John H.L. Hansen · Pinar Boyraz
Kazuya Takeda · Hüseyin Abut *Editors*

# Digital Signal Processing for In-Vehicle Systems and Safety

Springer

# Digital Signal Processing for In-Vehicle Systems and Safety

John H.L. Hansen • Pinar Boyraz
Kazuya Takeda • Hüseyin Abut
Editors

# Digital Signal Processing for In-Vehicle Systems and Safety

Springer

*Editors*
John H.L. Hansen
Center for Robust Speech Systems (CRSS)
Department of Electrical Engineering
The University of Texas at Dallas
Richardson, TX 75080-3021, USA
john.hansen@utdallas.edu

Pinar Boyraz
Mechatronics Education and Research
Center (MERC)
Istanbul Technical University
Gumussuyu, Istanbul 34437, Turkey
boyraz.pinar@googlemail.com

Kazuya Takeda
Department of Media Science
Nagoya University
Furo-cho, Chikusa-ku
Nagoya 464-8603, Japan
kazuya.takeda@nagoya-u.jp

Hüseyin Abut
ECE Department (Emeritus)
San Diego State University
San Diego, CA 92182, USA
*and*
Department of Electrical
and Electronics Engineering
Boğaziçi University
Bebek, Istanbul, Turkey
abut@anadolu.sdsu.edu

ISBN 978-1-4419-9606-0 e-ISBN 978-1-4419-9607-7
DOI 10.1007/978-1-4419-9607-7
Springer New York Dordrecht Heidelberg London

Library of Congress Control Number: 2011941657

# Foreword

The automobile has been in existence for more than 100 years and has evolved significantly during the past three decades. Early automobiles were designed to move the driver and passengers from point A to point B. Performance, comfort, style, and safety have all emerged to be core components in today's automotive market. The level of Digital Signal Processing contained within vehicles continues to grow significantly. This is due in part to the rapid growth of sensor technology within cars, as well as the motivation to make cars safer and more fuel efficient. In recent years, the concept of a "Smart Car" has also emerged, in part due to the advancements of artificial intelligence and computer design being introduced into vehicles. In the United States, the DARPA Grand Challenge [1] represents an effort which was started in 2004 to develop driverless vehicles. These vehicles would be fully automated and allow GPS, multi-sensor fusion, and advanced decision directed and feedback controls/artificial intelligence to navigate as an autonomous vehicle over long distances (i.e., 10–150 mi). To date, more than 195 teams from 36 US States and four countries have entered the competition. While the integration of advanced intelligence for smart cars is an admirable goal to achieve, it is clear that the majority of individuals who own their own car enjoy the freedom of driving, and are not likely to want to give up the ability to control their vehicle soon. As such, dealing with the introduction of new technologies into the vehicle environment represents a major challenge in the field.


As people spend more time in their vehicles, and commuting time to and from work continues to increase as urban populations grow, drivers are attempting to perform many more tasks than simply driving their vehicles. The introduction of wireless technology, digital audio/music players, mobile Internet access, advanced entertainment/multimedia systems, and smart navigation technologies into the car has placed increased cognitive demands on drivers. Yet, the typical driving test in countries continues to focus exclusively on the logistics of operating the vehicle itself and does not consider the management of these outside technologies as part of the driver assessment for a license. The United States [2] as well as many countries [3] have therefore moved to pass laws that restrict the use of cell phones and text messaging while operating a vehicle. The recent book *Traffic: Why We Drive The Way We Do*, by Tom Vanderbilt [4], offers a number of perspectives on society, culture, and government engagement on driving and drivers. Driver distractions in the car are many and have been documented by countless research studies. The average driver adjusts their radio 7.4 times/h of driving, turn their attention to infants 8.1 times/h, and are generally searching for something (e.g., sunglasses, coins, etc.) 10.8 times/h (p 78, [4]). The average driver looks away from the road 0.06 s every 3.4 s. Mobile devices with "intense displays" such the iPod require more concentration to search for songs, pausing, or skipping a song. While there are some differences of opinion, researchers have noted that any task that requires a driver to divert his/her attention (typically visual) away from the road for more than 1.5 s

(some believe this is up to 3.0 s) is viewed as a distraction. Irrespective of the exact time threshold, such a guideline is important as a general rule, but it should be clear that not all drivers are equally skilled, and even advanced/experienced drivers go through periods of fatigue, or can be unfamiliar with a new vehicle, which change their driving abilities and can impact safety, even if only during short periods.

The majority of driver-based vehicle research is based on (1) simulator studies, (2) field test studies, and (3) naturalistic studies. Simulator studies can allow research to consider high-risk conditions without putting test subjects at real risk; however they may not completely reflect how drivers would actually respond in such scenarios. Field test studies focus on vehicles which are outfitted with additional sensors/technology to record driver data on the streets; however, the driver clearly knows or feels this is a recording platform and may not be as familiar with the vehicle (i.e., their driving pattern would be different than if they were driving their own car). Finally, naturalistic driving represents the next major effort in the field, where miniaturization of data capture technology results in a recording platform that is seamlessly embedded into the driver's own vehicle, so it becomes a continuous window into everyday driving. The U.S. Transportation Research Board (TRB) is undergoing the SHRP2 [5] program, which would capture drivers from +1,500 vehicles continuously for 2 years. This corpus clearly would provide rich opportunities to integrate new digital signal processing advancements for built-in safety monitoring in the future.

In 2009, the fourth Biennial Workshop for In-Vehicle Systems and Safety took place in Dallas, Texas. This meeting served to bring together researchers from diverse research areas to consider advancements in digital signal processing within vehicles to improve safety and potentially contribute to reduce driver distraction. A total of 34 peer-reviewed conference papers were presented with researcher participation from universities, automotive and technology companies, as well as government research laboratories. The workshop included three keynote presentations from internationally recognized leaders in the field, including:

- Bruce Magladry – National Transportation Safety board (NTSB), USA
- Jon Hankey – Virginia Tech Transportation Institute (VTTI), USA
- Gerhard Schmidt – SVOX and Darmstadt University, Germany

As research involving advanced in-vehicle systems, smart-car technology, and intelligent transportation systems continues to advance, care must be taken to incorporate the skills and cognitive load and context of the driver, as well as the tasks and challenges faced when operating a vehicle in today's modern transportation network. Those authors of the fourth Biennial Workshop on DSP for In-Vehicle Systems and Safety, including those authors who have contributed chapters to this book, have dedicated themselves to advancements which will ultimately improve driver experience and safety.

The answers to questions relating to improved in-vehicle systems for safety are complex and require experts from diverse research fields. Significant advancements that leverage knowledge from such areas as human factors, control systems, signal processing, transportation engineering, artificial intelligence, machine learning, telecommunications/mobile technologies, and automotive design will ultimately lead to the next generation vehicles which continue to move drivers and their passengers from point A to point B, but also contribute to a safer driver experience as well as a more efficient transportation system.


Fourth Biennial Workshop on DSP for In-Vehicle Systems and Safety: (Closing Reception at the Fort Worth Rodeo for participants)

Richardson, TX, USA John H.L. Hansen

## References

[1] http://en.wikipedia.org/wiki/DARPA_Grand_Challenge
[2] http://www.iihs.org/laws/cellphonelaws.aspx
[3] http://www.cellular-news.com/car_bans/
[4] T. Vanderbilt, “Traffic: Why we drive the way we do,” A.A. Knopf, 2008.
[5] http://www.trb.org/StrategicHighwayResearchProgram2SHRP2/

# Preface

In June 2009, the *fourth Biennial Workshop on DSP (Digital Signal Processing) for In-Vehicle Systems and Safety* took place in Dallas, Texas, USA. The workshop was organized and hosted by the Center for Robust Speech Systems (CRSS): Speech/Speaker Modeling and UTDrive In-Vehicle Groups from The University of Texas at Dallas (UTDallas). This workshop follows a series of workshops organized first in 2003 (Nagoya, Japan), 2005 (Sesimbra, Portugal), and 2007 (Istanbul, Turkey). World-class experts from a diverse series of branches encompassing in-vehicle signal processing participated and shared cutting edge studies on road safety, in-vehicle technologies, and demos of state-of-art systems.

This workshop at UTDallas was broader in scope, with contributions from various realms such as: signal processing, control engineering, multi-modal audio-video processing, bio-mechanics, human factors, and transportation engineering which opened doors for fruitful discussions and information exchange in an exciting interdisciplinary area. The main focus areas were as follows:

- DSP technologies in adaptive automobiles
- Driver status monitoring and distraction/stress detection
- In-vehicle dialogue systems and human machine interfaces
- Challenges in video and audio processing for in-vehicle products
- Multi-sensor fusion for driver ID and robust driver monitoring
- Vehicle to vehicle, vehicle to infrastructure wireless technologies
- Human factors and cognitive science in enhancing safety
- Transportation engineering venues

The workshop included three keynote talks from internationally recognized leaders. Bruce Magladry, Director of the Office of Highway Safety, U.S. National Transportation Safety Board (NTSB), Washington, DC, USA, gave the opening keynote talk entitled “Highway Safety, Where We Are and Where We Are Going.” The second keynote address was from Jon Hankey from VTTI (Virginia Tech. Transportation Institute), Blacksburg, Virginia, USA, with a presentation entitled “Improving Transportation Safety – The Role of Naturalistic Driving Data”. VTTI

has been a leader in this domain with their well-known 100 car study on naturalistic driving. That work has been credited with motivating the SHRP2 Program from the U.S. National Transportation Board which will have +1,500 vehicles recorded continuously for 2 years. The third keynote speech was by Gerhard Schmidt, from SVOX and Darmstadt University, Germany, which focused on "Recent Trends for Improving Automotive Speech Communication Systems." A panel discussion was also organized which included Bruce Magladry (NTSB, USA), Jon Hankey (VTTI, USA), Gerhard Schmidt (SVOX, Darmstadt Univ., Germany), Hanseok Ko (Korea University, Korea), and Kazuya Takeda (Nagoya University, Japan) and offered opportunities for participants to engage in discussion on future directions for vehicle systems and safety. From this workshop, 21 papers and one additional contribution stemming from the next workshop were selected to make up the 22 chapters within this book. These chapters are grouped into four parts, each addressing key areas within in-vehicle digital signal processing:

Part A: Driver Behavior and Modeling Systems
Part B: In-Vehicle Interactive/Speech Systems
Part C: Vehicle Dynamics, Vision, Active Safety, and Corpora
Part D: Transportation, Vehicle Communications, and Next Generation Vehicle Systems

First, Part A consists of four chapters that consider driver behavior and modeling. The first chapter considers multi-modal signal processing based on speech, video, and CAN-Bus signals for robust stress detection in urban driving scenarios including multitasking, dialog system conversation, and medium-level cognitive tasks. The second chapter considers a classifier-based approach for assessing the emotion of a driver using speech into three emotions of anger, sadness, and happiness. The third chapter focuses on driving behavior signals stemming from vehicle control units such as gas/brake pedal use, steering wheel, etc. which differ among various driving tasks. Chapter 4 considers a hierarchical mode segmentation of observed driving behavioral data based on multiple levels of abstraction as applied to driving behavioral on an expressway.

The next nine chapters make up Part B of the textbook which focuses on In-Vehicle Interactive Systems. Chapter 5 considers advancements for in-car communication systems, and Chapter 6 focuses on wideband hands-free interaction in the car. Chapters 7 and 8 consider novel ways to start speech dialogs in cars, and cognitive dialog systems for dynamic environments respectively. Next, Chapter 9 considers corpus development for speech and vehicle noise for development of advancements on in-vehicle human–machine interactions. The next two chapters consider improved schemes for speech recognition in car environments, a necessary challenge in order to reduce distraction. The last two chapters in Part B focus on speech enhancement advancements for use in car environments. The next seven chapters make up Part C, which considers vehicle dynamics, vision, and active safety, and corpora. Chapter 14 develops advanced methods to generate reference views of traffic intersections. Chapter 15 considers computer vision systems for context-aware active safety and driver assistance. Chapter 16 investigates an

emerging area for integrating pedestrian detection and depth location with stereo cameras. Another safety area which is considered in Chapter 17 is driver overtaking judgments based on human perceptual for driver-assistant advancement. Driver emotional assessment based on multimedia using video/facial information is considered in Chapter 18. Meanwhile, Chapter 19 looks at modeling lane change trajectories using probabilistic strategies. An alternative scheme for active safety advancement is employing CAN-bus signal analysis based on stochastic models. The last portion of the book is Part D, which considers transportation, vehicle communications, and next generation vehicle systems. The highway driving infrastructure in many countries is expanding and become "smart", as well as vehicle to vehicle and vehicle to infrastructure communications. Chapter 21 considers multimedia streaming data over inter-vehicle communication networks. The final chapter offers some unique perspectives of next generation intelligent transportation infrastructures. MATISSE is a large-scale multi-agent system for simulating traffic safety scenarios.

As co-editors, we hope this book provides an up-to-date perspective of vehicle-based signal processing, with novel ideas for researchers with a comprehensive set of references for engineers and scientists in the field. We wish to thank all those who participated in the 2009 workshop. We wish to acknowledge support from a number of groups, in particular NEDO in Japan, funding agencies both from the USA, Turkey, Japan, and across all countries and from participating researchers who recognize the importance of research advancements for in-vehicle systems and safety. The co-editors would like to recognize and sincerely thank *Rosarita Lubag*, University of Texas at Dallas, who served as publications coordinator for the book, and assisted in layout, proof-reading, and ensuring quality control on each of the chapters. Her tireless efforts significantly contributed to a final version of the book which reflects the quality of the authors and presentations that took place in the fourth Biennial Workshop. We wish to express our continued appreciation to Springer Publishing for a smooth and efficient publication process for this book. Specifically, we would like to thank both Alex Greene and Ms. Allison Michael of Springer Publishing for their extensive efforts to work to enhance the structure and content of this book, as well as providing our community a high-quality and scholarly platform to stimulate public awareness, scientific research, and technology development in this field.

Richardson, TX, USA — John H.L. Hansen
Istanbul, Turkey — Pinar Boyraz
Nagoya, Japan — Kazuya Takeda
San Diego, CA, USA — Hüseyin Abut

# Contents

# Contributors

**Hüseyin Abut** San Diego State University, San Diego, USA
Boğaziçi University, Istanbul, Turkey

**Robert H. Baran** Korea University, Seoul, South Korea

**Jounghoon Beh** University of Maryland, College Park, USA

**Hynek Bořil** University of Texas at Dallas, Richardson, USA

**Pinar Boyraz** University of Texas at Dallas, Richardson, USA
Istanbul Technical University, Istanbul, Turkey

**Paolo Bucciol** French-Mexican Laboratory of Informatics and Automatic Control, Puebla, Mexico

**Carlos Busso** University of Texas at Dallas, Richardson, USA

**Ovidiu Daescu** University of Texas at Dallas, Richardson, USA

**Engin Erzin** Koç University, Istanbul, Turkey

**Tim Fingscheidt** Technische Universität Braunschweig, Braunschweig, Germany

**Balázs Fodor** Technische Universität Braunschweig, Braunschweig, Germany

**Hans W. Gierlich** HEAD Acoustics GmbH, Herzogenrath, Germany

**John H.L. Hansen** University of Texas at Dallas, Richardson, USA

**Soichiro Hayakawa** Mie University, Tsu, Japan

**Shinkichi Inagaki** Nagoya University, Nagoya, Japan

**Mark Hasegawa-Johnson** University of Illinois at Urbana-Champaign, Champaign, USA

**Jinesh Jain** University of Texas at Dallas, Richardson, USA

**Norhaslinda Kamaruddin** University Technology MARA, Shah Alam, Malaysia

**Jien Kato** Nagoya University, Nagoya, Japan

**Frank Kettler** HEAD Acoustics GmbH, Herzogenrath, Germany

**Kihyeon Kim** Korea University, Seoul, South Korea

**Lae-Hoon Kim** Qualcomm Incorporated, San Diego, USA

**Norihide Kitaoka** Nagoya University, Nagoya, Japan

**Wooil Kim** University of Texas at Dallas, Richardson, USA

**Tristan Kleinschmidt** Queensland University of Technology, Brisbane, Australia

**Hanseok Ko** Korea University, Seoul, South Korea

**Nitish Krishnamurthy** University of Texas at Dallas, Richardson, USA
Texas Instruments, Dallas, USA

**Rosarita Lubag** University of Texas at Dallas, Richardson, USA

**Juan Carlos De Martin** Politecnico di Torino, Torino, Italy

**Michael Mason** Queensland University of Technology, Brisbane, Australia

**Chiyomi Miyajima** Nagoya University, Nagoya, Japan

**Ato Nakano** Nagoya University, Nagoya, Japan

**Yoshihiro Nishiwaki** Nagoya University, Nagoya, Japan

**Hiroyuki Okuda** Nagoya University, Nagoya, Japan

**Emre Öztürk** Koç University, Istanbul, Turkey

**Matteo Petracca** Scuola Superiore Sant'Anna di Pisa, Pisa, Italy

**Zelam Purohit** University of Texas at Dallas, Richardson, USA

**Felix Putze** University of Karlsruhe, Karlsruhe, Germany

**Amardeep Sathyanarayana** University of Texas at Dallas, Richardson, USA

**David Scheler** Technische Universität Braunschweig, Braunschweig, Germany

**Gerhard Schmidt** University of Kiel, Kiel, Germany

**Tanja Schultz** University of Karlsruhe, Karlsruhe, Germany

**Antonio Servetti** Politecnico di Torino, Torino, Italy

**Sridha Sridharan** Queensland University of Technology, Brisbane, Australia

**Travis L. Steel** University of Texas at Dallas, Richardson, USA

**Tatsuya Suzuki** Nagoya University, Nagoya, Japan

**Kazuya Takeda** Nagoya University, Nagoya, Japan

**Anand Tharanathan** Texas Tech University, Lubbock, USA

**Anne Theiß** University of Kiel, Kiel, Germany

**Abdul Wahab** International Islamic University, Kuala Lumpur, Malaysia

**Yu Wang** Nagoya University, Nagoya, Japan

**Rym Zalila-Wenkstern** University of Texas at Dallas, Richarson, USA

**Jochen Withopf** University of Kiel, Kiel, Germany

**Arthur Wolf** SVOX Deutschland GmbH, Ulm, Germany

**Xuebo Yang** University of Texas at Dallas, Richardson, USA

**Jongsung Yoon** Korea University, Seoul, South Korea

# Part A
# Driver Behavior and Modeling Systems

# Chapter 1
# Towards Multimodal Driver's Stress Detection

**Hynek Bořil, Pinar Boyraz, and John H.L. Hansen**

**Abstract** Non-driving-related cognitive load and variations of emotional state may impact the drivers' capability to control a vehicle and introduce driving errors. The availability of stress detection in drivers would benefit the design of active safety systems and other intelligent in-vehicle interfaces. In this chapter, we propose initial steps towards multimodal driver stress (distraction) detection in urban driving scenarios involving multitasking, dialog system conversation, and medium-level cognitive tasks. The goal is to obtain a continuous operation-mode detection employing driver's speech and CAN-Bus signals, with a direct application for an intelligent human–vehicle interface which will adapt to the actual state of the driver. First, the impact of various driving scenarios on speech production features is analyzed, followed by a design of a speech-based stress detector. In the driver-/maneuver-independent open test set task, the system reaches 88.2% accuracy in neutral/stress classification. Second, distraction detection exploiting CAN-Bus signals is introduced and evaluated in a driver-/maneuver-dependent closed test set task, reaching 98% and 84% distraction detection accuracy in lane keeping segments and curve negotiation segments, respectively. Performance of the autonomous classifiers suggests that future fusion of speech and CAN-Bus signal domains will yield an overall robust stress assessment framework.

**Keywords** Active safety • CAN-bus signal processing • Distraction detection • Stress

H. Bořil (✉) • P. Boyraz • J.H.L. Hansen
Center for Robust Speech Systems, Erik Jonsson School of Engineering
& Computer Science, University of Texas at Dallas, Richardson, TX, USA
e-mail: hynek@utdallas.edu; boyraz.pinar@googlemail.com; john.hansen@utdallas.edu

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_1, 

## 1.1 Introduction

Recent advancements in the electronic industry have made access to information and entertainment easier than ever before. While undoubtedly benefiting many areas of our daily lives, there are situations where the presence of electronic gadgets has the opposite effect. In a current study, the Virginia Tech Transportation Institute (VTTI) reports that dialing on a handheld device while driving increases the risk of an accident by a factor of 3, and communicating via hands-free set increases the risk by one third. This suggests that performing secondary cognitive tasks while driving may severely impact driving performance. Besides cognitive load, drivers' emotions have also been shown to adversely affect driving performance, e.g., by the means of larger deviations of lane offset and steering wheel angle, and shorter lane crossing times in anger and excitation situations – signs of reduced lane control capability. Availability of an automated system assessing stress in drivers would benefit the design of active safety systems and other intelligent in-vehicle interfaces, making them capable of adapting to the driver's current state (e.g., by decreasing the frequency of navigation prompts when detecting high-cognitive-load situations).

A number of studies have analyzed the impact of emotions [1–4] and stress (including cognitive load) on speech parameters [5–9]. However, relatively limited attention has been paid to the impact of emotion, stress, or distraction on the speech of car drivers [10, 11]. In [10], speech from subjects driving a simulator was categorized into seven emotional states, using a classifier trained on a corpus of emotional speech from professional actors. The emotional states in drivers were evoked during conversation with a dialog system. Also, Jones and Jonsson [11] used speech data collected in a driving simulator and categorized them into four stress classes. Different stress levels were induced by requesting the driver to maintain a certain speed (60 mph or 120 mph) and solve simple math tasks prompted at slow and fast rates by a synthesizer over the phone. The obtained classification performance in the driver-independent task was relatively low (~51%). We note that both studies utilize simulated driving scenarios, and in the case of [10] also employ simulated emotions from actors to establish classification categories. Acted emotions represent exaggerated traits that are effective in convincing listeners of the individual speaker state, but are not accurate representatives of natural emotions. Using driving simulators also introduces differences from real driving scenarios since there is less or no consequence for making errors in the primary task. In addition, a significant drawback of approaches utilizing only speech is that the emotion or stress assessment can be conducted only in time intervals when the driver is engaged in conversation.

To address these issues, the present study is conducted on the database UTDrive [12] collected in real driving conditions and aims at utilizing both speech and CAN-Bus signals in the stress assessment. The term stress here represents the modality of the driver's speech production or driving behavior conducted under cognitive load. In the course of this chapter, the terms stress and distraction are used interchangeably, where the primary task is driving.

The remainder of the chapter is organized as follows: First, the data acquisition procedure and distraction/stress scenarios in UTDrive corpus are described.

Second, an analysis of speech production parameters in three cognitive load scenarios is conducted, and a speech-based stress classifier is introduced. Third, a classifier operating on CAN-Bus signals is proposed and evaluated.

## 1.2 UTDrive Corpus, Data Subsets, and Transcription Protocols

The data collection vehicle is a Toyota RAV4 equipped with the following sensors (illustrated in Fig. 1.1):

- Two CCD cameras for monitoring the driver and the road scene through front windshield
- Microphone array (five mics) to record driver's speech as well as noise conditions in the vehicle
- A close-talk microphone to obtain driver's speech with reduced noise content
- Optical distance sensor to obtain headway distance between equipped vehicle and other vehicles in traffic
- GPS for location tracking
- CAN-Bus OBD II port for collecting vehicle dynamics: vehicle speed, steering wheel angle, gas and brake inputs from driver
- Gas/brake pedal pressure sensors to collect information concerning pressure patterns in car-following and braking behavior

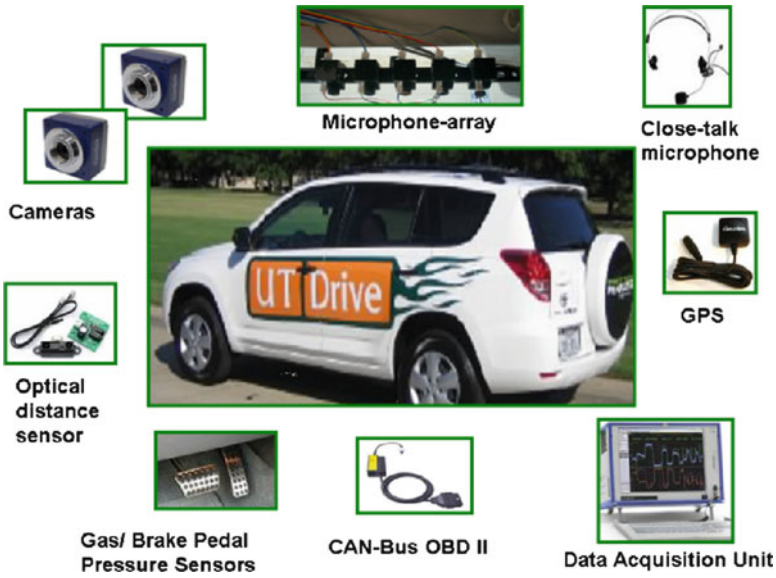



**Fig. 1.1** Instrumented data collection vehicle: UTDrive


**Fig. 1.2** Data collection: residential (left) and business (right) routes segmented according to assigned tasks

The UTDrive corpus includes data from the above-mentioned sensor channels (13 separate data streams: two video, six audio, one GPS, one optical distance, one CAN-Bus, two pressure sensors on gas/brake). The corpus is organized to have a balance in gender (37 males, 40 females), age (18–65), and different experience level (novice–expert) in driving. In order to examine the effect of distraction and secondary common tasks on these driver groups, a close-to-naturalistic data collection protocol is used.

The routes taken during data collection are given in Fig. 1.2, comprising a mixture of secondary, service, and main roads in residential (left-hand side map) and business (right-hand side map) districts in Richardson, TX. Each driver participating in the study is required to drive these two routes at least twice in each session to obtain a baseline and a distracted version of the same route. A session includes a mixture of several secondary tasks as listed in Table 1.1, taking place in road segments depicted in Fig. 1.2. According to this protocol, a participant performs 12 runs of data, with six being baselines for that day and that route, the other half featuring several distraction conditions. Each session is separated at least by 2 weeks in order to prevent driver complacency with the route and vehicle. Almost 60% of the data in the corpus have a full session profile from drivers. The remaining part contains incomplete sessions and data portions due to the consent of the participant not to continue data collection or several sensor failures. The secondary driver tasks are low to medium level of cognitive load while driving.

In this study, cell phone dialog parts including interaction speech with automated portals Tell-Me (information system) and American Airlines (reservation system) are utilized and analyzed using driver's speech and CAN-Bus signals. The cell phone conversation takes place in route segment two which includes lane keeping and lane curvature negotiation tasks while the driver is engaged in cell phone dialog. In order to segment the data in terms of driving event and task timelines and find overlapping portions, two different transcription protocols are applied. First, using the audio and video, a task transcription is performed, having 13 labels to annotate the segments of the data in terms of where the driver and passenger talk and where other types of distractions occur. The second is called "event transcription" and performed

**Table 1.1** UTDrive data collection protocol

| | | Secondary tasks | | |
|---|---|---|---|---|
| Part | | A | B | C |
| Route1 | 1 | Lane changing | Common tasks (radio, AC etc.) | Sign reading |
| | 2 | Cell phone dialog | Cell phone dialog | Conversation |
| | 3 | Common tasks | Sign reading | Spontaneous |
| | 4 | Conversation | Spontaneous | Cell phone dialog |
| Route2 | 1 | Sign reading | Lane changing | Common tasks (radio, AC etc.) |
| | 2 | Cell phone dialog | Cell phone dialog | Conversation |
| | 3 | Common tasks (radio, AC etc.) | Sign reading | Lane changing |
| | 4 | Spontaneous | Conversation | Sign reading |

| Session | Route | Task |
|---|---|---|
| 1 | 1 | Just drive |
| | 1 | Secondary tasks A |
| | 2 | Secondary tasks A |
| | 2 | Just drive |
| 2 | 1 | Just drive |
| | 1 | Just drive |
| | 2 | Secondary tasks B |
| | 2 | Secondary tasks C |
| 3 | 2 | Secondary tasks C |
| | 1 | Secondary tasks C |
| | 2 | Just drive |
| | 2 | Just drive |

to have six labels to denote different maneuvers of the driver. A color-coded driving timeline is developed to observe aligned task and event transcriptions to obtain more insight into the data as well as to see the overlapping sections between tasks and events. A detailed explanation is given in [13] for transcription labels and color-coded driving timeline.

It should be noted that cell phone dialog includes different types of distractions: manual (dialing and holding), cognitive (interaction and processing), and auditory (listening). Therefore, the segment of the road containing the cell phone dialog can be considered as the highest possibility of observing high levels of distraction and divided attention. Although the cell phone in the car interfaces via a bluetooth device and the manual tasks from the driver minimized, the initial dialing might cause momentary distraction.

## 1.3 Stress Detection Using Speech Signal

This section focuses on the stress assessment from the driver's speech. First, it should be noted that the real level of stress in the driver caused by the cognitive load is not known. To define stress levels in the speech segments, we apply a *cause-type*

annotation of the data, as presented in [10]. Here, we hypothesize that a certain task the driver is asked to perform has a potential to cause a deviation of the driver's speech production from neutral, and hence, represents a stress condition.

In particular, we expect that the interaction with the automated call centers *Tell-Me* and *American Airlines* (*AA*) puts an extensive cognitive load on the driver compared to the driver's casual conversations with the passenger. This is expected partly due to the high demands of the automated call center on clear articulation, explicit formulation of the requests within a limited vocabulary of the system, and frequent requests for reentering the query due to the automatic speech recognition failure. For this reason, we denote spontaneous conversations with the passenger as *neutral* speech and calls to *Tell-Me* and *AA* as *stressed* speech. It is noted that even spontaneous communication with the passenger represents a certain level of cognitive load on the driver compared to silent segments and that due to the variable level of car noise, the driver is likely to exhibit various levels of Lombard effect [5, 14, 15].

In order to verify whether there are any measurable differences in the "neutral" and "stressed" portions of speech data and, hence, whether our hypothesis concerning the presence of stress in the higher-cognitive-load scenarios is reasonable, we first analyze the distributions of speech production parameters and compare them across hypothesized stress classes. Subsequently, we train separate Gaussian Mixture Models (GMMs) for neutral and stressed classes and evaluate the class discriminability using maximum likelihood classification. The gender-independent training and testing of the neutral/stress classifier is performed on disjunctive data sets from different speakers in order to evaluate the generalizing properties of the classification system.

### *1.3.1 Speech Production Analysis*

Sessions from 15 drivers (seven females, eight males) are used in the speech analysis and stress classification experiments. An inspection of the close-talk microphone channel revealed a strong presence of "electric" noise completely masking the driver's speech. For this reason, a middle microphone channel from the microphone array is used instead.

The following speech signal parameters are analyzed on the data down-sampled from 25 kHz to 16 kHz: signal-to-noise ratio (SNR), mean noise and speech power spectrum, fundamental frequency, first four formant frequencies and bandwidths, and spectral slope of voiced speech segments. SNR was estimated from (1) segmental SNR estimator [16], (2) average *noise* power spectrum, and (3) average *noisy speech* power spectrum. The SNR distribution obtained from the first method is shown in Fig. 1.3; the mean SNR reaches −2.7 dB, with the standard deviation of 4.4 dB. Note that the SNR values in the distribution are quite low due to the distant microphone placement from the driver.

To verify the estimate from the segmental detector, in the next step, SNR is estimated directly from the average *noise* power spectrum ($N$) extracted from all

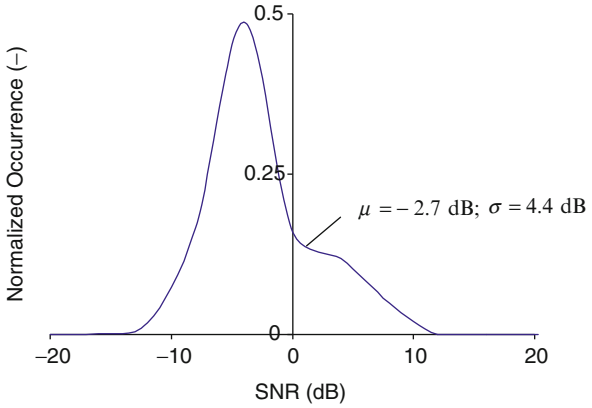



**Fig. 1.3** Distribution of SNR across all sessions

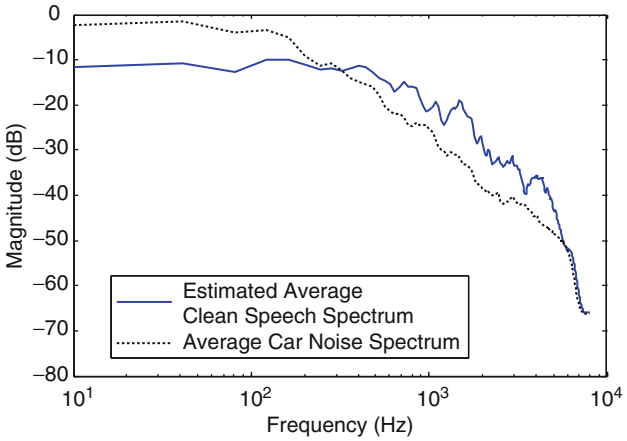



**Fig. 1.4** Average amplitude spectrum of noise and clean speech – averaged across all sessions

nonspeech segments, and the average *noisy speech* power spectrum (*SN*) is estimated from all passenger conversation, Tell-Me and AA segments:

$$\widehat{SNR} = 10 \cdot \log \sum_{k} \frac{SN_k - N_k}{N_k}, \tag{1.1}$$

where $k$ denotes the power spectrum frequency bin index. The SNR estimate obtained from the power spectra reaches −3.2 dB, confirming a reasonable accuracy of the segmental SNR estimation. The average power spectrum of noisy segments without speech and of clean speech estimated by subtracting N from SN is shown in Fig. 1.4. It can be seen that the car noise spectrum dominates over speech at low frequencies while speech becomes dominant, in spite of the low SNR, at frequencies higher than 300 Hz.

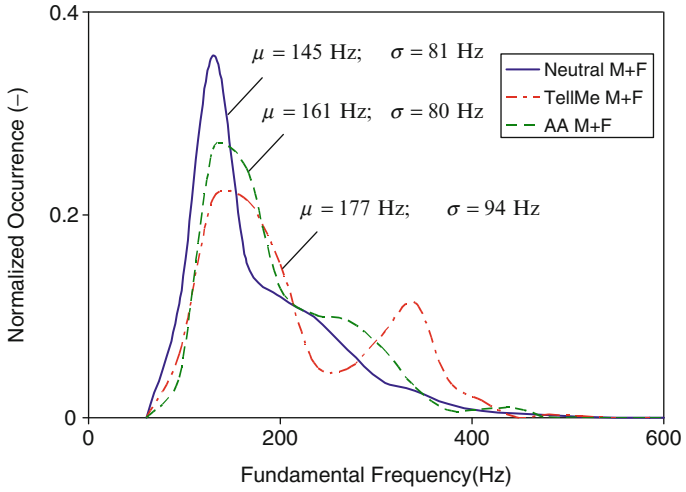



**Fig. 1.5** Distribution of fundamental frequency in neutral, Tell-Me, and AA sessions

**Table 1.2** Formant center frequencies and bandwidths (in parentheses)

| | | Formants and bandwidths (Hz) | | | |
|---|---|---|---|---|---|
| Gender | Scenario | F1 | F2 | F3 | F4 |
| F | Neutral | 555 | 1,625 | 2,865 | 4,012 |
| | | (219) | (247) | (312) | (327) |
| | Tell-Me | 703 | 1,612 | 2,836 | 3,855 |
| | | (308) | (276) | (375) | (346) |
| | AA | 710 | 1,667 | 2,935 | 4,008 |
| | | (244) | (243) | (325) | (329) |
| M | Neutral | 450 | 1,495 | 2,530 | 3,763 |
| | | (188) | (209) | (342) | (343) |
| | Tell-Me | 472 | 1,498 | 2,525 | 3,648 |
| | | (205) | (214) | (341) | (302) |
| | AA | 503 | 1,526 | 2,656 | 3,654 |
| | | (188) | (215) | (330) | (369) |

In the next step, speech production parameters are analyzed. Distributions of fundamental frequency in passenger conversations (denoted *Neutral*), and Tell-Me and AA conversations are depicted in Fig. 1.5, where $M + F$ stands for mixed-gender data sets. Both Tell-Me and AA samples display a consistent increase in mean fundamental frequency (177 Hz and 161 Hz) compared to neutral (145 Hz).

Mean center frequencies and bandwidths of the first four formants were extracted from voiced speech segments using WaveSurfer [17]. They are compared for neutral, Tell-Me, and AA conversations in Table 1.2. The voiced segments were identified based on the output of the pitch tracking algorithm implemented in [17] (RAPT [18]).

Mean center frequencies and standard deviations of F1 are displayed in Fig. 1.6. A consistent increase in F1 can be observed for Tell-Me and AA data. In AA,

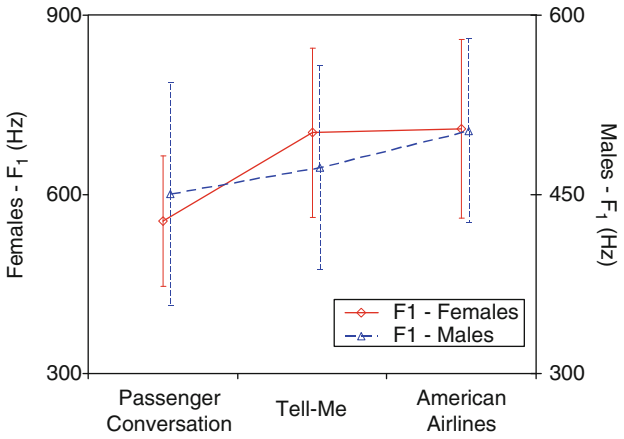



**Fig. 1.6** Mean $F_1$ center frequency in neutral, Tell-Me, and AA sessions (accompanied by standard deviations in error plots)

also F2 and F3 increase in both genders while remaining relatively steady in Tell-Me. Note that F1 and F2 increases have been previously reported for stressed speech, including angry, loud, and Lombard speech modes [5, 14, 15]. Finally, spectral slopes of the voiced speech segments were extracted by fitting a straight line to the short-term power spectra in the log amplitude/log frequency plane by means of linear regression [14]. The mean spectral slope reaches values around −10.4 dB/Oct, displaying no significant differences across stress classes. Note that the average slope is somewhat higher than that reported in the literature for clean neutral speech, presumably due to the strong presence of background car noise, which introduces additional spectral tilt.

The analysis conducted in this section revealed differences in fundamental frequency, $F_1$, and $F_2$ center frequencies between the selected neutral and stressed classes, confirming that the initial hypothesis about the presence of stress in Tell-Me and AA segments due to increased cognitive load is valid.

### *1.3.2 Automatic Classification of Stress*

In this section, speech-based neutral/stress classification is proposed and evaluated. For the purposes of classifier training and testing, the data from 15 drivers were split into a training set comprising of speech samples from two male and two female drivers, and test set comprising six male drivers and five female drivers.

Gaussian Mixture Models (GMMs) are chosen to represent probability density functions (PDFs) of the neutral and stressed classes. The probability of observation vector $\mathbf{o}_t$ being generated by the $j$th GMM is calculated as

$$b_j(\mathbf{o}_t) = \sum_{m=1}^{M} \frac{c_{jm}}{\sqrt{(2\pi)^n |\Sigma_{jm}|}} \cdot \mathrm{e}^{-\frac{1}{2}(\mathbf{o}_t - \boldsymbol{\mu}_{jm})^T \boldsymbol{\Sigma}_{jm}^{-1} (\mathbf{o}_t - \boldsymbol{\mu}_{jm})}, \tag{1.2}$$

where $m$ is the index of the Gaussian mixture component, $M$ is the total number of mixtures, $c_{jm}$ is the mixture weight such that

$$\sum_{m=1}^{M} c_{jm} = 1, \tag{1.3}$$

$n$ is the dimension of $\mathbf{o}_t$, $\Sigma_{jm}$ is the mixture covariance matrix, and $\boldsymbol{\mu}_{jm}$ is the mixture mean vector. The GMM representing neutral speech was trained on the passenger conversations and the stressed speech GMM on joint Tell-Me and AA conversations from the training set. In the neutral/stress classification task, the winning model is selected using a maximum likelihood criterion:

$$j_{win} = \begin{cases} 1, & \sum_{t=1}^{T} \log(b_1(\mathbf{o}_t)) - \sum_{t=1}^{T} \log(b_2(\mathbf{o}_t)) \geq Th, \\ 2, & \sum_{t=1}^{T} \log(b_1(\mathbf{o}_t)) - \sum_{t=1}^{T} \log(b_2(\mathbf{o}_t)) < Th, \end{cases} \tag{1.4}$$

where $t$ is the time frame index, $T$ is the total number of frames in the classified utterance, and $Th$ is the decision threshold.

In our experiments, the frame length was set to 25 ms, skip rate 10 ms, and the decision threshold to a fixed value $Th = 0$. Depending on the feature extraction scheme, the GMMs comprise 32–64 mixtures, and only diagonals are calculated in the covariance matrices. Unless otherwise specified, $c_0$–$c_{12}$ form the static observation feature vector. In all evaluation setups, delta and acceleration coefficients are extracted from the static coefficients and complete the feature vector. A variety of features, including Mel Frequency Cepstral Coefficients (MFCC), are considered.

In the UTDrive sessions, the amount of *neutral* spontaneous conversation data considerably exceeds the number of Tell-Me and AA samples. In this case, possible misclassification of small amount of stressed samples would have little effect on the overall classification accuracy, while classifying correctly only neutral data would assure high overall accuracy. To eliminate the impact of different sizes of the neutral and stressed sets, and to allow for accuracy-based selection of the optimal front-end for both AA and Tell-Me conversation scenarios, the overall classification accuracy is determined as

$$Acc = \frac{2Acc_{N-N} + Acc_{TellMe-S} + Acc_{AA-S}}{4} \ (\%), \tag{1.5}$$

where $Acc_{N-N}$ is the accuracy of neutral samples being classified as neutral, $Acc_{TellMe-S}$ is the accuracy of Tell-Me samples being classified as stressed, and $Acc_{AA-S}$ is the accuracy of AA samples being classified as stressed.

Efficiency of several feature extraction front-ends was evaluated in the neutral/stress classification task. In particular, Mel Frequency Cepstral Coefficients (MFCC [19]), Perceptual Linear Prediction (PLP) cepstral coefficients [20], Expolog cepstra [21],

**Table 1.3** Classification performance; normalized accuracy (%)

| | Front-end | | | | | | |
|---|---|---|---|---|---|---|---|
| NS | MFCC | PLP | Expolog LPC | Expolog DCT | 20Bands LPC | 20Bands DCT | 20Bands DCT11 |
| None | 83.7 | 83.1 | 81.4 | 81.9 | 84.2 | 84.1 | 83.6 |
| FWSS | 85.6 | 85.1 | 86.2 | 85.4 | 83.5 | 87.6 | 88.2 |

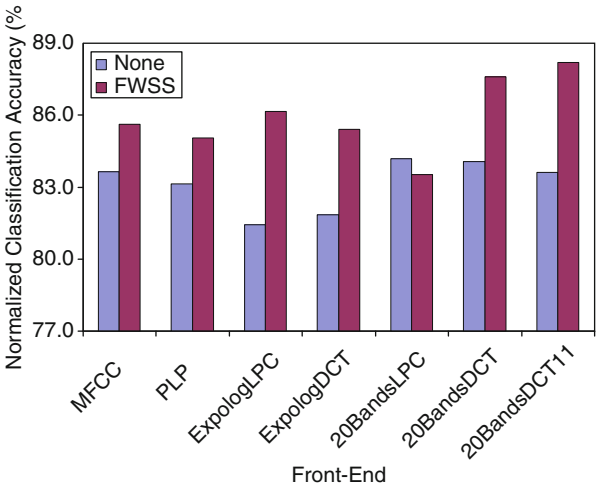



**Fig. 1.7** Front-end's classification performance

and cepstra extracted from a uniform filterbank of 20 non-overlapping rectangular filters distributed on a linear frequency scale (20Bands) [15] were compared. MFCC represent a common baseline front-end in speech/speaker recognition, and PLP has been shown by numerous studies to provide comparable or better performance to MFCC in various speech-related tasks [14].

Expolog is an outcome of studies on accent classification and stressed speech recognition, and features based on 20Bands filterbank have shown superior properties in noisy neutral and Lombard speech recognition [15].

In this study, Expolog and 20Bands filterbanks were used either as a replacement for the triangular Mel filterbank in MFCC, yielding front-ends denoted Expolog DCT and 20Bands DCT, or as a replacement for PLP trapezoid Bark filterbank, yielding setups denoted Expolog LPC and 20Bands LPC. In order to reduce the impact of strong background noise on classification, Full Wave Spectral Subtraction (FWSS) utilizing Burg's cepstral-based voice activity detector [14] was incorporated in the feature extraction. The classification results are summarized in Table 1.3 and Fig. 1.7. The first row of results in Table 1.3 represents the performance of a classifier without noise subtraction (NS), denoted "none."

It can be seen that in the majority of cases, FWSS considerably improves performance. Among front-ends employing 13 static coefficients and their first-and

second-order time derivatives, 20Bands DCT with FWSS provided the highest classification accuracy (87.6%). In addition, it was observed that decreasing the size of the static cepstral coefficients vector from 13 to 11 ($c_0$–$c_{10}$), denoted 20Bands DCT11, provides further accuracy increase to 88.2%. In this setup, the individual accuracies were $Acc_{N-N} = 91.4\%$, $Acc_{TellMe-S} = 70.0\%$, and $Acc_{AA-S} = 100.0\%$. Note that the accuracy and intraclass confusability can be further balanced by adjusting *Th* in Eq.1.4. However, for that, the availability of additional development data is required.

## 1.4 Distraction/Stress Detection Using CAN-Bus Signals

In this part of the study, we develop a distraction detection module based on a subset of CAN-Bus signals (mainly steering wheel angle and speed) using driver performance metrics, signal processing tools, and statistics. A generic distraction detection system without having the maneuver/context information and driver baselines for that particular maneuver is very difficult to design simply because the generic baseline for the nominal values of metrics/features has a wide range of variation due to driver characteristics and route/maneuver/context dependency.

CAN-Bus signals can reveal the distraction level of the driver when the variability due to maneuvers and driver characteristics are eliminated or dealt with so that they do not cause false alarms. Therefore, a methodology using a baseline for each individual driver and particular maneuver is proposed. A general flow diagram of the methodology is given in Fig. 1.8. The variation in the signals due to the maneuver/particular road segment is eliminated here by maneuver classification.

After the feature extraction process, distraction detection is performed by taking the driver's baseline for a given maneuver obtained from the same route segment (marked by two in Fig. 1.2) as when the conditions were neutral. Since UTDrive corpus includes multiple sessions collected from the same route and same driver under different conditions, hence, baselines can easily be obtained. The algorithm flow for distraction detection is shown in Fig. 1.9.

A normalized comparison ratio ($\alpha$) is calculated for each element in the feature vector. The comparison ratio is used in multiple interval thresholds. Each threshold

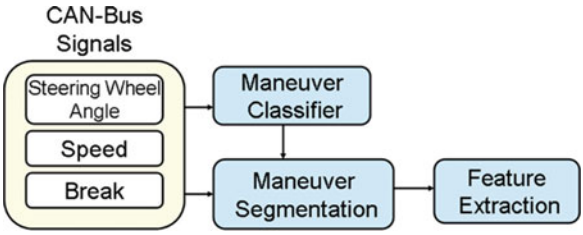



**Fig. 1.8** Flow diagram of general methodology used for CAN-Bus-based analysis

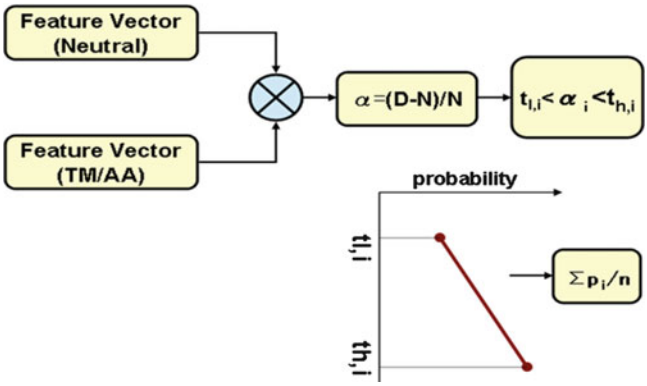



**Fig. 1.9** Distraction detection algorithm flow based on features extracted from CAN-Bus signals

interval is assigned to a probability. For example, if the ratio is between 0.1 and 1, the probability of distraction is 0.7, and if the ratio is larger than 20, it is 1. This assignment approach allows for a probabilistic assessment of the distraction or can give an idea of the distraction level.

Comparison values larger than 0.1 in magnitude are considered to indicate a significant distraction. If the comparison value magnitude is below 0.1, the session is assumed to be close enough to baseline to be considered neutral. As the comparison ratio increases, the probability of being distracted increases, with the highest value being 1 as shown in Fig. 1.9. At the end of this probability mapping, the probabilities are summed along the feature vector (now comprised by comparison ratios) and normalized by dividing the resultant likelihood value in the feature vector dimension. The next section explains the feature extraction process and motivation behind the feature vector elements selected.

### *1.4.1 CAN-Bus-Based Features*

The features are selected based on their relevance to distraction and definition of the maneuver. Using the color-coded driving timeline plots, it was observed that the route segment two contains lane keeping and curve negotiation tasks in terms of driving. For the lane keeping, several driver performance metrics are suggested in the literature mostly using steering wheel angle (SWA) to calculate a metric indicating the fluctuations or microcorrections in SWA input. Among these metrics, a widely accepted method is the sample entropy [22] and standard deviation. If available, the lane deviation measurements also give away if the driver is fully attentive and in control. The reversal rate of steering wheel is also considered to be a reliable metric to measure driver performance in a lane keeping task. Boer [23] recently updated his previous work and suggested some adjustments, taking high-frequency terms

**Table 1.4** Feature vector and definitions

| Notation | Definition |
|---|---|
| WDE_SWA | Wavelet decomposition detail signal energy for SWA |
| WDE_speed | Wavelet decomposition detail signal energy for speed |
| SampEnt_SWA | Sample entropy of SWA |
| SampEnt_speed | Sample entropy of SWA |
| STD_SWA | Standard deviation of SWA |
| STD_speed | Standard deviation of SWA |
| STD_SWAR | Standard deviation of SWA rate |

into account. It was also pointed out in a thorough analysis [24] that the speed interval for which the SWA-dependent metric is being calculated is important since the lower speeds require more SWA inputs to achieve the same amount of lateral movement of the car compared to a higher speed. For the curve negotiation, a constant input of an angle required using the visual input of the road curvature.

The novice or distracted driver may have fluctuating inputs in the SWA, and the general trend is that the speed should be reduced while taking the curves to balance the centrifugal force. Although different in nature, lane keeping and curve negotiation can be seen as regulatory control tasks from the driver's point of view. Therefore, we selected a seven-dimensional feature vector using available information and observations about driver performance/behavior including: energies of high-frequency components wavelet decomposition (WD), sample entropy, standard deviation, and standard deviation of rate of change (R-STD). All features are extracted for SWA, and speed channels except R-STD are only applied to SWA. The time window length is taken as equal to the maneuver length, and the effect of the signal length is eliminated in the calculation of features. The entries of the feature vector are listed with their definitions in Table 1.4.

For the wavelet decomposition, Daubechies [25] wavelet kernel with fourth order is used, and detail signal is taken at the sixth level. Daubechies wavelet is chosen since it can approximate to signals with spikes and discontinuous attributes well. The level and order is adjusted to be able to extract the high-frequency content in the signal which is in the limitation of human control; the higher details are ignored since they might be caused by other disturbances in the measurement rather than driver. Scaling functions (a), wavelet function coefficients (b), scaling function (c), and wavelet function (d) for DB4 are given in equation group (1.6):

$$h_0 = \frac{1+\sqrt{3}}{4\sqrt{2}}, h_1 = \frac{3+\sqrt{3}}{4\sqrt{2}}, h_2 = \frac{3-\sqrt{3}}{4\sqrt{2}}, h_3 = \frac{1-\sqrt{3}}{4\sqrt{2}}, \tag{1.6a}$$

$$g_0 = h_3, g_1 = -h_2, g_2 = h_1, g_3 = -h_0, \tag{1.6b}$$

$$a_i = h_0 s_{2i} + h_1 s_{2i+1} + h_2 s_{2i+2} + h_3 s_{2i+3}, \tag{1.6c}$$

$$c_i = g_0 s_{2i} + g_1 s_{2i+1} + g_2 s_{2i+2} + g_3 s_{2i+3}. \tag{1.6d}$$

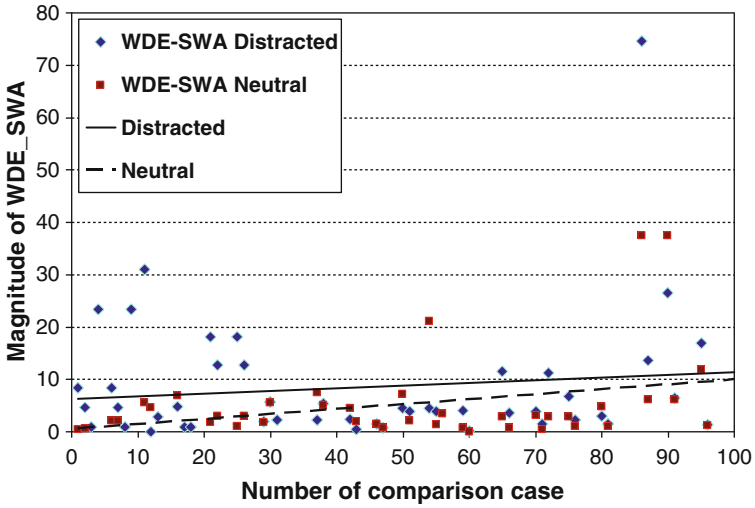



**Fig. 1.10** Wavelet decomposition details signal energy for SWA calculated for 96 comparison cases of lane keeping

Sample entropy (SampEnt), which is used as a measure to quantify regularity and complexity of the signal, is a perfect match measuring the regularity of SWA signal. It is known that the measures based on entropy have long been employed in biosignal processing such as EEG, ECG, and EMG to measure regularity and detect abnormality. The method to calculate the sample entropy follows the work described in [26]. The standard deviation is calculated in a canonical form with statistics.

### *1.4.2 Distraction Detection Performance*

Using the algorithm flow depicted in Fig. 1.9 and feature vectors explained in Table 1.4, 96 comparison cases for lane keeping and 113 cases for curve negotiation were examined using 14 drivers' (20 sessions, seven female and seven male drivers) data. As an insight, WDE_SWA feature member is given for lane keeping maneuvers in Fig. 1.10. It can be easily seen that the distracted sessions are generally greater than the baseline for this metric. The accuracy of the distraction detection is given in Table 1.5 using seven-dimensional feature vector (LKS) and using four-dimensional feature vector subset containing only SWA-related features (LKC) with threshold values of 0.2, 0.1, and 0 for the final classification result.

From Table 1.5, it can be seen that if any probability value higher than zero is taken into account, the distraction can be detected with 98% accuracy using lane keeping segments (LKS) and by 84% accuracy using curve negotiation segments (LKC) during Tell-Me/AA conversations.

**Table 1.5** Accuracy of distraction detection

| | | Threshold | | | | | |
|---|---|---|---|---|---|---|---|
| Maneuver | Measure | 0.2 | | 0.1 | | 0 (Binary) | |
| LKS | Count | 72/96 | 62/96 | 84/96 | 76/96 | 95/96 | 76/96 |
| | Acc (%) | 75 | 64 | 87 | 79 | 98 | 79 |
| LKC | Count | 65/113 | 64/113 | 82/113 | 79/113 | 95/113 | 79/113 |
| | Acc (%) | 57 | 56 | 72 | 69 | 84 | 69 |

The system offers a low-cost, driver-dependent, and reliable distraction detection submodule. Future work will focus on generic distraction detection using sums within the same feature space.

## 1.5 Conclusions

In this study, the impact of cognitive load on drivers was analyzed using the UTDrive database that comprises real-world driving recordings. In particular, driver's speech signal and CAN-Bus signals were studied and subsequently utilized in the design of autonomous speech and CAN-Bus domain neutral/stress (distraction) classifiers. The speech-based neutral/stress classification reached an accuracy of 88.2% in the driver-/maneuver-independent open test set task. The distraction detector exploiting CAN-Bus signals was evaluated in a driver-/maneuver-dependent closed test set task, providing 98% and 84% distraction detection accuracy in lane keeping segments and curve negotiation segments, respectively. The results suggest that future fusion of speech and CAN-Bus-based classifiers could yield a robust continuous stress (distraction) assessment framework.

## References

1. Neiberg D, Elenius K, Karlsson I, Laskowski K (2006) Emotion recognition in spontaneous speech using GMMs. In: Proceedings of ICSLP'06, Pittsburgh, pp 809–812
2. Lee CM, Narayanan SS (2005) Toward detecting emotions in spoken dialogs. IEEE Trans Speech Audio Proc 13:293–303
3. Ijima Y, Tachibana M, Nose T, Kobayashi T (2009) Emotional speech recognition based on style estimation and adaptation with multiple-regression HMM. In: Proceedings of IEEE ICASSP'09, Taipei, pp 4157–4160
4. Callejas Z, Lopez-Cozar R (2008) Influence of contextual information in emotion annotation for spoken dialogue systems. Speech Comm 50(5):416–433
5. Hansen JHL (1996) Analysis and compensation of speech under stress and noise for environmental robustness in speech recognition. Speech Comm 20, 151–173
6. Cummings K, Clements M (1990) Analysis of glottal waveforms across stress styles. In: Proceedings of IEEE ICASSP'90, Albuquerque, vol 1, pp 369–372

7. Bou-Ghazale SE, Hansen J (1998) HMM-based stressed speech modeling with application to improved synthesis and recognition of isolated speech under stress. IEEE Trans Speech Audio Proc 6:201–216
8. Sarikaya R, Gowdy JN, (1998) Subband based classification of speech under stress. In: Proceedings of ICASSP'98, pp 569–572
9. Zhou G, Hansen J, Kaiser J (1998) Linear and nonlinear speech feature analysis for stress classification. In: Proceedings of ICSLP'98, Sydney, vol 3, pp 883–886
10. Fernandez Raul, Picard RW (2003) Modeling drivers' speech under stress. Speech Comm 40 (1–2):145–159
11. Jones CM, Jonsson I (2005) Automatic recognition of affective cues in the speech of car drivers to allow appropriate responses. In: Proceedings of 17th Australian conference on computer–human interaction, Canberra, pp 1–10
12. Angkititrakul P, Petracca M, Sathyanarayana A, Hansen JHL (2007) UTDrive: driver behavior and speech interactive systems for in-vehicle environments. IEEE intelligent vehicles symposium, Istanbul, pp 566–569
13. Boyraz P, Sathyanarayana A, Hansen JHL (June 2009) CAN-bus signal modeling using stochastic methods and structural pattern recognition in time series for active safety. 4th biennial workshop on DSP for in-vehicle systems and safety, Dallas
14. Bořil H (2008) Robust speech recognition: analysis and equalization of Lombard effect in Czech corpora. Ph.D. dissertation, Czech Technical University in Prague http://www.utdallas.edu/~hynek
15. Bořil H, Hansen, JHL (2009) Unsupervised equalization of Lombard effect for speech recognition in noisy adverse environment. In: Proceedings of IEEE ICASSP'09, Taipei, pp 3937–3940
16. Vondrášek M, Pollák P (2005) Methods for speech SNR estimation: Evaluation tool and analysis of VAD dependency. Radioengineering 14:6–11
17. Sjolander K, Beskow J (2000) WaveSurfer – an open source speech tool. In: Proceedings of ICSLP'00, vol 4, Beijing, pp 464–467
18. Talkin D (1995) A robust algorithm for pitch tracking (RAPT). In: Kleijn WB, Paliwal KK (eds) Speech coding and synthesis. Elsevier, Amsterdam, pp 495–518
19. Davis SB, Mermelstein P (1980) Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans Acoust Speech Signal Proc 28:357–366
20. Hermansky H (1990) Perceptual linear predictive (PLP) analysis of speech. J Acoust Soc Am 87:1738–1752
21. Bou-Ghazale SE, Hansen JHL (2000) A comparative study of traditional and newly proposed features for recognition of speech under stress. IEEE Trans Speech Audio Proc 8:429–442
22. Boer E (2001) Behavioral entropy as a measure of driving performance. In: Proceedings of the first international driving symposium on human factors in driver assessment, training, and vehicle design, Aspen, 14–17 August 2001
23. Boer E (2005) Steering entropy revisited. In: Proceedings of the third international driving symposium on human factors in driver assessment, training, and vehicle design, Rockport
24. Boyraz P, Sathyanarayana A, Hansen JHL (2009). Lane keeping metrics for assessment of auditory-cognitive distraction, [preprint] to appear in SAE Book, Chapter 10, Driver performance metrics
25. Daubechies I (1988) Orthonormal bases of compactly supported wavelets. Comm Pur Appl Math 41:909–996
26. Xie HB, He WX, Liu H (2008) Measuring time series regularity using non-linear similarity-based sample entropy. Phys Lett A 372:7140–7146

# Chapter 2
# Driver Emotion Profiling from Speech

**Norhaslinda Kamaruddin, Abdul Wahab, and Hüseyin Abut**

**Abstract** Humans sense, perceive, and convey emotion differently from each other due to physical, psychological, environmental, cultural, and language differences. For example, as recognized and studied by psychologists more than a century, it is easier for someone of the same culture to judge and recognize emotion correctly compared to those from different culture. In this chapter, we attempt to study the speech emotion recognition problem by using two speech corpora from the Berlin dataset and the NAW datasets. We have investigated the universality as well as diversity of two different cultural speech datasets recorded by German and American speakers, respectively. Experiments were conducted for identifying three basic emotions, namely, angry, sad, and happy with neutral as emotionless state from these datasets. MFCC coefficients were used as feature sets in the experiments, and MLP was employed as classifiers to compare the performance of these datasets. In addition, real-time recorded speech from drivers was also tested to see the performance in a vehicular setting. Finally, speech emotion profiling approach was introduced to explore the universality and diversity of the speech emotion features.

N. Kamaruddin (✉)
Faculty of Computer Science and Mathematics, University Technology MARA (UiTM), Shah Alam, Selangor, Malaysia
e-mail: norhaslinda@fskm.uitm.edu.my

A. Wahab
Faculty of Information and Communication Technology, International Islamic University (IIUM), Kuala Lumpur, Selangor, Malaysia
e-mail: abdulwahab@pmail.ntu.edu.sg

H. Abut
ECE Department (Emeritus), San Diego State University, San Diego, CA, USA

EE Engineering Department, Boğaziçi University, Istanbul, Turkey
e-mail: abut@anadolu.sdsu.edu

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_2, 

**Keywords** Berlin dataset • Mel frequency cepstral coefficients (MFCC) • Multilayer perceptron (MLP) • NAW dataset • Speech emotion profiling • Speech emotion recognition

## 2.1 Introduction

During the last century, many researchers from different disciplines have tried to postulate a few basic emotions out of the entire range of emotions that are tinged and enlivened. One of the models suggests that every emotion is composed of different levels of certain basic components including arousal, intensity, aversion, self-directedness, and others. Among many models, the prevailing one conjectures that emotions arise much the same way as colors do – presenting a myriad of hues out of the basic few constituents [7] To date, cognitive science does not possess a test to decide between various competing models of the basic emotion. However, researchers in various disciplines agree that some emotions are universally accepted as basic and many others as secondary. Cornelius has labeled six emotions as the "Big Six" [11], which are *angry*, *happy*, *sad*, *fear*, *surprise*, and *disgust*. These were chosen in this study. However, we focus only on *angry*, *sad*, and *happy* emotions with *neutral* as emotionless state in this chapter.

*Emotion recognition* from engineering perspective is a fairly new field of research compared to the psychologists' community. With the understanding that human convey and perceive underlying emotion in the interaction, scientists and researchers are able to analyze massive amount of information transmitted from a speaker to the listener using the tools of signal processing today. Yet, we are struggling to understand emotion and, more critically, capture and/or process it in a form that is useful for technical purposes.

In 2001, Sherer et al. have conducted a study in nine different countries in Europe, United States, and Asia on vocal emotion portrayals using content-free sentences containing anger, sadness, fear, joy, and neutral voice [9]. They found that generally the accuracy decreased with increasing language dissimilarity in spite of the use of language-free speech samples. It is concluded that culture-and language-specific paralinguistic patterns may influence the emotion recognition process.

In this chapter, we address this issue by proposing Mel Frequency Cepstral Coefficients (MFCC) as our features for speech emotion recognition. Our feature extraction method based on Slaney's [8] approach coupled with the WEKA multilayer perceptron (MLP) [12] classifier. These are adopted to identify the three basic emotions, namely, *angry*, *sad*, and *happy* emotional states. Initially, two different speech emotion datasets – using the NAW dataset (American actors) and Berlin dataset (German actors) – were employed to train and test the accuracy of the proposed system based on the K-fold validation technique. Next, we have extended our scope by using speech data recorded while driving in real time, to analyze and understand the driver behavior [6]. The driver was asked to interact with the passenger as well as

talking to a caller using a mobile phone equipped with a hands-free module as a safety precaution while driving. Three different scenarios were recorded based on:

- Driver under stress when talking on the mobile phone while driving
- Laughing while driving
- Driver feeling very sleepy

Data from these three driving scenarios were then compared with the two standard datasets, i.e., NAW and Berlin datasets.

In addition to the speech emotion recognition system under study, we also explore speech emotion profiling as an alternative tool to better understand speech emotion and in analyzing inter-and intra-cultural behavior. Such tool seems to provide a deeper insight to the hidden characteristics of speech emotion.

This paper is organized as follows: In Sect. 2, we present the theoretical and experimental framework for the proposed speech emotion recognition system based on a feature extraction method using Mel Frequency Cepstral Coefficients (MFCC) as features and MLP as classifier. In Sect. 3, experiments for the proposed speech emotion profiling system will be presented with some analysis of the driving dataset as compared to those using the NAW and Berlin datasets. Section 4 discusses the findings and conclusion of the study with some future work that can help extend the idea of profiling to its next level.

## 2.2 MFCC-MLP Speech Emotion

During the past couple of decades, MFCC feature has been successfully used for high-end speech recognition and speaker identification problems. However, there are many variations in applications in terms of number of filters, shape of filters, bandwidths, and the manner in which the spectrum is warped. In classification experiments, Slaney's approach [8] – founded on a study by Ganchev et al. [3] – gives a slightly better performance than many earlier works. Hence, we have adopted the approach described in Slaney's Auditory Toolbox for Matlab [8] in this study.

Once the MFCC features from the speech are extracted, the speech emotion is then classified/recognized using a multilayer perceptron (MLP) technique which is based on Bishop's work [1] wherein the preliminary experiments sought to determine the initial accuracy of the speech emotion recognition system. MLP uses a combination of several perceptron layers that are interconnected to each other and exhibit a high degree of connectivity, which is determined by the synapses of the network. It consists of three main layers which are the input layer, the hidden layer, and the output layer. In the input layer, the data is given to the network, thus the number of input neurons must be equivalent to the number of features for the data. Each data entry is given a weight by the network to be passed to the hidden layer where a nonlinear calculation will be carried out with the activation function. The output layer is the sum of the entire hidden layer outcomes. MLP uses the ubiquitous back-propagation algorithm as its learning procedure.

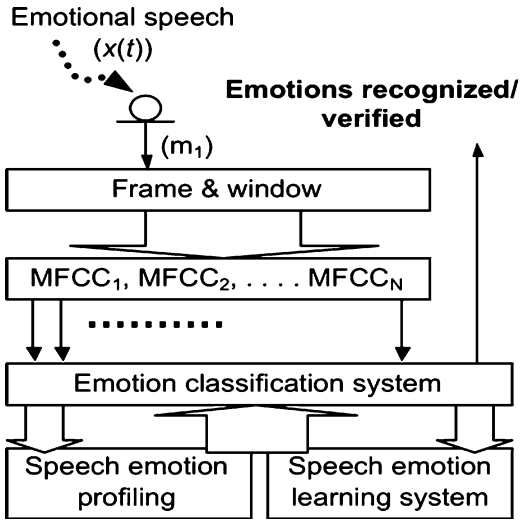



**Fig. 2.1** Shows the proposed speech emotion recognition system

Figure 2.1 shows the proposed speech emotion recognition system where the emotional speech is first filtered and framed. Forty MFCC features were then extracted and later classified using the MLP. The other additional module on speech emotion profiling and learning system is meant to enhance the speech emotion recognition system to cater for inter-and intra-cultural differences. In this chapter, only the emotion classification and some preliminary work on emotion profiling are presented next.

### *2.2.1 Berlin Dataset*

The Berlin Emotional Speech Database [2] contains ten sentences that have little emotional content textually. It is in German and covers seven emotion classes, namely, anger, fear, happy, sad, disgust, boredom, and neutral. The content of the spoken material is predefined and presented to five male and five female professional actors, respectively. The recording was done under studio conditions using a high-quality recording equipment and saved in mono wave format with an 8.0 kHz sampling rate. The complete database has been pre-evaluated through a manual perception test by 20 human subjects.

### *2.2.2 NAW Dataset*

The NAW dataset [13] was collated using some video clips from movies and television sitcoms obtained from the Internet. The participants are native speaker of American English. The emotions portrayed by the speakers have been analyzed and

**Table 2.1** Confusion matrix for human recognition performance for NAW dataset

| | Happy | Angry | Disgust | Surprised | Sad | Neutral |
|---|---|---|---|---|---|---|
| Happy | 76.5 | 0.0 | 1.5 | 12.0 | 0.0 | 10.0 |
| Angry | 0.0 | 90.0 | 5.0 | 0.0 | 4.0 | 1.0 |
| Disgust | 2.0 | 32.5 | 34.5 | 6.5 | 3.0 | 21.5 |
| Surprised | 9.0 | 2.0 | 8.0 | 64.5 | 1.5 | 15.0 |
| Sad | 0.0 | 0.0 | 0.5 | 0.0 | 98.0 | 1.5 |
| Neutral | 1.0 | 0.0 | 2.5 | 0.0 | 0.0 | 96.5 |

identified based on speech semantics, facial expression of the speaker, as well as basic understanding of the situations of the video clips occurrences. These video clips were converted to MP3 audio files at a sampling rate of 8.0 kHz, mono stream, and their amplitudes were scaled in the range $(-1, +1)$ V. A number of findings using this dataset have been reported earlier in [4, 5, 13].

### *2.2.3 Human Perception Test*

In order to ensure that the video clips obtained for the NAW dataset were correctly perceived, manual perception test were subsequently carried out. In this test, a total of 40 human subjects – 11 from Nanyang Technological University, Singapore (nine males and two females), and 29 from International Islamic University, Malaysia (15 male and 14 female), with an age mean of 23 years – have volunteered to provide their perceived assessment of the speech emotion audio files presented.

The participating subjects have reported that they have experienced neutral emotion prior to the commencement of the human perception test. The survey was conducted in a laboratory environment where the judges can listen to the speech emotion audio files with minimal distraction. They sat in front of a computer and listened to the speech emotion audio files via a headphone to ensure that judges can hear audio files without interruption. For each speech emotion audio file, they indicated the perceived emotion on a six-force-choice format representing the emotion classes with neutral as shown in Table 2.1.

In order to avoid any misled perception, each speech emotion audio file's name was labeled using a file number that has no relation to the respective emotion. In addition, the file numbering was also randomized to avoid any prediction of the emotion pattern. The human judges were allowed to listen to any of the speech emotion audio files repeatedly prior to making an appropriate decision.

Table 2.1 shows the confusion matrix for human recognition performance for the NAW dataset. Here, it can be seen that most judges were able to identify *sad*, *angry*, *neutral*, and *happy* quite easily with at least 76% accuracy. This is followed by *surprised* with 64% accuracy and *disgust* with only 34% accuracy, respectively. Disgust yielded very low recognition, which shows that the judges were not clear with its definition that they might have perceived disgust as mild anger thus resulting in higher percentage of anger being perceived. Similarly, surprised

emotion also scored fairly low perception performance due to the judges' mixed perception that most of them categorized surprised as happy for positive surprised or disgust for negative surprised. Sad is the highest correctly perceived emotion with 98% recognition accuracy performance since it was observed from features that it has the most acoustically distinct features.

## 2.3 Speech Emotion Recognition and Profiling Experiments

### *2.3.1 Emotion Identification Experiments*

Identification experiments were carried out to investigate the performance of our proposed system in determining the emotion for a given speech segment. As shown in Fig. 2.2, our proposed system can yield accuracy ranging from 47.9% to 75.4% for Berlin dataset and 61.4–71.2% for NAW dataset, respectively.

As it can be seen from Fig. 2.2, the maximum and minimum accuracy percentages for both datasets are consistent wherein *sad* emotion resulted in the highest accuracy and the *happy* emotion the lowest accuracy. Based on these results, we can see that NAW dataset result is comparable to the Berlin dataset, and the combination of MFCC as feature extraction coupled with MLP can achieve reasonable accuracy performance. This indicates that our proposed approach has potential to recognize emotion in speech.

### *2.3.2 Understanding Driver's Emotion*

The approach was next applied to a pre-recorded driving data to identify emotional state of drivers while driving under varying scenarios. There were four scenarios for the driver emotional state, namely, *stress*, *laughing*, *neutral*, and *sleepy*, which were tested in these set of experiments. Stress data is taken while the driver was talking through a mobile phone while driving with the assumption that he/she

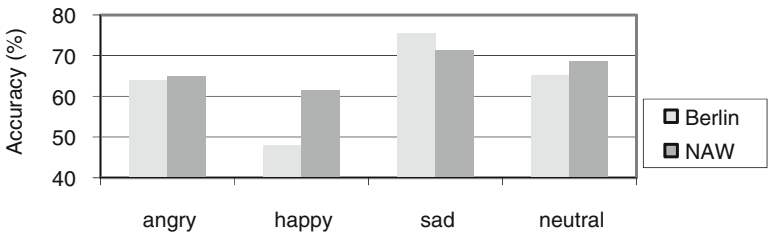



**Fig. 2.2** Identification result for Berlin and NAW dataset

**Table 2.2** Confusion matrix for identification results for real-time driving dataset

| | Stress (%) | Laugh (%) | Neutral (%) | Sleepy (%) |
|---|---|---|---|---|
| Stress | 55.6 | 19.9 | 13.3 | 11.2 |
| Laugh | 22.3 | 57.0 | 12.1 | 8.6 |
| Neutral | 5.4 | 6.9 | 74.5 | 13.2 |
| Sleepy | 8.5 | 7.4 | 17.6 | 66.5 |

**Table 2.3** Confusion matrix for identification results from combination of Berlin, NAW, and real-time driving datasets

| | Happy (%) | Sad (%) | Neutral (%) | Stress (%) | Angry (%) | Sleepy (%) |
|---|---|---|---|---|---|---|
| Happy | 50.6 | 11.5 | 14.9 | 4.9 | 15.4 | 2.8 |
| Sad | 12.7 | 59.5 | 15.4 | 1.0 | 9.4 | 2.1 |
| Neutral | 13.5 | 10.5 | 62.3 | 1.8 | 6.5 | 5.5 |
| Stress | 22.0 | 6.1 | 22.4 | 39.3 | 2.5 | 7.7 |
| Angry | 22.4 | 9.2 | 9.2 | 1.0 | 56.1 | 2.2 |
| Sleepy | 12.9 | 4.8 | 21.4 | 2.8 | 1.2 | 56.9 |

**Table 2.4** Confusion matrix for identification result of combination Berlin, NAW, and real-time driving dataset without neutral

| | Happy (%) | Sad (%) | Stress (%) | Angry (%) | Sleepy (%) |
|---|---|---|---|---|---|
| Happy | 59.4 | 13.5 | 5.8 | 18.1 | 3.2 |
| Sad | 15.0 | 70.3 | 1.2 | 11.1 | 2.5 |
| Stress | 28.4 | 7.9 | 50.6 | 3.2 | 9.9 |
| Angry | 24.6 | 10.1 | 1.0 | 61.8 | 2.4 |
| Sleepy | 16.4 | 6.1 | 3.6 | 1.5 | 72.4 |

needed to multitask between concentrating on his/her driving and at the same time providing appropriate responses to the caller prompts. The results are tabulated in Table 2.2.

From Table 2.1, we could see that the proposed system can identify at least 55.6% and can reach up to 74.5% of the driver emotional state using the same driving dataset. Neutral yielded the highest accuracy, and stress obtained the lowest accuracy.

In order to have better understanding of the proposed system performance, we have combined the three datasets consisting of Berlin, NAW, and the driving datasets, and have conducted identification experiments. Since *laughing* is a reaction when the driver is *happy*, we assumed that laughing is a subset of happy emotion. The identification results are provided in Table 2.3. It is clearly seen that the lowest accuracy yielded for stress data with only 39.3% accuracy while the maximum accuracy obtained by neutral with 62.3%.

The accuracy of such system can be improved if the neutral state is removed from the dataset. From the understanding of Schlosberg's affection space model [10], the neutral state is the speech emotion basis regardless of their emotion primitives' axes. Thus, pure emotion can be extracted from the processed speech if we can remove neutral from our findings. Confusion matrix of Table 2.4 shows a rather interesting result when the neutral is removed. The accuracy of the proposed system is increased by approximately 10%.

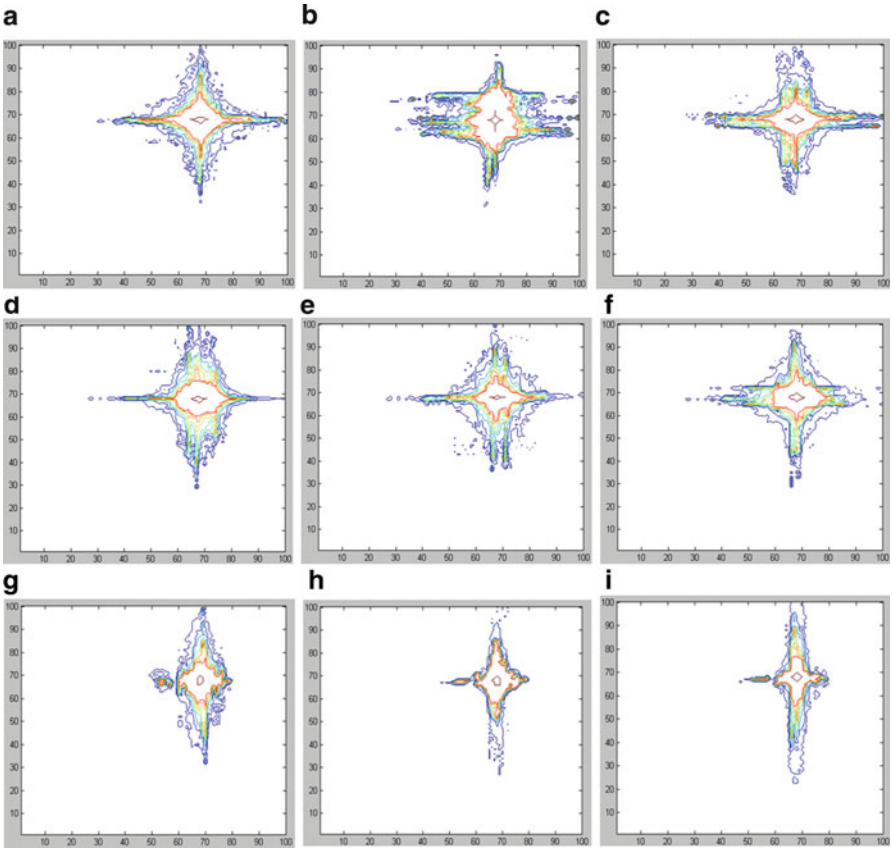



**Fig. 2.3** Speech emotion profiling for Berlin, NAW, and driving datasets: (**a**) Berlin – angry; (**b**) Berlin – sad; (**c**) Berlin – happy; (**d**) NAW – angry; (**e**) NAW – sad; (**f**) NAW – happy; (**g**) Driving – stress; (**h**) Driving – sleepy; (**i**) Driving – laugh

### *2.3.3 Speech Emotion Profiling*

Based on the results presented in Sect. 2, we apply speech emotion profiling method to the data in order to visualize the correlation between speech emotion signal and the neutral state. It is interesting to note from Fig. 2.3**a–i** that even though the data used is different, the pattern is similar for the same emotion across datasets, and yet the distinction is clearly observable for different emotion within a given dataset.

The most obvious example is the profile plot of *happy* emotion which has a cross pattern for all three dataset, although the data is completely different. Figure 2.3 also indicates that it is possible for us to visualize the inter- and intra-cultural variations of the speech emotion, which can lead us to better understand the effect of these cultural artifacts to improve speech emotion recognition globally.

## 2.4 Summary, Conclusion, and Future Work

Speech emotion profiling can be an effective tool for investigating intra- and inter-cultural variations from various perspectives. It enables one to visualize the interaction of the emotion that may give important information which is not observable using normal signal analysis tools, namely, the speech recognition and speaker identification. More work in understanding the profile especially in extracting the relevant features as well as the appropriate data processing are needed to benefit from such visualization tool. The speech emotion profile coupled with a three-dimensional affective-space model may be able to provide a better understanding of the dynamics of driver behavior. This work also illustrated that there are strong correletions between driver behavior and emotion which can be emperically measured using speech signals.

## References

1. Bishop C (1997) Neural networks for pattern recognition. Clarendon, Oxford
2. Burkhardt F, Paeschke A, Rolfes M, Sendlmeier W, Weiss B (2005) A database of German emotional speech. In: Proceedings of INTERSPEECH-ISCA, Lisbon, pp 1517–1520, 2005
3. Ganchev T, Fakotakis N, Kokkinakis G Comparative evaluation of various MFCC implementations on the speaker verification task. In: Proceedings of the 10th international conference on speech and computer (SPECOM 2005). Patras, vol 1, pp 191–194, 2005
4. Kamaruddin N, Wahab A (2008) Feature extraction for speech emotion. In: Proceedings of the 17th international conference on software engineering and data engineering (SEDE ‘08), Los Angeles, pp 120–125
5. Kamaruddin N, Wahab A (2008) Speech emotion verification system (SEVS) based on MFCC for real time applications. In: Proceedings of the 4th international conference on intelligent environments (IE ‘08), Seattle, pp 1–7
6. Khalid M, Wahab A, Kamaruddin N (2008) Real time driving data collection and driver verification using CMAC-MFCC. In: Proceedings of the 2008 international conference on artificial intelligence (ICAI ‘08), Las Vegas, pp 219–224
7. Plutchik R (2003) Emotions and life: perspective from psychology, biology and evolution, 1st edn. American Psychological Association, Washington, DC
8. Slaney M (1998) Auditory toolbox: Version 2. Technical Report #1998-010, Interval Research Corporation
9. Scherer KR, Banse R, Wallbott HG (2001) Emotion inferences from vocal expression correlate across languages and cultures. J Cross-Cultural Psychol 32(1):76–92
10. Schlosberg H (1954) Three dimensions of emotion. Psychol Rev 61:81–88
11. Cornelius, R. R. (1996). *The Science of Emotion: Research and Tradition in the Psychology of Emotion*, Upper Saddle River, NJ: Prentice-Hall
12. Witten, I. H., Frank, E., Trigg, L., Hall, M., Holmes, G. & Cunningham, S. J. (1999) Weka: Practical Machine Learning Tools and Techniques with Java Implementations. *In: N. Kasabov & K. Ko (Eds.). Proceedings of the ICONIP/ANZIIS/ANNES’99 International Workshop on Emerging Knowledge in Engineering and Connectionist-Based Information Systems*. Dunedin, New Zealand, 192–196
13. Kamaruddin N. & Wahab A. (2009). Features Extraction for Speech Emotion. *Journal of Computational Methods in Science and Engineering (JCMSE), 9 (Supplement 1)*, S1–S12, 2009

# Chapter 3
# Driver Status Identification from Driving Behavior Signals

**Emre Öztürk and Engin Erzin**

**Abstract** Driving behavior signals differ in how and under which conditions the driver uses vehicle control units, such as pedals, driving wheel, etc. In this study, we investigate how driving behavior signals differ among drivers and among different driving tasks. Statistically significant clues of these investigations are used to define driver and driving status models. Experimental results over the UYANIK database are presented. Driver identification over 23 drivers achieves a 57.39% identification rate with the fusion of gas and brake pedal pressure classifiers. Driver identification system with reduced number of drivers fits better on real-life scenarios. Driver identification rate within groups of three drivers is computed as 85.21%. Driver status identification over ten drivers with task and no-task classes yields a promising 79.13% task identification rate. Driving behavior is strongly related to past actions of drivers. In this study, we investigate driving behavior prediction from past driving signals. We propose a behavior prediction system, which performs temporal clustering of behavior signals and computes linear estimators for each temporal cluster. The temporal clustering is performed with hidden Markov model (HMM). Experimental evaluations show that distractive conditions have a certain effect on driving behavior, where the prediction errors are significantly increasing in these conditions. Road conditions are also influential on driving behavior prediction.

**Keywords** Driver status identification • Drive-safe • Driving behavior prediction • Driving behavior signal • Driving distraction

E. Öztürk • E. Erzin (✉)
Multimedia, Vision and Graphics Laboratory, Koç University, Sarıyer, Istanbul, Turkey
e-mail: eerzin@ku.edu.tr

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_3, 

## 3.1 Introduction

Recent developments in man–machine interaction have created a wide range of applications. Among those applications, human–vehicle interfaces have been studied extensively in the recent literature. Next-generation human–vehicle interfaces will likely incorporate biometric person recognition using speech, video, images, and analog driver behavior signals to provide more efficient and safer vehicle operation. Furthermore, driving behavior signals, such as pedal signals, velocity, and car-following distance, yield important clues on driving behavior status and driver's cognitive stress/distraction.

There have been significant efforts on the investigation of driving behavior patterns using driving behavior signals. Kurahashi et al. used driving behavior signals to quantify workload factors for driving behavior modeling [1]. The Center for Acoustic Information Research at Nagoya University has been collecting multimodal driving behavior signals since 1999 [2]. Their early studies investigate cepstral analysis of driving behavior signals [3] and modeling driver behavior as car-following and pedal-operation patterns [3, 4]. Recently, they investigate near-miss accidents by conducting interviews to determine driver behavior and cognitive state immediately before the incident [5]. Driver identification from biometric and driving behavior signals have been investigated within a multimodal decision fusion system [6].

Car-following data collection and modeling have also been investigated across different research centers [4, 7]. Predicting driver's future actions with the resultant behavior and past observations have been studied with implications to model the impact of Intelligent Transport Systems (ITS) [5, 8]. Tezuka et al. investigate prediction of driving behavior signals by capturing time-series steering angle data at the time of lane change with conditional Gaussian models and Bayesian networks [8]. Kishimoto and Oguri applied to Dynamic Bayesian Networks to construct a behavior model for inference of stop behavior [14]. They have revealed that using past movements has a great influence for predicting stop probability. A multimodal signal processing system for robust stress detection in urban driving scenarios has been proposed in [9]. Marinova, Devereaux, and Hansman have studied the effects of cell-phone conversations on driver reaction time and situation awareness at different levels of cognitive with hands-free and hand-held cell-phone configurations [10]. Cognitive workload and driver experience, using a secondary task method, the peripheral detection task (PDT) in a field study, has also been explored [11].

The Nagoya University CIAIR center leads the effort on international research coordination of driving behavior signal processing based on large-scale real-world database [2]. Within this research coordination, UTDrive of University of Texas at Dallas collects multimodal driving behavior data [12]. UTDrive investigates driver's cognitive stress/distraction to adapt interactive systems for improved safety. Similarly, the Drive-Safe consortium, which has partners from the academia and industry in Turkey, collects a similar multimodal driving behavior corpus to create conditions for prudent driving in [13].

In this chapter as a partner of the Drive-Safe consortium, we investigate driver identification, driving status identification, and driver behavior prediction under

different cognitive stress/distraction conditions using driving behavior signals. Our objective is to search out and examine the effects of cognitive distraction conditions on driving behavior and inquire whether driving behavior signals are characteristic information for every driver. We investigate task identification performances, where our earlier findings are presented in [17].

In this chapter, we present our contributions on the following three major problems:

- *Driver Identification*: Identification of a driver using behavioral signals is one the most interesting in-vehicle signal-processing problems. In this study, we use driving behavior signals such as vehicle speed, gas pedal pressure, brake pedal pressure, and distance from the vehicle in front for driver identification. First, we investigate the characteristics of these signals and present a selected set of driving statistics. Then we define a statistical driver identification system and evaluate this system experimentally.
- *Driver Status Identification*: Distractive conditions cause important safety problems to drivers. Studies have shown that nearly 80% of traffic accidents occur due to driver inattention, which are commonly results of distractive conditions. Navigation systems and other services in vehicles introduce many secondary driving tasks that can increase accident risk. Thus, developing a distraction detection method would be very beneficial for in-vehicle system to reduce the effects of distraction. In this study, driving experiments were done under some distractive conditions, which can be considered as the secondary driving tasks stated above. These tasks are dialog on cell phone, including route navigation and online banking, conversation with passenger on-board, and signboard and license plate reading. We investigate the statistical nature of driving behavior signals under different driving tasks, which are defined as distractive conditions. Then we attempt to detect distractive conditions using statistical classifiers.
- *Driver Behavior Prediction*: Human factors play a big role in traffic accidents. Predicting driving behavior is an important issue since it has a significant effect on decreasing human-caused accidents. Drivers' behavior is strongly related to their past actions, so in this study, we construct a driver behavior prediction model using drivers' past behavior signals. The driver behavior prediction model consists of temporal clustering with hidden Markov models (HMM) and minimum mean-square error (MMSE) estimation within each temporal segment. We also investigate the influence of road conditions and distractive conditions on our prediction model.

## 3.2 Driving Behavior Signal Characteristics

Driving signals differ in how and under which conditions the driver use vehicle control units, such as pedals, driving wheel, etc. We aim to model individual differences among the selected drivers and identify the drivers by using gas

pedal pressure, brake pedal pressure, vehicle velocity, and fusion of these signals. We also benefit from car-following distances. Driving behavior characteristics differ from person-to-person under different distractive conditions. In order to examine the effects of these distractive conditions, we investigate how driving behavior signals differ across driving tasks. Statistically significant clues of this investigation are used to define a driving status model. This section presents general characteristics and statistics of the driving behavior signals from the UYANIK database, feature representation of these driving behavior signals, and the statistical clustering, identification framework for the driver and driving status and predicting driver behavior.

### *3.2.1 Data Collection*

Driving behavior data was supplied by the Drive-Safe Consortium in Turkey with the test vehicle, UYANIK, which is a sedan car equipped with various sensors. The UYANIK database includes synchronous audio-visual recordings, CAN-Bus readings, pedal-sensor recordings, 180° laser range finder, and XYZ accelerometer recordings [13].

The data collection route is around 25 km at about 40 min, starting and ending at the OTAM Research Center in the ITU Campus in Ayazaga. It consists of two 1.5 km-very-busy city sections, followed by the TEM highway with much less traffic. Next, the route goes through the city, and then it goes back to the OTAM at ITU campus. The last segment is very busy with local traffic. The route is the same for all drivers. However, road conditions may differ depending on traffic jam and weather in Istanbul. We use a subset of the UYANIK database including driving behavior signal recording sessions of 20 male and 3 female drivers.

There are four primary tasks in the UYANIK database: (1) *reference driving* which includes no specific driving task, (2) *dialog on cell phone* which includes online banking application and navigational dialog, (3) *signboard reading* in which driver reads road-by signs and license plates aloud, and (4) *dialog with passenger* where driver talks with the on-board passenger.

### *3.2.2 Driving Behavior Signals*

We consider gas and brake pedal pressure signals, velocity from CAN-Bus, and car-following distance from the laser range finder as driving behavior signals. The gas, brake, and velocity signals are all sampled at 32 Hz, and the laser range finder sweeps 180° at every 2 s. Samples of driving behavior signals are given in Fig. 3.1.

The laser range finder in front of the vehicle records two-dimensional (x, y) data consisting of horizontal and vertical distances. Figure 3.2 shows the Laser Scan Reading and the photo for a selected driver recorded at 12:56 PM on April 6, 2007.

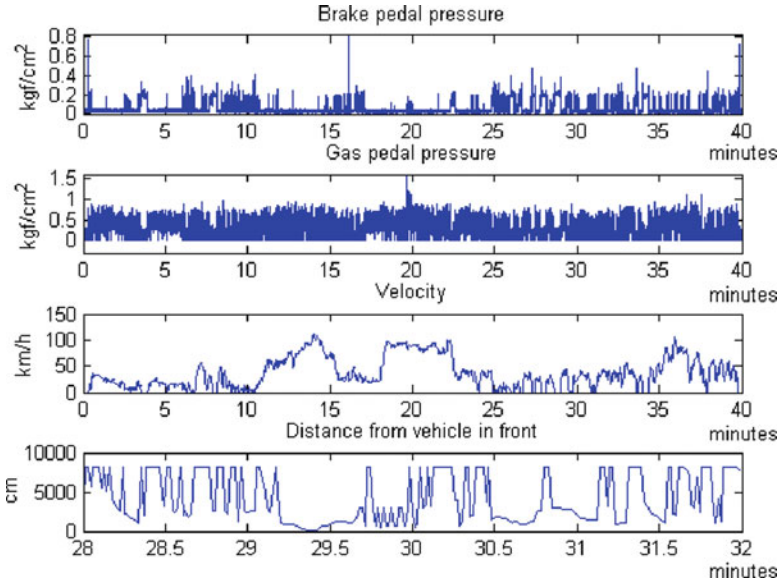



**Fig. 3.1** Driving behavior signals of a driver from the UYANIK database

The truck on the right is between −200 cm and +800 cm, the white truck is 22 m away, and the vehicle on the next lane (left) is about 23 m ahead [13].

The histograms of the driving behavior signals for all 23 drivers driving under highway and city traffic conditions are shown in Fig. 3.3. It shows that on the highway, drivers rarely use the brake pedal and steps on the gas pedal much more. The maximum range that the laser can sweep is about 80 m and generally most of the drivers exceed this distance on the highway.

Meanwhile, histograms of the driving behavior signals, taken from two randomly selected drivers, are shown in Fig. 3.4. The driver on the left side of the figure prefers driving faster and rarely uses the brake pedal. Also, he or she generally keeps distance from the vehicle in front for all road conditions, while the other driver prefers following a vehicle with closer distances where he or she faces traffic jam, and uses the brake pedal much more. Such differences are clear indications that driving behavior signals differ among drivers.

## 3.3 Driver Behavior Modeling

Modeling driver behavior is very important in enhancing the safety of drivers and pedestrians. Driver authentication, early warning systems for vehicles, and other technologies for security purposes can be given as the application areas of driver behavior modeling. Driving behavior, in itself, is a cyclic process [16].

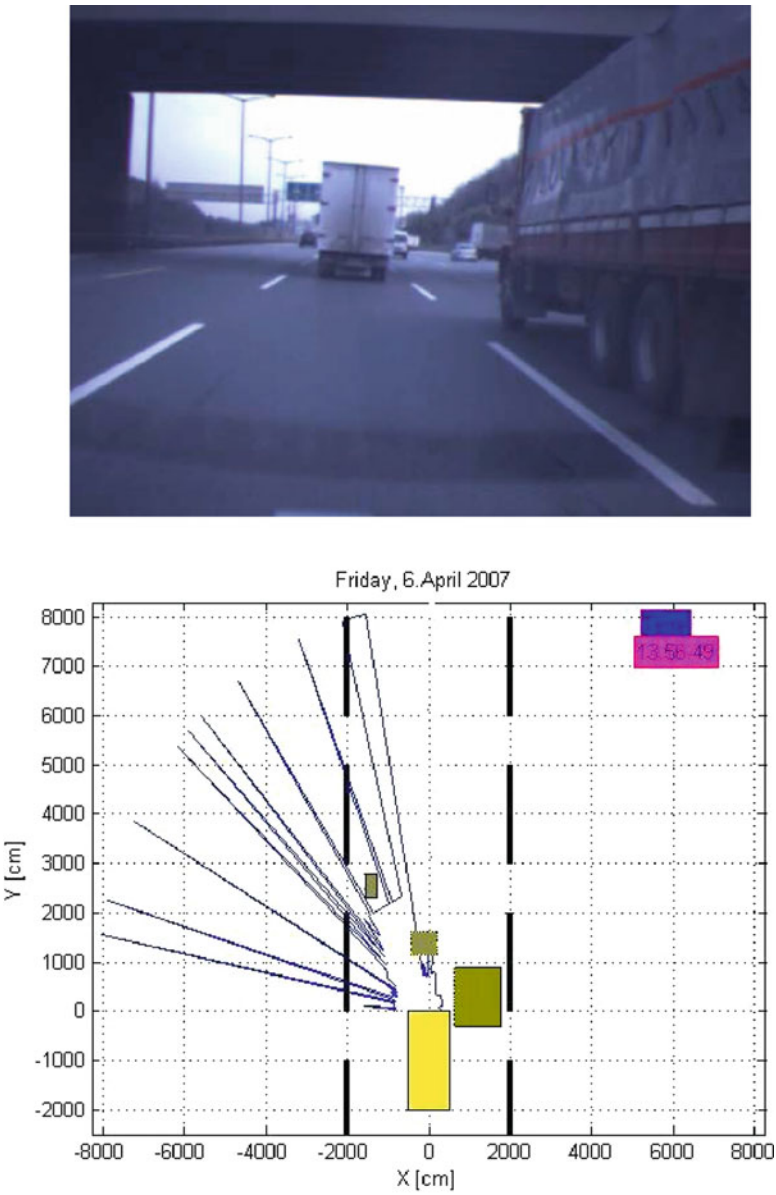



**Fig. 3.2** Laser scan reading and the photo for a selected driver

A driver determines the action to take by considering the road environment and operates the gas or brake pedal. The velocity of the vehicle changes according to the driver's operation and the distance from the vehicle in front (road environment), and it also changes according to the vehicle's status.

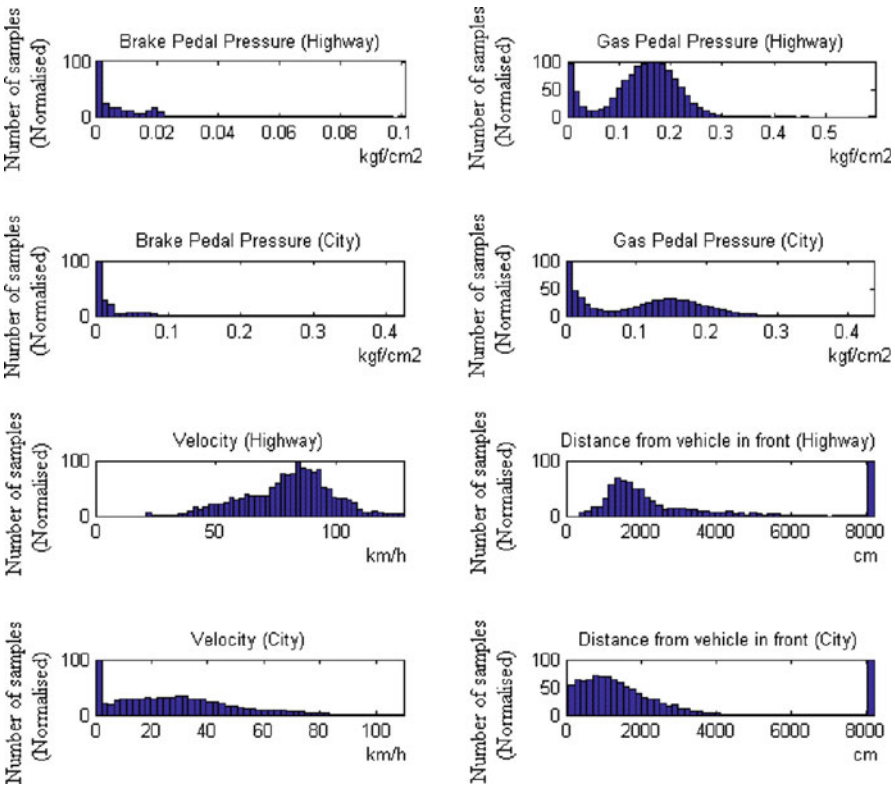



**Fig. 3.3** Histograms of the driving behavior signals from highway (top) and city (bottom) traffic

In this section, we discuss feature extraction, driver identification, and driver behavior signal prediction, and their role for driver behavior modeling. Driver identification is based on recognition of driving feature vectors using a statistical model. Our model is designed using a training and test procedure. In the training part, our algorithm learns the statistical nature of the data from a training set constructed by extracting the driver behavior features. In the testing part, the algorithm's accuracy is measured on a testing set, which is completely different from a training set.

### *3.3.1 Feature Extraction*

A preprocessing step, which is the high-pass filtering of the driving signals including gas pedal pressure, brake pedal pressure, and vehicle velocity, is applied to remove the DC component. Then, we apply cepstral analysis, which is a known source/filter

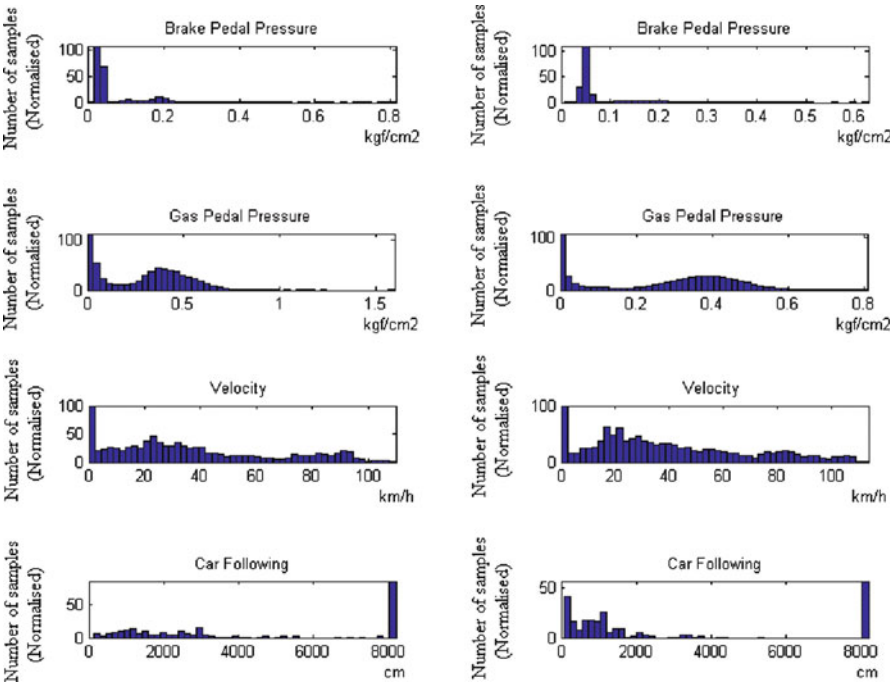



**Fig. 3.4** Driving behavior signal histograms of two drivers: driver one on the left, driver two on the right columns

separation method, and it has been used for driving behavior signals. Cepstral analysis captures significant information from driving behavior signals. In driver modeling, hitting a gas or brake pedal is filtered with a driver model represented as the spectral envelope. Spectral envelopes of pedal-operation signals represent the differences in pedal-operation patterns. These spectral envelopes are similar in the same driver and vary across different drivers.

In this study, we extract cepstral features for the gas and brake pedal pressure and velocity signals, which are sampled at 32 Hz. The cepstral features are extracted over 800-ms windows for every 96-ms frames. The cepstral feature is defined as the first $K$ coefficients of the discrete cosine transform of band-pass filtered log-magnitude spectra,

$$f_k = DCT\{BPF\{\log|F\{x_w(n+kT)\}|\}\} \tag{3.1}$$

where $k$ is the frame index, $x_w(n+kT)$ is the windowed signal of duration $T$. In order to eliminate high-frequency noise, we apply band-pass filtering with 1–13 Hz cutoffs for brake signal and with 1–6.5 Hz cutoffs for gas and velocity signals. The dimension of the feature vector is set as $K = 10$.

### 3.3.2 Driver Identification Model

The ability to identify a driver and his/her driving behaviors is related with how he/she hits the gas and brake pedals. We model the statistical nature of these pedal-operation patterns with Gaussian mixture models (GMM). The maximum posteriori probability approach to the N-class identification problem requires computation of conditional probability $P(\lambda_n|f)$ for each class $\lambda_n$, $n = 1,\ldots,N$, given a feature vector $f$ representing the sample data of an unknown class. An alternative is to employ the maximum likelihood solution, which maximizes the class-conditional probability,

$$\lambda^* = \arg\max_{\lambda_n} \log P(f|\lambda_n). \tag{3.2}$$

Furthermore, the likelihood scores coming from different classifiers can be combined at decision level (decision fusion) using weighted summation rule,

$$\lambda^* = \arg\max_{\lambda_n} \sum_k \alpha_k P(f_k|\lambda_n), \tag{3.3}$$

where $0 \leq \alpha_k \leq 1$ is the weight of the *k-th* classifier and $\sum_k \alpha_k = 1$ .

The computation of class-conditional probabilities needs a prior modeling step, through which we estimate a probability density function of feature vectors for each class $\lambda_n$, $n = 1,\ldots,N$ from available training data. The class-conditional probability density functions are modeled using the Gaussian mixture densities,

$$P(f|\lambda_n) = \sum_{k=1}^{M} \omega_k N(f; m_k, C_k), \tag{3.4}$$

where $m_k$ and $C_k$ are respectively mean vector and covariance matrix of the *k-th* mixture, and $M$ is the total number of mixtures.

### 3.3.3 Driver Behavior Prediction

We propose a driver behavior prediction system, which performs temporal clustering of behavior signals and computes linear estimators for each temporal cluster, based on the work in [15]. The temporal clustering is performed with hidden Markov model (HMM). Within each temporal segment linear estimators predict current driving behavior sample from $N$ recent samples of all behavior signals. The consistency of the predicted signal and the actual signal is expected to give us an idea about driving quality. We employ brake, gas pedal strokes, and velocity for our prediction model. Flowchart of predicting driver behavior is shown in Fig. 3.5.

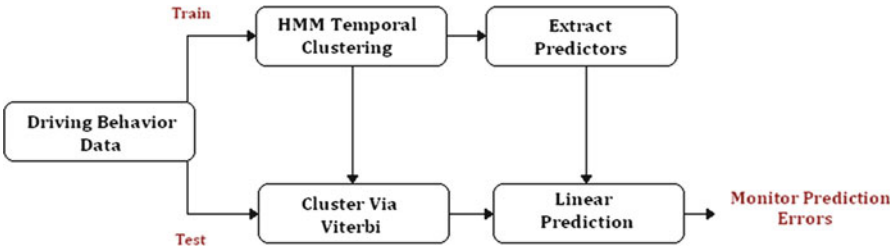



**Fig. 3.5** Flowchart of the driving behavior prediction system

First we build a temporal clustering model for all driving signals using HMM structure. Then we apply linear prediction to predict the desired driving behavior signal within temporal segments. The state sequence, which defines segment boundaries, is determined using the Viterbi algorithm. In each segment, we perform a linear prediction analysis to estimate current driving behavior sample from $N$ recent driving behavior samples. We construct the feature vector $d(n) = [b(n), g(n), v(n)]$, where $b$, $g$, and $v$ denote the direct samples taken from brake pedal pressure signal, gas pedal pressure signal, and velocity signal of the corresponding segment, respectively. We construct a temporal feature vector $x_n$ by combining $p$ past samples of the driver behavior samples:

$$x_n = [d(n-1), d(n-2), ..., d(n-p)]. \tag{3.5}$$

The optimal MMSE predictor to estimate the driving behavior signal at time instant $n$ can be given as,

$$\hat{y}(n) = \bar{y} + C_{YX} C_{XX}^{-1} (x_n - \bar{x}), \tag{3.6}$$

where $\hat{y}(n)$ is the driving behavior signal sample to be estimated, $\bar{y}$ is the mean driving behavior signal, $\bar{x}$ is the mean temporal feature vector, and $C_{YX}$ and $C_{XX}$ are the cross- and autocorrelation functions. The driving behavior signal $y(n)$ can be taken as any one of the brake pedal pressure, gas pedal pressure, and velocity signals. Note that the minimum mean squares error (MSE) is calculated as:

$$MSE = E\{||y_n - \hat{y}_n||^2\}, \tag{3.7}$$

where $y_n$ is a sequence of driving behavior signal.

## 3.4 Experimental Results

In the experimental evaluations, we use two subsets from the UYANIK database. The first subset, U-DRIVER, includes 23 drivers to be used for the driver identification evaluation (three of these drivers are not used for the car-following task since

they miss laser range information). The second subset, U-TASK, includes ten drivers to be used for the driving task identification. Driver identification performance for a particular task domain depends on the selection of accurate training database of interest in that domain. So, in order to achieve more realistic identification results, we divide the U-DRIVER into three groups. Assuming that a vehicle is generally used by a limited number of different drivers, each of these groups is arranged in 20 subgroups including three, four, and five drivers respectively. Driver identification is performed for all these 60 subgroups independently. Also, we benefit from this subset in order to predict driver behavior signals. The four primary tasks are transcribed on the U-TASK subset. In all driver and task identification evaluations, we use fivefold cross validation, where the available database is divided into five equal-length segments (the first segment starts with the beginning of the driving session, the second one starts with the end of the first segment, and the others follow the same procedure), and evaluations are performed over leave-one-segment-out train and test scheme. In driver behavior prediction evaluations, we use four-fold cross validation.

### *3.4.1 Results on Driver Identification*

Every driver has different driving behavior characteristics. Drivers vary in how they use the gas and brake pedals and how much distance they keep when following a vehicle. As described earlier, the gas, brake, and velocity signals are all sampled at 32 Hz, and the cepstral features are extracted over 800 ms (25 samples) of windows for every 96 ms (three samples) of frames. Figure 3.6 shows the driver identification performance of a GMM classifier on the U-DRIVER database including 23 drivers for brake pedal pressure, gas pedal pressure, and velocity signals using cepstral coefficients with varying number of mixture components. For identification purpose, we use different decision-window lengths and calculate the features for every 30 s of frames. Since brake pedal is not used frequently on highway, driver identification using brake pedal is performed only on city driving recordings.

As shown in Fig. 3.6, the gas pedal pressure signal yields better performance than the brake pedal pressure signal. This is possibly due to the more frequent use of gas pedal by drivers. The best identification results for all behavior signals are obtained by using GMM classifiers with 16 mixtures over 8–10 min of decision windows. The unimodal driver identification rates are all below 60%, which presents a fair driver identification system with possible room for improvement.

Decision fusion of classifiers with different driver behavior signals can improve unimodal identification rates. We investigate the fusion of classifiers with gas, brake, and velocity signals, and identify fusion structures with improved identification rates. Figure 3.7 presents decision fusion results of the driver identification system over different decision-window sizes. The optimal weights of the classifiers in the decision fusion are set experimentally over a partition of the training data. The resulting weights are set as $\alpha_g = 0.77$ in the brake (B) and gas (G) fusion, $\alpha_g = 0.79$ in the velocity (V) and gas (G) fusion. The best identification result is

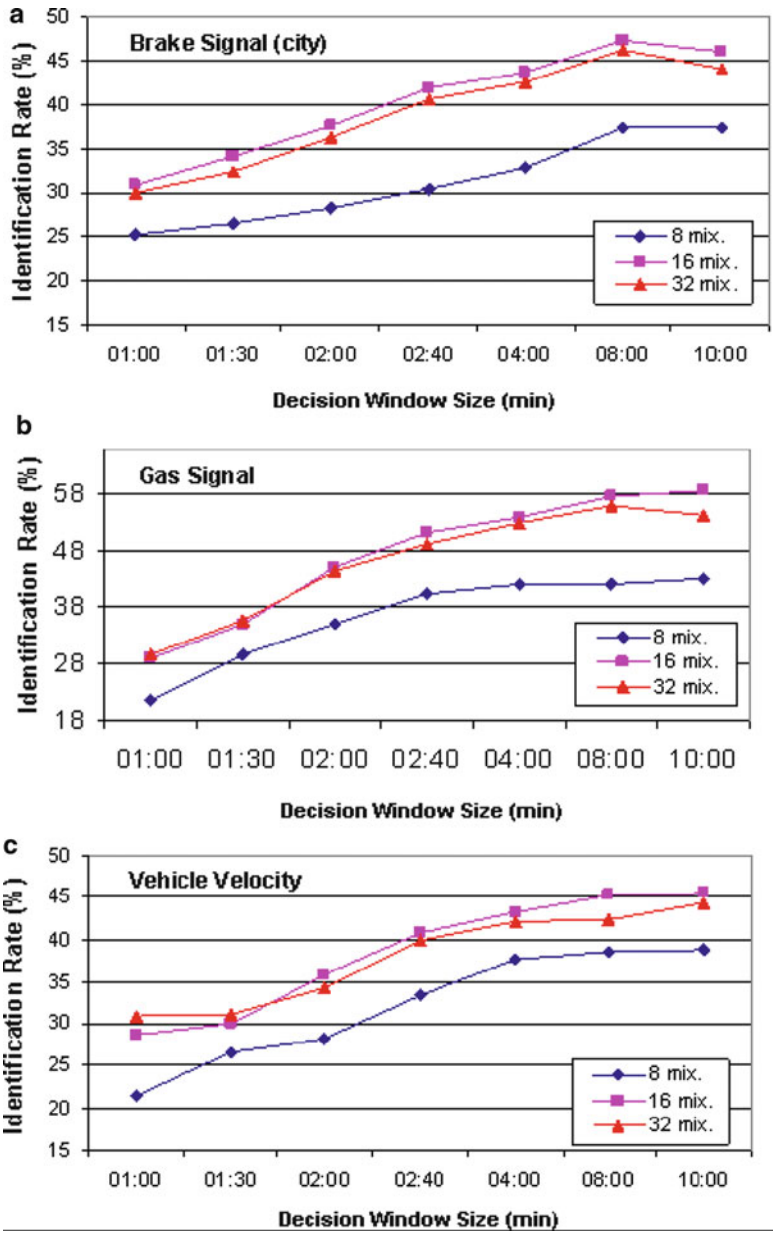



**Fig. 3.6** Driver identification rates with the (**a**) gas pedal pressure, (**b**) brake pedal pressure, and (**c**) velocity signals

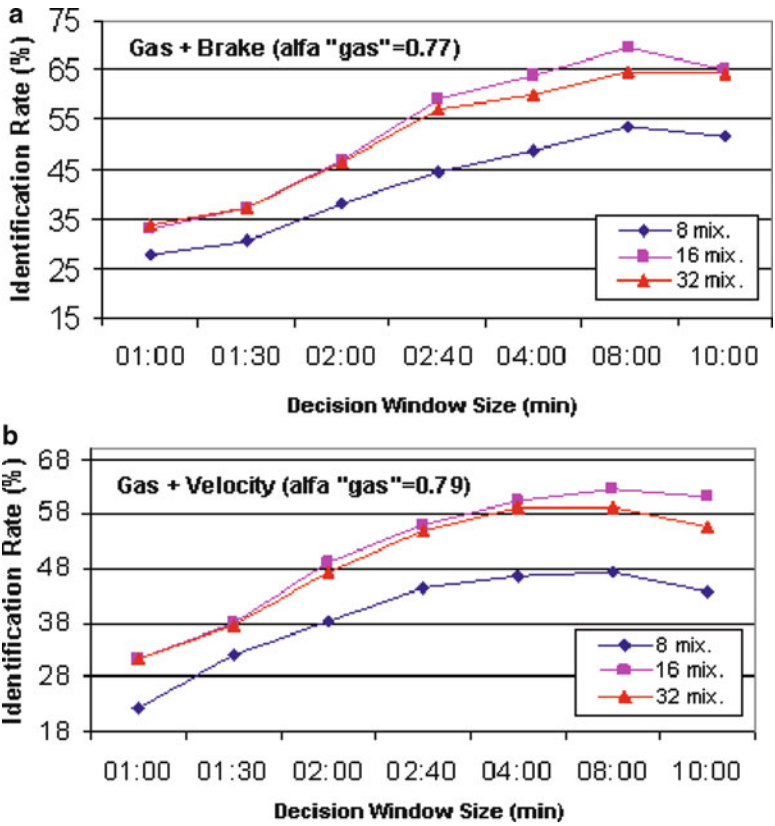



**Fig. 3.7** Driver identification rates with the decision fusion of (**a**) gas pedal pressure + brake pedal pressure, (**b**) gas pedal pressure + velocity

obtained as 69.5% with the fusion of gas (G) and brake (B) pedal pressure signals by using 16 mixtures of GMM. The best scenarios for all modalities are summarized in Fig. 3.8. We can observe from these results that decision fusion method significantly increases our system performance.

We also investigate the car-following distance measurements for the driver identification problem. Car-following distance measurements are collected using a laser range finder which sweeps 180° at every 2 s and measures the distance to the nearest object at each angle. The distance from the vehicle in front is acquired when the laser range finder is at 90°. Since the maximum range that the laser can sweep is about 80 m and most of the drivers exceed this distance on both highway and two-way roads, we only employ car-following distance signals on one-way roads. For one-way roads, the car-following task is around 4°min long for each driver.

Figure 3.9 shows the identification results for the car following distance signals over the U-DRIVER database including 20 drivers at different test lengths. Since the length of the task is rather short, selecting the decision window size is very crucial.

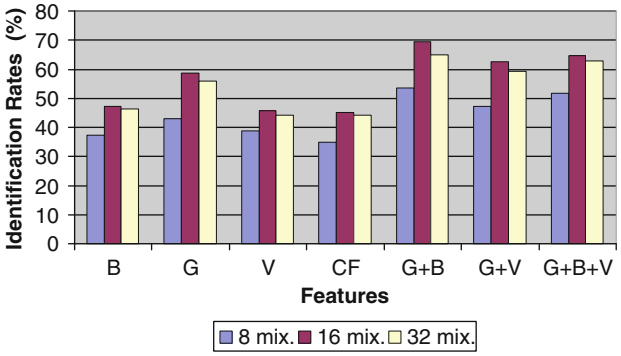



**Fig. 3.8** Comparison of driver identification rates for unimodal and multimodal classifiers

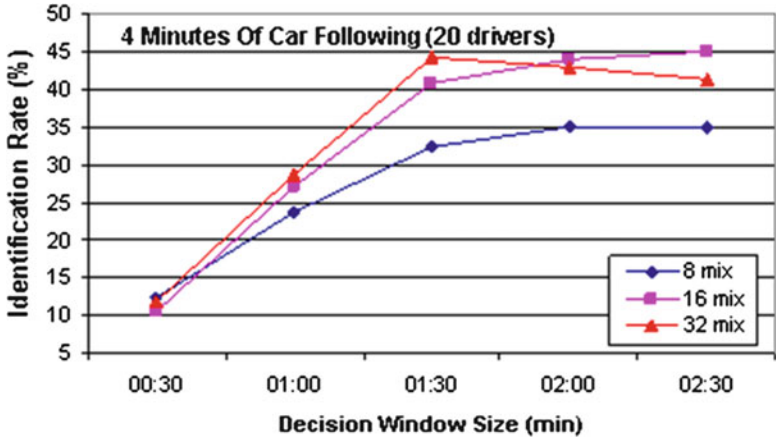



**Fig. 3.9** Driver identification rates with the car-following distance signals

The best performance is achieved as 45% with the 16 mixture GMM classifier with 150 seconds decision windows.

As the accelerator pedal is operated directly by the driver, it yields us the best feature to identify driver characteristics. Since the distance from the vehicle in front and vehicle velocity are the results of the driver's pedal operations, we achieve poor results by using only these features.

In a real-life scenario, typically a car is used by several drivers. Hence we investigate the performance of the driver identification system with reduced number of drivers. The dataset is divided into three groups, which are made up of 20 different subgroups including 3, 4 and 5 drivers respectively. We employ the same 16 mixture GMM classifier with 8 minutes decision window, which have been observed to achieve better identification rates in the earlier experiments.

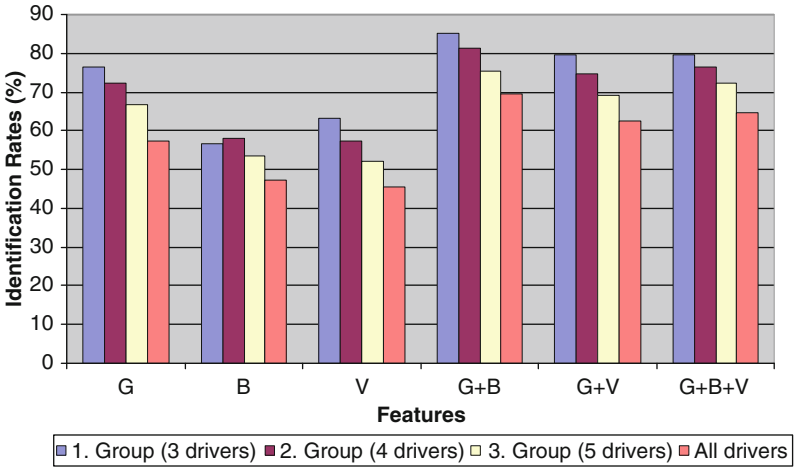



**Fig. 3.10** Comparison of driver identification rates for different group of drivers

We employ the driver identification task for each subgroup by using a fivefold cross validation and evaluate an average driver identification rate for all three groups. Figure 3.10 shows the average identification performances for each group, using different features. We achieve 85.21% of success with the fusion of gas (G) and brake (B) pedal pressure signals among the three drivers.

### *3.4.2 Results on Driving Status Identification*

In this section, we investigate the influence of distractive conditions on driving performance and develop a technique for quantifying driver stress levels under various conditions with different tasks. In the UYANIK database, nearly half of the driving sessions include driving under specific tasks. Driving tasks include dialog on cell phone, dialog with passenger, and signboard reading, which are expected to cause lack of cognitive engagement. The details of the tasks are described as follows:

- *No-Task*: A driver drives without any task.
- *Signboard Reading*: A driver reads aloud the words on signboards/plates during driving.
- *Dialoge on Cell phone*: A driver heads to an unfamiliar place while being guided by a navigator over cell phones. Also, online banking application is done by cell phone.
- *Conversation with Passenger*: A driver talks with the on-board passenger.

In order to investigate the use of driving behavior signals to classify different driving tasks, we build a driving task identification system and perform identification performance analysis over the U-TASK database. For task identification, the

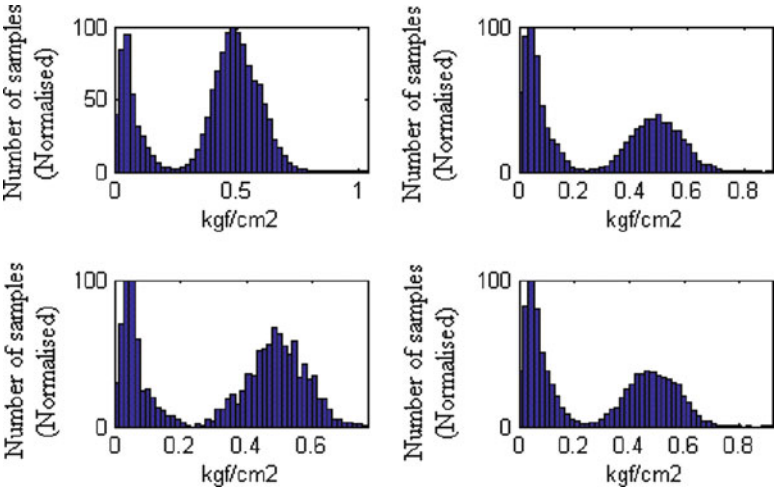



**Fig. 3.11** Histograms of gas pedal pressure signals under reference driving (top left), dialog on cell phone (top right), signboard reading (bottom left), and dialog with passenger (bottom right)

cepstral features are calculated for every 1-s frame. Figure 3.11 shows the histograms of the gas pedal pressure signal under different driving task conditions. Statistical differences between reference driving and driving under a task are observable. However, driving-task-specific histograms, especially dialog on cell phone and with passenger, are close to each other.

We first consider a two-class classification system to identify reference driving and driving with a task. The two-class identification system is expected to show whether distractive conditions are influential on driving performance. Among U-TASK dataset reference, driving lasts 190 min (47.8% of all data) and driving under a task lasts 207.5 min (52.2% of all data) totally. To evaluate the task identification, we use 16-mixture Gaussian classifiers and fivefold cross validation for classification.

The average identification rates of the classifiers using gas and brake pedal pressure signals, and their decision fusion with different decision-window sizes are given in Fig. 3.12. The best scenario is achieved by using 360 s of decision windows.

Table 3.1 shows the identification rates of each class for this scenario. In this table, the last column presents the prior reference distribution of events in the database. We identify a reference driving session with 93.2% of success and identify a driving session under a specific task with 72.5% of success by using the fusion of gas and brake pedal signals. The average task vs. no-task identification result is obtained as 83.3% with the fusion of 16-mixture GMM classifiers of gas and brake signals. Note that, these identification rates are significantly higher than random classifier performances with possible uniform distributions.

We also consider identification of individual tasks from driving behavior signals. Among all driving sessions under a specific task, dialog on cell phone lasts 97.5 min

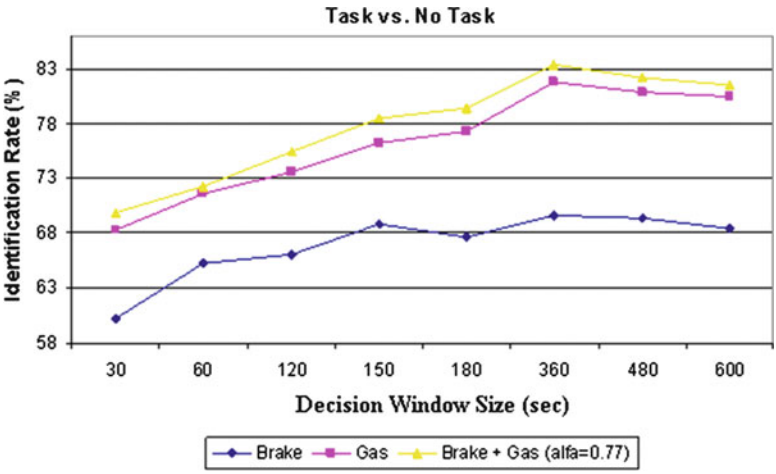



**Fig. 3.12** Average task identification rates of the classifiers using gas and brake pedal pressure signals and their decision fusion with different decision-window sizes

**Table 3.1** Task vs. no-task identification rates (%) of 16 mixture GMM classifiers with 360 s decision window for gas (G), brake (B), gas and brake (G + B) fusion, and reference random (R) classifier

| | G | B | G + B | R |
|---|---|---|---|---|
| No-Task | 91.1 | 76.6 | 93.2 | 52.2 |
| Task | 71.6 | 61.9 | 72.5 | 47.8 |
| Avg. | 81.8 | 69.6 | 83.3 | 50.1 |

**Table 3.2** Task identification rates (%) of 16 mixture GMM classifiers with 60 s decision window for gas (G), brake (B), gas and brake (G + B) fusion and reference random (R) classifier

| | G | B | G + B | R |
|---|---|---|---|---|
| Dialog on cell phone | 56.4 | 49.7 | 58.5 | 47.6 |
| Signboard reading | 17.5 | 32.5 | 25.0 | 9.7 |
| Conversation with passenger | 50.3 | 44.1 | 52.6 | 42.7 |
| Avg. | 50.0 | 45.7 | 52.7 | 41.8 |

(47.56% of all driving with a task data), conversation with passenger lasts 87.5 min (42.68% of all driving with a task data), and signboard and license plate reading lasts 20 min (9.76% of all driving with a task data) totally. To evaluate task identification, we use 16-mixture Gaussian classifiers and fivefold cross validation for classification.

Table 3.2 shows the identification rates of each class for the best scenario. Dialog on cell phone task is identified with 58.5%, signboard reading with 25%, and conversation with passenger with 52.6% of success by using the fusion of gas and brake pedal signals. The average task identification result is obtained as 52.7%

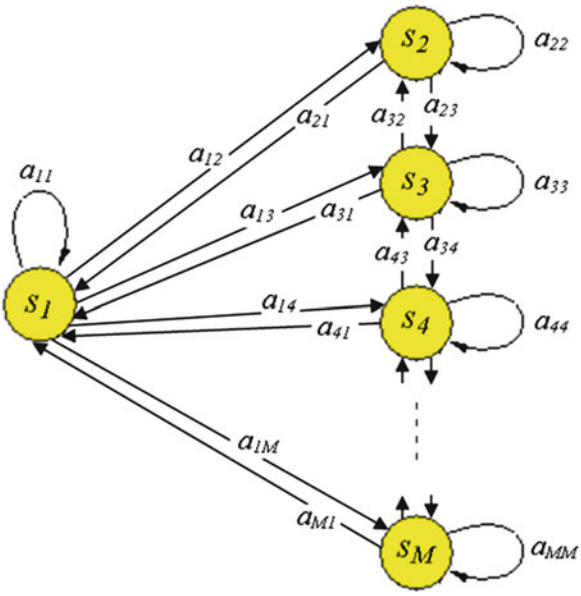



**Fig. 3.13** HMM structure for temporal clustering

with 60 s of decision windows and the fusion of 16-mixture GMM classifiers of gas and brake signals. Identification rates are observed to be higher than random for all task classes.

### *3.4.3 Results on Driving Behavior Prediction*

We use the U-DRIVER database to evaluate driver behavior prediction. Also, we analyze the effects of cognitive distraction conditions on predicting driver behavior. For this purpose, we use the transcribed U-TASK database. In driving behavior prediction, we use a fourfold cross validation for all estimation experiments. The four partitions in the fourfold cross validation are made up of equally numbered segments constructed by the HMM clustering. Since the lengths of these segments are not equal, the ratio of test/training data over time is different for all drivers.

In each temporal segment, which is constructed by HMM clustering, we perform an MMSE estimate of the current driving behavior sample from $N$ recent driving behavior samples. The optimal number of recent samples for the estimation is found experimentally as six samples from velocity signal, one sample from gas signal, and one sample from brake signal. Then we investigate the effect of the number of states for HMM clustering. Figure 3.13 plots how the prediction error changes as a function of number of states in the HMM structure for training and test data. The three-state HMM structure is chosen as an adequate model for the classification of driving behavior signals.

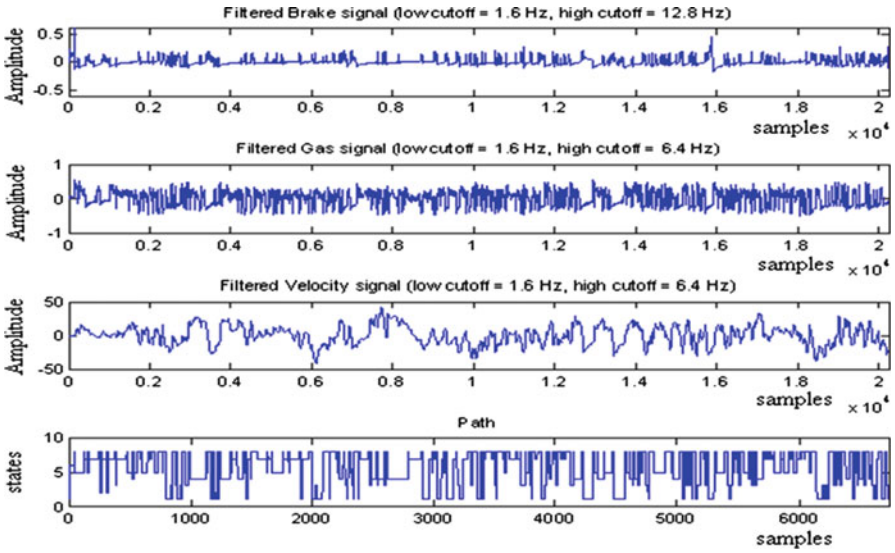



**Fig. 3.14** State sequence of a test data

The MMSE estimation for each temporal segment is performed under two different scenarios. In the first scenario, we use direct samples of behavior signals that are actual recent readings from sensors, at each estimation step. In the second scenario, within each temporal cluster we start using recent actual samples but continue with the estimated samples to estimate upcoming samples. Hence, in the second scenario, estimation looks forward based on the current actual signal readings. The second scenario creates a more realistic system to foresee expected driving behavior characteristics.

The driving behavior signals are predicted using a window of past behavior samples. All behavior signals are decimated by four for the driving behavior prediction experiment. The cepstral features for the HMM clustering are extracted over 800 ms windows (25 samples) for every 96 ms frames (three samples). First, we build a temporal correlation between all three signals using HMM structure shown in Fig. 3.14. This structure is specified by the following parameters:

- Set of discrete states $S = \{S_i, i = 1, 2, \ldots, M\}$
- State transition probability $a_{ij}$, $i = 1, 2, \ldots, M$; $j = 1,2,\ldots,M$

where $M$ denotes the number of states. Transition probabilities from one state to another are set equal initially, and system starts with probability 1 at the first state. The HMM model is then trained using EM algorithm with the training data. In the testing phase, the Viterbi decoding algorithm determines the state sequence of the test data. A sample state sequence using eight-state HMM clustering is given in Fig. 3.15.

Figures 3.16 and 3.17, respectively, present samples of driving behavior signal prediction based on the first and second scenarios for a randomly selected driver.


| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 11 | 13 |
|---|---|---|---|---|---|---|---|---|---|---|
| Brake | −5,45 | −27,54 | −42,7 | −40,3 | −43,2 | −42,35 | −42,68 | −39,94 | −43,55 | −38,56 |
| Gas | 0,12 | −22,45 | −35,6 | −33,65 | −34,66 | −31,75 | −38,55 | −35,32 | −38,1 | −33 |
| Velocity | 55,76 | 24,6 | 5,45 | 8,34 | 9,98 | 10,33 | 12,41 | 10,3 | 7,43 | 10,34 |

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 11 | 13 |
|---|---|---|---|---|---|---|---|---|---|---|
| Brake | 5,5 | −0,34 | −21,8 | −21,82 | −22,68 | −23,45 | −23,1 | −23,82 | −23,5 | −22,5 |
| Gas | 8,89 | 2,99 | −11,57 | −12,98 | −10,44 | −11,5 | −14,34 | −11,27 | −14,5 | −6,87 |
| Velocity | 68,84 | 55,23 | 25,05 | 24,6 | 26,02 | 22,1 | 28,28 | 28,58 | 24,22 | 24,5 |

**Fig. 3.15** Driving behavior prediction error plots for (**a**) training and (**b**) test data

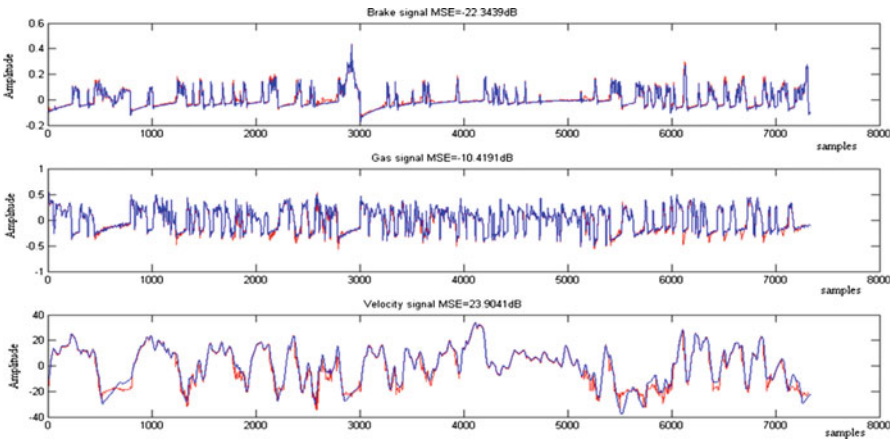



**Fig. 3.16** Driving behavior prediction for the first scenario

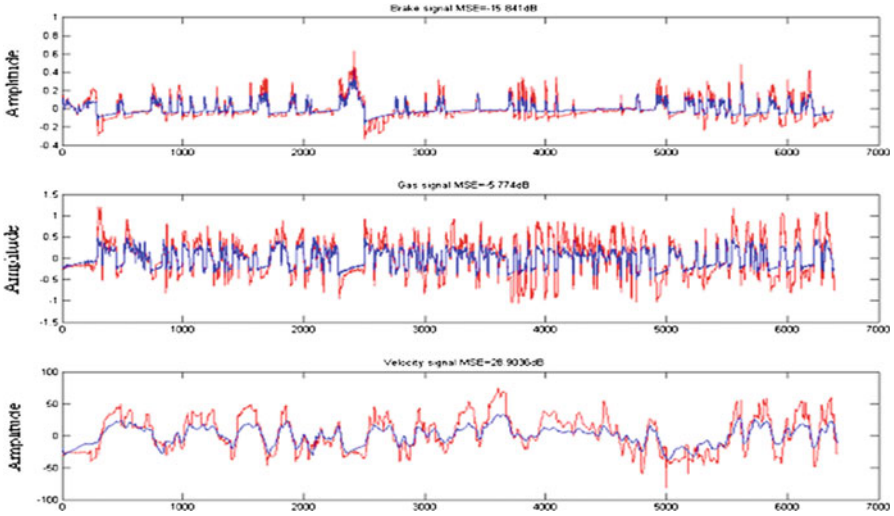
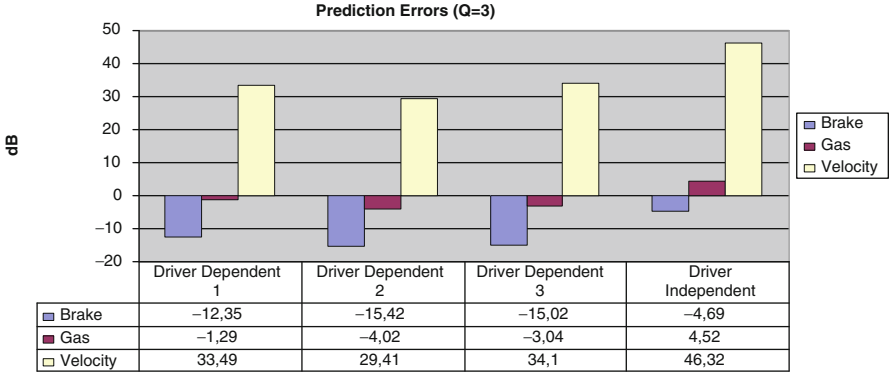



**Fig. 3.17** Driving behavior prediction for the second scenario




| | Driver Dependent 1 | Driver Dependent 2 | Driver Dependent 3 | Driver Independent |
|---|---|---|---|---|
| Brake | −12,35 | −15,42 | −15,02 | −4,69 |
| Gas | −1,29 | −4,02 | −3,04 | 4,52 |
| Velocity | 33,49 | 29,41 | 34,1 | 46,32 |

**Fig. 3.18** Comparison of driver-independent and dependent-driving behavior prediction errors

In these figures, signal plotted in blue (darker) represents the actual signal, and the red (lighter) one represents the estimated signal.

We also perform driver-independent experiments for the driving behavior prediction problem. From the database, we select 20 drivers for training and the remaining three drivers for testing. Test data for each three drivers is the same with the one that we used in driver-dependent experiment. In both driver-dependent and independent experiments, classifiers are maintained with the same parametric settings. We calculate the average prediction error for each test driver within the second estimation scenario. Figure 3.18 plots the prediction errors for driver-independent experiment

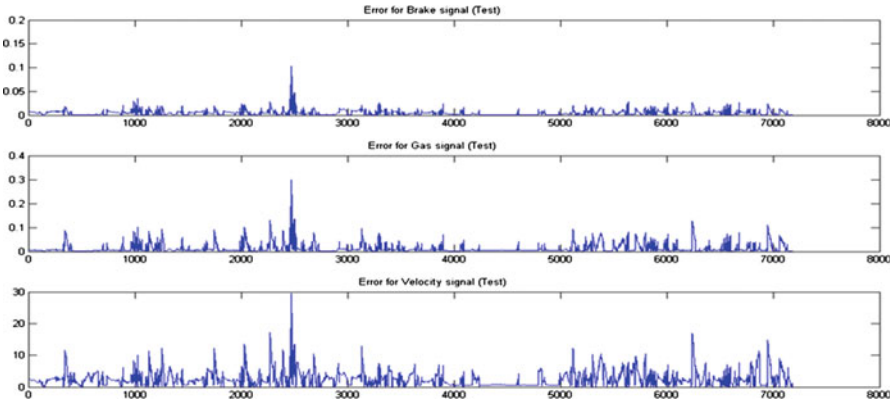



**Fig. 3.19** Prediction errors for one driver's (brake, gas, velocity) behavior signals

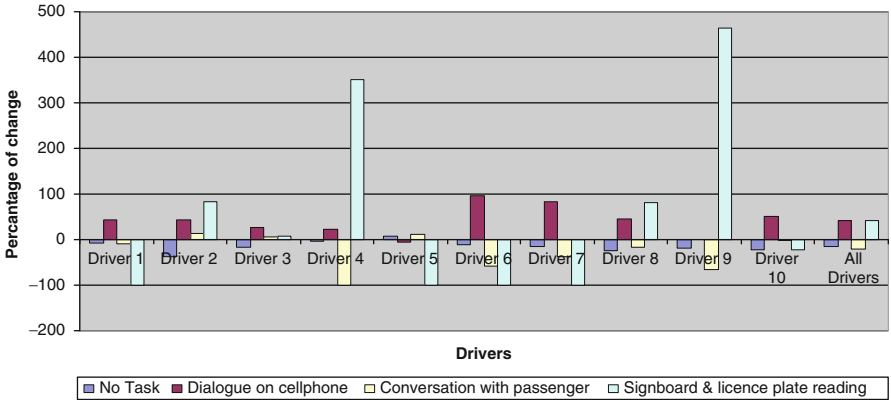



**Fig. 3.20** Percentage of change of driving task lengths over erroneous parts for each driver to entire sessions

with comparison to driver-dependent one. As expected, driver behavior prediction in driver-independent experiments is more difficult than driver-dependent experiments.

Figure 3.19 plots a set of sample prediction error signal for a randomly selected driver. In Fig. 3.19 some segments contain high level of prediction errors. To investigate the driving conditions where these errors appear, it is necessary to determine and transcribe high erroneous parts clearly. We select the 20% of the highest prediction error as the threshold value and define the segments higher than this threshold as high erroneous parts. The correlation between erroneous parts and the driving conditions can yield important findings. Hence, we calculate the ratio of change of driving task and road type durations over erroneous parts to the whole durations. The percentage of change is shown in Figs. 3.20 and 3.21 for driving tasks and road conditions, respectively.

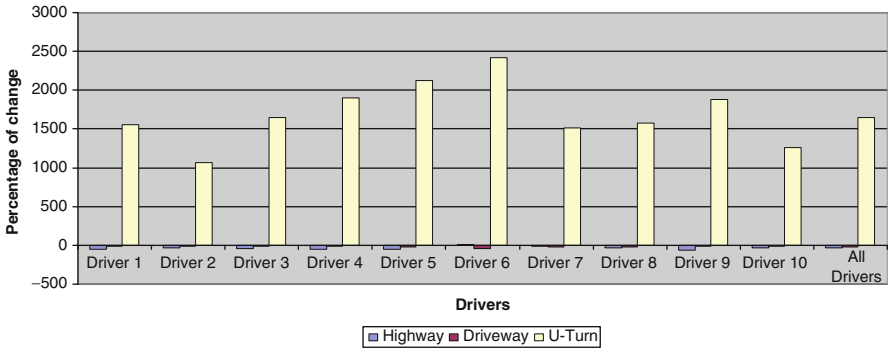



**Fig. 3.21** Percentage of change of driving session lengths on different types of roads over erroneous parts for each driver to entire sessions

In Fig. 3.20, we can observe that the prediction of behavior signals under distractive conditions is more erroneous than prediction under no secondary task.

The experiments show that the ratio of all task lengths over erroneous parts is higher than the ratio of all task lengths over all segments. Hence, distractive conditions have a certain observable effect on driving behavior. Figure 3.20 also shows that among driving tasks, the dialog on cell phone is a more effective indicator of driving behavior than the other tasks. Figure 3.21 shows that road conditions are also effective on predicting driver behavior. It is hard to predict driver behavior on U-Turns, where the highway connects to driveway. Sudden maneuvers and unsteady use of pedal operations creates high prediction errors at U-Turns.

## 3.5 Conclusions

In this study, we consider the problem of driver and driver status identification under different cognitive stress/distraction conditions. Also, we try to predict driving behavior signals by using the past movements of the drivers. Our objective is to construct a system to facilitate driver–vehicle interaction by analyzing driving behaviors. To study and determine the nature of driving behavior, we benefit from the characteristic driving signals including brake pedal pressure, gas pedal pressure, vehicle velocity, and the distance from the vehicle in front signals. The system is expected to be more reliable due to the availability of sufficient amounts of driving behavior signals.

In driver identification experiments, test results show that the decision fusion method significantly increases our system performance. We achieve 69.5% of success with the fusion of gas and brake pedal pressure signals, while these signals can reach up to 58% of success at most, individually, among 23 drivers. Driver identification results of the car-following task is lower than the pedal operation models; however, it is feasible to use them to recognize a driver. We also investigate the driver

identification problem for a reduced number of drivers to achieve more realistic results. The best identification result is obtained as 85.21% with the fusion of gas and brake pedal pressure signals among the three drivers.

Distraction detection is an important issue because cognitive/stress conditions have a great influence on driving behavior. We achieve 93.2% of success in detecting driver behavior signals under no specific task while the random rate is about 52% for ten drivers. In our database, nearly half of the driving sessions are done under specific task. Among these tasks, dialog on mobile phone, conversation with passenger on-board, sign reading, and license plate reading are the most effective ones.

Warning drivers about future incidents is an important application area because many of the traffic accidents are caused by drivers. In this study, we propose a method of predicting driving behavior based on gas pedal pressure, brake pedal pressure, and vehicle velocity signals. Predicting driving behavior signal using past samples yields encouraging results. We performed driver-dependent and independent driving behavior prediction experiments. Although prediction error profiles for the driver-independent experiment are higher than the driver-dependent experiment, driver-independent driving behavior prediction attains sufficiently low error rates. Distractive conditions are expected to have a great influence on driving behavior. Our driving behavior prediction results are also supporting this finding. Prediction of driving behavior signals under distractive conditions is 20% more erroneous than prediction under no secondary task.

**Acknowledgments** This work has been supported by TUBITAK under project EEEAG-104E176 and by the state planning organization of Turkey (DPT) under Drive-Safe project.

## References

1. Kurahashi T, Ishibashi M, Akamatsu M (2003) Objective measures to assess workload for car driving. In: Proceedings of the SICE 2003 Annual Conference, vol 1, pp 270–275
2. Miyajima C, Kusakawa T, Nishino T, Kitaoka N, Itou K, Takeda K (2008) On-going data collection of driving behavior signals. In: Takeda K, Erdogan H, Hansen JHL (eds) Corpus and signal processing for driver behavior. Springer Business-Science, New York
3. Wakita T, Ozawa K, Miyajima C, Takeda K (2005) Parametric versus non-parametric models of driving behavior signals for driver identification. In: Kanade T, Jain AK, Ratha NK (eds) AVBPA, vol 3546, Lecture Notes in Computer Science. Springer, New York, pp 739–747
4. Miyajima C, Nishiwaki Y, Ozawa K, Wakita T, Itou K, Takeda K, Itakura F (2007) In: Proceedings of the IEEE, driver modeling based on driving behavior and its evaluation in driver identification, vol 95, no 2, pp 427–437
5. Miyaji M, Danno M, Oguri K (2008) Analysis of driver behavior based on traffic incidents for driver monitor systems. In: Proceedings of the IEEE intelligent vehicles symposium, Eindhoven, pp 930–935
6. Erzin E, Yemez Y, Tekalp AM, Ercil A, Erdogan H, Abut H (2006) Multimodal person recognition for human–vehicle interaction. IEEE Multimedia 13(2):18–31
7. Ma X, Jansson M (2007) Model estimation for car-following dynamics based on adaptive filtering approach. In: Proceedings of the IEEE intelligent transportation systems conference ITSC 2007, Seattle, pp 824–829

8. Tezuka S, Soma H, Tanifuji K (2006) A study of driver behavior inference model at time of lane change using Bayesian networks. In: Proceedings of the IEEE international conference on industrial technology ICIT 2006, Mumbai, pp 2308–2313
9. Boril H, Boyraz P, Hansen JHL (2009) Towards multi-modal driver's stress detection. In: Proceedings of the 4th biennial workshop on DSP for in-vehicle systems and safety, Dallas
10. Marinova M, Devereaux J, Hansman RJ (2007) Experimental studies of driver cognitive distraction caused by cell phone use. Transportation Research part F, MIT Cambridge, MA
11. Patten CJD, Kircher A, Östlund J, Nilsson L, Svenson O (2006) Driver experience and cognitive workload in different traffic environments. Accident Anal Prev 38(5):887–894
12. Angkititrakul P, Hansen JHL (2008) UTDrive: The smart vehicle project. In: Takeda K, Hansen JHL, Erdogan H, Abut H (eds) Corpus and signal processing for driver behavior. Springer Business-Science, New York
13. Abut H, Erdogan H, Ercil A et al (2008) Data collection with UYANIK: too much pain; but gains are coming. In: Takeda K, Hansen JHL, Erdogan H, Abut H (eds) Corpus and signal processing for driver behavior. Springer Business-Science, New York
14. Kishimoto Y, Oguri K (2008) A modeling method for predicting driving behavior concerning with drivers past movements, Columbus
15. Erzin E (September 2009) Improving throat microphone speech recognition by joint analysis of throat and acoustic microphone recordings. IEEE Trans Audio, Speech Language Processing 17(7):1316–1324
16. Wakita T, Ozawa K, Miyajima C, Igarashi K, Itou K, Takeda K, Itakura F (2006) Driver identification using driving behavior signals. IEICE Trans Information Sys E89(3):1188–1194
17. Ozturk E, Erzin E (2009) Driving status identification under different distraction conditions from driving behaviour signals. In: Proceedings of the 4th biennial workshop on DSP for in-vehicle systems and safety, June 2009, UTD, TX, pp 25–27

# Chapter 4
# Multilayer Modeling of Driver Behavior Based on Hierarchical Mode Segmentation

**Hiroyuki Okuda, Ato Nakano, Tatsuya Suzuki, Soichiro Hayakawa, and Shinkichi Inagaki**

**Abstract** This chapter presents a new hierarchical mode segmentation of the observed driving behavioral data based on the multiple levels of abstraction of the underlying dynamics. By synthesizing the ideas of a feature vector definition revealing the dynamical characteristics and an unsupervised clustering technique, the hierarchical mode segmentation is achieved. The identified mode can be regarded as a kind of symbol in the abstract model of the behavior. Some applications of the proposed model are also discussed.

**Keywords** Driver behavior • Formal grammar • Hierarchical clustering • Hybrid system

## 4.1 Introduction

Recently, several ideas about driver modeling from the viewpoint of control technology and information processing have been explored. The common goal of which is to attain driving safety and develop human-friendly cars [1–4].

In studies about driving behavior, it is often found that a driver appropriately switches from complex nonlinear control law to simple control laws. This idea can be verified by executing "mode segmentation" of the observed driving data. Mode segmentation is based on the classification of a behavioral data's dynamical characteristics [5–7]. Assigning each obtained mode to each symbol can be regarded as one of the solutions for "symbolic grounding" problem. Furthermore, the transition between modes can be viewed as a form of driver decision making involving

H. Okuda (✉) • A. Nakano • T. Suzuki • S. Inagaki
Nagoya University, Nagoya, Japan
e-mail: h_okuda@nuem.nagoya-u.ac.jp; t_suzuki@nuem.nagoya-u.ac.jp

S. Hayakawa
Mie University, Tsu City, Japan

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_4, 

complex driving tasks [7]. Thus, the introduction of the mode segmentation leads to higher level of understanding of driving behavior whereby motion control and decision-making aspects are synthesized.

Another important characteristic of driving behavior is described by its hierarchical structure. Many behaviors can be understood using hierarchical modeling or characterization based on different levels of abstraction of dynamics. From this perspective, it is quite natural to introduce the "hierarchical mode segmentation" in analyzing human behavior. As a consequence, the hierarchical symbolization of human behavior can be realized solely on observed behavioral data (without any prior knowledge). The hierarchical symbolization is expected to play an essential role in the design of intelligent human support system; thanks to its high describability and understandability of complex behavior.

Based on abovementioned considerations, we propose a new hierarchical mode segmentation of the observed driving behavioral data based on multiple levels of abstraction of the underlying dynamics. In order to realize this idea, a PieceWise AutoRegressive eXogenous (PWARX) model is being introduced. This approach is often used as the identification model of hybrid dynamical systems [8,9] wherein each ARX model represents the corresponding dynamics of each mode. In our problem setting, the number of modes (number of symbols) is supposed to be controllable in order to obtain the hierarchical structure. Of course, the number of modes is assumed to be fixed in the standard framework of hybrid system identification. By synthesizing the ideas of definition of the feature vector revealing the dynamical characteristics [8] and an unsupervised clustering technique, the hierarchical mode segmentation is achieved.

All in all, the usefulness of both the hierarchical mode segmentation and symbolized human behavior from the viewpoint of the symbolic grounding are demonstrated by applying them on highway-driving behavioral data.

## 4.2 Hierarchical Mode Segmentation

In this section, we discuss how to define "mode" in driving behavioral data and how to obtain the hierarchical structure. We begin by defining driver input and output.

### *4.2.1 Definition of Input and Output*

Throughout this chapter, we focus on the driving behavior on the highway which consists of "following the leading vehicle," "lane changing," "overtaking," and so on. The driver input, i.e., the sensory information of the driver, is defined as follows (Fig. 4.1):

- Range from the leading car: $u_1$
- Range rate between the leading and examinee's cars: $u_2$

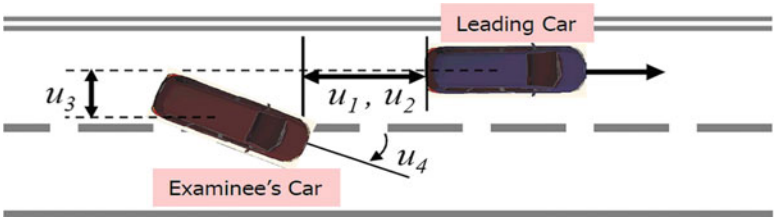



**Fig. 4.1** Definition of input signals

- Lateral displacement from the leading car: $u_3$
- Yawing angle of examinee's car: $u_4$
- Index for approaching (KdB): $u_5$
- Amount of time duration that the examinee looks at the left side mirror in the latest 10 s (*TL*): $u_6$
- Amount of time duration that the examinee looks at the right side mirror in the latest 10 s (*TR*): $u_7$

*KdB* is an index which represents the logarithm of a time derivative of the area behind the leading car as projected on a driver's retina [10]. The *KdB* can be expressed by using $u_1$ and $u_2$ as follows:

$$KdB = \begin{cases} -10 \times \log(|-2 \times \frac{u_2}{u_1} \times \frac{1}{5\times10^{-8}}|) & \text{if } u_2 > 0 \\ 10 \times \log(|-2 \times \frac{u_2}{u_1} \times \frac{1}{5\times10^{-8}}|) & \text{if } u_2 < 0 \end{cases} \tag{4.1}$$

The large *KdB* implies that the driver is facing dangerous situation. Also, the driver output is defined as follows:

- Steering angle: $y_1$
- Pedal operation: $y_2$

These input and output variables are chosen so that the resulting model can express the behavioral characteristics underlying the observed data. Furthermore, they can be observed in real driving situations using existing sensors.

### 4.2.2 PWARX Model as Mathematical Representation of Multimode Driving Behavior

In this subsection, the PWARX model is introduced as a mathematical model of driving behavior. It consists of the several ARX models, i.e., modes, and can appropriately control the number of modes. We consider the following first-order PWARX model which has $s$ modes:

$$y(k) = f(r(k)) + \varepsilon(k) \tag{4.2}$$

$$f(r(k)) = \begin{cases} \theta_1 r(k) \;\; if\; r(k) \in R_1 \\ \theta_2 r(k) \;\; if\; r(k) \in R_2 \\ \qquad \vdots \\ \theta_s r(k) \;\; if\; r(k) \in R_s \end{cases} \tag{4.3}$$

where $y(k)$ and $r(k)$ are defined as follows:

$$y(k) = (y_1(k)\, y_2(k))^T \tag{4.4}$$

$$r(k) = (u_1(k-1)\; u_2(k-1) \cdots \\ u_7(k-1)\; y_1(k-1)\; y_2(k-1))^T \tag{4.5}$$

The subscript $k$ denotes the sampling index (k = 1, 2, …, $n$). Furthermore, $\theta_i$ ($i$ = 1, 2, …, $s$) is a (2 × 9) unknown matrix to be identified from the data and is supposed to have a form:

$$\theta_i = \begin{pmatrix} {\theta_i}_{,1}^T \\ {\theta_i}_{,2}^T \end{pmatrix} \tag{4.6}$$

In the PWARX model, not only parameters $\theta_i$ but also the partitions of the subspaces $R_1$, …, $R_s$ are unknown. Therefore, it is not straightforward to assign each observation ( $y(k)$, $r(k)$) at sampling instant $k$ to the corresponding mode. To resolve this problem, a clustering-based technique is developed in [8] under the definition of interesting feature vector which represents the local dynamical characteristics underlying ($y(k)$, $r(k)$). In the next subsection, this feature vector is introduced.

### 4.2.3 Definition of Input and Output

1. Assume that the set of sample data $\{(y(j), r(j))\}(j = 1, 2, \ldots, n)$ is given. For each sample data ($y(j)$, $r(j)$), collect the neighboring $c$ data in the ($y$, $r$) space, generate the local data set $LDs_j$, and calculate the feature vector $\xi_j$ (Fig. 4.2). Note that the index $j$ indicates the order not in the time space but in the data space. The feature vector $\xi_j$ consists of the local parameters $\left((\theta_1^{LDs_j})^T, (\theta_2^{LDs_j})^T\right)^T$ in the local ARX model for the $LDs_j$ and the mean value $m_j$ of the data $r$ in the $LDs_j$. $(\theta_l^{LDs_j})^T$ ($l$ = 1, 2) and $m_j$ are calculated as follows:

$$\theta_l^{LDs_j} = (\Phi_j{}^T \Phi_j)^{-1} \Phi_j{}^T y_l^{LDs_j} \tag{4.7}$$

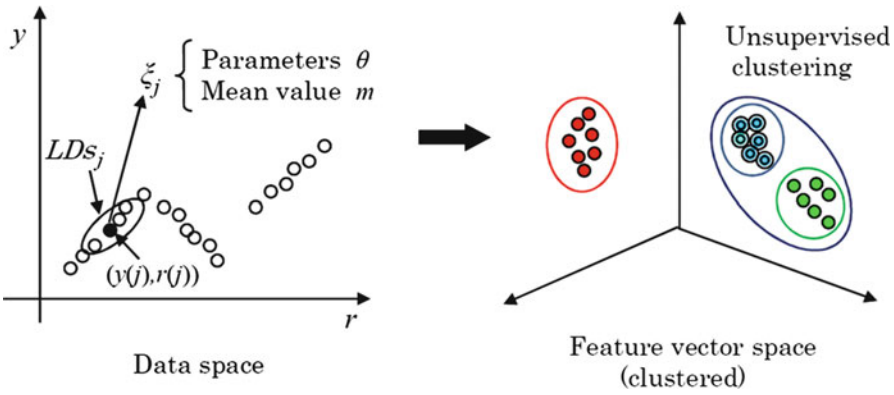



**Fig. 4.2** Transformation from data space to feature vector space

where $y_l^{LDs_j}$ $(c \times 1; l = 1, 2)$ is the output samples in the $LDs_j$, and $\Phi_j$ is given by

$$\Phi_j = (r_1\, r_2\, \cdots\, r_c)^T\, (r \in LDs_j). \tag{4.8}$$

As the result,

$$\xi_j = ((\theta^{LD}{}_{j,1})^T,\ (\theta^{LD}{}_{j,2})^T,\ m_j{}^T)^T. \tag{4.9}$$

2. For each feature vector $\xi_j$, the following covariance matrix $R_j$ is calculated:

$$R_j = \begin{pmatrix} V_{j,1} & 0 & 0 \\ 0 & V_{j,2} & 0 \\ 0 & 0 & Q_j \end{pmatrix} \tag{4.10}$$

where

$$V_{j,l} = \frac{SSR_{j,l}}{c - (9+1)} (\Phi_j{}^T \Phi_j)^{-1} \tag{4.11}$$

$$SSR_{j,l} = y_l^{LDs_j T} (I - \Phi_j (\Phi_j{}^T \Phi_j)^{-1} \Phi_j{}^T) y_l^{LDs_j} \tag{4.12}$$

$$Q_j = \sum_{r \in LDs_j} (r - m_j)(r - m_j)^T. \tag{4.13}$$

The feature vector $\xi_j$ represents the combination of the local dynamics and data. By this definition, the data is classified based not only on the value of data but also on the similarity of the underlying dynamics. Furthermore, the covariance matrix $R_j$ represents the confidence level of the corresponding feature vector $\xi_j$. $R_j$ is used as the weighting matrix in the calculation of the dissimilarity between feature vectors in the clustering procedure.

### 4.2.4 Clustering Procedure

The unsupervised hierarchical clustering is applied to the feature vectors $\xi_j$ ($j = 1, 2, \ldots, n$). The clustering algorithm is listed below:

3. Regard each feature vector $\xi_j$ as each cluster $C_j$, i.e., each cluster consists only of one feature vector. Calculate the dissimilarity $D_{p,q}$ between any two clusters $C_p$ and $C_q$ by using the following dissimilarity measure:

$$D_{p,q} = \left\|\xi_p - \xi_q\right\|^2_{R_{p,q}{}^{-1}} = (\xi_p - \xi_q)^T R_{p,q}{}^{-1}(\xi_p - \xi_q) \tag{4.14}$$

   Where

$$R_{p,q}{}^{-1} = R_p{}^{-1} + R_q{}^{-1}_{.} \tag{4.15}$$

4. Unify two clusters $C_x$ and $C_y$ which shows the smallest $D_{x,y}$. The unified cluster is denoted by $C_r$. If all clusters are unified, terminate the algorithm. Otherwise, go to step 3.
5. Calculate the dissimilarity $D_{x,y}$ between $C_r$ and $C_t$ for all $t(t \neq r)$ by using the following dissimilarity measure:

$$D_{r,t} = \frac{n_r n_t}{n_r + n_t} \sum_{\xi_{i_r} \in C_r} \sum_{\xi_{i_t} \in C_t} \left\|\xi_{i_r} - \xi_{i_t}\right\|^2_{R_{i_r,i_t}{}^{-1}}. \tag{4.16}$$

6. Where $n_r$ and $n_t$ are numbers of feature vectors belonging to clusters $C_r$ and $C_t$, respectively. Go to step 2.

After this clustering procedure, the classification of the feature vector space is achieved together with a dendrogram which shows the hierarchical classification for different number of modes. Since the transformation from the feature vector ($\xi$) space to the original observed data ($y$, $r$) space is straightforward, the mode segmentation of the observed data is obtained together with the hierarchical structure.

Note that once mode segmentation of the data is achieved, the identification of the parameters $\theta_i$ and the partitions of the subspaces $R_1, \ldots, R_s$ in the PWARX model [2] is straightforward.

## 4.3 Analysis of Driving Behavioral Data

### 4.3.1 Driving Environment

In this chapter, the following driving environment on the highway was designed on the driving simulator which provides a stereoscopic immersive vision [7].

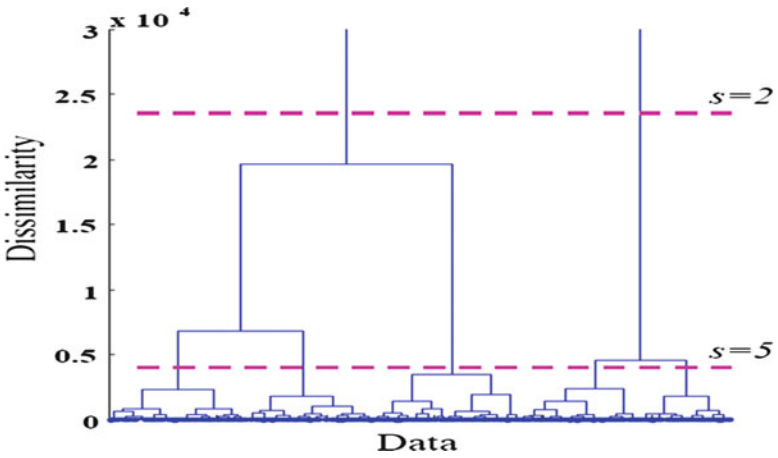



**Fig. 4.3** Dendrogram of clustering (Examinee A)

- The expressway is endless and has two lanes – the cruising lane and passing lane.
- There are ten cars on the cruising lane. Five of them are in front of the examinee's car. The remaining five are behind the examinee's vehicle. Their velocities vary from 70 to 85 km/h. Once the examinee's car overtakes the lead vehicle, then the tail-end car is moved in front of the lead car. The examinee is not aware of the switch.
- There are ten cars on the passing lane. Five of them are ahead of the examinee's car. The remaining five cars are behind the examinee's vehicle. Their velocities vary from 90 to 110 km/h. Once the lead car passes the examinee's car, then the tail end is moved in front of the lead car. The examinee is not aware of this change.
- The range between cars is set at 50–300 m, and there is no collision between cars except the examinee's car.
- The examinee's car change lanes while the other vehicles stay on their lanes.

Five examinees performed the test driving using the driving simulator. Note that the examinees were provided with the instruction "Drive the car according to your usual driving manner." Since this instruction is "broad," the examinees did not concern themselves much with the environmental information. As a result, each examinee drove his/her usual way.

### *4.3.2 Observed Behavioral Data and Clustering Results*

The unsupervised clustering based on the feature vector shown in the previous section has been applied to the observed driving behavioral data. The dendrogram obtained from the proposed strategy is shown in Fig. 4.3 wherein the vertical axis represents the dissimilarity between clusters.

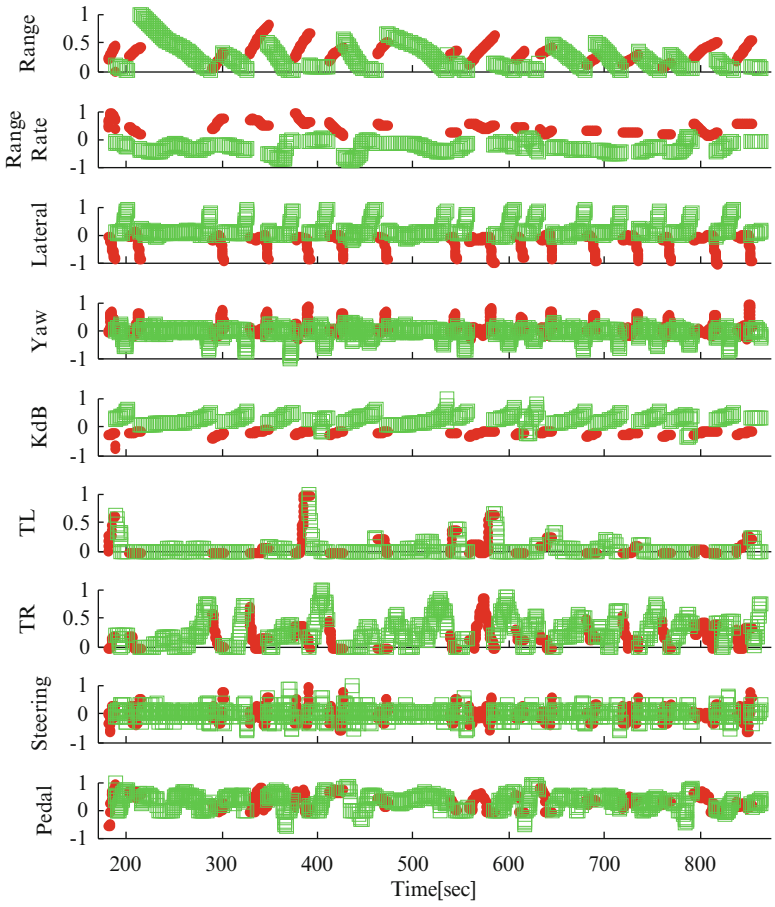



**Fig. 4.4** Observed profiles and mode segmentation result (Examinee A, two modes)

Figure 4.3 shows that when the two clusters are unified, the corresponding dissimilarity is designated by the horizontal bar. The horizontal axis represents the data which has been rearranged after clustering to indicate the hierarchical structure. From this figure, we can clearly understand the hierarchical structure of driving behavior. Two dashed horizontal lines are superimposed. The upper line indicates the number of modes (clusters) $s$, i.e., the number of the ARX models in [2] is set at two. Meanwhile, the lower line shows that $s$ is set at five.

In Figs. 4.4 and 4.5, the observed driving (input–output) profiles of examinee A are shown. All profiles are normalized before clustering. In the profile of the lateral displacement, it takes positive value when the examinee's vehicle is on the right side of the lead car. The steering angle takes positive value when the examinee steers it clockwise. Also, the pedal operation takes positive value when the accelerator is stepped on, and takes negative value when the examinee hits the braking pedal. Note that the range, range rate, and lateral displacement profiles demonstrate discontinuity. Since these variables are defined by relative

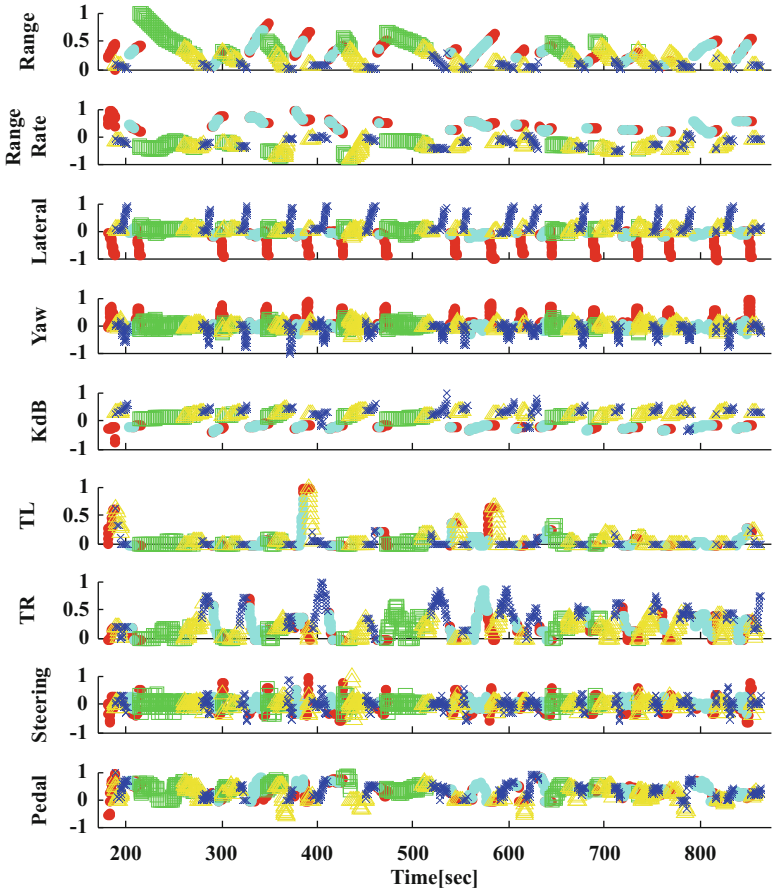



**Fig. 4.5** Observed profiles and mode segmentation result (Examinee A, five modes)

displacement from the lead car, if the examinee's car changes the driving lane, these variables change discontinuously.

The clustering results for the two modes modeling are indicated by colors in Fig. 4.4, while the clustering results in the case of five modes modeling are indicated in Fig. 4.5. Thus, mode segmentation works well. In order to investigate the behavioral meaning of each mode, profiles of the lateral displacement are enlarged in Figs. 4.6(**a**), 4.7(**a**), and 4.8(**a**). In addition, data distributions in range – range rate space – are shown in Figs. 4.9(**a**), 4.10(**a**) and 4.11(**a**). The horizontal axis highlights the range, while the vertical axis features the range rate. It is evident in these figures that the two modes of examinees A and B can be understood as "Following on Cruising Lane + Passing" (Mode 1: FC + P mode) and "Following on Passing Lane + Returning" (Mode 2: FP + R mode), respectively. This result implies that symbolization of behavior can be achieved based on the "dissimilarity" of the underlying dynamics.

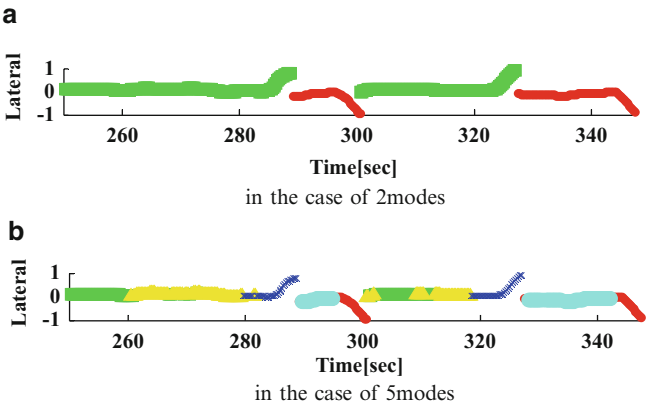



**Fig. 4.6** Enlarged profile of lateral displacement (Examinee A) (**a**) In the case of two modes (**b**) In the case of five modes

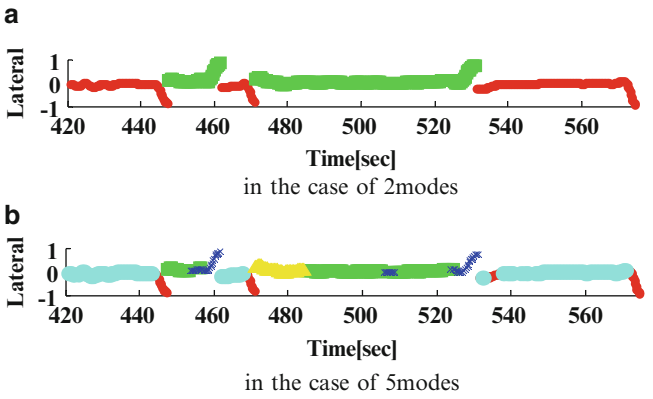



**Fig. 4.7** Enlarged profile of lateral displacement (Examinee B) (**a**) In the case of two modes (**b**) In the case of five modes

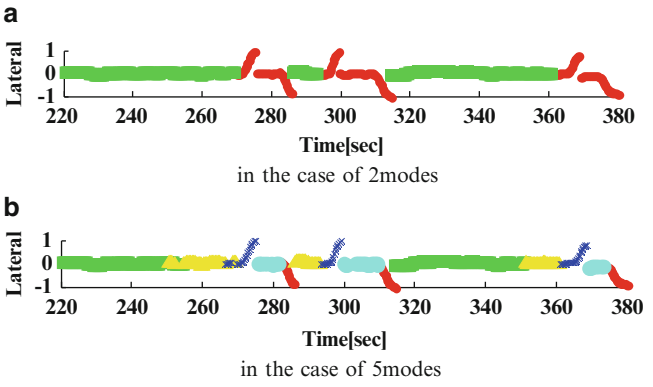



**Fig. 4.8** Enlarged profile of lateral displacement (Examinee C) (**a**) In the case of two modes (**b**) In the case of five modes

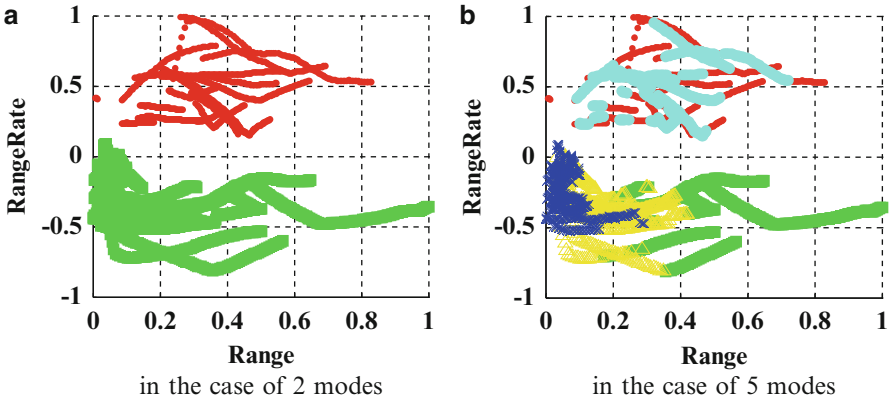



**Fig. 4.9** Observed data distribution and mode segmentation result (Examinee A) (**a**) In the case of two modes (**b**) In the case of five modes

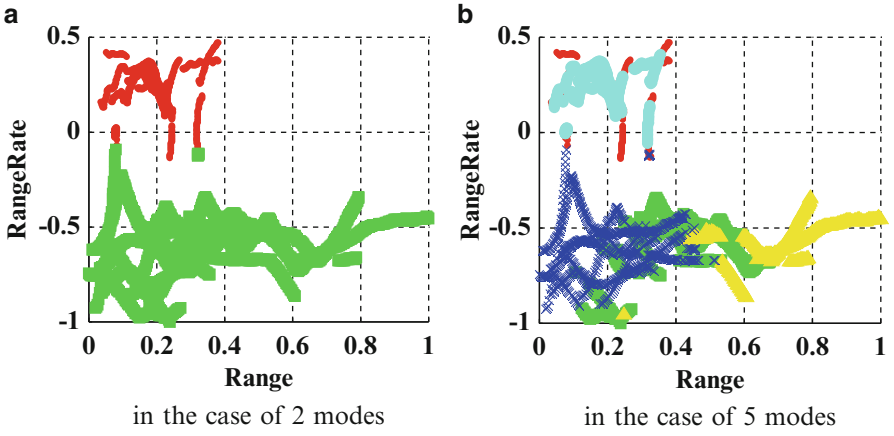



**Fig. 4.10** Observed data distribution and mode segmentation result (Examinee B) (**a**) In the case of two modes (**b**) In the case of five modes

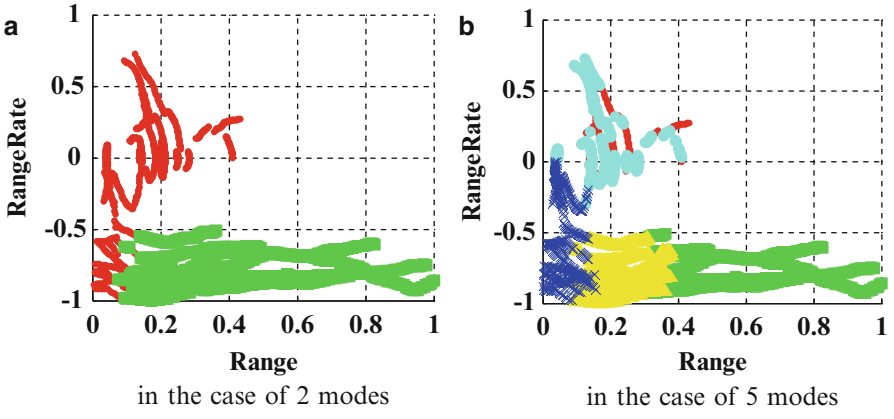



**Fig. 4.11** Observed data distribution and mode segmentation result (Examinee C) (**a**) In the case of two modes (**b**) In the case of five modes

Meanwhile, Fig. 4.8(**a**) shows the different tendency of the mode segmentation in the case of examinee C. The meaning of two modes can be understood as FC and P + FP + R in the case of examinee C, respectively.

## 4.4 Discussion

In order to analyze the hierarchical structure of the behavior, note the clustering results for five-mode modeling of examinee A as shown in Fig. 4.5, the enlarged lateral displacement of three examinees as illustrated in Figs. 4.6(**b**), 4.7(**b**) and 4.8(**b**), as well as the data distributions in range – range rate space as indicated in Figs. 4.9 (**b**), 4.10(**b**) and 4.11(**b**), respectively. From these figures, we can see that the two-mode model is further decomposed into several local behaviors which are:

- "Long Range Following on Cruising Lane" (Mode 1: LRFC mode)
- "Short Range Following on Cruising Lane" (Mode 2: SRFC mode)
- "Passing" (Mode 3: P mode)
- "Following on Passing Lane" (Mode 4: FP mode)
- "Returning" (Mode 5: R mode)

We could find similar symbolization for all examinees. The switching between these modes is caused by the driver's decision making. Figs. 4.12 and 4.13 illustrate the hierarchical relationship between the modes on the dendrogram. Thus, the hierarchical structure of the driving behavior can be obtained in a quite consistent manner. One of the significant contributions of this work is that this hierarchical structure is obtained automatically based only on observation (including the definition of the

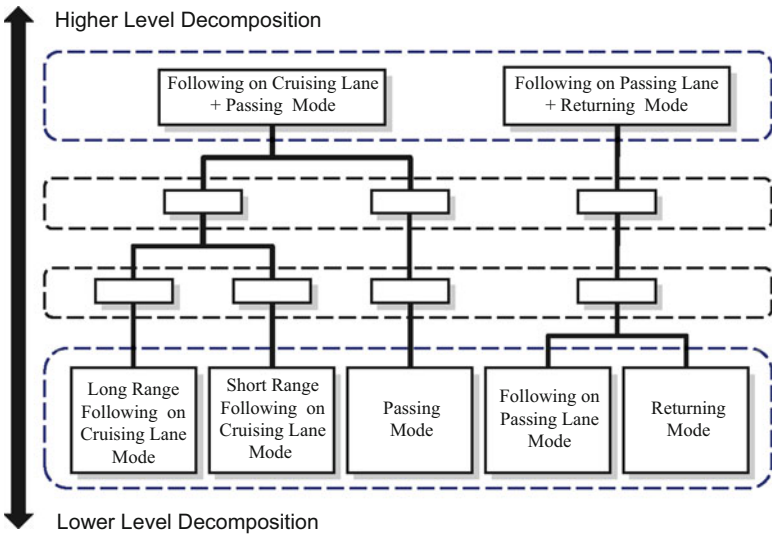



**Fig. 4.12** Identified hierarchical structure of driving behavior (Examinee A and B)

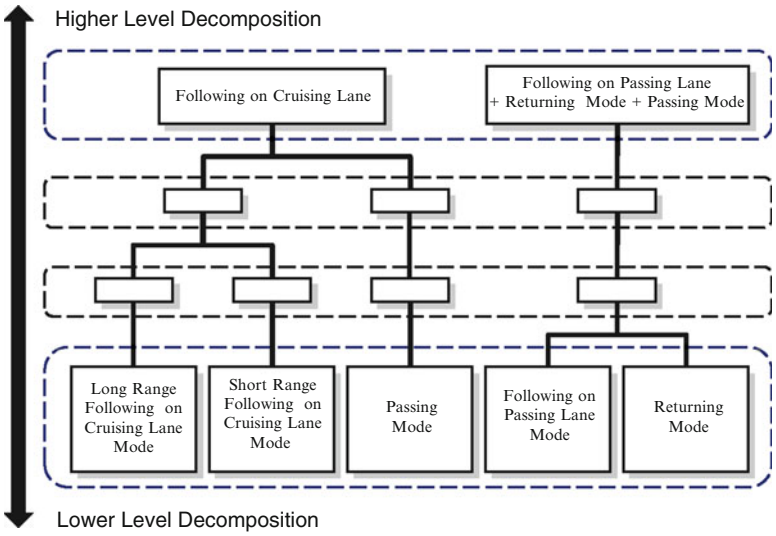



**Fig. 4.13** Identified hierarchical structure of driving behavior (Examinee C)

input and output signals) and data processing. Since this hierarchy clearly expresses multiple abstraction levels of human behavior, the proposed framework is expected to serve as basis for the design of several human-centric systems.

### *4.4.1 Development of Symbolic Behavior Model and its Application to Behavior Prediction*

In this section, some applications using proposed human behavior model is discussed. As already mentioned, this hybrid system modeling is considered as a solution to "symbolic grounding" using symbolization. With this approach, the human behavior can function as an entity (linguistic source) capable of generating a specific language (set of symbol strings). This assumption makes it possible to analyze and model the human behavior along with formal behavioral grammar such as Production Rule. Furthermore, the grammatical modeling of higher human behavior based on hybrid system symbolization will make long-term prediction of human behavior possible.

## 4.5 Conclusions

This chapter has presented a new hierarchical mode segmentation of the observed driving behavioral data based on multiple levels of abstraction of the underlying dynamics. By synthesizing the ideas of the feature vector definition revealing the dynamical characteristics and unsupervised clustering technique, the hierarchical

mode segmentation has been achieved. The identified mode can be regarded as a kind of symbol in the abstract model of the behavior. The proposed framework enables us to make a bridge between signal space and the symbolic space towards understanding human behavior. The construction of higher human behavior model based on some formal grammatical framework and its application towards the prediction of human behavior are our future works.

## References

1. MacAdam C (1981) Application of an optimal preview control for simulation of closed-loop automobile driving. IEEE Trans Syst Man Cybern 11(9):393–399
2. Modjtahedzadeh A, Hess R (1993) A model of driver steering control behavior for use in assessing vehicle handling qualities, ASME. J Dynam Syst Meas Contr 15:456–464
3. Pilutti T, Ulsoy AG (1999) Identification of driver state for lane-keeping tasks. IEEE Trans Syst Man Cybern A 29(5):486–502
4. Nechyba MC, Xu Y (1997) Human control strategy: abstraction, verification, and replication. IEEE Control Syst Mag 17(5):48–61
5. Kim JH, Hayakawa S, Suzuki T, Hayashi K, Okuma S, Tsuchida N, Shimizu M, Kido S (2005) Modeling of driver's collision avoidance maneuver based on controller switching model. IEEE Trans Syst Man Cybern B 35(6):1131–1143
6. Sekizawa S, Inagaki S, Suzuki T, Hayakawa S, Tsuchida N, Tsuda T, Fujinami H (2007) Modeling and recognition of driving behavior based on stochastic switched ARX model. IEEE Trans Intell Trans Syst 8(4):593–606
7. Suzuki T, Akita, Inagaki S, Hayakawa S, Tsuchida N (2008) Modeling and analysis of human behavior based on hybrid system model. In: Proceedings of the international symposium on advanced vehicle control, pp 614–619
8. Ferrari-Trecate G, Muselli M, Liberati D, Morari M (2003) A clustering technique for the identification of piecewise affine system. Automatica 39:205–217
9. Roll T, Bemporad A, Ljung L (2004) Identification of piecewise affine systems via mixed-integer programming. Automatica 40:37–50
10. Wada T, Doi S, .Imai K, Tsuru N, Isaji K, Kaneko H (2007) Analysis of drivers' behaviors in car following based on a performance index for approach and alienation. In: Proceedings of SAE2007 World Congress, SAE technical paper, 2007, doi:10.4271/2007-01-0440

# Part B
# In-Vehicle Interactive/Speech Systems

# Chapter 5
# Evaluation of In-Car Communication Systems

**Gerhard Schmidt, Anne Theiß, Jochen Withopf, and Arthur Wolf**

**Abstract** Due to high background noise and sound absorbing materials, communication between front and rear passengers inside a vehicle is often difficult. In-car communication (ICC) systems distribute the seat-dedicated microphone signals via the car's sound system in order to improve speech intelligibility and communication quality. Due to interfering signals and the closed loop operation of ICC systems, various signal processing techniques are required to reduce feedback, echo, and noise, as well as, to prevent system instability.

In this chapter, a basic overview about the involved signal processing schemes and some ideas for a methodical evaluation of the processing units as well as of the overall ICC system are presented. The evaluation considers different requirements for both talking and listening passengers. The evaluation is performed using four different types of systems and setups. The first one is an *ideal* ICC system. Here, a simulated system without any noise or feedback problems is computed in real time and is presented to listeners and to the measurement equipment. This ideal system is used to obtain upper performance levels or thresholds such as the maximum desired gain. Furthermore, a real ICC system is evaluated in order to analyze the achieved speech intelligibility and system quality. Some measurements are based on the assumption of linear time-invariant systems. This assumption is usually violated by real ICC systems. For that reason, ICC systems should freeze some of their algorithmic components to allow LTI-based measures to obtain suitable results. Usually, this is possible only for research ICC systems but not for commercially available ones. Finally, the measurements should be performed without any ICC system to obtain a basis for comparison.

G. Schmidt (✉) • A. Theiß • J. Withopf
University of Kiel, Kiel, Germany
e-mail: gus@tf.uni-kiel.de; ath@tf.uni-kiel.de; jow@tf.uni-kiel.de

A. Wolf
SVOX Deutschland GmbH, Ulm, Germany
e-mail: arthur.wolf@svox.com

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_5, 

**Keywords** In-car communication • SNR improvement • Speech transmission index • System evaluation

## 5.1 Introduction

When a car is driven at medium or high speed, the communication between passengers in the front and in the rear may be difficult. This is because of low signal-to-noise ratios (SNRs) due to engine, tire, and wind noise which lead to an increased noise level. Another reason is that the received speech level is decreased by sound absorbing interior materials. Furthermore, the driver and front passenger speak toward the windshield (see Fig. 5.1). Thus, their speech is hardly intelligible for the rear passengers in the noisy environment of a car driven at high speed. Figure 5.2 schematically shows the passenger compartment of a car with two front passengers and three rear passengers. As an example, a communication between the front passengers and the rear passenger sitting behind the driver is depicted. To improve the speech intelligibility, the passengers start speaking louder and lean or turn toward the listening communication partners, which increases the SNR by up to 20 dB. For longer conversations, this is usually tiring and uncomfortable. If the driver starts also to turn around, road safety becomes a concern as well.

The speech quality and intelligibility within a passenger compartment can be improved by an ICC system [1–6].[1] In Fig. 5.1, the passenger compartment of a car that is equipped with a commercially available ICC system is depicted. Since several

**Fig. 5.1** Communication with a passenger

[1] In the literature also the terms *in-vehicle communication* (IVC) system and *digital voice enhancement* (DVE) system are used.

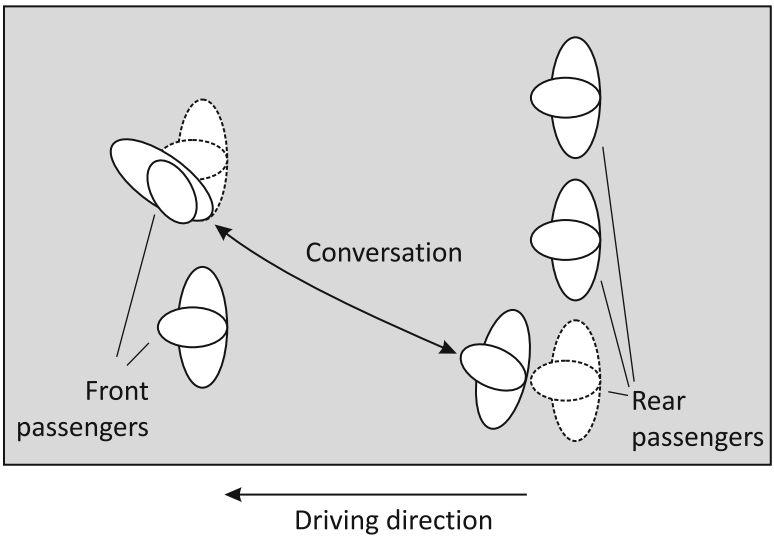



**Fig. 5.2** Schematic communication in a passenger compartment


**Fig. 5.3** Positioning of loudspeakers and microphones (With permission from SVOX)

of the results described in the remainder of this chapter are based on this system, we will show some more details about that system in the next few sections.

To improve the communication, the speech of the talking passengers is recorded by microphones and played back via those loudspeakers that are located close to the listening passengers. The positions of the microphones and of the loudspeakers of the system that we used for evaluation can be seen in Fig. 5.3. Four microphones are placed at the driver position; the loudspeakers for the rear seat passenger behind him are placed in the doors and on the hat rack.

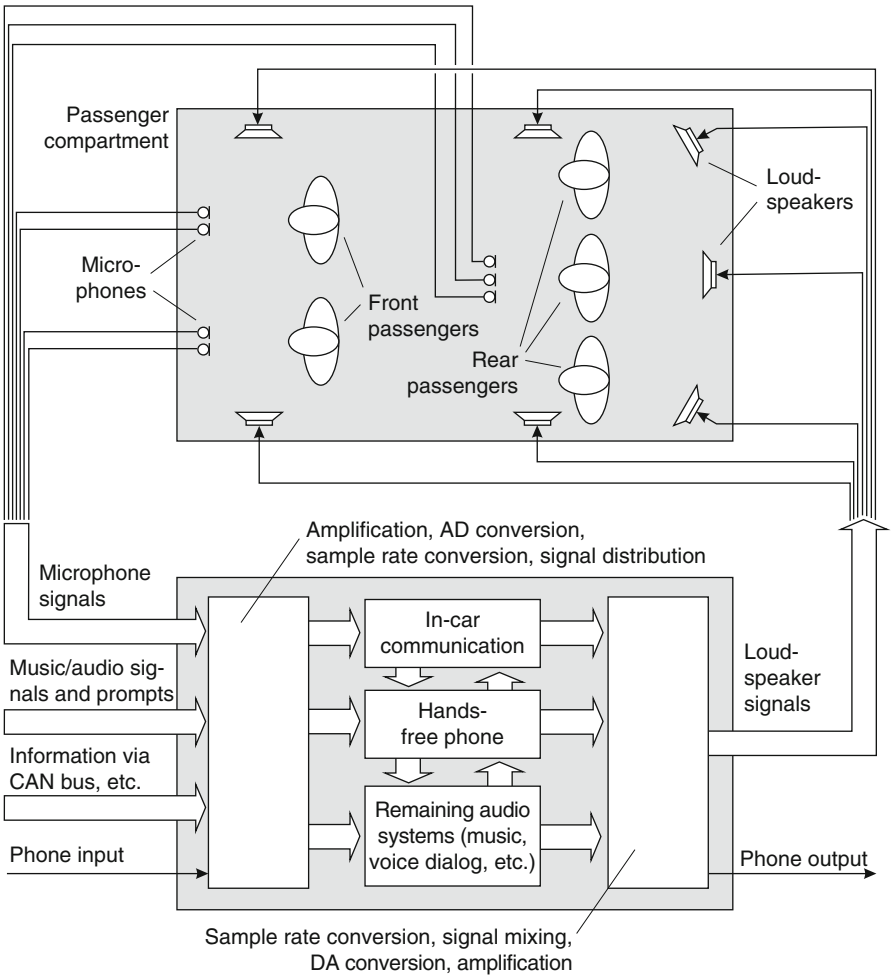



**Fig. 5.4** Structure of an ICC system embedded in the sound system of a vehicle

Figure 5.4 sketches the structure of an ICC system with several microphones and loudspeakers for both front and rear passengers. The ICC systems operate in a closed electroacoustic loop since the microphones pick up at least a portion of the loudspeaker signals. If this portion is not sufficiently small, sustained oscillations appear which can be heard as howling or whistling. The howling margin depends on the output gain of the ICC system, on the gains of the analog microphone and loudspeaker amplifiers, as well as on the acoustic properties of the passenger compartment. For this reason, all gains within an ICC system need to be adjusted carefully.

Since ICC systems are usually incorporated into the audio system of a vehicle, not only several restrictions but also additional possibilities arise. Cars that are equipped

with ICC systems usually also have hands-free and speech dialog systems installed. If these systems operate together, special care has to be taken. For example, if the ICC system adapts the playback volume in dependence of the background noise level, an echo cancellation unit that is included in hands-free or speech dialog systems will have to reconverge. Some details about those restrictions can be found in [3, 32].

The connection and interaction with other subsystems of the vehicle also leads to several advantages that can be exploited beneficially. External signals[2] such as music, navigation prompts, or warning signals can be used not only to identify the feedback paths in terms of the critical frequencies (those where howling would start) or the delay but also to estimate the impulse or frequency responses between all loudspeakers and all microphones. Furthermore, information that can be extracted from the automotive bus systems such as the CAN[3] bus also can help to improve the operation of ICC systems. For example, the results of weight sensors that are installed in seats in order to warn the passengers about not having fastened their seat belts can be used to deactivate the ICC processing for seats that are not occupied. However, going into the details of the interaction between ICC systems and other subsystems of a car is beyond the scope of this chapter. In the following, we will give a brief description of those signal processing components that are necessary for ICC systems.

To improve the stability margin, signal processing such as beamforming, feedback and echo cancellation, adaptive notch filtering, noise suppression, adaptive gain adjustment, equalization, and nonlinear processing can be applied. We will describe these components briefly in Sect. 5.3. In this section, we will focus on the general concept of ICC systems as depicted in Fig. 5.4.

Even if most automotive signal and data transfer options such as the MOST[4] bus are able to transport audio signals on a sample-by-sample basis, virtually all audio signals are transported and processed as blocks of samples, mostly using a minimal block size of 64 samples in today's cars. For that reason, it is of advantage since signal enhancement units such as ICC systems use block-based algorithmic approaches, but it leads, as a side effect, to a lower computational complexity compared to sample-by-sample processing. As a result, the very left and the very right signal processing blocks in Fig. 5.5 are transformations such as FFTs or more sophisticated filter banks that transform a block of signals into the subband domain and vice versa, respectively. As we will highlight later on, delay is a very critical issue for ICC systems. As a result, low-delay filter bank approaches are of special importance here. In addition, some pre- and postprocessing such as preemphasis or de-emphasis filters are usually performed. More details are described in Sect. 5.3.1.

After changing the signal representation from the time domain to the subband (or frequency) domain, some algorithmic components are applied to all input

[2] In this case, *external* is meant to be from the point of view of the ICC system.

[3] CAN stands for controller area network.

[4] MOST is the abbreviation for *media oriented systems transport*. This bus system is often used for the transport of audio data.

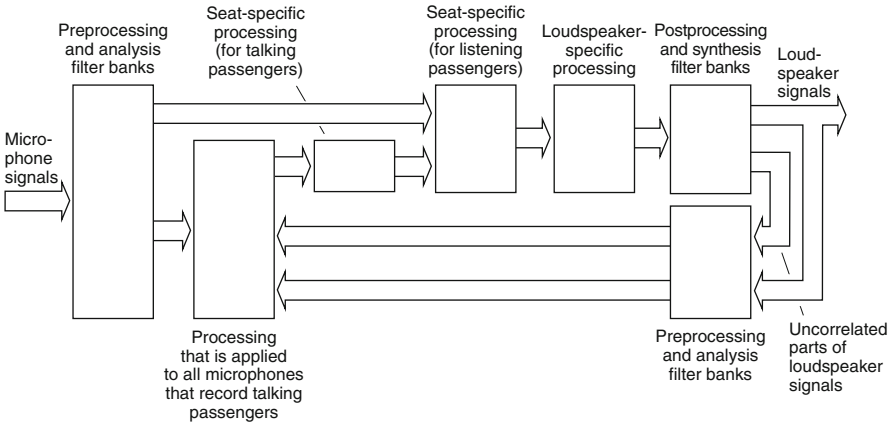



**Fig. 5.5** Basic signal processing units of in-car communication systems

(e.g. noise suppression) and to all output spectra (e.g. equalization). Thus, we have inserted two appropriate signal processing blocks in Fig. 5.5. Some further information about these two signal processing parts is presented in Sects. 5.3.2 and 5.3.5.

All signal components involved in between the microphone and the loudspeaker signal enhancement units can be grouped into two conversion parts. One group has to extract a dedicated signal for each seat that is occupied by a talking passenger. This can be done simply by selecting one of the microphones or by combining several of them (e.g. by means of beamforming). The outputs of this first signal processing group will be called the seat-specific signals and parameters of the *talking* passengers. The second group takes these talker signals and maps them onto signals that are specifically designed for the individual listeners. This includes a mixing process as well as gain adjustments in dependence of the noise level estimated at each listener seat. Finally, each listener signal is mapped onto the loudspeakers that are assigned to the listener seat using appropriate gain and delay settings.

In comparison to hands-free telephones or speech recognition engines, no methods for evaluating the quality of ICC systems have been standardized yet, and only a few have been published (e.g. [6, 7]). Thus, evaluation is more challenging as in most other speech and audio applications. Considering a basic ICC system, we focus on the analysis of ICC systems and give a general idea of how automatic evaluations can be performed.

Before we start with the evaluation, the restrictions and the boundary conditions of ICC systems as well as their consequences are discussed in the next section. Since a basic understanding of the involved signal processing units of a system is necessary in order to be able to design appropriate measurements and tests, we will continue with brief summaries of the individual processing units shown in Fig. 5.5 in Sect. 5.3.

When evaluating ICC systems, usually comparisons between the communication with and without the support of ICC systems are investigated. In addition, evaluation

results of an optimally operating ICC system (e.g. noise can be removed without any distortions for the remaining speech, and feedback can be suppressed ideally) can be used as an upper bound to benchmark the achieved communication improvement. Furthermore, it would be desirable if the well-understood theory of linear time-invariant systems could be applied in at least a few of the measurements. Since a lot of time-variant processing is involved in ICC systems, some sort of *freezing mode* would be helpful in some measurements and investigations. A detailed explanation about these system types used for the evaluation is given in Sect. 5.4.

In Sect. 5.5, we describe several aspects of the evaluation of ICC systems. We differentiate between subjective and objective schemes on one hand and between the quality perception of the talking and the listening passengers on the other hand. The chapter closes with a summary.

## 5.2 Boundary Conditions

During the design process of the ICC system, limiting conditions and certain system requirements must be considered. These are given by the physical behavior of the system itself, as, e.g. the electroacoustic feedback, and of course by the passenger's expectations on the ICC system.

Since not only one person is interacting in a conversation, the requirements of both the talking and the listening passenger have to be found to determine all boundary conditions. These conditions resulting from physical effects which are occurring during the operation, and the requirements of the passengers, restrict the system in terms of its adjustment.

The following section is therefore separated in three main parts that will give a closer look at the occurring effects and the resulting boundary conditions.

### *5.2.1 Physical Effects*

The main physical effect, which restricts the system in the practicable gain, is the closed electroacoustic loop or rather the resulting feedback. This feedback may lead to an instable overall system, if the system gain is chosen too high or if the applied countermeasures do not work properly. Therefore, the maximum amplification has to be defined in such a way that no howling due to the feedback arises. As mentioned before, this maximum allowed amplification can be increased by implementing some adequate signal processing techniques, for example, feedback suppression or equalization.

All the signal processing algorithms that should be applied are limited in their computational complexity by the signal processing power that is available. Fortunately, the computational power of the signal processing unit in cars is constantly

increasing and nowadays clearly higher than during the development of the first hands-free systems. This allows higher sampling rates and thus a larger signal bandwidth, which is essential for natural-sounding speech output. But still limitations in the algorithmic complexity are given and lead to additional boundary conditions that need to be considered during the design of an ICC system. In addition, limitations due to the characteristic of the electroacoustic transducers such as the bandwidths and the positions of the utilized microphones have to be considered for the design of ICC systems.

### 5.2.2 *Listening Passenger*

The main objective of ICC systems is to increase the speech intelligibility for the listening passenger. This is done by amplifying the speech signal of the talking passenger and thereby increasing the SNR at the ear of the listening passenger. This amplification cannot be increased to an arbitrarily high gain, due to the already mentioned physical limits. But how much improvement in terms of amplification is actually desired by the listening passenger? In most cars, the communication from the rear to the front passengers is marginally impaired, even at medium or high velocities. The usage of an ICC system would make the car unreasonably more reverberant and thus decrease the communication quality from front to rear. However, the opposite direction (from rear to front) behaves differently. The communication quality is impaired much more, especially at higher velocities.

The reason for the quality difference between both communication directions originates from the directionality of the human mouth. Figure 5.6 shows the average directionality for two frequency ranges when the driver is speaking [8]. In addition, the sound propagation to the other passengers is shown schematically.[5] For frequencies between 1,400 and 2,000 Hz, an attenuation of more than 10 dB between the front side (at 0°) and the back of the head (at 180°) can be measured. Due to this directionality and the arrangement within the passenger compartment, the rear passengers (especially the one directly behind the talking passenger) have problems to understand the front passengers.

Another effect that influences the boundary conditions of the amplification is the background noise level. As already mentioned above, the background noise consists of different sources, for example, engine noise, wind noise, and tire noise. Due to the fact that at standstill or low velocities the noise level is very low, the SNR is quite high, and therefore, there is no need to use an ICC system.[6] But if the car accelerates to higher velocities, the background noise level increases by more than 30 dB at all relevant frequencies, and the usage of ICC systems can improve communication quality.

[5] Effects due to the boundaries of the passenger compartment are not mentioned here.

[6] This may not be true for vans with more than two seat rows or even for buses.

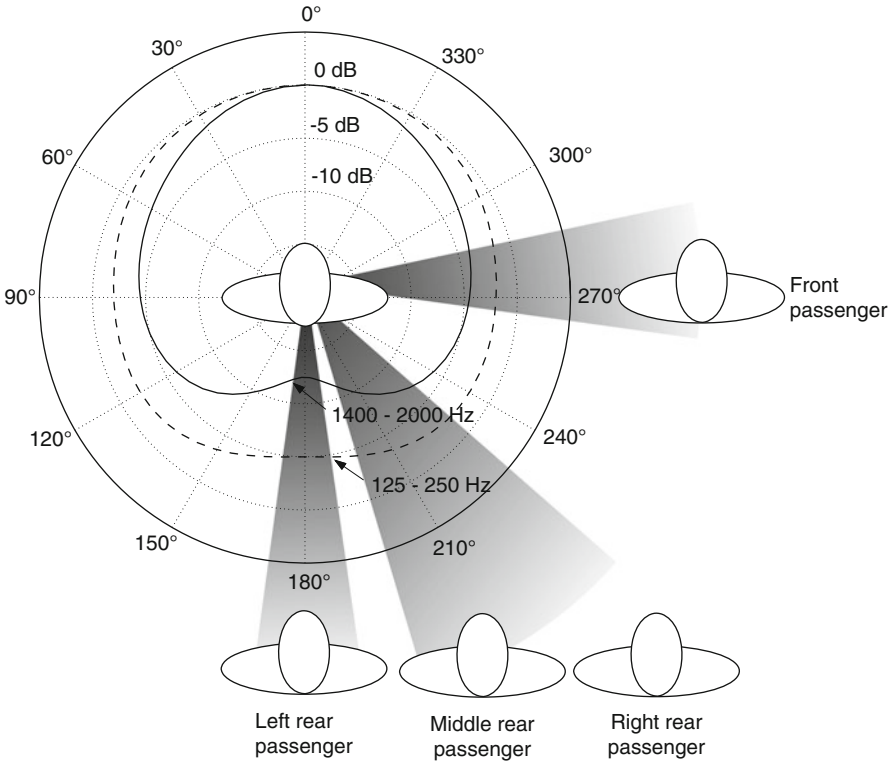



**Fig. 5.6** Average directionality of a speaking human head with additional passengers inside the vehicle (According to [8])

Hence, depending on the level of the background noise, different amplification values are needed to improve speech intelligibility.

Fortunately, the ICC system does not have to compensate the whole degradation of the SNR due to the increased background noise. Because of the Lombard effect, any person speaking in a noisy environment automatically will raise his voice in order to increase the efficiency of the communication [9]. This leads to an increased overall speech level while increasing the noise level. In literature rates of about 0.3–0.7 dB, speech power increment per 1 dBA increase of the background noise level (A-weighted) has been published [6, 10]. Due to the increased background noise as well as the higher speech level (Lombard effect), the system has to exceed an amplification of about 6–12 dB higher (dependent on the vehicle) in comparison to the standstill case [6].

All these boundary conditions enforce a trade-off between sufficient amplification for increasing the speech intelligibility and disturbances for the talking and listening passengers. This compromise has to be found independently for every type of vehicle.

In order to decrease the computational complexity, often block-based signal processing is applied, which leads to the insertion of a delay of a few milliseconds. Further delay is caused by the AD and DA converters,[7] by the signal transport between the processing units,[8] by the amplifiers for loudspeakers and microphones, and by the acoustical paths.[9] If the overall delay exceeds 15–20 ms, then the listening passengers are able to separate the two sound sources (direct wave front from the talking passengers and second wave front originating from the ICC system), which sounds very annoying [7]. Therefore, it is desirable to keep the delay as low as possible.

Because of the delay, another undesirable effect is created: the localization mismatch between visually and acoustically defined sources. In particular, this is a problem of the listening passengers located at the rear. The reason for this is the location of the rear loudspeakers, which are often behind the listening passenger, e.g. back shelf. If the reproduced speech signal is highly amplified, the listening passengers have the (acoustic) impression that the talking persons are located behind them. This mismatch of acoustical sensation and knowledge about the actual position of the talker causes a very unnatural impression of the communication. To solve this problem, the gain of the rear loudspeakers has to be limited dependent on the delay between the two sources. The amount of amplification until the localization mismatch effect appears is given by the so-called *precedence effect* [11]. The maximum amplification of the ICC system can be achieved at a delay of about 10–12 ms. Very small delays do not allow the second signal to be louder than the first one without affecting the spatial localization. As a result, the delay introduced by low-order (low-delay) block processing can be tolerated, and the advantages offered by this processing structure can be exploited.

However, in very high noise scenarios, the localization mismatch may not concern the listening passengers that much due to the fact that an improvement of the whole communication situation can be achieved by the higher gain. This raises the question, if small violations of the precedence curves can be tolerated and still a "good" communication can be achieved. The question to ask and to answer is: does the listening passenger prefer a higher system gain without the correct localization but with increased speech intelligibility? Up to now there is no detailed information or publication about this topic.

Another important issue is the increase of the overall noise level due to the reproduction of the recorded speech signal via the ICC system. The microphones record the talker's speech signal which is of course corrupted with background noise. Even though noise reduction methods are applied to enhance the signal

[7] Often delta-sigma converters are used in audio processing. This type of converter usually causes a delay of 0.3–0.7 ms, both for the DA as well as for the AD case.

[8] The delay is caused by waiting until enough samples are available in order to move them as a block. This helps to reduce the processing load of interrupt handling.

[9] A distance of 1 m between a loudspeaker and the ears of listening passengers results in a delay of about 3 ms (assuming 340 m/s for the speed of sound).

quality, potentially remaining noise components are amplified and played back over the loudspeakers. This may lead to an undesirable SNR reduction. In the worst case, the residual noise components are fluctuating, and the listening passengers are disturbed by the noise level changes while the ICC system is activated. As a result, the gain of the ICC system and the maximum attenuation of the noise suppression have to be adjusted such that neither a noise level increase nor noise level fluctuations due to the ICC system can be perceived by the listening passengers.

### *5.2.3 Talking Passenger*

The talking passenger has one main requirement for the ICC system: He simply does not want to be aware of the system.[10] The ideal case for the talking passenger would be if the situation within the compartment is the same whether with or without an ICC system. But since the speech signal is amplified by the ICC system and reproduced via the loudspeakers, it may occur that he hears himself while speaking. This effect is highly correlated with the gain and the delay of the ICC system. If the delay is too high, the talking passenger hears his own speech even at low gain levels as an echo.

Finally, it should be mentioned that ICC systems increase the degree of reverberation. This increase also depends on the delay and the gain of the system. The larger both are, the more the reverberation increases. This effect is tolerated up to a certain degree if the communication quality increases at the same time. However, if the reverberation time is increased up to more than 80–140 ms (depending on the type of car), passengers start complaining about the reverberant character of the vehicle [7].

## 5.3 Signal Processing for In-Car Communication Systems

As mentioned during the introduction of this chapter, some basic knowledge of the involved signal processing units of ICC systems is necessary if appropriate evaluation procedures should be investigated. For that reason, we will present brief descriptions of the individual signal processing blocks shown in Fig. 5.5 in this section before starting with details about the evaluation in the upcoming sections.

[10] This statement can be slightly modified for the listening passengers: The listening passengers should not be aware of a proper operating ICC system (in terms of localization, reverberation, noise increment, etc.) unless the system is switched-off.

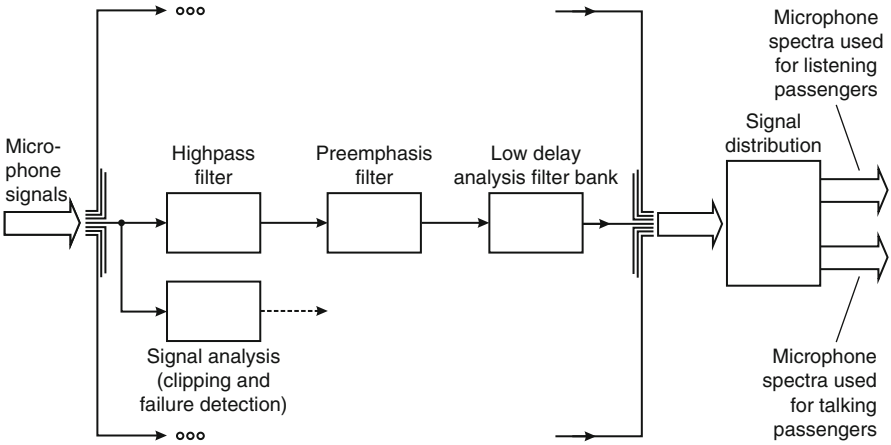



**Fig. 5.7** Preprocessing and analysis filter bank

### *5.3.1 Analysis and Synthesis Filter Banks*

Since typical background noise in automotive environments is dominating over speech components in the low-frequency range, high-pass filtering is applied as a first processing stage applied to all microphones – see Fig. 5.7. The cutoff frequency of the high-pass filters should be chosen between 80 and 300 Hz depending on the preferences of the user and on the type of vehicle (higher frequencies for sports cars, lower frequencies for sedans and vans).

A few signal processing components such as feedback and echo cancellation or adaptive beamforming (these units will be described in the following sections) assume a linear transmission between the loudspeakers and the microphones installed in the vehicle. If a microphone clips, this assumption is definitely violated. In order to exclude these periods, e.g. during the adaptation phase of cancellation filters, each microphone is analyzed not only in terms of a clipping detection but also in terms of a complete failure as caused, for example, by a breakdown of the power supply for the microphone amplifier. If the failure of a microphone is detected, the sensor is excluded from the succeeding processing.

A preemphasis filter and a low-delay analysis filter bank follow the high-pass filter. The preemphasis stage can be realized, e.g. as a fixed prediction-error filter that operates in the time domain. Its coefficients are adjusted to *whiten* the signal, leading to a smooth high-pass characteristic. The inverse filter will be applied behind the corresponding synthesis filter bank. Preemphasis and corresponding de-emphasis filters help to utilize the limited spectral resolution of filter banks (due to aliasing effects) in the best way.

Finally, the spectra of the microphone signals are grouped into two classes: Spectra of the first class are used by those processing units that enhance the signals of the talking passengers. Spectra of the second class are utilized for adjusting algorithmic

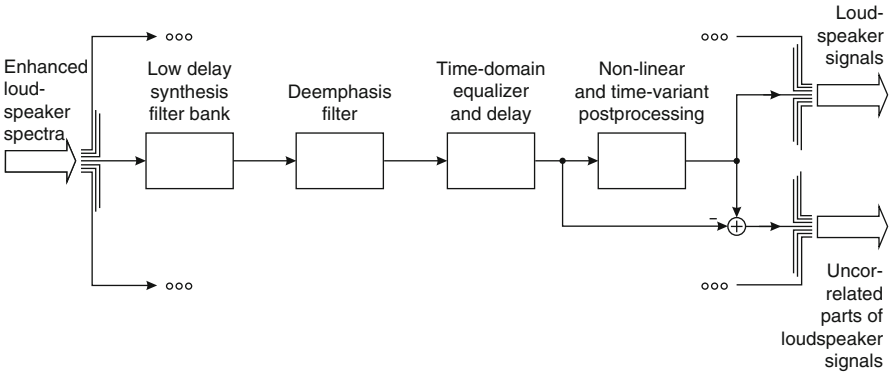



**Fig. 5.8** Synthesis filter bank and time-domain postprocessing

components for improving the playback for the listening passengers. If, for example, a setup with seven microphones, as depicted in Fig. 5.4, is used and only front-to-rear communication should be supported, then the four front microphones would be grouped as *microphones used for the talking passengers*. The three rear microphones could be used, e.g. for estimating the background noise level at each rear seat. These estimated noise levels can be utilized for determining appropriate playback volumes. Thus, the three rear microphones would be grouped as *microphones used for the listening passengers*.

Figure 5.8 depicts the synthesis part of ICC systems. In a first step, the spectra of the loudspeakers signals are transformed back into the time domain by an appropriate filter bank. As mentioned before, special emphasis should be placed on using low-delay approaches, e.g. as described in [12, 13].

If a preemphasis filter has been applied before the analysis filter bank, its inverse has to be computed after the corresponding synthesis filter bank. Furthermore, each loudspeaker might be equalized, either in terms of correcting undesired ripple in its frequency response or in terms of attenuating those frequencies where the feedback to the microphones is largest. Equalization can be applied either in the time domain, e.g. by means of allpass-based structures [14], in the subband domain by means of weighting factors, or as a combination of both. Usually, time-domain approaches are used to realize sharp notch or peak filters, and frequency domain approaches are used for smooth corrections. If spatial effects that require a certain delay for each loudspeaker should be exploited, also the delays might be adjusted individually for each loudspeaker.

Finally, signal processing that reduces the mutual correlation between loudspeaker signals can be applied. This is of special importance if the feedback and echo paths from the loudspeakers to the microphones are to be identified individually. The solution of this MIMO[11] system identification problem is not unique if the

[11] MIMO is the abbreviation for multiple input multiple output.

loudspeaker signals are fully correlated (measured in terms of coherence values close to one). In order to reduce the correlation, nonlinear or time-variant processing can be applied as known from stereo or multichannel echo cancellation [15]. For this type of postprocessing, usually a compromise between audible signal distortions on the one hand and a significant reduction of correlation, on the other hand, has to be found.

The synthesis filter bank and postprocessing unit produces two types of output signals: those that contain the entire signal components and those that contain only the uncorrelated signal parts (computed, e.g. by subtraction of the input and output of the nonlinear processing unit). The outputs first mentioned are designated for playback via the loudspeakers installed in the vehicle. The uncorrelated signals are used for computing the update terms of system identification algorithms as used in feedback or echo cancellation. Since both output types are required in the subband domain, analysis filter banks are applied afterward (see Fig. 5.5).

### *5.3.2 Processing Applied to All Microphones*

After being transformed into the subband (or frequency) domain, all microphone signals are enhanced by means of echo and feedback cancellation. We will denote the cancellation of those loudspeaker signals that are located in the direct vicinity of the microphone as echo cancellation. These loudspeakers usually do not play back the signal of the talking passenger that dominates the microphone signal. Signals that are emitted by loudspeakers being assigned to the other seats usually contain this signal component – as a consequence, we will call the compensation of such signals feedback cancellation. An overview about the subband-domain signal processing that is applied to all microphones is depicted in Fig. 5.9.

Since echo and feedback cancellation cannot guarantee a certain amount of distortion reduction, a Wiener-like suppression filter is usually applied [16, 17] after the cancellation units. For adjusting the filter weights, estimations of the residual echo and the residual feedback components in terms of their short-term power spectral densities are required. This can be implemented using the same methods as known from dereverberation of audio signals [18, 19].

In addition, stationary background noise [20, 21] as well as so-called wind buffets (distortions caused by turbulent air flow on the microphones) [22, 23] can be suppressed by spectral weighting. The weighting factors can be adjusted individually for each microphone if only a signal combination according to the criterion “best SNR wins” is applied (for details, see the next section). However, if beamforming is applied afterward, one has to make sure that the same weighting factors are applied to all microphone signals or the weighting has to be applied to the output spectra of the beamformer.

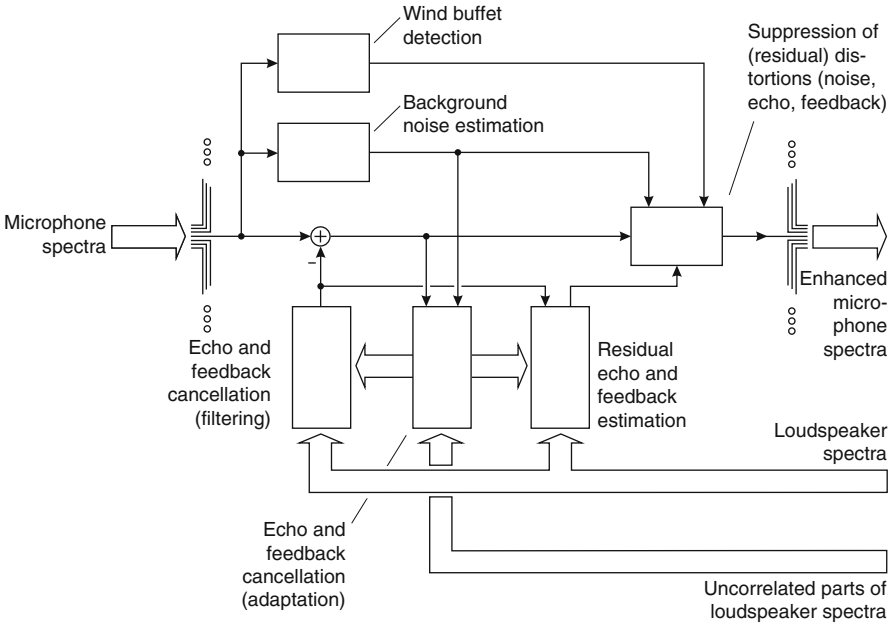



**Fig. 5.9** Signal processing applied to all microphones

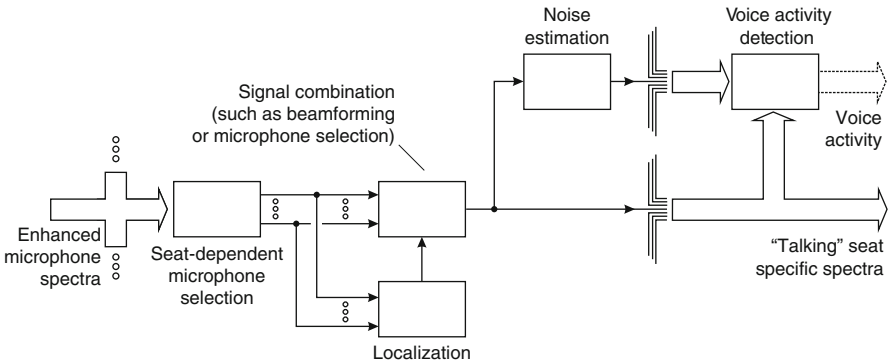



**Fig. 5.10** Seat-specific processing for the talking passengers

### *5.3.3 Seat-Specific Processing for the Talking Passengers*

The enhanced microphone spectra are assigned to the individual seats in a next processing stage. This could be also implemented in an overlapped manner. For example, all of the three rear microphones which are depicted in Fig. 5.4 could be

assigned to all of the three rear seats. In a next processing stage, three beamformers can be computed; each steered to one of the rear passengers (Fig. 5.10).

Since only the approximate position of each passenger is known a priori, a refinement of the steering directions of the beamformers can be achieved by using appropriate localization approaches [24]. However, also single microphone selection can be utilized. Here, in most cases, the sensor with the best SNR among all assigned microphones is selected. This is usually the microphone located closest to the mouth of the assigned passenger. However, if this microphone signal is disturbed by local noise (e.g. due to an open window or air condition), a wind buffet, or by signals emitted from a loudspeaker, it might be beneficial not to use the closest microphone but the one with the best SNR.

After combining all microphones that are assigned to one specific seat to a single signal, voice activity can be detected individually for each seat. To achieve this, the current background noise level as well as the short-term SNR is estimated. By comparing the individual SNRs as well as the noise and speech levels separately, robust voice activity detection can be achieved. The result of this detection unit is used in the following signal processing stages. For the sake of clarity, this signal is not shown in the overview diagram depicted in Fig. 5.5.

### *5.3.4 Seat-Specific Processing for the Listening Passengers*

The processing unit entitled "seat-specific processing (for listening passengers)" in Fig. 5.5 contains a two-stage signal mapping. In a first stage, the signals of the talking passengers are mapped onto the signals that are played back for the listening passengers. Usually, the signal of a talking passenger is not coupled back into those channels that are assigned to his seat for playback. Furthermore, the signals of talking passengers located closer to listening passenger are mapped with a much lower gain compared to signals of passengers being further away. As a last rule, nonactive talking passengers are attenuated by several decibels in order to keep the reverberation as small as possible. For that reason, the first mapping unit is also connected to the voice activity detection described in the previous section (Fig. 5.11).

The second mapping unit has the objective to distribute the playback signal of the individual seats onto the assigned loudspeakers. This mapping can (and should) be implemented for each loudspeaker individually since the individual loudspeaker-microphone paths differ in terms of their robustness against feedback.

In addition, for each seat and the assigned loudspeakers, a dedicated playback volume is computed by analysis of the seat-specific noise levels (which are also estimated within this processing unit). It is advantageous to compute this noise-dependent gain control individually for each seat, since the noise level can vary significantly from seat to seat, e.g. due to open windows or different adjustment of

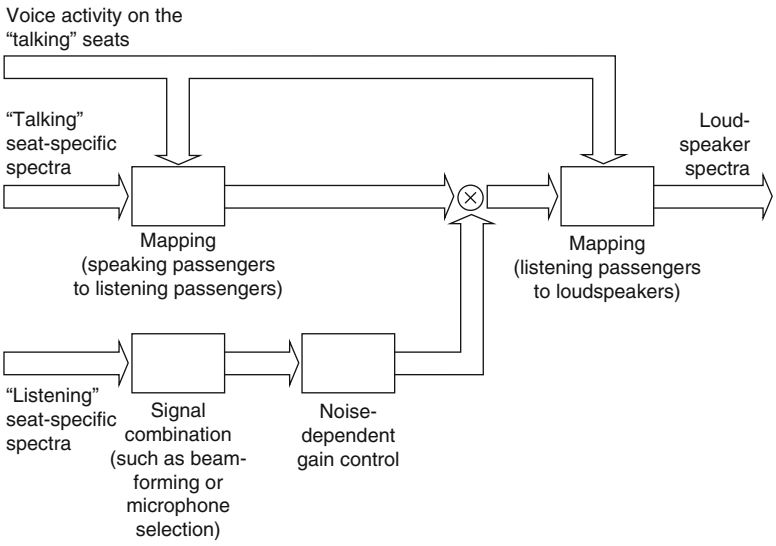



**Fig. 5.11** Seat-specific processing for the listening passengers

background music playback.[12] The gain adjustment is usually applied between the two stages of the signal mapping or in the second signal mapping stage if realized in a loudspeaker-specific manner. For reliable estimation of the background noise, those microphones that are located close to the listening positions should be used. In case of nonsymmetric setups, e.g. if only front-to-rear communication should be supported, these microphones can be operated at a much lower sample rate since only a rough estimate of the low-frequency part of the background noise is required.

### *5.3.5 Loudspeaker-Specific Processing*

As a last processing unit, loudspeaker-specific processing is described (see Fig. 5.12). As mentioned before, zero-phase equalization in terms of real-valued weighting in the subband domain is applied before playing back the loudspeaker signals. This type of equalization usually has lower complexity than equivalent time-domain structures if the complexity of the necessary analysis and synthesis filter banks is neglected. The objective of loudspeaker equalization is manifold:

[12] CD or radio playback is usually allowed during the operation of ICC systems. The overall playback level of the audio components, however, is often reduced in order to allow a comfortable voice communication.

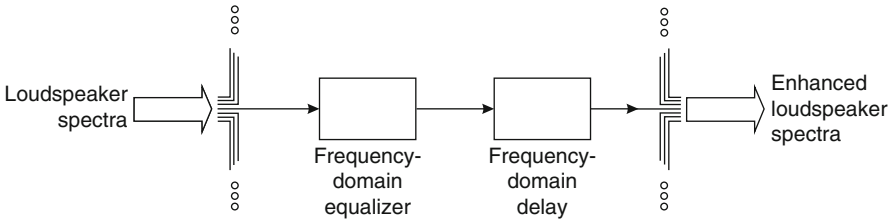



**Fig. 5.12** Loudspeaker-specific processing

- Midrange loudspeaker and tweeters should be driven only in the appropriate frequency range.
- Corrections for a better sound impression can be performed.
- Optimization of the feedback properties (reducing the gain at those frequencies that exhibit the largest coupling to the microphones).

If spatial effects based on appropriate delay adjustments are to be realized in high precision, the time-domain delay of Sect. 3.1 can be combined with a frequency selective phase shift in the subband domain: The rounded amount of delay in terms of samples is introduced in the time domain; the remaining fraction of samples is realized as a phase shift.

## 5.4 Types of ICC Systems Relevant for the Evaluation

When ICC systems are evaluated in an objective way, often the SNR or other measures are compared for different situations with an activated system and without any system support at all. This is usually a convenient comparison in order to highlight the advantages of ICC systems. However, to evaluate other properties of the system, for example, reverberation time, another system type besides the activated and deactivated one is necessary.

Some objective measures can be derived from the impulse response of the ICC system. However, an impulse response is defined only for a linear and time-invariant system, which the ICC system certainly is not. Therefore, we introduce the *frozen* system where all the nonlinear and adaptive (i.e. time-variant) elements of ICC systems have to be deactivated or stopped in terms of their adaption. Of course, such a system does not reflect the complete achievable improvement due to the usage of an ICC system, but it allows to classify the current state in which the system is and to calculate some properties of this *snapshot*. These measures then can be compared to other time instances or to other ICC systems.

One other interesting question during the evaluation process is how far away the *real* system from the optimal one is. To answer this question, the *ideal* system

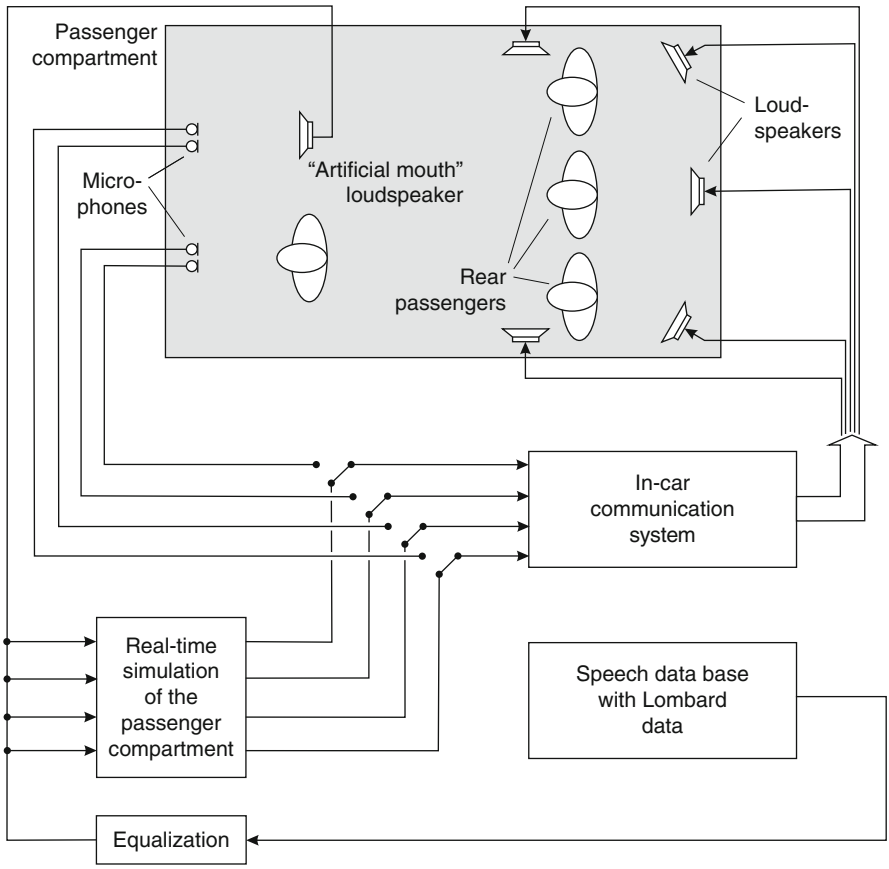



**Fig. 5.13** Ideal ICC system: the microphone signals are replaced by a clean speech signal convolved with appropriate impulse responses

must be defined first. The idea behind the *ideal* system is to excite the ICC system with perfect microphone signals as input, i.e. without any feedback and noise components. In Fig. 5.13, the realization of this *ideal* system is depicted. The talking passenger is simulated by an artificial mouth loudspeaker[13] which is fed by a speech signal. To achieve nearly the same speech spectrum as if passengers within the compartment would have spoken, a speech signal from an appropriate database has to be equalized vehicle specifically in order to create a natural artificial speech signal. In addition, the speech database should contain speech which reproduces the Lombard effect if the playback is used while driving the car. This signal will be reproduced via the artificial mouth in order to simulate the talking passenger on the front passenger's seat and is passed also to a simulation of the passenger compartment. This simulation is capable

[13] A loudspeaker with the same radiation patterns as a human head

of simulating the transmission from an artificial mouth loudspeaker via the microphones to the input of the ICC system. In addition, this device also allows adding characteristic car noise to ensure that algorithmic components that depend on the background noise level, such as the noise-dependent gain control, operate as in real scenarios. The output of the compartment simulation replaces the original microphones and is used as the input of the ICC system. In the end, the in this way designed system does not have any feedback and an adjustable noise level. Therefore, it reflects the upper boundary of the achievable improvement of the speech intelligibility. In the same way, the turned-off system equals the lower boundary.

## 5.5 Evaluation of In-Car Communication Systems

A first and intuitive approach to evaluate ICC systems would be to simply let subjects test and evaluate such systems. Tests carried out like this are called subjective methods. These tests are quite expensive since a large number of test subjects have to evaluate the system in order to obtain a representative result. In addition, the evaluation process after the experience phase has to be defined in an appropriate way. Such subjective tests have one big advantage: If the group of test subjects is well chosen with a sufficient number of subjects and an adequate questionnaire is designed, these tests give a meaningful assessment of the communication situation.

However, other methods, which are less expensive, easier to reproduce, and more reliable, would be desirable. Such methods are grouped under the generic term *objective test methods*. In order to evaluate an ICC system by means of objective methods, two main questions should be examined at the beginning:

- What does *improve* the communication between the passengers?
- What does *impair* the communication between the passengers?

Due to the fact that in a communication, at least two persons are necessary, the evaluation of the ICC system should not only be done for the listening passenger but also for the talking one. Of course, both communication partners do have different requirements on the system, and therefore, different factors have to be examined.

As mentioned in Sect. 5.1, up to now, there are no defined standards for evaluating an ICC system as they exist, for example, for speech or audio codecs or for hands-free systems [25, 26]. In [27] some first investigations about the analysis of the performance of an ICC system in an objective way were carried out. Therefore, in the following section, some first approaches and ideas of how to evaluate the here introduced ICC system are described. For that purpose, three main topics will be considered:

- First, the improvement for the listening passenger is evaluated by subjective methods.
- Secondly, objective approaches for listening quality evaluation are investigated.

- Finally, an approach for determining the degradation of the communication for the listening and the talking passenger due to the utilization of an ICC system is described.

### *5.5.1 Quality Improvement for the Listening Passengers*

The auditory impression of the listening passenger has the greatest influence on the quality evaluation of an ICC system. This is founded in the fact that the system was designed for increasing the communication quality for this passenger. Because of this, the evaluation is more widely diversified for this passenger as compared to the talking passenger (see Sect. 5.3).

#### 5.5.1.1 Subjective Methods

To determine the quality improvement for the listening passenger with respect to speech quality and speech intelligibility, at least two subjective methods may be utilized:

- Changes in the speech intelligibility can be measured by *diagnostic* or *modified rhyme tests* [28]. These tests are using a list of rhyming words, such as *game* and *name*, to focus on the intelligibility of each syllable.
- The speech quality can be derived using so-called *comparison mean opinion score* (CMOS) tests [29]. Here, well-known phrases are used as test sentences. This allows the listening subjects to focus on factors such as artifacts, reverberation, or naturalness of speech which are influencing the speech quality.

To create a comparable test situation which is identical for every test subject, the test phrases or words are played back within the passenger compartment via an artificial mouth loudspeaker located on the passenger's seat. This situation is recorded by binaural microphones mounted in the ears of the passenger sitting behind the artificial mouth loudspeaker, as shown in Fig. 5.14. The vehicle was driven at several speeds during the recording sessions. This allows the reproduction of the same hearing impression for every test subject by presenting the recorded binaural signal via a pair of calibrated headphones.[14]

The results of subjective tests depend on various boundary conditions. For example, a group of well-educated (in terms of acoustical analyses) test subjects will lead to a different result than a group of nonexperts. Other influences are gender and age of the subjects.

[14] If the test subjects would be sitting inside the car during the test, it would be complicated to ensure that the same amount of background noise is present in all tests. For that reason, it was decided to make recordings and using calibrated playback equipment instead of in situ tests.

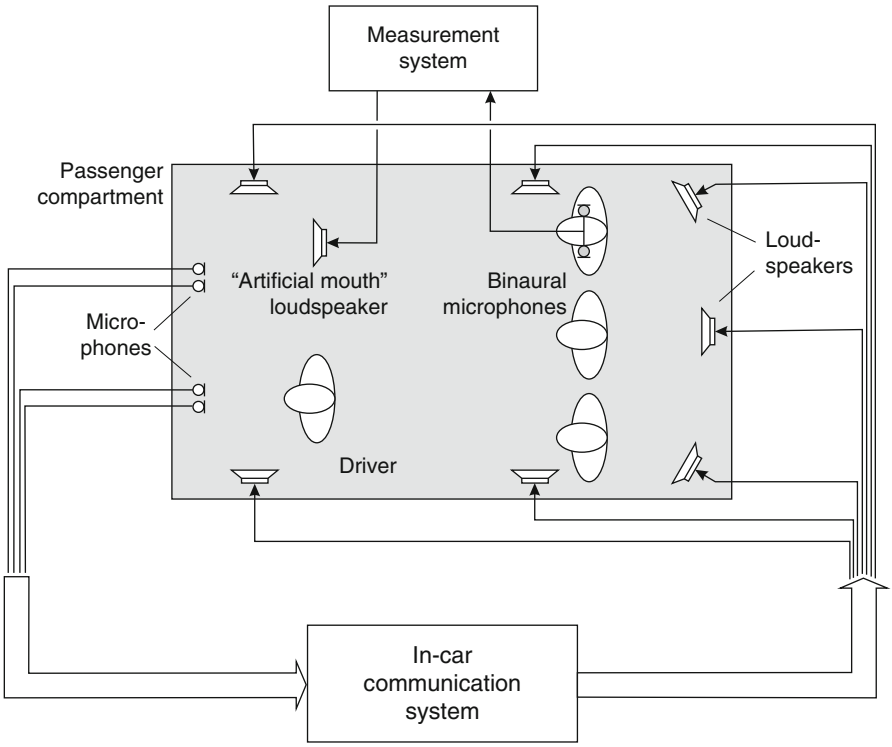



**Fig. 5.14** Measurement and evaluation of in-car communication systems

A rhyme test and a CMOS test were performed using four different scenarios [6]: tests with an activated ICC system and without any support of the system, both at standstill and at a speed of about 130 km/h.[15] The *ideal* system and the *frozen* system are not considered in this particular case.

For carrying out the rhyme test, first the rhyming word pair is visualized (with a computer screen) and then one of the words is randomly chosen and played back via the headphones. Afterward, the test subjects have to decide which word was reproduced acoustically.

Since the ICC system that was tested adjusts its amplification in dependence of the current background noise level, it can be assumed that there is no improvement in terms of speech intelligibility during standstill. This can also be concluded from the results of a rhyme test. Analyzing the results of 12 test subjects, each has voted on 40 pairs of rhyming words, showed no significant difference with and without the ICC system during standstill of the vehicle. However, at a speed of 130 km/h, the number of correctly understood words increased drastically by activating the ICC system [6].

[15] The subjective tests described here were executed using a different ICC system and not the one that is described in this chapter and has been used for the objective tests.

The CMOS test is carried out similar to the rhyme test. First, the test phrases are recorded within the vehicle at the different scenarios and afterward played back to the test subjects via a pair of calibrated headphones. To obtain a direct comparison between, e.g. a turned-on and turned-off ICC system, the playback of the audio files is performed in pairs of signals. In this case, the test subjects have to grade the two heard scenarios on a seven level grading scale ranging from much worse, worse, slightly worse, about the same, slightly better, better to much better. Again, this kind of test also reflects that an ICC system is not necessary during standstill, and in addition, it may disturb the passengers within the passenger compartment. However, at higher speeds, the result shows that the test subjects would prefer the activation of the ICC system. Nearly 90% preferred the ICC system to be activated at a speed of 130 km/h. Further and more detailed information about subjective tests and additional results can be found in [6].

#### 5.5.1.2 Objective Methods

After evaluating the system by two subjective methods, the quality improvement achieved by the ICC system should also be defined by objective methods.

At the beginning of this section, the ICC systems are evaluated by analyzing the impulse responses; the second part is the examination of the SNR improvement for the listening passenger. The last topic is the determination of the speech transmission index.

**Analysis of the Impulse Response**

The impulse response or rather the frequency response can be used as a first indication for the improvement or the degradation of the speech quality due to ICC systems. For this purpose, the impulse response from the mouth of the talking passenger to the ears of the listening one has to be measured with and without the ICC system.

To measure the impulse responses, the *frozen* system mentioned in Sect. 5.4 is excited by a test signal, e.g. white noise, and the output at the binaural microphones is recorded [8]. Due to the stopped and deactivated elements, it is important to create a well-suited test signal, which excites all relevant frequencies and does not stress the microphones of the ICC system which might produce nonlinear effects.

Once the impulse response is identified, parameters such as system delay, reverberation time, and frequency response can be extracted. In Fig. 5.15, the frequency responses measured with an activated and a deactivated ICC system are depicted. These frequency responses were measured between the artificial mouth located on the passenger's seat and the right ear of the listening passenger sitting directly behind the artificial mouth.

Assuming the arising background noise is suppressed by the ICC system and, therefore, not amplified, the difference between the frequency responses of a

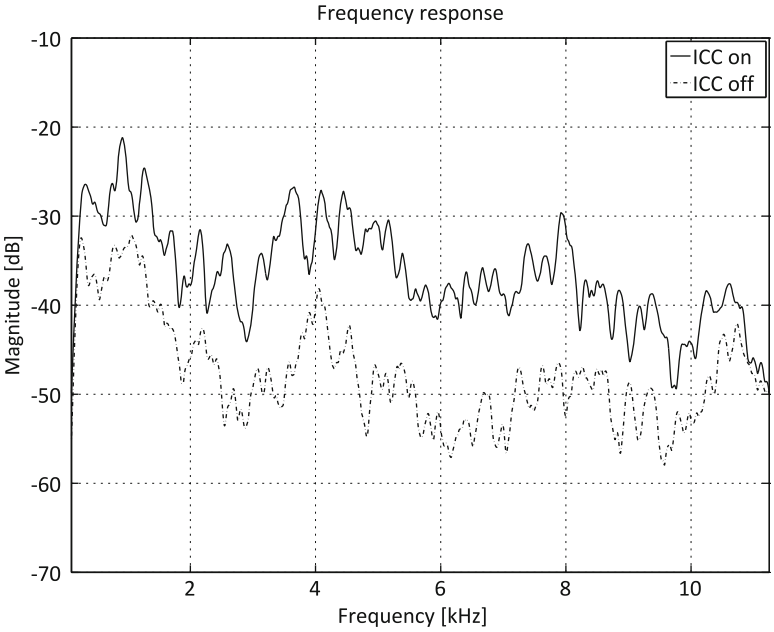



**Fig. 5.15** Frequency response (magnitude) at the listening person's right ear

turned-on ICC system and the turned off one shows the accomplished frequency-selective SNR improvement of the ICC system. By comparing the frequency responses of the ideal system, it can be identified how close the real system is to the ideal case.

### SNR Improvement

Another objective possibility to determine the quality improvement for the listening passenger is to measure the different SNRs with an activated ICC system and a deactivated one directly. These two measurements related to each other give a statement about the improvement in terms of the SNR achieved by the ICC system. Using this direct method, the LTI assumption is not necessary any more, and more realistic conditions (in terms of not only the measured signal but also of the ICC system [no usage of the *frozen* system]) can be applied.

To determine the SNR at the ears of the listening passenger, a predefined speech signal is used as a test signal. The test signal contains speech passages of female and male subjects and also speech pauses. The overall system is excited by means of the test signal using an artificial mouth at the passenger's position. This signal is again recorded by the binaural microphones located in the ears of the listening passenger sitting directly behind the passenger's seat, as shown in Fig. 5.14.

By using the predefined test signal $s(n)$, the speech and silent passages within this signal can be detected. Therefore, the squared magnitude of $s(n)$ is calculated and smoothed over the time:

$$\overline{|s(n)|^2} = \alpha \cdot |s(n)|^2 + (1-\alpha) \cdot \overline{|s(n-1)|^2}. \tag{5.1}$$

The smoothing factor is chosen as $\alpha \in [0.001,\ 0.01]$. By means of this smoothed discrete signal, the set of the sampling points for the speech and pause passages can be derived by

$$T_{speech} = \left\{ n \;\middle|\; \overline{|s(n)|^2} > S_0 \right\}, \tag{5.2}$$

$$T_{pause} = \left\{ n \;\middle|\; \overline{|s(n)|^2} < N_0 \right\}, \tag{5.3}$$

where $S_0$ gives the threshold for the speech passage detection and $N_0$ for the pause passages; in addition, $S_0 \geq N_0$ must hold. It is assumed that the signal recorded by the binaural microphones is defined as

$$y_i(n) = h_i(n) \ * \ s(n) + b(n) = u_i(n) + b(n), \tag{5.4}$$

where $b(n)$ indicates the additive noise, $h_i(n)$ the impulse response between the artificial mouth and the corresponding $i$-th binaural microphone, and $u_i(n)$ is equal to the convolution of the impulse response and the test signal. By combining these definitions, the noise power $P_{B,i}$ and the noisy speech power $P_{Y,i}$ can be estimated by

$$P_{B,i} = \frac{1}{\#T_{pause}} \cdot \sum_{n \in T_{\text{pause}}} |y_i(n)|^2 \tag{5.5}$$

and

$$P_{U,i} = \left( \frac{1}{\#T_{speech}} \cdot \sum_{n \in T_{\text{speech}}} |y_i(n)|^2 \right) - P_{B,i}, \tag{5.6}$$

where $\#$ defines the cardinality of a given set. Hence, the logarithmic SNR in dependence of the ear is defined as

$$\begin{aligned} SNR_i &= 10 \cdot \log_{10}\left( \frac{P_{U,i} - P_{B,i}}{P_{B,i}} \right) \\ &= 10 \cdot \log_{10}\left( \frac{P_{U,i}}{P_{B,i}} - 1 \right). \end{aligned} \tag{5.7}$$

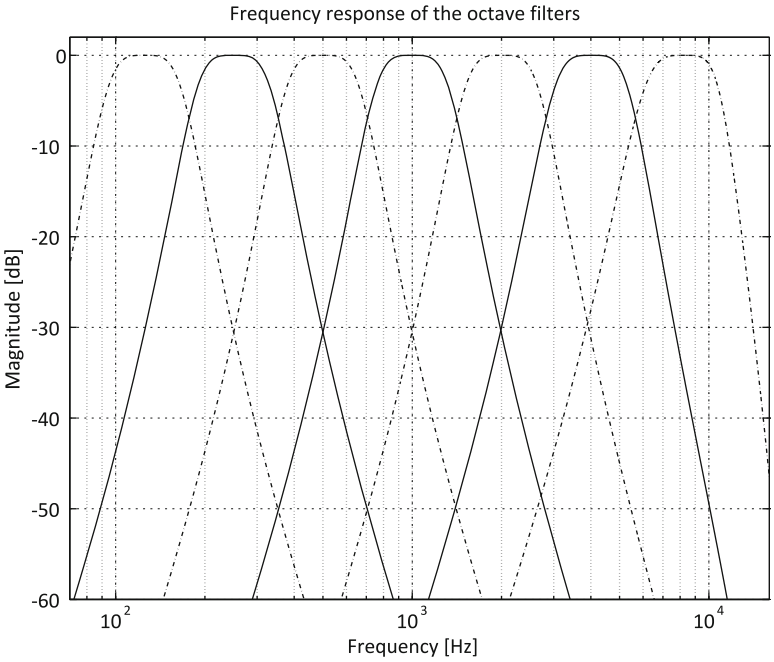



**Fig. 5.16** Frequency response (magnitude) of the octave filters

**Table 5.1** Center frequencies of the octave filters

| Octave band | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Center frequency | 125 Hz | 250 Hz | 500 Hz | 1 kHz | 2 kHz | 4 kHz | 8 kHz |

To obtain a frequency-selective observation, the SNR is determined in seven different octave bands. In Fig. 5.16, the corresponding octave filters are depicted. These octave bands are the same ones as used for calculating the speech transmission index as described in the next section. The test signals recorded using the binaural microphones have to be filtered by octave filters before the SNR is calculated according to the same approach as explained before in Eqs. 5.1–5.7. In Table 5.1, the center frequencies of the octave filters are depicted.

To compare the results, the SNR determination was done for the activated and the deactivated ICC system. Figure 5.17 shows the results for the right ear of the listening passenger. A significant increase of the SNR, especially in the octave bands 5–7, is observed.

At this point, only the SNR improvement achieved at the ears of the listening passenger is determined. Another interesting issue is the increase of the noise power inside the passenger compartment caused by the ICC system. The desirable situation in this case would be to avoid an increase in the noise power. Therefore, a measurement of the noise power difference between an activated and a deactivated ICC system measured at the listening passenger's ears gives also an indication about the quality of the ICC system.

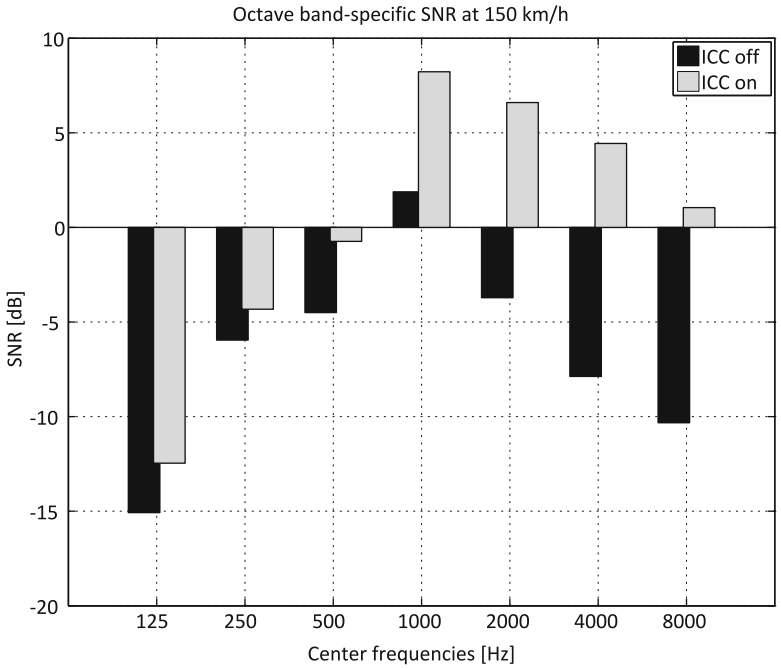



**Fig. 5.17** SNR improvement of the ICC system at 150 km/h

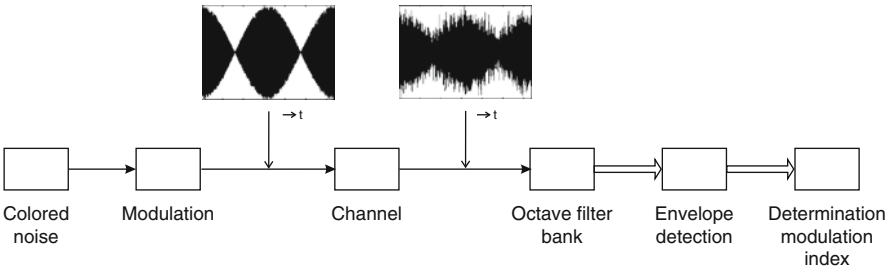



**Fig. 5.18** Basic concept for determining the modulation index

## Speech Transmission Index

The basic idea behind the *speech transmission index* (STI) is that the quality of a speech transmission can be described by the change in the modulation index. For that purpose, a special test signal is designed, and the reduction of the modulation index due to the transmission of this test signal is measured [30, 31].

In this particular case, we restrict ourselves to transmission channels with only linear and additive distortions, i.e., reverberation, noise, and echoes. In Fig. 5.18, the basic concept for measuring the modulation index is depicted.

The test signal is generated by a carrier signal $e(n)$, whose spectrum is similar to the long-term spectrum of speech. This carrier signal is then amplitude modulated which leads to the following definition of the test signal.

$$s_l(n) = \mathrm{e}(n) \cdot \sqrt{1 + \cos(2\,\pi\, F_l\, n\, T)}. \tag{5.8}$$

The received signal, in the following denoted by $y_l(n)$, is divided into seven bands by applying an octave filter bank – see Fig. 5.18. The different center frequencies of these filters are already mentioned in Table 5.1. The received signals depending on the octave band $k$ can be denoted as

$$y_{k,l}(n) = \tilde{e}(n) \cdot \sqrt{1 + m_{k,l} \cdot \cos(2\,\pi\, F_l\, n\, T)}, \tag{5.9}$$

where $\tilde{e}(n)$ corresponds to the carrier signal changed by the transmission.

Due to this approach, the envelope of each signal and, hence, the modulation index $m_{k,l}$ can be found. Up to this point, only one modulation frequency is considered. In order to investigate the influence of linear distortions in more detail, the number of modulation frequencies is extended up to 14 different frequencies which is indicated by the index $l$. These frequencies are distributed between 0.63 and 12.5 Hz in 1/3-octave steps [30].

To generate a test signal, the individual frequencies are excited sequentially over time. Afterward, the modulation index $m_{k,l}$ depending on the octave band and the modulation frequency can be estimated.

Using these modulation indices, the speech transmission index (STI) can be determined. Therefore, first the so-called *equivalent* SNR for the octave band $k$ has to be calculated as

$$\overline{SNR}_{k,l} = 10 \cdot \log_{10}\left(\frac{m_{k,l}}{1 - m_{k,l}}\right) \mathrm{dB}\,. \tag{5.10}$$

This *equivalent* SNR is limited to a certain range $R$ and normalized in order to generate a scale between zero and one. The resulting *transmission index* (TI) is defined as

$$TI_{k,l} = \min\left\{1,\ \max\left\{0,\ \frac{SNR_{k,l} - S}{R}\right\}\right\}, \tag{5.11}$$

where $S$ indicates the shift for the normalization. The values $S$ and $R$ are usually chosen as $R = 30\,\mathrm{dB}$ and $S = -15\,\mathrm{dB}$ [31]. Subsequently, the *modulation transmission index* (MTI) can be calculated as

$$MTI_k = \frac{1}{N} \sum_{m=1}^{N} TI_{k,l}\ , \tag{5.12}$$

**Table 5.2** Weighting factors of the octave bands corresponding to [30]

| Octave band $k$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $W_k$ | 0.129 | 0.143 | 0.114 | 0.114 | 0.186 | 0.171 | 0.143 |
| Center frequencies | 125 Hz | 250 Hz | 500 Hz | 1,000 Hz | 2,000 Hz | 4,000 Hz | 8,000 Hz |

where $N$ represents the number of used modulation frequencies. Finally, the STI can be derived by a weighted sum of the transmission indices

$$STI = \sum_{k=1}^{7} W_k \cdot MTI_k \ , \tag{5.13}$$

where $W_k$ represents the octave band–weighting factor. The weighting factors are chosen due to the psychoacoustic importance of each octave band. Octave bands, which are essential for the hearing impression, get a larger weight than others. In this particular case, the weights were chosen as referred in Table 5.2 corresponding to [30].

Using this approach, the single STI values were measured within different scenarios. To measure the necessary signals, the same configuration as shown before in Fig. 5.14 was utilized. In addition, breaks were inserted into the test signal in order to prevent that the ICC system changes too much in its characteristics, e.g. the gain, by detecting the test signal as noise. During the pauses, the ICC system can adjust back to the initial settings. The test signal was transmitted via the artificial mouth loudspeaker and again recorded the binaural ear microphones. This was carried through different velocities and with an activated and a deactivated ICC system in order to compare the obtained results.

Figure 5.19 depicts the results for the right ear of the listening passenger. The measuring shows that the STI values, due to the activation of an ICC system, are increased by about 0.15 due to the usage of an ICC system. In addition, the STI decreases in the case of the deactivated ICC system by accelerating to higher speed. If the ICC system is turned on, the STI values for 90 km/h and 120 km/h are nearly the same. In this case, the ICC system compensates the decreased SNR by using a higher amplification. The reduction of the STI from 120 to 150 km/h can be justified by the maximum amplification of the ICC system which is obtained in between these two velocities. Therefore, no further amplification is provided by the ICC system, and the STI decreases due to the increased background noise level. Further evaluation results of a similar ICC system by deriving the STI can be found in [27].

### 5.5.2 *Quality Degradation for the Listening Passenger*

The quality degradation for the listening passenger has two main reasons. The first one is the increase of reverberation through the ICC system. The second is the mismatch in the localization of the acoustical and visual sources of the speech

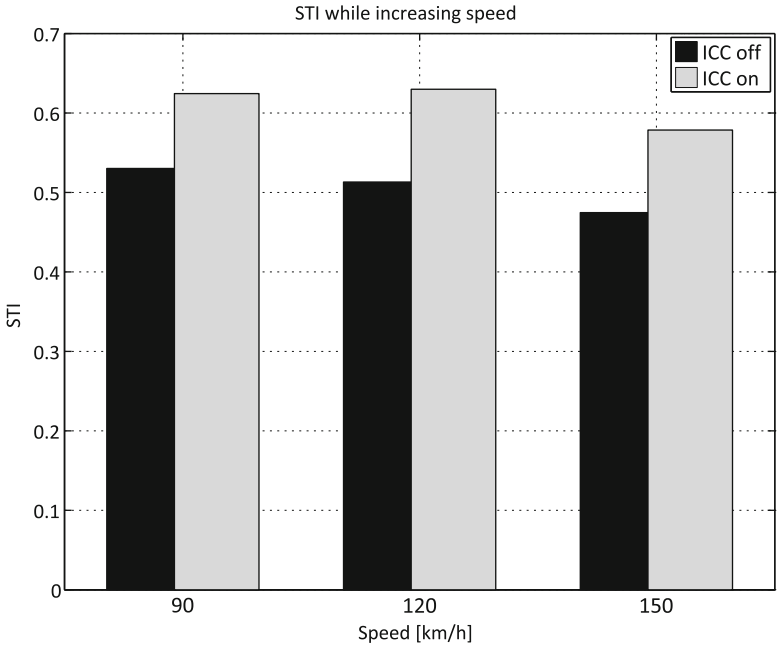



**Fig. 5.19** Measurement of the speech transmission index for the right ear of the listening passenger

signal. As mentioned in Sect. 5.2, the amount of reverberation is dependent on the gain and the delay of the ICC system, but also the localization is influenced by these two factors.

#### 5.5.2.1 Reverberation Time

The degree of reverberation can be expressed, e.g. by the reverberation time $T_{60}$, which can be derived from the energy decay curve (EDC) of the corresponding impulse response $h_n$ [8]. The normalized EDC is defined as

$$D(i) = 10 \log_{10} \left\{ \frac{\sum_{n=i}^{\infty} h_n^2}{\sum_{n=-\infty}^{\infty} h_n^2} \right\}. \tag{5.14}$$

The gradient of the EDC gives the reverberation time, which is defined as the time it takes the EDC (or equivalently the impulse response) to decay by 60 dB.

Figure 5.20 illustrates the EDCs of an activated ICC system and a deactivated one as an example. It can be seen that the activated ICC system increases the reverberation time in comparison to the deactivated case.

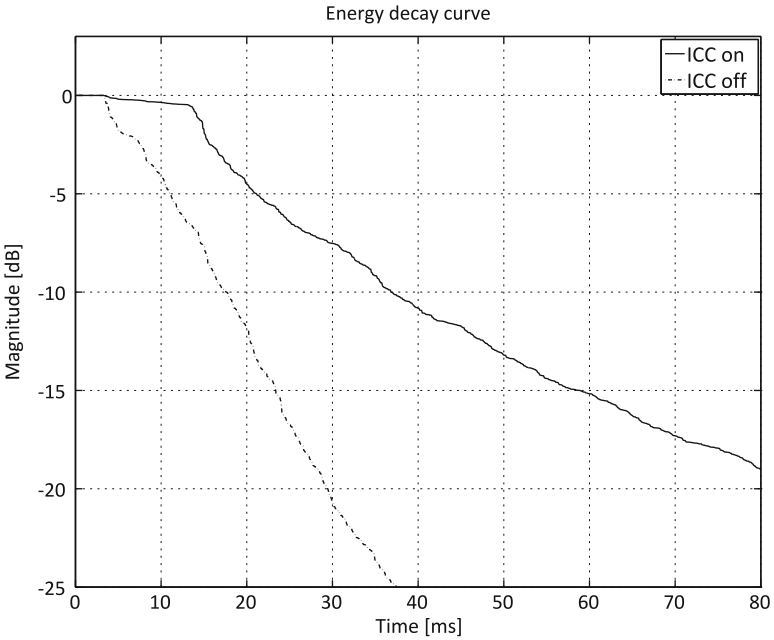



**Fig. 5.20** Energy decay curves of the impulse responses at the listening person right ear

The ratio between the reverberation time of the turned-off and the activated system describes the increase of the reverberation caused by the ICC system. The less the reverberation time is increased, the better the quality of the system is. The reverberation time with an activated ICC system should not exceed a certain threshold which is also depending on the type of the car. For example, a vehicle with only two seat rows should not exceed a reverberation time of 80–150 ms (depending on the type of car).

#### 5.5.2.2 Localization

The localization mismatch of the acoustical source is also one factor, which may decrease the communication quality of the listening passenger. As mentioned before in Sect. 2.2, this is correlated with the gain and the delay of the system as well as with the location of the loudspeakers. Therefore, in any case, the loudspeakers which support the correct localization of the acoustical source (e.g. loudspeakers located in the rear side of the front seat) should be utilized by an ICC system.

Up to now, the evaluation of the localization quality can only be done by subjective test methods. To perform an evaluation, a group of test subjects who are educated in terms of acoustical analysis should experience the ICC system and grade the localization impression.

### 5.5.3 Quality Degradation for the Talking Passenger

Because of the reproduced speech signal, the reverberation within the passenger compartment is increased. This disturbs not only the listening passenger but also the talking passenger. Therefore, the determination of the reverberation time can be utilized to define the quality degradation for the talking passenger as well.

If the delay is too large, the acoustic wave fronts (the one of the talking passenger and the playback of the ICC system) are perceived separately which impairs the communication for the talking passenger. Therefore, also the delay is an interesting quality measure.

However, if the delay is sufficiently small but the gain is too large, the talking passengers perceive their voices as echoes. To evaluate this phenomenon, the impulse responses between the mouth and the ears of the talking passenger are measured. By means of these impulse responses, the corresponding frequency responses can be calculated and evaluated. The difference between the frequency responses of an activated ICC system and a deactivated one gives a measure of the frequency-selective amplification of the speech signal. If this amplification is too large, there is a high probability that the talking passengers hear themselves. In addition, arising echoes due to the feedback can be detected by an inspection of the impulse response. Therefore, a masking envelope can be defined and compared with the impulse response. If the impulse response exceeds, this masking envelope echoes should be audible [6].

## 5.6 Some Ideas for an Automatic Evaluation of ICC Systems

Even if we have explained a range of different objective measures which are necessary to obtain an automatic evaluation of the ICC system, a complete definition of such an evaluation process is difficult. However, some first approaches and some ideas for further considerations will be given in this section.

As seen in the previous sections, there are two main questions which have to be answered when evaluating an ICC system:

1. Does the ICC system improve the speech quality for the listening passenger?
2. Is the communication quality reduced for the talking passenger by the ICC system?

By answering these questions, a statement about the quality of the complete ICC system can be made.

The first question can be answered by dividing this into two partial answers. The first considers measures which indicate the improvement for the listening passenger. The second partial answer deals with the factors which impair the communication quality for the listening passenger. For example, the speech intelligibility improvement can be identified by determining the SNR improvement in the ears

of the listening passenger. One advantage of measuring the SNR improvement is that it takes also the increase of the noise level within the passenger compartment into account. Also, the STI is a measure that helps to answer this question. This measure has a further advantage: It accounts for linear distortions such as reverberation. Other objective methods like the analyses of the transfer function or even new methods and indices would be conceivable at this point. The degree of speech quality degradation can be defined by determining, for example, the reverberation time. If this time exceeds a certain value, it can be concluded that the listening passenger senses an impaired communication. Further indicators are the delay induced by the ICC system and the loudness of the reproduced speech signal. By combining these two indicators, the localization mismatch of the acoustical source and the visual source can be analyzed. However, not all factors creating the subjective hearing impression for the listening passenger can be measured in an objective way. Therefore, in this particular case, some new objective methods have to be found first in order to design an automatic evaluation.

The second question can be answered by analyzing the transfer functions or the impulse responses from the mouth to the ears of the talking passenger with and without an ICC system. The increase of the transfer functions gives a first indication of how much the talking passenger hears the feedback of his speech signal. By analyzing the impulse response, appearing echoes can be discovered. However, also in this case, new objective measures have to be found in order to reproduce subjective hearing impressions like, for example, naturalness of speech.

To achieve a comprehensive evaluation of the ICC system, our suggestion would be to create some sort of weighted measure of at least one of the objective measures within each question. Thus, all important factors concerning the communication quality are mapped and evaluated in this measure. To overcome the problem with the up to now hardly measureable factors (e.g. naturalness of speech), a correlation between subjective and objective evaluations would be conceivable. Therefore, subjective and objective methods are carried out in the same scenario and related to each other. In addition, by this correlation, the reliability of the objective measures, in terms of the capability to reflect the actual communication quality, can be estimated. All these approaches are only suggestions from the authors. In any case, further research on the topic of automatic evaluation is required.

## 5.7 Conclusions

In this chapter, an overview of ICC systems and their evaluation has been given. There are several boundary conditions that have to be taken into account when an ICC system is designed, and it has been argued that the gain and also the delay introduced by the system are two factors that largely influence the quality of such a system.

Listening tests showed that ICC systems are able to improve the communication quality in cars significantly if the car is moving at moderate or high speed. Several

of these subjective methods have been suggested for describing the quality improvement or degradation for the listening passenger. Similar evaluations can be carried out for the talking passenger.

Since subjective tests are rather time-consuming, the aim is to develop an automatic system evaluation based on objective criteria. Examples for these measures are the SNR and the STI, which also proved to be capable of reproducing the results of some of the subjective tests. However, in some cases, it is difficult to find appropriate indicators, e.g. for judging on localization effects. An approach for the design of an automatic evaluation scheme has been presented by pointing out which questions should be answered by such a system. Even though some meaningful objective measures have been found, further research in this particular field is necessary to obtain more indicators that are correlated to the auditory perception of humans.

Since ICC systems are starting to enter the market, the demand for standardization of quality evaluation procedures arises. Evaluation systems could not only help to compare different ICC systems but also assist during the design and parameterization process.

## References

1. Cifani S, Montesi LC, Rotili R, Principi E, Squartini S, Piazza F (2009) A PEM based algorithm for acoustic feedback control in automotive speech reinforcement systems. In: Proceedings of ISPA 2009, Chengdu, China, pp 656–661
2. Freudenberger J, Pittermann J (2008) Noise and feedback suppression for in-car communication systems. ITG Fachtagung Sprachkommunikation, Aachen
3. Haulick T, Schmidt G (2006) Signal processing for in-car communication systems. Signal Process 86(6):1307–1326
4. Ortega A, Lleida E, Masgrau E (2001) Acoustic echo control and noise reduction for cabin car communication. Proc EUROSPEECH 2001 3:1585–1588
5. Ortega Gimenez A, Lleida Solano E, Masgrau Gómez EJ, Buera Rodríguez L, Miguel Artiaga A (2006) Acoustic echo reduction in a two-channel speech reinforcement system for vehicles. In: Abut H, Hansen JHL, Takeda K (eds) Digital signal processing for in-vehicle and mobile systems 2. Springer, New York
6. Schmidt G, Haulick T (2006) Signal processing for in-car communication systems. In: Hänsler E, Schmidt G (eds) Topics in acoustic echo and noise control. Springer, Berlin, pp 553–605
7. Haulick T, Schmidt G, Wolf A (2009) Evaluation of in-car communication systems. In: Proceedings of DSP workshop for in-vehicle systems and safety, Dallas, USA
8. Kuttruff H (2000) Room acoustics, 4th edn. Spon Press, London
9. Lombard E (1911) Le signe de l'elevation de la voix. Ann Maladies Oreille, Larynx, Nez Pharynx 37:101–119, In French
10. Hanson JHL (1994) Morphological constrained feature enhancement With adaptive cepstral compensation (MCE-ACC) for speech recognition in noise and lombard effect. IEEE Trans Speech Audio Process T-SA-2(4):598–614
11. Haas H (1972) The influence of a single echo on the audibility of speech. J Audio Eng Soc 20:145–159

12. Kurbiel T, Göckler HG, Alfsmann D (2009) A novel approach to the design of oversampling low-delay complex-modulated filter bank pairs EURASIP J Adv Signal Process, Article ID 692861, vol 2009
13. Mauler D, Martin R (2007) A low delay, variable resolution, perfect reconstruction spectral analysis-synthesis system for speech enhancement. In: Proceedings of EUSIPCO 2007, Poznań, Poland, pp 222–227
14. Zölzer U (ed) (2002) DAFX – digital audio effects. Wiley, Hoboken
15. Benesty J, Morgan DR, Sondhi MM (1996) A better understanding and an improved solution to the specific problems of stereophonic acoustic echo cancellation. Bell Labs Technical Memorandum.
16. Hänsler E, Schmidt G (2004) Acoustic echo and noise control: a practical approach. Wiley, Hoboken
17. Vary P, Martin R (2006) Digital speech transmission. Enhancement, coding and error concealment. Wiley, Hoboken
18. Habets EAP, Gannot S, Cohen I (2008) Dereverberation and residual echo suppression in noisy environments. In: Hänsler E, Schmidt G (eds) Speech and audio processing in adverse environments. Springer, Berlin
19. Naylor PA, Gaubitch ND (eds) (2010) Speech dereverberation. Springer, Berlin
20. Benesty J, Chen J, Huang Y, Cohen I (2009) Noise reduction in speech processing. Springer, Berlin
21. Heute U (2006) Noise reduction. In: Hänsler E, Schmidt G (eds) Topics in acoustic echo and noise control. Springer, Berlin
22. Elko G (2007) Reducing noise in audio systems. US Patent 7,171,008 B2
23. Hetherington P, Li X, Zakarauskas P (2003) Wind noise suppression system. US Patent 7,885,420 B2
24. Doblinger G (2006) Localization and tracking of acoustical sources. In: Hänsler E, Schmidt G (eds) Topics in acoustic echo and noise control. Springer, Berlin
25. Heute U (2008) Telephone-speech quality. In: Hänsler E, Schmidt G (eds) Speech and audio processing in adverse environments. Springer, Berlin
26. Kettler F, Gierlich HW (2008) Evaluation of hands-free terminals. In: Hänsler E, Schmidt G (eds) Speech and audio processing in adverse environments. Springer, Berlin
27. Ortega A, Lleida E, Masgrau E (2005) Speech reinforcement system for car cabin communications. IEEE Trans Speech Audio Process 13(5):917–929
28. Voiers W (1983) Evaluating processed speech using the diagnostic rhyme test. Speech Technol 30–39, vol. 1 Jan/Feb
29. ITU-T Recommendation P.800 (1996) Methods for subjective determination of transmission quality. International Telecommunications Union, Geneva
30. Steeneken HJM, Houtgast T (1980) A physical method for measuring speech-transmission quality. J Acoust Soc Am 67(1):318–326
31. Steeneken HJM, Houtgast T (1985) A review of the MTF concept in room acoustics and its use for estimating speech intelligibility in auditoria. J Acoust Soc Am 77:1069–1077
32. Haulick T, Iser B, Schmidt G, Wolf A (2008) Hands-free telephony and in-vehicle communication. European Patent Application, EP 2 151 983 A1

# Chapter 6
# Wideband Hands-Free in Cars – New Challenges for System Design and Testing

**Hans W. Gierlich and Frank Kettler**

**Abstract** Wideband hands-free technology in cars provides the capability to substantially improve the quality of the perceived speech for the driver as well as for the far-end communicational partner. However, in order to achieve a superior wideband speech quality, a variety of requirements – different from narrowband telephony – have to be taken into account. A few important parameters most critical for the success of wideband in cars are discussed. Since wideband transmission is at least partially IP-based, a higher delay can be expected as compared to narrowband calls. The impact of higher delay on the communicational quality is shown, and the different elements contributing to the delay in car hands-free systems are shown. Also, the impact of delay on conversational quality is discussed. The other aspects of wideband communication include speech sound quality in sending and receiving direction. A new objective test procedure 3QUEST for speech quality with background noise and its application to wideband car hands-free is introduced. For echo performance in wideband, new subjective test results are shown, and results of a new objective echo analysis method based on the hearing model "Relative Approach" are shown.

**Keywords** Human perception • System design • Wideband hands-free technology

## 6.1 Introduction

The deployment of wideband hands-free technology in cars provides the capability to substantially improve the quality of perceived speech for the driver as well as for the far-end communicational partner. In-vehicle hands-free terminals would benefit from wideband than traditional communication terminals. The difference in sound

H.W. Gierlich (✉) • F. Kettler
HEAD Acoustics GmbH, Herzogenrath, Germany
e-mail: H.W.Giderlich@head-acoustics.de

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_6, 

quality would immediately be noticeable to the driver since she/he will always have a perceptual comparison of the high-quality audio playback in the car for other media. Speech intelligibility in the car will be significantly increased, which is highly beneficial, especially in background noise situations while driving. As a consequence, the listening effort for the driver is reduced, the distraction from the primary task (driving) will be reduced as well. Thus, the driver's distraction may be reduced substantially if wideband technology is implemented properly. However, in order to achieve a superior wideband speech quality, a variety of requirements different from narrowband telephony have to be taken into account. This includes careful system design of all components involved in the transmission. The impact of delay and the components contributing to delay are described in Sect. 6.2. The listening speech quality analyses for wideband car hands-free systems are described in Sect. 6.3, and the special requirements on echo performance are given in Sect. 6.4.

## 6.2 Transmission Delay

Since wideband transmission is most likely IP-based when connecting to a fixed line network, a higher delay can be expected as compared to narrowband calls. The higher delay not only contributes to a degraded communicational quality but also requires a more thorough investigation of the echo loss required for wideband systems. This concerns spectral as well as temporal characteristics and is discussed in Sect. 6.4.

An overview of the components of a typical hands-free system and their effect on delay is given in Fig. 6.1.

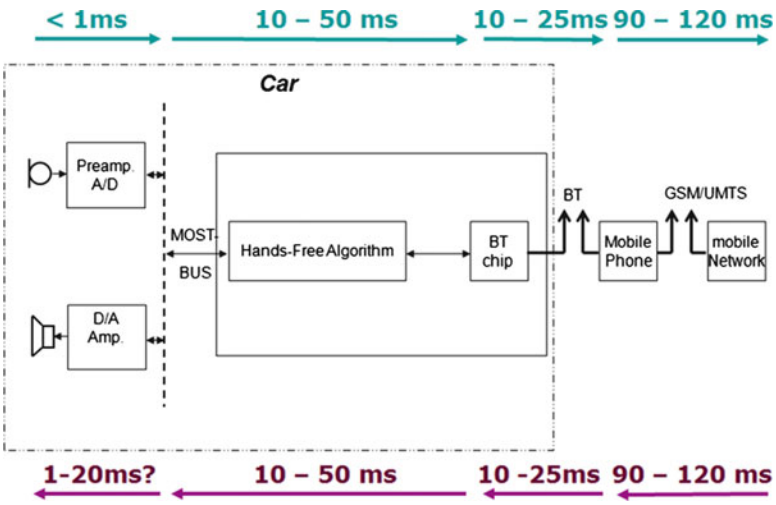



**Fig. 6.1** Typical components of a car hands-free system and their contribution to transmission delay

While the microphone and its connection to the in-car audio or in-car bus system typically introduces low delay, the hands-free algorithm in the uplink (sending direction) may introduce a significant transmission delay. In uplink, the most important signal processing is active: echo cancelation and noise cancelation. Both require substantial signal processing capacity, and in wideband systems, it is likely that these algorithms are realized in the frequency domain and/or in sub-bands. These technologies are known not only to provide good performance [1] but also to introduce higher delay – compared to simple LMS-type algorithm.

Signal processing in downlink may also introduce more delay than known in narrowband systems. This is caused, e.g. by advanced adaptive signal enhancement techniques, such as adaptive equalization or compression, and especially by wideband extension techniques. Such techniques can be used to generate a pseudowideband signal from narrowband speech and would help to minimize the perceived speech sound quality between wideband and narrowband calls (see [2, 3]). An additional source of delay might be the audio processor which is used to enhance the audio presentation of other audio sources in the car.

The Bluetooth® connection is the most typical link between the hands-free system and the mobile phone today. Currently, the Bluetooth® wideband specification is not yet available. In order to achieve a superior speech sound quality in conjunction with a low delay, tandem-free coding would be desirable. This would require the support of the AMR wideband transmission over the Bluetooth® link and the realization of speech coding and decoding in the hands-free system. However, an additional coder for the Bluetooth® link is in discussion. This would introduce additional distortion to the speech signal and increase significantly the overall delay in a connection. For a superior wideband service, such implementation is not desirable.

Summing up the delays assumed from Fig. 6.1, the transmission delay would be around 200 ms from car to car in the best case. Assuming an average Bluetooth® delay of about 30 ms and a fixed network delay of 50 ms, it is quite likely that the transmission delay in such a connection exceeds 400 ms.

The effect of delay in transmission systems is well known and described in ITU-T Recommendation G.131 [4] and G.107 [5]. While in [4], the impact of delay on the required echo loss is described, ITU-T Recommendation G.107 [5] gives an insight of delay on users' satisfaction. Although these investigations are still based on narrowband transmission, a similar impact can be expected in wideband systems. Figure 6.2 shows the impact of delay on user satisfaction [5], assuming ideal performance of all components in a connection except echo loss.

It can be seen that even with perfect echo loss, many users will be dissatisfied when exposed to a transmission delay of 400 ms or more. This is clearly not advisable for a superior service. But even with lower transmission delays, an excellent echo loss is required in order to achieve good users' satisfaction.

As a consequence, any component in a car hands-free system should be designed in such a way that only a minimum of delay is inserted.

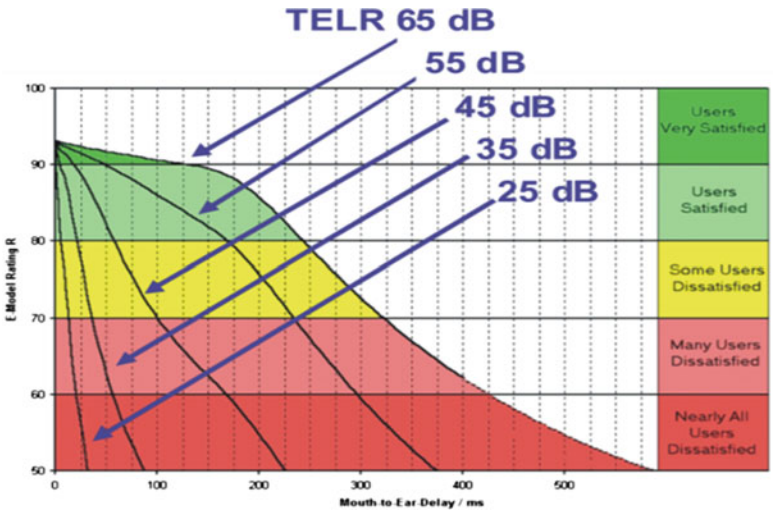



**Fig. 6.2** Users' satisfaction depending on delay and TELR (TELR = SLR + RLR + Echo Loss) from [5]

## 6.3 Listening Speech Quality

The performance requirements for the speech quality in receiving are probably easiest to fulfill due to the high quality of built-in car audio systems. For aftermarket hands-free systems, this is much more challenging. The extension of the frequency range in sending direction not only provides better representation of the low-frequency components of the transmitted speech but also increases the amount of noise transmitted by the microphone. This is of particular importance because the in-car noise is dominant in the low-frequency range. It imposes additional quality requirements on all speech enhancement techniques such as beamforming for microphones, noise cancelation, and others.

An objective measure 3QUEST according to ETSI EG 201 396-3 [6] is capable of determining the speech, noise, and overall quality, and such can be used in the optimization of wideband hands-free systems. The algorithm calculates correlation between the processed signal – typically recorded in sending direction of a hands-free system (uplink) – and two references, the original clean speech signal and the signal recorded close to the hands-free microphone. This signal consists of the near-end speech and the overlapped in-car noise. The algorithm is described in [6] and [7] in detail. Statistical analyses lead to a one-dimensional speech quality score (S-MOS), a noise quality score (N-MOS), and an overall quality score representing the general impression (G-MOS). The algorithm is narrowband and wideband capable and provides correlations in the range of $>0.91$ to the results of subjective tests.

The model was developed and trained with a certain amount of given randomized data (179 conditions). The rest of the databases were used for own validation only. During the development of the algorithm in the STF 294 project [8], the subjective S-, N-, and G-MOS results of 81 conditions remained unknown until the end of the algorithm development.

The 179 different test conditions included existing hands-free terminals and hands-free simulations in combination with different background noise scenarios such as in-car noise and outdoor road noise. The following plots show a very small amount of these data comparing subjective and objective results for the narrowband and wideband test case in hands-free conditions.

The subjective and objective results (S-MOS, N-MOS, and G-MOS) do not differ by more than 0.5 MOS in the narrowband case (see Figs. 6.3–6.5). This can be regarded as very reliable, especially when considering the complexity of this listening situation and amount of signal processing typically involved. The same can be analyzed for wideband scenarios, as shown in Figs. 6.6–6.8.

The correlation coefficient and root mean square error (RMSE) between the subjective and objective MOS data are shown in Table 6.1 for the entity of all 179 wideband test conditions.

This analysis method provides comprehensive quality scores for uplink transmission quality. It needs to be combined with further detailed parameter analyses like measurements of loudness ratings, frequency responses, signal-to-noise ratio, and others in order to provide the "whole picture" for a given implementation. Furthermore, the combination of comprehensive quality scores, on one hand, and detailed parameter analyses, on the other, may provide important hints for quality improvement and tuning, if necessary.

## 6.4 Echo Performance

Conversational aspects of wideband communication are important as well for the success of wideband services. Therefore, the requirements for conversational parameters such as double-talk capability and echo performance are to be revisited with respect to different perceptions between narrowband and wideband telephony.

As seen before, the delay plays a crucial role for echo perception. Furthermore, the extended transmission range in wideband scenarios and the spectral content of echoes strongly influence echo perception. This also demands new analysis techniques and requirements for wideband echo perception.

Current echo analyses combine various single measurements like echo attenuation or spectral echo loss and verify the compliance to requirements and tolerances. These parameters are incomplete, neither perception-oriented nor aurally adequate. They do not appropriately consider wideband-specific aspects. New investigations on wideband echo perception further point out that the spectral echo content in the frequency range between 3.1 and 5.6 kHz is especially crucial for echo disturbance [9]. New tolerances for the spectral echo attenuation have therefore been introduced in [9].


**Fig. 6.3** S-MOS, narrowband HFT

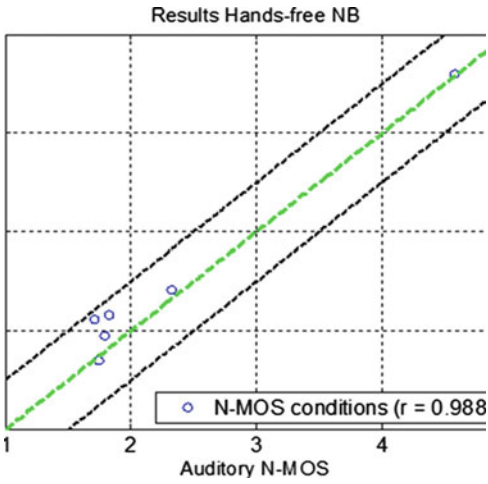



**Fig. 6.4** N-MOS, narrowband HFT

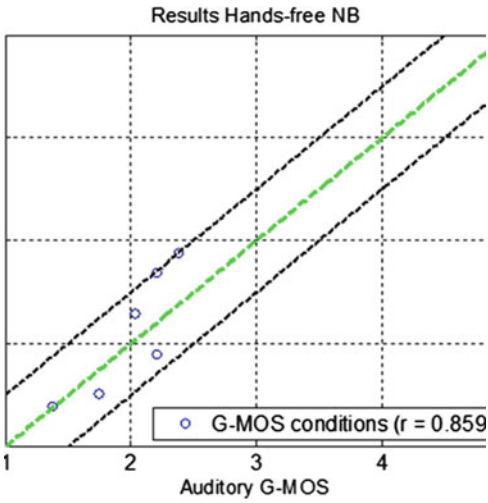



**Fig. 6.5** G-MOS, narrowband HFT

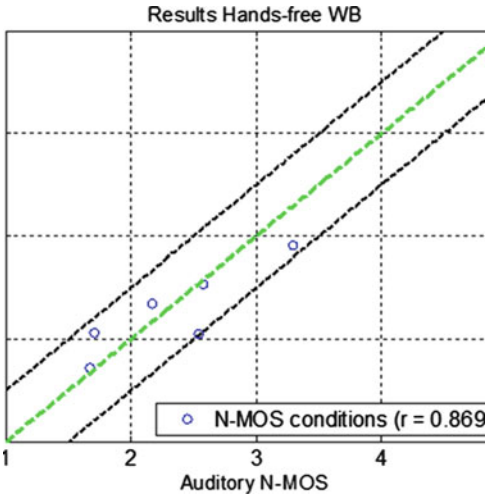



**Fig. 6.6** S-MOS, wideband HFT

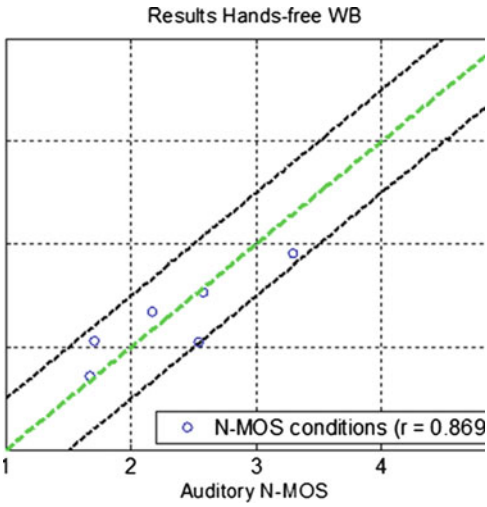



**Fig. 6.7** N-MOS, wideband HFT

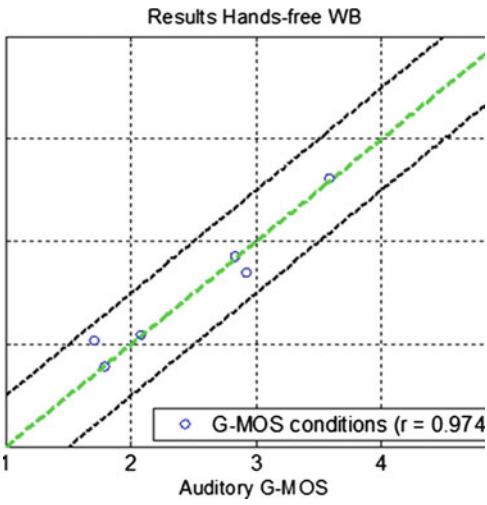



**Fig. 6.8** G-MOS, wideband HFT

**Table 6.1** Correlation and RMSE of prediction for wideband database

| | Training | | Validation | |
|---|---|---|---|---|
| | corr. | RMSE | corr. | RMSE |
| S-MOS | 91.2% | 0.37 | 93.0% | 0.33 |
| N-MOS | 94.3% | 0.27 | 92.4% | 0.32 |
| G-MOS | 94.6% | 0.25 | 93.5% | 0.28 |

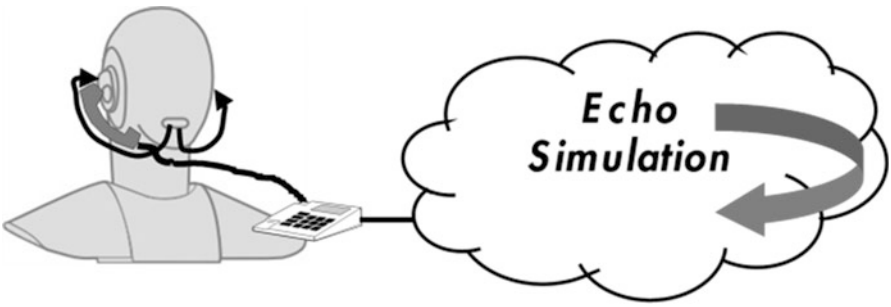



**Fig. 6.9** Principle of binaural recordings for third-party listening tests (Type A [15, 16])

A consequent next step in the field of analysis techniques is the development of an objective model providing one-dimensional values with high correlation to the MOS results from subjective tests. Models providing good correlations for echo assessment have already been evaluated for narrowband telephony, distorted sidetone, and room reverberations [10]. A new model based on the Relative Approach [11] may be applicable for narrowband and wideband telephony and may deliver hints for improvement of devices under test such as acoustic or network echo cancellers. The Relative Approach method is especially sensitive to detect unexpected temporal and spectral components and can be used as an aurally adequate analysis to assess temporal echo disturbances [12–14].

The Third-Party Listening Tests were carried out with 20 subjects in total, 14 naïve and 6 expert listeners. The speech material consists of male and female voices.

The basis for a new echo model – like for all other objective analyses – must be the subjective impression of test subjects. Therefore, subjective echo assessment tests were carried out first under wideband conditions. In principle, these tests can be conducted as so-called Talking-and-Listening Tests according to ITU-T P.831 [15] or as Third-Party Listening Tests based on artificial head recordings (ITU-T P.831, Test A [15, 16]). The principle of the recording procedure is shown in Fig. 6.9. A wideband-capable handset was simulated at the right ear of the HATS [17]. Besides the more efficient test conduction – a group of test subjects can perform the tests at the same time – the listening tests provide the advantage that the same audio files, as assessed in the subjective test, can be used for the objective analyses.

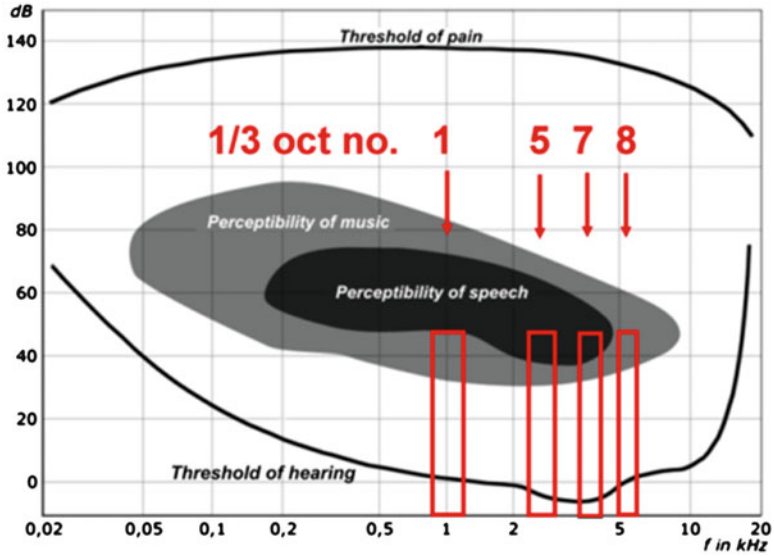



**Fig. 6.10** Filter characteristics (subset of test conditions)

A total number of 33 test conditions, including the reference scenarios (infinite echo attenuation) and different combinations of delay, echo attenuation, and spectral shaping, were included:

- Round-trip delays between 100 and 500 ms
- Echo attenuation between 35 and 55 dB
- Simulation of nonlinear residual echoes

The spectral echo content was realized by the following filter characteristics (subset of test conditions):

- NB: narrowband filter, 300–3.4 kHz
- HF1: 3.1–5.6 kHz
- HF2: 5.2–8 Hz
- 1/3 oct.no 1: 900–1,120 Hz
- 1/3 oct.no 5: 2.24–2.8 kHz
- 1/3 oct.no 7: 3.55–4.5 kHz
- 1/3 oct.no 8: 4.5–5.6 kHz

The 1/3 octave filter characteristics are shown in Fig. 6.10 together with the hearing and speech perception threshold. These filters seek a more detailed analysis of the critical frequency range between 1 and 5 kHz which provides the highest sensitivity for sound and speech perception.

A 5-point annoyance scale was used (5 points: Echo is inaudible, ..., 1 point: Echo is very annoying [18]). The stimuli were presented without pair comparison. The results were analyzed on a MOS basis together with confidence intervals based

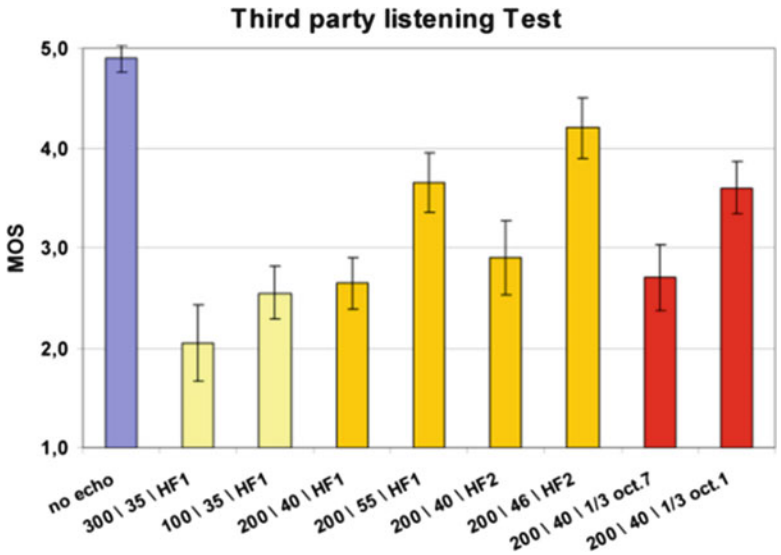



**Fig. 6.11** Subset of test results [14]

on a 95% level. Its first analysis pointed out that the quality rating for both groups (naïve, expert listeners) was very similar. The results were therefore combined.

A small subset of results from the listening-only test is shown in Fig. 6.11. The blue bar indicates the echo-free test condition. The rating of 4.8 MOS must be expected under this condition.

One example proving the importance of spectral content on echo perception is given by the red bars in Fig. 6.11. Both conditions represent a 200-ms round-trip delay in combination with a 40-dB echo attenuation. The two different filter characteristics "1/3 oct.1" and "1/3 oct.7" are introduced in Fig. 6.10. The results differ by approximately 1 MOS and point out the strong influence of spectral echo shaping on subjective assessment.

Figure 6.12 shows an example of a Δ 3D Relative Approach between the echo signal $e$ and the reference ear signal $r$. The echo signal is recorded at the artificial ear of the HATS. The reference signal $r$ represents the sidetone signal in the artificial ear as a combination of acoustical sidetone from mouth to ear and electrical sidetone via microphone and loudspeaker of a wideband-capable handset.

In the first approach, the two-dimensional mean value mΔRAe-r is calculated according to formula:

$$m\Delta RA_{e-r} = \frac{1}{KM} \sum_{k=1}^{K} \sum_{m=1}^{M} \Delta RA_{e-r}(k,\ m) \tag{6.1}$$

where $K$ = no. of freq. bands and $M$ = no. of samples per band.

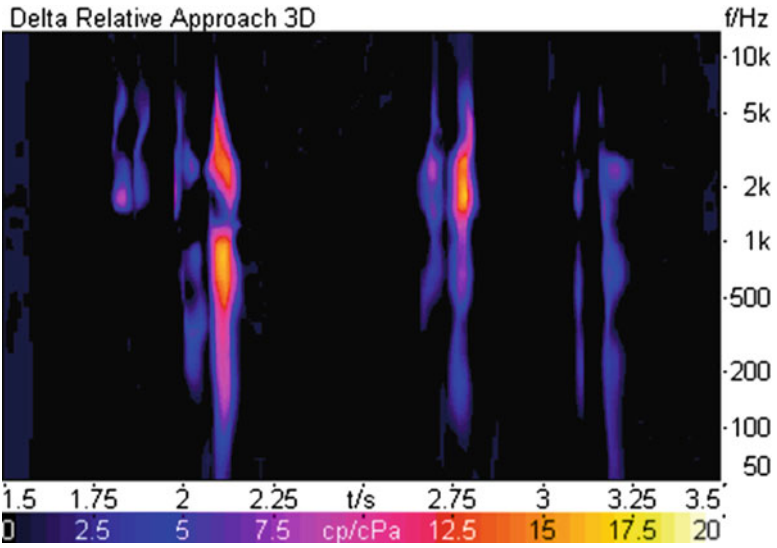



**Fig. 6.12** Δ 3D Relative Approach ΔRAe-r(t,f) between the echo signal *e* and the reference ear signal *r*

The parameters echo loss, echo delay, and mΔRAe-r are used as input signal for a linear regression in order to correlate the objective results to the subjective MOS for the echo model.

In the first step, only the two parameters echo loss and echo delay were used in the regression. The result is shown in the left-hand scatterplot in Fig. 6.13. A correlation of $r = 0.80$ is achieved, but the comparison of auditory MOS and objective MOS shows systematical errors: clusters of identical objective MOS occur in Fig. 6.13 (see arrows), which spread over a wide range of auditory MOS (between approximately 1.7 and 3.7 MOS). This can be explained by the different spectral content of these echo signals leading to significant different echo ratings in subjective tests – although the objective parameters (echo delay, echo attenuation) are identical.

The plot on the right-hand side in Fig. 6.13 shows the correlation between the auditory MOS and the objective results based only on the two-dimensional mean value *mΔRAe-r*. The correlation factor increases to $r = 0.84$. The systematical error is implicitly solved using the Relative Approach–based analysis. In principal, this could be expected because the Relative Approach considers the sensitivity of human hearing, especially for different frequency characteristics of transmitted sounds.

The combination of the three parameters *mΔRAe-r*, echo loss, and echo delay to the objective MOS further increases the correlation ($r = 0.90$). The scatterplot is shown in Fig. 6.14 (left-hand side) together with the error distribution in the right-hand picture. The residual error between objective and auditory MOS is below 0.5 MOS in 84% of test conditions.

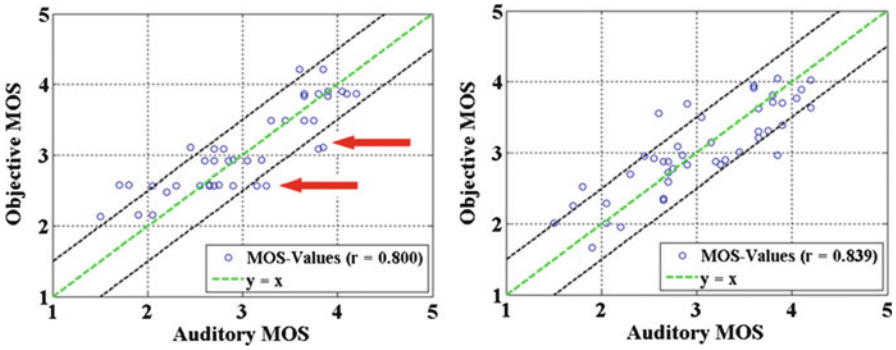



**Fig. 6.13** Objective vs. auditory MOS; *left*: input echo loss and echo delay right: *input* mΔRAe-r

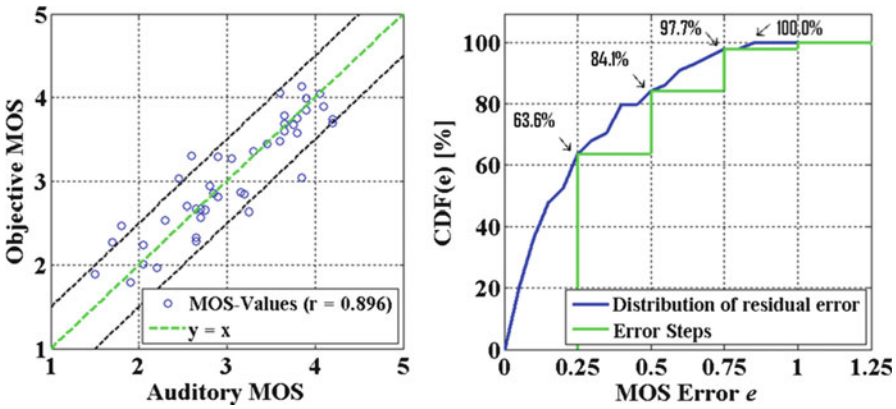



**Fig. 6.14** Objective vs. auditory MOS and residual error distribution; input parameter mΔRAe-r, echo loss, and echo delay

Next steps during the development of the echo model are the further adaptation of the Relative Approach on speech characteristics and the application of postprocessing on the resulting Δ 3D Relative Approach $\Delta RAe\text{-}r(t,f)$.

## 6.5 Conclusions

This chapter introduces several parameters critical to the success of wideband hands-free communication in cars. The impact of delay is shown and discussed. New test results and analysis techniques based on hearing model approaches are

given for the speech quality analysis in background noise as well as for echo performance.

Further work is required to derive new analysis techniques and performance criteria for double-talk in wideband systems. It is also clear that the narrowband performance requirements and testing techniques would benefit from such work.

## References

1. Hänsler E, Schmidt G (ed) (2008) Speech and audio processing in adverse environments. Springer, Berlin. ISBN:978-3-540-70601-4
2. Vary P, Jax P (2003) On artificial bandwidth extension of telephone speech. Signal Process 83:1707–1719, ISSN 0165-1684
3. Havelock D, Kuwano S, Vorländer M (eds) (2008) Handbook on signal processing in acoustics. Springer, New York. ISBN:978-0-387-77698-9
4. ITU-T Recommendation G.131 (2003). Talker echo and its control
5. ITU-T Recommendation G.107 (2008) The E-model: a computational model for use in transmission planning
6. ETSI EG 202 396-3 V.1.2.1 (2008–11) Speech processing, transmission and quality aspects (STQ); Speech quality performance in the presence of background noise; Part 3: Background noise transmission – Objective test method
7. Gierlich HW, Kettler F, Poschen S, Reimes J (2008) A new objective model for wide – and narrowband speech quality prediction in communications including background noise. In: Proceedings of the EUSIPCO 2008, Lausanne
8. STF 294 project. http://portal.etsi.org/STFs/STF_HomePages/STF294/STF294.asp
9. Poschen S, Kettler F, Raake A, Spors S (2008) Wideband echo perception. In: Proceedings of the IWAENC, Seattle
10. Appel R, Beerendts J (2002) On the quality of hearing one's own voice. JAES 50:237
11. Genuit K. (1996) Objective evaluation of acoustic quality based on a relative approach. In: Proceedings of the Inter-Noise 1996, Liverpool
12. Kettler F, Poschen S, Dyrbusch S, Rohrer N (2006) New developments in mobile phone testing. In: Proceedings of the DAGA 2008. Dresden
13. Lepage M. Evaluation of aurally-adequate analyses for assessment of interactive disturbances. Diploma thesis, IND Aachen
14. Kettler F, Lepage M, Pawig M (2009) Evaluation of aurally-adequate analyses for echo assessment. In: Proceedings of the NAG/DAGA 2009 Rotterdam
15. ITU-T Recommendation P.831 (1998) Subjective performance evaluation of network echo cancellers. International Telecommunication Union, Geneva
16. Kettler F, Gierlich HW, Diedrich E, Berger J (2001) Echobeurteilung beim Abhören von Kunstkopfaufnahmen im Vergleich zum aktiven Sprechen. In: Proceedings of the DAGA 2001, Hamburg
17. ITU-T Recommendation P.58 (1996) Head and torso simulator for telephonometry. International Telecommunication Union, Geneva
18. ITU-T Recommendation P.800 (1996) Methods for subjective determination of transmission quality. International Telecommunication Union, Geneva

# Chapter 7
# A Novel Way to Start Speech Dialogs in Cars by Talk-and-Push (TAP)

**Balázs Fodor, David Scheler, and Tim Fingscheidt**

**Abstract** The obligation to press a push-to-speak button before issuing a voice command to a speech dialog system is not only inconvenient but it also leads to decreased recognition accuracy if the user starts speaking prematurely. In this chapter, we investigate the performance of a so-called talk-and-push (TAP) system, which permits the user to begin an utterance within a certain time frame before or after pressing the button. This is achieved using a speech signal buffer in conjunction with an acoustic echo cancelation unit and a combined noise reduction and start-of-utterance detection. In comparison with a state-of-the-art system employing loudspeaker muting, the TAP system delivers significant improvements in the word error rate.

**Keywords** Acoustic echo cancellation • Frequency-domain adaptive filter (FDAF) • Noise reduction • Automatic speech recognition • In-car speech dialog • Push-to-speak

## 7.1 Introduction

Modern in-car speech dialog systems require the user to press a push-to-speak (PTS) button to initiate a dialog. The button press is normally followed by an acoustic acknowledgment tone indicating that the user may start speaking.

In practice, this procedure often causes degraded system performance due to nonconforming user behavior. For example, an inexperienced user cannot be expected to wait for the acknowledgment tone before they start speaking. Instead, the start of utterance (SOU) is likely to occur before the beep or, even worse, before

B. Fodor (✉) • D. Scheler • T. Fingscheidt
Technische Universität Braunschweig, Institute for Communications Technology, Braunschweig, Germany
e-mail: Fodor@ifn.ing.tu-bs.de; scheler@ifn.ing.tu-bs.de; fingscheidt@ifn.ing.tu-bs.de

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_7, 

the PTS button has been pressed. Similarly, even experienced users may not always conform to the required sequence simply for impatience or because they are concentrating on the driving task. As a consequence, the portion of speech uttered prematurely will not be processed by the system, resulting in recognition errors.

Another source of degradation is acoustic leaking of music or speech being presented via the car audio system into the hands-free microphone. Since the automatic speech recognition (ASR) engine generally cannot distinguish such signal components from the user's voice commands, the result will be recognition errors. In many commercial systems, this problem is approached by muting the loudspeakers upon PTS button actuation. However, muting cannot be performed instantaneously, thus leaving some disturbances in the microphone signal. Moreover, it is not always advisable to mute the loudspeaker signal. For example, the car computer may need to deliver urgent voice notifications at any time, regardless of whether the system is engaged in a speech dialog.

Instead of muting, some state-of-the-art systems employ acoustic echo cancelation (AEC) methods [1, 2], which strive to estimate and remove the acoustic signal component captured by the hands-free microphone originating from the car loudspeakers. While AEC makes muting unnecessary, this method alone still does not provide for intuitive dialog initiation. An extended and more flexible solution, the so-called talk-and-push (TAP) system, has been proposed in [3]. It allows the user to start speaking within a certain time frame before or after PTS button actuation. This is achieved by employing a look-back speech buffer in conjunction with an AEC unit and a robust SOU detection. The experiments in [3] were conducted at a sampling frequency of 8 kHz and using the normalized least-mean-square (NLMS) algorithm for AEC.

In this chapter, we investigate the performance of a TAP system operating at 16 kHz sampling frequency and employing the frequency-domain adaptive filter (FDAF) as proposed in [4] for AEC. While the higher sampling rate was chosen to open the prospect of more complex ASR tasks, the FDAF offers lower computational complexity than a 16 kHz NLMS algorithm, as well as a built-in postfilter for residual echo suppression.

The remainder of this chapter is organized as follows: Section 7.2 outlines the TAP system architecture. The implementation of the system components—AEC, noise reduction, and SOU detection—is described in Sects. 7.3 and 7.4. Section 7.5 then summarizes the experimental setup, followed by a discussion of the simulation results in Sect. 7.6.

## 7.2 The Talk-and-Push System

We assume the typical setup of an in-car speech dialog system: It consists of a speaker (e. g., the driver) seated in a vehicle, a hands-free microphone for voice control, and an in-car loudspeaker system reproducing voice prompts or music from the FM radio. In the microphone, the speaker's speech signal $s$ is disturbed by additive background noise $n$ and the reverberated loudspeaker signal $d$. In the

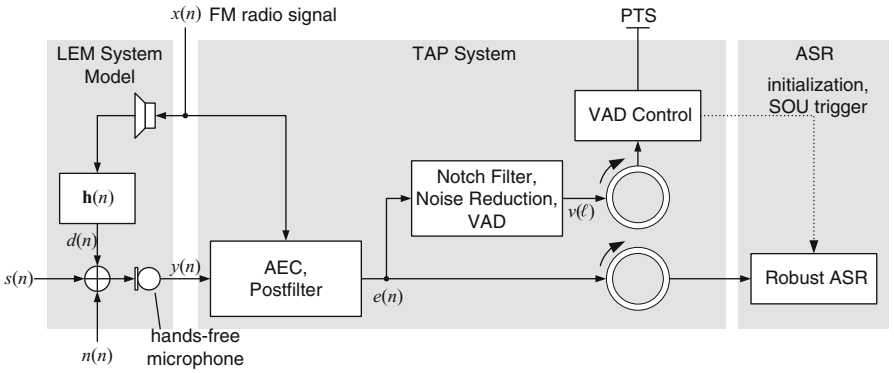



**Fig. 7.1** Block diagram of the talk-and-push (TAP) system

discrete-time domain, using $n$ as discrete-time index at sampling frequency $f_s = 16$ kHz, the microphone signal can thus be expressed as the sum:

$$y(n) = s(n) + d(n) + n(n) \tag{7.1}$$

This relation is depicted on the bottom left of Fig. 7.1.

To model the acoustic leaking from the loudspeaker into the microphone, we assume that the echo signal $d(n)$ results from the loudspeaker source signal $x(n)$ by convolution with a discrete-time, time-variant impulse response

$$\mathbf{h}(n) = [h_0(n), h_1(n), \ldots, h_{N-1}(n)]^T, \tag{7.2}$$

where $N$ denotes the finite impulse response length and $(\cdot)^T$ is the transpose.

For simplicity, a mono source signal $x(n)$ is assumed. The impulse response $\mathbf{h}(n)$ models the entire loudspeaker–enclosure–microphone (LEM) system—i. e., the path from the digital–to–analog converter before the loudspeaker via the acoustic enclosure to the analog–to–digital converter after the microphone.

Hence, the reverberated loudspeaker signal can be written as

$$d(n) = \mathbf{h}^T(n) \cdot \mathbf{x}(n), \tag{7.3}$$

where $\cdot$ denotes the scalar product and $x(n) = [x(n), x(n–1), \ldots, x(n\text{-}N + 1)]^T$ is a time-inverted segment of the loudspeaker signal of length $N$.

As shown in Fig. 7.1, the first stage of the TAP system is an acoustic echo cancelation (AEC) unit. It computes an estimate $\hat{d}(n)$ of the echo component according to [4] and subtracts it from the microphone signal.

For this purpose, the LEM system transfer function is estimated using the FDAF described in Sect. 7.3. The FDAF furthermore contains a postfilter, which reduces residual echo components as well as some background noise $n(n)$ present in the microphone signal.

The resulting error signal $e(n)$ is processed in two different branches: As shown at the bottom of Fig. 7.1, it is stored in a circular buffer to be fed into the ASR engine without further processing. In the upper branch of the TAP system, it is analyzed by an integrated additional noise reduction and voice activity detection (VAD) as described in Sect. 7.4. The latter's output is a voice activity signal which is buffered and evaluated by a control unit. Upon receiving a PTS event, the control unit locates the speech onset using buffered voice activity signal both from the past and present. The control unit also initializes and triggers the ASR engine, which is then supplied with a correct portion of the error signal from the lower buffer, depending on the detected SOU.

## 7.3 Acoustic Echo Cancelation and Postfilter

The AEC stage of our system employs the FDAF as described in [4], which unifies AEC and a postfilter for residual echo and noise suppression in the frequency domain. While most echo cancellers model the impulse response $\mathbf{h}(n)$ of the LEM system—or its transfer function—deterministically, the FDAF is based on a statistical model.

As proposed in [4], the impulse response $\mathbf{h}(n)$ is modeled as a random process with the expectation $\mathbf{h}_0(n)$ and covariance vector $\Phi_{hh}(n)$.

Actual estimation is performed in the frequency domain. Assuming that variations of the LEM path over time are gradual, the LEM system transfer function estimate $\widehat{H}_\ell(k)$ is updated recursively according to

$$\widehat{H}_{\ell+1}(k) = A\widehat{H}_\ell(k) + \Delta H_\ell(k), \tag{7.4}$$

where $\ell$ is the time frame index, $k$ is the frequency bin index, $A = 0.9995$ is the transmission factor, and $\Delta H_\ell(k)$ is the echo path update as computed according to [4].

Multiplying the estimated LEM transfer function $\widehat{H}_\ell(k)$ with a short-time Fourier transform (STFT) $X_\ell(k)$ of the loudspeaker source signal yields the estimated echo component $\hat{D}_\ell(k)$ in the short-time spectral domain. This estimate is then subtracted from the STFT $Y_\ell(k)$ of the microphone signal, resulting in an error signal $\tilde{E}_\ell(k)$. Note that before applying the STFT to the signals $x(n)$ and $y(n)$, they are subject to a high-pass filter with a cutoff frequency $f_c = 200$ Hz to remove low-frequency noise.

To reduce the noise component and to suppress the residual echo that is still present in the error signal $\tilde{E}_\ell(k)$, the FDAF includes an additional frequency-domain postfilter. Its application to the error signal yields an improved estimate of the desired speech signal as

$$E_\ell(k) = \tilde{E}_\ell(k) \times W_\ell(k), \tag{7.5}$$

where the postfilter is given by the generalized Wiener filter

$$W_\ell(k) = \frac{\Phi_{ss,\ell}(k)}{\Phi_{ss,\ell}(k) + |X_\ell(k)|^2 \times \Phi_{hh,\ell}(k) + \Phi_{nn,\ell}(k)}, \tag{7.6}$$

with $\Phi_{ss,\ell}(k)$, $\Phi_{hh,\ell}(k)$, and $\Phi_{nn,\ell}(k)$ denoting the power spectral density (PSD) of the desired speech signal $s(n)$, the echo path covariance in the frequency domain, and the PSD of the background noise $n(n)$, respectively. Since the covariance $\Phi_{hh,\ell}(k)$ can be taken as an uncertainty measure of the LEM system identification, the product $|X_\ell(k)|^2 \times \Phi_{hh,\ell}(k)$ represents the PSD of the residual echo. The PSDs $\Phi_{ss,\ell}(k)$ and $\Phi_{nn,\ell}(k)$ are estimated according to [4]. Finally, the postfilter gain $W_\ell(k)$ is floored to $W_{\min} = -12.6$ dB.

## 7.4 Integrated Noise Reduction and Voice Activity Detection

Subsequent to echo cancelation, residual vehicle noise $n(n)$ as well as some remains of the beep may still be contained in the error signal $e(n)$. In the upper path of the TAP system, robust detection of the speech onset therefore requires these disturbances to be distinguished from the desired speech component $s(n)$. This problem is here approached with a combined additional noise reduction and VAD operating on the short-time spectrum $E_\ell(k)$ of the error signal.

For the removal of the beep, all frequency bins corresponding to the frequency range from about 1.83 to 2.45 kHz are set to zero. For each frame $\ell$ and frequency bin $k$, the estimated clean speech spectrum $\hat{S}_\ell(k)$ is obtained from the error signal $E_\ell(k)$ by applying a Wiener filter based on the a priori signal-to-noise ratio (SNR) as described in [5] and [6]. For the computation of this SNR, the power spectral density of the noise is estimated by employing a 3-state time- and frequency-dependent VAD [3].

The output of the VAD is transformed into a per-frame voice activity signal $v \in [0, 1]$ by averaging over relevant frequency bins (see [3]) and then stored in the upper circular buffer as shown in Fig. 7.1. The final decision about the time of the speech onset is made by the VAD control unit: The hypothesized speech onset frame $\ell_{\text{SOU}}$ is the latest nonspeech frame (i. e., $v(\ell_{\text{SOU}}) \approx 0$) before $v(\ell)$ exceeds an empirical threshold [3].

## 7.5 Experimental Setup

For experimental evaluation, we performed an offline batch simulation of the TAP system using the Cambridge Hidden Markov Model Toolkit (HTK) for ASR. Instead of a physical LEM system, we used a digital LEM impulse response measured inside a vehicle. In the next two subsections, the near-end speech files as well as the noise and echo signals are described.

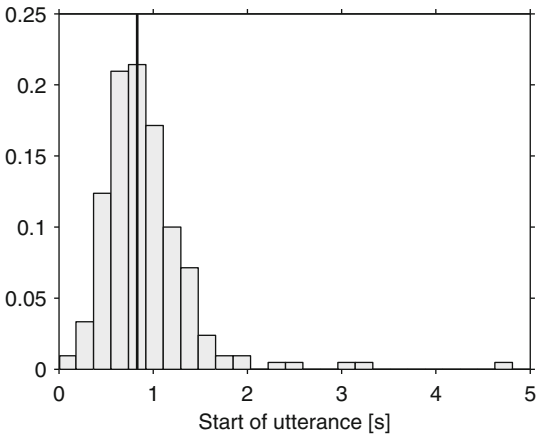



**Fig. 7.2** Normalized histogram of the start of utterance (SOU) with respect to the beginning of the speech file; the thick black line marks the median of 0.83 s [3]

For reference, we performed a similar experiment where the TAP system is replaced by the state of the art: Upon PTS button actuation, the in-car audio system is muted—i. e., no echo component is added at the microphone—and the unprocessed microphone signal is passed to the ASR engine. Any speech parts preceding the PTS event are discarded because there are no look-back buffers.

### *7.5.1 Test Speech Data*

The test speech data consisted of a subset of the US-English SpeechDat-Car connected-digit corpus [7]. The set comprised 210 utterances spoken by 35 speakers, each utterance containing four to sixteen digits. Since the test files were artificially degraded with background noise (see next section), we used close-talk recordings only, which approximately represent clean speech.

As described in [3], PTS actuation was assumed to occur 0.83 s relative to the beginning of each test speech file. Since the actual time of the speech onset varied from file to file, a probabilistic displacement of the SOU with respect to the PTS event was achieved. The histogram in Fig. 7.2, which was generated by forced Viterbi alignment, visualizes the distribution of the speech onset we found in the test speech files. By assuming the PTS event at the median of the SOUs, both premature and delayed speech were simulated.

### *7.5.2 Artificial Degradation with Echo and Noise*

We used different loudspeaker source signals to excite the LEM system as well as a set of vehicle noise files to simulate the disturbance of the desired speech on the microphone. Two different simulations were performed: In one case, the loudspeaker

signal $x(n)$ contained only music, which was randomly chosen from six files of varying musical styles. In the other case, $x(n)$ consisted of speech files, which were randomly chosen from 96 speech files taken from the English subset of the NTT-AT Multilingual database; the files were spoken by four female and four male speakers. In addition, a beep signal at 2.1–2.4 kHz was added to all loudspeaker source signals 0.25 s after the virtual PTS event. In the baseline reference case, however, no beep was added because we assumed strict muting of the loudspeakers. To obtain the simulated echo signals $d(n)$, the loudspeaker source signals were convolved with a time-invariant LEM system impulse response measured in a Volkswagen Passat car type.

For simulating the background noise component $n(n)$, four different vehicle noise files recorded in two different cars at two different velocities were used randomly.

Noise and echo components were added to the test speech signals at different signal-to-noise ratios (SNRs) and signal-to-echo ratios (SERs), respectively. By this means, we were able to investigate the system behavior under varying disturbance conditions. As in [3], we performed the SNR and SER adjustment based on the active speech level (ASL) according to ITU-T recommendation P.56 [8]. However, all signals were subject to a 50–7,000 Hz band-pass filter prior to the P.56 level measurement to eliminate speech-irrelevant frequency components.

### *7.5.3 Automatic Speech Recognition Setup*

The ASR experiments were conducted using a feature extraction frontend for mel-frequency cepstral coefficients (MFCCs) and a set of hidden Markov models (HMMs) trained on American English connected-digit strings.

The frontend settings were as follows: A pre-emphasis value of 0.9, a frame shift of 10 ms, a frame length of 25.6 ms, a Hamming window, and a 512-point FFT. No noise reduction was applied in the frontend, but the HMMs were trained on recordings containing slight vehicle noise. For each frame, twelve MFCCs (without the zeroth coefficient) were computed using 26 uniform, triangular filterbank channels on the mel scale and ignoring frequencies below 50 Hz and above 7 kHz. A log energy coefficient as well as first and second order time derivatives were appended. Cepstral mean normalization was performed separately for each utterance.

For acoustic modeling, we employed 42 tied-state HMMs representing acoustic–phonetic units, differentiating also by the immediate left and right context via triphone modeling within words. Each HMM consisted of one to three emitting states, each of which was assigned a continuous output probability density function modeled by a Gaussian mixture model with 32 components each. Diagonal covariance matrices were assumed. The training material consisted of 3,325 utterances spoken by 245 speakers and was taken from the connected-digit corpus of the US-English SpeechDat-Car database [7]; to ensure speaker independence, two disjunct sets of speakers were used for training and testing.

Recognizing the undegraded set of test utterances with the trained HMM set yielded a word error rate (WER) of 0.59%, which posed a lower bound to the remaining recognition experiments.

## 7.6 Results

Our experimental results are summarized in Table 7.1, which lists the obtained WERs in % for different disturbance conditions. In case (a), the echo signal was music, whereas in case (b), the echo signal was speech. For reference, the lines labeled "Muting" contain the results obtained with the baseline system. Since this system was assumed to mute the car loudspeakers instantly upon receiving a PTS event, its performance is independent of echo type and SER. Note that the baseline results must be interpreted with care as they strongly depend on the timing of the PTS event relative with the SOU. If, in practice, more speakers than the assumed 50% start speaking *after* PTS actuation, better baseline performance will result. Nevertheless, an actual state-of-the-art system may suffer from additional impediments not considered here: For example, the muting of the loudspeakers will occur with additional delay; moreover, the beep would not be omitted in practice.

The results in Table 7.1 show that the TAP system outperforms the reference system under all test conditions. In the absence of noise SNR $\rightarrow \infty$, the TAP system yields WERs of 0.73–2.29%, which is much closer to the limit of 0.59% than the 4.20% WER obtained in the reference case. Moreover, the dependence on the SER is negligible for SER $< \infty$, indicating that the AEC works reliably even when there is noise. This seems to be a major advantage over the NLMS algorithm when considering the results obtained in [3] and might be attributed to the residual echo

**Table 7.1** WER in % achieved with the TAP system under different SNR and SER conditions. For comparison, the performance of a state-of-the-art system employing muting is included

| | | SNR [dB] | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | −5 | 0 | 5 | 10 | 15 | 20 | $\infty$ |
| | Muting | 73.41 | 37.90 | 14.93 | 7.17 | 5.02 | 4.54 | 4.20 |
| **(a) Echo signal is music** | | | | | | | | |
| | 0 | 43.22 | 22.83 | 10.29 | 5.02 | 2.88 | 2.24 | 1.90 |
| SER [dB] | 5 | 42.83 | 22.83 | 10.44 | 4.83 | 2.98 | 2.34 | 1.95 |
| | 10 | 42.73 | 22.49 | 10.59 | 4.88 | 2.88 | 2.29 | 1.95 |
| | $\infty$ | 43.85 | 24.63 | 11.71 | 6.10 | 3.27 | 2.68 | 0.73 |
| **(b) Echo signal is speech** | | | | | | | | |
| | 0 | 43.02 | 22.39 | 10.63 | 5.32 | 3.17 | 2.39 | 2.29 |
| SER [dB] | 5 | 43.46 | 22.39 | 10.68 | 4.88 | 3.02 | 2.34 | 2.10 |
| | 10 | 42.98 | 22.54 | 10.78 | 5.12 | 2.93 | 2.20 | 2.49 |
| | $\infty$ | 43.85 | 24.63 | 11.71 | 6.10 | 3.27 | 2.68 | 0.73 |

suppression of the postfilter. However, the TAP system exhibits decreased performance when there is background noise but no echo signal (SNR $< \infty$, SER $\rightarrow \infty$); this may indicate that in the absence of LEM excitation, the operation of the postfilter is suboptimal.

When judging the SNR dependence of the TAP system, note the following: Since the test speech files were close-talk recordings made in a vehicle environment, they are not entirely clean with respect to background noise. As a consequence, the SNR values shown in Table 7.1 are biased towards higher values as they only reflect the amount of noise added artificially.

## 7.7 Conclusion

We have investigated the performance of a so-called TAP system, which tolerates imperfect user behavior when initiating a speech dialog. As in [3], we have demonstrated that the TAP system significantly improves recognition performance assuming that half of the users actuate the push-to-speak button shortly after they start speaking. This is achieved by means of two synchronized circular buffers providing a look-back capability and a robust speech onset detection. We have included an AEC and noise reduction unit operating in the frequency domain to eliminate loudspeaker signal as well as background noise leaking into the microphone. Further investigations will include AEC for multichannel source signals as well as improved methods to measure the SNR and SER. In addition, more complex ASR tasks will be evaluated using the TAP system.

## References

1. Shozakai M, Nakamura S, Shikano K (1998) Robust speech recognition in car environments. In: Proceedings of ICASSP'98, Seattle, pp 269–272
2. Matassoni M, Omologo M, Zieger C (2003). Experiments of in-car audio compensation for hands-free speech recognition. In: 2003 IEEE workshop on automatic speech recognition and understanding. pp 369–374
3. Fodor B, Scheler D, Suhadi S, Fingscheidt T (2009) Talk-and-Push (TAP) – towards more natural speech dialog initiation. In: AES 36th international conference, Dearborn
4. Enzner G, Vary P (2006) Frequency-domain adaptive Kalman filter for acoustic echo control in hands-free telephones. Signal Process 86(6):1140–1156, Elsevier
5. Scalart P, Filho J (1996) Speech enhancement based on a priori signal to noise estimation. In: Proceedings of ICASSP 1996, Atlanta, pp 629–632
6. Ephraim Y, Malah D (1984) Speech enhancement using a inimum Mean-square Error Short-time Spectral Amplitude Estimator. IEEE Trans Acoustics Speech Signal Process 32(6):1109–1121
7. Moreno A, Lindberg B, Draxler C, Richard G, Choukri K, Euler S, Allen J (2000) SpeechDat-Car: a large database for automotive environments. In: Proceedings of LREC 2000, Athens
8. International Telecommunication Union (1993) ITU-T recommendation P.56

# Chapter 8
# Cognitive Dialog Systems for Dynamic Environments: Progress and Challenges

**Felix Putze and Tanja Schultz**

**Abstract** In this chapter, we present our existing setup and ongoing research on the development of cognitive dialog systems for dynamic environments like cars, including the main components that we consider necessary to build dialog systems to estimate the user's mental processes (hence, cognitive) and adapt their behavior accordingly. In conducting realistic testing and recording environment to produce real-life data, a realistic driving simulator was used. We also needed to observe the user during these interactions in a multimodal way to estimate the current user state based on this data. This information is integrated with cognitive modeling components that enrich the observational data. We finally needed a dialog management system which is able to use this information for adapting its interaction behavior accordingly. In this chapter, we report our progress in building these components, give an overview over the challenges we identified during this work and the solutions we aim for.

**Keywords** Cognitive dialog system • Cognitive model • Human machine interaction • User state detection

## 8.1 Introduction

Spoken dialog systems have matured to a point where they find their way to many real-world applications. However, their application in very dynamic scenarios remains an open and very interesting task. Spoken dialog systems as an interface for in-car services are very desirable and at the same time very challenging. On one hand, they offer eyes-free and hands-free control without visual or manual distraction from the primary driving task. On the other hand, this task uses the user's cognitive capacity, so

F. Putze (✉) • T. Schultz
Cognitive Systems Lab, University of Karlsruhe, Karlsruhe, Germany
e-mail: felix.putze@kit.edu; tanja@ira.uka.de

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_8, 

we can no longer assume to deal with a fully attentive and perfect interaction partner as in more static environments. Another important aspect is the adaptation to individual preferences. As dialog sessions in driving scenarios may last for several hours, we have to take into account both changing user states, i.e., cognitive workload or emotions, as well as lasting user traits, e.g., gender or personality. Both types of individual differences influence the optimal interaction behavior which the system should use for maximizing user satisfaction, as user studies like [1] show. There is potential for a large range of adaptation measures: One example is reacting to increased cognitive workload by taking the initiative from the user, delaying noncritical information, or reducing its complexity. Another one is adjusting the system to the user's emotional state and personality by selecting appropriate wording, voice, and turn-taking behavior. We propose to use systematic multimodal observation and state classification of the user derived from a variety of different biosignals. This metadata is augmented with a more detailed model-based representation of the user's mental processes and helps to select appropriate adaptation measures. Combining a global model of the user's cognition and affective states for the purpose of building adaptive interaction strategies is new to the field of spoken in-car dialog systems.

After a review of related work, the following sections describe all components which are necessary to develop and evaluate cognitive interaction systems for in-car applications: a driving simulator to create a realistic environment for recordings, an interaction system as a platform for human–machine interaction, a recording setup to collect data for training and testing of systems, a recording software to deal with the challenges of multiple input streams, and a user state detection framework and components to model human cognition.

## 8.2 Related Work

In the last years, many approaches for user models for application in adaptive in-car dialog systems exist. Like [2], most of them rely on heuristics and indirect user state detection.

The authors of [3] describe a dialog system that bases its handcrafted dialog strategy for a gaming interface on the user's emotional state, derived from prosody, language, and visual features. Together with the history of interaction, the current user command, and other discourse features, the user state can be accessed by the dialog strategy in the form of a decision tree.

Fatma Nasoz and Christin Lisetti [4] describe a user-modeling approach for an intelligent driving assistant. This model is based on a Bayesian network which allows to derive the most useful system action (in terms of driving safety) given the estimated driver state, which consists of emotional state, personality, and other features and is partially derived from physiological measurements like the user's heart rate. The score for each action is calculated using a utility node which measures the probability of safety improvement given the current user state. Similar decision-theoretic, user-model-based action evaluation approaches are used in [5], which

also include an active sensor selection mechanism. Cristina Conati [6] presents an educational dialog system that can decide for different user assistance options, given the user's emotional state (derived from different modalities). This work bases its network on the cognitive OCC (by Ortony, Clore, and Collins) appraisal theory, which relates the users' emotions with their goals and expectations.

In the area of user state detection from biosignals, Liang, Reyes, and Lee [7] developed a real-time workload classifier in the car using facial features, like pupil diameter or gaze direction, extracted from videos of the driver. The ten participants followed a car with varying speed while performing a secondary memory and comparison task. Using support vector machines, the authors achieved a recognition rate of 81.1% on average for the recognition of cognitive workload. Healey and Picard [8] developed a classifier to monitor the stress levels in daily life car-driving tasks. They collected data from 24 real-life drives of at least 50-min duration and used the biosignals electromyography, electrocardiography, and skin conductance for their system. Linear discriminant analysis (LDA) was used for dimensionality reduction, and a classifier using a linear decision function was able to discriminate the three classes with accuracies of 100% (low workload), 94.7% (medium workload), and 97.4% (high workload).

## 8.3 Driving Simulator

Testing and evaluation of different interaction strategies requires a realistic experimental environment which reproduces all important effects and distractions seen in real-life applications. While recording in a real car in real traffic situations creates the most authentic sessions, the downsides of this approach are safety concerns with early prototypes, the lack of reproducibility, and the missing ability of reliably provoking scenarios which are relevant for the current investigation. Therefore, we decided to build a driving simulator which is designed to create a realistic driving experience. The main focus was not to build a physically correct car test bed but to simulate the most important influences and distractions that occur during real driving tasks, especially in situations where the application of a dialog system plays an important role. We based our driving simulator on a real car and kept the interior fully intact and functional to provide a realistic in-car feeling. The car is surrounded by a projection wall, covering the view of the frontal and lateral windows. The simulator features acoustic feedback via engine sound and environmental surround sound and haptic feedback in the seat (via tactile transducers) and steering wheel (via force feedback).

The simulator software is based on a modified gaming engine[1]. It was extended using a multiscreen display, steering wheel support, and simple ambient traffic control.

[1] MTA:SA: http://www.mtasa.com

**Fig. 8.1** The CSL driving simulator in action

Its support for scripting scenarios in LUA allows us to configure individual driving stages: We can position the driver in a wide artificial environment with realistic urban and rural areas, where we define a route represented by navigation directions for the system. It is possible to trigger events at defined points to generate specific traffic situations, position new elements in the environment, or influence the position or driving characteristics of the car (Fig. 8.1).

## 8.4 Interaction Setup

While the user is driving, they interact (via close-talking microphone to reduce noise) with a dialog system. In our current scenario, this constitutes a virtual co-driver which acts as interactive tour guide and navigation system for the virtual environment. To investigate the phenomena we are interested in, e.g., different levels of workload, we created several scenarios specially designed for studying man–machine interaction. This includes the handling of a variety of secondary tasks, urban and rural routes, and several triggered events.

The virtual co-driver is present on a screen in the cockpit on which it is displayed using the ThinkingHead[2], a morphable 3D avatar, and is equipped with a grammar-based speech recognition system and a speech synthesis component to vocally communicate with the driver. The co-driver is driven by a lightweight interaction manager which was designed especially for the purpose of adaptive dialog systems. The interaction manager uses a rule-based engine which executes one or more rules with preconditions that match the current interaction state, according to the

[2] http://thinkinghead.edu.au

Information State Update paradigm [9]. The interaction state also comprises variables that describe the detected user state to allow adaptive selection of speech acts based on the user's current situation.

The system is also able to switch its behavior between different styles for the realization of one selected speech act, depending on the user's state. Different behavior styles can change the processing of speech acts in many aspects. For example, the content of a speech act realization can differ in its length and complexity based on the user's workload. It is also possible to adjust the speaking speed, the volume of the voice, and the stress of certain key phrases according to this parameter. Using those parameters, the co-driver realizes a verbose, chatty, and entertaining behavior if it detects a state of low cognitive workload. It presents much information, tells occasional jokes, and shows expressive mimic. For situations with high cognitive workload, the co-driver switches to a different, more concise, and unobtrusive behavior to use the limited available cognitive resources for the transmission of the most critical information. In this style, the system also takes more initiative in the interaction, taking most noncritical decisions from the user.

A user study [10] showed that a behavior which adapts to the changing user's cognitive load is both more efficient and also more satisfying for the user than a nonadaptive one. By changing the information throughput depending on the workload level, the system can optimally use the available cognitive resources of the user without risking overload. This behavior was evaluated as empathic and desirable by the users in a satisfaction questionnaire. It is therefore critical for a cognitive interaction system to provide this kind of adaptation.

## 8.5 Recording Setup

During the interaction, we employ a variety of signals to observe the user in the car. This is done for multiple reasons. First, an adaptive dialog system needs online data streams from which it can extract meaningful features describing the user's state. Second, to train automatic recognizers that perform this user state classification, we need to provide large amounts of labeled training data. To that end, we installed multiple biosignal sensors in the car to get a reliable, continuous data stream without obstructing or distracting the user too much.

We employ the following equipment to observe the user:

- Small cameras to record videos of the face and the upper body of the driver to catch facial expressions and body pose.
- A close-talking microphone to record the user's utterances
- Brain activity is measured using electroencephalography (EEG) with one of two possible alternatives:
  - A 16-electrode EEG cap with active electrodes for optimal signal quality and coverage of all brain regions
  - A 14-electrode gaming device (Epoc Emotiv) with saline electrodes for increased usability and reduced setup time

- A light sensor glove which measures skin conductance and heart rate
- A respiration belt on top of the clothes to measure respiration frequency
- Two facial electromyography (EMG) electrodes to record facial activity which is not captured by the cameras.

The last three items all use the same recording interface and are either attached to a universal signal recorder[3] or directly connected via Bluetooth, which reduces obstruction to a minimum. In addition, we employ indirect motion monitoring by continuously recording the angle of the steering wheel and the acceleration and brake pedals in the car.

In this recording setup, we already collected more than 100 interaction sessions in the tour guide scenario, interacting with a virtual co-driver controlled by a human wizard. Each interaction session comes with a collection of recorded biosignals, a manual transcription, and the results of several questionnaires on user personality, satisfaction, and task performance. This large collection allows a systematic investigation of interaction behavior under changing workload conditions.

## 8.6 Recording Software

Metadata extraction for dynamic dialog systems is required to work in real time. To this end, we need to record multiple biosignal streams in a robust, fast, and convenient way, offering interfaces to read data from very different signal sources and output it to very different receivers like recognizers or visualization components. To fulfill all requirements, we developed a new recording software called *BiosignalsStudio* [11]. BiosignalsStudio is designed in a modular fashion and allows to connect arbitrary input modules for data collection from a specific device with arbitrary output modules which write data to files, visualize the data, or send it to an external recognizer software via sockets. All modules share a common generic data format which stores multiple data channels and a meta-information block which contains the sampling frequency, detected errors, etc. Each module can be connected to several receivers, allowing data from one source to be stored to disk and visualized in parallel. There exists a number of intermediate modules which can be installed between input and output modules to augment, filter, or transform the data. Currently, input modules for all connected biosignal recording devices and several others (like gyro and acceleration sensors) are available (Fig. 8.2).

As we operate with very different and asynchronous data streams, it is important to store timestamps with each data block to ensure that only data which belongs together is merged in the multimodal fusion of the recognition engine. These timestamps are

[3] Varioport, Becker MediTec

**Fig. 8.2** (Part of) the recording setup with EEG cap, audio headset, and sensor glove in the driving simulator

generated at the earliest point possible which is usually when receiving the data block from the hardware interface (some devices are able to generate hardware timestamps, which is preferable). Timestamps within blocks of data are linearly interpolated. They are stored together with one data file for each modality and detailed log files in one directory per session, allowing easy and standardized access for all components, regardless of the specific recording setup. For distributed recording on multiple machines, timestamps are automatically synchronized via the NTP protocol. In this situation, the software is also able to remotely control the recording from one machine which starts and monitors the recording on the others.

## 8.7 User State Detection

The collected biosignal streams are passed on to a generic biosignal classification framework that performs the following steps. First, the data is filtered and cleaned to remove technical and physiological artifacts from the signals. For this purpose, we employ several source separation techniques, e.g., independent component analysis (ICA) to remove eyeblinks from the EEG signal or canonical correlation regression (CCR) to deal with EMG artifacts. From the cleaned signals, we then calculate features to describe them. Features are extracted on overlapping windows of varying length, depending on the signal type and on characteristics of the user state in question. For the biosignals, we extract features both from the time and the frequency domain. Typical time domain features are mean, variance, or zero-crossing rate, calculated on the raw feature or on the first or second derivative. Frequency domain features are especially relevant for EEG signals. Classical features here describe the band power in the $\alpha$-,$\beta$-,$\gamma$-,$\delta$-, and $\theta$-bands

(see [10, 13] for details), but other features, e.g., derived using the Wavelet transform, are also available.

For the speech signal recorded during the interaction, we use the software Praat[4] to extract prosodic features like pitch, jitter, or shimmer from the user's voice. To capture linguistic features, we use the Linguistic Inquiry and Word Count[5] that categorizes each word in its vocabulary in one or more groups, e.g., "negative emotional word" or "self reference." Active Appearance Models [12] are used to capture information on the facial expression and activity of the user as recorded by the camera in the car.

To arrive at a person-independent system, features are normalized using range normalization or z-normalization. The normalization statistics are calculated on additional holdout data which is not used for other steps of training and evaluation. This kind of data can also be collected in an unsupervised fashion as enrolment data to bootstrap the system for a new user.

As we generate a very large initial feature set, we employ Forward Feature Selection during the training step to reduce the dimensionality of the feature space, preceded by a correlation-based filtering to reduce the runtime of the selection.

For classification, the final feature vectors are then passed into a statistical classifier of which multiple variants are available, e.g., a support vector machine (SVM) using Radial Basis Function kernels or a classifier based on linear discriminant analysis (LDA). More exactly, there is one classifier for each modality as this allows dynamic weighting of input channels, e.g., to account for noise or defective sensors. To arrive at a final classification result, the output of all classifiers is combined using majority voting.

One important application for user state detection is the recognition of cognitive workload from multiple biosignals. In a large evaluation, we developed a user-independent classification system that discriminates between low and high workload. For each participant, we record a number of different sessions in a driving scenario. Relaxing phases or simple driving tasks are labeled as low workload while driving sections with different secondary tasks (visual and auditive cognitive tests) are labeled as high workload sessions. From a prestudy [13], we know, from evaluation of subjective workload using the NASA TLX questionnaire, that this assignment corresponds to experienced load levels. Figure 8.3 summarizes the recognition rates achieved in the evaluation using a cross-validation scheme to classify data from relaxing phases and high workload phases induced by driving with secondary task. We see that a person-independent discrimination of the two conditions is possible, and that a decision fusion approach yields the best results.

[4] http://www.praat.org

[5] http://www.liwc.net

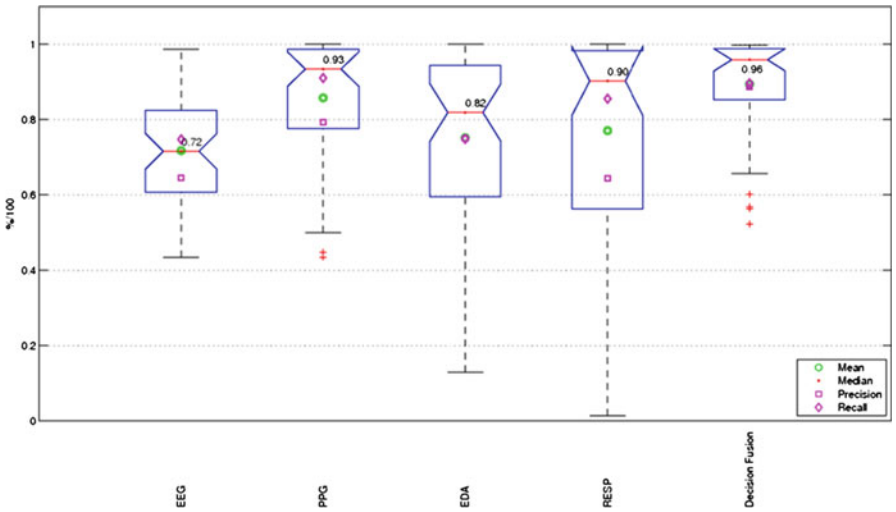



**Fig. 8.3** Recognition rates of a multimodal biosignal classifier for discriminating two conditions of low and high workload in a driving scenario. We show recognition rates for EEG, photoplethysmography (PPG), skin conductance (EDA), respiration (RESP), and decision fusion

## 8.8 Cognitive Modeling

Cognitive architectures like ACT-R [14] aim to provide a general model of human cognition for simulation or prediction. For the use in adaptive dialog systems, they help to represent and estimate nonobservable user states and are also able to predict future user behavior from a given state. This is very useful for two purposes. On one hand, a cognitive model can support the empirical, biosignal-based classification of user states by proving information derived from the evaluation of more formal models of cognition which are backed with a priori knowledge from psychology and cognitive science. On the other hand, a cognitive model is able to simulate human behavior in situations where no real user is available; this is a typical use case in evaluation and training situations in early phases of the development of a new system.

As a first cognitive modeling component, we implement a memory and interest model to represent the user's activation of, and interest in, the actual and potential discourse items. Our focus here is to reflect the fact that the user cannot remember all discourse items correctly and with the same intensity. This is of special importance in situations where the dialog system interrupts an ongoing dialog for more important information or during time-critical situations.

The memory model represents for each time slice an activation value for each possible discourse item in the domain ontology, including relations between those items. The activation determines how present each item currently is in the user's memory and how it can be used to derive the chance of successful retrieval of this

item and the time necessary to perform such a retrieval process. We based our system on the connectionist approach presented in the LTM$^c$ model [15], which was proposed to solve some issues with the memory model of ACT-R. Here, each item is represented as a node, connected with edges to other items that are semantically, linguistically, or hierarchically related. These edges are used to spread activation between nodes when one becomes activated, e.g., through a system speech act. We also extended the LTM$^c$ model to better reflect the dynamics of a memory system which is important to model topic switches in an interaction.

The interest model reflects the user's current interest in each item. This is a dynamic variable that depends not only on the situational context (spatial proximity, expressed interest) but also on more general, static factors. To represent this variety of influences, we employ a Bayesian network for the interest model.

Both models are currently used to determine a general value of importance of giving additional information to the user. This value allows us to weigh the speech act of information presentation against other goals like navigation pointers or entertainment. We do this by summing up the negative activation for all items, weighed by the interest values of each item. This score is called *competence urge*, based on the more general concept of urges that describe the needs of an individual and that influence its emotions and actions [16]. This score is also used to determine the items the system will present next to the user as they maximize the reduction of the competence urge.

In a user study in the tour guide scenario [17], we showed that it is possible to simulate plausible interactions using cognitive models. Utterances of the user where generated using the memory model which was stimulated from the perception of external stimuli and queried for the most highly activated items. The system in those simulations generated its utterances using its own model of the user's memory with a similar structure than the generative model, but separate activation scores to track what is going on in the user's mind. The behavior of both the system and the simulated user was learned in a Reinforcement Learning-based manner, using the urge mechanism to weight the goals of the agents. The generated interactions were played back to human judges and perceived as similar to a handcrafted gold standard and as significantly better than the baseline behavior.

Future applications for the memory model comprise its influence on the user understanding model, by making the chance for misunderstandings dependent on the activation level of the relevant items and the application for coherent user simulation in evaluation and training of interaction strategies.

## 8.9 Conclusions

The development of flexible, generic, and natural adaptation mechanisms for cognitive interaction systems has seen great progress as reported in this chapter. We implemented and tested a realistic driving simulator which will allow a large number of experiments under controlled but nevertheless authentic conditions.

We presented an adaptive dialog system that can change its behavior depending on the state of its user. We have implemented a framework of biosignal recording components and statistical classifiers that are able to determine the user's current state, for example his cognitive workload. We investigate cognitive modeling architectures to structure the user's adversarial desires and to model the user's memory. The next step will bring all components together to create a system which uses both biosignal-based user state detection and predictive models to a dialog strategy which can adapt flexibly to changes in the user's state.

## References

1. Nass C et al (2005) Improving automotive safety by pairing driver emotion and car voice emotion. In: Proceedings of CHI, Oregon
2. Hassel L, Hagen E (2006) Adaptation of an automotive dialogue system to users expertise and evaluation of the system. Lang Res Eval 40(1):67–85
3. Gnjatović M, Rösner D (2008) Emotion adaptive dialogue management in human-machine interaction: adaptive dialogue management in the NIMITEK prototype system. In: 19th European meetings on cybernetics and systems research, University of Vienna, Vienna
4. Nasoz F, Lisetti C (2007) Affective user modeling for adaptive intelligent user interfaces. In: Proceedings of 12th HCI international conference, Beijing
5. Li X, Ji Q (2005) Active affective state detection and user assistance with dynamic Bayesian networks. IEEE Trans Syst Man Cybern 35:93
6. Conati C (2002) Probabilistic assessment of user's emotions during the interaction with educational games. Appl Artif Intell 16:555–575
7. Liang Y et al (2007) Real-time detection of driver cognitive distraction using support vector machines. IEEE Trans Intell Transp Syst 8(2):340–350
8. Healey J, Picard R (2005) Detecting stress during real-world driving tasks using physiological sensors. IEEE Trans Intell Transp Syst 6(2):156–166
9. Larsson S, Traum DR (2002) Information state and dialogue management in the TRINDI dialogue move engine toolkit. Nat Lang Eng 6:3–4
10. Heger D, Putze F, Schultz T (2010) An adaptive information system for an empathic robot using EEG data. In: 2nd international conference on social robotics, Singapore
11. Heger D, Putze F, Amma C, Wielatt T, Plotkin I, Wand M, Schultz T (2010) Biosignals studio: a flexible framework for biosignal capturing and processing. In: 33 rd annual German conference on artificial intelligence 2010, Karlsruhe
12. Gao H (2009) Robust face alignment for face retrieval. University of Karlsruhe (TH), Karlsruhe
13. Putze F, Jarvis J-P, Schultz T (2010) Multimodal recognition of cognitive workload for multitasking in the car. In: 20th international conference on pattern recognition, Istanbul
14. Anderson J et al (2004) An integrated theory of the mind. Psychol Rev 111:1036–1060
15. Schultheis H et al (2006) LTM-C – an improved long-term emory for cognitive architectures In: Proceedings of the 7th international conference on cognitive modeling, Trieste
16. Bach J (2003) The micropsi agent architecture. In: Proceedings of international conference on cognitive modeling, Bamberg
17. Putze F, Schultz T (2009) Cognitive memory modeling for interactive systems in dynamic environments. In: Proceedings of 1st international workshop on spoken dialog systems, Kloster Irsee, 2009
18. Waibel A et al (2001) A one pass-decoder based on polymorphic linguistic context assignment. In: Proceedings of ASRU, Trento

# Chapter 9
# In-Vehicle Speech and Noise Corpora

**Nitish Krishnamurthy, Rosarita Lubag, and John H.L. Hansen**

**Abstract** As in-vehicle speech systems become prevalent, there is a need for specific compilation of data in vehicle scenarios to develop/benchmark algorithms for speech systems. This paper describes the collection efforts and analysis of two corpora: (1) the UT-Dallas Vehicle Noise (UTD-VN) corpora and (2) the CU-Move in-car speech and noise corpora. The UTD-VN corpus is focused on addressing the variability of in-car noise environments. This corpus includes compilation of unique noise scenarios within the car (Engine idling, AC windows closed, etc.) as well as variability of these scenarios across different makes and models. Another aspect that in-car speech systems need to address along with noise is the emotional and task stress of the driver while performing the driving task. The CU-Move corpus focuses on collection of data to describe the variability of conversational speech in an in-car environment. A sample study is carried out where it is shown that these environments are unique across different vehicles using the UT-Dallas Vehicle Noise corpora. This shows that a detailed analysis of variability across vehicle platforms is necessary for successful deployment of speech systems. In our opinion, these corpora are the first to describe the environment variability along with conversational speech in an in-car environment.

**Keywords** Car noise • command and control • Enhancement • Environment variability • Environmental noise • Navigation • Speech • Speech recognition • Speech systems • Stress

N. Krishnamurthy (✉)
University of Texas at Dallas, Richardson, USA

Texas Instruments, Dallas, USA
e-mail: nitish@ti.com

R. Lubag • J.H.L. Hansen
University of Texas at Dallas, Richardson, USA
e-mail: john.hansen@utdallas.edu

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_9, 

## 9.1 Introduction

Car environments are becoming a standard/core location for conducting voice-based commerce in dialog systems, message or information exchange, and other business or entertainment exchanges. However, one of the main challenges faced by speech and audio systems today is maintaining performance under varying acoustic environmental conditions caused by different driving conditions, car make and models, along with speech variability due to task-induced and emotional stress caused during driving. Efficient use of speech systems in cars require technology to be robust across variations in vehicle environments encountered. In fact, the diversity and rich structure of noise and speech in car acoustic environments create the need for application-specific speech solutions. This is a challenging task since effective communications systems in the car need to address the issue of diversity in transportation platforms and operating conditions. Another aspect along with the environment is the emotional and task stress caused due to task variability and distraction within typical car environments. These factors lead to significant acoustical variations in speech and noise encountered within vehicle environments. The focus here is not the social or legal issues associated with speech system deployment in car environments, but the description of corpora development to address the variability encountered in car environments.

The environment in transportation platforms varies due to the different makes and models of transportation platforms along with the changing operating environments encountered. Examples of the changes in acoustic variations include road characteristics, weather, and the state of the car. Road characteristics are a significant source of noise variation in cars, and the surface properties of the road can change the properties of noise encountered (e.g., asphalt versus concrete, and smooth vs. cracks or potholes). Also, noise changes are dictated by weather conditions such as rain, snow, winds, etc. Depending on the severity, these conditions can sometimes mask other noise events/types in cars. The focus here is to study the variability in normal weather conditions.

Even though significant efforts have been made in the past to study the impact of car noise on speech, there remains a need for a corpus to enable the study of noise events across vehicles and their impact on speech systems. The UT-Dallas Vehicle Noise (UTD-VN) corpus aims to compile the variability observed across vehicles and driving conditions for a fixed set of environmental conditions. This collection is unique as it contains a comprehensive collection of noise events across different vehicle platforms. A sample analysis here formulates noise in a car environment and shows that the noise types are distinguishable across different vehicle makes and models demonstrating the necessity for noise-specific corpora. This corpus opens up new research opportunities where the knowledge gathered from studying car noise events can be exploited by in-vehicle speech systems.

Another aspect of variability in car environments is the speech variability due to task and emotional stress. The CU-Move corpus is a compilation of speech

data collected during natural conversational interaction between the user and an in-vehicle system. In the past, studies have analyzed the impact of in-vehicle noise on speech systems including use of fixed noise and speech collection in lab environments without the variability induced in either speech or noise. Recently, some studies like [1] by Kawaguchi et al. have incorporated these variations. Their corpus focuses on spontaneous conversational Japanese where the speech data was collected under car idling and driving conditions. This study does not include the environment variability of the speech due to the task-induced stress. CU-Move focuses on compiling these variations in speech under diverse acoustic conditions in the car environment along with various environments encountered in realistic driving task. This data was collected from six different vehicles. The core of this corpus includes over 300 speakers from six US cities, with five speech style scenarios including route navigation dialogs. The noise collected during this corpus identified over 14 different unique noise scenarios observed in the car environment.

The goal of CU-Move is to enable the development of algorithms and technology for robust access and transmission of information via spoken dialog systems in mobile, hands free environments. The novel aspects of CU-Move include corpora collection using microphone arrays on corpus development on speech and acoustic vehicle conditions. This setup enables research utilizing environmental classification for changing in-vehicle noise conditions and back-end dialog navigation information retrieval subsystem connected to the WWW. While previous attempts at in-vehicle speech systems have generally focused on isolated command words to set radio frequencies, temperature control, etc., the CU-Move system is focused on natural conversational interaction between the user and in-vehicle system. Since previous studies in speech recognition have shown significant losses in performance when speakers are under task or emotional stress, it is important to develop conversational systems that minimize operator stress for the driver. System advances using CU-Move include intelligent microphone arrays, auditory- and speaker-based constrained speech enhancement methods, environmental noise characterization, and speech recognizer model adaptation methods for changing acoustic conditions in the car.

Here, the focus will be on the UTD-VN corpus with mention of relevant aspects in the CU-Move corpus. In conjunction, these corpora address most of the variations in the environment and speech encountered for holistic development of in-car speech and communication systems.

## 9.2 The UT-Dallas Vehicle Noise Corpora

In the UTD-VN corpus, noise data samples were collected from 20 cars, five trucks, and five SUVs across 10 different noise events. To enable portable recording across vehicles, a portable, lightweight, high-fidelity data collection setup was used

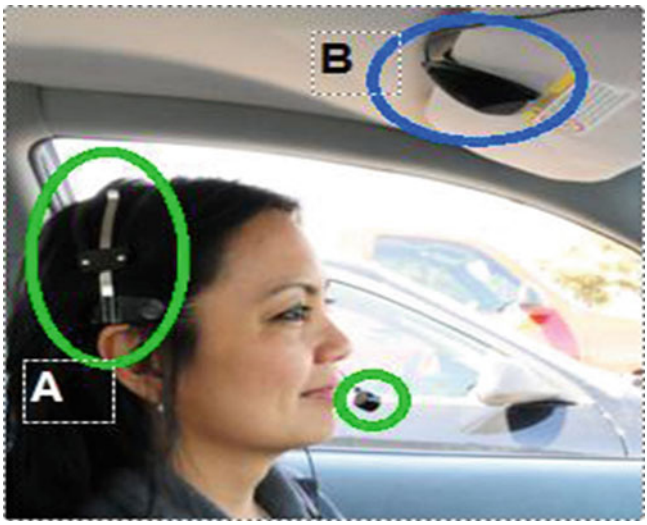



**Fig. 9.1** In-vehicle portable recording setup. (**a**) Shure SM10A close talk microphone and (**b**) Shure MX 391S omni-directional microphone.

to obtain accurate recording of the noise data. The equipments used for in-vehicle data collection were:

(a) Shure SM10A close talk microphone
(b) Shure MX 391S omni-directional microphone
(c) Edirol R-09 recorder

Figure 9.1 shows the recording setup. The close-talk microphone (marked as (a)) is worn by the driver during the data collection to record the noise data observed in the closed-talking microphone under different conditions. The far field microphone (b) has been secured onto the sun visor located above the driver's seat. Meanwhile, the data collector (not in the picture) managed the recording setup.

Data is collected under the following events within the vehicle acoustic environment:

(a) NAWC: No air-conditioning with windows closed
(b) ACWC: Air-conditioner engaged with windows closed
(c) NAWO: No air-conditioning with windows open
(d) HNK: Windows closed with car horn
(e) TRN: Turn signal engaged
(f) IDL: Engine idling
(g) REV: Engine revving
(h) LDR/RDR: Left/right door opening and closing

For these fixed events, the noise varies due to weather and road conditions. To minimize the number of independent variables, such as external noise and road characteristics, the driving routes were fixed for all recordings. The average speed

of the car during the recordings was 40 mph, and data was collected on a 4-mile route of concrete roads. The route was selected so as to consist of a combination of 6-lane city roads with higher traffic density and 2-lane concrete community roads with lower traffic density. The car noise data recording was timed so as to minimize external traffic noise due to peak hours.

The data collection consisted of two parts. The first set of in-vehicle noise events were recorded in the University of Texas at Dallas parking lot. For these recordings, the vehicle was stationary, the windows were closed, and the AC was turned off. Under these vehicle conditions, the following sound events were collected:

(a) Turn signals (TRN)
(b) Horn (HNK)
(c) Front doors opening and closing (LDR/RDR)
(d) Engine idling (IDL)
(e) Revving (REV)

The average total recording time for these conditions was about 6 min.

The second set of recordings took place on roads around the University of Texas at Dallas campus. The data was collected only in dry weather conditions, where the vehicles completed the route twice. The route was 2 mi. long, with two-to-three lane roads and speed limits ranging from 30 to 40 mph. For this corpus, the route was divided into seven sections, and a particular noise condition was assigned to each section.

Figures 9.2 and 9.3 show the designated route. The seven sections of the route are also shown in the figures. As shown in Fig. 9.2, two noise conditions (ACWC, NAWC) were collected in the first loop. During sections A to D of the route, the windows and AC remained closed. In sections E to G of the route, the windows were closed and the AC was turned on with blower at full capacity. Meanwhile, Fig. 9.3 shows the four noise conditions recorded during the second loop. Section K of the route included a speech exercise. Here, the driver was asked to count out aloud from 0 to 9, three times, with the windows closed and AC turned off. Data for NAWC condition was recorded again in sections L and N. The final recording condition was ACWC in section H. The average on-the-road recording time was about 21:25 min. The priorities of this exercise were the NAWC and ACWC conditions as speech systems encounter these conditions frequently in car environments. Collection setup was designed to allow for data collection in multiple sessions to ensure that the audio recording of the car events contained variability due to different road/traffic conditions. The corpus contains a total of 8 h of car noise data.

## 9.3 CU-Move

The UTD-VN corpus deals with variability in fixed environments across car makes and model. Another aspect of in-car acoustics as mentioned in sect. 1 includes speech variability due to stress along with noise. These are the major causes of acoustic mismatch in a car environment. The CU-Move corpus focuses on

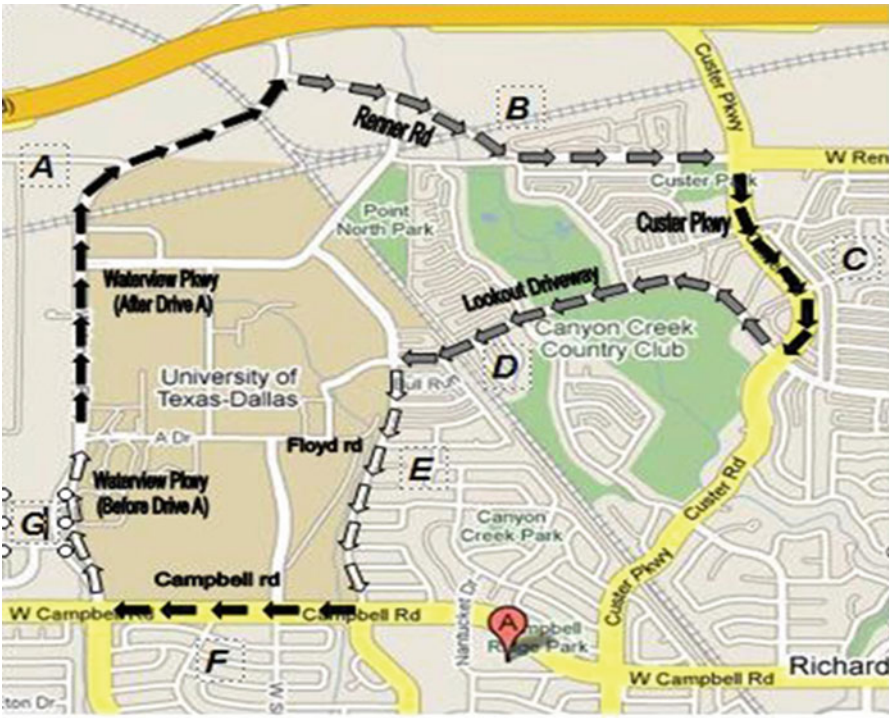



**Fig. 9.2** Loops for data collection. The *dotted lines* indicate unique recording conditions

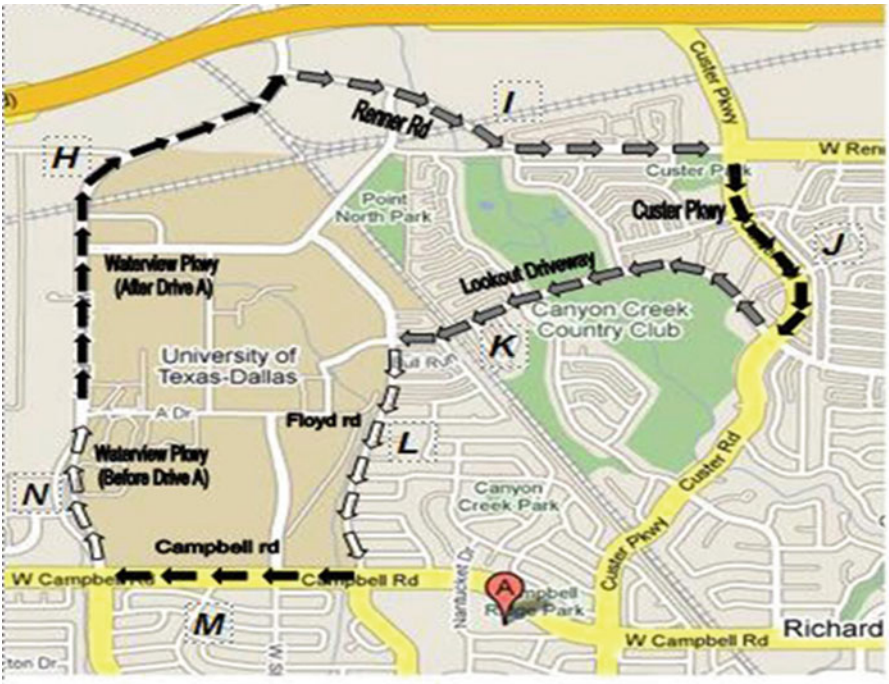



**Fig. 9.3** Loops for data collection. The *dotted lines* indicate unique recording conditions

compiling speech variability for an in-car task along with the environment variability encountered for different task scenarios. This corpus consists of 2 phases:

- Phase I: Speech & speaker data collection
- Phase II: Acoustic noise data collection (CU-Move Noise)

### *9.3.1 Phase I: Speech and Speaker Data Collection*

The speech and speaker data collection is divided in two sections. First (part 1) is structured text where the user is prompted to utter text and numbers similar to what is observed in a command and control application. The second section (part 2) is a dialog system scenario with a real person on the other end.

#### 9.3.1.1 Part 1: Structured Text Prompts

The driver performs a fixed route that includes a combination of several driving conditions (city, highway, traffic noise, etc.). For each speaker, prompts were given for specific tasks listed below from a laptop display situated around the glove compartment of the vehicle. This portion is 30 min long. There are four subsections that include:

- Navigation direction phrases section: a collection of phrases which are determined to be useful for in-vehicle navigation interaction (prompts fixed for all speakers)
- Digits prompts section: strings of digits for the speaker to say (prompts randomized)
- Streets/address/route locations section: street names or locations within the city; some street names will be spelled, some just spoken (prompts randomized)
- Sentences – general phonetically balanced sentences section: collection of phonetically balanced sentences for the speaker to produce (prompts randomized)

#### 9.3.1.2 Part 2: Dialog Wizard of Oz Collection

Here, the user calls a human "wizard" (WOZ) who guides the subject through various routes determined for that city. More than 100 route scenarios particular to each city were generated so that users would be traveling to locations of interest for that city. The human WOZ had access to a list of establishments for that city where subjects would request route information (e.g., "How do I get to the closest police station?", "How do I get to the Hello Deli?"). The user would call in with a modified cell phone in the car, which allows for data collection using one of the digital channels from the recorder.

### 9.3.2 Phase II: Acoustic Noise Data Collection (CU-Move Noise)

One of the primary goals of the CU-Move corpus is to collect speech data within realistic automobile driving conditions for route navigation and planning. Prior to selection of the vehicle used for phase II data collection across the United States, an in-depth acoustic noise data was collected on six vehicles in Boulder, Colorado. This section briefly summarizes the noise data collection scenarios.

#### 9.3.2.1 Vehicles

A set of six vehicles were selected for in-vehicle noise analysis. These vehicles were from model years of 2000 or 2001 (all had odometer mileage readings which ranged between 11 and 8,000 mi.). The six vehicles were:

- [Cav] Chevy Cavalier compact car
- [Ven] Chevy Ventura minivan
- [SUV] Chevy SUV Blazer
- [S10] Chevy S10 extended pickup truck
- [Sil] Chevy Silverado pickup truck
- [Exp] Chevy Express cargo van

All acoustic noise conditions are collected across six vehicles: Blazer, Cavalier, Venture, Express, S10, and Silverado. The noises were labeled into 14 categories which include:

1. Idle noise: the sound of the engine after starting and not moving, windows closed
2. Noise at 45 mph, window opened 1″
3. Noise at 45 mph, window closed
4. Noise at 45 mph, window opened half way down
5. Noise at 65 mph, window opened 1″
6. Noise at 65 mph, window closed
7. Acceleration noise, window closed
8. Acceleration noise, window opened half way down
9. AC (high) noise, window closed
10. Deceleration noise, window opened 1″
11. Turn signal noise at 65 mph, window closed
12. Turn signal noise, window opened 1″
13. Turn signal noise, window closed
14. Wiper blade noise, window closed

A total of 14 noise conditions were extracted from the same environment and locations for each of the 6 GM vehicles. This noise corpus focused on describing the unique variations in noise scenarios encountered in a car environment as opposed to focusing on variations across cars. This is described in Fig. 9.4. The CU-Move

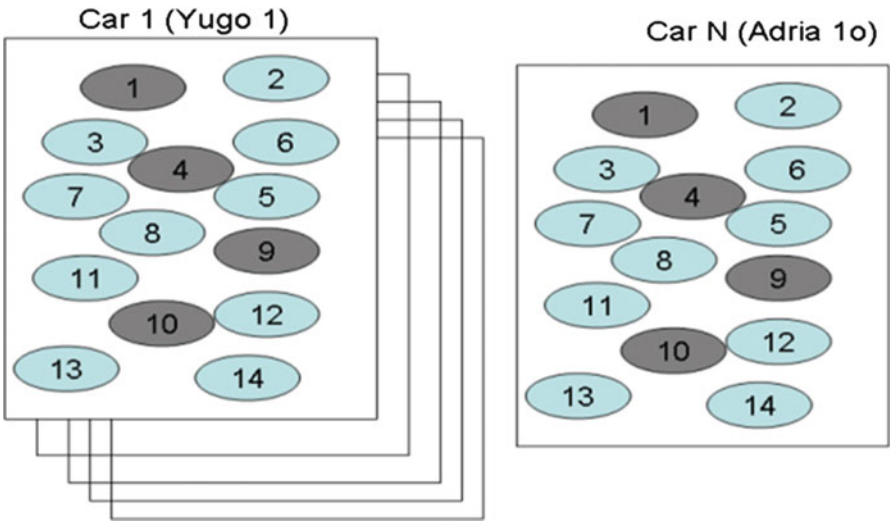



**Fig. 9.4** The Scope of UTD-VN vs. CU-Move

corpus is a compilation of events encountered in a car-driving scenario as described earlier whereas the UTD-VN is a compilation of a few events across various cars and conditions.

## 9.4 Noise Analysis and Modeling

The car noise–environment noise samples in both the corpora can be described as a combination of noise sources active in the car, as well as the acoustic environment of the vehicle itself. In other words, the resultant car noise is a function of car-independent noise $(n_e)$ and car-dependent noise $(n_{ce})$. Here, an additive model for $(\hat{n}_{ce})$ is assumed. This is illustrated in Fig. 9.5. Depending on the relative dominance of the constituent noises, the overall resultant noise observed can be of three primary types.

- *Car Internal Dominant Noise*: If car-dependent sounds such as air conditioning, horn, and engine sounds dominate, then the resulting noise $n_e$is unique to the specific car producing the sound (i.e., if $n_e << n_{ce}$ then $(\hat{n}_{ce}) \approx (n_{ce})$). For purposes of car verification/platform identification, this forms the most conducive scenario. For speech systems, it means that car specific models might be optimal for the best performance in specific car environments.
- *Car–Environment Dominant Noise*: If the observed sound is the sound of the car interacting with its environment, such as the sound of wheels on the road or wind

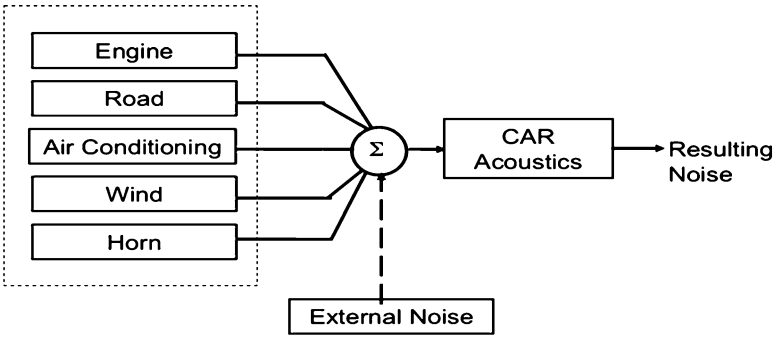



**Fig. 9.5** Model of acoustic environment in the car

noise, then the resulting car noise is less car specific/dominant (i.e., $n_e > n_{ce}$). This scenario is less favorable for car verification than the previous case.

- *Environment Dominant Noise*: Finally, noise sources external to the car such as horn from a nearby car or engine sounds from a passing truck are considered outside the scope of this study. This is because these sounds are least car specific (if $n_e << n_{ce}$ then$(\hat{n}_{ce}) \approx (n_{ce})$). This would cause increased confusability in acoustic vehicle platform identification. This case would require the most generic noise models for speech systems.

In practice, it is very difficult to obtain these noise types in isolation since all noise sources cannot be controlled simultaneously in naturalistic driving. However, in the process of car noise data collection, we have minimized external noise by carefully choosing the recording conditions.

For analysis, three noise conditions in the same vehicle are analyzed for their spectral content and variability. These conditions consist of NAWC, ACWC, and NAWO, as shown in Fig. 9.6. These environments were chosen because of their high probability of occurrence. Furthermore, these noise scenarios represent unique environments because the dominant sounds in each case are different (e.g., in ACWC, AC noise is dominant).

The spectral content of the vehicle acoustic environments under ACWC, NAWC, and NAWO conditions are shown in Fig. 9.6. As seen in Fig. 9.6b when the AC is on and the windows are closed, the car noise is least time varying. The main noise sources in this environment are AC, car engine, and road noise, but the AC is the dominant source of noise. The spectral slopes indicate that the ACWC scenario has the most high-frequency content compared to the other two noise types. Also, this condition is the most conducive for car verification since the AC and the fan/air blower are the most dominant noise sources. In the other two cases, wind noise and road noise are the main noise sources. When AC is turned off, as seen in Fig. 9.6a, the car noise is a mixture of road and engine noises. The only car-dependent noise type when the AC is off and windows closed is the car engine noise which is masked by the road noise. Finally, the last plot shows NAWO condition, where the main noise sources are wind noise, road noise, and engine noise. NAWO

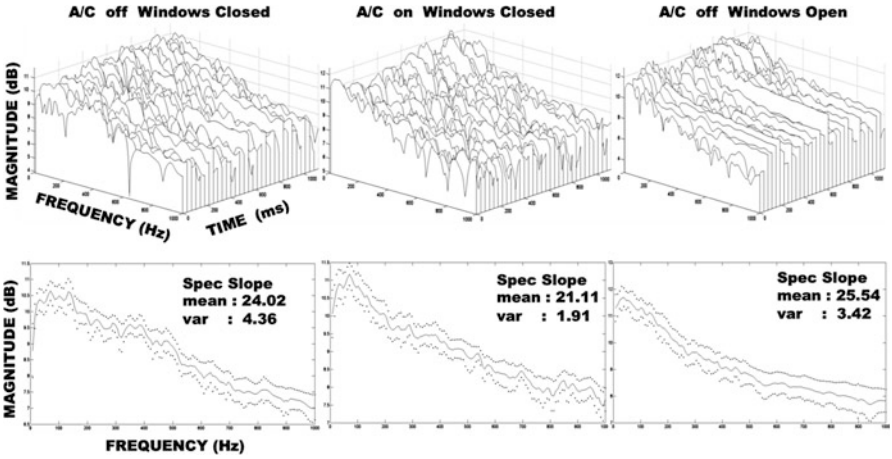



**Fig. 9.6** Vehicle acoustic environments: (**a**) road and engine noise is predominantly low frequency, (**b**) road, engine, and air-conditioning shows structure in higher frequencies, and (**c**) wind noise wipes out all structure and only the aggregate remains

has the least car-dependent information as compared to the other two environments since the wind noise is external to the car and masks all car-dependent information. As seen here, car-dependent noise types are the best indicators of car types and the car-dependent AC noise, which can be viewed as a potential excitation source for the interior vehicle compartment, enables the noise to carry more car-dependent information. To study the uniqueness and the variability in different acoustic conditions across cars, the acoustic data was modeled using 13 dimensional Gaussians and the Kullback–Leibler distance was employed to analyze the in-class and across-car differences. This is illustrated in Fig. 9.7, where solid areas represent the acoustic space for a single car in a particular environment, and the smaller shaded areas represent models of the session-to-session variability in the same acoustic event.

To estimate the separation across different vehicles, the in-class and across-class KL distances are measured. If the vehicle sound events are separable within this framework, the average in-class distances will be much lower than the out-of-class distances. These distances are evaluated for three vehicles, and box plots of these distances are presented in Fig. 9.8. As seen for each of these vehicle conditions, the in-class (IS) distances are clearly separable from the out-of-class (OS) distances, indicating that under the ACWC are spectrally unique and differentiable from each other.

As evident from this discussion, car noise environment is a unique environment with a mix of car-dependent and independent noise sources. Depending on the driving conditions and road scenarios, the environments may rapidly change from one condition to the next. The analysis also reinforces the need to collect data under different scenarios as the intervehicle variations might be a significant factor to normalize for generic speech systems.

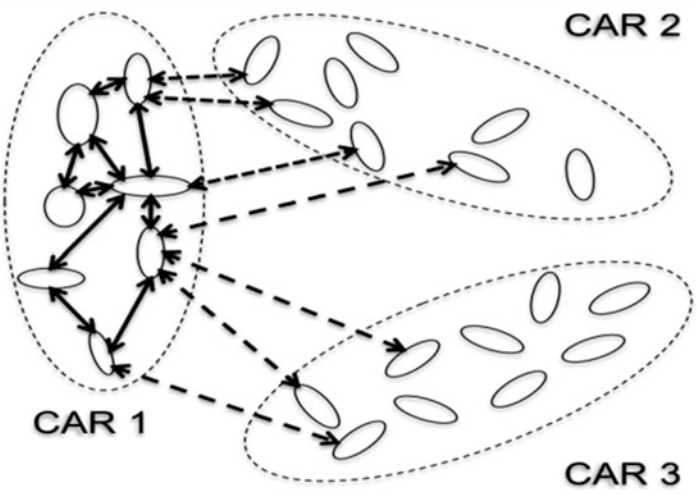



**Fig. 9.7** Illustration of in-class vs. out-of-class distances for each noise event in a car. Each *dotted region* denotes a car and it encloses solid regions that denote session instances

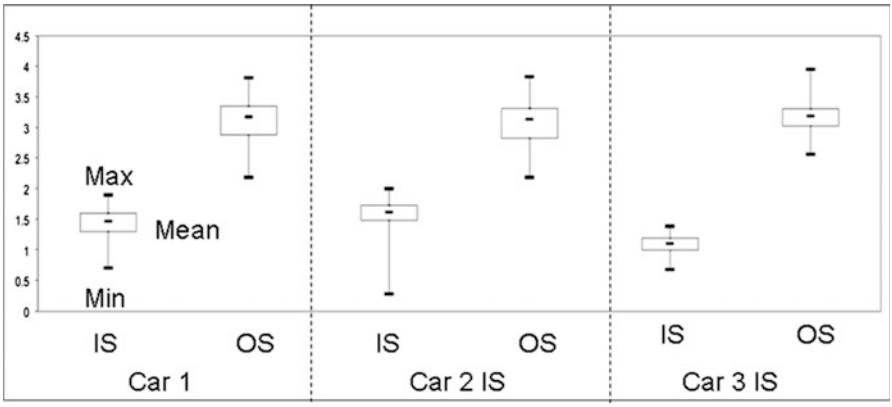



**Fig. 9.8** Inset and out-of-set distances for ACWC in three cars

The CU-Move corpus has been used extensively to understand the noise properties in car environments and leveraging these properties for speech systems. Examples of these studies include [3, 4], and [5] by Akbacak and Hansen to "environmental sniffing" of variations in environment to use most appropriate models using the Rover scheme. In [6], the authors used the CU-Move corpus for advancing voice-activated route navigation in car systems. Hansen [7] includes a detailed description of the corpus along with the usage scenarios of CU-Move.

## 9.5 Conclusion

This paper summarizes collection efforts of the UTD-VN corpus and the CU-Move corpus. The UTD-VN corpus includes a rich variety of noise types that are frequently encountered in car environments. The UTD-VN corpus contains noise data that reflects the variability in vehicle noise events across different makes and models of cars, whereas the CU-Move corpus includes the diversity in the car environments with variations in speech due to task and driving stress. Using the UTD-VN corpus, a model for car noise was formulated and used to demonstrate the uniqueness of noise types across different vehicles. The volume, diversity, and real-world nature of these corpora make it very valuable for researchers exploring in-vehicle speech technology. The next stage of data collection would be ubiquitous data collection for in-car environments that would use multiple sensors for aiding development of integrated multi-input systems that are most suited for in-car environments.

## References

1. Kawaguchi N, Matsubara S, Iwa H, Kajita H, Takeda K, Itakura F, Inagaki F (2000) Construction of speech corpus in moving car environment. In: Proceedings of the Interspeech-2000, vol 3, Beijing, pp 362–365
2. Hansen JHL, Plucienkowski J, Gallant S, Pellom B, Ward W (2000) CU-move: robust speech processing for in-vehicle speech systems. In: Proceedings of the Interspeech-2000, vol 1, Beijing, pp 524–527
3. Akbacak M, Hansen JHL (2003) Environmental sniffing: robust digit recognition for an in-vehicle environment. In: interspeech-2003/Eurospeech-2003, Geneva, pp 2177–2180
4. Akbacak M, Hansen JHL (2003) Environmental sniffing: noise knowledge estimation for robust speech systems. In: IEEE ICASSP-2003: international conference on acoustics, speech, and signal processing, vol 2, Hong Kong, pp 113–116
5. Akbacak M, Hansen JHL (2007) Environmental sniffing: noise knowledge estimation for robust speech systems, IEEE Trans Audio Speech Lang Process 15(2):465–477
6. Hansen JHL, Zhang X, Akbacak M, Yapanel U, Pellom B, Ward W (2003) CU-Move: advances in in-vehicle speech systems for route navigation. In: IEEE workshop in DSP in mobile and vehicular systems, paper 6.5, Nagoya, 4–5 April 2003, pp 1–6
7. Hansen JHL (2002) Getting started with the CU-Move corpus. CU-Move documentation

# Chapter 10
# A Likelihood-Maximizing Framework for Enhanced In-Car Speech Recognition Based on Speech Dialog System Interaction

**Tristan Kleinschmidt, Sridha Sridharan, and Michael Mason**

**Abstract** Speech recognition in car environments has been identified as a valuable means for reducing driver distraction when operating noncritical in-car systems. Under such conditions, however, speech recognition accuracy degrades significantly, and techniques such as speech enhancement are required to improve these accuracies. Likelihood-maximizing (LIMA) frameworks optimize speech enhancement algorithms based on recognized state sequences rather than traditional signal-level criteria such as maximizing signal-to-noise ratio. LIMA frameworks typically require calibration utterances to generate optimized enhancement parameters that are used for all subsequent utterances. Under such a scheme, suboptimal recognition performance occurs in noise conditions that are significantly different from that present during the calibration session – a serious problem in rapidly changing noise environments out on the open road. In this chapter, we propose a dialog-based design that allows regular optimization iterations in order to track the ever-changing noise conditions. Experiments using Mel-filterbank noise subtraction (MFNS) are performed to determine the optimization requirements for vehicular environments and show that minimal optimization is required to improve speech recognition, avoid over-optimization, and ultimately assist with semi-real-time operation. It is also shown that the proposed design is able to provide improved recognition performance over frameworks incorporating a calibration session only.

**Keywords** Automatic speech recognition (ASR) • In-car speech recognition • LIMA frameworks • Mel-filterbank noise subtraction (MFNS)

T. Kleinschmidt (✉) • S. Sridharan • M. Mason
Speech & Audio Research Laboratory, Queensland University of Technology, Brisbane, QLD, Australia
e-mail: t.kleinschmidt@qut.edu.au; s.sridharan@qut.edu.au; m.mason@qut.edu.au

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*, DOI 10.1007/978-1-4419-9607-7_10, 

## 10.1 Introduction

With the increased desire from consumers to integrate electronic devices such as MP3 players, navigation systems, and mobile phones for use in their vehicles comes the need to provide more intuitive human–machine interfaces (HMI) than currently seen in low- to midrange vehicles. Automatic speech recognition (ASR) can provide a safe and easy-to-use HMI, and technological advancements have enabled low-cost hardware implementations of ASR systems – a key requirement to widespread adoption in the automotive industry.

Most ASR systems are trained for use in controlled scenarios (e.g., office environments or telephone-based systems) and fail to produce satisfactory performance under the continually changing noise conditions found in automotive environments [1]. This is a key challenge to deployment of in-car ASR – drivers demand high-accuracy recognition, but high levels of noise restrict recognition performance of conventional ASR systems.

Speech enhancement is a common method for making ASR systems more robust against noise. Enhancement techniques aim to reduce the noise levels present in speech signals, allowing clean speech models (which are easily trained due to the availability of large amounts of data) to be utilized by the recognizer. This is a popular approach as enhancement algorithms are typically easily integrated with existing ASR front-end processing, as well as requiring little-to-no prior knowledge of the operating environment in order to achieve improvements in recognition accuracy. Both of these aspects are particularly attractive for in-car applications where hardware and software overheads must be minimized and where the system is continually subjected to changes in acoustic conditions.

Popular speech enhancement algorithms such as filter-and-sum beamforming (using multiple microphone speech acquisition) and spectral subtraction were originally designed to improve intelligibility and/or quality of speech signals without considering the effects on other speech processing systems such as recognition [2]. Optimization of parameters in these algorithms focuses on signal-based measures (e.g., maximizing signal-to-noise ratio or minimization of the mean-squared signal error). Enhancement techniques operating in this manner may still produce word accuracy improvements, but these improvements are by-products of the optimization process rather than its objective [2].

Promising results have been shown in studies that use speech recognition likelihoods as the optimization criteria as opposed to quality or intelligibility measures [2–4]. Enhancement techniques are placed within likelihood-maximizing (LIMA) frameworks, which attempt to *jointly* optimize both the recognized acoustic state sequence as well as enhancement parameters. There are three main types of LIMA framework – calibrated, unsupervised, and supervised.

Calibrated LIMA frameworks require a known adaptation utterance in order to optimize the enhancement parameters. Adaptation is typically performed using a dedicated calibration session for each speaker, with the optimized enhancement parameters kept constant for all other utterances for that speaker [2, 3]. This approach assumes constant noise conditions and therefore has limited potential for achieving optimal performance in rapidly changing vehicular environments.

An unsupervised LIMA framework was also proposed in [2] whereby online optimization takes place on an utterance-by-utterance basis using the hypothesized transcription as opposed to the true transcription. Whilst this method removes the restriction of a calibration session and showed considerable reductions in word error rates [2], it is highly reliant on the initial accuracy of the speech recognizer. Whilst the word error rate of the recognizer used in these experiments was high (approximately 60%), the test recordings were obtained at relatively high signal-to-noise ratios in a constant noise environment. Systems operating in the nonstationary vehicular environment exhibit even higher word error rates, resulting in reductions in accuracy of the hypothesized transcriptions. Optimization on unreliable transcriptions should be avoided as it could lead to suboptimal parameter estimation and therefore further reductions in recognition performance.

In this chapter, we consider the third alternative (i.e., a supervised LIMA framework) and propose a dialog-based design that allows regular optimization iterations in order to track the ever-changing noise conditions. The chapter reviews LIMA frameworks employing Mel-filterbank noise subtraction (MFNS) specifically for in-car speech recognition. The analysis involves testing a number of calibrated adaptation scenarios, as well as development of a novel online optimization framework, based on speech dialog systems which exploit user confirmation of correctly recognized voice commands to provide adaptation data for the LIMA framework.

## 10.2 LIMA Mel-Filterbank Noise Subtraction for In-Car Environments

### *10.2.1 Likelihood Maximization*

Speech enhancement algorithms aim to produce improvements in human intelligibility of speech signals. Automatic speech recognition systems hypothesize the most likely sequence of statistical models produced by the observed feature vectors. As a result, traditional optimization of spectral subtraction algorithms based on waveform criteria such as signal-to-noise ratio maximization [5, 6] does not necessarily translate into improvements in ASR word accuracy [2]. With the primary aim of using speech enhancement to improve speech recognition accuracy, Seltzer et al. [2] proposed a likelihood-maximization framework for enhancement parameter optimization. This framework was originally proposed for filter-and-sum beamforming but has since been applied to subtraction factors in multiband spectral subtraction [3].

In a recognition system incorporating speech enhancement, feature vectors are a function of the speech enhancement process. The recognition hypothesis provided by an optimal Bayes classifier regularly used in ASR systems is given by

$$\hat{w} = \arg\max_{w \in W} P(Z(\xi)|w) \cdot P(w), \tag{10.1}$$

where dependence of the feature vectors $Z$ on the enhancement parameters $\xi$ is clearly shown. The acoustic score $P(Z(\xi))$ is the measure of importance in LIMA systems as the transcription on which the optimization takes place is assumed to be known, and therefore, the language model score $P(w)$ will not change. The aim of likelihood maximization for MFNS is therefore to optimize the parameters to maximize the acoustic score of the recognized word sequence $\hat{w}$.

An initial decode pass is performed using default enhancement parameters to generate a state sequence $s$ on which to optimize $\xi$. In order to find the optimal values of $\xi$, gradient-based optimization is used on the total log-likelihood of the observed features, which is defined by

$$L(\xi) = \sum_i \log(P(z_i(\xi)|s_i)). \tag{10.2}$$

For a Hidden Markov Model (HMM) speech recognizer using Gaussian mixture state models (as used in this chapter), the gradient of the total log-likelihood is given by [2]

$$\nabla_\xi L(\xi) = -\sum_i \sum_{m=1}^{M} \gamma_{im}(\xi) \frac{\partial z_i(\xi)}{\partial \xi} \sum_{im}^{-1} (z_i(\xi) - \mu_{im}), \tag{10.3}$$

where $\gamma_{im}(\xi)$ is the a posteriori probability of the $m$th mixture component in state $s_i$ given the observed feature vector $z_i(\xi)$. The mean vector $\mu$ and covariance matrix $\sum$ from the acoustic model are required for each state $i$ and mixture component $m$ in order to calculate the gradient. The remaining term in Eq. 10.3 is the Jacobian matrix, $\partial z_i(\xi)/\partial \xi$, which consists of the partial derivatives of each element of the feature vector with respect to each of the enhancement parameters. Each Jacobian element is derived directly from the enhancement procedure (refer to Sect. 10.2.3). Once the gradient-based optimization converges, the new enhancement parameters are used to generate another set of feature vectors, and a subsequent decode pass is performed. A new state sequence is generated, and the enhancement parameters are further optimized for this new state sequence. The process continues until the recognition likelihood (and state sequence) converges, ensuring joint optimization of the recognized state sequence and the speech enhancement parameters.

## *10.2.2 Optimization Methods for In-Car ASR*

### 10.2.2.1 Calibrated LIMA Framework

The simplest and most common approach for optimizing the enhancement parameters is to use a calibration session with a known transcription $w_C$. Previous studies used a single known utterance for each speaker in order to determine

optimal enhancement parameters for that particular speaker [2, 3]. Whilst this procedure ensures that optimization takes place on a state sequence which is correct, calibrated LIMA frameworks inherently assume that background noise conditions do not change between the calibration and testing sessions. This is a major challenge for in-car speech recognition since vehicular environments are subjected to continually changing noise levels and conditions which mean calibration utterances would be required every time noise conditions changed significantly from the previous optimization. To overcome this, optimized enhancement parameters could be stored for each common noise condition; however, this still requires a calibration utterance to be used at some point in the system. Since there is a wide range of noise conditions, the user would be continually asked to repeat the adaptation utterance in order to obtain the optimal set of parameters. This operation is an unnecessary annoyance for the driver and is likely to lead drivers to become frustrated with the speech dialog system; such emotions could lead to further repercussions on ASR and driving performance [7].

An alternative solution is to calibrate once only for each driving session (e.g., a common startup utterance such as "Start dialog" could be used for adaptation), but this introduces the risk of inferior recognition in noise conditions significantly different to those present during calibration.

The calibration framework is also reliant on the words contained in the adaptation utterance; therefore, it is necessary for the adaptation utterance to be phonetically balanced and sufficiently long enough to provide as much acoustic model coverage as possible in order to generalize the optimized enhancement parameters. This is in direct conflict with the majority of dialog systems which promote simpler linguistic structures than human conversation and are therefore unlikely to be phonetically balanced. Thus, a separate utterance unrelated to the dialog transaction is required which is likely to be seen by the user as a further inconvenience and therefore impractical for this particular application.

#### 10.2.2.2 Unsupervised LIMA Framework

The unsupervised LIMA framework proposed in [2] may be a more appropriate choice for in-car environments. Unsupervised adaptation removes the restriction of a calibration utterance (thereby making the adaptation process transparent to the user), and instead, optimization takes place on an utterance-by-utterance basis. The major issue with the unsupervised operation is that it uses a hypothesized transcription, $w$, rather than the true transcription$w_C$. The hypothesis transcription is highly reliant on the effectiveness of the underlying acoustic models and state sequence generated by Viterbi alignment; therefore, the hypothesis transcription is likely to be less than 100% correct.

Since the true transcription $w_C$ is unknown, it is possible that states in the hypothesized transcription $\hat{w}$ are incorrect due to misrecognition and frame alignment errors (N.B. frame alignment errors will occur even when the transcription is known a priori, but should be limited). These inaccurate states will lead to the

resulting enhancement parameters being suboptimal since optimization is performed on the wrong state models. In turn, suboptimal enhancement parameters could lead to further decreases in accuracy in the subsequent decoding state. This effect is particularly likely when the number of incorrectly labeled frames is greater than the correctly labeled frames, as may be the case in high-noise conditions.

#### 10.2.2.3 Proposed Dialog-Based LIMA Framework

Having identified problem with the existing LIMA frameworks, we propose to exploit a confirmation-based speech dialog system to drive optimization. Dialog systems requiring users to verify commands with simple "Yes/No" replies are a well-established mechanism in voice recognition applications. A block diagram of the proposed framework within the dialog exchange is shown in Fig. 10.1.

This system mimics the calibrated and unsupervised frameworks by performing an initial decode using default enhancement parameter values in the feature extraction stage. This framework differs from previous work following the initial ASR pass. Instead of immediately performing optimization, the hypothesized word sequence is first verified through the grounding process which is required in the dialog system in order to detect any misrecognition errors which need to be corrected prior to executing a desired action such as determining route navigation.

Since it is cumbersome for the dialog manager to request confirmation from the user after each response, grounding often occurs once the dialog systems have gathered a number of pieces of information, for example the suburb, street name, and number of a destination address. In the case where the user states the information is incorrect, the dialog manager will attempt to recover from these errors by either asking for corrections to specific information or restarting the dialog transaction. In this instance, the enhancement parameters remain unaltered.

It is also possible to incorporate knowledge of the state of the car environment to alter the enhancement parameters should the noise condition change drastically between optimizations. The purpose of this chapter is not to suggest how this should be done but to analyze the performance of existing and proposed LIMA frameworks and make recommendations on how these are best utilized in automotive environments.

When the user confirms the information to be correct, this affirmation is fed back to the dialog manager for further processing (e.g., a call to an external information source such as the navigation system) but also triggers the optimization of the enhancement parameters. In order to interface the optimization process with the grounding procedure, it is required to store the user responses as well as the hypothesized state sequences – this is reflected in Fig. 10.1. On confirmation, this stored information is used in the optimization process; if rejected, the stored state sequence is therefore unreliable, and so, the memory can be cleared in preparation for responses in the error-recovery stage.

The primary advantage of the proposed dialog-based LIMA framework is that optimization never takes place on inaccurate transcription hypotheses, which overcomes the limitation of the unsupervised framework. Another advantage is the

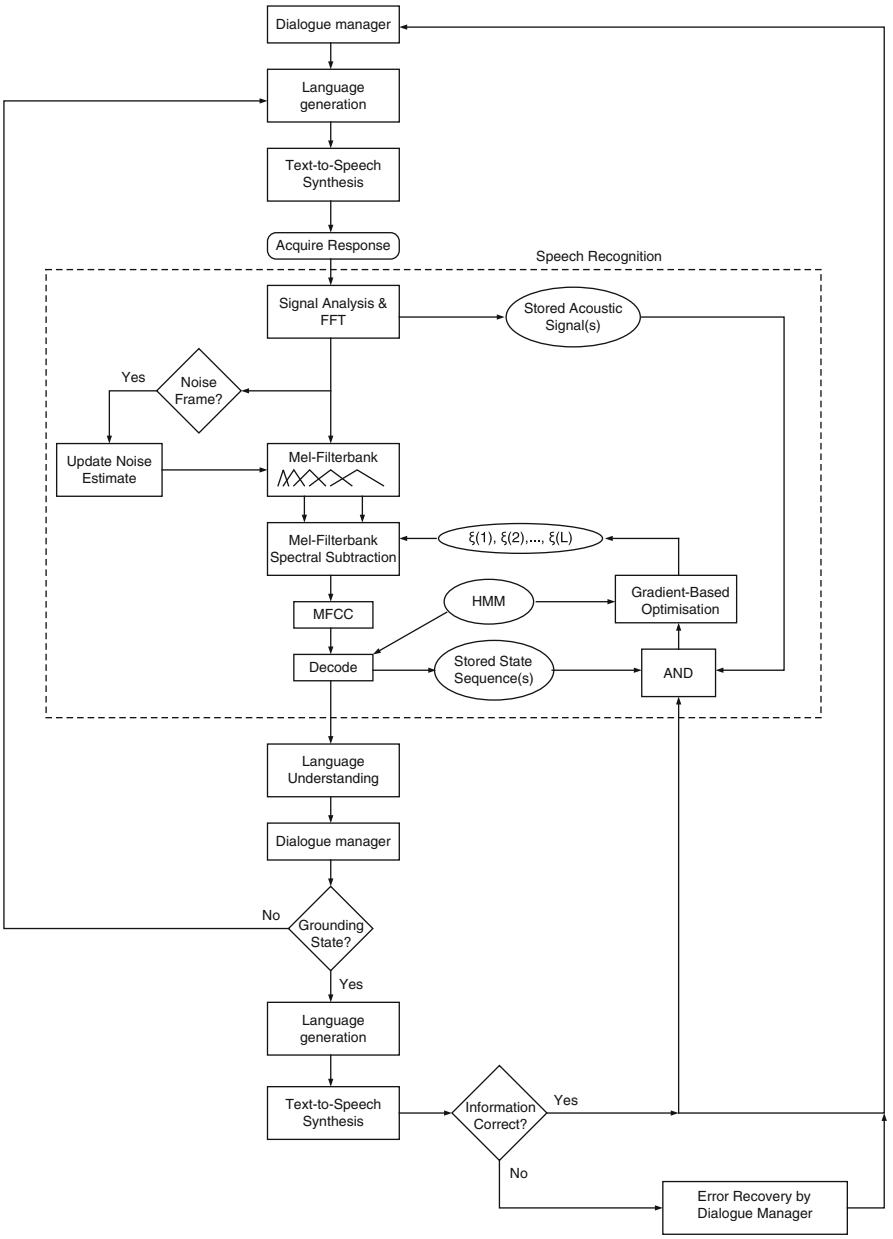



**Fig. 10.1** Proposed confirmation-based speech dialog system for in-car speech recognition using LIMA speech enhancement

ability to continually update the enhancement parameters as the noise conditions inside the vehicle change. This is achieved by maintaining the previous enhancement parameters until the next successful dialog transaction, by which time the noise conditions may have changed. As a result, the dialog-based system is able to overcome the need for matched noise conditions required for calibrated operation to be fully effective.

### *10.2.3 Mel-Filterbank Noise Subtraction*

In this chapter, we concentrate on spectral subtractive enhancement algorithms for this application. Spectral subtraction for speech enhancement was originally proposed by Boll in 1979 [8]. Enhancement is typically performed in the frequency domain; however, subband subtraction techniques such as the Mel-filterbank noise subtraction (MFNS) method proposed in [9] have become popular for use with recognition systems. BabaAli et al. [7] recently utilized the framework introduced in [2] to optimize the subtraction scaling factors in multiband spectral subtraction in the frequency domain.

In a noisy environment, speech $S(f)$ is assumed to be corrupted by uncorrelated additive background noise $D(f)$ to produce corrupted speech $Y(f)$:

$$Y^i(f) = S^i(f) + D^i(f), \tag{10.4}$$

where frequency spectra are obtained from the short-time Fourier transform of frame $i$.

Generally, an estimate of the background noise magnitude spectrum is subtracted from the magnitude spectrum of the noisy signal to give an estimate of the clean speech magnitude. Noise estimates are calculated during nonspeech periods and are typically kept constant throughout speech periods. In the following, the frame index $i$ has been removed from the noise estimate to reflect this operation.

In this chapter, however, we consider Mel-filterbank noise subtraction [9]. Using the Mel-frequency scale commonly used in speech recognition, the frequency spectrum is divided into a number of subbands with $f_U^k$ and $f_L^k$ being the upper and lower cutoff frequencies for the $k$th Mel-filterbank, respectively. Using this definition, Mel-filterbank noise subtraction is described by

$$\begin{aligned}
E_Y^i(k) &= \int_{f_L^k}^{f_U^k} |Y^i(f)| \mathrm{d}f \\
E_{\hat{D}}(k) &= \int_{f_L^k}^{f_U^k} |\hat{D}(f)| \mathrm{d}f \\
E_{\hat{S}}^i(k) &= \begin{cases} E_Y^i(k) - \alpha(k) E_{\hat{D}}(k) & E_Y^i(k) > \frac{\alpha(k)}{1-\beta} E_{\hat{D}}(k) \\ \beta E_Y^i(k) & \text{otherwise} \end{cases}
\end{aligned} \tag{10.5}$$

where $E_Y^i(k)$, $E_{\hat{D}}(k)$, and $E_{\hat{S}}^i(k)$ are the energies of the $k$th Mel-filterbank of the noisy speech, noise estimate, and the clean speech estimate, respectively. The scaling factor $\beta$ enforces a maximum level of signal energy attenuation and ensures that output filterbank energies remain positive. Filterbank-dependent subtraction factors – $\alpha(k)$ – are included to compensate for estimation inaccuracies of the instantaneous noise energy. In the experiments that follow, only the subtraction factors are optimized, that is:

$$\xi = [\alpha_1, \alpha_2, ..., \alpha_K]. \tag{10.6}$$

The expression for the Jacobian elements $\partial z_i(\alpha(k))/\partial\alpha(k)$ for each enhancement parameter can be derived as per [10] to produce

$$\frac{\partial z_i(\alpha(k))}{\partial\alpha(k)} = -\frac{1}{2}\sum_{k=0}^{K-1}\frac{\Phi_{ck}E_{\hat{D}}^i(k)}{\hat{E}_S^i(k)} \times \left(1 + \frac{E_Y^i(k)(1-\beta) - \alpha(k)E_{\hat{D}}^i(k)}{|E_Y^i(k)(1-\beta) - \alpha(k)E_{\hat{D}}^i(k)|}\right), \tag{10.7}$$

where $\Phi_{ck}$ are elements of the DCT matrix for cepstral coefficient $c$.

## 10.3 Experimental Procedures

### *10.3.1 Experimental Data*

Digit strings comprising the phone numbers task of the AVICAR database collected by the University of Illinois [11] were used as the test data. The AVICAR database contains real speech recorded in five different driving conditions: idle (IDL), 35 mph with windows up (35U) and down (35D), and 55 mph with windows up (55U) and down (55D). All experiments utilized an altered version of the first five experimental folds of the AVICAR evaluation protocol developed in [12]. The data for this evaluation consists of 38 speakers, all of which have at least one utterance available in all of the noise conditions.

### *10.3.2 Speech Recognizer*

Utterance decoding was performed using the HMM Toolkit [13]. Speaker-independent, context-dependent 3-state triphone HMM acoustic models were trained using the Wall Street Journal 1 corpus. Each HMM state was represented using a 16-component Gaussian mixture model.

For each observation, 39-dimensional MFCC feature vectors were generated consisting of 13 MFCC (including $C_0$) plus 13 delta and 13 acceleration coefficients. Cepstral mean subtraction was applied to each feature. The elements of the Jacobian were derived from this feature representation as per Eq. 10.7.

The recognition task uses an open word loop grammar [12]; therefore, no restrictions are made to ensure that exactly ten digits are recognized.

All speech recognition results quoted in this chapter are word accuracies (in %) and are calculated as

$$\text{Accuracy} = \frac{N - D - S - I}{N} \times 100, \tag{10.8}$$

where $N$ represents the total number of words, $D$ the number of deletions, $S$ the number of substitutions, and $I$ the number of insertions [13].

### *10.3.3 Optimization Iterations*

Since LIMA is an optimization problem, over-optimization of the enhancement parameters to a specific noise condition, speaker, or subset of acoustic state models is highly possible and should be avoided. This suggests that the number of optimization iterations should not be large in order to maintain generality across conditions, but too little iteration may result in the LIMA framework operating less effectively than a standard enhancement system. Considering real-time operation (another important consideration for in-car ASR) also points to limited iterations.

To address this issue, two experiments were designed to determine a suitable balance between ASR performance and pseudo real-time operation using the noise-only calibration framework described in Sect. 10.3.4. This framework was used since the belief was that noise conditions have a greater effect on the resulting enhancement parameters than individual speakers since speaker-independent acoustic models are being used.

In the first experiment, the number of gradient-descent iterations was varied whilst using a single joint optimization iteration (i.e., full recognition and parameter optimization cycles). The second experiment varied the number of joint optimization iterations whilst the gradient-descent iterations (determined from the former experiment) were kept constant. The combined outcomes of these experiments dictated the levels of optimization used for assessing the frameworks detailed in Sect. 10.3.4.

For all experiments, the enhancement parameters were initialized to $\alpha(k) = 1$ for all 26 Mel-filterbanks. These values were an appropriate initial guess since standard MFNS using these values provides improvements in speech recognition accuracy over a system without enhancement [10].

### *10.3.4 Likelihood-Maximization Frameworks*

The AVICAR database enables analysis of LIMA frameworks based on speaker or noise calibration as well as a combination of both. The following LIMA frameworks have been tested:

- Calibrated LIMA framework using optimization on a noise-by-noise basis
- Calibrated LIMA framework using optimization on a speaker-by-speaker basis under a single, randomly chosen noise condition
- Calibrated LIMA framework using optimization for each speaker in each noise conditions (i.e., matched conditions)
- Proposed dialog-based LIMA framework without calibration
- Proposed dialog-based LIMA framework with a single calibration utterance in a random noise condition
- Proposed dialog-based LIMA framework with a single calibration utterance in the idle noise condition

The unsupervised LIMA frameworks were not assessed in this chapter as the overall performance of the speech recognizer is low (less than 50% average word accuracy), making the hypothesis transcriptions (and therefore the optimized parameters) unreliable.

Each calibrated LIMA framework used a single, randomly generated utterance treated as adaptation session. For the noise-only calibration framework, a random utterance from a random speaker was chosen for each experimental fold in the evaluation protocol. For speaker-based calibration (applied in both calibrated and dialog frameworks), a single utterance from a random noise condition was used for each speaker, with the remaining utterances ordered randomly to simulate realistic driving conditions.

The proposed dialog system was run using no prior calibration, and optimization occurred every time the decoder correctly recognized *all* ten digits in the phone number. Utterances which occur prior to the first optimization exhibit the same performance as the static MFNS system and are therefore ignored in the final evaluation (N.B. this is why baseline results differ across the experiments).

In order to simulate a priori knowledge relating to previously optimized enhancement parameters, the dialog-based framework was also tested using an initial adaptation utterance which was either randomly chosen or from the idle condition. The idle condition was chosen as this is a likely scenario for users to first communicate with the in-car speech dialog system – for instance, for entering a destination address before setting off on the journey. Again, all utterances which occurred prior to the first subsequent optimization (excluding calibration) were ignored in the evaluation.

## 10.4 Data Analysis and Recommendations

### *10.4.1 Gradient-Descent Iterations*

The effect on ASR word accuracy as the number of gradient-descent iterations increases is shown in Table 10.1. Recognition results with no enhancement (baseline) and MFNS with static subtraction parameters ($\alpha(k) = 1$) are shown for comparison.

Analysis of these results shows that the optimal number of gradient-descent iterations is considerably different for each noise condition. For the more quiet conditions (idle and 35 mph with windows up), best performance is obtained with more than 20 iterations of gradient-descent optimization. For the noisier conditions, less than five optimization iterations provide the best performance (particularly for the 55-mph-with-windows-down noise condition). These three conditions also show trends of decreasing word accuracy as the number of iterations is increased above five. Since the noise conditions are approximately ordered by increasing levels of noise, it can be concluded that as the noise levels in the vehicle increase (i.e., higher speeds or open windows), the level of gradient-descent optimizations needs to be reduced in order to avoid over-optimization of the enhancement parameters.

The application of only one gradient-descent iteration provides a minimum of 0.3% improvement static MFNS, with both 35-mph scenarios improving by approximately 1%. A single iteration shows the effectiveness of a LIMA framework for improving ASR performance with minimal optimization.

The best overall performance across all five noise conditions is seen at three iterations. At this level of optimization, the 55-mph conditions both exhibit maximum performance, with two other noise conditions being only 0.1% below their best performance (IDL and 35D). The 35-mph-with-windows-up condition is the only

**Table 10.1** ASR accuracies for increasing gradient-descent iterations used in parameter optimization

| # Iterations | IDL | 35U | 35D | 55U | 55D |
|---|---|---|---|---|---|
| Baseline | 70.4 | 48.8 | 36.2 | 41.8 | 23.5 |
| $\alpha(k) = 1$ | 73.3 | 47.8 | 36.8 | 44.5 | 26.1 |
| 1 | 73.9 | 48.7 | 37.9 | 44.8 | 26.4 |
| 2 | 74.2 | 49.3 | 37.7 | 44.8 | 26.4 |
| 3 | 74.1 | 49.1 | 38.1 | 45.1 | 26.4 |
| 4 | 74.2 | 49.5 | 37.8 | 45.1 | 26.1 |
| 5 | 74.1 | 49.6 | 38.2 | 45.0 | 25.9 |
| 10 | 74.2 | 49.7 | 37.7 | 44.6 | 26.1 |
| 15 | 74.2 | 49.8 | 37.5 | 44.8 | 25.6 |
| 20 | 74.2 | 49.9 | 37.6 | 44.7 | 25.7 |
| 25 | 74.2 | 49.9 | 37.6 | 44.7 | 25.7 |

**Table 10.2** ASR results for increasing number of joint optimization iterations

| # Iterations | IDL | 35U | 35D | 55U | 55D |
|---|---|---|---|---|---|
| Baseline | 70.4 | 48.8 | 36.2 | 41.8 | 23.5 |
| $\alpha(k) = 1$ | 73.3 | 47.8 | 36.8 | 44.5 | 26.1 |
| **1** | **74.1** | **49.1** | **38.1** | **45.1** | **26.4** |
| 2 | 74.1 | 49.4 | 37.7 | 44.8 | 26.1 |
| 3 | 73.9 | 49.9 | 37.2 | 44.8 | 26.0 |
| 4 | 74.0 | 50.1 | 37.2 | 44.5 | 26.3 |
| 5 | 74.0 | 50.3 | 37.1 | 44.4 | 26.1 |
| 10 | 74.1 | 50.2 | 37.5 | 44.1 | 25.9 |

one which is well below its best performance (0.8%) but still provides improvement over the baseline and static MFNS systems. As a result, three gradient-descent iterations have been used for the remainder of the experiments in this chapter.

### *10.4.2 Joint Optimization Iterations*

Having established the most effective number of gradient-descent iterations, the number of joint optimization iterations was analyzed. Table 10.2 shows these results with the best performance across all noise conditions highlighted for clarity.

Apart from the 35-mph-with-windows-up noise condition, the results indicate that only one joint optimization iteration is required for in-car speech recognition. This result indicates that only minor changes are made to the decoded state sequences and therefore appears to be no advantage in performing more than one joint optimization iteration. Relating this observation to the results of the gradient-descent iterations experiment, if the state sequence did not change at all, the parameter optimization would continue from exactly the same position that it finished previously, and therefore, over-optimization is likely to occur as the number of joint optimization iterations increases.

This result combined with that of Sect. 10.4.1 indicates that over-optimization is a serious issue for LIMA frameworks operating in vehicular environments. It is therefore suggested that optimization iterations be kept to a minimum in order to keep the enhancement parameters generalized. The practical advantage of these findings is the ability to achieve improved ASR using LIMA frameworks whilst creating minimal processing delays due to the need for only a few optimization iterations.

### *10.4.3 LIMA Frameworks*

The LIMA frameworks listed in Sect. 10.3.4 were tested using the results obtained in the previous experiments. Table 10.3 presents the ASR results for all three

**Table 10.3** ASR results for the calibrated LIMA frameworks

| Adaptation condition | IDL | 35U | 35D | 55U | 55D |
|---|---|---|---|---|---|
| Baseline | 70.4 | 48.8 | 36.2 | 41.8 | 23.5 |
| $\alpha(k) = 1$ | 73.3 | 47.8 | 36.8 | 44.5 | 26.1 |
| Noise | 74.1 | 49.1 | 38.1 | 45.1 | 26.4 |
| Speaker | 73.6 | 49.5 | 38.2 | 44.9 | 26.5 |
| IDL | **73.7** | 49.3 | 37.8 | 44.6 | 26.8 |
| 35U | 73.8 | **49.9** | 38.6 | 45.0 | 27.0 |
| 35D | 73.0 | 49.4 | **39.2** | 45.1 | 26.7 |
| 55U | 74.2 | 49.7 | 37.9 | **45.5** | 26.8 |
| 55D | 73.1 | 49.1 | 38.2 | 44.7 | **27.1** |

calibrated frameworks. The matched calibrate-test conditions for speaker-based calibration are highlighted for clarity. Regardless of the calibration method used, the results show a global improvement over an enhancement system which does not utilize a LIMA framework.

Using matched conditions for speaker-based adaptation (i.e., employing calibration for each speaker in each noise condition) provides the best word accuracies in all cases except idle. Whilst the idle noise condition shows a 0.5% absolute decrease in word accuracy in its matched condition (as opposed to optimizing in 55U), the word accuracy performance is still an improvement over the static MFNS case (73.7% versus 73.3%). As a result, this is not seen to be a significant issue at this point in time.

In order to assess the effectiveness of the proposed dialog-based LIMA framework, all utterances occurring prior to the first optimization (or first optimization after calibration) for each speaker were ignored. This approach was required since the proposed technique requires 100% word accuracy in order to trigger optimization, a result which was achieved on only 3% of all utterances and mostly in the idle noise condition. This low number of optimization instances is due to the relatively low performance of the ASR system and nature of the recognition task which requires all ten digits to be recognized correctly.

These results of this final evaluation are summarized in Table 10.4. It should be noted that word accuracies in this table are better than in previous tables because this analysis removed a lot of utterances exhibiting poor ASR performance.

Almost all comparisons in Table 10.4 show that the proposed dialog-based LIMA framework for in-car ASR provides improved performance over the baseline enhancement system. Applying this framework can also recover losses in word accuracy incurred when using standard Mel-filterbank noise subtraction (e.g., in the two 35-mph noise conditions).

The results of this evaluation also prove the effectiveness of the proposed dialog-based framework when used with or without explicit calibration even though there are a very low number of optimization instances. For the case without calibration – which is the ideal operational behavior of such a framework since the user would be completely unaware of adaptation – global improvements over both baseline systems can be observed, with the best relative performance improvement over a

**Table 10.4** ASR results for all LIMA frameworks

| Framework | IDL | 35U | 35D | 55U | 55D |
|---|---|---|---|---|---|
| Baseline | 79.1 | 55.8 | 42.1 | 49.8 | 27.6 |
| $\alpha(k) = 1$ | 81.8 | 53.9 | 41.6 | 51.7 | 30.1 |
| Proposed dialog system | 82.6 | 55.9 | 42.3 | 53.1 | 31.1 |
| Baseline | 80.7 | 55.5 | 43.3 | 49.5 | 28.6 |
| $\alpha(k) = 1$ | 81.4 | 53.3 | 45.3 | 50.0 | 33.6 |
| Calibrated system (random) | 82.5 | 55.7 | 46.4 | 52.5 | 33.3 |
| Proposed dialog (random) | 82.3 | 57.7 | 45.5 | 52.7 | 32.3 |
| Baseline | 80.4 | 57.7 | 44.7 | 53.3 | 28.4 |
| $\alpha(k) = 1$ | 82.2 | 52.5 | 42.9 | 53.9 | 30.3 |
| Calibrated system (IDL) | 82.4 | 55.4 | 44.6 | 54.9 | 31.0 |
| Proposed dialog (IDL) | 82.9 | 55.9 | 46.0 | 55.5 | 30.9 |

system without enhancement being 16.7% in the idle condition. This particular result demonstrates the true potential of the framework to improve ASR accuracy, since utterances spoken during idle are most likely to trigger the optimization process. In comparison to the baseline enhancement system, the proposed framework shows relative improvements of between 1.2% and 4.4% in this mode of operation.

There are also noticeable improvements of the calibration-only LIMA framework, particularly one performing calibration during idle. In this case, the relative improvements range from 1.2% to 2.8% (excluding the marginal decrease in performance in the 55D noise condition). Given that most users will first speak to in-car dialog systems when entering their vehicle, this result verifies the potential of the proposed framework to be incorporated with a calibration session to produce further improvements in system performance.

Considering the operation of the proposed dialog-based system, there is potential for a loss of generality if a particular noise condition is consecutively optimized (as per the results in Table 10.2). The consistent improvements in Table 10.4, however, indicate that this is not an issue as regular changes in noise conditions seem to allow the optimization process to effectively track the internal noise conditions and set the enhancement parameters appropriately.

## 10.5 Conclusions

This chapter has reviewed likelihood-maximizing frameworks using Mel-filterbank noise subtraction for in-car speech recognition. A new LIMA framework based on a user-confirmation speech dialog system has been proposed. This framework has been evaluated against calibrated LIMA frameworks utilizing different adaptation scenarios.

Experiments have shown that with the proposed LIMA framework, minimal optimization is required for the best average recognition performance in car environments. This permits pseudo real-time operation of LIMA frameworks whilst

still providing improvements over standard speech enhancement techniques. The proposed dialog-based framework provides improved recognition performance over calibration-only systems; this effect is attributed to the ability to continually update enhancement parameters according to changes in noise conditions.

**Acknowledgments** Parts of the work presented here were funded through the Australian Cooperative Research Centre for Advanced Automotive Technology (AutoCRC).

## References

1. Kleinschmidt T, Mason M, Wong E, Sridharan S (2009) The Australian english speech corpus for in-car speech processing. In: Proceedings of ICASSP. IEEE Computer Society, Washington, DC, pp 4177–4180
2. Seltzer ML, Raj B, Stern RM (2004) Likelihood-maximizing beamforming for robust hands-free speech recognition. IEEE Trans Speech Audio Process 12(5):489–498
3. BabaAli B, Sameti H, Safayani M (2009) Likelihood-maximizing-based multiband spectral subtraction for robust speech recognition. EURASIP J Adv Signal Process 878105:1–15
4. Shi G, Aarabi P, Jiang H (2007) Phase-based dual-microphone speech enhancement using a prior speech model. IEEE Trans Audio Speech Lang Process 15(1):109–118
5. Lockwood P, Boudy J, Blanchet M (1992) Non-linear spectral subtraction (NSS) and hidden Markov models for robust speech recognition in car noise environments. In: Proceedings of the ICASSP, San Francisco, pp 265–268
6. Wu K-G, Chen P-C (2001) Efficient speech enhancement using spectral subtraction for car hands-free applications. In: Proceedings of IEEE international conference on consumer electronics, Los Angeles, pp 220–221
7. Kleinschmidt T, Boyraz P, Bořil H, Sridharan S, Hansen JHL (2009) Assessment of speech dialog systems using multi-model cognitive load analysis and driving performance metrics. In: Proceedings of IEEE international conference on vehicular electronics & safety, Pune, pp 167–172
8. Boll S (1979) Suppression of acoustic noise in speech using spectral subtraction. IEEE Trans Acoust Speech Signal Process 27(2):113–120
9. Nasersharif B, Akbari A (2006) A framework for MFCC feature extraction using SNR-dependent compression of enhanced Mel-filter bank energies. INTERSPEECH, 1632-Mon1A20.3
10. Kleinschmidt T (2010) Robust speech recognition using speech enhancement. PhD thesis, Queensland University of Technology
11. Lee B, Hasegawa-Johnson M, Goudeseune C, Kamdar S, Borys S, Liu M, Huang T (2004) AVICAR: audio-visual speech corpus in a car environment. In: Proceedings of INTERSPEECH, Jeju Island, pp 2489–2492
12. Kleinschmidt T, Dean D, Sridharan S, Mason M (2007) A continuous speech recognition protocol for the AVICAR database. In: Proceedings of ICSPCS, Gold Coast, pp 339–344
13. Young S, Evermann G, Gales M, Hain T, Kershaw D, Liu X, Moore G, Odell J, Ollason D, Povey D, Valtchev V, Woodland P (2006) The HTK Book. Cambridge University: Engineering Department, Cambridge

# Chapter 11
# Feature Compensation Employing Variational Model Composition for Robust Speech Recognition in In-Vehicle Environment

**Wooil Kim and John H.L. Hansen**

**Abstract** This chapter proposes a novel model composition method to improve speech recognition performance in time-varying background noise conditions. It is suggested that each order of the cepstral coefficients represents the frequency degree of changing components in the envelope of the log-spectrum. With this motivation, in the proposed method, variational noise models are generated by selectively applying perturbation factors to a basis model, resulting in a collection of various types of spectral patterns in the log-spectral domain. The basis noise model is obtained from the silent duration segments of input speech. The proposed Variational Model Composition (VMC) method is employed to generate multiple environmental models for our previously proposed feature compensation method. Experimental results prove that the proposed method is considerably more effective at increasing speech recognition performance in time-varying background noise conditions with +20.80% relative improvement in word error rates for the CU-Move real-life in-vehicle corpus, compared to an existing single model–based method.

**Keywords** Feature compensation • In-vehicle environment • Multiple model • Robust speech recognition • Variational model composition (VMC)

## 11.1 Introduction

Acoustic difference between training environments and conditions where actual speech recognition systems operate is one of the primary factors that degrade speech recognition accuracy, and the presence of background noise is one major

W. Kim (✉) • J.H.L. Hansen
Center for Robust Speech Systems (CRSS), Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas, Richardson, TX, USA
e-mail: wikim@utdallas.edu; john.hansen@utdallas.edu

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_11, 

factor. This is typically true for in-vehicle speech systems which face the problem of robust speech recognition in order to address a range of severe changing background noise conditions.

To minimize this mismatch, extensive research has been conducted in recent decades with the goal of achieving successful results for slowly changing background noise, including many types of speech/feature enhancement methods and model adaptation techniques [1–10]. However, these methods continue to suffer from ineffectiveness in time-varying background noise conditions, where the noise characteristics need to be effectively estimated as time elapses. Recently, missing-feature methods have shown promising results [11, 12], which utilize no prior knowledge of the background noise [13]. Unfortunately, they are highly dependent on the ability of reliable component estimation, still resulting in performance degradation in time-varying noise conditions.

In this study, a novel model composition method is proposed to address time-varying background noise such as in-vehicle environments for improved speech recognition. Our motivation is that each order of the cepstral coefficients represents a frequency degree of the changing components in the log-spectrum envelope [14]. In the proposed method, variational noise models are generated by selectively applying perturbation factors to a basis model in the cepstral domain to obtain various types of spectral patterns. The proposed variational model composition method is employed to generate multiple environmental models for our previously proposed feature compensation method [9, 10]. The proposed method will be evaluated on the CU-Move corpus which contains a range of acoustic signals expected to be observed during real-life car driving.

This chapter is organized as follows. We first review the CU-Move [15] corpus used for this study in Sect. 11.2. In Sect. 11.3, the motivation of the proposed variational model composition method is presented and the detailed procedure described. A multiple model–based feature compensation method as an application of the proposed study is presented in Sect. 11.4, which has been developed in our previous study. Representative experimental results are presented and discussed in Sect. 11.5. Finally, in Sect. 11.6, we conclude our work.

## 11.2 CU-Move Corpus

The CU-Move project [15] was designed to develop reliable car navigation systems employing a mixed initiative dialog. This requires robust speech recognition across changing acoustic conditions. The CU-Move database consists of five parts: (1) command and control words, (2) digit strings of telephone and credit card numbers, (3) street names and addresses, (4) phonetically balanced sentences, and (5) Wizard of Oz interactive navigation conversations. A total of 500 speakers, balanced across gender and age, produced over 600 GB of data during a 6-month collection effort across the United States. The database and noise conditions are discussed in detail in [15]. We point out that the noise conditions are changing with time and are quite different in terms of SNR, stationarity, and spectral structure. The challenge in

addressing these noise conditions is that they might be changing depending on the specific car and road conditions. In this study, we select 20 speakers from approximately 100 speakers in Minneapolis, Minnesota (i.e., Release 1.1A) and employ the connected single digits portion that contains speech under a range of varying complex in-vehicle noise events/conditions.

## 11.3 Variational Model Composition

In this section, a novel method is proposed to effectively estimate the time-varying background noise contained in a speech utterance by using information contained in non-speech segments. As initial knowledge for our discussion, the effect on log-spectral coefficients caused by adding a gain to the cepstral coefficients is presented. From fundamentals of the cepstrum, which is obtained by a discrete cosine transformation (DCT) of the log-spectrum, each order of the obtained cepstral coefficients represents a frequency of the log-spectrum envelope changes (i.e., frequency [14]). For example, the lower-order cepstral coefficients indicate a measure of the slowly changing components in the envelope of the log-spectrum, having the 0th cepstral coefficient represent a DC component (i.e., energy) of the log-spectrum at a frame. Therefore, applying a weight to each order of the cepstral coefficients could generate a variation of the original cepstrum in terms of the frequency of envelope change along the log-spectral axis.

Assume that a vector of cepstral coefficients $\mathbf{x}$ consists of 0th to $(N-1)$th coefficients. A variation of the cepstrum vector can be obtained by adding a gain vector g as follows:

$$\tilde{\mathbf{x}} = \mathbf{x} + \mathbf{g} \tag{11.1}$$

If the gain is applied only on the 0th coefficient such as $\mathbf{g} = [\pm g, 0, 0, \ldots, 0]$, the log-spectral coefficients of the obtained variation will have a different energy level from the original log-spectrum, which can be obtained by an inverse DCT of the cepstral coefficients. Figure 11.1, (a) shows log-spectra of the variations which are generated by weighting the zeroth cepstral coefficient. The plain solid line indicates the original log-spectral coefficients, and the lines with solid or empty circles indicate the resulting log-spectrum by weighting + g and − g at the zeroth cepstral component, respectively. We can see the two variations have different energy levels while maintaining an identical spectral envelope shape with the original coefficients. Plots (b) and (c) present the log-spectra of the variations generated by applying weights only to the first and fourth cepstral components, respectively. The variations in (b) show a smooth change of the envelope, and plots of the variations in (c) are varying relatively faster.

With this motivation, we believe that a range of models could be generated by applying a combination of weights to an original model in the cepstral domain. In our proposed method, it is assumed that (1) a basis noise model can be obtained from periods of "silence" (e.g., non-speech) within the speech stream and (2) the target

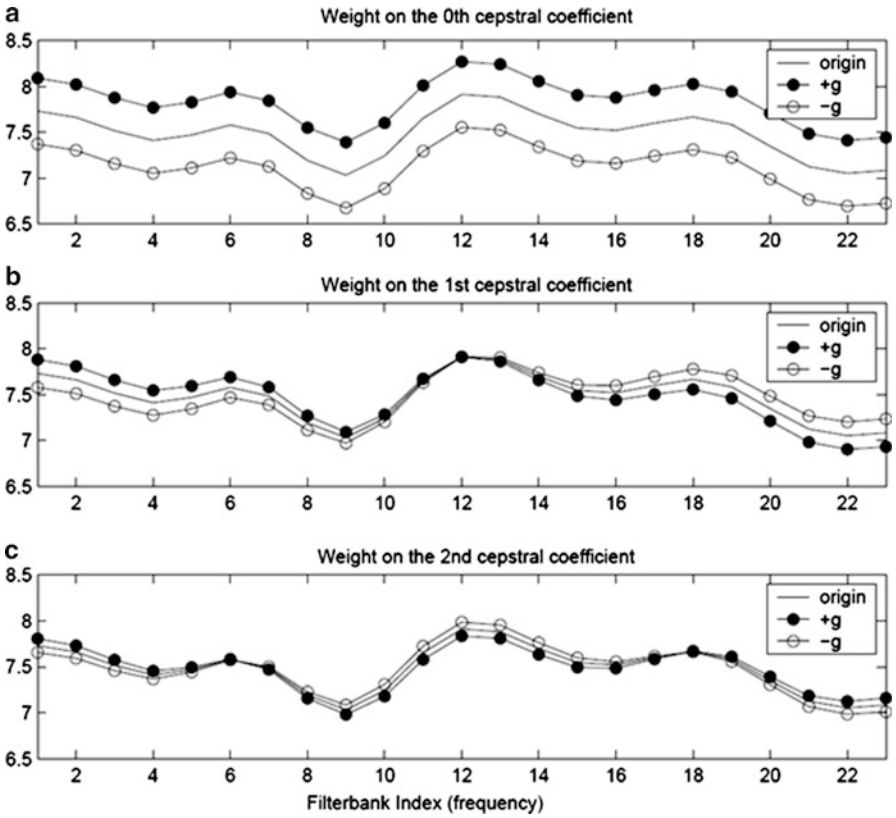



**Fig. 11.1** Example of variations of log-spectral coefficients generated by applying a weight to the (**a**) zeroth, (**b**) first, and (**c**) fourth cepstral coefficients

time-varying noise included in the speech duration would reflect variations of the estimated basis model. The variational models are generated by selectively applying weights to each component of the mean vector of the basis model in the cepstral domain. Here, we propose a novel algorithm to generate a collection of variational noise models as follows:

Step 1 – Basis Model Estimation
A basis noise model is obtained from silent segments within the input speech, which generally exists at the beginning and end parts of an utterance. The model is estimated as a Gaussian pdf $(\mu,\sigma^2)$ in the cepstral domain.

Step 2 – Variational Component Determination
The $V$ largest components $\{v_1,v_2,\ldots,v_V\}$ in the variance vector $\sigma^2$ are selected. These are named *variational components,* which are considered highly variable components and are size-ordered ranked as follows:

$$\sigma_{v_1} \geq \sigma_{v_2} \geq \cdots \geq \sigma_{v_V} \tag{11.2}$$

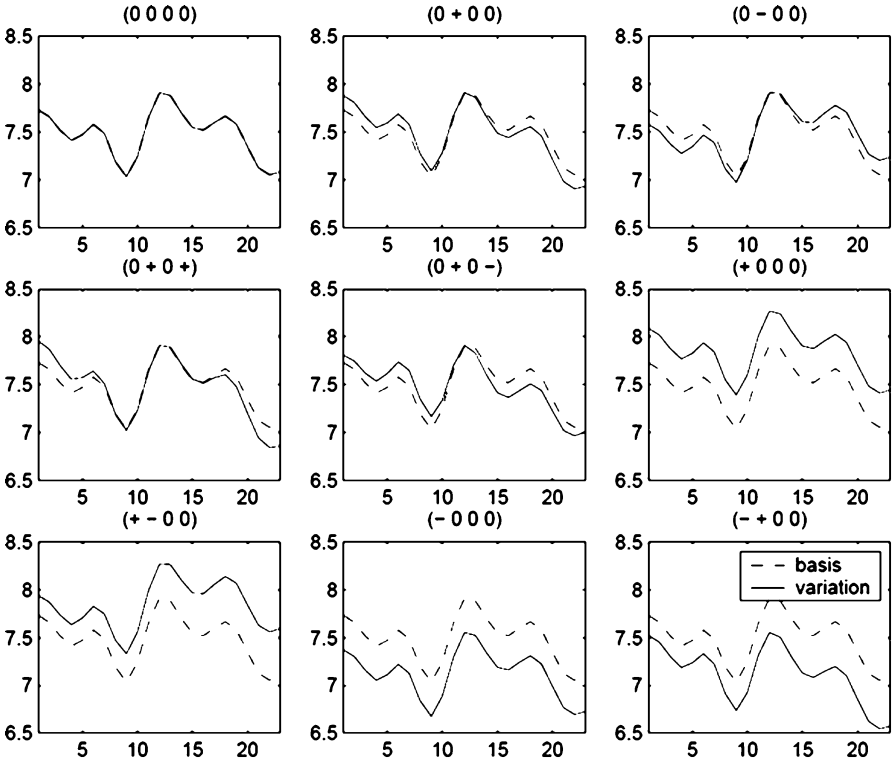



**Fig. 11.2** Mean parameters of variational models in log-spectral domain generated by the proposed model composition methods. Four-digit symbol of each plot indicates a combination of perturbation factors (i.e., $-\alpha$, 0, or $+\alpha$) for the selected four variational components

Step 3 – Model Composition by Mean Perturbation
A variation of the mean vector is generated by selectively applying the *perturbation factor* $f_p$ to the determined variational components of the cepstral coefficients $v_1$ to $v_V$ as follows:

$$\tilde{\mu}_i = \begin{cases} \mu_i(1+f_p) & \text{if } i \in \{v_1, v_2, \ldots, v_V\} \\ \mu_i & \text{otherwise} \end{cases}, \tag{11.3}$$

where $f_p = -\alpha$, 0, or $+\alpha$ and the $\alpha$ is a small positive value which we determine heuristically. The obtained model collection $\{\tilde{\lambda} = (\tilde{\mu}, \sigma^2)\}$ consists of a total $3^V$ number of generated variational models as a result of combinations of the 3-type gains of the $V$ variational components.

In this study, we employed four variational components for the proposed model composition method. Figure 11.2 demonstrates several representative variational noise models (i.e., mean parameters in the log-spectral domain) obtained by the proposed model composition algorithm, showing various types of spectral patterns generated by

applying combinations of weights to the selected variational cepstral components using a basis model which is presented as the dashed line in each figure. Here, after selecting the four coefficients with heist variance from Eq. 10.2, we apply a trilevel perturbation factor (i.e., either $-\alpha$, 0, or $+\alpha$). The figure clearly illustrates the effect of perturbation of the original basis model (i.e., (0, 0, 0, 0)) in the upper left corner.

## 11.4 PCGMM-Based Feature Compensation Employing Variational Model Composition

In this section, to address time-varying background noise for speech recognition, the Parallel Combined Gaussian Mixture Model (PCGMM)–based feature compensation algorithm [9, 10] employing the proposed variational model composition method is presented. In the PCGMM method, the parameters of the noise-corrupted speech model are obtained through a model combination procedure using clean speech and noise models independently. A constant bias transformation of the mean parameters of the clean speech model is assumed in the cepstral domain under an additive noisy environment as follows:

$$\boldsymbol{\mu}_{\mathbf{y},k} = \boldsymbol{\mu}_{\mathbf{x},k} + \mathbf{r}_k, \tag{11.4}$$

where $\boldsymbol{\mu}_{\mathbf{y},k}$ and $\boldsymbol{\mu}_{\mathbf{x},k}$ denote mean vectors of the $k$th component of GMMs for noise-corrupted speech $\mathbf{y}$ and clean speech $\mathbf{x}$, respectively. The bias term $\mathbf{r}_k$ is estimated with Eq. 11.4 once the mean parameters of the clean speech model and corresponding noise-corrupted speech model are obtained.

Utilizing multiple numbers of environmental models is considered to be effective for compensating input features adaptively under time-varying noisy conditions [10]. In the multiple model method, a sequential posterior probability of each possible environment is estimated over the incoming noisy speech. Given the input noisy-speech feature vectors $\boldsymbol{Y}_t = [y_{t-d+1}, y_{t-d+2}, \ldots, y_t]^{\mathrm{T}}$ over a $d$ interval, the sequential posterior probability of a specific environment GMM $G_i$ among all models can be written as:

$$p(G_i|Y_t) = \frac{P(G_i)p(Y_{t-1}|G_i)p(y_t|G_i)}{\sum_{e=1}^{E} P(G_e)p(Y_{t-1}|G_e)p(y_t|G_e)}, \tag{11.5}$$

where $p(Y_{t-1}|G_i) = \prod_{\tau=1-d+1}^{t-1} p(y_\tau|G_i)$ and $P(G_i)$ is the a priori probability of each environment $i$, represented as a GMM. Based on Eq. 11.5, the clean feature at frame $t$ is reconstructed as the weighted combination of the compensation terms obtained from a set of $E$ multiple environments as follows:

$$\hat{x}_{t,MMSE} \cong y_t - \sum_{e=1}^{E} p(G_e|Y_t) \sum_{k=1}^{K} r_{e,k} p(k|G_e, y_t), \tag{11.6}$$

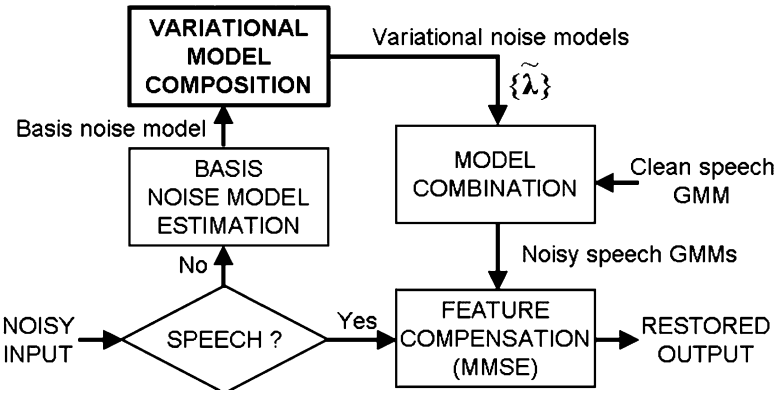



**Fig. 11.3** Block diagram of the PCGMM method employing the proposed variational model composition

where $r_{e,k}$ is a constant bias term from the $k$th Gaussian component of the $e$th environment model, and $p(k|G_e, y_t)$ is the posterior probability for environment $G_e$.

The variational noise models obtained by the proposed variational model composition method in this study are used to generate the environmental models $\{G_e\}$, which are estimated through the model combination procedure using the clean speech GMM and the obtained variational noise models. A uniform prior probability is set on all obtained noise models in this study. Figure 11.3 demonstrates the resulting block diagram of PCGMM-based feature compensation employing the proposed new variational model composition method.

## 11.5 Experimental Results

As test data for performance evaluation, connected single digits portions from CU-Move corpus were selected. We established an experiment setup which is identical to the Aurora2 evaluation framework [16]. The task is connected English-digits consisting of 11 words. Each whole word is represented by a continuous density HMM with 16 states and three mixtures per state. In addition to the digits, two silence models (i.e., normal silence and short pause) are used.

The feature extraction algorithm suggested by the European Telecommunication Standards Institute (ETSI) was employed for the experiments [17]. The zeroth cepstral coefficient was used instead of log energy for the sake of convenience in model combination implementation. After extracting the 13th order cepstrum, the first and second order time derivatives are included during the decoding procedure (a total of 39 dimensional feature vector). The HMM parameters were estimated using 8,840 clean speech training samples included in Aurora2, and performance was evaluated on the selected test set of CU-Move corpus. The test set consists of 464 utterances (length of 50 min) spoken by ten different speakers (five males

**Table 11.1** Performance of baseline system and existing methods on CU-Move corpus (WER,%)

| | |
|---|---|
| Baseline | 70.02 |
| SS + CMN | 39.90 |
| ETSI AFE | 48.31 |
| VTS | 31.45 |

and five females) in real-life in-vehicle conditions, which were collected in Minneapolis, Minnesota [15]. Data was down-sampled to 8 kHz and reflected a 9.50 dB SNR on average which was obtained using NIST Speech Quality Assurance software [18].

The performance of the baseline system (no compensation) is examined with comparison to several existing preprocessing algorithms in terms of environmental robustness for speech recognition. Spectral Subtraction (SS) and Cepstral Mean Normalization (CMN) were selected as conventional algorithms. These represent the most commonly used techniques for additive noise suppression and removal of channel distortion, respectively. In spectral subtraction [19], the subtraction factor and flooring factor are set at 4.0 and 0.2, respectively, and background noise is estimated using the minimum statistics method with a time delay of approximately 250 ms. For cepstral mean normalization, the average value of the cepstrum over the current input utterance was subtracted from each frame. AFE (Advanced Front-End) algorithm developed by ETSI was also evaluated as one of state-of-the-art methods which contains an iterative Wiener filter and cepstral histogram equalization [20]. We also evaluated another feature compensation method, the VTS (Vector Taylor Series) algorithm, for performance comparison where the noisy speech GMM is adaptively estimated using the EM algorithm over each test utterance [7]. Table 11.1 demonstrates performance of the baseline system and existing algorithms.

Next, we discuss the determination of perturbation factor for the proposed variational model composition by showing performance versus a change in the perturbation factor. Performance was evaluated using the speech recognition ability of the reconstructed speech by the PCGMM method which employs the variational model composition method. To see the performance in various types of background noise conditions, Aurora2 test database [16] was used. Here, we employed Subway, Babble, Car, and Exhibition noise conditions which were included in "Set A" of Aurroa2 database. Figure 11.4 presents the performance dependency on the perturbation factor $f_p$. The WER performance was plotted as a function of $\alpha$ from 0 to 0.1 for $f_p$ over four kinds of background noise conditions. Here, the WER is an average value of all SNR conditions (i.e., 0, 5, 10, 15, and 20 dB) for each background noise, and the plot with the solid circle presents the average performance of four kinds of noise conditions. The performance of the case with $\alpha = 0$ indicates the basic PCGMM method employing only a basis model without the variational model composition method, which is a target system for performance comparison of the proposed VMC-PCGMM. It is interesting to note that each plot shows a concave shape formulating a local minimum around 0.05–0.07 of $\alpha$ values. These results suggest that a suitable value for $\alpha$ needs to be determined to bring an effective performance to the proposed variational noise model composition method.

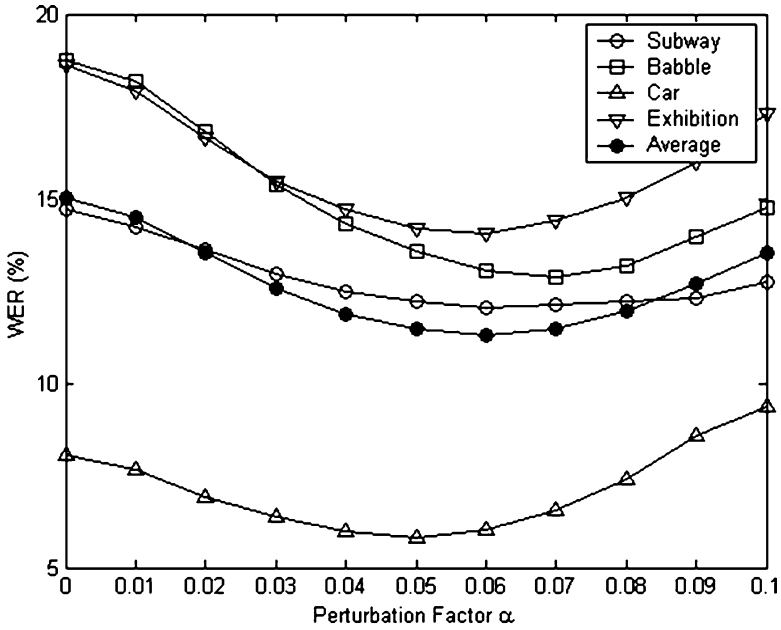



**Fig. 11.4** Recognition performance of VMC-PCGMM versus change of $\alpha$ for perturbation factor on Aurora2 database (WER,%)

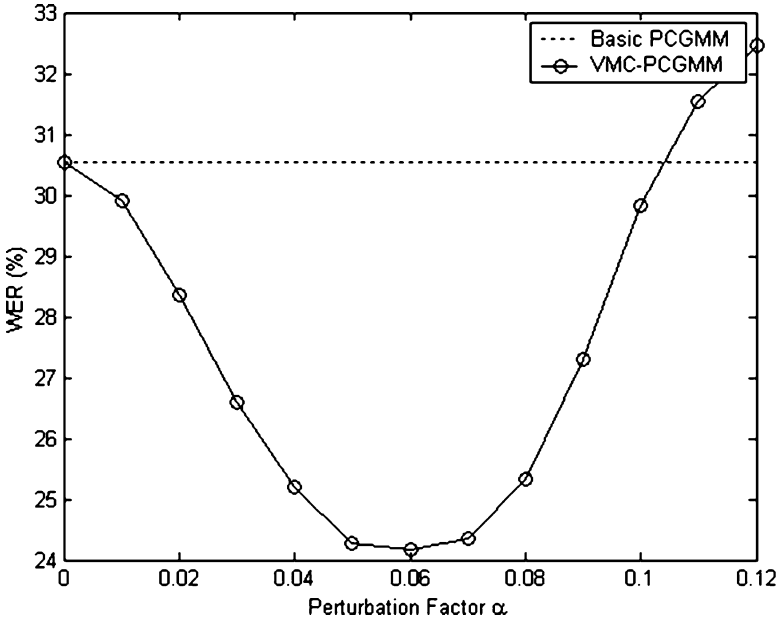



**Fig. 11.5** Recognition performance of VMC-PCGMM versus change of $\alpha$ for perturbation factor on CU-Move corpus (WER,%)

**Table 11.2** Performance comparison for the CU-Move corpus (WER,%)

| | |
|---|---|
| PCGMM | 30.53 |
| VMC-PCGMM | 24.18 |
| (Relative Improvement) | (20.80) |

Figure 11.5 shows the recognition performance of the reconstructed speech by the proposed VMC-PCGMM method as a change in the perturbation factor on the CU-Move corpus. Results here are similar to the Aurora2 database (i.e., Fig. 11.4) with the lowest WER at $\alpha$=0.06 for the perturbation factor $f_p$. This result indicates that our experiments using Aurora2 database would have suggested a guideline for determining the perturbation factor when applying the proposed VMC method to real-life in-vehicle environments. Table 11.2 shows a performance comparison of the proposed VMC-PCGMM ($\alpha$=0.06) to the basic PCGMM over the CU-Move corpus with a +20.80% relative improvement in WER. These results demonstrate that VMC-PCGMM method brings a significant improvement compared to the basic PCGMM and other conventional methods on real-life in-vehicle conditions.

## 11.6 Conclusion

In this study, a novel model composition method was proposed to improve speech recognition in time-varying background noise conditions such as in-vehicle environments. In the proposed method, a basis noise model was estimated from non-speech segments, and variational noise models were generated by selectively applying the perturbation factors on the variational cepstral components which are determined by the variance of the basis model. The proposed model composition method was employed to generate multiple environmental models for the PCGMM algorithm. Experimental results demonstrated that the proposed method is considerably more effective at increasing speech recognition performance in time-varying background noise conditions. We obtained a +20.80% relative improvement in WER for CU-Move real-life in-vehicle corpus compared to the single-model PCGMM method. This proves that the variational noise model composition generates a noise space that can effectively address the time-varying nature of background noise.

## References

1. Boll SF (1979) Suppression of acoustic noise in speech using spectral subtraction. IEEE Trans on Acoustics, Speech and Signal Proc 27:113–120
2. Ephraim Y, Malah D (1984) Speech enhancement using minimum mean square error short time spectral amplitude estimator. IEEE Trans on Acoustics, Speech and Signal Proc 32 (6):1109–1121

3. Hansen JHL, Clements MA (1991) Constrained iterative speech enhancement with application to speech recognition. IEEE Trans on Signal Proc 39(4):795–805
4. Gauvain JL, Lee CH (1994) Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. IEEE Trans on Speech and Audio Proc 2(2):291–298
5. Leggetter CJ, Woodland PC (1995) Maximum likelihood linear regression for speaker adaptation of continuous density HMMs. Comput Speech Lang 9:171–185
6. Gales MJF, Young SJ (1996) Robust continuous speech recognition using parallel model combination. IEEE Trans on Speech and Audio Proc 4(5):352–359
7. Moreno PJ, Raj B, Stern RM (1998) Data-driven environmental compensation for speech recognition: a unified approach. Speech Commun 24(4):267–285
8. Kim NS (2002) Feature domain compensation of nonstationary noise for robust speech recognition. Speech Commun 37:231–248
9. Kim W, Kwon O, Ko H (2004) PCMM-based feature compensation schemes using model interpolation and mixture sharing. ICASSP-2004 1:989–992
10. Kim W, Hansen JHL (2009) Feature compensation in the cepstral domain employing model combination. Speech Commun 51(2):83–96
11. Cook M, Green P, Josifovski L, Vizinho A (2001) Robust automatic speech recognition with missing and unreliable acoustic data. Speech Commun 34(3):267–285
12. Raj B, Seltzer ML, Stern RM (2004) Reconstruction of missing features for robust speech recognition. Speech Commun 43(4):275–296
13. Kim W, Stern RM (2006) Band-independent mask estimation for missing-feature reconstruction in the presence of unknown background noise. ICASSP-2006 305-308, May 2006.
14. Jr Deller JR, Hansen JHL, Proakis JG (2000) Discrete-Time Processing of Speech Signals. IEEE Press, New York
15. Hansen JHL, Zhang X, Akbacak M, Yapanel U, Pellom B, Ward W, Angkititrakul P (2004) CU-Move: Advances for in-vehicle speech systems for route navigation. In: Abut H, Hansen JHL, Taketa K (eds) DSP for in-vehicle and mobile systems. Springer, USA, Chap. 2
16. Hirsch HG Pearce D (2000) The AURORA experimental framework for the performance evaluations of speech recognition systems under noisy conditions. ISCA ITRW ASR2000. Paris, France
17. ETSI standard document (2000) ETSI ES 201 108 v1.1.2 (2000–04)
18. NIST Speech Quality Assurance (SPQA) package version 2.3, http://www.nist.gov/speech
19. Martin R (1994) Spectral subtraction based on minimum statistics. EUSIPCO-94 1182–1185
20. ETSI standard document (2002) ETSI ES 202 050 v1.1.1 (2002–10)

# Chapter 12
# Dual-Channel Speech Enhancement Using a Perceptual Filterbank for Hands-Free Communication

**Jongsung Yoon, Kihyeon Kim, Jounghoon Beh, Robert H. Baran, and Hanseok Ko**

**Abstract** We investigate a dual-channel speech enhancement method using perceptual adaptive noise suppressor, which improves perceptual quality of speech in automobile environment for hands-free communication. In particular, the perceptual adaptive noise suppressor, which is composed of a Mel-based perceptual filterbank, an adaptive filter, and a speech modification block, estimates the envelope of the desired speech by suppressing the nonspeech components. Experiments indicate that the proposed scheme shows 8.06 dB of NR improvement and 0.70 of PESQ score improvement compared to the Transfer Function Generalized Sidelobe Canceller structure alone.

**Keywords** Driver assistance • Dual-channel speech enhancement • Hands-free communication • In-vehicle speech technology

## 12.1 Introduction

Recently, the significance of multi-microphone-based speech enhancement has increased as the needs of hands-free communication systems grow, especially in in-vehicle situations. In this chapter, an efficient multichannel speech enhancement algorithm is presented, which improves the speech quality while minimizing the directional interference and ambient noise.

J. Yoon • K. Kim • R.H. Baran • H. Ko (✉)
Department of Electronics and Computer Engineering, Korea University, 5Ka-1 Anam-dong, Seongbuk-Gu, Seoul 136713, South Korea
e-mail: hsko@korea.ac.kr

J. Beh
Institute for Advanced Computer Studies, University of Maryland, College Park, MD, USA

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_12, 

The conventional beamforming methods, such as the linearly constrained minimum variance (LCMV) [1] and the generalized sidelobe canceller (GSC), can reduce interference from undesired directions by exploiting the correlation among the noise signals of different sensors [2]. However, the beamformer cannot avoid suffering from high computational burden when the adaptive filter must be long enough to effectively suppress the noise. Hence, this aspect is not favorable for the system to be embedded on vehicular communication devices.

To solve this problem, we propose a novel algorithm which is based on spectral magnitude modification using the structure of the generalized sidelobe canceller. The envisioned algorithm applies an auditory filterbank on the primary signal, output of the fixed beamformer, and the noise reference signal, output of the blocking matrix, in order to estimate the spectral samples of noise components. Then, these samples are fed to the gain filter for spectral modification so that the optimal spectral envelope of the desired signal can be obtained. This structure provides unique advantages over traditional beamforming methods including improvement of the perceptual quality of speech, robustness against the stationary ambient noise, and high computational efficiency. We develop the envisioned algorithm on the basis of a dual-microphone array structure. In order to obtain the improved performance, we consider the optimal combination using conventional adaptive noise cancellation which is executed in general short-time Fourier transform domain.

## 12.2 Dual-Channel Speech Enhancement

### *12.2.1 Transfer Function Generalized Sidelobe Canceller (TFGSC)*

The basic GSC structure is composed of a fixed beamformer (FBF), a blocking matrix (BM), and a noise canceller filter (NC). The FBF forms a beam in the look direction so that the acoustic signal from the desired speaker is passed while interfering noises are suppressed. Then, the BM blocks the desired signal and produces a noise reference signal. The NC generates a replica of the component which is included in the FBF output and is correlated with the interference. An enhanced speech signal is obtained by subtracting the replica from the output of the FBF. Conventionally, these processes are often described in terms of sampled data representation. The broadband GSC expression, which is based on general transfer functions (TF) of room impulse responses (RIR), has recently been introduced [3]. Compared with the simple attenuation-and-delay assumption on RIRs, the TF-based BM forms a sharp null in the look direction so that the leakage signal of desired speech is more favorably attenuated. Ideally, the BM would convey a pure noise reference input to the NC. Moreover, use of TFs in the FBF provides the ability to keep the desired signal free from distortion in a highly reverberant room condition. Gannot et al. developed this concept based on the transfer function ratio (TFR) and constructed an adaptive GSC, so-called TFGSC [2]. In Fig. 12.1,

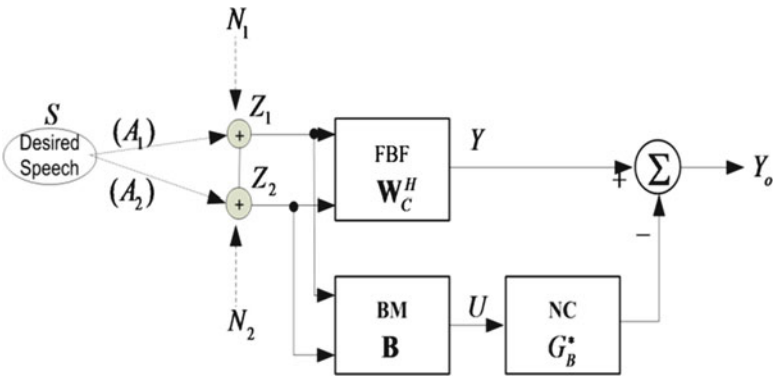



**Fig. 12.1** Schematic diagram of TFGSC

a schematic diagram of the dual-channel TFGSC is shown with the signal propagation model in the frequency domain.

The transfer function ratio $H$ is defined by

$$H = \frac{A_2}{A_1}. \tag{12.1}$$

Through the FBF, primary signal is given by

$$Y = \mathbf{W}_C^H \mathbf{Z} = \frac{1}{1+|H|^2}\begin{bmatrix}1 & H^*\end{bmatrix}\begin{bmatrix}Z_1\\ Z_2\end{bmatrix} = A_1 S + \frac{1}{1+|H|^2}[N_1 + H^* N_2]. \tag{12.2}$$

The FBF forms a beam in the look direction to pass speech and outputs a signal consisting of the distortionless speech and noise components including both the directional interference and the in-vehicle ambient noise. Next, BM forms a null beam to block speech and produces the noise reference signal:

$$U = \mathbf{B}^H \mathbf{Z} = \begin{bmatrix}-H & 1\end{bmatrix}\begin{bmatrix}Z_1\\ Z_2\end{bmatrix} = -HN_1 + N_2. \tag{12.3}$$

The noise reference signal generated goes to an NC block, and it constructs a filter, $\hat{G}_B^*$, to estimate and eliminate the noise component in FBF output via the general wiener filter solution as [4]

$$\hat{G}_B^* = \frac{E[UY_c]^H}{E[UU^H]} = \frac{\Phi_{UY}^*}{\Phi_{UU}} \tag{12.4}$$

$$Y_o = Y - \hat{G}_B^* U. \tag{12.5}$$

Normalized least mean squares (NLMS) algorithm is implemented for adaptive noise canceller [5, 6]:

$$\hat{G}_B(k,t+1) = \hat{G}_B(k,t) + \mu \frac{U(k,t)Y_o(k,t)}{P_{est}(k,t)}, \tag{12.6}$$

in which the time–frequency index returns to describe the update in short-time Fourier transform domain. In (12.6), the adaptation term is controlled by the power estimate of the input sensor signals:

$$P_{est}(k,t) = \alpha P(k,t-1) + (1-\alpha)\sum_{i=1}^{2} |Z_i|^2, \tag{12.7}$$

where $\alpha$is a forgetting factor. Then, the resulting system output is given by

$$Y(k,l) = Y_C(k,l) - \hat{G}_B^*(k,l)U(k,l). \tag{12.8}$$

A high computational burden occurs in the TFGSC when the number of adaptive filter coefficients is large enough to cover the signal path in a reverberant chamber. A save/add method is applied to perform a linear convolution using FFT. It necessitates a computationally efficient adaptive noise suppression filter while keeping the advantage of TFGSC.

### 12.2.2 *Perceptually Adaptive Noise Suppressor (PANS) Based on TFGSC*

The structure of the PANS based on TFGSC is shown in Fig. 12.2. The PANS is composed of three blocks: a fixed beamformer (FBF), a blocking matrix (BM), and a perceptually adaptive noise suppressor (PANS). It is used to estimate the spectral envelope (SE) of the desired speech signal. As shown in Fig. 12.3, an auditory filterbank such as the Mel-filterbank or the equivalent rectangular bandwidth characterizes the PANS [7, 8]. The filterbank is composed of band-pass filters imaging the effect of auditory masking. Accordingly, specific frequency resolution of a human auditory system is provided. As shown in Fig. 12.2, the filterbank outputs the auditory SE of the primary signal $\tilde{Y}$ and that of the reference noise $\tilde{U}$. Then, an adaptive filter estimates the noise SE $\tilde{N}$ of the primary signal with the input $\tilde{U}$. Given the estimate $\hat{\tilde{N}}$, the spectral modification is executed to obtain the desired speech as

$$\hat{S} = F_{itp}\left(\left[1 - \alpha\hat{\tilde{\xi}}\right]^{0.5}\right)Y, \tag{12.9}$$

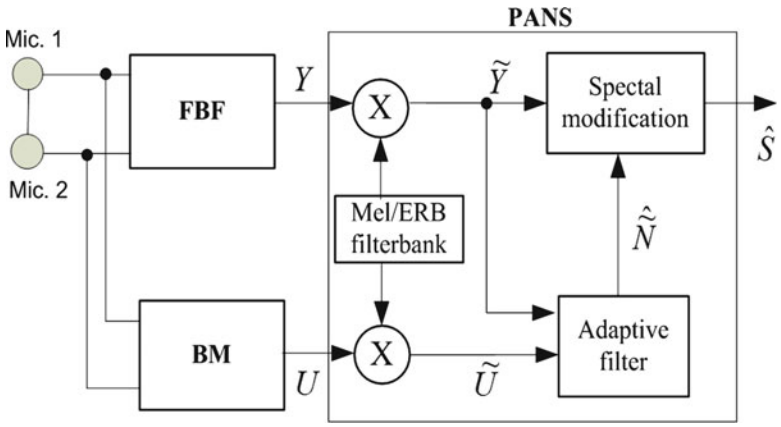



**Fig. 12.2** Schematic diagram of PANS based on TFGSC

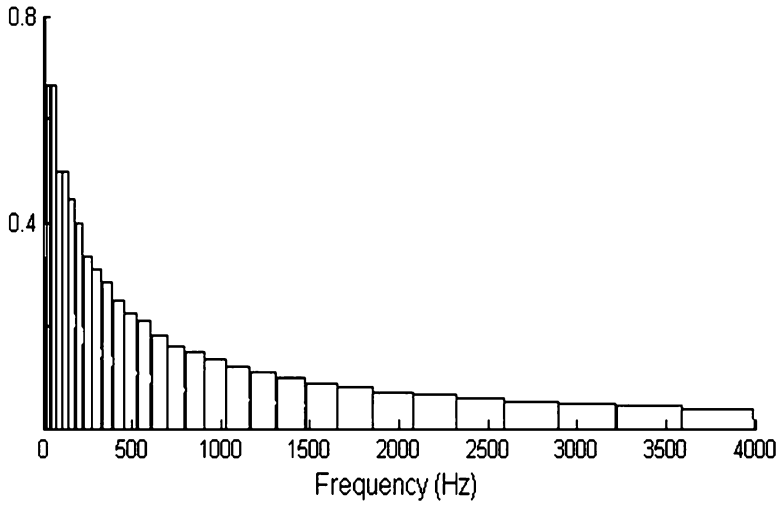



**Fig. 12.3** Frequency response of an ERB filterbank [10]

where $\alpha$ is a parameter to control the noise suppression, and the power ratio $\hat{\tilde{\xi}}$ is defined by $\hat{\tilde{N}}/\tilde{Y}$. Since the SE samples only appear at center frequencies of the filterbank, the function $F_{itp}$ is used to interpolate the power ratio samples $\hat{\tilde{\xi}}$ in the frequency domain. With the auditory filterbank and spectral modification, the proposed structure has the improved perceptual quality of an enhanced speech while minimizing the number of coefficients in the adaptive filter. Moreover, the system also promises to have the robustness against the in-vehicle ambient noise. This is based on the fact that the adaptive filter provides an SE estimate including overall noise components.

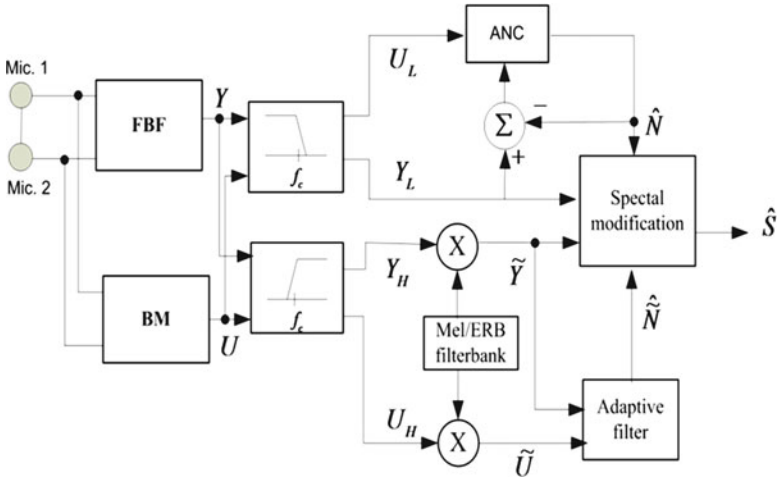



**Fig. 12.4** Schematic diagram of combination of the PANS and ANC

However, this approach cannot avoid speech distortion due to the interpolation process and the adaptive power estimation without using any phase information. Speech distortion becomes notable, especially, in a low frequency range where the energy of the speech is concentrated.

To overcome this degradation, a combination of PANS and the conventional adaptive noise canceller (ANC) is also considered. In a low frequency range, the ANC filter is applied to produce an accurate power estimate of the directional interference. Then, the spectral modification uses the noise estimate in order to enhance the speech without distortion. In a high frequency range, however, the PANS still applies the spectral modification with the auditory SE.

### *12.2.3 Combination of the PANS with Adaptive Noise Canceller (ANC)*

The structure of the PANS based on TFGSC is shown in Fig. 12.4. At low frequency range, noise is estimated by a conventional ANC filter. At high frequency range, the noise is estimated by the PANS filter. Enhanced speech is obtained by spectral modification. Since the energy of voiced speech is concentrated in low frequency range, in order to prevent the speech distortion, the spectral power of directional interference is estimated for each frequency bin rather than using filterbank. The PANS is applied to high frequency range so that the perceptual quality of speech is preserved and it also saves the computational load compared to the conventional ANC approach as well.

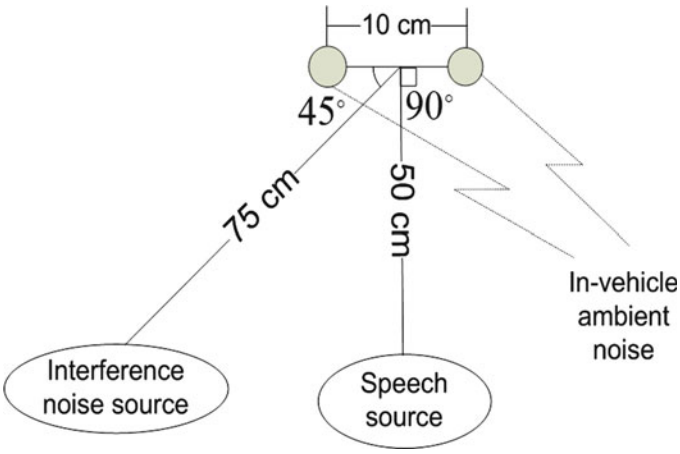



**Fig. 12.5** Microphone array aperture and location of signal sources for the RIR measurement

## 12.3 Experiments

To generate dual-channel speech signal and nonstationary interference signals, room impulse responses (RIRs) were measured in a vehicular chamber which has a reverberation time, $T_{60} = 250$ ms. The desired speech source was modeled to be located 50 cm from the microphone array along the broadside direction (90°) and the nonstationary interference source to be 75 cm along the 45° line. The array was located in front of the speech source with a 10-cm aperture. Figure 12.5 describes the experimental setup for signal generation.

Each RIR was convoluted with a single-channel clean speech signal to produce a dual-channel speech signal, and with an interfering human voice for a dual-channel nonstationary interference noise at a sampling rate of 8 kHz. Brownian noise was added as the in-vehicle ambient noise. Next, the interference plus the ambient noise was combined with the speech signal to simulate various signals with interference and noise ratios (SINR) ranging from −5 to 20 dB. The speech signal in experiments was formed from Korean digit strings and a nonstationary interference noise generated by using arbitrary Korean words.

To evaluate the performance of the noise suppression and the perceptibility of the enhanced speech signal, the noise reduction (NR) in log-domain and the perceptual evaluation of the speech (PESQ) are used as measures [9], respectively. Table 12.1 shows the performance of the proposed dual-channel speech enhancement system, where "PANS" and "PANS+ANC" denote the usage of PANS only and PANS with the ANC to estimate the desired spectral envelopes, respectively. The findings of the proposed algorithms is compared with the conventional transfer function–based GSC (TFGSC) method [2].

As shown in Table 12.1, the proposed PANS and PANS with ANC show superior performance over that of the TFGSC. Although PANS shows similar speech quality with TFGSC in adverse noise environment, this problem is solved by combining it with an ANC.

**Table 12.1** Experimental results of proposed algorithm

| Input SINR (dB) | | −5 | 0 | 5 | 10 | Avg |
|---|---|---|---|---|---|---|
| NR (dB) | TFGSC | 13.74 | 13.74 | 13.73 | 13.70 | 13.73 |
| | PANS | 22.45 | 22.43 | 22.23 | 21.72 | 22.21 |
| | ANC+PANS | 22.00 | 21.94 | 21.79 | 21.43 | 21.79 |
| PESQ | TFGSC | 2.02 | 2.44 | 2.64 | 3.02 | 2.53 |
| | PANS | 2.05 | 2.51 | 3.04 | 3.45 | 2.76 |
| | ANC+PANS | 2.76 | 3.12 | 3.41 | 3.62 | 3.23 |

## 12.4 Conclusions

We have proposed a dual-channel speech enhancement method using perceptual adaptive noise suppressor, which has improved the perceptual quality of speech for hands-free communication inside the auto chamber. The proposed method used an auditory filterbank based adaptive filter to estimate the noise SE and combined it with the ANC. This method resulted in reduced number of adaptive filter coefficients and has improved perceptual quality of speech. The usage of auditory SE was demonstrated to ensure robust noise suppression in the presence of in-vehicle ambient noise without additional postprocessing as demonstrated by the experimental results.

## References

1. Frost OL III (1972) An algorithm for linearly constrained adaptive array processing. Proc IEEE 60:926–935
2. Gannot S, Burshtein D, Weinstein E (2001) Signal enhancement using beamforming and nonstationarity with applications to speech. IEEE Trans Signal process 49(8):1614–1626
3. Brandstein M, Ward D (2001) Microphone arrays, Signal processing techniques and applications. New York, Springer
4. Meyer J, Simmer K U (1997) Multichannel speech enhancement in a car environment using Wiener filtering and spectral subtraction. In: Proceedings of ICASSP, IEEE Computer Society Washington DC, 1997, pp 1167–1170
5. Widrow B, Stearns S (1985) Adaptive signal processing. Prentice Hall, Englewood Cliffs, N.J.
6. Haykin S (2002) Adaptive filter theory, 4th edn. Prentice Hall, Upper Saddle River, N.J.
7. Faller C, Chen J (2005) Suppressing acoustic echo in a spectral envelope space. IEEE Trans Speech Audio Process 13(5):1048–1062
8. Wallin F, Faller C (2004) Perceptual quality of hybrid echo canceler/suppressor. ICASSP 4:157–160
9. Rix AW, Beerends JG, Hollier MP, Hekstra AP (2001) Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech coders. ITU-T Recommendation, pp 862, Feb 2001
10. Malcolm Slaney. Auditory toolbox, version 2. Technical Report #1998-010, Interval Research Corporation, 1998

# Chapter 13
# Optimal Multi-Microphone Speech Enhancement in Cars

**Lae-Hoon Kim* and Mark Hasegawa-Johnson**

**Abstract** Hands-free speech telephony and speech recognition in cars suffer from additive noise and reverberation. We propose an iterative blind room impulse response (RIR) estimation algorithm based on an analysis-by-synthesis loop closed around a multi-path generalized sidelobe canceller (GSC). By combining a post-filter with the proposed scheme, optimal speech enhancement in practical situations can be achieved. The algorithm is tested using simulated data and real speech recordings from the AVICAR database.

**Keywords** Hands-free communication • In-car speech recognition • Multi-path generalized sidelobe canceller (GSC) • Room impulse response (RIR)

## 13.1 Introduction

In recent years, although many systems have used multi-microphone arrays for speech enhancement [1, 2] and robust speech recognition [3], few approaches have presented a theoretical basis for multi-microphone speech signal processing under the assumed statistical model of source speech signal, room impulse response (RIR), and noise. One of the few published systems considering a theoretical basis for speech enhancement is that of Balan and Rosca [1], which showed that multi-microphone MMSE spectral amplitude estimation can be

*This work was done when Lae-Hoon Kim was with University of Illinois at Urbana-Champaign. He has since joined Qualcomm Inc.

L.-H. Kim (✉)
Qualcomm Incorporated, San Diego, CA, USA
e-mail: laehoonkim@gmail.com

M. Hasegawa-Johnson
University of Illinois, Urbana-Champaign, USA
e-mail: jhasegaw@uiuc.edu; jhasegaw@gmail.com

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_13, 

factored into a sufficient statistics followed by a single-microphone post-filter. As a straightforward extension of [1], if we know the RIRs, optimal estimation of the speech signal can be achieved using the simple two-step method. However, it is actually not easy to satisfy the assumption of the known RIRs. In this chapter, we address a realistic implementation of the sufficient statistics with unknown RIRs.

If we know the source signal, we can adaptively estimate the RIRs based on an acoustic echo cancelation scheme [4]. Because more correctly beamformed output is nearer to the original source signal, we might be able to use the beamformed output as a reference signal to estimate the RIRs [5]. In this chapter we propose using a delay-and-sum beamformer (DSB) to provide the information necessary for an initial constrained estimate of the RIR, which is then updated iteratively using a multi-path generalized sidelobe canceller (GSC) based on the evolving RIR estimate. Good RIR estimation makes the multi-path GSC more accurate, and this again guarantees better RIR estimation. We demonstrate that, with a reasonable constraint on the sparsity of the room impulse response, the algorithm converges to a useful approximate RIR. Even though we may not get perfect RIR identification, the converged RIR is nevertheless sufficient to compute coefficient vectors for a multi-path fixed beamformer (FBF) which outperforms the naive DSB. By leveraging the converged RIR, we are able to mitigate the common practical problem of multi-path GSC, namely, its tendency to cancel the target signal due the indistinguishability of signal from reverberation at the beamformer.

To visualize the situation in a tractable way, we first show the convergence of a simplified version of the proposed scheme. A simple simulation test shows that this method achieves sufficient blind deconvolution at the output of FBF. We then evaluate the proposed algorithm using real-world moving-car recordings [6].

## 13.2 Proposed Method

### *13.2.1 Multi-path GSC*

Multi-path GSC can be formulated as an optimization problem as shown in (13.1), which is a generalized version of GSC [7] under a known multi-path environment, represented by the RIR as coded into a constraint matrix C:

$$\underset{\vec{w}}{\operatorname{argmin}} E\left\{\vec{w}^T\vec{y}(n)\vec{y}(n)^T\vec{w}\right\} \text{ subject to } C^T\vec{w}=\vec{f}, \tag{13.1}$$

where $\hat{s}(n)=\vec{w}^T\vec{y}(n)$ is an estimated source signal at the current time $n$, $\vec{f}=[1\ \ 0\ \ \cdots\ \ 0]^T$. $\vec{y}(n)$ is a noisy signal vector measured by the microphone array, the array of filter coefficients is $\vec{w}=[w_1\ \ w_2\ \ \cdots\ \ w_{NL}]^T$ encoding the estimated L-tap inverse RIR filters for all of the N recorded signals, and

$$\vec{y}(n) = \left[ y_1^{[1:L]}(n) \quad y_2^{[1:L]}(n) \quad \cdots \quad y_N^{[1:L]}(n) \right], \tag{13.2}$$

$$y_i^{[1:L]}(n) = [y_i(n-(i-1)n_0) \;\; y_i(n-(i-1)n_0-1) \;\; \cdots \;\; y_i(n-(i-1)n_0)-L+1], \tag{13.3}$$

where i = 1, 2, ... N, steered to a look direction of $\theta = \arcsin(-n_0 \frac{c}{F_s d})$ for uniform microphone spacing d, sampling rate Fs, and speed of sound c. Note that $n_0$ is introduced in (5) to compensate the inter-microphone channel delay so that the signal from all microphone channels can be aligned. However, n0 may not be an integer number; therefore we may need to deal with non-integer delay compensation [3].

To derive multi-path GSC, we need to manipulate the constraint part in (1). The constraint part has the following convolution form:

$$C^T \vec{w} = [C_{h_1} \quad C_{h_2} \quad \cdots \quad C_{h_N}] \cdot \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{NL} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{13.4}$$

where $l_h$ (length of the RIR) + L-1 by L matrix $C_{h_i}$ is constructed from the response $\vec{h_i} = [h_i(0) \;\; h_i(1) \;\; \cdots \;\; h_i(l_h-1)]$,

$$C_{h_i} = \begin{bmatrix} h_i(0) & 0 & \cdots & 0 \\ h_i(1) & h_i(0) & \ddots & \vdots \\ \vdots & h_i(1) & \ddots & 0 \\ h_i(l_h-1) & \vdots & \ddots & h_i(0) \\ 0 & h_i(l_h-1) & \vdots & h_i(1) \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & h_i(l_h-1) \end{bmatrix}, \tag{13.5}$$

i = 1, 2, ..., N. $C_{h_i}$ is a typical linear convolution matrix, which has Toeplitz structure. The solution to Eq.13.5 is a channel deconvolution filter, $\vec{w}$ [8]. In standard multi-path GSC, the solution to (1) is computed by projection of $\vec{w}$ onto a blocking matrix, which can be constructed as the null space of the multichannel convolution matrix, C. Now, the problem of identifying the FBF coefficient vector $\vec{w}$ can be regarded as a general multichannel deconvolution problem; therefore it need not be computed directly as the least-squares solution of Eq.13.1; instead, if desired, we can apply any kind of multichannel deconvolution scheme [8–10]. The blocking matrix can also be constructed by using an echo cancelation scheme as in [5], because ideally the output of the fixed beamformer is

the deconvolved and beamformed source signal. Although we might be able to apply any kind of multichannel deconvolution scheme for FBF, in the subsequent sections we propose a blind multichannel RIR identification algorithm, which is in fact based on the unique structure of the multi-path GSC.

## *13.2.2 Iterative Blind Estimation of RIR Based on Multi-path GSC*

### 13.2.2.1 Problem Formulation

The channel response estimation follows the optimization process below:

$$\hat{h}_i(n) = \underset{\hat{h}_i(n)}{\arg\min} \left\| s(n) * (h_1(n) * w_1(n) + \cdots + h_N(n) * w_N(n)) * \hat{h}_i(n) - s(n) * h_i(n) \right\|^2, \tag{13.6}$$

where * stands for a convolution, and (7) can be represented in the following with vector notation $\vec{h}_i$:

$$\hat{\vec{h}}_i = \underset{\hat{\vec{h}}_i}{\arg\min} \left\| \hat{C}\hat{\vec{h}}_i - \vec{h}_i \right\|^2 = \left( \hat{C}^T \hat{C} \right)^{-1} \hat{C}^T \vec{h}_i, \tag{13.7}$$

where $\hat{C} = C^T \vec{w}$ is the convolution matrix obtained with the beamformed outputs of RIRs. Ideally, if $\hat{C} = I$, in other words, if the FBF of RIRs produces perfectly deconvolved output, then we can obtain the real RIRs. In addition to (13.7), the estimated RIRs are obtained with the constraint of forcing the magnitude to be zero values except at the estimated time stamps of each dominant reflection in RIRs. This constraint can be interpreted as a sparseness constraint of RIRs.

### 13.2.2.2 Algorithm

The proposed algorithm is introduced below step by step based on the assumption that we know the time stamps $r_{i,d}$ of the dominant echo paths occurring in impulse response $\vec{h}_i$, where i $= 1, \ldots,$ N and the number of dominant echo paths d $= 1, 2, \ldots,$ D. Here, we focus on the RIR estimation and deconvolution, because the noise suppression after the deconvolution is straightforward. Estimation of the time stamps for the dominant echo paths will be discussed in Sect.13.2.2.3:

1. Initialize estimated impulse response.
2. $h_i(n) = 1 + \varepsilon\delta(n - r_{i,1}) + \cdots + \varepsilon\delta(n - r_{i,D})$.
3. Perform multi-path GSC using (13.6) and update $h_i(r_{i,d})$ with the solution of (13.7). Enforce $h_i(n) = 0$ for other n.

4. Iterate 2 until there is no more significant change in the magnitude of the reflection.

If you follow the first iteration, you will get the first update at the time stamp for the dominant echo paths,

$$\hat{h}_i(r_{i,d}) \approx h_i(r_{i,d}) - \frac{1}{N}\left(h_1(r_{i,d}) + \cdots + h_i(r_{i,d}) - \varepsilon + \cdots + h_N(r_{i,d})\right) \tag{13.8}$$

(13.8) can be illustrated by the situation in which there is only one dominant reflection, with magnitude $\varepsilon$. Then, the deconvolution filter coefficient for the channel at time $r_{i,1}$ becomes near to $-\varepsilon$, if the deconvolution filter is long enough, to meet the following condition:

$$\left(\left(1 + \varepsilon\delta(t - r_{i,1})\right) * \left(1 - \varepsilon\delta(t - r_{i,1})\right)\right)(r) = 0. \tag{13.9}$$

Under this circumstance the deconvolution output at $r_{i,1}$ with RIR as input becomes $h_i(r_{i,1}) - \varepsilon$, and the beamformed output with this deconvolution output becomes $\frac{1}{N}\left(h_1(r_{i,1}) + \cdots + h_i(r_{i,1}) - \varepsilon + \cdots + h_N(r_{i,1})\right)$. Note that $h_i(n) = 0$ for every n except n = 0 and n=$r_{i,1}$. Now, by applying (13.7) for the channel estimation at channel i, (13.8) is obtained and $\hat{h}_i(r_{i,1})$ can be considered as an updated $\varepsilon$; therefore

$$\varepsilon_{k+1} = h_i(r_{i,1}) - \frac{1}{N}\left(h_1(r_{i,1}) + \cdots + h_i(r_{i,1}) - \varepsilon_k + \cdots + h_N(r_{i,1})\right) \tag{13.10}$$

at the kth iteration, which can be expressed as follows:

$$\varepsilon_{k+1} - h_i(r_{i,1}) + \frac{1}{N-1}\left(h_1(r_{i,1}) + \cdots + h_{i-1}(r_{i,1}) + h_{i+1}(r_{i,1}) \cdots + h_N(r_{i,1})\right)$$
$$\left(\frac{1}{N}\right)^n \left(\varepsilon_k - h_i(r_{i,1}) + \frac{1}{N-1}\left(h_1(r_{i,1}) + \cdots + h_{i-1}(r_{i,1}) + h_{i+1}(r_{i,1}) \cdots + h_N(r_{i,1})\right)\right) \tag{13.11}$$

By induction,

$$\varepsilon_\infty = \hat{h}_i(r_{i,1}) = h_i(r_{i,1}) - \frac{1}{N-1}\left(h_1(r_{i,1}) + \cdots + h_{i-1}(r_{i,1}) + h_{i+1}(r_{i,1}) \cdots h_N(r_{i,1})\right) \tag{13.12}$$

which can be interpreted as follows: If $\hat{h}_i(r_{i,1})$ is bigger than $\varepsilon$, it will be updated until there is no change of $\hat{h}_i(r_{i,1})$. In the early part of the RIR, echo paths are infrequent, typically $h_1(r_{i,1}), \cdots, h_{i-1}(r_{i,1}), h_{i+1}(r_{i,1}), \cdots, h_N(r_{i,1}) \ll h_i(r_{i,1})$; therefore $\hat{h}_i(r_{i,1}) \approx h_i(r_{i,1})$ in (13.12). Even with background noise in a real situation, $\hat{h}_i(r_{i,1}) \approx h_i(r_{i,1})$ still holds, since we can easily assume that the noise

process is zero mean and we take a mean of iteration measurements in (13.12). This one dominant reflection scenario can be extended to the case of multiple reflections, because after deconvolving the most dominant reflection path, the next dominant path can be deconvolved. Note that by specifying all the time stamps for the dominant echo paths, this sequential deconvolution can be performed implicitly. However, also note that due to practical issues such as low-pass filtering (due to sampling of the signal, and/or frequency-dependent reflection coefficients at the walls of the room), the response may not contain perfect impulses.

Such imperfection may produce some errors in estimation of the channel, since the assumption of sparse RIRs will no longer hold with precision. In particular, echo paths with similar direction of arrival (DOA) may not be estimated exactly using this scheme. In most cases, the restriction of channel estimation to sources with DOA different from those of the dominant echoes is not as difficult to meet as the restrictions imposed by other channel estimation algorithms, and in practice, even the imperfections of overlapping and negative-valued echoes seem not to harm channel estimation results using the proposed algorithm.

Figure 13.1 shows the converged result of a two-channel measurement with a seven dominant reflection paths in RIRs, including one negative component and one overlapped component as follows:

$$\begin{aligned} h_1 &= [1 \;\; 0 \;\; 0 \;\; 0 \;\; 0.5 \;\; 0 \;\; 0 \;\; 0.4 \;\; 0 \;\; 0.05 \;\; 0.3 \;\; 0 \;\; 0 \;\; -0.1 \;\; 0.09 \;\; 0 \;\; 0 \;\; 0.04]^T \\ h_2 &= [1 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0.5 \;\; 0 \;\; 0.45 \;\; 0 \;\; 0 \;\; 0.3 \;\; -0.1 \;\; 0 \;\; 0 \;\; 0.09 \;\; 0.04 \;\; 0]^T \end{aligned} \tag{13.13}$$

The first three reflection time stamps are assumed to be known and the others are set as zero. We can confirm that with the correct time stamps for a few of the dominant early echo paths (not all), we can estimate the channel responses and perform deconvolution.

#### 13.2.2.3 Algorithm with Reflection Time Stamp Estimation

In this section, we propose a heuristic method for estimating dominant reflection time stamps. The algorithm is as follows:

1. Initially we choose DSB as a first FBF and perform normalized least mean square algorithm to estimate the RIR FIR coefficients using the output of DSB.
2. Select the time stamps, in which the estimated RIR magnitudes are above a predefined threshold, which determines the significance level of the reflection.
3. Perform the proposed algorithm presented in Sect. 13.2.2.2.
4. Iterate 2 and 3 until there is no significant increase on the selected time stamps.

Figure 13.2 shows the converged result, where the simulated output of two channels has been obtained by convolving the channel response with a white Gaussian-noise source and the threshold has been set to 0.08. Note that most of the significant reflection points above the threshold can be estimated almost correctly.

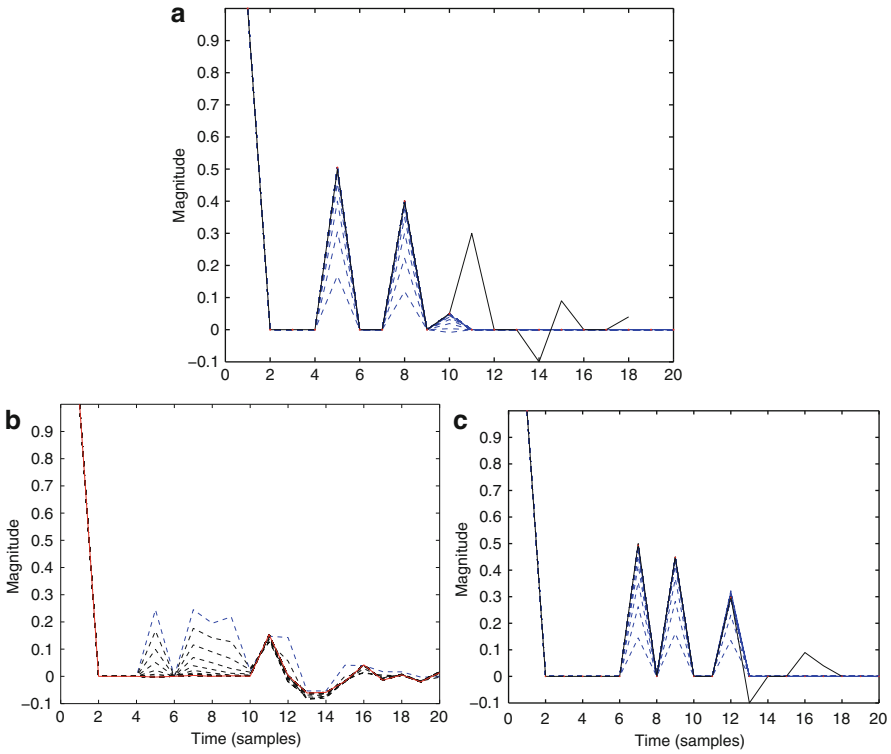



**Fig. 13.1** (**a**) FBF output: Blue dotted line is DSB output, black dotted lines are updated FBF output: Red line is the final FBF output after 20 iteration. Updated FBF output produces more impulse-like output by eliminating the effect of the designated echo paths, in other words, more deconvolved output. (**b**) Estimated channel h1. (**c**) Estimated channel h2: Red dots show the converged channel response after 20 iteration, and the blue dotted lines are updated responses. The black line is for original RIR. The designated channel responses are almost perfectly identified

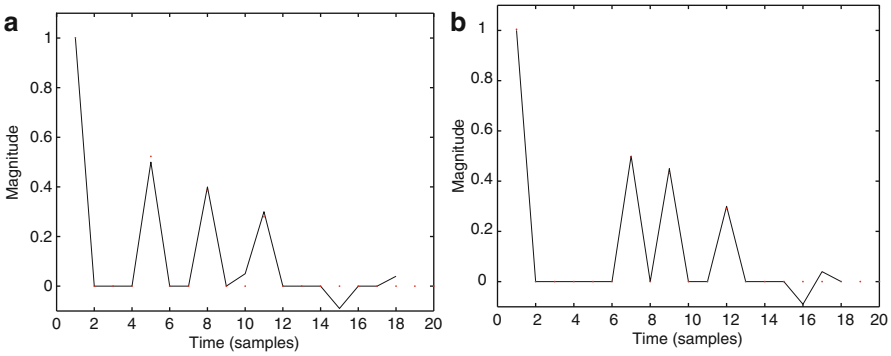



**Fig. 13.2** (**a**) Estimated channel h1. (**b**) Estimated channel h2: Red dots show the converged channel response after 20 iteration, and the black line is for the original RIR. The designated channel responses above the predefined threshold are almost correctly identified

## 13.3 Experiment with Real Car Data

In this section, we test the proposed algorithm using real multichannel sources measured in cars. Before running the algorithm, inter-channel delays are estimated using GCC-PHAT [11] to formulate the DSB. Figure 13.3 shows the two-channel

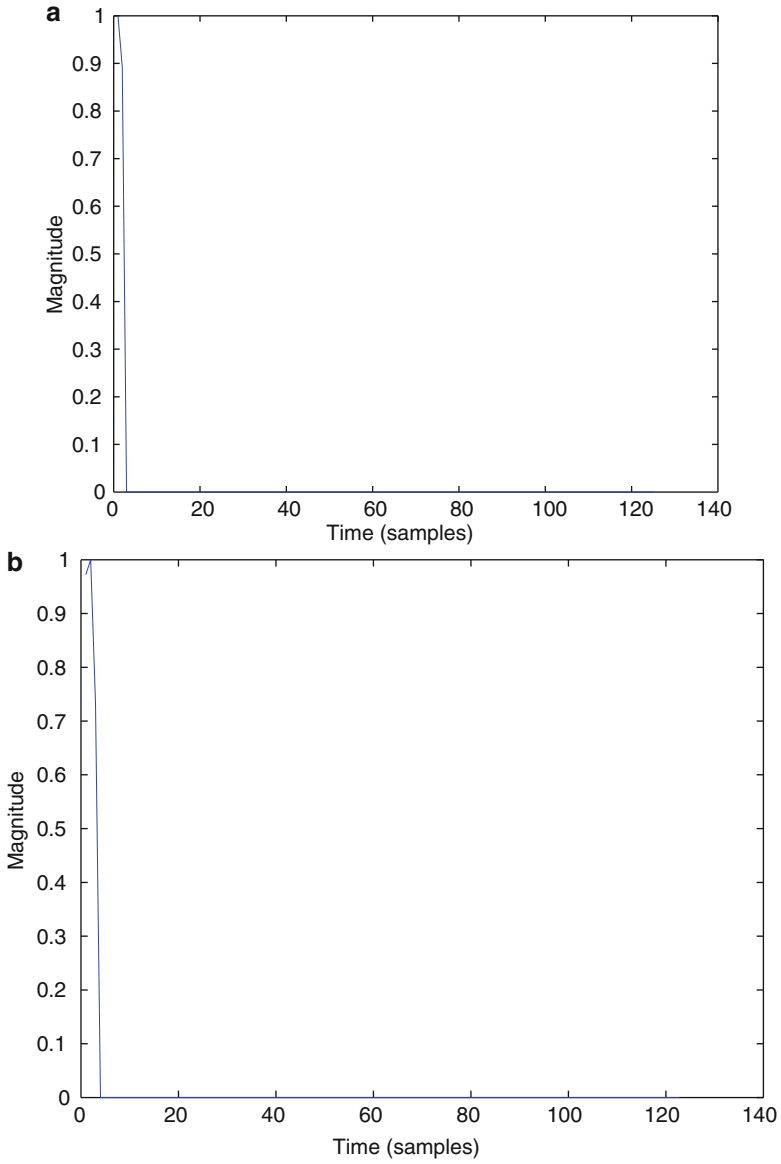



**Fig. 13.3** (**a**) Estimated channel hl. (**b**) Estimated channel h2

identification results using one of the single-digit utterances from the AVICAR database [6], and no distinctive reflection other than direct path has been estimated. Possible explanation about the result is that the space inside of a car is too small to have sparsely separable, distinctive echo paths. However, because this result also means that there is no significantly correlated reflection in the original signals with the beamformed output using the direct path information as in DSB, we can avoid the signal canceling problem when we use conventional GSC structure only with DSB as FBF. Optimal signal enhancement and isolated digit recognition results with conventional GSC followed by MMSE spectral amplitude estimation have been already reported in [12].

## 13.4 Conclusion

In this chapter, we propose a multi-path GSC–based blind channel identification method, which can be plugged in as a realistic replacement of the sufficient statistic for optimal speech enhancement. The simulation with artificially generated sparse channels demonstrates that the proposed algorithm can converge to good estimates of all components in the original channel responses that are above a predefined threshold. Channel estimation experiments with real data measured in a car show that there exists no distinctive significant reflection and support that a conventional GSC followed by a post-filter can produce optimal speech estimation.

## References

1. Balan R, Rosca J (2002) Microphone array speech enhancement by Bayesian estimation of spectral amplitude and phase. In: Proceedings of sensor array and multichannel signal process workshop, 2002 209–213
2. Gannot S, Cohen I (2004) Speech enhancement based on the general transfer function GSC and postfiltering. IEEE Trans Speech Audio Process 12:561–571
3. Oppenheim AV, Schafer RW (1999) Discrete-time signal processing, 2nd edn. Prentice Hall, Upper Saddle River
4. Gay SL, Benesty J (2000) Acoustic signal processing for telecommunication. Kluwer Academic Publishers, Norwell
5. Hoshuyama O, Sugiyama A, Hirano A (1999) A robust adaptive beamformer for microphone arrays with a blocking matrix using constrained adaptive filters. IEEE Trans Signal Process 47:2677–2684
6. Lee B, Hasegawa-Johnson M, Goudeseune C, Kamdar S, Borys S, Liu M, Huang T (2004) AVICAR: an audiovisual speech corpus in a car environment. In: Proceedings of international conference on spoken language processing, 2004
7. Griffiths LJ, Jim CW (1982) An alternative approach to linearly constrained adaptive beamforming. IEEE Trans Antenn Propag 30:27–34
8. Miyoshi M, Kaneda U (1988) Inverse filtering of room acoustics. IEEE Trans Acoustics Speech Signal Process 36:145–152

9. Delcroix M, Hikichi T, Miyoshi M (2007) Precise dereverberation using multichannel linear prediction. IEEE Trans Audio Speech Lang Process 15:430–440
10. Huang Y, Benesty J, Chen J (2005) A blind channel identification-based two-stage approach to separation and dereverberation of speech signals in a reverberant environment. IEEE Trans Speech Audio Process 13:882–895
11. Knapp GH, Carter GC (1976) The generalized correlation method for estimation of time delay. IEEE Trans Acoustics Speech Signal Process 24:320–327
12. Kim LH, Hasegawa-Johnson M, Sung KM (2006) Generalized optimal multi-microphone speech enhancement using sequential minimum variance distortionless response (MVDR) beamforming and postfiltering. In: Proceedings of international conference on acoustics, speech, and signal processing

# Part C
# Vehicle Dynamics, Vision, Active Safety, and Corpora

# Chapter 14
# Generating Reference Views of Traffic Intersection for Safe Driving Assistance

**Jien Kato and Yu Wang**

**Abstract** In this chapter, we address the problem of driving assistance along traffic intersections by providing drivers with additional visual information to expand their visual field. Our goal is to generate image stream of a virtual viewpoint which follows the host vehicle from a higher position, using images from multiple roadside cameras. Our approach is based on view morphing, but we extend it by integrating robust fundamental matrix estimation and sparse key point matching. This enables some tasks which previously rely on manual operation to be done automatically.

**Keywords** Driver visual field • Driving assistance • Image-based rendering (IBR) • Intersection assistance • Reference view • Vehicle blind spots

## 14.1 Introduction

Driving is becoming more and more stressful due to increasing traffic density. The situation seems to be more serious at intersections. According to the Annual Report 2007 from the National Police Agency of Japan, 46.3% of all traffic accidents in Japan occurred near intersections. In addition, a very large percentage of them happened either because of blind spots in the vehicles or inter-object occlusion due to traffic density at the intersections. These issues limit a driver's visual field. As a result, drivers find it more challenging to monitor their surroundings and forthcoming situations.

In the context of intersection assistance, Benmimnoun et al. presented a system [1] that utilizes intervehicle communication which updates the position measurements received from the onboard GPS and transmits all warning information to vehicles via

J. Kato (✉) • Y. Wang
Nagoya University, Nagoya 464-8603, Japan
e-mail: jien@is.nagoya-u.ac.jp; ywang@mv.ss.is.nagoya-u.ac.jp

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_14, 

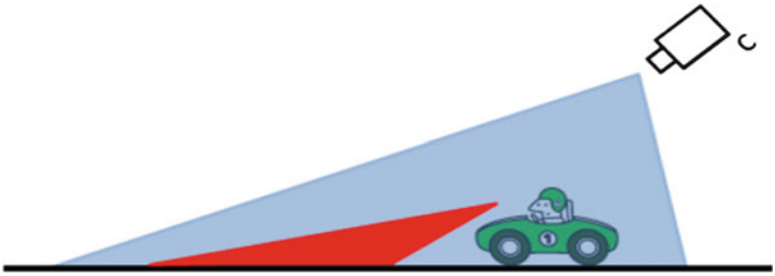


**Fig. 14.1** Reference view

roadside-vehicle communication. With the use of a well-designed human–machine interface, their system could improve traffic safety to some extent by providing resulted warning signals to drivers. However, the final information which the driver receives is that of danger warning. Such information is helpful, but it is not easily accessible compared to what drivers directly obtain using their eyesight. Also, it is difficult and awkward to handle at times. In another work that pertains to image processing, Ichihara et al. extended their NaviView [2] to suit the environment at the intersections. But a simple affine transformation can only provide drivers with mirror image of views from a roadside camera. Though that system could extend the driver's visual field to the next intersection, it is still power-limited, and the information obtained is difficult to handle.

We believe that visual information is intuitive. It enhances the driver's ability in handling the surrounding situation. Note that a driver's visual field is limited by the vehicle's structure and inter-object occlusion. Broadening it will make it more efficient. With this in mind, we propose a method for generating a reference view (Fig. 14.1) which follows the vehicle's movement from a higher ground. The resulted view not only extends the driver's visual field but also provides information about the vehicle itself. This leads to the strengthening of robustness against forthcoming occlusions. Since this viewpoint is aligned with the vehicle's direction, then it has a direct relation with what the driver could see. Also, it is natural for the driver to handle such view as reference information.

To generate such kind of view, we expect to use roadside cameras located at the intersections. Nowadays, roadside cameras have been installed in places where traffic accidents occur frequently, especially at intersections. Using image data from those cameras will be cost-effective.

We choose image-based rendering (IBR) method to achieve our goal because it could provide a realistic novel view. Since the shapes in novel view have to be preserved, the IBR method based on implicit geometry, such as view morphing [3], needs to be adopted. The accuracy of these methods has increased in the last decades. But they have not been widely used in real applications due to their excessive dependence on manual operations and need for prior knowledge of scene geometry. In this work, we extend and apply view morphing in a real application by integrating robust fundamental matrix estimation and feature matching. Our method only requires a slight adjustment of existing camera settings to make it amenable for practical use.

## 14.2 Approach

We assume that plural cameras have already been set around the given intersection. Obviously, the more cameras there are, the better reference view could be generated. In our work, evaluation was done by positioning six cameras at uniform heights. The detailed arrangement is shown in Fig. 14.2 (left). Our method does not restrict the detail position of cameras technically. Such a symmetrical setting is used only for an easy explanation. Each camera and its clockwise neighboring camera form a pair which are denoted as $C_{n0}$ and $C_{n1}$. Here, $n$ is the number of the pair. Like most actual situations, cameras are not calibrated in advance.

Our onboard system is supposed to receive image streams that are generated by each roadside camera while approaching the intersection. The camera pair which produces an orientation closest to the host vehicle is selected. The two images are then prewarped to make their image planes become parallel without changing the optical center of the cameras. Afterwards, we produce a novel view by linearly interpolating the positions and color of the two prewarped images. The resulting image is parallel with the prewarped two images, and it is shape-preserved. The position of the perspective view is determined by the angle between the vehicle's direction and directions of two selected cameras. Then, the images are again warped to align with the host vehicle's direction. In this way, we generate a view for the virtual camera as $C_s$ shown in Fig. 14.2. After a zooming stage via driver interaction, the final output of the system is the approximate view following the host vehicle's motion.

View morphing [3] is the inspiration for our method here. It could generate image from any viewpoint by linking two original cameras together. Note that the original method requires prior knowledge of the camera's projection matrices and excessive reliance on manual operation. Our team broadened it by integrating robust fundamental matrix estimation and sparse key point matching. The following paragraph further describes this method.

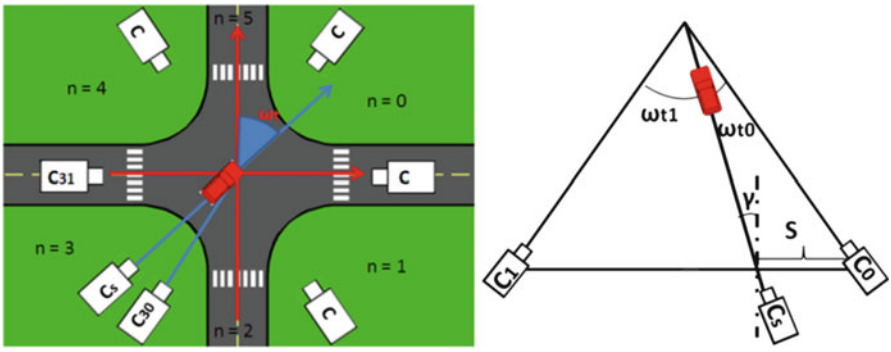



**Fig. 14.2** Actual and virtual cameras

## 14.3 Actual and Virtual Cameras

As mentioned previously, the detailed position of roadside cameras is not restricted. We made it a precondition because the existing roadside cameras in many intersections are not set for our purposes. The number may not be enough, and the settings may not suit our needs. Adding one or two cameras or adjusting the existing settings will make it convenient to use. At the same time, a robust way is needed to combine the images in a direct way. Moreover, to be able to align it with the host vehicle's direction, the virtual camera's position and direction should also be determined via the online measurement of the host vehicle's motion.

### *14.3.1 Estimate Fundamental Matrix*

For the camera pair n, its fundamental matrix $F_n$ is invariable and only needs to be computed once. The first step we perform is to take two images $I'_0$ and $I'_1$ from two cameras $C_{n0}$ and $C_{n1}$ and establish correspondence between them. Since $I'_0$ and $I'_1$ are from disparate viewpoints, such feature matching across a wide baseline is an error-prone task.

To achieve good estimation of the $F_n$, we first use SIFT key point detector [4] to select a set of key points from each image. We choose SIFT key point because it is robust against image transformation, and with a descriptor associate with each key point, we can easily establish potential correspondence with high confidence. Then, we match the key points between the image pair by finding the nearest neighbor of their descriptors in Euclidean distance. Since it may still contain many outliers of the matching, we adapt RANSAC [5] to estimate the $F_n$. During each RANSAC loop, eight corresponding pairs are randomly selected, and associated fundamental matrix is estimated using eight-point algorithm [6]. The quality of the estimated fundamental matrix in each loop is assessed by counting the number of inliers. A match is treated as inlier when there is reprojection error under a threshold. After many iterations of RANSAC for each camera pair, we get consistent results of $F_n$.

### *14.3.2 Virtual View Point*

Since our goal is to generate a view that dynamically follows the host vehicle, the virtual viewpoint's direction should be the same as that of the host vehicle. In this chapter, we assume that the online direction of the host vehicle is known as $\omega_t$ (Fig. 14.2 left), where $t$ is the time index. Based on it, we take the camera pair $C_{t0}$ and $C_{t1}$, which has the closest direction with $\omega_t$, and compute the corresponding angles $\omega_{t0}$ and $\omega_{t1}$. In order to produce the view of the virtual camera $C_s$, the $s$ and the camera tilt $\gamma$ (Fig. 14.2 right) are needed. The $s$ determines the morphing rate when producing the intermediate

parallel view by interpolation, while the $\gamma$ is needed when rotating the interpolated image to align the host vehicle's direction. We treat the position of $C_{t0} = 0$ and $C_{t1} = 1$; then, the s could be worked out approximately via $S = \omega_{t0}/(\omega_{t0} + \omega_{t1})$ while $\gamma = (\omega_{t1} - \omega_{t0})/2$.

## 14.4 Generating Reference View

In this section, we will introduce our method of generating the reference view. In each time step, one camera pair is selected, and the images $I_0$ and $I_1$ from $C_0$ and $C_1$ are used as source. Our method is an extension of view morphing [3]. We integrate a feature-matching procedure to avoid manual operations. Our method could be summarized as a four-step procedure as described in the following section.

### *14.4.1 Feature Correspondence*

In order to produce a morph, the complete correspondence maps between each pixel of two source images should be specified. Previously, the user manually determines correspondences by specifying a sparse set of matching features. The remaining correspondences are then ascertained based on these matches by interpolation [7]. Excessive reliance on manual operation makes the process ambiguous and not easy to manipulate [8].

In our work, it is also necessary to obtain the correspondence maps to synthesize a shape-preserved novel view. To ensure the quality of the novel view, a sufficient number of matches and ample distribution of the images should be guaranteed. In this situation, again comes the issue of establishing correspondence between images across a wide baseline. Differences in estimation of the fundamental matrix arise. Here, the quality of potential matches is more important. At the same time, there is additional need for quantity and distribution concerning such matches.

We apply SIFT detector [4] and Harris Corner Detector [9] on both $I_0$ and $I_1$, and collect responses from the image pair. We again use the descriptor of SIFT key points to establish potential correspondence as we have done in Sect. 14.3.1. For each corner key point, we use the normalized cross-correlation criterion to find its best match [6]. The reason we use two detectors is that they have different properties. With a local descriptor, SIFT key point is efficient in establishing correspondence with high confidence. Beside SIFT, using Harris corner key point could ensure that sufficient shape-related correspondence can be found. We then collect the matches generated in this manner. In order to eliminate false matches, we further use the precomputed fundamental matrix to remove outliers by enforcing Epipolar constraint. This way, we obtained a set of sufficient correspondence with high confidence. These correspondences will then be used in the following view synthesis procedure.

### 14.4.2 Prewarping

In order to produce a shape-preserved morph, the two images should be rotated twice to align the image planes and scan lines. Then, the linear interpolation on the warped image could produce new perspective views as the camera moves along the line linking two cameras together. Therefore, in each time step, we need to perform projective transformations $H_0$ and $H_1$ on both $I_0$ and $I_1$.

We denote $R_{\theta_i}^{d_i}$ and $R_{\phi_i}$ $(i = 0, 1)$ are both 3 by 3 matrix. $R_{\theta_i}^{d_i}$ is a rotation of angle $\theta_i$ about axis $d_i$ in depth, which makes the two image planes become parallel, while $R_{\phi_i}$ corresponds to an affine warping to align the scan lines. Given the fundamental matrix $F$ of $I_0$ and $I_1$, the four matrixes could be determined by choosing a rotation axis $d_0$.

The first thing we do is to factorize the precomputed $F$ with singular value decomposition and obtain two unit eigenvectors (epipoles) $\mathrm{e}_0 = \left[\mathrm{e}_0^{\mathrm{x}}, \mathrm{e}_0^{\mathrm{y}}, \mathrm{e}_0^{\mathrm{z}}\right]^{\mathrm{T}}$ and $e_1 = \left[e_1^x, e_1^y, e_1^z\right]^T$ of $F$ and $F^T$ respectively. We follow the recommended choice in [3] and select the rotation axis as$d_0 = [-e_0^y, e_0^x, 0]^T$. Then, we compute a vector $[x, y, z]^T = Fd_0$, and take $d_1 = [-y, x, 0]^T$. The angles of rotation in depth about $d_i$ could be computed via

$$\theta_i = -\frac{\pi}{2} - tan^{-1}\left(\frac{d_i^y e_i^x - d_i^x e_i^y}{e_i^z}\right). \tag{14.1}$$

In this way, the two rotations in depth are determined.

Following the depth rotation is another affine warp $R_{\phi_i}$ to make the Epipolar lines parallel. After the first rotation, the new epipoles become $\left[\tilde{e}_i^x, \tilde{e}_i^y, 0\right]^T = R_{\theta_i}^{d_i} e_i$. Then, the angles of rotation $\phi_0$ and $\phi_1$ could be obtained via

$$\phi_1 = -tan^{-1}\left(\tilde{e}_i^y / \tilde{e}_i^x\right). \tag{14.2}$$

After each image has been rotated twice, the original fundamental matrix is formed:

$$\tilde{F} = R_{\phi_1} R_{\theta_1}^{d_1} F_n R_{-\theta_0}^{d_0} R_{-\theta_0} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & a \\ 0 & b & c \end{bmatrix}. \tag{14.3}$$

To make sure $F$ is in the form:

$$\left(H_1^{-1}\right)^T F H_0^{-1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}. \tag{14.4}$$

Another translation is then applied on $I_1$ as

$$T = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -a & -c \\ 0 & 0 & b \end{bmatrix} \tag{14.5}$$

Now, two prewar transforms can be computed via $H_0 = R_{\phi_0} R_{\theta_0}^{d_0}$ and $H_1 = TR_{\phi_1} R_{\theta_1}^{d_1}$.

With the obtained $H_0$ and $H_1$, we perform the projective transformations on the two images $I_0$ and $I_1$, and obtain $\hat{I}_0$ and $\hat{I}_1$. In the previous step, we have obtained a set of feature matches. For the following interpolation step, we also perform the same projective transformation on their coordinates.

### 14.4.3 Image Interpolation

We have shown that in view morphing [3], the linear interpolation of parallel images is another parallel view. After the prewarping, both images $\hat{I}_0$ and $\hat{I}_1$ are capable for such kind of interpolation. In addition, during the transformation of a coordinate, the matching point's coordinates changed as well. The correspondence of the original images is preserved and represented as the new coordinate of the warped images. We then determine the maps of non-key points between two warped images using MATLAB 4 griddata method. This method could produce smooth surfaces for all pixels between $\hat{I}_0$ and $\hat{I}_1$ from a set of correspondence, namely two mapping function $T_0 : \hat{I}_0 \rightarrow \hat{I}_1$ and $T_1 : \hat{I}_1 \rightarrow \hat{I}_0$.

Using the morphing rate $s$, what we have estimated previously, and the two mapping functions in our hand, we then compute the displacement of each pixel $P_0 \in \hat{I}_0$ and $P_1 \in \hat{I}_1$via :

$$W_0(p_0, s) = (1 - s)p_0 + sT_0(p_0), \tag{14.6}$$

$$W_1(p_1, s) = (1 - s)T_1(p_1) + s(p_1). \tag{14.7}$$

Then, we integrate their colors by cross-dissolve procedure.

### 14.4.4 Postwarping and Zooming

After the interpolation, we have produced a novel view of the intersection. Since such view is parallel with the line linking two cameras together, we then perform a postwarp to make it align along the host vehicle's direction. The warping is a plane rotation of angle $\gamma$ in depth. After postwarping, the driver may need to zoom to finally approximate or obtain the reference view he/she will use during decision making while driving through an intersection.

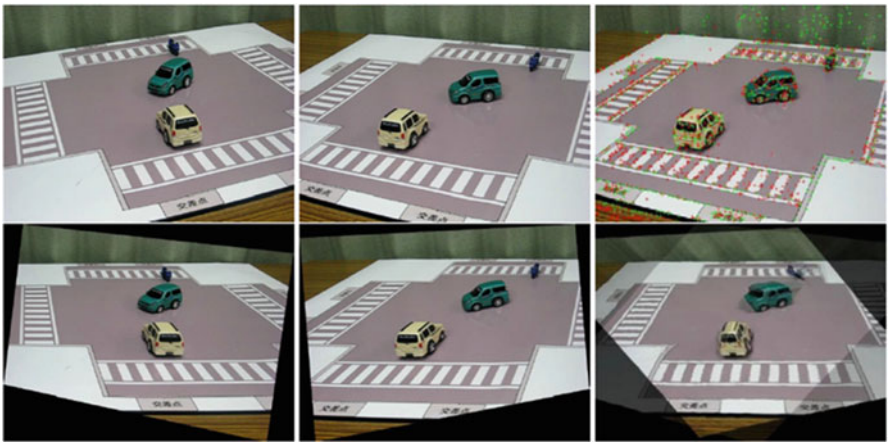


**Fig. 14.3** Experimental results

## 14.5 Experimental Result

Our evaluation experiment is done by using an intersection model at 1:38 scale. We use six cameras with resolution 640 by 480. The camera setting is approximately the one shown in Fig. 14.2 (left). Remote toy cars and bikes are used to obtain test image sequences. Figure 14.3 (top left and middle) shows a pair of our sample input from the left and right cameras respectively.

First of all, the fundamental matrices are estimated in the way mentioned in Sect.14.3.1. The prewarping transformations $H_0$ and $H_1$ are then calculated based on $F$. We take a pair of cameras' images as examples as shown in Fig. 14.3 (top left and middle). By jointly using SIFT and Harris detectors, about two thousand key points were selected in each image, and the distribution is normalized as shown in Fig. 14.3 (top right, green: SIFT, red: Harris). Using the matching criterion in Sect. 14.4.1 and followed with a manually operated refining step, two hundred and ten features were finally selected as correspondence. We then make the projective transformations on the two images (Fig. 14.3 bottom left and middle) as well as the matching points' coordinates. Without automatic estimation of vehicle's direction, we then produce a reference image by manually assigned morphing rate $s$ and the camera tilt angle $\gamma$. The resulting image is shown in Fig. 14.3 (bottom right). Even though the resulting image contains some ghost effect, it is evident that the proposed method works well.

## 14.6 Conclusion

In this chapter, we proposed a method to generate the reference view of traffic intersection for safe driving assistance. We adapted the view morphing approach and broadened it using robust fundamental matrix estimation and automatic feature

matching. This allows us to achieve the goal without any prior knowledge of scene geometry and excessive manual operation, which were crucial obstacles under the original model. Our experiment shows that our method works fine even for the images from large baseline disparate viewpoints.

During the processing, since the original images were resampled many times and occlusion exists, the resulted novel view contains some ghost effect. In order to solve these effects, we will optimize the raw output by introducing smoothness prior to the future work.

## References

1. Benmimoun A, Chen J, Suzuki T (2007) Design and practical evaluation of an intersection assistant in real world tests. In: Proceedings of IEEE intelligent vehicles symposium Istanbul, pp 606–611
2. Ichihara E, Takao H, Ohta Y (1999) NaviVIew: bird's-eye view for drivers using roadside cameras. The transactions of the IEICE J82-D-II(10):1816–1825
3. Seitz SM, Dyer CR (1996) View morphing. In: ACM Proceedings of SIGGRAPH 96, New York, pp 21–30
4. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int'l J Computer Vision 60(2):91–110
5. Fischler MA, Bolles C (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Comm of the ACM 24(6):381–395
6. Ma Y, Soatto S, Kosecka J, Sastry S (2003) An invitation to 3-D vision: from images to geometric models. Springer, New York
7. Beier T, Neely S (1992) Feature-based image metamorphosis. In: ACM Proceedings of SIGGRAPH 92, Chicago, USA, pp 35–42
8. Shum H, Kang SB (2000) A review of image-based rendering techniques. IEEE/SPIE Visual Commun Image Process, pp 2–13
9. Harris C, Stephens MJ (1988) A combined corner and edge detector. In: Proceedings of Alvey vision conference, Manchester, England, pp 147–152

# Chapter 15
# Computer Vision Systems for "Context-Aware" Active Vehicle Safety and Driver Assistance

**Pinar Boyraz*, Xuebo Yang, and John H.L. Hansen**

**Abstract** Recent developments of information technology and mobile lifestyle have forced drivers to multitask while they drive. The in-vehicle "infotainment" technology is already taking its place in the transformation of vehicles towards more intelligent and interactive devices rather than staying as mere transportation convenience. This transformation has several advantages such as easy route navigation, real-time traffic information, and staying connected with work or people while traveling. However, it has several drawbacks concerning the impact on driver cognitive load and attention sources. Therefore, it is crucial to take advantage of state-of-the-art in-vehicle technology to produce counter-measure systems that monitor the driver status and reduce driver workload adaptively depending on the context. In recognition and analysis of the driving context together with driver status monitoring, computer vision applications supply crucial information both in the vehicle (i.e., driver head and eye tracking) and out of the vehicle (i.e., lane, pedestrian, and vehicle detection and tracking, and road sign recognition). In this chapter, we provide a broad range of computer vision applications for CA-IVS from the literature and our previous studies, and we report our current research efforts.

* Pinar Boyraz was with the University of Texas at Dallas, CRSS UTDrive modeling group when this work was done. She has since joined Istanbul Technical University, Turkey.

P. Boyraz (✉)
Mechatronics Education and Research Center (MERC), Department of Mechanical Engineering, Istanbul Technical University, Inonu Cd. No.65, Gumussuyu, Istanbul 34437, Turkey

Center for Robust Speech Systems (CRSS), Department of Electrical Engineering
The University of Texas at Dallas, Richardson, TX 75080-3021, USA
e-mail: boyraz.pinar@googlemail.com

X. Yang • J.H.L. Hansen
Center for Robust Speech Systems (CRSS), Department of Electrical Engineering
The University of Texas at Dallas, Richardson, TX 75080-3021, USA
e-mail: john.hansen@utdallas.edu

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_15, 

**Keywords** Computer vision • Context aware • Lane tracking

## 15.1 Introduction

In their brief report, Fletcher et al. [1] provide an overall summary of promising computer vision systems applied in the vehicle. They determine areas where vision systems could be useful such as driver fatigue or inattention detection, pedestrian spotting, blind-spot checking, lane keeping, traffic sign recognition, and human factors aids. These applications are built based on several computer vision systems which are surveyed and/or presented here in this chapter as well. Building on what is already achieved in this area, we provide a systems engineering survey of computer vision systems for in-vehicle applications together with our previous and current findings. This study also presents a system utility analysis that ties all systems in a mechatronics integration approach, reducing complexity and cost of the final in-vehicle computer vision system, while maximizing the utility factor of the resultant design. In Sect. 15.2, applications are grouped into two main areas: driver status monitoring (inside the vehicle) and vehicle peripheral monitoring (outside the vehicle). These systems can be thought of as the *eyes of the cyber copilot* in the vehicle which (or who) is aware of driver's condition as well as the environment and the current situation (i.e., situation/context awareness). Next, in Sect. 15.3, all systems are analyzed from the perspective of utility in their *projected impact* on reducing the number of accidents or fatality rates. After determining the utility factors of systems, an example of mechatronics system integration for in-vehicle systems is presented. Finally, conclusions are drawn in Sect. 15.4 pointing to future research directions in this area.

## 15.2 Computer Vision Systems for In-vehicle Applications

In this section, we briefly survey different CV systems, reporting our progress in some areas with focus on the UTDrive research team. CV systems are seen as crucial components of future DAS and AVS systems; however, there is still a need for further development to achieve robust operation on board. Before providing details on each system, a list of requirements from onboard CV systems are presented here to emphasize the challenges in this area, some of which require hardware solutions and development of novel systems:

- Robust against illumination change
- Reliable in vibration and high accelerations
- Durable to low/high temperatures and weather conditions (especially cabling and mounting parts)
- Nonintrusive to the driver
- Compact/mobile
- Minimal power and computing source use

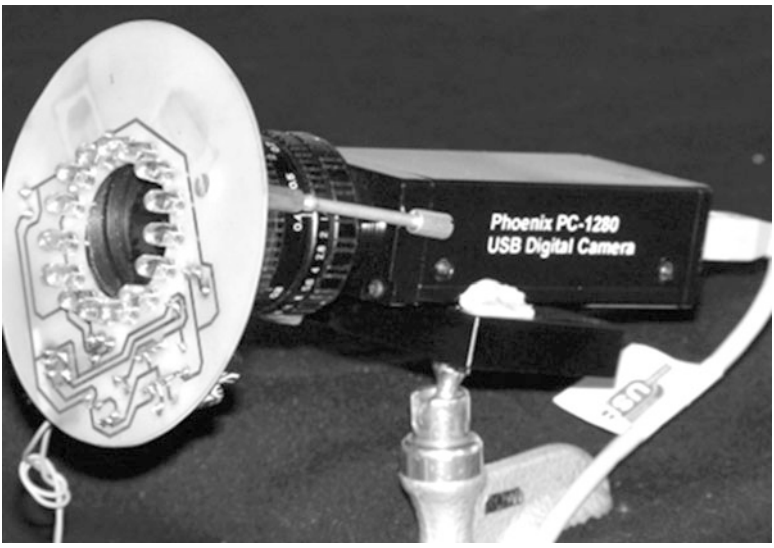



**Fig. 15.1** NIR eye tracker designed as a part of driving monitoring system

Here, the CV systems are grouped under two main groups focusing on driver and on the environment covering all aspects of the driving context.

### *15.2.1 Eye and Head Tracking*

Eye-tracking applications are originally motivated by research in human–computer interface development to create novel ways of interfacing [2] or helping people with motor disabilities [3]. In [4], an extensive survey of eye-tracking applications is given. For particular applications of eye tracking in driver monitoring systems, the systems can have a wide range from bright-pupil technique [5] utilizing co-centric near-infrared lights around the camera lens with active illumination to systems using an off-the-shelf webcam and visible light [6] and head-mounted systems [7]. There are also commercial eye-tracking applications already being used in studies utilizing eye-gaze information [8–10]. For head tracking, several applications using eye location, skin color, or motion in the image can be found [11,12]. A recent real-time system for monitoring driver vigilance is reported in [13]. In our previous study [14], an evolutionary computational approach was used to obtain an adaptive eye-tracking system to provide robustness in illumination changes. The system used bright-pupil technique which is based on retro-reflection property of the eye retina. The components of the system can be seen in Fig. 15.1, comprising a CMOS camera, co-centric ring of NIR LEDs to create bright-pupil effect, and optical absorption filters to block daylight.

Using the CV system in Fig. 15.1, eye tracking was performed to measure the pupil area as an indirect measurement of eyelid closure, eye gaze in x–y coordinate system, and head motion in 2-D image plane. The system can measure these three important indicators revealing drowsiness (i.e. PERCLOS [21] and [22]), attention level, as well as driver activity.

### *15.2.2 Affective Computing: Emotion Recognition*

Emotion recognition can be multimodal using speech and video/image. The emotion recognition task is very difficult to achieve. It has been reported that even human coders are able to recognize the universal six archetypal emotions with accuracy of between 40% and 60%, especially when they are given the cues in single modality (i.e., only audio or only visual) [15], visible light [6], and head-mounted systems [7].

Although a tremendous amount of work exists in the face recognition area, emotion recognition remains a challenge since it has a temporal dimension as well and deals with nonrigid motion of the face. It is also a very young area which needs substantial work to reach the maturity of face recognition. However, there have been efforts to develop real-time and automatic units for emotion recognition using video modality. Anderson et al. [16] designed a fully automated multistage system for real-time recognition of facial expression. First, the faces are located using a spatial ratio template tracker algorithm, and optical flow of the face is subsequently determined using a real-time implementation of a robust gradient. The head motion is dealt with averaging and was canceled. The motion signatures from optical flow algorithm were classified using an SVM into non-expressive or six basic emotion types, as most of the work action units (AUs) were used. Shan et al. [17] investigated new subspace methods for reducing the features for facial expression analysis. Pantic et al. [18] especially emphasized the temporal characteristic of emotion sequences and had a detailed analysis of motion sequences using the profile face videos. However, they commented that this area needs to have a possible multi-camera system to deal with different viewing angles of the face and dynamic head motion cancelation. A survey of state-of-the-art automatic facial expression analysis can be found in Pantic et al. [19].

### *15.2.3 Vehicle Peripheral Monitoring*

In this category, all road object detection and tracking systems can be included. Among them, the most promising systems are lane detection and tracking, road sign recognition, vehicle detection tracking, and finally pedestrian detection and tracking. Under the UTDrive research project, lane detection/tracking and road sign recognition systems are currently being developed with a context-aware framework.

### 15.2.4 Road Object Detection and Tracking

Video streams, whether processed online or off-line, contain rich information content regarding road scene. It is possible to detect and track vehicle, lane markings, and pedestrians and recognize road signs using a frontal camera and some additional sensors such as radar.

It is of crucial importance to be able to detect, recognize, and track road objects for effective collision avoidance or driver assistance system. In this chapter, we present our current progress in lane tracking and road sign recognition also reported in [22], adding a system utility analysis here.

### 15.2.5 Lane Detection and Tracking

There has been extensive work in developing lane tracking systems in the area of computer vision. These systems can be potentially utilized in driver assistance systems related to lane keeping and lane change. In [23], a comprehensive comparison of various lane-position detection and tracking techniques is presented. From that comparison, it is clearly seen that most lane tracking algorithms do not perform adequately so as to be employed in actual safety-related systems; however, there are encouraging advancements towards obtaining a robust lane tracker. A generic lane tracking algorithm has the following modules: a road model, feature extraction, post-processing (verification), and tracking. The road model can be implicitly incorporated as in [24] using features such as starting position, direction, and gray-level intensity. Model-based approaches are found to be more robust compared to feature-based methods. For example, in [25], a B-snake is used to represent the road. Tracking lanes in real traffic environment is an extremely difficult problem due to moving vehicles, unclear/degraded lane markings, and variation of lane marks, illumination changes, and weather conditions. In [26], a probabilistic framework with particle filtering was suggested to track the lane candidates selected from a group of lane hypotheses. A color-based scheme is used in [27]; shape and motion cues are employed to deal with moving vehicles in the traffic scene as well.

### 15.2.6 Road Sign Recognition

Methods used for automatic road sign recognition can be classified into three groups: color based, shape based, and others. The challenges in recognition of road signs

from real traffic scenes using a camera in a moving vehicle has been listed as lighting condition, blurring effect, sign distortion, occlusion by other objects, and sensor limitations. In [28], a nonlinear correlation scheme using filter banks is proposed to tolerate in/out of plane distortion, illumination variance, background noise, and partial occlusions. However, the method has not been tested on different signs in a moving vehicle. Broggi et al. has addressed real-time road signs recognition in three steps: color segmentation, shape detection, and classification via neural networks; however, vehicle motion problem is not explicitly addressed. Jilmenez et al. used FFT signatures of the road sign shapes and SVM-based classifier. The algorithm is claimed to be robust in adverse conditions such as scaling, rotations, and projection deformations and occlusions.

## 15.3 System Utility Analysis and Mechatronics Integration

In recent years, whisper speech processing has attracted several researches. In this section, a system utility analysis is performed projecting the effect and cost of the surveyed CV systems onto 2007 FARS accident causation data [31, 32]. First, a query is run on FARS database to obtain the number of fatalities as the column and several driver-related factors on the rows. This table is rearranged into a more compact form and shown in Appendix 15.1. In this table, categories of causation are grouped under three major groups: driver impairment, driver errors, and in-vehicle devices. Redefining of these major groups in seven categories and matching them with appropriate CV systems that have the potential of preventing the accidents resulted in a new table shown in Appendix 15.2. The refined categories are: driver impairment, poor decision making, reckless driving, poor lateral control, poor longitudinal control, poor maneuvering, and in-vehicle devices. The distribution of the database is shown in Fig. 15.2. From this figure, it can be seen that only 34% of the fatalities are caused by driver-related factors; however, 66% of the data is unclassified and not reported clearly. Therefore, we may say that 34% is an underestimated figure. Nevertheless, the distribution within this 34% of the fatalities in terms of causation gives us important information about which types of driver errors should be prevented and where the drivers require the most assistance. From the distribution of the causation of accidents, we can clearly see that poor lateral and longitudinal control and maneuvering accounts for up to 65%. This figure can be reduced by proper DAS, warning, or active safety systems. Using the figures from the refined table in Appendix 15.2, a simple utility analysis is performed, and the results are shown in Table 15.1.

From the analysis results in Table 15.1, the most beneficial systems are determined to be lane tracking, optical flow, and traffic sign recognition. If an integrated system is used and integrated using the same sensor with modulation according to the imminent situation, the most beneficial system is traffic scene analysis. In the light of this justification, we report our recent efforts in designing a traffic scene analysis system with initial components being a lane tracker and traffic sign recognizer. The system

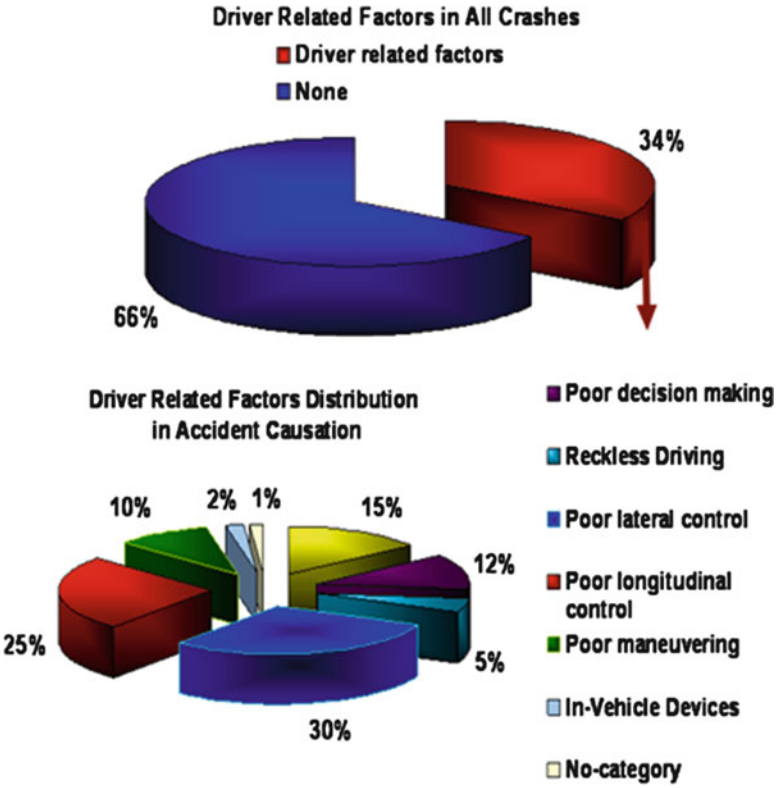



**Fig. 15.2** Driver-related factors in crashes and its distribution

**Table 15.1** Utility analysis results using projected prevention rate and unit cost of systems

| CV system | Name | Projected prevention | % | Cost | Utility |
|---|---|---|---|---|---|
| Eye and head tracking | EHT | 4,307 | 14.3 | 100 | 0.143 |
| Emotion recognition | ER | 1,683 | 5.6 | 100 | 0.056 |
| Lane tracking | LT | 10,304 | 34.3 | 100 | **0.343** |
| Optical flow | OF | 9,279 | 30.9 | 80 | **0.386** |
| Lane change recognition | LCR | 219 | 0.73 | 80 | 0.009 |
| Road area recognition | RAR | 66 | 0.22 | 50 | 0.004 |
| Vehicle detection and tracking | VDT | 1,585 | 5.28 | 100 | 0.053 |
| Pedestrian detection and tracking | PDT | 31 | 0.1 | 100 | 0.001 |
| Traffic sign recognition | TSR | 8,657 | 28.8 | 80 | **0.36** |
| *Integrated systems* | | | | | |
| Traffic scene analysis | TSA | 20,664 | 68.8 | 100 | **0.688** |
| Driver warning system | DW | 24,971 | 83.2 | 200 | **0.416** |

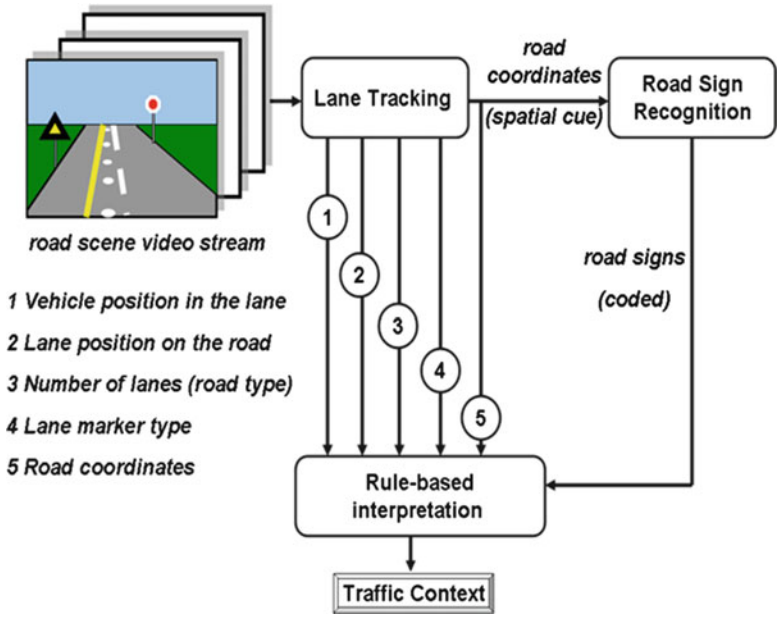



**Fig. 15.3** General framework for TSA system. The details of the versatile lane tracker algorithm are given in Fig. 15.4

is presented in detail in [22]. The general framework is depicted in Fig. 15.3 with the aim to extract overall traffic context. The details of the lane tracking algorithm is given in Fig. 15.4. Sample outputs from road sign recognition module is shown in Fig. 15.5.

The fusion of information between different image processing modules can be realized using a rule-based expert system as a first step. Here, we present a set of rules combining the outputs of vision algorithms with the output options of warning, information message, and activation of safety features.

*Case 1*: If road sign is 0, standard deviation of lane position $<$ 10 pixels, standard deviation of vehicle speed $<$ 10 km/h, context: normal cruise.

*Case 2*: If road sign is 0, standard deviation of lane position $<$ 10 pixels, standard deviation of vehicle speed $>$ 10 km/h, context: stop-go traffic, likely congestion, and output: send information to traffic control center.

*Case 3*: If road sign is 1, vehicle speed $>$ 20 km/h, context: speed limit is approaching, output: warning.

*Case 4*: If road sign is 2, vehicle speed $>$ 20 km/h, context: stop sign is approaching and the driver did not reduce the vehicle speed yet, output: warning and activation of speed control and brake assist.

*Case 5*: If road sign is 3, vehicle speed $>$ 20 km/h, context: pedestrian sign is approaching, output: warning and activation *of brake assist*.

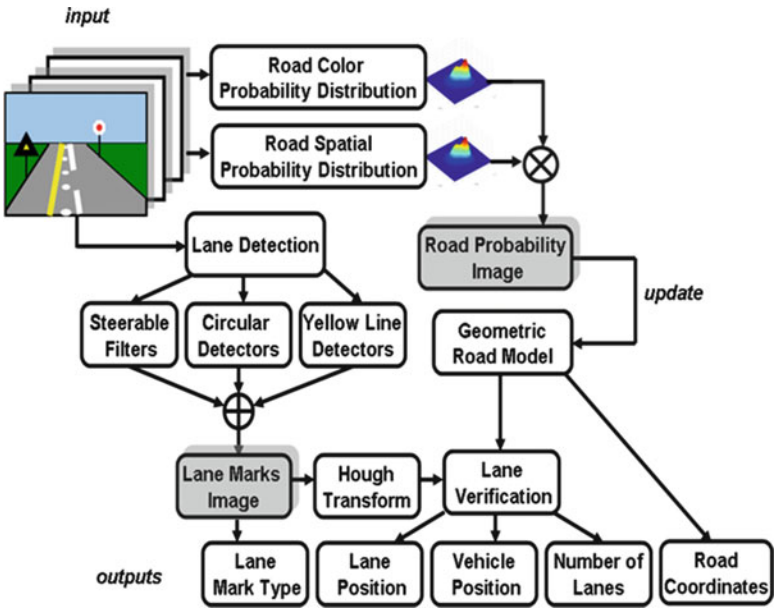



**Fig. 15.4** *Versatile lane tracking algorithm.* Some example results from lane tracking and road sign recognition parts are shown in Fig. 15.5

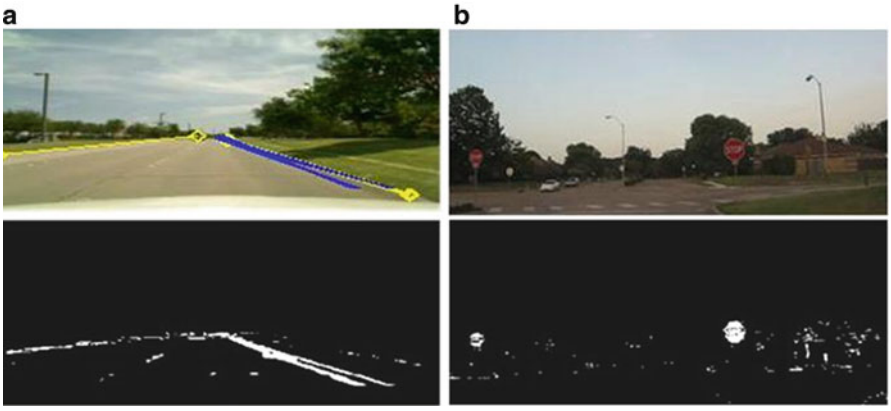



**Fig. 15.5** (**a**) An example output of road area detection and lane tracker. (**b**) Color segmented stop signs after dilation in road sign recognition module

These cases represent only a subset of the rule-based schemes, and it is possible to use more advanced rule-based construction methods such as fuzzy logic (Figs. 15.4 and 15.5).

## 15.4 Conclusion

In this chapter, a brief survey of state-of-the-art computer vision systems for in-vehicle applications was presented. In a critical approach to gauge these systems with their benefits, a utility analysis is performed given that an integrated traffic scene analysis system would be the most optimal to work on. With the encouragement from the utility analysis, the recent efforts of UTDrive in combining different image/video processing algorithms with an integrated mechatronics approach using the same sensor were reported.

## References

1. Fletcher L, Apostoloff N, Petersson L, Zelinsky A (2003) Vision in and out of vehicles. IEEE Intelligent Systems pp 12–17
2. (Online). www.fmcsa.dot.gov/documents/tb98-006.pdf
3. Morimoto CH, Mimica MRM (2005) Eye gaze tracking techniques for interactive applications. Comp Vis Im Und 98:4–24
4. Hutchinson TE, White KP Jr, Reichert KC, Frey LA (1989) Human–computer interaction using eye-gaze input. IEEE Trans Syst Man Cybern 19:1527–1533
5. Duchowski AT (2002) A breadth-first survey of eye tracking applications. Behav Res Meth Instrum Comput (BRMIC) 34(4):455–470
6. Zhu Z, Ji Q (2005) Robust real-time eye detection and tracking under variable lighting conditions and various face orientations. Comp Vis Im Und 98:124–154
7. Hansen DW, Pece AEC (2005) Eye tracking in the wild. Comp Vis Im Und 98:155–181
8. Sodhi M, Reimer B, Cohen, JL, Kaars R, Vastenburg E (2004) On-road driver eye movement tracking using head mounted devices. In: Proceedings of the Eye Tracking Research and Application Symposium (ETRA 2002), New Orleans. ACM Press, New York, pp 61–68
9. faceLab. http://www.seeingmachines.com/
10. Tobii Eye tracker. http://www.tobii.com
11. SR-Research. http://www.sr-research.com/
12. Xiao J, KanadeT, Cohn J (2002) Robust full motion recovery of head motion by dynamic templates and re-registration techniques. In: IEEE 5th international conference on automatic face and gesture recognition, Washington, DC, pp 156–162
13. Xu G, Sugimoto T (1998) Rits eye: a software-based system for realtime face detection and tracking using Pan-Tilt-Zoom controllable camera. In: Proceedings of fourteenth international conference on pattern recognition, vol 2. Brisbane, pp 1194–1197
14. Bergasa LM, Nuevo J, Sotelo MA, Barea R, Lopez ME (2006) Real-time system for monitoring driver vigilance. IEEE Trans ITS 6(1):63–72
15. Boyraz,P, Acar M, Kerr,D (2008) Multi-sensor driver drowsiness monitoring. IMechE PartD, J Automob Eng, 222(D11):2041–2063
16. De Silva LC, Pei CN, (2000) Bimodal emotion recognition. In: Proceedings of 4th IEEE international conference on automatic face and gesture recognition, Grenoble, 2000
17. Anderson K, McOwan PW (2006) A real time automated system for the recognition of human facial expressions. IEEE Trans Syst Man Cybern B Cybern 36(1):96–105
18. Shan C, Gong S, McOwan PW (2006) A comprehensive empirical study on linear subspace methods for facial expression analysis. In: 2006 conference on computer vision and pattern recognition workshop, New York City, New York. 17–22 June 2006, p 153, ISBN: 0-7695-2646-2

19. Pantic M, Patras I (2006) Dynamics of facial expression: recognition of facial actions and their temporal segments from face profile image sequences. IEEE Trans Syst Man Cybern B Cybern 36(2):433–449
20. Wierwille WW, Wreggit SS, Knipling RR (1994) Development of improved algorithms for on-line detection of driver drowsiness. In: International congress on transportation electronics (Dearborn), leading change. Warrendale, Society of Automotive Engineers
21. Knipling R, Rau P (2005) PERCLOS: a valid psycho-physiological measure of alertness as assessed by psychomotor vigilance. Technical Report of Federal Highway Administration, Office of Motor Crashes, USA FHWA-MCRT-98-006
22. Boyraz P, Yang X, Sathyanarayana A, Hansen JHL, Context-aware active vehicle safety and driver assistance, Enhanced Safety for Vehicles (ESV), 2009, 13–15 June, Stuttgart, Germany
23. McCall JC, Trivedi MM, Video-Based Lane Estimation and Tracking for Driver Assistance: Survey, System and Evaluation, IEEE Transactions on Intelligent Transportation Systems, vol. 7, no.1, pp. 20–37, March 2006
24. Kim YU, Oh, S-Y, Three-Feature Based Automatic Lane Detection Algorithm (TFALDA) for Autonomous Driving, IEEE Transactions on Intelligent Transportation Systems, vol. 4, no.4, pp. 219–225, Dec 2003
25. Wang Y, Teoh EK, Shen D, Lane detection and tracking using B-snake, Image and Vision Computing, vol 22, pp.269–280, 2004
26. Kim Z, Robust Lane Detection and Tracking in Challanging Scenarios, IEEE Transactions on Intelligent Transportation Systems, vol. 9, no.1, pp. 16–26, March 2008
27. Cheng HY, Jeng BS, Tseng PT, Fan, KC., 'Lane Detection with Moving Vehicles in Traffic Scenes', IEEE Transactions on Intelligent Transportation Systems, vol. 7, no.4, pp. 16–26, Dec 2006
28. Perez E, Javidi B, 'Non-linear Distortion- Tolerant Filters for Detection of Road Signs in Background Noise', IEEE Transactions on Vehicular Technology, vol. 51, no.3, pp. 567–576, May 2002
29. Broggi A, Cerri P, Medici P, Porta PP, Ghisio G, 'Real Time Road Signs Recognition', Proceedings of 2007 IEEE Intelligent Vehicles Symposium, Istanbul, Turkey, June 13–15, 2007
30. Jilmenez PG, Moreno G, Siegmann P, Arroyo SL, Bascon SM, 'Traffic sign shape classification based on Support Vector Machines and the FFT of the signature blobs', Proceedings of 2007 IEEE Intelligent Vehicles Symposium, Istanbul, Turkey, June 13–15, 2007
31. FARS Database: http://wwwfars. nhtsa.dot.gov/QueryTool/QuerySection/Report.aspx
32. Fatality Analysis Reporting System: http://www-fars.nhtsa.dot.gov/Main/index.aspx

# Chapter 16
# Integrated Pedestrian Detection and Localization Using Stereo Cameras

**Yu Wang and Jien Kato**

**Abstract** Detecting and localizing other traffic participants, especially pedestrians, from a moving vehicle have many applications in smart vehicles. In this work, we address these tasks by utilizing image sensors, namely stereo cameras mounted on a vehicle. Our proposed method integrates appearance-based pedestrian detection and sparse depth estimation. To benefit from depth estimation, we map the prior distribution of a human's actual height onto the image to update the detection result. Simultaneously, the depth information that contributed to correct pedestrians' hypotheses is used for a better localization. The difference with other previous works is that we take the trade-off between accuracy and computational cost in the first place of consideration and try to make the most efficient integration for onboard applications.

**Keywords** Histogram of Oriented Gradients (HOG) • INRIA data • Pedestrian detection • Stereo cameras

## 16.1 Introduction

Pedestrian detection is a very fundamental component in many applications, such as smart vehicles and robot navigation. In this chapter, we address this task by using image sensor which has obvious advantages with regard to visibility and low setup cost. In utilizing an image sensor, the common method of finding pedestrians is to slide a window over all the scales and positions of the image, extract features from each window to match with a pretrained model, and return a set of detections with high-matching scores. Obviously, more distinctive features and more representative

Y. Wang (✉) • J. Kato
Nagoya University, Nagoya 464-8603, Japan
e-mail: ywang@mv.ss.is.nagoya-u.ac.jp; jien@is.nagoya-u.ac.jp

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_16, 

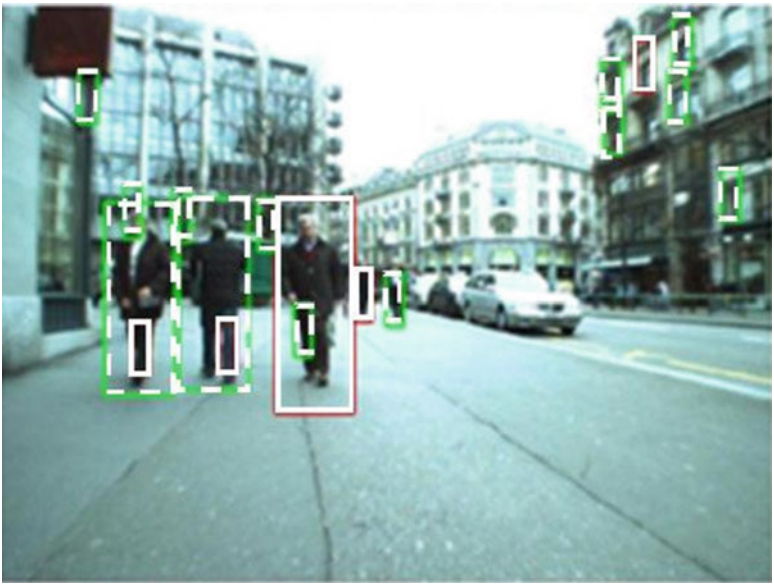


**Fig. 16.1** Select candidates strictly (*continuous line bounding boxes*); use looser criterion, more candidates were found (*dashed line bounding boxes*)

models will lead to better accuracy. However, improvement in this approach sometimes comes with additional processing time which usually slows down the entire system's speed [1].

In most real-world applications, speed and accuracy are crucial issues and should be addressed simultaneously. Of course, time-consuming methods are not recommended. At the same time, the simplest and fastest methods are not robust enough by themselves. An example is illustrated in Fig. 16.1. We apply a very simple pedestrian detector described in [2] on the street view image. When selecting the candidates using strict standards, as shown with continuous line bounding boxes, many true occurrences for pedestrians were missed. As we make the selection standard a little looser, some missed true occurrences were successfully found. But the second approach has a drawback. The false number increased. This means that a detector using a simple feature and coarse model is not, by itself, discriminative enough. The inadequacy, however, could be compensated to some extent by using other cues from the image and background knowledge.

Several studies have tried to use other cues for pedestrian detection. Leibe et al. [3] proposed the use of scene geometry to improve object detection. By assuming that pedestrians can only be possibly supported by the ground plane, some false detection results could be filtered out. In another work, Gavrila and Munder [4] presented a system which involves a cascade of modules wherein each unit utilizes complementary visual criteria to narrow down the image searching space. These two were both excellent works; however, additional cues are mainly used to get rid of false results but unable to support a true one.

In a more recent publication, Hoiem et al. [5] showed how to take advantage of scene elements to jointly determine the camera's viewpoint, object identities, and surface geometry efficiently from a single image. By utilizing the probabilistic relationship among the scene elements, their integration makes a simple detector become much more discriminative. However, since the geometric estimation module costs too much time, their method has limited usage.

In this chapter, we build upon these ideas and expand them by integrating a simple appearance-based object detector with sparse depth estimation. By properly modeling the interdependence between object hypotheses and their location, our method could not only reject object hypotheses with unreasonable depth but also let sensible depth information to support a true one. In addition, the way we use depth is independent of prior assumptions and could be done quite fast.

## 16.2 Overall Strategy

Taking stereo images as input, our system mainly has two complementary modules which are able to run in parallel. The first one is a pedestrian detector which processes images from the left camera to find pedestrian hypotheses with image features only. For every single pedestrian hypothesis in the image, the detector will assign a bounding box around it and a detection score to indicate its confidence. The second module is sparse depth estimation which utilizes the stereo images together to estimate a sparse depth map of images from the left camera.

In order to integrate the two modules together, we use a probabilistic way. We assume that an object's imaged height is conditioned on the object category and its distance with respect to the camera. But the object identity and their distance are independent from each other. Using a graphical model, we can represent the conditional independence over the object identities $o_i$, their imaged height $h_i$, and the corresponding 3D distance $d_i$, as shown in Fig. 16.2. The $I$ denotes the left camera image, and $D$ means sparse depth map which could be estimated using the stereo image pair, both are observed evidences in our model. Typically, we have $n$ object hypotheses in an image, where $n$ varies by image.

With this model, the overall joint probability of the scene elements could be written in the following equation as

$$P(o,d,h,I,D) = \prod_i P(o_i)P(d_i)P(D|d_i)P(I|o_i)P(h_i|o_i,d_i) \tag{16.1}$$

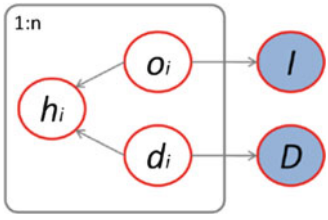



**Fig. 16.2** Graphical model

With observed evidences $I$ and $D$, we can use Bayes rule to give the likelihood of the scene elements conditioned on the evidences as

$$P(o, d, h|I, D) \propto \prod_i P(h_i|o_i, d_i)P(o_i|I)P(d_i|D) \tag{16.2}$$

The proportionality equation is with respect to $I$ and$D$ which is constant evidence from stereo images. On the right-hand side, $P(o_i|I)$ means the confidence of an object hypothesis given image evidence, which could be estimated by our pedestrian detector. $P(h_i|o_i, d_i)$ indicates the probability of a hypothesis observed with imaged height $h_i$, conditioned on its category and 3D depth. In our case, it could be estimated by introducing a prior distribution of the pedestrians' actual height. That $P(d_i|D)$ is the confidence of depth estimation given the depth evidence from a depth map.

In this work, we estimate depth in an explicit way wherein the depth for each object hypothesis is exact and without any probabilistic description. This allows us to margin out the $d$ on both left- and right-hand sides; for a single object hypothesis, we then get

$$P(o_i, h_i|I, D) \propto P(h_i|o_i, d_i)P(o_i|I) \tag{16.3}$$

where $P(o_i, h_i|I, D)$means, given the image evidences $I$ and $D$, the probability of an object hypothesis $o_i$ with its imaged height $h_i$. It is propagated with the $P(h_i|o_i, d_i)$ and $P(o_i|I)$, and could be considered as an improved confidence estimation of object hypothesis which not only takes into account the image evidence but also the depth information. We get the improved detection result by sorting the score of $P(o_i, h_i|I, D)$ for each object hypothesis and selecting the high ones. In the following paragraph, we will introduce the way we get $P(h_i|o_i, d_i)$and $P(o_i|I)$ from stereo images.

## 16.3 Pedestrian Detection

In order to obtain a set of pedestrian hypotheses, we built a baseline detector similar to the one described in [6]. As classifier, the Histogram of Oriented Gradients (HOG) feature and linear support vector machine was used. To distinguish this from the original 36-dimensional HOG feature used in [6], we employed an alternative 31-dimensional implementation from [1] to replace it. Also, to simplify the training process and speed up the runtime performance, a lower-dimensional feature set which could make a classifier with less parameters was utilized.

While training our detector, we used an existing package SVMPerf [7], which is highly optimized for training binary two-class classification SVMs with large data set. For this study, the INRIA person data set which has been organized into 3,610 positive samples of pedestrian with the size 70 by 134 was utilized. The negative

samples contain a fixed number of 15,000 patches that randomly selected from 1,239 person-free images of that data set. The training returns a 3,255-dimensional linear classifier (the size of 70 by 134 patch image's feature vector).

When a novel image emerges, we slide a window over the scales and positions to find the hypotheses. For each subwindow, we evaluate a score by doing dot product of the pretrained linear model and feature vector of the image patch. If the score is larger than the threshold, we either take it as a hypothesis or discard it. Typically, for an image portion that is likely to be a pedestrian instance, the score for the boxes around it will be very high. In order to eliminate any overlapped bounding boxes for the same instance, we perform non-maxima suppression to select only one box for each instance.

In this way, we get a set of hypotheses which is expected to have a pedestrian instance, each one with a bounding box and a classification score. However, the classification score is within the interval $(-\infty, +\infty)$. Since our graphical model wants a probabilistic input $P(o_i|I)$ which should be in the interval (0, 1), we therefore transform the SVM output into a probability form with logistic regression:

$$P = \frac{1}{1 + e^{Ax+B}} \tag{16.4}$$

where x is the classification score output from the dot product, P is the corresponding probability form of the score, and A and B are parameters which could be estimated by collecting a set of x and p. With novel classification score x′, we take the corresponding p′ as $P(o_i|I)$.

## 16.4 Localization of Pedestrian Instance

The use of a descriptor-based matching approach to obtain a sparse depth map distinguishes our work from the previous studies on how to estimate depth in a dense way. Though it could only provide a sparse representation of the scene, it is less ambiguous than dense matching which suffers from occlusion and nontexture regions. To make the depth map not "too sparse," we use two different kinds of key points as in [8] to relate the stereo images (Fig. 16.3).

We extract scale-invariant key points using Difference-of-Gaussian operator [10] and corner key point with Harris operator. For the scale-invariant key points, we utilize a GPU implementation of SIFT to compute their descriptors and match them by measuring the Euclidean distance. This implementation benefits from the Nvidia's CUDA technology and can get a speed of 25 Hz when processing images with size 640 by 480, which we think is enough for general real-world applications. The corner points are matched with a correlation window by normalized cross-correlation. Using two kinds of key points could help establish sufficient raw correspondences fast.

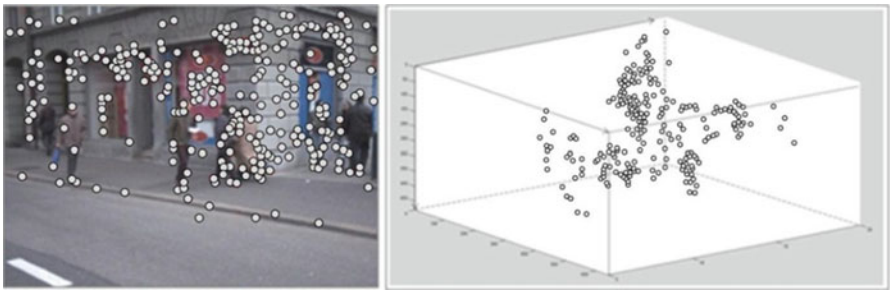


**Fig. 16.3** Key points (*left*) and their 3D coordinates

With the raw matching result, we further refine them by enforcing Epipolar constraint and perform linear triangulation to get their 3D coordinates through precalibrated camera matrices. We set the left camera's optical center as the world origin, and then the z coordinate is the depth of each matched key point.

For each object hypothesis that we obtained, we collect all the matched key points inside its bounding box and select one representative for that bounding box's depth. Here we use a simple way to select the representative point by finding the nearest feature point around the diagonals' intersection and take the depth as the hypothesis' depth $d_i$.

Despite its simplicity, this solution performs reasonably well compared with other approaches such as using mean-shift to directly find the coordinates of the mass center. The reason may be that a lot of matched point is found around the object's boundary, and the mean-shift stops at local maxima frequently.

## 16.5 Utilize a Prior Height Distribution

The probability for the imaged height of a pedestrian hypothesis $P(h_i|o_i, d_i)$ is obtained by a product of the observed height of its bounding box $h_i$ and a distance-conditioned height distribution $P(h_i|o_i, d_i)$. The later one is obtained using depth $d_i$. and a prior distribution of human's actual height.

Given a class-conditioned object hypothesis$o_i$, its distance $d_i$, and the camera's focal length $f$ which we already know from the camera's calibration, we further model the height of an adult human using a simple Gaussian. The parameters of this Gaussian could be estimated from statistical data. We follow [5] to use a mean of 1.7 m and a standard derivation of 0.085 m for the pedestrian height distribution; therefore, we have the height distribution as $H \sim N\left(1.7, 0.085^2\right)$.

Given the prior distribution of pedestrian's actual height H, by using similarity relation, we can represent the imaged pedestrian's height as $h = Hf|D$. Because of $H \sim N\left(1.7, 0.085^2\right)$, h is also a simple Gaussian with $1.7f/d_i$ as mean and $0.085f/d_i$ as standard derivation. Therefore, we get

$$P(h|o_i, d_i) \sim N\left(1.7\frac{f}{d_i}, \left(0.085\frac{f}{d_i}\right)^2\right) \tag{16.5}$$

With this imaged height distribution and the observed height $h_i$ of each bounding box, the confidence of every single hypothesis could be updated by taking the product of the detector output $P(o_i|I)$ and the $P(h|o_i, d_i)$. The updated confidence obtained in this way has thus taken into account the depth information and is expected to be more discriminative than the visual-features-only estimated result.

## 16.6 Experimental Results

We now present the experiment to show the performance of our method. The test data we used is collected from the ETHZ pedestrians' data set [9], which contains 5,235 pairs of stereo images that have been taken from either moving vehicles or mobile robots. All these images are from precalibrated cameras, with pedestrians on the left camera images annotated with bounding boxes as ground truth. The data were taken as sequences, so there are some continuous frames with almost the same scene. Since our work is only trying to evaluate the detection performance of single frame, we rearrange the data set by picking out image pairs with different scene structures. The final test set contains 133 pairs of stereo images with 798 annotations as ground truth.

In our experiment, we test three detection systems. First is our baseline detector, which uses HOG feature and linear support vector machine. The second is our proposed system which integrates this baseline detector and sparse depth estimation. The third one is UoCTTI detector [1], which employs mixtures of multiscale deformable part models. This is one of the best detectors in the PASCAL object detection challenge.

Some example detection results of the three systems on difficult images from our 133 stereo pairs' data set are shown in Fig. 16.4. The three columns from left to right show the output from our baseline detection system, proposed integration system, and UoCTTI system, respectively, on the same image. For a fair comparison, only the detections within the top ten confidences of each system are treated as output.

In general, the UoCTTI detector performed the best, as a result of more advanced modeling. Besides robust low-level feature, this detector uses a hierarchical structure model called deformable part model to represent the object category. In general, their detector finds pedestrians not only because they look like a person but also because they have parts (such as head, hands, legs), and these parts have appropriate positions. This makes the detector especially robust against occlusion. When distinguishing different human parts in crowded scene and large pedestrian volume conditions, the UoCTTI performs much better than our baseline detector system and our integration.

**Fig. 16.4** Experimental results: (*left*) baseline system, (*middle*) integration system, (*right*) UoCTTI system

When compared to the raw output of our baseline detector, our integration system did quite well and shows significant improvement in the different scenarios. The reason is that we integrated the depth cue. Through it, the system could find pedestrians better by taking into account the observed height of detections and update the detection confidence to become more reasonable.

From the experiment, in some board scene images, as shown in the second row of Fig. 16.4, our integration system could perform better than the UoCTTI detector. We think the reason is the trade-off between different sources of information. While the UoCTTI detector utilizes both a deformable part model and the position of body parts to improve the detection, at the same time, there are drawbacks to this approach. Because the final detection result is partially based on the parts and the corresponding locations, in cases of low image resolution (parts are not visually clear) or the pedestrian instance is small (parts are not distinguishable), their model will penalize the detection and result in a low detection score. In contrast, our integration system uses depth information which is not dependent on any kind of condition (as long as the depth is accurately estimated). For the pedestrian instances that are small in the image, depth will help more because depth information itself does not depend on the resolution of the image.

Our quantitative experiment uses precision–recall (PR) curve to measure how a detection system performs in practice. It says a big deal about how the objects are

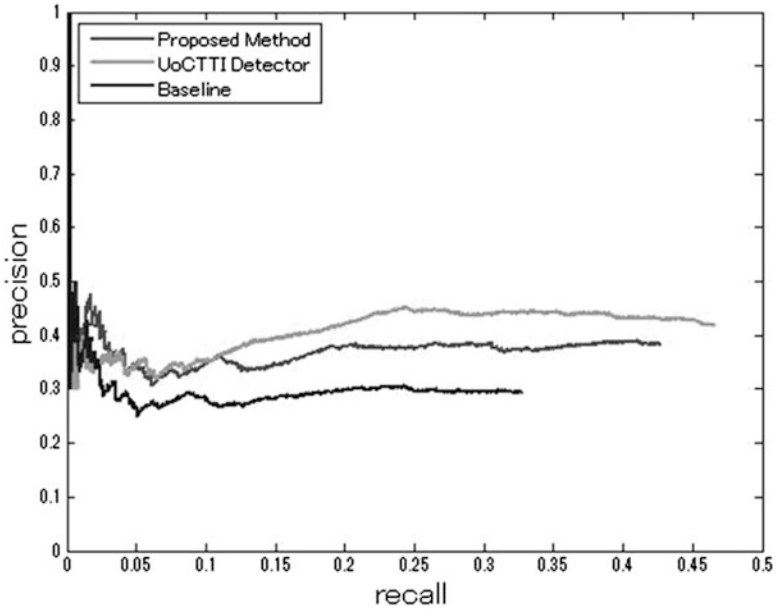



**Fig. 16.5** PR curve for the detection performance

detected in practice. For a fair comparison, we also take top ten ranked hypotheses as system output. The comparison of the three systems' performance on the 133 stereo pairs is plotted in Fig. 16.5.

In most cases, the detector with deformable-part-based model has maintained a precision near 0.5. By integrating depth information, our proposed system outperforms the baseline detector significantly and closes to the best one.

We also compute an average precision for the three methods to show the overall performance. The results are 0.2325 (our method), 0.1738 (baseline), and 0.2530 (UoCTTI), respectively.

Without any optimization of speed, on a 2.83-G Intel Core 2 Quad CPU with 4 G RAM, the average speed of the three methods are 1.73 s (our method), 1.7 s (baseline), and 8.4 s (UoCTTI) on a single 640 by 480 image. The UoCTTI detector is quite time-consuming. It uses nearly five times more than our baseline detector. Since the UoCTTI detector also uses HOG as low-level feature set, its disadvantage in runtime may mainly boil down to the complicated model it uses. Therefore, even if it is powerful, it could not be used in some applications before the runtime issue could be resolved.

By carefully selecting the efficient cues, our integration system could also be very fast. Though this runtime performance is not good enough for some applications, it still has room for improvement. Currently, in our system, the most time-consuming part is the HOG feature pyramid computation and sliding window searching. Since these two kinds of processing can be done much more faster by using GPU programming, our integration system still have the potential to be used in real-time applications.

## 16.7 Conclusion

In this chapter, we proposed a method for pedestrian detection in traffic areas. We integrate typical object detection method with sparse depth estimation. This enables us to use 3D depth information naturally and improve detection accuracy by taking into account the human knowledge that "things become smaller when they move farther."

The efficiency of our integration was shown in our experiment. Without adding too much processing time, our method could improve the performance of our baseline detection system to a significant level, even close to a state-of-the-art detection system [1]. For the latter one, processing time for the detection with the same image size will cost nearly five times. Besides efficiency, another thing that we found out from the experiment is that the utilization of depth is independent of image resolution and instance size. This leads to stable improvement over the baseline system for all different kinds of scenes.

However, some issues still exist in the current system. First, the depth information that we introduced is obtained in an explicit way. This will, in some level, make the system sensitive against error in depth estimation. Secondly, our system is not good in handling occlusion and therefore quite weak in some crowd scenes. In the future work, we will mainly focus on robust depth estimation and occlusion handling.

## References

1. Felzenszwalb P, Girshick RB, McAllester D, Ramanan D (2010) Object detection with discriminatively trained part based models. IEEE Trans Pattern Anal Mach Intell 32 (9):1627–1645
2. Torralba A, Murphy KP, Freeman WT (2004) Sharing features: efficient boosting procedures for multiclass object detection. In: IEEE Conference on computer vision and pattern recognition, Washington, 2004
3. Leibe B, Schindler K, Cornelis N, Van Gool L (2008) Coupled object detection and tracking from static cameras and moving vehicles. IEEE Trans Pattern Anal Mach Intell 30(10):1683–1698
4. Gavrila DM, Munder S (2007) Multi-cue pedestrian detection and tracking from a moving vehicle. Int J Comput Vision 73(1):41–59
5. Hoiem D, Efros AA, Hebert M (2008) Putting objects in perspective. Int J Comput Vision 80 (1):3–15
6. Dalal N, Bill Triggs B (2005) Histograms of oriented gradients for human detection In:. IEEE conference on computer vision and pattern recognition, San Diego, 2005
7. Joachims T (1999) Making large-scale support vector machine learning practical, MIT Press Cambridge, MA, USA
8. Wang Y, Kato J (2010) Reference view generating of traffic intersection. ICIC Expr Lett 4(4):1083–1088
9. Ess A, Leibe B, Schindler K, Van Gool L (2009) Robust multi-person tracking from a mobile platform. IEEE Trans Pattern Anal Mach Intell 31(10):1831–1846
10. Lowe DG (2008) Distinctive image features from scale-invariant keypoints. Int J Comput Vision 60(2):91–110

# Chapter 17
# An Examination of Overtaking Judgments Based on Limitations in the Human Perceptual System: Implications for the Design of Driver-Assistance Systems

**Anand Tharanathan**

**Abstract** Traffic accidents that occur during an overtaking maneuver is a worldwide problem. Such accidents lead to several injuries and fatalities each year. Similarly, rear-end collisions also comprise a large proportion of accidents each year. However, there is a high disparity in the number of studies that has examined the underlying causes for the two types of accidents. Several studies have investigated driver performance during a car-following task, which have in turn led to the development of driver-assistance systems to avoid rear-end collisions. However, only few studies have attempted to study judgments during an overtaking task, and even fewer studies have investigated the perceptual demands on a driver during such a maneuver. Since driving is primarily a visual task, in this paper, I conduct a detailed examination of an overtaking task with an emphasis on the limitations in the human perceptual system. Also, to better understand the complexity of an overtaking task, I compare and contrast an overtaking task with a car-following task. As an implication for design, I address certain disadvantages in using a typical forward-collision-avoidance-warning-system (designed to avoid rear-end collision) to aid an overtaking maneuver. Considering such limitations, seven functional requirements have been described that are important to be considered in the design of driver-assistance systems to support safe overtaking. Finally, I propose a model for the design of driver-assistance systems that emphasizes overcoming drivers' perceptual limitations by enhancing the effectiveness of the visual information that is available from the traffic environment.

**Keywords** Driver-assistance systems • Overtaking • Perceptual judgments

A. Tharanathan (✉)
Department of Psychology, Texas Tech University, Visual Performance Laboratory, Lubbock, TX, USA
e-mail: anandtharanathan@gmail.com; Anand.Tharanathan@Honeywell.com

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_17, 

## 17.1 Introduction

Traffic accidents account for several deaths and injuries each year. In Europe, 42,500 people are killed and 3,500,000 are injured every year due to traffic accidents [1]. Similarly, in the United States, over 42,000 people are killed in a year due to motor vehicle crashes [2]. Therefore, it is important to understand the underlying reasons for such collisions. It has been suggested that approximately 90% of all traffic accidents can be attributed to human error [3], and studies have investigated driver performance during car following [4], left and right turn maneuvers, [5] and at traffic intersections [6]. Also, research on computer vision and intelligent transportation systems have led to the development of several driver-assistance systems (DAS) that aid drivers, for example, in car following, inattention detection, pedestrian spotting, and lane keeping [7–9].

However, fewer studies have focused on drivers' judgments during overtaking maneuvers [10, 11]. This is quite surprising because overtaking maneuvers lead to many fatal accidents each year [12]. For example, between 1995 and 2000, about 26 traffic participants died each year in Netherlands due to overtaking failures [12]. In addition, it was reported that overtaking maneuvers led to a considerable proportion of injury-causing accidents in Nottinghamshire, England [13]. Furthermore, in the United States, there were 138,000 accidents due to overtaking in the year 2000, and such accidents accounted for 2.1% of all fatal crashes and 1.1% of injury crashes [14]. In short, global accident data suggests that it is critical to identify the underlying causes for accidents during overtaking maneuvers. Since driving is primarily a visual task [15], it will be beneficial to identify limitations in the human perceptual system that lead to erroneous judgments during such maneuvers. Consequently, it is essential to explore ergonomically appropriate solutions to overcome such limitations by developing DAS to help drivers during overtaking maneuvers.

In this article, we address five specific topics. First, we identify the sources of visual information that drivers rely on during an overtaking maneuver and the perceptual judgments that they typically make during such maneuvers. Second, to better understand the complexity of an overtaking maneuver, we compare and contrast an overtaking task with car following, especially because judgments during car following have been widely studied [16–18]. We outline the critical differences in the available visual information and associated judgments during the two types of tasks. Third, since forward-collision-avoidance-warning-systems (FCAWS) are available to aid drivers during car following, we investigate the possibility of using typical FCAWS to aid overtaking maneuvers. Based on the known limitations in the human perceptual system and the functional capabilities of the currently available FCAWS, we report certain disadvantages in using such FCAWS to support overtaking maneuvers. Fourth, we describe seven functional requirements that are important to be considered in the design of an ergonomically efficient DAS to support overtaking maneuvers. Finally, we propose a model for the design of DAS that emphasizes on overcoming drivers' perceptual limitations by enhancing the effectiveness of the available visual information.

### *17.1.1 Identifying the Problem*

Drivers perform maneuvers like lane changes, left and right turns, car following, and overtaking, and it is important to avoid collisions during such maneuvers. It has been suggested that an overtaking maneuver is especially a complex one that has a high probability for human error [19]. Although studies suggest that an overtaking maneuver is a complex task, the sources of visual information that guide such maneuvers are largely unknown. Past research suggests that drivers use different strategies to complete an overtaking maneuver [11]. Also, it has been noted that drivers make erroneous judgments about the temporal gap required to complete a safe overtaking maneuver [20]. Also, drivers are typically not accurate in judging the distance required to pass [11, 21]. In short, first, it is important to clearly identify the sources of visual information that drivers use during an overtaking maneuver. Then, it is essential to examine the effectiveness of such sources. If the effectiveness of the visual information is low, then the quality of the consequent perceptual judgments will be poor. In contrast, if the effectiveness of the visual information is high, then the quality of the consequent perceptual judgments will be better. An important contribution from DAS can be in enhancing the effectiveness of visual information.

## 17.2 Visual Information that Drivers Rely on During Overtaking Maneuvers

A recent study reported that after drivers decide to overtake, they determine whether the distance until the first oncoming car is sufficient to initiate the maneuver. Additionally, it was suggested that since the self, lead car, and oncoming car are in motion, it is essential to perceive the velocity and time-to-contact (TTC) information of the cars before initiating the overtaking maneuver and while passing [22]. Furthermore, the oncoming car might be accelerating or decelerating. Therefore, drivers have to accurately judge the rate of change of velocity of the oncoming car. In short, judgments about distance, velocity, acceleration, deceleration, and TTC are critical during overtaking maneuvers. Needless to state, an overtaking maneuver is perceptually more demanding to a driver because he or she has to make such judgments for more than one vehicle – the lead car and the oncoming car. Next, I examine the effectiveness of such visual information.

### *17.2.1 Time-to-Collision*

Lee (1976) noted that the time-to-collision with an approaching vehicle is optically specified by the ratio of the angular extent (e.g., visual angle subtended on the driver's eye by the front bumper of the oncoming car) to the rate of change of that

angular extent [23]. He named this ratio *tau,* and he suggested that the human perceptual system is sensitive to *tau*. However, *tau* has an important limitation, which is indirectly due to the limitation in the spatiotemporal resolution of the human visual system [24]. Specifically, the threshold to detect an increase in an approaching object's optical size (or visual angle subtended at the eye) is 0.017 deg [25] or 0.172 deg/s [26]. *Tau* is effective only if the optical expansion rate exceeds threshold (Gray and Regan 1998). Due to such limitations in the human perceptual system, drivers cannot accurately judge the TTC with an approaching vehicle when the actual TTC is relatively high. However, during an overtaking maneuver, the TTC with the oncoming car might be relatively large, sometimes around 6 s [22].

### *17.2.2 Distance of the Oncoming Car*

The vergence angle of the eyes can provide information about the distance of a fixated object, but such a source of information is accurate only when the fixated object is less than 10 m from the self [11]. Importantly, during an overtaking maneuver, an oncoming car can be more than 100 m from the driver [22]. Also, the effectiveness of different sources of visual information varies with distance [27]. At far distances, optical expansion rate (i.e., the rate of change of the visual angle subtended on the driver's eye) of an oncoming vehicle and *tau,* which specifies the TTC with the oncoming vehicle, may be below threshold.

### *17.2.3 Velocity of the Oncoming Car*

Due to the limitation in the spatiotemporal resolution of the human visual system, drivers will have more difficulty in perceiving the velocity of a vehicle that is small in size, or one that is approaching slowly, compared with a large vehicle or one that approaches fast. This is because the optical expansion rate is smaller for vehicles that are smaller in size, or ones that move slowly, compared with those that are larger or move faster [24].

### *17.2.4 Acceleration*

Studies have reported that the human perceptual system is incapable of accurately perceiving accelerated motion [28–30]. Specifically, studies have shown that observers overestimate the TTC with a car that approaches at an accelerated rate [31]. An overestimation of TTC implies that drivers judge an approaching vehicle to collide with them much later than it actually would. Such overestimations of TTC can be fatal while driving.

### *17.2.5 Relative Direction of Approach*

Gray et al. (2004) reported that the perceived velocity is relatively lower when the observer and an approaching object are moving in opposite directions [32]. They suggested that this leads to erroneous judgments of TTC. Such erroneous judgments are critical during an overtaking maneuver because while overtaking, the self and the oncoming car are typically moving in opposite directions, head on.

### *17.2.6 Motion Adaptation*

Studies have reported effects of motion adaptation on certain judgments involved in driving [33] and especially during overtaking maneuvers [11]. Based on driver performance in a driving simulator, such studies showed that while driving on rural roads, after drivers have been exposed to a nonchanging gap between the self and the lead car for an extensive period of time, the thresholds for detecting a change in gap between the self and the lead car or the oncoming car is higher [11]. In other words, it takes longer for drivers to detect the change in distance between them and the oncoming car when overtaking on rural roads. From a safety perspective, this is quite dangerous. Congruent with such studies, Hegeman et al. (2005) reported that most of the overtaking accidents in Netherlands occur on rural roads [22].

In sum, there are several limitations in the human perceptual system to perceive distance and motion. Such limitations become more critical under certain traffic conditions which might lead to erroneous judgments about distance, velocity, and TTC with an oncoming car. At this point, to better understand the complexity of an overtaking task and the additional demands it imposes on a driver, it will be beneficial to compare and contrast an overtaking task with a closely related type of driving task that has been widely studied; car following. I do so in the following section.

## 17.3 Critical Differences Between an Overtaking Task and Car Following

Inaccurate judgments during overtaking maneuvers can result in overtaking accidents. Similarly, inaccurate judgments during car following can result in rear-end collisions. Twenty-five percent of accidents on the road are rear-end collisions [34]. Therefore, several studies have investigated the types of perceptual judgments that are critical for safe car following. For example, studies have investigated deceleration judgments [16, 18, 35], headway estimation [36], and TTC judgments [1, 2, 6, 14, 17, 18, 29, 31, 34–43] during car following. Interestingly, all judgments that are crucial for safe car following are also critical during overtaking maneuvers. However, there are three

specific differences between an overtaking maneuver and a car-following task which makes an overtaking maneuver typically more complex.

First, during an overtaking maneuver, drivers have to judge the motion of both the lead car (car to be overtaken) and the oncoming car. In contrast, during a car-following task, the driver typically needs to judge only the motion of the lead car. Therefore, the perceptual demands associated with an overtaking task might be higher than a car-following task. Second, driver distraction is one of the leading causes for rear-end collisions. In other words, rear-end collisions typically happen because drivers do not attend to the road for adequate amount of time [40]. In contrast, during an overtaking maneuver, drivers typically have their eyes on the road. Therefore, inaccurate judgments during overtaking maneuvers are primarily due to the perceptual limitations in human beings to process motion, rather than a lack of attention. Third, rear-end collisions typically happen at shorter headways [44] when the self is following the lead car very closely. In contrast, an overtaking maneuver might be initiated when the distance between the self and the oncoming car is quite large. Furthermore, during poor driving conditions, such as during rain, snow, or night, overtaking maneuvers can become even more perceptually taxing to drivers than usual.

In sum, there are critical differences between a typical car-following task and an overtaking task. It is clearly evident that limitations in the human perceptual system contribute highly towards inaccurate judgments during an overtaking maneuver, rather than a lack of driver's attention on the road. Therefore, it is important for designers to consider such limitations in the human perceptual system in the design of intelligent transportation systems that can enhance drivers' visual performance during such complex maneuvers. Currently, there are forward-collision-avoidance-warning-systems (FCAWS) that can aid drivers in safe car following. In the next section, I outline the typical functional requirements considered in the design of such FCAWS and how it might be critically disadvantageous to utilize such systems to assist drivers during overtaking maneuvers.

### *17.3.1 Current Forward-Collision-Avoidance-Warning-Systems*

Due to the differences between the two types of maneuvers, FCAWS that are specifically designed to aid car following [45] might not always be effective to aid overtaking maneuvers. Specifically, there are three important limitations regarding the use of FCAWS to aid overtaking maneuvers. First, FCAWS typically aid in detecting the dynamics of only one host vehicle; the lead car [46]. However, in an overtaking maneuver, there are two cars involved, the lead car and the oncoming car. Second, one of the functional requirements of a typical FCAWS is that it should be able to detect the presence of a lead car only up to 100 m in front of the self [46]. However, during an overtaking maneuver, an oncoming car might be even further [22]. Third, FCAWS typically issue warnings when the TTC with the host vehicle reaches a threshold value, which is relatively small, for example, 2 s [47].

However, an overtaking maneuver can take up to 6 s [22], and prior research shows that the accuracy of TTC judgments decreases as the actual TTC increases [17]. In sum, the typical functional requirements and thresholds considered in the design of FCAWS to aid safe car following cannot be generalized to aid overtaking maneuvers. Therefore, as an aid to overtaking maneuvers, DAS needs a separate set of functional requirements and thresholds.

## 17.4 Driver-Assistance Warning Systems for Overtaking Maneuvers

In this section, I identify seven functional requirements that have to be considered as guidelines while designing DAS to support overtaking maneuvers. Importantly, the requirements have been developed to overcome the perceptual limitations of drivers while carrying out overtaking maneuvers. Also, the *human factor* involved in such a human–automation interaction is considered.

### *17.4.1 Focus on Object-Motion Processing in Addition to Object Recognition*

Most rear-end collisions occur when the driver is distracted and does not view the traffic for adequate amount of time [40]. Therefore, forward-collision-avoidance-warning-systems designed to support safe car following typically focus on object recognition because it assumes that the driver is not attentive to the traffic scene. However, during an overtaking maneuver, the drivers typically have their eyes on the road throughout the maneuver and hence are attending to the traffic scene. Under such conditions, it is the incapability of the human perceptual system to perceive distance and motion that leads to inaccurate judgments rather than a lack of attention to the traffic scene. Therefore, the DAS should be developed to overcome such limitations.

### *17.4.2 Capability to Detect the Motion Parameters of a Vehicle at Far Distances from the Self*

Human perceptual system is typically incapable of accurately detecting a vehicle's motion when its distance from the self is very far. However, overtaking maneuvers might have to be performed at such distances (e.g., 200 m).

### 17.4.3 Capability to Detect Motion Parameters of the Oncoming Vehicle

The DAS must be capable of accurately detecting in real time all parameters of motion of the oncoming vehicle, relative to the self (velocity, acceleration, deceleration, TTC, direction of movement, etc.). The human perceptual system is incapable of accurately detecting a vehicle's optical expansion rate when the vehicle is small in size, at a far distance, or moving at a slow velocity [24]. In addition, the perceptual system cannot accurately judge accelerated motion, especially when the TTC is greater than 1 s [30]. Furthermore, accuracy of velocity and TTC judgments vary with respect to the relative direction of movement between the oncoming car and the self [32]. Finally, judgments of TTC are not accurate when the optical expansion rate is below threshold [48].

### 17.4.4 Capability to Detect the Motion Parameters of the Lead Vehicle

The DAS must be capable of accurately detecting in real time all parameters of the lead vehicle's motion (the vehicle which is to be overtaken), relative to the self (velocity, acceleration, deceleration, time-to-passage, direction of movement, etc.). All the justifications from functional requirement "c" apply here. In addition, research [43] suggests that it is easier for human beings to attend to one object and search (or judge) for a particular parameter (e.g., presence of a distracter) compared with simultaneous searches between two objects for the same parameter (e.g., TTC with oncoming car and time-to-passage with the lead vehicle, both of which are temporal parameters). Furthermore, [49] showed that TTC judgments are affected when the number of objects in the scene increases.

### 17.4.5 The DAS Must Be Sensitive to the Physical and Dynamic Capabilities of the (Self's) Car

The DAS should be able to continuously sense and process the possible gain (e.g., possible acceleration within a temporal window) of the (self's) car under the given set of environmental conditions (e.g., gyroscopic capabilities of the car, friction of road during rain, snow, etc.). Such information must be processed simultaneously with the dynamics of other vehicles (or objects) as mentioned in functional requirements "b" through "d." When drivers have to process such complex information and make respective calculations, it will significantly increase their mental workload. However, studies suggest that warning systems should be designed to minimize work load on drivers (Miller and Huang 2002).

## 17.5 The Human Factor in Human–Automation Interaction

Driver-assistance systems are automated systems, and it is important to consider the implications of such automated systems on driver performance. According to the Parasuraman–Sheridan–Wickens model [41] of automation, any automation can be classified into four levels: (1) information acquisition, (2) information analysis, (3) decision selection, and (4) action implementation. Automation can be done at any or all the four levels. However, a major limitation in completely automating a system (i.e., across all four levels) is that it leaves the human with only a supervisory role. Therefore, if the system has to return back to manual functioning, or when there is a malfunction in the automation, it leads to a decrease in the driver's situation awareness and an increase in workload [38]. Therefore, automation across all four levels is not always ideal. The next two functional requirements have been proposed considering such implications for human–automation interaction and to enhance the overall performance of the driver–automation system.

### *17.5.1 Automate the Decision Selection Stage*

Functional requirements "b" through "e" imply that the DAS should be automated across the information acquisition and information analysis stages. Additionally, the decision selection stage should also be automated. Based on existing traffic conditions, the DAS should provide drivers with a decision. Such a decision should be based on the possibility of overtaking, and the decision should be conveyed to the driver in an effective manner. Specifically, based on the information analyses, the DAS should be able to determine whether it is *possible* or *not possible* to safely complete the overtaking maneuver. Let us consider a scenario. The self is going uphill, on a rainy night. The self initiated the overtaking maneuver by moving to the overtaking lane. However, there is an oncoming car, coming downhill. Based on the information available from the environment (that of the lead car and the oncoming car), the DAS calculates the time to complete the maneuver as 8 s, and the minimum speed to be 100 kph. However, based on the calculation of the underlying dynamics of the self's car and an additional margin of safety, the DAS decides that it is *not possible* to complete a safe overtaking maneuver. After analyzing this *possibility* of completing a safe overtaking maneuver, a decision should be provided to the driver in an effective manner.

Automating the decision selection stage is important for DAS designed for overtaking maneuvers. When drivers have to process complex information and make necessary calculations and analyses, it will significantly increase the mental workload on the driver. This can affect the perceptual judgments and consequent actions. Also, based on currently available research on motion perception, it is unclear about the capability of humans to judge the ideal acceleration required to complete an overtaking maneuver. The directly available optic information to sense the ideal acceleration is

very complex, and research suggests that the human perceptual system might not be able to compute such complex calculations [43, 50]. Therefore, it will be beneficial to automate the decision selection stage such that the DAS can provide drivers with a decision on whether it is safe or not to carry out the overtaking maneuver.

### *17.5.2 Provide an Auditory Warning, but Do Not Impose It on the Driver*

The modality and type of warning is important. The warning should be auditory, and the DAS should not impose the warning on the driver. Also, the action implementation stage should not be automated. This functional requirement is supported by two specific reasons. First, during an overtaking maneuver, the visual system is overloaded. Under such conditions, an auditory warning is more effective and helps to capture the driver's attention more readily (e.g., [42]). Second, research suggests that human performance deteriorates when automation imposes actions on them (e.g., [38]). Therefore, it is important that the driver is always in active control of the vehicle. The DAS should only provide the driver with a decision; it should be the driver who finally decides whether to accept the decision provided by the DAS.

## 17.6 Designing DAS to Overcome Perceptual Limitations in Drivers

As is evident from the above analysis of an overtaking maneuver, it is critical to consider the limitations in the human perceptual system while designing better DAS. One of the major assumptions of the current DAS is that such systems will be helpful when drivers are not attending to the road, or in other words, are distracted. For example, it has been suggested that if drivers are looking at a potential problem in the traffic scene, a warning is irrelevant [8]. This might be true for car following, when the lead vehicle is much closer to the driver. However, an overtaking maneuver is a good example of how a warning might be effective, even when drivers are looking at a potential problem in the traffic scene. During an overtaking maneuver, deterioration in driver performance is primarily due to the incapability of drivers to process distance and motion information when vehicles are far away, approaching at a relatively slow velocity, or are smaller in size. Therefore, it is critical to consider such perceptual limitations in the design of better DAS. Here, I present a model (Fig. 17.1) that will be helpful in the design of better DAS. The primary contribution of the proposed model is in emphasizing the fact that an important aspect of collision avoidance lies in the ability of drivers to *effectively* use the available visual information. If the visual information is not effective,

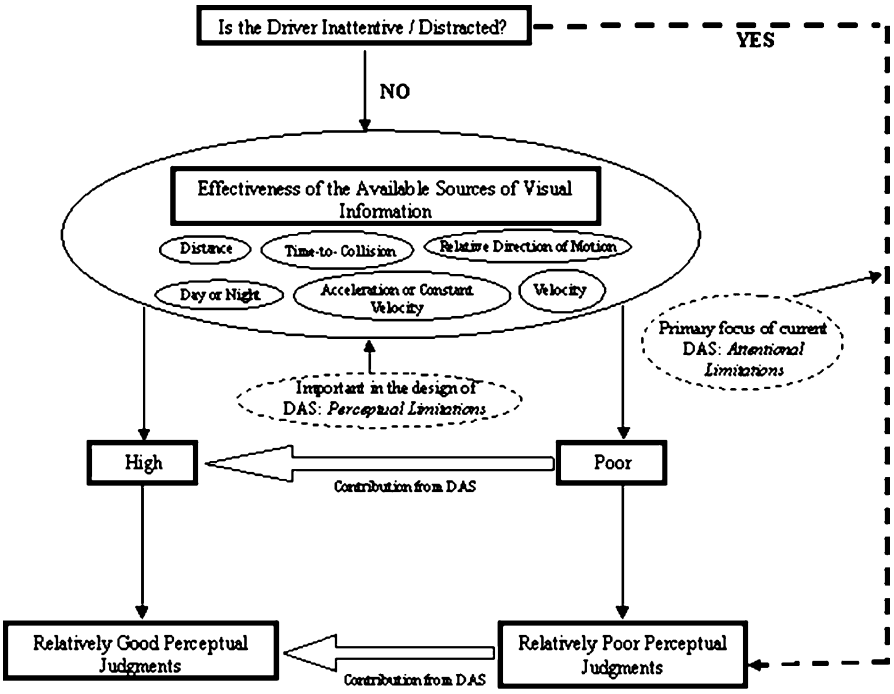



**Fig. 17.1** A model for designing driver-assistance systems to overcome perceptual limitations in drivers

allocating more attention will not improve driver performance. Therefore, DAS can contribute significantly toward enhancing the effectiveness of visual information which will lead to better perceptual judgments, reduce collisions, and increase driving safety.

## 17.7 Conclusions

The global data on overtaking accidents suggest that such traffic maneuvers require immediate attention. Additionally, the limitations in the human perceptual system suggest that drivers' judgments are not always accurate while performing such maneuvers. Therefore, it is important to design DAS that can aid the drivers during these complex maneuvers. Current FCAWS are typically designed for rear-end collisions. However, the visual information and environmental characteristics involved in a car-following task is different from an overtaking maneuver. In short, it is important to design DAS that can help drivers during overtaking maneuvers. Also, I have identified seven functional requirements that are important to consider in the design of such DAS. Finally, a model has been proposed for the

design of an effective DAS with emphasis on overcoming the limitations of the human perceptual system.

In conclusion, the proposed requirements take in to account human perceptual capabilities, limitations, and psychological factors associated with interacting with the automated DAS. Future research should address how such functional requirements affect the cognitive aspects of driver performance, i.e., situation awareness, human–automation interaction, and driver workload.

## References

1. Marchau V, Wiethoff M, Penttinen M, Molin E (2001) Stated preferences of European drivers regarding advanced driver assistance systems (ADA). Eur J Transport Infrastructure Res 1:291–308
2. National Highway Traffic Safety Administration (2006) Traffic safety facts 2004: a compilation of motor vehicle crash data from the fatality analysis reporting system and the general estimates system (NHTSA Publication No. DOT HS 809 919). U.S. Department of Transportation, Washington, DC
3. Brookhuis KA, De Ward D, Janssen WH (2001) Behavioral impacts of advanced driver assistance systems – an overview. Eur J Transport Infrastructure Res 1:245–253
4. Kiefer RJ, LeBlanc DJ, Flannagan CA (2005) Developing an inverse time-to-collision crash alert timing approach based on drivers' last second braking and steering judgments. Accid Anal Prev 37:295–303
5. Federal Highway Administration (2002) Safety effectiveness of intersection left-and-right-turn lanes. (FHWA Report No. FHWA-RD-02-089). McLean: U.S. Department of Transportation
6. Manser MP, Hancock PA (1996) Influence of approach angle of estimates of time-to-contact. Ecol Psychol 8:71–99
7. Carsten OMJ, Nilsson L (2001) Safety assessment of driver assistance systems. Eur J Transport Infrastructure Res 1:225–243
8. Fletcher L, Apostoloff N, Petersson L, Zelinsky A (2003) Vision in and out of vehicles. IEEE Intell Syst 8:12–17
9. Franke U, Gavrila D, Gern A, Goerzig S, Janssen R, Paetzold F, Whler C (2001) From door to door – principles and applications of computer vision for driver assistant systems. Intelligent vehicle technologies: theory and applications: Arnold
10. Gray R (2004) The use and misuse of visual information for "go/no-go" decisions in driving. In: Hennessy DA, Wiesenthal DL (eds) Contemporary issues in road user behavior. Nova Science, New York, pp 123–132
11. Gray R, Regan D (2005) Perceptual processes used by drivers during overtaking in a driving simulator. Hum Factors 47:394–417
12. Hegeman G (2004) Overtaking frequency and advanced driver assistance systems. In: IEEE intelligent vehicles symposium. Parma, Italy, pp 431–436
13. Clarke DD, Ward PJ, Jones J (1998) Overtaking road-accidents: differences in manoeuvre, as a function of driver age. Accid Anal Prev 30:455–467
14. National Highway Traffic Safety Administration (2001) Traffic safety facts 2000: a compilation of motor vehicle crash data from the fatality analysis reporting system and the general estimates system. U.S. Department of Transportation, Washington, DC
15. Gibson JJ, Crooks LE (1938) A theoretical field analysis of automobile driving. Am J Psychol 51:453–471

16. De Lucia PR, Tharanathan A (2005) Effects of optic flow and discrete warnings on deceleration detection during car-following. In: Proceedings of the 49th annual meeting of the human factors and ergonomics society. Human Factors and Ergonomics Society, Santa Monica
17. Schiff W, Detwiler ML (1979) Information used in judging impending collision. Perception 8:647–658
18. Tharanathan A, DeLucia PR (2007) Detecting deceleration of a lead car during active control of self motion: implications for rear-end collisions. In: Proceedings of the 51st annual meeting of the human factors and ergonomics society. Human Factors and Ergonomics Society, Santa Monica
19. Hills BL (1980) Vision, visibility and perception in driving. Perception 9:183–216
20. Jones HV, Heimstra NW (1964) Ability of drivers to make critical passing judgments. J Eng Psychol 3:117–122
21. Gordon DA, Mast TM (1970) Driver's judgments in overtaking and passing. Hum Factors 12:341–346
22. Hegeman G, Brookhuis K, Hoogendoorn S (2005) Opportunities for advanced driver assistance systems towards overtaking. Eur J Transport Infrastructure Res 5:281–296
23. Lee DN (1976) A theory of visual control of braking based on information about time-to-collision. Perception 5:437–459
24. De Lucia PR, Warren R (1994) Pictorial and motion-based depth information during active control of self-motion – size arrival effects on collision-avoidance. J Exp Psychol Hum Percept Perform 20:783–798
25. Hills BL (1975) Some studies of movement perception, age and accidents. Transport and road research laboratory: Technical report. Crowthorne, England
26. Hoffmann ER, Mortimer RG (1994) Drivers' estimates of time to collision. Accid Anal Prev 26:511–520
27. Cutting JE, Vishton PM (1995) Perceiving layout and knowing distances: the integration, relative potency, and contextual use of different information about depth. In: Epstein W, Rogers S (eds) Perception of space and motion. Academic Press, San Diego, pp 69–117
28. Benguigui N, Ripoll H, Broderick MP (2003) Time-to-contact estimation of accelerated stimuli is based on first-order-information. J Exp Psychol Hum Percept Perform 29:1083–1101
29. Tharanathan A (2009) Effects of constant and non-constant velocity motion on judgments of collision-avoidance action gap. In: Proceedings of the human factors and ergonomics society 53 rd annual meeting, Surface Transportation. San Antonio, Texas, USA, 53, pp 1762–1765
30. Tresilian JR (1991) Empirical and theoretical issues in the perception of time to contact. J Exp Psychol Hum Percept Perform 17:865–876
31. Tharanathan A, De Lucia PR (2006). Time-to-contact judgments of constant and non-constant velocities: implications for rear-end collisions. In: Proceedings of the 50th annual meeting of the human factors and ergonomics society. Human Factors and Ergonomics Society, Santa Monica
32. Gray R, Macuga KM, Regan D (2004) Long range interactions between object motion and self-motion in the perception of movement in depth. Vision Res 44:179–195
33. Gray R, Regan D (2000) Risky driving behavior: a consequence of motion adaptation for visually guided motor action. J Exp Psychol Hum Percept Perform 26:1721–1732
34. Shinar D, Rotenberg E, Cohen T (1997) Crash reduction with an advance brake warning system: a digital simulation. Hum Factors 39:296–302
35. Summala H, Lamble D, Laasko M (1998) Driving experience and perception of the lead car's braking when looking at in-car targets. Accid Anal Prev 30:401–407
36. Taieb-Maimon M (2007) Learning headway estimation in driving. Hum Factors 49:734–744
37. Li Z, Milgram P (2004) An empirical investigation of the influence of perception of time-to-collision on gap control in automobile driving. In: Proceedings of the 48th annual meeting of the human factors and ergonomics society. New Orleans, pp 2271–2275

38. Lorenz B, Parasuraman R (2007) Automated and interactive real-time systems. In: Durso F, Nickerson R, Dumais S, Lewandowsky S, Perfect T (eds) Handbook of applied cognition. Wiley, New York, pp 415–441
39. Miller R, Huang Q (2002) An adaptive peer-to-peer collision warning system. IEEE, VTC 1:317–321
40. National Highway Traffic Safety Administration (2000) NHTSA driver distraction expert working group meetings: summary and proceedings. NHTSA, Washington, DC
41. Parasuraman R, Sheridan TB, Wickens CD (2000) A model of types and levels of human interaction with automation. IEEE Trans Syst, Man Cybernetics – Part A: Syst Humans 30:286–297
42. Sanders MS, McCormick EJ (1993) Human factors in engineering and design, 7th edn. McGraw-Hill, New York
43. Strayer DL, Drews FA (2007) Attention. In: Durso F, Nickerson R, Dumais S, Lewandowsky S, Perfect T (eds) Handbook of applied cognition. Wiley, New York
44. Veeramallu SR (2000) Collision avoidance systems. In: MTC transportation scholars conference. Ames
45. Yang L, Yang JH, Feron E, Kulkarni V (2003) Development of a performance-based approach for a rear-end collision warning and avoidance system for automobiles. In IEEE Intelligent Vehicles Symposium, Columbus OH, pp 316–321
46. United States Department of Transportation (2005) Concept of operations and voluntary operational requirements for forward collision warning systems (CWS) and adaptive cruise control (ACC) systems on-board commercial motor vehicles. Washington, DC. Report Number FMCSA-MCRR-05-007
47. Dagan E, Mano O, Stein GP, Shashua A (2004) Forward collision warning with a single camera. In: IEEE intelligent vehicles symposium. Parma, pp 37–42
48. Gray R, Regan D (1998) Accuracy of estimating time to collision using binocular and monocular information. Vision Res 38:499–512
49. De Lucia PR, Novak JB (1997) Judgments of relative time-to-contact of more than two approaching objects: toward a method. Percept Psychophys 59:913–928
50. Kaiser MK, Johnson WW (2004) How now, broad tau? In: Hecht H, Savelsbergh GJP (eds) Advances in psychology 135, time-to-contact. Elsevier, San Diego, pp 287–302

# Chapter 18
# Advances in Multimodal Tracking of Driver Distraction

**Carlos Busso and Jinesh Jain**

**Abstract** This chapter discusses research efforts focused on tracking driver distraction using multimodal features. A car equipped with various sensors is used to collect a database with real driving conditions. During the recording, the drivers were asked to perform common secondary tasks such as operating a cell phone, talking to another passenger, and changing the radio stations. We analyzed the differences observed across multimodal features when the driver was engaged in these secondary tasks. The study considers features extracted from the controller area network bus (CAN-bus), a frontal camera facing the driver, and a microphone. These features are used to predict the distraction level of the drivers. The output of the proposed regression model has high correlation with human subjective evaluations ($\rho = 0.728$), which validates our approach.

**Keywords** Attention • CAN-bus data • Distraction • Driver behavior • Driver distraction • Head pose estimation • Multimodal feature analysis • Real traffic driving recording • Secondary task • Subjective evaluation of distraction

## 18.1 Introduction

New advances in sensing technologies and signal processing have opened interesting opportunities for in-vehicle systems that aim to increase road safety. One important direction is to monitor driver behaviors that can lead to car accidents. According to the *National Highway Traffic Safety Administration* (NHTSA), more than 25% of the police-reported accidents involved distracted drivers [1]. This fact is supported by the "100-Car Naturalistic Driving Study," which concluded that over 78% of crashes and 65% of near crashes were the result of inattentive drivers [2]. These statistics are

C. Busso (✉) • J. Jain
The University of Texas at Dallas, Richardson, TX 75080, USA
e-mail: busso@utdallas.edu; jjj081000@utdallas.edu

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_18, 

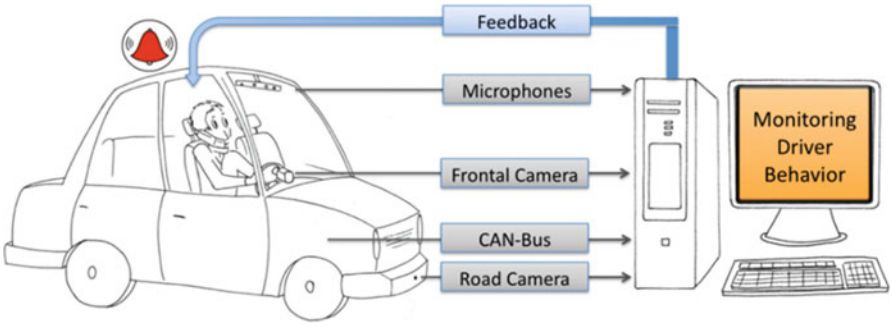



**Fig. 18.1** Monitoring driver behavior system using multimodal information

not surprising since 30% of the time when a car is moving, the driver is involved in secondary tasks that are potentially distracting [3]. With the development of new in-vehicle technologies, these numbers are expected to increase. Therefore, it is important to identify and develop feasible monitoring systems that are able to detect and warn inattentive drivers. These systems will play a crucial role in preventing accidents and increasing the overall safety on the roads.

Distraction can affect visual, cognitive, auditory, psychological, and physical capabilities of the driver. Distraction is defined by the Australian Road Safety Board as "the voluntary or involuntary diversion of attention from the primary driving tasks not related to impairment (from alcohol, drugs, fatigue, or a medical condition)" [4]. Under this well-accepted definition, the driver is involved in additional activities that are not related to the primary driving task, which include talking to a passenger, focusing on events or objects, and manipulating in-car technologies. As a result, the driver reduces his/her situational awareness, which affects his/her decision making, increasing the risk of crashes.

We have been working in detecting inattentive drivers by combining different modalities including *controller area network–bus* (CAN-bus) data, video cameras, and microphones [5, 6]. Our long term goal is to develop a multimodal framework that can quantify the attention level of the driver by using these noninvasive sensors (Fig. 18.1). Instead of relying on simulations, the study is based on recordings with actual drivers in real-world scenarios using the UTDriver platform – a car equipped with multiple sensors [7]. First, we have studied the changes observed in features across modalities when the driver is involved in common secondary tasks such as operating navigation systems, radio, and cell phone [5]. Then, we have proposed a regression model based on relevant multimodal features that can predict driver distraction. The results have shown that the outputs of the proposed system correlate with human subjective evaluations.

This chapter discusses the state-of-the-art in detecting inattentive drivers using multiple sensing technologies. It describes previous studies and our own contributions in the field. Notice that we only focus on distractions produced by secondary tasks. We do not include distractions or impairments produced by alcohol, fatigue, or drugs [8, 9].

The chapter is organized as follows: Section 18.2 gives a brief overview of previous work related to the study presented in this chapter. Section 18.3 describes the protocol used to collect the database. Section 18.4 presents subjective evaluations to quantify the perceived distractive behaviors. Section 18.5 reports our analysis of features extracted from the CAN-bus signal, a frontal camera, and a microphone. We study the changes in behaviors observed when the driver is engaged in secondary tasks. Section 18.6 demonstrates that the multimodal features can be used to recognize drivers engaged in secondary tasks, and to infer the distraction level of the drivers. Section 18.7 concludes the chapter with discussion and future directions.

## 18.2 Related Work

Several studies have attempted to detect inattentive drivers. These studies have proposed different sensing technologies including controller area network–bus (CAN-bus) data [7, 10, 11], video cameras facing the driver [12–14], microphones [10], and invasive sensors to capture biometric signals [8, 15, 16]. Some studies have analyzed data from real driving scenarios [7, 11, 17], while others have considered car simulators [16, 18, 19]. They also differ on the secondary tasks considered in the analysis. Bach et al. presented an exhaustive review of 100 papers that have considered the problem of understanding, measuring, and evaluating driver attention [20]. This section gives a brief overview of the current approaches to detect driver distractions.

### *18.2.1 Modalities*

Features derived from the vehicle such as speed, acceleration, and steering wheel angle are valuable in assessing driver behaviors [13, 18, 19, 21–23]. Relevant information can be extracted from CAN-bus data. Sathyanarayana et al. used CAN-bus signals to model driver behaviors [21]. They extracted steering wheel angle and gas and brake pedal pressures. The proposed information was used to detect driving maneuvers such as turns, stops, and lane changes. After maneuver recognition, they recognized distraction using driver-dependent *Gaussian mixture model-universal background model* (GMM-UBM). Unfortunately, accessing the CAN-bus information is not always possible, since the car manufactures protect this information. Accessing car information is easier in studies that use car simulators. These interfaces usually provide detailed information about the car. For example, Tango and Botta used features such as the steering angle, lateral position, lateral acceleration, and speed of the host vehicle to predict the reaction time of the drivers [19]. Along with other features, Liang et al. used the steering wheel position, steering error, and lane position to assess cognitive distraction [13]. Ersal et al. built a

radial-basis neural network model to characterize normal driving behavior [18]. The proposed system used features derived from the pedal position. They used this normal model to identify variations on the behaviors displayed when the driver was engaged in secondary tasks. Likewise, Yang et al. used a GPS signal to approximate information that can be extracted from the CAN-bus signal such as the velocity and steering wheel angle. They use computer simulations for the study [23].

Cameras have been used to detect and tract inattentive drivers. Several studies have attempted to infer the head pose and/or eyelid movements of the driver from frontal cameras [11, 13, 14, 17]. Liang et al. showed that eye movements and driving performance measures (e.g., steering wheel position, lane position) were useful to detect cognitive distraction [13]. They proposed a classifier trained with *support vector machine* (SVM), achieving an average accuracy of 81.1%. Su et al. presented a simple approach to monitor driver inattention using eyelid movements and facial orientation [14]. They used a low-cost CCD camera mounted in the car dashboard. Bergasa et al. also considered eyelid movements and head pose to detect fatigue [17]. They estimated the *percent eye closure* (PERCLOS), eye closure duration, blink frequency, face position, fixed gaze, and nodding frequency, which were fused with fuzzy classifiers. In addition to head rotations, Kutila et al. used the gaze of the driver and lane tracking data to detect visual and cognitive workload. [11]. They used a stereo camera system to estimate the head and gaze features. An important challenge in this field is processing video in real driving conditions due to lighting variations. Fortunately, advances in computer vision have led to robust tracking algorithms that are effective and suitable for in-vehicle systems [24]. Furthermore, Bergasa et al. showed that IR illuminator can be used to reduce changes in illumination [17].

Other studies have considered physiological signals, which are correlated with the driver workload, attention, and fatigue [8, 15, 16]. Among all physiological signals, *electroencephalography* (EEG) is the predominant and most used modality [20]. Putze et al. used multiple biometric signals such as *skin conductance* (SC), *photoplethysmography* (PPG), respiration and EEG [16]. Damousis and Tzovaras proposed a fuzzy fusion system to detect alert versus drowsy drivers using electrooculogram (EOG) [15]. They used this signal to infer different eyelid activity indicators that were used as features. Lin et al. used EEG to detect drowsiness [8]. Sathyanarayana et al. extracted CAN-bus information along with body sensors (accelerometer and gyroscope) to detect driver distraction. They attached body sensors to the driver's head and legs. The drivers were also recorded with cameras, which were used to manually and automatically segment the corpus into normal and task conditions. They reported accuracies above 90% for the detection of distraction with k-NN classifiers.

### *18.2.2 Inducing Visual and Cognitive Distractions*

Different approaches have been used to induce visual and cognitive distractions. They aim to increase the driver workload, affecting the primary driving task. Therefore, the recordings can include samples of inattentive behaviors.

The most common approaches to induce cognitive distractions include solving math problems [11, 16, 25], talking to another passenger [11, 21], and paying attention to cognitive activities (e.g., follow stock market) [13]. For visual distractions, common techniques are "look and find" tasks [13, 16, 19], operating devices (e.g., touch screen, cell phone, GPS) [6, 18], and reading numbers [11]. In our work, we are interested in analyzing behaviors when the driver is engaged in secondary tasks that are commonly performed in real driving scenarios.

### *18.2.3 Driving Platforms*

Bach et al. reported that most of the studies in this research area have considered recording from car simulators (51% of the studies considered in their review) [20]. In many cases, using simulators is the only feasible and secure approach. For example, studies that aim to detect fatigue are usually conducted in laboratory setup [8, 15]. As an exception, Bergasa et al. studied fatigue in real driving condition. However, the subjects were asked to simulate drowsy behaviors [17]. Likewise, car simulators are normally used when physiological signals are used [8, 15, 16]. Some of these signals are difficult to collect in real car. Also, the invasive nature of the sensors makes this approach less suitable for real-world driving scenarios.

Some studies have used recording from cars in real roads [11, 17, 21, 22]. For this purpose, different car platforms have been designed to collect driver behaviors with data acquisition systems consisting of invasive and noninvasive sensors in real-world driving scenarios. Examples include Argos [26], UYANIK [27], and UTDrive [7, 10] (this study uses the UTDrive platform). These cars will provide more realistic data to study driver behaviors.

## 18.3 Recording Drivers in Real Road Conditions

The scope of our work is to use multimodal sensors to track driver behaviors during real driving conditions. This requirement implies that the study cannot rely on recordings based on driving simulations. Therefore, we recorded subjects driving the UTDrive platform on real roads near the main campus of *the University of Texas at Dallas* (UTD). The UTDrive car is a 2006 Toyota RAV4 equipped with data acquisition systems with multiple sensors including camera and microphone arrays (Fig. 18.2a). The system records the CAN-bus data, which provides relevant car information such as the brake, gas, acceleration, vehicle speed, and steering wheel angle. A frontal camera (PBC-700H) facing the driver was placed on the dashboard behind the steering wheel (Fig. 18.2b), which records 30fps at a resolution of $320 \times 240$. It provides valuable information about facial expressions and head orientation of the driver. There is also a camera facing the road ahead, which records 15fps at a resolution of $320 \times 240$. Although we have not used this camera

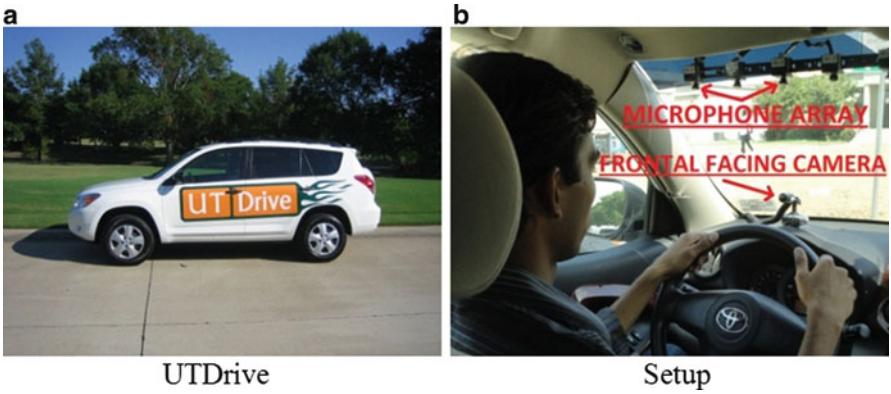



**Fig. 18.2** UTDrive car and sensors setup. (**a**) UTDrive, (**b**) Setup

in our work, it provides important information that can be used for lane tracking [11, 28]. The information is simultaneously stored in a Dewetron computer. In addition, a *global positioning system* (GPS) is placed at the center of the front windshield for the experiments. The database was collected in dry days with good light conditions to reduce the impact of environment in the study. The readers are referred to [7] for further details about the UTDrive project.

In our study, we are interested in analyzing the observed behaviors when the driver is involved in secondary tasks. We decided to include common activities such as changing radio stations, talking by cell phone, operating a GPS, and having spontaneous conversation with a passenger. There are other common secondary tasks such as texting, eating, drinking, grooming, and smoking that were not included for security reasons.

A multimodal database was collected from 20 subjects consisting of students and employees of the university. They were asked to drive the UTDriver car following a predefined, 5.6-mile route, described in Fig. 18.3. This route includes many traffic lights, stop signs, heavy and low traffic zones, residential areas, and a school zone. Each subject completed this route twice (12–16 minutes per lap).

In the first lap, drivers are asked to sequentially perform common tasks as described in Fig. 18.3. The first task corresponds to change the built-in car radio to targeted stations (route A in Fig. 18.3). The second task requires the driver to input a specific predecided address into the GPS and then follow the instructions to the desired destination (route B in Fig. 18.3). Preliminary analysis on the data indicated that driver behaviors during operating and following the GPS were different. Therefore, we subdivided this task in two. Then, the subject is asked to make a phone call from his/her cell phone to obtain flight information between two US cities (route C in Fig. 18.3). Due to similar reasons observed in the GPS task, we subdivided this task in operating and talking on the cell phone. Notice that at the time of the recording, the State of Texas allowed drivers to use cell phones. After that, a passenger shows


**Fig. 18.3** Route used for the recording (5.6 miles long). Subjects drove this route two times. In the first lap, the subjects performed the tasks in order, starting with the *Radio* task and ending with the *Conversation* task. In the second lap, the subjects drove the same route without performing any task. The second lap involves normal driving without any of the aforementioned tasks, which is used as a normal reference. Since the same route is used for both normal and task conditions, the analysis is less dependent on the selected road

randomly selected pictures and the driver is asked to describe them (route D in Fig. 18.3). The purpose of this task is to collect approximated (and maybe exaggerated) data from distractions caused by billboards, sign boards, and shops. The last task corresponds to spontaneous conversation between the driver and a passenger, who asked few general questions (route E in Fig. 18.3). After subdividing the phone and GPS tasks, the database includes the following seven tasks: *Radio*, *GPS Operating*, *GPS Following*, *Phone Operating*, *Phone Talking*, *Picture*, and *Conversation*.

## 18.4 Assessing Driver Distraction

After collecting the database, the first research question is to assess the level of distraction induced on the drivers by the selected secondary tasks [6]. Defining the ground truth for driver distraction is a crucial problem for training and testing systems that aim to identify inattentive drivers. However, this is a nontrivial task, since different in-cab activities will create specific demands on the drivers causing visual, cognitive, auditory, psychological, and/or physical distractions.

As a first approximation, we conducted perceptual evaluations to assess driver distraction. A *graphical user interface* (GUI) was created for this subjective evaluation (Fig. 18.4). This GUI allows the evaluators to watch videos extracted from the frontal camera. They can evaluate the perceived distraction level of the driver on a scale from 1 (*less distracted)* to 5 (*more distracted)*. Notice that evaluators should be able to identify visual distractions with high accuracy. However, they may not be able to assess more challenging type of distractions. Quantifying cognitive or psychological distractions remain an open challenge.

The database contains over 7 h of data. However, we decided to assess only a portion of this corpus to reduce the time and resources needed for the evaluation. The corpus was automatically split into 5-s videos. For each of the driver, three videos were randomly selected per task. In addition, we include three videos of the drivers under normal conditions. Therefore, we selected 480 5-s videos for the

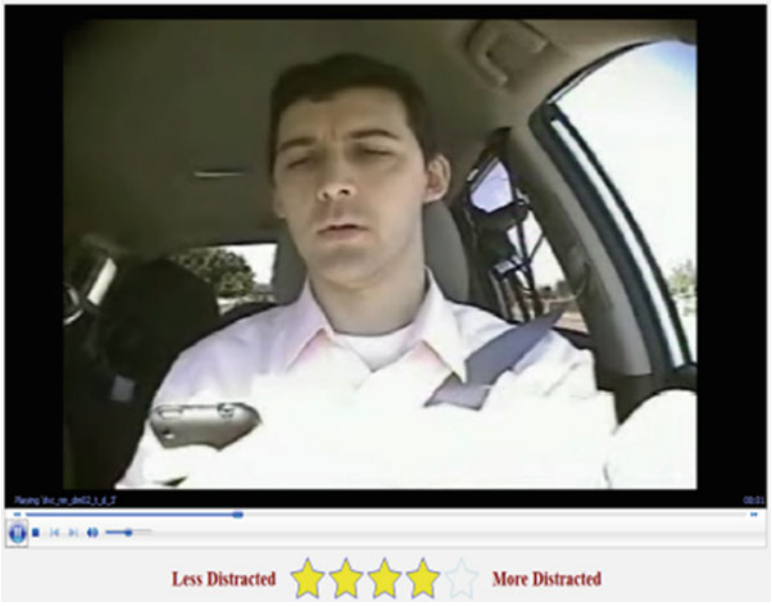



**Fig. 18.4** Subjective evaluation GUI. The subjects are required to rate the video based on how distracted they feel the driver is (1 for *less distraction*, 5 for *more distraction*)

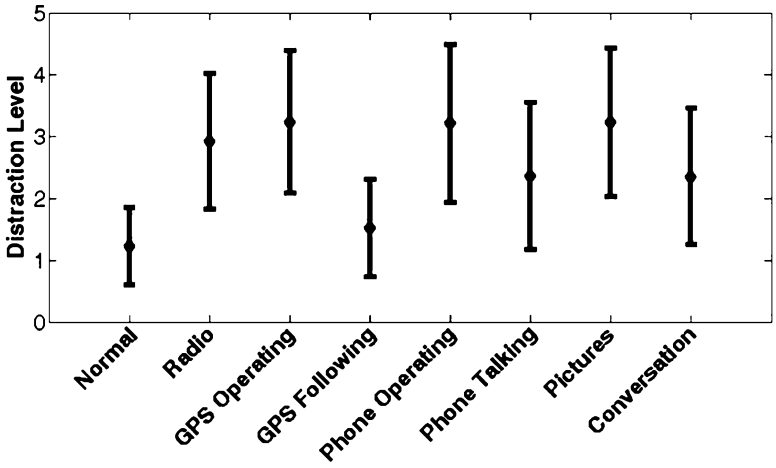



**Fig. 18.5** Perceived distraction levels based on subjective evaluations. The figure shows the means and standard deviations for each task across drivers and evaluators

evaluation (3 videos $\times$ 8 conditions $\times$ 20 drivers $=$ 480). Nine students participated in the subjective experiment. They assessed only 160 videos (1 video $\times$ 8 conditions $\times$ 20 drivers $=$ 160). We read the adopted definition of distraction (Sect. 18.1) before the evaluation to unify their understanding of distraction. The presentation of the videos was randomized to avoid biases. With this setup, each video was rated by three independent evaluators.

Figure 18.5 gives the means and standard deviations of the perceived distraction level across secondary tasks. The results indicate that the tasks *GPS Following* and *Phone Talking* are not perceived as distracting as the tasks *GPS Operating* and *Phone Operating*, respectively. Notice that talking on a cell phone increases the cognitive load of the drivers. Studies have reported that the use of cell phone affect the driver performance (e.g., missing traffic lights, fail to recognize billboard) [20, 29]. This subjective evaluation does not seem to capture this type of distraction. Likewise, *Radio* and *Picture* are perceived as distractive tasks.

## 18.5 Analysis of Multimodal Features

Our next research problem is to identify multimodal features that can characterize inattentive drivers. As mentioned in Sect. 18.3, the corpus includes CAN-bus information, videos, and audio. Identifying informative features from these noninvasive modalities is an important step toward detecting distracted drivers.

**CAN-Bus**: One important source of information is provided by the CAN-bus, which includes steering wheel angle, brake value, vehicle speed, and acceleration. The car has also sensors to measure and record the brake and gas pedal pressures.

From these continuous streams of data, we estimate the derivative of the brake and gas pedal information. In addition, we estimate the jitter in the steering wheel angle, since we expect that drivers involved in secondary tasks will produce more "jittery" behaviors. Vehicle speed is also considered, since it is hypothesized that drivers tend to reduce the speed of the car when they are engaged in a secondary task.

**Frontal Video Camera**: The camera captures frontal views of the drivers. From this modality, we estimate head orientation and eye closure count. The head pose is described by the yaw and pitch angles. Head roll movement is hypothesized to be less important, given the considered secondary tasks. Therefore it is not included in the analysis. Likewise, eye closure percentage is defined as the percentage of frames in which the eyelids are lowered below a given threshold. This threshold is set at the point where the eyes are looking straight at the frontal camera. These variables are automatically extracted with the AFECT software [30]. Previous studies have shown that this toolkit is robust against large datasets and different illumination conditions. Another advantage of this toolkit is that the information is independently estimated frame by frame. Therefore, the errors do not propagate across frames. Unfortunately, some information is lost when the head is rotated beyond a certain degree or when the face is occluded by the driver's hands. The algorithm produces empty data in those cases.

**Microphone Array**: The acoustic information is a relevant modality for secondary tasks characterized by sound or voice activity such as *GPS Following*, *Phone Talking*, *Pictures*, and *Conversation*. Here, we estimate the average audio energy from the microphone that is closest to the driver.

The proposed monitoring system segments the data into small windows (e.g., 5 s), from which it extracts relevant features. We estimate the mean and standard deviation of each of the aforementioned data, which are used as features. Details of other preprocessing steps are described in Jain and Busso [5].

After the multimodal features are estimated, we compare their values under task and normal conditions. Notice that segments of the road have different speed limits and number of turns. Therefore, the features observed when the driver was engaged in one task (first lap – Sect. 18.3) are only compared with the data collected when the driver was not performing any task over the same route segment (second lap – Sect. 18.3). This approach reduces the variability introduced by the route.

We conducted a statistical analysis to identify features that change their values when the driver is engaged in secondary tasks. A matched pair hypothesis test is used to assess whether the differences in the features between each task and the corresponding normal condition are significant. We used matched pairs instead of independent sample, because we want to compensate for potential driver variability. For each feature $f$, we have the following hypothesis test [31]:

$$\begin{aligned} H_0 &: \quad \mu^f_{normal} - \mu^f_{task} = 0 \\ H_1 &: \quad \mu^f_{normal} - \mu^f_{task} \neq 0 \end{aligned} \tag{18.1}$$

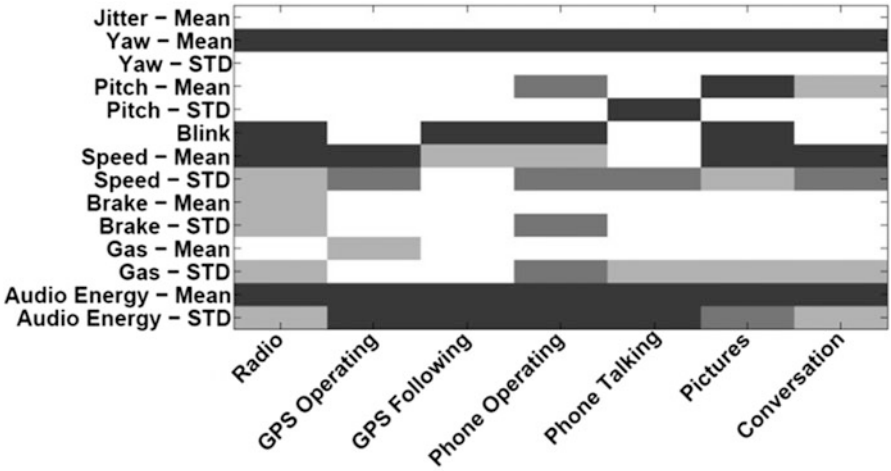



**Fig. 18.6** Results of the matched pairs *t*-test: features vs. tasks. For a particular task, gray regions indicate the features that are found to have significant differences (*dark gray*, $p$-value = 0.05; *gray*, $p$-value = 0.10; *light gray*, $p$-value = 0.20)

where $\mu^f_{normal}$ and $\mu^f_{task}$ are the means of $f$ in normal and task conditions, across speakers. Since the database consists of 20 drivers, we use a t-test for small sample,

$$t^f = \frac{\bar{d}^f}{s^f_d/\sqrt{n}} \tag{18.2}$$

where $\bar{d}^f$ and $s^f_d$ represent the mean and standard deviation of the sample of differences across the $n$ (=20) drivers. Figure 18.6 shows the features that are found significant at $p$-value = 0.05 (dark gray), $p$-value = 0.10 (gray), and $p$-value = 0.20 (light gray). The figure shows that mean of the energy and head yaw movement are significantly different for all the tasks ($p$-value = 0.05). Eye closure percentage (blink) is also significantly different for tasks such as *Radio*, *GPS Following*, *Phone Operating*, and *Pictures*. The figure also shows that there are tasks such as *GPS Following*, *Phone Talking*, and *Conversation* in which few of the selected features present significant differences at $p$-value = 0.05. Interestingly, these tasks are perceived less distracting than other (see Fig. 18.5). Notice that for different tasks, there are significant features across all the modalities considered in this study (CAN-bus, video, and microphone).

While Fig. 18.6 identifies features that are significantly different under normal and task conditions, it does not show the characteristic patterns for each condition. Therefore, we estimated the mean and standard deviation of the features for each of the secondary task, across speakers. We also estimated these statistics for features observed in normal condition over the corresponding route segments.

Figure 18.7 shows the errors plot for (a) head yaw movement, (b) head pitch movement, (c) vehicle speed, and (d) steering wheel jitter. Figure 18.7a reveals that

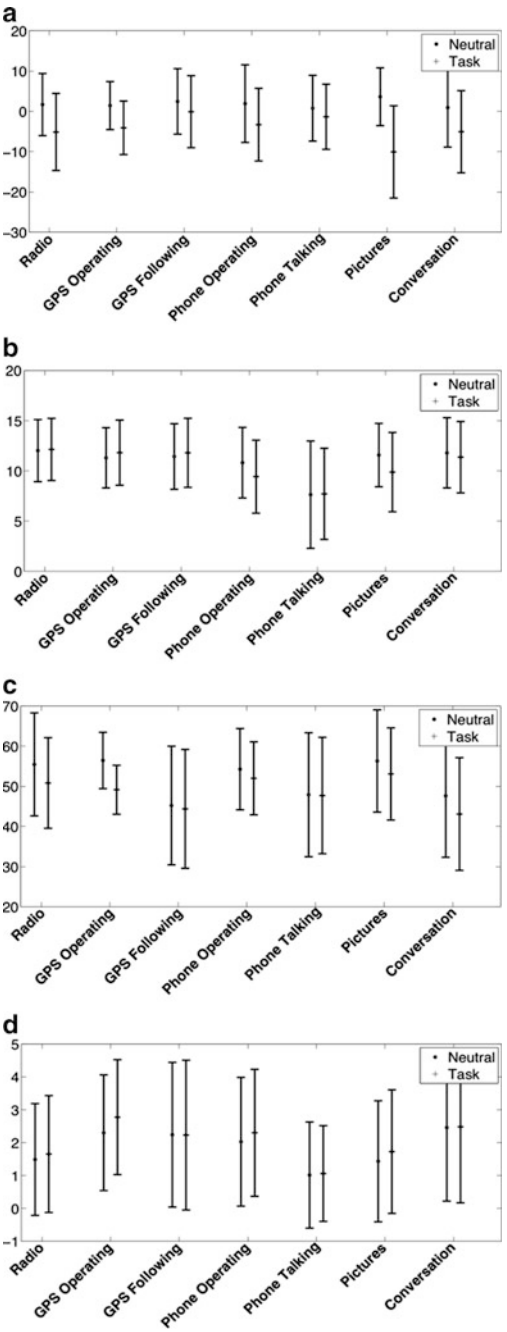



**Fig. 18.7** Error-bar plots displaying the mean and standard deviation for: (**a**) head yaw, (**b**) head pitch, (**c**) vehicle speed, and (**d**) jitter in the steering wheel

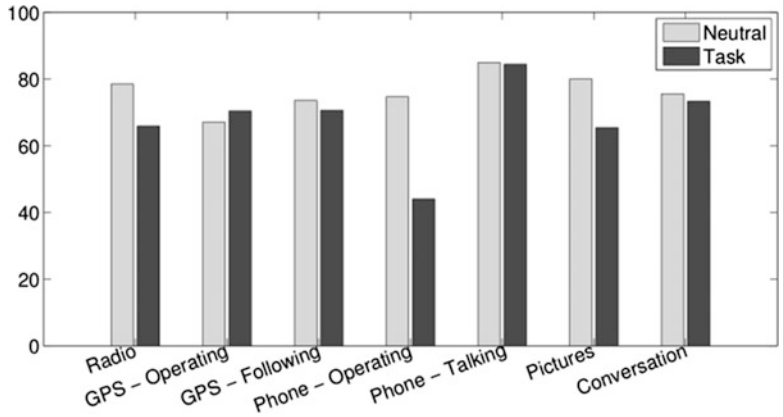



**Fig. 18.8** Percentage of eye closure in task and normal conditions. The values for normal conditions are estimated over the features observed in the corresponding route of the tasks

drivers tend to look at their right while performing the tasks. Changes in head pitch movements are more evident in tasks such as *Phone Operating* and *Picture* (Fig. 18.7b). In those cases, drivers tend to look down. Figure 18.7c shows that drivers reduce the car speed when they are engaged in secondary tasks. This result is consistently observed across task. Figure 18.7d shows that the jitter in the steering wheel is slightly higher in *GPS Operating* and *Phone Operating*. However, these differences are not significant (see Fig. 18.6). Figure 18.7 also shows differences in the features during normal conditions across tasks. These differences are inherently dependent on the road. This result suggests that the characteristic of the route is an important variable that should be considered in the design of automatic feedback systems [21]. Figure 18.8 shows the percentage of eye closure for the normal and task conditions. It can be seen that the closure rates differ from the patterns observed during normal condition for tasks such as *Radio*, *Phone Operating*, and *Pictures*. For these tasks, the drivers tend to keep their eyelid more open.

Figure 18.9 provides further information about the differences observed in head yaw movements for the tasks (a) *Radio* and (b) *Conversation*. The figure provides the feature distributions for normal and task conditions. Notice that these are among the most common secondary tasks performed by drivers. The figure shows that both distributions present positive skewness for the task conditions. The implication is that drivers shift their attention from the road, which may affect their situational awareness.

## 18.6 Prediction of Driver Distractions

After studying relevant features that can signal driver distraction, this section explores whether the proposed multimodal features can be used to detect driver distractions. The study includes two evaluations. First, we train a classifier to

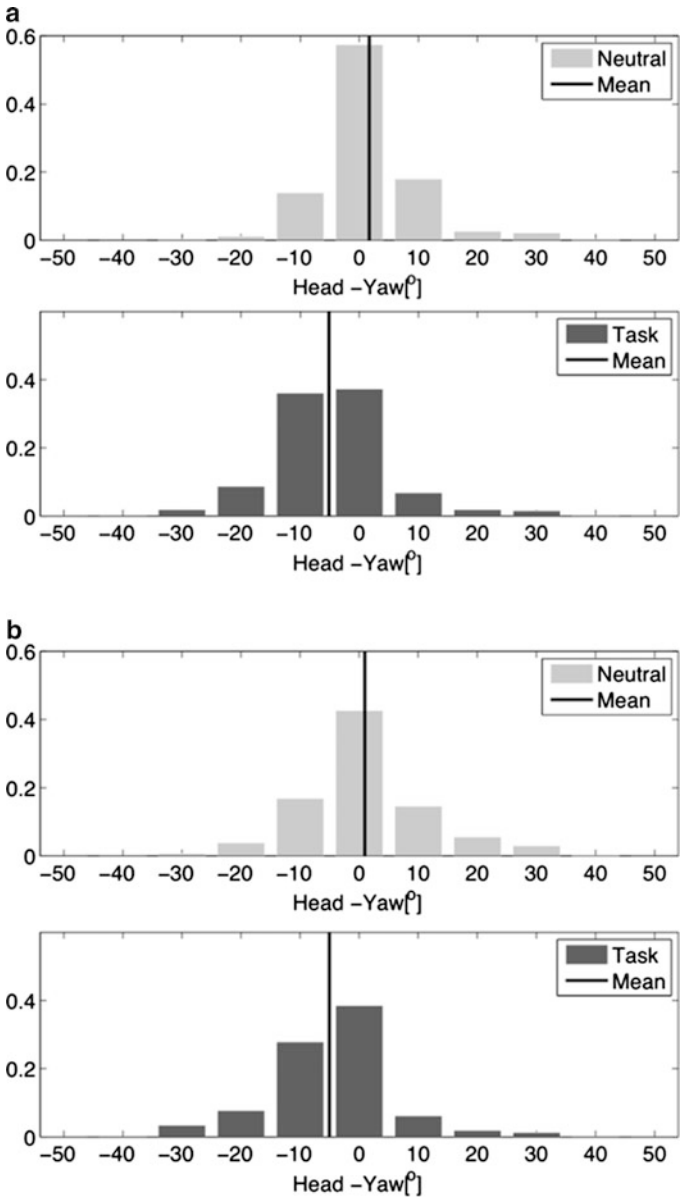



**Fig. 18.9** Distribution of head yaw movements for the tasks (**a**) *Radio* and (**b**) *Conversation*. The distributions are estimated from data recorded during task (*dark gray*) and normal (*light gray*) conditions. The vertical lines represent the corresponding means

**Table 18.1** Accuracies for the multi-class k-NN classifier for different values of $k$

| k | 4 | 8 | 12 | 16 | 20 |
|---|---|---|---|---|---|
| Accuracy | 0.3582 | 0.3958 | 0.4021 | 0.4218 | 0.4272 |

recognize whether the driver is performing any of the secondary tasks (Sect. 6.1). Then, we build a regression model that aims to predict the level of distraction of the driver (Sect. 6.2).

### *18.6.1 Classification of Secondary Tasks*

The perceptual evaluation in Sect. 18.4 shows that some secondary tasks are perceived more distracting than others. This result suggests that recognizing a driver engaged in secondary tasks is an important problem. We have proposed binary classifiers to distinguish between individual tasks and normal conditions [5]. Here, we are interested in the multi-class problem of recognizing between the seven tasks and normal conditions (eight-class problem). We argue that this is a more practical approach that can lead to useful applications.

For this evaluation, we trained a k-nearest neighbor classifier. The database is split in 5-s windows which are considered as independent samples. The task labels are assigned to the samples according to the task in the corresponding route segment. To ensure driver-independent results, samples from one speaker are included either on the training or testing sets. This was implemented using a "leave-one-driver-out" cross validation scheme. Table 18.1 gives the average accuracies across folds for different values of $k$. The best performance is obtained for $k = 20$, which gives an accuracy of 42.72%. This accuracy is significantly higher than chances (12.5%).

### *18.6.2 Regression Model for Driver Distraction*

The second evaluation consists in building a regression model to predict the distraction level of the driver. The baseline for this experiment is the average of the subjective evaluation results presented in Sect. 18.4. Only the samples that were perceptually evaluated are considered in this analysis (480, 5-s segments with balanced number of samples per task and driver). The multimodal features are included as dependent variables. The proposed model includes interaction and quadratic terms, because they improved the performance. The coefficient of determination for this model is $R^2 = 0.53$, which corresponds to a correlation of $\rho = 0.728$. This result shows that the proposed features can be used to predict the distraction level of the driver.

## 18.7 Discussion and Conclusions

Any distraction can affect the situational awareness of the driver leading to disastrous consequences. This chapter summarized our current research effort in detecting distracted drivers using multiple modalities. The study was based on a database collected with real driving conditions, in which 20 drivers were asked to perform common secondary tasks. Our analysis identified features extracted from the CAN-bus data, a camera, and a microphone that present characteristic differences during task conditions. Our results show that these multimodal features can be used to predict the distraction level of the drivers. The proposed metric estimated from a regression model was highly correlated with human subjective evaluations ($\rho = 0.728$).

One weakness in our approach is the ground truth for the distraction level, which is derived from subjective evaluations. The proposed perceptual evaluation may not capture cognitive and psychological distractions. Indicators for these types of distraction may be derived by conducting workload ratings (e.g., SWAT, NASA-TLX) [20], by measuring brain activity with invasive sensors [32], or by indirectly inferring the underlying cognitive state of the driver (i.e., emotions, stress). In this area, we are working toward detecting the emotional state of the drivers, especially for negative reactions.

Another area of interest is expanding the set of features. For example, advances in the field of computer vision can lead to algorithms to directly detect distractive objects (e.g., cell phone) or actions (e.g., eating). The outputs of these algorithms can be used as discrete variables in the proposed regression models.

The work presented in this chapter represents our first step toward obtaining a metric to determine the attention level of the drivers. A real-time algorithm with such capability will facilitate the design of a feedback system to alert inattentive drivers, preventing potential accidents, and therefore, improving the overall driver experience.

## References

1. TA Ranney, WR Garrott, MJ Goodman (2001) NHTSA driver distraction research: past, present, and future. Technical Report Paper No. 2001-06-0177, National Highway Traffic Safety Administration, June 2001
2. V Neale, T Dingus, S Klauer, J Sudweeks, M Goodman (2005) An overview of the 100-car naturalistic study and findings. Technical Report Paper No. 05-0400, National Highway Traffic Safety Administration, June 2005
3. TA Ranney (2008) Driver distraction: a review of the current state-of-knowledge. Technical Report DOT HS 810 787, National Highway Traffic Safety Administration, April 2008
4. I Trezise, EG Stoney, B Bishop, J Eren, A Harkness, C Langdon, T Mulder (2006) Inquiry into driver distraction: report of the road safety committee on the inquiry into driver distraction. Technical Report No. 209 Session 2003–2006, Melbourne: Road Safety Committee, Parliament of Victoria, August 2006

5. J Jain, C Busso (2011) Analysis of driver behaviors during common tasks using frontal video camera and CAN-Bus information. In: IEEE international conference on multimedia and expo (ICME 2011), Barcelona, July 2011
6. JJ Jain, C Busso (2011) Assessment of driver's distraction using perceptual evaluations, self assessments and multimodal feature analysis. In: 5th Biennial workshop on DSP for in-vehicle systems, Kiel, September 2011
7. P Angkititrakul, D Kwak, S Choi, J Kim, A Phucphan, A Sathyanarayana, JHL Hansen (2007) Getting start with UTDrive: driver-behavior modeling and assessment of distraction for in-vehicle speech systems. In: Interspeech 2007, Antwerp, August 2007, pp 1334–1337
8. Lin C-T, Wu R-C, Liang S-F, Chao W-H, Chen Y-J, Jung T-P (2005) EEG-based drowsiness estimation for safety driving using independent component analysis. IEEE Trans Circ Syst I: Regular Papers 52(12):2726–2738
9. T Rahman, S Mariooryad, S Keshavamurthy, G Liu, JHL Hansen, C Busso (2011) Detecting sleepiness by fusing classifiers trained with novel acoustic features. In: 12th annual conference of the international speech communication association (Interspeech'2011), Florence, August 2011
10. P Angkititrakul, M Petracca, A Sathyanarayana, JHL Hansen (2007) UTDrive: driver behavior and speech interactive systems for in-vehicle environments. In: IEEE intelligent vehicles symposium, Istanbul, June 2007, pp 566–569
11. M Kutila, M Jokela, G Markkula, MR Rue (2007) Driver distraction detection with a camera vision system. In: IEEE international conference on image processing (ICIP 2007), vol 6. San Antonio, September 2007, pp 201–204
12. Y Dong, Z Hu, K Uchimura, N Murayama (2009) Driver inattention monitoring system for intelligent vehicles: a review. In: IEEE intelligent vehicles symposium, Xi'an, June 2009, pp 875–880
13. Liang Y, Reyes ML, Lee JD (2007) Real-time detection of driver cognitive distraction using support vector machines. IEEE Trans Intell Transport Syst 8(2):340–350
14. MC Su, CY Hsiung, DY Huang (2006) A simple approach to implementing a system for monitoring driver inattention. In: IEEE international conference on systems, man and cybernetics (SMC 2006), vol 1. Taipei, October 2006, pp 429–433
15. Damousis IG, Tzovaras D (2008) Fuzzy fusion of eyelid activity indicators for hypovigilance-related accident prediction. IEEE Trans Intell Transportation Syst 9(3):491–500
16. F Putze, J-P Jarvis, T Schultz (2010) Multimodal recognition of cognitive workload for multitasking in the car. In: International conference on pattern recognition (ICPR 2010), Istanbul, August 2010
17. Bergasa LM, Nuevo J, Sotelo MA, Barea R, Lopez ME (2006) Real-time system for monitoring driver vigilance. IEEE Trans Intell Transport Syst 7(1):63–77
18. Ersal T, Fuller HJA, Tsimhoni O, Stein JL, Fathy HK (2010) Model-based analysis and classification of driver distraction under secondary tasks. IEEE Trans Intell Transport Syst 11(3):692–701
19. Tango F, Botta M (2009) Evaluation of distraction in a driver-vehicle-environment framework: an application of different data-mining techniques. In: Perner P (ed) Advances in data mining. Applications and theoretical aspects, volume 5633 of lecture notes in computer science. Springer, Berlin/Heidelberg, pp 176–190
20. KM Bach, MG Jaeger, MB Skov, NG Thomassen (2009) Interacting with in-vehicle systems: understanding, measuring, and evaluating attention. In: Proceedings of the 23rd British HCI Group annual conference on people and computers: celebrating people and technology, Cambridge, UK, September 2009
21. A Sathyanarayana, P Boyraz, Z Purohit, R Lubag, JHL Hansen (2010) Driver adaptive and context aware active safety systems using CAN-bus signals. In: IEEE intelligent vehicles symposium (IV 2010), San Diego, June 2010

22. A Sathyanarayana, S Nageswaren, H Ghasemzadeh, R Jafari, JHL Hansen (2008) Body sensor networks for driver distraction identification. In: IEEE international conference on vehicular electronics and safety (ICVES 2008), Columbus, September 2008
23. J Yang, TN Chang, E Hou (2010) Driver distraction detection for vehicular monitoring. In: Annual conference of the IEEE industrial electronics society (IECON 2010), Glendale, November 2010
24. Murphy-Chutorian E, Trivedi MM (2010) Head pose estimation and augmented reality tracking: an integrated system and evaluation for monitoring driver awareness. IEEE Trans Intell Transport Syst 11(2):300–311
25. Harbluk JL, Noy YI, Trbovich PL, Eizenman M (2007) An on-road assessment of cognitive distraction: impacts on drivers' visual behavior and braking performance. Accid Anal Prev 39(2):372–379
26. Perez A, Garcia MI, Nieto M, Pedraza JL, Rodriguez S, Zamorano J (2010) Argos: an advanced in-vehicle data recorder on a massively sensorized vehicle for car driver behavior experimentation. IEEE Trans Intell Transport Syst 11(2):463–473
27. Abut H, Erdoğan H, Ercil A, Curuklu B, Koman HC, Taş F, Argunşah AO, Coşar S, Akan B, Karabalkan H et al (2009) Real-world data collection with "UYANIK". In: Takeda K, Erdoğan H, Hansen JHL, Abut H (eds) In-vehicle corpus and signal processing for driver behavior. Springer, New York, pp 23–43
28. P Boyraz, X Yang, JHL Hansen (2009) Computer vision applications for context-aware intelligent vehicles. In: 4th Biennial workshop on DSP for in-vehicle systems and safety, Dallas, June 2009
29. DL Strayer, JM Cooper, FA Drews (2004) What do drivers fail to see when conversing on a cell phone? In: Proceedings of human factors and ergonomics society annual meeting, vol 48. New Orleans, September 2004
30. MS Bartlett, GC Littlewort, MG Frank, C Lainscsek, I Fasel, JR Movellan (2006) Automatic recognition of facial actions in spontaneous expressions. Journal of Multimedia 6(1):22–35, September 2006
31. Mendenhall W, Sincich T (2006) Statistics for engineering and the sciences. Prentice-Hall, Upper Saddle River
32. Berka C, Levendowski DJ, Lumicao MN, Yau A, Davis G, Zivkovic VT, Olmstead RE, Tremoulet PD, Craven PL (2007) EEG correlates of task engagement and mental workload in vigilance, learning, and memory tasks. Aviat Space Environ Med 78(5):231–244

# Chapter 19
# A Stochastic Approach for Modeling Lane-Change Trajectories

**Yoshihiro Nishiwaki, Chiyomi Miyajima, Norihide Kitaoka, and Kazuya Takeda**

**Abstract** A signal-processing approach for modeling vehicle trajectory during lane changes while driving is discussed. Since individual driving habits are not a deterministic process, we develop a stochastic method to model them. The proposed model consists of two parts: a dynamic system represented by a hidden Markov model and a cognitive distance space represented with a hazard-map function. The first part models the local dynamics of vehicular movements and generates a set of probable trajectories. The second part selects an optimal trajectory by stochastically evaluating the distances from surrounding vehicles. Through experimental evaluation, we show that the model can predict vehicle trajectory in given traffic conditions with a prediction error of 17.6 m.

**Keywords** Driving behavior • Generation • Hazard map • Hidden Markov model (HMM) • Lane change • Prediction • Sampling • Stochastic modeling

## 19.1 Introduction

Driving safety and fuel-efficient driving are central issues in modern societies. Even though the traffic accident fatality rate has dropped significantly in Japan, traffic accidents were still responsible for approximately 5,000 fatalities in 2010 [1]. Energy and environmental problems are also serious threats to modern society. Technologies such as precrash safety and hybrid vehicles have contributed to solving some of these problems [2–4]. On the other hand, technologies focused on drivers, such as driver monitoring and hands-free, in-vehicle interfaces, are still not commonly utilized. Furthermore, there are an insufficient number of studies that

Y. Nishiwaki (✉) • C. Miyajima • N. Kitaoka • K. Takeda
Graduate School of Information Science, Nagoya University, Furo-cho, Chikusa-ku, Nagoya 464-8603, Japan
e-mail: nisiwaki@sp.m.is.nagoya-u.ac.jp

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_19, 

model human driving behavior, although vehicular behavior has been widely studied from the viewpoint of control theory [5–9]. Since human behavior is not deterministic, research that models driving behavior from the viewpoint of stochastic signal processing is important.

In this chapter, we propose a stochastic method of predicting vehicle trajectories during lane changes. In our proposed method, a trajectory model can be trained by a set of collected data based on the maximum likelihood principle without predetermined parameters. In addition, using a hidden Markov model (HMM), our method can model the multistate behavior of lane changing without explicit knowledge about state transitions or predetermined parameters.

Various approaches have been taken to predict vehicular behavior. Danielsson et al. [10] generated vehicle trajectories of surrounding vehicles for a few seconds. However, driver characteristics were not considered, and this method was not evaluated quantitatively. Althoff et al. [11] stochastically modeled the presence of trucks, cars, and pedestrians in traffic for a few seconds. However, the effectiveness of the modeling characteristics was not clear, nor were driver characteristics considered.

The most important contribution of this study is that we develop a model that can predict vehicle behavior for an interval of about 20 s, based on stochastic signal processing. Such long-term prediction was not discussed in previous works on control theory, as they assumed that sensing data are updated much more frequently. We propose a stochastic method to model characteristics of drivers' lane-change behavior and to predict a lane-change trajectory for a given initial condition and traffic environment. Our proposed method consists of two parts as shown in Fig. 19.1.

The first uses a hidden Markov model to characterize the stochastic dynamic properties of vehicular movements that originate from the driver's habitual characteristics. Since lane-change activity consists of multiple states (i.e., examining the safety of traffic environments, assessing the positions of other vehicles, moving into the next lane, and adjusting driving speed to traffic flow), a single dynamic system cannot model vehicle trajectory. In addition, the boundaries between states cannot be observed from its trajectory. HMMs can model such a stochastic state transition systems, and estimation-maximization (EM) algorithms can train HMMs without explicit information about state boundaries [12]. Furthermore, once the joint probability of a signal and its time derivatives, i.e., $z[n]$ and $\triangle z[n]$, are trained, the most probable signal sequence, $\{z[n]\}_{n\,=\,1,\,\ldots,\,N}$, can be calculated for a given state transition pattern [13]. Therefore, the first part of our model can generate trajectory hypotheses that represent characteristics of drivers' lane-change behavior.

The second part is a cognitive hazard map calculated from vehicle following distance distributions of training data. Here, the driver's sensitivity to the distance to a nearby vehicle in a particular location is modeled. Such sensitivities to surrounding vehicles are then integrated into a hazard map in a probability domain. Therefore, this function can be used for trajectory selection.

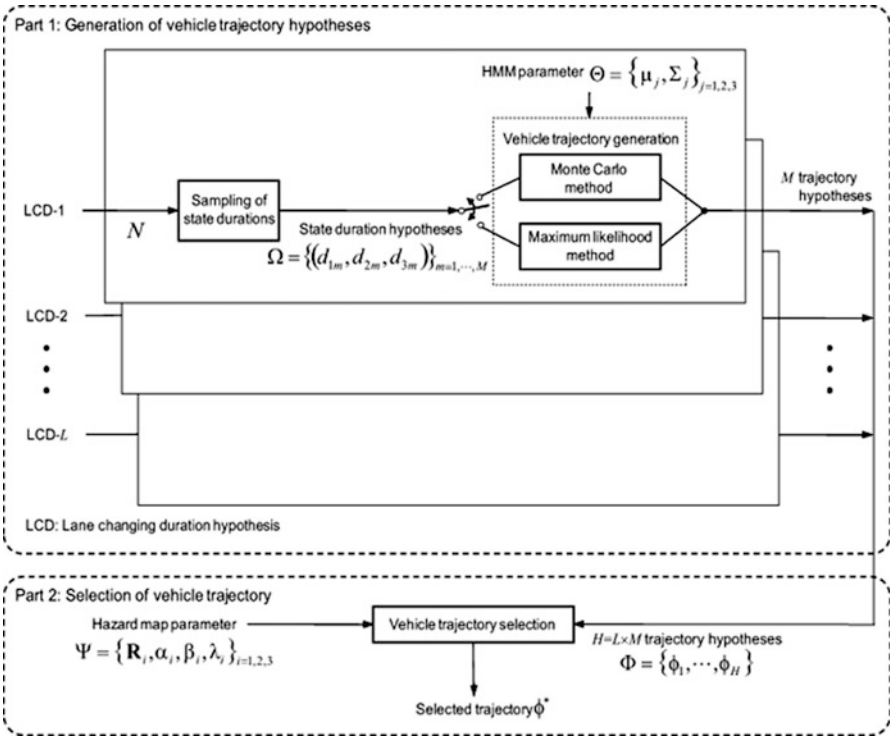



**Fig. 19.1** Overview of lane-change trajectory generation

Finally, the two processes are combined into a trajectory-predicting algorithm that first generates a set of probable trajectories by sampling from the HMM probability distributions and then selects the optimal trajectory based on the cognitive hazard map of the surrounding traffic.

## 19.2 Modeling Trajectories Using Hidden Markov Models

### *19.2.1 Trajectory Data*

A set of vehicle-movement observations was measured using a driving simulator. Relative longitudinal and lateral distances from the vehicle's position when starting the lane change, $x_i[n]$, $y_i[n]$, and the velocity of the vehicles, $\dot{x}_i[n]$, $\dot{y}_i[n]$, were recorded every 160 ms. Here, $i = 1, 2, 3$ is an index for the location of surrounding vehicles (Fig. 19.2), and $(x_0[n], y_0[n])$ represents the position of the driver's

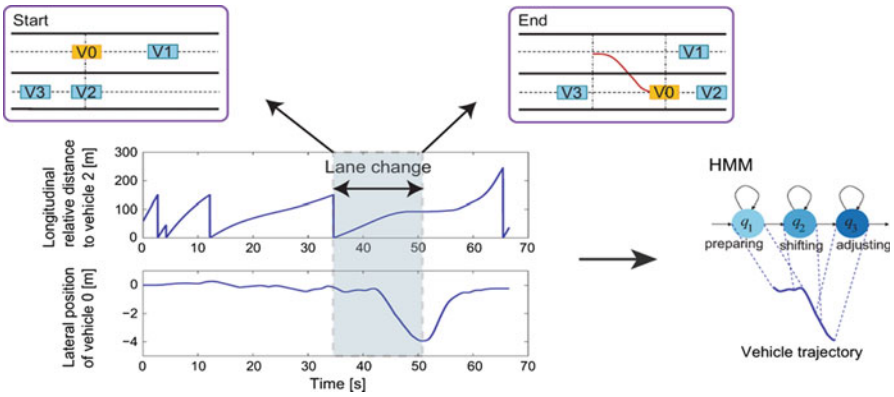



**Fig. 19.2** Lane-change trajectory and geometric positions of surrounding vehicles

own vehicle. The duration of lane-change activity, $n = 1, 2, \ldots, N$, starts when V0 (driver's own vehicle) and V2 are at the same longitudinal position and ends when V0's lateral position reaches the local minimum as shown in Fig. 19.2.

### *19.2.2 Hidden Markov Model*

We used a three-state HMM to describe the three different stages of a lane change: preparation, shifting, and adjusting.

In the proposed model, each state is characterized by a joint distribution of eight variables:

$$v = \left[\dot{x}_0,\ y_0,\ \Delta\dot{x}_0,\ \Delta y_0,\ \Delta^2\dot{x}_1,\ \Delta^2 y_0,\ \dot{x}_1,\ \dot{x}_2\right]^t \tag{19.1}$$

In general, longitudinal distance, $x_0$, monotonically increases in time and cannot be modeled by an i.i.d. process. Therefore, we use longitudinal speed $\dot{x}_0$, as a variable to characterize the trajectory. We calculated a higher-order time derivative$\dot{x}$ (or $\triangle x$) for signal $x$ by linear regression as follows:

$$x[n] = \frac{\sum_{k=-K}^{K} k \cdot x[n-k]}{\sum_{k=-K}^{K} k^2} \tag{19.2}$$

Finally, after training the HMM using a set of recorded trajectories, the mean vector $\mu_j$ and covariance matrix $\sum_j$ of the trajectory variable $\boldsymbol{v}$ are estimated for each state $j = 1, 2, 3$. The distribution of duration $N$ is modeled using a Gaussian distribution.

### 19.2.3 Trajectory Generation from a Hidden Markov Model

As shown in the following experiments, the shape of a trajectory is controlled by the HMM and the duration of the lane-change activity. When the driver performs a lane change in a shorter time, this results in a sharper trajectory. We generate a set of probable lane-change trajectories by determining state durations $\{d_j\}$ and by sampling the corresponding PDFs as follows.

First, we determine lane-change duration $N$ by sampling from its trained distribution. Then we determine state durations $d_j$ by uniformly sampling from the state duration distribution by

$$d_j = \left\lceil \frac{\xi_j N}{\sum_{k=1}^{K} \xi_k} \right\rceil \tag{19.3}$$

where $\lceil\ \rceil$ is a ceiling function, and $\xi_j$ is a random variable that follows a uniform distribution between zero and one. Once a set of state durations is determined, the maximum likelihood HMM signal synthesis algorithm (ML method) [13] or the sampling algorithm [14] generates the most probable trajectory. Simply repeating this process will produce a set of probable vehicle trajectories which characterize a trained driver's typical lane-change behavior.

## 19.3 Trajectory Selection

Although various natural driving trajectories may exist, due to the surrounding vehicle conditions, the number of lane-change trajectories that can be realized under given traffic circumstances is limited. Furthermore, the selection criteria of the trajectory, based on the traffic context, differ among drivers, e.g., some drivers are more sensitive to the position of the front vehicle than that of the side vehicle, etc. Therefore, we model the selection criterion of each driver with a scoring function for lane-change trajectories based on vehicular contexts, i.e., relative distances to the surrounding vehicles.

In the proposed method, a hazard-map function M is defined in a stochastic domain based on the histograms of the relative positions of the surrounding vehicles $\boldsymbol{r}_i = [\tilde{x}_0, x_0, \tilde{y}_0, y_0]^t$.

To model sensitivity to surrounding vehicles, we calculated covariance matrix $\mathbf{R}_i$ for each of three distances, $\boldsymbol{r}_i$, $i$ =1, 2, 3, using training data. Since the distance varies more widely at less sensitive distances, we use the quadratic form of inverse covariance matrices $\mathbf{R}_i$ as a metric of the cognitive distance. Then we calculate hazard-map function M for surrounding vehicle $\mathrm{V}_i$ as follows:

$$\mathcal{M}\left(\boldsymbol{r}_i^t \boldsymbol{R}_i^{-1} \boldsymbol{r}_i\right) = \frac{1}{1 + \exp\left\{\alpha_i\left(\boldsymbol{r}_i^t \boldsymbol{R}_i^{-1} \boldsymbol{r}_i - \beta_i\right)\right\}} \tag{19.4}$$

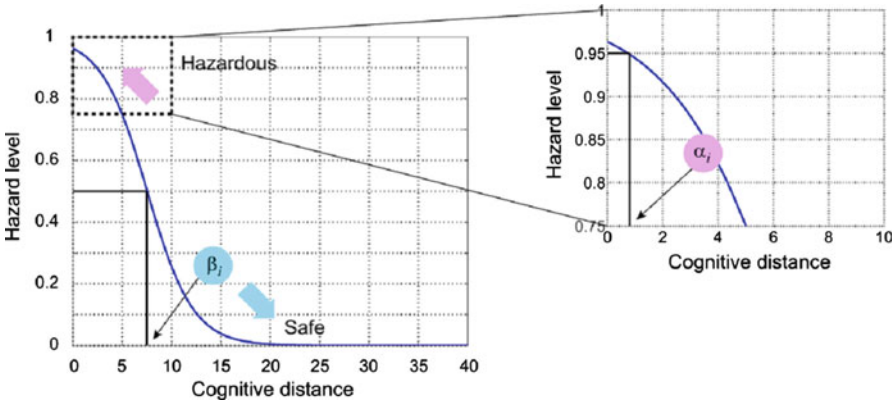



**Fig. 19.3** Parameters of a hazard map

where $\alpha_i$ is a parameter of the minimum safe distance defined so that the minimum value of cognitive distance $\boldsymbol{r}_i^t \boldsymbol{R}_i^{-1} \boldsymbol{r}_i$ of the training data corresponds to the lower 5% distribution values, and $\beta_i$ is the 50% distribution value (mean value) of $\boldsymbol{r}_i^t \boldsymbol{R}_i^{-1} \boldsymbol{r}_i$ (Fig. 19.3).

$$\mathcal{M}\left(\min \boldsymbol{r}_i^t \boldsymbol{R}_i^{-1} \boldsymbol{r}_i\right) = 0.95 \tag{19.5}$$

$$\mathcal{M}\left(\overline{\boldsymbol{r}_i^t \boldsymbol{R}_i^{-1} \boldsymbol{r}_i}\right) = 0.5 \tag{19.6}$$

The hazard-map parameters $\alpha_i$ and $\beta_i$ are obtained as follows by solving Eqs. 19.5 and 19.6 with respect to $\alpha_i$ and $\beta_i$:

$$\alpha_i = \frac{\log(0.05) - \log(0.95)}{\min\left\{ \boldsymbol{r}_i^t \boldsymbol{R}_i^{-1} \boldsymbol{r}_i \right\} - \overline{\boldsymbol{r}_i^t \boldsymbol{R}_i}^{-1} \boldsymbol{r}_i} \tag{19.7}$$

$$\beta_i = \overline{\boldsymbol{r}_i^t \boldsymbol{R}_i}^{-1} \boldsymbol{r}_i \tag{19.8}$$

Hazard map $M$ can take values within the interval (0, 1); the higher the value, the more hazardous the situation.

Hazard map $M$ can be regarded as an a posteriori probability of being in the safe driving condition under range distances $\Pr\{\text{safe} \mid \boldsymbol{r}\}$, when the likelihood is given as an exponential quadratic form, i.e.,

$$\Pr\{r|\text{safe/unsafe}\} \propto \exp\left(-\frac{1}{2}\boldsymbol{r}^t \boldsymbol{A} \boldsymbol{r}\right) \tag{19.9}$$

where $\boldsymbol{A}$ is an invertible square matrix. Therefore, integrating the hazard maps for three surrounding vehicles can be done simply by interpolating three probabilities with weights $\lambda_i$ into an integrated map:

$$\mathcal{M}' = \sum_i \frac{\lambda_i}{1 + \exp\left\{\alpha_i \left(\boldsymbol{r}_i^t \boldsymbol{R}_i^{-1} \boldsymbol{r}_i - \beta_i\right)\right\}} \tag{19.10}$$

Once the positions of the surrounding vehicles at point in time $n$, $\boldsymbol{r}_i[n]$ are determined, $M$ can be calculated for each point in time, and by averaging the value over the lane-change duration, we can compare the possible trajectories. Then the optimal trajectory that has the lowest value is selected from the possible trajectories.

## 19.4 Evaluation

### *19.4.1 Data Collection and Setup*

Thirty lane-change trials were recorded for each of two drivers using a driving simulator which simulated a two-lane urban expressway where the traffic was moderately dense. The velocities of the vehicles in the passing lane ranged between 82.8–127.4 km/h, and the distances between two successive vehicles in the passing lane ranged between 85–315 m. The drivers were instructed to pass the preceding vehicle once during each trial, when they were able to. The average velocity of the two drivers when passing the lead vehicle was 112.4 km/h.

Figure 19.4 shows the lane-change duration at each trial and its most probable state durations. The distribution of lane-change duration characterizes a driver's lane-change behavior. For example, on average, driver B required more time than A to complete a lane change.

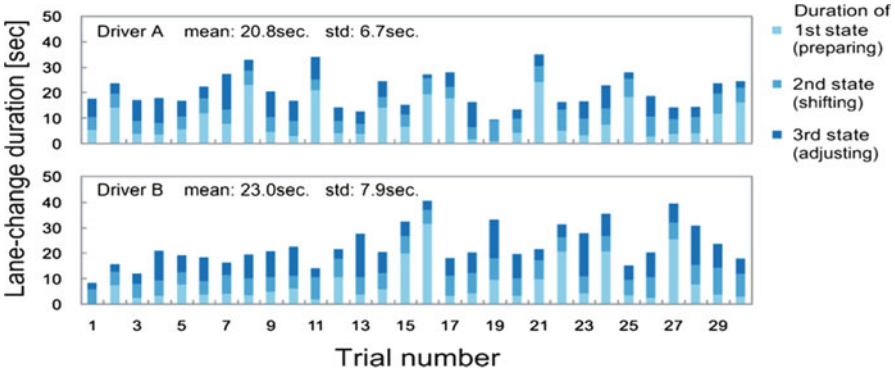



**Fig. 19.4** Lane-change duration and its most probable state durations calculated using HMM

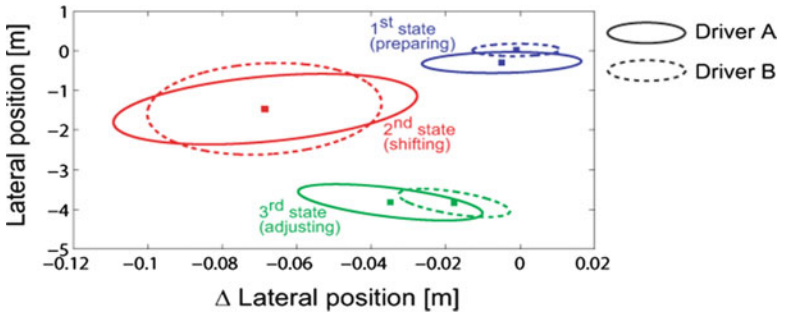



**Fig. 19.5** Joint PDFs of trajectory variables trained with three states of HMM for two drivers (y-Δy plain is plotted). Square dots show the means and contours represent "one sigma" boundaries

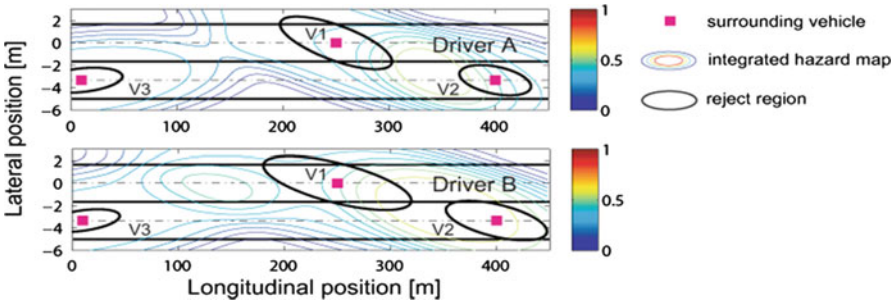



**Fig. 19.6** Hazard maps for two drivers when the same positions of surrounding vehicles were given

Thirty trials were used for threefold cross validation tests: twenty for training and ten for tests. Each state of an HMM was characterized by a joint Gaussian PDF of trajectory variables and was trained using an HTK [15].

Four hundred possible trajectories were generated from an HMM. First, 20 lane-change duration values $N$ were sampled from the distribution of a driver's own lane-change durations. Then, for each sampled lane-change duration $N$, 20 sets of state durations $\{d_j\}$ were hypothesized, also by sampling from the uniform distribution using Eq. 19.3. To select the optimal trajectory, we integrated three hazard maps into a single hazard map with equal weights, i.e., $\lambda_1, \lambda_2, \lambda_3 = 1/3$ in Eq.19.10. We assumed that the surrounding vehicle speeds $\dot{x}_i$, $\dot{y}_i$ were constant throughout the lane-change activity.

### *19.4.2 Results*

The trained joint PDFs of the trajectory variables are plotted for each HMM state of the two drivers in Fig. 19.5. We confirmed that the habitual differences in lane-

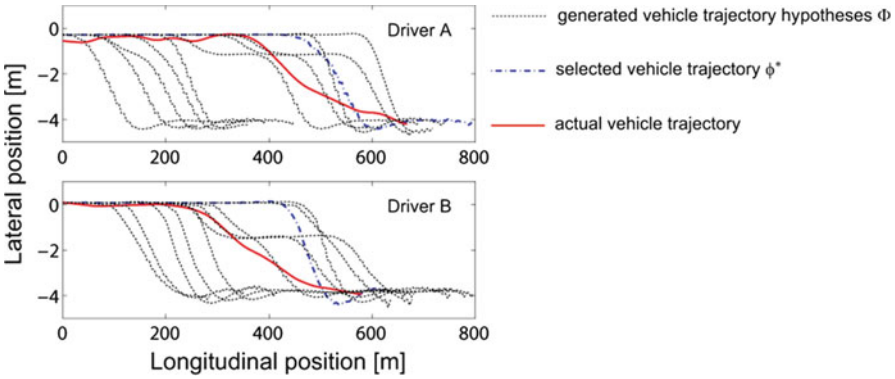



**Fig 19.7** Examples of generated (dotted line) and the selected (dashed line) trajectories for maximum likelihood (ML) signal synthesis algorithm. The actual trajectory observed under the given condition is also plotted (solid line)

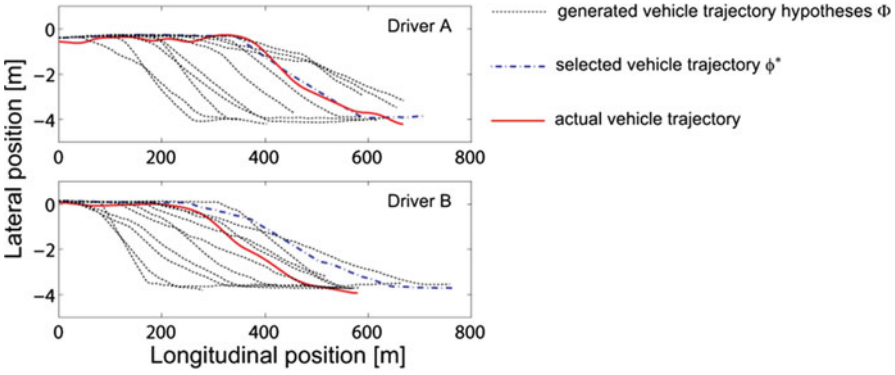



**Fig 19.8** Examples of generated (*dotted line*) and the selected (*dashed line*)trajectories for sampling method. The actual trajectory observed under the given condition is also plotted (*solid line*)

change behavior can be modeled with HMM parameters. The trained hazard maps M' for the two drivers shown in Fig. 19.6 also depict differences in sensitivity to surrounding vehicles.

We generated possible lane-change vehicle trajectories over a 20-s period using two methods: a maximum likelihood (ML) method [13] and a sampling method. The possible generated trajectories and the selected optimal trajectory using the ML method are shown in Fig. 19.7, and those generated using the sampling method are shown in Fig. 19.8. The vehicles traveled about 600 m while changing lanes. The trajectories of the two drivers are clearly different.

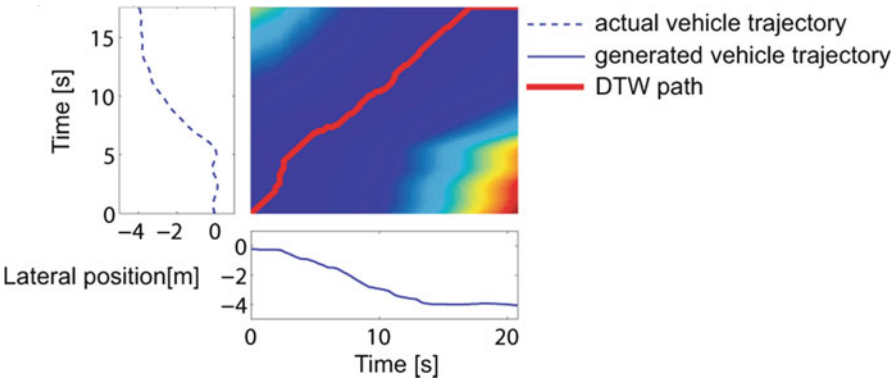



**Fig 19.9** Examples of DTW recursion between actual and generated vehicle trajectories

For further quantitative evaluation, we calculated the difference between the predicted and actual trajectories based on dynamic time warping (DTW), using the normalized square difference as a local distance:

$$D(i,j) = \min\begin{cases} D(i-1,j) \\ D(i-1,j-1) \\ D(i,j-1) \end{cases} + \frac{1}{I+J-1}\left\{\frac{(x_0[i]-\hat{x}_0[j])^2}{\sum_{n=1}^{I} x_0^2[n]} + \frac{(y_0[i]-\hat{y}_0[j])^2}{\sum_{n=1}^{I} y_0^2[n]}\right\} \tag{19.11}$$

where $I$ and $J$ are the length of the actual and predicted trajectories, respectively. An example of the DTW result is shown in Fig. 19.9. The DTW recursion proceeds from $D(0, 0) = 0$ to $D(I, J)$. We used $10{\cdot}\log(D)$ as a signal-to-deviation (SDR) measure for the prediction. This is because the lengths of the actual and predicted trajectories are different.

Average SDRs of the best trajectory hypothesis (best) and all trajectory hypotheses (mean) using the maximum likelihood method (left) and the sampling method (right) are shown in Fig. 19.10. The resultant SDR of the sampling method for 60 tests was 38.0 dB. The sampling method was better at generating vehicle trajectories similar to actual driver trajectories than the ML method.

Figure 19.11 shows the resultant SDRs when driver A's model was used for predicting driver B's trajectory and vice versa. The SDR decreased by 2.2 dB when the other driver's model was used to make the prediction. This result confirmed the effectiveness of the proposed model for capturing individual characteristics of lane-change behavior. We also tested our method using the actual lane-change duration, i.e., $I = J$. When the actual lane-change duration $N$ is given, the root mean square error (RMSE) between the predicted and actual trajectories can be calculated. The average RMSE for 60 tests was 17.6 m, which was a good result for predicting vehicle trajectories over a distance of about 600 m (i.e., for a 20-s time period).

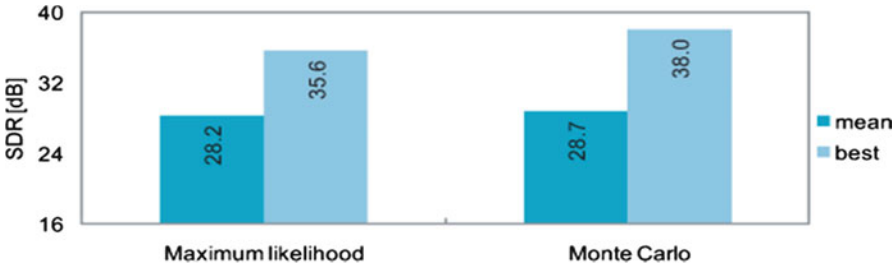



**Fig 19.10** Average SDRs of the best trajectory hypothesis (*best*) and all trajectory hypotheses (*mean*) using maximum likelihood method (*left*) and sampling method (*right*)

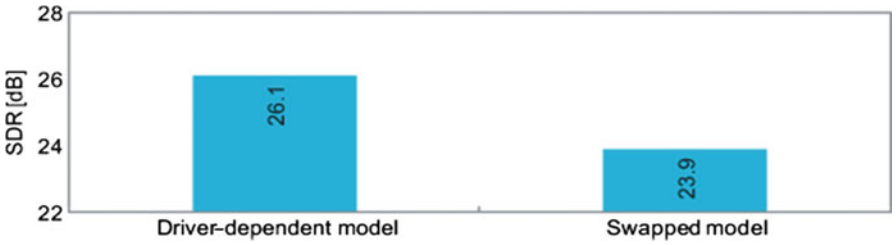



**Fig 19.11** Average SDRs of trajectories selected using a driver's own models (*left*) and using the other driver's models (*right*)

## 19.5 Summary and Future Work

In this chapter, we proposed a stochastic framework for modeling driving behavior where a driver's habitual behavior and cognitive characteristics are modeled using an HMM and a geometrical probability function. The proposed method can predict lane-change trajectory of about 20 s in length given only the initial conditions by generating a set of probable trajectories with the HMM and then selecting the optimal trajectory with the geometric function. Since model parameters can be trained based on statistical training criteria, a driver's personal driving style can be easily characterized using training data.

Based on experimental evaluations for two drivers, we confirmed that the model can generate a reasonably accurate personalized trajectory. However, further study is needed. First, more analytical and quantitative evaluation of the method is necessary, using a larger amount of data. Also, the model should be tested using real driving data collected under actual traffic conditions. Integrating the generation and selection processes, based on a consistent criterion, is also a very challenging but important task.

**Acknowledgments** This work was supported by the Strategic Information and Communications R&D Promotion Program (SCOPE) of the Ministry of Internal Affairs and Communications of Japan and by the Core Research for Evolutional Science and Technology (CREST) of the Japan Science and Technology Agency. We are also grateful to the members of these projects for their valuable comments.

## References

1. National Police Agency Traffic Bureau, Traffic accident statistics 2010 (in Japanese). http://www.npa.go.jp/toukei/index.htm#koutsuu. Accessed 08 Apr 2011
2. An PE, Harris CJ (1996) An intelligent driver warning system for vehicle collision avoidance. IEEE Trans Syst Man Cybern A 26(2):254–261
3. Onken R (1994) DAISY, an adaptive, knowledge-based driver monitoring and warning system. In: Proceedings of the vehicle navigation and information systems conference, Yokohama, Aug 1994, pp 3–10
4. Gong Q, Li Y, Peng ZR (2008) Computationally efficient optimal power management for plug-in hybrid electric vehicles based on spatial-domain two-scale dynamic programming. In: Proceedings of 2008 international conference on vehicular electronics and safety (ICVES 2008), Columbus, Sept 2008
5. Brackstone M, McDonald M (1999) Car-following: a historical review. Transport Res F 2(4):181–196
6. Fritz H, Gern A, Schiemenz H, Bonnet C (2004) CHAUFFEUR assistant: a driver assistance system for commercial vehicles based on fusion of advanced ACC and lane keeping. In: Proceedings of 2004 IEEE intelligent vehicles symposium (IV 2004), Parma, June 2004, pp 495–500.
7. Ishida S, Gayko JE (2004) Development, evaluation and introduction of a lane keeping assistance system. In: Proceedings of 2004 IEEE intelligent vehicles symposium (IV 2004), Parma, June 2004, pp 943–944
8. Chee W, Tomizuka M (1994) Vehicle lane change maneuver in automated highway systems. Publication of PATH project, ITS, UC Berkeley, UCB-ITS-PRR-94-22, 1994
9. Wenzel TA, Burnham KJ, Williams RA, Blundell MV (2005) Closed-loop driver/vehicle model for automotive control. In: Proceedings of the international conference on systems engineering (ICSEng 2005), Las Vegas, Aug. 2005, pp 46–51
10. Danielsson S, Petersson L, Eidehall A (2007) Monte carlo based threat assessment: analysis and improvements. In: Proceedings of 2007 IEEE intelligent vehicles symposium (IV 2007), Istanbul, June 2007, pp 233–238
11. Althoff M, Stursberg O, Buss M (2008) Stochastic reachable sets of interacting traffic participants. In: Proceedings of 2008 IEEE intelligent vehicles symposium (IV 2008), Eindhoven, June 2008, pp 1086–1092
12. Rabiner LR (1989) A tutorial on hidden Markov models and selected applications in speech recognition. In: Proceedings of the IEEE, Feb 1989, pp 257–286
13. Tokuda K, Yoshimura T, Masuko T, Kobayashi T, Kitamura T (2000) Speech parameter generation algorithms for HMM-based speech synthesis. In: Proceedings of 2000 IEEE international conference on acoustics, speech, and signal processing (ICASSP 2000), Istanbul, June 2000, pp 1315–1318
14. Nishiwaki Y, Miyajima C, Kitaoka N, Takeda K (2009) Stochastic modeling of vehicle trajectory during lane-changing. In: Proceedings of 2009 IEEE international conference on acoustics, speech, and signal processing (ICASSP 2009), Taipei, Apr 2009, pp 1377–1380
15. Young S, Evermann G, Gales M, et. al (2001–2006) The HTK book (for HTK version 3.4). Cambridge University Engineering Department

# Chapter 20
# CAN-Bus Signal Analysis Using Stochastic Methods and Pattern Recognition in Time Series for Active Safety

**Amardeep Sathyanarayana, Pinar Boyraz, Zelam Purohit, and John H.L. Hansen**

**Abstract** In the development of driver-adaptive and context-aware active safety applications, CAN-Bus signals play a central role. Modern vehicles are equipped with several sensors and ECU (electronic control unit) to provide measurements for internal combustion engine and several active vehicle safety systems, such as ABS (anti-lock brake system) and ESP (electronic stability program). The entire communication between sensors, ECU, and actuators in a modern automobile is performed via the CAN-Bus. However, the long-term history and trends in the CAN-Bus signals, which contain important information on driving patterns and driver characteristics, has not been widely explored. The traditional engine and active safety systems use a very small time window ($t < 2s$) of the CAN-Bus to operate. On the contrary, the implementation of driver-adaptive and context-aware systems requires longer time windows and different methods for analysis. In this chapter, a summary of systems that can be built on this type of analysis is presented. The CAN-Bus signals are used to recognize the patterns in long-term representing driving subtasks, maneuvers, and routes. Based on the analysis results, quantitative metrics/feature vectors are suggested that can be used in many ways, with two prospects considered here: (1) CAN-Bus signals can be presented in a way to distinguish distracted/impaired driver behavior from normal/safe and (2) driver characteristics and control strategies can be quantitatively identified so that active safety controllers can be adapted accordingly to obtain the best driver–vehicle response for safe systems. In other words, an optimal human–machine cooperative system can be designed to achieve improved overall safety.

**Keywords** Active safety • CAN-Bus • Time-series analysis

A. Sathyanarayana (✉) • P. Boyraz • Z. Purohit • J.H. Hansen
Center for Robust Speech Systems, University of Texas at Dallas,
Richardson, TX, USA
e-mail: amardeep@utdallas.edu; boyraz.pinar@googlemail.com; zelam.purohit@gmail.com;
john.hansen@utdallas.edu

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_20, 

## 20.1 Introduction

The last 20 years have witnessed a transformation of modern automobiles, turning them into vehicles packed with sensors, microchips, and actuators, all forming integrated and modular subsystems of safety, infotainment, and energy management. In fact, automobiles have been perhaps the first merger between mechanical and electrical/electronic components offering flexibility for better control of (1) energy production and use (i.e., timed/controlled internal combustion engine cycle or hybrid technology energy cycle/system switch management), (2) vehicle dynamics (i.e., ABS, ESP) and (3) instrument cluster (i.e., better displays, adaptable controls, setting points, etc.), and (4) driver assistance systems (i.e., LKS, ACC, Blind Spot Warning, Parking Assistance, etc.). In the center of these developments is a protocol that made it all possible to communicate the messages between sensors, processing units, and actuators. That protocol and system called CAN-Bus (Controller Area Network) was introduced in early 1990s [1]. While this transformation is taking place, another dimension has caught the attention of researchers. All the technology in modern vehicles needs to consider the human component: driver. Although the pursuit of understanding or modeling human driver behavior is not new [2–4], the long-awaited merger between advanced vehicle concepts and human-centered systems has just began. To be able to design truly cooperative and effective driver assistance, safety, or infotainment systems, driver behavior needs to be better understood, modeled, and incorporated into the system design. The subject of this chapter is to utilize CAN-Bus signals and demonstrate the new opportunities of using it to model driver behavior and suggest system implementations incorporating intelligent CAN-Bus processing. In this chapter, newly developed CAN-Bus data analysis tools are presented in Sect. 2. Next, the systems and applications based on CAN-Bus analysis are demonstrated. Finally, conclusions are drawn from the findings of UTDrive project over the last 1.5 years and future directions are shown to attract more research into this very exciting new area.

## 20.2 CAN-Bus Data Analysis

CAN-Bus data analysis requires a multimedia data annotation tool and a common protocol to be able to segment the data into meaningful parts and use them efficiently in modeling. Therefore, a multimedia data annotation tool (UTDAT) using video channels (driver and road scene videos), driver's speech, and CAN-Bus is designed.

Accompanying this tool, a color code for driving timeline (CCDT) has been designed to interpret the driving data from multiple channels for event detection. Using these two tools, it is possible to zoom into particular sections of the data and run specific analysis on the CAN-Bus or accompanying channels. The database

used to develop these tools is the UTDrive Corpus. More extensive information can be found on data collection procedure, data structure, and properties of the UTDrive Corpus in [9].

### *20.2.1 Data Annotation Tool: UTDAT*

Data annotation is the most crucial step in the analysis of multisensor data analysis since it provides the basis for further signal processing. It should be noted that although the segments of the roads are assigned to different tasks and driving events can be also detected using this information, data collection is highly dynamic in nature taking place in real traffic. Therefore, it is required to tag the events and tasks to record their time tags (begin–end). For this particular study, the interest is to recognize the driving maneuvers and detect distractions; therefore, two different transcription files are prepared for each run. First, using video streams and CAN-Bus channels, driving events are tagged as having six different labels: right turn (RT), left turn (LT), lane change (LC), lane keeping in straight segment (LKS), lane keeping in curved segment (LKC), and stops (ST). The events constitute the driving event timeline parsing the session into meaningful parts which need to be examined separately. Second, transcription involves time-tagging of 12 important task-related events using the audio signal together with the video. These 12 labels are: driver talks (DT), experimenter talks (ET), navigation instruction (NI), silence (SI), Tell-Me Dialog system (TM), American Airline Dialog system (AA), lane change prompts (LP), common tasks (CT), sign reading (SR), music playing (MP), and two additional driver response–related tags, interrupted utterance (IU) and response delay (RD). UTDAT data annotation tool is written using the MATLAB GUI and is shown in Fig. 20.1.

### *20.2.2 Color-Coded Driving Timeline (CCDT): A Novel Way to Look at CAN-Bus*

In order to facilitate the analysis of large-size multisensory driving data, a color code for driving timeline is prepared, visually marking each event and task label with certain colors and projecting them as two parallel timelines. An example of the timeline is shown in Fig. 20.2 with the legend of the Color-Coded Driving Timeline (CCDT).

Using CCDT, it is possible to observe the events and secondary tasks in a session simultaneously. This visualization tool is heavily used in further analysis stages for building the distraction/workload hypotheses exploiting overlaps between tasks and events in the timeline.

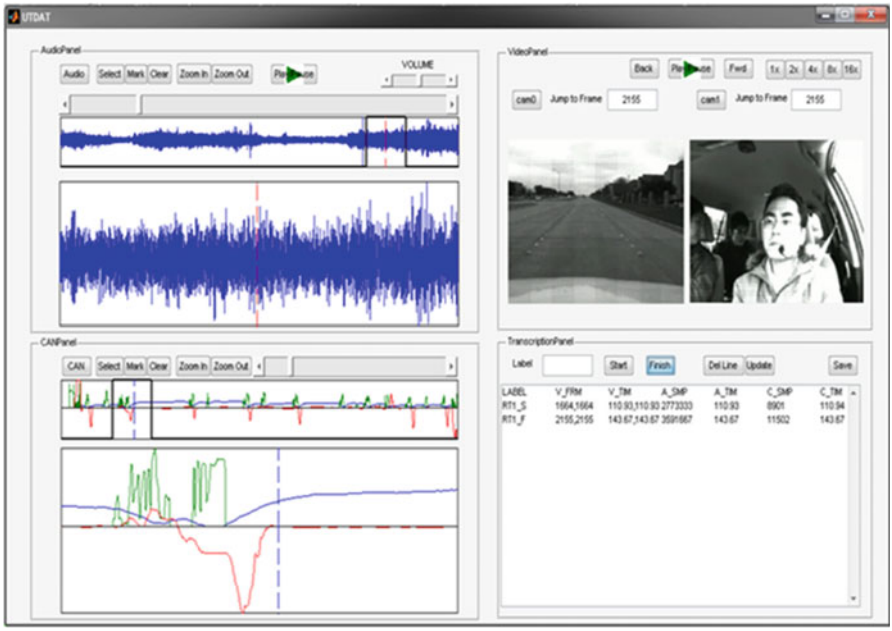



**Fig. 20.1** UTDAT multimedia data annotation tool is capable of cross-referencing and synchronization of two videos, one audio, and CAN-Bus streams

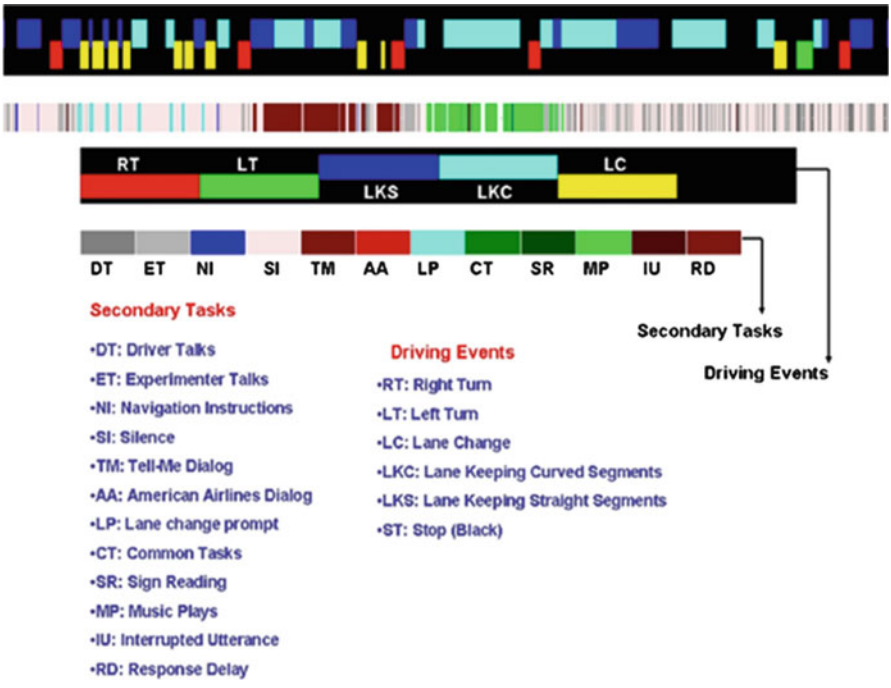



**Fig. 20.2** Timeline of driving events (*black band*) and tasks (*white band*) depicted in CCDT

## 20.3 Systems and Applications

A categorization of applications is given in Fig. 20.3 leading to different active vehicle safety (AVS) structures. For both context recognition and abnormality detection, the application can be either (1) generic or (2) person-specific. Generic systems are expected to take 95% of the drivers with reasonable reliability and acceptable false alarm rates (i.e., less than 2%). Designing such a generic system is difficult because of the highly dynamic nature of the driving task, including the variations between drivers, conditions, and even discrepancies between two sessions of driving on the same route by the same driver. Previous work has concentrated on designing a generic system for context recognition and abnormality detection using stochastic methods with a non-optimal feature vector [7]. The opposite of a generic approach, these systems can be person-dependent in order to reduce the effect of the inter-driver variation on performance for recognition. However, driver-dependent AVS systems require that personal driving characteristics and/or biometrics be stored on the in-vehicle system. Driver-dependent AVS is expected to have at least three submodules: (1) driver identification – use speaker and/or face recognition or smart key to reduce complexity of driver monitoring, (2) maneuver/context recognition – monitor and recognize driving context to reduce the complexity of abnormality detection task, and (3) abnormality

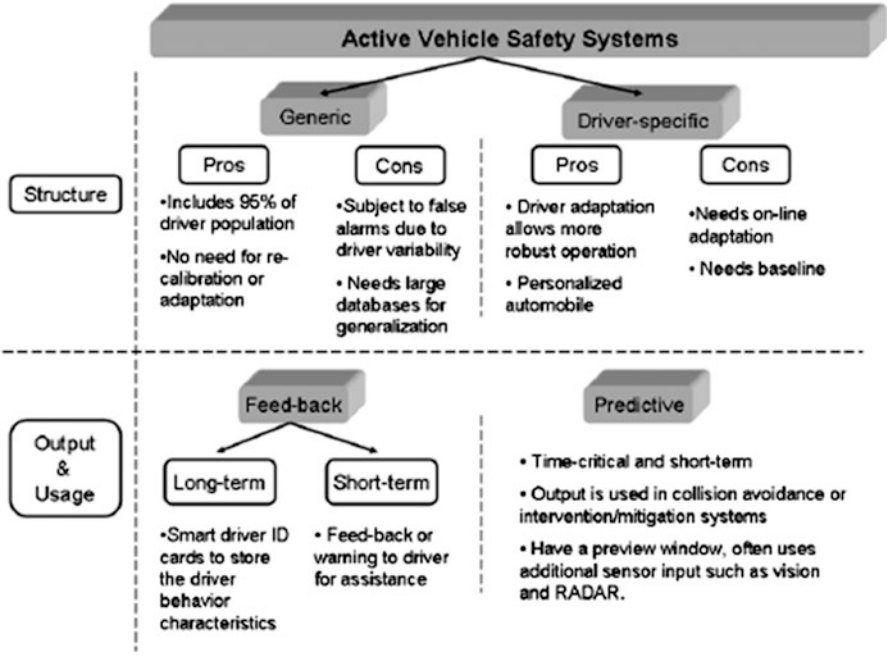



**Fig. 20.3** Active Vehicle Safety (AVS) systems categorized according to their data/structure and output/end-use

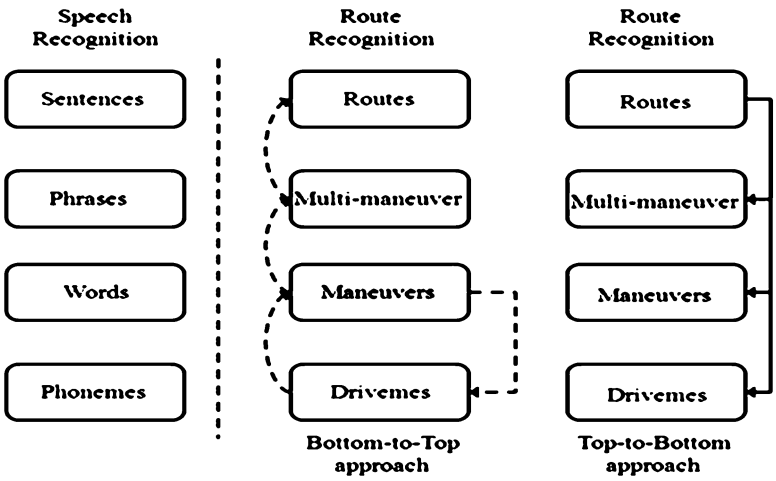



**Fig. 20.4** Hierarchy among the units of speech recognition and maneuver recognition

detection – given the specific driver characteristics/ models and context, this model is expected to detect the abnormalities (i.e., due to distraction, sleepiness, inattention). A driver-dependent framework has been previously designed and evaluated [4, 8].

### *20.3.1 Generic Maneuver Recognition and Distraction Detection*

In the generic approach, it is assumed that no domain information is available; however, the patterns in the signals can be recognized using general signal processing approaches. Driving signals are considered based on the analogy with speech signals and how they are processed. This analogy is given in Fig. 20.4.

This approach uses HMMs to model the maneuvers and neutral/distracted versions of the maneuvers, and the comprehensive results can be found in [6]. In the bottom-to-top (BtT) approach, an isolated subunit is the main interest of the overall recognition algorithm. After obtaining a separate HMM for each maneuver defined in the route, the route model can be constructed by internal semantics and syntax structure.

According to this approach, "drivemes" common to all maneuvers can be discovered and used to build up maneuver models. These maneuver models can be used to build multi-maneuver models and finally complete routes. Alternatively, in the top-to-bottom (TtB) approach, a single HMM with a large number of states is trained. In this manner, we assume that there is no a priori information known about the individual maneuvers. We further assume that we have a record of a meaningful data sequence which is constructed by some units; however, we do not insert restrictions on their duration. After training this HMM framework, certain pruning techniques, including clustering and the Viterbi algorithm, are used to determine

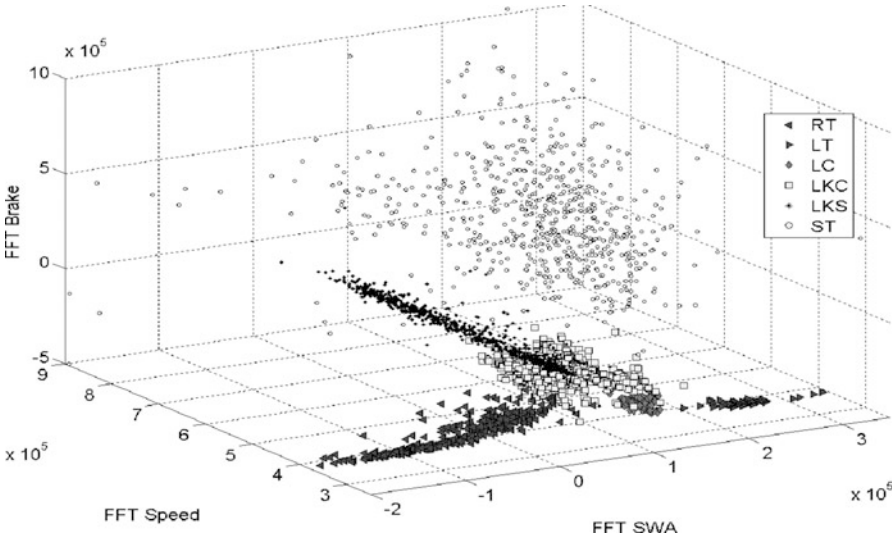



**Fig. 20.5** 3-D scatter plot of first FFT coefficients of CAN-Bus signals for six maneuvers

which states are dominant in the single HMM, while a portion of the route known as a certain maneuver is presented as an observation sequence. The discovered dominant states can be concatenated (state-tying) to represent certain maneuvers; therefore, a finer model of the HMM can be obtained for the route. Using the HMM framework, it is possible to recognize 100%, 93%, and 81% of 112 right turns, 29 left turns, and 70 lane change events, respectively. As for distraction detection, distracted drivers were recognized with 100% rate for LT and LC maneuvers; however, we could not obtain the same performance for RT. Since the approach did not use CCDT segmentation at that time, the train data was not representing ground truth in terms of maneuvers tagged with distraction. It is assumed that all the maneuvers in distracted sessions were representing distracted data; however, this is not necessarily so.

To improve the ground truth and perform finer generic maneuver recognition, UTDAT and CCDT were utilized and a much simpler approach utilizing FFT was performed. From this new analysis, very well-separated clusters of maneuvers were obtained as shown in Fig. 20.5. Using geometrically defined decision surfaces, the maneuvers were recognized with better performance. The results are given in Table 20.1 [4]. The recognition results were further improved by optimizing the decision surfaces using SVMs, giving 99% accuracy, with confusion occurring only between LKS and LKC maneuvers. It was understood that drivers have different baselines and there is also variation among the same driver's data from the same route; therefore, a driver-dependent approach was pursued for distraction detection.

**Table 20.1** Generic maneuver recognition performance using FFT and geometric decision surfaces

| | | |
|---|---|---|
| True positive rate | TPR = TP/P | 93.7% |
| False positive rate | FPR = FP/P | 0.8% |
| Accuracy | ACC = (TP + TN)/(P + N) | 93.7% |
| Specifity | SPC = 1-FPR | 99.0% |
| Positive prediction value | PPV = TP/(TP + FP) | 95.8% |
| Negative prediction value | NPV = TN/(TN + FN) | 99.1% |
| False discovery rate | FDR = FP/(FP + TP) | 4.1% |

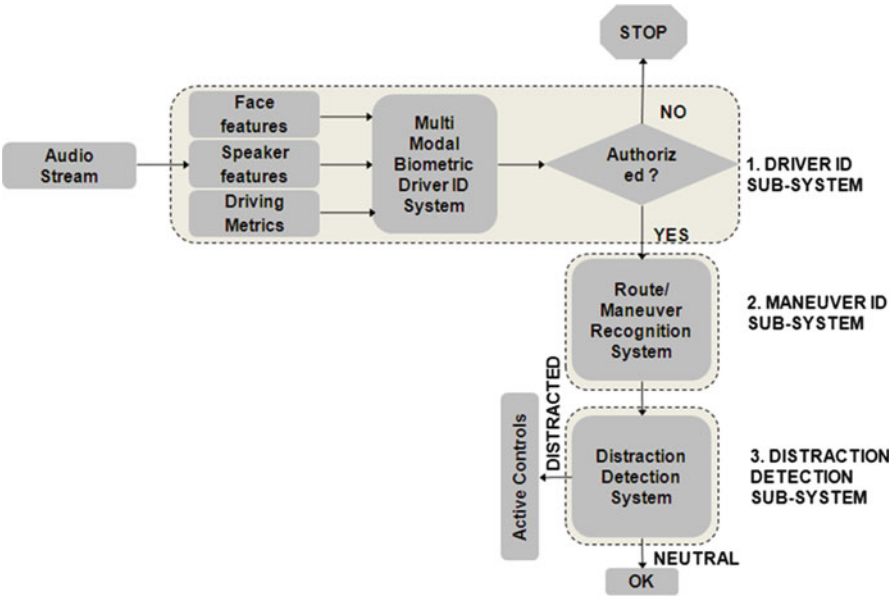



**Fig. 20.6** Driver-dependent maneuver recognition and distraction detection system

### *20.3.2 Driver-Specific Distraction Detection*

For the driver-specific approach, this is particularly suitable for detecting distraction by eliminating the variation from driver characteristics. If the algorithm is applied after maneuver recognition, a further reduction of variations can be achieved. Motivated by the need to have a robust system depending on driver individual traits, an integrated narrowing-down approach has been proposed [7, 8]. The driver-specific system is shown in a block diagram in Fig. 20.6. To recognize the driver's identity, a speaker ID system was implemented using 30 s of driver's speech for in-set based training including nine drivers with 100% accuracy. When duration of the data is reduced to 10 s, the accuracy drops to 91%, with further reductions with 5 s and 2 s to 86% and 68%, respectively.

We suggest that already-proven biometric signals such as speech, fingerprint, and face recognition should be utilized for driver identification. Although CAN-Bus signals carry important personal traits, it was found that the performance of a CAN-Bus–based identification was much lower than other biometric systems (between 83% and 90% recognition accuracy).

The driver-dependent system described here used a GMM/UBM-based structure for distraction detection. The average distraction detection performance was always above 70% for all maneuvers. However, the system is not able to recognize neutral cases better than 70% as well. Therefore, the false alarm rate is expected to be approximately 30%, which is unacceptable for a final safety application. Again, after using UTDAT and CCDT tools, better data pools are obtained representing ground truth in the driving timeline. These tools and a finer analysis on distraction detection improved the results to 95% for distraction detection as reported in [9] using specific driver performance metrics based on high-frequency content, sample entropy, and standard deviation.

## 20.4 Conclusion

CAN-Bus signal analysis performed in long-term time windows open the door to truly human-centric systems which are capable of recognizing the context/maneuver and detecting distraction which can become an important module in driver status monitoring and assistance systems. This chapter summarized the recent findings in CAN-Bus analysis in the UTDrive project during the past 1.5 years. Two important data mining tools were developed and found to be extremely beneficial for multimedia data analysis. It was understood that if examined carefully, CAN-Bus signals carry important traces on context information and driver status. This concealed information pieces can be made explicit and interpreted for the benefit of active safety systems incorporating human factors into system design.

**Acknowledgments** The authors would like to acknowledge the diligent work in audio/task transcriptions by CRSS staff Rosarita Lubag.

## References

1. CAN-Bus technical specifications from Bosch http://www.semiconductors.bosch.de/pdf/can2spec.pdf
2. McRuer D, Weir D (1969) Theory of manual vehicular control. Ergonomics 12:599–633
3. MacAdam C (1981) Application of an optimal preview control for simulation of closed-loop automobile driving. IEEE Trans Syst Man Cybern SMC-11:393–399
4. Boyraz P, Hansen JHL Active vehicle safety systems based on intelligent CAN-Bus signal modules, in preparation, to be submitted to IEEE Trans. on ITS, June 09

5. Michon JA (1985) A critical view of driver behavior models: what do we know, what should we do? In: Evans L, Schwing RC (eds) Human behavior and traffic safety. Plenum Press, New York, pp 485–520
6. Sathyanarayana A, Boyraz P, Hansen JHL (2008) Driver behaviour analysis and route recognition by Hidden Markov Models. In: IEEE international conference on vehicular electronics and safety, Ohio, USA, 22–24 Sept 2008
7. Boyraz P, Sathyanarayana A, Hansen JHL (2008) In-vehicle multi-channel signal processing and analysis in UTDrive project: driver behaviour modeling and active safety systems development. In: 3rd international conference ESAR, Hannover, Germany, 5–6 Sept 2008
8. Sathayanarayana A, Boyraz P, Hansen JHL (2011) Information fusion for context and driver aware active vehicle safety systems, Elsevier Journal on Information Fusion, 12(4), Oct 2011 ISSN: 15662535, DOI: 10.1016/j.inffus.2010.06.004
9. Boril H, Boyraz P, Hansen JHL Towards multi-modal driver's stress detection, 4th Biennial-Workshop on DSP for In-Vehicle Systems and Safety, 25–27 June 2009, TX, USA

# Part D
# Transportation, Vehicle Communications, and Next Generation Vehicle Systems

# Chapter 21
# Adaptive Error Resilient Mechanisms for Real-Time Multimedia Streaming over Inter-Vehicle Communication Networks

**Matteo Petracca, Paolo Bucciol, Antonio Servetti, and Juan Carlos De Martin**

**Abstract** To allow real-time streaming of loss tolerant flows such as multimedia streams in inter-vehicle communication network, we propose a cross-layer technique based on proactive error correction and interleaving algorithms. The proposed technique optimizes the FEC/interleaving channel coding parameters based on network layer information under real-time constraints. It is implemented at packet level to allow a straightforward adaption in the existing wireless devices. By resorting to standard compliant, real-time RTCP reports, we also develop and optimize an adaptive technique that is able to match fast channel variations and reduce both the overhead required by the proactive error recovery scheme and the additional delay introduced by the interleaver. Simulations based on Gilbert–Elliott wireless channel model show that the proposed adaptive technique without optimizations is able to gain over 0.9 dB in terms of video PSNR with respect to the standard transmission, while in its optimized version, the gain is over 1.5 dB PSNR, with a total overhead of about 12%.

**Keywords** Forward Error Correction (FEC) • Inter-vehicle communication networks • Inter-vehicle multimedia • Multimedia signal processing • VANETs

M. Petracca (✉)
Scuola Superiore Sant'Anna di Pisa, Pisa, Italy
e-mail: matteo.petracca@sssup.it

P. Bucciol
French-Mexican Laboratory of Informatics and Automatic Control (UMI LAFMIA 3175 CNRS), San Andrés Cholula, Puebla, Mexico
e-mail: paolo.bucciol@polito.it

A. Servetti • J.C. De Martin
Politecnico di Torino, Torino, Italy
e-mail: antonio.servetti@polito.it; demartin@polito.it

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_21, 

## 21.1 Introduction

The strong evolution of inter-vehicle communications in the Intelligent Transportation System (ITS) sector, along with the widespread adoption of portable devices equipped with IEEE 802.11 wireless interfaces, fosters the deployment of innovative wireless communication services based on the real-time streaming of multimedia flows. Inter-vehicle multimedia streaming has countless applications, ranging from safety services to collaborative driving and generic value-added services such as advertising and infotainment.

However, the high variability of intervehicle communication channels based on the IEEE 802.11 standard makes the transmission of real-time multimedia information a very challenging problem [1]. Among the main drawbacks of streaming applications over VANETs is the high percentage of packet losses which can be experienced over the wireless channel [2]. Even if multimedia information is tolerant to some packet losses, high losses do not actually allow the faithful reconstruction of the media with respect to its original version, thus not guaranteeing the quality necessary for object and speech recognition algorithms.

In this chapter, we address the problem of protecting real-time multimedia communications over intervehicle networks to guarantee the necessary quality for sophisticated multimedia signal processing techniques.

Let us consider the following scenario, depicted in Fig. 21.1, where a video-communication software is installed in two cars that are going over the same path. The front car is transmitting real-time video information to the back car. As cars move along the path, the wireless channel experiences noise due to environmental elements, thus suffering from multipath fading. It causes variable bit error rate, depending on several parameters such as the distance between the two cars, the presence of objects between the cars, and the relative speed. Certain combinations of these parameters can also generate very long bursts of packet losses resulting in intermittent connectivity. When it happens, the real-time transmission of the considered video flow is unfeasible, unless appropriate counteractions are taken.

Packet-level Forward Error Correction (FEC) techniques are able to recover packet losses without resorting to packet retransmission requests (which would generate too high delays in real-time constrained scenarios). Packets to be transmitted are grouped in blocks, and their loss can be recovered until the packet loss rate of a given block exceeds the percentage of redundant packets inserted. If made aware of the channel conditions, the sender can then adapt the percentage of FEC to match the actual channel conditions.

This mechanism works well with the assumption of uniformly distributed losses. However, VANET transmissions heavily suffer from burst losses that strongly impact the possibility to recover data packets belonging to such bursts. To overcome this problem, in this chapter, we resort to packet-level interleaving to split long consecutive error bursts into smaller lost packets sequences. With adequate constraints, we show that a joint FEC/interleaving technique is able to consistently improve the transmission quality while respecting real-time constraints. We then present the

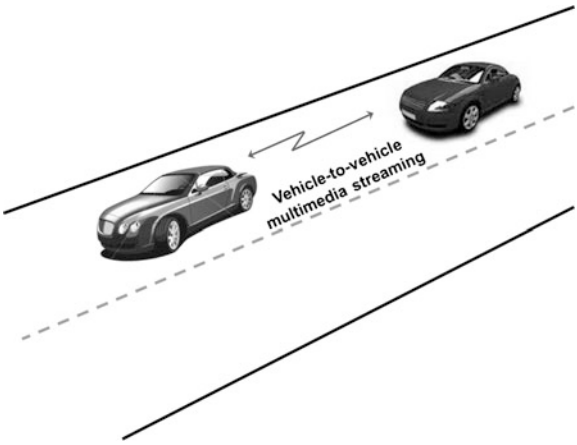



**Fig. 21.1** Multimedia streaming scenario in vehicular ad-hoc networks

proposed FEC and Interleaving Real-time protection technique (FIR) and its optimized version (FIRO) which dynamically adapts protection strength and transmission delay to channel variations. The proposed techniques are validated with simulations based on the Gilbert–Elliott wireless channel model, showing gains of up to more than 0.9 dB and 1.5 dB in PSNR with respect to plain transmission.

The remainder of the chapter is organized as follows: In Sect. 21.2, the principles of real-time multimedia streaming are presented. In Sects. 21.3 and 21.4, the building blocks of our solution are described, namely, the FEC and interleaving techniques, and nonadaptive optimal parameters are obtained. In Sect. 21.5, the proposed solution to the adaptive case is derived, both for the FIR and FIRO algorithms, and their performance evaluated with respect to isolated adaptive FEC, isolated adaptive interleaving, and plain transmission. Finally, Sect. 21.6 concludes the chapter.

## 21.2 Real-Time Multimedia Streaming

The requirements for real-time data transport mechanisms are distinctively different from those for traditional data communications. For example, real-time delivery requirements restrict the use of retransmissions to recover from packet losses so that the Transmission Control Protocol (TCP) is not suitable for this scenario. Instead, the Real-time Transport Protocol (RTP), specified in RFC 3550 [3], is the de facto standard for delivering data with real-time content over IP networks.

To enable real-time transmission and playout at the receiver, the RTP packet header carries sensitive information such as the sequence number and the timestamp. An RTP packet may contain one or more codec frames, with the sequence number incrementing by one for each packet sent and the timestamp increasing at the rate of

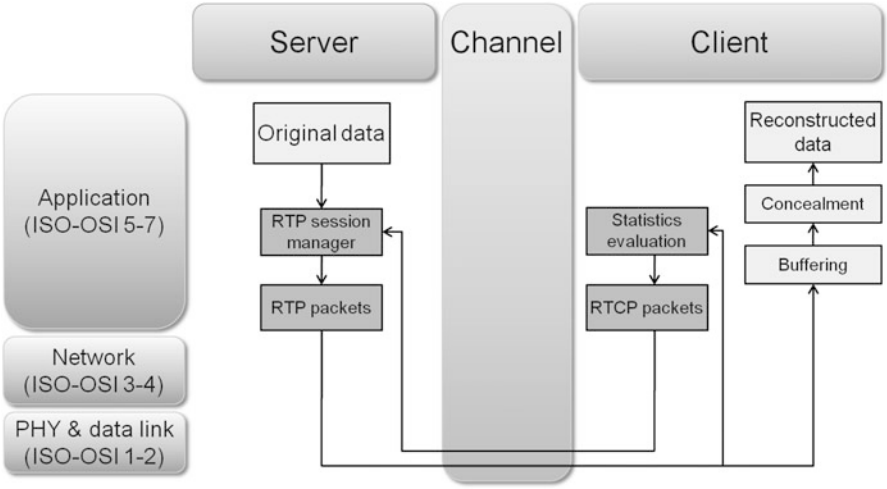



**Fig. 21.2** Real-time multimedia streaming by means of the RTP and RTCP protocols

the sampling clock. The RTP receiver uses the sequence number to detect lost packets and the timestamp field to determine when to play out received data.

The RTP Control Protocol (RTCP) [3] is used to monitor the quality of service and to convey information about the participants in an ongoing session. Basically, RTCP carries long-term statistic information (e.g., mean packet loss rate (PLR), round trip time, jitter, etc.) related to the RTP session participants. The full real-time multimedia streaming procedure is shown in Fig. 21.2.

In this work, we discuss how RTCP reports can support RTP transmission to track frequent variations of the wireless channel in order to provide the streaming server with regular feedbacks from the receiver on the suffered packet loss rate. Timely feedback is used, at the sender, to adapt the transmission policy to the channel characteristics in order to achieve the best video quality as perceived by the end user. Error control techniques are introduced to improve communication reliability against time-varying and bursty packet losses. In fact, IEEE 802.11 link-layer retransmissions are efficient only in a shorter timescale and in the face of short-term fluctuations (fast fading); more persistent fluctuations (slow fading) in a high-mobility scenario render these mechanisms inefficient. Application-level error control techniques may provide additional reliability on a longer timescale and, as described in the next sections, cross-layer integration can be exploited to regulate the trade-off between error control aggressiveness and transmission overhead according to the channel loss trends reported by the RTCP protocol.

## 21.3 Forward Error Correction

Generic forward error correction is a codec-independent method of protecting the information conveyed in data packets against packet erasures by adding redundant data to the transport stream. In this work, we use a common method for generating

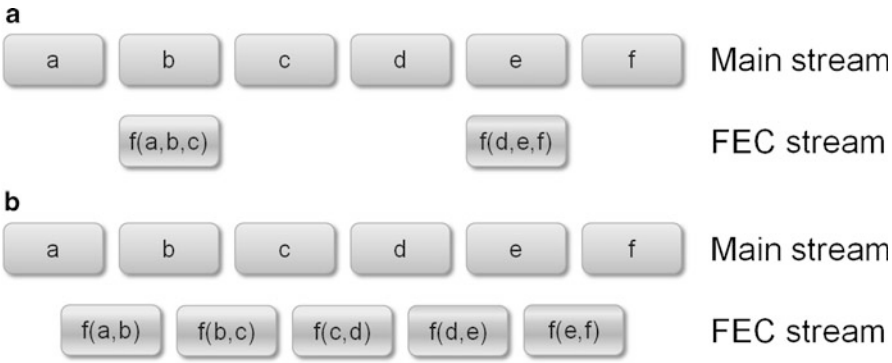



**Fig. 21.3** Two basic sample schemes using generic FEC as defined in RFC 5109

FEC data that takes a set of packet payloads and applies the binary exclusive or (XOR) operation across the payloads. This scheme allows the recovery of missing data in the case where *one* of the original packets is lost, but the FEC packet is received correctly. The RTP payload format for using generic FEC based on XOR operations has been published in RFC 5109 [4].

In recent years, several proposals have been made to use well-known error-correcting codes, such as Reed–Solomon [5] codes, for packet loss recovery as well. However, the weakness of the more complex schemes is the computational complexity, which may cause performance problems with long packets and a large number of parity packets. This is why we limit the scope of this chapter to XOR-based FEC codes only. Nevertheless, the basic principles discussed here can be easily extended to other kinds of linear codes.

Figure 21.3 shows two basic schemes using the generic FEC defined in RFC 5109. In this chapter, we adopt the definition of function *f(x, y, ...)* to denote the resulting FEC packet when the XOR operation is applied to the packets *x, y, ...* . In example (a), a single packet loss every three packets (in the original media stream) can be recovered, and in example (b), every packet loss can be recovered, assuming that the FEC stream is received correctly in both cases.

Clearly, both schemes require more network bandwidth because of the redundancy overhead. Example (a), that is denoted FEC 3:1, introduces an overhead of 33% since an FEC packet is sent every three data packets, while example (b), that is denoted FEC 1:1, introduces an overhead of 100%. In general, an FEC i:1 introduces an FEC packet for every i data packets, causing an overhead of *(100/i)*%.

In practice, the media stream and the FEC stream are usually transmitted using the same transport medium. This is why we cannot expect packet losses to occur only in the media stream as both streams are likely to suffer from similar error characteristics. In the network perspective, it is realistic to assume the media stream and the FEC stream to form a single stream containing both media and FEC packets. Given a sequence of media and FEC packets, we can easily see the variation in error recovery rates when we examine the residual media data loss rate after applying different kinds of FEC patterns to the sequence. In Fig. 21.4, we plot the packet loss rate at the network level for a real wireless inter-vehicle

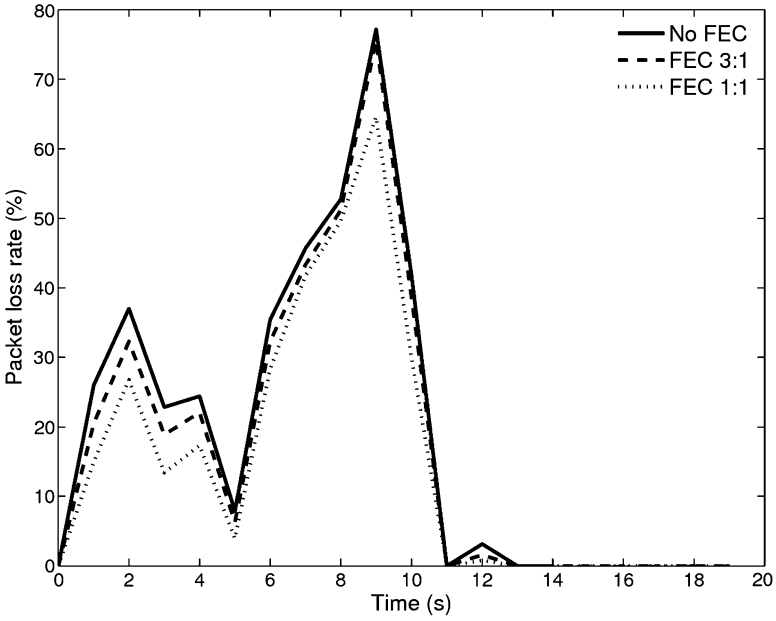



**Fig. 21.4** Application-level packet loss rate as a function of time for two generic XOR FEC schemes compared to the case of no FEC. FEC overhead is 100% for FEC 1:1 and 33% for FEC 3:1

transmission trace together with the application-level data loss rate for FEC examples (a) and (b). Clearly, the more overhead is introduced, the more media data loss rate decreases.

Nevertheless, the loss rate reduction is lower than expected. This is because the high packet loss rates of wireless transmission usually occur through correlated (adjacent) packet losses. In this case, loss distribution (i.e., loss pattern) is a key parameter that determines the FEC performance. Clustered losses considerably reduce the efficiency of FEC and decrease the decoding quality. It is clear that the packet loss rate at the application level does not depend only on the packet loss rate but also on *which* packets are lost.

A method that can be used to tackle this problem is to use interleaving to spread adjacent frames in different packets [6] as described in the next section.

## 21.4 Packet Interleaving

We explore a simple packet interleaving scheme to convert burst losses into an equivalent number of isolated losses which are easier to recover from using forward error control. Compared to other types of error resilience techniques, packet

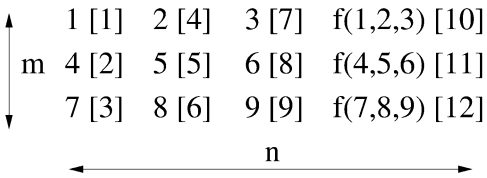



**Fig. 21.5** Packet interleaver with block size $n = 4$ and interleaving depth $m = 3$. Packets are transmitted by columns following the sequence numbers in brackets

interleaving provides the advantages of (1) being computationally simple and (2) not requiring any increase in bit rate. Furthermore, packet interleaving can easily be coupled with FEC techniques.

A potential drawback of packet interleaving is that it requires additional delay. Interleaving delay is of particular concern in high interactive applications, such as Internet telephony, that cannot tolerate a delay above 400 ms [7]. However, the required delay, which depends on channel burst length characteristics, can generally be bound to relatively short values, so even in this kind of applications, the end-to-end delay introduced by this technique is usually acceptable. Since many approaches for interleaving exist, we introduce the specific packet interleaving strategy used in this study.

A simple packet interleaver that permutes the packet transmission order is represented in Fig. 21.5. At the sender, packets are first written into the interleaver in rows, with each row corresponding to a block of $n$ packets; among them, $k = n - 1$ are data packets, and the last one is a XOR-based FEC packet. Then the packets are transmitted by columns as soon as $m$ rows of packets fill up. At the receiver, when packets are reordered using their timestamp and sequence number, loss bursts are converted into separated losses. Let us consider, for example, the case of a transmission channel afflicted by a burst loss of length three occurring during the transmission of the first three packets. Using the *(n, m)* interleaver shown in Fig. 21.5, the burst loss affects separated packets 1, 4, and 7 instead of successive packets 1, 2, and 3.

The effectiveness of the interleaver depends on the block size and the interleaving depth as well as the loss characteristics of the channel. With an interleaving depth of $m$, a burst loss of length $B$ can be converted into a shorter burst with a maximum length of $\lceil B/m \rceil$, where $\lceil x \rceil$ denotes the smallest integer not smaller than $x$. In an ideal case, when $m \geq B$, the burst loss can be converted into isolated losses.

In this case, the separation between any two losses is either $n$ or $n - 1$. A larger interleaver is more effective in that it can convert a longer burst loss into isolated losses or increase the separation of the converted isolated losses. However, this is at the cost of higher latency. At the client, an interleaved packet received cannot be used until all the packets it depends on are received. For an *(n, m)* interleaver, the nth packet in the original order suffers from the highest delay, as it has to be transmitted in the $((n - 1) \times m) + 1$th place. Hence, the decoding delay corresponding to an *(n, m)* interleaver is

$$(n - 1) \times m \tag{21.1}$$

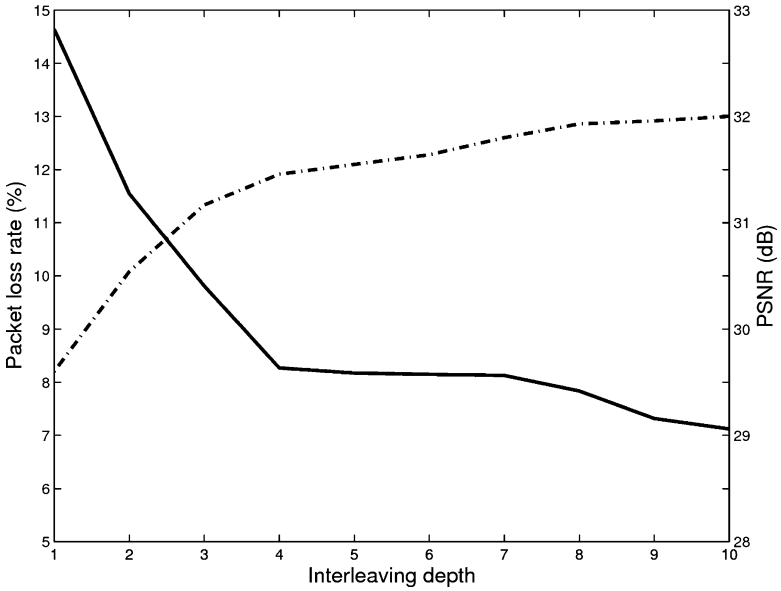



**Fig. 21.6** Application-level packet loss rate and PSNR (*dotted line*) as a function of the interleaver depth with FEC 1:1 for the *Foreman* sequence and network trace of Fig. 21.4

and a trade-off exists between the effectiveness in permuting the packets and the latency. It should be noted that the total delay here is not the typical $n \times m$ which arises in channel coding situations, since we do not have the delay of applying FEC across the entire interleaved data [8].

Figure 21.6 illustrates the advantage of using different interleaving lengths for the same FEC scheme in the real wireless intervehicle transmission trace shown in Fig. 21.4. It is observed that the interleaver leads to lower packet loss rate by converting the burst losses into isolated losses so that the XOR FEC scheme can effectively recover the missing data packets. The figure also shows the corresponding quality of the received video stream, measured by means of Peak Signal-to-Noise Ratio (PSNR). Note that at the network level, the total number of losses is the same in both cases; the difference is only in the pattern of the losses. In addition, we clearly see that after a certain interleaving depth, there is nearly no gain in increasing the interleaver depth. This is because the interleaving depth is equal or greater than the mean burst length of the network channel and that this value is large enough to benefit from burst loss spreading. In the next section, we determine the optimal combination of FEC redundancy and interleaver length *(n, m)* under certain application-related delay constraints.

## 21.5 Adaptive FEC and Interleaving Techniques

Researchers have been working for a long time to improve FEC-based error control mechanisms. The major research interest is still how to make the FEC code size adaptive instead of using a fixed FEC code under all communication environments.

Several works proposed adaptive FEC schemes that adjust the code size according to an optimization model based on the assumption that the packet loss in a network follows a Bernulli process [9], a Gilbert–Elliott model [10], etc. However, the method of employing fixed models to determine the characteristics of wireless channels works reasonably well for an environment where the end nodes are fixed or have low mobility. For an environment that changes dynamically in time and speed, finding an appropriate model is still a major research issue. So we propose an alternative solution, that is to use a feedback loop to determine the changing channel conditions and consequently to adjust the strength of the FEC code depending on the notification of corrupted packets at the receiver end. By means of RTCP reports the loss pattern at the network level is regularly sent back to the receiver, thus giving the streaming server the possibility to adapt the FEC strength along the video stream. The schematic implementation of the proposed closed-loop FEC and Interleaving Real-time protection technique (FIR) and its optimized version (FIRO) is shown in Fig. 21.7.

In the following of the chapter, the FIR and FIRO techniques are first presented; then, their performance is evaluated with respect to isolated adaptive FEC, isolated adaptive interleaving, and plain transmission. In all the simulations, the channel has been modeled with a 2-state Gilbert–Elliott model.

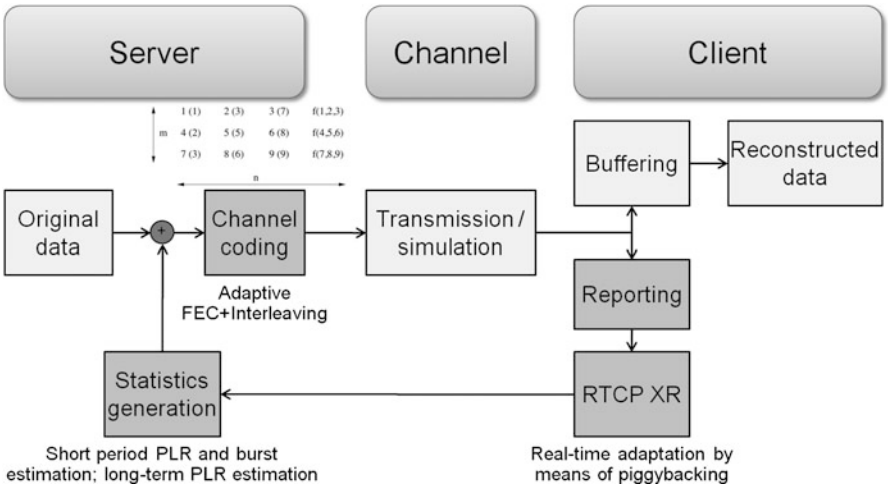



**Fig. 21.7** Implementation of the FEC and interleaving real-time protection technique

### 21.5.1 FEC and Interleaving Real-Time Protection Technique (FIR)

The proposed XOR-based adaptive FEC scheme uses the averaged loss rate $p$ reported periodically by RTCP to adjust the amount of redundancy (FEC) to be transmitted. XOR-based FEC protocol produces an additional redundancy packet from $k$ media packets, and it has the capacity to overcome a single packet loss over the $n = k + 1$ consecutive packets. This provides resiliency against a maximum packet loss rate of $p = 1/n$ when considering that even FEC packets may be affected by loss. Thus, based on the averaged packet loss rate measurements such as that provided by the RTCP feedback, it is possible to constantly adjust the redundancy amount by changing the number of media packets ($k$) covered by the FEC packet as follows:

$$k = \left\lfloor \frac{1}{p} \right\rfloor - 1 \tag{21.2}$$

The maximum acceptable loss rate threshold beyond which the streaming server triggers FEC adaptation may differ depending on the nature of the audiovisual content and its loss resiliency characteristics (e.g., according to Eq. 21.2, if the maximum threshold is set to 10%, the maximum value of $k$ has to be set to 9).

The other dimension of the interleaving matrix, i.e., the number of rows ($m$), depends on the overall delay that can be tolerated by the real-time application. The total end-to-end delay consists of three components: the codec delay, the network delay, and the playout delay. The latter is set according to the jitter introduced by the network transmission, and when interleaving is used, it should be increased so that it can accommodate also the interleaving delay. Playout buffer size is set by the receiver at the beginning of the transmission, before media decoding and, in the simplest scenarios, it is usually kept constant. So, if we denote it by $d_{po} = d_j + d_i$, where $d_j$ corresponds to the jitter component and $d_i$ to the interleaving component, the value of $m$ can be dynamically calculated from Eq. 21.1 as a function of $d_i$ and $n$ as

$$m = \left\lfloor \frac{d_i}{n-1} \right\rfloor \tag{21.3}$$

An additional issue that must be considered is that the FEC adaptation model poses a problem when dealing with channels that exhibit varying packet loss rates over time. The frequency of the receiver reports, which give to the sender an estimate about the network loss rate and other parameters, may reduce the responsiveness of the FEC scheme, leading to suboptimal FEC efficiency. A high frequency would enhance the responsiveness at the sender while causing high variations between successive measurements and possibly leading to instability, not to mention

excessive feedback traffic overhead. On the other hand, a low frequency would have good stability and low overhead, but poor responsiveness.

Every time the sender receives an RTCP packet with a report of the current PLR, it calculates the PLR estimate $p$ (also identified as long-term PLR) for the subsequent time interval $(\hat{p}(i))$ using the reported PLR value $(p(i-1))$ and the previous PLR estimate $(\hat{p}(i-1))$ according to

$$\hat{p}(i) = \hat{p}(i-1) \times \alpha + p(i-1) \times (1-\alpha) \tag{21.4}$$

where the value of the memory factor $\alpha$ has to be chosen in order to give a good noise reduction ratio while maintaining a reasonable rate of convergence.

The entire FIR algorithm can be summarized as follows:

```
1. Compute the max allowed delay for the interleaver
2. Define maximum allowed PLR
3. Set the long-term PLR equal to 0
4. Set the FEC and interleaver parameters equal to 0
   (no FEC and interleaver)
5. While sending packets
  5.1 If a RTCP report is received then
      5.1.1 Update the long-term PLR and other statistics
      5.1.2 If the longterm PLR is bigger than the maximum allowed PLR then
            5.1.2.1 Update FEC and interleaver parameters
   End
```

### 21.5.2 FEC and Interleaving Real-Time Optimization Protection Technique (FIRO)

In the FIR technique, the FEC and interleaver parameters are adjusted according to the experienced packet loss rate which is reported to the sender by the RTCP reports. The interarrival period of each RTCP report is considered fixed. This choice from one hand simplifies the FEC and interleaving adaptation algorithm, which does not depend from other variables, but on the other hand reduces the adaptation capability of the algorithm.

The FIRO algorithm extends its previous version by adding the possibility of changing the interarrival period of the RTCP reports. FIRO starts by setting an initial frequency of the RTCP interarrival period, by defining a minimum and maximum value for the interarrival period and the granularity of the variations (steps). The RTCP interarrival period is then updated based on the actual channel conditions. If there is the need of adapting more quickly to the channel variations or to monitor the channel with more accuracy, that is, if the current estimate is not accurate or if the long-term PLR is higher than the maximum allowed PLR, the time between two consecutive RTCP reports is decreased by one step until the minimum value of the interarrival period is reached. The interarrival period is increased by one step (so to decrease the frequency of the reports) until the maximum allowed

value is reached in all other cases. The new interarrival period is then appended as RTP extension to the next packet to be transmitted.

The entire FIRO algorithm can be summarized as follows:

```
1. Compute the max allowed delay for the interleaver
2. Define maximum allowed PLR
3. Define the PLR accuracy
4. Set the long-term PLR equal to 0
5. Set the FEC and interleaver parameters equal to 0
   (no FEC and interleaver)
6. While sending packets
       6.1 If a RTCP report is received then
                 6.1.1 Update the long-term PLR and other statistics
                 6.1.2 If the long term PLR is bigger than the maximum allowed PLR
                       then
                            6.1.2.1 Update FEC and interleaver parameters
                            6.1.2.2 Decrease, if allowed, the RTCP interarrival
                                    period
                 6.1.3 Else If the long-term PLR variation between its old and
                       current value is bigger than the PLR accuracy then
                            6.1.3.1 Decrease, if allowed, the RTCP interarrival
                                    period
                 6.1.4 Else
                            6.1.4.1 Increase, if allowed, the RTCP interarrival
                                     period
                 6.1.5 If the RTCP interarrival period has changed
                             6.1.5.1 Append the new RTCP interarrival period to
                                     the next data packet
   End
```

### *21.5.3 Performance Evaluation*

To test the performance of the FIR and FIRO algorithms, the transmission of a 65 s video stream between two cars in a highway scenario has been simulated. The input video stream [11] has been compressed with the H.264/AVC codec [12] at 30 frames/s, 9 packets/frame, 600 kbit/s. The channel has been modeled with a 2-state Gilbert–Elliott model, with average packet loss rate of 10% and average burst error length of 3 packets. The maximum allowed transmission delay has been set to 400 ms, according to [7], while the maximum allowed packet loss rate has been set to 5%. For the FIR algorithm, the RTCP interarrival time has been set equal to 1 s, while for the FIRO, it can vary from a minimum of 0.1 s to a maximum of 1 s with granularity of 0.1 s for increments/decrements. The received quality of the video stream has been evaluated by means of Peak Signal-to-Noise Ratio (PSNR). The performance of the following transmission techniques has been evaluated: plain transmission, adaptive FEC only (without interleaving), adaptive interleaving only (without FEC), FIR, and FIRO. Overall results are presented in Table 21.1.

The FIR algorithm outperforms a plain transmission and the adaptive techniques in which FEC and interleaving are used in isolation both in PLR and PSNR results. By resorting to interleaving, FIR outperforms the adaptive FEC technique in terms of reduction of the application-level PLR (−0.62%). The error bursts are split between multiple FEC blocks, allowing the FEC to be more effective. In terms of

**Table 21.1** Performance comparison between plain transmission, adaptive FEC, adaptive interleaving, FIR, and FIRO

| Transmission algorithm | PLR (%) [Δ] | PSNR (dB) [Δ] | Overhead (%) |
|---|---|---|---|
| Plain transmission | 9.98 [ ] | 37.88 [ ] | 0.00 |
| Adaptive FEC | 8.94 [−1.04] | 3.10 [+0.22] | 14.34 |
| Adaptive interleaving | 10.03 [+0.05] | 38.64 [+0.76] | 0.00 |
| FIR | 8.32 [−1.66] | 38.83 [+0.95] | 14.34 |
| FIRO | 6.18 [−3.80] | 39.41 [+1.53] | 11.51 |

perceived quality, the adaptive interleaving technique performs better than the adaptive FEC technique, even if its PLR performance is worse, since the error concealment algorithm of the video decoder can recover single packet losses more easily than burst losses.

The FIRO algorithm shows better performance with respect to its version without optimizations (FIR) both in PLR and PSNR. The use of variable RTCP interarrival time guarantees a much more effective adaptation to the channel conditions, which results in a reduction of about 2% in terms of PLR and a gain of about 0.5 dB with respect to the FIR algorithm. Moreover, FIRO shows a substantial reduction in the additional transmission overhead, which is equal to 11.51% with respect to a plain transmission and lower than the 14.34% experienced by the FIR algorithm.

## 21.6 Conclusions

This chapter discusses the implementation of adaptive communication techniques aimed at the proactive protection of multimedia streams by means of combined FEC and interleaving in the context of inter-vehicle communications. An adaptive technique, FIR, and its optimized version, FIRO, for the real-time transmission of loss-tolerant information flows targeted to V2V communication have been presented. The two techniques are based on two well-known packet-level error-resilience techniques and on periodic receiver feedbacks. FIR resorts to the periodic receiver reports sent at a fixed interarrival frequency to dynamic update the FEC and interleaving parameters, thus improving the communication quality at both network and application layer. FIRO is an optimized version of the FIR technique in which the interarrival frequency of the receiver reports is dynamically updated, thus improving performance both in PLR and PSNR. The proposed FEC and interleaving adaptation technique guarantees a gain of over 1.5 dB PSNR, with a total overhead of about 12%, in its optimized version.

**Acknowledgement** This work was supported in part by Regione Piemonte through the VICSUM project.

## References


1. Sun W, Yamaguchi H, Kusumoto S (2006) A study on performance evaluation of real-time data transmission on vehicular ad hoc networks. In: Proceedings IEEE mobile data management. 126–130
2. Blum JJ, Eskandarian A, Hoffman LJ (2004) Challenges of intervehicle ad hoc networks. IEEE Trans Intell Transportation Syst 5(4):347–351
3. Schulzrinne H, Casner S, Frederick R, Jacobson V (2003) RTP: A transport protocol for real-time applications – RFC 3550. IETF Network Working Group
4. Li A (2007) RTP payload format for generic forward error correction – RFC 5109. IETF Network Working Group
5. Reed IS, Solomon G (1960) Polynomial codes over certain finite fields. SIAM J Appl Math 8:300–304
6. Perkins C, Crowcroft J (2000) Effects of interleaving on RTP header compression. In: Proceedings of IEEE INFOCOM. 111–117
7. ITU-T (1993) Rec. G.114 One way transmission time. ITU-T Technical Report
8. Liang YJ, Apostolopoulos J G, Girod B (2002) Model-based delay-distortion optimization for video streaming using packet interleaving. In: Proceedings of Asilomar conference on signals, systems and computers. 2:1315–1319
9. Bolot J-C, Fosse-Parisis S, Towsley D (1999) Adaptive FEC-based error control for Internet telephony. In: Proceedings of IEEE INFOCOM. 3:1453–1460
10. Yao J, Huang W-F, Chen M-S (2006) DFEC: Dynamic forward error control for DVB-H. In: Proceedings of IEEE international conference acronym is SUTC
11. Acticom Reference Sequence: Highway (CIF). http://trace.eas.asu.edu/mirrors/h26l/1581.html. Accessed 1 Mar 2011
12. ITU-T H.264 & ISO/IEC 14496-10 AVC (2008) Advanced video coding for generic audiovisual services. ITU-T Technical. Report

# Chapter 22
# Matisse: A Large-Scale Multi-Agent System for Simulating Traffic Safety Scenarios

**Rym Zalila-Wenkstern, Travis L. Steel, Ovidiu Daescu, John H.L. Hansen, and Pinar Boyraz**

**Abstract** In this study, we discuss the high level architecture of MATISSE, a large-scale multi-agent system for simulating traffic safety and congestion scenarios. MATISSE includes three main components: the Agent–Environment System (AES) creates simulation instances where the environment is modeled as a graph, the Data Management System stores and processes the information collected from the AES, and the Visualization Framework provides 2D and 3D virtual representations of simulated entities.

**Keywords** Multi-agent systems • Simulation • Safety • Traffic management

## 22.1 Introduction

The root causes of traffic congestion have long been understood, and several strategies have been defined to address this problem [Dot07]. Transportation technologies known as Intelligent Transportation Systems (ITS) have been considered as possible solutions [1]. In this paper, we discuss Soteria,[1] a multilayered, integrated traffic super-infrastructure for safety enhancement and congestion reduction, and MATISSE, a tailor-made, large-scale multi-agent-based simulation system designed to support this infrastructure.

[1] In Greek mythology, Soteria is the spirit of safety, preservation, and deliverance from harm.

R. Zalila-Wenkstern (✉) • T.L. Steel • O. Daescu • P. Boyraz
University of Texas at Dallas, Richardson, TX, USA
e-mail: rymw@utdallas.edu; steel@utdallas.edu; daescu@utdallas.edu; boyraz.pinar@googlemail.com

J.H.L. Hansen
Center for Robust Speech Systems (CRSS), Department of Electrical Engineering
The University of Texas at Dallas, Richardson, TX 75080-3021, USA
e-mail: john.hansen@utdallas.edu

J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7_22, 

Several advanced traffic simulation tools have been implemented in the last decade (e.g., CORSIM [2], CONTRAM [3], CORFLO [4], PARAMICS [5]). These tools are based on a conventional top-down view of the traffic problem and produce models that are rigid and idealistic. In our work, we approach the traffic problem from a bottom-up perspective and consider the traffic system as a large set of small interacting autonomous entities. The global system behavior emerges from the behavior and interactions of the individual entities.

The rest of this paper is organized as follows: in Sect. 22.2, we give a brief overview of Soteria; in Sect. 22.3, we describe MATISSE's high level architecture; and in Sect. 22.4, we discuss a model execution through a case study.

## 22.2 Overview of the Soteria Super-Infrastructure

Soteria [6] is a novel super-infrastructure for improving safety and reducing congestion on roads and highways. This infrastructure aims at enforcing communication, interaction, and collaboration between all the stakeholders at the micro and macro levels.

The proposed super-infrastructure is based upon two underlying concepts:

- In order to manage the traffic environment efficiently, it is necessary to partition the physical space into smaller defined areas called *cells*.
- Each cell is assigned a physical entity called a *controller*. A cell controller is responsible for (1) autonomously managing and controlling a portion of the physical environment (i.e., cell) including vehicles and traffic lights and (2) notifying other controllers of changes that may affect their cells.

As shown in Fig. 22.1, our proposed super-infrastructure consists of three components:

- The *Cell Controller Infrastructure* consists of cell controllers equipped with interactive devices. The purpose of this infrastructure is to keep the Vehicle Infrastructure and the Traffic Flow Infrastructure up to date with respect to traffic and safety information.
- The *Context-Aware Intelligent (CAI) Vehicle Infrastructure* consists of vehicles equipped with devices that allow them to (1) monitor the driver's behavior in order to prevent possible accidents, (2) communicate with other vehicles, and (3) interact with cell controllers to obtain traffic information in real time.
- The *Traffic Flow Infrastructure* consists of three types of stationary traffic devices: traffic lights, traffic collection devices, and relay units. The purpose of this infrastructure is to improve safety and traffic flow on roads and highways by providing information about the physical traffic infrastructure and congestion condition.

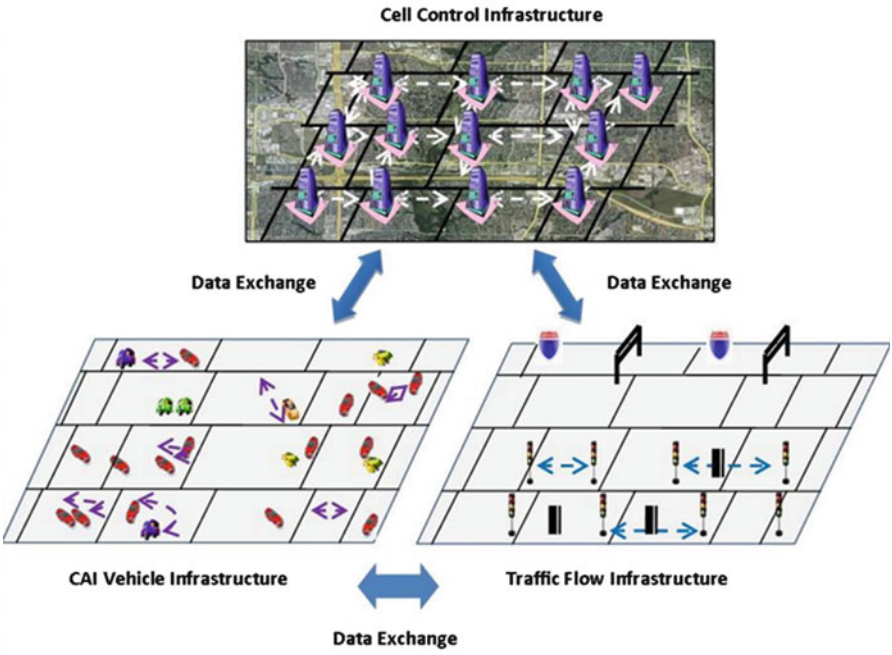



**Fig. 22.1** The super-infrastructure

## 22.3 Matisse: A Simulation Platform for Soteria

As mentioned in Sect. 22.1, MATISSE (**M**ulti-**A**gent based **T**raff**I**c **S**afety **S**imulation syst**E**m) is a "tailor-made" simulation framework[2] designed to specify and execute simulation models for Soteria. More precisely, it allows the simulation of various traffic safety improvement and congestion reduction scenarios at the macro level under nominal and hypothetical conditions.

MATISSE's artificial world consists of a large number of agents and a virtual environment. Agents can be of type *vehicle*, *traffic light*, or *information collection device* and are either mobile or stationary. In MATISSE, the environment is a bidirectional graph ($G$) in which nodes represent locations and edges represent paths between locations. Agents move in the environment using the map specified by $G$. Because of the dynamic and distributed features of the environment and due to the large amount of information exchanged between the agents and the environment, it is necessary to partition the space into a network of *cells*. Cell information is managed by individual *cell controllers*.

[2] The term "framework" refers to a system of systems.

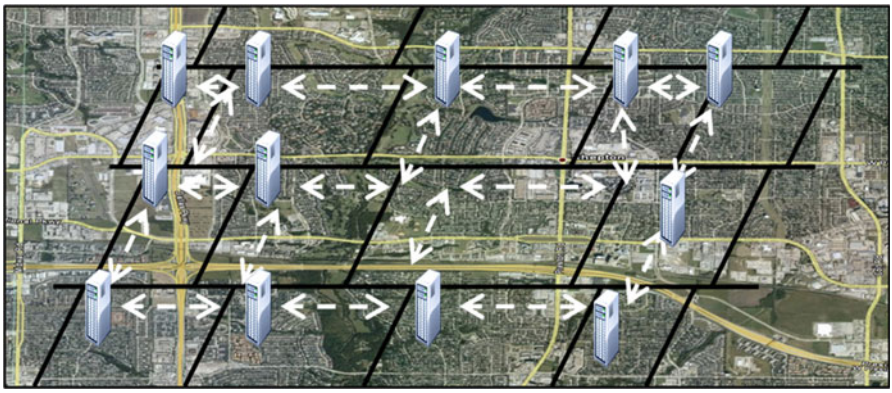


**Fig. 22.2** MATISSE cell structure

MATISSE's architecture is an extension of the DIVAs platform [7–9]. It includes three main components (see Fig. 22.3):

- The *Agent Environment System* (AES) creates simulation instances.
- The *Data Management System* (DMS) stores and processes information collected from the AES.
- The *Visualization Framework* receives information from the DMS to create 2D or 3D images of the simulation.

MATISSE's main constituent, i.e., the Agent Environment System, consists of three components:

- The *Context-Aware Intelligent Vehicle (CAI) platform* creates and manages mobile agents that represent vehicles.
- The *Traffic Device platform* creates and manages stationary agents that represent traffic lights and information collection devices.
- The *Environment platform* creates and manages the environment Fig. 22.2.

These platforms interact with one another through three *Message Transport Services*.

### *22.3.1 Agent Architecture*

In MATISSE, each agent, irrespective of its type, has an internal structure consisting of *interaction modules*, *information modules*, a *task module,* and a *planning and control module* (see Fig. 22.4). Concepts such as goals, tasks, and constraints are defined for each specific agent:

- *Interaction Modules*. An agent is able to perceive the environment through the *environment perception module*. It communicates with other agents through the *agent communication module*.

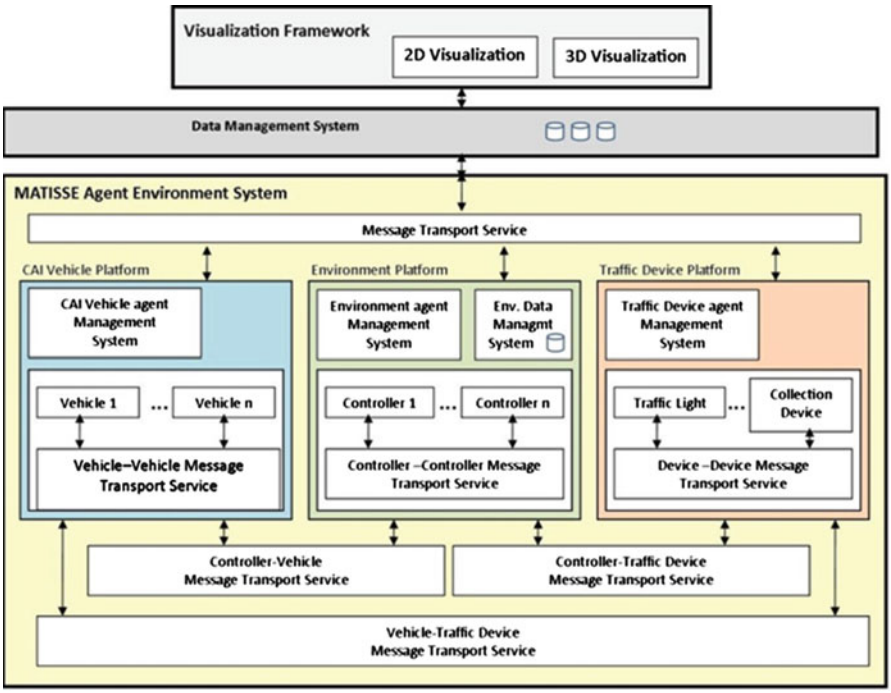



**Fig. 22.3** MATISSE high level architecture

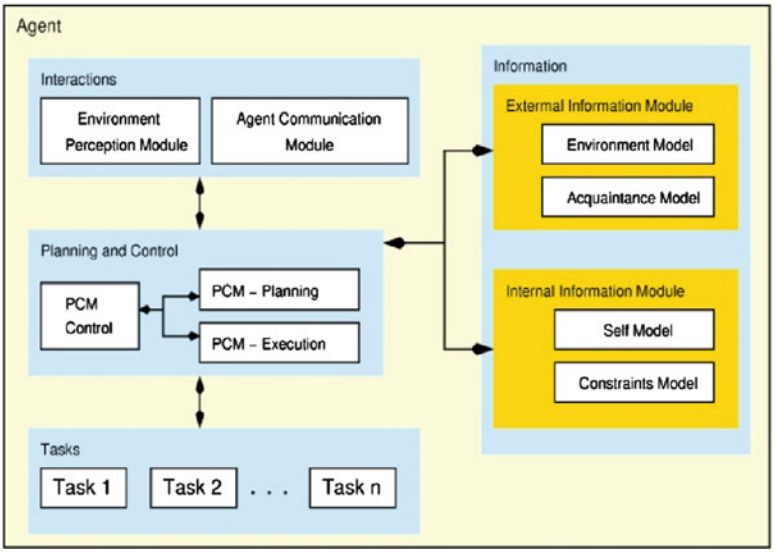



**Fig. 22.4** Agent architecture

- *Information Modules*. This is partitioned into External Information Module (EIM) and Internal Information Module (IIM). It serves as the portion of the agent's memory that is dedicated to maintaining knowledge about entities external to the agent. It consists of the *Environment Model* and the *Acquaintance Model*. The Environment Model is maintained according to the agent's perception of its environment, while the Acquaintance Model is maintained according to the agent's collaboration with other agents.
  It acts as the portion of the agent's memory that is dedicated for keeping information that the agent knows about itself. This module consists of the agent's *Self Model* and the *Constraint Model*. The Self Model maintains the essential properties of the agent, while the Constraint Model maintains the agent's physical and collaborative limitations.
- *Task Module*. This module manages the specification of the atomic tasks that the agent can perform in the domain in which it is being deployed. MATISSE allows the user to define these tasks or assign them from a library of predefined tasks.
- *Planning and Control Module*. This serves as the brain of the agent. It uses information provided by the other agent modules to plan, execute tasks, and make decisions.

## 22.4 Cell Controller Architecture

A cell controller is responsible for managing and controlling its own portion of the environment. It informs its local agents (e.g., vehicles, traffic lights) about changes in their surroundings and informs neighboring cells of any changes that may affect them. These characteristics reveal a strong correlation between the cell controller and the agent architectures; thus, it is clear that a cell controller can be modeled as a simple agent, as depicted in Fig. 22.5.

Similar to the agent architecture, the main four components of a cell controller are the *interaction modules*, the *information modules*, a t*ask module*, and the *planning and control module*.

- *Interaction Modules*. These modules handle asynchronous communication among cell controllers as well as synchronous communication between cell controllers and agents.
- *Information Modules*. These modules contain the data a controller needs to function. It is composed of the:
  - *Agent Model*. This model contains minimal information about the agents within the cell's environment region such as their identifiers and locations.
  - *Linked Cell Model*. This model maintains a list of neighboring cells whose graphs share a path with this cell's graph. Information such as cell identifiers and path identifiers of all shared paths are included in this model.
  - *Graph Model*. This model contains information regarding the nodes and edges contained within the cell.

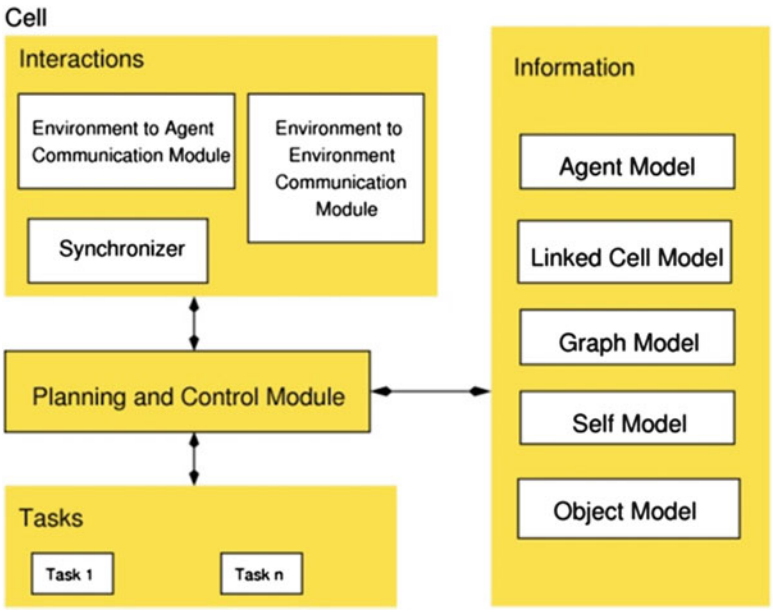



**Fig. 22.5** Cell controller architecture

- *Self Model*. This model contains information regarding the essential characteristics of the cell such as its identifier and region boundaries.
- *Object Model*. This model includes information detailing physical entities that are situated within the cell region but are not actual agents.

## 22.5 Case Study: Model Execution

In this section, we will consider the accident scenario depicted in Fig. 22.6. An accident has occurred in cell 12 (shown by the 'X'), and the vehicle onboard Collision Avoidance System informs cell controller C12 of the accident. Under this scenario, C12 immediately performs the following steps:

(a) Informs all the vehicles in the cell of the accident. The C12 cell controller achieves this by broadcasting an accident notification over the cell-controller-to-vehicle MTS to all vehicle agents currently stored in the cell's vehicle agent model.
(b) Informs adjacent cell controllers of the accident. The C12 cell controller sends accident notifications over the cell-controller-to-cell-controller MTS to neighboring cell stored in the linked cell model.
(c) Communicates with higher level controllers to obtain broader traffic information that is passed onto the vehicles. A hierarchy of cell controllers enables lower cell controllers to exchange information with cell controllers on a broader scale.


**Fig. 22.6** Accident scenario

All the vehicles in the cell make use of the broader traffic information to determine the best exit route (to avoid creating congestion on secondary streets).

Traffic lights communicate with other traffic lights to optimize traffic flow (e.g., approaching cars are not allowed to enter the cell) and may decide to turn green to allow traffic to flow.

The adjacent cell controllers, upon receipt of the accident notification, act in a similar manner as C12 to notify vehicles, traffic light controllers, and adjacent cell controllers of the accident.

## 22.6 Conclusion

In this paper, we discussed the high level architecture of MATISSE, a large-scale multi-agent system for the specification and execution of traffic safety scenarios. We approach the traffic simulation problem using a bottom-up approach, where the global system behavior is the result of the combination of individual micro-level behaviors. The design of MATISSE is based upon sound software engineering principles (i.e., separation of concerns, information hiding, and modularity). This leads to an extensible, reusable architecture.

## References

1. Research and Innovative Technology Administration, *Intelligent Transportation Systems*, available online at http://www.its.dot.gov/index.htm. Accessed 30 Apr 2009
2. CORSIM, Microscopic Traffic Simulation Model, available online at http://mctrans.ce.ufl.edu/featured/tsis/Version5/corsim.htm. Accessed 30 Apr 2009
3. CONTRAM, Continuous Traffic Assignment Model, available online at http://www.contram.com/. Accessed 30 Apr 2009
4. Lieu H, Santiago AJ, Kanaan A (1991) CORFLO: an integrated traffic simulation system for corridors. In: Proceedings of the engineering foundation conference, Palm Coast, Florida, 1–6 Apr 1991
5. Quadstone Paramics, Cutting Edge Microsimulation Software, available online at http://www.paramics-online.com/. Accessed 30 Apr 2009
6. Boyraz P, Daescu O, Fumagalli A, Hansen J, Wenkstern R Soteria: An integrated macro-micro transportation super-infrastructure system for management and safety. Technical Report, Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas, Feb 2009
7. Mili R, Oladimeji E, Steiner R DIVAs: Illustrating an abstract architecture for agent-environment simulation systems. *Multi agent and grid systems*, Special Issue on Agent-Oriented Software Development Methodologies, No. 4, vol 2, pp. 505–525, IOS Press, 2006
8. Mili R, Oladimeji E, Steiner R (2006) Architecture of the DIVAs simulation system. In: Proceedings of agent-directed simulation symposium ADS'06. Society for modeling and simulation, Huntsville, Alabama, USA, 2006
9. Steiner R, Leask G, Mili R (2006) An architecture for MAS simulation environments. In: Weyns D (ed) Proceedings of environments for multi-agent systems, (E4MAS'05), ACM conference on autonomous agents and multi agent systems, Utrecht, The Netherlands, 15–29 Jul 2005. Lecture notes in computer science, vol 3830. Springer, pp 50–67

# Index



J.H.L. Hansen et al. (eds.), *Digital Signal Processing for In-Vehicle Systems and Safety*,
DOI 10.1007/978-1-4419-9607-7, 