ORIGINAL RESEARCH PAPER

Color image segmentation in HSI space for automotive applications

Calin Rotaru · Thorsten Graf · Jianwei Zhang

Abstract This article presents a method for classifying color points for automotive applications in the Hue Saturation Intensity (HSI) Space based on the distances between their projections onto the SI plane. Firstly the HSI Space is analyzed in detail. Secondly the projection of image points from a typical automotive scene onto the SI plane is shown. The minimal classes relevant for driver assistance applications are derived. The requirements for the classification of the points into those classes are obtained. Several weighting functions are proposed and a fast form of an euclidean metric is investigated in detail. In order to improve the sensitivity of the weighting function, dynamic coefficients are introduced. It is shown how to compute them automatically in order to get optimal results for the classification. Finally some results of applying the metric to the sample images are shown and the conclusions are drawn.

Keywords Color image processing · HSI space · Image segmentation · Automotive applications · Driver assistance

C. Rotaru (✉) · T. Graf
Group Research Electronics,
Volkswagen AG,
38436 Wolfsburg, Germany
e-mail: calin.rotaru@rosoftware.com

T. Graf
e-mail: thorsten.graf@volkswagen.de

J. Zhang
Department of Computer Science,
University of Hamburg,
Vogt-Kölln-Straße 30, 22527 Hamburg, Germany
e-mail: zhang@informatik.uni-hamburg.de

1 Introduction

Color vision has been around for more than a few decades. Mostly due to hardware limitations and sensor costs the applications in the automotive domain remained very limited.

Nowadays both limitations are nearly overcome by the development of the hardware. The mobile phone market is driving down the costs for color imaging sensors and optical devices, while current embedded systems have gained computing powers which were until shortly hard to imagine except in personal computers.

In the near future, both hardware and sensor cost limitations will be outpaced and the color vision sensors together with the processing hardware will be largely available. This will make them a common choice for comfort oriented driver assistance applications. Furthermore, the automotive industry is increasingly faced with safety regulations. In order to comply with them, sensors able to perform classification and multisensorial approaches are required. In both these scenarios the color camera will be a better choice than a grayscale one, provided that the research and development of the processing algorithms will be able to keep pace with the market developments.

The current research on color vision for the automotive industry is limited. The major automotive research areas in which color vision has proved its specific advantages are detecting and interpreting the chromatic man made structures (like yellow markings, road signs, etc.), shadow detection and removal, object detection and detection and removal of the surrounding background. All of these methods share an image segmentation procedure in which the relevant image pixels are extracted from the image.

Detection and interpretation of the chromatic man made structures is a broad topic that includes traffic sign

recognition, information-sign recognition, yellow marking recognition, controlled environment segmentation. Most of these methods are based on the following two processing steps: (1) segmentation of the image in various color spaces (RGB, HSI and YUV are often used) and (2) the recognition of relevant objects using form and color information. The color segmentation of the scene is treated below, along with the object detection.

Shadow detection and removal remains a difficult task as long as the scene geometry, materials of the surfaces present in the scene and the complete characterization of the flux of light through the scene are not known in detail. In the automotive environment none of these prerequisites are met. Therefore distinguishing shadows from changes in material color or reflectance is a very challenging task. Solutions range from simple thresholding methods of the intensity information of the local image areas [18, 25] to using the color information and geometry. The color spaces frequently used are RGB (e.g. [8]) and HSI/V (e.g. [24]).

Object detection in automotive scenes using color is relatively new. Developing color segmentation algorithms that fulfill the real-time requirements of the automotive industry is difficult. The first step in recognizing vehicles moving on the road surface is to segment the image by classifying each pixel of an image into one of a discrete number of color classes. Image segmentation has been acknowledged to be one of the most difficult tasks in computer vision and image processing [5].

The most relevant approaches to accomplishing this task without considering position information include linear color thresholding, clustering, region-growing, nearest-neighbor classification, color space thresholding, histogram thresholding and probabilistic methods.

Linear color thresholding works by partitioning the color space with linear boundaries (e.g. planes in 3 dimensional spaces). A particular pixel is then classified according to the partition it lies in. This method is convenient for learning systems such as neural networks, or multivariate decision trees [2].

Clustering techniques [4, 6, 28] identify homogeneous clusters of points in the feature space (such as RGB color space, HSV color space, etc.) and then label each cluster as a different region. The homogeneity criterion is usually that of color similarity, i.e., the distance from one cluster to another cluster in the color feature space should be smaller than a threshold. The disadvantage of this method is that it does not consider local information between neighboring pixels.

Region growing algorithms such as seeded region growing [1] and the watershed transform [17] are widely used to solve image segmentation problems. They begin with a set of markers or seeds which are grown until each image pixel is allocated to one marker. This growth is controlled by a priority queue mechanism which is dependent on the statistics of the image (or its gradient) covered by a region.

Similar to clustering and region growing is the nearest-neighborhood classification. Typically several hundred preclassified samples are employed, each having a unique location in the color space and an associated classification. To classify a new pixel, a list of the K nearest samples is determined and the pixel is classified according to the largest proportion of classifications of the neighbors [3]. Both linear thresholding and nearest neighbor classification provide reliable results in terms of classification accuracy.

Yet another approach is to use a set of constant thresholds defining a color class as a rectangular block in the color space [7, 11]. This approach performs well, but is unable to take advantage of potential dependencies between the color space dimensions. A variant of the constant thresholding has been implemented in hardware by Newton Laboratories.

Another similar approach, using variable thresholds, is the histogram thresholding [15]. Histogram thresholding is one of the popular techniques for monochrome image segmentation [26]. This technique considers that an image consists of different regions corresponding to the gray level ranges. The histogram of an image can be separated using peaks (modes) corresponding to the different regions. A threshold value corresponding to the valley between two adjacent peaks can be used to separate these regions [22]. One of the weaknesses of this method is that it ignores the spatial relationship information of the pixels. The main advantage of this method is that it does not require a priori knowledge about the number of objects in the image. An algorithm for color image segmentation using local threshold values was studied in [19]. This technique divides an image into homogeneous regions by using a local threshold values. It calculates the threshold values automatically with the help of a merging process.

Finally a related approach is to store a discretized version of the entire joint-probability distribution [23]. For example, to check whether a particular pixel is a member of the color class, its individual color components are used as indices to a multidimensional array. When the location is looked up in the array the returned value indicates the probability of the membership to a particular class.

When the position information for the pixels is considered, then the following methods are also largely employed to segment color images: edge detection [13], region growing [1, 4, 27], neural networks [10, 12], markov random field models [14] and fuzzy logic [16]. Since this paper presents a low level segmentation approach, in which no position information is used, a detailed presentation of these higher level image segmentation techniques, as for example in [7], is beyond our scope.

Once the color image is segmented, the obtained regions are identified as objects depending on their form, position, motion or structure.

Background detection and subtraction are used in many driver assistance applications where two different approaches are encountered. The first is to "black-out" the detected background (render it insignificant by setting the pixels to a neutral value; black or white are widely used). Consequently, the algorithms process the whole image. The second approach is to use regions of interest in the algorithm definition so that the algorithm does not process the detected background. Background detection relies normally on multiple clues like texture, color, position in the image and so on.

The HSI space is often used in automotive applications. However, in most cases it is not investigated in detail for its suitability. The current state of the art solutions are mostly higher-level approaches, in which no analysis of the HSI space is performed. This article tries to fill the gap by providing an analysis of the HSI space in the automotive context, showing what particularities of the HSI spaces are to be taken into consideration for segmentation algorithms, and introducing a generic segmentation method based on intensity and saturation. This new method is analyzed in detail and the dynamic computation of its parameters is presented. In the end, the new method is compared to the linear color thresholding and nearest neighbor methods and the conclusions are drawn.

2 Characteristics of the HSI space

2.1 RGB to HSI transformation

The transformation from the source RGB space to the HSI space is performed with the already established formulas (R, G, B are floats in the range 0–1):

$$I = \frac{R + G + B}{3} \tag{1}$$

$$S = 1 - \frac{3}{R + G + B} \times \min(R, G, B) \tag{2}$$

$$H = \arccos\left(\frac{\frac{1}{2} \times [(R - G) + (R - B)]}{\sqrt{(R - G)^2 + (R - B)(G - B)}}\right) \tag{3}$$

if $B > G$, then $H = 2\pi - H$.

The conversion process from RGB to HSI is similar to a transformation from rectangular to polar coordinates. A new axis is placed in the RGB space between (0, 0, 0) and (1, 1, 1). This axis passes through all the achromatic points (i.e. those with $R = G = B$), and is therefore called the achromatic axis. A function $F(R, G, B)$ is then chosen

which calculates the brightness, luminance or lightness of color. The form chosen for F defines the shape of the iso-brightness surfaces. The iso-brightness surface K contains all the points with a brightness of F_k, i.e. all the points satisfying the relation $F(R, B, G) = F_k$. These iso-brightness surfaces are then projected onto a plane perpendicular to the achromatic axis and intersecting it at the origin, called the chromatic plane as it contains all the color information. The hue and saturation or chroma coordinates of each point are then determined within the plane, where the hue corresponds to the angular coordinate around the achromatic axis, and the saturation or chroma corresponds to a distance from the achromatic axis.

To visualize the shape of the resulting space, the points of each iso-brightness surface K are projected onto a chromatic plane intersecting the achromatic axis at K. The solid corresponding to a color space is obtained by merging all these projections. The form of this solid depends on the brightness function [9].

2.2 Form of the HSI space

The sensor of choice in typical automotive applications is an RGB sensor (CCD or CMOS) with an 8 bit signal for each channel, i.e. the RGB 24 bit format. The HSI image has to be obtained from the RGB one by applying the conversion formulas. Since the R, G, B elements are represented on 8 bit, they can take integer values in the range 0–255. Therefore the transformation to HSI coordinates will lead to a subset of the theoretical HSI color space. The form of this subset is relevant in order to understand its properties.

To plot the form of the HSI space, all 256^3 possible input triplets (R, G, B) were generated, scaled in float and then transformed to HSI using the formulas in (1)–(3). The three axes of the space are $X = S \times \cos(H)$, $Y = S \times \sin(H)$, $Z = I$.

In Fig. 1a the form of the HSI space projected on the ZY plane (Z axis is Intensity, Y axis is Saturation $\times$ sin(Hue)) is presented. Studying the Fig. 1a, two conclusions can be
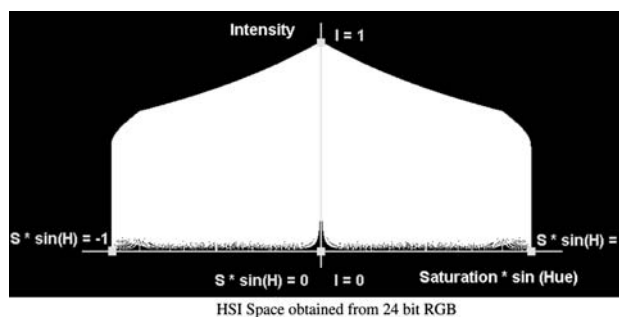

Fig. 1 Analysis of the HSI space form

drawn. In the upper part of the figure the values converge to the $I = 0$, $S = 0$ point. In the lower part the values are not converging to a defined point, but are spread in the theoretical cylindrical space. This characteristic of the HSI Space could be also formulated in the following terms: "for the lower intensity region, the saturation values do not converge to 0, but are exhibiting high values".

2.3 Specifics of the HSI space used

As seen from Fig. 1a, one characteristic is that for the upper half of the space (intensity values higher than 0, 5) the saturation will converge respecting the known conical representation. For the lower half of the space (intensity values lower than 0, 5) this property does not hold anymore and the saturation takes large values.

Due to the discretization effects, the obtained HSI points are sparse for the low intensity region (as it can be also seen from the lower part of the Fig. 1a). For example the RGB triplet (0, 1, 1) gives $S = 1$ (maximum possible value) while the triplet (1, 1, 1) gives 0 (minimum possible value). Both triplets represent very dark (almost black) image elements, that are practically indistinguishable for the human eye. Extreme changes in saturation are affecting only dark image areas. This is the other important characteristic of the HSI subspace obtained from 24 bit RGB images.

There are several ways to deal with these issues. One is to accept by default in the algorithms that if I is below a certain threshold, than the S component is either invalid or has higher values than expected. This leads to slightly more complicated algorithms, but their complexity remains manageable. Another solution is to replace the formulas in the low-intensity cases with predefined values for the saturation. This will not require any special handling in the algorithms, but the resulting space even if related to HSI, will not be identical to it, therefore the solution cannot be compared to the ones based on the HSI/HSV representations.

3 Projection of road scenes in HSI space

In order to be able to perform a segmentation in the HSI space, one has to answer at least the following questions:

1. Where in the HSI space are the features to be extracted located?
2. Which of the three dimensions of the HSI space are most relevant for the segmentation process?
3. Which functions can be used to determine the membership of a pixel to a class?
4. What is(are) the value(s) of threshold(s) in the segmentation process?

In order to obtain the answer to these questions some characteristics of a typical road scene in the HSI space were analyzed. The example used is the one in Fig. 2a.

The automotive environment specifics for an in-vehicle system (typical highway or country roads scene) are determined by the composition of the scene. There is a well defined infrastructure that is always present (environment). For example the road surface and lane markings are elements of the automotive environment. The actors (pedestrians, cars, trucks) present and moving in this environment are of capital importance for most driver assistance functions.

The next features characterize the overwhelming majority of automotive scenes:

– the road surface ranges from dark gray to light gray depending on the material and illumination (few exceptions can be found, for example dark red surfaces),
– the lane markings are white or yellow, usually reflective,
– the traffic signs are chromatic elements (even though there is no standard color across nations, they are almost always reflective and usually share vivid colors: blue, green, yellow, etc.),
– other infrastructure elements (road delimiters, markings, vegetation, buildings, etc.).

The obstacles present on the road relevant to driver-assistance functions are cars, trucks, motorcycles, bicycles, pedestrians, animals, etc. In this paper we focus on cars, trucks and motorcycles. Pedestrians and animals exhibit large variations in their aspect. A successful segmentation of pedestrians has to take into consideration the form, movement and position history in order be able to correctly recognize them. This would exceed the limits proposed for this article.

To summarize, the segmentation process should at least be able to distinguish the road and lane markings from possible obstacles (i.e. vehicles and the environment like vegetation and road signs). Vehicles exhibit a dark lower part because of tires and under-vehicle shadows. Furthermore they may also be chromatic, while the road infrastructure is usually achromatic and its intensity ranges from dark gray to bright white.

So the answer to question (1) where in the HSI space are the features to be extracted located?—in case of automotive applications can be "the relevant image pixels are usually having a low intensity or a high saturation".[1]

In order to be able to answer the question (2) which of the three dimensions of the space are most relevant to the segmentation process?—an analysis of the HSI components was

---

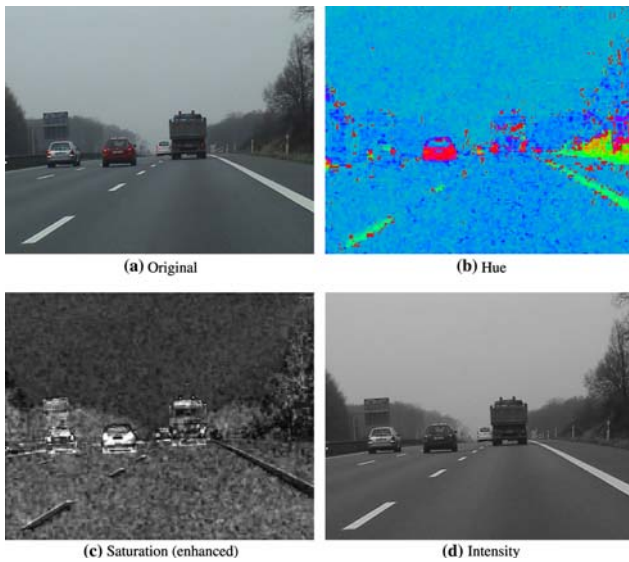[1] This is the same as saying that they either belong to chromatic elements or are very dark.

Fig. 2 Sample Image and its *H*, *S*, *I* components

conducted [20], including the intensity information, the saturation, the combination of saturation and intensity (SI). Finally, all three HSI components were analyzed. One image which was part of the investigation is presented in Fig. 2.

The following conclusions can be drawn by looking at Fig. 2 (for more details see [20]):

– intensity carries most information as can be seen in Fig. 2d. This comes as no surprise since most of the infrastructure is monochromatic and besides color humans rely on form and position to identify and track objects in their field of view.

– Saturation, even if well defined, has small values. The previous remarks from 3 about the saturated dark pixels are confirmed on the saturation image in Fig. 2c.
– Hue data from Fig. 2b alone is extremely hard to interpret even for a person that has already seen the original picture. Hue alone does not carry enough information. Combined with the saturation, the hue image would be more recognizable.

Hue values are describing color characteristics which vary greatly even between members of a same class of pixels as defined at the beginning of 3. For example saturation values are similar between a red and a blue car, but hue values are completely different. This is why hue is not a relevant feature for the classification methods that are the subject of this paper. Hue has to be used in higher level algorithms that have a basic knowledge of the structure of the environment (for example identification of the brake lights).

This situation is commonly encountered in automotive scenes. It is also one of the main reasons why color processing seems not to give any advantage over the more commonly encountered grayscale approaches. One dimensional histograms for intensity, saturation (as shown in Fig. 3a) and 2 dimensional histograms in the saturation—intensity plane (as shown in Fig. 3b) were built. The histograms are analyzed in order to answer the question: "is it possible to isolate specific effects on the histograms due to the vehicles present in the image?" Briefly, the conclusions drawn are:

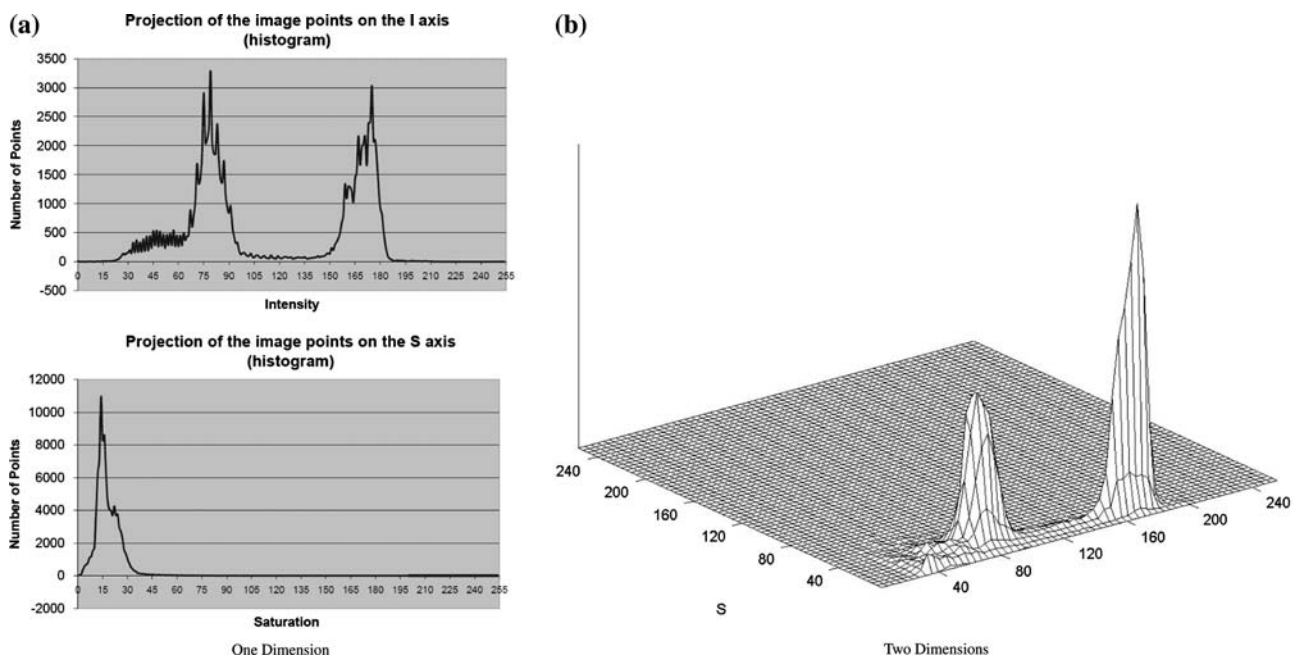Intensity values alone (without position information) cannot be used to make a segmentation of the image that is

Fig. 3 Histograms of the intensity and saturation of the image pixels

able to separate the objects from the road and the lane markings, even if certain information is available. In few particular cases when all objects are significantly brighter than the road and darker than the markings, the segmentation may yield some results, but generally the bright colored cars will mix with the lane markings while the dark ones will mix with the road. This problem can be solved using the position information or environment models, but this would be too complex for a low-level segmentation method.

The same problems arise with the projection on the S axis. Moreover, all saturation values are relatively small. Saturation values alone are noisy and that they are hard to interpret. Under better lighting conditions, they may improve. Saturation seems to carry less information than intensity.

The situation on the SI plane is easier to interpret 3. There are two major peaks corresponding to the road and the sky (most of the time the lane markings generate a separate third peak as well). The rest of the small histogram peaks correspond to other elements in the image: vegetation and cars. If the image has a high contrast, the two peaks are distant in the SI plane. If the image has a low contrast the two peaks are close in the SI plane. In all cases tested (all day-light traffic scenes) the two peaks were present.

This implies that given a proper discrimination function, one pixel could be classified as belonging to the road if the average road S and I values are known and the pixel has a minimum distance to the peak generated by the average values (with respect to the I and S axis). Both peaks resemble cones. This suggests that the corresponding color classes of the image elements in the SI plane may have circular forms (i.e. a central point and a given radius). If this holds, the separation into classes could be done by using a metric computing the distance to the center and comparing it with the radius of the disc.

The answer to the question (3) which functions can be used to determine the membership of a pixel to a class?— may read: the function used to make such a pixel classification based on its S and I value can be a metric since it computes the distance between the two points (current and peak average) in the SI plane.

For example, given the euclidean metric, a point in the plane (center of the disc) and a distance will define a class. This class is the disc including all points situated closer to the center than the given distance.

The answer to question (4) depends on the chosen segmentation function and will be only given in Sect. 5.

4 Weighting functions for automotive applications

In 1–3 the I, S and H components are represented as floats. This is inconvenient because the float representation uses

3×4 bytes per pixel and the computations involving floats are slower than the ones involving integers. Therefore in what follows we scaled the values such that I and S are represented in the range 0–255 (8 bit) and H in 0–65,535 (16 bit). Accordingly, the HSI value of a pixel can be represented on 4 bytes and processing algorithms can be written using only integer arithmetic, improving their performance. In the following formulas the integer representations are used and correspondingly all comments will refer to values in these ranges.

4.1 Metrics in the SI plane

As discussed in Sect. 3, cars, lane markings, road and sky generate clusters of points in the SI plane. This makes it possible to classify the image points, based on their distance to the class average in the SI plane, into the classes.

This subsection presents several weighting functions[2] based on their suitability for the segmentation of image points. It also discusses the suitability of each weighting function and assesses its usability by means of segmentation accuracy and computational complexity.

The euclidean metric can be used in the SI plane. One point is chosen and the distances for others points are computed with respect to its SI values (S_p, I_p) using the formula (4). Choosing the reference point is one of the key aspects when applying the metric. In our case the reference point can belong neither to the road (S_r, I_r) nor to the markings (S_m, I_m), because they both have to be distinguished from the relevant objects. One solution would be to use a combination of both as in (5) or to choose a point that normally belongs to cars (for example the high saturation, average intensity point that has $S = 255$, $I = 128$).

$$F_1 = \sqrt{(S - S_p)^2 + (I - I_p)^2} \qquad (4)$$

$$F_2 = \min(F_1(S_r, I_r), F_1(S_m, I_m)) \qquad (5)$$

The semimetric in (6) is derived from the euclidean metric. Since the values for S and I range between 0 and 255 the square root can be replaced (the accuracy decreases) by division with an integer. The smallest value scaling all possible output values in the interval given by [0–255] is $255^2 \times 2/255 = 510$. In order to obtain an acceptable resolution for the output values, the divider has to be computed dynamically as detailed in Sect. 5.

---

[2] The metric term is used here in a relaxed notation, without giving both points as function parameters. The set for all metrics is the integer SI plane, i.e. 2-tuples of S and I values where both S and I take integer values between 0 and 255. If the weighting function has the mathematical properties of a metric or semimetric, then it is called so. If not, then it will be simply called "weighting function".

$$F_3 = \frac{(S - S_p)^2 + (I - I_p)^2}{\text{Divider}} \tag{6}$$

One weighting function which is particularly efficient for the extraction of lane markings is (7). It was obtained starting with F_3. After setting $S_p = 0$ and $I_p = 255$, $I - I_p$ was reversed to get both terms always positive. The addition was replaced by multiplication to improved the sensitivity even further. It performs well when S has very low and I relatively high values; white lane markings are the image elements possessing this property.

For the extraction of yellow lane markings the function in 7 can be changed to (8) where S_y is the typical saturation for the yellow markings. The accuracy of the classification depends on the strength of the yellow footprint in the image (related to the reflectiveness of the marking), the direction of the light and day/night conditions. Several lane marking points were chosen for each frame and their averaged S values were used to compute the S_y parameter.

$$F_4 = \frac{(255 - I) \times S}{\text{Divider}} \tag{7}$$

$$F_5 = \frac{\text{abs}(S_y - S) \times (255 - I)}{\text{Divider}} \tag{8}$$

Shadows have a small impact on the performance of the presented weighting functions unless combined with direct illumination (sunrise/sunset in front of the camera). In this case, the contrast of the scene decreases significantly and the saturation information is hardly usable. In all other cases, the saturation values change significantly for shadowed areas, while the change in intensity is limited and may also happen gradually. Applying a metric that equally weighs the S and I components makes the elimination of shadows by a simple thresholding impossible. A solution was found by applying a semimetric that weighs S and I differently as in (9). Since S carries the most significant changes, it should be weighted the most.

$$F_6 = \frac{W_1 \times (S - S_p)^2 + W_2 \times (I - I_p)^2}{\text{Divider} \times (W_1 + W_2)}, \quad W_1 > W_2 \tag{9}$$

The semimetric F_6 (9) performs much better for the extraction of road/lane marking information than any of the presented metrics for the case of strong shadows. Still the problem remains partially unsolved. The problem of shadows can be only solved by using a high level representation of the environment.

4.2 Consideration of the characteristics of the used HSI subspace

The class of pixels that is most interesting for this work is the one that contains dark and chromatic elements as explained in 3. In this paragraph the behavior of the weighting functions is analyzed for the elements of this class.

It is important to understand how the weighting functions perform in case of lower intensity values (both HSI space characteristics presented in 3 are related to lower intensity values). F_3 is analyzed here, since it was the one most used in this work. In order to be able to compute the value of F_3, one has to choose the reference values for S_p and I_p. The analysis is done regardless of these values.

In the case of chromatic elements in the term $(S - S_p)^2$ from F_3, S has high values.

In practice, there are very few image points in which the three RGB components have the same values and therefore saturation equal to 0. In the case of very dark elements the saturation values are high as seen from 3. In this case, in the term $(S - S_p)^2$ in F_3, S has high values.

In both cases S in the term $(S - S_p)^2$ exhibits high values and therefore the output of F_3 for chromatic and dark elements is in the upper range of all possible results. E.g. for $S_p = 0$, $I_p = 0$ the point $(R, G, B) = (0, 3, 0)$ yields $(S - S_p)^2 + (I - I_p)^2 = 255^2 + 1^2 = 65{,}026$. A similar value is obtained for the point $(R, G, B) = (20, 120, 10)$, that is $(S - S_p)^2 + (I - I_p)^2 = 204^2 + 50^2 = 44{,}116$. If one takes the square root, these distances correspond to about 255, respectively 210. This is a significant result, because it demonstrates that the classification of both relevant image features can be done with a single function.

4.3 Improving the metric F_3

The semimetric F_3 was designed for areas that differ from the road or lane markings. Due to shadows present under the car, the semimetric usually succeeds in extracting an almost horizontal area at the bottom of the cars. The body of the car is not always extracted due to the way the reference point was chosen and its distant SI values from the tires.

An example of applying this metric relative to the reference point having $S = 255$, $I = 128$ is shown in Fig. 4, where all values lower than 170 were plotted. It can be observed that the metric extracts a significant number of points from objects and the nearby landscape, while only isolated points from the road or lane markings are extracted.

Another example is presented in Fig. 5b; the original image is presented in Fig. 5a. In this example the results for traffic signs can be accurately observed: most of the blue points are extracted. The function parameters were computed dynamically as exemplified in Sect. 5.

A greater difficulty is posed by question (4): what is(are) the value(s) of threshold(s) in the segmentation process? One possibility is to use the threshold which eliminates

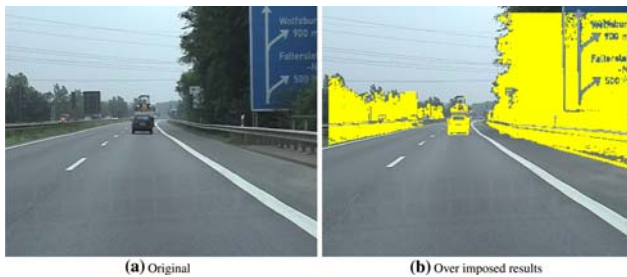Fig. 4 Over imposed results using the weighting function F_3

(a) Original   (b) Over imposed results

Fig. 5 Over imposed results using the weighting function F_3

both road and lane markings, i.e. to compute the threshold from the average SI values of the road.

Another alternative is to use the reference point on an already detected tire or car body. This will decrease the number of points belonging to the outer environment, but will increase the number of points detected in the lower part of the vehicle. This is how the metric was applied in object detection algorithms. Composed weighting functions (best result of more metrics) even if very promising from the researcher's point of view, have high computational cost. For example, a weighting function composed of two sub-functions will require at least twice as much computing power as the fastest of the sub-functions since both sub-functions have to be computed for each pixel.

The reference point and the divider are the parameters of the metric F_3. In order to improve its sensitivity the two parameters can be dynamically computed for every frame. The threshold used to separate the two point classes can also be adjusted dynamically to counteract the frequent changes in the illumination typical for automotive scenes. Section 5 deals with this question.

5 Adaptive SI metric coefficients

The parameterizable elements for the function F_3 are:

- the reference point (center of measurement),
- the divider (e.g. for F_3 510 brings all output values within the range 0–255 but also compresses the areas of interest within this range too much),
- the threshold which has to be used to decide if a point belongs to a class or not.

The new form of the metric F_3 from (6) is show in (10). The main differences to (6) are:

- the addition of the reference point for the class to the function definition (in order to obtain consistent results, the reference point was computed once per frame and remained the same for all pixels of the frame),
- the presence of the divider as a variable quantity (also computed once for each frame).

$$F_3(S, I, S_p, I_p) = \frac{(S - S_p)^2 + (I - I_p)^2}{\text{Dynamic divider}}. \quad (10)$$

Besides computing the divider dynamically, the threshold can also be determined dynamically. The algorithm requires two points (in SI plane). The first point should be chosen in such a way that it becomes a positive match from the metric (an "insider"), while the second should be chosen outside the relevant area in the SI plane (an "outsider").

Defining the "insider" and "outsider" points should pose no problem since driver assistance systems based on video identify the road and the lane markings. Therefore the road detection algorithm delivers a road region that can be used to allow for a selection of average road S, I values. Selecting an "outsider" would then be the same as specifying the average road values. Specifying an "insider" has to do with the scope of the detection algorithm which uses the metric. If a reference point for a lane marking is needed, then the point should be chosen to belong to a lane marking. The object detection algorithms will typically specify the point on the tires of the car (a common image characteristic of all vehicles).

Having obtained both reference points, the metric is computed in such a way that the distance between the "insider" and "outsider" values is maximal (255). This is the same as saying that $F_3(S_o, I_o, S_i, I_i) = 255$. Factoring out the divider and replacing the metric value with 255 in 10, it yields the computation formula for the divider (11).

$$\text{Divider}(S_o, I_o, S_i, I_i) = \frac{(S_i - S_o)^2 + (I_i - I_o)^2}{255} \quad (11)$$

In (11) the symbols mean: S_i, I_i saturation and intensity of the "insider" point. S_o, I_o for the pixel corresponding to the

"outsider" (typically average values of the detected road surface).

The threshold defines the outer boundary of the positive matches. For each point that is evaluated the result obtained by apply the formula 10 to its S, I values is compared with the threshold. If it is smaller, than the point is a positive match. In order to compute the threshold, one has be able to obtain the "worst outsider". In other words one has to compute 12:

$$\text{Threshold} = \min(\{F_3(S, I, S_p, I_p)\})$$
$$\forall (S, I) \in \text{"outsiders"} \tag{12}$$

The final segmentation step in which the threshold is compared to the current values, is a separate logical step. It is also possible to use different thresholds in different image areas. For example one may use a different threshold in brighter areas. In this case the threshold has to be computed using a subset of "outsiders" belonging to that region.

One may use statistical methods to find out an average for a specific scene type (for example daylight, normal contrast and brightness). Analyzing several daylight scenes the authors obtained good results by using a threshold placed at 30% of the distance between the "outsider" (255) and the "insider" (0).

6 Comparison with other segmentation algorithms

The image segmentation based on SI metrics is not using any position information when sorting the pixels into classes. Therefore it is directly comparable with the linear color thresholding, nearest-neighbor classification, color space thresholding and probabilistic methods that do not require the position of the pixels in the image during the segmentation.

We compare with linear color thresholding and nearest-neighbor. Color space thresholding is a generalized version of the linear color thresholding. Since hue information available in typical automotive scenes is not directly related to a certain class, this method will deliver similar results to linear color thresholding. Using the discretized version of the entire joint probability distribution (probabilistic methods) allows to model any segmentation method. For example, if the F_3 metric is used for the computation of the probability distribution, then the method delivers the same results as our segmentation based on SI metrics. The memory requirements to store the entire lookup table and the computing resources required to update it, make the last method unsuitable for real time image processing implementations in the automotive domain and therefore also for the direct comparison.

For the linear color thresholding, two classes were defined. One contains the road and the lane markings and the second contains everything else (all relevant objects, etc.) Of course, it is possible that the bright objects are classified as belonging to the wrong class, but such problems are common to all methods that rely solely on the color information. The hue information is not relevant for the road, therefore is ignored for the classification. In order to improve the results of the classification the boundaries of the classes were automatically computed using the results of the road and lane detection (for example the minimum road intensity of the analyzed region.

For the nearest-neighbor classification the preclassified exemplars were automatically updated (road, lane marking, object detection algorithms have provided the required values). In order to compensate for situations in which no objects were detected (therefore no exemplars could be provided for the object class), some predefined (manually obtained from images in the sequence) values were also used. The distance to the preclassified exemplars was computed using the euclidean metric.

Before the results are presented, one remark has to be made. Due to performance reasons, the SI segmentation was implemented only from and up to 8 pixels distance from the image margins. The other two methods were implemented to process the complete image.

The results from Figs. 6, 7, 8, 9, 10 are ordered in the following manner: original image (top-left), color segmentation based on SI metrics using F_3 (top-right, marked with yellow), nearest-neighbor (bottom-left, marked with cyan) and linear color thresholding (bottom-right, marked with magenta).

The scene in Fig. 6 is a low contrast scene. Few observations can be done without looking into detail:
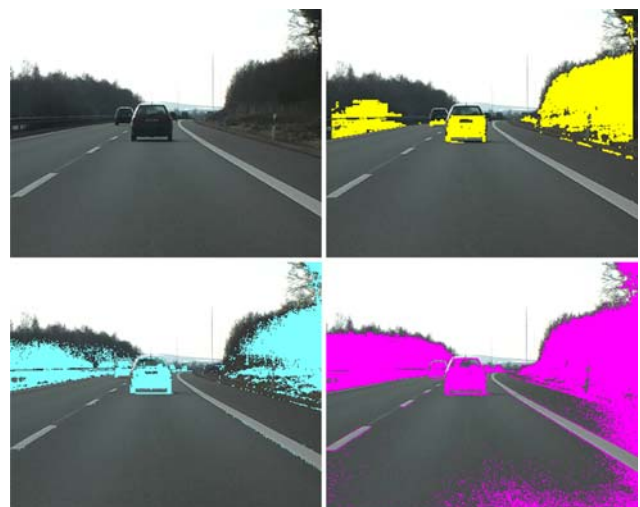

Fig. 6 Comparison with other segmentation algorithms—low contrast scene

Fig. 7 Detail comparison—low contrast scene

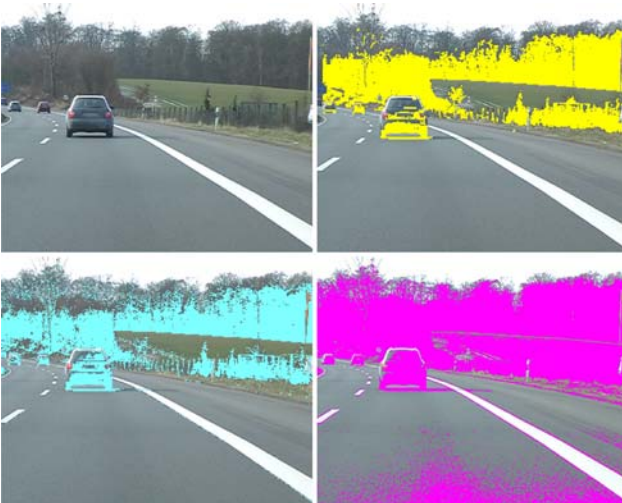**Fig. 8** Comparison with other segmentation algorithms—close scene

Fig. 9 Comparison with other segmentation algorithms—far objects

– the linear thresholding is wrongly classifying road pixels as relevant object pixels,
– the SI metric and nearest-neighbor have similar results. The SI metric classifies less from the shadow under the closest car as relevant object pixels,
– the results of the SI metric are denser in most areas as the one of the nearest-neighbor algorithm.

If the results are investigated into detail as in Fig. 7, the following supplementary observations can be done:

– the SI metric segmentation provides for the automotive use the best selectivity in the far regions. Sometimes
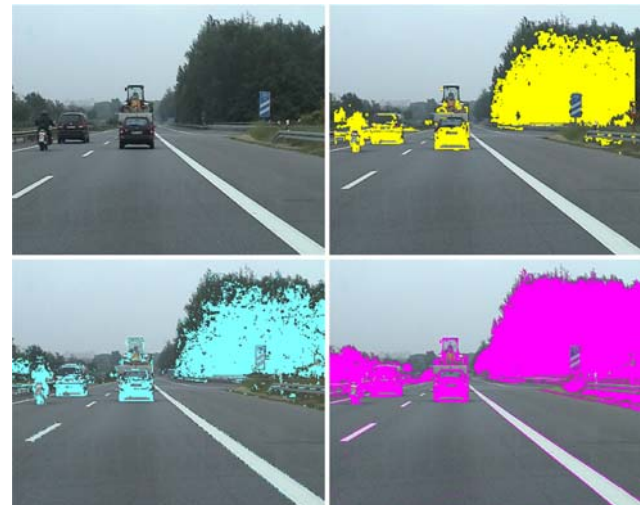

Fig. 10 Comparison with other segmentation algorithms—different objects

there is a price to pay, for example in this case the price is the missing detection of the distant car on the right side of the detailed image. It is filtered out by the SI segmentation, while the other two algorithms extract relevant pixels in that area,
– the linear color thresholding incorrectly classifies many dark points near the lane markings as relevant object points.

The comparison in Fig. 8 shows the behavior of the algorithms in the case of near objects and shadows. The following observations hold:

– all algorithms are experiencing problem with strong cast shadows,
– the linear color segmentation classifies many road and shadowed areas as object relevant pixels. This can be improved using a more constrictive threshold (the one used is obtained from the road detection algorithm and is influenced for example by bright road regions).

In Fig. 9 is presented the situation for far objects. One characteristic of the far objects is that their shadows are not visible in the picture and therefore the segmentation algorithms will accurately detect the intersection line of the object base with the road. The tires can also be distinguished easily. The following observations can be done:

– all algorithms work properly, extracting at least the lower part of the cars,
– for the far objects, the algorithm based on SI metric works less satisfying than the other two.

The last example is presented in Fig. 10. This image sequence was taken with another camera having a single

CCD imager as opposed to the other sequences which were acquired using a 3 CCD camera. There are two observations to be done:

– the nearest-neighbor and the linear color thresholding extract many dark points near the lane markings as relevant object points. In this example, the lane marking on the right is practically doubled by a line of wrongly classified points by the two algorithms,
– due to the fact that there was no direct illumination of the scene, the linear color thresholding has less classification errors in the road regions,
– the linear color thresholding has difficulties in extracting the motorbike driver from the background. The other two algorithms are working fine in this case.

Summing up the following conclusions can be drawn based on these examples and others that were investigated for this work, but not presented here:

– the linear color thresholding is not able to correctly separate between object, shadows and road pixels. It extracts often pixels corresponding to road, shadow and background as significant pixels. Its weakness is easy to link to the way the color space is divided for the classification. Sections using planes generates rectangular structures in the SI plane. Their form is not matching the conical shapes that are characteristic for elements of the automotive scenes.
– Nearest-neighbor classification works almost as good as the classification using the SI metric. Since the nearest-neighbor algorithm computes the distance to the preclassified exemplars by using the euclidean metric, it can be sometimes very similar to the SI metric. If the preclassified exemplars for the irrelevant class are given by the average road, lane marking and sky values then this method is founded on the same basis as the one based on the SI metric. Unfortunately the costs of going over the list with preclassified exemplars for all pixels, make this method significantly slower that the other two.
– The classification employing the SI metric performs in most automotive scenes better than the other two methods. The weakness that was identified (detection of the far objects) is related to the fact that in those cases the saturation exhibits small values (due to the small quantity of reflected light, the sensitivity of the imager is limiting the color definition). The intensity is mainly defining the object footprint in the picture. Since the F_3 metric used is weighting both saturation and intensity equally, its sensitivity is in these cases almost cut in half. This weakness should be compensated either by using a specialized detection algorithm for far objects or by employing a different threshold in image regions that are containing far objects.

7 Conclusion

We demonstrated that the SI plane of a color image contains significant information for automotive applications. Under normal conditions in automotive scenarios this information can be used to perform a segmentation of the image in such a way that possible obstacles (cars, trucks, vegetation, etc.) can be distinguished from the road background as clusters of points.

This article analyzed the HSI space, showing which components are relevant for image segmentation. It introduced several weighting functions to be used during segmentation. In order to improve the sensibility of the method, adaptive coefficients were proposed. The presented method can be used as a direct input to an object detection algorithm that can use the clusters of points to validate an object using its form (e.g. [21]).

Another approach is to use this method as a preprocessing step for higher-level detection algorithms. In this case the output of the weighting function is not thresholded, but directly passed to the next processing step. This is similar to a contrast enhancement method.

Even if presented in the automotive context, the method itself is a general segmentation method in the HSI space. It can also be used in other applications requiring the extraction of an object from an image. For example, if the background saturation and intensity values can be computed (or are known), then the weighting function F_3 can be applied directly to extract the object(s) from the background.

The proposed method was integrated with promising results [21] in a vision sensor in use at Group Research Electronics, Volkswagen AG.

References

1. Adams, R., Bischof, L.: Seeded region growing. IEEE Trans. Pattern Anal. Mach. Intell. pp. 641–647 (1994)
2. Brodley, C., Utgoff, P.: Multivariate decision trees. Mach. Learn. **19**, 45–77 (1995)
3. Brown, T., Koplowitz, J.: The weighted nearest neighbor rule for class dependent sample sizes. IEEE Trans. Inf. Theory **25**, 617–619 (1979)
4. Chen, T., Lu, Y.: Color image segmentation-an innovative approach. In: Proceedings of the IEEE ICPR Conference, pp. 395–405 (2001)
5. Cheng, H.D., Jiang, X.H., Sun, Y., Wang, J.: Color image segmentation: advances and prospects. Pattern Recognit. **34**, 2259–2281 (2001)
6. Comaniciu, D., Meer, P.: Robust analysis of feature spaces: color image segmentation. In: Proceedings of 1997 IEEE Conference on Computer Vision and Pattern Recognition, pp. 750–755 (1997)
7. Deshmukh, K., Shinde, G.: An adaptive color image segmentation. Electron. Lett. Comput. Vis. Image. Anal. **5**, 12–23 (2005)

8. Funka-Lea, G., Bajcsy, R.: Combining color and geometry for the active, visual recognition of shadow. Fifth International Conference on Computer Vision, Cambridge, MA, USA, pp. 203–209 (1995)

9. Hanbury, A., Serra, J.: A 3d-polar coordinate colour representation well adapted to image analysis. Lecture Notes in Computer Science, pp. 804–811 (2003)

10. Haykin, S.: Neural networks a comprehensive foundation (1999)

11. Jain, R., Kasturi, R., Schunck, B.: Recovery of egomotion using region alignment. Machine vision. McGrawHill, NY (1995)

12. Ji, S., Park, H.: Image segmentation of color image based on region coherency. International Conference on Image Processing, pp. 80–83 (1998)

13. Jie, X., Fei, S.P.: Natural color image segmentation. Proc. Int. Conf. Image Process. 1, 973–976 (2003)

14. Kato, Z., Pong, T.C., Lee, J.C.M.: Color image segmentation and parameter estimation in a markovian framework. Pattern Recognit. Lett. 22, 309–321 (2001)

15. Kurugollu, F., Sankur, B., Harmani, A.: Color image segmentation using histogram multithresholding and fusion. Image Vis. Comput. 19, 915–928 (2001)

16. Li, N., Li, Y.: Feature encoding for unsupervised segmentation of color images. IEEE Trans. Syst. Man. Cybern. 33, 438–446 (2003)

17. Meyer, F., Beucher, S.: Morphological segmentation. J. Vis. Commun. Image Represent. 1, 21–46 (1990)

18. Nagao, M., Matsuyama, T., Ikeda, Y.: Region extraction and shape analysis in aerial images. Comput. Graph. Image Process. 10, 195–223 (1979)

19. Navon, E., Miller, O., Averabuch, A.: Color image segmentation based on adaptive local thresholds. Image vis. comput. 23, 69–85 (2005)

20. Rotaru, C., Schmidt, R.: Color based segmentation in automotive image processing using hsi metrics. 11th World Congress on ITS, Japan (2004)

21. Rotaru, C., Schipritt, J., Graf, T., Zhang, J., Buchwald, W.P.: High accuracy object detection and following in color images for automotive applications. IEEE International Conference on Information Acquisition (2005)

22. Sahoo, P.K., Soltani, S., Wong, A.K.C., Chen, Y.C.: A survey on thresholding techniques. Comput. Vis. Graph. Image Process. 41, 233–260 (1988)

23. Silk, E.: Human detection and recognition for an hors d'oeuvres serving robot. Data structures and network algorithms (1983)

24. Soto, A., Saptharishi, M., Ollennu, A.T., Dolan, J., Khosla, P.: Cyber-atvs: dynamic and distributed reconnaissance and surveillance using all terrain ugv. In: Proceedings of the International Conference on Field and Service Robotics, pp. 329–334 (1999)

25. Wang, C., Hunag, L., Rosenfeld, A.: Detecting clouds and cloud shadows on aerial photograps. Pattern Recognit. Lett. 12, 55–64 (1991)

26. Weszka, J.: A survey of threshold selection techniques. Comput. Graph. Image Process. 7, 259–265 (1978)

27. Yining Deng, B.M.: Unsupervised segmentation of color texture region in images and video. IEEE Trans. Pattern Anal. Mach. Intell. pp. 1–25 (2001)

28. Zhang, C., P.Wang: A new method of color image segmentation based on intensity and hue clustering. International Conference on Pattern Recognition (ICPR), pp. 613–616 (2000)

Author Biographies

Calin Rotaru is a PhD candidate at the Department of Computer Science, University of Hamburg, Germany. His PhD work focuses on the topic color machine vision for driver assistance systems and is supported by Volkswagen AG, Group Research Electronics. He graduated (2002) with the topic "Stereo Camera Based Object Recognition" for Driver Assistance Systems from the Faculty of Automation and Computer Science of the Technical University of Cluj-Napoca, Romania. His research interests include color machine vision, smart vision systems, multisensorial data fusion and vision in driver assistance systems.

Thorsten Graf received the diploma (M.Sc.) degree in computer science and the Ph.D. degree (his thesis was on "Flexible Object Recognition Based on Invariant Theory and Agent Technology") from the University of Bielefeld, Bielefeld, Germany, in 1997 and 2000, respectively. In 1997 he became a Member of the "Task Oriented Communication" graduate program, University of Bielefeld, funded by the German research foundation DFG. In June 2001 he joined Volkswagen Group Research, Wolfsburg, Germany. Since then, he has worked on different projects in the area of driver assistance systems as a Researcher and Project Leader. He is the author or coauthor of more than 40 publications and owns several patents. His research interests include image processing and analysis dedicated to advanced comfort/safety automotive applications.

Dr. Jianwei Zhang is full professor and director of the Institute of Technical Aspects of Multimodal Systems, Department of Computer Science, University of Hamburg, Germany. He is one of the Chair Professors "Human-Computer Interaction" of the Department of Computer Science of Tsinghua University. He received his Bachelor (1986) and Master degree (1989) from the Department of Computer Science of Tsinghua University, and his PhD (1994) from the Department of Computer Science, University of Karlsruhe, Germany. His research interests include multimodal information processing, robot learning, service robots, smart vision systems and Embodied Intelligence. In these areas he has published over 120 journal and conference papers, six book chapters and two research monographs. He leads numerous basic research and application projects, including the EU basic research programs and the Collaborative Research Centre supported by the German Research Council. Dr. Zhang has received multiple awards including the IEEE ROMAN Best Paper 2002.