Chapter 4
A Hybrid CRF/HMM for One-Shot Gesture Learning

Selma Belgacem, Clement Chatelain and Thierry Paquet

Abstract This chapter deals with the characterization and the recognition of human gestures in videos. We propose a global characterization of gestures that we call the *Gesture Signature*. The gesture signature describes the location, velocity, and orientation of the global motion of a gesture deduced from optical flows. The proposed hybrid CRF/HMM model combines the modelling ability of hidden Markov models and the discriminative ability of conditional random fields. We applied this hybrid system to the recognition of gesture in videos in the context of one-shot learning, where only one sample gesture per class is given to train the system. In this rather extreme context, the proposed framework achieves very interesting performance which suggests its application to other biometric recognition tasks.

4.1 Introduction

A gesture is a short human body motion, in the range of seconds, achieved primarily with arms to generally perform an action. In some situations of disability or constrained environment, the gesture is the only possible mean of communication between humans or between the human being and the machine. In the latter case, the machine identifies gestures using computer vision techniques.

Gesture analysis field includes several themes: characterization, tracking, recognition, segmentation, spotting, etc. As part of our study, we focus on gesture characterization and recognition. Gesture characterization involves extracting information from the data in the aim to discriminate the classes of gestures. Gesture characterization is a necessary step for gestures recognition.

In the case of continuous sign language, recognition must integrate articulated gestures, it must combine segmentation and classification as well. Segmentation consists in determining the limits of gestures in the sequence of video frames. Classification consists in assigning a label belonging to a given vocabulary of gestures to

S. Belgacem · C. Chatelain · T. Paquet (✉)
LITIS EA 4108, University of Rouen, Saint-etienne du Rouvray, France
e-mail: thierry.paquet@univ-rouen.fr

A. Rattani et al. (eds.), *Adaptive Biometric Systems*,
Advances in Computer Vision and Pattern Recognition,
DOI 10.1007/978-3-319-24865-3_4

each sequence of video frames that compose a specific gesture. As stated by Sayre [30], segmentation and classification are two tasks that must be performed simultaneously. The classification task must also integrate knowledge a priori on data such as the vocabulary of gestures, gesture duration, the recording environment, etc. The segmentation step has to face the variability of the duration of gestures, while the classification step has to face the variability of instances of a same gesture.

A gesture is a set of movements performed mainly with hands. It can be represented in a simplified three-dimensional space consisting of its two-dimensional projection and its variation through time. In addition, the recognition system must be robust to recording environment variations. Indeed, the recording conditions are not usually identical between two sequences representing the same gesture. We can observe changes in brightness, backgrounds, colours, objects, etc. Note that the appearance of the involved human may also change (clothes, skin colour, height, etc.).

Markov models are widely applied to the recognition and segmentation of sequential data. They model the temporal dependencies in sequences. They are based on the Markovian assumption that account for the short-term dependencies only, omitting the long-term dependencies in the model.

Although introducing some simplification in the model, generative Markov Models such as hidden Markov models (HMM) [27] allow to introduce a temporal structure between classes that account for high-level knowledge such as a language model. Some other Markov models such as conditional random fields (CRF) [17] are more oriented toward the local discrimination of patterns. In this work, we propose to combine the advantages of these two types of Markov models to provide a hybrid system. We will show that this hybrid system allows the integration of knowledge while being robust to different sources of variability.

Gestures are characterized using an original global description that account for shapes and motions in the video frames. This method describes the location, the velocity and the direction of the motion, based on the optical flow velocity information.

This system was tested using the "*Gesture Challenge 1–2*" dataset proposed by ChaLearn 2011–2012 [11]. The subject of this competition is one-shot gesture learning [11, 40]. We will show later that the lack of training data is another problem that the Markov models are able to solve to a certain extent.

In Sect. 4.2 of this chapter, we present an overview of the gesture recognition applications in the literature, especially the hybrid models combining HMM with other classification methods. In Sect. 4.4, we show the principle of our hybrid model CRF/HMM and explain its interest. Then, in Sect. 4.5, we describe our gesture characterization model. In Sect. 4.4.2.1, we explain how we adapted our hybrid system to the one-shot learning context, in order to cope with the lack of training data. Finally, we will present in Sect. 4.6, the experimental protocol and the evaluation results of our system and its properties.

4.2 Related Works to Gesture Recognition

During the last decade, many studies have been devoted to gesture recognition, and especially in order to design automatic systems that would recognize the sign language. Such systems would allow deaf people to better communicate with machines or with other humans. For example, Vogler and Metaxas [36], Agris et al. [37] and Ong et al. [25] designed a parallel HMM model for signed sentences recognition. They distinguished gesture descriptors such as position, orientation and distance to facilitate the learning process of the HMM and optimize the use of these descriptors. This decomposition is manifested by the generation of one HMM for each descriptor and for each sub-unit of the model.

For gesture sequences recognition, the use of global parallel HMM models is common in the literature [13, 16, 25, 36–38]. HMM models have also been used with a very small number of training examples [13, 16, 38, 40]. This paper addresses the lack of data problem, which is a major problem in the field of machine learning. Konecny et al. [16], Jackson [13] and Weiss [38] proposed a global HMM model for gesture sequences recognition using single-instance learning databases. The global model is a set of left–right interconnected HMM's modelling each gesture. From each state of each HMM, it is possible to remain in that state or to jump to a subsequent internal or external state. In the model proposed by Jackson [13], each frame of the gesture video is represented by a state. This model remains complex due to the large number of states involved.

The idea of combining HMM with other classification scheme is not new. Such hybrid framework is intended to introduce a better discrimination between classes, than generative models can do. One of the first combination scheme was proposed in the 1990s by the integration of neural networks to HMM's [34]. Such combination is prevalent in the literature in various fields. This type of hybrid models was applied to speech recognition [14, 21, 24, 29, 32, 41], handwriting recognition [3, 9, 15, 19, 20, 22, 33] and gesture recognition [6]. HMM models have also been combined with SVM models for handwriting recognition [8] and with dynamic programming methods for gesture recognition [28]. We noticed that the application of these hybrid models to gesture recognition is recent and not much studied in the literature.

To our knowledge, the only work addressing CRF and HMM combination is the work of Soullard et al. [31], based on the work of Gunawardana et al. [10]. In this work, the authors constrain the learning step of a hidden CRF by initialising it with the parameters of a pretrained HMM. This method ensures the convergence of the hidden CRF learning step and shows the difficulty of learning convergence of such models. The idea of our approach is different and is inspired from neuro-Markovian approaches. The principle of these approaches is to replace the HMM data model, consisting of a mixture of Gaussians, by a discriminative model that classifies local observations. This model is traditionally composed of a neural network which provides local a posteriori probabilities of each class associated to each local observation in the sequence. In this work, we propose the use of a CRF in order to perform this discriminative layer. The CRF layer will discriminate local observations and provide

local class posteriors to the HMM layer. These local posteriors are then combined during the HMM decoding stage that integrates more global information embedded in the HMM transition model (known as the language model). According to the principle of our hybrid model, the HMM learning step and the CRF learning step are performed separately. Details of the new hybrid model we propose are presented in Sect. 4.4.

4.3 Markovian Models

4.3.1 Hidden Markov Models (HMM)

The Hidden Markov Models (HMM) [2] are probabilistic generative statistical models used for sequence recognition. Their principle is to generate observations based on some hidden states. The joint probability $p(y_{1:T}, x_{1:T})$ (Eq. 4.1) for the observation sequence $x_{1:T}$ and the hidden state sequence $y_{1:T}$ is derived from the particular generative graphical model depicted on Fig. 4.1. This simple graphical model is obtained at the expense of two restrictive assumptions: each observation x_t depends only on the current hidden state Y_T (thus assuming observations to be conditionally independent between each other) and each hidden state Y_T depends only on the previous state y_{t-1} (for an order 1 Markov model). Finally, these assumptions lead to the factorization of Eq. 4.1.

$$p(y_{1:T}, x_{1:T}) = p(y_1)p(x_1|y_1)\prod_{t=2}^{T} p(y_t|y_{t-1})p(x_t|y_t) \tag{4.1}$$

Through the inference phase, the most likely sequence of hidden states Y^* that describes the given sequence of observations X is determined. Viterbi algorithm [35] is used to find this best sequence.

The graphical data modelling with a HMM model is very interesting. This model is used to guide the decoding process by preserving the structural consistency over time. This model makes it possible to integrate high-level *a priori* knowledge such as syntactical information or duration. Another advantage of HMM's is that they do not require having labelled frames, as the EM-based training process is able to infer local labels from global label given at gesture level.

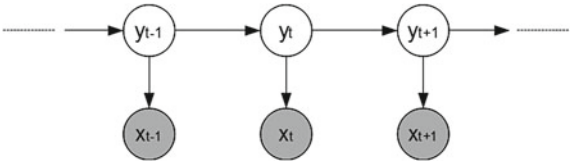

Fig. 4.1 Graphical representation of a HMM: each observation x_t depends only on the hidden state y_t and each hidden state y_t depends only on the previous state y_{t-1}

Generative models such as HMM use Gaussian mixtures to approximate the data distribution. When training data are too few, modelling becomes poor and inadequate which is a major drawback of HMM's. However, discriminative models can remedy this problem. We present in the next section a discriminant sequential Markov model: CRF. This model was proposed by Lafferty et al. [17]. It has some advantages that can address HMM problems.

4.3.2 Conditional Random Fields (CRF)

Conditional random fields (CRF) [17] are discriminative Markov models known for their classification ability. They have been designed in order to model the decision process of labelling a sequence. Therefore, they account for the a posteriori probability of a particular sequence of labels. As depicted in Fig. 4.2, at each time step, a label depends on the previous label (Markov assumption) and may depend on the whole observation sequence X. Making no requirement about the conditional independence of the observation data. The graphical representation of a CRF model is a linear undirected graph with a HMM similar structure. Weights associated to each arc are no longer probabilities but potential functions reflecting the adequacy (or the link) between the two nodes.

The probability of a state sequence $Y = y_{1:T}$ knowing the sequence of observations $X = x_{1:T}$ is computed by:

$$p(Y|X) = \frac{1}{Z(X)} \exp\left(\sum_{t=1}^{T}\sum_{k=1}^{K} \lambda_k f_k(y_{t-1}, y_t, X, t)\right), \tag{4.2}$$

where $Z(X)$ is a normalization term.

f_k, $\forall k \in [1, K]$ are the feature functions. There are two types of feature functions: feature functions of transitions between successive states representing Markov dependencies and observation feature functions. λ_k is the f_k function weight. The weights λ_k, $\forall k \in [1, K]$ are the parameters to be optimized during the CRF training procedure.

As opposed to HMM, CRF are not able to model high-level information such as a language model or syntactical rules. They are local classifiers in a sequential process. Thus, the high-level knowledge must be introduced in postprocessing as

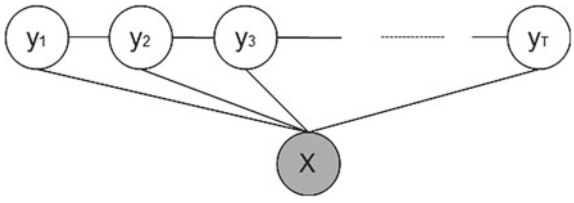

Fig. 4.2 A representation of the graphical structure of the linear CRF

an additional step of filtering in order to guaranty the structural labelling consistency. The HMM's generative framework has this ability of coping with high-level structuring information.

Finally, if we compare the advantages and disadvantages of CRF and HMM, we find a certain complementarity between the two models. Therefore, we propose to combine these two models in a hybrid framework that we present in the next section.

4.4 Hybrid CRF/HMM Model

4.4.1 Overview of the CRF/HMM Model

In this section, we present our hybrid CRF/HMM system for gesture recognition. It combines the discriminative ability of CRF with the modelling ability of HMM. Combining the two models is performed in an easy and straightforward way derived from the literature. The discriminative CRF stage provides local class posterior probabilities that are fed to the HMM stage that account for more global constraints regarding the label sequence. Figure 4.3 shows the proposed hybrid system.

Following this model, the HMM probability $p(y_{1:T}, x_{1:T})$ (see Eq. 4.3) depends on the posteriors computed using the CRF.

$$p(y_{1:T}, x_{1:T}) = p(x_1|y_1)p(y_1)\prod_{t=2}^{T} p(x_t|y_t)p(y_t|y_{t-1}) \tag{4.3}$$

However, $p(x_t|y_t)$ is a likelihood, while the CRF outputs posteriors $p(y_t|x_t)$. Therefore, $p(x_t|y_t)$ is computed from $p(y_t|x_t)$ using Bayes' rule:

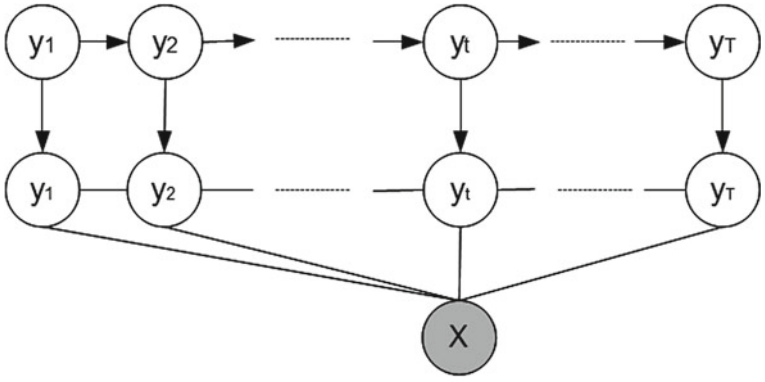

Fig. 4.3 The graphical model CRF/HMM

$$p(x_t|y_t) = \frac{p(y_t|x_t)p(x_t)}{p(y_t)} \tag{4.4}$$

As every gesture class are considered to be equally likely, $p(y_t)$ is a constant $\forall t \in \mathbb{N}$. The aim of the decoding process is to find the state sequence $y_{1:T}$ that maximizes $p(y_{1:T}, x_{1:T})$. As the observation probability $p(x_t)$ is time independent, $p(x_t)$ is not involved in the maximization of $p(x_t|y_t)$. Hence, the maximization of $p(x_t|y_t)$ turns toward the maximization of $p(y_t|x_t)$.

Given that the CRF are able to take into account the whole observation sequence to compute the posteriors of each class, one can state that $p(y_t|x_t) = p(y_t|x_{1:T})$. Let us recall that $y_{1:T}$ and $x_{1:T}$ are noted Y and X.

This is computed within the CRF using the forward-backward algorithm [1], where the forward probability α_t and the backward probability β_t are computed using the following recurrences:

$$\alpha_t(i) = p(x_1 x_2 \dots x_t, y_t = s_i) = \sum_{j=1}^{N_s} \alpha_{t-1}(j)\psi_t(s_i, s_j, o_l), \tag{4.5}$$

$$\beta_t(i) = p(x_{t+1} x_{t+2} \dots x_T, y_t = s_i) = \sum_{j=1}^{N_s} \beta_{t+1}(j)\psi_{t+1}(s_i, s_j, o_l), \tag{4.6}$$

where

$$\psi_t(s_i, s_j, o_l) = \exp(\sum_{k=1}^{K} \lambda_k f_k(y_t = s_i, y_{t-1} = s_j, x_t = o_l)) \tag{4.7}$$

and s_i, s_j are hidden state that belong to $\mathscr{S}$, and o_l is an observation that belong to $\mathscr{O}$. Finally, following the forward-backward procedure, we have:

$$p(X) = \sum_{j=1}^{N_s} \alpha_T(j) = \sum_{j=1}^{N_s} \beta_1(j) = \sum_{j=1}^{N_s} \alpha_t(j)\beta_t(j), \tag{4.8}$$

$$p(y_t = s_i|X) = \frac{p(y_t = s_i, X)}{p(X)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N_s} \alpha_t(j)\beta_t(j)} = \gamma_t(i). \tag{4.9}$$

4.4.2 Training the CRF/HMM Model

We chose to achieve a separated training of HMM and CRF. The HMM training provides the transition matrix between gesture states. Transition models are learned separately for each gesture class and gathered into a global model for decoding gesture sequences. This model is described in Sect. 4.4.4.

As CRF do not benefit from an embedded training stage like HMM, it is necessary to build a frame-labelled learning dataset. This is achieved using the initial HMM model of gesture trained on the dataset that are used in a forced alignment mode that provides the desired frame labelling. Then, the CRF learns a single model for every gestures, considering as many classes in the model as there are sub-gestures. The number of sub-gestures is equal to the number of states in the HMM model of gesture.

4.4.2.1 CRF/HMM Adaptation to One-Shot Learning

In this section, we focus on the learning of the recognition system using a unique sample per class. These learning conditions are interesting since the annotation efforts are extremely reduced in this case. Furthermore, using a single sample per class allows to speed up the learning process.

The one-shot learning framework has been quite extensively used for gesture analysis and recognition [13, 16, 38–40]. These system are generally made up of a standard recognition method that has been adapted to the one-shot learning framework. We now describe the adaptation of our models (HMM and CRF) to one-shot learning.

To model the feature space, the HMM relies on Gaussian mixtures estimated on the learning database. When considering a very reduced number of samples, the Gaussian distribution parameters are very difficult to estimate, especially the variance. Therefore, first we limited the mixture to one Gaussian per gesture class. Second, the variance is computed on every gesture class in order to increase the amount of data and improve the estimation. Doing that, each gesture class has the same variance. Although these two tricks are a limitation of the initial method, the experiments showed the interest of such an adaptation.

In its initial form, the CRF method is mathematically able to deal with either discrete or continuous features; however, since the CRF classification stage is derived from a logistic regression, it is more adapted to discrete features than continuous. This is even more true when the number of samples is small. Therefore, we turned toward the use of a feature quantization procedure. It allows to efficiently tune the parameters linked to each discrete feature value. Notice that some recent developments have introduced hidden CRF models in order to cope with continuous features [26]. But such a framework would require more data than possible in the one-shot learning context.

The quantification is achieved using a uniform scalar quantifier that maps each continuous feature into $\mathtt{N_Q}$ discrete features, according to the following equation:

$$\begin{aligned} Q : [-\mathtt{V_{max}}, \mathtt{V_{max}}] &\longrightarrow [-\mathtt{N_Q}, \mathtt{N_Q}] \\ x &\longmapsto \tfrac{x \times \mathtt{N_Q}}{\mathtt{V_{max}}}. \end{aligned} \tag{4.10}$$

We empirically tuned the value $\mathtt{N_Q}$ in order to reach the best recognition performance using a validation procedure. We found that $\mathtt{N_Q} = 16$ was the best value.

4.4.3 Structure and Parametrization of the CRF/HMM Model

As for a standard HMM, the HMM of our hybrid structure is made of states describing each gesture. Although the gesture duration can be modelled through the state autotransitions, it is known that a better modelization can be achieved by setting a variable number of states per gesture. We experimentally checked that this strategy outperforms the performance of the same system with a fixed number of states per gesture. The number of states of each gesture i is determined automatically depending on its frame length $\mathtt{f_g}(i)$. The theoretical number of frames per state, denoted $\mathtt{f_s}$, is one hyperparameter of the system. We denote the number of states of a gesture model i; $\mathtt{N_e}i = \mathtt{f_g}(i)/\mathtt{f_s}$. As we already mentioned, we limit the data model to have only one Gaussian per state.

The CRF part of our hybrid model has a standard linear structure, as shown in Fig. 4.3. The CRF training leads to a single model that discriminates all the gestures of the dataset. As explained in the previous section, the CRF formulation allows to consider an observation window, including the current observation and a neighbouring context to be determined. To adapt the system to the gesture duration variability, we chose a variable size $\mathtt{f_w}$ of the observation window *CRFwind*. $\mathtt{f_w}$ is statically estimated on the learning databases. In order to avoid overfitting the CRF, a regularization term has been empirically tuned to a value of 1.5.

4.4.4 Decoding Using the CRF/HMM Model

The gesture sequence to recognize may contain an arbitrary number of gestures, in an arbitrary order. Therefore, the model should evenly switch between the gesture models. This can be modelled by gathering all the gesture model within a global sequence model, as shown in Fig. 4.4. In this model, each line represents an isolated gesture, with a variable number of state. This global model allows to describe any arbitrary gesture sequence with equiprobable gesture transition probabilities.

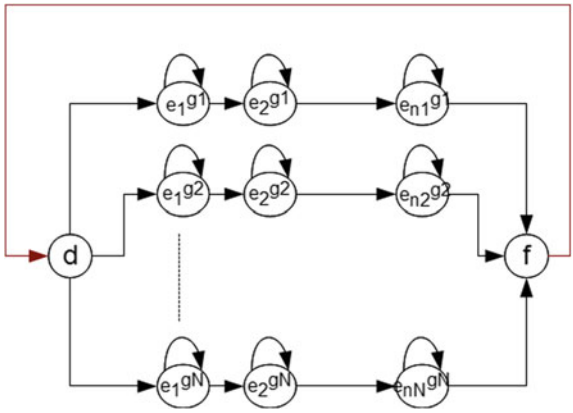

Fig. 4.4 The recognition model of gesture sequences using HMM. $e_j^{g_i}$ represents the state j of the gesture i

4.5 Global Gestures Characterization

Gestures characterization requires velocity descriptors and shape descriptors as well. Considering that signers can wear clothes in different colours and have different skin colours; colour descriptors are not included in our characterization model.

In this section, we present a set of motion descriptors deduced from optical flows velocities. We call this set of descriptors *Gesture Signature* (GS). We also propose to include shape descriptors extracted with histogram of oriented gradients (HOG). Such descriptors will account for shape descriptors.

4.5.1 Characterization with Optical Flows: Gesture Signature

Optical flows describe local velocities at the pixel level. They are known for their robustness to brightness changes [4]. They are invariant to colours and object distortion. Optical flows are able to describe simultaneously all movements in the scene without any segmentation. Therefore, this method seems adequate to simultaneously extract a maximum of information on body motion, while being robust to variability of colour, shape and brightness. In what follows, we propose a feature vector whose components are combinations of velocity values computed with optical flows.

Hand movements are usually located on the left and the right part of the image, so it is advantageous to divide the image into two vertical sections as shown in Fig. 4.5. Thus, the description of the movement is better localized and motions are characterized in these two distinct regions.

Each part of the image is described by a gesture signature which consists of nine descriptors derived from positive and negative horizontal components $\mathrm{V}_{\mathrm{X}}^{+}$ and $\mathrm{V}_{\mathrm{X}}^{-}$, and nine descriptors derived from vertical components $\mathrm{V}_{\mathrm{Y}}^{+}$ and $\mathrm{V}_{\mathrm{Y}}^{-}$. These components

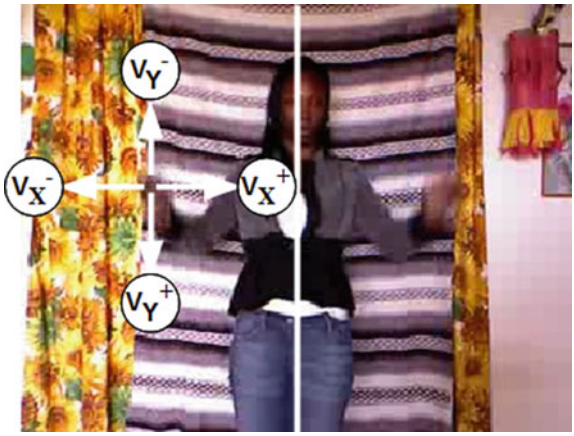

Fig. 4.5 The directions of the optical flow components (image from a ChaLearn database video)

are derived from optical flows at each pixel of the image at position p (Fig. 4.5). Obviously, for each pixel p, two of these four values are null; one pixel can have only one direction according to the x-axis and one direction according to the y-axis.

For a given direction, these nine descriptors consist of four movement *location* descriptors, two movement *velocity* descriptors and three movement *orientation* descriptors. Although these features are simple, they are complementary and describe precisely the gesture changes since location, velocity and orientation are the main components of a gesture.

Table 4.1 shows the 18 features set.

The eight horizontal and vertical location features are related to inertia centre coordinates. They represent the vertical and horizontal positions of velocity centres with respect to the global movement of the considered portion of the image.

There are four features of movement velocity and strength. The first descriptor gives an energy information of the movement. It is inversely proportional to the quadratic mean of the moving pixels velocities. For normalization reasons, we use the inverse of this quadratic mean. The second descriptor gives information about the motion amplitude. It is the median of the moving pixels velocities. The median integrates information about the linear momentum, where the mass is replaced in our case by the number of moving pixels. The median also reduces the noise effect. $\mathrm{V}_{\mathrm{X}}^{*}$ and $\mathrm{V}_{\mathrm{Y}}^{*}$ components are the medians of a threshold velocity vector which is computed with optical flows. Values of the threshold are given below.

$$S_{V_X} = \frac{\sum_{p=1}^{\mathrm{N}_{\mathrm{px}}^{s}} |\mathrm{V}_{\mathrm{X}}(p)|}{\mathrm{N}_{\mathrm{px}}^{s}} \qquad S_{V_Y} = \frac{\sum_{p=1}^{\mathrm{N}_{\mathrm{px}}^{s}} |\mathrm{V}_{\mathrm{Y}}(p)|}{\mathrm{N}_{\mathrm{px}}^{s}}$$

The six movement orientation features are statistics on pixels moving in the same direction, positive or negative. The first two descriptors characterize the amount of pixels moving in the same direction. The third descriptor characterizes the dominant direction of the movement. Those three descriptors characterize the relationship or the symmetry between the two main movement groups whose orientations are opposite. Figure 4.6 shows the interest of these descriptors and illustrates the symmetry information. Thus, by analysing the variation of these three descriptors, we can deduce the type of associated movement. Hence the importance and the complementarity of these three orientation descriptors.

4.5.2 Characterization with HOG

For a complete gesture characterization, we add global contour features extracted with a classic shape descriptor; histograms of oriented gradients (HOG). To apply this descriptor, we resumed the implementation of Dalal et al. [7]. nine directions are used to quantify gradients inclination angles calculated on the image. According to the work of Dalal et al. [7], detecting people with these nine orientations is efficient.

Table 4.1 The eight movement **location** features, the four motion **velocity** features and the six movement **orientation** features

	Descriptor	Horizontally	Vertically
Location	Average Abscissa of pixels moving in the positive direction (AAP)	$\frac{1}{I_w} \times \frac{\sum_{p=1}^{N_{px}^+} \lvert V_X^+(p)\rvert x_p}{\sum_{p=1}^{N_{px}^+} \lvert V_X^+(p)\rvert}$	$\frac{1}{I_w} \times \frac{\sum_{p=1}^{N_{px}^+} \lvert V_Y^+(p)\rvert x_p}{\sum_{p=1}^{N_{px}^+} \lvert V_Y^+(p)\rvert}$
	Average ordinate of pixels moving in the Positive direction (AOP)	$\frac{1}{I_h} \times \frac{\sum_{p=1}^{N_{px}^+} \lvert V_X^+(p)\rvert y_p}{\sum_{p=1}^{N_{px}^+} \lvert V_X^+(p)\rvert}$	$\frac{1}{I_h} \times \frac{\sum_{p=1}^{N_{px}^+} \lvert V_Y^+(p)\rvert y_p}{\sum_{p=1}^{N_{px}^+} \lvert V_Y^+(p)\rvert}$
	Average Abscissa of pixels moving in the negative direction (AAN)	$\frac{1}{I_w} \times \frac{\sum_{p=1}^{N_{px}^-} \lvert V_X^-(p)\rvert x_p}{\sum_{p=1}^{N_{px}^-} \lvert V_X^-(p)\rvert}$	$\frac{1}{I_w} \times \frac{\sum_{p=1}^{N_{px}^-} \lvert V_Y^-(p)\rvert x_p}{\sum_{p=1}^{N_{px}^-} \lvert V_Y^-(p)\rvert}$
	Average ordinate of pixels moving in the negative direction (AON)	$\frac{1}{I_h} \times \frac{\sum_{p=1}^{N_{px}^-} \lvert V_X^-(p)\rvert y_p}{\sum_{p=1}^{N_{px}^-} \lvert V_X^-(p)\rvert}$	$\frac{1}{I_h} \times \frac{\sum_{p=1}^{N_{px}^-} \lvert V_Y^-(p)\rvert y_p}{\sum_{p=1}^{N_{px}^-} \lvert V_Y^-(p)\rvert}$
Velocity	Global velocity inverse (GVI)	$\sqrt{\frac{N_{px}}{\sum_{p=1}^{N_{px}} (V_X(p))^2}}$	$\sqrt{\frac{N_{px}}{\sum_{p=1}^{N_{px}} (V_Y(p))^2}}$
	Maximum velocities median (MVM)	$\frac{1}{S_{V_X}} \times \lvert V_X^*\rvert$	$\frac{1}{S_{V_Y}} \times \lvert V_Y^*\rvert$
Orientation	Proportion of the pixels moving in the positive direction (PPP)	$PPP_X = \frac{N_{px}^{V_X^+}}{N_{px}}$	$PPP_Y = \frac{N_{px}^{V_Y^+}}{N_{px}}$
	Proportion of the pixels moving in the negative direction (PPN)	$PPN_X = \frac{N_{px}^{V_X^-}}{N_{px}}$	$PPN_Y = \frac{N_{px}^{V_Y^-}}{N_{px}}$
	Dominant orientation (DO)	$DO_X = \frac{N_{px}^{V_X^+} - N_{px}^{V_X^-}}{N_{px}}$	$DO_Y = \frac{N_{px}^{V_Y^+} - N_{px}^{V_Y^-}}{N_{px}}$

In HOG, these nine directions are weighted by the corresponding gradients norms in the image computing cells. It applies sliding window self-superposition, generating redundant histograms and a very large HOG feature vector (with size equal to 3780). To alleviate this vector, we used the average operator on two levels. The first simplification level is based on the HOG visualization algorithm proposed by Jurgen Brauer.[1] The idea of this algorithm is to average the redundant histograms on image

[1] http://www.juergenwiki.de/work/wiki/doku.php?id=public%3ahog_descriptor_computation_and_visualization.

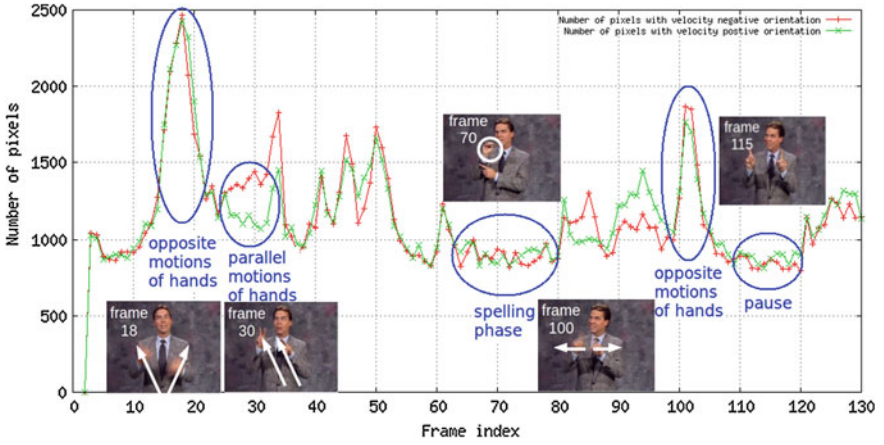

Fig. 4.6 Evolution of the descriptors PPP_X and PPN_X in a video from SignStream database [23]. Two *curves* superimposed with a presence of a peak correspond to an opposite movement of the two hands. A strong difference between the two curves corresponds to a parallel movement of both hands in the dominant direction. A stagnation of the two curves corresponds to fixed hands (frame 70)

cells keeping nine gradient directions in each cell. The second level of the HOG descriptor simplification, applied in our case, is to average the gradient amplitudes on larger image blocks that we call meta-blocks. We partitioned the image blocks to meta-blocks. For each meta-block and for each orientation, we compute mean amplitude in all meta-block cells. Tests were carried out using 4 and 16 meta-blocks. We obtain then nine amplitude averages per meta-block which leads to a descriptor of size $9 \times 4 = 36$ or $9 \times 16 = 144$. In our case, HOG are computed on the difference of two successive images in order to characterize only moving patterns.

4.6 Experimental Protocol

In this section, we explain the experimental protocol: databases, evaluation methods, feature vector variants and implementation tools.

4.6.1 Databases

Our recognition system has been evaluated on public databases designed for the ChaLearn 2011–2012 competition [11]. We did not participate to this competition but we were able to compare our system to those of the participants thanks to the

evaluation platform proposed by the competition organizers.[2] We detail the results of this evaluation in Sect. 4.7.

ChaLearn databases are made of three types of resources: 480 system development sub-databases named *devel*, 20 system validation sub-databases named *valid* and 40 system final evaluation sub-databases named *final*. The 1–20 *final* sub-databases were tested in the first round of the competition and 21–40 *final* sub-databases were tested in the second round of the competition. This final evaluation classifies participants in the ChaLearn competition.

Each of these sub-databases contains 47 pairs of videos. Each video pair presents the same scene in two formats: RGB colour format and depth format. These videos are recorded using a Kinect (TM) camera at a frequency of 10 frames per second, with a resolution of 240 $\times$ 320 pixels. Videos of the same sub-database share the same scenic features: same actor, same background, same recording conditions, same theme and same gesture vocabulary. However, these scenic characteristics vary from sub-database to another. 20 players participated in the making of these databases, one actor per sub-database. These databases present 30 vocabularies composed of 8–15 gestures belonging to various themes such as video games, distance education, robot control, sign language, etc.

Each sub-database includes two sets of video: a training set $\mathbb{G}$ and a test set $\mathbb{S}$. The training set $\mathbb{G}$ consists of 10 videos. Each video contains a single and isolated instance of a gesture: *one-shot learning* databases. The test set $\mathbb{S}$ consists of 40 videos. Each video includes a sequence of 1–5 successive gestures separated by a common break point. Gestures organization in each test sequences is random, there is no specific gestures grammar.

We summarize in the following subsection the various feature vectors used for the tests.

4.6.2 Feature Vector Variants

Table 4.2 presents the different variants of the feature vector $\mathbf{c}$ we used in our experiments. We index each variant by its size $l(\mathbf{c})$. $l(\mathbf{v}(GS))$ is the number of gesture signature features. $l(\mathbf{v}(HOG))$ is the number of HOG features. Some variants of the feature vector $\mathbf{c}$ are applied to two data formats (RGB image and depth image).

4.6.3 Evaluation Metric

The organizers of the ChaLearn competition defined a global evaluation metric on all test sequences based on the Levenshtein distance, also called edit distance [18]. This form of global error is denoted by $\mathscr{L}_{\mathrm{ch}}$ and given by Eq. 4.11.

[2] https://www.kaggle.com/c/GestureChallenge2.

Table 4.2 Feature vector variants adopted in the experiments

Total size $l(\mathbf{c})$	Descriptor			
	Gesture signature GS		HOG	
	$l(\mathbf{c}(GS))$	Description	$l(\mathbf{c}(HOG))$	Description
54	18	No image division	36	4 meta-blocks
52	16	No median, no image division	36	4 meta-blocks
180	36	Image division into 2 parts	144	16 meta-blocks
360	72	Image division into 2 parts, 2 data formats	288	16 meta-blocks, 2 data formats
72	72	Image division into 2 parts, 2 data formats	0	HOG not applied

$$\begin{aligned} \mathscr{L}_{\mathtt{ch}} : \mathbb{D} &\longrightarrow \mathbb{R} \\ \mathbb{S} &\longmapsto \frac{\sum_{s\in\mathbb{S}} \mathrm{L}(\mathscr{R}(s), \mathscr{T}(s))}{\sum_{s\in\mathbb{S}} l(\mathscr{T}(s))}, \end{aligned} \tag{4.11}$$

where $\mathbb{D}$ is the set of test databases, $\mathbb{S}$ is the set of test sequences, $\mathtt{s}$ is the sequence of gestures, $\mathscr{R}(s)$ is the system recognition result of sequence $\mathtt{s}$, $\mathscr{T}$ is a function giving the ground truth sequence $\mathtt{s}$, $\mathrm{L}(.,.)$ is the Levenshtein distance and $l(v)$ gives the size of a vector v.

We use the ChaLearn form of the error $\mathscr{L}_{\mathtt{ch}}$ to compare our recognition system to ChaLearn participants recognition systems. However, let us emphasize that $\mathscr{L}_{\mathtt{ch}}$ is slightly different from the classical Levenshtein distance (see Eq. 4.12), which is bounded and seems more generic. Thus, to present the main results of our various tests, we use the classic error form.

$$\begin{aligned} \mathscr{L} : \mathbb{D} &\longrightarrow [0, 1] \\ \mathbb{S} &\longmapsto \frac{1}{|\mathbb{S}|} \sum_{s\in\mathbb{S}} \frac{\mathrm{L}(\mathscr{R}(s), \mathscr{T}(s))}{l(\mathscr{R}(s)) + l(\mathscr{T}(s))} \end{aligned} \tag{4.12}$$

4.6.4 Implementation Tools

We used the OpenCV library [5] to develop image and video processing methods. HMM gesture recognition methods have been implemented thanks to Torch library,[3] while CRF gesture recognition methods rely on the CRF++ library.[4]

[3] http://torch.ch/torch3/.

[4] http://crfpp.googlecode.com/svn/trunk/doc/index.html.

4.7 Gesture Recognition Results

In this section, we present the results of our system, using different variants. We first evaluate the effect of the quantification of continuous features for a discrete CRF in Sect. 7.1. Then, we demonstrate the robustness of the hybrid model CRF/HMM with respect to the number of states and to the various feature vectors in Sect. 7.2. Finally, we compare the recognition results of the hybrid system CRF/HMM to the classic and adapted versions of HMM and CRF in Sect. 7.3. We conclude this section by presenting our rank compared to participants at the ChaLearn competition.

All recognition performance results of the hybrid system CRF/HMM presented in this section are obtained with tests performed with an adapted CRF/HMM as explained in Sect. 4.4.2.1 unless otherwise stated. Adapted HMM and adapted CRF recognition systems cited in this section are also adapted as explained in Sect. 4.4.2.1.

4.7.1 Evaluation of the Features Quantization for CRF

Although CRF are able to cope with continuous features, it has been shown that discretizing the feature set could increase its performance, especially when the number of training examples is small [12].

Indeed, continuous CRF put a single weight for all values of a descriptor. Whereas a reduced value of this descriptor does not necessarily mean that it has no importance and a high value of this descriptor does not mean that it is really important. This way of managing weights can be suitable to weight a score function whose values have a monotonous importance. However, for a descriptor, distinctive ranges of values can change from one descriptor to another. Thus, discrete CRF, which give a distinct weight for each discrete feature value, provide more specification to features, which subsequently increases the discrimination of classes. Therefore, discrete CRF seems an adequate model for one-shot learning case, as we noted in the Sect. 4.4.2.1.

Figure 4.7 (left) presents the CRF/HMM recognition performance in both continuous and discrete characteristics cases by varying the number of frames per state for the HMM component. Discrete system performances clearly outperform continuous system performances, which demonstrate the interest of quantification. Let us also mention that the learning time of continuous CRF (estimated in hours) largely exceeds the learning time of discrete CRF (estimated in minutes). This is another advantage of discrete CRF.

4.7.2 Robustness of the CRF/HMM Approach

In this section, we analyse the influence of several parameters on our CRF/HMM approach results: number of frames per states, gesture duration and feature vector.

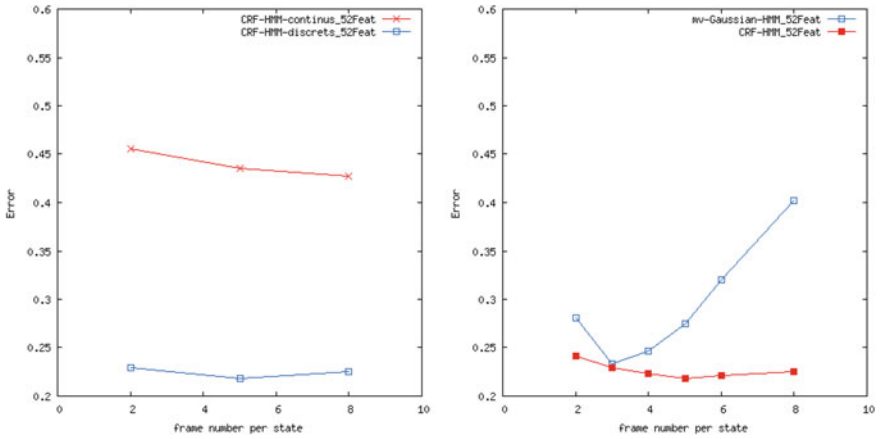

Fig. 4.7 *Left* CRF/HMM gesture recognition results with continuous and discrete component. *Right* CRF/HMM and adapted HMM systems robustness to the variation of the number of frames per state

4.7.2.1 Robustness to Changes in the Number of Frames per State

Figure 4.7 (right) shows the recognition error $\mathscr{L}$ of adapted HMM and CRF/HMM systems with respect to the number of frame per state $\mathtt{f_S}$. For each value of $\mathtt{f_S}$, the recognition system is re-learned. One can observe that the CRF/HMM system outperforms HMM, and that the CRF/HMM system provides extremely stable results, while the system performance of HMM is strongly variable. This is an interesting feature since it does not require a fine hyperparameter tuning for reaching good results.

4.7.2.2 Robustness to Changes in the Gesture Duration

The change in the number of frames per state has a direct impact on the CRF/HMM robustness to the gesture duration variation. With a large number of frames per state, CRF/HMM system is able to handle the temporal elasticity of a gesture. In other words, when a gesture expands or narrows through the number of frames in the test data, CRF/HMM system is able to align the gesture model on the data and decode them. In addition, CRF component are able to implicitly manage narrowing and expansion of data through their local decision which is independent from the data global model, unlike HMM which are dependent on a graph-oriented model without jumps. Thus, to manage the temporal elasticity of gestures, a simple structure of the hybrid model with a reduced number of states can replace a complex HMM system with jumps between states and a complete connection as adopted by some participants of the ChaLearn competition [13, 16, 38].

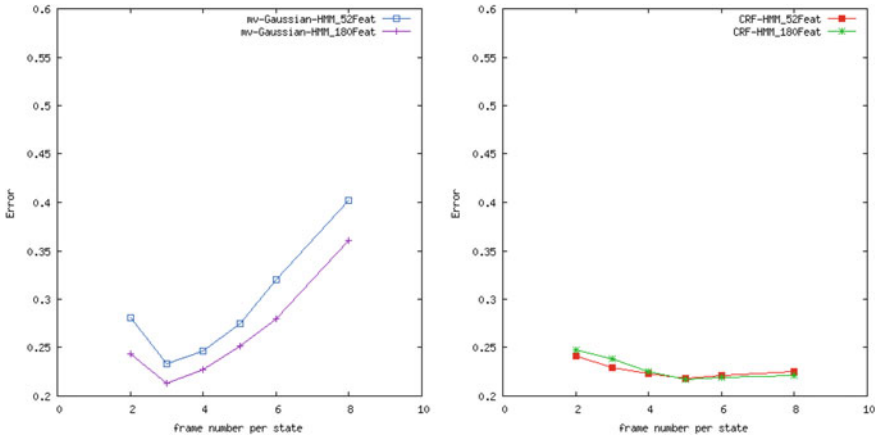

Fig. 4.8 Adapted HMM (*left*) and CRF/HMM (*right*) robustness to the variation of the feature vector

4.7.2.3 Robustness to Changes in the Feature Vector

Figure 4.8 present the variation of the error $\mathscr{L}$ in terms of the number of frames per state $\mathtt{f_s}$ for two HMM systems (left) and for two CRF/HMM systems (right). Each pair of systems is evaluated with two different feature vectors. When the feature vector size decreases, CRF/HMM keep almost the same performance. In other words, a minimum of features is sufficient for CRF HMM, whereas for classic HMM, features addition increases greatly the recognition performance. This recognition ability with a reduced number of features makes features extraction task easier and faster.

These three CRF/HMM robustness property prove that with a simple system, it is possible to reach high recognition performance thanks to CRF and HMM advantages combination and disadvantages compensation. We can see the simplicity of the CRF/HMM system at three levels: (a) a simple model structure with a reduced number of state without jumps nor complete connection; (b) a reduced number of features; and (c) a training dataset reduced to an example by class.

4.7.3 Evaluation of the CRF/HMM Using the ChaLearn Platform

We present in this subsection the recognition results of our best hybrid system CRF/HMM on the *valid* and *final* databases, as well as our ranking in the ChaLearn competition.

We first present a comparison of the performance of the main recognition systems that we studied (Table 4.3) on the *devel* databases. The 52 feature vector has been

Table 4.3 The recognition results of various recognition systems based on HMM and CRF and tested on 20 *devel* databases

System	$l(\mathbf{c})$	$\mathtt{f_s}$	Error: $\mathscr{L}$
classic HMM	52	6	0.36
adapted HMM	52	3	0.23
classic CRF (continuous)	52	$\mathtt{f_g(g)}$	0.29
adapted CRF (discrete)	52	$\mathtt{f_g(g)}$	0.28
CRF/HMM (adapted)	52	5	**0.22**

Table 4.4 The recognition results of our best hybrid system CRF/HMM on 20 *valid* databases, 20 *final* 1–20 databases and 20 *final* 21–40 databases (each database category contains about 750 total sequences test in the order of 200 frames each)

Database category	Error		Ranking
	$\mathscr{L}$	$\mathscr{L}_{\mathtt{ch}}$	
Valid	0.177193	0.348812	–
Final 1–20 (1st round)	0.147924	0.296440	7th
Final 21–40 (2nd round)	0.122398	0.252357	7th

chosen since it provides good results while keeping a compact representation (see Table 4.2). It is identical for all the systems. The number of frames per state $\mathtt{f_s}$ has been optimized for each system. $\mathtt{f_g(g)}$ represents the size of the learned gesture, which means that every gesture is represented by a single class, subclasses that correspond to states in the case of HMM do not exist in the case of CRF. On the other hand, a postprocessing step is applied to the classic and adapted CRF in order to filter their recognition results. Without this step recognition error exceeds 0.5. Table 4.3 shows that the performance of the proposed hybrid system CRF/HMM clearly outperforms the recognition performances of other systems.

In order to rank our system in the ChaLearn 2011–2012 competition, we tested the hybrid system on *valid* and *final* databases provided during the competition. Table 4.4 shows the hybrid system CRF/HMM recognition error values computed with both evaluation methods $\mathscr{L}$ and $\mathscr{L}_{\mathtt{ch}}$ on *valid* and *final* databases. Table 4.4 presents the CRF/HMM system rank on both database categories using the $\mathscr{L}_{\mathtt{ch}}$ error. It appears that we ranked at the 7th position among 559 systems from 48 participants for both first and second rounds. The complete list with their score (the $\mathscr{L}_{\mathtt{ch}}$ error) is available on the Kaggle website for the first[5] and the second round.[6] We achieved this rank using only RGB format data.

[5] https://www.kaggle.com/c/GestureChallenge/leaderboard.

[6] https://www.kaggle.com/c/GestureChallenge2/leaderboard.

Beside the competition, for a data size equal to 750, we demonstrated with the statistical unilateral student test that our hybrid model CRF/HMM significantly outperforms classic models HMM and CRF. CRF/HMM also outperforms the adapted HMM[7] with a confidence level of 99 % and the adapted CRF (see footnote 7) with a confidence level of 99.5 %.

These results and this study show that the CRF/HMM hybrid system is a system that has better performance than other classic systems (HMM and CRF), is robust to different variations, and is interesting and practical in the real-world problem such as one-shot learning.

4.8 Conclusion

In this chapter, we proposed a new hybrid system for gesture recognition CRF HMM. We demonstrated that this combination of Markov models benefits from each model advantages without undergoing its drawbacks. These Markovian models have been adapted to one-shot learning context in order to improve their recognition ability. We also proposed a new gesture characterization model which is a gesture Signature based on optical flows.

We demonstrated that these gesture characterization and recognition models constitute a robust recognition hybrid system that opens up new perspectives for sequential Markov models. An interesting perspective of our gesture recognition work concerns the gesture detection task, called the gesture *spotting*. Gesture spotting consists on locating and labelling specific gestures in videos. It can be applied in video documents management contexts such as video retrieval, categorization and indexing. Our recognition model could be adapted to the spotting task by representing false examples through an additional class to the gestures vocabulary.

Finally, we demonstrated in this chapter Markov systems ability to model and manage spatio-temporal variations of sequential data, including gestures. The modelling evolution of the human activity contributes to the evolution of computer vision techniques and subsequently contributes to the evolution of human–machine interaction systems.

References

1. Austin, S., Schwartz, R., Placeway, P.: The forward-backward search algorithm. In IEEE ICASSP, pp. 697–700 (1991)
2. Baum, L.E., Petrie, T.: Statistical inference for probabilistic functions of finite state Markov chains. Ann. Math. Stat. **37**, 1554–1563 (1966)
3. Bengio, Y., LeCun, Y., Nohl, C., Burges, C.: LeRec: a NN/HMM hybrid for on-line handwriting recognition. neural Comput. **7**(6), 1289–1303 (1995)

[7]The mentioned adaptation is the model adaptation to the one-shot learning context.

4. Bhandarkar, Suchendra M., Luo, Xingzhi: Integrated detection and tracking of multiple faces using particle filtering and optical flow-based elastic matching. CVIU **113**(6), 708–725 (2009)
5. Bradski, Gary, Kaehler, Adrian: Learning OpenCV: Computer Vision with the OpenCV Library. O'Reilly, Cambridge (2008)
6. Corradini, A.: Real-time gesture recognition by means of hybrid recognizers. In: Gesture Workshop, vol. 2298, pp. 34–46. Springer (2001)
7. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: International Conference on Computer Vision & Pattern Recognition, vol. 2, pp. 886–893 (2005)
8. Ganapathiraju, A., Hamaker, J., Picone, J.: Hybrid SVM/HMM architectures for speech recognition. In: INTERSPEECH, ISCA, pp. 504–507 (2000)
9. Gilloux, M., Lemarie, B., Leroux, M.: A hybrid RBF network/hidden Markov model handwritten word recognition system. In: ICDAR, pp. 394–397 (1995)
10. Gunawardana, A., Mahajan, M., Acero, A., Platt, J.C.: Hidden conditional random fields for phone classification. In INTERSPEECH, ISCA, pp. 1117–1120 (2005)
11. Guyon, I., Athitsos, V., Jangyodsuk, B., Hamner, P., Escalante, H.: ChaLearn gesture challenge: Design and first results. In CVPR Workshops, pp. 1–6. IEEE (2012)
12. Hebert, D., T. Paquet, Nicolas, S.: Continuous CRF with multi-scale quantization feature functions application to structure extraction in old newspaper. In: ICDAR, pp. 493–497 (2011)
13. Jackson, E.: An HMM-based approach for gesture recognition using edge features. In: CVPR 2012 Workshop on Gesture Recognition (2012)
14. Johansen, F.T.: A comparison of hybrid HMM architectures using global discriminative training. In: ICSLP, ISCA (1996)
15. Knerr, S., Augustin, E.: A neural network-hidden markov model hybrid for cursive word recognition. ICPR **2**, 1518–1520 (1998)
16. Konencny, J., Hagara, M.: One-shot learning gesture recognition using HOG/HOF features. In: ICPR 2012 Workshop on Gesture Recognition (2012)
17. Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: ICML, pp. 282–289 (2001)
18. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions and reversals. Soviet Physics Doklady **10**, 707 (1966)
19. Marukatat, S., Artieres, T., Gallinari, P., Dorizzi, B.: Sentence recognition through hybrid neuro-Markovian modeling. In: ICDAR, pp. 731–737 (2001)
20. Matan, O., Burges, C., Lecun, Y., Denker, J.S.: Multi-digit recognition using a space displacement neural network. In: Advances in Neural Information Processing Systems, vol. 4, pp. 488–495 (1992)
21. Morgan, N., Bourlard, H., Renls, S., Cohen, M., Franco, H.: Hybrid neural network/hidden Markov model systems for continuous speech recognition. IJPRAI **7**(4), 899–916 (1993)
22. Morita, M.E., Sabourin, R., Bortolozzi, F., Suen, C.Y.: Segmentation and recognition of handwritten dates: an HMM-MLP hybrid approach. IJDAR **6**(4), 248–262 (2003)
23. Neidle, Carol, Sclaroff, Stan, Athitsos, Vassilis: SignStream: a tool for linguistic and computer vision research on visual-gestural language data. Behav. Res. Methods Instrum. Comput. **33**(3), 311–320 (2001)
24. Niles, L.T., Silverman, H.F.: Combining hidden Markov models and neural network classifiers. In: ICASSP, pp. 417–420 (1990)
25. Ong, S.C.W., Ranganath, S.: Deciphering gestures with layered meanings and signer adaptation. In: IEEE International Conference on Automatic Face and Gesture Recognition, p. 559 (2004)
26. Quattoni, A., Wang, S., Morency, L., Collins, M., Darrell, T.: Hidden conditional random fields. IEEE Trans. Pattern Anal. Mach. Intell. **29**(10), 1848–1852 (2007)
27. Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. In: Proceedings of the IEEE, pp. 257–286 (1989)
28. Rajko, S., Qian, G.: A Hybrid HMM/DPA adaptive gesture recognition method. In: ISVC, vol. 3804, pp. 227–234 (2005)
29. Rigoll, G.: Maximum mutual information neural networks for hybrid connectionist-HMM speech recognition systems. IEEE Trans. Speech Audio Process. **2**(1), 175–184 (1994)

30. Sayre, Kenneth M.: Machine recognition of handwritten words: a project report. Pattern Recogn. **5**(3), 213–228 (1973)
31. Soullard, Y.: Hybrid HMM and HCRF model for sequence classification. Bruges, Belgium (2011)
32. Tebelskis, J., Waibel, A., Petek, B., Schmidbauer, O.: Continuous speech recognition by linked predictive neural networks. In: NIPS, pp. 199–205 (1990)
33. Thomas, S., Chatelain, C., Heutte, L., Paquet, T., Kessentini, Y.: A deep HMM model for multiple keywords spotting in handwritten documents. Accepted in Pattern Anal. Appl. (2015)
34. Trentin, E.: A survey of hybrid ANN/HMM models for automatic speech recognition. Neurocomputing **1–4**, 91–126 (2001)
35. Viterbi, A.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans. Inf. Theory **13**(2), 260–269 (1967)
36. Vogler, C., Metaxas, D.: A framework for recognizing the simultaneous aspects of American Sign Language. Comput. Vis. Image Underst. **81**, 358–384 (2001)
37. von Agris, U., Zieren, J., Canzler, U., Bauer, B., Kraiss, K.-F.: Recent developments in visual sign language recognition. Univ. Access Inf. Soc. **6**(4), 323–362 (2008)
38. Weiss, D.: HMM based one shot gesture recognition. In: CVPR 2012 Workshop on Gesture Recognition (2012)
39. Wu, D., Zhu, F., Shao, L.: One shot learning gesture recognition from RGBD images. In: CVPR, IEEE, pp. 7–12 (2012)
40. Yang, Yang, Saleemi, I., Shah, M.: Discovering motion primitives for unsupervised grouping and one-shot learning of human actions, gestures, and expressions. IEEE Trans. Pattern Anal. Mach. Intell. **35**(7), 1635–1648 (2013)
41. Zavaliagkos, G., Austin, S., Makhoul, J., Schwartz, R.M.: A hybrid continuous speech recognition system using segmental neural nets with hidden Markov models. IJPRAI **7**(4), 949–963 (1993)