Gesture sequence recognition with one shot learned CRF/HMM hybrid model

Selma Belgacem[a], Clément Chatelain[a], Thierry Paquet[a]

[a]LITIS EA 4108, University of Rouen, Saint-Etienne du Rouvray, France

Abstract

In this paper, we propose a novel markovian hybrid system CRF/HMM for gesture recognition, and a novel motion description method called *gesture signature* for gesture characterisation. The gesture signature is computed using the optical flows in order to describe the location, velocity and orientation of the gesture global motion. We elaborated the proposed hybrid CRF/HMM model by combining the modeling ability of Hidden Markov Models and the discriminative ability of Conditional Random Fields. In the context of one-shot-learning, this model is applied to the recognition of gestures in videos. In this extreme case, the proposed framework achieves very interesting performance and remains independent from the moving object type, which suggest possible application to other motion-based recognition tasks.

Keywords: gesture recognition, one-shot-learning, hybrid system, hidden Markov model, conditional random field, gesture characterisation.

1. Introduction

Following the increasing demand for intuitive and simple human/computer interaction, the gesture analysis and recognition research field has received a lot of attention these last years. A gesture can be defined as a short human body motion, achieved primarily with arms. In some particular situations such as disability or constrained environment, the gesture is the only human/machine communication channel. This study falls into gesture characterization and recognition.

Email address: belgacemselma@yahoo.fr (Selma Belgacem)

The recognition of gesture sequences combines both segmentation and classification. As stated by Sayre [1], segmentation and classification are two tasks that must be performed simultaneously. 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 video gesture can be represented in a simplified three-dimensional space consisting of its two-dimensional projection and its variation through time. The recognition system must be robust to recording environment variationssuch as changes in brightness, backgrounds, colors, objects, signer appearance (clothes, skin color, height, etc.).

Markov models, which are widely applied to the recognition and segmentation of sequential data, model the temporal dependencies in sequences. They are based on the Markovian assumption that accounts 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) [2] allow to introduce a temporal structure between classes representing a high-level knowledge such as a language model. The principle of the HMM is to model the observation generation based on some hidden states. Each observation only depends from the current hidden state (thus assuming observations to be conditionally independent between each other) and each hidden state only depends from the previous state (for an order 1 Markov model). Then, through the inference phase, the most likely sequence of hidden states that describes the given sequence of observations is determined using Viterbi algorithm [3]. On the other hand, HMM's use Gaussian Mixtures (GM) to model the data distribution. When training data are too few, modeling becomes poor and inadequate with GMM, which is a major drawback of HMM's. However, discriminative models, such as Conditional Random Fields (CRF) [4] which are also Markov models, can remedy this problem. The CRF model was proposed by Lafferty et al.[4] in 2001. It has some advantages that can address HMM problems.

CRF's 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. Similar to HMM's, at each time step a label depends on the the previous label (Markov assumption), but may depend on the whole observation sequence making no requirement about the conditional independence of the observation data. 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 post-processing as 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, in this work we propose to combine these two models in a hybrid framework, allowing the integration of knowledge while being robust to different sources of variability. We also propose to characterize gestures using an original global description of 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. In one-shot gesture learning context, this system was tested using the "*Gesture Challenge 1-2*" dataset proposed by ChaLearn 2011-2012 [5, 6]. We will show that the lack of training data is another problem which can be solved by Markov models to a certain extent.

We will show mainly the principle of our hybrid model CRF/HMM and explain how we adapt it to the one-shot learning context, in order to cope with the lack of training data. We will describe also our gesture characterization model and present the experimental protocol and the evaluation of our system recognition results.

2. Related works

Human gesture analysis is an active research domain with a lot of applications. Among them, many studies have been devoted to gesture recognition, especially the design of automatic systems for recognizing the sign language. Such systems would allow deaf people to better communicate with machines or with other humans.

For gesture sequences recognition, the use of global parallel HMM models is common in the literature [7, 8, 9, 10, 11, 12]. For example Vogler et al. [7], Agris et al. [8] and Ong et al. [9] 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.

Another issue when dealing with real-world problem such as gesture recognition is the lack of labeled examples.

Konecny [10] et al., Jackson [11] and Weiss [12] 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 modeling 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 [11], each frame of the gesture video is represented by a state. This model remain 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 pure generative models can do. One of the first combination scheme was proposed in the 1990s by the integration of neural networks to HMM's [13]. Such combination is prevalent in the literature in various fields. This type of hybrid models was applied to speech recognition [14, 15, 16, 17, 18, 19], handwriting recognition [20, 21, 22, 23, 24, 25, 26] and gesture recognition [27]. HMM models have also been combined with SVM models for handwriting recognition [28] and with dynamic programming methods for gesture recognition [29]. We noticed that the application of these hybrid models to gesture recognition is recent and not much studied in the literature.

To the best of our knowledge, the only work addressing CRF and HMM combination is the work of Soullard et al. [30], based on the work of Gunawardana et al. [31]. In this work, the authors constrain the learning step of a hidden CRF by initialising it with the parameters of a pre-trained 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 posteriors 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 section 3.

3. Hybrid CRF/HMM model

3.1. 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 modeling 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. Let us recall that a label is noted y_t, corresponding to a gesture segment which can span over multiple video frames. An observation is noted x_t, corresponding to a feature vector extracted from one frame. The feature vector is a real valued vector when using the first HMMs devoted to the frame labelling task (the dimension is the feature vector size, see experimental results). This feature vector is later quantified into multiple bins when used by the CRF (see section 3.3) for the gesture recognition task. Its size is defined in section 4. The number of a sequence frames is noted T, it depends of the gesture size. $y_{1:T}$ and $x_{1:T}$ are respectively noted by Y, and X. X_d presents the quantified feature vector. Figure 1 shows the proposed hybrid system.

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

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

In the classic form of HMMs, $p(x_t|y_t)$ is a Gaussian mixture. In our new model, this distribution will be, in some way, replaced by the categorical distribution $p(y_t|x_t)$ computed by the CRFs. Indeed, $p(x_t|y_t)$ 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 :

$$p(x_t|y_t) = \frac{p(y_t|x_t)p(x_t)}{p(y_t)} \tag{2}$$

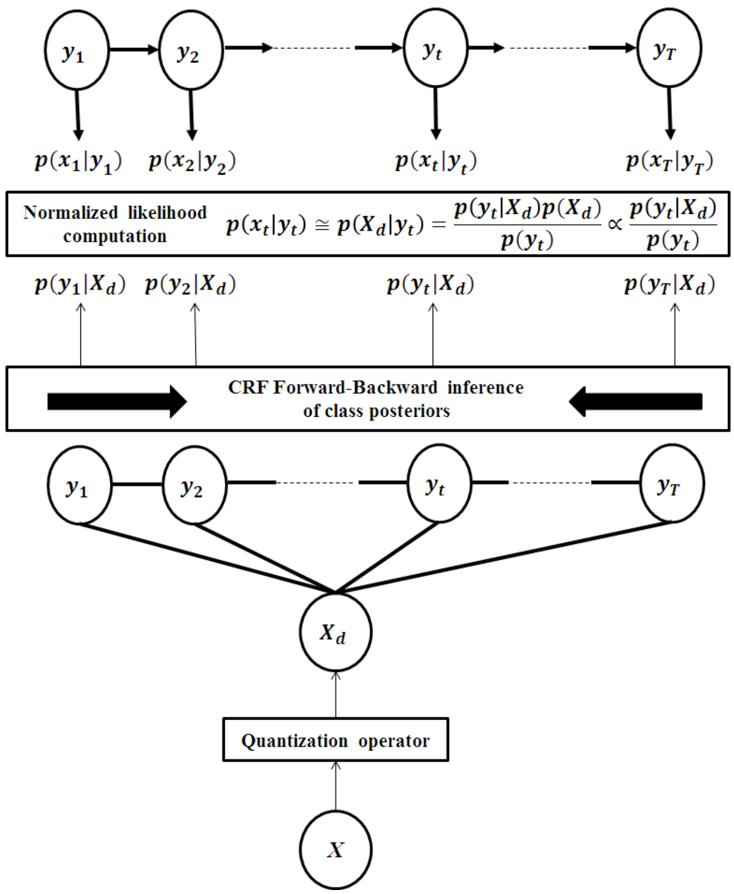

Figure 1: The graphical model CRF/HMM : the HMM joint probability $p(y_{1:T}, x_{1:T})$ for the observation sequence X and the state sequence $y_{1:T}$ is computed using CRF local class posterior probabilities $p(y_t|x_t)$

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 maximises $p(y_{1:T}, x_{1:T})$. 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, we assume that $p(y_t|x_t) = p(y_t|x_{1:T}) = p(y_t|X) \cong p(y_t|X_d)$.

This is computed within the CRF using the forward-backward algorithm [32], where the forward probability α_t and the backward probability β_t are computed using the following recurrences:

$$\alpha_t(i) = \sum_{j=1}^{N_s} \alpha_{t-1}(j)\psi_t(s_i, s_j, o_l) \tag{3}$$

$$\beta_t(i) = \sum_{j=1}^{N_s} \beta_{t+1}(j)\psi_{t+1}(s_i, s_j, o_l) \tag{4}$$

where

$$\psi_t(s_i, s_j, o_l) = \exp(\sum_{k=1}^{K} \lambda_k f_k(y_t = s_i, y_{t-1} = s_j, x_t = o_l)) \tag{5}$$

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

$$p(X_d) = \sum_{j=1}^{N_s} \alpha_T(j) = \sum_{j=1}^{N_s} \beta_1(j) = \sum_{j=1}^{N_s} \alpha_t(j)\beta_t(j) \tag{6}$$

$$p(y_t = s_i|X_d) = \frac{p(y_t = s_i, X_d)}{p(X_d)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N_s} \alpha_t(j)\beta_t(j)} = \gamma_t(i) \tag{7}$$

3.2. Training the CRF/HMM model

We chose to achieve a separate training of the HMMs and the CRF. As first stage, HMMs are trained with the standard Baum Welch algorithm which means that the target function is the likelihood of the global gesture model. Transition probability Matrices are learned separately for each gesture class, and gathered into a global model for decoding gesture sequences. This model is described in section 3.5.

In the second stage, CRF are trained with the classic LBFGS algorithm. As CRF do not benefit from an embedded training procedure like HMM, the target function of this training phase is the local frame level classe (state) posterior. Therefore local frame level labels are necessary. In this respect we introduce a frame level labelling stage that consists in using the HMM model of gesture trained on the dataset, in a forced alignment mode. The frame labels produced serve as the objective target of the of the cost function for training the CRF. During this second training phase the CRF learns a single model for all 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.

The training chain is furthermore explained in figure 3.

3.3. 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 [10, 11, 12, 33, 6]. These system are generally made 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 parameters of the Gaussian distribution $p(x_t|y_t)$ 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[34, 35]; however, since the CRF classification stage is derived from a logistic regression, it is more adapted to discrete features than continuous [36]. This is even more true when the number of samples is small. Indeed, in the context of one shot learning the loss of information induced by the discretization of continuous features may have a regularization effect when training the CRF with one single example. Feature quantization also allows to efficiently tune the parameters linked to each discrete feature value. Although quantization involves a loss of information, the integration of a large set of features allows to capture a global representation of the whole gesture. Therefore, we turned toward the use of a feature quantization procedure. Notice that some recent developments have introduced Hidden CRF models in order to cope with continuous features [37]. 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:

$$\begin{array}{rccc} Q: & [-\mathtt{V_{max}}, \mathtt{V_{max}}] & \longrightarrow & [-\mathtt{N_q}, \mathtt{N_q}] \\ & x & \longmapsto & \frac{x \times \mathtt{N_q}}{\mathtt{V_{max}}} \end{array} \tag{8}$$

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.

3.4. 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 auto-transitions, 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 hyper-parameter 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 figure 1. 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 $\mathtt{w_o}$ (equation 9). $\mathtt{f_w}$ is statistically 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.

$$\mathtt{f_w}(\mathtt{w_o}(\mathbb{G})) = \min(\frac{3}{4}\min_{\mathtt{g}\in\mathbb{G}} \mathtt{f_g}(\mathtt{g}), \text{threshold}) \tag{9}$$

3.5. 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 Figure 2. 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.

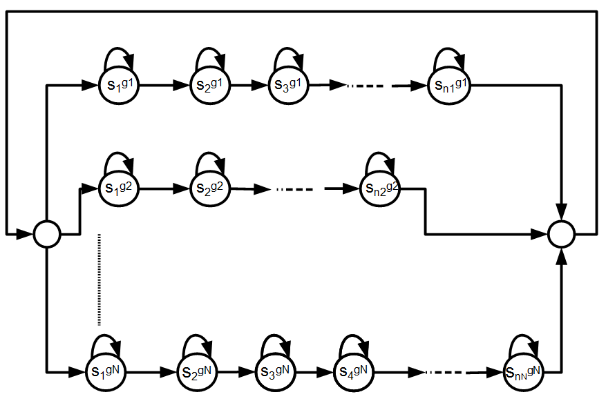

Figure 2: The recognition model of gesture sequences using HMM. $s_j^{g_i}$ represents the state j of the gesture i

3.6. CRF/HMM algorithm

Algorithm 1 summarizes the training and decoding process of our hybrid CRF/HMM model for gesture sequences recognition. Table 1 details algorithm functions and variables description. GSHOG characterisation function extract features from videos using Gesture Signature and HOG methods explained in section 4. CRF/HMM procedure is furthermore explained by the diagram represented in figure 3.

```
Begin CRFHMMRecognition(Databases:videos):sequences
    for all Databases do
        for all GestClassVideos do
            LFeatFile←GSHOG(GestClassVideo)
        end for
        BawmWelch(HMMParams,LFeatFiles)
        UnifyHMMGaussVar(HMMParams.GaussVar)
        LabGestClassVideos←Viterbi(HMMParams,LabList,LFeatFile)
        QLFeatFiles←Quantify(LFeatFiles)
        LBFGS(CRFParams,QLfeatFiles,LabGestClassVideos)
        for all TestSeqVideos do
            TFeatFile←GSHOG(TestSeqVideo)
        end for
        QTFeatFiles←Quantify(TFeatFiles)
        PosterioriProbas←ForwardBackward(CRFParams,TFeatFiles)
        GestSeqs←CRFHMMViterbi(HMMParams.TransitionProbas,
         PosterioriProbas,LabList,SizeSeqs)
    end for
End
```

Algorithm 1: CRF/HMM learning and decoding algorithm

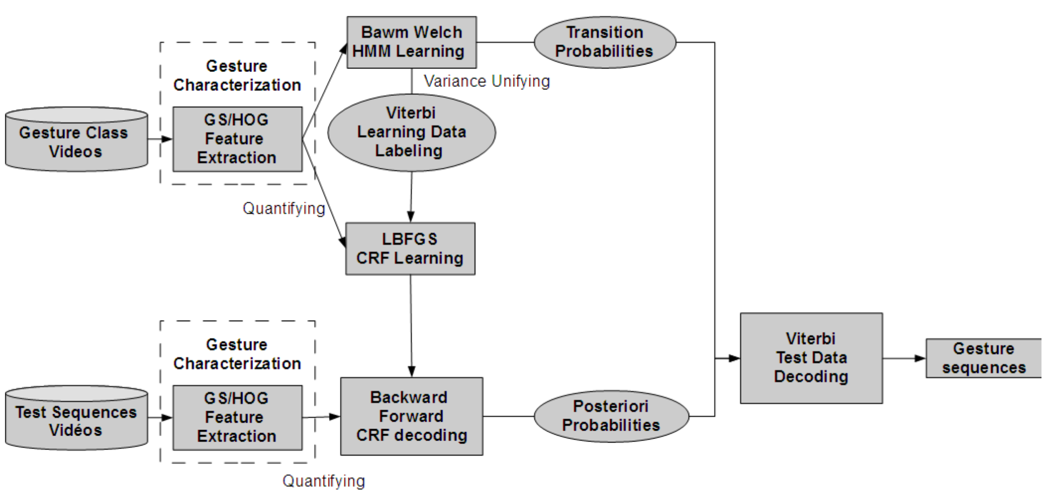

Figure 3: The CRF/HMM training and decoding process diagram

Function	Description
GSHOG	feature extraction (see section 4)
BawmWelch	HMM learning algorithm
UnifyHMMGaussVar	HMM adaptation
Viterbi	Learning videos labeling
Quantify	CRF adaptation
LBFGS	CRF learning method
ForwardBackward	posteriori probabilities computing
CRFHMMViterbi	Test sequences decoding
Variable	**Description**
Databases	learning and test databses
GestClassVideos	videos of gesture classes
LFeatFiles	feature files of gesture class videos
HMMParams	HMM parameters
HMMParams.GaussVar	HMM Gaussian variables
LabGestClassVideos	labeled video frames of gesture classes
LabList	gesture class labels
QLFeatFiles	quantified feature files of gesture class videos
CRFParams	CRF parameters
TestSeqVideos	videos of test sequences
TFeatFiles	feature files of test sequence videos
QTFeatFiles	quantified feature files of test sequence videos
PosterioriProbas	posteriori probabilities
HMMParams.TransitionProbas	global transition probabilities
SizeSeqs	size of test sequences
GestSeqs	recognized gesture sequences

Table 1: Algorithm 1 Functions and variables description

4. Global gestures characterization

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

In this section, we propose a second contribution presenting a model for the gesture characterisation : a set of motion descriptors deduced from optical flows velocities. We call this set of descriptors *Gesture Signature* (GS).

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.[38]. 9 directions are used to quantify gradients inclination angles calculated on the image. Such descriptors will account for shape descriptors.

4.1. Characterization with optical flows : Gesture Signature

Optical flows describe local velocities at the pixel level. They are known for their robustness to brightness changes [39]. They are invariant to colors 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 color, 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. 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 9 descriptors derived from positive and negative horizontal components $\mathtt{V}_{\mathtt{X}}^{+}$ and $\mathtt{V}_{\mathtt{X}}^{-}$, and 9 descriptors derived from vertical components $\mathtt{V}_{\mathtt{Y}}^{+}$ and $\mathtt{V}_{\mathtt{Y}}^{-}$. These components are derived from optical flows at each pixel of the image at position p. 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 9 descriptors consist of 4 movement *location* descriptors, 2 movement *velocity* descriptors and 3 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 2 shows the 18 features set (related variables are defined in table 3). The 8 horizontal and vertical location features are related to inertia center coordinates. They represent the vertical and horizontal positions of velocity centers with respect to the global movement of the considered portion of the image.

There are 4 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. $\mathtt{V}^*_{\mathtt{X}}$ and $\mathtt{V}^*_{\mathtt{Y}}$ components are the medians of a thresholded velocity vector which is computed with optical flows. Values of the threshold are given below.

$$S_{V_X} = \frac{\sum_{p=1}^{\mathtt{N}^s_{\mathtt{px}}} |\mathtt{V}_{\mathtt{X}}(p)|}{\mathtt{N}^s_{\mathtt{px}}} \tag{10}$$

$$S_{V_Y} = \frac{\sum_{p=1}^{\mathtt{N}^s_{\mathtt{px}}} |\mathtt{V}_{\mathtt{Y}}(p)|}{\mathtt{N}^s_{\mathtt{px}}} \tag{11}$$

The 6 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 shows the interest of these descriptors and illustrates the symmetry information. Thus, by analyzing 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.

5. Experimental protocol

In this section, we explain the experimental protocol : databases and evaluation methods

Table 2: The 8 movement **location** features, the 4 motion **velocity** features and the 6 movement **orientation** features of the *Gesture Signature* characterisation model.

	Descriptor	horizontally	vertically
Location	Average Abscissa of pixels moving in the Positive direction (AAP)	$\frac{1}{I_w} \times \frac{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{+}} \lvert\mathtt{V}_{\mathtt{X}}^{+}(p)\rvert x_p}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{+}} \lvert\mathtt{V}_{\mathtt{X}}^{+}(p)\rvert}$	$\frac{1}{I_w} \times \frac{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{+}} \lvert\mathtt{V}_{\mathtt{Y}}^{+}(p)\rvert x_p}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{+}} \lvert\mathtt{V}_{\mathtt{Y}}^{+}(p)\rvert}$
	Average Ordinate of pixels moving in the Positive direction (AOP)	$\frac{1}{I_h} \times \frac{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{+}} \lvert\mathtt{V}_{\mathtt{X}}^{+}(p)\rvert y_p}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{+}} \lvert\mathtt{V}_{\mathtt{X}}^{+}(p)\rvert}$	$\frac{1}{I_h} \times \frac{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{+}} \lvert\mathtt{V}_{\mathtt{Y}}^{+}(p)\rvert y_p}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{+}} \lvert\mathtt{V}_{\mathtt{Y}}^{+}(p)\rvert}$
	Average Abscissa of pixels moving in the Negative direction (AAN)	$\frac{1}{I_w} \times \frac{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{-}} \lvert\mathtt{V}_{\mathtt{X}}^{-}(p)\rvert x_p}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{-}} \lvert\mathtt{V}_{\mathtt{X}}^{-}(p)\rvert}$	$\frac{1}{I_w} \times \frac{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{-}} \lvert\mathtt{V}_{\mathtt{Y}}^{-}(p)\rvert x_p}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{-}} \lvert\mathtt{V}_{\mathtt{Y}}^{-}(p)\rvert}$
	Average Ordinate of pixels moving in the Negative direction (AON)	$\frac{1}{I_h} \times \frac{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{-}} \lvert\mathtt{V}_{\mathtt{X}}^{-}(p)\rvert y_p}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{-}} \lvert\mathtt{V}_{\mathtt{X}}^{-}(p)\rvert}$	$\frac{1}{I_h} \times \frac{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{-}} \lvert\mathtt{V}_{\mathtt{Y}}^{-}(p)\rvert y_p}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}^{-}} \lvert\mathtt{V}_{\mathtt{Y}}^{-}(p)\rvert}$
Velocity	Global Velocity Inverse (GVI)	$\sqrt{\frac{\mathtt{N}_{\mathtt{px}}}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}} (\mathtt{V}_{\mathtt{X}}(p))^2}}$	$\sqrt{\frac{\mathtt{N}_{\mathtt{px}}}{\sum_{p=1}^{\mathtt{N}_{\mathtt{px}}} (\mathtt{V}_{\mathtt{Y}}(p))^2}}$
	Maximum Velocities Median (MVM)	$\frac{1}{S_{V_X}} \times \lvert\mathtt{V}_{\mathtt{X}}^{*}\rvert$	$\frac{1}{S_{V_Y}} \times \lvert\mathtt{V}_{\mathtt{Y}}^{*}\rvert$
Orientation	Proportion of the Pixels moving in the Positive direction (PPP)	$PPP_{\mathtt{X}} = \frac{\mathtt{N}_{\mathtt{px}}^{\mathtt{V}_{\mathtt{X}}^{+}}}{\mathtt{N}_{\mathtt{px}}}$	$PPP_{\mathtt{Y}} = \frac{\mathtt{N}_{\mathtt{px}}^{\mathtt{V}_{\mathtt{Y}}^{+}}}{\mathtt{N}_{\mathtt{px}}}$
	Proportion of the Pixels moving in the Negative direction (PPN)	$PPN_{\mathtt{X}} = \frac{\mathtt{N}_{\mathtt{px}}^{\mathtt{V}_{\mathtt{X}}^{-}}}{\mathtt{N}_{\mathtt{px}}}$	$PPN_{\mathtt{Y}} = \frac{\mathtt{N}_{\mathtt{px}}^{\mathtt{V}_{\mathtt{Y}}^{-}}}{\mathtt{N}_{\mathtt{px}}}$
	Dominant Orientation (DO)	$DO_{\mathtt{X}} = \frac{\mathtt{N}_{\mathtt{px}}^{\mathtt{V}_{\mathtt{X}}^{+}} - \mathtt{N}_{\mathtt{px}}^{\mathtt{V}_{\mathtt{X}}^{-}}}{\mathtt{N}_{\mathtt{px}}}$	$DO_{\mathtt{Y}} = \frac{\mathtt{N}_{\mathtt{px}}^{\mathtt{V}_{\mathtt{Y}}^{+}} - \mathtt{N}_{\mathtt{px}}^{\mathtt{V}_{\mathtt{Y}}^{-}}}{\mathtt{N}_{\mathtt{px}}}$

Variable	Description
I_w	image width
I_h	image height
N_{px}	total pixel number
$N_{px}^{V_X^+}$	number of pixels moving in the positive horizontal direction
$N_{px}^{V_X^-}$	number of pixels moving in the negative horizontal direction
$N_{px}^{V_Y^+}$	number of pixels moving in the positive vertical direction
$N_{px}^{V_Y^-}$	number of pixels moving in the negative vertical direction
$V_X^+(p)$	positive horizontal velocity component of a pixel p
$V_X^-(p)$	negative horizontal velocity component of a pixel p
$V_Y^+(p)$	positive vertical velocity component of a pixel p
$V_Y^-(p)$	negative vertical velocity component of a pixel p
$V_X(p)$	horizontal velocity component of a pixel p
$V_Y(p)$	vertical velocity component of a pixel p
V_X^*	median of horizontal components (absolute value) of pixel velocities
V_Y^*	median of vertical components (absolute value) of pixel velocities
S_{V_X}, S_{V_Y}	velocity thresholds (see equations 10 and 11)
PPP_X	Proportion of the Pixels moving in the Positive horizontal direction
PPN_X	Proportion of the Pixels moving in the Negative horizantal direction
PPP_Y	Proportion of the Pixels moving in the Positive vertical direction
PPN_Y	Proportion of the Pixels moving in the Negative vertical direction

Table 3: Gesture Signature variables description

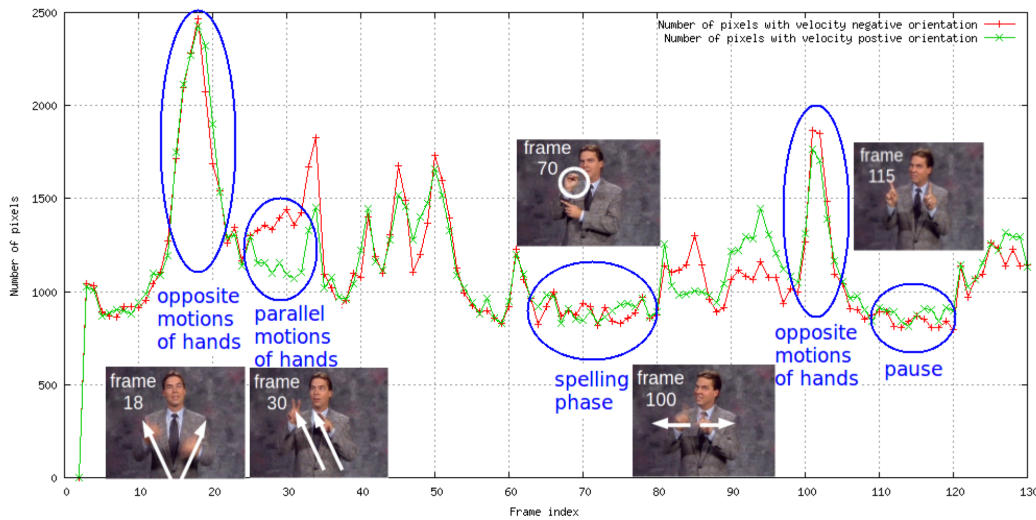

Figure 4: Evolution of the descriptors $PPP_{\mathtt{X}}$ (Proportion of the Pixels moving in the Positive horizontal direction) and $PPN_{\mathtt{X}}$ (Proportion of the Pixels moving in the Negative horizontal direction) in a video from SignStream database [40]. Two curves superimposed with a presence of a peak correspond to an opposite movement of the two hands. A strong difference between the two curves correspond to a parallel movement of both hands in the dominant direction. A stagnation of the two curves correspond to fixed hands (frame 70).

5.1. Databases

Our recognition system has been evaluated on public databases designed for the ChaLearn 2011-2012 competition [5]. 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 [1]. We detail the results of this evaluation in section 6.

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 color format and depth format. These videos are recorded using a Kinect (TM) camera. 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 to 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.

5.2. Feature vector variants

Table 4 presents the different variants of the feature vector $\vec{c}$ we used in our experiments. We index each variant by its size $l(\vec{c})$. $l(\vec{v}(GS))$ is the number of gesture signature features. $l(\vec{v}(HOG))$ is the number of HOG

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

features. Some variants of the feature vector $\vec{c}$ are applied to two data formats (RGB image and depth image).

Table 4: Feature vector variants adopted in the experiments

total size $l(\vec{c})$	**Descriptor**			
	Gesture Signature GS		**HOG**	
	$l(\vec{c}(GS))$	**description**	$l(\vec{c}(HOG))$	**description**
52	16	no median, no image division	36	4 image blocks × 9 gradient directions
54	18	no image division	36	4 image blocks × 9 gradient directions
72	72	image division into 2 parts, 2 data formats	0	HOG not applied
180	36	image division into 2 parts	144	16 image blocks × 9 gradient directions
360	72	image division into 2 parts, 2 data formats	288	16 image blocks × 9 gradient directions, 2 data formats

5.3. 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 [41]. This form of global error is denoted by $\mathcal{L}_{\texttt{ch}}$ and given by equation 12.

$$\begin{array}{rcl} \mathcal{L}_{\texttt{ch}}: \quad \mathbb{D} & \longrightarrow & \mathbb{R} \\ \mathbb{S} & \longmapsto & \frac{\sum_{s\in\mathbb{S}} \mathrm{L}(\mathcal{R}(s),\mathcal{T}(s))}{\sum_{s\in\mathbb{S}} l(\mathcal{T}(s))} \end{array} \tag{12}$$

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

We use the ChaLearn form of the error $\mathcal{L}_{\texttt{ch}}$ to compare our recognition system to ChaLearn participants recognition systems. However, let us emphasize that $\mathcal{L}_{\texttt{ch}}$ is slightly different from the classical Levenshtein distance (see

Equation 13), which is bounded and seems more generic. Thus, to present the main results of our various tests, we use the classic error form.

$$\begin{array}{rccl} \mathcal{L}: & \mathbb{D} & \longrightarrow & [0,1] \\ & \mathbb{S} & \longmapsto & \frac{1}{|\mathbb{S}|}\sum_{s\in\mathbb{S}} \frac{\mathrm{L}(\mathcal{R}(s),\mathcal{T}(s))}{l(\mathcal{R}(s))+l(\mathcal{T}(s))} \end{array} \quad (13)$$

6. Gesture recognition results

In this section we present the results of our system, using different variants. We first compare the recognition results of the hybrid system CRF/HMM to the classic and adapted versions of HMM and CRF in subsection 6.1. Then, we present our rank compared to participants at the ChaLearn competition. Next, in subsection 6.2, we present some properties of the hybrid model CRF/HMM including its robustness with respect to the number of states and to the various feature vectors, and we conclude this section by demonstrating the advantage of the gesture signature model.

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 section 3.3 unless otherwise stated. Adapted HMM and adapted CRF recognition systems cited in this section are also adapted as explained in section 3.3.

6.1. 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 5) on the *devel* databases. The feature vector is identical for all the systems ($l(\vec{c}) = 52$). The number of frames per state $\mathtt{f_s}$ is 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 post-processing step (see algorithm 2) is applied to the classic and adapted CRF in order to filter their recognition results. Without this step recognition error exceeds 0.5. Table 5 shows that the performance of the proposed hybrid system CRF/HMM clearly outperform the recognition performances of other systems. Indeed, for a data size equal to 750, we demonstrated with the statistical unilateral Student test that our hybrid model CRF/HMM significantly outperforms classic and adapted HMM and CRF models with a confidence level detailed in table 5.

Begin ***CRFpostProcessing(RecognizedSeqs):FilteredSeqs***
  for *all RecognizedSeqs* S_i **do**
    for *all gestures* G_j **do**
      Fix window size $Sz(w_j) = \begin{cases} \frac{3}{4}Sz(G_j) & \text{if } 1 < \frac{Sz(S_i)}{Sz(G_j)} \\ Sz(S_i) & \text{if } 0 < \frac{Sz(S_i)}{Sz(G_j)} \leq 1 \end{cases}$
      for *all shifted window positions (shifting step is* $Sz(w_j)$*)* **do**
        Search for the most occurent gesture G_m in the current window
        if $(G_m = G_j)$ **then**
          save G_m at the current window position (if conflict, keep the shortest gesture)
        end
      end for
    end for
    FilteredSeq $\leftarrow$ all saved G_m
  end for
End

Algorithm 2: Post-filtering and segmentation algorithm for CRF recognized sequences, where *FiltredSeqs* are filtered and segmented sequences extracted from *RecognizedSeqs* which are the recognized sequences with CRF Backward-Forward method

Table 5: a) The recognition results of various recognition systems based on HMM and CRF and tested on 20 *devel* databases ($\mathtt{f_s}$ is optimized for each system, $l(\vec{c}) = 52$ and images are RGB). b) Unilateral student test results (confedence level): CRF/HMM campared to classic and adapted HMM and CRF models.

System	**(a) Error : $\mathcal{L}$**	**(b) Student confidence level (%)**
classic HMM	0.3615	99.95
adapted HMM	0.2354	85
classic CRF (continuous)	0.2930	99.95
adapted CRF (discrete)	0.2757	99.95
CRF/HMM (adapted)	**0.2228**	-

In order to rank our system in the ChaLearn 20111-2012 competition, we tested the hybrid system on *valid* and *final* databases provided during the competition. Table 6 shows the hybrid system CRF/HMM recognition error values computed with both evaluation methods $\mathcal{L}$ and $\mathcal{L}_{\texttt{ch}}$ on *valid* and *final* databases. Table 6 presents the CRF/HMM system rank on both database categories using the $\mathcal{L}_{\texttt{ch}}$ error. It appears that we ranked at the 7^{th} position among 559 systems from 48 participants for both first and second rounds. The complete list with their score (the $\mathcal{L}_{\texttt{ch}}$ error) is available on the Kaggle website for the first[2] and the second round[3]. We achieved this rank using only RGB format data.

Table 6: 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, images are RGB and $l(\vec{c}) = 180$)

database category	**Error**		**ranking**
	$\mathcal{L}$	$\mathcal{L}_{\texttt{ch}}$	
valid	0.1772	0.3488	-
final 1-20 (1^{st} round)	0.1479	0.2964	7^{th}
final 21-40 (2^{nd} round)	0.1224	0.2523	7^{th}

6.2. Properties of the CRF/HMM system

6.2.1. Robustness to changes in the number of frames per state

Figure 5 shows the recognition error $\mathcal{L}$ of the adapted HMM and the CRF/HMM systems with respect to the number of frames per state $\texttt{f}_{\texttt{s}}$. Those systems were trained for each value of $\texttt{f}_{\texttt{s}}$. One can observe that the CRF/HMM system outperforms the HMM, and that the CRF/HMM system provides extremely stable results, while the performance of the HMM is strongly variable. This is an interesting aspect of this system since it does not require a fine tuning of the hyper-parameter for reaching good results.

6.2.2. Robustness to changes in the gesture duration

The average 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, the CRF/HMM system is able to handle the

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

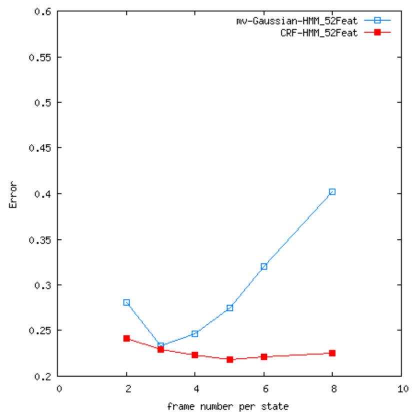

Figure 5: CRF/HMM and adapted HMM systems robustness to the variation of the number of frames per state.

Table 7: Recognition results with different variants of the feature vector on 20 *devel* databases (RGB image and depth image)

System	**Error : $\mathcal{L}$**	
	GS $l(\vec{c}) = 72$	(GS,HOG) $l(\vec{c}) = 360$
adapted HMM	0.2525	0.2425
CRF/HMM	0.2559	0.2255

temporal elasticity of a gesture. In other words, when a gesture expands or narrows through the number of frames in the test data, the CRF/HMM system is able to align the gesture model on the data and decode them. In addition, the CRF component is able to implicitly manage duration variation of gestures in a more straightforward way than HMMs do. It appears that the temporal elasticity of gestures is better captured by the simple structure of the hybrid model with a reduced number of states, compared to the totally connected structure of the HMM system alone as adopted by some participants of the ChaLearn competition [10, 11, 12].

6.2.3. Robustness to changes in the feature vector

Figures 6 present the variation of the error $\mathcal{L}$ 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, feature addition greatly increases 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

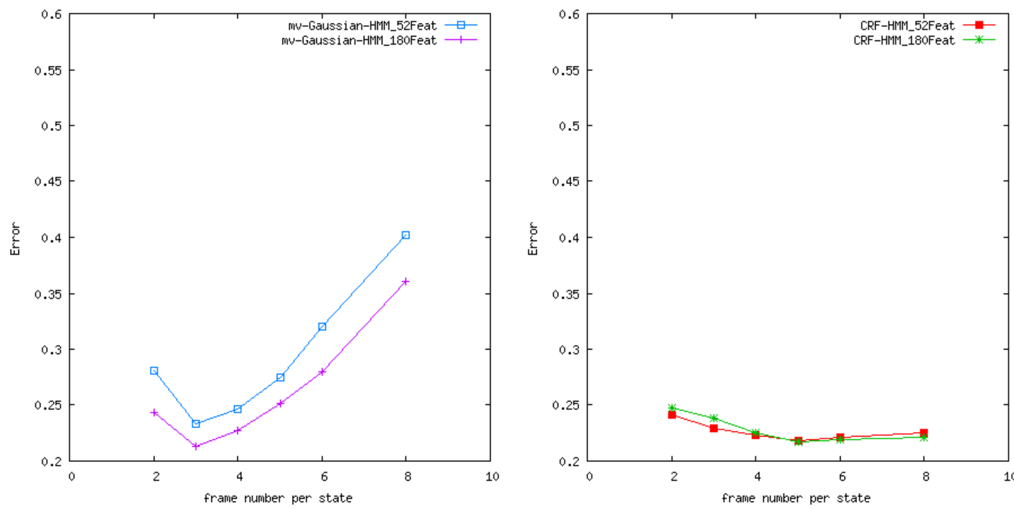

Figure 6: Adapted HMM (left) and CRF/HMM (right) robustness to the variation of the feature vector

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. (c) a training data set reduced to an example by class.

6.2.4. Evaluation of the Gesture Signature

Table 7 shows the recognition results of two systems, adapted HMM and CRF/HMM, on *devel* databases applying three variants of the feature vector. The purpose of these tests is not to compare these two recognition systems but to validate the interest of the feature vector GS. According to table 7, we notice that the performance of recognition systems with the feature vector GS is very close to the performance of these recognition systems using a feature vector that combines GS features and the HOG features. Moreover, these error values are low and exhibit valuable recognition performance. Thus, the gesture signature GS can represent a complete characterization model.

Finally, 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.

7. Conclusion

In this article, an hybrid CRF/HMM system for gesture recognition has been proposed. This HMM and CRF combination benefits from each model advantages without undergoing its drawbacks. These systems have been adapted to the one-shot learning context in order to suit to the real-world constraints of small labelled datasets.

A new gesture characterization model has also been proposed, which is a gesture signature that rely on optical flows. This model is able to describe any dynamic scene using its motion, making it independent from the moving object type.

We demonstrate that the CRF/HMM system are able to efficiently model and manage spatio-temporal variations of sequential data and constitute a robust recognition hybrid system that opens up new perspectives for sequential Markov models. Among them, an interesting perspective concerns the gesture detection task, called the gesture *spotting*, which consists on locating and labelling specific gestures in videos. It can be applied in video retrieval and indexing context. Our recognition model could be adapted to the spotting task by representing false examples through an additional class to the gestures vocabulary.

References

[1] K. M. Sayre, Machine recognition of handwritten words: A project report, Pattern Recognition 5 (3) (1973) 213 – 228.

[2] L. R. Rabiner, A tutorial on hidden markov models and selected applications in speech recognition, Proceedings of the IEEE (1989) 257–286.

[3] A. Viterbi, Error bounds for convolutional codes and an asymptotically optimum decoding algorithm, IEEE Transactions on Information Theory 13 (2) (1967) 260–269.

[4] J. Lafferty, A. McCallum, F. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data, ICML (2001) 282–289.

[5] I. Guyon, V. Athitsos, P. Jangyodsuk, B. Hamner, H. Escalante, Chalearn gesture challenge: Design and first results., CVPR Workshops (2012) 1–6.

[6] Y. Yang, I. Saleemi, M. Shah, Discovering motion primitives for unsupervised grouping and one-shot learning of human actions, gestures, and expressions, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (7) (2013) 1635–1648.

[7] C. Vogler, D. Metaxas, A framework for recognizing the simultaneous aspects of american sign language, Computer Vision and Image Understanding 81 (2001) 358–384.

[8] U. von Agris, J. Zieren, U. Canzler, B. Bauer, K. K-F, Recent developments in visual sign language recognition, Univers. Access Inf. Soc. 6 (4) (2008) 323–362.

[9] S. Ong, S. C. W.and Ranganath, Deciphering gestures with layered meanings and signer adaptation, IEEE International Conference on Automatic Face and Gesture Recognition (2004) 559.

[10] J. Konencny, M. Hagara, One-shot learning gesture recognition using hog/hof features, ICPR 2012 Workshop on Gesture Recognition.

[11] E. Jackson, An hmm-based approach for gesture recognition using edge features, CVPR 2012 Workshop on Gesture Recognition.

[12] D. Weiss, Hmm based one shot gesture recognition, CVPR 2012 Workshop on Gesture Recognition.

[13] E. Trentin, A survey of hybrid ANN/HMM models for automatic speech recognition, Neurocomputing (2001) 91–126.

[14] N. Morgan, H. Bourlard, S. Renls, M. Cohen, H. Franco, Hybrid neural network/hidden markov model systems for continuous speech recognition, IJPRAI 7 (4).

[15] L. T. Niles, H. F. Silverman, Combining hidden Markov models and neural network classifiers, in: ICASSP, 1990, pp. 417–420.

[16] J. Tebelskis, A. Waibel, B. Petek, O. Schmidbauer, Continuous speech recognition by linked predictive neural networks, NIPS (1990) 199–205.

[17] F. Johansen, A comparison of hybrid hmm architectures using global discriminative training., ICSLP.

[18] G. Rigoll, Maximum mutual information neural networks for hybrid connectionist-hmm speech recognition systems., IEEE Transactions on Speech and Audio Processing 2 (1) (1994) 175–184.

[19] G. Zavaliagkos, S. Austin, J. Makhoul, R. Schwartz, A hybrid continuous speech recognition system using segmental neural nets with hidden markov models., IJPRAI 7 (4) (1993) 949–963.

[20] S. Thomas, C. Chatelain, L. Heutte, T. Paquet, Y. Kessentini, A deep hmm model for multiple keywords spotting in handwritten documents, Pattern Analysis and Applications 18 (4) (2015) 1003–1015.

[21] Y. Bengio, Y. LeCun, C. Nohl, C. Burges, Lerec: A nn/hmm hybrid for on-line handwriting recognition, Neural Comput. 7 (6) (1995) 1289–1303.

[22] S. Knerr, E. Augustin, A neural network-hidden markov model hybrid for cursive word recognition, ICPR 2 (1998) 1518–1520.

[23] S. Marukatat, T. Artieres, P. Gallinari, B. Dorizzi, Sentence recognition through hybrid neuro-markovian modeling., in: ICDAR, 2001, pp. 731–.

[24] M. Gilloux, B. Lemarie, M. Leroux, A hybrid rbf network/hidden markov model handwritten word recognition system., in: ICDAR, 1995, pp. 394–397.

[25] M. Morita, R. Sabourin, F. Bortolozzi, C. Suen, Segmentation and recognition of handwritten dates: an hmm-mlp hybrid approach., IJDAR 6 (4) (2003) 248–262.

[26] O. Matan, C. Burges, Y. Lecun, J. Denker, Multi-digit recognition using a space displacement neural network, in: Advances in Neural Information Processing Systems 4, 1992, pp. 488–495.

[27] A. Corradini, Real-time gesture recognition by means of hybrid recognizers., in: Gesture Workshop, Vol. 2298, Springer, 2001, pp. 34–46.

[28] A. Ganapathiraju, J. Hamaker, J. Picone, Hybrid svm/hmm architectures for speech recognition., in: INTERSPEECH, ISCA, 2000, pp. 504–507.

[29] S. Rajko, G. Qian, A hybrid hmm/dpa adaptive gesture recognition method., in: ISVC, Vol. 3804, 2005, pp. 227–234.

[30] Y. Soullard, T. Artieres, Hybrid hmm and hcrf model for sequence classification, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning.

[31] A. Gunawardana, M. Mahajan, A. Acero, J. Platt, Hidden conditional random fields for phone classification., in: INTERSPEECH, ISCA, 2005, pp. 1117–1120.

[32] S. Austin, R. Schwartz, P. Placeway, The forward-backward search algorithm, IEEE ICASSP (1991) 697–700.

[33] D. Wu, F. Zhu, L. Shao, One shot learning gesture recognition from rgbd images., CVPR, IEEE (2012) 7–12.

[34] C. Sutton, A. McCallum, An introduction to conditional random fields, Foundations and Trends in Machine Learning 4 (4) (2012) 267373.

[35] V. Radosavljevic, S. Vucetic, Z. Obradovic, Continuous conditional random fields for regression in remote sensing, in: Proceedings of the 2010 Conference on ECAI 2010: 19th European Conference on Artificial Intelligence, IOS Press, Amsterdam, The Netherlands, The Netherlands, 2010, pp. 809–814.

[36] D. Hebert, T. Paquet, S. Nicolas, Continuous crf with multi-scale quantization feature functions application to structure extraction in old newspaper, in: ICDAR, 2011, pp. 493–497.

[37] A. Quattoni, S. Wang, L. Morency, M. Collins, T. Darrell, Hidden conditional random fields, Pattern Analysis and Machine Intelligence, IEEE Transactions on 29 (10) (2007) 1848–1852.

[38] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: International Conference on Computer Vision & Pattern Recognition, Vol. 2, 2005, pp. 886–893.

[39] S. M. Bhandarkar, X. Luo, Integrated detection and tracking of multiple faces using particle filtering and optical flow-based elastic matching, CVIU 113 (6) (2009) 708–725.

[40] C. Neidle, S. Sclaroff, V. Athitsos, Signstream: A tool for linguistic and computer vision research on visual-gestural language data, Behavior Research Methods, Instruments, & Computers 33 (3) (2001) 311–320.

[41] V. Levenshtein, Binary Codes Capable of Correcting Deletions, Insertions and Reversals, Soviet Physics Doklady 10 (1966) 707.