IDIAP RESEARCH REPORT

HMM AND IOHMM FOR THE RECOGNITION OF MONO- AND BI-MANUAL 3D HAND GESTURES

Agnès Just [a] Olivier Bernier [b]
Sébastien Marcel [a]

IDIAP–RR 04-39

JULY 2004

TO APPEAR IN
British Machine Vision Conference, 2004

Dalle Molle Institute for Perceptual Artificial Intelligence • P.O.Box 592 • Martigny • Valais • Switzerland

phone +41 − 27 − 721 77 11
fax +41 − 27 − 721 77 12
e-mail secretariat@idiap.ch
internet http://www.idiap.ch

[a] Institut Dalle Molle d'Intelligence Artificielle Perceptive (IDIAP), CH-1920 Martigny
[b] France Telecom Research & Development, FR-22300 Lannion

IDIAP Research Report 04-39

HMM AND IOHMM FOR THE RECOGNITION OF MONO- AND BI-MANUAL 3D HAND GESTURES

Agnès Just Olivier Bernier Sébastien Marcel

JULY 2004

TO APPEAR IN
British Machine Vision Conference, 2004

Abstract. In this paper, we address the problem of the recognition of isolated complex mono- and bi-manual hand gestures. In the proposed system, hand gestures are represented by the 3D trajectories of blobs obtained by tracking colored body parts. In this paper, we study the results obtained on a complex database of mono- and bi-manual gestures. These results are obtained by using Input/Output Hidden Markov Model (IOHMM), implemented within the framework of an open source machine learning library, and are compared to Hidden Markov Model (HMM).

1 Introduction

Nowadays, Human-Computer Interaction (HCI) is usually done using keyboards, mice or graphic boards. The use of hand gestures for HCI can help people to communicate with computers in a more intuitive way. The potential power of gestures has already been demonstrated in applications that use the hand gesture input to control a computer while giving a presentation for instance. Other possible applications of gesture recognition techniques include computer-controlled games, teleconferencing, robotics or the manipulation of objects by CAD designers. In gestural HCI, the use of video cameras is more natural than any dedicated acquisition device (such as data-gloves for instance) but is also much more challenging. In video-based hand gesture recognition (HGR) it is necessary to distinguish two aspects of hand gestures: the *static* aspect and the *dynamic* aspect. The *static* aspect is characterized by a pose or configuration of the hand in an image. The *dynamic* aspect is defined either by the trajectory of the hand, or by the sequence of hand postures in a sequence of images.

Furthermore, there is two sub-problems to address when dealing with dynamic hand gesture recognition: spotting and classification. On one hand, spotting aims at identifying the beginning and/or the end of a gesture given a continuous stream of data. Usually, this stream of data is made by a random sequence of known gestures and non-gestures. On the other hand, given an isolated gesture sequence, classification outputs the class the gesture belongs to.

In this paper, we will focus on the classification of isolated hand gestures. First, we present an overview of related work on HGR. In section 3, we describe our approach to capture mono- and bi-manual 3D hand gestures, and we describe the database. Then, we introduce Input/Output Hidden Markov Model (IOHMM) and we present experimental results. Finally, we discuss the results and conclude.

2 Related Work

Dynamic HGR is a **sequence processing** problem that can be accomplished by using various techniques. Darell and Pentland in [6] used a vision-based approach to model both object and behavior. The object views were represented using sets of view models. This approach allowed them to learn their model by observation. The disadvantage of this method is that complex articulated objects have a very large range of appearances. Therefore, they used a representation based on interpolation of appearance from a relatively small number of views. The gesture classification is performed by stereotypical space-time patterns (i.e. the gestures) matched with stored gesture patterns using dynamic time warping (DTW). This system was tested on only two gestures. And images were focused on the hand. The experiment was also user-dependent since each of the seven users were involved in both the training and testing phases.

Finite state machine (FSM) was the first technique applied to sequence processing. It was applied to gestures by Davis and Shah [7]. Each gesture was decomposed into four distinct phases. Since the four phases occurred in fixed order, a FSM was used to guide the flow and to recognize seven gestures. Experiments were using a close-up on the hand. Hong et al. [9] proposed another approach based on FSM, that used 2D positions of the centers of the user's head and hands as features. Their system permitted to recognize in real-time four mono-manual gestures. But the most important technique, widely used for dynamic HGR, is Hidden Markov Models. This approach is inspired by the success of the application of HMMs both in speech recognition and in hand-written character recognition fields [1]. Starner and Pentland in [2] used an eight element feature vector consisting of each hand's x and y position, angle of axis of least inertia, and eccentricity of bounding ellipse. They used networks of HMMs to recognize a sequence of gestures taken from the American Sign Language. Training was performed by labeling each sign with the corresponding video stream. They used language modeling to segment the different signs. The Viterbi decoding algorithm was both used with and without a strong grammar based on the known form of the sentences. With a lexicon of forty words, they obtained 91,9% of accuracy in the test phase. Unfortunately, these results are almost impossible to reproduce.

More recently, Marcel et al. [4] have proposed Input/Output Hidden Markov Models (IOHMMs). An IOHMM is based on a non-homogeneous Markov chain where emission and transition probabilities depend on the input. On the opposite, HMMs are based on homogeneous Markov chains since the dynamic of the system is determined only by the transition probabilities which are time independent. Their system was able to recognize four types of gestures with 98,2% of accuracy. Furthermore, this database is publicly available from the Internet [1]. But in their article, they used IOHMM for a small gesture vocabulary (only four gestures).

In most of the studies on hand gestures, small vocabulary has been used. In the next section, we describe a more important database for the recognition of mono- and bi-manual 3D hand gestures.

3 Mono- and Bi-manual 3D Hand Gestures

Most of human activities involve the use of two hands. Furthermore, gestures occur in a 3D space and not in a 2D image plane. In the proposed system, mono- and bi-manual hand gestures are represented by the 3D trajectories of blobs. Blobs are obtained by tracking colored body parts in real-time using the EM algorithm. This approach is similar to the statistical region approach for person tracking [5], for gesture recognition [10].

3.1 Tracking Blobs in 3D

A detailed description of the 3D blob tracking algorithm can be found in [3]. This algorithm tracks head and hands in near real-time (12Hz) using two cameras (Figure 1).

Figure 1: Left: left and right captured images. Center: left and right images with projected ellipsoids. Top right: ellipsoids projection on the frontal plane. Down right: ellipsoids projection on the side plane (the cameras are on the left side).

The algorithm is based on simple preprocessing followed by the use of a **statistical model** linking the observations (resulting from the preprocessing stage) to the parameters: the position of the hands and the head. Preprocessing consists of background subtraction followed by specific colors detection, using a simple color lookup table. The **statistical model** is composed of four ellipsoids, one for each hand, one for the head and one for the torso. Each one is projected on both camera planes as an ellipse. A Gaussian probability density function with the same center and size is associated with each ellipse. The parameters of the model (positions and orientations of the ellipsoids) are adapted to the pixels detected by the preprocessing stage. This adaptation simultaneously takes into account the detected pixels from the two cameras, and is based on the maximum likelihood principle. The EM algorithm is used to obtain the maximum of the likelihood.

3.2 Gesture Database

The database used in this paper has been obtained using the tracking method described above. The database consists of 16 gestures (Table 1) carried out by 20 different persons. Most of gestures are mono-manual and some are bi-manual (*fly*, *swim* and *clap*).

The use of gloves with distinct colors permits to avoid occlusion problems that occur with bi-manual gestures. The person performing the gesture wears gloves of different colors and a sweat-shirt of a specific color

[1] http://www.idiap.ch/~marcel/Databases/main.html

Name	Description	**R**	**M/B**
Stop/yes	Raised hand on the head level and facing palm		M
No/wipe	Idem with movements from right to left	R	M
Raise hand	Raised hand higher than the head		M
Hello/wave	Idem with movements from right to left	R	M
Left	Hand on the hip level, movements to the left	R	M
Right	Hand on the hip level, movements to the right	R	M
Up	Hand on the hip level, movements to the up	R	M
Down	Hand on the hip level, movements to the down	R	M
Front	Hand on the hip level, movements to the front	R	M
Back	Hand on the hip level, movements to the back	R	M
Swim	Swimming mimic gesture	R	B
Fly	Flying mimic gesture	R	B
Clap	On the torso level, clap the hands	R	B
Point left	On the torso level, point to the left		M
Point front	On the torso level, point to the front		M
Point right	On the torso level, point to the right		M

Table 1: Description of the 16 gestures. A hand gesture could involve one hand (**M**ono-manual) or both hands (**B**i-manual). The gesture could be also a **R**epetitive movement such as *clap*.

Figure 2: Left: Example of images of the "Vinci" sequence from the point of view of the left camera (on the left) and from the point of view of the right camera (on the right). Right: 3D coordinates of the center of each blob (head, torso, left hand and right hand) for a "swim" gesture.

different from the skin color and different from the glove colors in order to help the segmentation of hands, head and torso.

For each person and each gesture, there are 5 sessions and 10 shots per session. All the gestures start and end in the same rest position (the hands lying along the thighs). The temporal segmentation was manually accomplished after a recording session. For each gesture, a trajectory for each blob has been generated. Finally, the database is composed of 1000 trajectories per gesture. Gesture trajectories correspond to 3D coordinates of center of the head, of the two hands and of the torso. They are produced with the natural hand (left hand for left-handed and right hand for right-handed persons). For the left-handed persons, trajectories have been mirrored.

Figure 3 shows an example of the swim gesture sequence from the point of view of the right camera. Furthermore, for each person and each session, a "Vinci" sequence has been recorded (Figure 2). This sequence gives the maximum arm spread. This figure presents also in a three dimensional space[2] the coordinates of the center of each blob (head, torso and hands) for a "swim" gesture sequence.

[2]the z axis is the vertical axis of the person.

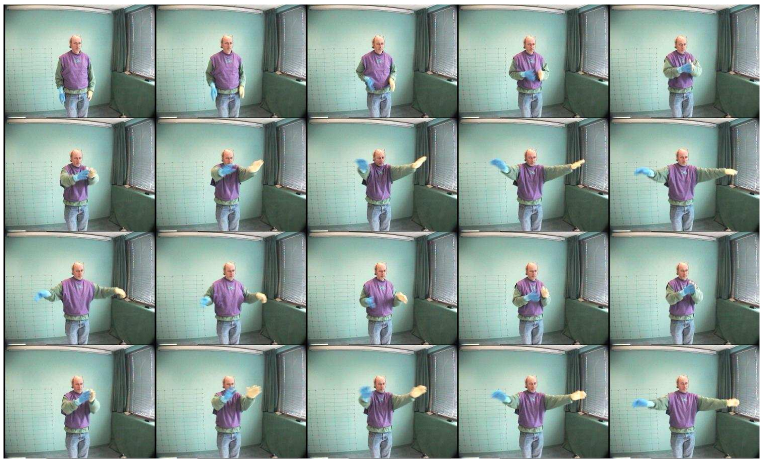

Figure 3: From top-left to bottom-right, a frame-by-frame decomposition of a "swim" gesture from the point of view of the right camera.

4 Hidden Markov Model versus Input Output Hidden Markov Model

4.1 Hidden Markov Model

A Hidden Markov Model (HMM) [1] is a statistical machine learning algorithm which models sequences of data. It consists of a set of N states, called hidden states because non-observable. It also contains transition probabilities between these states and emission probabilities from the states to model the observations.

The data sequence is thus factorized over time by a series of hidden states and emission from these states. Let q_t be the state, y_t be the output (observation) at time t. The emission probability $P(y_t|q_t = i), \forall i = 1 \ldots N$ depends only on the current state q_t. The transition probability between states $P(q_t = i|q_{t-1} = j), \forall i, j = 1 \ldots N$ depends only on the previous state. Then, the training of a HMM can be carried out using the *Expectation-Maximization* (EM) algorithm [8].

As we try to recognize 16 gesture classes, we have one HMM per class and we use a naive Bayes classifier to perform the classification.

4.2 Input Output Hidden Markov Model

An Input Output Hidden Markov Model (IOHMM) is an extension of the HMM described previously. First introduced by Bengio and Frasconi [11], IOHMMs are able to discriminate temporal sequences using a supervised training algorithm. IOHMMs map an input sequence to an output sequence. In our case, output sequence correspond to the class of the gesture. Let q_t be the state, x_t the input and y_t the output at time t. Thus q_t depends on q_{t-1} and x_t. y_t depends on q_t but also depends x_t (cf. Figure 4).

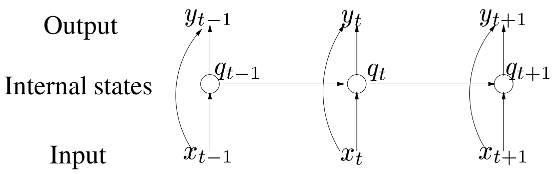

Figure 4: An IOHMM showing dependencies between the input x, output y and hidden states q of the model.

IOHMMs are composed of emission probabilities $P(y_t|q_t, x_t)$ and transition probabilities $P(q_t|q_{t-1}, x_t)$.

They are time dependent since the emission and transition probabilities depend on x_t. Hence IOHMMs are based on non-homogeneous Markov chains contrary to HMMs. Consequently, the dynamic of the system is not fixed *a priori* such as in HMMs, but evolves in time and is function of the input sequence. The architecture of IOHMM also consists of a set of states q. With each state are associated two conditional distributions: one for transition probabilities and one for emission probabilities. The data sequences to model can be of two types: discrete or continuous. In the discrete case, codebooks or multinomial distributions can be used to model the conditional distributions. In the continuous case, models such as Multi Layer Perceptron [12] can be used to represent the conditional distributions. Another solution to deal with continuous observations is to perform a quantization in order to discretize the data. In an IOHMM, there are several ways to classify an input sequence $\mathbf{x}_1^T$. The most restrictive way is to compute $P(\mathbf{y}_1^T|\mathbf{x}_1^T)$, where $\mathbf{y}_1^T$ represents the output vector sequence. We can also compute the average $\frac{1}{N}\sum_{t=1}^{T} P(\mathbf{y}_t|\mathbf{x}_1^t)$ where N is the number of classes. The less restrictive method is to compute $P(\mathbf{y}_T|\mathbf{x}_1^T)$. In all these cases, the classification is achieved by finding the class c which maximizes the probability. Here we have chosen the class c such as $P(\mathbf{y}_1^T|\mathbf{x}_1^T)$ is maximum.

5 Experimental Results

In this section, we present baseline results obtained using HMMs and IOHMM on the proposed mono- and bi-manual database. In the case of IOHMM, we have conducted experiments using discrete conditional distributions. Thus we have performed a quantization step on the data. This quantization is explained later in this section. The open source machine learning library used for all experiments is Torch *http://www.torch.ch*.

5.1 Preprocessing the Database

Normalization: As a first step, a normalization has been performed on all gesture trajectories. We suppose that each gesture occurs in a cube centered on the torso and of vertex size the maximum spread given by the "Vinci" sequence. This cube is then normalized to reduce the vertex to one. Finally, the range of x, y and z coordinates varies between -0.5 and 0.5. The 3D coordinates of the head and torso are almost stationary. Thus, we keep the normalized 3D trajectories of both hands only. This leads to an input feature vector of size 6.

Feature Extraction: We also computed the difference between the coordinates of each hand for two consecutive frames. These features have been multiplied by 100 in order to have values with the same order of magnitude than x, y and z. The final feature vector is $[x_{left},y_{left},z_{left},x_{right},y_{right},z_{right},\Delta x_{left},\Delta y_{left},\Delta z_{left},\Delta x_{right},\Delta y_{right},\Delta z_{right}]$ of size 12.

Quantization: The second step is the quantization of the data to efficiently use discrete IOHMM. The output sequence still encodes the gesture class: $y_t = \{0,\ldots,15\}, \forall t$. In order to model more closely the class distribution of the data, we apply a K-means algorithm [13] class per class on the input features. For each gesture class, a K-means model with 75 clusters has been trained using the training set. The 16 resulting K-means models have been merged into a single one (1200 clusters). Finally, each frame of each sequence is quantized into one discrete value $\in [1\ldots1200]$ (which is the index of the nearest cluster).

5.2 Parameter Tuning

In our experiments, we have used a left/right topology for both the IOHMM and the 16 HMMs. In order to find the optimal hyper-parameters (number of states of the discrete IOHMM, number of states and number of Gaussians per state for the HMMs), the following experimental protocol has been used. For experimental purposes, the database has been split into three subsets: the training set T, the validation set V and the test set Te. T and V contain 5 subjects each. Te contains 10 subjects. For each subject, all recordings from all shots have been used. Table 2 provides the minimum/average and maximum number of frames per sequence for each subset of the database. Different possibilities for the number of states and for the number of Gaussians have been tried on T. The selection of the best parameters has been done on V. Finally, a model has been trained on both T and V and tested on Te. We obtained the best results with 5 states for the IOHMM, and 15 states and 1 Gaussian per state for each HMM.

	Training set	Validation set	Test set
minimum number of frames	12	6	10
average number of frames	25	24	28
maximum number of frames	64	71	89

Table 2: Minimum, average and maximum number of frames for the different subsets

5.3 Results

Figure 5 provides comparative results between discrete IOHMM and baseline continuous HMMs. HMM and IOHMM achieve respectively 64% and 74% average classification rate. From the results, we observe that bi-manual gestures are very well classified. Few mistakes happen between “swim” and “clap” gestures.

If we now have a look to the mono-manual gestures, we notice that first, there is a misclassification between the “stop”, “no/wipe”, “raise” and “hello/wave” gestures. If we refer to table 1, the only differences between these four gestures are the hand level and the oscillatory movement of the hand from the left to the right. Thus, HMMs have problems to model the oscillatory movement of the “no/wipe” and “hello” gestures. On the contrary, IOHMM has no real problem to model these oscillations (average recognition rate: 85%). With the HMMs, the “stop” and “no” gestures, such as the “raise” and “hello” gestures are misclassified one to the other. But still the non-oscillatory movements are misclassified in both cases, with the IOHMM and with the HMMs.

Let us consider the positioning gesture category (“left”, “right”, “up”, “down”, “front” and “back” gestures). The block around the diagonal of the matrix shows first that HMMs and IOHMM differentiate quite accurately this category of gestures from the others. It shows also that it has difficulties to provide the correct class within this category, even if IOHMM give better results than HMMs. Only the “left” and “right” gestures are well classified. For the others, it seems that the discriminant aspect of these gestures which is the dynamic of the hand (Table 1) is not sufficient for a good classification.

Finally, if we consider pointing gestures, HMMs and IOHMM differentiate also quite accurately this category of gestures from the others. But IOHMM give better results than HMMs as the recognition rate is around 70% for the 3 gestures, and only 60% for the HMMs. Concerning the “point left” gesture, HMMs misclassified it with the “point front” and “point right” gestures and IOHMM misclassified it only with the “point front”gesture. It shows that the location of the hand at the end of the pointing gesture is not precise enough to give a discriminant information to the IOHMM, and to the HMMs.

6 Conclusions

In this paper, we addressed the problem of the recognition of isolated complex mono- and bi-manual hand gestures. Hand gestures were represented by the 3D trajectories of blobs obtained by tracking colored body parts.

We provide recognition results obtained on a complex database of 16 mono- and bi-manual gestures by two sequence processing algorithms, namely Input Output Hidden Markov Model (IOHMM) and Hidden Markov Model (HMM), implemented within the framework of an open source machine learning library. The obtained results are encouraging. Bi-manual gestures are very well classified, and mono-manual gestures are fairly classified. We conclude that IOHMM performs the best on this database. We will perform complementary experiments using continuous IOHMM to verify if this conclusion is still valid.

Acknowledgments

This research has been carried out in the framework of the GHOST project, funded by France Telecom R&D (project number 021B276). This work was also funded by the Swiss National Science Foundation through the National Center of Competence in Research (NCCR) on ”Interactive Multimodal Information Management (IM2)”. The authors wish to thank J. Guerin and B. Rolland for recording and annotating the gesture database.

References

[1] L. R. Rabiner, “A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition”, *Proceedings of the IEEE*, Vol. 77, Nb. 2, 1989.

[2] T. E. Starner and A. Pentland, “Visual Recognition of American Sign Language Using Hidden Markov Models”, *Int. Workshop on Automatic Face- and Gesture-Recognition*, 1995.

[3] O. Bernier and D. Collobert, “Head and Hands 3D Tracking in Real Time by the EM algorithm” *Proceeding of the IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems (RATFG-RTS'01)*, 2001.

[4] S. Marcel and O. Bernier and J.E. Viallet and D. Collobert, ”Hand Gesture Recognition using Input-Output Hidden Markov Models”, *Proc. of the FG'2000 Conference on Automatic Face and Gesture Recognition”*, 2000.

[5] C.R. Wren and A. Azarbayejani and T. Darrell and A. Pentland, ”Pfinder: Real-Time Tracking of the Human Body”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, pp. 780-785, 1997.

[6] T. Darrell and A. Pentland, “Space-time gestures”, *Proc. of the Conference on Computer Vision and Pattern Recognition*, pp. 335-340, 1993.

[7] J. Davis and M. Shah, ”Recognizing Hand Gestures”, *Proc. of European Conference on Computer Vision* Vol. 1, pp. 331-340, 1994.

[8] A.P.Dempster and N.M. Laird and D.B. Rubin, “Maximum-likelihood from incomplete data via the EM algorithm”, *Journal of Royal Statistical Society*, Vol. 39, pp. 1-38, 1977.

[9] P. Hong and M. Turk and T.S. Huang, “Gesture Modeling and Recognition Using Finite State Machines”, *Proc. of the fourth International Conference on Automatic Face and Gesture Recognition*, 2000.

[10] V.I. Pavlovic and R. Sharma and T.S. Huang, “Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review” *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 19, pp. 677-695, 1997

[11] Yoshua Bengio and Paolo Frasconi, “An Input Output HMM Architecture” *Advances in Neural Information Processing Systems* Vol. 7, pp 427–434, 1995

[12] D.E. Rumelhart, G.E. Hinton and R.J. Williams “Learning internal representations by back-propagating errors” *Nature* Vol. 323, pp 533–536, 1986

[13] J.A. Hartigan and M.A. Wong “A K-Means Clustering Algorithm” *Journal of Applied Statistics*, Vol. 28, pp 100–108, 1979

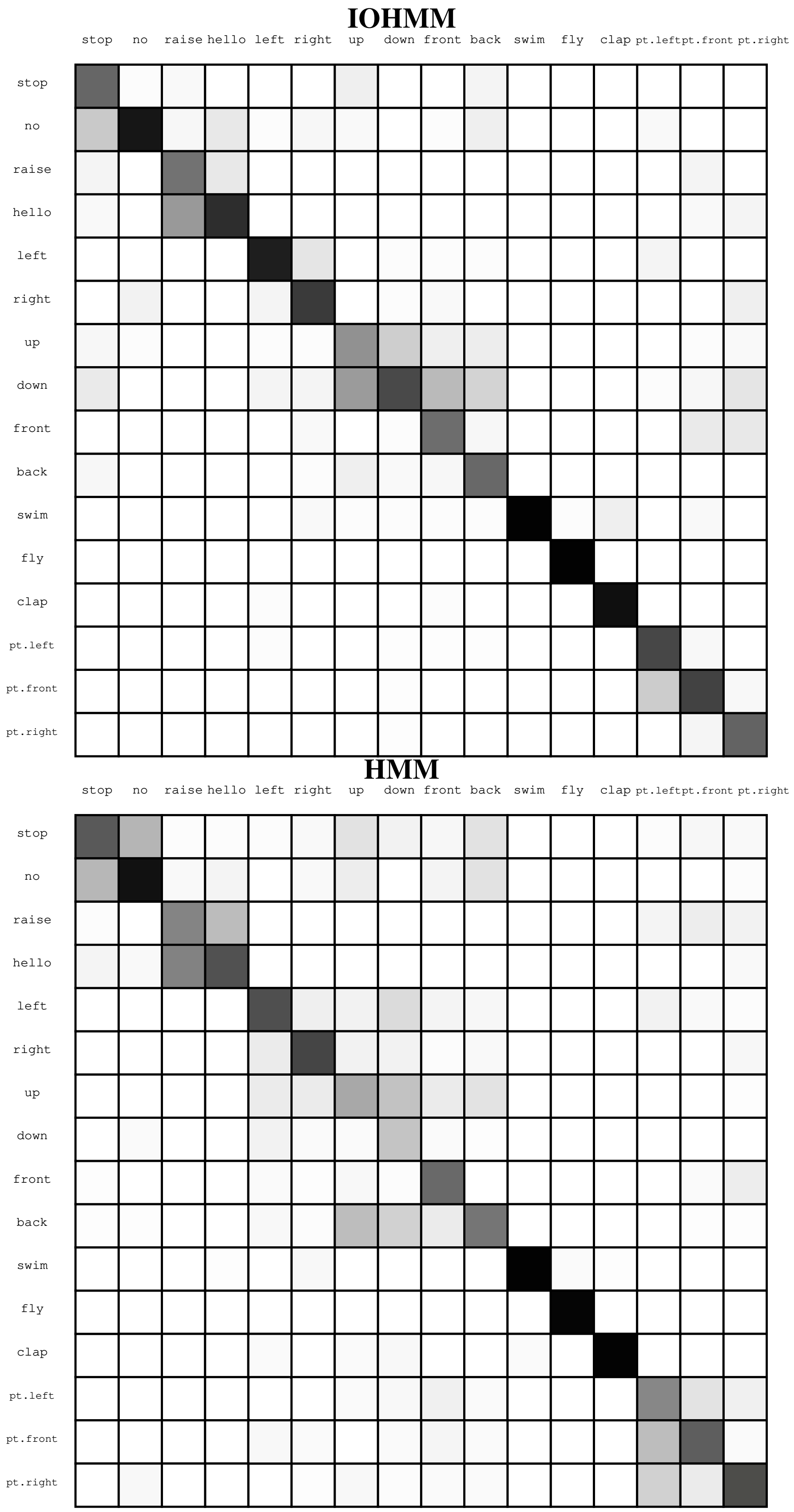

Figure 5: Confusion matrix for IOHMM and HMM on the test set (rows: desired, columns: obtained). Black squares correspond to the well-classified gestures.