Dynamic Hand Gesture Recognition Using the Skeleton of the Hand

4 authors, including:

Bogdan Ionescu
Polytechnic University of Bucharest
150 PUBLICATIONS **881** CITATIONS

SEE PROFILE

Didier Coquin
Université Savoie Mont Blanc
89 PUBLICATIONS **529** CITATIONS

SEE PROFILE

Patrick Lambert
Université Savoie Mont Blanc
120 PUBLICATIONS **1,046** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project   Deep learning based Object Recognition and Semantic Segmentation View project

Project   MediaEval Benchmarking Series in "Multimedia Recommendation" View project

Dynamic Hand Gesture Recognition Using the Skeleton of the Hand

Bogdan Ionescu

University "Politehnica", LEU, Complex-Room B135A, 1–3 Liliu Maniu Boulevard, Bucharest, Romania
Email: bionescu@alpha.imag.pub.ro

Didier Coquin

Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance (LISTIC),
Université de Savoie, 74016 Annecy Cedex, France
Email: didier.coquin@univ-savoie.fr

Patrick Lambert

Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance (LISTIC),
Université de Savoie, 74016 Annecy Cedex, France
Email: patrick.lambert@univ-savoie.fr

Vasile Buzuloiu

University "Politehnica", LEU, Complex-Room B135A, 1–3 Liliu Maniu Boulevard, Bucharest, Romania
Email: buzuloiu@alpha.imag.pub.ro

This paper discusses the use of the computer vision in the interpretation of human gestures. Hand gestures would be an intuitive and ideal way of exchanging information with other people in a virtual space, guiding some robots to perform certain tasks in a hostile environment, or interacting with computers. Hand gestures can be divided into two main categories: static gestures and dynamic gestures. In this paper, a novel dynamic hand gesture recognition technique is proposed. It is based on the 2D skeleton representation of the hand. For each gesture, the hand skeletons of each posture are superposed providing a single image which is the dynamic signature of the gesture. The recognition is performed by comparing this signature with the ones from a gesture alphabet, using Baddeley's distance as a measure of dissimilarities between model parameters.

Keywords and phrases: hand gesture recognition, skeleton, orientation histogram, Baddeley distance.

1. INTRODUCTION

The goal of the dynamic hand gesture recognition technique proposed in this paper is to offer a method of guiding robots from a distance using hand gestures. It is obvious that we can use the hand gesture recognition techniques to control a computer or a hi-fi device, as well. In order to command a robot, we have to "send" some information to it, somehow. This information will be "encapsulated" in the hand gesture. Our task is to translate the information into a command which can be sent to the robot.

To perform the hand gesture recognition, we have to create a model of the gestures and to use it to do the recognition task.

1.1. Hand gesture modeling

From the initial intention to the final performance, a gesture follows a characteristic motion pattern in space and time [1]. Kendon distinguishes three motion phases that a single gesture is made of: preparation, stroke, and retraction [2]. The properties of this pattern are universal and permanent, and can, therefore, be used to describe any particular gesture. The following question arises: "Is every hand movement a gesture?" The answer is given by Quek who developed a set of rules for the distinction of real gestures from other hand movements [3].

(1) Gestures are contained in movements that start with a slow initial move from the rest position, continue with a phase with substantially increased speed (the stroke), and end by returning to the rest position.

(2) The hand assumes a particular configuration during the stroke.

(3) Slow motions between resting positions are not gestures.

(4) Hand gestures should be constrained within a certain volume space.

(5) Static hand gestures require a finite period of time to be recognized.

(6) Repetitive movements can be gestures.

From the rules above, one can infer that hand gestures could be classified as *static hand gestures* (characterized by the hand posture which is determined by a particular finger-thumb-palm configuration) and *dynamic hand gestures* (characterized by the initial and final stroke hand configuration and the general stroke motion) [1].

1.2. Hand gesture analysis

There are three main approaches in the hand gesture analysis: *glove-based analysis*, *vision-based analysis*, and *analysis of drawing gestures* [1].

Vision-based analysis of hand gestures is probably the most natural way of constructing a human-computer gesture interface. Yet it is also the most difficult one to implement in a satisfactory manner because of the limitations in machine vision today. The vision-based analysis is performed by using one or more cameras. The visual information about a person in some visual environment is acquired and the necessary gesture is extracted. However, there are some difficult problems: the segmentation of the moving hand from a sometimes complex environment, the analysis of hand motions, the tracking of hand positions relative to the environment, and finally the recognition of hand postures. To facilitate these operations, some use markers, marked gloves, or restrictive setups: uniform background, very limited gesture vocabulary, or just a simple posture analysis.

The work related in this paper is a part of the vision-based analysis techniques. The most common visual-based hand gesture recognition techniques are *3D hand model analysis*, *analysis using markers and marked gloves*, and *image-property-based analysis* [1]. The technique proposed here uses the properties of the images in order to perform the gesture recognition.

1.3. A classification of hand gesture recognition techniques

There is a growing literature on hand gesture recognition. An overview has been proposed by Pavlovic et al. in [1]. The process of hand gesture recognition can be divided into two main tasks as follows. *Feature extraction* where low-level information from the raw data is analyzed to produce higher-level semantic information. Once the features are collected, the second phase of hand gesture recognition is the *classification* when the data collected will be used to detect a hand gesture.

The features are built on distance, velocity and acceleration information, energy measurements, or angle information and are based on techniques such as active shape models, principal component analysis, linear fingertip models, and spatiotemporal vector analysis [4, 5, 6].

As classification techniques, we can mention the template-based approach, statistical methods (hidden Markov model [7, 8]), and miscellaneous algorithms (neural networks [7, 9], causal analysis). In general, a template-based approach creates data records for each posture and gesture in the set and uses them to classify new postures and gestures. There are several types of template-based classification methods: *conventional template matching*, *instance-based learning* [10], *the linguistic approach* [11], and *appearance-based motion analysis* [12].

Conventional template matching, which is generally the simplest way, is performed in two steps. In the first part, the templates are created by collecting data values for each posture in the posture set; generally each posture is performed a number of times and the average of the raw data for each sensor (camera, glove) is taken and stored as the template. The second part consists in comparing the current sensor readings with the given set of templates to find the posture template most closely matching the current data record. The dynamic gesture recognition technique proposed in this paper uses as classification method the conventional template matching approach.

2. GENERAL PRESENTATION OF THE PROPOSED APPROACH

In this paper, a new dynamic hand gesture recognition technique is proposed. Our algorithm is a vision-based algorithm. Images of hand gestures are captured with a single digital video camera and then processed by a computer. The images are 2D grey-level images. To simplify the hand segmentation, a restrictive context is used:

  (i) the hand gestures are performed against an approximately uniform background,
 (ii) the distance between the camera and the hand is nearly constant and the gesture is performed in an *a priori* known region of the space,
(iii) the gesture alphabet is *a priori* known.

This does not affect the generality of the recognition process. In order to preserve the generality of the method, no markers or special gloves are used.

Because of the efficiency in terms of computation time, the simple feature extraction techniques are used for the extraction of the gesture features. Dealing directly with pixel intensities is less time-consuming than training some neural networks or extracting the statistics from the gesture. The features extracted in this manner represent the physical and spatial properties of the gesture. Based on that, computing a certain distance between these features is the fastest way to compare them.

Also the classification step requires a very *minimal* training stage consisting in a set of examples for each gesture. As classification method, the conventional template matching will be used. All these considerations lead to fast and efficient hand recognition techniques.

The hand gestures can be divided into static hand gestures, which are represented by a single image of the hand, and dynamic hand gestures which are represented by a sequence of images, each one corresponding to a hand posture
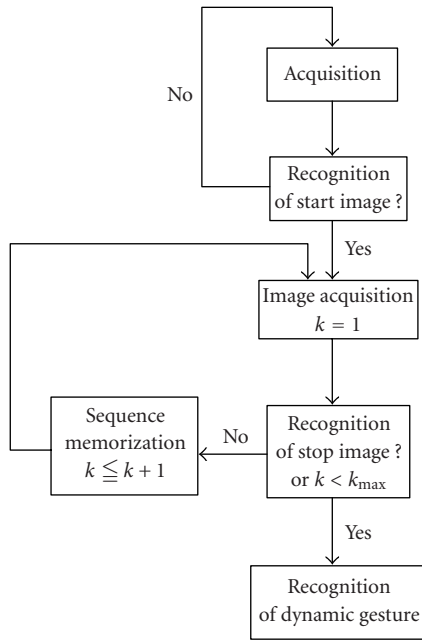
FIGURE 1: Gesture recognition process.

FIGURE 2: The gesture recognition process diagram.

within the gesture (hand movement). The recognition of a gesture is performed using the fusion of a static and a dynamic recognition technique. The static recognition is used to detect the start and end positions of the hand during the gesture (denoted as *start* and *stop* images or start and end signatures, in the following). The dynamic recognition is only performed between the *start* and *stop* images.

The gesture frames are sequentially given by the digital camera and processed. When a beginning gesture signature is recognized, the dynamic signature is computed as the frames arrive. If the end signature is recognized, the recognition process is ended, else the recognition process will stop after k_{max} acquisitions. Typically $k_{max} = 50$, which corresponds to the maximum duration of the gesture. Static gestures can be recognized if the *start* and the *end* static signatures are the same and the dynamic signature does not present any hand skeleton movements. In order to prove the efficiency of the proposed dynamic gesture recognition algorithm, only dynamic gestures were used in our tests. The diagram of the dynamic gesture recognition system is presented in **Figure 1**.

As such, the general process of the proposed method is composed of four main parts.

(i) *A preprocessing step* to focus on the gesture.
(ii) *A feature extraction step* where two techniques are used.
   (1) The first one, dedicated to static recognition, uses *histograms of local orientations* in the gesture image (by computing the local gradient on the image). It is based on an algorithm proposed by Freeman and Roth [13]. The orientation histogram will act as the feature set of the gesture.
   (2) The second one, used for dynamic gesture recognition, extracts *"the dynamic signature"* and is
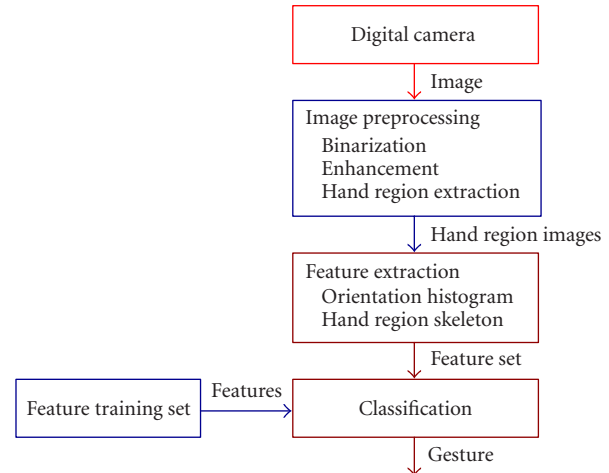
defined using a new original method which gives a compact and efficient representation of the gesture. The dynamic signature is a binary image which represents the superposition of all the skeletons of the hand region for all images (hand postures) within the gesture image sequence. It reflects the motion information of the gesture and also the hand spatial configuration during the gesture motion. Using the proposed dynamic signature, the gesture can be segmented into its constitutive steps such as the preparation and the stroke [3]. Each gesture will be represented by a particular dynamic signature.

(iii) *A training step.* In order to perform the gesture classification, the gesture alphabet has to be known. A training stage is required. Using a set of samples for each gesture, a training set of data is generated for both the static and the dynamic recognition. Using different representations in space of the same gesture gives invariance to small translations and rotations. The influence of the size of this set will be studied in **Section 5.3**.

(iv) *A classification step.* The unknown gesture features will be compared with the features stored in the training data. The best match will lead to the recognition of the unknown gesture.

The gesture recognition process diagram is illustrated in **Figure 2**. The hand region obtained after the preprocessing stage will be used as the primary input data for the feature extraction step of the gesture recognition algorithm.

3. IMAGE PREPROCESSING FOR FEATURE EXTRACTION

In order to extract the features needed for the classification step of the gesture recognition process, we have to preprocess the images given by the digital camera. The main goal is to isolate and enhance the hand region. The preprocessing chain consists of the following operations: *binary image*
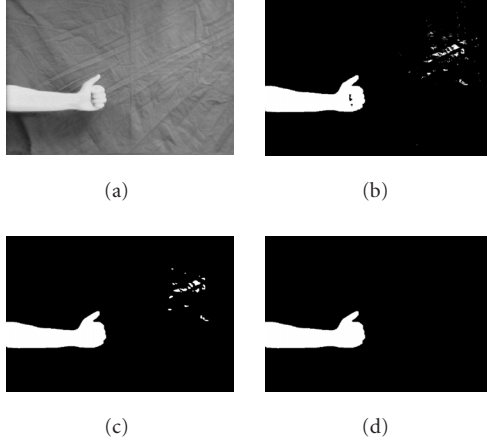
(a)

(b)

(c)

(d)

FIGURE 3: Hand region extraction: (a) original grey-level image, (b) binary image after histogram segmentation, (c) binary image after morphological enhancement and hole filling, and (d) hand region.

computation, *binary image enhancement,* and *hand region extraction*. The background is considered almost uniform and darker than the hand region. This restriction is used to focus our attention on the performance of dynamic hand gesture recognition. The enhancement operation is done by applying morphological operations (opening, closing, and hole filling). The detection of the biggest convex region of white pixels in the binary image can also be used as a hand presence detector. The results can be seen in Figure 3.

4.   STATIC HAND GESTURE RECOGNITION

A static hand gesture is represented by a single grey-level image consisting of a particular finger-palm-thumb hand configuration. The static hand gesture recognition is performed by using the *local orientation histograms*.
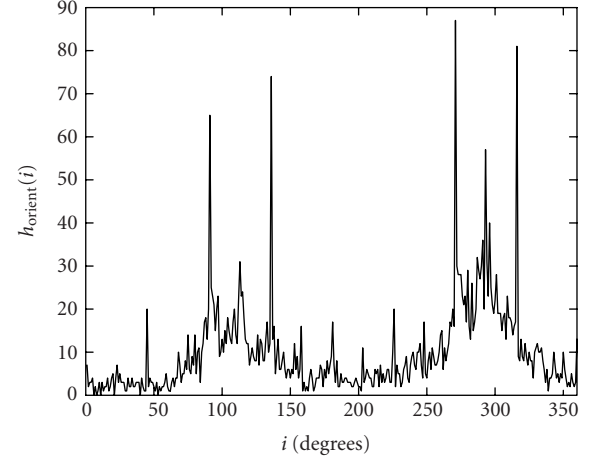
4.1.   *Local orientation histograms*

Computing the local orientation histogram is a simple and fast technique which can easily run in real time on a workstation. It is also robust in the case of illumination changes in the input images. The technique used is based on a method proposed by Freeman and Roth [13].

The orientation histograms are used as features for the gestures. For each gesture image, the binary image will be computed and preprocessed as shown in Section 3 in order to obtain the hand region. Based on that, the original pixel intensities are retrieved from the initial image. The result is a grey-level image which contains the hand posture against a black background (the original background is erased).

The local orientation histograms are based on local pixel orientations which are described by directions of the local gradients. The gradient amplitudes and orientations are obtained using the Sobel operator only for the pixels within the hand region.

In order to take into account only the pixels with a significant local variation, the pixels with a gradient amplitude

(a)
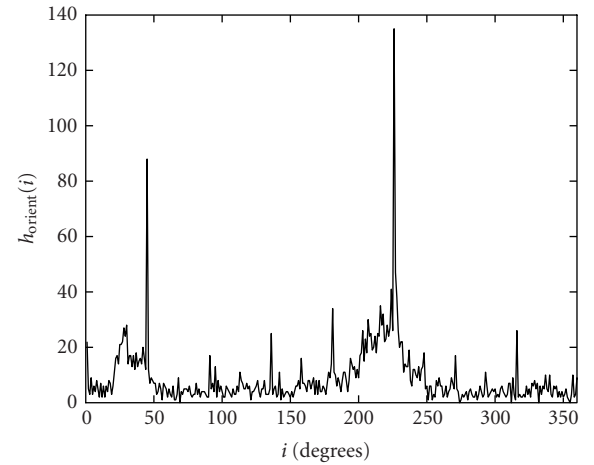
(b)

FIGURE 4: Orientation histograms of sequence "GET" ((a) *start* of the gesture; (b) *end* of the gesture).

smaller than the mean gradient amplitude multiplied by a coefficient (the optimal experimental value is 1.5) are eliminated. The orientation histogram is computed as a statistic of gradient orientations for the remaining pixels.

$$h_{\text{orient}}(i) = \sum_{m=1}^{M} \sum_{n=1}^{N} \delta\big(f_{\text{grad orient}}(m,n) - i\big), \qquad (1)$$

where i is the angle in degrees (from 0 to 359 degrees), $f_{\text{grad orient}}$ is the matrix of the gradient orientations, $M \times N$ is the image dimension and $\delta(x)$ is set to 1 if $x = 0$ and set to 0 otherwise.

The idea of using the orientation histogram is that each gesture has a particular orientation histogram, and it can be used to distinguish between different *start* or *stop* images of the dynamic gestures (start and end signatures). The orientation histograms are also fast to compute. Figures 4a and 4b show orientation histograms for the start and the end images of sequence "GET."
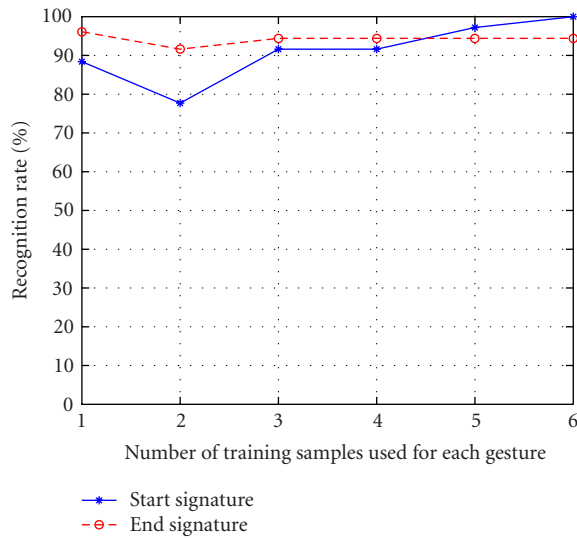
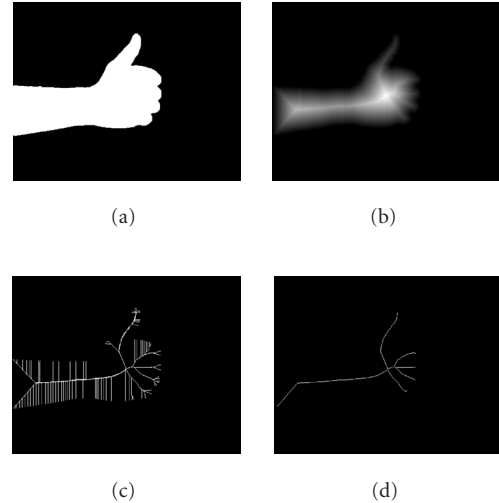FIGURE 5: Recognition rates for static signatures.

FIGURE 6: Skeleton extraction: (a) hand region (binary image), (b) chamfer distance image (white corresponds to the greatest distance), (c) the skeleton obtained after connecting the centers of maximal discs, and (d) the skeleton obtained after spurious hole filling, pruning, and beautifying the previous skeleton.

4.2. Static gesture recognition

The Euclidean distance is used as a measure of distance between different orientation histograms.

The training stage of the static hand gesture recognition algorithm consists in recording the mean orientation histogram in order to get a set of samples for each gesture within the gesture alphabet. The new gesture orientation histogram is computed and compared (using the Euclidean distance) with all the histograms from the training set. The minimum Euclidean distance gives the recognized static gesture.

The method was tested on a small hand gesture alphabet, containing ten different gestures in order to determine its performance. Thirty-six sequences were used for the tests. The local orientation histogram training set was computed using the mean of three samples for each *start* and each *end* image of the gestures in order to increase the robustness of the recognition process.

Figure 5 shows the recognition rate for the static signatures (start and end signatures) versus the number of training sequences. The recognition rate remains between 77% and 100%. However, these errors consist always in some confusions between different start and stop image signatures. So, in many cases, the dynamic gesture will be correctly delimited. A higher number of training sequences lead to a higher recognition rate.

5. DYNAMIC HAND GESTURE RECOGNITION

A dynamic gesture consists of a particular movement of the hand which encapsulates a particular piece of information (the gesture purpose is to transmit meaningful information). The dynamic gestures are represented by a sequence of grey-level images (typically between 40 or 50 images in our case).

5.1. Dynamic signature

The dynamic signature is the fusion of all the hand region skeletons for all the frames between the start and end gestures.

A skeleton is a compact representation of an object (hand region in our case). The desired properties of the skeleton are preserving the topology of the object (the same number of connected components and the same number of holes), robust against translation, rotation, and scaling, and thin (no more than 2 pixels in width) [14]. The extraction of the hand region skeleton is performed using a distance transformation-based method: the chamfer distance transformation [15]. The distance transformation is computed to extract the median axis of the hand region (local maxima). Then the resulting set is connected in order to preserve the object topology. For more information on computing skeleton using chamfer distance, see [16]. Figure 6 shows the different steps to extract a hand region skeleton from a binary image.

The dynamic signature is the binary image composed by the superposition of all the skeletons calculated within the dynamic sequence. It is used to capture the motion of the gesture. Figure 7 shows examples of dynamic signatures for "GET" and "SLOW" gestures.

We can see that on a dynamic signature the three stages of a gesture [2] are very clearly illustrated: the preparation (at the beginning of the gesture a lot of skeletons are overlaid forming a thicker skeleton), the stroke (the skeletons are apart because the hand movement is fast), and the resting position (the skeletons are also overlaid).

This technique is very interesting because it summarizes the entire gesture in a single binary image easy to analyze. However, the gesture chronology is lost. We cannot determine which position is the starting position or the end one. This drawback could be solved using the start and the end static signatures introduced for the static gesture recognition in Section 4.
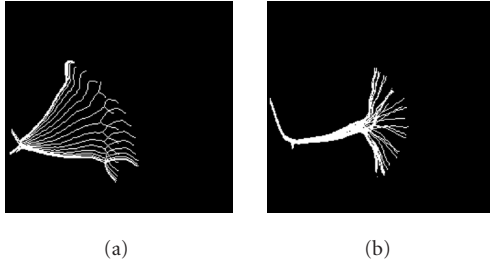
(a)   (b)

FIGURE 7: Dynamic signatures for (a) "GET" and (b) "SLOW" sequences.

The used technique requires that operator positions be nearly always the same in the field of the camera. Dynamic gesture recognition depends on the position of the gestrure in the image. To improve the robustness of our method against translation, we have determined the extreme-left point and the extreme-low point of the superposed skeletons, and a normalization is carried out so that the superposed skeletons images have always the extreme-left point on the same abscissa and the extreme-low point on the same ordinate.

If this technique limits the sensitivity to translation, the proposed method remains sensitive to rotation. A certain invariance to rotation is obtained using in the training stage different representations in space of the same gesture (each gesture is performed using different angles and positions). However, the gesture alphabet supposes that all the gestures are performed with an almost similar angle position of the arm.

5.2. Recognition process

The diagram of the dynamic gesture recognition system is presented in Figure 2. For each gesture, the static (orientation histograms) and the dynamic (binary image of the superposition of the hand postures skeletons) signatures are computed.

The recognition process is performed by comparing the obtained signatures with some references obtained from the training stage. The static signature recognition is dealt with as presented in Section 4. The comparison is performed on binary images and used the Baddeley's distance which is less sensitive to object translation and rotation than the Euclidean distance. This distance [17, 18] is defined as follows.

If S is a reference system ($S \subset Z^2$) on which our images are digitized, a binary image can be seen as a function f_I : $S \rightarrow \{0, 1\}$. If the two images to be compared are I_1 and I_2, Baddeley's distance is defined as

$$d_B(I_1, I_2) = \left[\frac{1}{N \times M} \sum_{p \in S} |d_{I_1}(p) - d_{I_2}(p)|^q \right]^{1/q}, \quad (2)$$

where $N \times M$ is the total number of pixels from S, $d_{I_j}(p)$ is the distance from point p to the binary object contained in the image I_j with $j = 1, 2$, and q is an exponent controling the weight of the distance magnitude. Here q is set to 2.

To compare two dynamic signatures, we have computed Baddeley's distance within the smallest frame containing the skeletons. We can notice that Baddeley's distance is an average measure. Hence, it is less sensitive than the Hausdorff distance to small and localized distortions.

In order to perform the classification step of the gesture recognition process, a training set of features is computed. For each gesture from the gesture alphabet, a set of sample signatures is used. Instead of using the mean binary image of the dynamic signatures as a training feature for each gesture (which is irrelevant in the case of binary images), m sample signatures are used.

In conclusion, the training set consists of m training sequences (dynamic signatures) for each hand gesture within the gesture alphabet (see Figure 8). Using n gestures in the gesture alphabet and having m sets of example signatures for each one, there are $n \times m$ sample signatures for the training sequence.

5.3. Performance of the gesture recognition system

The hand gesture recognition system was tested on an alphabet of 10 gestures representing robot commands (gestures: whap-whap, behind, get, throw, left, right, slow, stop, turn, and go). The gesture alphabet is shown in Figure 8.

Using a set of samples for each gesture varying from one sample to six samples (for each gesture from the gesture alphabet, there is a set of sample signatures in the training set), the recognition system performed very well and achieved a high recognition rate (see below).

In Figure 9, the recognition rates for each dynamic signature are shown. Experiments have been performed with 40 gestures (10 different gestures and 4 tests for each gesture). A 100% recognition rate was achieved using 2 to 6 training samples. Even using only one training sequence, the recognition rate was greater than 95%. Because the dynamic signature reflects the motion of the gesture, it is likely to be different for different gestures.

5.4. Computation time

Finally, the proposed dynamic gesture recognition algorithm consists in the computation of two types of signatures: the static signatures (for the beginning and the ending of the gesture) and the dynamic signature.

The static signatures are the local orientation histograms for the *start* and the *end* images of the gesture. The computation of an orientation histogram is very fast and only requires a conventional workstation in order to be performed in real time. The image is only scanned once in order to compute the gradient orientation of the pixels. The static gesture recognition using the orientation histograms can be performed in real time [13].

In contrast, the dynamic signature requires the computation of the hand region skeleton, which is a computational time-consuming algorithm and cannot be performed in real time on a conventional workstation. The skeleton computation requires several analyses of the image pixels having a polynomial computational complexity.
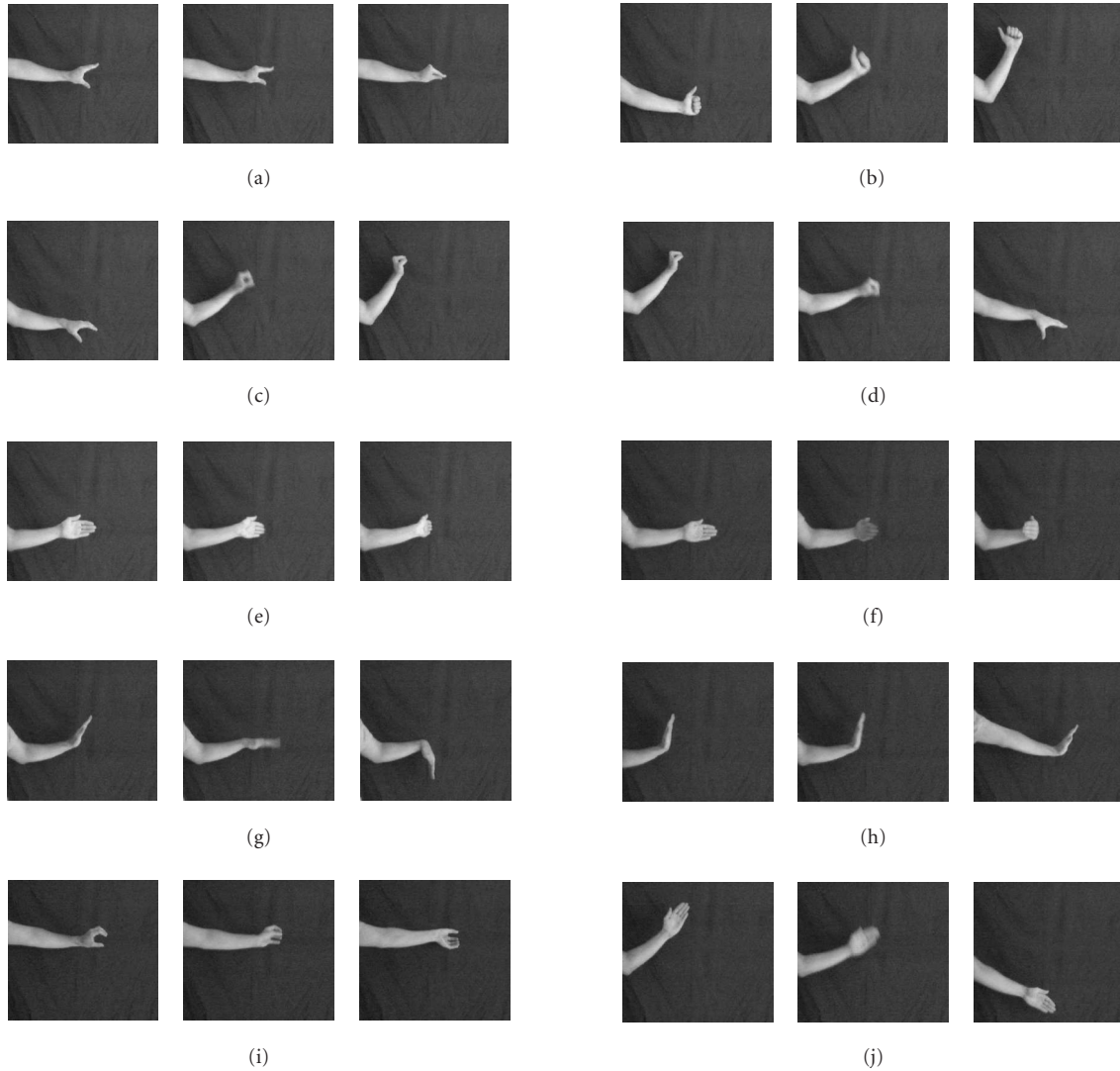
Figure 8: Dynamic alphabet of ten hand gestures: (a) whap-whap, (b) behind, (c) get, (d) throw, (e) left, (f) right, (g) slow, (h) stop, (i) turn, and (j) go.

Actually, the proposed gesture recognition system is working on a 2 GHz Pentium 4 CPU, equipped with a video acquisition and processing system (Matrox card). The overall system allows to reach a 12 images per second processing [19]. Some other experiments have been successfully realized to control a robot in real time by using five gestures (stop, forward, backward, left, and right).

6. CONCLUSION

In this paper, we have proposed a gesture recognition method using both static signatures and an original dynamic signature.

The static signature, based on the method proposed by Freeman and Roth (see Section 4.1) [13], uses the local orientation histograms in order to classify the hand gestures.

It is a fast method due to the easiness of the orientation histogram computation, which works in real time on a workstation and is also relatively robust to illumination changes. This approach would not be appropriate to make inferences about occluded data. A little training is required, the mean orientation histograms are computed for a training set of a minimum of three samples for each gesture within the gesture alphabet. The current gesture orientation histogram is compared with the training set.

The proposed dynamic gesture recognition algorithm is an original one. Each gesture is represented by a sequence of images. The dynamic signature used for the classification is the superposition of all hand region skeletons for each image within the sequence. It reflects the motion and the spatial location of the gesture. For the gesture classification, only a little training is required. For each gesture from the gesture
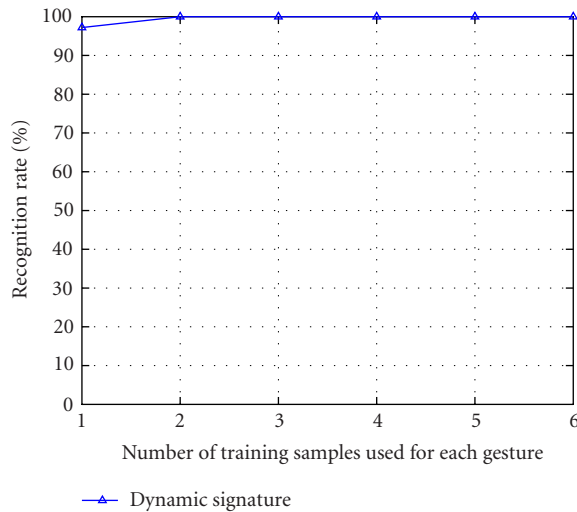
FIGURE 9: Recognition rates for dynamic signatures.

alphabet, a set of sample signatures is computed. The algorithm performed very well on a ten-gesture alphabet leading to a recognition rate of 100% in simplified conditions (uniform background, limited alphabet) in order to facilitate the hand region extraction. A general hand region segmentation could be done by detecting the skin color in the hand gesture images (skin detection algorithms [20])

REFERENCES

[1] V. I. Pavlovic, R. Sharma, and T. S. Huang, "Visual interpretation of hand gestures for human-computer interaction: a review," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, no. 7, pp. 677–695, 1997.

[2] A. Kendon, "Current issues in the study of gesture," in *The Biological Foundations of Gestures: Motor and Semiotic Aspects*, J.-L. Nespoulous, P. Perron, and A. R. Lecours, Eds., pp. 23–47, Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 1986.

[3] F. K. H. Quek, "Toward a vision-based hand gesture interface," in *Proc. Virtual Reality Software and Technology Conference (VRST '94)*, pp. 17–29, Singapore, Republic of Singapore, August 1994.

[4] H.-S. Yoon, J. Soh, Y. J. Bae, and H. S. Yang, "Hand gesture recognition using combined features of location, angle and velocity," *Pattern Recognition*, vol. 34, no. 7, pp. 1491–1501, 2001.

[5] S. Nayaga, S. Seki, and R. Oka, "A theoretical consideration of pattern space trajectory for gesture spotting recognition," in *Proc. 2nd IEEE International Conference on Automatic Face and Gesture Recognition (FGR '96)*, pp. 72–77, Killington, Vt, USA, October 1996.

[6] B. Raytchev, O. Hasegawa, and N. Otsu, "User-independent online gesture recognition by relative motion extraction," *Pattern Recognition Letters*, vol. 21, no. 1, pp. 69–82, 2000.

[7] C. W. Ng and S. Ranganath, "Real-time gesture recognition system and application," *Image and Vision Computing*, vol. 20, no. 13-14, pp. 993–1007, 2002.

[8] F.-S. Chen, C.-M. Fu, and C.-L. Huang, "Hand gesture recognition using a real time tracking method and hidden Markov models," *Image and Vision Computing*, vol. 21, no. 8, pp. 745–758, 2003.

[9] R. Kjeldsen and J. Kender, "Toward the use of gesture in traditional user interfaces," in *Proc. IEEE International Conference on Automatic Face and Gesture Recognition (FGR '96)*, pp. 151–156, Killington, Vt, USA, October 1996.

[10] F. Quek, "Unencumbered gestural interaction," *IEEE Multimedia*, vol. 4 , no. 3, pp. 36–47, 1996.

[11] H. Chris, I. Sexton, and M. Mullan, "A linguistic approach to the recognition of hand gestures," in *Proc. Designing Future Interaction Conference*, University of Warwick, Coventry, UK, April 1994.

[12] J. Davis, J. William, and M. Shah, "Gesture recognition," Tech. Rep. CS-TR-93-11, Department of Computer Science, University of Central Florida, Orlando, Fla, USA, 1993.

[13] W. T. Freeman and M. Roth, "Orientation histograms for hand gesture recognition," Tech. Rep. TR-94-03a, Mitsubishi Electric Research Laboratories, Cambridge Research centre, Cambridge, Mass, USA, 1995.

[14] C. Arcelli and G. Sanniti di Baja, "Euclidean skeleton via centre-of-maximal disc extraction," *Image and Vision Computing*, vol. 11, no. 3, pp. 163–173, 1993.

[15] G. Borgefors, "Distance transformations in digital images," *Computer Vision, Graphics and Image Processing*, vol. 34, no. 3, pp. 344–371, 1986.

[16] Y. Chehadeh, D. Coquin, and H. Bolon, "A skeletonization algorithm using chamfer distance transformation adapted to rectangular grids," in *Proc. 13th IEEE International Conference on Pattern Recognition (ICPR'96)*, vol. 2, pp. 131–135, Vienna, Austria, August 1996.

[17] A. J. Baddeley, "An error metric for binary images," in *Robust Computer Vision*, pp. 59–78, Wichmann, Karlsruhe, Germany, 1992.

[18] D. Coquin and Ph. Bolon, "Applications of Baddeley's distance to dissimilarity measurement between gray scale images," *Pattern Recognition Letters*, vol. 22, no. 14, pp. 1483–1502, 2001.

[19] B. Ionescu, D. Coquin, and P. Lambert, "Reconnaissance de gestes dynamiques de la main," in *Proc. 19eme Colloque sur le traitement du signal et des images (GRETSI '03)*, pp. 22–25, Paris, France, September 2003.

[20] L. Gu and D. Bone, "Skin colour region detection in MPEG video sequences," in *Proc. IEEE International Conference on Image Analysis and Processing (ICIAP '99)*, pp. 898–903, Venice, Italy, September 1999.

Bogdan Ionescu received an Engineer's degree in applied electronics (2002) and an M.S. degree in computing systems (2003) both from University "Politehnica" of Bucharest, Romania. Since 2003, he has been a Ph.D. student in image processing and informatics with Image Processing and Analysis Laboratory (LAPI), University "Politehnica" Bucharest and Laboratoire d'Informatique, Systémes, Traitement de l'Information et de la Connaissance (LISTIC), Université de Savoie, Annecy, France. His scientific interests cover electronics engineering, artificial intelligence, image processing and computer vision, software engineering, and computer science. Since 2001, he has attended the Spring International School "Multidimensional Signal Processing and Analysis: Methods, Algorithms, Technologies, Applications," which is organized yearly at the University "Politehnica" of Bucharest. In 2003 and 2004, he participated in the International Animated Film Market, Annecy, France, with the project "Semantic Analysis of the Animation Movies."
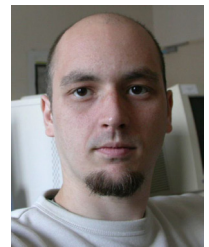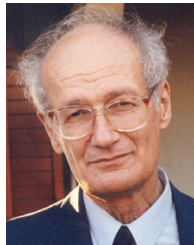
Didier Coquin received the Ph.D. degree in signal processing and telecommunication from University of Rennes I, France, in 1991. He is currently an Assistant Professor of telecommunication and network engineering at the Savoie University (Technological Academic Institute) and works in the Informatics, Systems, Information and Knowledge Processing Laboratory (LISTIC), Annecy, France. His research interests include image processing, pattern recognition, image and video indexation, data fusion, and fuzzy logic.

Patrick Lambert gratuated with a major in electrical engineering from the Grenoble National Polytechnic Institute (INPG), Grenoble, France, in 1978. He received the Ph.D. degree in signal processing in 1983. He is currently an Assistant Professor of electrical engineering at the Savoie University (Technological Academic Institute) and works in the Informatics, Systems, Information and Knowledge Processing Laboratory (LISTIC), Annecy, France. His research interests include image and video indexation, data fusion, and fuzzy logic.

Vasile Buzuloiu is a Professor with the Department of Applied Electronics and Information Engineering. He also heads the Image Processing and Analysis Laboratory, University "Politehnica" of Bucharest, Romania, and is a Research Associate with CERN Geneva, Switzerland. His scientific interests cover mathematical modeling, statistical decisions, encryption, digital signal processing, image processing and analysis systems, and image processing applications. He holds an M.S. degree in electronics from the University "Politehnica" of Bucharest (1959), an M.S. degree in mathematics from "Universitatea Bucuresti" (1971), and a Ph.D. degree in electronics from University "Politehnica" of Bucharest (1980). He received the "Traian Vuia" Award of the Romanian Academy (1985) for the first digital image analysis system developed in Romania. He also holds the title of Honorary Professor of the University "Transilvania" of Brasov, Romania (2001). Since 1994, he has held various positions of Invited Professor at French Universities (INSA de Lyon, ESIA-Université de Savoie, Université Saint Etienne, Université Poitiers, INP Toulouse, etc.). He is a Member of IEEE, SPIE, Color Group (Great Britain), and the Romanian Society for Applied Mathematics. Since 1995, he has been the Director of the Spring International School "Multi-dimensional Signal Processing and Analysis: Methods, Algorithms, Technologies, Applications," which is organized yearly at University "Politehnica" of Bucharest.

EUSIPCO 2011

19th European Signal Processing Conference
August 29- September 2, 2011, Barcelona (Spain)

Photograph © Turisme de Barcelona / J. Trullàs

Preliminary call for papers

The 2011 European Signal Processing Conference (EUSIPCO-2011) is the nineteenth in a series of conferences promoted by the European Association for Signal Processing (EURASIP, www.eurasip.org). This year edition will take place in Barcelona, capital city of Catalonia (Spain), and will be jointly organized by the Centre Tecnològic de Telecomunicacions de Catalunya (CTTC) and the Universitat Politècnica de Catalunya (UPC).

EUSIPCO-2011 will focus on key aspects of signal processing theory and applications as listed below. Acceptance of submissions will be based on quality, relevance and originality. Accepted papers will be published in the EUSIPCO proceedings and presented during the conference. Paper submissions, proposals for tutorials and proposals for special sessions are invited in, but not limited to, the following areas of interest.

Areas of Interest

- Audio and electro-acoustics.
- Design, implementation, and applications of signal processing systems.
- Multimedia signal processing and coding.
- Image and multidimensional signal processing.
- Signal detection and estimation.
- Sensor array and multi-channel signal processing.
- Sensor fusion in networked systems.
- Signal processing for communications.
- Medical imaging and image analysis.
- Non-stationary, non-linear and non-Gaussian signal processing.

Submissions

Procedures to submit a paper and proposals for special sessions and tutorials will be detailed at www.eusipco2011.org. Submitted papers must be camera-ready, no more than 5 pages long, and conforming to the standard specified on the EUSIPCO 2011 web site. First authors who are registered students can participate in the best student paper competition.

Important Deadlines:

Proposals for special sessions	15 Dec 2010
Proposals for tutorials	18 Feb 2011
Electronic submission of full papers	**21 Feb 2011**
Notification of acceptance	23 May 2011
Submission of camera-ready papers	6 Jun 2011

Webpage: www.eusipco2011.org

Organizing Committee

Honorary Chair
Miguel A. Lagunas (CTTC)

General Chair
Ana I. Pérez-Neira (UPC)

General Vice-Chair
Carles Antón-Haro (CTTC)

Technical Program Chair
Xavier Mestre (CTTC)

Technical Program Co-Chairs
Javier Hernando (UPC)
Montserrat Pardàs (UPC)

Plenary Talks
Ferran Marqués (UPC)
Yonina Eldar (Technion)

Special Sessions
Ignacio Santamaría (Unversidad de Cantabria)
Mats Bengtsson (KTH)

Finances
Montserrat Nájar (UPC)

Tutorials
Daniel P. Palomar
(Hong Kong UST)
Beatrice Pesquet-Popescu (ENST)

Publicity
Stephan Pfletschinger (CTTC)
Mònica Navarro (CTTC)

Publications
Antonio Pascual (UPC)
Carles Fernández (CTTC)

Industrial Liaison & Exhibits
Angeliki Alexiou
(University of Piraeus)
Albert Sitjà (CTTC)

International Liaison
Ju Liu (Shandong University-China)
Jinhong Yuan (UNSW-Australia)
Tamas Sziranyi (SZTAKI -Hungary)
Rich Stern (CMU-USA)
Ricardo L. de Queiroz (UNB-Brazil)

EURASIP
European Association
for Signal Procesing

CTTC
Centre
Tecnològic
de Telecomunicacions
de Catalunya

UPC