Context-Based Gesture Recognition

José Antonio Montero[1,3] and L. Enrique Sucar[2]

[1] Instituto Tecnológico de Acapulco
Av. Instituto Tecnológico S/N
Acapulco, Guerrero, México
[2] Inst. Nacional de Astrofísca, Óptica y Electrónica
Luis Enrique Erro #1
Tonantzintla, Puebla, México
[3] ITESM Campus Cuernavaca
Av. Reforma 182-A, Lomas Cuernavaca, Morelos, México

Abstract. Most gesture recognition systems are based only on hand motion information, and are designed mainly for communicative gestures. However, many activities of everyday life involve interaction with surrounding objects. We propose a new approach for the recognition of manipulative gestures that interact with objects in the environment. The method uses non-intrusive vision-based techniques. The hands of a person are detected and tracked using an adaptive skin color segmentation process, so the system can operate in a wide range of lighting conditions. Gesture recognition is based on hidden Markov models, combining motion and contextual information, where the context refers to the relation of the position of the hand with other objects. The approach was implemented and evaluated on two different domains: video conference and assistance, obtaining gesture recognition rates from 94 % to 99.47 %. The system is very efficient so it is adequate for use in real-time applications.

1 Introduction

In everyday life, humans make intensive use of their hands to communicate with other humans, or to manipulate their environment. We denote such hand motions as *gestures*. The automatic recognition of human gestures is useful for many applications, such as human computer interaction (HCI), surveillance, collaborative environments, training and entertainment systems, and medical support. In many domains, gestures are characterized by the spatio-temporal structure of their motion patterns. These structures are intrinsically probabilistic and often ambiguous. In general, they can be treated as temporal trajectories in a high dimensional feature space representing closely correlated measurements on visual observations. For example, the spatio-temporal structure of a simple behavior such as moving the hand towards a key, could be represented by the trajectory of an observation vector given by the position and displacement of the hand centroid (Fig. 1). However, there are many gestures, in particular *manipulative* gestures [6], where the motion trajectory information is not sufficient to discriminate the gestures; for example, gestures realized with the same object (*erasing/writing* on a notebook). In these cases it is necessary to use addi-

J.F. Martínez-Trinidad et al. (Eds.): CIARP 2006, LNCS 4225, pp. 764–773, 2006.

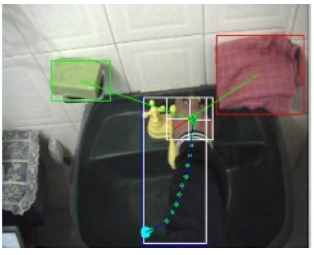

Fig. 1. Trajectory described by the hand centroid while realizing the gesture *opening the key*

tional information relative to the interacting objects (context) so the recognition process is more reliable.

An adequate selection of the visual features is very important for the success of gestures recognition systems. Current non intrusive approaches use a single camera (or a stereo system) to extract relevant features while detecting and tracking the hands, or other parts of the body of a person; which are used as input to a recognition system. The most commonly used methods for feature representation are: trajectory-based features [6], optical flow, and region-based features [11]. The choice is closely related to application and environmental conditions, and as mentioned before, this type of features are not enough for manipulative gestures.

The task of gesture recognition is a very challenging problem in computer vision. In most previous approaches, the gestures are confined to a predefined set, which requires that the subjects are well trained and the motions are uniform. The features and recognition algorithms are totally data-driven without any high level context information. For example, in [3], functions of different coordinates (cartesian, polar, angular) are used as feature vectors and applied in the recognition of six Tái Chi movements. Joint arm angles have been used by Quan, as a feature vector for human activities recognition [8]. However, the gestures realized in many real environments (office, washstand) where the person interacts with surrounding objects, are much more difficult than the above. In these cases, the motion is more natural, complex and dependent on the scenario. Then, it is necessary to use context information.

Hidden Markov Models (HMMs) are widely used for modeling temporal structures. They have been applied to speech recognition [9], learning and more recently to gesture recognition [6]. In this work we propose a novel approach that integrates motion and context features using HMMs to recognize manipulative gestures. These features are obtained using a single ceiling mounted camera observing the user hand interacting with surrounding objects. The motion or trajectory-based features selected are based on a previous study performed by the authors [12], and consist of the orientation and the magnitude in polar coordinates. The context features consider the relative position of the objects that interact with the user hand. We use these features to train HMMs and perform recognition. We evaluated experimentally the gesture recognition system in

two different domains: a video conference environment and a scenario in which a someone is washing his hands (for assisting senior or disabled persons). We focus our attention on human gestures performed by the hand that include interaction with surrounding objects and do not have a characteristic trajectory.

Given that the features are continuous, we discretized them into a set of symbols using vector quantization [4]. We then tested the recognition rate using HMMs with different number of hidden states and different number of training data. The results show a variation on the recognition rate depending on these parameters, ranging from 94 % to 99.47 %. We contrasted our models to those based only on motion information.

This paper is organized as follows. In Section 2 we describe hand localization and tracking using color-based, adaptive histograms techniques. Section 3 describes the feature extraction process. In Section 4 we present the learning and recognition system using HMMs. Section 5 includes the experimental results. We conclude with a summary and directions for future work.

2 Hand Detection and Tracking

For this work we consider hand gestures performed in two environments (office and washstand). Hand gestures are made on a planar space. The view of the scene is provided by a downward pointing, ceiling-mounted camera which offers several advantages for hand and object tracking, such as a less unobstructed perspective of the gestures. Our hand detection and tracking approach is divided in two phases. The first phase is the hand localization process that obtains the hand region using an adaptive color histogram. The second phase consists of the tracking algorithm, that generates the gesture trajectory by connecting the hand centroid along the continuous time sequence.

2.1 Hand Detection

For detection of the hand region in an image we use a color-based approach [10]. The human skin color is usually more distinctive and less sensitive to illumination changes if we use the *rgy* normalized color space proposed by [5]. Table 1 compares the *rgy* color space with others models for skin detection. Color histograms is the technique used to model the skin color space. To determine if a blob in the image contains skin pixels we apply the technique proposed by Ballard and Swain known as *histogram intersection* [10]. However, to improve the constraint of a fixed threshold value used by Swain, we are using Otsu's algorithm [7]. Based on Otsu's algorithm, our method incorporates adaptive thresholding, so it is able to tolerate changes in lighting conditions. Initial detection of the hand combines the color-based approach with motion information (Fig. 2), to make it more robust with respect to occlusions and illumination changes in real environments.

2.2 Hand Tracking

Once we have detected skin regions in an image sequence, the next step consists on tracking the hand using only color information. (Currently we assume that

Table 1. Average recognition obtained with the Bayesian classifier used to determine the skin class on different colors spaces

Color Space	Average Recognition
RGB	94.42
HSV	93.50
Yc_rc_b	91.46
RGY	97.20

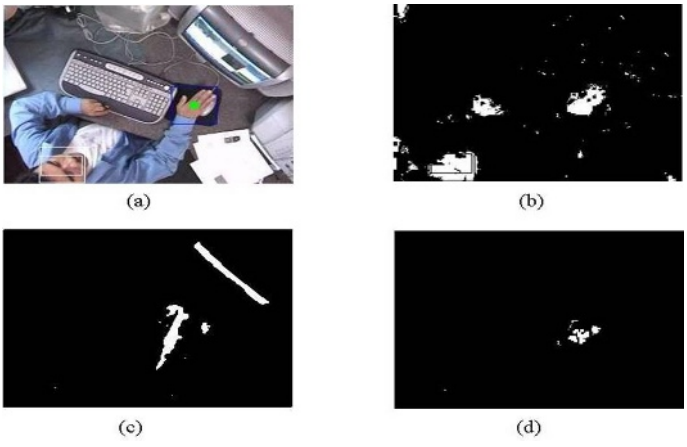

Fig. 2. Detection of human body parts using integration of color and motion information. (a) image original, (b) image segmented using color information, (c) image segmented using motion information, (d) integration of regions using color and motion segmentation.

the person performs all the gestures with the right hand.) For hand tracking, we first have to decide if the skin regions in the image are the face or the hand of a person. Hand/face detection is based on three rules. The first rule considers that only the hands and face of the person cause a significant movement in the images sequence. The second rule establishes a minimum threshold (number of skin labeled pixels) than a region must have to be considered a hand or face of a person. Experimentally we found that the region with the biggest skin area corresponds to the face of the person. The third rule indicates that the person is near the objects of interest, and his hand is the only skin region that establishes contact with the objects.

Based on these rules we detected the user hand in an image sequence, so the system starts tracking it. During tracking, we obtain the center points of the hand region. The center point of an object is defined using the centroid (X_c, Y_c):

$$X_c = \frac{\sum_x \sum_y B(x,y)x}{A}, \quad Y_c = \frac{\sum_x \sum_y B(x,y)y}{A} \quad . \tag{1}$$

where A is the number of pixels in the object and B is the binarized input object which takes two values, 1 for the hand and 0 for the background. Then we adjust a search window over the region defined by X_c, Y_c.

Hand tracking is realized by applying the hand detection process over the search window based on motion heuristics (maximum motion between frames), in the images sequence. The sequences of centroid points are detected by the hand localization algorithm, and thus, the gesture trajectory, G, is produced by connecting centroid points (see Fig. 1):

$$G = (x_1, y_1), ..., (x_n, y_n) \ . \tag{2}$$

Our system detects the gestures realized by a person in real environments when his hand interacts with relevant objects.

3 Object Detection and Tracking

Context information for supporting gesture recognition has been used in others works [1,11,6], but their methods impose many restrictions or are computationally complex, so it difficult apply them to real environments. We propose a simple approach based on the color and position of relevant objects in a domain. We use this information to support the gesture recognition process. The object detection and tracking process is a color-based approach. The objects existing in a scenario represent the contextual information that supports the gesture recognition process. The detection and localization of relevant objects is done using an adaptation of the work of Swain [10] and Bradsky [2]. Objects are modeled using color histograms for *hue-saturation* in the *hsv* color space. Training images are used to generate color histograms for each object using $30x32$ bins. Initially each object is searched over the full image, and then only in a search window (Bradsky applies the search over the entire image). Objects are detected using histogram intersection [10], so we obtain an image in gray scale where pixels close to 255 are from the object detected. Once an object is detected, we use a tracking algorithm proposed by Bradsky [2] over an appropriate search window

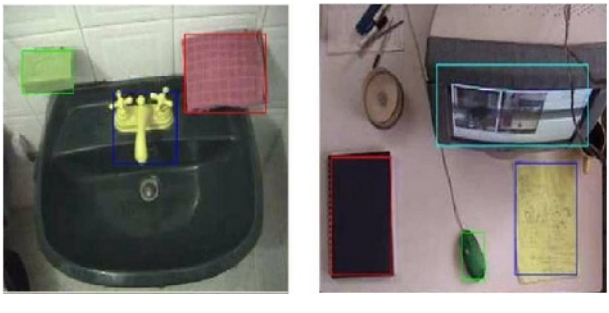

Fig. 3. Objects detected in two domains. Left image: towel, soap and key in a washstand. Right image: screen, book, mouse and note pad in an office environment.

for each object. The system maintains the position of each object in the scene. Figure 3 illustrates the object detection process in two domains: washstand and office environment. Objects detected in the washstand domain are: (1) a towel (pink rectangle), (2) a soap (green rectangle), and (3) a key (blue rectangle). Objects detected in the office environment are: (1) a notepad (blue rectangle), (2) a book (red rectangle), (3) a mouse (green rectangle), (4) a screen (aquamarine rectangle).

Objects interacting with the hand of a person represent contextual information. The gesture recognition system integrates this information with the motion attributes from the hand trajectory.

4 Recognition System

Gesture recognition is based on hidden Markov models (HMMs), integrating motion and contextual features.

4.1 Feature Selection

Motion features are obtained from the trajectory described by the right hand centroid when interacting with surrounding objects. In a previous analysis, we found that magnitude and orientation in polar coordinates are the best motion features for describing this type of gestures [12]. Context features include the distance from the hand centroid to each of the relevant objects that are detected in the scene. An example is shown in Fig. 4. Thus, we use the following set of features: (i) orientation, ρ, (normalized value) in polar coordinates, ii) magnitude, ϕ, (normalized value) in polar coordinates, and iii) context information (relative position of the objects in the scene). To obtain those features the following procedure is applied.

The center point, (C_x, C_y), of the gesture trajectory is obtained:

$$(C_x, C_y) = (\frac{1}{n}\sum_{t=1}^{n} X_t, \frac{1}{n}\sum_{t=1}^{n} Y_t) \ . \tag{3}$$

Based on this center point, the angle, θ_t, and distance, r_t, of each sample in the trajectory is obtained, relative to the center point:

$$\theta_t = tan^{-1}(\frac{Y_t - C_x}{X_t - C_y}) \ . \tag{4}$$

$$r_t = \sqrt{(X_t - C_x)^2 + (Y_t - C_y)^2} \ . \tag{5}$$

By calculating the longest distance from the center point to any point in one gesture, r_{max}:

$$r_{max} = max_{t=1}^{n}(r_t) \ . \tag{6}$$

the normalized $(0-1)$ distance, ρ_t; and normalized angle, ϕ_t, are:

$$\rho_t \frac{r_t}{r_{max}}, \phi_t = \frac{\theta_t}{2\pi} \ . \tag{7}$$

By integrating the motion and context parameters, we obtain the following feature vector:

$$F = (\rho, \phi, Context) \ . \tag{8}$$

Given that these are continuous variables and we are using discrete models, these features are codified in 64 discrete observation symbols using the k-means algorithm [4].

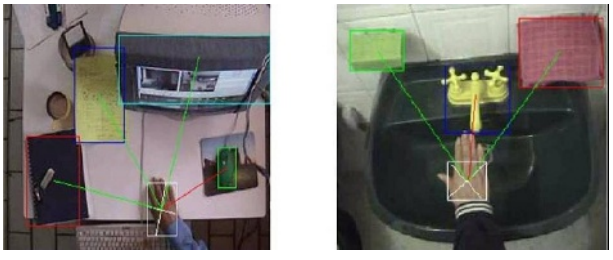

Fig. 4. The distance between the user hand and objects determine the context. In this case, the red line shows the closest object.

4.2 Hidden Markov Models

HMMs have the ability to accurately characterize data exhibiting sequential structure in the presence of noise, such as human gestures, finding the most likely sequence of states that may have produced a given sequence of observations. The HMM topology used in this paper is the classical left–right structure, which is typical for motion ordered paths. As usual, one model was trained for each gesture class, and for recognition we selected the model with the highest probability. The complete parameter set of the HMM can be expressed compactly as $\lambda = (A, B, \Pi)$, where A is the probability transition matrix, B is the observation probability matrix, and Π is the initial probability vector. Three basic problems must be solved for the application of HMMs: evaluation (classification), decoding, and training. We approach the above problems with the standard techniques [9]: forward algorithm, Viterbi algorithm, and the Baum-Welch algorithm.

5 Experimental Results

In this study, we focus our attention on human gestures performed by the hand that include interaction with known objects. The type of gestures considered in the experiments are the realized in two domains: office environment and washstand room. The gestures that recognition system will try to identify are the following: *erasing, writing, using the mouse and turning the leaves of a book*, in the office environment (see Fig. 5); and *using soap, opening the key, closing the key, drying the hands, taking the towel, and washing hands*, in the washstand domain (see Fig. 5).

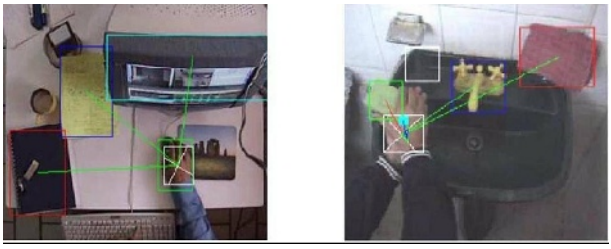

Fig. 5. Example of gestures recognized in the two domains: left image office environment, right image washstand

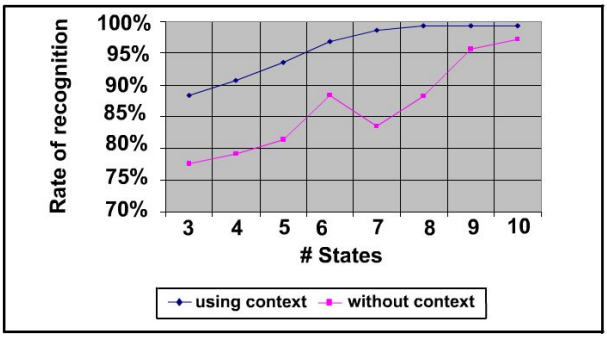

Fig. 6. Gesture recognition rates with and without context information vs. number of hidden states for the office domain

5.1 Training

We use two different training data set: one set for the office domain and another for the washstand domain. In the office domain a data base with 4200 data points was generated for training and testing. Data stored in the DB was obtained from 100 gestures sequences with 5 different gestures realized by a person in a sitting position and interacting with objects in the office environment. We used 2100 data points for training and 2100 for testing. In the washstand domain, we use a data base with 3600 data points generated for training and testing. These data were obtained from 50 gestures sequences with 6 different gestures realized by a person in a standing position and interacting with surrounding objects. In this case, we used 2560 data points for training and 1040 data points for testing. In both cases, gesture data was captured by a ceiling mounted camera pointed downward, under normal illumination conditions in an office and in a bathroom.

5.2 Results

We tested the gesture recognition system for the 11 different types of gestures, related to the manipulation of each object in the two scenarios.

For the office environment, we compared the recognition rate with only motion information vs. motion and context, for different number of hidden states in the HMM. Figure 6 shows graphically the difference between both schemes. With only motion, the recognition rate varies from 75% for 3 hidden states to 97% for 10 hidden states. In the other case, when we integrate motion and context, the recognition rate varies from 88% using 3 hidden states to 99.47% using 10 hidden states. There is a significant improvement by incorporating context.

For the washstand domain, we only show the results with motion and context features. The confusion matrix is depicted in Table 2. The recognition rate varies from 80% to 100%. The average recognition in this domain is 94%.

Table 2. Confusion matrix for the washstand domain

	Open/close Key	Washing hands	Take soap	Soaping	Take towel	Dry hands	Av %
Open/close Key	15	0	0	0	0	0	**100.00**
Washing hands	0	14	0	0	1	0	**99.33**
Take soap	0	1	12	0	1	1	**80.00**
Soaping	0	0	0	14	1	0	**93.33**
Take towel	0	0	0	0	14	1	**93.33**
Dry hands	0	0	0	0	0	15	**100.00**

5.3 Implementation

The system was implemented in a personal computer, Intel Pentium 4, with a 1.3 Ghz processor; and a Sony TRV19 CCD color video camera. The video card (PixelView) captures 30 frames of 320x240 pixels per second. The processing rate of the recognition system is between 12 and 15 fps. The system is codified in Visual C++ 6.0 over Windows XP.

6 Conclusions

In this work we propose a novel approach that integrates motion and context features using HMMs to recognize manipulative gestures. These features are obtained using a single ceiling mounted camera observing the user hand interacting with the surrounding objects. The motion features consist of the orientation and the magnitude in polar coordinates. The context features consider the relative position of the objects that interact with the user hand. We evaluated experimentally the gesture recognition system in two different domains: a video conference environment and a scenario in which a someone is washing his hands. The results in both domains are very good, showing a significant improvement in relation to using only motion features. We believe that using additional context information, such as the relation between hands, or hands and face, could provide another important set of features to improve gesture recognition for more complex scenarios.

References

1. D. Ayers and Mubarack Shah. Monitoring human behavior in an office environment. *PAMI*, 7:780–794, July 1997.
2. G.R. Bradski. Computer vision face tracking as a component of a perceptual user interface. *Workshop on Applications of Computer Vision*, 1:214–219, 1998.
3. Lee W. Campbell, David A. Becker, Ali Azarbayejani, Aaron F. Bobick, and Alex Pentland. Invariant features for 3-d gesture recognition. *In Proc. of the Intl. Workshop on Automatic Face and Gesture Recognition, Zurich*, 5:157–162, 1996.
4. Robert M. Gray. Vector quantization. *IEEE ASSP Magazine*, 7:407–467, Apr 1984.
5. Miriam Martnez and Luis E. Sucar. Learning and optimal naive bayes classifier. *36vo Congreso de Investigacion y Desarrollo del Tecnolgico de Monterrey: Impulsando la Economa Basada en Conocimiento*, 2006.
6. Darnell J. Moore, Irfan A. Essa, and Monson H. Hayes. Exploiting human actions and object context for recognition tasks. *Proc. IEEE of the 7th International Conference on Computer Vision*, 1, Sep 2000.
7. N. Otsu. A threshold selection method from gray-evel histograms. *IEEE, Trans. Sys, Man, and Cybernetics*, 1:62–66, Jan 1979.
8. D. L. Quan. Gesture recognition with a data glove. *IEEE Proc. National Aerospace and Electronics Conf.*, 2, 1990.
9. L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):267–293, 1989.
10. M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, pages 11–32, 1991.
11. M. Thonnat and N. rota. Video sequence interpretation for visual surveillance. *Third International Workshop on Cooperative Distributed Vision*, pages 1–9, November 2000.
12. Jose Antonio Montero V. and Luis Enrique Sucar S. Feature selection for visual gesture recognition using hidden markov models. *Fifth Mexican International Conference on Computer Science*, 1:196–203, 2004.