Hand gesture recognition for man - machine interaction

Tomasz Kapuscinski, Marian Wysocki
Rzeszow University of Technology, Control and Computer Science Chair
W. Pola 2, 35-959 Rzeszow, Poland
{tomekkap,mwysocki}@prz.rzeszow.pl

Abstract

The paper addresses some basic problems of hand gesture recognition. Three steps important in building gestural vision - based interfaces are discussed. The first step is the hand location through detection of skin colored regions. Representive methods based on 2D color histograms are described and compared. The next step is the hand shape (posture) recognition. An approach that uses the morphological hit - miss operation is presented. Finaly, application of hidden Markov models in recognition of dynamic gestures is explained. An experimantal system that uses the discussed methods in real time is described and recognition results are presented.

1. Introduction

Human computer interaction has become an increasingly important part of our daily lives. One can expect that computers of the future, and future robots, as the special case, will interact with humans in a natural way. Recent examples of human computer interfaces include integration of speach understanding, gaze tracking, lip reading and gesture recognition [8]. Gesture is a very intuitive way for human communication and it is also very useful as interface for human robot interaction. Cleary one way for machines to communicate with people more effectively is for them to be able to interpret our gestures. Hand is the most functional part of human body: pointing, handling or expressing some symbols, and hand gestures comprise the majority of gestures used by humans. In this paper we will use the term gesture to refer to hand gesture.

Visual based automatic gesture recognition has recently acquired much attention [11, 16]. Gestural interfaces for robots must satisfy several constraints [19]. The system should:

- cope with variable and possible complex background,
- be person independent,
- not recquire the user to wear markers or colored gloves,
- cope with various lighting conditions,
- be capable of real time performance.

This paper reflects our experience in building a vision - based interface for gesture recognition. In section 2 we discuss the human skin segmentation problem. Representative methods based on 2D color histograms are described and compared. Section 3 considers hand shape (posture) classification. We describe an approach that uses mathematical morphology. Section 4 explains recognition of dynamic gestures with hidden Markov models. Section 5 presents an experimantal system and sample results of gesture recognition.

2. Skin segmentation

Locating and tracking patches of skin - colored pixels through an image sequence is a tool used in many gesture recognition systems. An important chalange of any skin-color detecting system is to accomodate varying illumination conditions that may occur within the image sequence. Some robustness may be achieved via the use of luminance invariant color spaces. An important aspect of any skin color recognition is choosing a color space that is relatively invariant to minor illumination changes. This is because the shape of the skin and nonskin distributions depends on the chrominance space. Two important cirteria for implementing an efficient skin - color based hand segmentation and gesture recognition are[17]: (1) how well a given chrominance model can describe the distributions in a given space, (2) the amount of overlap between the skin and non -

skin distributions in that space. One of the most popular color spaces that have proved to be robust to minor illuminant changes is the normalized RGB. The color pixels are normalized by dividing out the luminance component[14]

$$r = \frac{R}{L},\ g = \frac{G}{L},\ b = \frac{B}{L},\ L = R + G + B \qquad (1)$$

This removes the effect of changes in the orientation of the skin surface with respect to a light source. The intensity - normalized pixels from a region of an image known to contain skin can be used to define a normalized two dimensional histogram[9]

$$\tilde{h}_s(r,g) = \frac{1}{N_s} h_s(r,g) \qquad (2)$$

of skin color, where N_s - sum of the histogram $h_s(r,g)$ over r and g. The histogram $\tilde{h}_s$ is used as a representation of the probability density function. The main disandvantage is that histograms in general are bad for representing sparse data, where only a fraction of the necessary samples is available. This can be dealt with via interpolation or Gaussian filtering of the histogram. Now we explain three histogram - based approaches to human skin detection.

1. The histogram $\tilde{h}_s(r,g)$ calculated during the initialization phase is smoothed by Gaussian filtering and then used in transformation of the input image into a gray level probability image

$$z(i,j) = 255\tilde{h}_s(r(i,j), g(i,j)) \qquad (3)$$

where (i,j) represent pixel coordinates. The gray level image is convolved with Gaussian kernel and then thresholded using the threshold obtained after approximation of the gray level histogram by two normal distributions. The binary image is improved by morphological opening - closing (OC) filtering [15]. The extraction process is summarized in Fig. 1.

2. The probability distribution of the skin color is approximated by 2D Gaussian function $G(r,g)$. The 2x1 mean value vector and the 2x2 covariance matrix are estimated using known formulae and the histogram $\tilde{h}_s(r,g)$, obtained during initialization. The Gaussian distribution model $G(r(i,j), g(i,j))$ is used instead of the $\tilde{h}_s$ to construct the gray level probability image. Further procedure is the same as that in 1. It has been shown in [17] that the single Gaussian model of the skin color distribution in the normalized (r,g) space produces results that are comparable to those obtained with more sophisticated mixture Gaussian density models, constructed

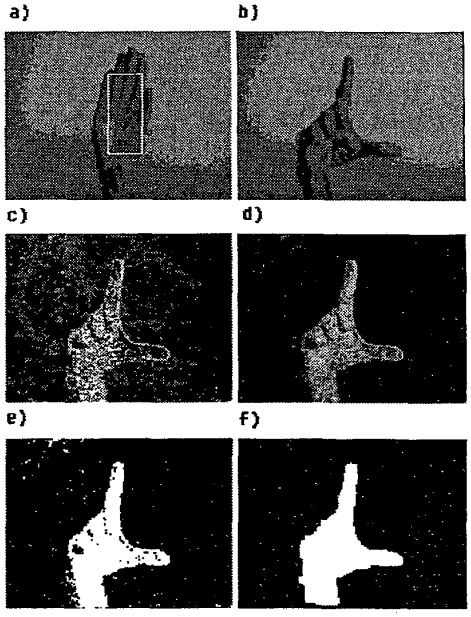

Figure 1. Skin color extraction: a) skin region used for calculation of the histogram $\tilde{h}_s$, b) input image, c) gray level (probability) image, d) smoothed probability image, e) image (d) after thresholding, f) image (e) after OC filtration.

by the EM (expectation maximization) method.

3. The method uses Bayes theorem [4, 9] to choose the most likely hypothesis, given the value of a feature. The second normalized two dimensional histogram is build during the initialization phase, i. e. the histogram

$$\tilde{h}_{ns}(r,g) = \frac{1}{N_{ns}} h_{ns}(r,g) \qquad (4)$$

of non - skin color, where N_{ns} - sum of the histogram $h_{ns}(r,g)$ over r and g. The probability of a color vector (r,g) given skin is approximated by

$$P(r,g|skin) = \tilde{h}_s(r,g) \qquad (5)$$

the probability of a color vector (r,g) given non - skin is approximated by

$$P(r,g|nonskin) = \tilde{h}_{ns}(r,g) \qquad (6)$$

Bayes rule states that the probability of skin given a color vector (r,g) is

$$P(skin|r,g) = \frac{P(r,g|skin)P(skin)}{P(r,g)}$$

and the probability of nonskin given (r, g) is

$$P(nonskin|r,g) = \frac{P(r,g|nonskin)P(nonskin)}{P(r,g)}$$

For any pixel, if the ratio (7)

$$\frac{P(skin|r,g)}{P(nonski|r,g)} = \frac{P(r,g|skin)P(skin)}{P(r,g|nonskin)P(nonskin)} \cong \frac{\tilde{h}_s(r,g)P(skin)}{\tilde{h}_{ns}(r,g)P(nonskin)} \quad (7)$$

is greater than one, then the pixel will be classified as skin, otherwise it will be classified as non - skin. There are two possible assumptions: (1) $P(skin) = P(nonskin)$ (2) the values $P(skin)$ and $P(nonskin)$ are estimated from the training data. The first case corresponds to maximum likelihood (ML) estimation, the second one to maximum a posteriori (MAP) estimation.

Fig. 2. shows comparison of results obtained for sample data set of 162 hand pictures prepared in different illumination conditions. Normalized RGB space and some other color spaces[14] have been used. To reduce the sensitivity of the segmentation to changes in illumination only chromatic coordinates of those spaces were considered. Thus 2D histograms and simple adaptation of the methods described earlier could be applied. The skin pixels were segmented by hand and compared with those segmented automatically. E denotes percent of errors (skin detected as non - skin + non - skin detected as skin).

The methods of skin detection described in this paper work well within rather narrow set of conditions. They address only the time - varying illumination defined over a narrow range (white light). Further research is needed to cope with multiple sources with time varying illumination and single or multiple colored sources.

3. Hand posture classification

There are various techniques proposed in literature for hand posture classification. Some approaches generate 3D hand models and evaluate their matching confidence with the 2D image [13]. This method is computionally expensive. Other authors use a principal component analysis (PCA) of the distribution of hand images [9], image moments [3, 5], elastic graph matching [18] or orientation histograms [3]. In this section we present another method developed by our team [6, 7]. The method uses morphological hit - miss transform [15] applied to binary images obtained by skin detection. Binary hit - miss operation extracts geometrical

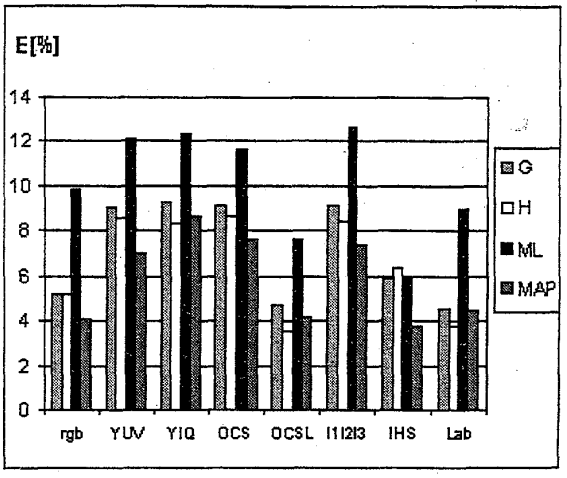

Figure 2. Comparison of detection methods in various color spaces, H, G, (ML, MAP) - the detection methods 1, 2, 3, respectively

features from a binary object. The masks used in the operation determine the type of features that are extracted. The operation is defined as

$$Hit\ Miss(X, B_1, B_2) = (X \ominus B_1) \cap (X^c \ominus B_2) \quad (8)$$

where X is the object which is being operated on, X^c denotes its complement (background), $\ominus$ denotes the morphological erosion operation and B_1 and B_2 are the specified structuring elements with the requirement

$$B_1 \cap B_2 = \emptyset \quad (9)$$

B_1 and B_2 are expressed with respect to a local origin Ω (called reprezentative point). The hit - miss operation applied to the object X means that the reprezentative point Ω is moved systematically across the object. For a point p to be in the resulting set, two conditions should be fulfilled simultaneously. Firstly the part B_1 of the composite structuring element that has its reprezentative point at p should be contained in X and secondly, the part B_2 of the composite structuring element should be contained in X^c. Fig. 3. shows structuring elements developed for two simple gestures. A distance between the structuring elements B_1 and B_2 ensures efficiency of the method under small variations of hand size and orientation. Orientation of the structuring elements is computed using central moments M_{20}, M_{02}, M_{11} of the hand and the following formula to derive the slope θ of the maximum axis

$$tan^2\theta + \frac{M_{20} - M_{02}}{M_{11}} tan\theta - 1 = 0 \quad (10)$$

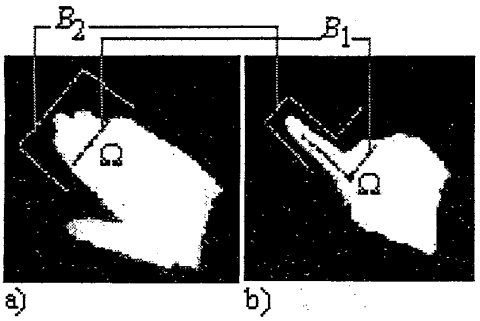

Figure 3. Sample structural elements of the hit - miss operation

The hand image is rescaled by a scaling factor obtained during the initialization phase on the basis of image of the open hand. More complicated hand postures such as e.g. Polish Finger Alphabet are classified with neural network using feature vectors obtained by hit - miss operations.[6, 7].

4. Recognition of dynamic gestures

Human hand gestures are spatiotemporal entities. Usually the performance of the gesture is not perfect. The same gesture changes in time and space even if one person performs it twice. Human performance involves two distinct stochastic processes: human mental states and resultant actions [20]. The mental state is immeasurable, the action is measurable. Therefore many researches use hidden Markov models to recognize hand gestures[1, 10, 11, 16]. The HMM are double stochastic processes as governed by an underlying Markov chain with a finite number of states, and a set of random functions each of which is associated with one state [12]. In the discrete time instant $t = 1, 2, \ldots, T$ the process is in one of the states $s(t)$ and generates an observation $o(t)$ according to the random function corresponding to the current state. The model is hidden in the sense that all that can be seen is a sequence of observations. Markov model composed of N states $s_1, s_2, \cdots s_N$ is definied by $\lambda = (A, B, \pi)$, where

$$A = [a_{ij}],\ i, j = 1, 2, \cdots, N$$

is a transition matrix between states,

$$B = [b_1, b_2, \cdots, b_N]$$

is an observation probability matrix and

$$\pi = [\pi_1, \pi_2, \cdots, \pi_N]$$

is the initial probability distribution for the states. The elements a_{ij}, b_i and π_i are defined as follows:

$$\begin{aligned} a_{ij} &= P(s(t+1) = s_i | s(t) = s_j) \\ b_i &- \textit{probability function of observations} \\ &\quad \textit{given state } s_i \\ \pi_i &= P(s(1) = s_i) \end{aligned}$$

Hidden Markov Models are discrete or continuous [12]. In the first case the observations are interpreted as elements of a finite set V called vocabulary or codebook. Then the function $b_i = b_i(k)$ corresponds to the probability of observing the k-th symbol v_k of the codebook

$$b_i(k) = P(o(t) = v_k | s(t) = s_i)$$

If observations can not be restricted to a codebook, the HMM is continuous and b_i represents a probability density function, usually proposed as multidimensional Gaussian or a mixture of Gaussians.

An HMM can perform a number of tasks based on sequences of observations.

1. Learning: Given an observation sequence $O = \{o(1), o(2), \cdots, o(T)\}$ and a model λ, the model parameter of λ can be adjusted such that $P(O|\lambda)$ is maximized.

2. Sequence classification: For a given observation sequence $O = \{o(1), o(2), \cdots, o(T)\}$ by computing $P(O|\lambda_i)$ for a set of known models λ_i, the sequence can be classified as belonging to class i for which $P(O|\lambda_i)$ is maximized.

3. Sequence interpretation: Given observation sequence $O = \{o(1), o(2), \cdots, o(T)\}$ and an HMM λ, applying the Viterbi algorithm [12] gives a signle most likely state sequence $S = \{s(1), s(2), \cdots, s(T)\}$.

There exist many different types of HMMs. One which has the property that it can model signals whose characteristics change over time in a succesive manner is called a left - right model or Bakis model [12]. States in the Bakis model can be aligned in such a way that only left - to - right transitions are possible. Further, it has a well defined initial and final state. Frequently the model includes the additional constraint that no more than one state may be skipped in any transition and thus it allows transitions to the same state, the next and the one after the next, as shown in Fig. 4. For a given set of features, adequate for recognition, the number of states is usually determined experimentally. In principle, many possible sets of features contain enaugh information to reconstruct original gesture, but often

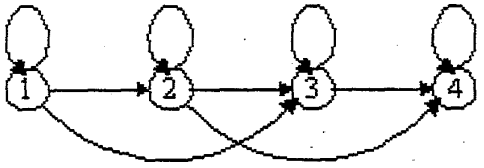

Figure 4. A four-state Bakis HMM

the choice is not obvious. A gesture in this paper is described as a sequence of codes characterizing direction of hand movement. Each segment of trajectory is coded into one of the 8 directions as shown if Fig. 5.

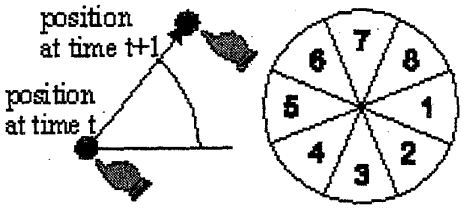

Figure 5. Hand movement and 8 directional code words

5. Experiments

An experimental system is implemented on a personal computer Celeron 333MHz with an image frame - grabber Matrox Meteor II. Input images are captured by a CCD Panasonic camera with the resolution 768 x 576 and with the frequency 25 frames per second. The system is enhanced by a video projector projecting on an ordinary white board observed by the camera. The software implemented in Visual C++ 6.0 on Windows 98 contains the following modules: camera calibration, building of skin color model, skin recognition, hand tracking, posture detection using morphological hit - miss operation and gesture recognition by HMMs. The calibration procedure determines 8 parameters of an afine transformation (translation, rotation and scaling). The parameters are computed on the basis of four corners of the white board, recognized by the Hough transform[15]. Such simple calibration turned out to be sufficient for hand tracking and recognition. Model of skin color distribution is constructed from a region of the image known to contain skin. Therefore in the model building phase the user is invited to place his hand in a rectangular projected on the board. Skin recognition is performed according to one of the methods described in section 2. The gestures are performed on the white board. To accelerate computations, small window 160 x 120 pixels containing the hand is considered. The center of the window is estimated by an observer

$$\hat{x}_{k+1} = F\hat{x}_k + H(y_k - C\hat{x}_k),\ k = 0, 1, 2, \cdots \quad (11)$$

where the state vector $\hat{x}$ contains estimated position and velocity of the window center with respect to the horizontal axis, and related estimates for the vertical direction. The observation vector y contains coordinates of the center of gravity of the detected hand. The state matrix F and the observation matrix C are as follows

$$F = \begin{bmatrix} 1 & \Delta T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta T \\ 0 & 0 & 0 & 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

where ΔT denotes sampling period = 40 ms. The gain matrix H is determined by the pole placement method [2]. Simplified model of motion, characterized by the matrix F, assumes constant velocity. We noticed that the observer ensures correct tracking of hand movements performed with maximal velocity about 1.1 m/s. Without observer, i. e. the estimated center of the window is taken at the gravity center of the hand detected in the preceding step, the maximal velocity is about 1.5 times slower. Correct tracking means that the whole hand is contained in the window. Precise initialization is essential for reliable tracking. One possible method is to mark the hot spot region on the white board. We assumed it at the top left corner. Only one hand is actually tracked.

Three hand postures are important for the system. The first two postures: "move the icon" and "draw" are shown in Fig. 3a, b, respectively. Third posture "do nothing" is formed by five open fingers. When the system recognizes the posture "move icon", a rectangle containing the hand is displayed on the board. When the posture "draw" is detected, the line drawn by the index finger is displayed as well as the related directional codewords (see Fig. 5.) are computed and registered. To remove garbage gestures, the user performs the gesture "do nothing" before meaningful gesture start and after meaningful gesture stop.

We defined the digits $0, 1, \cdots, 9$ as gestures and used two, three or four state HMM to model each gesture. Data set of 600 gestures (60 gestures of each type) performed by five persons was prepared. The gestures sampled at about 40 ms sampling period (25 frames

per second) were performed with various velocities. Lenghts of obtained codeword sequences change from about ten to about thirty. A training set of 400 gestures was used to estimate parameters of 10 HMMs by the Baum - Welch algorithm [12]. Remaining gestures were used for testing. We obtained the following recognition rates: 98 % for training, 97.5 % for testing, 97.83 % total. Two state models turned out to be sufficient for the digits 1, 7, three state models for 2, 3, 4 and four state models for the digits 0, 5, 6, 8, 9. Four state models contained additional transitions allowing one state to be skipped (see Fig. 4.). The HMMs are implemented in the system, and gestures (digits) can be recognized on - line using forward algorithm[12].

6. Conclusions

The paper addresses some basic problems of hand gesture recognition. We discussed three main steps in building of gestural interfaces. The first step is the hand location through detection of skin colored regions. The next step is the hand posture classification. Thirdly, the application of hidden Markov models in recognition of dynamic hand gestures is considered. An experimantal system that uses the direct camera input and the discussed methods in real time is described. Obtained recognition rate about 98 % is promising. Future research will focus on two areas. The first area is more reliable skin color detection under variation of illumination during tracking. The second one is automatic gesture spotting and recognition of sequences of dynamic gestures.

Acknowledgment The support of the State Comittee for Scientific Research (Warsaw) under grant 8T11A01117 is gratefully acknowledged.

References

[1] B. Bauer and H. Hienz. Relevant features for video - based continuous sign language recognition. *Proc. of the 4th Int. Conf. on Automatic Face and Gesture Recognition, Grenoble, France*, pages 440–445, 2000.

[2] C.-T. Chen. *Analog and Digital Control Systems Design.* Sanders College Publishing, New York, 1993.

[3] W. T. Freeman, K. Tanaka, J. Ohta, and K. Kyuma. Computer vision for computer games. *Proc. of the 2nd Int. Conf. on Automatic Face and Gesture Recognition, Killington, Vermount*, 1996.

[4] K. Fukunaga. *Introduction to statistical pattern recognition.* Acad. Press, London, 1972.

[5] E. Hunter, J. Schlenzig, and R. Jain. Posture estimation in reduced model gesture input systems. *Int. Workshop on Automatic Face and Gesture Recognition, Zurich, Switzerland*, 1995.

[6] J. Marnik. Polish finger alphabet signs recognition using mathematical morphology. *Archives of Theoretical and Applied Informatics*, 2(11):157–169, 1999. (in Polish).

[7] J. Marnik, T. Kapuscinski, and M. Wysocki. Polish finger alphabet signs recognition. *Proc. of the 45th Internat. Scientific Colloquium, Ilmenau Tech. Univ*, pages 61–66, 2000.

[8] I. Marsic, A. Medl, and J. Flanagan. Natural communication with information systems. *Proc. of the IEEE*, 8:1354–1366, 2000.

[9] J. Martin and J. L. Crowley. An apperance - based approach to gesture recognition. *Proc. of the 9th Int. Conf. on Image Analysis and Processing, Florence, Italy*, 1997.

[10] J. Martin and J.-B. Durand. Automatic handwriting gestures recognition using hidden markov models. *Proc. of the 4th Int. Conf. on Automatic Face and Gesture Recognition, Grenoble, France*, pages 403–409, 2000.

[11] V. I. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human - computer interaction. *IEEE Trans. PAMI*, 19(7):677–695, 1997.

[12] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–286, 1989.

[13] J. M. Rehg and T. Kanade. *Digit eyes: Vision - based human hand tracking.* Technical Report CMU-CS-99-220, Carnegi Mellon University, 1993.

[14] S. J. Sangwine and R. E. N. Horne (Eds.). *The Colour Image Processing Handbook.* Chapman & Hall, London, 1998.

[15] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision.* Chapman & Hall, London, 1994.

[16] T. Starner, J. Weaver, and A. Pentland. Real - time american sign language recognition using desk and wearable computer based video. *IEEE Trans. PAMI*, 20(12):1371–1375, 1998.

[17] J.-G. Terrillon, M. N. Shirazi, H. Fukamachi, and S. Akamatsu. Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images. *Proc. of the 4th Int. Conf. Automatic Face and Gesture Recognition, Grenoble, France*, pages 54–61, 2000.

[18] J. Triesch and C. von der Malsburg. Robust classification of hand postures against complex backgrounds. *Proc. of the 2nd Int. Conf. on Automatic Face and Gesture Recognition, Killington, Vermont*, 1996.

[19] J. Triesch and C. von der Malsburg. A gesture interface for human - robot - interaction. *Proc. of the 3rd Int. Conf. Automatic Face and Gesture Recognition, Nara, Japan*, pages 546–551, 1998.

[20] J. Yang, X. Yangsheng, and C. S. Chen. Human action learning via hidden markov model. *IEEE Trans. SMC, Part A*, 27(1):34–44, 1997.