

Online Visual Recognition of Dynamic Gestures Using Dynamic Bayesian Networks

Héctor Hugo Avilés-Arriaga and Luis Enrique Sucar

Instituto Tecnológico y de Estudios Superiores de Monterrey,
Campus Cuernavaca
Av. Paseo de la Reforma No. 182-A Col. Lomas de Cuernavaca
C.P. 82589 Cuernavaca Morelos México
00374765@academ01.mor.itesm.mx, esucar@campus.mor.itesm.mx

Abstract. Gestures are a natural and effective alternative to command mobile robots. This paper describes an online visual recognition system to recognize a set of 5 dynamic gestures executed with the user's right hand and oriented to command mobile robots. The system employs a radial scan segmentation algorithm combined with a statistical-based skin detection method to find the candidate face of the user and to track his right-hand. It uses 4 simple features to describe the user's right-hand movement and Dynamic Bayesian Networks as recognition technique. This system is able to recognize these five gestures in real time with an average recognition rate of 84.01%, better result than using hidden Markov models for recognition.

1 Introduction

In the last years, visual recognition of gestures has emerged as a very broad field of study in Human-Machine Interaction research. Diverse interesting applications have been developed in this field [3, 4, 9] in both, isolated gestures recognition [17] and its relation with spoken language [16]. The most common technique for gesture recognition are hidden Markov models (*HMM's*). These models are effective for simple gestures, but not rich enough for more complex gestures. An alternative, richer representation are dynamic Bayesian networks (*DBN's*). We present an online visual system to recognize a set of five right-arm dynamic gestures using *DBN's* as recognition technique. These gestures were chosen for their potential application in human-mobile robot interfaces. In contrast with other similar systems [5], these gestures are widely understood around the world [1]. The system employs a radial scan segmentation algorithm combined with a statistical-based skin detection method to find the user's candidate face and track his right-hand. Also, the system uses very simple features to describe the user's right-hand movement. These features are applied as motion observations for the dynamic Bayesian networks. The system is able to recognize these five gestures in real time with an average recognition rate of 84.01%, better results than using hidden markov Models for recognition.

2 Methodology

We divide our methodology to recognize dynamic gestures into the next four stages:

- i) Gestures selection
- ii) User localization by skin color
- iii) Tracking of the user's right-hand and extraction of motion features
- iv) Visual gestures recognition using dynamic Bayesian networks

Dynamic gesture recognition using online visual systems involves diverse fields of study. To improve the system usefulness, it is important to take into account social implications and variations of gestures in their selection. Locating users on images, as well as tracking them and extract movement features are problems related to computational vision. Finally, associating this information with previously stored data is a problem corresponding to pattern recognition.

2.1 Gestures selection

The system considers five gestures: *go-right* (Figure 1a), *go-left* (Figure 1b), *come* (Figure 1c), *attention* (Figure 1d) and *stop* (Figure 1e). These gestures are widely understood around the world. Moreover, we can naturally associate them simple tasks for the mobile robot like left or right motion.

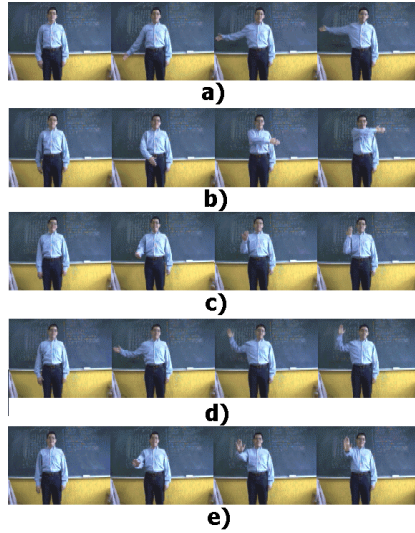


Fig. 1. Gestures consider by our system: a) go-right, b) go-left, c) come, d) attention and e) stop.

2.2 User localization by skin color

The skin pixels classification method used by our system is based on the method developed by Jones [6]. He suggests the construction of two histogram color models, one for skin pixels and other for non-skin pixels, both in 24 bits RGB color space. Using the histogram models, we built a skin classifier based on Bayes rule:

$$P(piel|rgb) = \frac{P(rgb|piel)P(piel)}{P(rgb|piel)P(piel) + P(rgb|\neg piel)P(\neg piel)}$$

where $P(skin)$ and $P(\neg skin)$ are *a priori* probabilities, and $P(rgb|skin)$ and $P(rgb|\neg skin)$ are taken directly from the skin and non-skin histograms respectively. Pixels are classified as skin if $P(skin|rgb) > \Theta$, where $0 \leq \Theta \leq 1$ is a threshold. However, to increase the speed of the classification stage, our system employs the rule $P(rgb|skin) \geq P(rgb|\neg skin)$, which is equivalent to use $P(skin|rgb) > \Theta$ with $\Theta = P(skin)$ [Jones 98]. Since $P(skin|rgb)$ and $P(skin|\neg rgb)$ are taken directly from the skin histogram model, other calculations are not necessary.

To segment skin regions, the algorithm developed in our system is based on the radial scan segmentation algorithm proposed by SAVI Group [7]. The algorithm traces lines with certain angular distance among them, from the center of the image to its edges, classifying pixels over these lines, as skin or non-skin pixels. At the same time, it uses some segmentation conditions to grow skin regions. The advantage of this algorithm is the fast speed to find skin regions. For example, if we consider an square image of size $N \times N$ with $N = 480$ pixels, then we have an image of 230,400 pixels. If we use 360 lines to sweep the image, then we have to visit only $360 \times 240 = 86,400$ pixels, *i.e.*, 37.5% of the total pixels in the image. These reductions in the search space of the image are very important when we are working with real-time systems.

To find the user's candidate face the skin segmentation algorithm is applied only in the upper half of the image, supposing that the user's face is the predominant skin region in this area. Figure 2 shows an example of face segmentation. Once the face has been located in the image, to find the user's right hand the skin segmentation module is applied on a small region that contains the right hand, using anthropometric measures and considering that the arm is initially at the rest position (see figure 2).

2.3 Tracking of the user's right-hand and extraction of motion features

Once the system has localized the hand at the rest position, the hand can be tracked in the image sequence. For tracking, in every image we define a 120×120 search window around the previous position of the hand. The size of this search window is based on the usual speed of arm movements, obtained experimentally. Some images that show hand tracking are presented in figure 3. With this strategy, the hand tracking is performed in real time *i.e.*, 30 images per second.

Motion feature extraction is derived from the method proposed by Starner [9]. Our system uses 4 simple features to describe the hand displacement: $\Delta area$

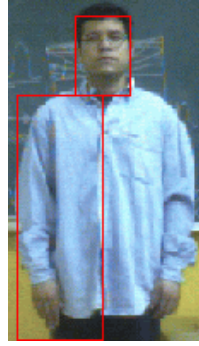


Fig. 2. Example of face segmentation and localization of the user's right hand.

or changes in area of the hand, Δx or changes in hand position on the x -axis of the image, Δy or changes in hand position on the y -axis of the image, and $\Delta posture$ or comparison between sides of the square region that segments the hand. To evaluate the hand motion between two images, these features take only one of three possible values, (+), (-) or (0) that indicate increment, decrement or no change, depending on the position and posture of the hand in the previous image.

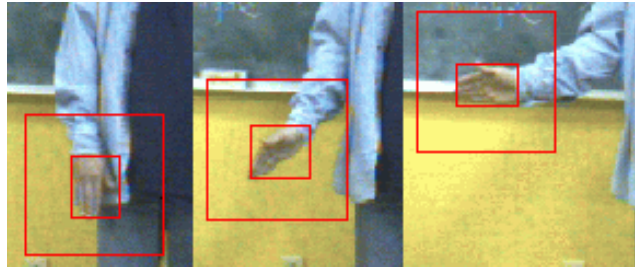


Fig. 3. Tracking of the user's right hand.

Consider the Δx feature. If the motion of the hand's centroid is to the user's right side, then $\Delta x = (+)$. If the motion is to the left side of the user, then $\Delta x = (-)$. If the system doesn't detect motion on the x -axis, then $\Delta x = (0)$. The Δy feature is obtained in a similar way. In addition to the $(\Delta x, \Delta y)$ descriptions, some information about the *posture* of the hand is useful to describe motion. This posture description can be obtained doing comparisons between sides of the square region that segments the hand. If the rectangle's side parallel to the x -axis ($side_x$) of the image is longer than the side parallel to the y -axis ($side_y$), then $\Delta posture = (+)$. If $side_y$ is longer than $side_x$, then $\Delta posture = (-)$. If $side_x = side_y$, then $\Delta posture = (0)$. To estimate depth motion in a simple way,

we use the $\Delta area$ feature. If the hand increases its area we suppose its forward motion, and then $\Delta area = (+)$. If the hand decreases its area, we suppose its backward motion, and then $\Delta area = (-)$. If there is no change in the area, then $\Delta area = (0)$.

2.4 Visual gestures recognition using Dynamic Bayesian Networks

Dynamic Bayesian Networks (*DBN's*) are acyclic graphical models to represent temporal processes inside a stochastic framework [10]. Many events of the world can be represented by dynamic systems -e.g. car or human motion, speech, etc. [10, 11]. DBN's are generalizations of Hidden Markov Models (*HMM's*) and dynamic linear systems like Kalman Filter, representations less structured than Dynamic Bayesian Networks [12]. To represent a DBN, it is common to employ two assumptions: i) Markovian property, that establishes independence of the future respect to the past given the present, and ii) the process is stationary, *i.e.*, that probability transitions among states are all the same through the time [13].

Usually, a DBN is composed by a *base network* X^t defining the instantaneous state of the system at time t and a *transition network* R that connects some nodes of X^t to X^{t+1} for $t = 0 \dots T - 1$ [11, 12]. Figure 4 shows an example of a simple base net and its transition network for $t = 1, 2, 3$.

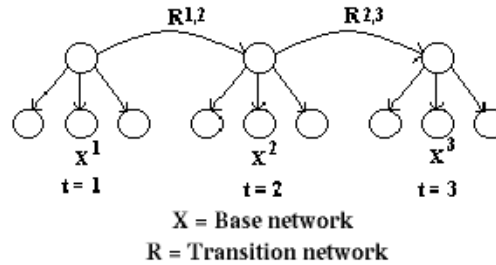


Fig. 4. Example of dynamic Bayesian network composed by a simple base network and its transition network for $t = 1, 2, 3$.

A DBN can contain hidden nodes to define the system state at each time t . However, just as Hidden Markov Models [14], that state is only accessible through the set of observation variables of the dynamic system. Hidden Markov Models uses a single observation node for each state. Because of this, it is necessary to represent explicitly at each observation node permutations of all possible values of the observation variables. With 4 variables and 3 possible values for each, we have $3^4 = 81$ the permutations or different observations attached at each hidden node [15]. With HMM's, the number of parameters needed to define them grows exponentially as we increment the number of states, observation variables and their possible values [14]. Dynamic Bayesian networks permit

more complex dependences among variables than HMM's. This makes possible to consider more than just a single observation node per state, decreasing the number or parameters needed to define the model.

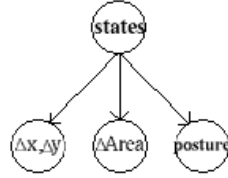


Fig. 5. Topology for the base network proposed in our system.

Figure 5 shows the Bayesian network proposed as base network for each gesture considered by our system. It considers 3 observation nodes that represent the variables $(\Delta x, \Delta y)$, $\Delta area$ and $posture$ described in the previous section. Δx and Δy are contained in a single node because with this topology we obtained better results than using one node per variable. The hidden node *states* represents states S_1 and S_2 of the DBN. With this models it is only necessary to specify 15 parameters per state, 9 permutations of possible values of Δx and Δy , plus 6 possible values of $\Delta area$ and $posture$. This is a reduction of 81.5% of the parameters needed to define a single HMM state. The transition net proposed for S_1 and S_2 is shown in figure 6.

To train our DBN's we utilized the Baum-Welch algorithm. To test the models we employed the Forward algorithm [14].

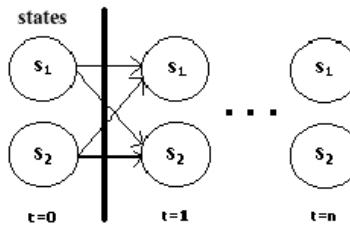


Fig. 6. States transition network.

3 Experiments and results

Initially we built the skin classification and segmentation modules. To construct the histogram color models employed to classify pixels as skin or non-skin, it was necessary to segment 2,101,612 skin pixels, and 19,552,655 non-skin pixels

Table 1. Gestures recognition rates using dynamic Bayesian networks. Lines present the percentage of correct classification in the execution of each gesture as well as the percentage of incorrect classification.

	Attention	Come	Go-right	Stop	Go-left
Attention	66.2%	12.41%		21.37%	
Come		100%			
Go-right		0.75%	99.25%		
Stop	26.66%	20.74%		52.59%	
Go-left					100%
Average recognition rate			=	84.01%	

by hand. The skin pixels were taken from 170 images of faces, arms and hands of 30 people at different lighting conditions, using 4 videocameras. To test the tracking and recognition stages, we supposed a laboratory environment. The image resolution employed was 640×480 pixels. The distance from the user to the camera ranges between 1.5m and 2.5m.

For training, we used an average of 596 sequences of observations for each gesture, taken from 11 people. Although training sequences must have a great amount of possible variations [9], it must be taken into consideration that whenever a gesture is made, this one must be executed with similar form, speed, force and amplitude so that it won't be confused with other gestures. In this manner, the ambiguity is reduced, and clarity in the message is obtained [1].

To test the recognition rate of our system, other user made an average of 146 executions for each gesture in front of the videocamera. The gestures we used to test the system were different from those used for training. In order to define the start and end of a gesture, a small tolerance region was established around the initial position of the hand. The return of the hand to its initial position defines when the gesture is completed.

Table 1 shows recognition rates and the average recognition rate of the system for each gesture using dynamic Bayesian networks. To compare this resultS with the standard recognition technique, Table 2 shows the recognition results using hidden Markov models with topologies of 3 and 5 states described in [15]. In this cases we used the same training and test sets of gestures. As it can be seen, although there is a slight improvement with DBN's, their use permit decrease the number of parameters needed to define the models respect to HMM's. At the same time, we obtain simpler and clearer representations than using hidden Markof models.

The two models obtained similar recognition results in go-left and go-right gestures. These results are easily explained considering the nature of the necessary movements in one or another case. Whereas the gestures go-left and go-right require displacements that predominate towards the left and the right (parallel to the image plane); the gestures come, attention and stop involve movements in depth (perpendicular to the image plane). So, gestures that involve depth motion have lower recognition rates, given that direct depth information is not

Table 2. Recognition results using hidden Markov models of 3 and 5 states.

	Attention	Come	Go-right	Stop	Go-left
Attention	90.35%	9.65%			
Come	2.79%	88.26%	7.26%	1.67%	
Go-right			100%		
Stop	60%	3.70%		36.29%	
Go-left					100%
Average recognition rate			=	83.05%	

considered. In both models, the main error in recognition appears between stop and attention gestures. Maybe, this is generated by the similarity of movements between these two gestures, which is related with the distance from the user to the camera. Although these are preliminary results, we suppose that it is possible to improve recognition rates of DBN's by finding better dependence relationships among variables.

4 Conclusions and future work

This document describes an online system to recognize 5 dynamic gestures making use of dynamic Bayesian networks. DBN's permit to define more complex dependencies among variables in the model. At the same time as we increase the number of variables or observation features to represent gestures, DBN's permit to maintain a clearer and simpler structure than HMM. Due to this simpler structure, the number of parameters needed to define the models is usually reduced.

As future work we plan increase the number of variables in the DBN, considering movement of the elbow and shoulder and in this way, test more complex DBN's structures. Also, we plan to implement structural learning to find the dependence relationships among variables, which would improve recognitions rates.

References

1. Morris, D.: El hombre al desnudo. Vol. 1. Ediciones Orbis. Barcelona, España (1977). (In Spanish).
2. Wasson, G., Kortenkamp, D., Huber, E.: Integrating Active Perception with an Autonomous Robot Architecture. *Agents*. (1998) 325–331
3. Kahn, R., E.: Perseus: An Extensible Vision System for Human-Machine Interaction. (PhD Thesis). The University of Chicago (1998).
4. Kortenkamp, D., Huber, E., Bonasso P.: Recognizing and interpreting gestures on a mobile robot. *Proceedings of the AAAI-96*. AAAI Press/The MIT Press. (1996). 915–921.
5. Waldherr, Stefan. *Gesture Recognition on a Mobile Robot*. Carnegie Mellon University. School of Computer Science. (1998).

6. Jones, M., J., Rehg, J.: Statistical Color Models with Application to Skin Detection. Cambridge Research Laboratory. CRL 98/11. (1996).
7. S. A. V. I. Group.: Available at: <http://www.cs.toronto.edu/~herpers/projects.html>. May 28. 1999
8. Avilés, H.: Reconocimiento de gestos dinámicos aplicado a robots móviles. Instituto Tecnológico y de Estudios Superiores de Monterrey. Campus Cuernavaca. (2000).
9. Starner, T., E.: Visual Recognition of American Sign Language Using Hidden Markov Models. MIT. Program in Media Arts and Science. (1995).
10. Forbes, F., Huang, T., Kanazawa, K., Russell, S.: The BATmobile: Towards a Bayesian Automated Taxi. Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence. (1995).
11. Zweig, G., Russell, S. Probabilistic Modeling with Bayesian Networks for Automatic Speech Recognition. Australian Journal of Intelligent Information Processing Systems. Vol. 5(4). (1999). 253–260.
12. Murphy K.: A Brief Introduction to Graphical Models and Bayesian Networks. Available at: <http://www.cs.berkeley.edu/~murphyk/Bayes/bayes.html>. March 3. 2001.
13. Boyen, X., Friedman, N., Koller, D.: Discovering the Hidden Structure of Complex Dynamic Systems. Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence. (1999).
14. Rabiner, L., R.: Readings in Speech Recognition. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Morgan Kaufmann Publishers. (1990).
15. Avilés, H., Sucar, E., Zárate, V.: Dynamical Arm Gestures Visual Recognition Using Hidden Markov Models. IBERAMIA/SBIA. Workshop on Probabilistic Reasoning in Artificial Intelligence. (2000).
16. Cassell, J.: Computer Vision in Human-Machine Interaction. A Framework For Gesture Generation And Interpretation. Morgan Kaufmann Publishers. (1998).
17. Martin, J. Durand, J-B.: Automatic Gestures Recognition Using Hidden Markov Models. Fourth IEEE International Conference on Automatic Face and Gesture Recognition. (2000).