A FAST ALGORITHM FOR VISION-BASED HAND GESTURE RECOGNITION FOR ROBOT CONTROL

Asanterabi Malima, Erol Özgür, and Müjdat Çetin

Faculty of Engineering and Natural Sciences,
Sabancı University, Tuzla, İstanbul, Turkey
malima@su.sabanciuniv.edu, erol@su.sabanciuniv.edu, and mcetin@sabanciuniv.edu

Abstract

We propose a fast algorithm for automatically recognizing a limited set of gestures from hand images for a robot control application. Hand gesture recognition is a challenging problem in its general form. We consider a fixed set of manual commands and a reasonably structured environment, and develop a simple, yet effective, procedure for gesture recognition. Our approach contains steps for segmenting the hand region, locating the fingers, and finally classifying the gesture. The algorithm is invariant to translation, rotation, and scale of the hand. We demonstrate the effectiveness of the technique on real imagery.

1. Introduction

Vision-based automatic hand gesture recognition has been a very active research topic in recent years with motivating applications such as human computer interaction (HCI), robot control, and sign language interpretation. The general problem is quite challenging due a number of issues including the complicated nature of static and dynamic hand gestures, complex backgrounds, and occlusions. Attacking the problem in its generality requires elaborate algorithms requiring intensive computer resources. What motivates us for this work is a robot navigation problem, in which we are interested in controlling a robot by hand pose signs given by a human. Due to real-time operational requirements, we are interested in a computationally efficient algorithm.

Early approaches to the hand gesture recognition problem in a robot control context involved the use of markers on the finger tips [1]. An associated algorithm is used to detect the presence and color of the markers, through which one can identify which fingers are active in the gesture. The inconvenience of placing markers on the user's hand makes this an infeasible approach in practice. Recent methods use more advanced computer vision techniques and do not require markers. Hand gesture recognition is performed through a curvature space method in [2], which involves finding the boundary contours of the hand. This is a robust approach that is scale, translation and rotation invariant on the hand pose, yet it is computationally demanding. In [3], a vision-based hand pose recognition technique using skeleton images is proposed, in which a multi-system camera is used to pick the center of gravity of the hand and points with farthest distances from the center, providing the locations of the finger tips, which are then used to obtain a skeleton image, and finally for gesture recognition. A technique for gesture recognition for sign language interpretation has been proposed in [4]. Other computer vision tools used for 2D and 3D hand gesture recognition include specialized mappings architecture [5], principal component analysis [6], Fourier descriptors, neural networks, orientation histograms [7], and particle filters [8].

Our focus is the recognition of a fixed set of manual commands by a robot, in a reasonably structured environment in real time. Therefore the speed, hence simplicity of the algorithm is important. We develop and implement such a procedure in this work. Our approach involves segmenting the hand based on skin color statistics, as well as size constraints. We then find the center of gravity (COG) of the hand region as well the farthest point from the COG. Based on these pre-processing steps, we derive a signal that carries information on the activity of the fingers in the sign. Finally we identify the sign based on that signal. Our algorithm is invariant to rotations, translations and scale of the hand. Furthermore, the technique does not require the storage of a hand gesture database in the robot's memory. We demonstrate the effectiveness of our approach on real images of hand gestures.

2. Hand Gesture Recognition

Consider a robot navigation problem, in which a robot responds to the hand pose signs given by a human, visually observed by the robot through a camera. We are interested in an algorithm that enables the robot to identify a hand pose sign in the input image, as one of five possible commands (or counts). The identified command will then be used as a control input for the robot to perform a certain action or execute a certain task. For examples of the signs to be used in our algorithm, see Figure 1. The signs could be associated with various meanings depending on the function of the robot. For example, a "one" count could mean "move forward", a "five" count could mean "stop". Furthermore, "two", "three", and "four" counts could be interpreted as "reverse", "turn right," and "turn left."

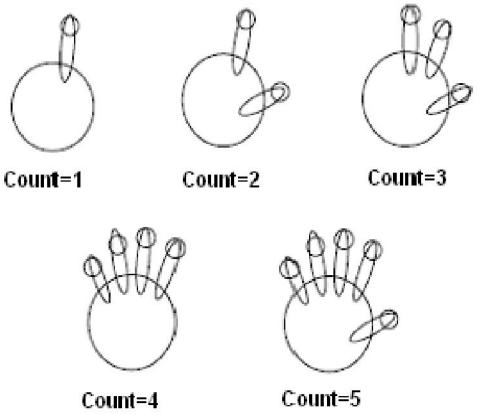

Figure 1: Set of hand gestures, or "counts" considered in our work. Adapted from [8].

Our proposed method of hand gesture recognition consists of the following stages:

- Localizing hand-like regions based on learned skin color statistics [9], producing a BW image output.
- Performing region-based segmentation of the hand, eliminating small false-alarm regions that were declared as "hand-like," based on their color statistics.
- Calculating the center of gravity (COG) of the hand region as well as the farthest distance in the hand region from the COG.
- Constructing a circle centered at the COG that intersects all the fingers that are active in the count.
- Extracting a 1D binary signal by following the circle, and classifying the hand gesture based on the number of active regions (fingers) in the 1D signal.

In the following subsections we describe each of the steps mentioned above.

2.1. Localizing Hand-like Regions by Skin Detection

We assume that the portion of the scene around the hand has already been extracted. Then our first task is to segment out the hand in the image from the background. We achieve that goal in two steps. First, we find the pixels in the scene that are likely to belong to the hand region, which we describe in this section. Then we refine that result, as we describe in the next section.

It has been observed that the red/green (R/G) ratio is a discriminative characteristic feature for human skin color [9]. Our statistical observations also support this claim. In particular, in Figure 2, we show three images we have acquired, each containing a hand gesture, together with scatter plots of the red versus green components of the pixel intensities for skin and non-skin regions in the images. We observe that the R/G ratio stays within a narrow band of values for skin pixels, whereas it is much more variable for non-skin pixels . Therefore, we could use this ratio to decide whether a pixel is likely to belong to the hand region or not. In particular, we empirically observe that the following two thresholds successfully capture hand-like intensities:

$$1.05 < R/G < 4.00 \qquad (1)$$

Using this thresholding scheme, we set all the pixels with color intensities within the thresholds to one, and all the rest to zero; resulting in a black and white image output. Of course, this simple scheme could produce many erroneous decisions, for example many background pixels having skin-like colors could be classified as "hand-like." We refine this output in the next section.

2.2. Segmentation and False-Region Elimination

The scheme described in the previous section could produce many disconnected regions in the image classified as hand-like. We use ideas from region-based segmentation [10,11] to alleviate this problem. Our assumption is that the largest connected white region corresponds to the hand. So we use a relative region size threshold to eliminate the undesired regions. In particular, we remove the regions that contain smaller number of pixels than a threshold value. The threshold value is chosen as 20% of total number of pixels in the white parts. Note that this is an image-size invariant scheme. The ideal outcome is the segmented hand region.

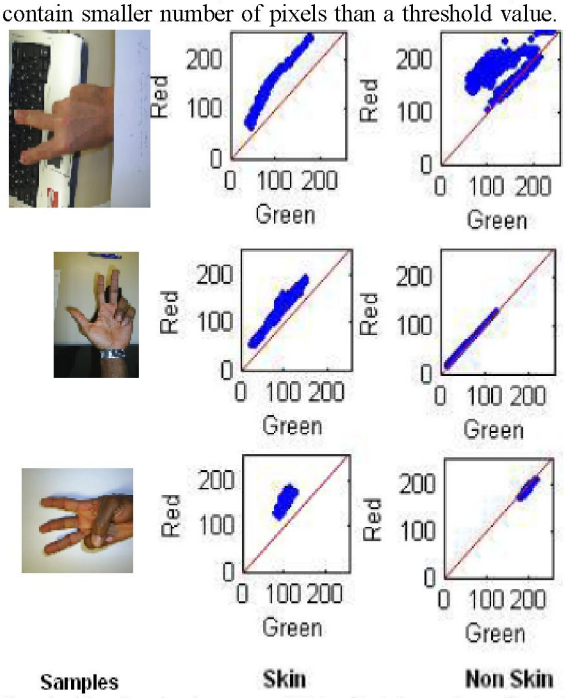

Figure 2: Scatter plots of the red versus green components of pixel intensities of skin and non-skin regions for three sample images. The line in the scatter plots indicates a slope of 1.

2.3. Finding the Centroid and Farthest Distance

Given the segmented hand region, we calculate its centroid, or center of gravity (COG), $(\bar{x}, \bar{y})$, as follows:

$$\bar{x} = \frac{\sum_{i=0}^{k} x_i}{k} \quad and \quad \bar{y} = \frac{\sum_{i=0}^{k} y_i}{k} \qquad (2)$$

where x_i and y_i are x and y coordinates of the i^{th} pixel in the hand region, and k denotes the number of pixels in the region.

After we obtain the COG, we calculate the distance from the most extreme point in the hand to the center; normally this farthest distance is the distance from the centroid to tip of the longest active finger in the particular gesture (see Figure 3).

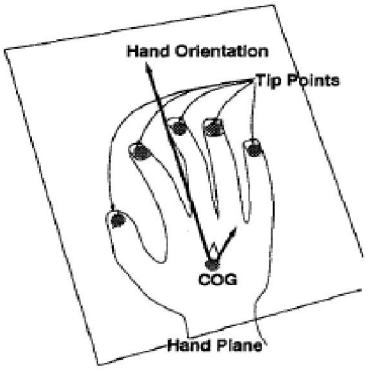

Figure 3: Center of gravity and extreme points of the hand. Taken from [3].

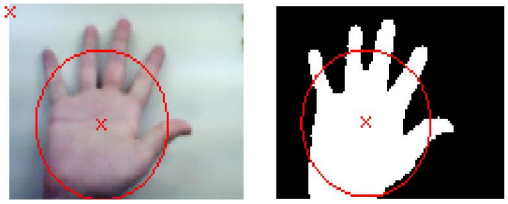

Figure 4: Demonstration of the execution of the processing steps described in Sections 2.1-2.4 on a sample image.

2.4. Constructing a Circle

We draw a circle whose radius is 0.7 of the farthest distance from the COG. Such a circle is likely to intersect all the fingers active in a particular gesture or "count." Just to provide a visual flavor, Figure 4 demonstrates the execution of the steps described so far on a sample image.

2.5. Extracting a 1D Signal and Classification

We now extract a 1D binary signal by tracking the circle constructed in the previous step. Ideally the uninterrupted "white" portions of this signal correspond to the fingers or the wrist. By counting the number of zero-to-one (black-to-white) transitions in this 1D signal, and subtracting one (for the wrist) leads to the estimated number of fingers active in the gesture. Estimating the number of fingers leads to the recognition of the gesture.

Note that our algorithm just counts the number of active fingers without regard to which particular fingers are active. For example, Figure 5 shows three different ways in which our algorithm would recognize a three count; rotation, orientation, or any other combination of three fingers would also give the same result. So an operator does not have to remember which three fingers he/she needs to use to express the "three count." While this feature may be preferable in some tasks, in other tasks one might be interested in associating different meanings to different finger combinations. We could modify and adapt our algorithm to such a setting by a number of modifications. For example, the analysis of the 1D signal described in this section would need to pay attention to the distances between the active fingers, as well as between the fingers and the wrist.

Figure 5: Different forms of three count.

2.6. Scale, Rotation, and Translation Invariance

Our proposed algorithm is scale invariant. Meaning that the actual size of the hand size and its distance from the camera do not affect interpretation. It is rotation invariant, since the orientation of the hand does not hinder the algorithm from recognizing the gesture. In addition, the position of hand is also not a problem.

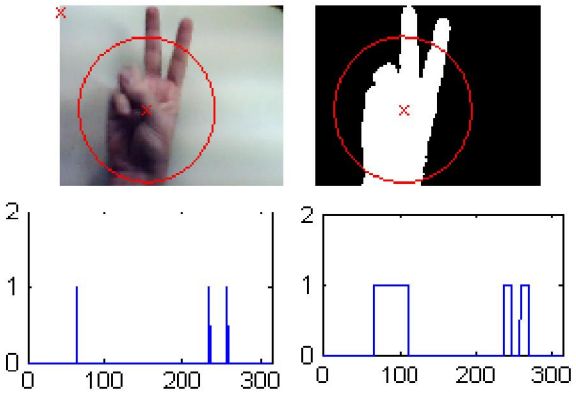

Figure 6: Sample result for a "two" count. Top left: Original image and the circle constructed. Top right: Segmented hand and the circle constructed. Bottom left: Zero-to-one transitions in the 1D signal extracted. Bottom right: The 1D signal itself.

3. Experimental Results

We have conducted experiments based on images we have acquired using a 4 Mega-Pixel digital camera as well as a simple webcam. We have collected these data on uniform as well as cluttered backgrounds. Figure 6 shows a sample result for a hand image displaying the count "two". We show the output of various stages of our algorithm. Note that through differencing operations, we obtain the zero-to-one transitions in the 1D signal whose extraction was described in Section 2.5, and is illustrated in Figure 6 as well. The number of these transitions minus one (for the wrist) produces the estimated count. Figure 7 shows comparable results for a hand image displaying the count "four." In both examples, our algorithm leads to the correct recognition of the gesture. Also, the computation time needed to obtain these results is very small, since the algorithm is quite simple.

Out of 105 samples taken from 21 members of our laboratory, we have obtained 96 correct classifications which is approximately 91% of all images used in our experiments. We have noted that images taken under insufficient light

(especially using the webcam) have led to the incorrect results. In these cases the failure mainly stems from the erroneous segmentation of some background portions as the hand region. Our algorithm appears to perform well with somewhat complicated backgrounds, as long as there are not too many pixels in the background with skin-like colors. Overall, we find the performance of this simple algorithm quite satisfactory in the context of our motivating robot control application.

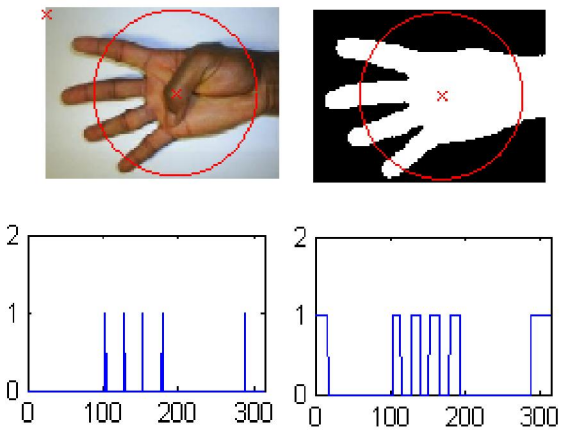

Figure 7: Same as Figure 6, except that the count here is "four".

4. Conclusion

We proposed a fast and simple algorithm for a hand gesture recognition problem. Given observed images of the hand, the algorithm segments the hand region, and then makes an inference on the activity of the fingers involved in the gesture. We have demonstrated the effectiveness of this computationally efficient algorithm on real images we have acquired.

Based on our motivating robot control application, we have only considered a limited number of gestures. Our algorithm can be extended in a number of ways to recognize a broader set of gestures. The segmentation portion of our algorithm is too simple, and would need to be improved if this technique would need to be used in challenging operating conditions. However we should note that the segmentation problem in a general setting is an open research problem itself. Reliable performance of hand gesture recognition techniques in a general setting require dealing with occlusions, temporal tracking for recognizing dynamic gestures, as well as 3D modeling of the hand, which are still mostly beyond the current state of the art.

5. Acknowledgment

We would like to thank all the researchers working on this field who in one way or another guided us on achieving our goals. We would also like to express our appreciation to all other researchers at Sabancı University who were kind enough to share their views with us and offered some suggestions in making this project a success.

This work was partially supported by the European Commission under Grant FP6-2004-ACC-SSA-2 (SPICE).

6. References

[1] J. Davis and M. Shah "Visual Gesture Recognition", *IEE Proc.-Vis. Image Signal Process., Vol. 141, No.2*, April 1994.

[2] C.-C. Chang, I.-Y Chen, and Y.-S. Huang, "Hand Pose Recognition Using Curvature Scale Space", *IEEE International Conference on Pattern Recognition*, 2002.

[3] A. Utsumi, T. Miyasato and F. Kishino, "Multi-Camera Hand Pose Recognition System Using Skeleton Image", *IEEE International Workshop on Robot and Human Communication*, pp. 219-224, 1995.

[4] Y. Aoki, S. Tanahashi, and J. Xu, "Sign Language Image Processing for Intelligent Communication by Communication Satellite", *IEEE International Conf. On Acoustics, Speech, and Signal Processing*, 1994.

[5] R. Rosales, V. Athitsos, L. Sigal, and S. Sclaroff, "3D Hand Pose Reconstruction Using Specialized Mappings", *IEEE International Con. on Computer Vision*, pp. 378-385, 2001.

[6] C. Tomasi, S. Petrov, and A. Sastry, "3D = Classification + Interpolation", *IEEE International Conf. on Computer Vision*, 2003.

[7] W. T. Freeman and M. Roth, "Orientation Histograms for Hand Gesture Recognition", *IEEE International Conf. on Automatic Face and Gesture Recognition*, 1995.

[8] L. Bretzner, I. Laptev, and T. Lindberg, "Hand Gesture Recognition using Multi-Scale Color Features, Hierarchical Models and Particle Filtering", *IEEE International Conf. on Automatic Face and Gesture Recognition*, 2002.

[9] J. Brand and J. Mason, "A Comparative Assessment of Three Approaches to Pixel-level Human Skin Detection", *IEEE International Conference on Pattern Recognition*, 2000.

[10] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice-Hall, 2nd edition, 2002.

[11] R. M. Haralick, and L. G. Shapiro, *Computer and Robot Vision*, Volume I, Addison-Wesley, 1992, pp. 28-48.