Recognition of ASL using Hand Gestures

Prof. Sayyad T.J.

Karmaveer Bhaurao Patil College of Engineering & Polytechnic, Satara.

Abstract: Gesture recognition is an area of active current research in computer vision. It brings visions of more accessible computer system. Gesture is often used to mean an initiation or conclusion of some human interaction. Hand gestures can be classified in two categories: static and dynamic. A static hand is a particular hand configuration and pose, represented by a single image. A dynamic gesture is a moving gesture, represented by a sequence of images. In this paper single-handed gestures, are considered which are of distinct hand shapes and hand region. The intention is to achieve correct perception of the hand gesture performed by the user in front of a camera with a uniform non-skin color background (darker than the skin color and not in the shades of red color). A hand gesture recognition system is introduced to recognize static hand gestures which are the subset of ASL (American Sign Language). The system is designed to recognize the alphabets from A to Z in terms of sign language i.e. hand gestures which will be taken from a video scene captured by camera. The system is designed for 26 gestures of alphabets which are used in American Sign Language.

Keywords: ASL, Hand Gestures, Neural network, Gesture Recognition.

1. INTRODUCTION

It is observed that using hand for performing gesture requires the user to wear unnatural device that cannot easily be ignored and which often requires significant effort to put on and calibrate. Even optical systems with markers applied to the body suffer from these shortcomings. Also there are some problems related to hand tracking in successive frames. Some trackers have been developed which use methods such as tracking contours with snakes, using Eigenspace matching techniques, maintaining large sets of statistical hypotheses, or convolving images with feature detectors but all these are computationally too expensive. Color-based tracking are simpler algorithms but slower at any given CPU speed due to their use of color correlation, blob and region growing and contour considerations. We want a tracker that will track a hand in successive frames of a video sequence with minimal computational cost and with reduced complexity in design.

The system consists of three modules:

1. Skin color detection: For detecting the skin colored pixels in the image the technique of Explicitly defined skin region is used. This technique provides effective results in extracting skin colored objects from the image with dark non-skin colored background.

2. Hand Tracking: After skin colored object extraction separation of region of interest i.e hand region in the images is separated by performing morphological operations on the images. In each frame the location of hand is found out. Then CAMSHIFT algorithm is implemented so as to track the region of interest i.e. hands in the successive frames of the captured video scene. In CAMSHIFT algorithm tracking is done by placing a bounding box or search window around the hand region in the successive images.

3. Hand gesture Recognition: In this module implementation of a neural network for hand gesture recognition is done. A single frame is selected with the completely tracked gesture and send as a input to the neural network. A standard Perceptron neural network is constructed. The network has an input layer and a Perceptron layer. The inputs are connected to all nodes in the Perceptron layer. Hexagonal arrangement of neurons is followed in the network. Gesture is then classified on the basis of comparison of feature vectors. The feature vector chosen for the comparison of the images is orientation histogram which makes the system robust against the different illumination values of the image.

2. ASL (AMERICAN SIGN LANGUAGE)

ASL is a complete, complex language that employs signs made with the hands and other movements, including facial expressions and postures of the body. It is the first language of many deaf North Americans, and one of several communication options available to deaf people. ASL is said to be the fourth most commonly used language. American Sign Language is a unique system of communication because it is both a visual language and manual language. Instead of expressing himself through sound, a speaker using ASL employs a combination of facial expressions, body language, gestures, palm orientations, and hand shapes. Learning the subtleties of communicating in this manner can often take years of intensive study. In American Sign Language, signs can be classified as transparent, translucent, or opaque. Signs that are transparent have meanings that are easily understood even by people who have not mastered the basics of the language. Translucent signs are understood by non-proficient speakers once the meaning has been explained. In comparison, a sign that is classified as opaque has a meaning that is not often guessed by someone who is not fluent in American Sign Language. Most of the signs needed to communicate clearly using American Sign Language are classified as opaque. In spoken language, the different sounds created by words and tones of voice (intonation) are the most important devices used to communicate. Sign language is based on the idea that sight is the most useful tool a deaf person has to communicate and receive information. Thus, ASL uses hand shape, position, and movement; body
movements; gestures; facial expressions; and other visual cues to form its words. Like any other language, fluency in ASL happens only after a long period of study and practice. ASL consists of approximately 6000 gestures of common words with finger spelling used to communicate obscure words or proper nouns. Finger spelling uses one hand and 26 gestures to communicate the 26 letters of the alphabet.

Figure 2.1: ASL signs

3 SYSTEM ARCHITECTURE

3.1 System Overview

The video sequence is captured by a camera. After capturing the scene it is then converted into no of frames where then each frame is stored in bitmapped image format for further processing.

Figure 3.1: Conversion of video into images

Next block represents the separation of skin colored objects from the images which also eliminates background in the image. This step is necessary for finding the location of hand in the image. The use of shape cue is alternatively possible for locating the hand, but they vary greatly during natural hand motion. The skin color is distinctive cue of hand and face and it is invariant to scale and rotation. Therefore, the technique of Explicitly defined skin region is chosen for the purpose of skin colored object separation from the background. The output of this block are the images with presence of only skin colored objects. The final goal of skin color detection is to build a decision rule, that will discriminate between skin and non-skin pixels. This is accomplished by introducing a metric, which measures distance (in general sense) of the pixel color to skin tone. The type of this metric is defined by the skin color modelling method.

3.1.1 Explicitly defined skin region

The method to build a skin classifier is to define explicitly (through a number of rules) the boundaries skin cluster in some colorspace. For example: (R,G,B) is classified as skin if:
- \( R > 95 \) and \( G > 40 \) and \( B > 20 \) and \( \max(R,G,B) - \min(R,G,B) > 15 \) and \( |R-G| > 15 \) and \( R > G \) and \( R > B \)

The obvious advantage of this method is simplicity of skin detection rules that leads to construction of a very rapid classifier. With the proper selection of values the technique works satisfactorily to separate out skin color pixels from the colors which are darker than skin color and not in the shades of red color because the hue of red color is most closest to hue of flesh color.

In the next block Image segmentation for the purpose of separating out the region of interest i.e. hand from other skin colored objects in the image is done. This is achieved by performing morphological operations like dilation, edge detecting, filling holes in the image and finally finding out the connected regions in the image.

Figure 3.2: Skin color detection

3.1.2 CAMSHIFT Derivation

The closest existing algorithm to CAMSHIFT is known as the mean shift algorithm. CAMSHIFT (Continuously Adaptive MEANSHIFT) is a variation of the
MEANSHIFT algorithm. The MEANSHIFT algorithm is an algorithm, which simply climbs the peak of a probability density function given a specific constraint window. This window is called the search window. The MEANSHIFT works well for analysis of static distribution through time. However, it fails when the distributions change. CAMSHIFT picks up where the MEANSHIFT fails. CAMSHIFT adaptively modifies the search window each of frame until a steady state results and adjusts the window for the next frame. By doing so, CAMSHIFT is able to track continuity of an object through time.

For discrete 2D image probability distributions, the mean location (the centroid) within the search window (Steps 3 and 4 in Mean Shift Algorithm [3] ) is found as follows:

\[ M_{00} = \sum x \sum y I(x,y) \]
\[ M_{01} = \sum x \sum y xI(x,y) \]
\[ M_{10} = \sum x \sum y yI(x,y) \]

Then the mean search window location (the centroid) is

\[ x = \frac{M_{10}}{M_{00}}, \quad y = \frac{M_{01}}{M_{00}} \]

where I(x,y) is the pixel (intensity) value at position (x,y) in the image, and x and y range over the search window. CAMSHIFT relies on the zeroth moment information, extracted as part of the internal workings of the algorithm, to continuously adapt its window size within or over each video frame. One can think of the zeroth moment as the distribution "area" found under the search window. Thus, window radius, or height and width, is set to a function of the the zeroth moment found during search. The CAMSHIFT algorithm is then calculated using any initial nonzero window size.

Continuously Adaptive Mean Shift Algorithm (CAMSHIFT):
1. Choose the initial location of the search window.
2. Mean Shift as above (one or many iterations); store the zeroth moment.
3. Set the search window size equal to a function of the zeroth moment found in Step 2.
4. Repeat Steps 2 and 3 until convergence (mean location moves less than a preset threshold).

3.1.3 Orientation Histogram
Next block is of feature extraction i.e the output of this block is a feature vector which will be fed to the next block for gesture classification. The feature vector chosen for extraction is the orientation histogram of the image. This feature is chosen because of the advantage that the object classification can be done irrespective of its location in the image. Also the illumination changes in the images can cause the other features to wrongly classify the object in the image. Also the hand’s pixel values vary considerably with lighting, while the orientation values remain fairly System Architecture constant. We can calculate the local orientation most simply from the direction of the image gradient. Then the local orientation angle, \( \theta \), as a function of position x and y, and image intensities I(x, y), is:

\[ \theta(x, y) = \arctan \left[ \frac{I(x, y) - I(x-1, y)}{I(x, y) - I(x, y-1)} \right] \]

We want gestures to be the same regardless of where they occur within the camera’s field of view. This translation invariance is achieved by the drastic step of ignoring position altogether, simply tabulating a histogram of how often each orientation direction occurred in the image.

3.1.4 Neural Network
Next block is of neural network used in the system for gesture perception. A standard neural network is constructed for the system. The architecture resembles the Perceptron Network. The numbers are chosen as per the requirement in the system implementation. The output is obtained by training the network with Cyclic order incremental training method. A hard limit function is used as a transfer function for getting the output. The network is also trained to adapt the changes in weights and biases for getting the necessary output from the
Supervised training is provided to the network where the network is provided with a series of sample inputs and comparing the output with the expected response. The training continues until the network is able to provide the expected response. In a neural net, for a sequence of training input vectors there exist target output vectors. These target output vectors represent a recognized gesture.

4 EXPERIMENTS AND RESULTS
Separation of Skin Colored Objects using Explicitly defined Skin region. Thus the skin colored objects i.e the hand and face are separated from background. The background pixels are set to value 255 (white color). Edges are detected in the images to find the connected regions in the images. As mentioned in testing Sobel operator is used to perform edge detection. Dilation is carried on the images to grow the thickness of the regions. Structuring element used is 'line' and holes in the image are filled.

Sorting out region of interest from different objects in the images. Sorting is based on region properties.

Figure 4.1: Edge detected images
Figure 4.2: Dilated images
Figure 4.3: Filled images
Figure 4.4: Sorted Hand region

Figure 4.5: Training of Neural Network
Figure 4.6: End of training
Figure 4.7: Plot of Errors Vs Epochs
Figure 4.8: Selected frames with two different gestures
Figure 4.9: Recognition of two different gestures

5 CONCLUSION
Thus a system is developed to recognize 26 hand gestures which represent the alphabets from A to Z in ASL (American sign Language). Technique of hand tracking i.e. implementation of CAMSHIFT avoids the use of any special hand tracking software in the gesture recognition system. The feature vector considered for gesture classification is orientation histogram of the image which is robust against the change in color of the image for different skin colors. Gesture perception is achieved through a simple neural network which perceives the gesture within short time.

References