Imperial Journal of Interdisciplinary Research (IJIR)
Vol-3, Issue-9, 2017
ISSN: 2454-1362, http://www.onlinejournal.in

Hand Gesture Recognition using Sign Language

Tharadevi. R.V
[1]PG Student, Department of MCA, CHMM College for Advanced Studies, Kerala, India

***Abstract**: Sign Language is mostly used by deaf and dumb people. In order to improve the man machine interaction, sign language can be used as a way for communicating with machines. Most of the applications which enable sign language processing are using data gloves and other devices for interacting with computers. This restricts the freedom of users. So to avoid this, this system we capture live video stream through normal webcam and pass it as input and process this video to detect human gestures . These gestures can be signs represented in any sign language like American Sign Language (ASL), Indian Sign Language(ISL) etc. By processing these signs, this system can make an efficient system for searching using signs. In future this method of searching can be implemented in many areas such as railway station for searching the details of any trains, etc. Searching uses signs can play a major role in searching technique because the latest existing technique of searching is based on voice. But this may not work properly because each person has a different pronunciation. This system can overcome the drawbacks of the current available searching technique and thus can make every person under an umbrella of searching.*

1. Introduction

Human recognition has continued to be an active area of research and has thus rightfully attracted much attention from the researchers over the years. In this system we capture the live video stream and pass it as input and process it to detect humans and gestures that occur in the video. The two main processes done in this project are Gesture recognition using Contour-based Object Detection and Skin recognition algorithm. Sign language recognition using finger spelling helps human machine interaction (HMI) helps humans to interact with machines. A gesture is defined as a movement that is usually made by hand. Sign language can be used to communicate with the machine by inputting the finger spelling assigned to each alphabet. In American Sign Language (ASL) each alphabet, A-Z, is assigned a unique finger spelling. Sign language is mostly used by people with disabilities. Our goal is to design a Human Computer Interface(HCI) system that can understand the sign language accurately so that the people with disabilities may communicate with the other people without the need of any other device. In our project, the image of the hand is captured using a simple web camera. The acquired image is then processed and some features are extracted. These features are then used as input to a classification algorithm for recognition. The recognized finger spelling may then be used to generate speech. In our project we are using ASL (American Sign Language). In ASL each alphabet of English vocabulary, A-Z, is assigned a unique finger spelling.

2. Sign Language Recognition

Sign languages are mostly used by deaf and dumb people for communication. Signs are movements made by hands for communication. So this sign can be used for many other purposes like searching. The existing technology only focuses on searching data based on voice. But this technology can't work properly because every person has different pronunciation of different words. This may limit the working of these systems.

There are may sign languages like Portuguese Sign Language(PSL),American Sign Language(ASL) etc. This paper is based on ASL. In ASL each alphabet of English vocabulary, A-Z, is assigned a unique finger spelling.

The scope of this project is to create a method to recognize finger spelling, based on a pattern recognition technique, employing histograms of local orientation The orientation histogram will be used as a feature vector for finger spelling classification. In practice, it involves the construction of human detectors, where the detectors search given images of human and localize them.

A human detector can be considered as a combination of two key factors: a feature extraction algorithm and a detector that is used to make human/non-human decisions. This thesis targets general purpose human detectors that do not make strong contextual assumptions.

Many applications, such as recognizing particular static hand signal need a richer description of the shape of the input object than image moments provided. If the hand signals fell in a predetermined set, and the camera views a close-up of the hand, we use an example-based approach, combined with a simple method top analyzes hand signals called orientation histograms. These examples-based applications involve two phases; training phase and running phase. In the training phase, the user shows the system one or more examples of a specific hand shape. The computer creates and saves the corresponding orientation histograms. In run phase, the computer compares the orientation histogram of the current image and stored image templates. Then selects the category of the closest match as appropriate. This method should be robust against small differences in hand size but sensitive to changes in hand orientation. Face monitor module communicate with the camera/Web cam connected and captures the image in accordance with user input to system. It uses the dynamic linked library named "avicap32.dll" and uses Windows messaging service to communicate to the driver of image capturing device to view and capture images.

When somebody gets in front of the webcam and makes some hands gestures like signs using finger spellings in front of the camera, the application should detect the type of the sign and when a hand gesture is detected, the system will raise an event and the application could perform different actions depending on the type of the sign. That is, each alphabet may be assigned a unique gesture. In this system, the image of the hand is captured using a simple web camera. The acquired image is then processed and some features are extracted. These features are then used as input to a classification algorithm for recognition. The recognized sign is outputted as text and voice. Few attempts have been made in the past to recognize the signs made using finger spelling method, but with limitations on recognition rate and time.

3. Architecture

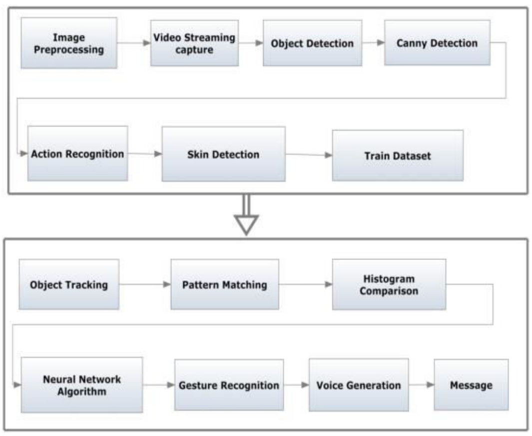

Figure 1: Architectural Design

3.1 Segmentation

Identifying and finding sharp discontinuities in an image is called edge detection. The purpose of edge detection in general is to significantly reduce the amount of data in an image, while preserving the structural properties to be used for further image processing. Edge detection involve convolving the image with an operator (a 2-D filter), which is constructed to be sensitive to large gradients in the image which returns value zero in uniform regions. There are various ways to perform edge detection. This work will compare two such techniques for detecting motion which are Canny edge detection.

The algorithm runs in five separate steps:

1. Smoothing:
   Blurring the image to remove noise.
2. Finding gradients:
   The edges are marked where the gradients of the image have large magnitudes.
3. Non-maximum suppression:
   Only local maxima are marked as edges.
4. Double thresholding:
   Potential edges are determined by thresholding strong and weak edges.
5. Edge tracking by hysteresis:
   Suppressing all the edges that do not connect to a very certain (strong) edge and hence determine final edges.

3.2 Edge Detection after Image Subtraction

In this algorithm, edge detection is performed after image subtraction. The algorithm is as follows:

1. Extract frames from the video stream.
2. Write the extracted frames as image files.
3. Subtract the previous image from the current Image.
4. Convert the image to grayscale.
5. Detect edges.
6. Label connected components.
7. Perform blob analysis.
8. Calculate the center of mass of each labeled region and label it.
9. Play the labeled images as a continuous video Stream to detect motion.

3.3 Motion Detection

Step1:

Identify objects that have moved between the two frames (using difference and threshold filter). The difference between each corresponding pixel of two frames is calculated and the pixels with difference greater than the specified threshold are marked in foreground color (white) and remaining pixels are marked as background color (black). So, the output will be a binary-image with only two colors (black and white). The intensity value (brightness level) of the pixels is used to calculate the difference. The intensity value of each pixel of a grayscale image will be in the range 0 to 255. If RGB images are used, then the grayscale intensity value can be calculated as (R + G + B / 3).

Step2 :
Filters out the noise that are wrongly identified as motions (using an erosion - filter) and identifies the significant movements. Erosion-filter filters pixels by removing the pixels that are not surrounded by enough amount of neighboring pixels. This gives the ability to shrink the objects, thereby removing the noisy pixels i.e. stand-alone pixels. The binary image (output of Step1) is scanned pixel-by-pixel. If enough number of pixels in current window are not turned on, then the entire window is turned off i.e. ignored as noise.

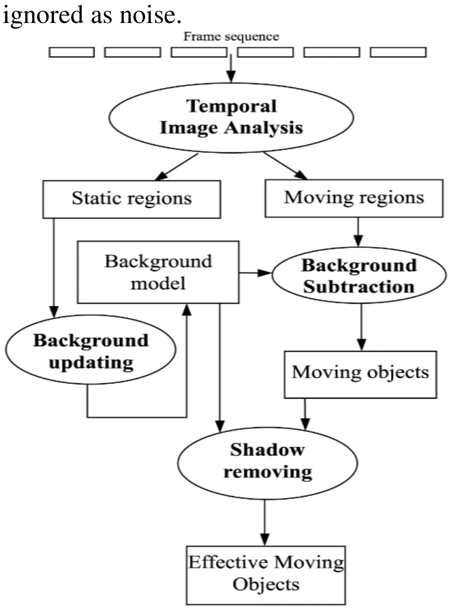

Figure 2: Motion Detection

3.4 Gesture Recognition (Contour)

Hand gestures, i.e., gestures performed by hand. The process by which gestures formed by a user are made known to the system is called Gesture Recognition. A pattern recognition system will be using a transform that converts an image into a feature vector, which will then be compared with the feature vectors of a training set of gestures. The final system will be implemented with a neural network.

1. Image creation:
   The images are taken by the camera. The camera is interfaced to the computer.
2. Preprocessing:
   It is used to filter out noise from the image. The image captured by the camera contains noise like impulse noise & Salt-pepper type noise. Salt –pepper noise is one of the types of the saturated impulse noise. This type of noise is removed by using a median filter.
3. Feature Extraction:
   In this step the required features are extracted from the image. The following steps are used:

i. Subtraction:
In the background subtraction method, the image of the background is captured as a reference image. The plain background image is subtracted from the image having a hand with a background that is the Target image. The resulting image is only hand. This is the simple & the easiest method of extraction.

ii. Segmentation:
Thresholding is used for clarity. Thresholding by Hysteresis method is used. The hysteresis will give the information regarding which point is to select as the threshold. Also, it will convert the image into the binary form. So that it will be easy for further processing.

iii. Thinning:
Thinning is an operation that is used to remove selected foreground pixels from binary images. In this object is converted into the set of digital arcs. These arcs are lying roughly along the medial axis. That is it will give reduced amount of the data, reducing the time required for processing. The border pixel having more than one neighbor is removed, converting into the thin line. It can be used for several applications, but is particularly useful for skeletonization. This mode is commonly used to clearly output the edge detectors by reducing all lines to single pixel thickness. Thinning is only applied to binary images, and produces another binary image as output.

3.5 Pattern Recognition

1. Creating Training sets:
   The pattern recognition block consists of creating the training sets of the images. Here the input & its expected output are known & according to that the Neural

network is Trained, which calculates the weight & bias This creates the Training sets. Inputs that are used to create the Training sets are given at the input layer. To the feed forward network the output of the input node connects to each of the inputs of hidden layer node. Each input has a weight which is updated according to the error generated by nodes associated with a mismatch between the actual output and the desired output. When there is perfect match the error is zero and the weights are fixed. In this way, the network is trained for a particular input and output combination.

$E= \Sigma(Y-D)^2$

Y=Actual Output

D=Desired Output

$W_{new}= W_{old} - \varepsilon\, (\delta E/\delta W)$

ε =Learning Constant

The images to be tested are given as input to the trained network. Depending on the comparison, we are getting output for a given image. The output depends on training system. i.e. how we train the system.

2. Testing:
   For testing, the unknown image is given as the input to network, which is then compared with the training set. Depending upon the match it gives the output.

3.6 Skin Detector

Step 1: Classify the pixels of the color image into two classes (skin / non-skin) using a skin detector, the result is a binary image.

Step 2: Segmenting the binary image, the result of the previous step, in connected and consistent regions using the well known segmentation technique: the line of the watershed.

Step 3: Elimination of non-significant regions.

Step 4: Location of significant skin areas.

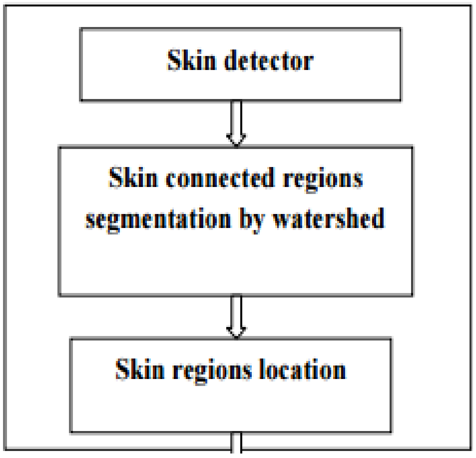

Figure 3: Skin Detection

The choice of a skin color model for a suitable color segmentation is always complex. In face detection, the detection of skin has been just a preliminary step and should be faster while having the best performance in terms of the true positive rate (TP: the probability that a pixel belongs to the skin class and is assigned to the skin class) and false positive rate (FP: the probability that a pixel does not belong to the skin class and is assigned to the skin class). For this reason, the choice was focused on the skin model with fixed thresholds which produces a very fast classification due to the simplicity of the decision rules used to discriminate the pixels of skin from those of non-skin.

4. Conclusion

The system discussed in this paper enables efficient Human-Machine Interaction in as far as it does not involve any use for extra tools. It is totally based on the user's body organ (the hand). Furthermore, such a dynamic system may be integrated as a part of other image processing systems to support more applications.

5. References

[1] Vajjarapu Lavanya, Akulapravin M.S, Madhan Mohan," Hand Gesture Recognition And Voice Conversion System Using Sign Language Transcription System", IJECT Vol. 5, Issue 4, Oct - Dec 2015.

[2] Panwar.M., "Hand Gesture Recognition System based on Shape parameters", In Proc. International Conference, Feb2012.

[3] Christopher Lee, Yangsheng Xu."Online, interactive learning of gestures for human robot interfaces", Carnegie Mellon University, the Robotics Institute, Pittsburgh, Pennsylvania, USA, 1996.

[4] Meenakshi Panwar, Pawan Singh Mehra, "Hand Gesture Recognition for Human Computer Interaction", In Proc. IEEE International Conference on Image Information Processing, Waknaghat, India, November 2011