International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

MSL Hand Gesture System

[1]Priyanka Tandale

PG student, SCOE, Pune-41, Maharashtra, India

Abstract: ***Gesture recognition is an important for developing alternative human-computer interaction. It enables human to interface with machine in a more natural way. A hand gesture consists of two types, static and dynamic. A static gesture 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. Four steps with regard to gesture are; acquisition, modelling, feature extraction and recognition. Gesture acquisition for dynamic hand gestures is carried using web camera. Gesture modelling consists of pre-processing steps. Segmentation is used to detect hand from its background. Gesture recognition from Marathi Sign Language(MSL) alphabets database consisting of total 40 gestures and total dataset of 200 images with different orientation will be carried out using feature extraction and training the gestures using ANN. Recognition includes multilayer perceptions.***

Keywords: MSL, HSV, ANN, MLP, Back Propagation

1. Introduction

Gesture recognition is an area of active current research in computer vision. It brings visions of more accessible computer system. In this paper we focus on the problem of real time hand gesture recognition using MLP. We have considered single-handed gestures, which are sequences of distinct hand shapes and hand region. A Gesture is defined as a motion of the hand to communicate with a computer.

Hand gestures are classified in two categories static and dynamic. A static gesture 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. Our main focus is on the recognition of dynamic images. In this each gesture will have a specific meaning and automatic the audio sound will be generated by using Artificial Neural Networks (ANNs) which provides a new suite of nonlinear algorithms for feature extraction (using hidden layers) and classification (e.g., multilayer perceptrons). Multilayer Perceptron (MLP) has three layers comprising of one input layer, one hidden layer and one output layer. The output layer uses linear transfer function. It is trained using back propagation algorithm. The network is trained by using a built in train function .This function trains the network on training data (Supervised Learning). The training algorithm used for the network is Gradient Descent (GD).

2. Literature Review

Table 1: Literature Review on [1] - [10]

Year	*Author*	*Work done*	*Classifier Used*	*Accuracy*
2003	Kouichi M. et al.	Human to interface with machine in a more natural way.	back propagati-on network & Elman recurrent network	71.4% 96%
2004	Sharma, A. et al.	Recognition of predefined hand and gross body actions using artificial neural network.	MLP	N/A
2005	Xiang Li et al.	Vision based gesture interaction	Fuzzy C-Means algorithm	85.83%
2006	Dang Binh et al.	Recognition of hand gesture based on pseudo three-dimensional hidden Markov model (P3DHMM)	P3DHMM	99.1%
2007	Qing Chen et al.	Real-time vision-based hand gesture classification.	AdaBoost	92.3%
2008	Lichtenauer et al.	Recognition of sign language	HMM	89.66%
2009	Nguyen et al.	Recognition of Vietnamese language	P2-DHMMs	90%
2010	Doe Hyung et al.	Game interface-ng using hand gesture recognition	Difference image entropy	85%
2011	Charkari, et al.	Recognition of the image-based numbers of Persian sign language (PSL) using thinning method on segmented image.	Real endpoints	96.6%
2012	Ghotkar et al.	Recognition of the alphabets of Indian Sign Language	Genetic Algorithm	N/A

3. System Overview

The main aim of this paper is creating visual biased analysis application to perform Hand Gesture Recognition of Marathi Sign Language (MSL) alphabets successfully without any error regardless the person hand sizes and other external causes.

Licensed Under Creative Commons Attribution CC BY

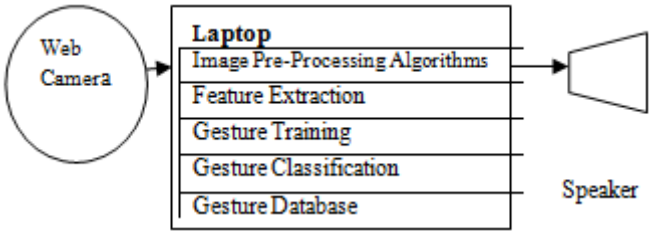


Figure 1: System Overview

3.1 Image Processing Algorithms

a) Hand Detection

In this main hand object which is the core of the work is detected this split the area of interest into two main segments, foreground which is the hand and the background**.** The background elimination algorithm uses simple algorithms to separate foreground regions from background to reduce the amount of computation required. A simple background elimination method keeps a reference background image and compares the input frame with that reference image. Foreground regions are those regions where input image is different from the reference image.

b) Image Segmentation

In this step, the hand is extracted as one object from the input image, we have applied image segmentation using color based model for human skin detection by RGB to HSV color space conversion in order to extract the hand object from a cluttered background, this segmentation algorithm has a significant property which is color seeking algorithm. For human-skin pigment detection,it is given by formula,

I (x,y) = hand cluster if $0 \leq H(x,y) \leq 0.1228$ and $0.2666 \leq S(x,y) \leq 0.8777$ background cluster otherwise

Where I(x, y) is the input image, H(x, y) is the hue value of the corresponding pixel and S(x, y) is the saturation value of the same for HSV color space.

c) Calibration to Hand Size

Instead of processing the whole image area, a small area of interest can be cropped and processed which speeds up the processing speed and helps of geometric features locating.

d) Noise Removal

The aim of this technique is to reduce the noise that is originally embedded into the image or the new emerged noise that creeps during the former processing steps. This includes median filtering. The median filter is also a sliding-window spatial filter, but it replaces the centre value in the window with the median of all the pixel values in the window.

3.2 Feature Extraction

a) Edge Detection

Here we have used canny approach to edge detection as it is optimal for step edges corrupted by white Gaussian noise. This edge detector is assumed to be output of a filter that both reduces the noise and locates the edges.

Canny Edge Detection Algorithm:

1) Smoothing: Blurring of the image to remove noise.
2) Finding gradients: The edges should be marked where the gradients of the image has large magnitudes.
3) Non-maximum suppression: Only local maxima should be marked as edges.
4) Double thresholding: Potential edges are determined by thresholding.
5) Edge tracking by hysteresis: Final edges are determined by suppressing all edges that are not connected to a very certain (strong) edge.

b) Mean

The mean is used as a feature extraction purpose. It calculates mean value of row and column then calculates mean of that calculated value.

3.3 Gesture Training and Classification

a) Artificial Neural Network

Artificial Neural Networks (ANNs) provide a new suite of nonlinear algorithms for feature extraction (using hidden layers) and classification (e.g., multilayer perceptrons). Multilayer Perceptron (MLP) having three layers comprising of one input layer, one hidden layer and one output layer has been used. The output layer uses linear transfer function. It is trained using back propagation algorithm. The network is trained by using a built in train function .This function trains the network on training data (Supervised Learning). The training algorithm used for this network is Back Propagation Algorithm.

b) Multilayer Perceptron Model (Three Layer)

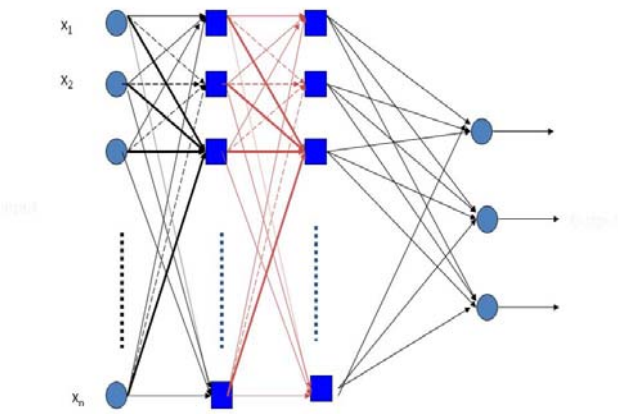


Figure 2: Multilayer Perceptron Model

Features of Multilayer Perceptron network:

1. No connection within layer
2. No direct connection between input and output layer
3. Fully connected between layers
4. Number of output units need not equal to number of input units

Licensed Under Creative Commons Attribution CC BY

5. Number of hidden units per layer can be less than or more than input or output units

c) Back Propagation Algorithm

Back propagation is a general purpose learning algorithm. A back propagation network with a single hidden layer of processing elements can model any continuous function to any degree of accuracy .There are literally hundreds of variations of back propagation in the neural network literature, and all claim to be superior to "basic" back propagation in one way or the other. Indeed, since back propagation is based on a relatively simple form of optimization known as gradient descent. The basic back propagation algorithm consists of three steps:

1) The input pattern is presented to the input layer of the network. These inputs are propagated through the network
2) Because back propagation is a supervised learning algorithm, the desired outputs are given as part of the training vector. The actual network outputs are subtracted from the desired outputs and an error signal is produced.
3) This error signal is then the basis for the back propagation step, whereby the errors are passed back through the neural network by computing the contribution of each hidden processing unit and deriving the corresponding adjustment needed to produce the correct output.

Two major learning parameters are used to control the training process of a back propagation network. The learn rate is used to specify whether the neural network is going to make major adjustments after each learning trial or if it is only going to make minor adjustments. Momentum is used to control possible oscillations in the weights, which could be caused by alternately signed error signals. While most commercial back propagation tools provide anywhere from 1 to 10 or more parameters for you to set, these two will usually produce the most impact on the neural network training time and performance. Actual algorithm for a 3-layer network (only one hidden layer):

Step1:
Initialize the weights in the network randomly.
DO
For each example e in the training set O = neural-net-output (network, e) ; forward pass T = teacher output for e.
Step 2:
Calculate error (T - O) at the output units.
Step 3:
Compute delta_wh for all weights from hidden layer to output layer .
Step 4:
Backward pass Compute delta_wi for all weights from input layer to hidden layer.
Step 5:

Backward pass continued Update the weights in the network Until all examples classified correctly or stopping criterion satisfied Return the network **.**

4. Results

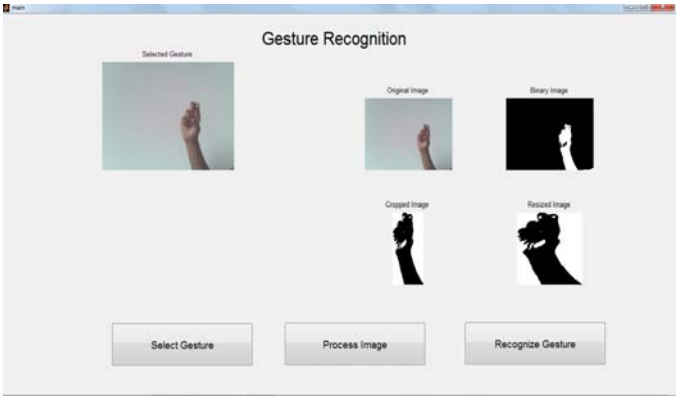


Figure 3: Gesture Recognition Results

Figure 3 shows result of static gesture recognition using GUI. The experimental setup used for this is MATLABR2012.The gesture is selected from database and pre-processing, feature extraction steps are carried out and finally a recorded sound is generated for that particular gesture. The results for MSL gesture is shown in the figure.

5. Conclusion and Future Scope

In this paper, we developed a system for the purpose of the recognition of sign language. The system have three stages: preprocessing stage, feature extraction stage and the gesture classification stage. The work was accomplished by training a set of input data. Without using any glove, an image for the sign is taken by a camera. After preprocessing, the feature extraction stage is applied on each of segmented image. Multilayer Perceptrons is applied in gesture classification stage. The importance of this work is to help primarily for deaf to solve their communication problem.

Area of Hand gesture based computer human interaction is very vast. This project recognizes hand gesture of recognition system can be useful in many fields like robotics and so make this offline system for real time will be future work to do. Line so work can be done to do it for real time purpose.

References

[1] Kouichi M, "Real-time hand shape recognition for human interface," *IEEE Comput. Graph. Appl.*, pp. 77–86, Sept. 2003.
[2] Sharma A, "Recognition of human actions using moment based features and artificial neural networks," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp.696–710, Jan 2004.
[3] Beifang Yi, "Real-time natural hand gestures," *IEEE Trans.Computing in Science & Engineering*, vol. 7, pp.92-96, June 2005.
[4] Nguyen Dang Binh, "Real-Time hand Gesture Recognition Using Pseudo 3-D Hidden Markov Model," IEEE *International Conference on Cognitive Informatics.*, vol. 4, pp. 820-824,July 2006.
[5] Qing Chen, "Real-time Vision-based Hand Gesture Recognition Using Haar-like Features," *IEEE*

Licensed Under Creative Commons Attribution CC BY

Instrumentation and Measurement Technology Conference Proceedings., vol. 2, pp. 1-6, May. 2007.

[6] Lichtenauer, "Sign Language Recognition by Combining Statistical DTW and Independent Classification," *IEEE Pattern Analysis and Machine Intelligence Anal. Machine Intell.*, vol. 30, pp. 798–804, Aug. 2008.

[7] Bao , " Real time hand gesture recognition using HMM ," *IEEE Trans.Fuzzy Systems and Knowledge Discovery*, vol. 4, pp. 3-6, July 2009.

[8] Doe-Hyung , "Game interfacing using hand gesture recognition," *IEEE Comput.Graph. Appl.*, vol. 14, pp. , Jan. 2010.

[9] Barkoky, A, "Static hand gesture recognition of Persian sign numbers using thinning method," *IEEE Multimedia Technology (ICMT)*, vol. 10, pp. 6548 - 6551, July 2011.

[10] Ghotkar, A.S, "Hand gesture recognition for Indian Sign Language," *IEEE Computer Communication and Informatics (ICCCI)*, pp. 6548 - 6551, Jan 2012.

Author Profile

Priyanka Tandale received the B.E. and M.E. degrees in Electronics and Telecommunication Engineering from Pune University. Now she is associated with one of the research institute in India.

Licensed Under Creative Commons Attribution CC BY