IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 3, Issue 2, April-May, 2015
ISSN: 2320 – 8791 (Impact Factor: 1.479)
www.ijreat.org

.

Sign Language To Voice Conversion Using Image Processing In MATLAB

Gaurav Sharma[1], Shubhi Tondon[2], Gauri Dave[3], Mr. Pankaj Agrawal[4]

[1, 2, 3] Students (4th Year), Department of Electronics and Communication Engineering, SRM University, NCR Campus, Modinagar.

[4]Asst. Professor, Department of Electronics and Communication Engineering, NCR Campus, Modinagar.

Abstract

In order to eliminate communication between gap between dumb & deaf people and physically abled people, this system converts their sign language to voice. This project shows a prototype of the same using image processing in MATLAB. Algorithm used for gesture recognition is SIFT which was designed by Dr. David Lowe. It consists of a microcontroller to read the MATLAB output from serial port and relay a signal to Voice Processor apr33a3.

Key Words: *SIFT- Scale Invariant Feature Transform [1], interest points, key points, descriptor, gesture recognition, training image, testing images, feature extraction and feature matching, ASL- American sign language.*

1. Introduction

ASL is universally acceptable sign language and has its own grammar. Including grammar is beyond the scope of this paper and has been included in future work. Gesture recognition is a prominent field of pattern recognition and SIFT is one of the most successful algorithm used in this. The reason of its success because it is scale invariant i.e. the recognition is independent of the size, orientation or depth of training and testing image. Both training and testing images are went under sift and key points are extracted and then their matching is done. The one with most no. of matches of the training images is chosen and a particular ASCII value is generated that is the output of the MATLAB that is collected by microcontroller ARDUINO DUEMILANOVE and relayed to voice processor APR33AR to produce the corresponding voice via speaker.

2. Image Processing

It is vast field to explore and gesture recognition is a part of it.

Using gesture recognition the sign of ASL is taken as the input and the recognised gesture is the output as a form of recognised training image.

2.1 Gesture recognition

There are many gesture recognition algorithms like DCT, skin detection, neural networks etc. The one implemented her is SIFT [1].

2.1.1 SIFT

It is a key point extractor and it transform image data into a scale invariant coordinates. The output of SIFT are scale invariant. Advantages of SIFT are locality (features are local so robustness to occlusion and clutter is provided) , distinctiveness (individual feature can be matched to a large number of database images), quantity (there are large number of features generated for an object) and efficiency (it is motivated from human neural networks so the efficiency is comparable).

3. Steps involved in SIFT

There are four steps involved namely Scale space peak detection, Key point localization, Orientation assignment and Key point descriptor.

3.1 Scale space peak detection

The peaks are the possible locations for finding features by changing the value of σ (scales) in the Gaussian Filter. Testing and training images are subjected to Gaussian Filter to interest points. The whole spectrum of σ values are used it is explained in [2].

3.1.1 Building Scale space

All scales must be examined to identify invariant features. An efficient function is to use Laplacian pyramid [3]. An image is takes and Gaussian filter with a particular σ (basically means blurring). Then the value σ of sigma is increased to kσ, then $k\sigma^2$ and so on. Then size of the image is reduced in both the rows and columns and then again the same set of σ values is applied.

.

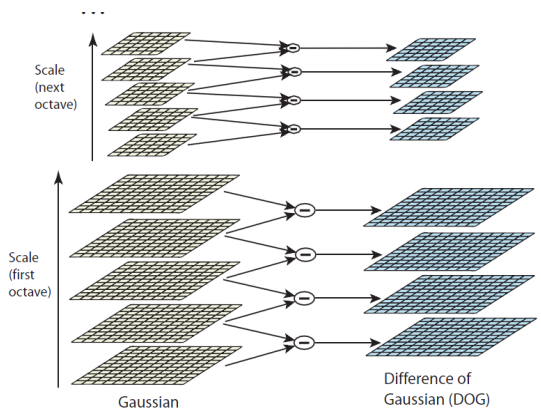


Fig 1- Set of blurred images at different scale of same size is called octave. After first octave the size of image is reduced and again applied Gaussian filter to get the next octave.

3.1.2 Laplacian of Gaussian (LOG)

After building the scale space interest points but be found and to do that peak must be determined. In order to find the peak LOG is performed. Interest points are the maxima or minima of a LOG. To apply LOG on scale space find out the difference between the Gaussian output at one scale to the one next higher scale. This difference of Gaussian (DOG) is approximated to LOG as shown in Fig 1. It is proved as follow.

Suppose we have a Gaussian Function G and the Laplacian of Gaussian as Δ. We use standard heat equation and replace the temperature variable by σ of the Gaussian filter.

$$\frac{\delta G}{\delta \sigma} = \sigma \Delta^2 G$$

$$\sigma \Delta^2 G = \frac{\delta G}{\delta \sigma} = (G(x, y, k\sigma) - G(x, y, \sigma))/(k\sigma-\sigma)$$

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 \Delta^2 G$$

The above equation shows that DOG can be approximated to LOG with typical values σ=1.6 and k=√2.

$$G(x, y, k\sigma) = \frac{1}{2\pi(k\sigma)^2} e^{-(x^2+y^2)/2k^2\sigma^2}$$

k=√2.

3.1.3 Peak detection

Compare a pixel (x) with 26 pixels neighbour scales. 8 on the same scale and 9 in the higher and lower scale. If the pixel (x) is maxima or minima of all the 27 points then it is an interest point. There will be large number of maxima and minima so chose only the most stable ones in all octaves.

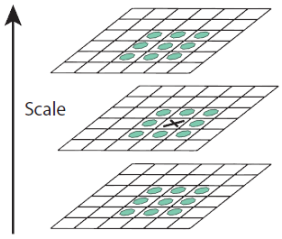


Fig 2- Pixel (x) is compared with neighbour 26 pixels, if it maxima or minima then it is an interest point.

3.2 Key point Localisation

The best scale for each interest point is to be located.

3.2.1 Initial outlier rejection

Low contrast interest points should be removed. Poorly localised interest points along an edge needs to be removed. To remove these Taylor series expansion of the interest points. A function can be approximated by Taylor series [1] as

$$D(x) = D + \frac{\delta D^T}{\delta x} + \left(\frac{1}{2}\right) x^T \delta^2 D/\delta x^2$$

x=(x, y, σ).

Maxima and minima are located at

$$\hat{x} = -\left(\frac{\delta^2 D^{-1}}{\delta x^2}\right)\left(\frac{\delta^2 D}{\delta x^2}\right)$$

If value of D(x) at maxima/minima must be large |D(x)|>threshold, then that will be selected which are stable rest will be ignored.

3.2.2 Further outlier rejection

DOG has strong response along edge because edges are ambiguous. Suppose DOG as a surface and compute the principle curvature (pc). Along the edge on the pc is very low and across the edge it is high. This is similar to Harris values used in Harris detector [4]. Analogous to Harris detector compute Hessian matrix of D

$$H = \begin{matrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{matrix}$$

Outliers are removed by evaluating the trace of the matrix.

$$T_r(H) = D_{xx} + D_{yy=} \lambda_1 + \lambda_2$$

.

$$Det(H) = D_{xx}D_{yy} - (D_{xy})^2 = \lambda_1\lambda_2$$

This is minimum when r=1, eliminates the key points if r>10.

3.3 Orientation Assignment

To achieve rotation invariance find the domination orientation of the interest point then align the other points in neighbourhood to that and result is a canonical representation of the orientation. Compute the central derivative and gradient magnitude & direction of L (smooth image) at the scale of key point (x, y).

$$m(x) = \sqrt{\big(L(x+1,y) - L(x-1,y)\big)^2 + \big(L(x,y+1) - L(x,y-1)\big)^2}$$

$$\theta(x,y) = tan^{-1}\left(\frac{L(x,y+1) - L(x,y-1)}{\big(L(x+1,y) - L(x-1,y)\big)}\right)$$

Now study the orientation of all the points around that key point and built a histogram of 36 bins. Then assign the weights to the pixels but where the gradient value is more it will count more. Then make the histogram having 36 bins. Weights are basically the gradient magnitude and spatial Gaussian function with σ=1.5 x<scale of key point>.

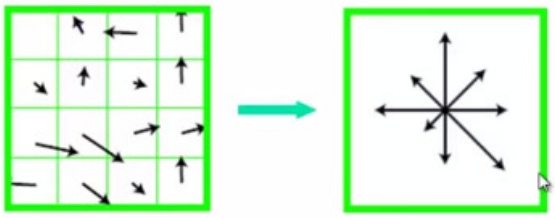


Fig. 3- Orientation of points are converted in to histogram of 36 bins.

From the histogram select the one with highest peak and that will be the orientation of the interest point. Introduce additional key points at local peaks of the histogram with different direction.

3.4 Key point descriptor

After having the key point it is needed to be described by some parameters like intensity. Here gradient orientation histogram is used as the descriptor because it is stable even if illumination changes. To extract local image descriptors at key points compute the relative orientation in 16*16 neighbourhood at a key point. For each key point there is a dominating orientation so we compute the direction w.r.t the dominating orientation of all points.

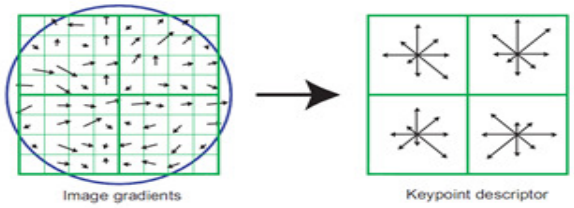


Fig. 7- A key point descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the key point location, as shown on the left. These are weighted by a Gaussian window, indicated by the overlaid circle. These samples are then accumulated into orientation histograms summarizing the contents over 4x4 sub regions, as shown on the right, with the length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region. This figure shows a 2x2 descriptor array computed from an 8x8 set of samples, whereas the experiments in this paper use 4x4 descriptors computed from a 16x16 sample array.

Take 16*16 neighbourhood and divide it in 4*4 block, then find histogram of each block then concatenate 16 histograms in to a 126 dimension long vector. 4*4 is an empirical value explained in [1]. Store these numbers in a vector and then normalise to unit vector (it will be helpful to illumination invariance. Remove high values using non-linear transform and bound unit vector item to maximum 0.2 and then normalise to unit vector because high values will affect the normalization.

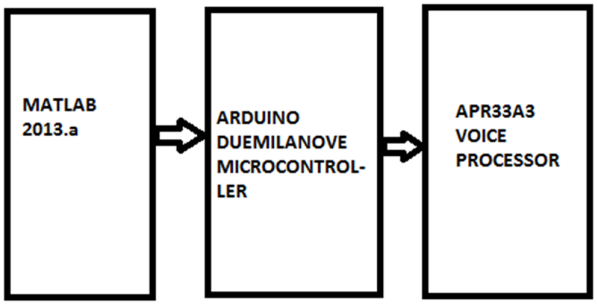
4. Key point matching-

To match the key points against the training images find the nearest neighbour i.e. a key point with minimum Euclidean distance. Every key point of testing image will be compared with every key point in training images to find the best match. To make efficient nearest neighbour match look at the ratio of distance between the best and 2nd best match. To generate candidate match, find patches that have most similar appearance or SIFT descriptor. If the ratio of first and second best match is low first match looks good, if high the match is ambiguous.

5. Hardware Implementation

The hardware consisted of a microcontroller Arduino duemilanove card and voice processor APR33A3. . Flow diagram of hardware working is as follows.

.



5.1 Arduino Duemilanove

It is card with ATmega168 microcontroller and other peripheral and interfacing components. Operating Voltage is 5v, input voltage is 7-12 v, Digital I/O pins are 14 of which 6 provide PWM outputs, analog input pins are 6, DC current per I/O is 40 mA. It has Flash memory of 16 kb of which 2kb is used by boot loader. SRAM is 1kb, 512 bytes EEPROM and clock is set at 16 MHz It reads the MATLAB 2013.a from the serial port and relay corresponding signal to the voice processor.

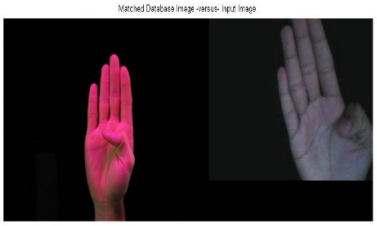
5.2 APR33A3

It is an APLUS manufacture 16 bit voice processor capable of 340-680 s recording and playback divided in 8 message slots. It operates at 3v-5v which was provided by Arduino. When Pin 12/Rec will be set low then whichever message pin will get high will record the message and when Pin 12 will be set high then giving low to message slots will playback the recorded sound [5]. This was controlled by MATLAB 2013.a – Arduino interfacing which was based on ASCII value to the letter representing the gesture.

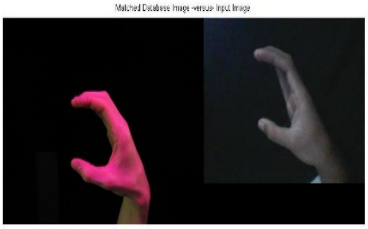
6. Results

Results were categorised in two parts, first the image processing results and the first with voice outputs. Efficiency was calculated by success results over total results.

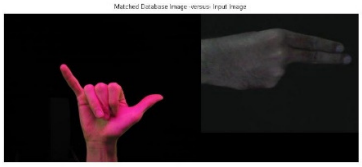
Input- b, output -b



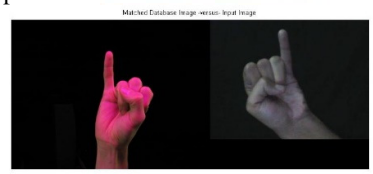
Input- c, output-c



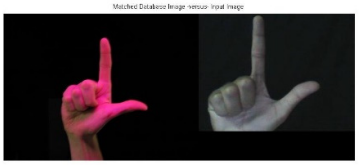
Input-h, output- y



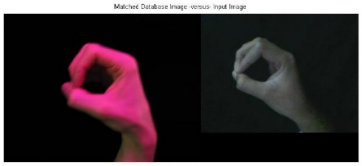
Input- i, output- i



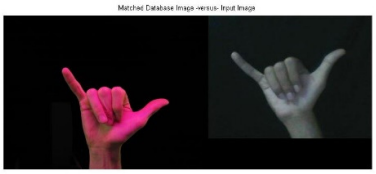
Input- l, output-l



Input- o, output- o



Input-y, output- y



Voice results were based on the hardware working according to the detected output. It is tabularised as follows.

.

Input	Output	Result	Voice result corresponding to output	Overall result
b	b	Success	Success	Success
c	c	Success	Success	Success
h	y	Fail	Success	Fail
i	i	Success	Success	Success
l	l	Success	Success	Success
o	o	Success	Success	Success
y	y	Success	Success	Success
Percentage		85.71	100	85.71

7. Future Work

The future includes the removal of having black background and apply it to the real time environment and inclusion of sign language grammar. The improvement in hardware is the removal of delay which is included due to discharging problem. Continuous real time conversion must be included.

References

[1]. David G. Lowe, "Distinctive Image Features from Scale-Invariant Key points", Computer Science Department, University of British Columbia Vancouver, B.C., Canada, 2004.

[2]. Andrew P. Witkin, "Scale-Space Filtering", Fairchild Laboratory for Artificial Intelligence Research.

[3]. Peter J. Burt and Edward H. Edelson, "The Laplacian Pyramid as a Compact Image Code", IEEE Transactions on Communications, VOL. COM-31, No. 4, April 1983.

[4]. Chris Harris & Mike Stephens, "A COMBINED CORNER AND EDGE DETECTOR", Plessey Research Roke Manor, United Kingdom, 1988.

[5] aPR33A3, Fixed 1/ 2/ 4/ 8 Message Mode (E2.1), Datasheet, Recording voice IC, APLUS, http://www.aplusinc.com.tw/proimages/Recording%20Ic/aPR33Ax/20140108/aPR33A3_E2.1_Datasheet_20131126.pdf.