Hand Gesture Recognition based on Digital Image Processing using MATLAB

By Tahir Khan under supervision of Dr. Amir Hassan Pathan
Faculty of Engineering, Sciences and Technology, IQRA University
Karachi, Pakistan
Email: khan.tahir@gmail.com

Abstract - This research work presents a prototype system that helps to recognize hand gesture to normal people in order to communicate more effectively with the special people. Aforesaid research work focuses on the problem of gesture recognition in real time that sign language used by the community of deaf people. The problem addressed is based on Digital Image Processing using Color Segmentation, Skin Detection, Image Segmentation, Image Filtering, and Template Matching techniques. This system recognizes gestures of ASL (American Sign Language) including the alphabet and a subset of its words.

Index Terms— Hand Gesture Recognition, Digital Image Processing, Skin Detection, Image Segmentation, Image Filtering, Template Matching technique.

1. Introduction

Communication is a Latin word derived from SCIO means to share. Communication means to share thoughts, messages, knowledge or any information. Since ages communication is the tool of exchange of information through oral, writing, visuals signs or behaviour. The communication cycle consider to be completed once the message is received by a receiver and recognizes the message of the sender. Ordinary people communicate their thoughts through speech to others, whereas the hearing impaired community the means of communication is the use of sign language.

Around 500,000 to 2,000,000 speech and hearing impaired people express their thought through Sign Language in their daily communication [1]. These numbers may diverge from other sources but it is most popular as mentioned that the ASL is the 3rd most-used sign language in the world.

2. Objective

This research work focuses on the problem of gesture recognition in real time that sign language used by the community of deaf people. Research problem identified is based on Digital Image Processing using Color Segmentation, Skin Detection, Image Segmentation, Image Filtering, and Template Matching techniques. This system recognizes gestures of ASL including the alphabet and a subset of its words.

The gesture recognition method is divided into two major categories a) vision based method b) glove based method. In glove based systems data gloves are used to achieve the accurate positions of the hand sign though, using data gloves has become a better approach than vision based method as the user has the flexibility of moving the hand around freely.

There are many possible vision based methods are available. Above all, Byong K. Ko and H. S. Yang developed a finger mouse system that enables a signer to specify commands with the fingers as in [2]. Apart from that, there are other different methods available such as colored hand-glove based method, Neural Network and PCA as in [3] to [5] etc.

Though, implementation of Neural Network is very simple, but it is used to be over-trained on such a limited training sample particularly obstructed gesture sign also may cause a problem. In these circumstances, it is very difficult to predict the response of a neural network. Also the Neural Network can potentially create erroneous results due to environment variation. In the other hand PCA, due to the very limited training set. PCA faces the same problem of over-specification of the gesture sign as well as may involve lowering the dimensionality of the image.

The main goal of this research paper is to demonstrate that how a good performance can be achieved without using any special hardware equipment, so that such a system can be implemented and easily used in real life.

Author Tahir Khan has accomplished his Masters of Philosophy program in Computer Science from Iqra University, Karachi, Pakistan.
E-mail: khan.tahir@gmail.com

The contribution of this research paper can be summarized in this manner that the paper emphasises to use Template Matching technique as the primary hand gesture sign detection method due to its conceptual simplicity and my confidence in it. Apart from that this method has ability to combine feature detection with gesture detection very easy which can be done by creating more templates.

3. What is ASL?

ASL (American Sign Language) is a language for hearing impaired and the deaf alike people, in which manual communication with the help of hands, facial expression and body language are used to convey thought to others without using sound. Since ASL uses an entirely different grammar and vocabulary, such as tense and articles, does not use "the", therefore, it is considered not related to English.. ASL is generally preferred as the communication tool for deaf and dumb people.

4. The Method

The idea behinds this method is that the software run in a mobile handset having a frontal camera while a disabled person (who is in the front of the mobile handset) makes the signs. This software recognizes these gestures of ASL, including letters and a subset of its words and produces text message for corresponding word or letter for normal people to understand.

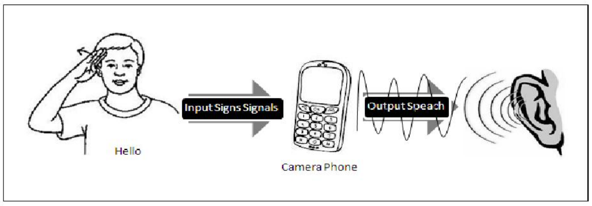

Figure 1: Sign Language Interpreter

In this sign language interpreter system, mobile frontal camera is the input device for observing the information of hands or fingers of the user and then these inputs presents to the system to produce the text message for normal people to understand the gestures.

The development of such a visual gesture recognition sys-tem is not an easy task. There are a numerous environmental concerns and issues are associated with this Sign Language Interpreter System from real world. Such as visibility, this is the key issue in the performance of such system, since it determines the quality of the input images and hence affects the performance.

5. Concerns and Issues

Visibility issue may arise due to several reasons For instance, the camera where the user has to stay in position, the various environmental condition like lighting sensitivity, background color and condition, electric or magnetic fields or any other disturbance may affect the performance.

- Occlusion can be occurred due to the occluded figures, while signing [6].
- The boundaries of gesture have to be automatically detected. For example the start sign and the end sign for alphabets especially "J" and "Z" have to be detected automatically.
- The sitting or standing position of the signer may vary in front of camera. Movements of the signer, like rotating around the body must be taken into account.
- Delay in the processing execution can be occurred due to the large amount or higher resolution of image. Hence, it is difficult to recognize in real time basis.

6. Image Acquisition

The most common method of Image Acquisition is done by digital photography with usually a digital camera but other methods are also considered. The Image Acquisition includes compression, processing, and display of images.

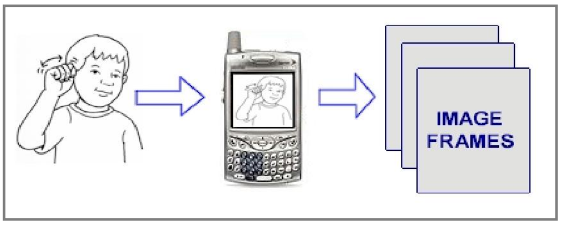

Figure 2: Image Capturing Process

The image/frames of the person conveying the message using hand gesture can be obtained by using a frontal camera of mobile phone. The reason for choosing mobile camera phone instead of a traditional camera for capturing the image is that, it is the easiest way to transfer text or voice message to the other ordinary person's mobile device through a mobile network.

Figure 2, shows the description of the image acquisition process. In this research work, I have considered that the mobile camera faces towards the signer to capture the image of the hand gestures of the signer.

7. Image Processing Steps

To satisfy and reduce the computational effort needed for the processing, pre-processing of the image taken from the camera is highly important. Apart from that, numerous factors such as lights, environment, background of the image, hand and body position and orientation of the signer, parameters and focus the of camera impact the result dramatically.

8. Color Segmentation

Color in an image is apparent by human eyes as a combination of R(red), G(green) and B(blue), these three colors i.e Red, Green and Blue are known as three primary colors. Other kinds of color components can be derived from R,G,B color represented by either linear or nonlinear transformations.

The RGB color components represent the incoming light, that is the brightness values of the image that can be obtained through (Red, Green and Blue filters) i.e RGB filters based on the following equations:

$$R=\int_{\lambda} E(\lambda) S_R(\lambda) d\lambda$$

$$G=\int_{\lambda} E(\lambda) S_G(\lambda) d\lambda$$

$$B=\int_{\lambda} E(\lambda) S_B(\lambda) d\lambda$$

$S_{R.}S_{G.}S_{B.}$ represent filters of the color on incoming light, whereas $E(\lambda)$ is radiance and (λ) is the wavelength of the image.

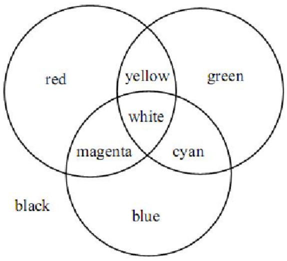

Figure 3 *RGB color model*

It has been highly praised that human eye can only distinguish two-dozen of color out of thousands of color shades and intensities. It is quite often difficult to extract an object or recognize a pattern from image using gray scale, the object can only be extracted using color information. Since color information provides additional information to the intensity as compared to grayscale. Therefore, the Color information is extremely necessary for pattern recognition.

Though, there is no any common theory available for color image segmentation up till now. The color image segmentation methods all we have are yet, either by nature or ad hoc basis. The color segmentation approaches are dependent on the application , there are no any common algorithms which is considered the best for color image segmentation. The color image segmentation is a psychophysical perception, since it is very essential to have pre-knowledge of mathematical solutions about the image information.

The main purpose of Color segmentation is to find particular objects for example lines, curves, etc in images. In this process every pixel is assigned in an image in such a way that pixels with the same label share certain visual characteristics.

The goal of color segmentation in this research work is to simplify and increase the ability of separation between skin and non-skin, and also decrease the ability of separation among skin tone.

9. Skin Detection

There are several techniques used for color space transformation for skin detection. Some potential color spaces that are considerable for skin detection process are:

- CIEXYZ
- YCbCr
- YIQ
- YUV

A performance metric that the other colorspaces have used is scatter matrices for the computation of skin and non-skin classes. Another drawback is to comparison through histogram of the skin and non-skin pixel after transformation of colorspace.

The YCbCr, colorspace performs very well in 3 out of 4 performance metrics used [7]. Thus, it was decided to use YCbCr colorspace in skin detection algorithm.

In this research work, Skin Detection process involves classification of each pixel of the image to identify as part of human skin or not by applying Gray-world Algorithm for illumination compensation and the pixels are categorized based on an explicit relationship between the color components YCbCr. In YCbCr colorspace, the single component "Y" represents luminance information, and Cb and Cr represent color information to store two color-difference components, Component Cb is the difference between the blue component and a reference value, whereas component Cr is the difference between the red component and a reference value [8].

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112.000 \\ 112.000 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

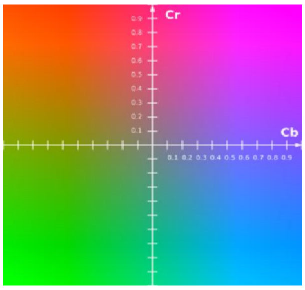

Figure 4: *The CbCr plane at constant luma Y*

Thus, a pixel is considered a human skin, if a set of pixel is falling into that particular category with a certain value of Cr and Cb having certain threshold.

$$f(Cr, Cb) = \begin{cases} 1 \ if (Cr, Cb) \in S \\ 0 \ if (Cr, Cb) \notin S \end{cases}$$

Where S is the tuples of Cr and Cb values that consider as skin.

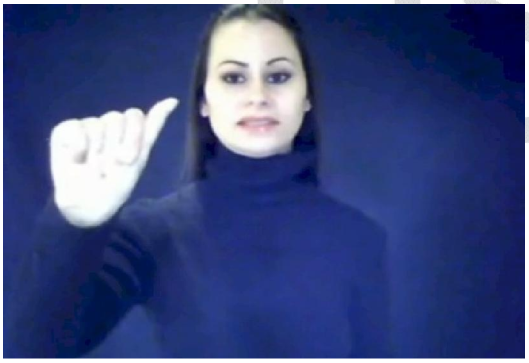

Figure 5: Original Image of the signer

$$\text{skin}=\begin{cases} 1 \ if (Cb \geq 77 \ and \ Cb \leq \ 127 \ and \ Cr \geq 133 \ and \ Cr \leq 173) \\ 0 \ otherwise \end{cases} .$$

Overlaid onto the image with human skin pixels marked in blue color, so that the gesture can easily be identified.

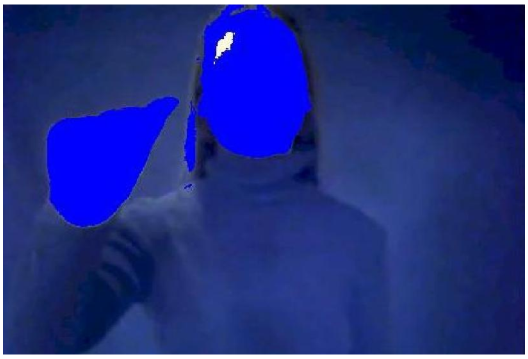

Figure 6: Image with skin pixels marked in Blue color

After the skin detection, image marked with Blue color converted into the binary with skin pixels as '1' and rest are "0". So that, the correlation of the image can be matched with the Template,

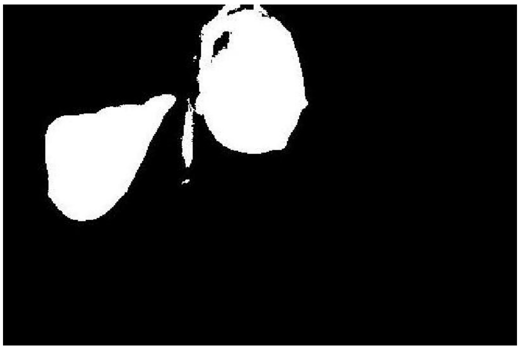

Figure 7: *Binary Image of the signer after skin detection*

The skin detection algorithm implements the following steps:

- Read the image(RGB color image) , and capture the dimensions (height and width)
- Initialize the output images
- Apply Grayworld Algorithm for illumination compensation
- Convert the image from RGB to YCbCr
- Detect Skin:
- Mark Skin Pixels with Blue

10. Image Segmentation

To reduce the computational time needed for the processing of the image, it is important to reduce the size of the image and only the outline of the sign gesture has to be process. After conversation of the image into binary, the outline of the hand gesture is cropped as a vector of (x, y) coordination, assuming that the hand gesture covers left corner of the image.

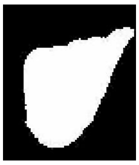

Figure 8: *Image after cropping, ready to match with template*

```
height = size(inputImage,1)/1.5;
width = size(inputImage,2)/2;
imcrop(inputImage,[0 0 width height]);
```

11. Image Filtering

In Image Filtering technique, the value of the pixel of any given image is determined by applying algorithm to the value of the pixels in the neighborhood. This Filter is referred to the sub image, mask, kernel, template or window.

There are mainly two types of Image Filtering.

1. Spatial Filtering
2. Linear Filtering

I have considered Linear Filtering as the primary hand gesture sign detection method due to the following reasons:

1. There are some limitations with Spatial Filtering. i.e it limits the outline of center of mask to be at distance no less than (n-1)/2 pixel from border. That resulting output image shorter than the original image. That means one or more column of the mask will be located outside image plane.

2. Linear Filtering conceptual simplicity and my confidence in it.

12. Implementation

The Template Matching cross-correlation involves simply multiplying together corresponding pixels of the signer image, here is called the Target image and the Template and then summing the result.

Template Matching is implemented by the following method:

- First, select a part of the search image that can be used as image template: called the search image. i.e S(x, y), where S represents Search Image, x and y represent the coordinates of each pixel in the search image.
- The template T(xt, yt), where T represents Template, xt and yt represent the coordinates of each pixel in the template.
- Then the center of the template T(xt, yt) moves over each x and y point in the search image. And then sum up the products between the coefficients in Search Image S(x, y) and the Template T(xt, yt) over the complete area of the target image.
- The search image considers all the position of the template.
- The largest value of the position is considered the best position of the object.

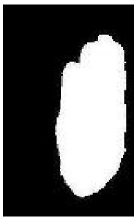

Figure 9: *Search Image S(x y)*

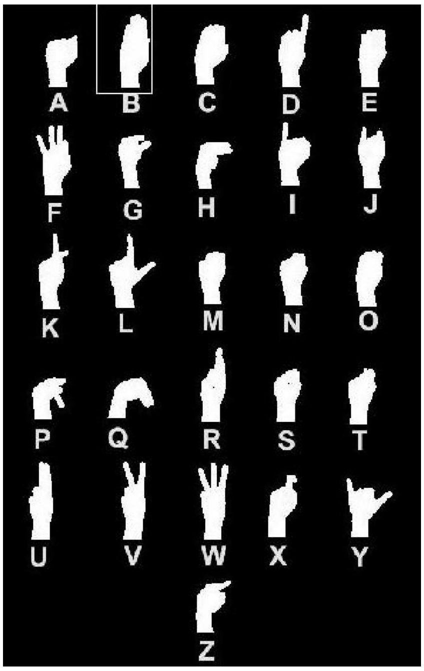

Figure 10: *Template Image T(xt, yt)*

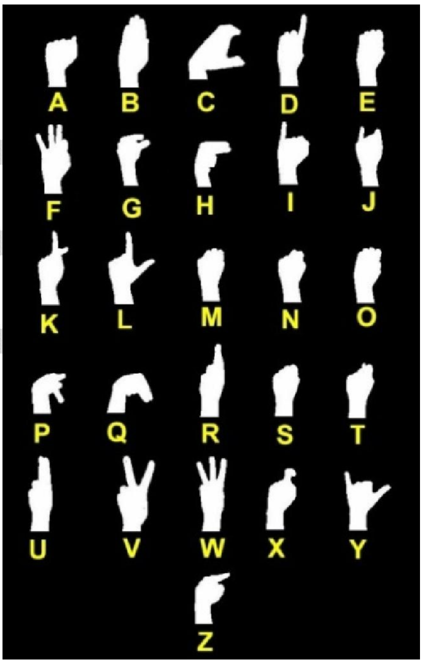

Figure 11: *Result: the output of the Convolution*

Cross-correlation is used to compare the intensities of the pixels using template matching to handle the translation issue on the signer image.

For our hand gesture recognize application in which the brightness of the input image of the signer can vary due to the various environmental condition like lighting sensitivity, background color and condition, electric or magnetic fields or any other disturbance and exposure conditions of the signer, the images has to be first normalized. The norma-lization has to be done at every step by subtracting the mean and dividing by the standard deviation. That algorithm is called the cross-correlation of a template and represented as follow:

$$\frac{1}{n-1}\sum_{x,y}\frac{\left(f(x,y)-\bar{f}\right)\left(t(tx,ty)-\bar{t}\right)}{\sigma_f\sigma_t}$$

Where n represents the number of pixels in the template t(tx,ty) and f(x,y) $\bar{f}$ is the average of f and σ_f is standard deviation of f.

13. Speed up the Process

Template matching as we know is a two-dimensional cross-correlation of a grayscale image. There are lot of other elements that have major influence on estimating the level of similarity between the template and the target image. Since, a two-dimensional cross-correlation calculation for a large image in term of size is not only very time-consuming but also it is very difficult to estimate the overall performance of the system as well. To overcome this situation, I have used the correlation in the frequency domain by multiplying the two-dimensional Fourier Transforms (FFT). And then took the inverse of FFT to obtain the output image. This reduces the processing time significantly.

Due to the complication of the process, image filtering was normally done in dedicated hardware systems in the past. In order to speeding up the matching process, the process can also be done through the use of an image pyramid. The Image Pyramid is basically a series of images, which has different scales, created by repeatedly filtering and sub-sampling the input image like we have here the signer's image to generate a sequence of reduced resolution images. Then, these lower resolution images are searched for the template having the similarly in term of resolution, to delay the possible start positions for searching the image at the larger scales. Then, these larger images are searched in a small window nearby the start position to find the best template position.

Apart from the above mentioned process using image pyramid there is another way of speeding up the process of template matching through filtering the image in the frequency domain, also called 'frequency domain filtering,' . Frequency Domain Filtering is done through the convolution theorem.

So, I have considered convolution theorem as because according to the convolution theorem, under suitable conditions the Fourier transform of a convolution is point wise product of Fourier transforms. Or we can say, in time domain, convolution is point-wise multiplication in frequency domain

$$\mathcal{F}\{f * g\}=\mathcal{F}\{f\}\,.\,\mathcal{F}\{g\}$$

Where f and g are two functions with convolution f * g. (the asterisk * denotes convolution and not multiplication). $\mathcal{F}$ denote the Fourier transform operator. Thus, $\mathcal{F}$ {f} and $\mathcal{F}$ {g} are Fourier transform of f and g respectively. And . denotes point-wise multiplication

14. Improving the Accuracy

In order to improve the accuracy of the template matching method, I have decided to use a secondary template of hand gesture (subimage or masking), this secondary template have slightly different gesture sign with different angle, considering an overall hit only if a proper and correct ges-ture sign is supplied. This is an additional advantage as this secondary template allows system to let go the individual thresholds to get all the possible correct hand gesture sign. Apart from that it also helps to determinate the start and end boundary of signs alphabets involve motion like "J" and "Z".

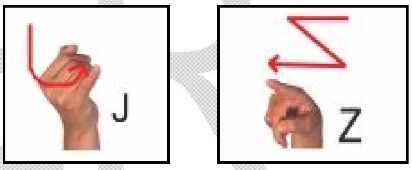

Figure 12: Signs alphabets involve motion

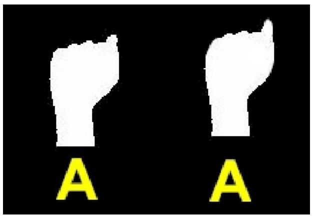

Figure 13: Primary and Secondary Templates with different angle for sign "A"

15. Result and Analysis

The purpose of this application is to recognize hand gesture. The design is very simple and the signer doesn't need to wear any type of hand gloves. Although this sign language recognition application can be run in an ordinary computer having a web camera, but ideally it requires Android Smart phone having frontal camera with at least 1GHz processor and at least 514MB RAM. The template set consists of all alphabets A to Z. The letters J and Z involves motions and hence required secondary templates to determinate start and end boundary of the alphabets. Table 1 represents result of algorithm presented using the code in MATLAB. The algorithm can detect all the alphabets from A to Z with 100% recognition rate if the signer supplies the correct sign.

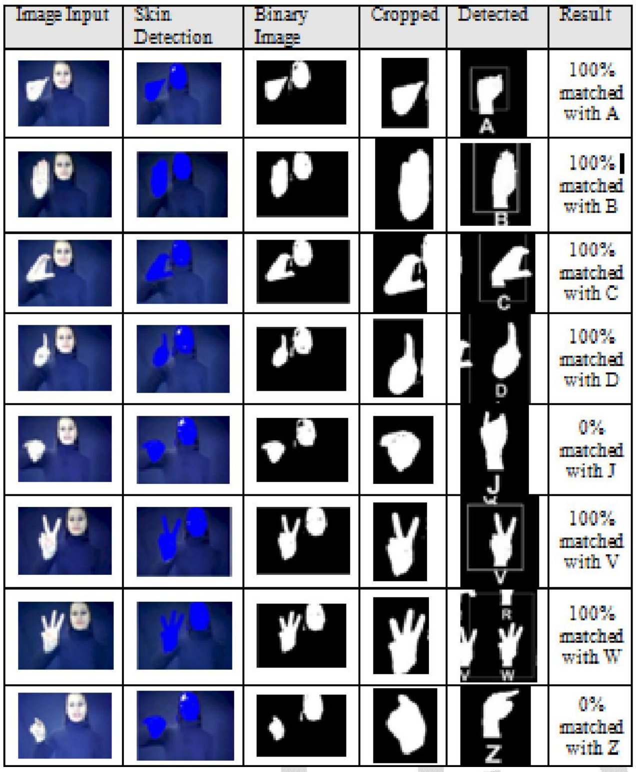

Image Input	Skin Detection	Binary Image	Cropped	Detected	Result
					100% matched with A
					100% matched with B
					100% matched with C
					100% matched with D
					0% matched with J
					100% matched with V
					100% matched with W
					0% matched with Z

Table 1*: Analysis and Result*

The system can recognize a set of 24 letters from the ASL alphabets: A, B, C, D, E, F, G, H, I, K, L, M, N, O, P, Q, R, S, T, U, V, W. The only issue found for alphabets involve motion like J and Z.

A statistic regarding the performance of this system using single template can be found:

Recognized alphabets	Recognition accuracy
A, B, C, D, E, F, G, H, I, K, L, M, N, O, P, Q, R, S, T, U V, W, X, Y	100%
J and Z	0%

Overall performance of the system: 92.30%

MATLAB code for skin detection

Filename: generate_skintone.m

```
function [out bin] = generate_skintone(inputimage)
%GENERATE_SKINTONE Produce a skinmap of an inputimage. Highlights patches of %skin" like pixels. Can be used in , gesture recognition,
.
  if nargin > 1 | nargin < 1
    error(generate_skinmap(inputimage)');
  end;

  %Read input image
  img_inputimg_input = imread(inputimage);
  img_height = size(img_input,1);
  img_width = size(img_input,2);

  %Initialize the images
  out = img_input;
  bin = zeros(img_height,img_width);

  %Apply Grayworld Algorithm    Img_gray =
grayworld(img_input);

  %Convert from RGB to YCbCr
  imgycbcr = rgb2ycbcr(img_gray);
  YCb = imgycbcr(:,:,2);
  YCr = imgycbcr(:,:,3);

  %Detect Human Skin
  [r,c,v] = find(YCb>=77 & YCb<=127 & YCr>=133
& YCr<=173);
  numind = size(r,1);

  %Mark Humain Skin Pixels
  for i=1:numind
    out(r(i),c(i),:) = [0 0 255];
    bin(r(i),c(i)) = 1;
  end
  img_show(img_input);
  figure; img_show(out);
  figure; img_show(bin);
end
```

Fliename: grayworld.m

```
function_out = grayworld(input_image)
%Color Balancing
%   input_image- 24 bit RGB Image
%   result - Color Balanced 24-bit RGB Image

  result = uint8(zeros(size(I,1), size(I,2), size(I,3)));

  %R,G,B components
  R = input_image(:,:,1);
  G = input_image (:,:,2);
  B = input_image (:,:,3);

  %Inverse of the Avg values
  mR = input_image /(mean(mean(R)));
  mG = input_image /(mean(mean(G)));
  mB = input_image /(mean(mean(B)));

  %Calculate the Smallest Avg Value
  max_RGB = max(max(mR, mG), mB);

  % Compute  the scaling factors
  mR = mR/max_RGB;
  mG = mG/max_RGB;
  mB = mB/max_RGB;

  %Calculate the scale values
   result(:,:,1) = R*mR;
   result(:,:,2) = G*mG;
   result(:,:,3) = B*mB;
end
```

MATLAB code for Template Matching

Filename: template.m

```
close all
clear all

% read the input image
img1=imread('template/template_bw.jpg');

% read the traget template Image
img2=imread('target/v_crop.jpg');

% apply templete matching algorithm using DC
components
result2=tmc(img1,img2);

figure,
imshow(img2);title('Target');
figure,imshow(result2);title('Matching Result using
tmc');
-----------------------------------------------------------------
----
Filename: tmcmain.m
function result1=tmc(img1,img2)

if size(img1,3)==3
  img1=rgb2gray(img1);
end
if size(img2,3)==3
  g2=rgb2gray(img2);
end

% recognize which one is target and which one is
template
if size(img1)>size(img2)
   Target_image=img1;
   Template_image=img2;
else
   Target_image=img2;
   Template_image=img1;
end
% calculate the size of both images
[r1,c1]=size(Target_image);
[r2,c2]=size(Template_image);

% calculate the mean of the template
image22=Template_image-
mean(mean(Template_image));


%corrolate target and template images
corrMat=[];
for i=1:(r1-r2+1)
   for j=1:(c1-c2+1)
      N_image=Target_image(i:i+r2-1,j:j+c2-1);
      N_image=N_image-mean(mean(N_image));
      corr=sum(sum(N_image.*image22));
      corrMat(i,j)=corr;
   end
end
% plot the box on the target image
result1=plotbox(Target_image,Template_image,corr
Mat);
```

16. CONCLUSION

The statistic of result of the implementation, it is therefore, concluded that the method is used for template matching and color segmentation work with high accuracy with hand gesture recognition. The results obtained are applicable, and can be implemented in a mobile device smart phone having frontal camera. However, only issue was found for alphabets involve motion like J and Z, which are recommended to be handled through multiple secondary templates.

17. References

[1] Paulraj M. P. Sazali Yaacob, Mohd Shuhanaz bin Zanar Azalan, Rajkumar Palaniappan, "A Phoneme based sign language recognition system using skin color segmentation", Signal Processing and its Applications (CSPA) – pp: 1 – 5, 2010.

[2] Byong K. Ko and H. S Yang, "Finger mouse and gesture recognition system as a new human computer interface", pp: 555-561, 1997.

[3] Manar Maraqa, Dr. Raed Abu Zaiter, "Recognition of Arabic sign Language using recurrent neural networks", Applications of

Digital Information and Web Technologies, pp: 478 – 481, 2008.

[4] Yang quan, "Chinese Sign Language Recognition Based on Video Swquence Appearance Modeling", ICIEA, the 5th IEEE Conference, pp: 1537 – 1542, 2010.

[5] K. Kawahigasi, Y. Shirai, J. Miura, N. Shimada "Automatic Synthesis of training Data for Sign Language Recognition using HMM, pp: 623 – 626, 2006.

[6] P. Mekala, R. Salmeron, Jeffery Fan, A Davari, J Tan, "Occlusion Detection Using Motion-Position Analysis" IEEE 42nd Southeastern Symposium, on System Theory (SSST"10), pp: 197-201, 2010.

[7] Jae Y. Lee and Suk I. Yoo "An Elliptical Boundary Model for Skin Color Detection" pp: 2-5, 2002.

[8] Digital Image Processing Using Matlab by Ganzalez, p: 205. 2009.