Real time static gesture detection using machine learning

|  |  |
| --- | --- |
| 1st Sandipgiri Goswami  Computer Science Department  Laurentian Unversity Sudbury,Canada [sgoswami@laurentian.ca](mailto:sgoswami@laurentian.ca) | 2nd Dr. K Passi  *Computer Science Department*  Laurentian UnversitySudbury,Canada [kpassi@cs.laurentian.ca](mailto:kpassi@cs.laurentian.ca) |

**Abstract: Sign gesture recognition is an important problem in human-computer interaction with signiﬁcant societal influence. However, it is a very complicated task, since sign gestures are naturally deformable objects. Gesture recognition contains unsolved problems for the last two decades, such as low accuracy or low speed, and despite many proposed methods, no perfect result has been found to explain these unsolved problems. In this thesis, we suggest a machine learning approach to translating sign gesture language into text.**

**In this study, we have introduced a self-generated image data set for American Sign Language (ASL). This dataset is a collection of 36 characters containing alphabets A to Z and numeric digits 0 to 9. The proposed system can recognize static gestures. This system can learn and classify specific sign gesture of any person. We used a convolutional neural network (CNN) algorithm for the classification of the images to text. An accuracy of 99.00% was achieved on the alphabet gestures and 100% accuracy on digits.**

*Keywords: Sign gestures, Image processing, Machine learning, Conventional neural network.*

1. Introduction

The World Health Organization (WHO) estimated that 250 million people in the world are deaf [1]. This group of people use symbolic language to communicate with other people. This symbolic language is called a sign language. Sign Language was developed for people with hearing-impairment so they could communicate with others. Sign language is not a unique language and there have been very improvements in the sign languages. There is proof that communication between hearing-impaired has been carried since the development of the sign languages [2]. Gesture-based communication is dependent on region and has significant differences from other languages. It is essential to understand sign language when we communicate with hearing-impaired and their families. Lack of understanding results in significant challenges in understanding this community and may result in miscommunication. Sign Language is used to convey messages using hand movements, facial expressions and body language. It has been used by deaf people who cannot hear but can speak. Sometimes family members and relatives need to learn sign language to communicate with the hearing-impaired. By using a sign language, a successful channel of communication can be established among the hearing-impaired community.

In this thesis, Image classification and machine learning have been used for interpreting American Sign Language. For image classification, computer vision algorithms were used to capture images and to process data set for filtering as well as reducing noise from images. Finally, the data set is trained to recognize hand gestures using machine learning algorithm. A conventional neural network (CNN) for measuring the accuracy of training data sets. The general overview of the derived approach is combining the image classification and machine learning for American sign language recognition as shown in Figure 1.

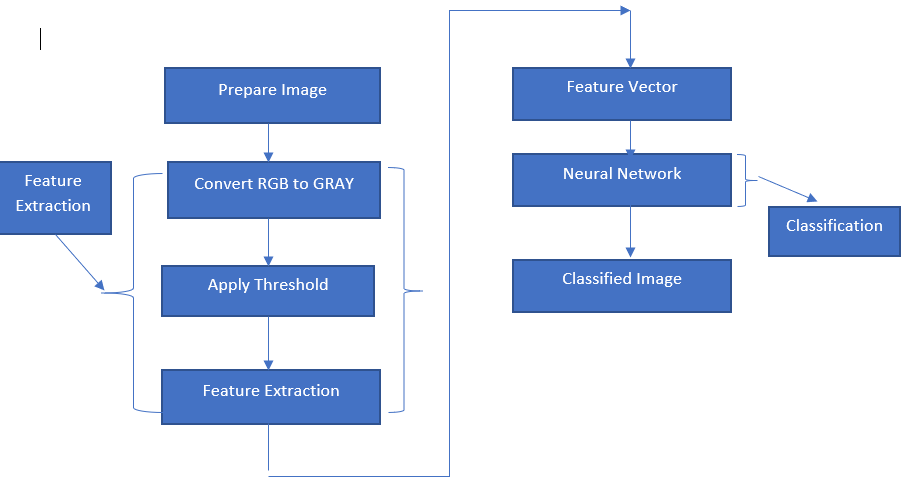


Fig. 1. System Architecture.

1. Related Work:

Machine learning has been used for image recognition. Hidden Markov Model (HMM) and Dynamic Time Warping (DTW), two kinds of machine learning methods, are widely applied to achieve high accuracies [5, 6, 7]. These are mostly good at capturing time-based patterns, but they require characterized models that were defined before learning. Sterner and Pentland [5] used a Hidden Markov Model and a 3-Dimensional glove that detects hand movement. Since the glove can attain 3-Dimensional detail from the hand regardless of spatial orientation, they achieved the best accuracy of 99.2% on the test set. Hidden Markov Model uses time series data to track hand actions and classify based on the position of the hand in new frames.

As per researcher’s point of view, a linear classifier is easy to work with because the linear classifiers are a relatively simple model, it requires sophisticated feature extraction and preprocessing methods to get good results [2, 3, 4]. Singha and Das [2] achieved an accuracy of 96% on Ten classes for images of gestures of one hand using Karhunen-Loeve Transforms. It translates and rotates the axes to build up a new framework based on the variance of the data. This technique is useful after using a skin colour detection, hand cropping and edge recognition on the images. They use a linear classifier to recognize number sign including a thumbs up, first and index finger pointing left and right, and numbers only. Sharma [4] has done research using Support Vector Machines (SVM) and k-Nearest Neighbors (KNN) to illustrate each colour channel after background noise deletion and noise subtraction [4]. Their research suggests using contours, which is very useful to represent hand contours. They achieved an accuracy of 62.3% using a Support Vector Machine on the segmented colour channel model.

Suk [8] suggested a system for detecting hand gestures in a continuous video stream using a dynamic Bayesian network or DBN model. They try to classify moving hand gestures, such as creating a circle around the body or waving. They attain an accuracy of nearly 99%, but it is worth noting that all hand gestures are different from each other and are not American Sign Language. However, the motion-tracking feature would be applicable for classifying the dynamic letters of ASL: j and z.

1. Data set and Variables

A dataset for American Sign Language was created as there is no standard dataset available for all countries/subcontinents [9]. In the present scenario, the requirement for large vocabulary dataset is in demand [9]. Moreover, the existing dataset is incomplete and additional hand gestures had to be included. In future, this research will help other researchers to develop their own dataset based on their requirements. This dataset is a collection of 36 characters which contains A to Z alphabets and 0 to 9 numerical digits. Right hand was used to capture 1000 images for specific alphabets and numbers. Code was implemented to flip the images from the right to the left-hand image. The height and width ratios vary significantly but average approximately 50x50 pixels. The dataset contains over 72,000 images in grayscale. Additionally, people can add their images to this dataset. Figure 2 shows images of A to Z alphabet. Table I describes the dataset properties.

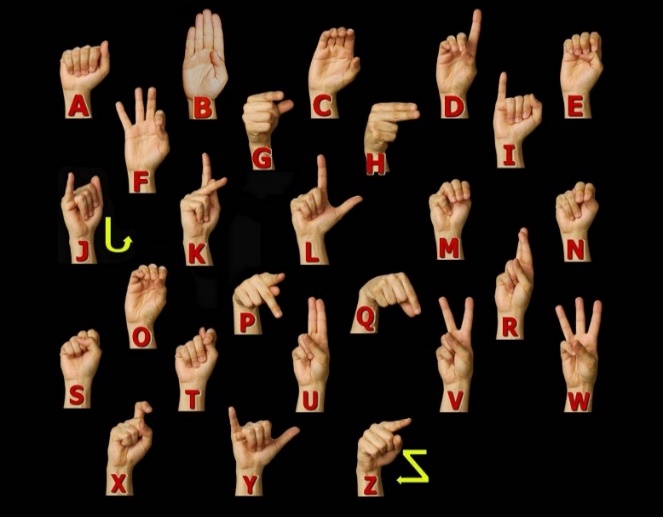


Fig. 2. American Sign language Manual Alphabet [10].

****

Fig. 3. American Sign language Manual Number [10].

Table I: Dataset Description and Image property.

|  |  |
| --- | --- |
| **Property** | **Description** |
| Alphabets | A to Z |
| Numbers | 0 to 9 |
| Color | Greyscale |
| Dimensions | 50x50 |
| Height | 50 pixels |
| Width | 50 pixels |
| File type | JPEG |

1. Approach for Hand Detection In This research

To detect hand gesture using skin colour, there are different approaches including skin color-based methods. In this thesis, after detecting and subtracting the face and other background, skin recognition and a contour comparison algorithm were used to search for the hand and discard other background colour objects for every frame captured from a webcam or video file. First, we need to extract and store the hand contour and skin color which is used to compare with the frame in order to detect the hand. After detecting the skin area for each frame captured, the contours of the detected areas were compared with the saved hand histogram template contours to remove other skin like objects existing in the image. If the contour comparison of the spotted skin area complies with any one of the saved hand histogram contours, then it captured hand gesture only.

An integrated system is proposed for detection, segmentation, and tracking of the hand in a gesture recognition system using a single webcam. Some other methods use colour gloves [11, 12], whereas our method can detect the plain hand posture by integrating two useful features: skin colour detection and contour matching. My proposed hand posture finding algorithm has real-time performance and is strong against rotations, scaling, cluttered background, and lighting conditions. Detecting the human hand in a plain background will boost the performance of hand gesture recognition systems. In this method, the speed and result of recognition will be the same for any frame size taken from a webcam such as 640×480, 320×240 or 160×120 and the system will also be robust against a plain background as only the hand posture area is detected. A smaller image size that holds the detected hand posture area is suitable for training image size of a training stage of the classification.

 To detect the hand gesture in the image, a four-phase system was designed as shown in Figure 5. First, load hand contour template which will be used to compare and detect hand skin area pattern from webcam using the contours comparison algorithm. Then we will open a camera which has a square box to capture hand gesture. The hand must be placed fully within the square box. The skin colour locus (captured skin contour template) for the image is removed from the user’s skin colour after face deletion. In the last step, the hand gesture is spotted by removing false positive skin pixels and identifying hand gesture and other real skin colour regions using contour matching with the loaded hand gesture pattern contours.

Fig. 5. Hand posture detection steps

1. *Skin Detection*

Skin detection is a useful approach for many computer vision applications such as face recognition, tracking and facial expression, abstraction, or hand tracking and gesture recognition. There are recognized procedures for skin colour modelling and recognition that will allow differentiating between the skin and non-skin pixels based on their colour. To get a suitable distinction between skin and non-skin areas, a colour transformation is needed to separate luminance from chrominance [14].

The input images normally are in Colour format (RBG), which has the drawback of having components dependent on the lighting situations. The misunderstanding between the skin and non-skin pixels can be decreased using colour space transformation. There are different approaches to detection of skin colour components in other colour spaces, such as HSV, YCbCr, TSL or YIQ to provide better results in parameter recovery under changes in lighting conditions. Researches have shown that skin colours of individuals cluster tightly in the colour space for all people from different societies, for example, colour appearances in human faces and hands vary more in intensity than in chrominance [13, 15]. Thus, take away the intensity V of the original colour space and working in the chromatic colour space (H, S) provides invariance against illumination situations. In [14], it was shown that removal of the Value (V) component and only using the Hue and Saturation components, can still permit detection of 96.83% of the skin pixels. In this research, hue, saturation, value (HSV) colour model was used, since it has been shown to be one of the most adapted to skin-colour detection [16]. HSV model is easier to represent a color than RGB color space. It is also well-matched with human colour perception. Also, it has real-time execution and it is more robust in cases of rotations, scaling, cluttered background, and changes in lighting condition. The projected hand gesture detection algorithm is real-time and sturdy against the mentioned previous changes. The other skin like objects existing in the image are removed from contour comparison with the loaded hand posture prototype contours.

The HSV colour space is gained by a nonlinear transformation of the essential RGB colour space. The conversion between RGB and HSV was described in [17]. Hue (H) is a section that characterizes a pure colour such as pure yellow, orange or red, whereas saturation (S) provides a measure of the degree to which a pure colour diluted by white light [18]. Value (V) attempts to represent brightness along the gray axis such as white to black, but since intensity is subjective, it is thus difficult to measure [18].

According to [18] and Figure 6, Hue is estimated in HSV colour space by a position with Red starting at 0, Green at 120 and Blue at 240 degrees. The black mark in the diagram at the lower left on the screen determines the hue angle.

Saturation is a ratio that ranges between 0.0 along the middle line of the cone (the V axis) to 1 on the edge of the cone. Value ranges, string from 0.0 (dark) to 1.0 (bright).



Fig. 6. HSV Color Space [18].

According to [12] , the HSV model can be resulting from the non-linear transformation from an RGB model according to the following calculations.

As per a classification point of view, skin-colour detection is divided into two class problem: skin-pixel vs non-skin-pixel classification. Currently, different known classification approaches exist such as thresholding, Gaussian classifier, and multilayer perceptron [20, 21, 22].

In this research, thresholding technique is used that allows getting a good result for higher computation speed when compared with other techniques, given our real-time requirements. In image processing, Gaussian blur is used to reduce noise and reduce details. Thresholding classification is used to find the values between two components H and S in the HSV model as the Value (V) component was removed. Usually, a pixel can be observed as being a skin-pixel when the following threshold values are satisfied: 0° < H < 20° and 75° < S < 190°. In the last step, skin detected object is stored into pickle object, called contour template. Contour template is a curve joining all continuous points having the same intensity. As we stated that contours are the boundary of a shape with the same intensity. It stores the values of x and y coordinates of boundary of the shape, to store all coordinate values.

1. *Contour Comparisons*

Once the contour template has been detected, the contours of the detected skin colour are recovered and then compared with the contours of the hand gesture patterns. Once skin colour contours are recognized as belonging to the hand gesture contour patterns, that area will be identified as a region of interest (ROI) which will then be used for tracking the hand movements and saving the hand posture in JPEG format in small images as shown in Figure 7. After that, stored images are further used to extract the features needed to recognize the hand postures in the testing stage.

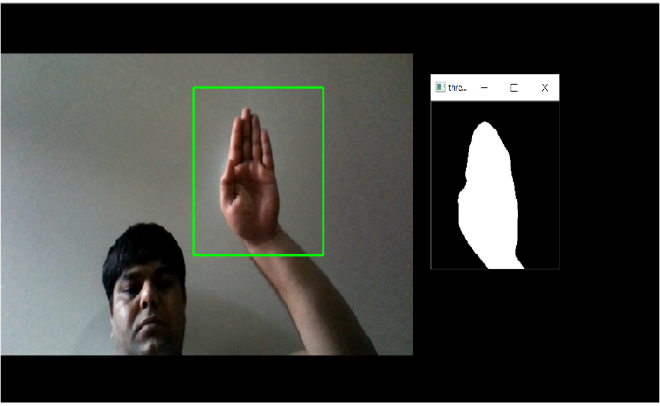


Fig. 7: Image of detecting hand postures.

1. Method

Supervised learning was used in this research as the dataset images have assigned labels. The supervised learning method is more commonly used when data is known. This method needs training data with a specific format. Each instance must have assigned label. These labels make available supervision for the learning algorithm. The training process of supervised learning is constructed on the following principle. First, the training data are fed into the model to produce estimates of output. This estimate is compared to the assigned label of the training data in order to evaluate the model error. Based on this error the learning algorithm alters model’s parameters in order to reduce it.

*A. Architecture*

Our architecture is commonly used in CNN architecture. This architecture consists of multiple convolution and dense layers. The CNN architecture includes three types of three convolution layers, and each layer has its max pooling layer and one group of fully connected layer followed by a dropout layer and the output layer as shown in Figure 8.



Fig. 8. CNN network architecture for Alphabets.

*B. Hardware and Software Configuration*

Training of Neural Networks is notoriously computationally expensive and it requires a lot of resources. From the bottom level perspective, it translates into many matrix multiplications. Modern Central Processing Units (CPUs) are not capable of such computations and therefore are not very efficient. On the other hand, modern GPUs are designed to perform exactly these operations.

At present there are two main parallel computing platforms, CUDA and OpenCL. They both have their advantages and disadvantages, but the major difference is that CUDA is proprietary, while OpenCL is available free. This divide translates into hardware productions as well. CUDA is mostly supported by NVIDIA and OpenCL is supported by AMD. NVIDIA with its CUDA platform is presently a leader in the domain of deep learning. Therefore, for the training of CNN models, GPUs from NVIDIA was selected. The selected model was GIGABYTE GeForce GTX 1080. Detailed information about the hardware configuration is given in Table 2.

Table II: Hardware Configuration

|  |  |
| --- | --- |
| GPU | GeForce GTX 1080 4GB |
| CPU | Intel(R) Core(TM) i7-8550 CPU @ 2.00GHz |
| Memory | DIMM 1333MHz 8GB |

From the list of considered software tools, Keras was selected as it is written in python, an easy to program language and as it satisfies all considered factors. Support for efficient GPU implementation in Keras relies on either Tensor flow or Theano back-end. From the different user perspectives, it doesn’t matter either way, but Tensor flow was selected because it was observed as faster of the two and has GPU-accelerated library package of primitives for deep neural networks. Detailed information about software configuration is given in Table 3.

Table III: Software Configuration

|  |  |
| --- | --- |
| Keras | 2.2.2 |
| Tensorflow | 1.9.0 |
| CUDA | 9.2 |
| Python | 3.6.0 |
| Operating System | Window 10 |
| Open CV | 3.4.3 |

1. Result

On our self-generated dataset, we achieved an accuracy of 99.00% to detect hand gestures for alphabets and 100% accuracy was achieved to detect hand gestures for digits. Real-time testing was done with five different students and estimation per student took 20 minutes for alphabets and approximately 7 to 8 minutes for digits. Testing was done in non-controlled form, i.e. in different lighting condition and with different backgrounds. For alphabets 50 epochs were applied, and for digits 20 epochs were applied. Both networks used Adam optimizer and a learning rate of 0.001. Loss function was cross-entropy due to multiple-class classiﬁcation. The training and testing sets contained 70:30 ratio, respectively in both models. Figure 9 shows the validation accuracy for different epochs for the digits. Figure 10 shows the validation accuracy for different epochs for alphabets. The confusion matrices of both networks are illustrated in Figures 11 and 12. From both confusion matrices, it is evident that the classiﬁcation accuracy of both models is almost identical. 

Fig. 9. Epochs vs. validation accuracy for digits.



Fig. 10. Epochs vs. validation accuracy for alphabets.



Fig. 11. Confusion matrix for 0 to 9 digits.

A screenshot of a cell phone

Description automatically generated

Fig. 12. Confusion matrix for A to Z alphabets.

ROC Area Under the Curve (AUC): The Receiver Operating Characteristics (ROC) charts is a method for organizing classification and visualizing the performance of the trained data using the classifier. ROC AUC for all the digit classes is 1, from which one can say that the performance evaluation of all classes is excellent in the data set. Also, the weighted value of the ROC AUC is 1 (100%) for the data classified using the CNN network, which correctly classified the data with the best accuracy and best performance. ROC AUC curves for the digits and alphabets are shown in Figures 13 and 14, respectively.

A screenshot of a cell phone

Description automatically generated

Figure 13. ROC AUC graph for 0 to 9 digits.

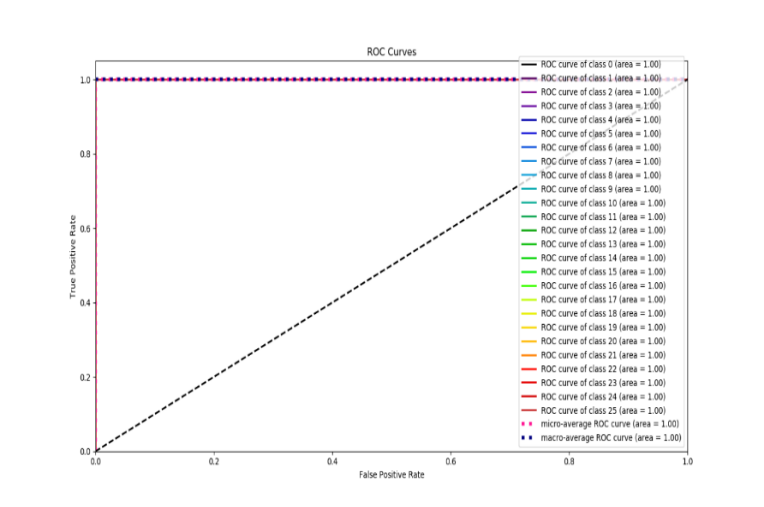


Figure 14. ROC AUC graph for A to Z alphabets

1. Conclusion and Feature work

In this paper we developed a system to recognize American Sign gesture using a skin color model, thresholding and CNN. We have tested with different lighting condition and in a different place. The dataset collected in the ideal conditions has proved to be the most efficient dataset in terms of accuracy and gives 99% accuracy on alphabets and 100% accuracy on digits.

Sign gesture recognition still has a long way to go in the research path, especially for 2D systems. This study offers fascinating ideas for future research. Some of these possibilities are defined in this section. As this thesis focused only on static sign gesture recognition, one next step forward is to recognize the dynamic sign gesture for the ASL. Even though that study introduces a self-generated a new dataset with a rather more gesture for American Sign Language, it still does not offer all the possible movements for American Sign Language. Videos with rotation in 3Dimension, words and expressions are examples of how this dataset can be extended.

1. References

1] WHO calls on private sector to provide affordable hearing aids in developing world. [Internet]. Current neurology and neuroscience reports. U.S. National Library of Medicine; 2001 [cited 2019Jan22]. Available from: <https://www.ncbi.nlm.nih.gov/pubmed/11887302>.

2] Emond A, Ridd M, Sutherland H, et alThe current health of the signing Deaf community in the UK compared with the general population: a cross-sectional studyBMJ Open 2015;5:e006668. doi: 10.1136/bmjopen-2014-006668

3] Das K, Singha J. Hand Gesture Recognition Based on Karhunen-Loeve Transform. 2013Jan17;:365–71.

4] Aryanie, D., & Heryadi, Y. (2015). American sign language-based finger-spelling recognition using k-Nearest Neighbors classifier. *2015 3rd International Conference on Information and Communication Technology (ICoICT)*, 533-536.

5] Sharma, Rohit & Nemani, Yash & Kumar, Sumit & Kane, Lalit & Khanna, Pritee. (2013). Recognition of Single Handed Sign Language Gestures using Contour Tracing Descriptor. Lecture Notes in Engineering and Computer Science. 2.

6] T. Starner and A. Pentland, "Real-time American Sign Language recognition from video using hidden Markov models," *Proceedings of International Symposium on Computer Vision - ISCV*, Coral Gables, FL, USA, 1995, pp. 265-270.  
doi: 10.1109/ISCV.1995.477012

7] Jebali, Maher & Dalle, Patrice & Jemni, Mohamed. (2013). Extension of Hidden Markov Model for Recognizing Large Vocabulary of Sign Language. International Journal of Artificial Intelligence & Applications. 4. 10.5121/ijaia.2013.4203.

8] Suk, Heung-Il & Sin, Bong-Kee & Lee, Seong-Whan. (2010). Hand gesture recognition based on dynamic Bayesian network framework. Pattern Recognition. 43. 3059-3072. 10.1016/j.patcog.2010.03.016.

9] Sahoo, Mishra A, S G, Ravulakollu, K K. Sign Language Recognition : State of the Art. Asian Res. State of the Art Asian Res. 9(2):116–34.

10] William Vicars / ASL University. ASL [Internet]. Children of Deaf Adults (CODA). [cited 2019Jan29]. Available from: http://www.lifeprint.com/.

11] Aran O, Keskin C, Akarun L. Computer Applications for Disabled People and Sign Language Tutoring. Proceedings of the Fifth GAP Engineering Congress. 2006Apr;:26–8.

12] Tokatlı H, Halıcı Z. 3D Hand Tracking in Video Sequences. MSc Thesis. 2005Sep;

13] He J, Zhang H. A Real Time Face Detection Method in Human-Machine Interaction. 2008 2nd International Conference on Bioinformatics and Biomedical Engineering. 2008;

14] Zhu Q, Wu C-T, Cheng K-T, Wu Y-L. An adaptive skin model and its application to objectionable image filtering. Proceedings of the 12th annual ACM international conference on Multimedia - MULTIMEDIA 04. 2004;

15] Kelly W, Donnellan A, Molloy D. Screening for Objectionable Images: A Review of Skin Detection Techniques. 2008 International Machine Vision and Image Processing Conference. 2008;:151–8.

16] Zarit B, Super B, Quek F. Comparison of five color models in skin pixel classification. Proceedings International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems In Conjunction with ICCV99 (Cat NoPR00378). :58–63.

17] Ford A, Roberts A. Color space conversions. Westminster University,London,UK. 1998Aug11;

18] Gonzalez R, Woods R, Eddins S. Digital Image Processing Using MATLAB. Englewood Cliffs, NJ. 2004;

19] Hughes JF. Computer graphics: principles and practice. Upper Saddle River, NJ: Addison-Wesley; 2014.

20] Nallaperumal K, Ravi S, Babu K, Selvakumar K, Fred A, Seldev C, et al. Skin detection using color pixel classification with application to face detection: A comparative study. Proc IEEE Int Conf Comput Intell Multimedia Appllication. 3:436–41.

21] Greenspan H, Goldberger J, Eshet I. Mixture model for face-color modeling and segmentation. Pattern Recognition Letters. 2001;22(14):1525–36.

22] Phung SL, Chai D, Bouzerdoum A. A universal and robust human skin colour model using neural networks. IJCNN01 International Joint Conference on Neural Networks Proceedings (Cat No01CH37222).