

# Blending of Face Segments

Akshay Kajekar<sup>#1</sup>, Shreyash Patel<sup>\*2</sup>, Sweta Chauhan<sup>#3</sup>, Prashasti Kanikar<sup>#4</sup>

Mukesh Patel School of Technology Management & Engineering,  
NMIMS University V. L. Mehta Road, Vile Parle (W), India

## Abstract

A novel algorithm for extracting the regions of interest (ROI) from face images followed by subsequent stitching of the extracted ROIs of various faces to form a face image is presented in this paper. The final face resulting from the fusion of the original face with exogenous features lacks the characteristic discontinuities that would have been expected if only a replacement operation was performed. The novelty of the algorithm comes from the fact that the approach is not only simple to implement and understands, but it gives near perfect results.

To achieve its task, the system uses five modules: face detection, feature detection, segmentation, shifting and blending.

## Keywords

**Feature extraction, human facial proportions, matlab functions, rgb color space, segmentation and ycbcr color space**

## 1. Introduction

Motivated by reports of increased self-confidence and driven by shifting cultural values, number of aesthetic maxillofacial plastic surgeries [1] has increased dramatically in the past few decades. This system offers patients and doctors an innovative way to visualize the outcomes of the procedure. Doctors may then provide patients with a before-and-after of the procedure helping patients significantly with their decision making process.

In this work, we solve the classical problems of face finding and feature detection; our system addresses and solves the problem of fusing a face with facial features belonging to another individual possible of different color. This is required in order to achieve a natural looking result.

Blending and mosaicing features require a two level processing. On one hand, it is required that the shape of the original shape be changed to that of the desired object, and on the other hand it is desired that the color of the replaced feature be same as the original feature.

The paper is organized as follows. the basic concept is explained in Sec. II. The Sec. III describes the approach used for face detection, feature detection and extraction.

This sec.III.A also describes the normalization of color concept proposed by us. Sec. IV discusses the proposed approach for normalization of size and color shades. Sec. V shows the results. Finally, we present the conclusion in the Sec. VI.

## 2. Basic Concepts

Figure 1 shows the block diagram representing the system where:-

O: Original Face Image

D: Desired Face Image

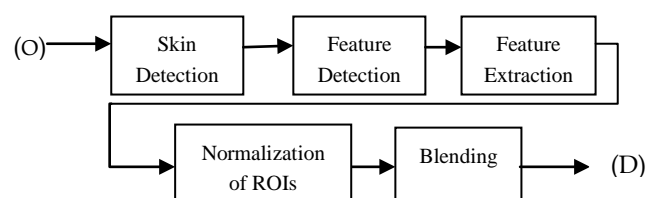


Fig. 1 Block diagram of proposed system.

The first step is to detect faces in the input images. The face detection module achieves this goal by using a skin detection module and the skin color of normalized to a certain shade. Once the face is detected, the feature detection module finds the required regions of interest and then these regions are extracted by the extraction module. The normalization module resizes the size of the ROIs. The blending module achieves smooth blending of the regions extracted such that the results look natural.

## 3. Related Work

Segmentation [1] is subdividing an image into its constituent regions or object. The level up to which the subdivision is carried out depends on the problem being solved.

In our approach, segmentation involves segregating the face into three regions which are:-

1. Eyes and the forehead
2. Nose and the cheeks
3. Lip and the chin

This is achieved by using the functions explained in the following sections.

The first 3 modules have been explained in detail in [1]. The overview is as follows.

### A. Skin Detection

As shown in [1], each pixel in the inputted image is checked based on the below equations.

If the pixels satisfy the condition then it's a skin pixel else it's not a skin pixel.

The most noticeable range which was used by algorithm to detect the skin for H (hue) value is:

$$0.01 \leq H \leq 0.1 \quad (1)$$

YCbCr space segments the image into a luminosity component and chrominance components. After experimenting with various thresholds the best result were found by using the following rule for detecting the skin pixel:

$$100 \leq Cb \leq 110 \quad (2)$$

$$140 \leq Cr \leq 150 \quad (3)$$

For each pixel of the image calculate:

And hue value of each pixel can be found by the built-in function `rgb2hsv()`.

We try to normalize the color using the following approach:

Normalization of Color

The algorithm of this step is as follows:

We want the image to have the same color as this average RGB value.

Consider:

Red=210; ... (R)

Green =140; ... (G)

Blue=100; ... (B)

1. Consider only the skin pixels.  
Calculate average of R, G, and B of the image  
Consider avg\_R as the average Red  
Consider avg\_G as the average Green  
Consider avg\_B as the average Blue
2. Subtract the difference between the R, G and B values stated above and the avg\_R, avg\_G and avg\_B in each skin pixel of the image.
3. If the difference between R and avg\_R:
  - is positive then subtract the difference with the avg\_R and allocate that value to the pixel.
  - is negative then add the difference with the avg\_R and allocate that value to the pixel
 Do the same to G and avg\_G.  
Do the same to B and avg\_B.

Hence the image is normalized to a particular skin pixel color.

Without color



With color



Fig. 2 Result of Normalization of Skin

#### B. Feature Detection[1]

1. Convert the output of skin detection 'image' into black/white form.
2. Find the 1st white point
  - (a) From left to right side of image.
  - (b) From right to left side of image.
3. Compute midpoint

Notations used:

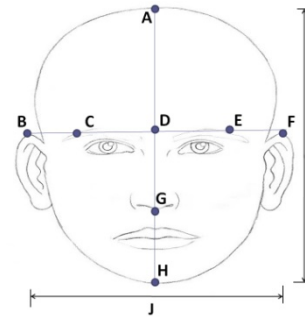


Fig. 3 Coordinates representation

Where:

A → head\_x, head\_y

F →

right\_xvalue, right\_yvalue

B → left\_xvalue, left\_yvalue

G → nose\_x, nose\_y

C → left\_eye\_x, left\_eye\_y

H → tip\_x, tip\_y

D → mid\_x, mid\_y

I → length\_face

E → right\_eye\_x, right\_eye\_y

J → length\_across\_face

We need to find (refer fig. 3)

- (a) Eye coordinate (C and E)
- (b) Face top coordinate (A)
- (c) Face tip coordinate (H)
- (d) Nose coordinate (G)

The following is the pictorial explanation of the concept used for feature detection.

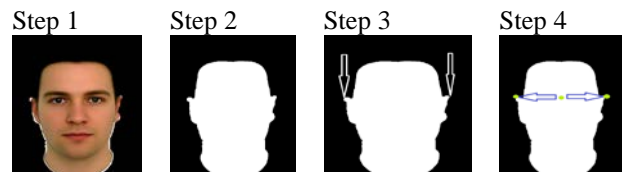


Fig. 4 Graphical representation of the feature detection

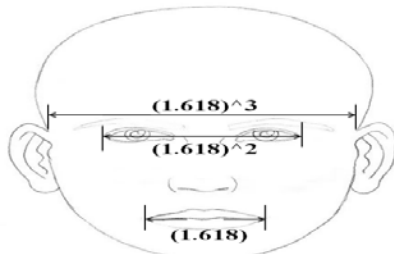


Fig. 5 Horizontal Proportion

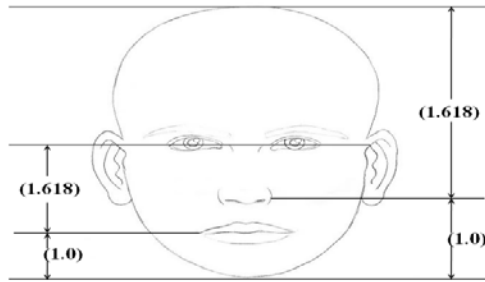


Fig. 6 Vertical Proportion

The eye points can be detected using the horizontal proportion as the coordinates B and F are known. Hence, after the calculations, the coordinates C and E are known.

From the vertical proportions, we can find coordinates A and H. the nose point G can also be computed from this vertical proportion as shown in fig. 6..

### C. Feature Extraction

We now proceed to the next step of extracting the given feature from the face [1].

The algorithm for the feature extraction:

Refer to fig.3 for following coordinates.

Start

1. Blacken image till midpoint i.e. mid\_x.
2. Blacken the eye parts.
3. Take feature extraction point. Save image as 'head less.jpg'
4. Blacken the face till the nose\_yvalue. Save the image as 'bottom.jpg'
5. Compute the top part image i.e. (temp\_img-headless). Save as 'top.jpg'
6. Compute middle part image i.e. middle = headless-bottom. Save as 'middle.jpg'

End

The following is the pictorial explanation of the concept used for feature extraction.

Step 1                      Step 2                      Step 4

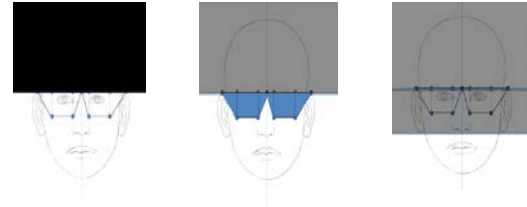


Fig. 7 Graphical representation of the feature extraction.

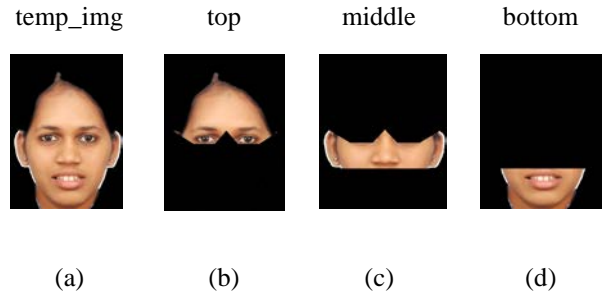


Fig. 8 (a) shows the output of the skin detection module. (b) shows the top.jpg image. (c) shows the middle.jpg image and (d) shows the bottom.jpg image.

### I. Proposed Work

#### A. Normalization of Size

Length across the face can be computed by the ear coordinates i.e. B and F (fig. 3).

$$l = F(x \text{ value}) - C(x \text{ value}) \quad (4)$$

where l=length across face  
similarly b is also known.

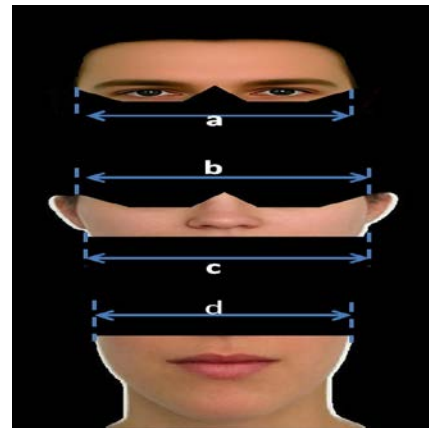


Fig.9 Length across Face of Top and Middle and bottom

Known variables:

- Length across face of top part=a
- Length across face of middle part=b
- Length across face of middle part (to resize with bottom)=c
- Length across face of bottom part=d

### Resizing the Middle

We size the face parts, according to the top part length.

2 cases arise:-

- Case 1: If middle part is greater than top part.
  - Compute the difference between their lengths.
  - Decrease the size (columns) of the middle part by that much difference.
- Case2: If middle part is smaller than top part.
  - Compute the difference between their lengths.
  - Increase the size (columns) of the middle part by that much difference.

### Resizing the Bottom

1. Compute new length of middle and bottom.

2. Two cases arise:-

2 cases arise:-

- Case 1: If bottom part is greater than middle part.
  - Compute the difference between their lengths.
  - Decrease the size (columns) of the bottom part by that much difference.
- Case2: If bottom part is smaller than middle part.
  - Compute the difference between their lengths.
  - Increase the size (columns) of the bottom part by that much difference.

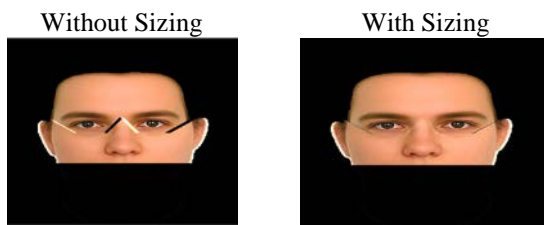


Fig. 10 Example of sizing

Compute the difference between the top and middle parts 'x and y coordinate.

Cases arise:-

Case 1: Top part is lower than middle part.

Compare the difference between the y coordinates. If the y coordinate of the top part is greater than the middle part. Shift the rows of the part by that much difference.

Case 2: Middle and Top part have a lot of distance.

Compare the difference between the y coordinates. If the y coordinate of the middle part is greater than the top part. Shift the rows of the part by that much difference.

Case 3: Middle part is more towards the left than the top part.

Compare the difference between the x coordinates. If the x coordinate of the top part is greater than the middle part. Shift the columns of the part by that much difference.

Case 4: Middle part is more towards the right than the top part. Compare the difference between the x coordinates. If the x coordinate of the middle part is greater than the top part. Shift the columns of the part by that much difference.

Shift the columns of the part by that much difference.

- These same cases arise between the middle and bottom part.

These same cases arise between the middle and bottom part.

Without Positioning

With Positioning



Fig. 11 Example of sizing

### B. Positioning the 3 Parts:

Known variables:

From fig. 3(coordinates), we know left ear coordinates i.e B.

Consider the top parts left ear coordinates as follows:

- a) Top values of left.
  - i.e left\_xvalue\_top , left\_yvalue\_top.
- b) Middle values of left.
  - i.e left\_xvalue\_middle , left\_yvalue\_middle.
- c) Bottom values of left.
  - i.e left\_xvalue\_bottom , left\_yvalue\_bottom.

Consider only top and middle parts:

### C. Color Shades

#### D. To change the shade of the face image:

Just change the Cb and Cr values:

Cb=105;	Cb=105;	Cb=99;	Cb=95;
Cr=163	Cr=163	Cr=160	Cr=160



Fig. 12 Example of Color shading

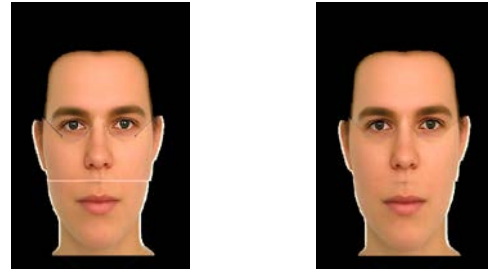


Fig. 13 Example of Blending

### E. Blending

The new face generated should be as natural as possible with the least seam and discontinuity. For this purpose blending needs to be performed.

The blending technique involves superimposing the three different layers. The images of the three different sections are added to with each other. The image so obtained contains the image of the generated face which is approximately the required image of the face. However it is not completely natural. This is because the resultant image may contain overlapping regions or gaps. Both these problems need to be addressed. While superimposing the images there are gaps left between the lines of contact. It is important to fill these gaps. This filling of the gaps is achieved by performing 'closing' on the image along the lines where the merging of the different sections takes place. Closing takes place as follows:

$$A \bullet B = (A \oplus B) \ominus B \quad (5)$$

Where the ' $\bullet$ ' indicates closing and ' $\oplus$ ' and ' $\ominus$ ' indicate dilation and erosion respectively. A is the image on which dilation needs to be performed which is the resultant image obtained from the superimposition of the three layers. B is the structuring element that will be used to achieve the purpose. Thus closing is a result of dilation followed by erosion on image A by structuring element B. The other problem that needs to be addressed is the problem of overlapping as a result of the superimposition. Since we are employing the addition method the resultant image may contain overlapping regions. Hence we need to take care that addition is not performed at overlapping regions so as to avoid creating seams at mosaicing area which would otherwise become very bright.

Without Blending

With Blending

## 4. Results



Fig. 14 Results

## 5. Conclusion

In this paper, we have presented a novel algorithm for extracting the regions of interest from frontal face images.

At the same time, we describe details of a system that achieves fusion of different face segments of different individual - possibly of different skin color. The resulting face looks natural and gives near-perfect result.

The possible applications where our approach could be applied are cosmetic surgery to extract features from face and also by detective agencies to identify people based on eye-witness accounts which is a good replacement to the manual technique now used.

## References

- [1] Akshay Kajekar, Shreyash Patel, Sweta Chauhan, Prof. Prashasti Kanikar, Extraction of Face Segments based on Proportion, Vol. 3, Issue. 5, pp. 2-5, May 2012.
- [2] Kuang-chih Lee, Dragomir Anguelov, Baris Sumengen and Salih Burak Gokturk, Markov Random Field Models for Hair and Face Segmentation.
- [3] Diedrick Marius, Sumita Pennathur, and Klint Rose, Detection Face Detection Using Color Thresholding, and Eigen-image Template Matching

- [4] Dr. Arti Khaparde, Sowmya Reddy.Y Swetha Ravipudi, Face Detection Using Color Based Segmentation and Morphological Processing –A Case Study
- [5] C.J. Veenman, M.J.T. Reinders, and E. Backer, A Cellular Coevolutionary Algorithm for Image Segmentation, Vol. 12, No. 3, pp. 304-316, March 2003.
- [6] Goin MK, Rees TD, A prospective study of patients' psychological reactions to rhinoplasty, *Annals of Plast Surgery*, 27(3):210, 5, Sep 1991.
- [7] S.Ravi and S.Wilson, Face Detection with Facial Features and Gender Classification Based On Face Detection with Facial Features and Gender Classification Based On Support Vector Machine, 2010 Special Issue - International Journal of Imaging Science and Engineering
- [8] Sanjay Kr. Singh, D. S. Chauhan, Mayank Vatsa, Richa Singh, A Robust Skin Color Based Face Detection Algorithm, *Tamkang Journal of Science and Engineering*, Vol. 6, No. 4, pp. 227-234 (2003)
- [9] Arindam Biswas, Suman Khara, Partha Bhowmick, Extraction of Regions of Interest from Face Images Using Cellular Analysis, Science and Technology Dept.
- [10] The golden proportion can be found on the site. '[http://majorityrights.com/weblog/comments/the\\_facial\\_proportions\\_of\\_beautiful\\_people](http://majorityrights.com/weblog/comments/the_facial_proportions_of_beautiful_people)'.
- [11] Gonzalez R. Woods. E., Digital Image Processing Using Matlab (Prentice Hall, Second Edition, 2002
- [12] Rudra Pratap, Getting Started With Matlab 7
- [13] Gonzalez R. Woods. E., Digital Image Processing (Prentice Hall, Second Edition, 2002)