International Journal of Engineering Research and Advanced Technology (IJERAT)

DOI: 10.31695/IJERAT.2020.3644

E-ISSN : 2454-6135

Volume.6, Issue 8
August -2020

Human Ear Print Detection Algorithm

Raniah Ali Mustafa[1] Haitham Salman Chyad[2] & Dena Nadir George[3]

[1-3]College of Education, Department of Computer Science

Mustansiriyah University, Iraq

ABSTRACT

In this paper, proposed detection algorithm for human ear print images, the algorithm consist of three-stage. the first stage The detection algorithm using HSV color space, canny algorithm and contrast enhancement for grayscale. This step aims to determine skin area in-ear image by first HSV color space converting RGB to HSV color space and applying certain rules to determine the skin area. the second stage applies skin ear segmentation for the split of the skin and non-skin areas where ear skin color detection. After the ear detection stage, the first stage in edge detection is image smoothing through using a Gaussian filter then converted to a grayscale image after then contrast enhancement is an important step in the algorithm detection ear. Finally applying Canny edge detection, in general, is to significantly reduce the amount of data in an image, while protecting the main structure to be utilized for further image processing. were obtained from the dataset, available in the Internet and detection algorithm implemented in programing language Visual Basic 6.0.

Key Words: Ear Detection, HSV Color Space, Gaussian Filter, Contrast Enhancement, Canny Edge Detection.

1. INTRODUCTION

EAR images can be acquired in a similar manner to face images and a number of researchers have suggested that the human ear is unique enough to each individual to al-low practical use as a biometric. Several researchers have looked at using features from the ear's appearance in 2D in-tensity images [1], whereas a smaller number of researchers have looked at using 3D ear shape [2].

"When you're born your ear is fully formed. The lobe de-scends a little, but overall it stays the same. It's a great way to identify people," said Mark Nixon, a computer scientist at the University of Southampton. and leader of the research, "There's real power in using the appearance of an ear for computer recognition, compared to facial recognition. It's roughly equivalent if not better," said computer scientist Kevin Bowyer of Notre Dame, who is pursuing his own ear recognition technology and not involved with Nixon's work. "If you've got a profile image for someone, this is a great way to use it."

Ear biometric has played an important role for many years in forensic science and its use by law enforcement agencies. Among various physiological biometric traits, ear has received much attention in recent years as it has been found to be a reliable biometrics for human recognition [2].

2. BIOMETRICS TECHNIQUES

The biometric techniques as demonstrated in Figure (1.1) can be classified into two types as indicated by the attributes that are used:

- Biometric Techniques Based on Physiological Characteristics

They include all biometrics methods which are based on straightforward computation on a specific part of person body, iris, face, fingerprint, Ear, hand and DNA are the mostly known and popular kinds. These kinds are reliable more than kinds depend on behavior [3].

- Biometric Techniques Based on Behavioral Characteristics

Depending on the behavior or action of person, this type of methods extracts attributes; the most prominent and successful kinds are voice, keystroke and signature, in this type of biometric; the time is used as measure to determine person attribute [3].

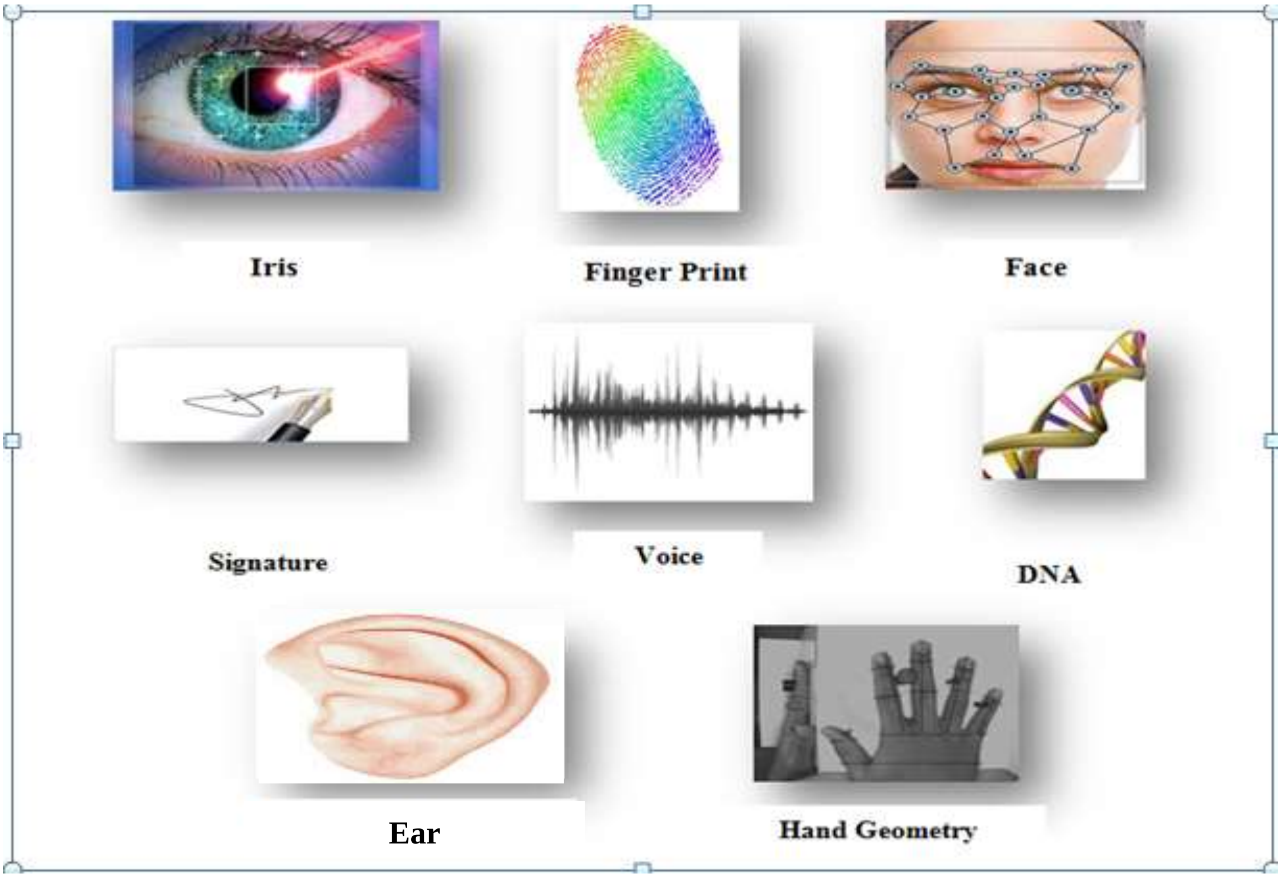


Figure 1.1: Biometric Technologies

3. HISTORY OF USING EAR BIOMETRICS

The term of biometrics is derived from Greek word bio (life) and metric (to measure). Among the first known examples of practiced biometrics was a form of member-printing used in China in the fourteenth century. The Chinese merchants were stamping children's palm and footprints on paper with ink to distinguish one baby from another. Biometrics use has been traced all the way back to the Egyptians. Thousands of years ago, they used the basic principles of biometric verification in everyday routine business situations. There are references to individuals being identified through unique physiological characteristics such as scars, body measurement, complexion, eye color, and height. This would often occur in agricultural transactions where grain and other necessities would be supplied to a central storage area, and in various legal proceedings.

In the 1890's, an anthropologist and police desk clerk in Paris named Alphonse Bretillon sought to fix the problem of identifying convicted criminals and tuned biometrics into distinct field of study. He developed a method of multiple body measurements that was named after him (the Bertillonage technique-measuring body length). Police throughout the word used this system until it proved to be exceedingly prone to error as many people shared the same measurements. After this failure, the police started using fingerprinting- developed by Richard Edward Henry of Scotland Yard- after the methods used by the Chinese centuries before [4]. Law enforcement has been using fingerprints as a method of identification since the 1865's. The first biometric system was developed in 1960. The question of uniqueness and classification of ears already has some history, although not always in a forensic context. Imhofer (1906) already found that in set of 500 ears he only needed 4 characteristics to uniquely distinguish them. Some other references on ear identification, although some more connected to either ear prints or research to in heredity factors are covered this field in some way. They are: Hirschi (1970), Oepen (1976), Rother (1976), Hunger and Hammer (1987), Hammer und Neubert (1989). The most prominent is by Iannarelli (1989), he considered over 10.000 ears and found no indistinguishable ears. Most of the above literature indicates that the variability between ears is large that it might be possible that ears are unique, and moreover possibly uniquely distinguishable on a limited number of features or characteristics. During the past decades in several countries, like The Netherlands, United Kingdom, Germany, Austria and the United States, already a substantial number of cases involving evidence based on ear prints and some isolated cases of video images of ears have appeared in court and in some cases accepted as evidence [4].

4. EAR DETECTION

The external ear structure, known as the pinna, is used in biometric recognition system. Its structure is relatively simple, but it varies significantly across individuals. The ear is made up of standard features including the helix, the antihelix, the lobe, and the u-shaped intertragic notch between the ear hole and the lobe [1-2]. Object detection is an important task in computer vision that aims to find instances of semantic objects belonging to searched classes (such as humans, buildings, or cars) in digital images and videos. This task involves extract-ing the regions of pixels in the image where the searched ob-jects are present. Object detection has many applications in areas of computer vision (e.g. image retrieval) and it is a cru-cial stage for further recognition processes. The usage of some contextual information about the searched object (e.g. its shape or color) is very useful to automatically detect it. The ear pattern is composed of several circles (some of them are con-centric) that form the outer and inner borders of the helix and antihelix ear regions. This morphological feature is a characteristic from ears [5-7].

5. CONVERT TO HSV COLOR SPACE

Hue Saturation value (HSV) color space is one of RGB color space transformation; it is also named Hue Saturation Brightness (HSB). The main characterize this color space is presents the property of color numerical. Hue represents the tint of color, saturation represents the shade and value represents the luminance. Hue describes the preeminent color like primary Red, Green and Blue; or secondary colors like Magenta, Yellow and Cyan, as explained in Figure (1.2). Equations (1), (2), and (3) represent transformation rules to obtain the (H, S, V) values from RGB color space [8].

$$H= \begin{cases} (\frac{G-B}{Max-Min})/\ 6 & if\ R = Max \\ (2+\frac{B-R}{Max-Min})/\ 6 & if\ G = Max \\ (4+\frac{B-R}{Max-Min})/\ 6 & if\ B = Max \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots(1)$$

$$S = (\frac{Max-Min}{Max}) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2)$$

$$V= Max \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3)$$

Where

R: Red value.

G: Green value.

B: Blue value.

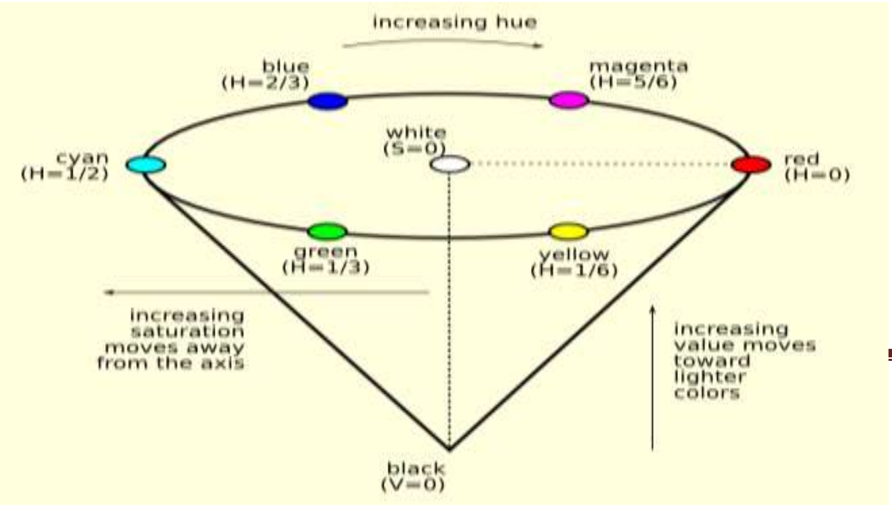


Figure 1.2: HSV color space

6. IMAGE SMOOTHING BASED ON GAUSSIAN FILTER

This type of filter is used for two objectives, first noise reducing and second for blurring. Blurring is used as preprocessing step; such as for removal of small details from an image prior to object extraction, also, for bridging of small gaps in lines or curves. Gaussian filter is a two dimensional (2-D) convolution operator. The form of Gaussian as follow [9]:

$$f(x,y) = \frac{1}{2\pi\sigma^2}\ \exp(-\frac{x^2+y^2}{2\sigma 2})\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4)$$

Smoothing degree is depending on the value of sigma of Gaussian form. [9].

6.1 CONVERT TO GRAY-SCALE IMAGE

The gray image can be used to describe the spatial distribution of light intensity (or brightness). Brightness value is the only constituent of image in gray-scale. In ideal case, each pixel value represents by 8 bits from 0 to 255. To compute the gray value from Red, Green, and Blue color components we can use one of the following equations [10]:

$$F(x, y)= 0.3R +0.59G + 0.11B \ldots\ldots\ldots\ldots..(5)$$

$$F(x,y)=\frac{1}{3}\ (R+G+B)\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots..(6)$$

$$F(x,y)= \text{Thresholds}\ (\frac{R}{255}+\frac{G}{255}+\frac{B}{255})\ldots\ldots\ldots\ldots\ldots\ldots..(7)$$

Where R, G, B are the red, green, and blue color components respectively; F(x, y) is the corresponding gray value.

6.2 CONTRAST ENHANCEMENT

The important aim of this technique is to process a given image so that the output is convenient for the next step. If many number of picture elements tacking limited place that means the image is poor contrast. By doing modification to the histogram, each pixel will modify to have new value which is increased occupying range of brightness levels. **Contrast Stretching** is one of image enhancement technique that increases the occupying range of brightness levels. The principal idea of contrast stretching is to stretch the distribution of brightness value to occupying all the range of the histogram. The calculation of the specific interval is selected by determining minimal and maximal intensity thresholds values [10]. The intensity range, Img(), is stretched to the full range [0-255]. The following linear mapping function is utilized to compute the new pixel's brightness values:

$$\mathrm{Img}(x,y) = \mathrm{round}\left(\frac{(\mathrm{img}(x,y) - \mathrm{Min})}{(\mathrm{Max} - \mathrm{Min})}\right)^{\alpha * 255} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(8)$$

where

Min: the minimum intensity value of the input image

Max: the maximum intensity value of the input image

α: the sigma value within the range [3, 10].

6.3 CANNY EDGE DETECTION

The main objective of applying edge detection filter is decreasing object's amount of data in an image, with preserves fundamental structural of that object to be used for further image analysis. Canny edge detection has become one of the standard edge detection methods, its algorithm runs in 4 separate steps [11]:

First step: smoothing the image (Gaussian filter is used for this purpose).

Second step: compute intensity gradient of the image (applying two convolution masks X, Y directions and find the gradient).

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (9)$$
$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$$

$$G = \sqrt{G_x^2 + G_y^2} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (10)$$

Third step: *Non-maximum* suppression which means labels only the pixels that are considered to be part an edge.

Fourth step: *Hysteresis which means* choose minimum and maximum thresholds, accepted the pixel gradient that is between these two thresholds.

7. THE PROPOSED SYSTEM

The main idea in our approach for ear detection is based on using HSV color space, canny algorithm and contrast enhancement for gray scale. In this paper, each phases in the proposed detection algorithm will explained with the results. The figure (1.3) illustrates each phases for the proposed system.

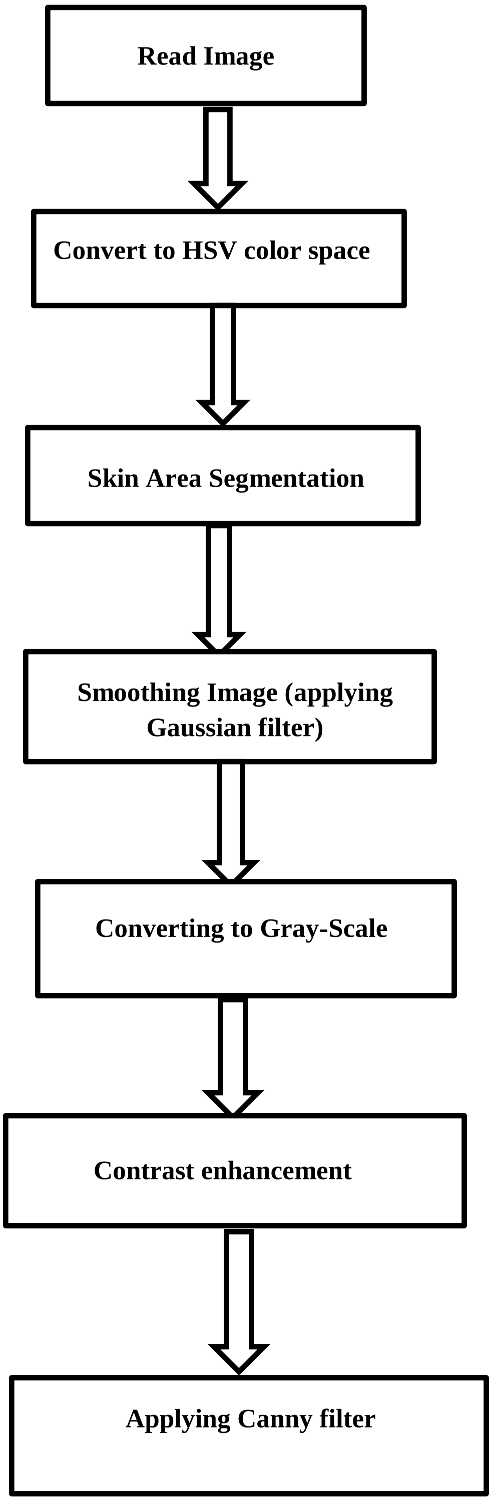


Figure 1.3: The detection algorithm for ear Images

7.1 READ IMAGE

DOI : 10.31695/IJERAT.2020.3644

The color ear image is fed to the system as a BMP image file; the color resolution of the image is taken as 24 bit/pixel. When BMP image file is loaded, the image header information is being read as first, then, the bitmap pixel data will be read. The three bands: red, green and blue are loaded and put into 3D array with record. Figure (1.4) presents the array. The algorithm (1.1) presents the applied steps for image loading task.

No Pixels	Row/ Column (x,y)	Z (Preprocessing on x and y)							
		Value			HSV				
		Red	Green	Blue	H	S	V	SkinHSV	Contrast
1	1,1								
	1,2								
	…								
2	2,1								
	2,2								
	…								

Figure 1.4: Example of 3D array contents

where

X: row pixel image, the value of x (0 to Image Width-1)

Y: column pixel image, the value of y (0 to Image High -1)

Z: containing the values of three bands also all preprocessing on x and y (the value of Red, Green, Blue, HSV, SkinHSV, …etc).

Algorithm (1.1) Read BMP Image
Goal: Convert a bitmap image to 3D arrays with recode.

```
Input: ImgFile // image file
Output:  Wid,Hgt // image's width and height
             ImagePixels(0,0 to Wid-1, 0 to Hgt-1).value   // Red component of image
             ImagePixels(1,0 to Wid-1, 0 to Hgt-1).value  // Green component of image
             ImagePixels(2,0 to Wid-1, 0 to Hgt-1).value  // Blue component of image

Step 1: Get width and height values from BMPH // BMPH is the BMP Header
          Wid ←BMPH.Width  // Get image's width and height values from its header
          Hgt ←BMPH.Height
Step 2: Check image pixel resolution
        If BMPH.BitPlan = 24 Then
             Set I ←0
             For all X, Y Do {where 0 to Wid-1, 0 to Hgt-1}
             Set ImagePixels(0,X, Y).value ←Img(I+2)
             Set ImagePixels(1,X, Y).value ←Img(I+1)
             Set ImagePixels(2,X, Y).value ←Img(I)
           Increment I by 3
End For
Return (Wid, Hgt, ImagePixels)
Else
Display message "The selected Image's BitPlan improper"
End If
End.
```

7.2 CONVERT TO HSV COLOR SPACE

In RGB color space too great changing in color values might not be detected by a human observer. Therefore, this color space is not suitable for skin detection, for this reason, HSV color space which is suitable for representing human skin color as mentioned in section (5), has been applied in the proposed system. This step aims to determine skin area in ear image by first converting RGB to HSV color space and applying certain rules to determine the skin area. algorithm (1.2) shows the implemented steps.

```
Algorithm (1.2) HSV Color Space Transformation

Goal: Conversion of RGB color space to HSV color space.
Input:  ImagePixels(0,0 to Wid-1, 0 to Hgt-1).value   // Red component of image
        ImagePixels(1,0 to Wid-1, 0 to Hgt-1).value  // Green component of image
        ImagePixels(2,0 to Wid-1, 0 to Hgt-1).value  // Blue component of image

Output:ImagePixels(0,0 to Wid-1, 0 to Hgt-1).HSV // H component of image
        ImagePixels(1,0 to Wid-1, 0 to Hgt-1).HSV // S component of image
        ImagePixels(2,0 to Wid-1, 0 to Hgt-1).HSV // V component of image

Step 1: Get the RGB three band value, apply certain rule to obtain HSV space
    For all X, Y Do {where 0 to Wid-1, 0 to Hgt-1}
      Set R1 ← ImagePixels(0,X,Y).Value
      Set G1← ImagePixels(1,X,Y).Value
      Set B1← ImagePixels(2,X,Y).Value
      Set  Max ← iMax(iMax(R1, G1), B1)
      Set  Min ←iMin(iMin(R1, G1), B1)

       If  R1 = Max  then
            Set ht ←($\frac{G1 - B1}{Max - Min}$)

       Else
         If G1 = Max  then
              Set ht ←$2 + (\frac{B1 - R1}{Max - Min})$
         Else
              Set ht ←$4 + (\frac{R1 - G1}{Max - Min})$
           End if
       End if
     End If
          Set ht ←$\frac{ht}{6}$
     If (ht < 0) Then

       $ht = ht + 1$
      End If
     Set ImagePixels(0,X,Y).HSV ← [ht ]
    Set imagepixels (1,X,Y).HSV ← $\frac{Max-Min}{Max}$
    Set imagepixels(2,X,Y).HSV←Max
End For
Return( ImagePixels)
End.
```

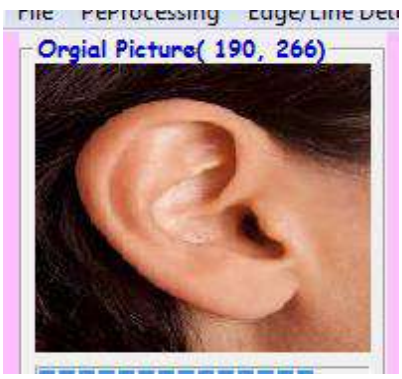
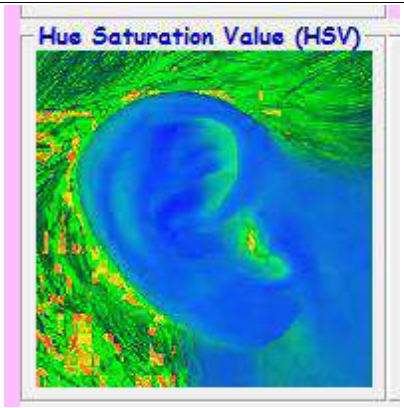
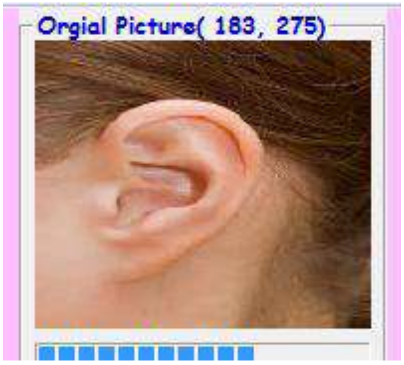
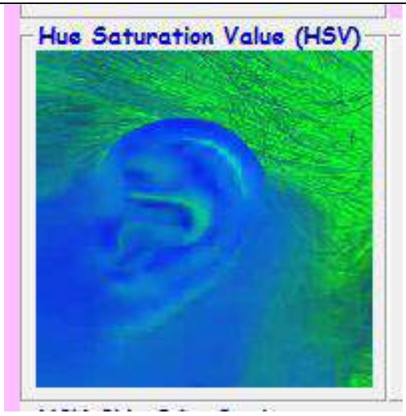
 	
Right Profile color ear Image	After convert to HSV
 	
Left Profile color ear Image	After convert to HSV

Figure 1.5: Convert color Image to HSV Color Space

7.3 SKINE EAR SEGMENTATION

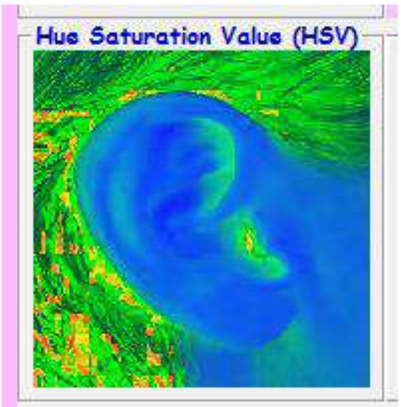
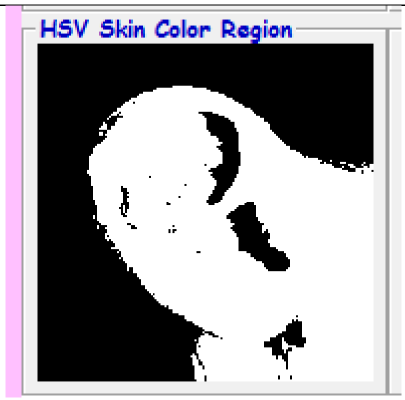
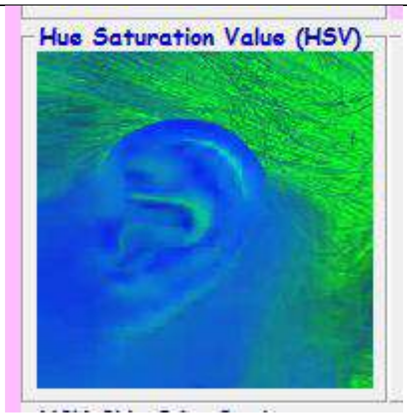
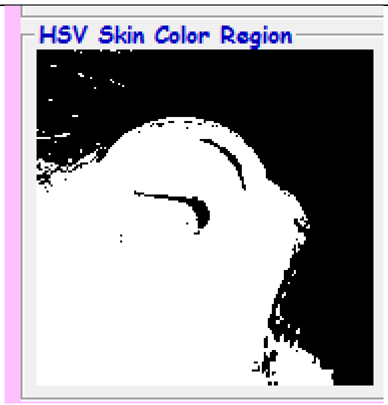
The blue values exhibit the most detectable split of the skin and non-skin areas as shown in Figure (1.5), The skin in band H is characterized by values between 0 and 40, in the band S from 30 to 160 and in the band V from 150 to 255. Make scan for all image pixels, each pixel that is achieved in equation (3.1) is classified as a skin color pixel and set to value 255, and non-skin pixels are fixed to 0 Figure (1.6) shows examples of ear skin color detection, algorithm (1.3) explain the implemented steps.

$$\begin{aligned}&(H > 0)\ and\ (H < 40)\ and\\&(S > 30)\ and\ (S < 160)\ and\\&(V > 150)\ and\ (V < 255)\end{aligned} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(11)$$

DOI : 10.31695/IJERAT.2020.3644

Algorithm (1.3) Skin Area Segmentation
Goal: DetectionSkin area.

```
Input: ImagePixels(0,0 to Wid-1, 0 to Hgt-1).HSV // H component of image

       ImagePixels(1,0 to Wid-1, 0 to Hgt-1).HSV // S component of image
       ImagePixels(2,0 to Wid-1, 0 to Hgt-1).HSV // V component of image

Output: ImagePixels(4 ,0 to Wid-1, 0 to Hgt-1).SkinHSV // Skin Color

Step 1: Apply a range rule to obtain skin HSV area
    For all X, Y Do {where 0 to Wid-1, 0 to Hgt-1}
        If (ImagePixels(0, X,Y).HSV > 0) And (ImagePixels(0, X,Y).HSV < 40) And (ImagePixels(1, X,Y).HSV>30)
        And (ImagePixels(1, X,Y).HSV<160) And (ImagePixels(2, X,Y).HSV > 150) And (ImagePixels(2, X,Y).HSV < 255) Then
            ImagePixels(4, X,Y).SkinHSV ← 255
        Else
            ImagePixels(4, X,Y).SkinHSV ← 0
        End If
End For
Return (ImagePixels)
End
```

	
Right HSV Profile ear Image	Skin Area Segmentation
	
Left HSV Profile ear Image	Skin Area Segmentation

DOI : 10.31695/IJERAT.2020.3644

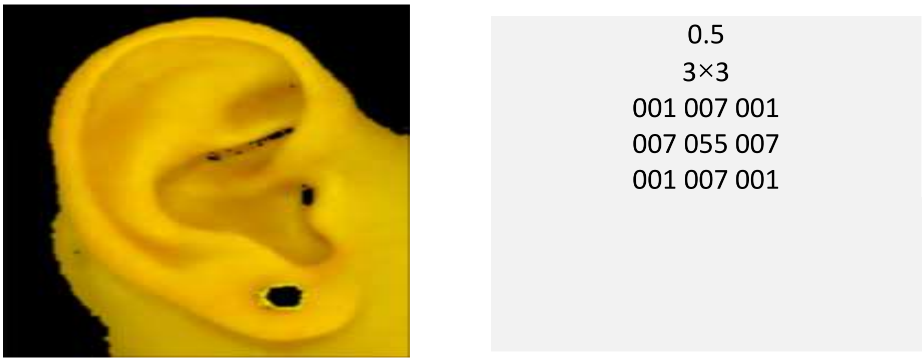
Figure 1.6: Skin Area Segmentation

8. EAR DETECTION STAGE

The main purpose of this stage is to detect the main edge in ear image. This stage includes three steps, image smoothing, convert to gray- scale, contrast Gray scale and applying canny edge detection.

8.1 EAR IMAGE SMOOTHING STEP

After ear detection stage, first stage in edge detection is image smoothing. The purpose of this stage is to reduce the existing noise and improve the shape of ear details. Gaussian filter as expressed in equation (4) is used in this step of proposed detection ear. The sigma which is equal to (0.8) and kernel size (3x3) have been chosen in the proposed system. Figure (1.7) explains application of Gaussian filter with kernel size (3x3) and sigma value (0.8) with its array value that are produced for case. The smoothing step is processed to achieve better edge extraction results.



0.5
3×3
001 007 001
007 055 007
001 007 001

Figure 1.7: The applying Gaussian filter with sigma value

8.2 CONVERT TO GRAY SCALE SUB STAGE

The colored image of ear will be converted to gray scale. Each colored input will be converted to gray scale image (this operation will convert 24-bit/pixel images to 256 gray scale image), the equation (7) is used in this step of proposed detection ear. The figure below explains the result of applying threshold value in the range (127) with the sigma value equal to (0.8) of Gaussian filter to obtain best results as mentioned in algorithm (1.4).

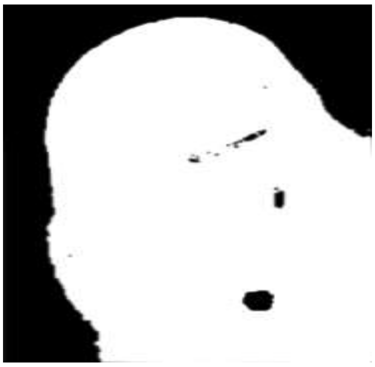


Figure 1.8: Ear Clip after converting to Gray-Scale

```
Algorithm (1.4) Grayscale
Goal: Convert to Grayscale.

Input:  ImagePixels(0,0 to  C_F_ Wid-1 , 0 to  C_F_ Hgt-1).Gaussian  //
        Red output of  Gaussian image
        ImagePixels(1,0 to  C_F_ Wid-1 , 0 to  C_F_ Hgt-1).Gaussian   //
        Green output of Gaussian image
        ImagePixels(2,0 to  C_F_ Wid-1 , 0 to  C_F_ Hgt-1).Gaussian  //
        Blue output of Gaussian image

Output: ImagePixels(6,0  C_F_ Wid-1 , 0 to  C_F_ Hgt-1 ).Grayscale //
        Grayscale component of image

Step 1: Convert to Grayscale.
     Set Threshold←140
For all X, Y Do {where 0  to  C_F_ Wid-1 , 0 to  C_F_ Hgt-1}
      Set Red ←ImagePixels(0,X, Y).Gaussian
      Set Green←ImagePixels(1,X, Y).Gaussian
      Set Blue←ImagePixels(2,X, Y).Gaussian
     Set ImagePixels(6,X,Y).Grayscale← $Threshold * (\frac{Red}{255} + \frac{Green}{255} + \frac{Blue}{255})$
  End for
Return (ImagePixels(6,0 to C_F_ Wid-1, 0 to C_F_ Hgt-1).Grayscale).
End.
```

8.3 CONTRAST ENHANCEMENT STEP

Contrast enhancement is an important step in this stage, which makes the ear details more suitable for the next step of the proposed detection ear. In order to accomplish this aim, contrast stretching has been made for gray value. The applied contrast stretching comprises two steps; the first is determining the minimum and maximum threshold values. After that, linear stretching is applied; it moves the low intensity values that are less than the determined (min value) toward 0, and the high level values higher than (max value) are moved toward 255. All values which lie between min and max values are linearly mapped (using equation 7). As a result, the range of intensity levels is stretched to the full range (0-255), A look table is established for speeding up the mapping process; this table is utilized to straightforwardly change each pixel value to its corresponding new value. Once, this look table is established, each pixel is mapped into its corresponding new value directly without the need to recalculate the same equation. Algorithm (1.5) shows the applied contrast enhancement steps.

```
Algorithm (1.5) Contrast Enhancement
Goal: Convert Gray to a more usable form.

Input:  ImagePixels(6,0 to C_F_ Wid-1, 0 to C_F_ Hgt-1).Grayscale  // For ear clip
        C_F_ Wid,  C_F_ Hgt                 //Width and Height of ear clip
        Sig                                 // sigma value
        Min                                 // Minimum gray value
        Max                                 // Maximum gray value

Output:  ImagePixels(0,0 to C_F_ Wid-1, 0 to C_F_ Hgt-1).Contrast_GrayScale //  face clip after contract enhancement

 Step 1: Initialize look table (0 to 255)

    For all I Do   { where 0 to 255}
```

```
        If I < Min Then
         Set lookTable (I) ← 0
         If I > Max Then
        Set lookTable(I) ← 255
           Else  Set LookTable (I) ← ((value(i) - min) / (max - min)) ^ Sig * 255
              End For

Step 2: Compute the new values (Contrast GrayScale)
    For all X, Y Do {where 0  to  C_F_ Wid-1, 0 to  C_F_ Hgt-1 }
         If ImagePixels(6,X,Y).Grayscale=255 then
    Set  ImagePixels (0, X, Y).Contrast_GrayScale← 255
      Else if ImagePixels(6,X,Y).Grayscale=0 then
    Set  ImagePixels (0, X, Y).Contrast_GrayScale← 0
      Else ImagePixels (0, X, Y).Contrast_GrayScale←LookTable(ImagePixels (6,X,Y).Grayscale)

End For

Return  (ImagePixels(0,0 to C_F_ Wid-1, 0 to C_F_ Hgt-1).Contrast_GrayScale).
End
```

8.4 CANNY EDGE DETECTION SUB STAGE

The motivation behind edge detection in general is to significantly reduce the amount of data in an image, while protecting the main structure to be utilized for further image processing.

Figure (1.9) explains applying Canny filter after (image smoothing step and convert it to Grayscale step) with maximum and minimum thresholds values (T Max is 100 and T min is 20). Algorithm (1.6) illustrates the steps of applying Canny filter.

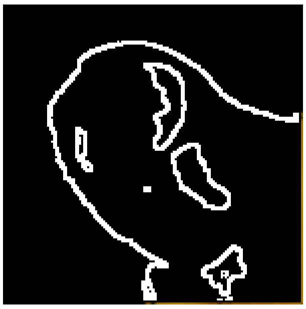


Figure 1.9: An example of applying Canny filter

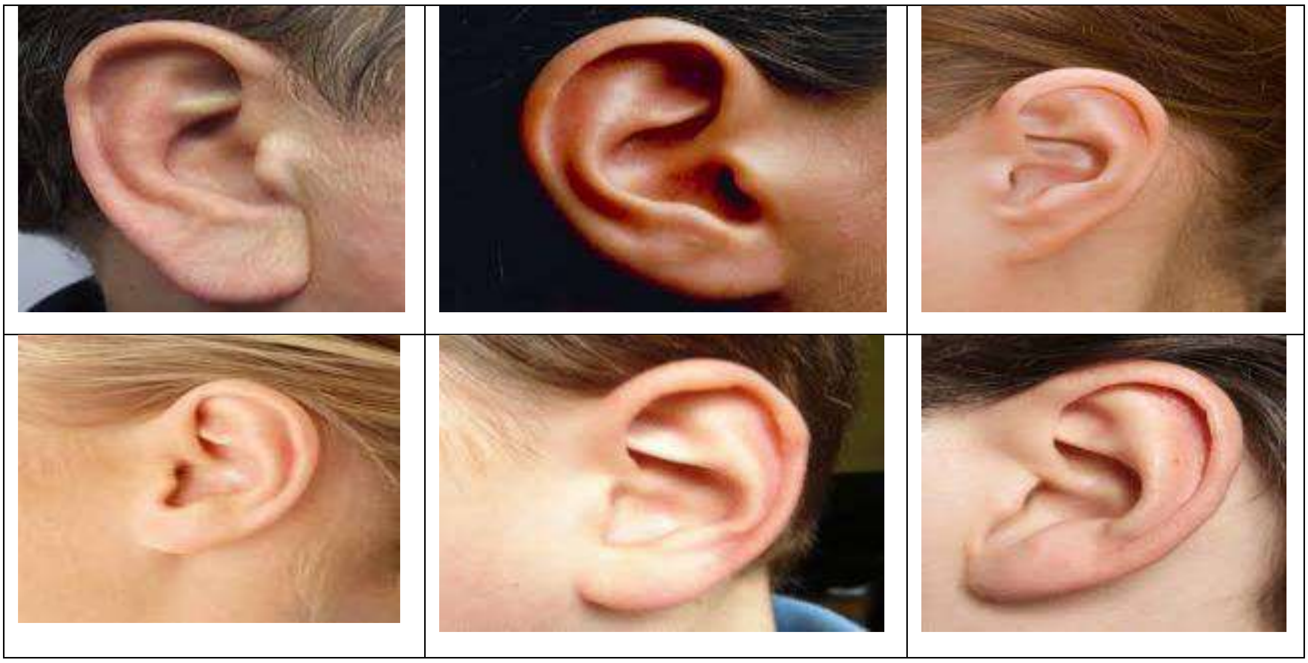
```
Algorithm (1.6) Canny Edge Detection
Goal: Edge Detection

Input:earM (0,0 to earX-1, 0 to earY-1).ContrastGrayScale   //ear
         ThresholdMin
         ThresholdMax

Output:  earM (7,0 to earX-1, 0 to earY-1).Canny Edge    // ear

Step 1: define 3*3 structuring element Op_X
        Set Op_X(-1, -1) ← -1:  Set Op_X(0, -1) ← -2:  Set Op_X(1, -1) ← -1
         Set Op_X(-1, 0) ← 0:  Set Op_X(0, 0) ← 0:  Set Op_X(1, 0) ← 0
        Set Op_X(-1, 1) ← 1:  Set Op_X(0, 1) ← 2:  Set Op_X(1, 1) ← 1

Step 2: define 3*3 structuring element Op_Y
         Set Op_Y(-1, -1) ← -1:  Set Op_Y(0, -1) ← 0:  Set Op_Y(1, -1) ← 1
          Set Op_Y(-1, 0) ← -2:  Set Op_Y(0, 0) ← 0:  Set Op_Y(1, 0) ← 2
         Set Op_Y(-1, 1) ← -1:  Set Op_Y(0, 1) ← 0:  Set Op_Y(1, 1) ← 1
Step 3:
```

DOI : 10.31695/IJERAT.2020.3644

```
        Set KernelSize = 1
       For all X, Y Do {where 0  to  earX-1 , 0 to  earY -1 }
           Set GradX ← 0
          Set  GradY ← 0
          Set  Grad ← 0
         For all i, j Do {where -1  to 1, -1 to 1 }
                Set  Itensity ← ImagePixels(0,X, Y). Contrast_GrayScale

                Set  GradX ← GradX + (Itensity * Op_X(i, j))
                Set GradY ← GradY + (Itensity * Op_Y(i, j))
            End for

         Set Grad ← $\sqrt[2]{(GradX * GradX) + (GradY * GradY)}$

            If Grad >= ThresholdMin And Grad <= ThresholdMax Then
            Set Grad ← Grad -(((ThresholdMax + ThresholdMin) / KernelSize)
            If Grad <= ThresholdMax Then  Set Grad ← 0
            If Grad >= ThresholdMax Then  Set Grad ←255

           If Grad = 0 then
                Set ImagePixels(7,X, Y).Canny Edge←0
           Else
                Set ImagePixels(7,X, Y).Canny Edge←1
            Endif
         Else
            If Grad <= 0 Then        Set ImagePixels(7,X, Y).Canny Edge←0
            If Grad >= 255 Then   Set ImagePixels(7,X, Y).Canny Edge←1
   End for

Return  (ear (7,0 to earX-1, 0 to earY-1).Canny Edge)
End.
```

9. DATASET

The proposed system applied on Dataset consists of 40 images in two cases (Left profile and Right profile) the image is read as BMP 24 bit/pixel (bit depth).



DOI : 10.31695/IJERAT.2020.3644

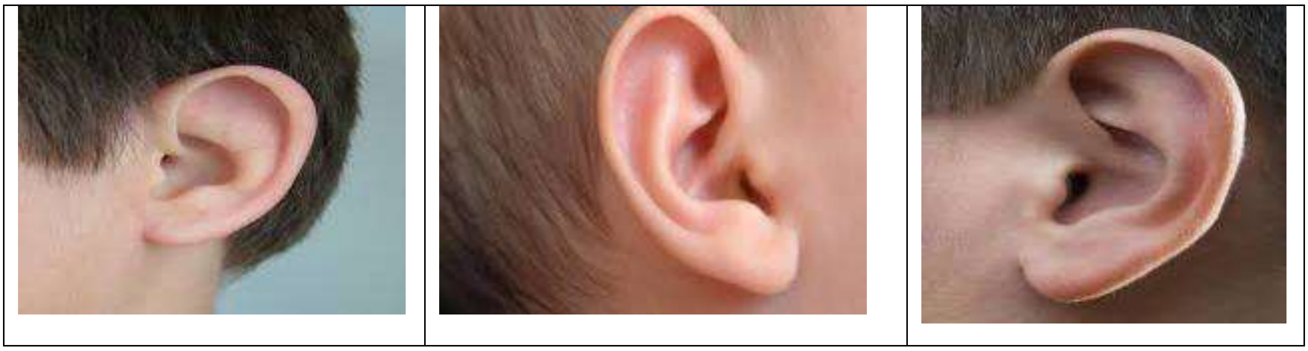


Figure 1.10: Samples of the Dataset used in this work

10. CONCLUSIONS

1. In this project, we have introduced a model to detect and extract ear in color image, by utilizing HSV color space which demonstrated clearer discrimination between skin and non-skin ranges. Skin area segmentation was performed using a certain range rules derived from the HSV color space.
2. The experimental results showed that our new approach in modeling skin color was able to achieve excellent rate of accuracy rate. It can be noticed that the model is able to separate the skin ranges from the back grounds.
3. The proposed syetem applied on Dataset consists of 40 images in two cases (Left profile and Right profile) the image is read as BMP 24 bit/pixel (bit depth).
4. Use of Canny edge detection leads to higher results because of use of the values of control parameters are: Kernel size of Gaussian Filter equal to (3x3) and its sigma equal to (0.8), threshold of gray scale equal to (127) and finally the values of maximum and minimum thresholds of canny filter are equal to (100) and (20) .

ACKNOWLEDGMENT

The authors would like to thank Mustansiriyah University (www.uomustansiriyah.edu.iq) Baghdad – Iraq for its support in the present work.

REFERENCES

1. P. Yan and K. W. Bowyer (2007). Biometric Recognition Using 3D Ear Shape. IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 29, NO. 8.

2. D. Singh and S. K. Singh (2014). A Survey on Human Ear Recog-nition System Based on 2D and 3D Ear Images. OPEN JOURNAL OF INFORMATION SECURITY AND APPLICATIONS, Volume 1, Number 2.

3. M. H. Farhan (2011). Fingerprint Recognition Using Fractal Geometry. M.Sc. thesis, Al Anbar university, college of coputer, computer science.

4. R. A.Al-Shamkhy (2006). A Proposed Passive Identification System Using Ear Biometrics Images. Ph.D. Thesis, Iraqi Commission for Computers and Inforamtics Informatics Institute for Postgraduate Studies.

5. S. Prakash and P. Gupta (2011). An Efficient Ear Recognition Technique Invariant to Illumination and Pose. Department of Computer Science & Engineering,Indian Institute of Technology Kanpur, Kanpur-208016, India.

6. V. S. Bhata and J. D. Pujar (2013). Face detection system using HSV color model and morphingoperations. International Jour-nal of Current Engineering and Technology.

7. S. Taha (2011). Human Face detection in color image By using different color space. Al-Mustansiriyah University, computer science.

8. S. Taha (2011). Human Face detection in color image By using different color space. Al-Mustansiriyah University, computer science.

9. C.Solomon and T.Breckon (2011).Fundamentals of Digital Image Processing. Physical Sciences, University of Kent, Canterbury, UK.

10. R. Gonzalez and R. Woods (2002). Digital Image Processing. Second Edition, Prentice Hall.

11. G. Mandloi (2014). A Survey on Feature Extraction Techniques for Color Images. International Journal of Computer Science and Information Technologies, Vol. 5 (3), 4615-4620.

DOI : 10.31695/IJERAT.2020.3644