Hand Print Recognition System based on FP-Growth Algorithm

Haitham Salman Chyad
Department of Computer Science, College of Education, Mustansiriyah University, Baghdad, Iraq.
E-mail: dr.haitham@uomustansiriyah.edu.iq

Raniah Ali Mustafa*
Department of Computer Science, College of Education, Mustansiriyah University, Baghdad, Iraq.
E-mail: rania83computer@uomustansiriyah.edu.iq

Kawther Thabt Saleh
Department of Computer Science, College of Education, Mustansiriyah University, Baghdad, Iraq.
E-mail: kawtherthabt@uomustansiriyah.edu.iq

Received August 20, 2021; Accepted December 02, 2021
ISSN: 1735-188X
DOI: 10.14704/WEB/V19I1/WEB19067

Abstract

Hand print recognition system received great interest in the recent few years such as human-computer interaction, computer vision, and computer graphics. In this paper, proposed system for recognition human handprint based on FP-growth algorithm, the system consists of three-stage. The first stage the detection algorithm using HSV color space, canny algorithm and contrast enhancement for grayscale. In this stage separate skin area in-handprint image through first HSV color space converting RGB to HSV color space as well as conducting specific rules for determining the skin area. And then applies skin hand segmentation for the split of non skin and skin areas where hand skin color detection. After the hand 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 hand. Finally applying canny edge detection. The second stage extract features through apply seven moment invariants. The three-stage applying FP-growth algorithm for recognition handprint image. The system which has been proposed utilize handprint images databases, the database proposed a large data-set of human hand images, 11K Hands, that consists of palmar and dorsal sides of the human hand images dataset that collective database from 190 various subject's handprint images is made publicly obtainable. The handprint recognition system achieved rate of 92.70%.

Keywords

Hand Print Detection, Seven Moment Invariants, Data Mining, Association Rules, Aprioi, FP-Growth Algorithm, Contrast Enhancement, Canny Edge Detection.

Introduction

Throughout the history, there was a high importance to find a way for identifying individuals, as different systems were used for such purpose. Traditional identification systems were operating on the basis of the fact that individuals must have the ability for memorizing a few passwords, also they should have certain coded card or photo ID, such systems have significant limitations, such as: possessions might be stolen, simply duplicated or lost; knowledge could be forgotten; knowledge and possessions might be stolen or shared. Recently, biometrics are getting more and more attention. There were various biometric systems according to many properties and different body parts, each one of the such biometrics has weaknesses and strengths based on its use and application. Systems using finger and hand geometry are effective for identification. In addition, the physical properties related to human's hand include a few information which is important to authenticate the individual's identity, as each hand has its unique finger's width, palm's width, finger's length, location or thickness of fingers, which are all distinguishing human beings. The biometric recognition systems are eliminating such obstacles. Also, biometrics are dealing with individuals' identification according to their behavioral or biological properties. A lot of biometric properties are used in many applications, such as finger and hand geometry, face, vice, gait, iris, fingerprint, odor, signature, DNA, keystroke, retinal scan, also infrared facial and hand vein thermo grams. In hand geometry systems, the major benefits are that they are simple and they might utilize low-resolution images as well as providing high-efficiency with excellent user acceptance (Saxena et al., 2013).

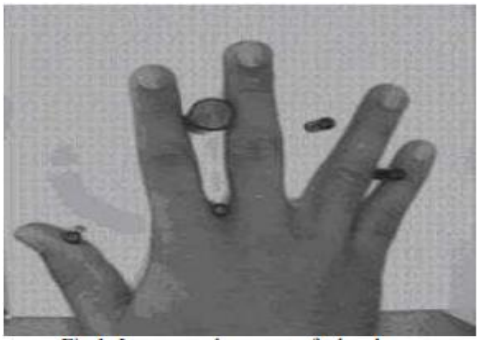


Figure 1 Incorrect hand placement

Conventional systems of hand geometry are using pegs for a purpose of fixing hand's placement (Saxena et al., 2013). Using pegs has 2 major drawbacks as they will be deforming the hand silhouette's shape and users could be incorrectly placing their hands as can be seen in the Figure 1, such issues might be reducing the biometric system's performance. A lot of studies regarding personal recognition on the basis of hand geometry were studied. A study conducted by (Jain & Duta 1999) designed a hand geometry based

system of verification which is utilized for prototype system of web security. A study conducted by (Sanchez 2000) chosen fingers' widths at various latitude values, palm and finger heights, finger deviations as a vector of features. A study conducted by Oden et. al. (Saxena et al., 2013) developed a system with the use of implicit polynomials. A study conducted by Sidlauskas (Saxena et al., 2013) discussed 3-D hand profile identification apparatus utilized in terms of hand recognition. Kumar et. al. (Saxena et al., 2013) utilized palm print and hand geometry; at the same time, Han et. al. (Saxena et al., 2013) utilized just palm print information with regard to personal authentication. In the presented work, the suggested algorithm depends on digital image processing involving recognition, feature extractions and segmentation of the hand.

Recently, there were a few studies associated with this study, including Tang et al. (2013) suggesting 2-phase HPR system in terms of Sign Language Recognition by means of Kinect sensor. In the 1st phase, the study suggested an efficient algorithm for implementing hand tracking and detection. The algorithm incorporated depth and color information, with no certain requirements on the uniformly colored or stable backgrounds. It has the ability for handling situations where the hands were extremely close to the other body parts or when the hands weren't the closest objects to camera, also it allowed hands' occlusion resulting from other hands or faces. With regard to the 2nd phase, the study applied DNNs for learning (automatically) features from the images of hand posture, which has been considered to be insensitive to rotation, scaling and movement. The experiments are verifying that the suggested system works accurately and rapidly, also it is achieving 98.12% recognition accuracy. A study conducted by Krzysztof (2020) introduced a novel algorithm of time-series classification for the problem of small training sets, such algorithm has been put to test on recordings of hand gesture in the tasks of gesture recognition and person identifications. In addition, the algorithm offers considerably more sufficient accuracy of classification in comparison to other algorithms of machine learning. With regard to a total of twenty-two hand gestures performed via ten individuals, while the size of the training set was five gesture execution records for each one of the classes, then the error rate regarding the newly-suggested algorithm was between (37% and 75%) lower compared to other algorithms. In the case when the training set includes just single sample for each one of the classes, the new algorithm will reach between (45-95) % lower error rate. The experimentations are indicating that the algorithm is outperforming modern approaches with regard to classification's accuracy in gesture recognition and person identification. BAMWEND and Özerdem (2019) suggested a static hand-gesture recognition system in real-time with utilizing 2 method of machine learning, which are Artificial Neural Networks (ANNs) and Support Vector Machines (SVMs), the newly-

introduced Microsoft Kinect sensor is used for image extraction. The sensor allows extracting hand images. The system is implemented on MATLAB due to the fact that it is majorly utilized via researchers in various fields and that it has the ability for handling complicated computations. With regard to model's training, one might collect 100 depth based Histogram regarding Oriented Gradient features for each one of the alphabets from hand gesture images which have been subjected to training, testing and validation with the use of SVM and ANN. From such dataset, one might have the ability for generating a generalized gesture model in terms of each one of the alphabet images. In the suggested system, the classification with ANN reported high performance in comparison to SVM. Also, the dataset has been organized as consisting static letters. In addition, the average of recognition accuracy has been 98.2 and 93.4 for ANN and SVM. For the suggested system, the better performance has been reported with ANN. MOHAMMED et al (2019) suggested an architecture, in which the entire image passes through one stage dense object detector for extracting hand region, that will be passing through light-weight CNN for the hand gesture recognition. For the purpose of evaluating this method, extensive experiments were conducted on 4 hand detection datasets, such as 5-signers, Oxford, Indian classical dance (ICD) and EgoHands datasets, in addition to the 2 hand gesture data-sets with various gesture vocabularies with regard to the recognition of the hand gestures, such as, Tiny Hands and LaRED data-sets. The experimental results are showing that the suggested architecture was robust and effective, it is outperforming other methods in gesture classification and hand detection tasks.

Moment Invariants

The method was majorly utilized for the image pattern recognition in various applications as a result of their un-changed features on the picture translation, scaling and rotation. The moment invariants were significant to calculate the region characteristics' region which might be used for the shape recognition. Also, the 2D geometrical moments with order (p+q) regarding a function were indicated in the Equation 1 (Mercimek et al., 2005), (Raniah et al., 2018):

$$m_{pq} = \sum_{x=0}^{x=N-1} \sum_{y=0}^{y=M-1} x^p \, y^q \, f(x,y) \qquad (1)$$

In which,

$$p, q \; 0,1,2, \dots, \infty$$

N denotes the number of columns
M denotes the number of rows.

The moments having a property of translation invariance were referred to as the central moments and were indicated via μ_{pq}, they have been specified in the Equations 2 (Mercimek et al., 2005), (Raniah et al., 2018):

$$\mu_{pq} = \sum_{x=0}^{x=N-1} \sum_{y=0}^{y=M-1} (x - \bar{x})^p \,.\, (y - \bar{y})^q \, f(x, y) \quad (2)$$

In which, $\bar{y}$ and $\bar{x}$ are the coordinates of centered and they were evaluated by Equations 3 and 4 (Mercimek et al., 2005), (Raniah et al., 2018).

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad (3)$$

$$\bar{y} = \frac{m_{01}}{m_{00}} \quad (4)$$

It might be specified that the central moments up to $p + q \leq 3$ order could be evaluated via the next Equations and formulas 5 - 14 (Mercimek et al., 2005), (Raniah et al., 2018):

$$\mu_{00} = m_{00} \quad (5)$$

$$\mu_{10} = 0 \quad (6)$$

$$\mu_{01} = 0 \quad (7)$$

$$\mu_{20} = m_{20} - \bar{x}\, m_{10} \quad (8)$$

$$\mu_{02} = m_{02} - \bar{y} \quad (9)$$

$$\mu_{11} = m_{11} - \bar{y}\, m_{10} \quad (10)$$

$$\mu_{30} = m_{30} - 3\,\bar{x}\, m_{20} + 2\,\bar{x}^2\, m_{10} \quad (11)$$

$$\mu_{12} = m_{12} - 2\,\bar{y}\, m_{11} - \bar{x}\, m_{02} + 2\,\bar{y}^2\, m_{10} \quad (12)$$

$$\mu_{21} = m_{21} - 2\,\bar{x}\, m_{11} - \bar{y}\, m_{20} + 2\,\bar{y}^2\, m_{01} \quad (13)$$

$$\mu_{03} = m_{03} - 3\,\bar{y}\, m_{02} + 2\,\bar{y}^2\, m_{01} \quad (14)$$

Scale invariance could be gained by means of normalized central moments η_{pq}, as Equations 15 and 16 (Mercimek et al., 2005), (Raniah et al., 2018).

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \quad (15)$$

In which,

$$\gamma = \left[\frac{(p+q)}{2}\right] + 1 \quad (16)$$

A 7 nonlinear absolute moment invariants, evaluated from the normalization of central moments through order 3 were provided in equations 17 to 23 (Mercimek et al., 2005), (Raniah et al., 2018):

$$\emptyset_1 = \eta_{20} + \eta_{02} \quad (17)$$

$$\emptyset_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (18)$$

$$\emptyset_3 = (\eta_{30} - 3\eta_{12})^2 + (3\,\eta_{21} - \eta_{03})^2 \quad (19)$$

$$\emptyset_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (20)$$

$$\emptyset_5 = (\eta_{30} - 3\eta_{12})\,(\eta_{30} + \eta_{12})\,[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})\,(\eta_{21} + \eta_{03})\,[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (21)$$

$$\emptyset_6 = (\eta_{20} - \eta_{02})\,[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}\,(\eta_{30} + \eta_{12})\,(\eta_{21} + \eta_{03}) \quad (22)$$

$$\emptyset_7 = (3\,\eta_{21} - \eta_{03})\,(\eta_{30} + \eta_{12})\,[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{30})\,(\eta_{21} + \eta_{03})\,[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (23)$$

Association Rules Methods

The association rules were one of the data mining major researched fields and have lately received a lot of focus from database community. They are of high importance in retail and marketing and many other fields (Mitchell 1999). First introduced in the year 1993, the association rules, were utilized for identifying the relations between the set of items in database, such relations aren't depending on the inherent properties related to data, yet rather depending on the co-occurrence regarding data items (Dunham et al., 2021).

Aprioi Algorithm

The algorithm is of high importance in terms of frequency item set for association rules (AR). It has been suggested via Agrawal and Srikant (Witten et al., 2011). Also, the algorithm of the names depends upon the fact that the algorithm uses prior knowledge regarding the frequency item set features. Algorithm that uses a repeated way also it has been characterized as a level-wise (L-w) searching, in which the k-item sets were used for exploring (k+1)-item sets. Firstly, the group of frequency values of 1-itemsetsis found out. This group was specified via L1. L1 and used for finding out L2, also the group of the frequency values of 2-itemsets applied for finding out L3, and others, even in the case where no more frequent k-item sets have been identified. One must do complete DB scan to find each L_k (Verhein 2008).

Algorithm of FP-Growth

Lately, the FP tree-based frequent pattern mining approach, referred to as FP- growth, introduced via Han et al have achieved high-efficiency, in comparison to Apriori-like method. In addition, the FP-growth approach utilizes divide-and-conquer strategy, applying just 2 full I/O scans regarding the data-base, also avoiding iterative candidate generations (Han et al., 2000). In addition, the FP-growth was utilized for mining complete group regarding frequent patterns through the growth of pattern fragment. There are 3 techniques used to do effective mining (Han et al., 2000):

1. FP-tree based mining uses an approach of pattern fragment growth for avoiding the costly generation regarding many candidates sets.
2. A large data-base has been compressed to much smaller, highly compressed data structures that are avoiding repeated and costly data-base scans.
3. Partitioning based divide-and-conquer approach was utilized for significantly reducing the size of the search space.

The problem of the frequent pattern mining might be defined in a formal manner as, assuming I = {i1, i2,..., im} is the group of items, while DB represents the transaction data-base, in which every one of the transactions T represents a group of the items and $T \subseteq I$. TID, a distinctive identifier, was allocated with each one of the transactions. Also, transaction T includes pattern P, set of items in I, if $P \subseteq T$. Furthermore, the support of pattern P represents the number of transactions which contain P in the database. It is assumed that P represents a frequent pattern, in the case when P is not more than pre-defined minimal support threshold ξ, frequent pattern mining that includes 2 phases (Han et al., 2000):

1. Constructing a compact data structure, FP-tree, that might be storing further information in a smaller amount of the space.
2. Developing FP-tree-based pattern growth (FP-growth) methodology for recursively uncovering all the frequent patterns.

Apriori algorithm might be suffering from the next 2 non-trivial costs (Mazman & Usluel 2009), (Agrawal & Srikant 1994):

1. It might still require generating a lot of candidates sets, for instance, in the case when there were 10^4 frequent 1-itemsets, the Apriori algorithm requires generating over 10^7 candidate 2-itemsets.

2. It might require repeated scan of the entire database as well as checking a large group of the candidates through pattern matching. In addition, it's expensive to try going over every one of the data-base transactions for determining the support regarding candidate item sets.

Suggested System

The major idea regarding the proposed system recognition human handprint is based on FP-growth algorithm there the system utilized for the feature extraction from human handprint image on the basis of 7 moment invariants and saved features in Training database Features (TRDBF) where training five images and testing five images them suggested system for recognition FP-growth algorithm. The suggested system is conducted as follows:

1. Preprocessing included (Read Image, Convert to HSV Color Space, Skin Area Segmentation, Hand Detection Stage (Hand Image Smoothing Step, Convert into Gray scale Sub stage, Contrast Enhancement Step, Canny Edge detection Sub Step).
2. Feature extraction based upon 7 moment invariants.
3. Recognition system on the basis of FP-growth algorithm. In the presented work, each one of the steps in suggested system will be explained with results. Each suggested system's step is shown in Figure 2.

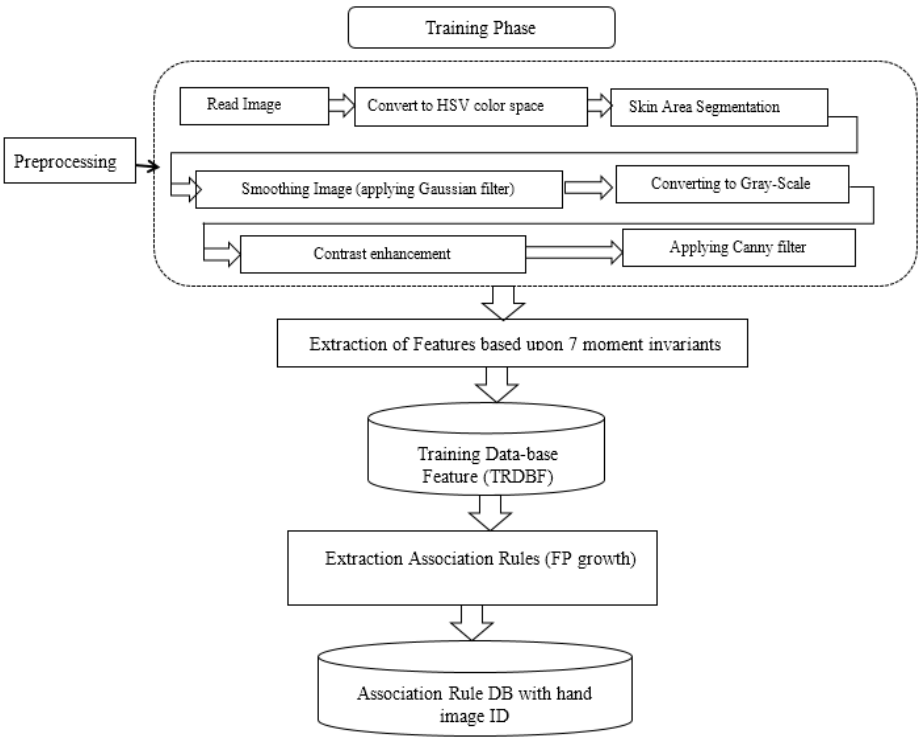


Figure 2 The suggested hand recognition system model

Read Image

Color hand image will be fed into system as the BMP image file; the image's color resolution is 24 bit/pixel. In the case when loading the BMP image, the header information of the image will be read first, after that, the bitmap pixel data is going to be read. The 3 bands: R, G and B will be loaded and put in a 3-D array with record.

Convert to HSV Color Space

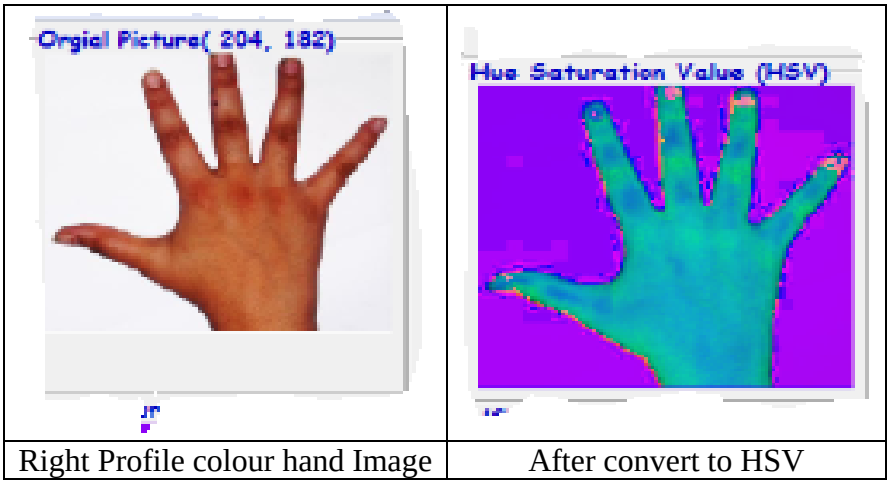
With regard to RGB color space, extreme changes in the color values could not be identified via humans. Thus, the color space isn't adequate for skin detection, as a result, the HSV color space that is adequate to represent the skin color of humans, was utilized in the suggested system. The aim of this step is determining skin area in the hand image via initially converting the RGB to HSV color space as well as utilizing specific rules for determining skin area. The conversion of color image to HSV color space is shown in the Figure 3. The Equations 25, 26, and 27 are used for representing the transformation rules for obtaining (H, S, & V) values from the RGB color space (Taha 2011), (Solomon 2011).

$$H= \begin{cases} (\frac{G-B}{Max-Min})/\,6 \; if \; R = Max \\ (2+\frac{B-R}{Max-Min})/\,6 \; if \; G = Max \\ (4+\frac{B-R}{Max-Min})/\,6 \; if \; B = Max \end{cases} \quad (25)$$

$$S = (\frac{Max-Min}{Max}) \quad (26)$$

$$V= Max \quad (27)$$

In which, R represents Red. B represents Blue. G represents Green.



Right Profile colour hand Image	After convert to HSV

Figure 3 Converting color Image to HSV Color Space

Skin Area Segmentation

Blue values are showing the major detectable split regarding non-skin and skin areas as can be seen in the Figure 3, also, the skin in band H has been specified via values in range of (0-40), in band S, between (30 and 160), while in band V, between (150 and 255). After scanning all the image pixels, each one of the pixels which is achieved in Equation 28 can be categorized as a skin color pixel and has been set to 255, and nonskin pixels will be set as 0 Figure 4 exhibits examples regarding the hand skin color detection.

$$(H > 0)\ and\ (H < 40)\ and\ (S > 30)\ and\ (S < 160)\ and\ (V > 150)\ and\ (V < 255) \quad (28)$$

Hand Detection Stage

In this stage, the major aim is detecting the main edge in hand image, such stage involves 3 steps, image smoothing, conversion into grayscale, contrast Gray scale and utilizing canny edge detection.

Hand Image Smoothing Step

Following the stage of hand detection, image smoothing is considered as the first stage in edge detection with aim of reducing the current noise and enhancing the hand details' shape. Also, the gaussian filter as indicated in the Equation 29 was utilized in such step of suggested detection hand (Mitchell 1999). In addition, the sigma which equals (0.8) as well as kernel size (3x3) were selected in the suggested system. Figure 4 discusses the application regarding Gaussian filter with sigma value (0.80) and kernel size (3 x 3) with the array value that has been created for the case. The step of smoothing is used for achieving excellent results of edge extraction.

$$f(x,y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma 2}\right) \quad (29)$$

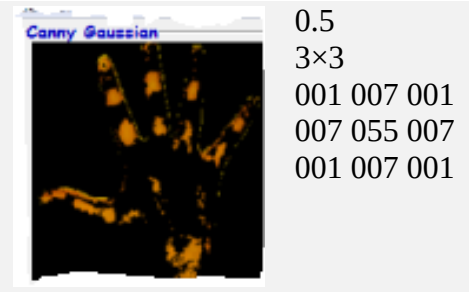


Figure 4 The applying Gaussian filter with sigma value

Convert into Gray Scale Substage

The hand's-coloured image will be converted to grey-scale, every one of the coloured inputs will be converted into grey-scale image (such process is used for converting 24bit/pixel images to 256 grey-scale image), also Equation 30 was utilized in such step of suggested detection hand. Figure 5 shows the result of using threshold value in a range of (127) with sigma value that equals (0.80) of Gauss filter for obtaining the optimal results.

$$F(x,y)= \text{Threshold } (\frac{R}{255} + \frac{G}{255} + \frac{B}{255}) \quad (30)$$

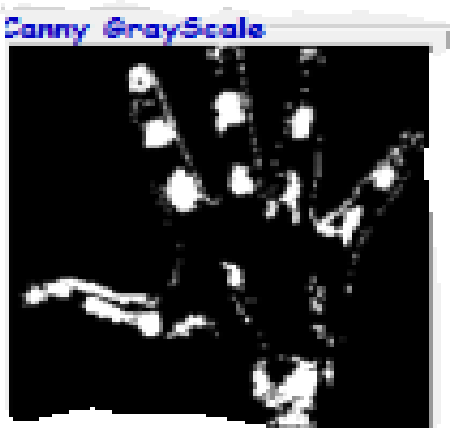


Figure 5 Hand after converting to Gray-Scale

Contrast Enhancement Step

In such stage, one of the significant steps is enhancement making the details of the hand better for the next-step regarding the suggested detection hand. For the purpose of accomplishing such aim, the contrast stretching was made for the gray value. Also, the used contrast stretching includes 2 steps; the first one is to determine the maximum and minimum threshold values. Then, linear stretching was used; it is used for moving low intensity values which were not more than the specified (minimal value) towards 0, also the high level values more than (maximal value) will be moved towards 255. In addition, all the values that range from minimal to maximal values were mapped (in a linear manner). Thus, the intensity levels' range was stretched to full range (0-255). In addition, a look table was established to speed up the process of mapping; the table was used for directly changing each one of the pixel values to its corresponding new value. As soon as establishing such look table, then each one of the pixels will be directly mapped into corresponding new value with no requirement for recalculating the same equation. The applied contrast enhancement steps are shown in Algorithm (1).

Algorithm (1) Enhancement of the Contrast
Goal: Conversion the Grey to a more usable manner.
Input: Image_Pixels(6,0 to C_F_ Wid-1, 0 to C_F_ Hgt-1).Gray-scale // For ear clip
C_F_ Wid, C_F_ Hgt // hand clip Width & Height
Sig // value of the sigma
Min // Minimum gray value
Max // Maximum gray value
Output: Image_Pixels(0,0 to C_F_ Wid-1, 0 to C_F_ Hgt-1).Contrast_GreyScale // hand clip following the enhancement of the contract

Step1: Initialization of the lookup table (0 - 255)
For all I **Do** { where 0 to 255}
If I > Max
Set lookTable(I) ← 255
If I < Min
Set lookTable (I) ← 0

Else Set Look_Table (I) ← ((value(i) - min) / (max - min)) ^ Sig x255
EndFor
Step2: Compute the new values (Contrast GreyScale)
For all X, Y Do {where 0 to C_F_ Wid-1, 0 to C_F_ Hgt-1 }
If Image_Pixels(6,X,Y).Greyscale=255 then
Set Image_Pixels (0, X, Y).Contrast_GreyScale← 255
Else if Image_Pixels(6,X,Y).Gray_scale=0 then
Set Image_Pixels (0, X, Y).Contrast_GreyScale← 0
Else Image_Pixels (0, X, Y).Contrast_GreyScale←Look-Table(Image_Pixels (6,X,Y).Grayscale)
EndFor
Return (Image_Pixels(0,0 to C_F_ Wid-1, 0 to C_F_ Hgt-1).Contrast_GreyScale).
End

The Substage of Canny Edge Detection

Generally, the aim of edge detection is considerably reducing the data amount in the image, whereas protecting the major structure that is used for more image processing.

Figure 6 shows using Canny filter following the (image smoothing step as well as converting it into Gray scale step) with minimum and maximum threshold values (T min equals 20, T Max equals 100). Algorithm (2) shows the steps of using canny filter.

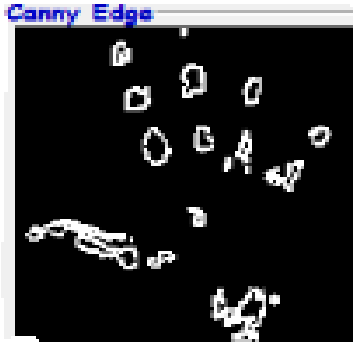


Figure 6 An example of Canny filter application

Algorithm2 Canny Edge Detector
Aim: Edge Detection
Input:earM (0,0 to earX-1, 0 to earY-1).Contrast_Gray_Scale //hand
Threshold_Min
Threshold_Max
Output: handM (7,0 to earX-1, 0 to earY-1).CannyEdge// hand

Step1: defining 3x3 structuring element OpX
Assign Op_X(-1, -1) ← -1: Assign Op_X(0, -1) ← -2: Assign Op_X(1, -1) ← -1
Assign Op_X(-1, 0) ← 0: Assign Op_X(0, 0) ← 0: Assign Op_X(1, 0) ← 0
Assign Op_X(-1, 1) ← 1: Assign Op_X(0, 1) ← 2: Assign Op_X(1, 1) ← 1
Step2: defining 3x3 structuring element OpY
Assign Op_Y(-1, -1) ← -1: Assign Op_Y(0, -1) ← 0: Assign Op_Y(1, -1) ← 1
Assign Op_Y(-1, 0) ← -2: Assign Op_Y(0, 0) ← 0: Assign Op_Y(1, 0) ← 2
Assign Op_Y(-1, 1) ← -1: Assign Op_Y(0, 1) ← 0: Assign Op_Y(1, 1) ← 1
Step3:
Assign Kernel_Size = 1
For each X, Y Do {where 0 to earX-1, 0 to earY -1 }
Assign Grad_X← 0
Assign Grad_Y← 0
Assign Grad ← 0
For each i, j Do {where -1 to 1, -1 to 1 }
Assign Itensity←Image_Pixels(0,X, Y). Contrast_GreyScale

Assign Grad_X←Grad_X + (Itensity x Op_X(i, j))
Assign Grad_Y←Grad_Y + (Itensity x Op_Y(i, j))
EndFor
Assign Grad $\leftarrow \sqrt[2]{(\text{Grad}_X * \text{Grad} - X) + (\text{Grad\_Y} * \text{Grad\_Y})}$
If Grad >= Threshold_Min & Grad <= Threshold_Max Then
Assign Grad ←Grad -(((Threshold_Max + Threshold_Min) / Kernel_Size)
If Grad <= Threshold_Max Then Assign Grad ← 0
If Grad >= Threshold_Max Then Assign Grad ←255
If Grad = 0 then
Assign Image_Pixels(7,X, Y).CannyEdge←0
Else
Assign Image_Pixels(7,X, Y).CannyEdge←1
End If
Else
If Grad <= 0
Assign Image_Pixels(7,X, Y).CannyEdge←0
If Grad >= 255
Assign Image_Pixels(7,X, Y).CannyEdge←1
EndFor
Return (ear (7,0 to earX-1, 0 to earY-1).CannyEdge)
End.

Feature Extraction based upon Seven Moment Invariants

Hand image is used in this stage for extracting the features. Moments' invariant approach extracts 7 values as features; section (2) mentions all the details. The 7 values have been extracted from the hand image following detecting the hand image via 7 moment invariants applied on the hand image. Using the process of feature extraction is achieved with the use of steps in algorithm (3).

Algorithm3. Steps of Feature Extraction (Training the hand Image)
Input: Colour hand Image
Goal: Hu's Moment for every one of the hand images

Step 1: Reading the hand image data
Step2: Detection of hand area
Step 3: Applying the hand image smoothing based upon the Gaussian Filter with the use of eq. (29).
Step4: Converting the hand Image into Grayscale based on equation (30).
Step5: Applying canny edge detection based on algorithm (2).
Step6: Getting the width of the image (**For** i=0 to ImageWidth -1)
Getting the height of the image (**For** j=0 to ImageHeight -1)
Step7: Calculating moment of $(m_{00}, m_{01}, m_{10}, m_{11}, m_{02}, m_{20}, m_{12}, m_{21}, m_{03}, m_{30})$ order from eq1.
Step8: Computing the mass center coordinates with the use of eq. (3) & eq (4).
Step9: Calculating central moment (which is represented by μ_{pq}) with the use if eq. (2) of order $\mu_{00}, \mu_{01}, \mu_{10}, \mu_{11}, \mu_{02}, \mu_{20}, \mu_{12}, \mu_{21}, \mu_{03}, \mu_{30}$ and for the easy calculations eq.(5) to eq.(14). are used
Step10: Calculating normalized central moment (which is represented by η_{pq}) utilizing eq. (15) & eq. (16) for ($\eta_{20}, \eta_{02}, \eta_{11}, \eta_{30}, \eta_{03}, \eta_{21}, \eta_{12}$) order.
Step11: Utilizing the normalized central moment compute the 7 moment invariants (Ø1, Ø2, Ø 3, Ø4, Ø 5, Ø6, Ø 7), eqs. (17-23).
Step12: Storing the 7 moment invariants (Ø 1, Ø2, Ø 3, Ø 4, Ø 5, Ø 6, Ø 7) in Training data-base feature for face images.
Step13: If there are more face images, redo steps (1-12) for each face image.
Step14: End.

In such stage, following analysis, the hand images are going to be utilized to normalized central moment for obtaining 7 moment invariants feature vectors extraction with regard to every one of the bands (7 band) and saving the in training data-base features (TRDBF). Table1 exhibits the extracted features' results (prior to normalization) for the sample related to hand image data-base.

Table 1 Results of obtained Features for hand Image Data-base sample (prior to the Normalization)

Class	Features							
C1	**Image**	**A**	**B**	**C**	**D**	**E**	**F**	**G**
	C1-1	1.731209274	0.474204702	0.201504482	0.007739139	0.001824354	-0.001474301	-0.000131658
	C1-2	1.744893126	0.43682177	0.122737652	0.012112309	0.003757153	-0.004143249	-0.000411777
	C1-3	1.742523433	0.097216296	0.246838215	0.068364968	0.008916204	0.020995025	-0.00214773
	C1-4	1.671138054	0.028090196	0.009634663	0.004221654	-0.000509171	0.000641826	2.69192E-05
	C1-5	1.401899797	0.015484811	0.023636958	0.003780747	0.001859901	-0.000238953	2.36449E-05
C2	**Image**	**A**	**B**	**C**	**D**	**E**	**F**	**G**
	C2-1	1.233005543	0.025439072	0.064208664	0.095150583	0.003795692	-0.009263798	-0.004143926
	C2-2	1.437604926	0.044622129	0.193871592	0.340123895	0.078537023	0.063421488	0.021062387
	C2-3	1.486013006	0.048958667	0.120350096	0.235939159	0.030447808	-0.024470006	-0.01316609
	C2-4	1.630076038	0.016587107	0.422122949	0.197861813	0.022840525	-0.025329724	-0.010768491
	C2-5	1.456876623	0.001871599	0.2474979	0.427610275	0.134943741	0.005415247	-0.016002467

In this substage, following the 7 moment invariants feature vector extraction normalization (f_{norm}) will be carried out through finding minimum (f_{min}) and maximum (f_{max}) values for every one of the features in every class of every sample in TRDBF. Table2 lists results of the (f_{max},f_{min}) that have been obtained for feature vectors for a sample hand image data-base.

Table 2 Results of obtained (f_{max}, f_{min}) for Texture Features from samples of hand Image Data-base

Class	**Features**						
C1	**A**	**B**	**C**	**D**	**E**	**F**	**G**
fmax	1.744893126	0.474204702	0.246838215	0.068364968	0.008916204	0.020995025	2.69192E-05
fmin	1.401899797	0.015484811	0.009634663	0.003780747	-0.000509171	-0.004143249	-0.00214773
C2	**A**	**B**	**C**	**D**	**E**	**F**	**G**
fmax	1.630076038	0.048958667	0.422122949	0.427610275	0.134943741	0.063421488	0.021062387
fmin	1.233005543	0.001871599	0.064208664	0.095150583	0.003795692	-0.025329724	-0.016002467

Following obtaining the values of the minimum (f_{min}) and maximum (f_{max}) for every one of the features in every class for every sample in TRDBF process normalization will be implemented. Table3 lists the results of the obtained features for some samples of hand image data-base.

Table 3 Results of the Obtained Features from a Sample of the Medical Image Data-base (following the normalization)

Class	Features							
C1	**Image**	**A**	**B**	**C**	**D**	**E**	**F**	**G**
	C1-1	1.731209274	0.474204702	0.201504482	0.007739139	0.001824354	-0.001474301	-0.000131658
	C1-2	1.744893126	0.43682177	0.122737652	0.012112309	0.003757153	-0.004143249	-0.000411777
	C1-3	1.742523433	0.097216296	0.246838215	0.068364968	0.008916204	0.020995025	-0.00214773
	C1-4	1.671138054	0.028090196	0.009634663	0.004221654	-0.000509171	0.000641826	2.69192E-05
	C1-5	1.401899797	0.015484811	0.023636958	0.003780747	0.001859901	-0.000238953	2.36449E-05
C2	**Image**	**A**	**B**	**C**	**D**	**E**	**F**	**G**
	C2-1	1.233005543	0.025439072	0.064208664	0.095150583	0.003795692	-0.009263798	-0.004143926
	C2-2	1.437604926	0.044622129	0.193871592	0.340123895	0.078537023	0.063421488	0.021062387
	C2-3	1.486013006	0.048958667	0.120350096	0.235939159	0.030447808	-0.024470006	-0.01316609
	C2-4	1.630076038	0.016587107	0.422122949	0.197861813	0.022840525	-0.025329724	-0.010768491
	C2-5	1.456876623	0.001871599	0.2474979	0.427610275	0.134943741	0.005415247	-0.016002467

APPLICATION OF THE FP-GROWTH: following getting features will apply the algorithm of the FP-growth where features encoding for the purpose of finding repetitions and drawing before finding the rule.

Algorithm4: Algorithm of Custom FP-Growth
Input: A conventional DB, minimal support threshold Goal: Its frequent pattern tree, tree structure
Start FP-tree is produced through the use of the steps below: Scanning DB once. Collecting a group of the frequent items F as well as their supports. Sorting F in the support descending order as L, the list of the redundant items; Creating the FP-tree root, T, and it is labeled as the "root"; While Trans in database: Selecting and sorting frequent items in the Trans based on order of L; Sorting the frequent item list = [p | P], where p represents the 1st element and P represents the rest of the list; Calling *InsertTreeFunction ([p | P], T);* *Calling the FP-growth (FP-tree, null). //* The FPtree is mined EndProcess; *Inserting the tree function ([p | P], T) :* Start If T has a child N such that (N.item-name = p.item-name) Incrementing N's count by 1; Else Creating a new node N, and letting its count be 1, its parent link will be linked to T, and its node-link will be linked to nodes with same item-name through node-link structure; EndIf; If P is non-empty, call InsertTree (P, N) recursively; EndIf; *FP-growth function (FP-tree, null):* // Process of the FPgrowth(Tree, A) Start If Tree includes one path P for every one of the combinations (which are represented as B) of nodes in path P produce pattern B∪A with the support=minimal support of the nodes in B Else for every one of the ai in Tree header { Generating the pattern B = ai∪A with support = ai.support; Constructing B's conditional pattern base through the traversal of FPtree through following link of every one of the frequent item and accumulating all the transformed prefix paths of that item ; Construct B's conditional FP_tree (Tree B) through the accumulation of the count for every one of the items in conditional pattern base >= minimal support; If Tree B <> Ø then call the FPgrowth(Tree B, B); EndIf; } EndIf; End;

In the present paper utilizing the FP-growth, where will be encoding features for finding frequency and that the benefits of algorithm is reducing the scan and after that, will build the tree and after that obtain rules for the retrievals of the eye images person. Table4 shows database following the encoding and table5 lists frequencies.

Table 4 Encoding data-base

Id Set (class)	Item_Set
C1	a,b,c,d,e
C2	e,b,e,d,e
C3	a,b,a,d,d
C4	a,e,d,e
C5	c,b,c,d,c

Table 5 Frequency Item set (1st FP-growth step)

Item ID	Support Count
A	10
B	8
C	7
D	12
E	9

After that, the tree will be built and after that, the rules have been extracted for retrieving the eye image of the person. Table 6 lists rules (id) is utilized for recognizing the hand images of the person.

Table 6 Results of the Association Rules Extracted database (FIAEDB) as well as their Confidence from the sample of the hand images of the persons.

Class	Association Rules			
	Symbols	**Equations**	**Relations**	**Confidences**
C1	C 26	A,C,/A,	0.10940/0.10940	A-->C
	C 26	B,G,/B,	0.10940/0.10940	B-->G
	C 26	C,D,/D,	0.10940/0.10940	D-->C
	C 26	E,G,/E,	0.10940/0.10940	E-->G

Criteria of the Evaluation

The evaluating efficiency of the system of the face recognition has been computed by 2 measures, which are Recognition Rate (RR) and False Alarm Rate (FAR). The formula for the calculation of those measures have been provided as in Equations 31 and 32 respectively.

1. **FAR**: has been described as ratio of numbers of the false recognition decision to entire number of the attempts.

$$\text{FAR} = \frac{\text{Number of the false recognition attempt cases}}{\text{Total number of the attempts}} * 100 \qquad (31)$$

2. **Rate of Recognition**: is ratio of number of the correct recognition decisions to total number of the attempts.

$$\text{RR} = \frac{\text{Number of the accurate attempt cases}}{\text{Total number of attempts}} * 100 \qquad (32)$$

In the Table (7) demonstrations the best recognition rate.

Table 7 The achieved recognition rate for the training and testing for each classes

Recognition Rate		
	FAR	RR
Training	0.0	100%
Testing	17.3%	92.70%%

Dataset

The suggested system is utilized on a dataset which includes 11,076 hand images taken from a total of 190 subjects, each one of the images has (1,600×1,200 pixels). Also, the subjects had different ages, in range between (18 and 75) years old. For getting variety in captured shapes of the hands, every one of the subjects was required to close and open their fingers of the left and right hands. Also, every one of the hands has been photographed from palmar and dorsal sides with uniform white background and after that approximately placed in same distance to the camera. There was a metadata record related to each one of the images that involves: (1) subject's ID, (2) gender, (3) age, (4) skin color, and (5) a group of the information of captured hand, for instance, left or right hand, hand side (palmar or dorsal), and logical indicators indicating if the hand image includes nail polish, irregularities or accessories. The suggested data-set has a lot of hand images with extra detailed meta-data. The figure (7) shows the samples of the dataset utilized in the proposes system.

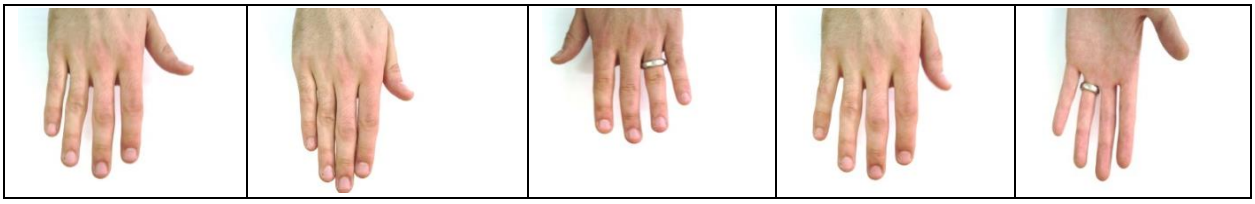


Figure 7 Samples of the Dataset used in this work

Conclusions

The aim of our paper is to design a system capable of recognizing the hand using FP-Growth Algorithm. We have introduced a model to detect and extract hand in color image, by utilizing HSV color space that shows clear discriminations between non skin and skin ranges. Skin area segmentation was performed using a certain range rules derived from the HSV color space. The experimental results specified that the novel approach used in this work to model the skin color has the ability of achieving good accuracy rates. It can be noticed that the model is able to separate the skin ranges from the back grounds. Use of Canny edge detection leads to higher results due to using the control parameters' values: Gaussian Filter's Kernel size which equals (3 x 3) and its sigma equals (0.80), grayscale threshold equals (127) and lastly the values of maximal and minimal thresholds of canny filter which equals 100 and 20. In addition, the 7-moment invariant that has been utilized for feature extraction exhibits the robustness due to the invariant features on rotation, translation and scaling of an image. Normalization used in this work for implementing features which have been extracted from 7 moment invariant for achieving the aim of decreasing the features size. Using association FP growth in the recognition provided excellent results due to using 7 moment invariants from were extraction. The suggested system achieve excellent percentage of 92.70%.

Acknowledgements

The authors would like to thank Mustansiriyah University for its supports.

References

Saxena, N., Saxena, V., Dubey, N., & Mishra, P. (2013). Hand geometry: A new method for biometric recognition. *International Journal of Soft Computing and Engineering (IJSCE), 2*(6), 2231-2307.

Jain, A.K., & Duta, N. (1999). Deformable matching of hand shapes for user verification. *In Proceedings International Conference on Image Processing (Cat. 99CH36348), 2,* 857-861.

Sanchez, R., R. (2000). Hand Geometry Pattern Recognition Through Gaussian Mixture Modeling. *15th, International Conference on Pattern Recognition*, Vol. 2, pp. 937-940.

Sanchez-Reillo, R. (2000, September). Hand geometry pattern recognition through gaussian mixture modelling. *In Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, *2*, 937-940. http://doi.org/10.1109/ICPR.2000.906228

Tang, A., Lu, K., Wang, Y., Huang, J., & Houqiang, L. (2013). A Real-time Hand Posture Recognition System Using Deep Neural Networks. *ACM Transactions on Intelligent Systems and Technology*, 9(4).

Rzecki, K. (2020). Classification Algorithm for Person Identification and Gesture Recognition Based on Hand Gestures with Small Training Sets. *Sensors*, *20*(24). http://doi.org/10.3390/s20247279

BAMWEND, J., & Özerdem, M., S. (2019). Recognition of static hand gesture with using ANN and SVM. *DUJE (Dicle University Journal of Engineering)*, vol. 10, no.2, pp. 561-568, DOI:10.24012/dumf.569357

MOHAMMED, A., A., Q., Lv, J., & Islam, M., S. (2019, November 30). A Deep Learning-Based End-to-End Composite System for Hand Detection and Gesture Recognition. *sensors*, doi: 10.3390/s19235282

Mercimek, M., Gulez, K., & Mumcu, T., V. (2005). Real Object Recognition Using Moment Invariants. *S¯adhan¯a*, Vol. 30, No. Part 6, pp. 765–775, DOI: 10.1007/BF02716709

Raniah, A. M., Kawther T. S., & Haitham, S. C. (2018 August 8). Feature Extraction Based on Wavelet Transform and Moment Invariants for Medical Image. *International Journal of Engineering Research and Advanced Technology (IJERAT)*,
doi.org/10.31695/IJERAT.2018.3315

Mitchell, T., M. (1999). Machine Learning and Data Mining. *Communications of the ACM*, doi.org/10.1145/319382.319388

Dunham, M., H., Xiao Y., Gruenwald, L., & Hossain Z. (2021). A Survey of the Association Rules.

Witten, I., H., Frank, E., & Hall, M., A. (2011). Data Mining Practical Machine Learning Tools and Techniques. *Elsevier Inc (Ed)*. ISBN 978-0-12-374856-0 (pbk.).

Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining Association Rules Between Sets of Items in Large Database. *Proceedings of the 1993 ACM SIGMOD Conference*, Washington DC, USA.

Verhein, F. (2008). Frequent Pattern Growth (FP-Growth) Algorithm An Introduction. *School of Information Technologies*, the University of Sydney, Australia.

Han, J., Pei, J., & Yin, Y. (2000). Mining Frequent Patterns without Candidate Generation. *University of Illinois at Urbana-Champaign*.

Han, J., & Yin, Y. (2000). Mining frequent patterns without candidate generation. In Proc. of the *ACM SIGMOD Conference on Management of data*.

Mazman, S., M., & Usluel Y., K. (2009). The Usage of Social Networks in Educational Context. *World Academy of Science, Engineering and Technology*.

Agrawal, R., & Srikant, R. (1994). Fast Algorithms for Mining Association Rules in Large Databases. *Proceedings of the Twentieth International Conference on Very Large Databases*.

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

Solomon, C., & Breckon, T. (2011). Fundamentals of Digital Image Processing. *Physical Sciences, University of Kent*, Canterbury, UK. www.wiley.com/wiley blackwell.