

**Republic of Iraq
Ministry of Higher Education
And Scientific Research
Baghdad University
College of Science**



Palm Recognition System Using Moment Feature

**A Project Report Submitted to the College of Science, Baghdad
University in Partial Fulfillment of the Requirements for the BSc
Degree of Science in Computer Science**

BY

Esra'a hassn hameed

Zahraa mageed abood

SUPERVIED BY

LECTURAL

Bushra Abdullah Sultan

2011-2012

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

أشهد الله ان لا اله الا هو
و الملائكة واولو العلم قائما
بالقسط لا اله الا هو العزيز الحكيم

صدق الله العظيم

لإهداء

إلى من أفنى عمره في العطاء الدائم, إلى الذي أرشدني ووقاني من زلاتي حتى أصبحت ما أنا عليه..... والذي الغالي

إلى من كان دعاؤها سر نجاحي وشمعة طريقي , عنوان التضحية ونبع العطاء.... إلى نسيم الحياة الجارية.... والذي الغالية

إلى من قاسموني الحياة , الورود العطرة التي أمدتني بعبق الحياة..... إخواني الأحباء

إلى بهجة النفس والروح , إلى رفاق الدرب الطويل , أصدقائي الأوفياء

إلى حملة العلم والقيم... أستاذتي الأعزاء

Acknowledgement

Before any thing

Praise be to god for all things

We would like to express our deepest gratitude to our supervisor Mrs. Bushra Abdullah for her great help and concern during the preparation of this work.

We would like to thanks all our families, without their support and assistant we could not get to this point.

Thanks to our teachers of computer science department for all teaching and encouragement during the study phase.

Finally thanks to all my friends for supporting and giving me advices.

تأييد المشرف

أؤيد إن المشروع الموسوم قد أعد تحت إشرافي في قسم علوم الحاسبات / كلية العلوم / جامعة بغداد كجزء من متطلبات الحصول على درجة البكالوريوس في علوم الحاسبات.

إسم المشرف:

التوقيع:

التاريخ:

تأييد لجنة المناقشة

نؤيد إننا اطلعنا على المشروع الموسوم كأعضاء لجنة المناقشة وقد تم إختبار الطالبة إسراء حسن حميد والطالبة زهراء مجيد عبود في محتوى التقرير وعليه نوصي بقبول المشروع كبحث للتخرج للحصول على درجة البكالوريوس في علوم الحاسبات.

الاسم والتوقي

الاسم والتوقيع

الاسم والتوقيع

الاسم والتوقيع

Background

The need for security and privacy rights is always paid to the development of ways to prevent intruders to the contents of the system.

The passwords of the first methods used to support systems to maintain security and privacy. But this system, which relies on passwords a lot of weaknesses to forget the most important of the possibility of that word or stolen.

To overcome the weaknesses introduced a system of smart cards, the smart cards that can save complex passwords, encrypted, but there has been a problem is how to verify the identity of the holder of the card, and was thinking about using the new technologies tested man himself, such as identifying the hallmarks of his voice or fingerprint sample or fingerprints or palm of his hand or finger prints, and others.

Because of the palm print is a hallmark unique to each person as it is difficult simulation or imitation, they support a more reliable and security.

So , This project was designed to obtain data of palm print that we need, through the calculations and algorithms applied to the project for analysis and identification of the user. The Delphi is used to design the project and implement the algorithms because the Delphi language of high productivity and one of the languages powerful and easy-together, and is said to combine ease of VISUAL BASIC language and the power of C + +.

In addition, the ease of dealing with this language and the availability of applications and functions that help the programmer is ready to write and implement its programs

<u>Index no.</u>	<u>page no.</u>
Back ground	6
Index	7
Chapter one: image analysis	8
Palm recognition system	9
Processing	9
Moment invariants	11
Chapter 2: system layout	12
System design	13
Image filtering	14
Image binarization	14
Moment invariant	15
Experimental & Result	17
Chapter three: practical part	18
Chapter four: conclusion	25
Conclusions	26
Reference	27

Chapter one

"image analysis"

1. Palm Recognition System

Palm Recognition: is a modal biometrics system for personal identification. It used only a single hand image to capture this feature simultaneously to form a new feature. The new feature was then used to recognize the user.

General stages of palm recognition system are as follows:

step 1. Preprocessing: is a stage to process the raw data into a specific target area or Region of Interest (ROI).

step 2. Feature extraction: extracted using specific methods (using moment invariants).

step 3. Palm recognition: in this stage the extracted features will be compared with the database to find the matching user with test feature.

1.1.Preprocessing

Preprocessing is aimed to change the raw data , to assist computational capabilities in feature extraction as well as to reduce noise.

We used digital image processing techniques which intended to change the intensity of image pixels to produce a more suitable form of data for subsequent operations.

The image types considered are:

i. true color (Bmp image).

Bmp image have two types:

1) file header.

A Bmp file begins with a large header. its total size is (54) byte. It defined every think you could possible want to know about the file except the names of the objects, people, animals, etc.

2) Bitmap header.

Is sequence of 3-byte color ,one for red, one for green, one for blue.

ii. gray-scale image.

The gray image contain brightness information only, no color information. the number of bits is used for each pixel determines the number of different brightness levels available. The typical image contain 8 bits/pixel data, which allow us to have 256 (0-255) different brightness (gray) levels.

iii. Binary image.

is the simplest type of image and can take on two values, typically black and white, or '0' and '1'. A binary image is referred to as 1bit/pixel image because it takes only 1 binary digit to represent each pixel. Binary images are often created from gray-scale images via threshold operation, where every pixel above the threshold value is turned white ('1'), and those below it are turned black ('0').

In preprocessing, the image captured by the sensor will be normalized in order to prepare the feature extraction.

Standard techniques were used in the preprocessing including:

Is refers to a monochrome ,or one-color image, covert RGB image To gray level image , which have allows us to have 256 (0-255) Different brightness (gray) levels.

1.1) Binarization (thresholding using average method)

Created from Gray-scaling image via a threshold operation, where every pixel is turned white '1' or turned black '0'.

1.2) Median filtering

Is one of the best edges preserving smoothing filters. it consist of Sliding window encompassing an odd number of pixels. The center Pixel in the window is replaced by the median of the pixels within The window.

In order to perform median filtering in a neighborhood of a pixel; We first sort the value of the central pixel. The principal function Of the median filter is to force points with very distinct intensities To be more like their neighbors, thus actually eliminating intensity Spikes that appear isolated in the area of the filter mask.

1.3) Region of Interest (ROI) determination

Often for image analysis, we want to investigate more closely A specifics area within the image .

1.5) Pattern recognition:

are large research areas in the field of image processing and computer vision. Recognition is largely based on the matching of descriptions of shapes. Numerous shape description techniques have been developed, such as analysis of scalar features (dimensions, area, number of corners etc.). Fourier descriptors, moment invariants and boundary chain coding. These techniques are well understood when applied to images and have been developed to describe shapes irrespective of position, orientation and scale. They can be easily applied to vector graphical shapes. After that we describes experiments, which apply moment invariants to the problem.

Moment invariants:

Moment invariants have been frequently used as features for image processing, remote sensing, shape recognition and classification. Moments can provide characteristics of an object that uniquely represent its shape. Invariant shape recognition is performed by classification in the multidimensional moment invariant feature space. Several techniques have been developed that derive invariant features from moments for object recognition and representation. These techniques are distinguished by their moment definition, such as the type of data exploited and the method for deriving invariant values from the image moments. It was Hu that first set out the mathematical foundation for two-dimensional moment invariants and demonstrated their applications to shape recognition. They were first applied to aircraft shapes and were shown to be quick and reliable ,These moment invariant values are invariant with respect to translation, scale and rotation of the shape.

Chapter two

"system layout"

2.1.System design :

The system layout is summarized in the following shape:

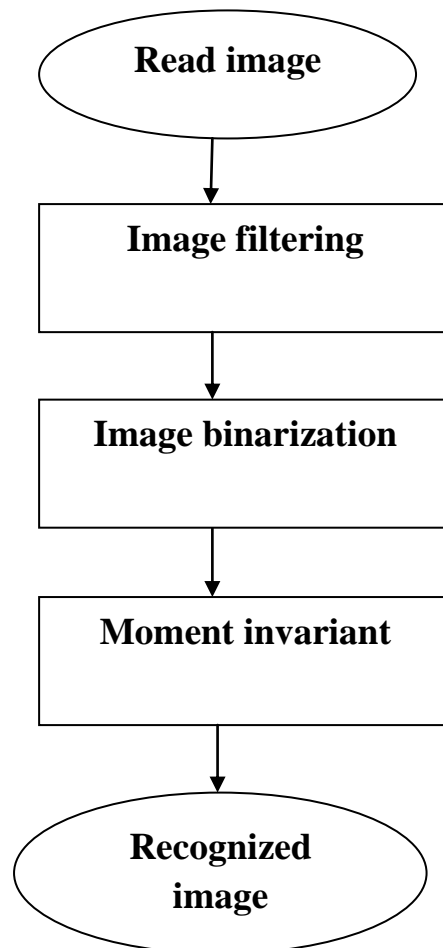


Figure (2.1)

2.2 Image filtering

In this work, the median filter has been used, which replaces the value of a pixel by the median of the gray levels in the neighborhood of that pixel (the original value of the pixel is included in the computation of the median). Median filter are quite popular because for certain types of random noise, they provide excellent noise-reduction capabilities, with considerably less blurring than linear smoothing filters of similar size. Median filter are particularly effective in the presence of impulse noise, also called salt- and-pepper noise because of its appearance as white and black dots superimposed on an image. in order to performed median filtering at a pixel in an image ,the following steps will be used:

- 1- First take 3*3 windows.
- 2- Sort the values of the pixel in window and its neighbors.
- 3- Determine their median, and assign this value to the central pixel.

2.3) Image binarization algorithm:

Input:

The filtered image
Height of image
width of image

output:

threshold

steps:

step1: calculate threshold using the following equation

$\text{threshold(average)} = \text{round}((\text{max of color} - \text{min of color}) * 0.3) + \text{min of color.}$

Step2: loop from 0 to height

Step3: loop from 0 to width

Step4: check pixel in image is > average

Then add the pixel value '1' to add1

Else add the pixel value '0' to add1

Move the value in add1 to any variable to be used.

2.4.moment invariant

2.4.1.background

Hu defines seven of these shape descriptor values computed from central moments that are independent to object translation, scale and orientation. Translation invariance is achieved by computing moments that are normalized with respect to the centre of gravity so that the centre of mass of the distribution is at the origin (central moments). Size invariant moments are derived from algebraic invariants but these can be shown to be the result of a simple size normalization. From the second and third order values of the normalized central moments a set of seven invariant moments can be computed which are independent of rotation.

2.4.2.theory

Traditionally, moment invariants are computed based on the information provided by both the shape boundary and its interior region. The moments used to construct the moment invariants are defined in the continuous but for practical implementation they are computed in the discrete form. Given a function $f(x,y)$, these regular moments are defined by:

$$M_{pq} = \iint x^p y^q f(x,y) dx dy \quad (1)$$

M_{pq} is the two-dimensional moment of the function $f(x,y)$. The order of the moment is $(p + q)$ where p and q are both natural numbers. For implementation in digital form this becomes:

$$M_{pq} = \sum_x \sum_y x^p y^q f(x,y) \quad (2)$$

To normalize for translation in the image plane, the image centroids are used to define the central moments. The co-ordinates of the centre of gravity of the image are calculated using equation (2) and are given by:

$$\bar{x} = \frac{M_{10}}{M_{00}} \quad \bar{y} = \frac{M_{01}}{M_{00}} \quad (3)$$

The central moments can then be defined in their discrete representation as:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q$$

(4)

The moments are further normalized for the effects of change of scale using the following formula:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}$$

(5)

Where the normalization factor $\gamma = (p + q / 2) + 1$. From the normalized central moments a set of seven values can be calculated and are defined by:

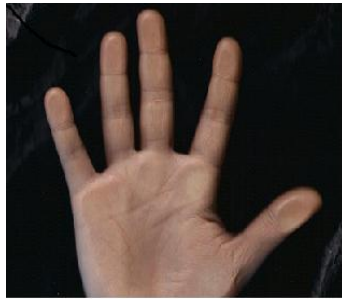
$$\begin{aligned}\phi_1 &= \eta_{20} + \eta_{02} \\ \phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\ \phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (\eta_{03} - 3\eta_{21})^2 \\ \phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{03} + \eta_{21})^2 \\ \\ \phi_5 &= (3\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 \\ &\quad - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \\ &\quad \times [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ \phi_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ &\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\ \phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 \\ &\quad - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03}) \\ &\quad \times [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{30})^2]\end{aligned}$$

(6)

These seven invariant moments ϕ_I , $1 \leq I \leq 7$ set out by Hu, were additionally shown to be independent of rotation. However they are computed over the shape boundary and its interior region.

Experimental & Result :

Experimental(1)



(a)original palm image



(b)gray scaling image



(c)binary image



(d) image after filtering



(e) Region of Interest (ROI).

Figure(2.2) preprocessing: convert the BMP color image to gray scale image by gray-scaling operation after that we use thresholding (binarization) and filtering to the gray scale image and get the (ROI) by removal finger to be ready used in moment invariant theory

Chapter three

"the practical part "

Gray scaling image

```
for i:= hight1-1 downto 0 do
for j := 0 to width1-1 do
begin
blockread(f,b,1);
blockread(f,g,1);
blockread(f,r,1);
b1[i,j]:=b;
g1[i,j]:=g;
r1[i,j]:=r;
gray[i,j]:=(b1[i,j]+g1[i,j]+r1[i,j]) div 3;
x:=gray[i,j];
image1.Canvas .pixels[j,i]:=rgb(r,g,b); {realimage}
image2.canvas.pixels[j,i]:= rgb(x,x,x); {gray image}
end;
```

min & max no. of image

```
max:=gray[0,0];
min:=gray[0,0];
for i:= 0 to hight1-1 do
for j := 0 to width1-1 do
begin
if gray[i,j]>max then max:=gray[i,j];
if gray[i,j]<min then min:=gray[i,j];
end;
memo1.Lines.Add(inttostr(max)) ;
memo2.Lines.Add(inttostr(min))
```

Binarization

```
th := round((max-min)*0.3) + min;
for i:= 0 to hight1-1 do
for j := 0 to width1-1 do
begin
if gray[i,j] >= th then
binary[i,j]:= 1
else
binary[i,j]:= 0;
t:=binary[i,j];
if t=1 then t:=255;
image3.canvas.pixels[j,i]:= rgb(t,t,t);
end;
end else exit end;
```

Filtering image

```
for i:= 1 to hight1-2 do
for j:= 1 to width1-2 do
begin
sum:=0;
for xx:=i-1 to i+1 do
for yy:=j-1 to j+1 do
sum:=sum+binary[xx,yy];
palm[i,j]:=round(sum / 9);
if palm[i,j]>=1 then palm[i,j]:=1
else palm[i,j]:=0;
if palm[i,j]=1 then r:=255 else r:=0;
image4.canvas.pixels[j,i]:= rgb(r,r,r);
end;
```

Remove the finger

```
s1:=140;
for i:= 0 to hight1-1 do
begin
f:=0; x1:=0;
for j:= 0 to width1-1 do
begin
if((f=0) and(palm[i,j]=1)) then
begin
f:=1 x1:=j
end
else if (f=1) and (palm[i,j]=0) then
begin
f:=0 x2:=j
dx:=abs(x2 - x1);
if dx < s1 then
begin
for i1:= x1 to x2 do
begin
palm[i,i1]:=0
image4.canvas.pixels[i1,i]:= 0
end
end
end
end
```

The moment invariant

```
for m:= 0 to 3 do
for n:= 0 to 3 do
begin
mom[m,n]:= 0;
end;
smx:=0;
smy:=0;
sm:=0;
for y:=0 to hight1-1 do
for x:=0 to width1-1 do
begin
if palm[x,y] <> 0 then
begin
smx:=smx + x * palm[x,y];
smy:=smy + y * palm[x,y];
sm:=sm + palm[x,y];
end;
end;
*****

Mom[0, 0] := Sm;
Mom[1, 0]:= SmX ;
Mom[0, 1] := SmY;
Xmean := SmX / Sm;
Ymean := SmY / Sm;
mom[1,1]:=0;
mom[1,2]:=0;
mom[0,2]:=0;
mom[2,0]:=0;
mom[2,1]:=0;
mom[3,0]:=0;
mom[0,3]:=0;

for y:=0 to hight1-1 do
for x:=0 to width1-1 do
begin
if palm[x,y]<>0 then
begin
xp1:=x - xmean;
xp2:=xp1 * xp1;
yp1:=y-ymean;
yp2:=yp1*yp1;
mom[1,1]:=mom[1,1]+xp1*yp1*palm[x,y];
mom[2,0]:=mom[2,0]+xp2*palm[x,y];
mom[0,2]:=mom[0,2]+yp2*palm[x,y];
mom[2,1]:=mom[2,1]+xp2*xp1*palm[x,y];
```

```

mom[1,2]:=mom[1,2]+xp1*xp2*palm[x,y];
mom[3,0]:=mom[3,0]+xp2*xp1*palm[x,y];
mom[0,3]:=mom[0,3]+yp2*yp1*palm[x,y];
end;
end;
*****

mu[0,0]:=mom[0,0];
mu[0,1]:=0;
mu[1,0]:=0;
mu[1,1]:=mom[1,1]-xmean*mom[0,1];
mu[2,0]:=mom[2,0]-xmean*mom[1,0];
mu[0,2]:=mom[0,2]-ymean*mom[0,1];
mu[2,1]:=mom[2,1]-2*xmean*mom[1,1]-
ymean*mom[2,0]+2*xmean*xmean*mom[0,1];
mu[3,0]:=mom[3,0]-3*xmean*mom[2,0]+2*xmean*xmean*mom[1,0];
mu[0,3]:=mom[0,3]-3*ymean*mom[3,0]+2*ymean*ymean*mom[0,1];
for m:=0 to 3 do
for n:=0 to 3 do
begin
if (m<>0)and(n<>0) then
eta[m,n]:=mu[m,n]/ (math.Power(mu[0,0],(1+(m+n)/ 2)));
end;
assignfile(f4,'d:\str.txt');
append(f4);
str1:=inputbox('please enter your name','name','');
if str1='yes' then
recognize
else
begin
writeln(f4,str1);
*****

q[1]:=eta[2,0]+eta[0,2];
writeln (f4,floattostr(q[1]));

q[2]:=((eta[2,0]-(eta[0,2]*eta[2,0]))+4*eta[1,1]*eta[1,1]);
writeln (f4,floattostr(q[2]));

q[3]:=eta[3,0]-3*(eta[1,2]*eta[1,2])+((3*eta[2,1]-eta[0,3])*(3*eta[2,1]-eta[0,3])) ;
writeln (f4,floattostr(q[3]));

q[4]:=((eta[3,0]+eta[1,2])*(eta[3,0]+eta[1,2]))+((eta[2,1]+eta[0,3])*(eta[2,1]+eta[0,3]
));
writeln (f4,floattostr(q[4]));

q[5]:=eta[3,0]-
3*eta[1,2]*((eta[3,0]+eta[1,2])*(eta[3,0]+eta[1,2])*(eta[3,0]+eta[1,2]))-
3*((eta[2,1]+eta[0,3])*(eta[2,1]+eta[0,3]))+(3*((eta[3,0]+eta[1,2])*(eta[1,2]+eta[3,0]
))-
((eta[2,1]+eta[0,3])*(eta[2,1]+eta[0,3])));
writeln (f4,floattostr(q[5]));

```

```

q[6]:= (eta[2,0]-eta[0,2])*(((eta[3,0]+eta[1,2])*(eta[3,0]+eta[1,2]))-
((eta[2,1]+eta[0,3])*(eta[2,1]+eta[0,3]))) + 4*eta[1,2]*(eta[3,0]+eta[1,2])*(eta[2,1]+eta
[0,3])*(eta[3,0]+eta[1,2])*(eta[2,1
]+eta[0,3]);
writeln (f4,floatostr(q[6]));

```

```

q[7]:= (3*eta[2,1]-
eta[0,3])*(eta[3,0]+eta[1,2])*((eta[3,0]+eta[1,2])*(eta[3,0]+eta[1,2]))-
3*((eta[2,1]+eta[0,3])*(eta[2,1]+eta[3,0])) + (eta[3,0]-
3*eta[1,2])*(eta[2,1]+eta[0,3])*(3*((eta[3,0]+eta[1,2])*(eta[3,0]+eta[1,2]))-
((eta[2,1]+eta[0,3])*(eta[2,1]+eta[0,3])));
writeln (f4,floatostr(q[7]));

```

```

q[8]:= eta[1,1]*(((eta[3,0]+eta[1,2])*(eta[3,0]+eta[1,2]))-
((eta[0,3]+eta[2,1])*(eta[0,3]+eta[2,1]))) - (eta[2,0]-
eta[0,2])*(eta[3,0]+eta[1,2])*(eta[0,3]+eta[2,1]);
writeln (f4,floatostr(q[8]));

```

```
closefile(f4);
```

```
end;
```

```
*****
```

recognize

```

procedure recognize;
var
i,j,z,u1,g,sum:integer;
min1,r:real;
str1,str2,k:string ;
begin
reset(f4);
i:=1;
while(not eof(f4)) do
begin
Readln (f4,k);
p1[i].name :=k;
for z:=1 to 8 do
begin
Readln (f4,k);
r:=strtofloat(k);
p1[i].q[z]:=r;
end;
i:=i+1;
end;
closefile(f4);
for j:=1 to i-1 do
begin
sum:=0;
for z:=1 to 8 do

```

```
sum:=sum+round(sqr(p1[j].q[z]-q[z]));
d[j]:=sum;
end;
min1:=d[1];
u1:=1;
for g:= 2 to i-1 do
begin
if d[g]<min1 then
begin
min1:=d[g];
u1:=g;
end;
end;
showmessage('the person is '+p1[u1].name );
end;
```


Chapter four

" conclusions "

Conclusions:

1 - Take the original image and is converted to gray image by separating the image to the colors red, green and blue in the three matrices, then the rate matrices were found and then draw the image.

2- Take the matrix containing the values of the gray scaling image is to find the min and max values of the image and then is to find the value of the threshold to be comparable with the pixels of the matrix, if greater than 1 we put in the matrix of the binary value of '1 'and if you do not put a '0' . The result is a matrix consisting of values of black and white or 0.1 to make it easier to deal with the data during the calculation values of the moment.

3- The candidate Mediterranean compile these numbers in the vicinity of a particular pixel on if compile all the numbers in the pixels adjacent to a given pixel. Then, the order of the candidate to take these numbers and the number is in the middle of the list. This number, which stands for a particular color is used for any pixel on the pixel takes on the color.

4- Normalized central moments a set of eight invariant moments can be computed which are independent of rotation. This is done using the moment invariant for pattern recognition .

5- Cut image and take part of the palm just because it is calculated from which the values of the moment, then eight values are calculated through the equations mentioned these values are independent of the rotation in terms of size, direction, change the location does not change the form of the body.

6-Recognize, are marked with the person inside the system through the work of checking the values of the moment for many people, if it is identical to the one he is allowed to enter and if no match is not allowed to enter the system.

References:

**USING MOMENT INVARIANTS FOR CLASSIFYING SHAPES ON
LARGE_SCALE MAPS (Laura Keyes ,Adam Winstanle)**

**AN APPROACH TO OFFLINE ARABIC CHARACTER RECOGNITION
USING NEURAL NETWORKS (S. N. Nawaz, M. Sarfraz, A. Zidouri, and W. G.
Al-Khatib) king Fahd University of Petroleum and Minerals**

Elsevier – 2009 ,Recognition A Kong, D Zhang... - Pattern

Elsevier - 1993 ,Pattern recognition - ...J Flusser

(Dudani Breeding and McGhee, 1977)

(Hu, 1962 Prokop and Reeves, 1992)

<http://eprints.nuim.ie/64/1/ceus.pdf>

http://zoi.utia.cas.cz/moment_invariants

<http://www.cs.nthu.edu.tw/~cchen/Research/1993PRmoment.pdf>

<http://www.freepatentsonline.com/article/American-Journal-Applied-Sciences/156719908.html>

<http://onlinelibrary.wiley.com/doi/10.1111/j.1600-065X.2008.00731.x/abstract?deniedAccessCustomisedMessage=&userIsAuthenticated=false>

http://scholar.google.com/scholar?q=palm+print+recognition&btnG=&hl=ar&as_sdt=0&as_vis=1