Int. J. Open Problems Compt. Math., Vol. 2, No. 2, June 2009

# Indian Coin Recognition System of Image Segmentation by Heuristic Approach and Houch Transform (HT)

**C.M.VELU[1] and P.VIVEKANANDAN[2]**

**[1] HOD of CSE, SKR Engineering College, Chennai – 602 103, India.**
**[2]Director, Knowledge Data Centre, Anna University, Chennai – 600 025, India.**
**e-mail: cmvelu41@gmail.com, vivek@annaunive.edu**

**Abstract**

***The objective of this paper is to classify the Indian coins of different denomination released recently. This paper is framed mainly to classify the coins offered in the Hundi by the devotees of Tirumala Tirupati Devasthanam(TTD), Tirupati, India. The objective is to count money by recognizing the coins and count the total sum based on its value. The system is proposed to design coin recognition by applying heuristic approach, based on the coin table. This table stores parameters of each coin. This method yields 97% of result in recognizing the coin image. It is also proposed to apply HT algorithm combining the features of a) Straight line detection HT algorithm, b) Curve detection HT algorithm and c) Circle detection HT algorithm. Using these three algorithms edge of the coin is recognized. The features of old coins and new coins of different denominations are considered for classification. Some coins are used in different countries have same parameters, but it has different value. This paper concentrates on affine transformations such as simple gray level scaling, shearing, rotation etc. The coins are well recognized by zooming processes by which a coin size of the image is increased. This paper presents a coin recognition method with rotation invariance. Indian Coins are classified based on different parameters for various values of coin such as shape, size, surface***

***design, weight and so on. Hence, it is easy for the automatic machine to classify Indian coins.. There are many coin classification machines are available, but , the machine has to be designed for recognition of Indian coin. To increase the efficiency of the machine, they are to be embedded with proper source code.In this paper, Sobel Filter, HA and HT are used to classify the coin image. It is proposed a method for realizing a simple automatic coin recognition system more effectively. The HT technique is used to recognize almost 100% of the coin image. Comparing to Sobel edge detection method, HA the HT gives better result.***

**Keywords***: Smoothing, Edge detection, Thresholding, Recognition, Classification*..

# 1 Introduction

Automatic machines are used for Coins classification and recognition to find the sum of the coin is quite complicate. In this paper, it is proposed a coin recognition method by heuristic approach. The effectiveness of the coins classification is ensured based on the parameters of the coin. Moreover, the variations in images obtained from new and old coins are also discussed. The polar coordinate image of coin on circles with different radii is used as the feature for coin recognition. Finally, the knowledge base of the coin is fed to the recognition system to classify the coin easily. For a given coin, image is scanned and stored as gray values in a matrix. The intensity of the pixel is maximized. The heuristic approach is adapted to classify the coin after pre-processing of the image. We present a vision-based approach to coin classification which is able to discriminate between many coin classes. The approach described is a multistage procedure. In the first stage a translationally and rotationally invariant description is computed. Correct decision into one of the different coin classes and the rejection class, i.e., correct classification or rejection, was achieved for 99.23% of coins in a test sample containing 10,000 coins. False decisions, i.e., either false classification, false rejection or false acceptance, were obtained for 0.67% of the test coins. The classification of these coins according to their denomination in the field of application for the method is presented in this paper.

The basic idea of the method can be seen from the coin image of detecting a straight line in coin image [Duda and Hart 72]. The criteria for coin classification can be based on gray-level, color, texture, shape, model, etc, are discussed by R.Bremananth[1]. The method which specifically addresses coin segmentation based on color or gray value is reported by P.Thumwarin and Petra Perner [2,6]. Many serious problems like shape, peak detection in surface of the coins are reported by Reinhold Huber[3]. An attempt has been made to use Heuristic and HT techniques to recognize most of the coin images.

In Section 2, coin feature extraction is discussed. Section 3 deals with coin segmentation based on labeling. Section 4 discusses coin classification using HT. Section 5, presents conclusion and result.

# 2 Coin Feature Extraction

The coin classification technique is successful based on the following assumptions and computations.

- i) The coins are moved on a conveyor belt.
- ii) Proper lighting is focused on the coin.
- iii) Each coin is separated and fed to the system for recognition.
- iv) Coins are weighed accurately.
- v) The coin images are collected both sides of the coin.
- vi) It is possible to capture the side view of the coin image.
- vii) The coin image can be rotated by any degree.
- viii) The Circular, Hexagon, Octagon, Polygon shape of coin's radius are measured
- ix) The coin Circumference/Perimeter and area are computed.
- x) The thickness of each coin is given to the system.
- xi) The coin images with 256 gray values are collected.
- xii) The coin average gray values are computed.

## 2.1 Rotation of the Coin

The coin image is represented by the Cartesian Co-ordinates and it is given by

$$f(x,y) = \sum_{n=0}^{Q} \sum_{m=0}^{Q} F(n,m)\,\phi(x - n, y - m) \qquad (1)$$

where, $\phi(x - n, y - m) = (\sin \pi x / \pi x)(\sin \pi y / \pi y)$,

and Q is an integer related to the image size. F(n,m) is the gray level at a pixel (n,m). Then the following function is defined by letting $x=r\cos\theta$, $y=r\sin\theta$ in equation(1) as

$$\hat{f}(r, \theta) = f(r\cos\theta, r\sin\theta), \qquad (2)$$

where $0 \le r \le R$, $0 \le \theta \le 360^{o}$ and $\hat{f}(r, \theta)$ defined inside the circle of radius R for coin recognition, in case of same size coins, the design of coin surface and weight of the coin are the important features[2]. To extract the rotation invariance feature of the coin image, it is assumed that,

$$\hat{f}(r, \theta) \text{ as } \hat{f}(r, \theta) = \hat{f}(r, \theta+2m\pi) \qquad (3)$$

where m=0,1,2,….. and $\hat{f}$ (r, θ) is expanded into Fourier series by letting r = $r_k$(Constant) as

$$\hat{f}(r_k, \theta) = \sum_{m=-M}^{M} a_m^k e^{im\theta} \quad (4)$$

where , $a_m^k = \frac{1}{2\pi}\int_0^{2\pi} \hat{f}(r_k, \theta)\, e^{-im\theta}\, d\theta$

where k is the radius of the coin image.

$$\text{Let } \hat{g}(r_k, \theta) = \hat{f}(r_k, \theta+\alpha) \quad (5)$$

be the rotated seal imprint of $\hat{f}$ ($r_k$, θ) by angle $\alpha$ about its origin. Then it can be seen that Fourier coefficients $b_m^k$ of $\hat{g}$ ($r_k$, θ) is given as $b_m^k = a_m^k\, e^{-im\alpha}$ (6)

Thus, the relation becomes $u_m^k = |b_m^k| = |a_m^k|$. It can be seen from the above equation that the absolute value of Fourier coefficients, $u_m^k$ of circle which is derived from the coin with different radii are rotation invariant [2]. Simple rotation of image involves movement of a coin to certain degree of rotation, by which one can match the coin with the existing knowledge base of coin image. A function that uses this scheme for rotating an angle of θ degree to the right into an image is given in MATLAB code.

## 2.2 Coin Image Magnification

Zooming and de-zooming are processes by which a coin image is increased or decreased in size, the special technique is introduced to recognize 100% of the coin image. The zooming helps us to make bigger size of a coin image, by which recognition rate is increased. The Coin Recognition of Pre-Processing of various stages like cropping, scaling, resizing, rotation are done in the following Matlab code.

```
% Coin Recognition Code Pre-processing
%Reading the coin image
img = imread('c1.bmp'); figure ; imshow(img) ;
%To convert into Gray value
imgGray = rgb2gray(img); figure ; imshow(imgGray) ;
% Manual Cropping  of coin image
imgCrop = imcrop(imgGray); figure ; imshow(imgCrop) ;
% Resizing of coin  image
imgLGE = imresize(imgCrop, 5, 'bicubic'); figure ;
imshow(imgLGE) ;
% Rotation of the coin image
imgRTE = imrotate(imgLGE, 35); figure ; imshow(imgRTE) ;
% Binary Image of the coin
imgBW = im2bw(imgLGE, 0.90455); figure ; imshow(imgBW) ;
```

## 2.3 Coin Centroid, Circumference and Area Computation

The area of a coin image is calculated by total number of pixels representing the coin image. The perimeter is the number of pixels along the outer contour of the coin. Using radius or side of coin image, we can compute circumference or perimeter of the coin depending upon its shape namely circle, square, hexagon, octagon, polygon using suitable formula. The centroid of the coin may be calculated as, $X_c=\sum X_i/n$, $Y_c=\sum Y_i/n$, where, i=1 to n. A is the area of the coin, N is the number of pixels in the coin, X,Y are the coordinates of the pixel in the coin. A function to calculate the centroid of a binary image of coin is given by the Pseudo code as follows:

```
centroid (float*X-C,*Yc)
{
 int area=0, i,j;
 *Xc=0.0; *Yc=0.0;
   for(i=0; I<no_of_cols; ++i)
     {
          for (j=0;j<no_of_rows; ++j)
          {
               if (x[i][j]>0)   /* test for a coin */
                {
                          /* accumulate x,y coordinates */
                    *Xc =*Xc+i ;
                    *Yc=*Yc+j ;
                           area=area+1;
                       }
          }
     }
       *Xc=*Xc/area;
      *Yc=*Yc/area;
}
```

## 2.4 Data Acquisition

Usually the ordinary Cartesian coordinate system is used to represent a pixel of an image. In this system, *g(x, y)* is the gray level at the pixel *(x, y)*. Images can alternatively be thought of as ordinary matrices in which the gray level of a pixel is represented as $g_1(i, j)$. *The coin* table stores values for parameters of each coin. A knowledgebase of the coins are fed to the system.

Table 1: Coin parameter table

| Coin Value in Paise | Type of Coin | Coin Diameter/side in mm | Coin Shape | Coin Weight (grams) | Coin area In $Cm^2$ | Coin average gray value | Coin Thickness In mm |
|---|---|---|---|---|---|---|---|
| 5 | Old | 18 | Square | 2.00 | 2.4500 | 250 | 1.1000 |
| 5 | New | 18 | Square | 2.00 | 2.4500 | 250 | 1.1000 |
| 10 | Old-1 | 26 | Octagon | 2.24 | 5.3114 | 225 | 1.0000 |
| 10 | Old-2 | 24 | Polygon | 1.73 | 4.5257 | 225 | 1.0000 |
| 10 | New | 16 | Circle | 1.99 | 2.0114 | 75 | .1.0000 |
| 20 | Old | 25 | Hexagonal | 2.21 | 4.0595 | 240 | 1.2000 |
| 20 | New | 25 | Circle | 2.30 | 4.5590 | 240 | 1.2000 |
| 25 | Old | 20 | Circle | 2.95 | 3.1428 | 90 | 1.0000 |
| 25 | New | 21 | Circle | 2.82 | 3.1467 | 90 | 1.0000 |
| 50 | Old | 24 | Circle | 4.98 | 4.5257 | 100 | 1.6000 |
| 50 | New | 22 | Circle | 3.72 | 3.8028 | 100 | 1.6000 |
| 100 | Old | 27 | Circle | 5.97 | 5.7278 | 100 | 1.7000 |
| 100 | New | 26 | Circle | 4.93 | 5.3114 | 100 | 1.7000 |
| 200 | Old | 27 | Polygon | 5.91 | 5.7278 | 110 | 1.8000 |
| 200 | New | 27 | Circle | 7.72 | 6.1600 | 110 | 1.8000 |
| 500 | Old | 24 | Circle | 9.03 | 4.5257 | 200 | 3.0000 |
| 500 | New | 24 | Circle | 6.14 | 4.5257 | 200 | 1.5000 |

## 2.5 Heuristic approach

To classify the coin it is proposed to use heuristic approach based on various parameters such as shape, weight, area, average gray value, thickness and diameter etc. The binary image of coins are shown in fig(1).

## 2.6 Coin Gray Scale Range Normalization

If the original image had gray levels ranging from a which is the lowest, to b which is the highest, and we want to make the range c to *d,* the change could be made in three steps:
***Step 1***. Subtract *a* from each gray level to make the range as 0 to *b - a.*
***Step 2***. Multiply the result by *(d – c)/(b - a)* to make the range as 0 to *d* - c.
***Step 3***. Add c to the result obtained from step 2, to obtain the range c to *d.*
These three steps are summarized to convert from the range [a, b] to the range [c, d].

$$g_2(x, y) = ( (d-c) / (b - a))[g_1(x, y) - a] + c. \qquad (7)$$

The coin grayscale image is shown in fig.(2).

## 2.7 Thresholding

The thresholding transformation sets each gray level <=T is set to 0, and each gray level > T is changed to *K* - 1. Thresholding is useful when one wants to separate bright coin image from a darker background or vice versa[8]. The thresholding transformation is shown in fig.(3) and is defined by

$$g_2(x,y) = \begin{cases} 0 & \text{if } g_1(x,y) <= T \\ K-1 & \text{if } g_1(x,y) > T \end{cases} \qquad (8)$$

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 7 | 0 |
| 0 | 7 | 7 | 7 | 0 |
| 0 | 7 | 7 | 7 | 0 |
| 0 | 0 | 0 | 0 | 0 |

Fig.(3) Thresholded Image.

The recognition of coins image can be classified into four classes based on gray levels according to $g_2(x,y)$, as follows, but, it is not giving accurate results[8].

$$g_2(x,y) = \begin{cases} 64, & \text{if } 0 <= g_1(x,y) <= 63 \\ 128, & \text{if } 64 < g_1(x,y) <= 127 \\ 192, & \text{if } 128 < g_1(x,y) <= 191 \\ 255, & \text{if } 192 < g_1(x,y) <= 255 \end{cases} \qquad (9)$$

## 2.8 The Inverse Transform

The inverse transformation reverses light as dark and dark as light. If the original image is denoted by $g_1(x,y)$ and the new image denoted by $g_2(x,y)$, then the inverse transformation is defined by $g_2(x,y) = K - 1 - g_1(x,y)$ (10)
If $K = 256$, then black, represented by gray level 0, is transformed to white, represented by gray level 256 - 1 - 0 = 255. If the inverse transformation is applied to the image, assume $K = 8$, the resulting image shown in fig.(6).

| | | | | |
|---|---|---|---|---|
| 6 | 5 | 5 | 5 | 5 |
| 4 | 5 | 3 | 2 | 5 |
| 5 | 1 | 1 | 0 | 7 |
| 5 | 1 | 1 | 2 | 6 |
| 7 | 5 | 4 | 5 | 6 |

Fig.(6). Inverse Transform image in graylevel.

## 2.9 Coin Brightness and Contrast Normalization

The contrast in an image is the amount of variation of its gray levels. One way of quantifying this variation is by the root-mean-squared difference of the gray levels from their mean, and standard deviation of the gray levels[8]. The average gray level in an $M$ x $N$ image $g_1(x, y)$ is

$$\mu = \frac{1}{M}\frac{1}{N}\sum_{x=1}^{M}\sum_{y=1}^{N} g_1(x, y) \qquad (11)$$

and the standard deviation of the gray levels is

$$\sigma = \sqrt{\frac{1}{M}}\sqrt{\frac{1}{N}}\sum_{x=1}^{M}\sum_{y=1}^{N} \{ g_1(x, y)-\mu \}^2 \qquad (12)$$

The following transformation will convert the image $g_1(x, y)$ with parameters $\mu$ and $\sigma$ into one with a new mean gray level $\mu_n$ and a new standard deviation $\sigma_n$ is given by

$$g_2(x, y) = [\{ g_1(x,y) - \mu \} / ( \sigma / \sigma_n )] + \mu_n \qquad (13)$$

Alternatively, brightness and contrast could be defined to be the median gray level and the Mean Absolute Deviation (MAD) of the gray levels from the median, where,

$$\text{MAD} = \frac{1}{M}\,\frac{1}{N}\sum_{x=1}^{M}\sum_{y=1}^{N} \left| g(x,y) - \text{median} \right| \qquad (14)$$

To adjust the brightness and contrast with the median and MAD, where $\mu$ and $\mu_n$ now represent the old and new medians, and $\sigma$ and $\sigma_n$ now represent the old and new MADs. If $K$ - 1 is the largest possible value for a display device and $g_1(x, y) > K - 1$ in an image, then the display device shows that pixel at the brightest gray level possible is $K$ - 1.

# 3 Coin Segmentation and Labeling

The algorithm scans, when an unlabeled pixel (x, y) is found, the algorithm will label all the pixels in the 4 connected region to which (x, y) belongs before it labels any pixels from other regions. We first obtain a new label *L*. We then label (x, y) as *L* and add (x, y) to an initially empty list of pixels whose neighbors are to be checked later. Next we remove the pixel (s,t) least recently placed in the list. We next label with *L* each unlabeled 4-neighbor of (s, t) that has the same gray level as (s, t) and insert each such 4-neighbor in the list. We then repeat this process. If the list is not empty, we remove from the list the pixel (s, t) least recently placed in the list. We next label with *L* each unlabeled 4-neighbor of (s, t) that has the same gray level as (s, t) and insert each such 4-neighbor in the list. If unlabeled pixel found, we obtain a new label and restart the labeling process.

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 8 | 4 | 8 | | 1 | 1 | 1 | | -1 | -1 | -1 |
| 4 | 0 | 4 | | 1 | 2 | 2 | | -1 | 2 | 2 |
| 8 | 4 | 8 | | 2 | 1 | 1 | | 2 | 1 | 1 |

(g) (h) (i)

Fig.(7). The center pixel denoted as 0 is 4-connected to its 4-neighbors (marked as 4) and 8-connected to all of its. 8-neighbors (marked as 4 or 8). Fig.(8). An image with regions to be labeled. Fig(9). The image after the region is labeled.

## 3.1 Region Labeling Algorithm

***Step 1:*** Let $g(x, y)$ represent the gray level of pixel $(x, y)$.

**Step 2:** As the algorithm executes, $g(x, y)$ is changed to the label of pixel $(x, y)$.

**Step 3:** Undefined gray levels outside the image such as $g(x, -1)$ and $g(-1, y)$ are considered to be unequal to any gray level in the image.

***Step 4:*** If an image has *n* pixels, the scanning part of the region-labeling algorithm takes *n* steps.

Since each pixel goes on the list once, the total number of times that the body of the "while" loop is executed is *n.* Thus, the body of the "for" loop is executed at most a total of *4n* times[7].

**Region Growing Pseudo-code**

```
Find seed pixel->x,y
Width=1,
height=1
while(hasGrown)
    {
        hasGrown=false
        if(pixel[x+width][y] to pixel[x+width][y+height-1] are set)
        {
                width ++
                hasGrown = true
        }
        if(pixel[x][y+height] to pixel[x+width-1][y+height] are set)
        {
                height ++
                hasGrown = true
        }
}
end while
```

## 3.1 Extracting features to classify labeled coin image

The coin’s image was thresholded and the pixels in each of the white regions were given a unique numeric label using the region-labeling algorithm. Next the area in pixels of each coin was computed by counting the number of pixels having each label. The average gray level for each coin was also computed. The Sobel edge detection of coin image is displayed in fig(10).

# 4 Hough Transform(HT)

The HT was initially developed to detect analytically defined shapes, such as lines, circles, or ellipses in general images. The generalized HT can be used to detect arbitrary shapes. However, the generalized HT requires the complete specification of the exact shape of the target coin to achieve precise segmentation. If an image represents coin with known shape and size, segmentation can be viewed as a problem of classifying this coin within an image. One of many possible ways to solve these problems is to move the mask with an appropriate shape and size along the image and the mask.

## 4.1 Straight line detection using HT

Let us consider all the possible lines which can go through an image point (s,t): t = m s + c. The parameters of all these lines form a straight line in the parameter space m, c. Both m and c can attain any value from $-\infty$ to $+\infty$. To parameterize the line is:

$$r = x \cos\theta + y \sin\theta \qquad (15)$$

where, $x = a + r \cos\theta$, $y = b + r \sin\theta$.

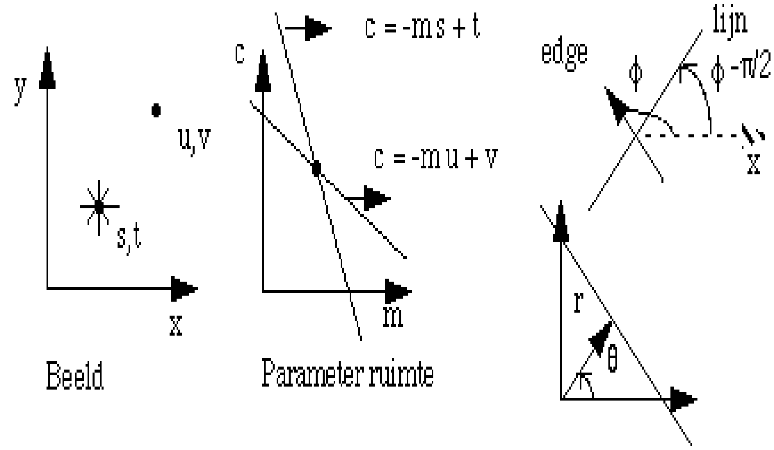
The $\theta'$ is from $-90°$ to $+90°$ and r is $\pm 1/2$ D (16)

where D is the diagonal of the image[9].

We have the following Hough algorithm to determine lines:

**HT line detection Algorithm**

***Step 1.*** Initialize $A(r_d, \theta_d)=0$ for all $r_d$ and $\theta_{d.}$
***Step 2.*** For every point (x,y) having a value > Threshold
***Step 3.*** Calculate the r's and θ's for all the possible lines through (x,y).
***Step 4.*** Discrete the values to $r_d$ and $\theta_{d.}$
***Step 5.*** Set $A(r_d, \theta_d) := A(r_d, \theta_d) + 1$ for all $r_d$ and $\theta_{d.}$
***Step 6.*** The local maximum in A yields the parameters of lines where a lot of points lie on.
***Step 7.*** For every point (x,y) with edge G(x,y) > Threshold and angle $\varphi$ : $m = t_g(\varphi - \pi/2)$ and $c = y - m x$.
***Step 8.*** If angle $\varphi$ is not exact, then take a range of ±45°, same for x,y: e.g. ±1




Fig(12). Line detection in the image is transformed to detection of local maxima

## 4.2 Curve detection using the HT

Generalization to more complex curves can be described by an analytic equation. Consider an arbitrary curve represented by an equation $f(x,$ a) = 0, where a is the vector of curve parameters. The processing results in a set of parameters of desired curves $f(x,$ a) = 0 that correspond to local maxima of accumulator cells in the parameter space; these maxima best match the desired curves. Parameters may represent unbounded analytic curves (for example, line, ellipse, parabola) etc., but to look for finite parts of these curves, the end points must be explicitly defined.

**Algorithm for Curve detection using the HT**

***Step 1.*** Quantize parameter space within the limits of parameters. The dimensionality *n* of the parameter space is given by the number of parameters of the vector a.

***Step 2.*** Form an n-dimensional accumulator array *A(a)* with structure matching the quantization of parameter space; set all elements to zero.

***Step 3.*** For each image point $(x_1, x_2)$ in the appropriately thresholded gradient image, increase all accumulator cells *A(a),* if $f(x,$ a) = 0 for all a inside the limits used in step 1.

$$A(a) = \mathrm{A(a)} + \Delta A. \qquad (17)$$

***Step 4.*** Local maxima in the accumulator array *A(a)* correspond to realizations of curves $f(x,$ a) that are present in the original image.

## 4.3 Circle detection through HT

Let the task be to detect a circle of a known radius r in an image. The method starts with a search for dark image pixel, after such pixel is found a locus potential center points of the circle associated with it can be determined. Such a locus potential center points forms a circle with the radius r [9]. If the loci of potential circle center are constructed for all dark pixel identified in the image, constructed for all dark pixels identified in the image, the frequency can be determined with which each pixel of the image space occurs as an element of the circle-center loci. The true center of the circle being sought is represented by the pixel with the highest frequency of occurrence in the circle-center loci. Thus, the center of the searched circle is determined. With the known circle radius, the coin image segmentation is performed [9].

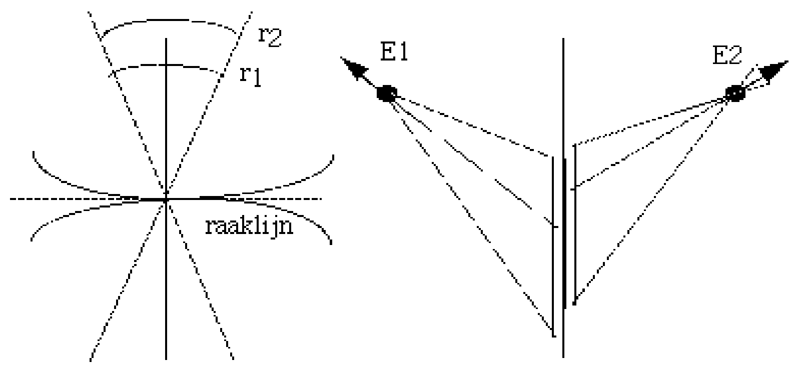



Fig (13). Catalogue of Hungarian denomination

A static r belongs to a 2-D parameter space A(a,b), a variable r belongs to a 3-D parameter space A(a,b,r). If we want to find both light and dark circles, two sides of every edge must be viewed. If we look at two edges in an image then the number of possible (a,b,r) values strongly decrease. The local maximums in the parameter space are easier to find. Within edge points in the image, there are n(n-1)/2 pairs to be viewed. Boundaries on r and testing on the φ's can restrict the number of (a,b,r) values to be calculated.

**Algorithm for Generalized HT**
**Algorithm for Curve detection using the HT**
***Step 1.*** Quantize parameter space within the limits of parameters a. The dimensionality *n* of the parameter space is given by the number of parameters of the vector **a**.
***Step 2.*** Form an n-dimensional accumulator array $A(\mathbf{a})$ with structure matching the quantization of parameter space; set all elements to zero.
***Step 3.*** For each image point $(x_1, x_2)$ in the appropriately thresholded gradient image, increase all accumulator cells $A(\mathbf{a})$, if $f(x, \mathbf{a}) = 0$, $A(\mathbf{a}) = A(\mathbf{a}) + \Delta A$ *(18)*

for all **a** inside the limits used in step 1.
***Step 4.*** Local maxima in the accumulator array *A(**a**)* correspond to realizations of curves *f(x,* **a**) that are present in the original image.

### 4.3 Coin Counting Machine

The application is suitable for an architecture of a coin counter system that incorporates a steady camera which monitor coins passing beneath the conveyor belt.

## 5 Conclusion and Results

The HT is very robust in the presence of additional structures in the image as well as being insensitive to image noise. Moreover, it may search for several occurrences of particular shape during the processing. The conventional sequential approach requires a lot of storage and extensive computation. Our scope is limited on recognizing only the Hungarian coins (Head OR Tail) in the denominations of 5, 10, 20, 25, 50,100,200 and 500 Pauses of Indian Coin. In this paper, the following three points are taken into the consideration for decreasing the maintenance cost. The perfect image of a coin is used for learning and recognition. The implementation to a real system ensures the following important points:

a) The Recognition rate is close to 100 percent.
b) It is a low cost system.
c) Recognition time is very less.

The proposed system by applying heuristic approach, based on the coin table yields 97% of result in recognizing the coin image. The HT algorithm combining the features of a) Straight line detection HT algorithm, b) Curve detection HT algorithm and c) Circle detection HT algorithm, we observe that the edge of the coin is recognized almost 100% of the coin image. Comparing to Sobel edge detection method the HT gives better results.

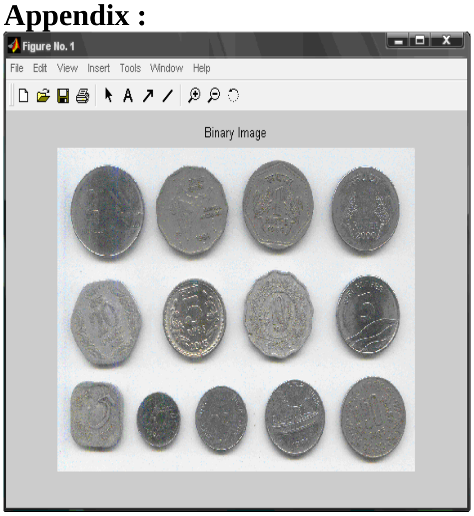
## 6 Open Problem

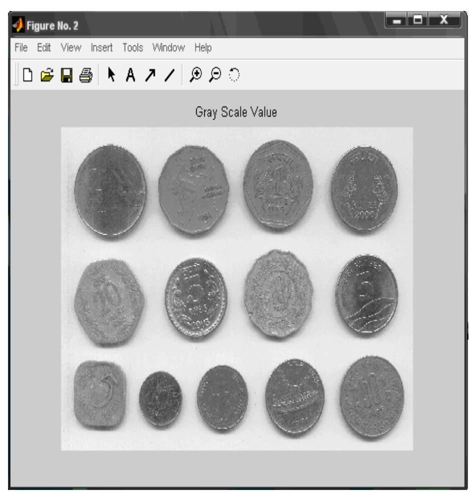
This paper can be extended to classify coins released during various time periods. Also, we can classify based on the coin shape, impression on the coin, metal of the coin etc. It may be used to measure similarity between a model and a detected coin on the basis of size and spatial location of peaks in the parameter space.

## References

[1] R.Bremananth, B.Balaji, M.Sankari, A.Chitra, 'A New approach to Coin recognition using Neural Pattern Analysis', IEEE Indicon Conference 2005, pp 366-370.
[2] P.Thumwarin, S.Malila, P.Janthawang, W.Pibulwej, T.Matsura, 'A Robust Coin Recognition method with rotation Invariance', IEEE, 2006, pp 520-523.
[3] Reinhold Huber, Herbert Ramoser, Konrad Mayer, Harald Penz, Michael Rubik, PRL, 26, 2005, 61-75.
[4] Yasue Mitsukura, Minoru Fukumi, Norio Akamatsu, 'Design and Evaluation of neural Networks for Coin Recognition by using GA and SA', IEEE, 2000, 178-183.
[5] Ph.A.Passeraub, P.A.Besse, C.de. Raad, O.dezuari, F.Quinet, R.S.Popovic,'Coin Recognition using an Inductive Proximity Sensor Microsystem', IEEE, 1997, 389-392.
[6] Petra Perner, 'Are case-based reasoning and dissimilarity-based classification two sides of the same coin?', Artificial Intelligence, 2002, 193-203.
[7] Minoru Fukumi, Sigeru Omatu, Fumiaki Takeda, Toshihisa Kosaka, 'Rotation Invariant Neural Pattern Recognition System with application to Coin Recognition', IEEE, 1992, 272-279.
[8] Earl Gose, Richard Johnson Baugh, Steve Jost, 'Pattern Recognition and Image Analysis',PHI, 1999.
[9] Milan Sonka, Vaclav Hlavac, Roger Boyle, 'Image Processing, Analysis, and Machine Vision', PWS Publishing Company, 1999.

**Appendix :**




Fig(1). Coin 5,10,20,25,50,100,200 and 500 Paise Fig.(2). Gray Scale value of the coin image

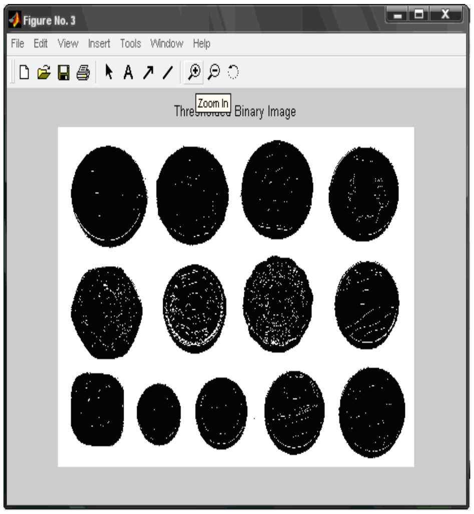



Fig.(4).Thresholded Binary Image.

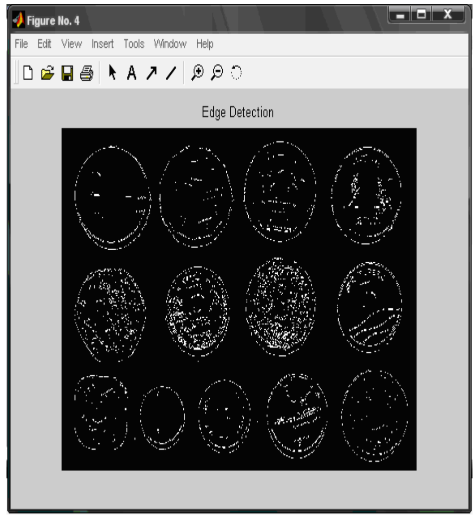



Fig.(10). Sobel Edge Detection.


Fig.(11). Coin recognition by Heuristic approach.

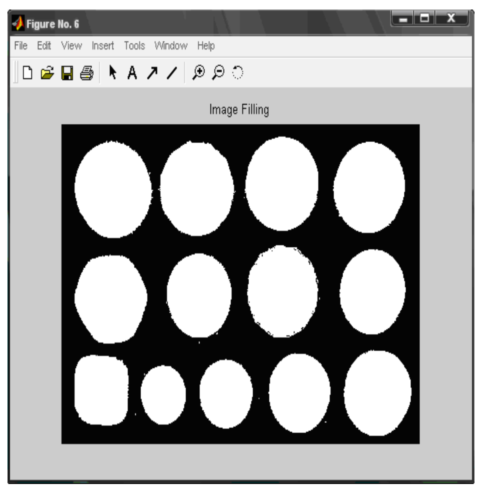



Fig.(5). Inverse Image.

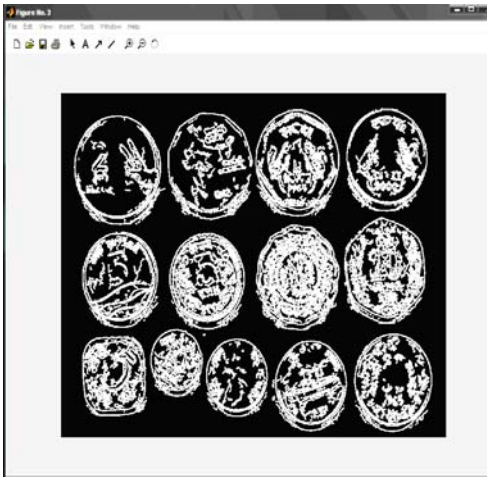


Fig.(12). Edge Detection by HT

Fig(13). Coin Classification Machine

**About the Authors**

**VIVEKANANDAN PERIYASAMY** received his Master of Science in Applied Mathematics from Madras University in 1978, and Doctor of Philosophy from Anna University in 1987. Also, he obtained his Master of Engineering in Computer Science and Engineering from Anna University in 1995. He is working as Professor of Mathematics, Department of Mathematics in Anna University from 1978. He visited Singapore, Malaysia, Bangaladesh, Sultanate of Oman, Germany and USA for presenting research papers and chairing sessions. He has published more than 70 research papers in national and international journals. His areas of research are Neural Network, Data Warehousing and Data Mining, Artificial Intelligence, Expert System, Digital Image Processing and Pattern Recognition, Internet Security and Software Reliability. He is one of the eminent professors in Anna University. His service to the student community earned him name and fame extending service to foreign countries. He was deputed for an assignment to Department of Information Technology, Higher College of Technology, Muscat, Sultanate of Oman from 2004 to 2008, where he served the students of international community. He is a prolific writer, thinker, philosopher and good orator. He frequently visits orphanages to help the poor and needy peoples. Currently, he is the Director, Data Research Centre, Anna University, Chennai, India.

**C.M.VELU,** received his M.Sc in Operations Research and Statistical Quality Control from Sri Venkateswara University, Tirupathi in 1985 and M.S in Computer Systems and Information from BITS, PILANI in 1994. He has visited UAE as a Computer faculty. He served as faculty of CSE for more than two decades. He has published five research papers in international journals. He presented five papers in national and international conferences. His area of interest is Data Warehousing and Data Mining, Artificial Intelligence, Artificial Neural Networks, Grid Computing, Parallel Processing. His area of research is Digital Image Processing and Pattern Recognition.