ISSN 1605-8607

COIN RECOGNITION BASED ON HEURISTIC SEGMENTATION AND IMAGE FUSION VIA BAYESIAN APPROACH

Akhtar H. Jalbani*, Pardeep Kumar**, M. Ibrahim Channa***, Chee Wai Mak****, Aness A.Soomro*****

Abstract

A simple algorithm that emulates the human ability and speed to learn and recognize coins and denominations of a foreign currency is presented. The use of color in hue, saturation and brightness indices to separate the coins from other objects / background and to determine the fusion of processed images results in a heuristic approach motivated by Bayesian learning framework. The complex process of coin detection and coin recognition reduces to a single threshold-limited image from which simple parameters are extracted and the value of the coins presented can be accurately determined. This method is applied to recognize Thai currency coins of 10-baht, 5-baht, 2-baht, 1-baht and 50-satang denominations and achieves 100% accuracy as compared to 44.29% using template matching in a trial of 14 test images acquired from a specific object distance.

Keywords: Bayesian approach, Bayesian learning framework, coin recognition, Hue-Saturation-Value, Histogram, Hough circle transform, image processing, pattern recognition, Thai coins, threshold.

1. INTRODUCTION

LECTRONIC payment methods may have increased in popularity in certain parts of the world in recent years, doing away with the need to carry "cash", but coins are still very much an integral part of life in many other parts of the world. In the metropolitan city of Bangkok, a foreign visitor often struggles with recognizing the value of the coins presented as some of the inscriptions on the coins are in Thai numerals (as shown in 0). While the conventional recognition by numeric is not possible, the human brain switches to visual differentiation methods. This subtle change often goes unnoticed but presents a powerful methodology that focuses on fundamental elements like color, shapes and size. The human brain quickly learns and remembers after a few occasions. Such methodologies are intriguing to study and emulate in machine learning.

Coin detection and coin recognition methods / systems are not new; many intelligent and complex systems have been researched and developed [1], using adaptive neural networks [3] and texton recognition methods [2]. The classification of coins based on parameters such as shape, size, surface design and weight was proposed in [1] for Indian coins. The use of color, particularly using special lighting matched to spatial properties of the camera was discussed in [10]. Yet another method that utilizes the HSV (hue, saturation and brightness) histograms to grade coins was presented in [11].

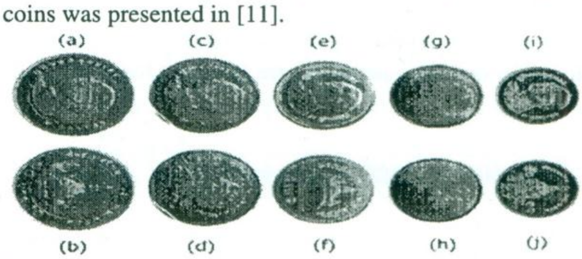


Figure 1: Thai coins with different denominations of interest; (a) head of 10-baht, (b) tail of 10-baht, (c) head of 5-baht, (d) tail of 5-baht, (e) head of 2-baht (type 1), (f) tail of 2-baht (type 1), (g) head of 1-baht, (h) tail of 1-baht, (i) head of 50-satang and (i) tail of 50-satang. Note: the 2-baht coin also comes in a silver colour version, while the 50-satang has a gold colour version.

*Associate Professor, Department of IT, Quaid-e-Awam University of Engineering, Science & Technology, Nawabshah, Sindh
,*** Assistant Professor, Department of CSE, Quaid-e-Awam University of Engineering, Science & Technology, Nawabshah, Sindh
***Assistant Professor, Department of IT, Quaid-e-Awam University of Engineering, Science & Technology, Nawabshah, Sindh
****Post graduate Student, Department of ISE, Asian Institute of Technology (AIT), Bangkok, Thailand

A multistage classifier based on eigenspaces capable of discriminating between hundreds of coin classes was presented in [7]. In the same, Bayesian fusion was also applied. A hybrid method of using eigenvalues of the covariance matrix, with circular hough transform and Bresenham's circle algorithm utilized with a raster scan algorithm that accurately pin-points position [12]. A gradient based method of recognizing coins using the discretized gradient direction went on to win the MUSCLE CIS Coin Competition 2006 [8]. Further methods for recognizing modern and Roman (ancient) coins were established in [9], where coin classification was accomplished with a simple k-Nearest Neighbor algorithm with k=5 after feature extraction. Many of these algorithms draw inspiration from more fundamental discussions on hough transform [5], histogram based methods [6], classification and segmentation techniques based on k-means [13 – 14]. A method of using R, G, B color together with area was proposed in [19] for single coin images.

With such a wealth of methods available, it is interesting to note that most of the methods focus on single coin, or extracted images of single coins at a time for detection and recognition. It is also noted that learning algorithms require hundreds of examples to allow the system to learn. This paper focuses on simplifying the learning process of differentiating the coins from the background (including other objects) so that processed images can be fused to form the optimum image where heuristic features can be extracted and evaluated to recognize coins accurately.

2. METHODOLOGY

A typical image used in this paper is shown in 0(a) below. Sample images do not necessarily have to contain all the denominations of interest. For ease of experimentation, the algorithms are implemented in OpenCV platform.

2.1 PRE-PROCESSING

The sample image of coins is first passed through a 7x7 Gaussian smoothing filter for noise reduction. The Gaussian smoothing operator is a 2-D convolution operator that is used to `blur' images and remove detail and noise. The weights are chosen according to the shape of a Gaussian function and the 2-D mask has the form:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{\frac{x^2+y^2}{2\sigma^2}} \qquad (1)$$

In parallel, the grey-scale version of the image is passed through a Canny operation.

The smoothed image is converted from RGB space into HSV space. The H (hue), S (saturation) and V (brightness) planes are then extracted separately from the converted HSV image. The results obtained are shown in 0.

(a) Heuristic Threshold for Coin Segmentation

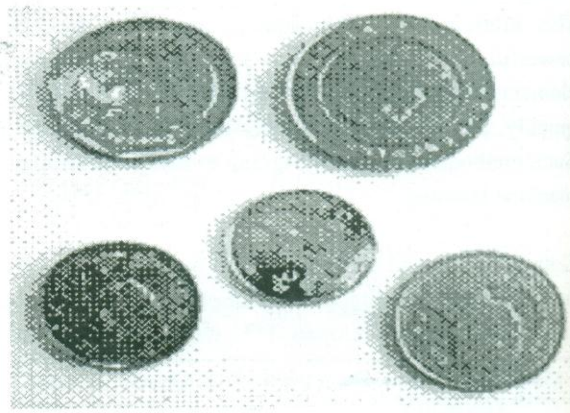
The primary task is to segment the image so that the coins become foreground and all other details become background. Motivated by the proposed method in [11], we turned to examining the histograms of the extracted H, S and V planes (0(d), (e) and (f) respectively). With an appropriate value selected (as marked approximately by the white arrow in 0), it is possible to set a threshold for each of the images to segment as many coins from the background as possible.

The apparent weakness of such a threshold scheme is over-reaction to "noise" in the histogram, which makes decisions difficult. The probability function of the coin in the H-plane is given by:

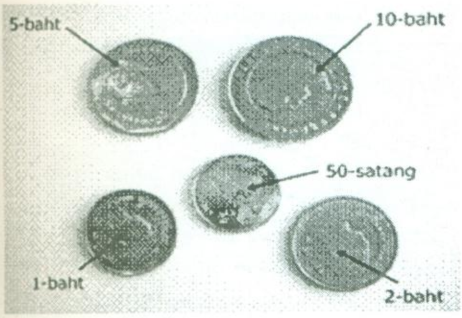
$$h(x;4,1) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-4)^2} \qquad (2)$$

The probability function of the background in the S-plane and V-planes are given by equations (3) and (4) respectively:

$$s(x;0,4) = \frac{1}{2\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x}{2}\right)^2} \qquad (3)$$



(a)



(b)

Figure 2: A typical image containing Thai coins of interest used in this paper; (a) shows the image of coins presented and (b) shows the image marked with denomination for reference.

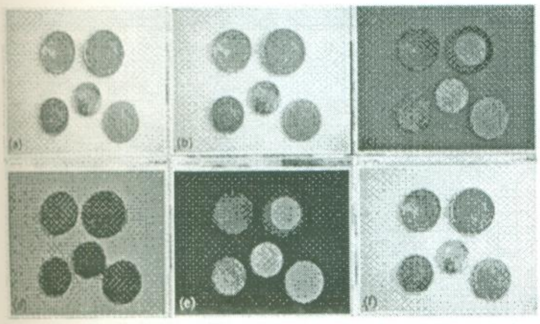


Figure 3: Pre-processing results obtained; (a) the original image as reference, (b) Gaussian smoothed image, (c) RGB image converted to HSV space and visualized, (d) extracted H (hue)-plane from HSV space, (e) extracted S (saturation)-plane from HSV space and (f) extracted V (brightness)-plane from HSV space.

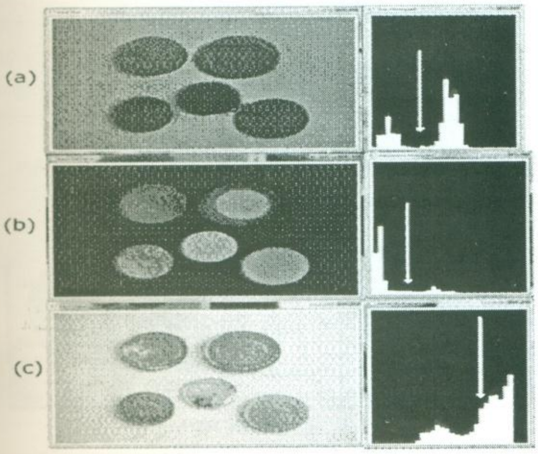


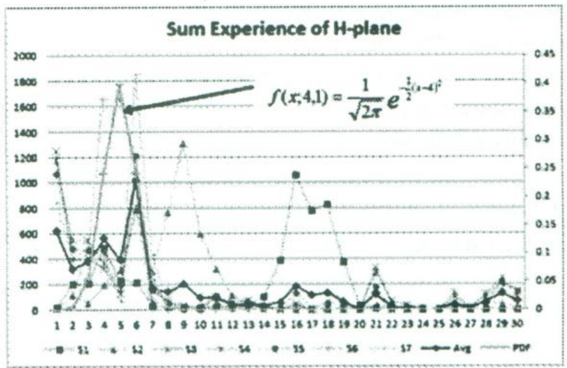
Figure 4: H-, S- and V-planes with respective histograms; (a) H-plane image with histogram, (b) S-plane image with histogram and (c) V-plane image with histogram. White arrow marks the approximate threshold position for each histogram to segment coins from background.

$$v(x;24,9)=\frac{1}{3\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x-24}{3}\right)^2} \qquad (4)$$

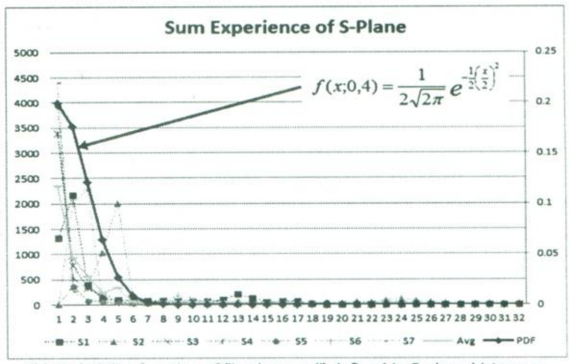
These probability functions can be derived from the heuristic behavior of histograms constructed from sample images in the H-, S- and V-planes, as shown in 0. Such histogram behavior is useful *prior knowledge* to the overall coin detection schema.

2.2 COIN DETECTION AND RECOGNITION

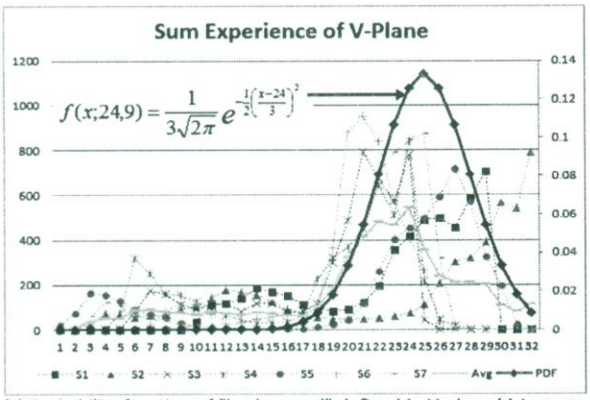
There are two important aspects of an object: shape and appearance [16]. We have defined Gaussian type probability density functions representing coins and background in H-, S- and V-planes in the prior section. It is important to recognize that a "coin" is a "round" object of certain "color" and "size", where "size" also corresponds to the "value". Where "round" is a property constituent of shape, "color" and "size" are property constituents of appearance.



(a) Probability function of "coin" defined in H-plane histogram



(b) Probability function of "background" defined in S-plane histogram



(c) Probability function of "background" defined in V-plane histogram

Figure 5: H-, S- and V-planes histograms characteristics; (a) probabilistic density function of "coin" casted in H-plane histogram, (b) probabilistic density function of "background" in S-plane histogram and (c) probabilistic density function of "background" in V-plane histogram.

(a) Shape

The simplest way to locate round shapes is to utilize the Hough Circle Transform [17]. The implementation via OpenCV first passes the image through a Canny edge detected. For every non-zero point, the local gradient is considered. The entire set of nonzero pixels in the edge image is considered for every candidate center. A circle with radius R and center (x_1, y_1) can be described with the parametric equations:

$$X = x_1 + R\cos\theta \quad (5)$$

$$Y = y_1 + R\sin\theta \quad (6)$$

When angle θ sweeps through 360°, the set of points (X,Y) traces the perimeter of the circle described.

The probability density function is given by:

$$P(\text{object is circle}) = \begin{cases} 1; \text{ iff object} \in \text{hough circle} \\ 0; \text{ otherwise} \end{cases} \quad (7)$$

The identification of circular objects via hough circle transform function is shown in 0.

(b) Size

Size can be determined either by the length of the perimeter of the circle traced by the hough circle transform function or determined by the area encircled by the perimeter of the object found. In this paper, the blob area function in OpenCV was used to determine the area of the object.

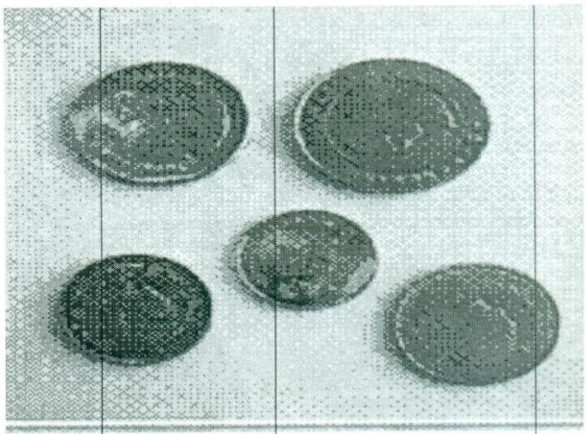
Intuitively, the "size" of the coin corresponds to the denomination or "value" of the coin. The distribution of size for different denominations as measured by the blob area function is examined, as shown in 0. A Gaussian type probability density function can be setup for each of the denominations of the coins of interest as the variation arises from the measurement itself with an error estimate that is Gaussian. The sizes of the coins are fixed; table 1 shows the relationship of sizes between denominations.

Table 1: Size relationship between Coin Denominations

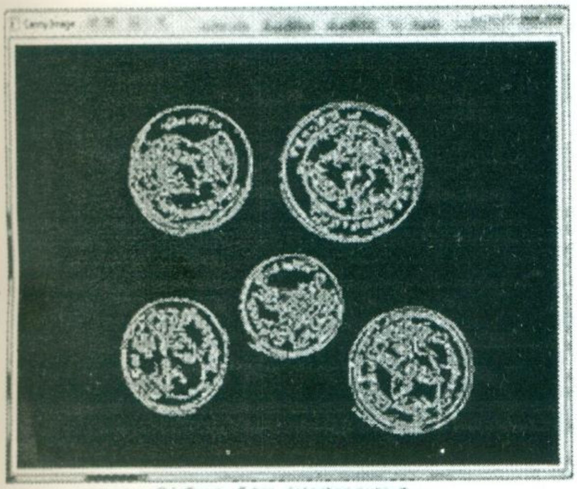
	10-baht	5-baht	2-baht	1-baht	50-satang
10 baht	1.00	1.22	1.40	1.70	2.08
5 baht	0.82	1.00	1.15	1.39	1.70
2 baht	0.71	0.87	1.00	1.21	1.48
1 baht	0.59	0.72	0.83	1.00	1.22
50 satang	0.48	0.59	0.68	0.82	1.00

(c) Coin Detection

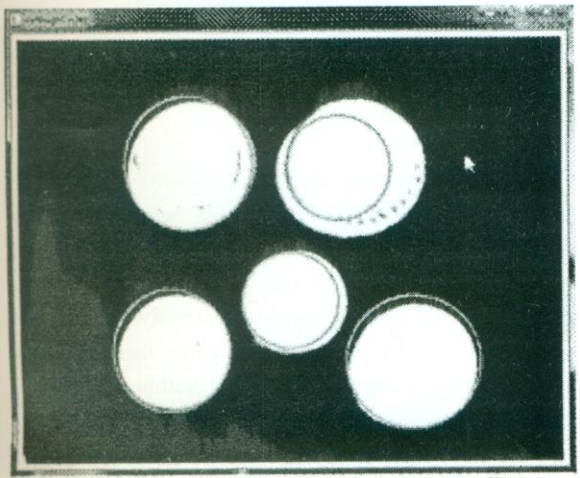
The motivation to construct probability density function for property descriptors of the coin (that is, a "round" object of a certain "color" and "size", where "size" corresponds to the denomination value) in shape and size, we explore the use of Bayesian methods. Bayesian methods allow us to use prior knowledge about the objects [15][16].



(a) Grey-Scale Image.

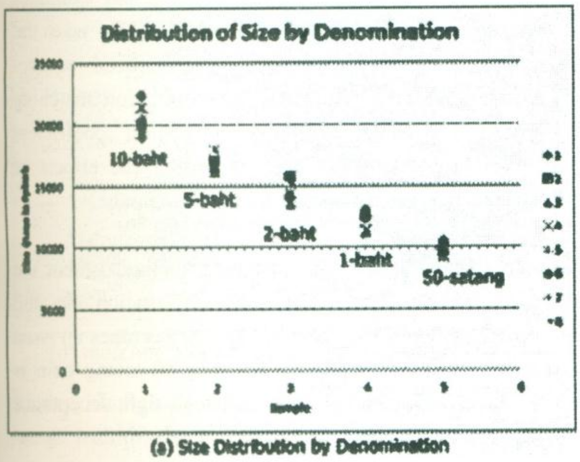


(b) Canny Edge detector output.

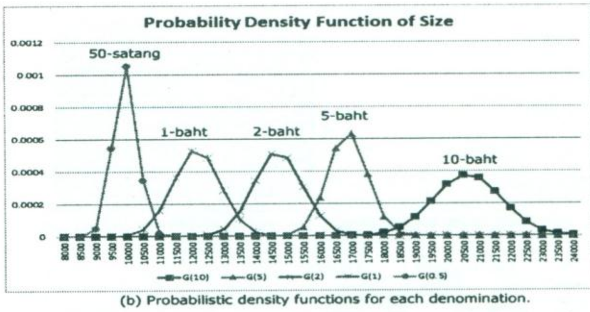


(c) Hough Circle Transform output; hough circles identified.

Figure 6: Identification of circular objects; (a) grey-scale version of original image, (b) Canny edge detector output and (c) identified circles from hough circle transform function.



(a) Size Distribution by Denomination



(b) Probabilistic density functions for each denomination.

Figure 7: Size distribution of different denominations studied in this paper; (a) distribution by sample and (b) probability function constructed from sample distribution.

The combination of the probabilistic evaluation of each of the object properties yields the final decision for the coin recognition. Implemented in a decision tree format, each of the probabilistic outcomes is accumulated sequentially in a manner well practiced by humans. For example, a "10-baht" coin is a "round object" of "a certain color that is different from the background" and "size that is between 19000 and 22000 pixels". Mathematically, it is given by:

$$P(10\text{-}baht) = P(object\ is\ circle) \bullet h(x) \bullet [1 - s(x)] \bullet [1 - v(x)] \bullet size(x) \qquad (8)$$

Using this Bayesian approach, the concept of "a certain color that is different color that is different from the background" is realized by the fusion of resultant binary images from apply appropriate thresholds previous derived from the probability density functions in the H-, S- and V-planes discussed previously.

3. RESULTS & DISCUSSIONS

The proposed method is applied to several images containing coin and non-coin objects and also of different backgrounds. In non-coin objects, we applied the method to round and irregular shaped objects. To test the robustness of the proposed method, we used round, non-coin objects of similar size as the coins.

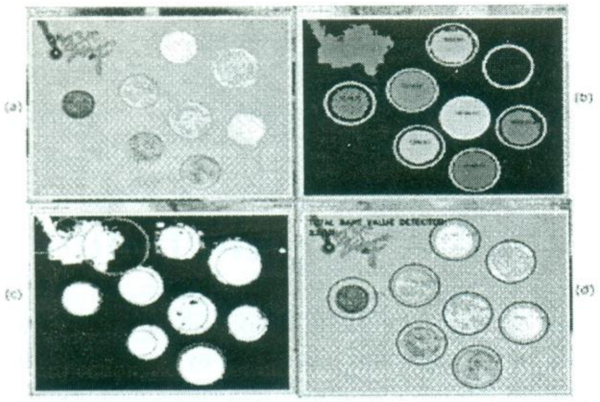


Figure 8: Image containing coin and non-coin objects; (a) original image, (b) blob area function, (c) hough circle transform and (d) coin identification and recognition mapped to original.

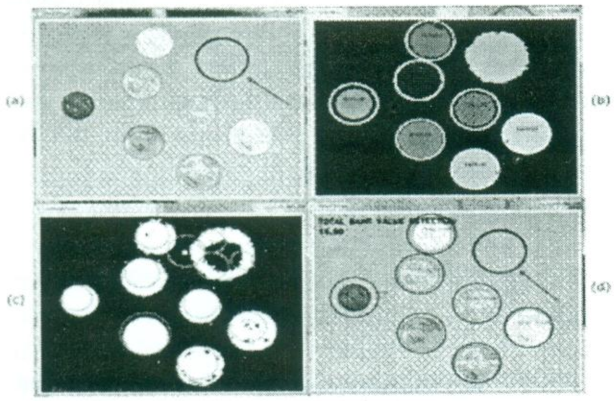


Figure 9: Image containing coin and round, non-coin object; (a) original image, (b) blob area function, (c) hough circle transform and (d) coin identification and recognition mapped to original.

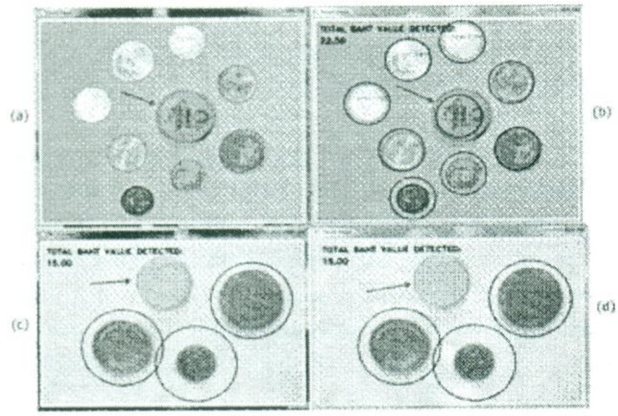


Figure 10: Images containing coin and round, non-coin object; (a) original image 1, (b) identification and recognition result of image 1, (c) original image 2 and (d) identification and recognition result of image 2.

3.1 COIN VS NON-COIN OBJECTS

As shown in 0(a)-(d), the proposed method is able to identify the coins of interest and also recognize the value correctly. The proposed method left out the irregular object placed in the image.

Another test image consisted of a round, non-coin object of similar size to the coins. Results are shown in 0; the round "rubber-band" was not selected by the algorithm. The values were correctly identified. Similarly, the same result is achieved with two other test images consisting of round, non-coin objects, as shown in Figure 10 In particular, the image in 0(c)-(d) contained a coin (of 25-satang denomination) that is not in the scope of this paper and the proposed method was able to discriminate this coin.

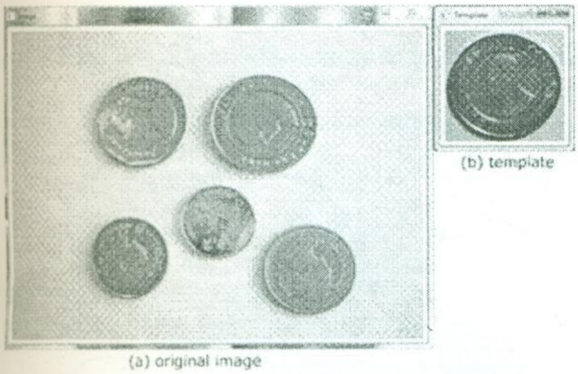
3.2 BACKGROUNDS

The proposed method was also applied to different backgrounds. Firstly a stone-table was chosen and secondly, the coins were placed on the palm of one's hand, as shown in 0. In both cases, the proposed method identified and recognized the coins correctly. Incidentally. Figure 11 (a) - (b) contained 3 coins there are of a denomination (25-satang) that is not within the scope of this paper and has been successfully discriminated.

3.3 COMPARISON WITH TEMPLATE MATCHING

The proposed method is compared with a template matching technique discussed in Chapter 7 of [18]. Template matching in OpenCV takes a template image and slides it over the image of interest and evaluates "correlation" with a specified method. We take the normalized versions, namely (i) normalized square-difference method, (ii) normalized correlation matching method and (iii) normalized correlation coefficient matching method as they help to reduce the effects of lighting differences between template and image.

The advantage of template matching method is that the matching techniques are invariant to rotation (0) and features (0). The invariance to features becomes an issue on accuracy, as shown in 0, where a 50-satang coin is found as a match to a 1-baht template; a tight acceptance criteria needs to be set.



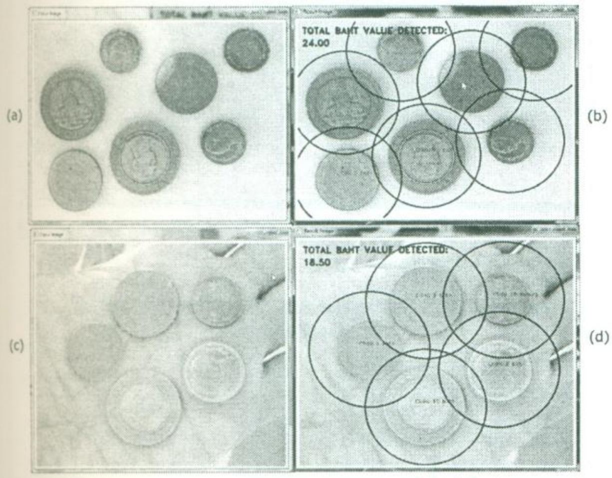


Figure 11: Images of coins on a stone-table background; (a) original image of coins on stone-bench, (b) identification and recognition result of image (a), (c) original image of coins on palm of hand and (d) identification and recognition result of image (c).

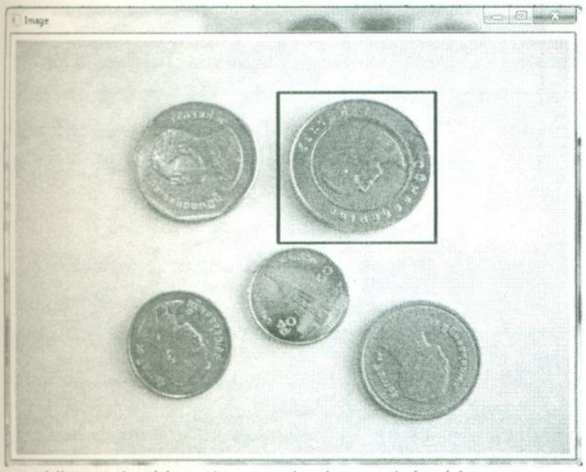


Figure 12: Results using template matching; (a) original image with (b) the template, (c) showing visualization of the matching results where the arrows mark the location of "best match" and (d) shows the "best match" marked on the original image.

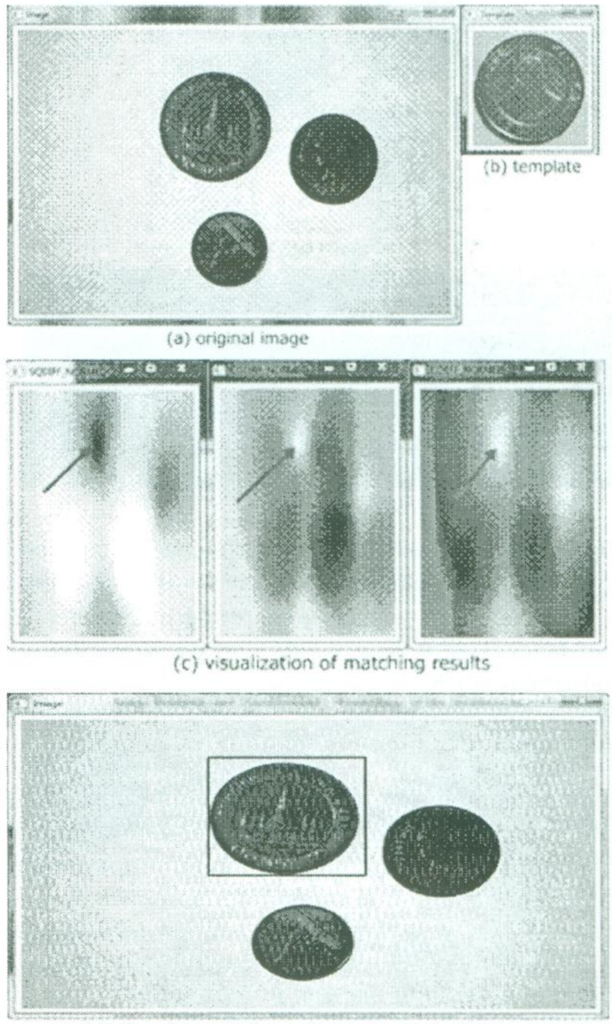
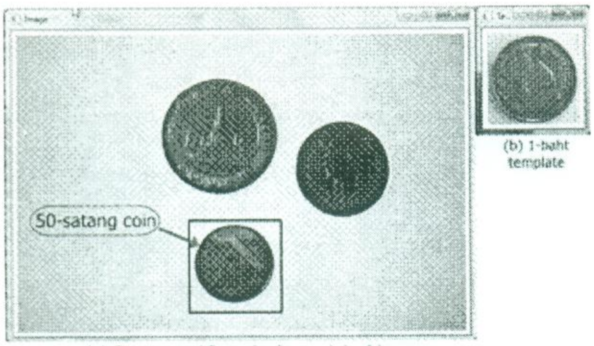


Figure 13: Results using template matching where template has different features than target in image; (a) original image with (b) the template, (c) showing visualization of the matching results where the arrows mark the location of "best match" and (d) shows the "best match" marked on the original image.

In another test image comparison, template matching method was unable to find a match due to size and/or color (0(a)); the best match was rejected as the values were outside the acceptance range. With a re-size template, the best match values showed an acceptance (99.99% confidence) but the match found the incorrect coin (0 (c)-(d)).

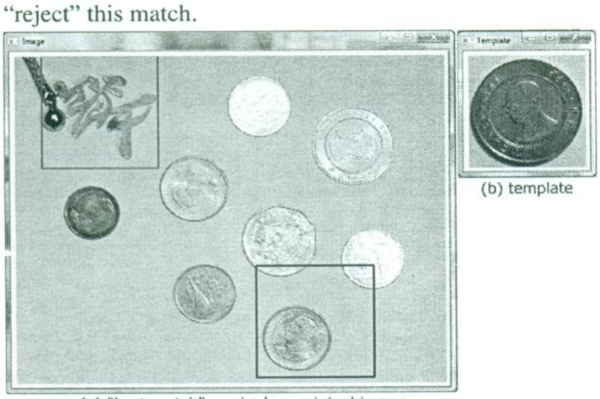


(b) 1-baht template

(a) "best match" marked on original image

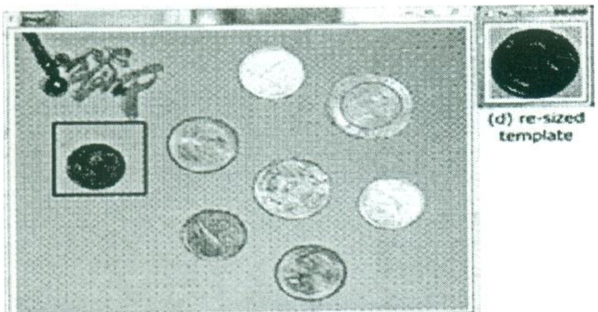
(c) acceptance criteria on min/max values need to be set

Figure 14: Incorrectly recognized coin; (a) a 50-satang coin in an image being found as a match to (b) a 1-baht template, therefore giving false results. The match criteria (c) needs to have a tight acceptance limit set to "reject" this match.



(b) template

(a) "best match" marked on original image



(d) re-sized template

(c) match (99.99% confidence) found but incorrect coin

Figure 15: Incorrectly recognized coin; (a) matches found that were rejected based on acceptance criteria with the template shown in (b). (c) shows the incorrect match still with (d) resized template (match accepted at 99.99% confidence).

Figure 16: Coins correctly identified and recognized using our proposed method.

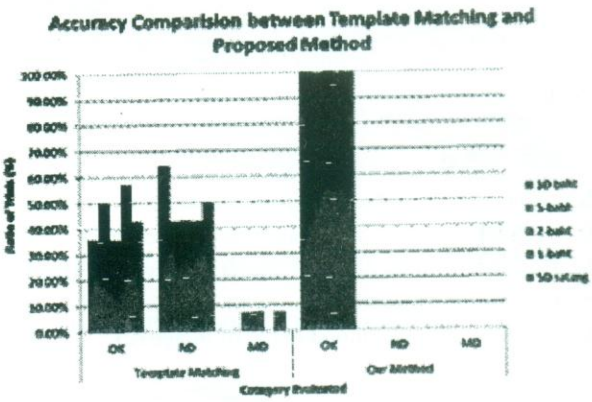


Figure 17: Graph showing comparison between template matching and proposed method. "OK" means correctly recognized, "ND" means not detected/recognized and "MD" means mis-recognized (recognized wrong denomination)

Whereas, our proposed method correctly detected and recognized all the coins in the same test image (0), showing superior performance over template matching method.

A further trial using 14 test images to compare between template matching and our proposed method is conducted. The brief algorithm for each method is summarized in table 2. A summary of accuracy comparison between template matching and our proposed method conducted using the same test images is shown in 0. In the comparison trial, template matching method achieved only an overall accuracy of 44.29% (31/70 coin-presences detected and recognized correctly) while our method achieved 100% accuracy (70/70 coin-presences detected and recognized correctly). The trial confirms the superior performance of our method over template matching.

Table 2: Summary of Algorithm

Process	Template Matching	Proposed Method
Pre-Processing	1. Gaussian Smoothing (noise reduction)	1. Gaussian Smoothing (noise reduction) 2. Convert RGB image to HSV image 3. Split HSV image into 3 separate images; H-, S- and V-planes 4. H-, S- and V-planes threshold at limits based on Gaussian probability density function on histograms. 5. Fusion of H-, S- and V-images based on behavior of PDF determined in step (4). Result is a binary image.
Coin Detection & Recognition	1. Load 10-baht template. 2. Apply template matching using normalized square-difference, normalized correlation and normalized coefficient matching algorithms. 3. From each algorithm, determine location of "best match" and compare "confidence" with limit of 95%. If match found, go to step 4. If nothing accepted, load next and repeat. If nothing accepted after all templates used, skip to step 5. 4. Remove matched ROI from original image and repeat from Step 1. Accumulate count for coin detected and recognized. 5. Repeat for 5-baht, 2-baht, 1-baht and 50-satang template sets.	1. Subject fused image to blob detection and blob area function. 2. Subject fused image to hough circle transform. 3. For each blob detected, cross check with hough circle list. If matched, assign coin value based on area and go to step 4. If not matched go to step 5. 4. Accumulate count for coin identified and recognized. 5. Repeat step 3-4 for all blobs detected in image.

4. Conclusions

In this paper, a simple approach of presenting a fused image (consisting of binary H, S, and V representation of coins in the presence of background and other objects) to a Hough Circle Transform and Blob Detection algorithm for coin recognition is presented. The decision of threshold values employed and the choice of images selected for fusion with a Bayesian approach applied on heuristic trends exhibited by coins in the presence of other objects and background. This method achieved superior performance with an accuracy of 100% compared to 44.29% using template matching in a trial conducted with 14 test images.

Future work will focus on the ability to differentiate coin denominations from currencies mixed into the same image.

Acknowledgment

We wish to thank Western Digital for supporting the Master scholarship and AIT-School of Engineering and

Technology for providing the learning opportunity on this project.

REFERENCES

[1] M. Velu, P. Vivekanandan, "India Coin Recognition System of Image Segmentation by Heuristic Approach and Hough Transform (HT)", Int. J. Open Problems Compt. Math., Vol. 2, Issue. 2, June 2009.

[2] Dong-Shin Kim, Young Gyun Yun, You-Kyung Cha, "Coin Detector", CS7495/4495 Term Project, SSIP6, 14th Summer School on Image Processing, Szeged, Hungary 2006, [ONLINE] available at URL:www.inf.u-szeged.hu/ssip/2006/ projects/team6/ coin.pdf

[3] Shatrughan Modi, Seema Bawa, "Automated Coin Recognition System using ANN", International Journal of Computer Application (0975-8887), vol. 26, Issue 4,pp. 0975-8887, July 2011.

[4] M. Nölle, H. Penz, M. Rubik, K. Mayer, I. Holländer and R. Granec, "Dagobert – A new Coin Recognition and Sorting System", Proc. VIIth Digital Image Computing: Techniques and Applications, Sun C., Talbot H., Ourselin S. and Adriaansen T. (Eds.), 10-12 Dec. 2003, Sydney.

[5] H. Ballard, "Generalizing the hough transform to detect arbitrary shapes", Pattern Recognition, 31-2, 1981.

[6] S. Siggelkow and H. Burkhardt, "Improvement of Histogram-Based Image Retrieval and Classification", Proceedings of the International Conference on Pattern Recognition 3, 367-370, 2002.

[7] R. Haber, H. Rosmoser, K. Mayer, H. Penz, M. Rubik, "Classification of Coins using an eigenspace approach", Pattern Recognition Letters, Vol. 26, Issue 6, pp. 61-75, 2005.

[8] Marco Reisert, Olaf Ronneberger, and Hans Burkhardt, "An Efficient Gradient Based Registration Technique for Coin Recognition", Proceedings of the MUSCLE CIS Coin Competition Workshop, Berlin, Germany, 11 September 2006, [ONLINE] available at URL: http://muscle.prip.tuwien.ac.at/coin_workshop2006_proceedings/reisert.pdf.

[9] Martin Kampel, Maia Zaharieva, "Optical Recognition of Modern and Roman Coins", Layers of Perception, Proceedings of the 35th International Conference on Computer Applications and Quantitative Methods in Archeology (CAA), Berlin, Germany, 2007.

[10] M. Ademeck, M. Hossfeld, M. Eich, "Three colour selective stereo gradient method for fast topography recognition of metallic surfaces", Proceedings of Electronic Imaging, Science and Technology (Martin A. Hunt and Jeffrey R. Price .eds), Machine Vision Applications in Industrial Inspection XI, vol. SPIE 5011, pp. 128-139, January 2003.

[11] Rick Bassett, Ping Gallivan, Xiang Guo, Eric Heinen, Akarsh Sakalaspur, "Development of an Automated Coin Grader: A Progress Report", Proceedings of MASPLAS'02, the Mid-Atlantic Student Workshop on Programming Languages and Systems, April 2002.

[12] J. Prakash, K. Rajesh, "A Novel Approach for Coin Identification using Eigenvalues of Covariance Matrix, Hough Transform and Raster Scan Algorithm", World Academy of Science, Engineering and Technology 44, 2008.

[13] R. M. Haralick and L. G. Shapiro, "Survey – Image segmentation techniques", Computer Vision Graphics and Image Processing vol. 29, pp. 100-132, 1985.

[14] M.K. Ng, "A note on K-means Algorithm", Pattern Recognition, vol. 33, pp. 515-519, 2000.

[15] Moshe Blank, Ita Lifshitz, "A Bayesian Approach to Recognition", Advanced Topics in Computer Vision, Faculty of Mathematics and Computer Science, [ONLINE] available at URL: http://www.wisdom.weizmann.ac.il/~vision/course

s/2004_2/files/bayessian_recognition.pps, April 2004.

[16] Fei Fei Li, Robert Fergus, Pietro Perona, "A Bayesian Approach to Unsupervised One-Shot Learning of Object Categories", Proceedings, 9th IEEE International Conference on Computer Vision (ICCV 2003), France, 2003.

[17] Kimme, D. H. Ballard, J. Sklansky, "Finding circles by an array of accumulators", Communications of the Association for Computing Machinery 18, pp. 120-122, 1975.

[18] Gary Bradski, Adrian Kaehler, "Learning OpenCV", O'Reilly Media, Inc., 2008.

[19] H. R. Al-Zoubi, "Efficient coin recognition using a statistical approach", 2010 IEEE Conference on Electro/Information Technology (EIT), pp. 1-5, 2010.