MATEMATIKA, 2009, Volume 25, Number 2, 147–156

Visual-Based Automatic Coin-Counting System Using Neural Network

[1]**Mohd. Syafarudy Abu & Lim Eng Aik**
Institut Matematik Kejuruteraan, Universiti Malaysia Perlis
02000 Kuala Perlis, Perlis, Malaysia
e-mail: [1]syafarudy@unimap.edu.my, [2]ealim@unimap.edu.my

Abstract A new intelligent coin-counting system is described in this paper. The proposed system is effective and flexible for the purpose of performing coin-counting using image captured from webcam. Image processing techniques are employed to prepare data for image understanding, and a Radial Basis Function (RBF) network is employed for performing the classification task. Extensive and promising results were obtained and the analysis suggests the proposed Radial Basis Function type classifier provides good results for high accuracy in coin-counting.

Keywords Coin-Counting; feature extraction; median filter; edge detection; image segmentation.

1 Introduction

Coin counting has been an issue for bank and store. Long ago, before the arrival of coin counting machine, man has to count the coins manually, and it is time consuming and yet tedious for those who handle the counting work. Mistakes on counting happen most of time due to many reasons: eyes tiredness, losing focus, too many tiny coins confuse the eyes and etc. Alternative coin-counting method can appear to be essential because an accurate coin-counting is able to provide a quantitative output and time saving. Recently, image processing is applied to estimation density of microbe in image based on black pixel density estimation in [1], [2]. Additionally, [3], [5], [6] are vision based techniques to classify object in images. Recently, an algorithm to count characters in an image [4] was reported. While [7], shows a way to identify shape in binary image and grey level images are reported. More recently, a neural network based vision system for estimating the crowd level of an underground station was reported in [8, 9, 10, 11].

In this paper, we describe an unconstrained visual-based system designed to perform accurate coin-counting. Obviously, there are few issues to be considered. The main issues discussed in this paper are the way to extract the essential features from unconstrained images and perform accurate coin-counting using RBF neural network.

In general, visual-based system includes two stages: image processing and image understanding. The first stage is to localize the objects from the background. This followed by feature extraction techniques are applied for image understanding. In [12, 13], radius seeking algorithm has been used to count and classify coins. Coins in images are then located and classified based on radius, and the total values of coins found in image are then obtained. But, this technique shows poor performance when occlusion of coins or many coin touching each other are found in images. Obviously, there are few problems in these solutions. When the system is directly referred to image information, coin-counting systems produce high input dimensions. Generally, this increases the RBF network training time and reduces its performance. The other problem is the occlusion of coins. That is, the overall result depends greatly on the performance of the image processing technique.

In this paper, we employ the approach of image segmentation and feature extraction. There the extracted information to be in the form of reduced data sequences. With these sequences, the images are being dealt with in spatial order. The RBF neural network type classifier is employed to assure a robust image understanding, due to its outstanding classification ability [14, 15, 16].

This paper is organized as follows: Section 2 briefly explains this coin- counting system; Section 3 deals with the image processing stage, where feature extraction is discussed; the Section 4 overviews the RBF network which is the basis of image understanding component; Section 5 presents the performances of our system in details; Section 6 is the conclusion.

2 System Outline

The system for coin-counting includes two stages: image processing, and image understanding. Data obtained through image processing produced a data curves. A series of data sub-curves correspond to an image. A data sub-curve describes an interested region in the image. In the stage of image understanding, our system is based on the RBF network. RBF networks had been used previously for sequence recognition [17, 18, 19]. In order to process the data curves, we extend this idea to our RBF network. Figure 1 shows the RBF network recognizes the data sub-curve. For every data sub-curve, a data window with the given size shifts from the left to the right step by step. The segment of the given sub-curve in the "window" is sampled into a single vector as the RBF network input. In the RBF network, the kernel nodes of hidden layer are provided with a series of overlapping segments of the data sub-curves in spatial order. Window by window, the output layer performs classification and memorized all these results. At the end of this data sub-curve operation, the output layer, the partial classification is performed and the final classification result is obtained for the given data sub-curve.

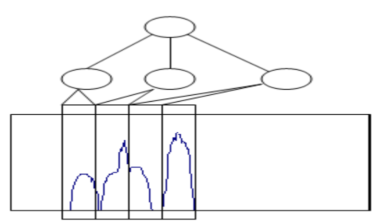


Figure 1: How RBF Network Recognizes a Curve

In the whole system, there are the three crucial points: feature extraction of image; the RBF network data window; the structure and parameters of the RBF network. The parameters *w_width* and *w_height* for RBF network data window depend strongly on the nature of data curves. *w_width* and *w_height* are chosen as 80 pixels and 60 pixels respectively in our experiments. In order to reduce the input dimension and avoid noises of data curves, a data window of the RBF network is partitioned into 9 frames evenly. The average in one data frame is calculated as one element of an input vector. In this way, the input dimension of RBF network is reduced to 9.

3 Image Processing

In coin-counting system, the coin values in the image are needed to be counted. Based on the lower level feature of the original images, the operation of image processing prepares data curves for image understanding. The data curves have lower dimension and are suitable for our classification task.

3.1 Image Segmentation

Segmentation is the partitioning of an image into parts that are coherent according to some criterion [20]. The aim of segmentation in this system is to differentiate the region of interest from other region of the image.

In our task, coins are region of interest. After segmentation, coins in images are detected. The detection is based on the change of detection with reference to the background. Here, the changes mean the difference of the grey value between image and the reference image. If the changes at a pixel in the image are bigger than a certain threshold value, the pixel at the corresponding position in the segmented image is in black; otherwise, this pixel is in white. After segmentation, we get the binary masked image corresponding to every original image via edge detector.

In order to filter out more real image noises, including the slight variation of illumination, such as camera noise, it is a need to discriminate the objects from non-objects using median filter. If detection of changes is based on overall changes of the pixels inside the sub-block rather than an individual pixel, the limited changes coming from image noises above are averaged and can be discarded easily. Hence, in the image segmentation, pixel is replaced by sub-block. There is a need to choose the size of the sub-block to guarantee that low resolution image work well in our task. The original image resolution is 300x250 in pixels. We choose the sub-block of 5x5 in pixels. The resolution of the segmented image is 60x50.

The average grey value of pixels in a sub-block of the original image is compared with the corresponding sub-block of the reference image. Figure 2 (a and b) showed the example of the image and the segmented image corresponding to it.

3.1.1 Edge Detector

To solve the problem in detecting coins edge, we used Sobel filter [21]. By applying Sobel filter on the image, the edges of the object are obtained. Then we analyzed the edge points in certain area, to estimate whether the area is belonged to the background or not. To build a Sobel filter, two 3x3 convolution kernels are built:

Figure 3(a) and 3(b) are the 3x3 convolution kernels that uses to calculate $O_1(i,j)$and $O_2(i,j)$ respectively. Figure 3(c) shows the coordinate system of the kernel. The formula of convolution is as below:

$$O(i,j) = \sum_{k=1}^{3}\sum_{l=1}^{3} I(i+k-2, j+l-2)K(k,l) \tag{1}$$

where $I(i,j)$ is the pixel value of the pixel (i,j) of the input image. $K(k,l)$ is the value of the convolution kernel. $O(i,j)$ is the calculated pixel value of the pixel (i,j) of the processing image.

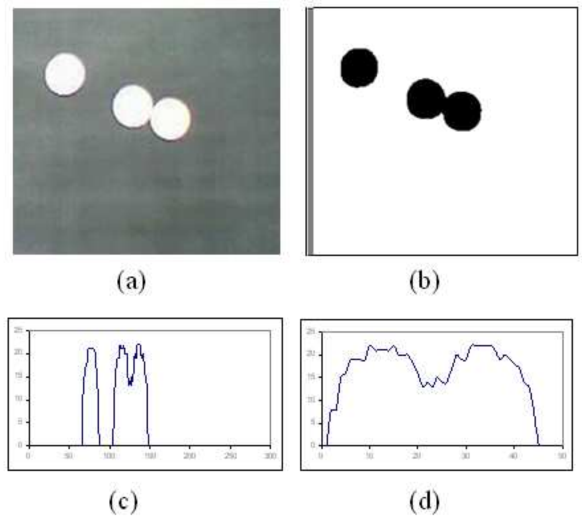


Figure 2: The Whole Procedure of Image Processing

The two kernels are applied to the pixels of the input image. Then $O_1(i,j)$ and $O_2(i,j)$ of each pixels of the image can be found. Then the gradient magnitude at each pixel $G(i,j)$ are calculated by using (2).

$$G(i,j) = \sqrt{O_1^2(i,j) + O_2^2(i,j)} \tag{2}$$

If $G(i,j)$ is greater than the threshold value τ, then the pixel (i,j) is considered as edge and is represented as white pixel. Figure 4(a) and 4(b) shows the result of median filtering and Sobel filter respectively. From the result obtained from Sobel filter image, we convert that particular image to binary thresholded image as shown in Figure 4(c).

3.1.2 *Median Filter*

In order to eliminate the noises and preserve the edge points in the output image when the image is processed by Sobel filter, some pre-processing works have to be done on the input image. Preprocessing work involve smoothing the input image to remove noises by median filter. When clearly notice in Figure 5(b), it can be seen that the image has been smoothed from the actual image (Figure 5(a)). Although the image is a little blurred, it has no harm effect for further processing. The algorithm of median filter in the paper is that it sorts the pixels in a 5x5 sub-block in ascending order according to the pixels value in the 5x5 sub-block and replaces the center pixel of the sub-block with the pixel which has median pixel value.

3.2 Feature Extraction

The data contained in images by nature has very high dimension, even the resolution of the segmented image have been reduced using the above method. Feature extraction can be seen as a special kind of data reduction to find a subset of informative variables based on image data.

-1	-2	-1
0	0	0
1	2	1

(a)

-1	0	1
-2	0	2
-1	0	1

(b)

K_{11}	K_{12}	K_{13}
K_{21}	K_{22}	K_{23}
K_{31}	K_{32}	K_{33}

(c)

Figure 3: Sobel Convolution Kernel

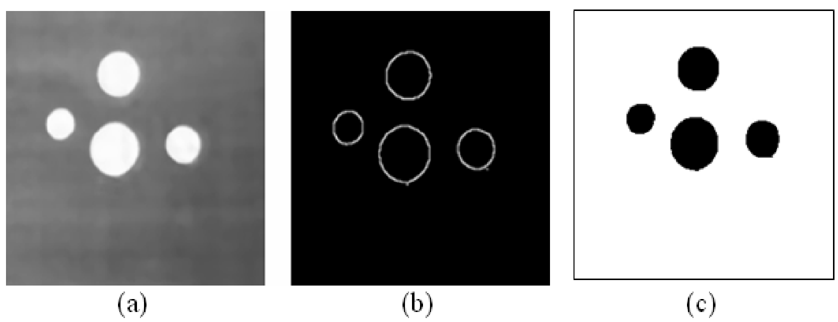
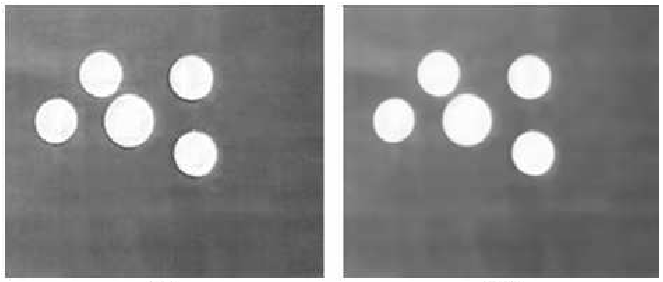


Figure 4: Result for Median Filter Image, Sobel Filter Image and Binary Threshold Image

After image segmentation, the area that we are interested has been transformed into black color and other parts into white color. The percentage of black pixel in a standard size template is recorded. Step by step, the standard template scanning the whole image from the left to the right. The percentage values of black pixel in the whole template are recorded at every position of the template. The curves of percentage of black pixel in a template are obtained from every segmented image. A template is shown in Figure 2(b), the rectangle in grey at the most left hand side. The step length of a shift is one pixel. Using this method of feature extraction, an image is transformed into a data curve. Figure 2(c) gives the data curve corresponding to the segmented image shown in Figure 2(b).

In the images, we are only interested in regions with coins. Two thresholds are set to filter out the data that have no useful information. One threshold ϕ is for the percentage values. Based on this threshold, one curve is partitioned into several sub-curves. The other threshold ψ is for the minimum width of sub-curves. Figure 6 shows the pseudo-code of the sub-curve extracting algorithm.

Figure 2(c and d) give an example of a curve and its corresponding sub-curves prepared for the RBF network. In fact, the images in Figure 2(a, b, c, and d) show an example of



(a) (b)

Figure 5: Median Filter Image of the Coin

the whole image processing from image segmentation to feature extraction. After image processing, the image is transformed to a series of data sub-curves. Sampling these data sub-curves in the way mentioned in section 2, we obtain the input data vectors for the RBF networks.

```
Begin
I = 1; percent = φ; length = ψ; L = null;
While (not sequence end)
        If G(i) > φ
                L = L + G(i);
        Else
            If L > ψ
                Save L for RBF networks
        end
        L = null;
        End if
End While
End
```

Figure 6: Pseudo-code of Sub-curve Extracting Algorithm

4 Radial Basis Function Network and its Training

Radial basis function (RBF) neural networks transform the n-dimensional inputs non-linearly to an m-dimensional space and then estimate a model using linear regression. In general, RBF neural networks have three layers, the input layer, hidden layer and output layer. Based on [22], when the number of basis functions (hidden units) are significantly larger than the number of input dimensions (input units), the role of the hidden layer is a pseudo-orthogonalization of the sample data through a mapping from the input space onto a higher dimensional space . The non-linear mapping is controlled by a set of m basis functions. Each function is decided by a centre vector c_i $(1 \leq i \leq m)$ in the input space and a width or radius vector $\sigma_i(1 \leq i \leq m)$. The active function of hidden neurons is typically chosen as Gaussian-like functions, as in (3).

$$f_i(x) = \exp\left(-\sum_{k=1}^{n} \frac{(x_k - c_{ik})^2}{2\sigma_{ik}^2}\right) \tag{3}$$

Each output unit can be computed by (4).

$$y_j(x) = \sum_{i=1}^{m} w_{ji} f_i(x) + b_j,\ 1 \leq j \leq l \tag{4}$$

The learning methods of RBF networks had been summarized in [23]. In our solution, SOM [24] is employed to calculate the centers of RBF networks c_i $(1 \leq i \leq m)$. The width vectors $\sigma_i(1 \leq i \leq m)$ are determined by the formula (5),

$$\sigma = r \cdot I \tag{5}$$

where r = 0.5 in our experiments.

The values in width matrix σcontrol the shape of the basis functions. Large or small values are making the neuronal response too flat or too peak respectively. We avoid two extreme situations. Hence, this simpler method is selected, and in fact, the experiments below have proved its effectiveness.

5 Experiments and Results

The digital images in our task come from a series of images capture by webcam on a background that we have setup for this experiment. All the coins used in these images are Malaysian coins, which define in "sen" in currency. There are four types of coin involved, the 5 sen coin, 10 sen coin, 20 sen coin, and 50 sen coin. We select 250 images. In these images, there are coins show up in various combinations; lying flat and isolated, lying flat and touching, occlusion of coins, etc. For training the RBF networks, we partition the whole data set into three parts: 120 images for training, 30 images for validating, and 100 images for testing. The images for testing are novel one for the classifier.

The ten top networks from 50 networks that have been evaluated on their performance on validation patterns are selected. The average classification accuracy for these selected ten classifiers are0.920 ± 0.028, and the best classification accuracy among these ten classifiers are 0.943. Figure 7 gives some images for testing and the results of our system. These show that the RBF-based classifiers can overcome the "noises" due to the various reasons, including image noise not filtered out and occlusion of coin.

6 Conclusion

In this paper, we introduce a new intelligent coin-counting system, which is based on segmented image. There are two important issues in image processing, namely, feature extraction and method of understanding the image. These points are our main focus.

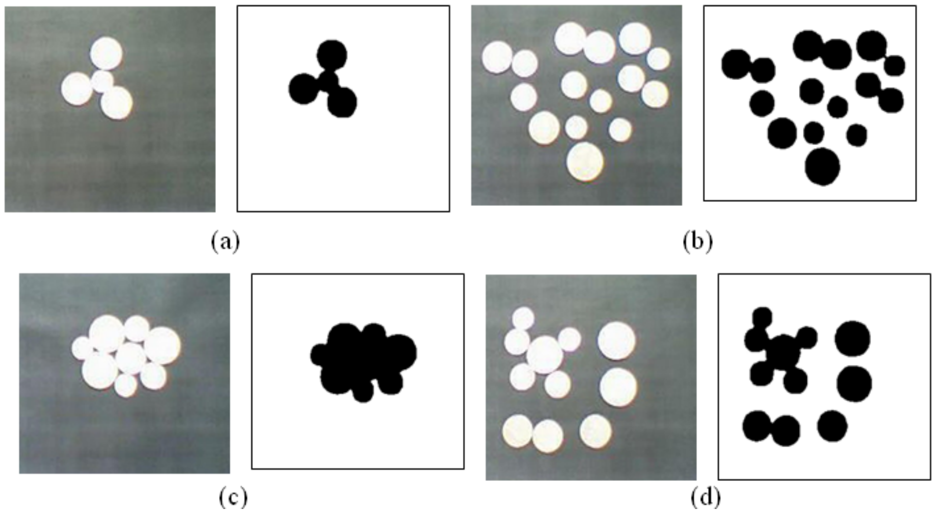


Figure 7: Some Examples of Our Result; (a) Result = 65sen, Coin Touching Each Other; (b) Result = 185sen, Coin Touching Each Other and Isolated; (c) Result = 200sen, Major Coin Touching; (d) Result = 250sen

In section 3, we describe the feature extraction method. Considering our goal of coin-counting in images, we reduce the image information effectively. The input dimension for our classifier is reduced greatly.

As discussed in section 4, the RBF network, an adaptive classifier which delivers a fast training and good performance in classification, is chosen. The RBF network used in our system is extension of a technique successfully used for understanding sequences [17, 18, 19]. Performances comparison of non-RBF classifier and the RBF classifier proved that the RBF network is suitable for classification tasks.

The performance of our coin counting system showed a high accuracy which exceeds 90%. Obviously, more work needs to be done to explore the capabilities of our system. The improvement of image segmentation gives our system more accurate classification performance. If the width of data can be adaptive, our system can deal with more complex situation. All of these are our further works.

References

[1] Q. Mu, J. Lim, C. W. Ji, B. K. Chung, H. Y. Kim, K. B. Uhm, C. S. Myung, J. Cho & T. S. Chon, *Density estimation of Bemisia tabaci (Hemiptera: Aleyrodidae) in a greenhouse using sticky traps in conjunction with an image processing system*, Journal of Asia-Pasific Entomology, 2008, 25-29.

[2] P. Zhao, N. H. Wang & Z. B. Pu, *A novel area measurement scheme based on a multi-resolution dynamic contour*, Optic 119, 2008, 34-40.

[3] B. Zion, V. Alchanatis, V. Ostrovsky, A. Barky & I. Karplus, *Classification of guppies (Poecilia reticulata) gender by computer vision*, Aquacultur Engineering 38, 2008, 97-104.

[4] K. Halina & P. Mariusz, *Resulted word counts optimization – A new approach for better automatic image annotation*, Pattern Recognition, vol. 41, 2008, 3562-3571.

[5] Y. P. Ginoris, A. L. Amaral, A. Nicolau, M.A.Z. Coelho & E. C. Ferreira, *Development of an image analysis procedure for identifying protozoa and metazoan typical of activated sludge system*, Water Research, Vol. 41(2007), 2581-2589.

[6] A. B. Castanon, J. S. Fraga, S. Fernandez, A. Gruber & L. D. F. Costa, *Biological shape characterization for automatic image recognition and diagnosis of protozoan parasites of the genus Eimeria*, Pattern Recognition, Vol 40(2007), 1899-1910.

[7] O. Schmitt, M. Hasse, *Radial symmetries based decomposition of cell clusters in binary and gray level images*, Pattern Recognition, Vol. 41(2008), 1905-1923.

[8] S. Y. Cho & T.W.S. Chow, *A fast neural learning vision system for crowd estimation at underground stations platform*, Neural Processing Letters, 10, no.2(Oct 1999), 111-120.

[9] S. Y. Cho, T.W.S. Chow & C.T. Leung, *A neural based crowd estimation by hybrid global learning algorithm*, IEEE Transaction on Systems, Man, and Cybernetics, Part B, Vol. 29, no 4(Aug 1999), 535-541.

[10] C. K. Ng, *A system for counting people using image processing*, Thesis for Master Philosophy, University of Hong Kong, August 2001.

[11] M. Rossi & A. Brozzoli, *Tracking and counting moving people*, Proceedings of IEEE International Conference on Image Processing 1994, 1994(3), 212-216.

[12] G. K. Gupta, *A coin counting system using adaptive statistical windowing*, Digital Image Processing, Dec 1998, 1-18.

[13] E. Purdy, *Lower bounds on Two coin-weighted problems*, Proceedings of IEEE International Conference on Image Processing 1994, Sept 2008, 1-16.

[14] T. Poggio & F. Girosi, *Regularization algorithm for learning that are equivalent to multilayer networks*, Science 247(1990), 978-982.

[15] S. Haykin, *Neural Networks: a comprehensive foundation,* Prentice Hall, 1999.

[16] C. Bishop, *Neural network for pattern recognition,* Clarendon Press, Oxford, 1995.

[17] A. Waibel & T. Hanazawa, *Phoneme recognition using time-delay neural network*, IEEE Transaction on Acoustic, Speech and Signal Processing, 37, no. 3(1987), 228-239.

[18] M. Ceccarelli & J.T. Hounson, *Sequence recognition with radial basis function networks: experiments with spoken digits*, Neurocomputing, 11(1996), 75-88.

[19] M. Egmont-Petersen, D. Ridder & H. Handels, *Image processing with neural networks - review*, Pattern Recognition, 2002, 312-316.

[20] H. Buxton, *Learning identity with radial basis function networks*, Neurocomputing, 20(1998), 15-34.

[21] W. K. Pratt, *Digital Image Processing,* John Wiley & Son Inc., New York, 1999.

[22] C. Bishop, *Improving the generalization properties of radial basis function neural networks,* Neural Computation, 3(1991), 559-588.

[23] F. Schwenker, H. A. Kester & G. Palm, *Three learning phases for radial basis function networks*, Neural Networks, 14(2001), 439-458.

[24] T. Kohonen, *Self-Organizing Maps*, Springer-Verlag, Berlin, Germany, 1997.