ISSN 0511-5728
The West Indian Journal of Engineering
Vol.38, No.1, July 2015, pp.108-114

Automated Money Detection Application for Trinidad and Tobago Currency Notes

Jesse B. Saitoo[a] and Akash Pooransingh [b,Ψ]

Department of Electrical and Computer Engineering, Faculty of Engineering,The University of the West Indies, St Augustine, Trinidad and Tobago, West Indies
[a]E-mail: jesse.saitoo@my.uwi.edu
[b]E-mail: akash.pooransingh@sta.uwi.edu

[Ψ] *Corresponding Author*

(Received 25 May 2015; Revised 09 July 2015; Accepted 20 July 2015)

Abstract: *One of the challenges faced by visually impaired persons is the recognition of currency. Mobile phones are becoming more affordable and are being used by the visually impaired community to help them in their everyday lives. This paper investigates the Local Binary Patterns (LBP) as a method to recognise Trinidad and Tobago currency notes. A mobile application was developed and the effectiveness of the LBP algorithm was tested in terms of speed, robustness to illumination, scale and rotation. The LBP algorithm realised a recognition rate of at least 95 percent for Trinidad and Tobago currency. The recognition rates on mobile devices were compared for the LBP and the ORB (Oriented FAST and Rotated BRIEF) methods.*

Keywords: *Local Binary Pattern, Multiple Object Detection, Object Recognition, Visually impaired*

1. Introduction

The Visually Impaired community has been quite progressive in their approach to leading a normal life within the sighted world. One expanding technology that is being used by visually impaired persons is the mobile phone (Wong et al. 2012). Medium end mobile phones incorporate digital cameras and information gathered from the camera may be essential to real time applications. According to Dinh et al. (2013), data can be sent to a processing centre for further evaluation using the mobile capabilities of the phone for immediate responses. The use of mobile phone applications can be used to improve the lives of the visually impaired community by allowing them to easily identify currency bills, hence making monetary transactions quicker and easier. Applications such as Looktel (Sudol et al., 2010) are readily available for popular currencies such as the US Dollars, but there is a current challenge in the Caribbean to recognise regional currencies. None exist of acceptable quality. Furthermore, it would aid in the research for developing similar applications and increase awareness for the needs of visually impaired persons.

This is an image processing problem in object recognition. According to Gevers and Smeulders (1999), colour gives strong information for object recognition. A simple and effective algorithm is to match the colour histograms of the images. Wang and Lin (2010) proposed a currency recognition system based on colour and shape. For the colour recognition, background subtraction and histogram equalisation was used to improve accuracy. The images were taken using a scanner under constant lighting conditions. The RGB (red–green–blue) colour space was used and the mean for each channel was calculated. A lookup colour table was used to determine the primary colour of the currency. From the results, they achieved 100 percent accuracy under constant lighting conditions using new currency bills. However, the recognition rate dropped drastically under varying light conditions and the use of old bills. The system proved to be effective only under ideal conditions and fails in the real world environment.

Singh et al. (2011) suggested a currency recognition using template matching. The templates were cropped from the original image and the matching was done using cross correlation. Since template matching is scale and rotation dependent, they used pre-processing techniques to achieve scale and rotation invariance. Canny edge detection combined with simple trigonometry was used to calculate the angle of rotation of the currency. The currency was then rotated to align with the template image. In order to achieve scale invariance, the currency was scaled to the original image size (512 x 512) using nearest neighbour interpolation. The template matching was found to be computationally expensive on both laptop and mobile phone. To achieve real time recognition, Singh employed an optimised template matching on a Field Programmable Gate Array (FPGA).

According to Grauman et al. (2009), image recognition has made significant progress over the past decade through the development of local invariant features. Local invariant features is extracted from the image using Feature Detectors and represented using Feature Descriptors. In the developer's paper on Oriented FAST and Rotated BRIEF (ORB), Rublee et al. (2011) proposed a fast binary descriptor which is based on BRIEF. ORB uses variations to the popular keypoint detector FAST (Features from Accelerated Segment Test) and recent descriptor BRIEF. ORB claims to surpass older methods that rely on computationally costly detectors and descriptors. It is a computationally efficient replacement to SIFT that has similar performance but is less affected by noise. ORB also outperforms SURF while being two orders of magnitude faster. Their motivation was to enhance image recognition on low power devices without GPU acceleration. ORB also performed well against image transformations such as scaling, rotation, lighting and occlusion making it suitable for real world applications.

Pietikäinen (2010) stated that Local Binary Pattern (LBP) is a simple yet effective texture operator. The basic LBP works by thresholding the neighbourhood of each pixel in an image and creating a binary representation. Due to LBP simplicity and computational efficiency, it has become a popular algorithm in many visual applications such as Face Recognition. Ahonen et al. (2004), used a variation to LBP known as Circular LBP. This variation used a variable radius and sample points allowing it to be more robust to scaling. Also, the image was separated into blocks allowing the Circular LBP to capture not only fine grain details but macro structures. The algorithm was tested on the Facial Recognition Technology database and it outperformed other methods such as Principal Component Analysis and Elastic Bunch Graph Matching. Circular LBP was able to achieve recognition of 97 percent when recognizing faces with different facial expressions. The algorithm was also tested against varying illumination conditions and achieved a recognition rate of 79 percent. Given the advantages in robustness to scaling, speed and accuracy, current literature has not shown the use of LBP for currency recognition.

In this paper, LBP was used as the main recognition algorithm for the currency recognition system. The algorithm was implemented on an android mobile phone and was used to extract texture features from the bills. The texture features can be then be represented in a compact binary form which allows for easy storage and fast recognition rates. The algorithm was adjusted to allow for multiple object detection and object recognition. The objects chosen was the birds, since this was the key distinguishing feature of Trinidad and Tobago bills. Multiple object detection was used to detect and segment the objects and then the object recognition module was used to verify the object detected in order to increase accuracy.

2. LBP for Currency Recognition

Ojala et al. (1996) proposed the original LBP operator. The original LBP operator considered a 3 x 3 neighbour around each pixel in the image and threshold the neighbours against its centre pixels to produce a binary number. The binary number was converted to a decimal value where it was concatenated into a single histogram to retain spatial information. A formal description of the algorithm can be described below:

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^p\, s(i_p - i_c) \qquad (1)$$

The central pixel can be defined as (x_c, y_c) with intensity i_c and the neighbour pixel i_p.
s(x) is the sign of the function and can be denoted as

$$s(x) = \begin{cases} 1 & if\ x \geq 0 \\ 0 & else \end{cases} \qquad (2)$$

From this description, very fine grain details can be captured from the image since the operator acts on a pixel level making it a very local descriptor. After the operator was published, it was found that a fix neighbourhood failed to encode details that varied in scale. Ahonen et al. (2004) extended the operator to use a variable neighbourhood. The basic idea was to use an arbitrary amount of neighbours and align them on a circle of variable radius. This method allowed Circular LBP to capture neighbourhoods such as spot, spot/flat, line, edge and corner which is shown in Figure 1.

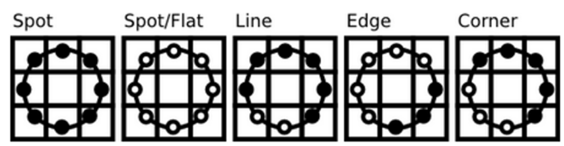

Figure 1: Circular LBP neighbourhoods
Source: Ahonen et al. (2004)

Any given point (x_c, y_c) in the image, the position of the neighbour (x_p, y_p), $\rho \in P$ can be calculated by:

$$x_p = x_c + Rcos\left(\frac{2\pi p}{P}\right) \qquad (3)$$

$$y_p = y_c - Rsin\left(\frac{2\pi p}{P}\right) \qquad (4)$$

R is given as the radius of the circle and P is the number of sample points.

This new operator is an extension of the original LBP and is called Circular LBP or Extended LBP. By using a variable circle, some of the sample points on the circle may not correspond to the pixel location and hence, the sample points get interpolated. There are many interpolation techniques; one that achieves good results with Circular LBP is bilinear interpolation. Bilinear interpolation can be represented by the function below:

$$f(x, y) \approx [1 - x \ \ x] \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1 - y \\ y \end{bmatrix} \qquad (5)$$

The histogram of the labelled image $f_i\ (x,\ y)$ can be defined as:

$$H_i = \sum I\{f_i(x,y) = i\}, i = 0, \ldots, n-1 \qquad (6)$$

The number of different labels produced by the LBP operator is given by n where

$$I \begin{Bmatrix} 1, & A\ is\ true \\ 0, & A\ is\ false \end{Bmatrix} \qquad (7)$$

The histogram now contains local micro patterns of edges, spots and flat areas over the whole image.

Ahonen et al. (2004) also incorporated spatial information in the Circular LBP algorithm. This was done by dividing the LBP image into m local regions and obtaining the histogram of each. The overall representation of the image was found by concatenating all the histograms to form one histogram. The final histogram of the image contains three levels of locality which makes it effective. The first level gives us a representation of the patterns on a pixel level, the second level is the sum of the histograms of the local regions, and the third level is the concatenation of the regional histograms to form a global histogram of the image. In order to match the histograms of the images, similarity measures such as Histogram intersection, Log-likelihood statistic and Chi square statistic may be used.

Chi square statistic was chosen as the dissimilarity measure. (Lancaster and Seneta, 2005) states that the Chi square algorithm can be represented by:

$$x^2(S,M) = \sum_i \frac{(s_i - M_i)^2}{(s_i + M_i)} \qquad (8)$$

Ahonen et al. (2004) concluded from the results that the Chi square algorithm performed better than Histogram intersection and Log-likelihood statistic for face recognition. Using the “Olivetti Research Laboratory, Cambridge database of face images” they were able to obtain excellent recognition rates up to 98 percent. The database also contained images that were scaled up to 10 percent and rotation of 20 percent, meaning the algorithm showed robustness to scale and rotation.

According to Liao et al. (2007), LBP can also be used for object detection. They proposed a variation to the original LBP called Multi-scale Block Local Binary Pattern (MB-LBP). The original LBP uses a 3x3 operator on every pixel in the image, MB-LBP however uses a 9x9 operator. The algorithm to calculate the MP-LBP is given below:

1) Find the average block intensity of block zero by adding the all the pixels in the block and dividing by 9.
2) Repeat steps 1 for blocks 1-8.
3) Compare block 1 to the centre block 0, if the intensity of block 1 is greater or equal to the centre block, set block 1 to one. If block 1 intensity is less than the centre block, set block 1 to zero.
4) Repeat step 3 for blocks 2-8.
5) Concatenate the values for each block (1-8) in a numerical order and convert to a decimal number.

After all the labels for each block have been found, the uniform patterns are extracted. Uniform patterns can reduce the length of the feature vector and also makes the LBP robust to rotation. A local binary pattern is defined as uniform if it contains at most two bit wise transitions from 0 to 1 or 1 to 0 if the bit pattern is traversed circularly. The following is a group of uniform patterns “00000000 (0 transitions), 01110000 (2 transitions), 11001111 (2 transitions)”. Non uniform patterns contains two or more transitions such as “11001001 (4 transitions) and 01010010 (6 transitions)”.

This type of LBP is called Uniform LBP or Rotation invariant LBP. MB-LBP histograms were plotted using the FRGC ver2.0 face database. It was observed that the top 63 bins of the histogram corresponded to the uniform binary patterns. From this, a new definition of uniform local binary pattern can be establish as the top 63 bins where the non-uniform patterns can be represented as one bin or label. This shortens the histogram to 64 bins in total. This new representation was called Statistically Effective MB-LBP (SEMB-LBP).

Figure 2 shows a system overview of the various sub modules that form the implemented currency recognition system. The android camera captures one frame in an RGBA format. The Input Image module pre-processes the image to send it to the multiple object detection module. This pre-processing is a conversion of the image from a RGBA format to grayscale. The multiple object detection module checks to see if any objects are present in the given frame. If there is an object, the object is cropped and sent to the object recognition module However, if no object is present or more than one object is present, the input image module sends another frame. The object recognition module receives the cropped frame and matches the cropped image to the images in the database. If there are no matches the system goes back to the Input Image module, but if there are matches, the results are sent to the Voice Output module. The Voice Output module receives results from the Object Recognition module and it speaks the results to the user. The system is then repeated, starting from the Input Image module.

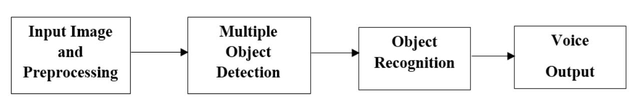

Figure 2: Sub-modules of the Currency Recognition System

3. Results

Each of the tests involved a different experimental setup. Table 1 shows the test used and its respective experimental setup. All of the testing was done using an

Intel 2.1 GHz dual core processor on a windows 7 OS. Each sub-section would contain the detail of the testing data and experimental setup. This section highlights the algorithm's performance in terms of robustness to illumination, speed, scale and rotation. The recognition rate of the algorithm was then compared to the popular ORB method from the developers' paper by Rublee et al. (2011). Figure 3 shows the testing data used in the experiments which consisted of Trinidad and Tobago currency. The images consisted of cropped images of the Front side of the Trinidad currency bills and each image shows a particular bird which is the object to be recognised.

Table 1: Experimental Setup

Test	Hardware/Software	Method
Original LBP Transformation Illumination Variations Colour Variations	Intel Pentium CPU B950 @ 2.10 GHz, 4Gb ram using Python 2.7 with OpenCV 3.0	The five and one dollar images from the dataset were imported using a Python script. The LBP algorithm was coded into a Python module and the binary LBP image transformation was performed.
Circular LBP: Radius Variation Chi-Square distance Added Noise Scaling and Rotation Variations	Intel Pentium CPU B950 @ 2.10 GHz, 4Gb ram using Visual Studio 2013 C++ with OpenCV 3.0 – Face Recognizer API	A sample one dollar was imported using C++ and converted to a LBP histogram and stored in a database. A copy of the sample dollar was used and altered according to the various tests. The copy dollar was then converted to a LBP histogram and compared to the original sample using Chi-Square.

Figure 3: Sample of Trinidad and Tobago Currency Notes used as Testing Data

Typically, training data contained 800 images per bill. The images consisted of cropped images of the birds on the bill in different lighting conditions. Four hundred images were taken at zero degrees and another four hundred images were taken of the object at 180 degrees. Figure 4 shows a sample of the Training data for one dollar bill.

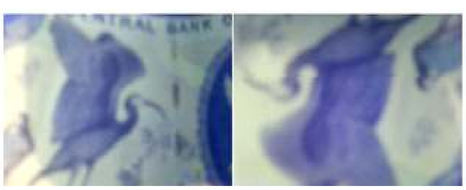

Figure 4: Sample of Trinidad and Tobago 1 Dollar Note used as Training Data

3.1 Robustness to Illumination

Figure 5 shows the verification of the Circular LBP robustness to illumination variations and validates the choice of the algorithm for a real world application. The dataset consisted of a sample one dollar bill, however the results can be replicated using other currency bills. The illumination was varied with Visual Studio C++ using OpenCV libraries. The altered image was then sent to a Python script where the Circular LBP is performed. The figure shows the transformation of the original image, the original image darkens and the original image brightens respectively. Comparing the Circular LBP transformations, it can be seen that they are relatively unchanged which shows Circular LBP robustness to illumination variations.

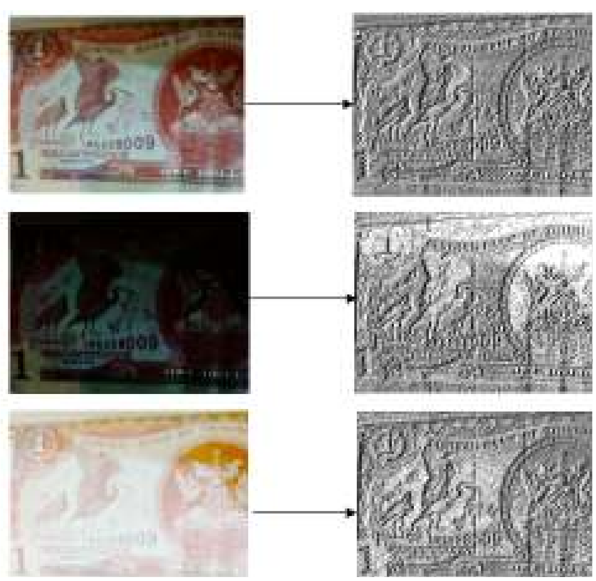

Figure 5: Example of LBP image transformation

3.2 Speed

Figure 6 shows the relationship between the Prediction time and the number of histograms in the database. The prediction time is the time to calculate the circular LBP of the input frame, convert to a histogram and compare it using chi-square to the histograms stored in the database. For 100 histograms in the database, the time taken would be 1.288 seconds. The graph also shows a linear relationship. Therefore, it can be said that the time complexity of the circular LBP plus the histogram comparison is proportional to the amount of histograms in the local database. When tested on the mobile device, prediction time was less than 20ms, making LBP suitable for real time performance.

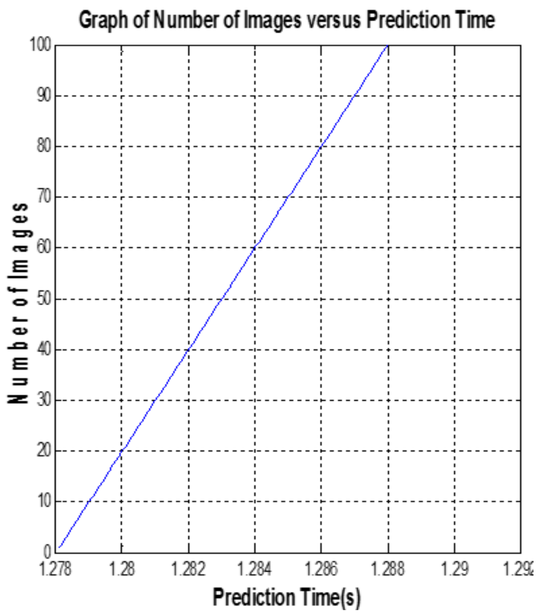

Figure 6: Number of images versus Prediction time

3.3 Scale and Rotation

Figure 7 shows Circular LBP robustness to scaling variations. It validates the choice of Circular LBP for this application where the currency is not at a fixed scaled. The test data consisted of a sample one dollar and a duplicate of the same one dollar except scaling and rotation variations were made. OpenCV libraries with Visual Studio C++ were used to vary the scale. The amount of scaling was varied and the Chi-square distance was observed. From the Figure 7, Circular LBP responds well to scaling, tolerating scale shifts up to 70 percent of the image with the Chi-Square distance remaining below 30. 100 percent represent no scaling.

Figure 8 shows how the Chi-Square distance responds to a changing angle of rotation. It validates the choice of Circular LBP for this application where the currency is not at a fixed orientation. From the data, the average confidence level is approximately 53. However the confidence level drops at the angles 90,180,270, and 360. This was due to interpolation, since at these angles no interpolation is performed.

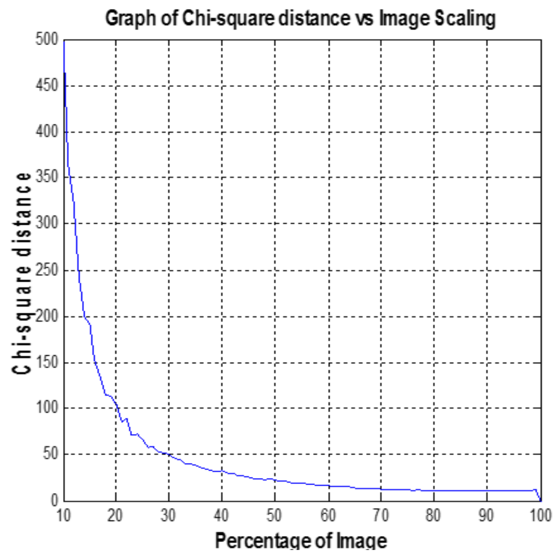

Figure 7: Figure 40-Graph of Chi-Square distance vs Image Scaling

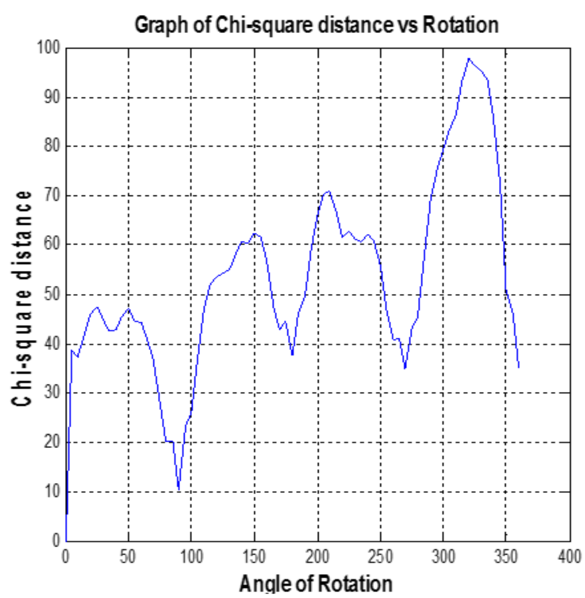

Figure 8: Graph of Chi-Square distance versus Rotation

In Figure 9, the recognition rate of LBP is shown. The dataset consisted of 100 images of the six currencies. The test was performed using Visual Studio C++. Both test for LBP and ORB were carried out separately but under similar lighting conditions, no scaling and no rotation. The Recognition Rate is a key factor in Currency Recognition. From Figure 9, it can clearly be seen that LBP performs on average above 95

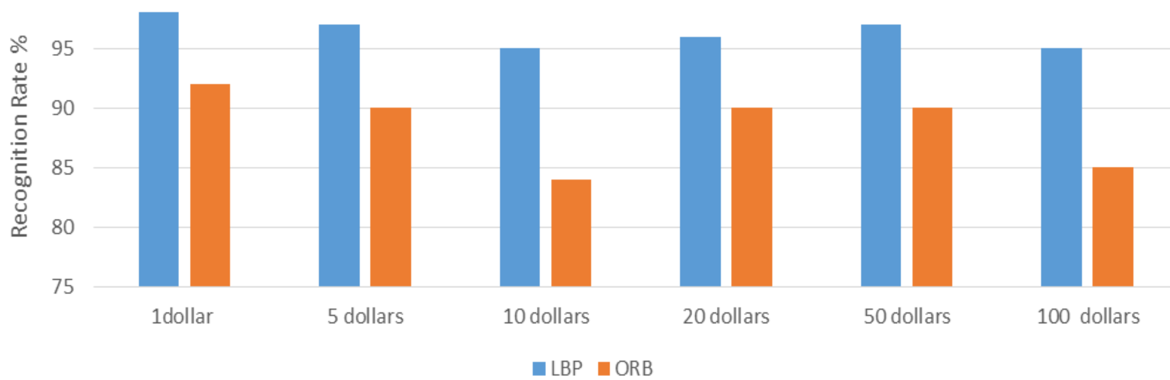

Figure 9: Recognition Rate of LBP versus ORB

percent in the Recognition Rate test compared to 85 percent of the ORB method.

4. Conclusion

LBP proves to be a simple yet powerful algorithm. LBP variations such as Circular LBP and Multi-scale Block LBP can accomplish object detection and object recognition. LBP effectively extracted fine grain texture features from the currency. These features were easily represented as a compact histogram which consumed little memory and allowed for fast histogram matching The low complexity of the histogram matching permitted real time and on-device matching on mobile phones.

From the results, LBP is computationally fast and achieves a prediction time of less than 20ms on mobile phones, allowing for real time currency scanning. Also, LBP showed high recognition rates of 95 percent and above, which is a key factor in currency recognition. LBP is also robust to various lighting conditions and exhibits excellent performance in poorly lit rooms. Also LBP is invariant to scaling and rotation making it practical for the currency recognition system as the visually impaired user would not know the exact distance from the bill or the orientation.

There were many limitations to this system; however, satisfactory results were still able to be obtained. Some major limitations were:

1. Lack of Resources- Training the Recognition System was a computationally intensive task. On a 2.1 GHz dual core CPU, training time took over half hour per currency bill.
2. Testing- testing was only performed on a Samsung galaxy S3 and a Samsung tab 3. The application compatibility with other android devices could not be tested due to time and resource constraints.

This paper presents a fast and accurate method to recognise Trinidad and Tobago currency notes. This method uses the LBP method for object recognition and proves to be robust to rotation and scale variation. It was implemented on an Android platform and provided a recognition rate of at least 95 percent which was better than the ORB method presented in (Rublee et al., 2011). Deployment of the work in this paper would greatly assist the visually impaired community of Trinidad and Tobago in making monetary easier cash transactions. In the future, this work may be extended to include other regional currencies.

References:

Ahonen, T., Hadid, A., and Pietikäinen, M. (2004). “Face recognition with local binary patterns”, *Proceedings of the ECCV 8th European Conference on Computer Vision*, Prague, Czech Republic, Springer Berlin Heidelberg, , May 11-14, pp. 469-481

Dinh, H. T., Lee, C., Niyato, D., and Wang, P. (2013), “A survey of mobile cloud computing: architecture, applications, and approaches”, *Wireless Communications and Mobile Computing*, Vol.13, No.18, pp.1587-1611.

Gevers, T., and Smeulders, A. W. (1999), “Color-based object recognition”, *Pattern Recognition*, Vol.32, No,3, pp.453-464.

Grauman, K., and Leibe, B. (2010), *Visual Object Recognition*, Morgan & Claypool Publishers.

Lancaster, H. O., and Seneta, E. (2005), *Chi Square Distribution*, John Wiley & Sons, Ltd.

Liao, S., Zhu, X., Lei, Z., Zhang, L., and Li, S.Z. (2007), “Learning multi-scale block local binary patterns for face recognition”, In: *Advances in Biometrics*, Springer Berlin Heidelberg, pp.828-837.

Ojala, T., Valkealahti, K., Oja, E., and Pietikäinen, M. (2001), “Texture discrimination with multidimensional distributions of signed gray-level differences”, *Pattern Recognition*, Vol.34, No.3, pp.727-739.

Pietikäinen, M., Hadid, A., Zhao, G., and Ahonen, T. (2011), “Computer vision using local binary patterns”, *Science and Business Media*, Springer, Vol. 40, pp.13-14.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011), “ORB: an efficient alternative to SIFT or SURF”, *Proceedings of 2011 IEEE International Conference on Computer Vision* (ICCV), IEEE, Barcelona, Spain.November, pp. 2564-2571.

Singh, B., Badoni, P., and Verma, K. (2011), “Computer vision based currency classification system”, *International Journal of Computer Applications*, Vol.2, No.4, pp.34-38.

Sudol, J., Dialameh, O., Blanchard, C., and Dorcey, T. (2010), “Looktel - A comprehensive platform for computer-aided visual assistance”, *Proceedings of 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, IEEE, San Francisco, CA, USA.June, pp. 73-80.

Wang, Y., and Lin, S. (2010), *Currency Recognition System Using Image Processing*, (BSc Thesis), Department of Industrial Development, IT and Land Management, Faculty of Engineering and Sustainable Development, University of Gävle.

Wong, M. E., and Tan, S. S. (2012), "Teaching the benefits of smart phone technology to blind consumers: Exploring the potential of the iPhone", *Journal of Visual Impairment & Blindness*, Vol.106, No.10, pp.646.

Authors' Biographical Notes:

Jesse B. Saitoo *is an undergraduate student at the Department of Electrical and Computer Engineering of The University of the West Indies (UWI). He is interested in software development and mobile application development. He also has a keen interest in image processing and researching innovative and efficient methods in image processing applications for mobile phones.*

Akash Pooransingh *is a lecturer at the Department of Electrical and Computer Engineering. He is a member of the IEEE. He completed his M.Sc. in Communications Systems with Distinction in 2005 and his PhD at the Department of Electrical and Computer Engineering of The University of the West Indies (UWI) St. Augustine Campus (STA) in 2012. His area of interest includes image processing and video analysis of sport broadcasts.*

■