

Design of Foreign Currency Recognition Application using Scale Invariant Feature Transform (SIFT) Method based on Android (Case Study: Singapore Dollar)

Mohammad Rizky Adhiguna, Budhi Irawan and Anggunmeka Luhur Prasasti
Department of Computer Engineering, Telkom University, 40257 Bandung, Indonesia
anggunmeka@telkomuniversity.ac.id

Abstract: Money is the most commonly used means of payment by the public. But without denying fake money is widely circulated and there are still many people who are less accurate in recognizing the authenticity of the money. This will be bad for social life as we known that money is main payment that can use by everyone. For people with disabilities that lack of visual itself will be hard to know the identity from the money. With this problem in this research will be designed and implemented an Android based mobile application that can recognize currency with image. Applications designed using the Scale Invariant Feature Transform (SIFT) method that can provide information to users about their nominal and authenticity of the money using Indonesian. This application can help people who are less aware of information about genuine money and people with disabilities to find informations about authenticity of Foreign currency. With this application people with disabilities, also can tell the identity of the money itself with more accurate considering this app has implemented by SIFT method on feature extraction but the process time will be longer because the SIFT method itself has a fairly complicated calculation process. From these complex calculations will also produce better accuracy.

Key words: Foreign currency, mobile application Android, SIFT, calculation process, accuracy

INTRODUCTION

Banknotes are the most commonly used means of payment by the public in which the currency itself has a nominal and unique uniqueness of each state. But it cannot be denied that counterfeit money is still circulating wildy and the public does not pay much attention to it (Bhurke *et al.*, 2015). Moreover, for the disabled themselves who have deficiencies in terms of vision will also experience difficulties in detecting the authenticity of money (Chhaya and Sheetal, 2016; Kulkarni *et al.*, 2016).

With these problems in this study will be designed an Android-based mobile application that can find out the nominal and the authenticity of a piece of Singapore Dollar banknotes. The money in question is Foreign currency where there are still many people who do not know the nominal money and also, whether the money is genuine or not, especially, for people with disabilities, especially the blind who have a deficiency in vision, it will be very difficult to detect the authenticity of these banknote (Chhaya and Sheetal, 2016). The reason

for using Singapore dollar denominations because the Singapore dollar is one of the most frequently used money in the world (Deil, 2014). This application will convey information in Indonesian by means of the user inputting data in the form of photos that can be taken through a smartphone camera (Fig. 1).

This application is designed using the SIFT (Scale Invariant Feature Transform) method which is a method that will be used for feature extraction where this method will select unique parts of the object (Hamida *et al.*, 2016). Then, the section will be used as data to match training image data that has been stored in the database (Hamida *et al.*, 2016). With this application can help people with disabilities in recognizing the authenticity of Singapore dollars.

In the journal Iyad Abu Doush, Sahar AL-Btoush entitled "currency recognition using a smartphone: Comparison between color SIFT and gray scale SIFT algorithms" explains Arab detection using SIFT method in color image and grayscale image using smartphone (Doush and Sahar, 2017).

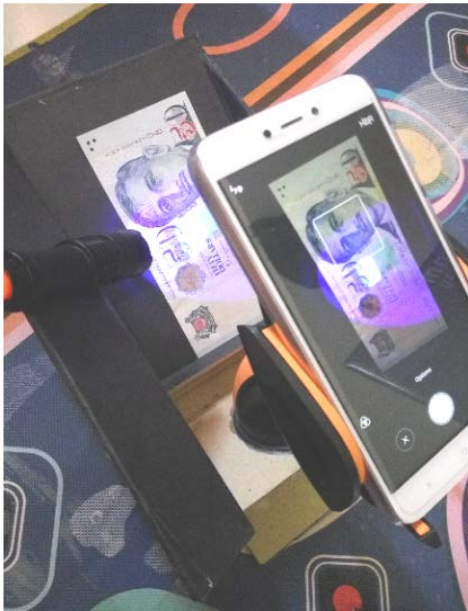


Fig. 1: The process of taking pictures of banknotes that will be detected using the application

MATERIALS AND METHODS

Singapore dollar banknotes: Money is the most used payment tool in each country (Kavya and Devendran, 2015). Money itself has two types, namely paper money and coins (Sneha *et al.*, 2017). The money used here is banknotes and the currency used is the Singapore currency.

Figure 2-4 are examples of images of Singapore dollars that will be used in this study. Nominal used is SGD 10, SGD 50 and SGD 100. Then, in the second picture is an example of Singapore dollar bill that has been irradiated UV light. The reason for choosing the nominal is because the nominal is one of the most commonly used in the world (Deil, 2014).

Scale Invariant Feature Transform (SIFT): SIFT is an algorithm used to detect features in an image where the end result of the detection process will be used as a descriptor of the image (Anonymous, 2018a-c). At the time of detection will be selected keypoint, then, the data that has been obtained will be stored in the database where the stored data will be used as a descriptor. The purpose of using this method is to find unique patterns that exist on objects or images that will be acquired (Kavya and Devendran, 2015). SIFT method itself has a fairly high level of accuracy but due to the high accuracy of the process time required is also takes a long time (Hamida *et al.*, 2016).



Fig. 2: An example of 100 Singapore dollars before being exposed to UV rays and exposed to UV light



Fig. 3: An example of money is 50 Singapore dollars before it is irradiated by UV rays and after being irradiated by UV rays

Finding extreme value: The extreme value search process is needed to get the keypoint candidate that will be used for the extraction process. To get extreme values, the Difference of Gaussian (DoG) function is used as follows (Anonymous, 2018a-c):

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \quad (1) \\ = L(x, y, k\sigma) - L(x, y, \sigma)$$

In the above function is explained that $D(x, y, \sigma)$ image results from extreme value search done, $G(x, y, \sigma)$ is the variable used in the Gaussian function, then $I(x, y)$ is an image that has been acquired and $L(x, y, \sigma)$ is an image to get extreme value. The constant value k is used to maximize keypoint detection (Anonymous, 2018a-c).



Fig. 4: An example of money is 10 Singapore dollars before it is irradiated by UV rays and after irradiated by UV rays

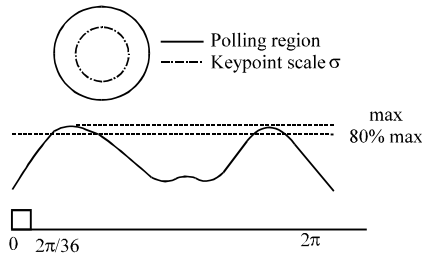


Fig. 5: Histogram of orientation gradient in DoG

Keypoint localization: If the required candidate keypoint has been obtained, then the keypoint localization will be performed (Anonymous, 2014). This process serves to determine the exact location of the keypoint by selecting a low value keypoint and eliminating it. To be able to remove the keypoint, it will be determined using the function of the extreme value $\mathcal{D}(\hat{x})$ obtained by the equation as follows:

$$\mathcal{D}(\hat{x}) = \mathcal{D} + \frac{1}{2} \frac{\partial \mathcal{D}^T}{\partial \mathbf{X}} \hat{x} \quad (2)$$

Value $\mathbf{x} = (x, y, \sigma)^T$ is the value used for balancing. Then extreme value \hat{x} obtained through the derivative of the function \mathbf{x} (Yuan, 2013) (Fig. 5).

Giving orientation: Giving orientation to keypoint aims for the keypoint location that has been set after, the keypoint localization will remain in the specified location. So, if the image is rotated, the location of the keypoint that has been set will not change (Anonymous, 2018a-c).

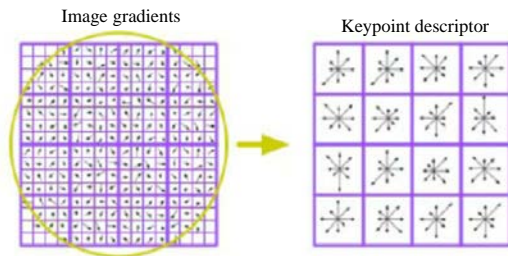


Fig. 6: Change of the image gradient to a keypoint descriptor

Keypoint descriptor: Descriptor is needed inside the keypoint to set the point of the keypoint that has been obtained despite the change in the image from the point of view and lighting. The descriptor on the keypoint itself will take the form of a vector obtained after passing through the histogram orientation process (Yuan, 2013). To get orientation histogram can be obtained from calculation of gradient magnitude and orientation at keypoint area (Fig. 6).

System planning: The system in this application is designed to be able to perform image processing where the input given in the form of banknotes of dollars Singapore and can provide output in the form of authenticity of money in the form of text.

Figure 7 is a general overview of the system on the application to be created. Like image processing in general, after the system receives input from the user, it will be pre-processing first on the input image. The purpose of the pre-processing stage itself is to prepare the image to be used to fit the required at a later stage. After the image in accordance with the required, then will go to the extraction stage where at this stage the system will choose which parts will be made a unique part of the image inputted to be used in the classification stage. The SIFT method itself is required in the extraction portion of the image. In the classification stage the results of the extraction itself will be used to test the compatibility of new inputs with data already stored in the database. After completion of the classification process, then the new system can provide output to the user in the form of text.

Use case diagram: Figure 8 it is explained about what users can do on the system. When the application is run, the user will get two choices on the main menu that is for the dataset and capture. But before entering the main menu, the user will be given information on how to use the application in the form of voice to facilitate disability in using the application. When in the main menu, the user needs to create a dataset that will be saved first before

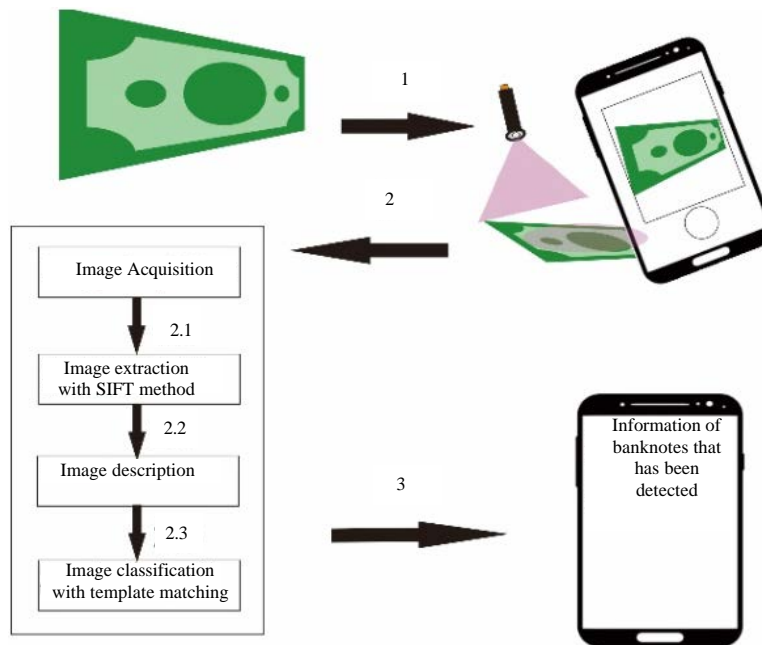


Fig. 7: Illustration of the entire system process

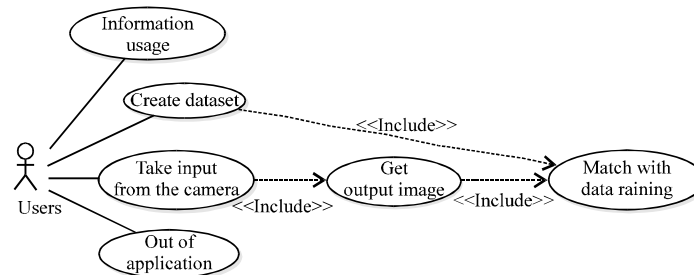


Fig. 8: Use Case Diagram based on the application that has been created

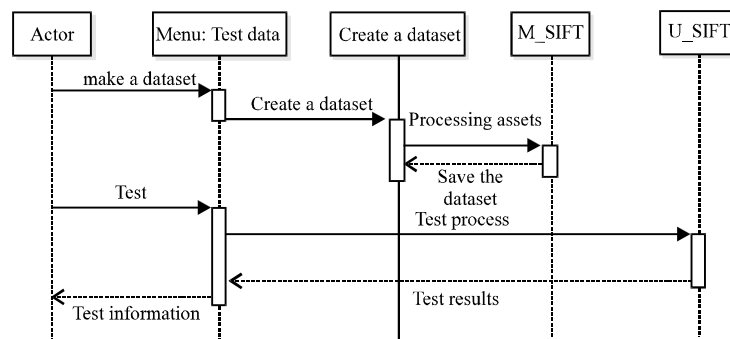


Fig. 9: Sequence diagram used in the application

capture. When you have created a dataset, the user can capture where the input will be matched with the stored dataset. After testing, the user will get output in the form of text and voice. In addition, users can also directly exit the application.

Sequence diagram: Sequence diagrams are diagrams that display interactions between objects in the system used. In this system 3 objects are used, namely actors, menu: Uji Data, buat Data set, M_SIFT, Dan U_SIFT.

Figure 9 shows the interaction between objects on the sequence diagram that will be used in the system. At



Fig. 10: Examples of training data used at each nominal used

the time the user will detect the image, then, the menu object will make the acquisition of the image. After acquiring the image, the system object will continue the next process. System object will extraction on the image, then the system will do the classification on the image. After the process is complete, the system will output the output to the menu object and the user will get the output from the menu.

Data training: Training data is data that will be stored as a dataset where this data is useful for matching with input data. The training data used has 3 different images based on the nominal banknotes used which are nominal SGD 10, SGD 50 and SGD 100. Each banknote has a watermark that will be seen when exposed to UV light. Figure 10, it can be seen that each nominal has a different watermark. This watermark will be the focus to detect the authenticity of money and nominal money. For nominal SGD 10 itself has a watermark inscribed number 10, nominal SGD 50 has a watermark inscribed number 50 and nominal SGD 100 has a watermark inscribed number 100.

Graphical User Interface (GUI): The GUI to display is a view that is inside an already created app. In the main menu of applications where it will be given two buttons, namely to create datasets and capture. When selecting for a dataset, the application will enter the menu for the dataset. In the menu for this dataset will process the training data that is stored, so that, it can be used in the matching process. Figure 11 is a display after the matching process which will display the capture output along with the nominal and authenticity of the banknote.

RESULTS AND DISCUSSION

Testing: In this test, several testing scenarios will be carried out. The purpose of the test itself is to find out what parameters can support the application to run optimally. Here is the test scenario that was done:



Fig. 11: Display results from matching

- Testing image resolution
- Testing amount of data test
- Environmental lighting test
- Image taking distance testing
- Testing system time response

In this test, the accuracy level will be obtained by calculating the average of successful test results by using this application. The accuracy calculation formula is as follows:

$$\text{Level of accuracy} = \frac{\text{Total testing is correct}}{\text{Total testing}} \times 100\% \quad (3)$$

Table 1: Testing image resolution

Resolution	Dataset time (sec)	Time process (sec)	Nominal accuracy (%)	Authenticity accuracy (%)
60×60	1.094	180	0	0
100×100	2.934	180	76	90
500×500	38.292	180	70	83

Table 2: Amount of data train

Amount of data	Accuracy (%)
9	66
18	76
24	63

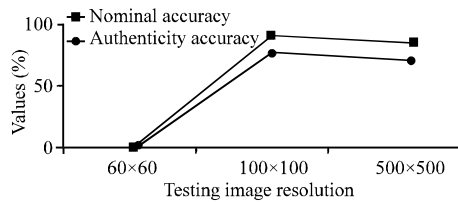


Fig. 12: Graph of image resolution testing

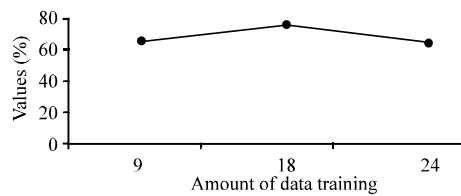


Fig. 13: Graph of testing the amount of training data

Tests on the system is done in order to know whether the system made has met the criteria that have been determined on the purpose of this final project. The test will be how the system can detect Singapore dollar banknotes using Scale Invariant Feature Transform (SIFT) method. Train data used is data that is captured manually and already done cropping on the data train. The data will be collected which will then be searched the keypoint when creating the data set. Then, the data will be matched with the results of the acquisition. Following are the results of the test:

Table 1 shows the results of testing on image resolution where the resolution used is 60×60 pixels, 100×100 pixels and 500×500 pixels. Based on the results obtained, the resolution of 100×100 pixels gets the best accuracy with an accuracy rate of 76% (Fig. 12).

Table 2 shows the results of testing the number of training data used where the amount of data used in the matching process is 9, 18 and 24 data. Based on the test results obtained, the best amount of data used is 18 data with 76% testing accuracy (Fig. 13).

Table 3 shows the results of an environmental light test where light is measured using a light meter application on a smartphone with a light intensity of

Table 3: Environmental light test

Light (lux)	Accuracy (%)
15000	43
7500	66
80	76

Table 4: Image of taking distance

Distance (cm)	Accuracy (%)
8	40
19	76
30	46

Table 5: System time response

Device	Android version	Processor (Ghz)	RAM (GB)	Time dataset (sec)	Time processor (sec)
1	5.0.2	Octa-core 2.0	3	1.774	180
2	7.1.2	Octa-core 1.4	4	2.793	108

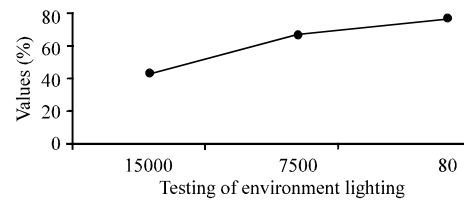


Fig. 14: Graph of environmental light testing

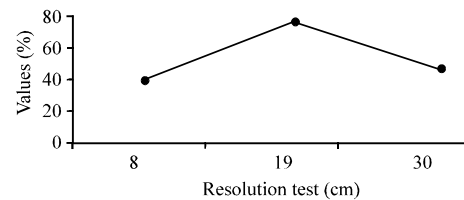


Fig. 15: Graph of image resolution testing

15000, 7500 and 80 lux. Based on the test results obtained, the best light for the test is in the intensity of 80 lux where the accuracy is 76% (Fig. 14).

Table 4, the results of the distance image retrieval are shown where the distance used is 8, 19 and 30 cm. Based on the results obtained, a distance of 19 cm to get the best accuracy results where the accuracy of 76% obtained (Fig. 15).

Table 5 shows the results of testing the system time response on two different devices where 1 use Android 5.0.2 OS, Octa core 2.0 processor and 3GB RAM. While the device 2 using OS 7.1.2, Octa processor cores 1.4 and 4GB RAM. On device I has a process time of 1,774 sec dataset and a matching time of 180 sec. On device 2 has a processing time of 2,793 sec dataset and 108 sec matching process time (Fig. 16).

By looking at the above test data, we can see that the accuracy level using the SIFT method can be determined by several parameters such as resolution, amount of training data used, image retrieval distance, ambient light and system response time. Resolution used is very

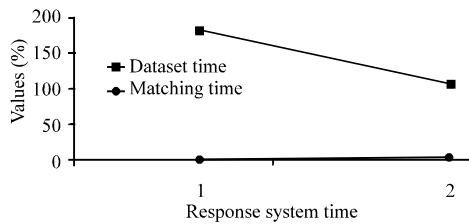


Fig. 16: Graph of system response time testing

influential to the test results where the greater the resolution used, the more keypoint obtained, so, the higher the accuracy too. However, due to high accuracy, the processing time becomes longer because the SIFT method itself is more focused on getting more keypoint. For the distance itself, also, has an important role in the accuracy which in this test is done two different distances. For 19 cm distance taking is based on the appropriate distance when using the tool. While the distance of 8 cm is when not using a tool, so, the user needs to measure the manual distance used with a distance of about 4 fingers. The smartphone used also determines the response time given for Random Access Memory (RAM) is large enough, the process is also faster.

CONCLUSION

Invariant Scale Feature Transform (SIFT) method can be implemented on an Android-based mobile application to detect Singapore dollar banknotes but in the testing process takes a long time to get results because the process required by the SIFT method is quite long.

Image resolution and number of images used can be a supporting factor in maximizing the accuracy of the SIFT method. The greater the resolution and the more amount of data used, the better the accuracy but the required processing time will be longer.

To get maximum results, the resolution used is 100×100 pixels. Another reason to use the resolution is to reduce the time duration of the process when image detection and accuracy are also quite high is 76%.

The cause of the length of the SIFT process is when passing the blob detection process where if using ORB in the process will take pixels in the image in a wide enough range in 1 point but the accuracy will decrease. On the other hand, the SIFT method in the process will take pixels in the image in a very small range, so that, the accuracy obtained is better but the processing time is longer.

ACKNOWLEDGEMENTS

The researcher thank the Telkom University providing the research facility.

REFERENCES

- Anonymous, 2014. Introduction to SIFT (Scale-Invariant Feature Transform). Opencv dev team, USA. https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_feature2d/py_sift_intro/py_sift_intro.html
- Anonymous, 2018a. Finding keypoint. AI Shack Ltd, SIFT, USA. <http://www.aishack.in/tutorials/sift-scale-invariant-feature-transform-keypoints/>
- Anonymous, 2018b. Keypoint orientations. AI Shack Ltd, SIFT, USA. <http://www.aishack.in/tutorials/sift-scale-invariant-feature-transform-keypoint-orientation/>
- Anonymous, 2018c. LoG approximations. AI Shack Ltd, SIFT, USA. <http://www.aishack.in/tutorials/sift-scale-invariant-feature-transform-log-approximation/>
- Bhavani, S.A., 2017. Currency recognition using SIFT. Intl. J. Comput. Appl., 167: 15-20.
- Bhurke, C., M. Sirdeshmukh and M.S. Kanitkar, 2015. Currency recognition using image processing. Intl. J. Innovative Res. Comput. Commun. Eng., 3: 4418-4422.
- Chhaya, W.V. and G.S. Sheetal, 2016. Android application on Fake Indian Currency Recognition. Intl. J. Mag. Eng. Technol. Manage. Res., 3: 426-429.
- Deil, S.A.F., 2014. The 10 Most widely used currency in the world. Liputan6.com, USA. <https://www.liputan6.com/bisnis/read/2017867/10-mata-uang-paling-banyak-digunakan-di-dunia>
- Doush, I.A. and A.B. Sahar, 2017. Currency recognition using a smartphone: Comparison between color SIFT and gray scale SIFT algorithms. J. King Saud Univ. Comput. Inf. Sci., 29: 484-492.
- Hamida, S.B., R. Azizi and A. Maaloul, 2016. Intelligent system of M-vision based on optimized SIFT. J. Comput. Commun., 4: 52-62.
- Kavya, B.R. and B. Devendran, 2015. Indian currency detection and denomination using SIFT. Intl. J. Sci. Eng. Technol. Res., 4: 1909-1911.
- Kulkarni, R., S. Poul, M. Mulik and S. Swami, 2016. Object detection, location finder for blind people using android application. Intl. J. Res. Advent Technol., 4: 30-33.
- Sneha, R., P.K. Deepika, S. Nagaveni, S.D. Shree and K.S. Asha, 2017. Automatic counterfeit currency detection using image processing. Intl. Res. J. Eng. Technol., 4: 1208-1211.