# Herbs Recognition Based on Android using OpenCV

**I Wayan Agus Suryawibawa**
Information Technology Department, Udayana University, Bali, Indonesia
Email: suryawibawa.32@gmail.com

**I Ketut Gede Darma Putra and Ni Kadek Ayu Wirdiani**
Information Technology Department, Udayana University, Bali, Indonesia
Email: ikgdarmaputra@gmail.com, ayu_wirdi@yahoo.com

*Abstract*—Herbs are used in traditional medicine. There are so many herbs are spread across the world, it is difficult to memorize it all. This paper describes an android application to recognize herbs by their leaf characteristics (shape, veins, and keypoints). Shape and veins of leaves are recognized by Invariant Moment Method as the feature extraction. City Block Distance used to calculate the distance between the features. Whereas for detection and keypoints extraction using Oriented FAST and Rotated BRIEF on OpenCV library. This keypoints distance calculation using Brute-Force Hamming. Matching is done by calculating the shortest distance between test image and reference image. If the result is less than or equal to threshold then image is match. Experiment result show this application can achieve 79% of success rate by using keypoints. This result is influenced by glossy leaf surface, so there is many reflected light that become noise.

*Index Terms*—Android, Herbs, OpenCV, Invariant Moment, Oriented FAST and Rotated BRIEF.

## I. INTRODUCTION

Plants have many types and usefulness, one is its uses as a medicine. Plants that used for medicine is call herbs. Herbs are very important as alternative medicine such as traditional medicine. Traditional medicine can be used to minimize the use of chemicals by using natural materials as much as possible. So the herbs can process into medicines manually without using machines. There are so many herbs spread across the world, it is difficult to memorize it all.

One way that can be used to recognize herbs is by using digital image processing on their leaves [1-4]. Digital image processing is a process to improve quality of the images and prepare it for ready to extract. This feature is used to determine the match of image. There are many methods that can be used in the extraction of features, depending on the features and speed of computation required.

Some characteristics of leaves that can be recognized are shape [1, 4] and veins [1,3]. Invariant Moment method can be used to recognize shapes and leaf veins [2].

This method gave 7 moment invariants as value of leaf feature [5]. Invariant Moment used because it can give faster computation for smartphone device. The advantages using Invariant Moment is to minimize the effect of position, rotation, and scaling of leaf image [5, 9]. The method that is used to calculate the distance of the Invariant Moment value is City Block Distance. This is a method of matching by searching the shortest value between the test image and the reference image. If the distance is less than or equal to the threshold then the image is match. Despite using different features, but the methods used are still the same that is Invariant Moment. So, one more method is required as a comparison that might be able to provide better results.

Oriented FAST and Rotated BRIEF (ORB) method from OpenCV Library can be used to detect and extract the value of keypoints on leaf [6, 7]. This method used as comparison of Invariant Moment. OpenCV or Open Computer Vision is an Application Programming Interface (API) Library that used in digital image processing. OpenCV Library can obtained higher performance than the self-made algorithm build in Android Library [8].

Herbs recognition has been implemented on desktop application. The problem is the computer hardware is big and can't mobile. This application will be more useful if it developed on mobile device such as Android Smartphone. Android is one of the best platforms for mobile device. Many people are already using Android Smartphone as their device, so it is expected a lot of people can use this application.

The remainder of the paper is organized as follows. In section II, a review of the necessary method is presented, like method that used on preprocessing, feature extraction, or matching. The proposed algorithm of this recognition system is described in Section III. After that, application of the proposed algorithm and the result of the application analysis are discussed in section IV. Section V shows the conclusion from section IV.

## II. METHODS

*A. Alpha Beta filter*

Grayscale is gray color image with color levels between value 0 to 255 that obtained by this following formula [9]:

$$G(x,y) = (R+G+B)/3 \qquad (1)$$

Grayscale value for each image pixel is obtained by calculating the average value from the color values of *R* (red), *G* (green), and *B* (blue) on each pixel.

### B. Gaussian Blur to Reduce Noise

Grayscale image is converted into smoother image by using Gaussian Blur that aims to reduce the noise in the image. Noise is reduced to get good results for the recognition of shapes and veins. This system use σ = 2 and *kernel* = 3x3. This process will generate a little blurred image [9].

### C. Binarization to Get The Shape of Leaf

Binary image is the image that only has 2 color value that is 0 (black) and 255 (white). Changing the image into binary, then on grayscale image (the image that has converted to Gaussian Blur) color value is determined by reference value (threshold). Threshold value is determined using Otsu method [10]. This method used the ratio of gray level value to obtain threshold value automatically. If the grayscale value less than or equal to the threshold value, then it is changed to 0, while if it more than the threshold value then the value is changed to 255. Converting images to binary form is intended to obtain leaf shape features using Invariant Moment.

### D. Canny Edge Detection

Canny edge detection is used to detect the edges of the herb image based on color differences, so that the herb veins can be seen. Features of the herb veins will also be extracted using Invariant Moment method. The first step is to convert the RGB image to grayscale and eliminates noise with Gaussian Blur. The next step is to perform edge detection, such as using Sobel operator to perform a search in horizontal (*Gx*) and vertical (*Gy*) [9].

$$G_x = \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix} \qquad (2)$$

$$G_y = \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \qquad (3)$$

The next step is to determine the direction of the edge by using the following formula:

$$\theta = arctan\left(\frac{G_x}{G_y}\right) \qquad (4)$$

And then, divide it into 4 colors so that lines with different directions have different colors. The distribution is from 0 to 22.5 and 157.5 to 180 degrees is blue, from 22.5 to 67.5 degrees is yellow, 67.5 to 112.5 degrees is red, dan112, 5 -157.5 degrees is green. Then to produce thin line, the line edge is minimized by applying Nonmaximum Suppression. The last step is to do binarization by using threshold. Threshold value of canny found using experiments that will be discussed in Section IV.

### E. Feature Extraction by Using Invariant Moment

Feature extraction is performed for shape and veins features of herb were Invariant Moment. Feature extraction is performed on the binary image for shape feature, and the canny edge detection for veins feature. Moment Invariant method will produce 7 moments that describe the area, position, orientation and other defined parameters. Invariant Moment can be used to minimize the difference value on same image when image experience translation, rotation or scaling. The formula of invariant moments can be seen on references [5, 9].

### F. Oriented FAST and Rotated BRIEF (ORB)

Oriented FAST and Rotated BRIEF (ORB) method from OpenCV Library can be used to detect and extract the value of keypoints on leaf [6]. This method used as a comparison that might be able to provide better results. OpenCV or Open Computer Vision is an Application Programming Interface (API) Library that used in digital image processing. OpenCV library has many functions for digital image processing, such as improvement of image quality, image feature extraction, and also image recognition. ORB use FAST method for keypoints detection and BRIEF method for feature extraction. OpenCV library can improve the performance while doing the processes of digital image processing on android devices [7-8].

Keypoint detection is a process that uses the Library OpenCV functions to make the detection of certain corners on images involving histogram of gradient calculation. The method used to detect keypoints is ORB Detector. This method is development of the FAST method which is the quickest method to detect keypoints in OpenCV. FAST can resolve detection of change in image scale by using pyramid scheme, but unfortunately have not been able to resolve with the orientation of the image. FAST development is used to resolve the rotation called the Oriented FAST by using the intensity centroid [6]. The first moment of an image is determined by the following formula:

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y) \qquad (5)$$

*(p + q)* is the ordo of the moment. *X* and *y* are pixel coordinates, while *(x, y)* is the pixel intensity. Then moment is used to find the centroid by the following formula:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right) \qquad (6)$$

Then a vector construct from corner's center (*O*) to the

centroid (*OC*). The orientation formula then simply is:

$$\theta = atan2(m_{01}, m_{10}) \qquad (7)$$

*atan2* is the quadrant-aware version of *arctan*. The following are some examples of program code when using ORB Detector to detect keypoint:

```
FeatureDetector detector =
FeatureDetector.create(FeatureDetector.ORB);
MatOfKeyPoint keypoints = new MatOfKeyPoint();
detector.detect(ImageSource, keypoints);
```

Feature extraction is performed for keypoint features using ORB descriptor method from OpenCV Library. ORB Descriptor is the development of BRIEF method in OpenCV to detect image changes rotation [6]. ORB Descriptor performed by computation the value of keypoint vectors that have been detected. The following are some examples of program code when using ORB Descriptor:

```
DescriptorExtractor extractor =
DescriptorExtractor.create (DescriptorExtractor.ORB);
Mat features = new Mat();
extractor.compute(ImageMat, keypoints, features);
```

### G. City Block Distance

City Block Distance is also call as Manhattan Distance / Boxar Distance / Absolute Value Distance. City Block Distance method computes the absolute value of the two difference vectors [9].

$$d_{ij} = \sum_{k=1}^{n} |x_{ik} - x_{jk}| \qquad (8)$$

The match image is determined by the shortest distance.

### H. Brute-Force Hamming

Brute-Force Hamming is also present on the OpenCV Library. This method used to compute the value that obtained from ORB Descriptor [5]. Same as City Block Distance, Brute-Force Hamming used the shortest distance to determine the match image.

### III. PROPOSED ALGORITHM

Herbs recognition research was done using 10 types of herbs with 10 samples for each type. This study uses two kinds of general description, that is registration and recognition.
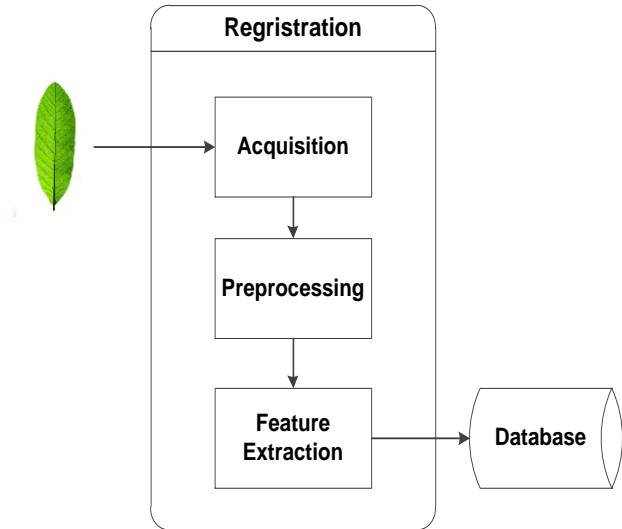


Fig. 1. General description of the registration system.

Fig. 1 shows registration process of the reference image features to database. The first phase starts with acquisition leaves image as a reference. Then image have some digital image processing on Preprocessing phase, so that the image is ready to be extracted as value. Feature Extraction phase is process to convert image to be numeric value and stored it in database. Herb Registration is done by storing 4 features value from 4 images as reference images for one type of herb.
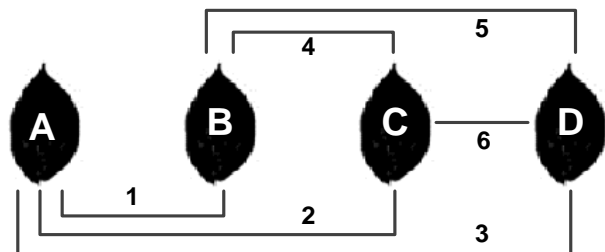


Fig. 2. Calculation of threshold.

The registration process calculated threshold value by using these four features that are stored in the database. Fig. 2 shows the calculation of threshold is done by calculating the average distance between leaf A - B, A - C, A - D, B - C, B - D, and C - D. The distance is then summed and divided by 6, so that obtain threshold value for reference image.

After the registration process completed, then the recognition process of leaves can be done. Beginning phases of the recognition process is same as registration process, until the feature extraction phase. It is aimed to obtain the image features in the same way.
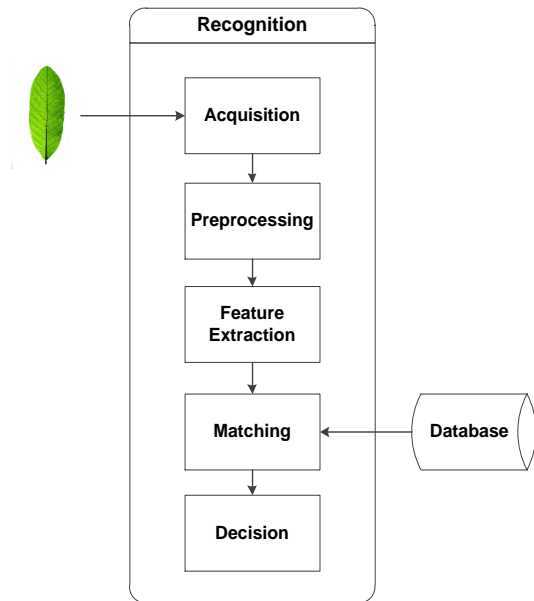
Fig. 3. General description of the recognition system.

Fig. 3 shows image recognition process of leaves. The recognition phase is the process of matching test image feature with reference image features that contained in database. The matching process is done by calculating the shortest distance between test image features with all the reference image features that exist in database. After obtained the shortest distance, the process continued on Decision phase. In this phase, the distance will be compared to threshold of reference image. If the distance is less than or equal to threshold then image is match, whereas if distance exceeds the threshold value, image does not match.

### A. Image Acquisition

Image acquisition is a step in collecting leaf images which is used as reference image in registration process and test image in recognition process. Image acquisition is done by taking leaf image using camera on Android Smartphone. The leaf images capture above the solid background, such as white to minimize the noise [4].

### B. Preprocessing

Preprocessing is phases in digital image processing to increase image quality and process it for ready to extract as value. There are some processes used in Preprocessing.

The first step is to Cropping and Resizing image. Cropping is the process of crop images into 1 x 1 ratio or square. This is done to avoid changes in shape of herb images or images stretching. Resizing is a process to change the size of image. Leaf image is resized to 150 x 150 pixels. This is intended to accelerate the preprocessing.

Then image converted to Grayscale to get one type of color value between 0 to 255. After that, noise on the image reduced by using Gaussian Blur.

Shape feature of leaf obtained from Binarization process of image. Whereas veins feature obtained by using Canny Edge Detection. The other feature that obtained by using OpenCV Library is keypoints. This

feature obtained from ORB Detector method (Oriented FAST).

### C. Feature Extraction

Feature extraction is a phase in changing the image into value that can be calculated by the hardware. Feature extraction data is set of numbers that will be used in performing the matching values by calculating the distance between image features.

Shape and veins features are using Invariant Moment method to extract the feature. This method will obtain 7 moments. The value of 7 Moment will be stored in the database while the registration process. The registration process use 4 reference images for each type of herbs.

Keypoints feature are extracted by using ORB Descriptor (Rotated BRIEF). ORB Descriptor performed by computation the value of keypoints vectors that have been detected. Then vector stored in the database as feature value of the image. Storage performed by using Json to change value of vector into string value, so it is easily stored in a database.

### D. Matching

Matching phase is process of doing the matching on test images with reference images by calculating the distance of feature extraction value. The matching process of shape and veins performed by using City Block Distance between test images with all the features of the image stored in database. Keypoints feature are computed by using Brute-Force Hamming. The shortest distance will be selected and then matched with the threshold.

### E. Decision

When the smallest value obtained from the calculation of City Block Distance and Brute-Force Hamming, then system check the value is less than or equal to the threshold multiplied by α (alpha). Alpha is a variable that is global to the entire reference image. If the features of the image is smaller or equal to the threshold multiplied by α, then that image declared match with the reference image, whereas if it is larger, then the test images declared not match with the reference image contained in the database.

## IV. EXPERIMENT SIMULATION AND RESULT ANALYSIS

Herbs Recognition Application was made by using OpenCV Library, so user need to install OpenCV Manager on their Android device. OpenCV Manager is free to download at Google Play Store. As was explained previously, there are two main processes in this application, that is the registration and recognition.

This application must have herbs data for recognition. The data that registered in this application is name, information about benefits and ways to use, and leaf features that are stored in the database, while the leaf images are stored on the *sdcard*.

Table 1 shows the list of herbs that used on this experiment.

Table 1. List of herbs

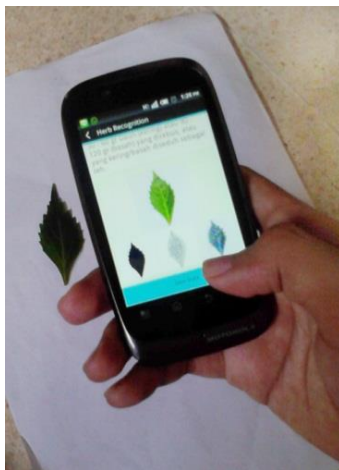| No. | Herbs Name | Samples Total |
|-----|-----------|---------------|
| 1 | Dadap Serap | 10 |
| 2 | Guava | 10 |
| 3 | Hibiscus | 10 |
| 4 | Kumis Kucing | 10 |
| 5 | Mengkudu | 10 |
| 6 | Papaya | 10 |
| 7 | Salam | 10 |
| 8 | Sirih | 10 |
| 9 | Cassava | 10 |
| 10 | Waru | 10 |



Fig. 4. Registration process.

Fig. 4 shows user used the application for register a leaf as herb. In this process, user entry 4 images as reference image and stored it in database.



Fig. 5. Herbs data.

Fig. 5 is the screenshot that shows herbs data that have been registered in application. The details of the process on Fig. 4 can be seen on Fig. 6.

Recognition process on this application used the leaf image of herbs. User can take that image from *sdcard* or capture with smartphone camera, then the image will processing to recognize.

Fig. 7 (a) is the screenshot of application while recognize a leaf. This process will show the herb data, if that image matched with images feature that exist in application database. Fig. 7 (b) shows the detail of recognition process.
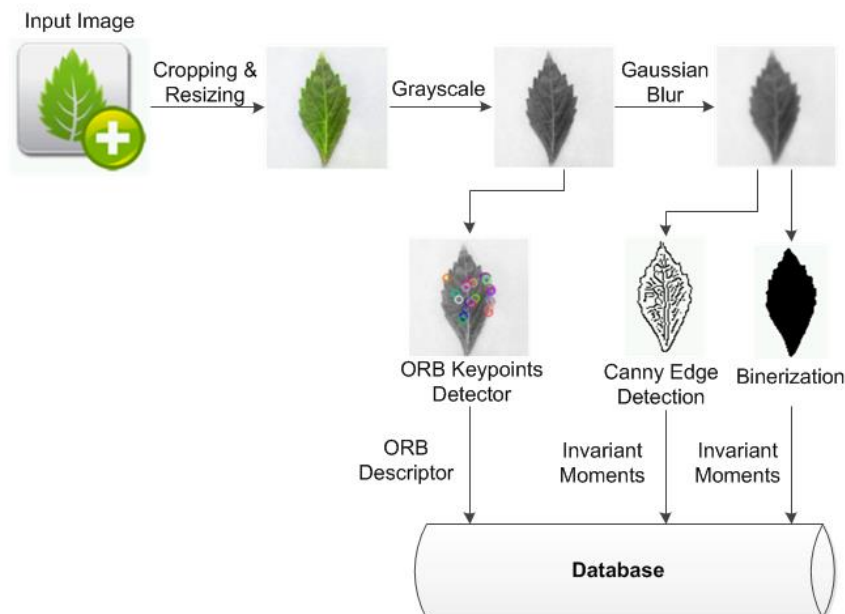


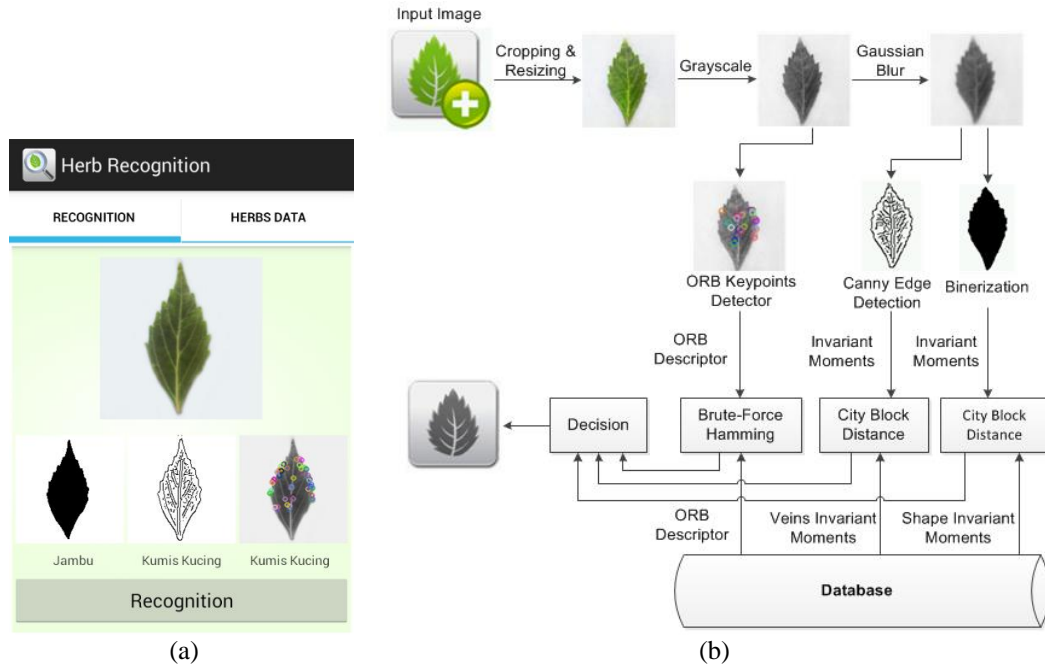Fig. 6. Details process of registration.

Fig. 7. (a) Recognition process, (b) Details process of recognition.

Analysis of this application is using 10 test images for 10 types of herbs, so that have 100 test images that capture by different android smartphone. First analysis is performed on veins recognition to get the best threshold for Canny Edge Detection. This analysis used threshold value between 10 to 100.
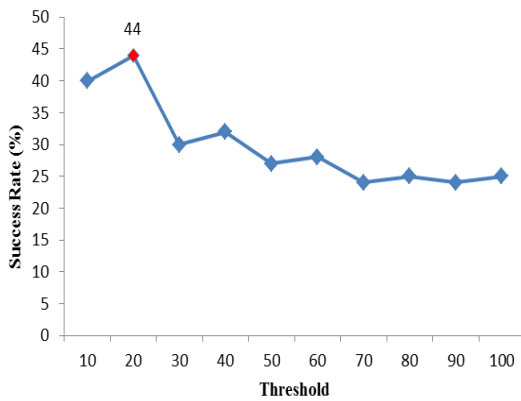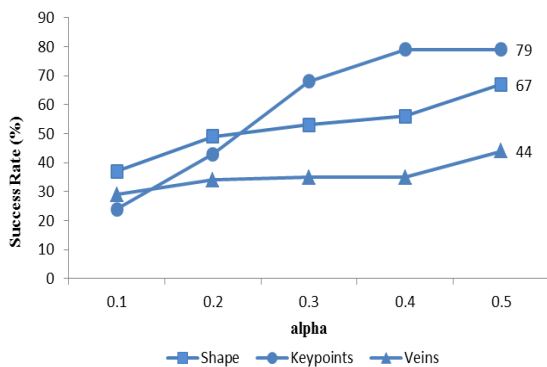
Fig. 8 shows the best threshold for Canny Edge Detection for veins recognition is 20. This threshold obtained 41% of success rate. The next analysis is performed on recognition the feature of shape, veins, and keypoints by using alpha (the multiplier threshold as global variable) between 0.1 and 0.5.

Fig. 9 shows the match rate when recognition for each feature. The highest success rate of recognition is about 79% obtained by keypoints feature when using alpha 0.4. Whereas the shape and veins recognition obtained results respectively 67% and 44%. The result shows the use of keypoints feature is highly effective in herbs recognition.



Fig. 8. Experiment result to find the best Canny threshold.



Fig. 9. Experiment result when using alpha.

Table 2. Experiment Result of Success Rate

| Herbs Name | Success Rate | | |
|---|---|---|---|
| | Keypoints | Shape | Veins |
| Dadap Serap | 8 | 10 | 4 |
| Guava | 9 | 6 | 4 |
| Hibiscus | 9 | 6 | 3 |
| Kumis Kucing | 8 | 3 | 4 |
| Mengkudu | 9 | 7 | 7 |
| Papaya | 9 | 3 | 3 |
| Salam | 5 | 9 | 3 |
| Sirih | 7 | 8 | 4 |
| Cassava | 9 | 7 | 5 |
| Waru | 6 | 8 | 7 |
| Percentage (%) | 79 | 67 | 44 |

Data in Table 2 shows the detail of success rate on recognition process. In this table, *Salam* and *Sirih* give bad result for keypoints recognition, that is also happened with veins recognition. This is because the glossy leaf surface, so there is a lot of reflected light that become

noise. Shape recognition give bad result on the leaves that have uneven edges like *Kumis Kucing* and *Papaya*.

## V. CONCLUSION

Based on the experiments, herbs recognition with use keypoints feature obtained 79% of success rate. That is the best match rate if compared with shape and veins feature that only obtained 67 and 44%. This result is influenced by glossy leaf surface, so there is a lot of reflected light that become noise. The match rate is also influenced by the image when captured by the smartphone camera. Poor smartphone camera will give bad result, it because the effect of brightness, contrast, sharpness of the image that can obtain more noise. Based on that result, ORB is the best method to recognize the leaf image. ORB can minimize the effect of noise in the leaf images, that is better than Invariant Moment method.

## REFERENCES

[1]  K. B. Lee and K. S. Hong. An Implementation of Leaf Recognition System using Leaf Vein and Shape. *International Journal of Bio-Science and Bio-Technology.* 2013; 5(2): 57-66.

[2]  A. Bhardwaj, M. Kaur, A. Kumar. Recognition of plants by Leaf Image using Moment Invariant and Texture Analysis. International Journal of Innovation and Applied Studies. 2013; 3(1): 237-248.

[3]  A. Kadir, L. E. Nugroho, A. Susanto, P. I. Santosa. Experiments Of Zernike Moments For Leaf Identification. *JATIT Journal of Theoretical and Applied Information Technology.* 2012; 41(1): 82-93.

[4]  Milan S. Image-based Recognition of Plants. Bachelor Thesis. Prague Postgraduate Czech Technical University. 2012.

[5]  Z. Huang and J. Leng. Analysis of Hu's Moment Invariants on Image Scaling and Rotation. *International Conference on Computer Engineering and Technology (ICCET).* 2010; 7(2): 476-480.

[6]  E. Rublee, V. Rabaud, K. Konolige, G. Bradski. ORB: An Efficient Alternative to SIFT or SURF. IEEE International Conference on Computer Vision. 2011: 2564-2571.

[7]  A.V.Kulkarni, J.S.Jagtap, V.K.Harpale. Object Recognition with ORB and its Implementation on FPGA. International Journal of Advanced Computer Research. 2013; 3(3): 164-169.

[8]  A. Anuar, K. M. Saipullah, N. A. Ismail, Y. Soo. OpenCV Based Real-Time Video Processing Using Android Smartphone. International Journal of Computer Technology and Electronics Engineering (IJCTEE). 2012; 1(3): 58-63.

[9]  Darma Putra. Pengolahan Citra Digital. Yogyakarta: CV. Andi Offset. 2010.

[10] Darma Putra. Binerisasi Citra Tangan dengan Metode Otsu. Teknologi Elektro. 2004; 3(2).

**Author's Profiles**

**I Wayan Agus Suryawibawa**, Male, is a student at Department of Information Technology, Udayana University, Bali and currently working toward the Bachelor of Information Technology. His research interests are digital image processing, system anaysis, and mobile programming.

**I Ketut Gede Darma Putra**, Male, is a lecturer in Department of Electrical Engineering and Information Technology, Udayana University Bali, Indonesia. He received his S.Kom degree in Informatics Engineering from Institute of Sepuluh November Technology Surabaya, Indonesia on 1997. He received his Master Degree on Informatics and Computer Engineering from Electrical Engineering Department, Gadjah Mada University, Indonesia on 2000 and achieved his Doctorate Degree on Informatics and Computer Engineering from Electrical Engineering Department, Gadjah Mada University, Indonesia on 2006. His research interests are Biometrics, Image Processing, Data Mining, and Soft computing.

**Ni Kadek Ayu Wirdiani**, Female, is a lecturer in Department of Information Technology, Udayana University Bali, Indonesia. She received her S.T degree in Communication Electronics, Electrical Engineering Department from Udayana University, Indonesia on 2004. She received her M.T degree in Information Technology Management, Electrical Engineering Department from Udayana University, Indonesia on 2011. Her research interests are Biometrics and Image Processing.