*International Journal of Engineering and Science Applications*
ISSN 2406-9833
IJEScA

# Software as a Service (SaaS) based Machine Learning for Digital Image Recognition

Syafaruddin[1)] and Andi Lukman[2)]
[1)]Department of Electrical Engineering of Universitas Hasanuddin, 90245 Tamalanrea-Makassar, Indonesia
e-mail: *syafaruddin@unhas.ac.id
[2)]Sekolah Tinggi Informatika dan Multimedia (STIMED) Nusa Palapa Makassar, Indonesia
e-mail: uke@stimednp.ac.id

**ABSTRACT**

Nowadays, the machine learning method and algorithms are varied with different capabilities and tasks. It is almost impossible to understand the algorithm in detail or to determine which method is appropriate for certain applications. For these reasons, the application system operates in cloud system according to Software as a Service (SaaS) method is proposed; therefore the system is accessible for multiusers with open source data mining. Wide range of algorithms in Waikato Environment for Knowledge Analysis (WEKA) machine learning are considered, for instance support vector machine (*SVM), K-Nearest Neighbor (KNN), Naïve Bayes*, *C4.5 Decision Tree*, *Logistic Regression* and *Random Forest* methods. The application facilitates the image recognition researcher of using SaaS method, due to the flexibility in purpose of research, such as search in algorithm analysis, optimal training results in digital image recognition and the implementation of application system. In addition, the system application can be accessed anytime without installation process, but through web browsing systems.

*Keywords: Saas, machine learning algorithm, digital image, and binary classification.*

Article history: Received 4 October 2014, last received in revised 20 November 2014

## 1. INTRODUCTION

Nowadays, digital image recognition has become one of the popular research themes in computer vision, especially pattern recognition and machine learning tasks. In machine learning task, certain algorithms based on mathematical theory and statistic are implemented to facilitate computers to recognize digital image patterns. The research in digital image processing may be helpful for the object classification in the fields of medical, marine and business. In this research, the application system for digital image recognition task is designed by considering the S*oftware as a Service* (SaaS) based cloud computing technology. The method may help the users for the online application since the internet connection exists without needing software installation [1]. The SaaS method based on the application systems can be used by multi-clients at the same time of utilization; therefore it provides high access flexibility for the users.

There were so many potential previous researches considering different digital image recognition tasks. For instance object (book) digital image recognition task [2], mammographic digital image for breast cancer [3], metastatic lymph glands for lung cancer [4]. However, the recognition tasks are individually depending on certain designed algorithms based machine learning methods, where the optimal parameters are considered according to the research aims. To overcome such problems, the general application system is proposed using collective machine learning algorithms, such as that one is available in Waikato Environment for Knowledge Analysis (WEKA) where the users are flexible to select the appropriate algorithms and preference the optimal parameters according to their target recognition tasks [5].

One of the common problems in digital image recognition research is the developed application system is only working properly on the research object; therefore for other tasks, the researchers need to build a new algorithms. It is not so convenient in the end because it needs high skills in computer programming area. In fact, the users and digital image researchers are coming from different background knowledge which the majority of them are not from computer science. On the other hand, if researchers are able to develop algorithms for digital image processing task, it cannot be utilized in multi-users scenarios and it ensures software installation process. This is the background idea why WEKA is coming with collective popular machine learning algorithms for digital image processing in default form, so that the users may modify the algorithms according to their preference.

For the open access utilization, the SaaS technology is implemented combined with application system that is available in WEKA collective machine learning algorithms. Therefore, the research aims is to develop application system for digital image recognition task taking the machine learning algorithm in WEKA operates in cloud system according to Software as a Service (SaaS) method. The outcome of this research is mainly addressed to help digital image researchers to select their proper algorithms for the research without spending time to develop computer algorithm in free access and flexible. The detail information about this research is provided as follows.

## 2. CONFIGURATION OF PROPOSED SYSTEM

The proposed research utilizes hardware, such as personal computer (PC) base Core i3 and software components, for instance Java 2 SDK (J2SE), Eclipse for Java EE developer, and Google Plugin for Eclipse (GPE), Google App Engine SDK, Google Web Toolkit (GWT) and internet browser. The reasons for adopting these components are the minimum processor requirement in designing the application system and the necessity of fast computational access when the software as a service (SaaS) based application system is deployed to Google app engine. The image data used in this research for training and validation process is taken from Caltech 101; i.e object collection of digital image in PNG type developed by computational vision research of California Institute of Technology [6]. Meanwhile, the input-output documentation system technique consists of application system based unified modeling language (UML) with learning process and digital image recognition as outputs.

The application systems of machine learning for digital image recognition is developed using algorithms collection in WEKA machine learning based data mining with open source. The application system operates in cloud system according to SaaS method; therefore the system is accessible for multiusers. The design architecture of SaaS based application system for digital image recognition is shown in Figure 1. The application may help the digital image research due to the flexibility in purpose of research, such as search in algorithm analysis, optimal training results in digital image recognition and the implementation of application system. Wide range of algorithm may be considered, for instance, support vector machine (SVM), K-Nearest Neighbor (KNN), Naïve Bayes, C4.5 Decision Tree, Logistic Regression and Random Forest methods. In addition, the application can be accessed anytime without installation process, but through web browsing systems.

In addition, Fig. 1 explains the mechanisms more clearly that there are 3 actors interact in the system, i.e digital image researchers, *Google app engine* (GAE) and WEKA *Machine Learning*. The digital image researchers are the users of application system, while GAE is the machine that processes any functions defined in the system. On the other hand, the WEKA machine learning provides machine learning algorithms required in the designed system. The successful implementation of designed system is assessed with black box testing method with sorts of criteria, such as login process, downloading data image for training process, saving and displaying training and validation data set, data conversion into ARFF format, implementation different algorithm, digital image recognition results and multiusers assessment. The success of digital image recognition process is measured by ratio in percentage between the numbers of recognized digital image to the total number of digital image taken in validation step.

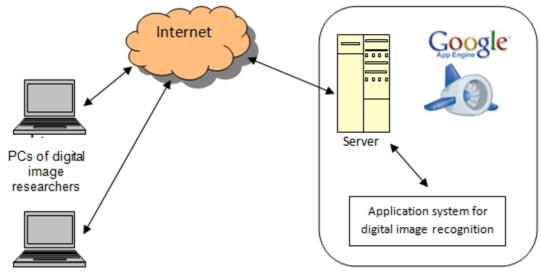


Figure 1. Design architecture of SaaS based digital image recognition

## 3. Simulation Results

### A. *Pre-processing Data*

Pre-processing data is required before the training process, testing and validation in order to obtain data set that is suitable for algorithms in WEKA classification. The pre-processing data consists of downloading digital image, normalization, bag of words (BoW), and data conversion to ARFF format before program deployment for system implementation. These stages are explained as follows.

The digital image is taken from the computer client and downloaded to server and stored in blobstore. Beside the image file, the url address is also stored including the unique key that can be used to access the image file. The process of storing and retrieving data in blobstore is performed by Google App Engine for Java (GAE/J) program utilizing import of API GAE Blobstore. Next, the digital image is saved for waiting the further process. The server reads the digital image in the blobstore according to the url address and its unique key. The normalization process is needed to transform the image size to 64X64 pixel in order to obtain the uniform image size that makes easy and convenient algorithms process. The normalization process can be performed using GAE/J programming utilizing API GAE Images. After the normalization, the digital image bag of word (BoW) is quantified.

The bag of word is the division of normalized image into 16 parts with maintaining each pixel position. The first step is to take pixel value which consists of RGB values of digital image and to calculate the mean value using equation (XX). The process is pretty similar to the gray scaling process and the mean value is stored in 2 dimensional arrays in order to keep the pixel value. As results, the matrix of 64x64 from (0,0) to (63,63) contains value of 0 - 255 is obtained.

The second step is to divide the matrix into 16 parts; therefore each part consists of 16x16 pixel with fixed position is maintained. The value of BoW is determined from these 16 matrixes, denoted by BoW01 to BoW16. The position of the first row of matrix BoW01 is from the pixel (0,0) to pixel (15,15), BoW02 is between pixel (0,16) and pixel (15,31) and so on until BoW16 which is from pixel (48,48) to pixel (63,63). The last stage is to calculate the mean of pixel of matrix from BoW01 to BoW16, and then saved into field according to the matrix name in datastore including the classification of field relation and field target. The normalization process is complete; however, the algorithms in WEKA machine learning cannot yet process it unless the data set has been converted to ARFF format.

The data conversion into ARFF format is merely done to fulfill the requirement of algorithms in WEKA machine learning. The process starts with creating header part of ARFF. Every class is defined using library *weka.core. FastVector* with method of addElement that contains yes and no answers. These library and method are also used to fill each attribute and class utilizing library *weka.core.Attribute* from BoW01 to BoW16. Then, it needs to input the header data set using library *weka.core.Instances*. The next process is to make data part by taking the data set that has already been stored in datastore GAE using query data technique according to input relation. The query results are sequentially red by personal computer (PC) and collected in dataset using library *weka.core.Instance*. This process is continuous until all data query is completely red. After header part and data are completely filled, the data set is ready to be processed using algorithms in WEKA classification including training, validation and digital image recognition.

### *B. Deployment diagram and Multiusers technique for Software as s Service (SaaS) method*

After the application system has been built and coded using programming language, the system is the deployed for the real implementation. The deployment diagram describes the configuration of components during installation process of application system. Also, it may point to the physical location of system, display the software parts amongst the hardware structures. In addition, it is suitable to modeling the client/server systems. The developed deployment diagram is shown in Figure 2. In this diagram, the multiusers (clients) can access the application system through internet browsing. The application system appears in java script form that has been converted by *Google Web Toolkit*. The clients request service to server of *Google App Engine* using *Remote Procedure Call* (RPC) technique; then server carries out this request by *J2EE Server*. The image data will be stored by *Blobstore*, while the data set is stored by *Datastore*. After the process of client's request is complete, server will return the data process results to the clients who appear on the *web browser*.

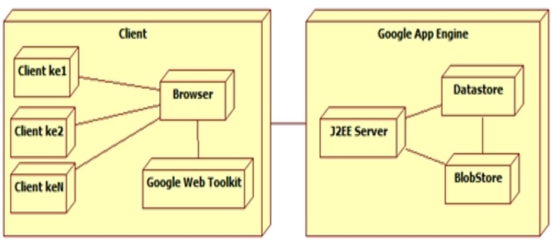


Figure 2. Deployment Diagram of application system

The deployment diagram shown the physical condition of application systems to be ready in online implementation using web browser and can be accessed by multiusers. It may also being considered as the interface technology between the client package and server package. The diagram indicates that the Software as a Service (SaaS) based application system is useful and reliable for the multi-clients access with *Google App Engine*. The design application system can be accessed in web browser address of http://recognizeimage.appspot.com.

The client package requests service to the server package through the *interface ImageMLService* and *ImageMLServiceAsync*. This kind of request utilizes *Remote Procedure Call* (RPC) technology. The RPC implements *asynchronous* method, where the clients do not need to wait long till the server responds it. Otherwise, the browser will hang up during busy server. In this case, the clients just send request to server then disconnect until the data process from server is received. It is an automatic reply once the server finishing the data processing. The speed and reliability of client data request are depending on the Google server that is globally connected. This capability allows the machine learning application can be accessed by multi-users with reasonable speed.

### *C. Results of Digital Image Recognition*

The designed computer system application based Software as a Service (SaaS) is used for different recognition tasks, for instance face, animal (butterfly), plant (sun flower),

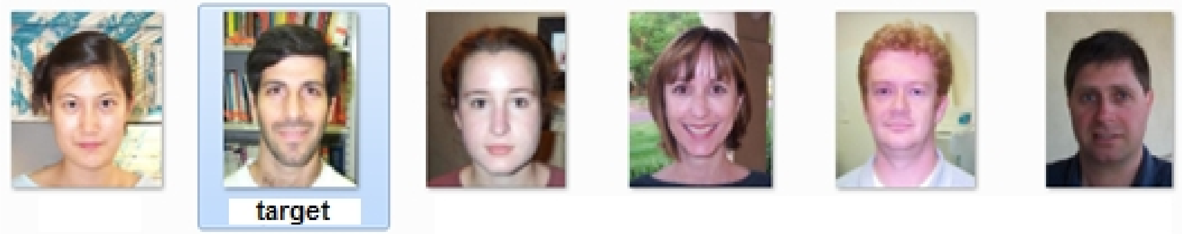


Figure 3. Digital image for face recognition task.

object (cup) recognitions and tested with algorithms of support vector machine (*SVM), K-Nearest Neighbor*

*(KNN), Naïve Bayes*, *C4.5 Decision Tree*, *Logistic Regression* and *Random Forest* methods. The results of digital image recognition are explained as follows.

In the case face recognition, we set the target picture considered as relation. For training and validation process, there are 25 pictures related to the target picture, called class data (yes). Meanwhile, other face pictures are varied of 5 pictures each, called class data (no). Therefore, there are 50 digital images of collective face picture as shown in Figure 3. The number of images for training data set is 60% of total images with normally distributed by 15 images from class 'yes' and 15 images from class 'no'. The remaining percentage of images is used for validation which consists of 10 images from class 'yes' and 10 images from class 'no'.

Using the list of algorithms for digital image recognition, the Support Vector Machine (SVM) may recognize the target face of 65% with the successful ratio of 50% for digital image of class 'yes' and 80% for class 'no'. Better recognition result of 80% is obtained using Naïve Bayes method with successful ratio of 80% for both image of class 'yes' and 'no'. Similar to Naïve Bayes method, the C4.5 Decision Tree may recognize the target digital image with 80%; composed of 70% for digital image of class 'yes' and 90% for class 'no'. On the other hand, the K-Nearest Neighbor method and random forests methods have similar performance to SVM method with 65% face recognition capability. The best method for face recognition using our proposed method is with Logistic Regression method with 95% successful rate; consists of 90% for digital image of class 'yes' and 100% for class 'no'; therefore the algorithms is recommended for face recognition task. The recapitulation of successful rate is shown in Table 1.

Table 1. Successful rate of machine learning algorithms for face recognition

| Machine Learning Algorithms | Class *'Yes'* | | Class *'No'* | | Successful percentage |
|---|---|---|---|---|---|
| | Success | Not Success | Success | Not Success | |
| Support Vector Machine | 5 | 5 | 8 | 2 | 65 % |
| Naïve Bayes | 8 | 2 | 8 | 2 | 80 % |
| C4.5 Decision Tree | 7 | 3 | 9 | 1 | 80 % |
| K-Nearest Neighbor | 6 | 4 | 7 | 3 | 65 % |
| Logistic Regression | 9 | 1 | 10 | 0 | 95 % |
| Random Forest | 9 | 1 | 4 | 6 | 65 % |

In the case animal image recognition, a butterfly image is set as target relation. There are 25 digital images categorized in class ‘yes, while other animal images, such as ant, dragon fly, bird, pigeon, rooster and starfish are categorized class ‘no’ with 5 digital images each. The digital image object for animal recognition is shown in Fig. 4.

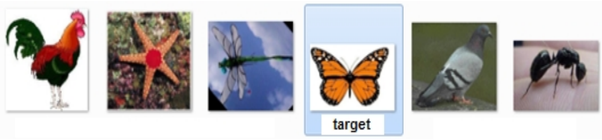


Figure 4. Collection of digital image for animal recognition task

Similar to the face recognition previously, the percentage for training data set is 60% of total digital image, with normal distribution of 15 images categorized class ‘yes’ and 15 images categorized class ‘no’. Meanwhile, 40% of total digital image is used for validation with 10 categorized class ‘yes’ and 10 images categorized class ‘no’. After testing all machine learning algorithms, the random forest method is claimed better than other methods for animal (butterfly) digital image recognition task.

The successful rate of algorithms for butterfly image recognition is shown in Table 2. Similar approach in selection digital image data for training and validation in the previous face and butterfly images recognition task is considered for plant (sun flower) recognition task. There are 25 digital images categorized in class ‘yes’ and other plant flowers categorized class ‘no’ such as bonsai tree, joshua tree, lotus flower, strawberry and waterlilly with 5 digital images each. The digital object for plant recognition is shown in Figure 5.

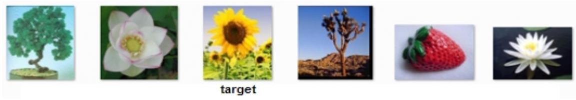


Figure 5. Digital object for plant recognition

From database, there are total of 50 digital images considered for training and validation process. Again, the percentage of training data set and validation are 60% and 40%, respectively. The implementation of machine learning algorithms show that method of C4.5 Decision Tree is better than other methods with the successful rate of 75%; therefore this method is recommended for plant digital image recognition. The recapitulation of successful rate of algorithms is shown in Table 3.

The last recognition task is using the machine learning algorithms for object (cup) digital image. All approaches from the previous tasks are similar, except the digital image object collection for training and validation process.

Table 2. The successful rate of algorithms for butterfly digital image recognition task

| Machine Learning Algorithms | Class ‘*Yes*’ | | Class ‘*No*’ | | Successful percentage |
|---|---|---|---|---|---|
| | Success | Not Success | Success | Not Success | |
| Support Vector Machine | 3 | 7 | 10 | 0 | 65% |
| Naïve Bayes | 2 | 8 | 7 | 3 | 45% |
| C4.5 Decision Tree | 6 | 4 | 6 | 4 | 60% |
| K-Nearest Neighbor | 6 | 4 | 7 | 3 | 65% |
| Logistic Regression | 5 | 5 | 8 | 2 | 65% |
| Random Forest | 6 | 4 | 8 | 2 | 70% |

Table 3. The successful rate of algorithms for sunrise flower digital image recognition task

| Machine Learning Algorithms | Class '*Yes*' | | Class '*No*' | | Successful percentage |
|---|---|---|---|---|---|
| | Success | Not Success | Success | Not Success | |
| Support Vector Machine | 2 | 8 | 10 | 0 | 60% |
| Naïve Bayes | 5 | 5 | 5 | 5 | 50% |
| C4.5 Decision Tree | 9 | 1 | 6 | 4 | 75% |
| K-Nearest Neighbor | 5 | 5 | 7 | 3 | 60% |
| Logistic Regression | 9 | 1 | 5 | 5 | 70% |
| Random Forest | 8 | 2 | 5 | 5 | 65% |

The digital image objects categorized in class 'no' are varied, such as scissors, stapler, chair, lamp and umbrella. The digital image object collection is shown in Figure 6.

The remaining things like the percentage of digital image for training and validation process is the same.

After testing all the machine learning algorithms, the Naïve Bayes algorithm shows better object recognition performance compared to other methods. The recapitulation of successful rate of algorithms for object recognition task is shown in Table 4.

However, the logistic regression method seems superior to other methods for all target recognition tasks in this research. The successful rank of algorithms for digital image recognitions task is shown in Table 5. One of the reasons in the high rank successful of *logistic regression* algorithm compared to other method is the original design of logistic *regression* algorithm for binary classification. Nevertheless, all algorithms requires digital image consistency and quality for better digital image recognition results.

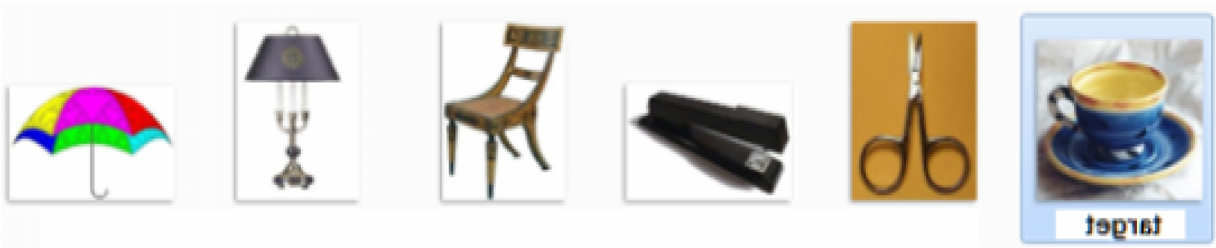

Figure 6. Digital object for object recognition

Table 4. The successful rate of algorithms for object (cup) digital image recognition task

| Machine Learning Algorithms | Class '*Yes*' | | Class '*No*' | | Successful percentage |
|---|---|---|---|---|---|
| | Success | Not Success | Success | Not Success | |
| Support Vector Machine | 1 | 3 | 10 | 0 | 55% |
| Naïve Bayes | 7 | 3 | 8 | 2 | 75% |
| C4.5 Decision Tree | 6 | 4 | 8 | 2 | 70% |
| K-Nearest Neighbor | 4 | 6 | 9 | 1 | 65% |
| Logistic Regression | 7 | 3 | 3 | 7 | 50% |
| Random Forest | 5 | 5 | 9 | 1 | 70% |

Table 5. The successful rank of algorithms digital image recognition task

| Objects | Algorithms | Class '*Yes*' | | Class '*No*' | | Percentage |
|---|---|---|---|---|---|---|
| | | Success | Not Success | Success | Not Success | |
| Face | Logistic Regression | 9 | 1 | 10 | 0 | 95% |
| Butterfly | Random Forest | 6 | 4 | 8 | 2 | 70% |
| Sun-flower | C4.5 Decision Tree | 9 | 1 | 6 | 4 | 75% |
| Cup | Naïve Bayes | 7 | 3 | 8 | 2 | 75% |

## 4. DISCUSSION

From the simulation results, the proposed technology of Software as a Service (Saas) brings the machine learning algorithms have capability and flexibility in different digital image recognition task. In developing the SaaS technique in *Google App Engine* (GAE), the researchers need to think the type of request in client side for the server process.

It is due to the limitation access of client for data processing with only HTML or Javascript facility available in web browser. The process is mainly concentrated in server location. Nevertheless, this is a kind of advantage of users using of SaaS technique because the digital image recognition task can be processed with only using web browser.

Although the GAE limits the programmers of using native library that is commonly used in Java programming, the GAE may only receive the low level library. In GAE, there is a term of *white list library/API*; the recommended one for GAE. One of the popular library in Java programming called imageIO with the capabilities of image processing, such as image retrieving, image manipulation, pixel reading and manipulation. However, this library is not part of white list of GAE. On the other hand the API is which popular for image processing in general cannot be used as well because API utilizes the imageIO library. That is way; the GAE has its own API in digital image manipulation including image sizing, rotation and flip. Nevertheless, the capability is still being improved especially for pixel image exploration; while the proposed computer system application requires pixel image exploration for developing data set and image processing. To overcome this problem, the imageIO library is replaced with PNGJ library.

The PNGJ library has similar function with imageIO library but only limited to image file with PNG extension. Fortunately, the PNG file is generally used for web based image application; therefore detailed exploration of PNG file image for GAE utilization run in web browser.

Another problem may rise beside the white list GAE as previously mentioned is the progress idea of designed program. Initially, the simplification program based desktop is performed. It runs successfully, in fact the program cannot be fully implemented in servlet GAE programming in Eclipse utilizing GAE plugging. In this respect, the program works in *Eclipse* with GAE plugging will not guarantee it runs in real GAE (online). Some module and function may not work properly. In this case, the black box testing is important to determine which module/function works or not before the online testing is performed.

The proposed system application is intended for the digital image recognition task. As shown in simulation results, the machine learning algorithms may be better on especial task. It is found that 2 factors affecting the successful of machine learning algorithms for each task. The first one is the pre-processing stage where the normalization and bag of word (BOW) are taken into account. In this research, the image size of 64x64 is changed to BoW16 to obtain the training data set and validation data. In machine learning process, different filtering method may influence the successful of machine learning algorithms. The filtering method effects to the level of image consistency and quality. The second factor, the machine learning used in this application is still in default setting. In fact, these algorithms still open being expanded, for instance the algorithm of *K-Nearest Neighbor*, where K can be 2, 3 and so on instead of using default value of 1.

The proposed application system is limited to the image data with PNG *true color* and binary classification (yes or no). In future works, the research *may* extend to the multi-classification task instead of binary classification. The clustering problems may be also taken into account with additional learning algorithms. The filtering method also holds important roles for diversity algorithms and task applications.

## 5. Conclusion

This paper has presented the application systems of machine learning for digital image recognition task using algorithms collection in WEKA machine learning based data mining with open source. The application system operates in cloud system according to SaaS method; therefore the system is accessible for multiusers. Wide range of algorithm may be considered, for instance, support vector machine (*SVM)*, *K-Nearest Neighbor (KNN)*, *Naïve Bayes*, *C4.5 Decision Tree*, *Logistic Regression* and *Random Forest* methods.

The application may help the digital image researchers due to the flexibility in purpose of research, such as search in algorithm analysis, optimal training results in digital image recognition and the implementation of application system. In addition, the application can be accessed anytime without installation process, but through web browsing systems. In future works, the research may extend to the multi-classification task instead of binary classification. The clustering problems may be also taken into account with additional machine learning algorithms.

## References

[1] Allen, B., et.al, 2012, Software as a Service for Data Scientists, Communications of The ACM, Vol. 55, No. 2, pp. 81-88,.

[2] Lukman, A., 2012, Implementasi Pengolahan Citra dan Algoritma LVQ Untuk Pengenalan Pola Buku, Seminar Nasional Informatika (SNIf), pp.145-151, Medan.

[3] Hajare, P. S., &, V. V. Dixit, 2012, Breast Tissue Classification Using Gabor Filter, PCA and Support Vector Machine, International Journal of advancement in

electronics and computer engineering (IJAECE) Vol. 1, No. 4, pp. 116-119.

[4] Chao Li, et.al, 2012, Using the K-Nearest Neighbor Algorithm for the Classification of Lymph Node Metastasis in Gastric Cancer, Journal of Computational and Mathematical Methods in Medicine.

[5] Desai, A & S. Rai, 2012, Analysis of Machine Learning Algorithms using WEKA, International Journal of Computer Applications, No.13.

[6] http://www.vision.caltech.edu/Image_Datasets/Caltech101/, accessed on 20 September 2013.