

Multipatch-GLCM for Texture Feature Extraction on Classification of the Colon Histopathology Images using Deep Neural Network with GPU Acceleration

¹Toto Haryanto, ¹Adib Pratama, ¹Heru Suhartanto, ¹Aniati Murni, ²Kusmardi Kusmardi and ³Jan Pidanic

¹Faculty of Computer Science, Universitas Indonesia, Depok, Indonesia

²Department of Pathology Anatomy Faculty of Medicine, Universitas Indonesia, Jakarta, Indonesia

³Faculty of Electrical Engineering and Informatics, University of Pardubice, Studentska 95, 532 10 Pardubice, Czech Republic

Article history

Received: 13-12-2019

Revised: 01-02-2020

Accepted: 04-03-2020

Corresponding Author:

Toto Haryanto

Faculty of Computer Science,
Universitas Indonesia, Depok,
Indonesia

Email: toto.haryanto@ui.ac.id

Abstract: Cancer is one of the leading causes of death in the world. It is the main reason why research in this field becomes challenging. Not only for the pathologist but also from the view of a computer scientist. Hematoxylin and Eosin (H&E) images are the most common modalities used by the pathologist for cancer detection. The status of cancer with histopathology images can be classified based on the shape, morphology, intensity, and texture of the image. The use of full high-resolution histopathology images will take a longer time for the extraction of all information due to the huge amount of data. This study proposed advance texture extraction by multi-patch images pixel method with sliding windows that minimize loss of information in each pixel patch. We use texture feature Gray Level Co-Occurrence Matrix (GLCM) with a mean-shift filter as the data pre-processing of the images. The mean-shift filter is a low-pass filter technique that considers the surrounding pixels of the images. The proposed GLCM method is then trained using Deep Neural Networks (DNN) and compared to other classification techniques for benchmarking. For training, we use two hardware: NVIDIA GPU GTX-980 and TESLA K40c. According to the study, Deep Neural Network outperforms other classifiers with the highest accuracy and deviation standard 96.72 ± 0.48 for four cross-validations. The additional information is that training using Theano framework is faster than Tensorflow for both in GTX-980 and Tesla K40c.

Keywords: GLCM, Histopathology, Deep Neural Network, Multipatch

Introduction

Cancer is one of the leading diseases that cause high levels of mortality worldwide. The statistics of the National Center for Health Statistics presents in 2018 shows that more than 1.7 million new cancer cases existed and about six hundred thousand death occurred in the United States (US) (Siegel *et al.*, 2018). The development of new image processing techniques and Computer-Assisted Diagnosis (CAD) play the main role in the analysis of histopathology tissues from full images (Gurcan *et al.*, 2009) as the basic tools for analysis, segmentation, and automatic detection. Research about histopathology images is very

challenging due to the specific structure of the images that tend to misidentified or misclassified. After the whole image scanner for capturing histopathology data discovered, it is then possible to analyze images with high-resolution directly (Mukhopadhyay *et al.*, 2018). Nevertheless, the problem for histopathology images is that the cancer cell is indicated by the nucleus whereas images include an enormous number of the nucleus that must be analyzed. Therefore, it is necessary to divide the full image to high-resolution patches and analyze high-resolution patches one by one to locate accurate pixel location and extract the relevant information. The second problem is related to the size of full-size images that consume a lot of processing time to complete the

analysis. The whole process can be divided into three subsequent steps: (1) Texture Extraction, (2) Training and (3) Classification or Identification.

Features of texture will be the first information extracted from the images to get histopathology information. There are various techniques for extracting texture from the images. One of the techniques that have been widely used in many research fields is Gray Level Co-Occurrence Matrix (GLCM) proposed by Haralick *et al.* (1973). The idea of GLCM is focusing on the spatial relationship of the pixel (angle and distance) in the gray level images. GLCM is used as the base texture feature extraction method and combined with other features like shape, temporal and spatial features that have been successfully applied in Dynamic Contrast-Enhanced Magnetic Resonance Imaging (DCE-MRI) data (Banaie *et al.*, 2018). The research managed to achieve accuracy up to 98.87% after combining spatial and temporal features. However, the accuracy is lower than 98.87% when using only spatial or temporal features.

There are four contributions to this research. First, the feature extraction from a multi-patch pixel that is applied to whole histopathology images. Second, handling the loss of information by using 15 pixels of the window-size on sliding windows step. Third, the elimination of noise by using the mean moving filter (Comaniciu and Meer, 2002). The last, design of DNN architecture for training the models of classification. The DNN models are also compared to other classifiers such as K-Nearest Neighbors (KNN), Linear Discriminant Analysis (LDA) and Decision Tree.

This article is divided into six sections described below. The first section is an introduction including background, motivation purposes, and contribution of the research. The second section is related to research in histopathology texture analysis and classification. The third section explains the histopathology dataset used in this research and also the multi-patch GLCM feature extraction methodology. This section also explains the theory related to the applied method. The results and discussion are described in sections four and five. The last section is the conclusion of the research.

Related Works

Handcrafted Feature Extraction

The study of feature extraction using GLCM combined with the Gabor filter conducted by Ali *et al.* (2018) is called G2LCM. The G2LCM is a hybrid method between Gabor and GLCM feature extraction that applied to histopathology images. These methods implemented on Chromoendoscopy (CH) images. The CH images are being classified into two types: Normal

and abnormal frames using Support Vector Machine (SVM) as the classifier algorithm. In this research, SVM has revealed to have the best performance compared to other classifiers in terms of sensitivity, specificity, and accuracy up to 91%, 82%, and 87% respectively. The GLCM is also successfully applied as a technique for feature extraction on histopathological images and is also combined with local binary patterns, called LBGLCM (Öztürk and Akdemir, 2018). Another research is proposed by Peyret *et al.* (2018) propose the multiscale multispectral local binary pattern and combine it with GLCM for histology tumor classification. The images are classified by Support Vector Machine and obtain the highest accuracy (%) up to 99.6±0.4 for Qatar dataset (Peyret *et al.*, 2018).

Other previous research is combining GLCM and some texture extraction methods such as first-order statistics, local binary pattern discrete wavelet and gray-level run-length matrix. In this case, the GLCM is used for the characterization of coronary plaque region for Intravascular Ultrasound (IVUS) images. The classification of data using ensemble classifiers in this study reveals the characterization accuracy of about 81% for Fibro-Fatty Tissue (FFT) and 75% for Necrotic Core (NC) images. In some cases, GLCM is often combined with other texture extracting methods to obtain higher accuracy (Dhahbi *et al.*, 2018). It is possible to combine GLCM with other handcrafted feature extraction not only texture-based but also shape-based or color-based. In this research, in order to accelerate and optimize the extraction, GPU is used in the GLCM feature extraction technique (Men *et al.*, 2013).

Mean-Shift Algorithm

The mean-shift algorithm is a robust method for feature space analysis proposed by Comaniciu and Meer (2002). The mean-shift filter algorithm has been applied for filtering, segmenting and clustering images. The challenge in the mean-shift algorithm is finding the proof convergence of the algorithm in high dimensional space. The characteristic of mean-shift is a nonparametric – iterative algorithm. The extension paper related to the mean-shift algorithm that proof the convergence step in the mean-shift algorithm is proposed by Ghassabeh (2013). In the medical field, the mean-shift algorithm has been applied to medical research (Vallabhaneni and Rajesh, 2017), (Aparajeeta *et al.*, 2018), (Guo *et al.*, 2018; Mure *et al.*, 2015; Bai *et al.*, 2013; Yang *et al.*, 2013). The idea of mean-shift algorithm is to find pixel with similar characteristics of the density and save the distinct pixel value. The mean-shift algorithm is widely used for image or object video tracking. Furthermore, the use of Graphics Processing Unit (GPU) on mean-shift

segmentation algorithm is aimed to accelerate the computation time (Men *et al.*, 2013).

Deep Neural Network

A Deep Neural Network (DNN) is a neural network with multiple hidden layers between input and output (Bengio, 2009; Schmidhuber, 2014). DNN had been proven successful in solving complex problems mathematically (Bianchini and Scarselli, 2014). In medical field, DNN has been implemented to classify brain tumors images and outperform other conventional classification techniques (Mohsen *et al.*, 2017). Recently, deep learning becomes state-of-the-art techniques in many research areas, including medical image processing. There are various deep learning architecture such as RNN, LSTM, CNN, DBN, LSTM, DSN, and GAN. Deep Neural Network is a type of deep learning. Usually, training using Deep Neural Network is very time-consuming due to a large amount of data, high dimensionality, and complexity of the neural network. However, the learning process can be accelerated by GPU computing. The research about GPU acceleration in Convolutional Neural Network has been conducted in medical image applications (Li *et al.*, 2017a; Du *et al.*, 2017; Zhang *et al.*, 2017; Li *et al.*, 2017b). Besides, a GPU-accelerated library called CNNdroid has been introduced. This library can execute trained CNN on Android-based mobile devices (Oskouei *et al.*, 2015).

Material and Methods

Histopathological Images Dataset

Histopathology images were obtained from the MICCAI 2015 contest. MICCAI is a non-profit society to collaborate in research, education, and practice in the field of image computing and computer-assisted medical intervention. This society organizing and operating the annual high quality of the international conference, workshop, contest, and publication. The real data consist of two classes of cancer status: Benign and malignant. It is proposed by previous research on nuclei detection and classification (Sirinukunwattana *et al.*, 2016). The data have been annotated by experts and were obtained from sixteen patients. The real images of the data have 755×522 pixels. The data comprises of 74 benign and 91 malignant of cancer grade. For training in machine learning, we have to enrich the data. Random subsample sampling is implemented to these data and yields 200×200 pixels images. This process resulted in 941 benign and 910 malignant images. After that, we use the sliding windows technique with 15-pixel size to obtain pixel patch of the images. Finally, we have about 29617 datasets for benign and malignant images.

Multipatch GLCM

The equations are an exception to the prescribed specifications of this template.

In this section, we will explain multipatch GLCM as one technique for texture feature extraction on histopathology images. Actually, GLCM can be directly applied to 200×200 pixels images. However, not all of the nucleus area is clearly differentiated or at times poorly differentiated between benign and malignant types. Cancer identification can be analyzed with the structure of the nucleus. Normal tissues usually have a regular shape of the nucleus. Meanwhile, the malignant tissue has an irregular shape of the nucleus. However, tissues with poor differentiation occasionally is indistinguishable. This technique will be able to capture detail pixel by pixel and that the texture information will deliver different information.

The whole images of tissues are filtered by the mean-shift filter algorithm for noise reduction of the structure of tissues. We start from the pixel {0,0} then we capture pixel patch with various dimension where {length = weight: 80, 100, 120,140 and 160}. Next pixel move with sliding 15-window so that the pixel information in each image patch preserved. This mechanism will produce more sub-image as the new dataset significantly. In this research, we can produce more than fourteen thousand images for each class. Subsequently, the RGB images are converted to the gray image before six variables of GLCM extracted. There are main variables of GLCM: Contrast, dissimilarity, homogeneity, angular second moment, energy, and correlation. The detail steps of the proposed technique illustrated in Fig. 1.

Mean-Shift Filter Algorithm

The mean-shift filter applied in our research is aimed to reduce noise before the texture information is extracted. The basic concept of this technique is for a low-pass filter by considering the surrounding pixels algorithm and is described in the pseudocode.

In mean-shift filtering, defining s or spatial radius is not trivial and that we have to select it experimentally. In this research, we adjust the value for $s = 21$. The value of r refers to parameters of the color distance of the mean-shift algorithm. We use $r = 51$ experimentally.

The idea of the mean-shift filter is to compute a new spatial center, and new color means. The procedure will iterate until spatial and color mean will stop changing or stopped by the maximum of iteration is achieved. Algorithm 1 describes the pseudo-code mean-shift filter. According to the algorithm, g is kernel function derived from Gaussian kernel or Parzen windows to determine probability density function. In the mean-shift filter, the value of the deviation standard σ will be replaced with the h parameters of the mean-shift filter.

Algorithm 1. Mean-shift filter

Input : images $x_n = (x_n^s, x_n^r), n = 1 \dots N$
parameters : $h = (s, r)$
Data : $c_i = (c_i^s, c_i^r), n = 1 \dots N$ #source_points
Data : $z_i = (z_i^s, z_i^r), n = 1 \dots N$ #filtered_point

for $i = 1, \dots, N$ **do**
 initialize $j = 1$ and $y_{i,1} = c_i = (x_i^s, c_i^r)$
 while not converged **do**
 calculate $y_{i,j+1}$ according to

$$y_{j+1} = \frac{\sum_{i=1}^n c_i g\left(\left\|\frac{y_{i,j} - c_i}{h}\right\|^2\right)}{\sum_{i=1}^n g\left(\left\|\frac{y_{i,j} - c_i}{h}\right\|^2\right)}$$

 $y_{i,j+1}$ is a new position of the kernel windows
 n : the number of point in the spatial kernel centered on $y_{i,j}$
 $y_{i,conv} = y_{i,j+1}$
 assign $z_i = y_{i,conv}$

The filtering process using mean-shift with g kernel function will be affected by the h parameter which replaces the mean and standard deviation if we use a Gaussian filter. The final output is filtered or smooth image with a similar dimension as the input image. Figure 2 depicts several results of the mean-shift filter for benign and malignant tissue.

Design GLCM Scenario

Pixel information from GLCM can be obtained from a distance and angle orientation. Too near distance causes the information in each pixel to be relatively homogenous. However, too far also cause the information between pixel not relevant. Therefore, we determine the distance $d = 5$ and design the scenario to cover four angles including $\{0^\circ, 45^\circ, 90^\circ \text{ and } 135^\circ\}$. Because we have some multipatch images, the scenario will be implemented to all of the patch images. The table below describes the scenario.

The output of the scenario above is table data which comprises six columns as features of GLCM images and 29617 rows of all total data benign and malignant. Overall, we have twelve sets of data before processing them in the Deep Neural Network.

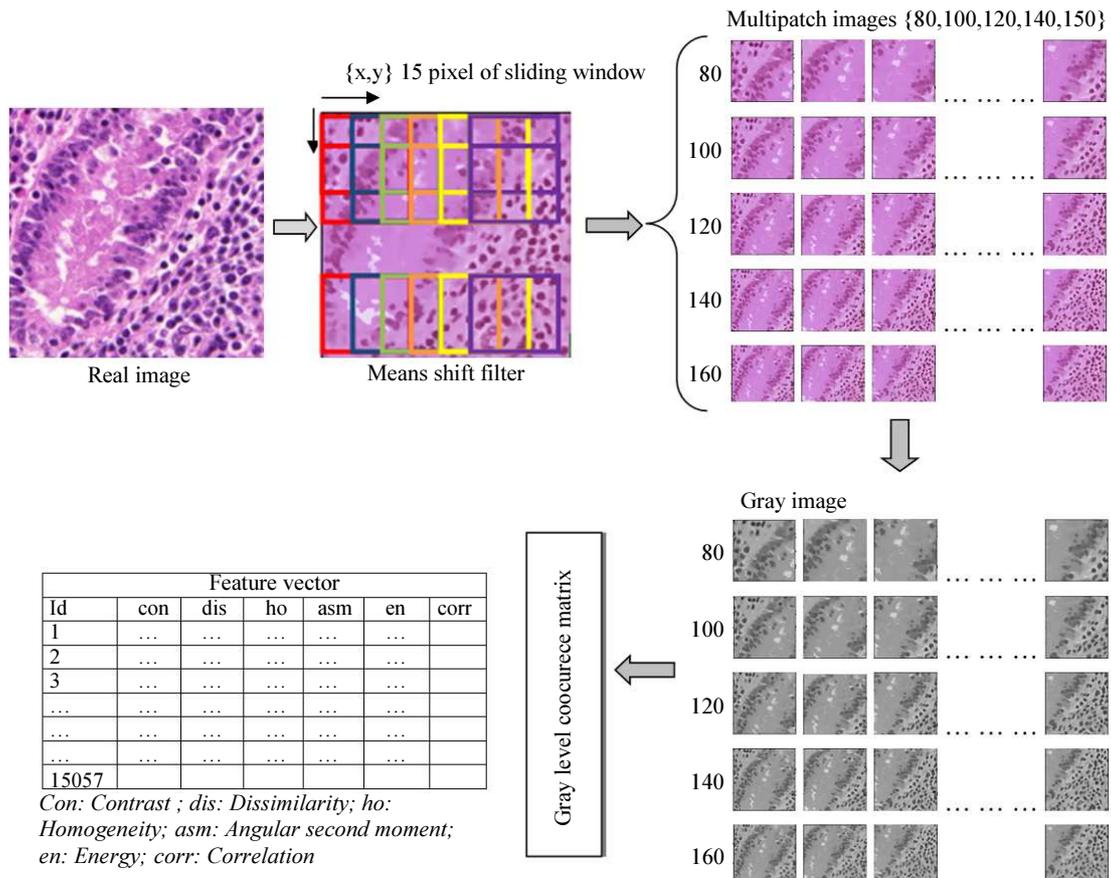


Fig. 1: Illustration of Multipatch-GLCM

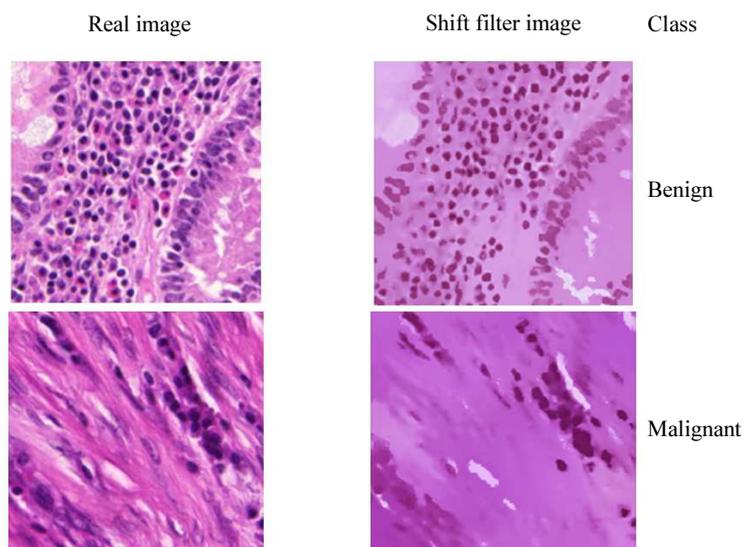


Fig. 2: Sample of mean shift filtering algorithm on histopathology images

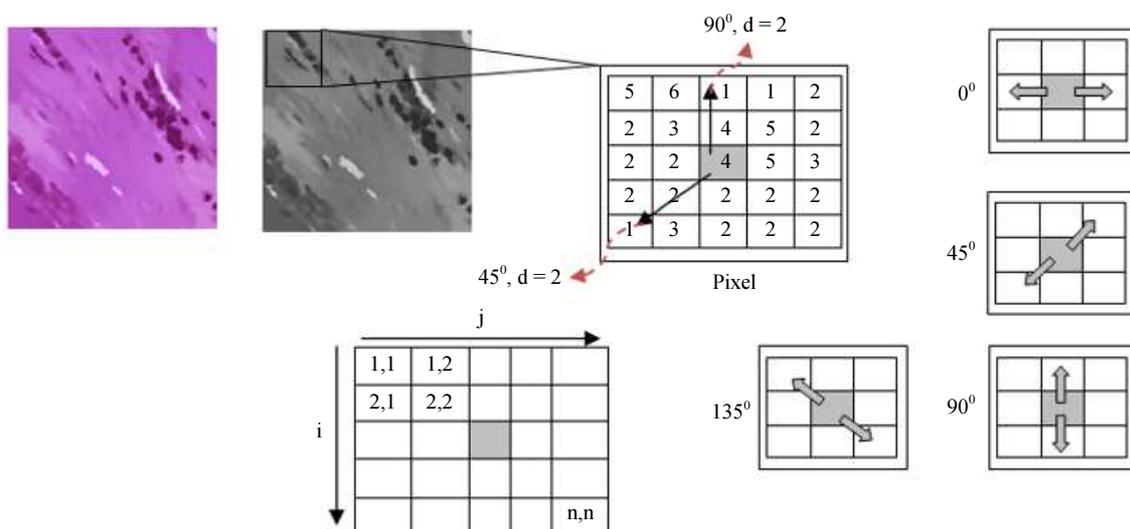


Fig. 3: Illustration of GLCM with angle ($0^\circ, 45^\circ, 90^\circ, 135^\circ$) and distance ($d = 2$) orientation from reference pixel

Table 1: Scenario of GLCM feature extraction with $d = 5$ and angle $\{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$

Angle	Pixel patch	GLCM variables
0°	120×120, 140×140, 160×160	contrast, dissimilarity, homogeneity, energy, angular second moment (asm), correlation
45°	120×120, 140×140, 160×160	contrast, dissimilarity, homogeneity, energy, angular second moment (asm), correlation
90°	120×120, 140×140, 160×160	contrast, dissimilarity, homogeneity, energy, angular second moment (asm), correlation
135°	120×120, 140×140, 160×160	contrast, dissimilarity, homogeneity, energy, angular second moment (asm), correlation

Feature extraction with GLCM is used to extract information with some variables. The main variables used in this research are Contrast, dissimilarity, homogeneity, energy, and correlation. GLCM will extract pixel information after RGB images converted to gray images. As mention above, two orientations in

GLCM are: Distance and angle. Fig. 3 will illustrate how the GLCM works.

Before calculating the variable of GLCM, the co-occurrence matrix will be normalized first. The normalization is calculated by dividing the total number of accumulated co-occurrence. After normalizing, the

element of the resulting sum to 1. Six variables as mention above will be calculated, this refers to the Haralick papers as a founder of GLCM (Haralick *et al.*, 1973). All of the variables are defined below:

$$contrast = \sum_{i,j=0}^{N_g-1} p(i,j)(i-j)^2 \quad (1)$$

$$dissimilarity = \sum_{i,j=0}^{N_g-1} p(i,j)|i-j| \quad (2)$$

$$homogeneity = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{p(i,j)}{1+(i-j)^2} \quad (3)$$

$$energy = \sum_i \sum_j \{p(i,j)\}^2 \quad (4)$$

$$asm = \sum_i \sum_j \{p(i,j)\}^4 \quad (5)$$

$$correlation = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} [(ij)p(i,j)] - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (6)$$

$p(i,j)$ with $(i,j)^{th}$ is the element of normalized symmetric of the co-occurrence matrix of GLCM. Contrast refers to the measurement of the intensity between the reference pixel and its neighboring pixels with the specified angle and distance. If there is a large amount of variation in an image, the contrast will be high. Dissimilarity will measure the distance between the pair of the object (pixels) in the region interest (distance and angle). A larger value implies a greater disparity in intensity values

among neighboring pixels. Homogeneity measures how close the distribution of a pixel is in a GLCM. This value will be inversely proportional to contrast. If homogeneity increases, the contrast decreases. ASM measures the uniformity of pixels in GLCM. As the pixels get more similar, the ASM value is also large. The energy in GLCM is derived from ASM. Energy is the root of ASM. Correlation shows the linear dependency of the gray level value in the GLCM that indicates a local gray-level dependency on the texture image. Higher values of correlation can be obtained for a similar gray-level area. The final target of all equations is six texture features for each image patch as the input for the deep neural network training process.

From Fig. 3 we can understand how the GLCM variables obtained. The various angles (0^0 , 45^0 , 90^0 and 135^0) represent the focus pixels position. Meanwhile, $d = 2$ is the distance from reference pixel to neighbor pixel. After the position is determined each variable is calculated.

Designing Deep Neural Network Architecture

Deep Neural Network is a network with the multi hidden layer between output and output. The study of the deep neural network gives upper bound and lower bound of activation function for the network with h hidden unit, l hidden layers and n input (Bianchini and Scarselli, 2014). According to this research, Deep Neural Network is able to address more difficult problems than other methods.

This research inspires us to design the two deep neural network architectures for our experiment. Our network has six input vector features obtained from GLCM and will be trained to various networks to the output layer. Illustration of our architecture 1 with four hidden layers shown in Fig. 4.

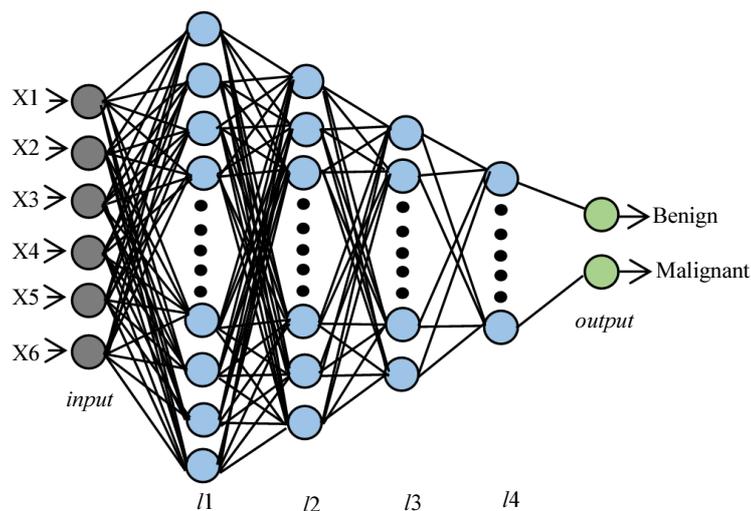


Fig. 4: Deep neural network in architecture 1 with 4 hidden layer

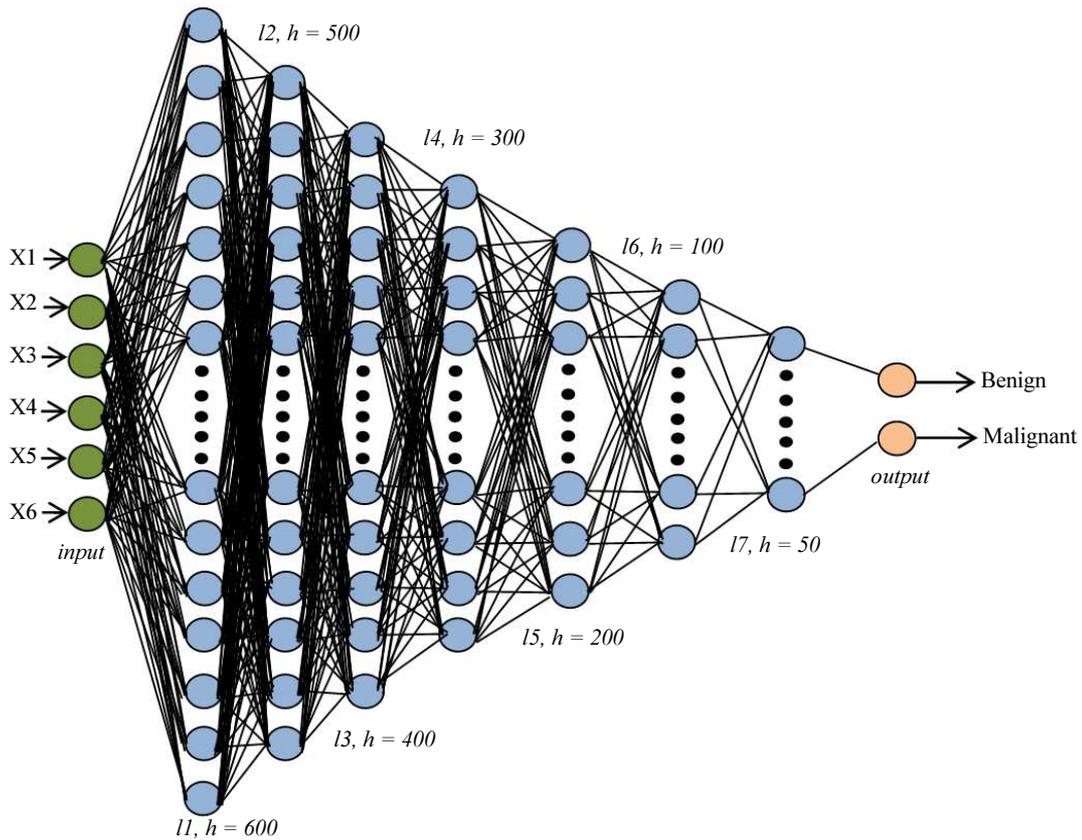


Fig. 5: Deep neural network in architecture 2 with 7 hidden layer

Our network is also modified to be deeper which we called architecture 2. In this architecture, we apply the additional hidden layers which become seven ($l = 7$). Some papers propose the number of hidden units which try to cover the optimal number of hidden units theoretically and experimentally (Sheela and Deepa, 2013). Too many hidden units will cause a network to be more complex and overfitting. On the contrary, too little hidden units will cause underfitting. The detail of the first architecture is illustrated in Table 2. Meanwhile, the second architecture is described in Fig. 5.

Adjusting parameters in a deep neural network is not trivial. For activation function, we use a Rectified Linear unit (ReLU) because of the ability to avoid vanishing gradient (Qayyum *et al.*, 2017) and to make training faster. The formula of ReLU is shown in Equation (7):

$$\text{ReLU} = \begin{cases} 0 & ; \text{for } x < 0 \\ x & ; \text{for } x \geq 0 \end{cases} \quad (7)$$

To reduce the bias, k-fold cross-validation is implemented when training with a neural network or deep neural network. In our model, we use four cross-validations. The accuracy with deviation standard is used to measure the validity of our data set. Deep

Neural Network needs to be tuned and optimized for obtaining a good model. One of the parameters is Adam for optimizing. Combining Adam with ReLU activation function results in the lowest training cost experimentally. Adam is an algorithm for first-order gradient-based optimization. In the training step, adam has some advantages such as little memory requirement, computationally efficient and the suite from the problem with large in terms of data and parameters (Kingma and Ba, 2015). Therefore, this technique is suitable for our case.

In the output layer, a softmax function is used. Softmax function will calculate the probabilities of each target class of the overall possible target class. The formula is shown in Equation (8):

$$f(x_i) = \frac{e^{x_i}}{\sum_i e^{x_i}} \quad (8)$$

The softmax function is used to classify the status of cancer at the end of architecture with a probability value for each predicted class. This value is straightforward to understand by the pathologist in practice. The sum of the probability value of all class is 1.

Table 2: Design of proposed deep neural network architecture

Architectures	Input	Output	Hidden layer (l), hidden unit (h)	Parameters
1	6	2	l1: h = 600, l2: h = 500, l3: h = 300, l4: h = 100	hidden layer activation: relu output layer activation: softmax
2	6	2	l1: h = 600, l2: h = 500, l3: h = 400, l4: h = 300, l5: h = 200, l6: h = 100, l7: h = 50	batchsize = 60 epoch = 5000 optimizer = Adam learning rate = 0,0001 loss = binary cross entropy

Table 3: Machine specification for training deep neural network

	Specification	
	Machine 2	Machine 1
Processor	Intel core i7® 5960X 16 cores @ 3.00 Ghz	AMD Opteron 6344 12 cores @ 1,3 GHz
RAM	65 GB	32 GB
GPU	NVIDIA GTX-980 Number cores: 2048 Clock: 1280 MHz GPU-RAM: 4GB	NVIDIA Tesla K40c Number cores: 2888 Clock: 745 Mhz GPU-RAM: 11 GB

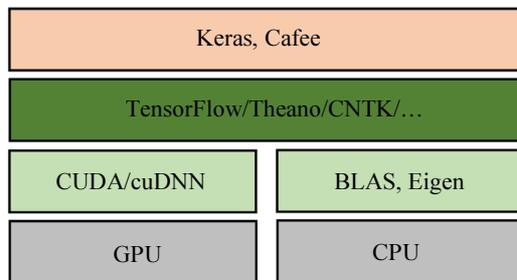


Fig. 6: Software and hardware layer for deep learning application

Experimental Setup

Training Deep Neural Network requires machine infrastructure with high specifications for computation. Therefore, in our experiment, we use two GPU machines, as listed in Table 3.

For the training architecture of DNN, we use Keras as the tool. Keras is a high-level framework for running deep learning. It needs the middle library to be able to run on GPU. Keras can run with Tensorflow and Thenao backend and access low-level library “cudnn” from Nvidia. Hierarchically, the software and hardware model layer of Deep Neural Network illustrated in Fig. 6 (Fallis 2013).

Our research is conducted on Linux 16.04 LTS (Long Term Support) and Debian 8.9 operating system with CUDA 9.0 version. Linux 16.04 LTS is a stable version of Ubuntu in this version with long support of packages.

Result

Gray Level Co-Occurrence Matrix Information

Handcrafted feature extraction using GLCM in this research is obtained from six variables for textures

information. The example of GLCM information is described in Fig. 7.

Training Step and Performance of Accuracy

The performance of accuracy also measured with various mini-batch that will become the basis of the number of the mini-batch size chosen. Mini-batch size is the number of the dataset in the feed-forward process in the training step to update the weight of our network. During training, we record the training step as shown in Fig. 8.

The detail of accuracy is represented by the accuracy rate for two types of our network architectures run on each machine (Fig. 9 and Fig. 10). Fig. 9 shows the accuracy with some number of batch-size that is conducted experimentally on Tesla. We also record the performance run on the other machine GTX-980.

The architectures are trained using four cross-validations to obtain the validation dataset. To reduce bias, we calculate the deviation standard for accuracy. The result of accuracy also will be compared to several supervised techniques such as K-Nearest Neighbors (KNN), Linear Discriminant Analysis (LDA) and Decision Tree. Table 4 shows the accuracy of our experiment.

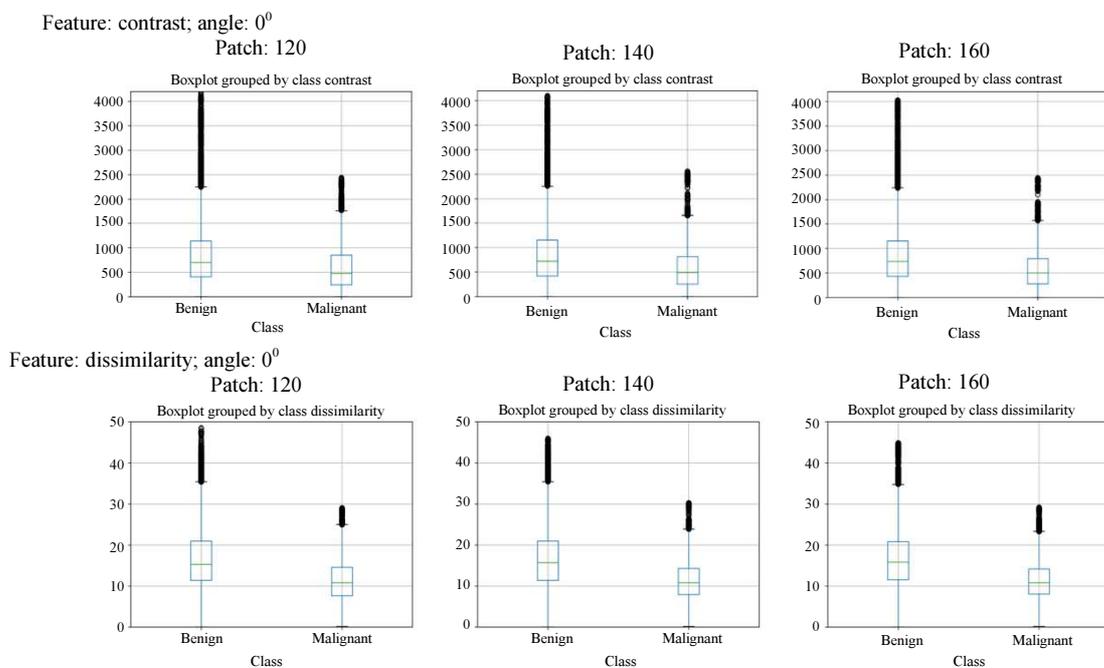
Training Time

Deep learning commonly needs training time to build a model for classification. According to the performance of accuracy in Table 4, Deep Neural Network (DNN) outperforms other methods. However, there is a trade-off between accuracy and computation or training time to build the DNN model. For that, we use two machines with NVIDIA GPU for training our models. Graphics in Fig. 11 and 12 show the training time of DNN by NVIDIA Tesla K40c and GTX-980 for the proposed architectures.

Tensorflow is one of the famous Deep Learning framework developed by Google (Abadi *et al.*, 2015). Meanwhile, Theano is a simple framework which is originated from the University of Montreal. The framework is developed by individual researchers and by a team of researchers under their collaborative work (TDT, 2016).

Table 4: Cross-validation accuracy of histopathology images with multiple patches

	Patch-size		
	120	140	160
Angle = 0, Distance = 5			
KNN	70.57 (0.87)	72.23 (0.23)	73.51 (0.62)
LDA	72.35 (0.43)	73.75 (0.66)	74.73 (0.68)
Decision Tree	84.34 (0.34)	88.40 (0.13)	90.64 (0.12)
DNN-Architecture 1 Machine 1	88.02(1.33)	92.36 (0.49)	93.24 (0.93)
DNN-Architecture 2 Machine 1	90.69 (1.43)	93.80 (0.89)	95.65 (0.60)
DNN-Architecture 1 Machine 2	87.93 (1.32)	90.88 (2.14)	93.50 (0.72)
DNN-Architecture 2 Machine 2	90.67 (0.55)	94.37 (0.85)	95.34 (0.38)
Angle = 45, Distance = 5			
KNN	71.05 (0.82)	72.32 (1.13)	73.60 (0.57)
LDA	73.04 (0.37)	74.63 (0.39)	75.46 (0.30)
Decision Tree	84.60 (0.05)	88.42 (0.32)	90.55 (0.39)
DNN-Architecture 1 Machine 1	89.73 (0.23)	92.46 (0.62)	92.20 (2.58)
DNN-Architecture 2 Machine 1	89.02 (1.29)	94.39 (0.44)	95.52 (0.95)
DNN-Architecture 1 Machine 2	89.30 (0.54)	92.29 (1.24)	94.78 (0.83)
DNN-Architecture 2 Machine 2	89.54 (2.89)	92.79 (2.62)	94.75 (1.44)
Angle = 90, Distance = 5			
KNN	69.84 (0.88)	71.48 (0.86)	72.97 (0.95)
LDA	72.39 (0.30)	73.67 (0.58)	74.50 (0.50)
Decision Tree	84.00 (0.7)	87.41 (0.21)	90.50(0.19)
DNN-Architecture 1 Machine 1	87.58 (2.46)	90.91 (1.18)	94.10 (1.70)
DNN-Architecture 2 Machine 1	91.73 (0.54)	94.53 (1.02)	96.72 (0.48)
DNN-Architecture 1 Machine 2	87.2(2.16)	91.73 (1.10)	94.98 (0.20)
DNN-Architecture 2 Machine 2	91.13 (0.67)	91.97 (1.98)	95.53 (1.72)
Angle = 135, Distance = 5			
KNN	70.27 (0.67)	72.03 (0.81)	73.18 (0.73)
LDA	71.08 (0.41)	72.32 (0.64)	72.52 (0.67)
Decision Tree	84.34 (0.42)	88.43 (0.13)	90.82 (0.17)
DNN-Architecture 1 Machine 1	88.88 (1.10)	90.97 (1.05)	92.50 (2.65)
DNN-Architecture 2 Machine 1	89.95 (1.88)	93.95 (1.09)	93.60 (1.06)
DNN-Architecture 1 Machine 2	89.06 (0.32)	89.99 (3.23)	93.68 (1.58)
DNN-Architecture 2 Machine 2	90.84 (1.03)	94.53 (0.56)	94.60 (1.22)



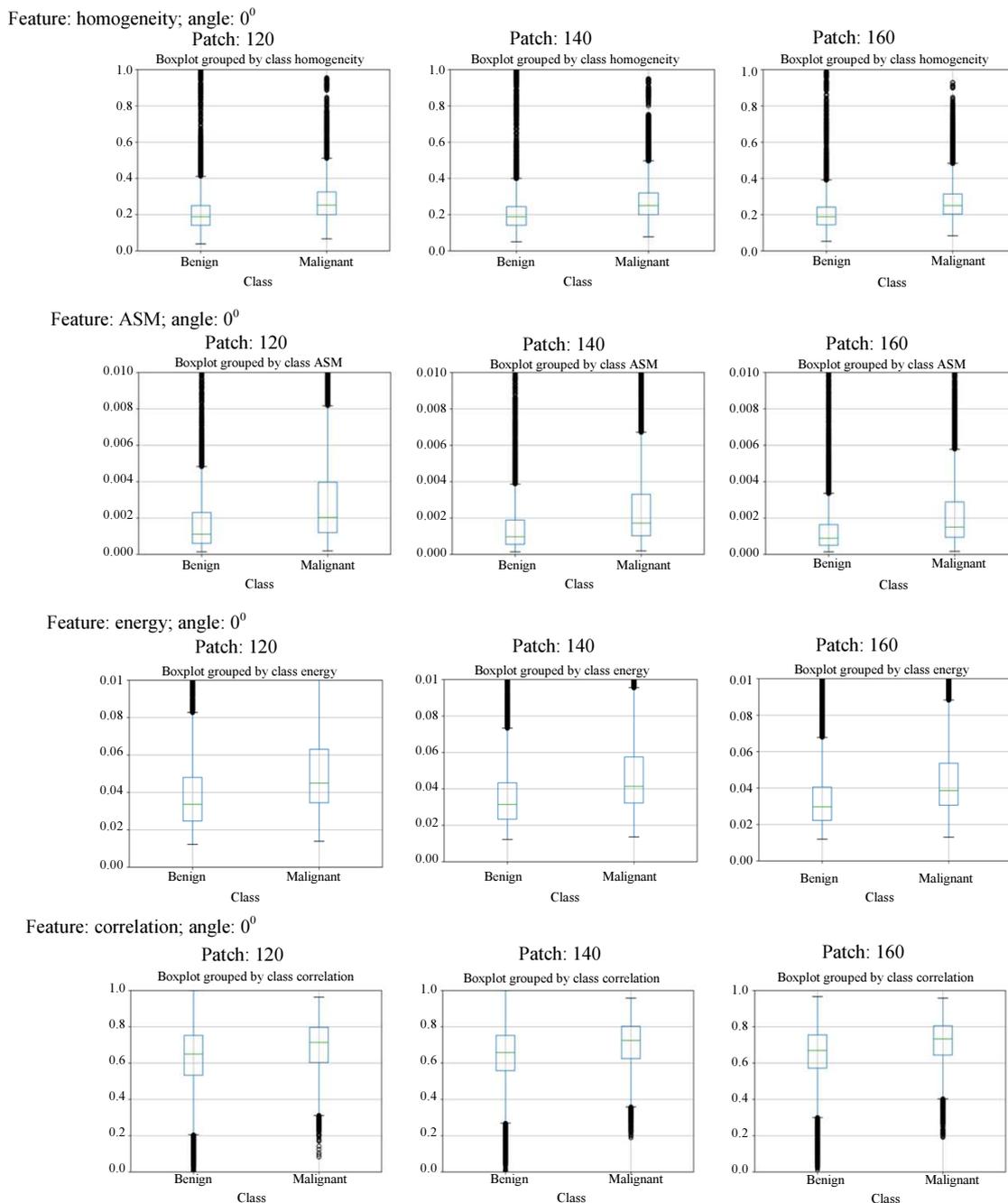


Fig. 7: Result of GLCM features extraction for angle 0^0 with the distance $d = 5$

Discussion

Histopathology image classification is still challenging to be exploited. Not only for a computer scientist but also for a pathologist. Feature extraction is an important part of histopathology classification. The GLCM is basically one of the texture features for image processing in many research field including histopathology. In this research, whole slide images are filtered using a mean-shift filter first. This filtering

process is aimed to reduce the noise. Then, the images are converted to gray images based on the formula (10):

$$RGB \rightarrow Gray : Y \leftarrow 0.299 * R + 0.587 * G + 0.114 * B \quad (10)$$

where, R , G , and B are red, green and blue channels. Before applying GLCM, the sliding windows step is conducted to enrich the dataset. To minimize the loss of information, we use 15 pixels of windows-size. The windows-size is used to minimize the selected images.

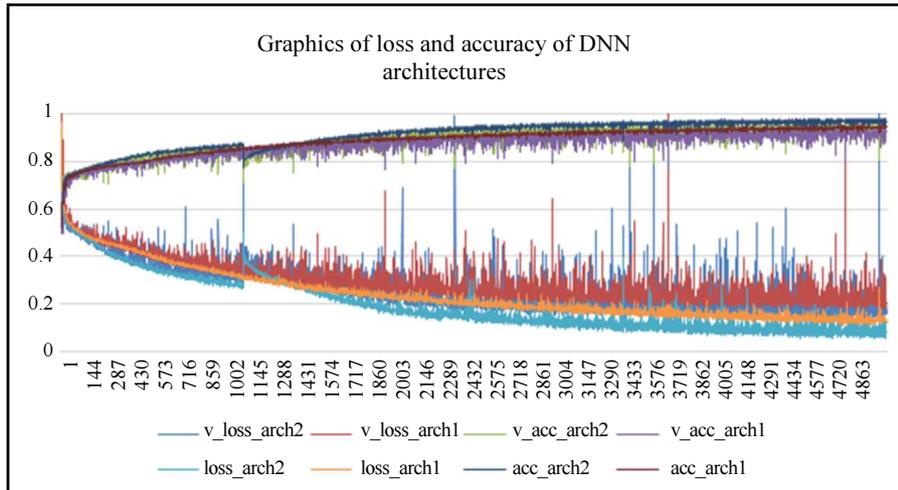


Fig. 8: Graphics of loss and accuracy from architecture 1 and architecture 2

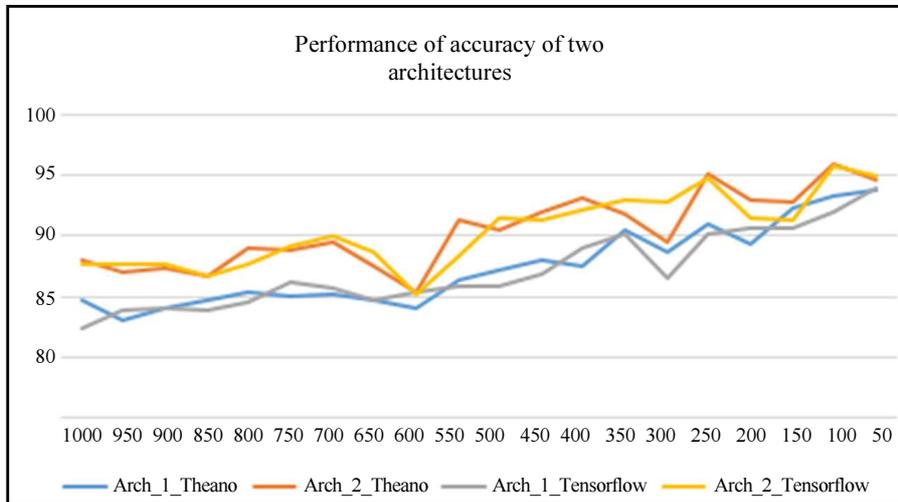


Fig. 9: Performance of accuracy based on the number of batch-size on Tesla K40C

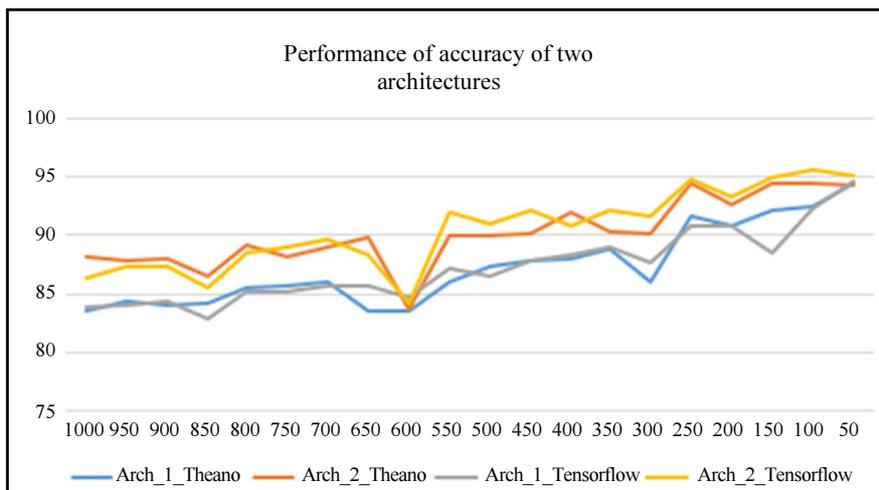


Fig. 10: Performance of accuracy based on the number of batch-size on GTX-980

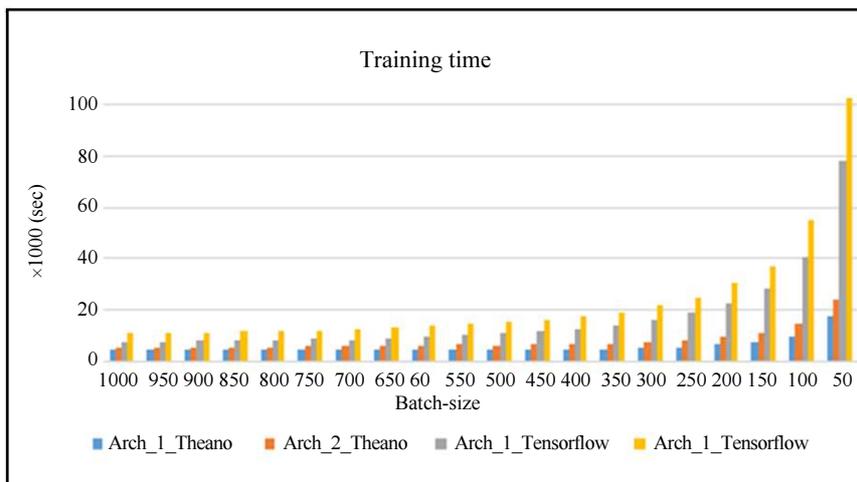


Fig. 11: The training time of two proposed DNN architectures on Tesla K40c using Theano and Tensorflow

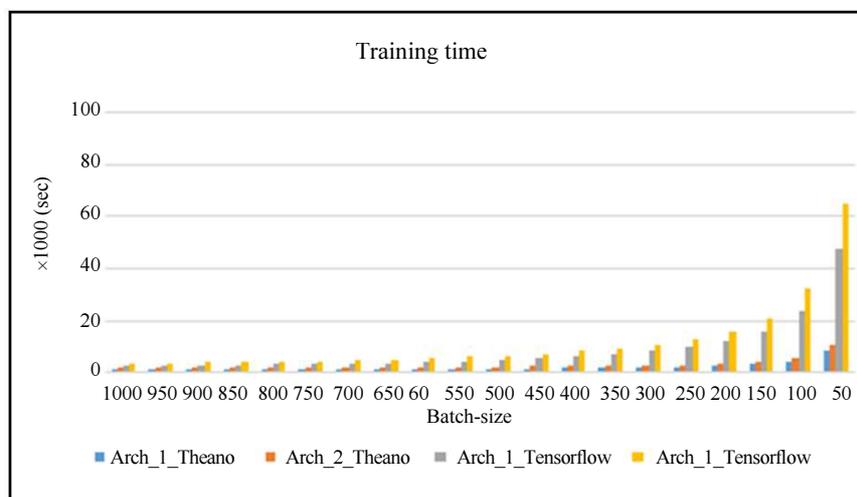


Fig. 12: Training time of two proposed DNN architectures on GTX-980 using Theano and Tensorflow

For each window, various pixel patches are captured and become the essence of our research. The usage of multipatch is that it can produce more images features significantly for the training set. In our research, there are 15056 images for benign and 14561 images for malignant. GLCM has some features to be extracted, and for this research, we use six parameters: contrast, dissimilarity, homogeneity, angular second moment, energy, and correlation.

In GLCM, the orientation of angle and distance of pixel will determine the information texture produced. Therefore, we design various angles from 0° , 45° , 90° , and 135° as illustrated in Fig. 3 to cover the surrounding pixels. Meanwhile, for the distance, we choose $d = 5$ combined with various angles experimentally. The result of GLCM is shown in the box-plot graphics in Fig. 7. From these figures, we understand the characteristic of the

parameters. The outlier of the parameters is applied to k-fold cross validation when training and validating the Deep Neural Network model.

In our study, we propose two types of architectures as shown in Fig. 4 and 5. The second architecture is deeper than the first one with seven hidden layers. Deeper architecture should produce a better model for predicting or classifying. The performance accuracy of our model is described in Table 4. From Table 4, it can be explained that the accuracy of multiple pixel patches calculated from 120, 140, and 160. Overall, Deep Neural Network (DNN) outperforms other classification methods such as KNN, LDA, and Decision Tree. Table 4 also explains that the highest accuracy for the validation dataset achieved on 160 pixels. There are two cases when the 140 pixels patch reveals higher accuracy 92.46 (0.62) and 93.95 (1.09) for angle 45° and 135° respectively.

Table 5: Accuracy of new dataset

Architecture 1	Architecture 2
0.78	0.80

Our model also tests the new dataset, as described in Table 5. The new dataset is data that never used for validating the model.

According to the hardware resources described in Table 3, the machine with Tesla K40c has more the number of cores than GTX-980. However, the clock speed of the Tesla K40c is not as fast as GTX-980. K40c has 745 MHz and GTX-980 has 1280 MHz of processor speed. In this research, we also record the training time for both the GPU machines when training the architectures. The fast performance of GTX-980 is due to the floating-point performance, which is 4.6 TFLOPS while Tesla is 4.29 TFLOPS. Tera-Floating Point Operation Per Second (TFLOPS) is a way of measuring the power of computer based more on computer mathematical capability. The TFLOPS refers to the capability of a processor to calculate one trillion floating-point operations per second. GTX-980 has 4.6 TFLOPS means this processor of GPU can calculate 4.6 floating-point operations every second on average. Meanwhile, Tesla K40c only calculate 4,2 floating-point operations every second. GTX-980 also has 1216 MHz GPU Turbo speed 341 higher than Tesla K40c with 875 MHz GPU turbo speed.

Conclusion

This research has succeeded in implementing multipatch GLCM as the feature extractor for histopathology images with its performance of accuracy that outperforms other classifiers. The Deeper architecture will result in better models for classification; however, the trade-off is the cost in training time. Nevertheless, the presence of GPU computing can solve the gap between accuracy and training time.

Acknowledgment

This research was supported by Penelitian Terapan Unggulan Perguruan Tinggi (PTUPT) grant number No. NKB-1691/UN2.R3.1/HKP.05.00/2019 from the Ministry of the Research Technology Republic of Indonesia. Contract Number: 1/E1/KP.PTNBH/2019 and 234/PKS/R/UI/2019. The work supported by ERDF/ESF "Cooperation in Applied Research between the University of Pardubice and companies, in the Field of Positioning, Detection and Simulation Technology for Transport Systems (PosiTrans)" (No. CZ.02.1.01/0.0/0.0/17_049/0008394).

Author's Contributions

Toto Haryanto: Participated in developing the program, experiment, and analysis. He also contributes writing of the manuscript.

Adib Pratama: His contribution develops the program and analysis. He also makes sure GPU machines are running well during training.

Heru Suhartanto and Jan Pidanic: His contribution is in GPU computing analysis and reviewing the manuscript.

Aniati Murni: Contributes to image processing and analysis. He also gives suggestions for the manuscript related to deep learning for image processing.

Kusmardi Kusmardi: His contribution to this paper is validating the histopathology images data for cancer classification.

Ethics

All authors declare that there is no ethical issue involved in this article.

References

- Ali, H., M. Yasmin, M. Sharif and M.H. Rehmani, 2018. Computer assisted gastric abnormalities detection using hybrid texture descriptors for chromoendoscopy images. *Comput. Methods Programs Biomed.*, 157: 39-47. DOI: 10.1016/j.cmpb.2018.01.013
- Aparajeeta, J., S. Mahakud, P.K. Nanda and N. Das, 2018. Variable variance adaptive mean-shift and possibilistic fuzzy c-means based recursive framework for brain MR image segmentation. *Expert Syst. Applic.*, 92: 317-333. DOI: 10.1016/j.eswa.2017.09.049
- Bai, P.R., Q.Y. Liu, L. Li, S.H. Teng and J. Li *et al.*, 2013. A novel region-based level set method initialized with mean shift clustering for automated medical image segmentation. *Comput. Biol. Med.*, 43: 1827-32. DOI: 10.1016/j.compbiomed.2013.08.024
- Banaie, M., H. Soltanian-Zadeh, H.R. Saligheh-Rad and M. Gity, 2018. Spatiotemporal features of DCE-MRI for breast cancer diagnosis. *Comput. Methods Programs Biomed.*, 155: 153-164. DOI: 10.1016/j.cmpb.2017.12.015
- Bengio, Y., 2009. Learning deep architectures for AI. *Foundat. Trends® Machine Learn.*, 2: 1-127. DOI: 10.1561/2200000006
- Bianchini, M. and F. Scarselli, 2014. On the complexity of shallow and deep neural network classifiers. *IEEE Trans. Neural Netw. Learn. Syst.*, 25: 1553-1565. DOI: 10.1109/TNNLS.2013.2293637
- Comaniciu, D. and P. Meer, 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Patt. Anal. Machine Intell.*, 24: 603-619. DOI: 10.1109/34.1000236

- Dhahbi, S., W. Barhouni, J. Kurek, B. Swiderski and M. Kruk *et al.*, 2018. False-positive reduction in computer-aided mass detection using mammographic texture analysis and classification. *Comput. Meth. Programs Biomed.*, 160: 75-83.
DOI: 10.1016/j.cmpb.2018.03.026
- Du, X., J. Tang, Z. Li and Z. Qin, 2017. Wheel: Accelerating CNNs with distributed GPUs via hybrid parallelism and alternate strategy. *Proceedings of the 25th ACM International Conference on Multimedia, (ICM' 17)*, ACM, California, pp: 393-401.
- Fallis, A., 2013. Deep learning with python.
- Ghassabeh, Y.A., 2013. On the convergence of the mean shift algorithm in the one-dimensional space. *Patt. Recognit. Lett.*, 34: 1423-1427.
DOI: 10.1016/j.patrec.2013.05.004
- Guo, Y., A. Şengür, Y. Akbulut and A. Shipley, 2018. An effective color image segmentation approach using neutrosophic adaptive mean shift clustering. *Measurement: J. Int. Measurement Confederat.*, 119: 28-40. DOI: 10.1016/j.measurement.2018.01.025
- Gurcan, M.N., L.E. Boucheron, A. Can, A. Madabhushi and N.M. Rajpoot *et al.*, 2009. Histopathological image analysis: A review. *IEEE Rev. Biomedical Eng.*, 2: 147-171.
- Haralick, R.M., K. Shanmugam and I. Dinstein, 1973. Textural features for image classification. *IEEE Trans. Syst. Man Cybernet.*, 6: 610-621.
DOI: 10.1109/TSMC.1973.4309314
- Kingma, D.P. and J. Ba, 2015. Adam: A method for stochastic optimization. *Proceedings of the International Conference on Learning Representations*, May 7-9, San Diego, pp: 1-15.
DOI: 10.1145/1830483.1830503
- Li, S., Y. Dou, Q. Lv, Q. Wang and X. Niu *et al.*, 2017. Optimized GPU acceleration algorithm of convolutional neural networks for target detection. *Proceedings of the 18th IEEE International Conference on High Performance Computing and Communications, 14th IEEE International Conference on Smart City and 2nd IEEE International Conference on Data Science and Systems*, Dec. 12-14, IEEE Xplore Press, Sydney, NSW, Australia, pp: 224-230.
DOI: 10.1109/HPCC-SmartCity-DSS.2016.0041
- Li, S., Y. Dou, X. Niu, Q. Lv and Q. Wang, 2017. A fast and memory saved GPU acceleration algorithm of convolutional neural networks for target detection. *Neurocomputing*, 230: 48-59.
DOI: 10.1016/j.neucom.2016.11.046
- Abadi, M., A. Agarwal, P. Barham, E. Brevdo and Z. Chen *et al.*, 2015. TensorFlow: Large-scale machine learning on heterogeneous systems.
- Men, L., M. Huang and J. Gauch, 2013. Accelerating Mean Shift Segmentation Algorithm on Hybrid CPU/GPU Platforms. In: *Modern Accelerator Technologies for Geographic Information Science*, Shi, X., V. Kindratenko and C. Yang (Eds.), Springer, Boston, MA, pp: 157-166.
- Mohsen, H., E.S.A. El-Dahshan, E.S.M. El-Horbaty and A.B.M. Salem, 2017. Classification using deep learning neural networks for brain tumors. *Future Comput. Inform. J.*, 3: 68-71.
DOI: 10.1016/j.fcij.2017.12.001
- Mukhopadhyay, S., M.D. Feldman, E. Abels, R. Ashfaq and S. Beltaifa *et al.*, 2018. Whole slide imaging versus microscopy for primary diagnosis in surgical pathology of 1992 cases (Pivotal Study). *Am. J. Surgical Pathol.*, 42: 39-52.
DOI: 10.1097/PAS.0000000000000948
- Mure, S., T. Grenier, D.S. Meier, C.R.G. Guttmann and H. Benoit-Cattin, 2015. Unsupervised spatio-temporal filtering of image sequences. A mean-shift specification. *Patt. Recognit. Lett.*, 68: 48-55.
DOI: 10.1016/j.patrec.2015.07.021
- Oskouei, S.S.L., H. Golestani, M. Hashemi and S. Ghiasi, 2015. CNNdroid: GPU-accelerated execution of trained deep convolutional neural networks on android. *Proceedings of the 24th ACM International Conference on Multimedia, (ICM' 15)*, ACM, pp: 1201-1205.
DOI: 10.1145/2964284.2973801
- Öztürk, Ş. and B. Akdemir, 2018. Application of feature extraction and classification methods for histopathological image using GLCM, LBP, LBGLCM, GLRLM and SFTA. *Proc. Comput. Sci.*, 132: 40-46. DOI: 10.1016/j.procs.2018.05.057
- Peyret, R., A. Bouridane, F. Khelifi, M.A. Tahir and S. Al-Maadeed, 2018. Automatic classification of colorectal and prostatic histologic tumor images using multiscale multispectral local binary pattern texture features and stacked generalization. *Neurocomputing*, 275: 83-93.
DOI: 10.1016/j.neucom.2017.05.010
- Qayyum, A., S.M. Anwar, M. Awais and M. Majid, 2017. Medical image retrieval using deep convolutional neural network. *Neurocomputing*, 266: 8-20. DOI: 10.1016/j.neucom.2017.05.025
- Schmidhuber, J., 2014. Deep learning in neural network: An overview. *Neural Netw.*, 61: 85-117.
DOI: 10.1016/j.eswa.2016.10.065
- Sheela, K.G. and S.N. Deepa, 2013. Review on methods to fix number of hidden neurons in neural networks. *Math. Problems Eng.* DOI: 10.1155/2013/425740
- Siegel, R.L., K.D. Miller and A. Jemal, 2018. *Cancer statistics, 2018*. CA: A Cancer J. Clinicians, 68: 7-30.
DOI: 10.3322/caac.21442

- Sirinukunwattana, K., S.E.A. Raza, Y.W. Tsang, D.R.J. Snead and I.A. Cree *et al.*, 2016. Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. *IEEE Trans. Med. Imag.*, 35: 1196-1206.
DOI: 10.1109/TMI.2016.2525803
- TDT, 2016. Theano: A python framework for fast computation of mathematical expressions. The Theano Development Team.
- Vallabhaneni, R.B. and V. Rajesh, 2017. Brain tumor detection using mean shift clustering and GLCM features with edge adaptive total variation denoising technique. *ARNP J. Eng. Applied Sci.*, 12: 666-671. DOI: 10.1016/j.aej.2017.09.011
- Yang, X., J. Pei and W. Sun, 2013. Elastic image registration using hierarchical spatially based mean shift. *Comput. Biol. Med.*, 43: 1086-1097.
DOI: 10.1016/j.combiomed.2013.05.006
- Zhang, J., J. Xiao, J. Wan, J. Yang and Y. Ren *et al.*, 2017. A parallel strategy for convolutional neural network based on heterogeneous cluster for mobile information system. *Mobile Inform. Syst.*
DOI: 10.1155/2017/3824765