BATIK CLASSIFICATION USING DEEP CONVOLUTIONAL NETWORK TRANSFER LEARNING

Yohanes Gultom[1], Rian Josua Masikome[2], and Aniati Murni Arymurthy[1]

[1]Faculty of Computer Science, Universitas Indonesia, Kampus UI, Depok, 16424, Indonesia
[2]Faculty of Mathematics, Computer Science and Natural Sciences, RWTH Aachen University, Templergraben 55, 52062 Aachen, Germany

E-mail: yohanes.gultom@ui.ac.id, rian.josua.masikome@rwth-aachen.de, aniati@cs.ui.ac.id

Abstract

Batik fabric is one of the most profound cultural heritage in Indonesia. Hence, continuous research on understanding it is necessary to preserve it. Despite of being one of the most common research task, Batik's pattern automatic classification still requires some improvement especially in regards to invariance dilemma. Convolutional neural network (ConvNet) is one of deep learning architecture which able to learn data representation by combining local receptive inputs, weight sharing and convolutions in order to solve invariance dilemma in image classification. Using dataset of 2,092 Batik patches (5 classes), the experiments show that the proposed model, which used deep ConvNet VGG16 as feature extractor (transfer learning), achieves slightly better average of 89±7% accuracy than SIFT and SURF-based that achieve 88±10% and 88±8% respectively. Despite of that, SIFT reaches around 5% better accuracy in rotated and scaled dataset.

Keywords: *batik, classification, deep learning, transfer learning*

Abstrak

Kain Batik adalah salah satu warisan kebudayaan Indonesia yang sangat berharga. Oleh karena itu, penelitian yang berkesinambungan perlu dilakukan untuk melestarikannya. Sekalipun telah menjadi topik penelitian yang umum, klasifikasi pola Batik secara otomatis masih memiliki beberapa tantangan yang perlu diselesaikan. Salah satu tantangan tersebut adalah masalah invariance dilemma. Convolutional neural network (ConvNet) adalah salah satu arsitektur deep learning yang mampu mempelajari representasi data dengan mengkombinasikan teknik local receptive inputs, weight sharing dan convolutions untuk mengatasi masalah invariance dilemma pada klasifikasi citra seperti pola Batik. Eksperimen menggunakan dataset 2,092 potongan foto Batik (5 kelas) menunjukkan bahwa model yang menggunakan ConvNet VGG16 sebagai ekstraktor fitur mencapai rata-rata akurasi 89±7% sedangkan model berbasis SIFT dan SURF mencapai rata-rata 88±10% dan 88±8%. Meskipun demikian, SIFT lebih akurat sekitar 5% pada dataset yang dirotasi dan diperbesar.

Kata Kunci: *batik, klasifikasi, deep learning, transfer learning*

1. Introduction

Batik fabric is one of the most profound cultural heritage in Indonesia. Hence, continuous research on understanding it is necessary to preserve it. One of the most popular research topic in computer science is batik classification. This topic can not be separated from another crucial subtopic: feature extraction. It is because in order to achieve high classification accuracy, a machine learning model requires numerical features extracted from Batik images.

Since the most prominent feature of Batik is its uniquely recurring pattern (motif), earlier researches have focused on finding a method to extract features from it. Earlier researches have shown good result using several method such as Generalize Hough Transform [1], Gabor, GLCM and LBP [2]. The recent methods that are currently considered as state of the art are Scale-Invariant Feature Transform (SIFT) [3] [4] and Speeded up robust features (SURF) [5]. Classifications using other features such as color and contrast are showing potentials but need to be researched further [6].

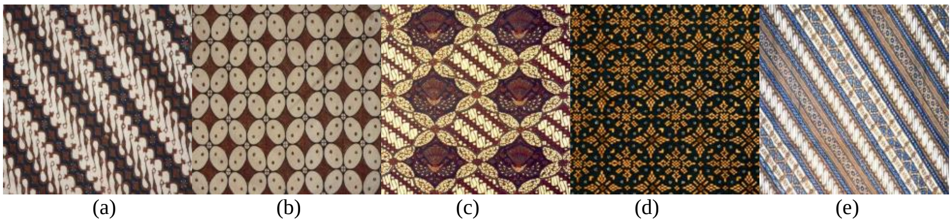

Figure 1. General Batik pattern classification, (a) Parang, (b) Kawung, (c) Ceplok, (d) Nitik, and (e) Lereng.

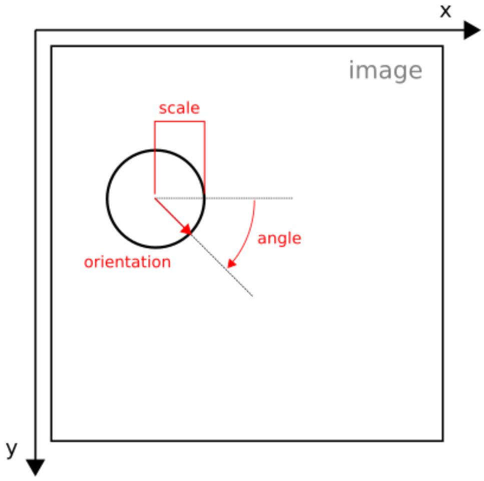

Figure 2. SIFT keypoint

Deep learning based models have outperformed state-of-the-art methods in many domains including image classification and object recognition [7]. One of the deep learning models, convolutional neural network (convnet) [8], is currently considered as the state-of-the-art of image classification model as it was used as the base structure by ILSVRC-2014 top achievers [9]. Therefore convnet has potential to improve result on other image classification problems such as Batik classification.

In this paper, we propose a neural network Batik clas- sification model that uses pre-trained deep convolutional network (VGG16) [9] as a feature extractor. Features from a dataset of five general classes of Indonesian Batik (shown in Figure 1) are extracted using VGG16, SIFT and SURF then classified using several machine learning classifiers. In order to test the capability of the model to solve invariance dilemma, tests are also done with rotated and scaled (zoomed) images.

2. Methods

Recent researches in Batik classification can be divided into two groups: (1) Researches on classification using handcrafted features and (2) researches on classification using automatically extracted features using deep learning.

Classification using Handcrafted Features

Since Batik classification has been researched for quite some time, current available methods are robust enough to noise addition, compression, and retouching of the input images. However most of them are still having difficulties with variance in transformations which involve either translation, rotation, scaling or combinations of them [4]. One of the initial work on Batik Classification was done using Generalized Hough Transform (GHT) to recognize Batik motifs as part of a content-based image retrieval (CBIR) [1]. The research focused on detection of repetitive motifs in a batik image but not yet addressed various orientations and scale.

One of the most recent research address the performance of several feature extraction methods (Gabor filter, log-Gabor filter, Gray Level Co-occurrence Matrix, and Local Binary Pattern) on rotated and scaled primitive Batik motifs [2]. It shows that applying Principal Component Analysis (PCA) to reduce dimensionality can improve the classification 17%. It also shows that applying Sequential Forward Floating Selection (SFFS) as feature selection makes the execution time 1,800 times faster.

Improvements on Batik classification were motivated by the emergence of Scale-Invariant

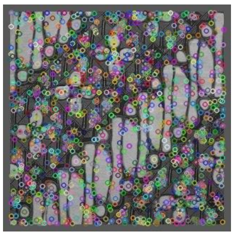

Figure 3. SIFT keypoint in Batik Parang.

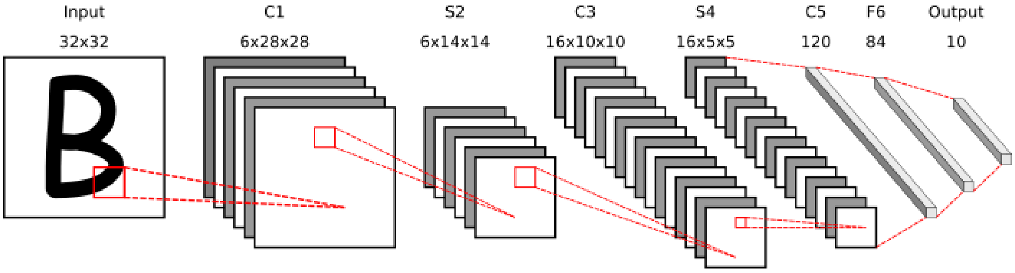

Figure 4. LeNet5convolutional network

Feature Transform (SIFT) [10] and Speeded up robust features (SURF) [11]. Both of these keypoint-based feature extraction methods are proposed to solve the transformation invariance dilemma. SIFT keypoint is a circular image region with an orientation which can be obtained by detecting extrema of Difference of Gaussian (DoG) pyramid [10]. It's defined by four parameters: center coordinates x and y, scale and its orientation (an angle expressed in radians) as shown in Figure 2. An image, for example Batik image, may contains multiple keypoints as shown in Figure 2. In order to be efficiently and effectively used as a feature for classification, the keypoint need to be represented as SIFT descriptor. By definition it is a 3-dimensional spatial histogram of the image gradients characterizing a SIFT keypoint.

Recent research proved that using SIFT descriptors to calculate similarity between Batik images can give 91.53% accuracy [4]. Voting Hough Transform was also applied to the descriptors to eliminate mismatched keypoint candidates hence improving the accuracy. This research suggested that the original SIFT descriptor matching should not be directly used to calculate similarity of Batik images due to many numbers of mismatched keypoints. This research uses fundamental templates of Batik patterns as a dataset instead of Batik photos. So it does not address issue related to noises which happen on non-processed images such as blur/unfocused, lightning issue, watermarks .etc.

Another research [3] proposed a classification method using support vector machine (SVM) fed by bag of words (BoW) features extracted using SIFT descriptors. In this research, SIFT descriptors also were not used directly as features for SVM but were clustered using k-means vector quantization algorithm to build vocabularies. These visual vocabularies then used to describe each images and fed to SVM classifier. This approach is required because SIFT descriptors have high dimensionality and vary between each images. The experiment results showed high average accuracy of 97.67% for normal images, 95.47% for rotated images and 79% for scaled images. Besides that SIFT and bag of words made a good feature extractor, this research also concluded that further works need to handle scaled Batik image cases.

An earlier research [5] proved that SURF can extract transformation invariant features faster than SIFT for classification of Songket, another Indonesian traditional fabric with motifs just like Batik. Unlike the others, this research used SIFT and SURF features directly to compute the matching scores between Songket images. The scores are calculated by (1) the number of matched keypoints and (2) the average total distance of the n-nearest keypoints. The result of experiments showed that the matching accuracy with SIFT features was 92-100% and 65-97% with SURF. With SURF features, the accuracy dropped quite significant if salt and pepper noises were added while SIFT was more stable. Apparently, this one was not paying much attention to transformation variance as it did not apply transformation noise as in other research [3].

Classification using Deep Learning

Deep learning is a multilayer representation learning in artificial neural network [7]. While representation learning itself is a method in machine learning to automatically extract/learn representa-

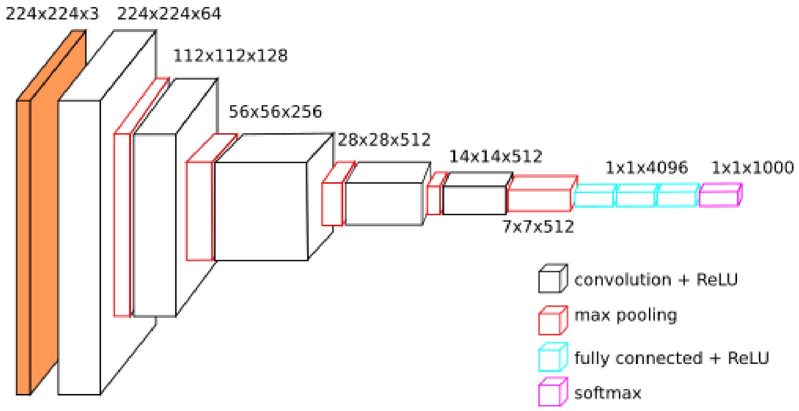

Figure 5. VGG16 deep convolutional network model of visual geometry group, Oxford.

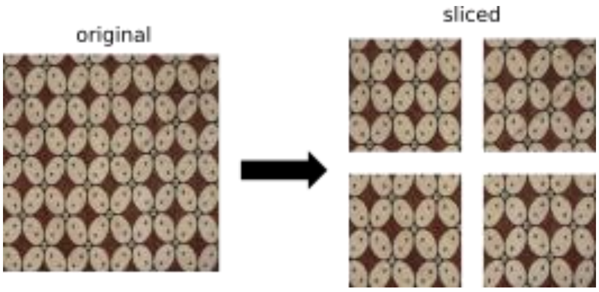

Figure 6. Example of generation of Batik Patches

tion (features) from raw data. The representation of the raw data then can be used for recognition or classification task. Some fundamental deep learning architectures for instances are convolutional neural network (ConvNet), deep belief network (DBN), autoencoder (AE) and recurrent neural network (RNN). Despite of being an old idea, it was recently emerged due to the several factors: (1) discovery of new techniques (eg. pretraining & dropout) and new activation functions (eg. ReLU), (2) enormous supply of data (big data), and (3) rapid improvement in computational hardware, especially GPU.

Proposed Method

We propose a deep convolutional neural network com- posed by a pre-trained VGG16 (without its top layer) as automatic feature extractor and a multi-layer perceptron (MLP) as classifier. The method of using pre-trained deep network as part of another neural network to solve different (but related) task can be considered as transfer learning or self-taught learning [14].

Convolutional Neural Network

Convolutional network is a special kind of neural net- work optimized to learn representation of an image [7]. It introduces 2 new types of hidden layers: convolutional and subsampling/pooling layers. Each layer in convnet connects neurons (pixels) from their input layer in form of local receptives (square patches) through a shared weights to a feature map [8]. On top of a set of convolutional and pooling layers, some fully-connected layers are added as classifier as described by Figure 4.

$$y_i = \log(1 + \exp x_i) \qquad (1)$$

$$y_i = \frac{e^{xi}}{\sum_{k=1}^{K} e^{xk}} \qquad (2)$$

$$\text{for } i=1..K$$

$$r_j^x \sim \text{Bernoulli}(p) \qquad (3)$$

$$\tilde{y}_i = r_i * y_i$$

VGG16 is a very deep convnet model made by Visual Geometry Group (VGG), University of Oxford [9]. It was trained on 1,000,000 images dataset from ImageNet and achieve state-of-the-art results on Large-Scale Visual Recognition Challenge (ILSVRC) 2014. It contains 16 hidden layers composed of convolutional layers, max pooling layers and fully-connected layers as shown in Figure 5. The convolution and fully-connected layers uses ReLu activation function (Equation 1), except the output layer that uses a SoftMax activation function (Equation 2) to estimate probability of multiple classes/labels. Dropout is also used as regularization after each tanh fully-connected layers to avoid overfitting by randomly drop/turn off (set value to zero) hidden

TABEL 1
MODELS ACCURACY COMPARISON ON ROTATED TEST DATA

Model	Accuracy on Rotation			Average
	90	180	270	
SIFT LogReg	98.28	97.34	95.74	96.45
SURF MLP	96.81	96.81	96.28	96.63
VGG16 MLP	88.30	96.28	90.96	91.84

TABEL 2
MODELS ACCURACY COMPARISON ON SCALED TEST DATA

Model	Accuracy on Zoom-In			Average
	10%	30%	50%	
SIFT LogReg	98.40	93.62	89.89	93.97
SURF MLP	93.62	87.23	79.79	86.88
VGG16 MLP	96.28	88.83	81.91	89.01

nodes (Equation 3) [15].

Transfer Learning

Deep neural networks usually require a lot of training data to learn the representation of the data. In case there is not enough training data, there are several techniques to help neural networks model learns data representation using small training data. One of the technique is transferring knowledge of another pre-trained neural network model to our model. This technique is known as transfer learning or self- taught learning [14].

Our proposed model uses transferred knowledge (layer weights) from pre-trained VGG16 model (provided by deep learning framework Keras[1]) which was pre-trained using 1,000,000 images dataset from ImageNet. Intermediate outputs of VGG16 can be used to extract generic features for any image classifier [13]. Therefore, even though VGG16 was not designed to classify Batik patterns, it should be able to extract useful generic features from Batik images which can be used further for classification.

Compared to SIFT/SURF BoW, using pre-trained VGG16 allows us to reduce time needed to extract features because no training required for the feature extractor. Moreover, since our proposed model is based on neural network, execution time may also be reduced significantly by utilizing GPU parallelization.

To improve comprehension and reproducebility, the proposed model and experiment codes are also available in public online code repository[2]. This research also utilizes opensource TensorFlow-backed Keras as deep learning framework and scikit-learn[3] library for classification and evaluation to reduce amount of codes written so they can be easily studied further.

Experiments

To measure the performance of our model, we trained our model and compared it with SIFT and SURF based models.

The dataset used in the experiments originally comes from Machine Learning and Computer Vision (MLCV) Lab, Faculty of Computer Science, University of Indonesia. The original dataset consists of 603 Batik photos (± 78.3 MB) gathered from various sources thus having different size, quality and view angle. But based on the previous research, this dataset is expanded by equally slicing each image to four patches (Figure 6) for better accuracy [12]. Hence, the dataset used in the experiments contains 2,092 images (patches) of five classes: Ceplok (504 images), Kawung (368 images), Lereng (220 images), Nitik (428 images), Parang (572 images).

In the first experiment, the objectives are to compare our performance of SIFT, SURF and VGG16 extractors on the dataset by using the result to train and test six different classifiers: Logistic Regression, Support Vector Machine (RBF Kernel), Multi-Layer Perceptron (1 ReLU hidden layer, 100 nodes), Decision Tree, Gradient Boosting and Random Forest. The dataset is minimally preprocessed by converting each image to grayscale before processed by three extractors: (1) SIFT BoW extracts 2800 features, (2) SURF BoW extracts 2800 features, and (3) VGG16 extracts 512 features.

While, VGG16 extractor does not require any training because pre-trained model (weights) are used, the SIFT and SURF features extractors are trained using the best methods (achieving highest accuracy) described in previous research [3] (illustrated in Figure 7): (1) Image descriptors were extracted according to their feature extractor

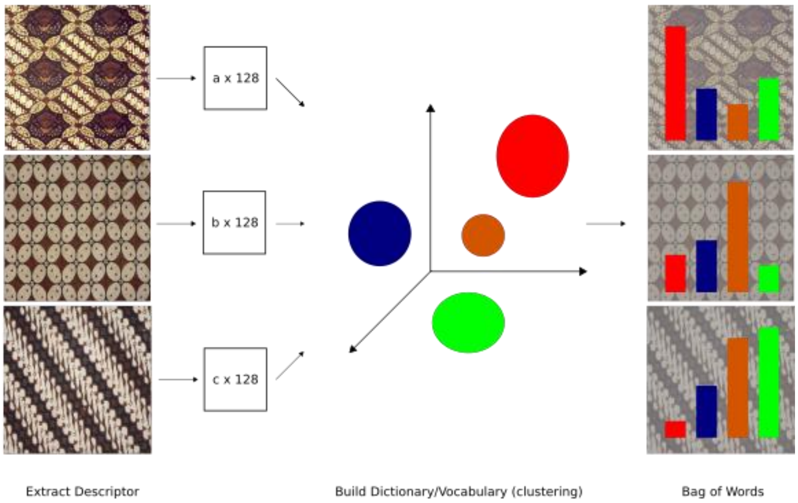

Figure 7. SIFT for building Bag of Words visual vocabularies.

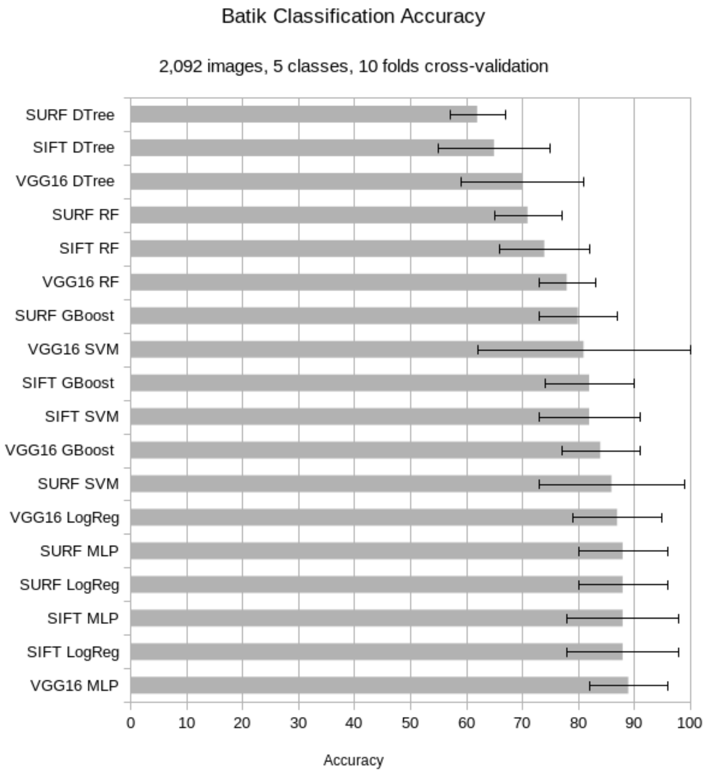

Figure 8. Model accuracy comparison

(SIFT or SURF), (2) Descriptors were clustered to 2800 clusters using K-Means to get visual vocabularies for BoW, and (3) Those 2800 visual vocalbularies then used to compute BoW features from SIFT/SURF image descriptors to produce 2800 features.

In the first experiment, each extracted feature is used to trains and tests six different classifiers mentioned above using 10 folds cross validation. So effectively, each classifier is trained using 1,883 images and tested using 209 images 10 times. The results are averaged and then compared to see which combination of extractor and classifier performs the best.

The best combinations of each three extractors from first experiment are tested for their capability to handle invariance dilemma in the second experiment. There are three steps of experiments. First step is each combinations of the best extractors-classifiers are trained using 2,092 images (without any transformation). Second step is a subset of 193 images are randomly chosen from dataset and transformed to six test datasets by applying six transformations: (1) 90 degrees rotation, (2) 180 degrees rotation, (3) 270 degrees rotation, (4) 10% zoom-in 1.1 scale, (5) 30% zoom-in 1.3 scale, and (6) 50% zoom-in 1.5 scale. The last step is each combinations of the extractors-classifiers are tested against those six transformed test datasets.

All experiments were conducted using Intel Core i7 5960X CPU, 66 GB RAM, NVIDIA GTX 980 4GB GPU, 240GB SSD, Debian 8 OS. The VGG16 extractor runs on GPU to reduce the execution time but the result should not be different than running it on CPU.

3. Results and Analysis

In the first experiment, the proposed model (VGG16 MLP), achieved slightly better (1%) accuracy and less deviation than the best SIFT and SURF models (SIFT LogReg and SURF MLP) as shown by chart in Figure 8. The average accuracy achieved by the proposed model is also ±8% better than Stacked-Autoencoder [12] that used dataset from same origin.

On general, the result also shows that VGG16 extractor performs as well as SIFT and SURF extractors despite of fewer features dimension (512 features against 2,800 features). Since VGG16 extractor does not require training, it is more efficient than SIFT/SURF BoW extractor. On top of that, neural network models such as VGG16 are known to run parallelly in GPU [16] to make it event more efficient.

It is also shown that decision-tree-based classifiers (Decision Tree, Random Forest and Gradient Boosting) generally achieve lower accuracy compared to non decision tree classifiers. Only SIFT Gradient Boosting and VGG Gradient Boosting which outperform SVM classifiers. This shows that the extracted features do not have nominal scale which is usually suitable with decision tree-based classifiers. Meanwhile, SVM, which represents non-linear classifier, is outperformed by logistic regression and single layer ReLU MLP that represent linear classifier. This result shows that the features extracted by SIFT, SURF and VGG16 are not linearly separable.

In the second experiment, the proposed model (VGG16 MLP) shows slightly less accurate results compared to SIFT and SURF models. For rotated and zoomed-in datasets, SIFT model is ±5% better than VGG16 model. While SURF model is ±5% better than VGG16 only on rotated datasets but ±3% worse than it. Despite of that, the accuracies of the proposed model are generally high (above 80%) and much better than self-trained Stacked-Autoencoder from previous research [12]. This shows that pre-trained VGG16 is able to handle invariance dilemma in Batik images almost as good as SIFT and SURF extractor.

4. Conclusion

Based on the experiment results and analysis, there are several points can be concluded: Pre-trained VGG16 extractor with MLP classi- fier slightly outperformed SIFT and SURF based models in term of accuracy for non-trans-formed dataset. Despite of not performing as good as SIFT and SURF models on transformed data-sets, it still achieves relatively high accuracy. This confirms that automatic feature extraction using pre-trained convolutional are able to handle transformation invariant features such as Batik motifs as good as SIFT and SURF as also concluded by related research [13].

Pre-trained VGG16 extractor is more efficient than SIFT and SURF bag of words (BoW) because it does not require any form of data fitting or training with Batik dataset. Meanwhile, SIFT/ SURF requires clustering of Batik SIFT/SURF descriptors in order to build visual vocabularies. On top of that, VGG16 extractor can be run parallely on GPU to further reduce execution time.

Features extracted by VGG16, SIFT and SURF do not scale like nominal data and are linearly separable. Hence, decision-tree-based (ID3, Gradient Boosting and Random Forest) and non-linear classifiers perform less accurate compared to linear classifiers (Logistic Regression and single hidden layer MLP).

There are also some aspects that can be explored to improve the research further: VGG16 is not the only pre-trained deep learning model available. So further research needs to compare performance of other pre-trained models such as VGG19 [9], Xception [17], ResNet50 [18] .etc on Batik datasets.

As majority of the data are mixed-motif Batik, each images should be classified to multiple classes at the same time (eg. Parang and Kawung). So current dataset should be relabeled to show the multi-label information of each Batik images.

Certain images in dataset often overlap each other (eg. Parang and Lereng motifs). This condition often confuses classifier during training and causes less accurate generalization. Therefore better (stricter) data labeling may further increase the accuracy of classification models.

Due to the various sources of data, the quality (resolution, noise, watermarks .etc) of the data are also various. Removing low quality data and preprocessing high quality ones may produce homogeneous data and improve classifier accuracy

References

[1] H. R. Sanabila and R. Manurung, “Recognition of batik motifs using the generalized hough transform,” University of Indonesia, 2009.

[2] H. Fahmi, R. A. Zen, H. R. Sanabila, I. Nurhaida, and A. M. Ary- murthy, “Feature selection and reduction for batik image retrieval,” in Proceedings of the Fifth International Conference on Network, Communication and Computing. ACM, 2016, pp. 47–52.

[3] R. Azhar, D. Tuwohingide, D. Kamudi, N. Suciati et al., “Batik image classification using sift feature extraction, bag of features and support vector machine,” Procedia Computer Science, vol. 72, pp. 24–30, 2015.

[4] I. Nurhaida, A. Noviyanto, R. Manurung, and A. M. Arymurthy, “Au- tomatic indonesian’s batik pattern recognition using sift approach,” Procedia Computer Science, vol. 59, pp. 567–576, 2015.

[5] D. Willy, A. Noviyanto, and A. M. Arymurthy, “Evaluation of sift and surf features in the songket recognition,” in Advanced Computer Science and Information Systems (ICACSIS), 2013 International Conference on. IEEE, 2013, pp. 393–396.

[6] V. S. Moertini and B. Sitohang, “Algorithms of clustering and clas- sifying batik images based on color, contrast and motif,” Journal of Engineering and Technological Sciences, vol. 37, no. 2, pp. 141–160, 2005.

[7] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[8] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[9] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[10] D. G. Lowe, “Distinctive image features from scale-invariant key- points,” International journal of computer vision, vol. 60, no. 2, pp. 91–110, 2004.

[11] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features,” in European conference on computer vision. Springer, 2006, pp. 404–417.

[12] R. A. Menzata, “Sistem perolehan citra berbasis konten dan klasifikasi citra batik dengan convolutional stacked autoencoder,” Universitas Indonesia, 2014.

[13] P. Fischer, A. Dosovitskiy, and T. Brox, “Descriptor matching with convolutional neural networks: a comparison to sift,” arXiv preprint arXiv:1405.5769, 2014.

[14] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, “Self-taught learning: transfer learning from unlabeled data,” in Proceedings of the 24th international conference on Machine learning. ACM, 2007, pp. 759–766.

[15] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting.” Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929–1958, 2014.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classifica- tion with deep convolutional neural networks,” in Advances in neural information processing systems, 2012, pp. 1097–1105.

[17] F. Chollet, “Xception: Deep learning with depthwise separable con- volutions,” arXiv preprint, 2016.

[18] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.