

Klasifikasi Jenis Batik Menggunakan Algoritma *Convolutional Neural Network*

Yesicha Amilia Putri^{*1}, Yufis Azhar², Agus Eko Minarno³

^{1,2,3}Universitas Muhammadiyah Malang

yesichaamilia@gmail.com^{*1}, yufis.az@gmail.com², agoes.minarno@gmail.com³

Abstrak

Beragamnya motif batik di Indonesia menyulitkan dalam pengenalan pola citra batik. Dalam hal ini diperlukannya klasifikasi citra batik untuk mengidentifikasi karakteristik citra. Dataset yang digunakan merupakan dataset batik berjumlah 300 data dengan 50 jenis batik dan setiap kategori terdapat 6 data. Algoritma yang digunakan dalam penelitian ini adalah *Convolutional Neural Network (CNN)*. *CNN* merupakan salah satu teknik *Deep Learning* yang memiliki hasil paling signifikan didalam pengenalan citra. Model *VGG16* digunakan pada pengujian untuk membandingkan akurasi yang diperoleh dan waktu yang dibutuhkan. Penelitian ini dilakukan untuk membuktikan model yang diusulkan mampu mengklasifikasi citra batik dengan baik. Hal ini dapat dibuktikan bahwa akurasi yang didapatkan menggunakan *CNN* sebesar 98% dan membutuhkan waktu lebih cepat dibandingkan model *VGG16*.

Kata Kunci: *Convolutional Neural Network, VGG16, Batik.*

Abstract

The diversity of batik motifs in Indonesia makes it difficult to recognize batik image patterns. In this case the classification of batik images is needed to identify the characteristics of the image. The dataset used is a batik dataset 300 data with 50 types of batik and each category contains 6 data. The algorithm used in this study is *Convolutional Neural Network (CNN)*. *CNN* is one of the *Deep Learning* techniques that has the most significant results in image recognition. The *VGG16* model is used in testing to compare the accuracy obtained and the time required. This research was conducted to prove the proposed model is able to classify the image of batik well. This can be proven that the accuracy obtained using *CNN* is 98% and requires faster time compared to the *VGG16* model.

Keywords: *Convolutional Neural Network, VGG16, Batik.*

1. Pendahuluan

Indonesia merupakan negara yang kaya akan sumber daya alam. Indonesia juga merupakan negara yang memiliki aneka ragam pulau, suku bangsa, bahasa dan budaya. Salah satu yang melekat dari Indonesia adalah batik. Batik sudah menjadi ciri khas dari negara Indonesia. Hal ini telah diakui oleh UNESCO pada tanggal 2 Oktober 2009 bahwa batik merupakan hak kebudayaan intelektual bangsa Indonesia[1]. Batik merupakan suatu pola yang unik dengan unsur ketradisional yang diterapkan pada kain yang diberi warna dan motif ciri khas dari Indonesia[2][3].

Beragam motif batik di Indonesia menyulitkan dalam pengenalan pola citra batik[4]. Dalam hal ini diperlukannya klasifikasi citra untuk mengidentifikasi karakteristik citra. Dalam klasifikasi citra terdapat metode lain seperti pada penelitian yang dilakukan oleh Yufis Azhar, Agus Eko Minarno, dan Yuda Munarko menggunakan metode *MTCD*. Penelitian tersebut menggunakan 2 dataset, yang pertama dataset citra batik sebanyak 300 data yang dibagi ke dalam 50 kategori. Yang kedua dataset *vacation*, dimana data tersebut merupakan kumpulan citra dari tempat-tempat liburan di seluruh dunia dengan menggunakan data sebanyak 812 data. Pengujian dilakukan untuk membandingkan waktu komputasi dan nilai presisi dari metode *MTCD*. Dataset dibagi menjadi data training dan data testing 10%. Dari hasil pengujian metode *MTCD* mampu menurunkan waktu komputasi hingga 16% untuk dataset *vacation*. Sedangkan untuk citra batik sebesar 4%. Pada penelitian tersebut menunjukkan bahwa metode yang digunakan dapat menurunkan waktu komputasi hingga 16% untuk dataset citra yang berdimensi besar. Tetapi dalam metode *MTCD* memiliki mekanisme yang kurang efektif karena sistem harus mengulang

proses konvolusi sebanyak *texton* yang digunakan. Hal tersebut berpengaruh cukup signifikan ketika data yang digunakan adalah citra yang berukuran besar[5].

MTCD merupakan salah satu metode untuk ekstraksi citra yang terdapat pada *Machine Learning*. Adapun kekurangan yang dimiliki oleh *Machine Learning* adalah ketika melakukan ekstraksi fitur dilakukan ekstraksi secara manual untuk menentukan *color*, *edge*, dan *texture*. Dari kekurangan tersebut dapat diperbaiki dengan *Deep Learning* menggunakan algoritma CNN. Algoritma ini tidak perlunya melakukan ekstraksi fitur *color*, *edge*, dan *texture* dikarenakan terdapat ekstraksi fitur tersendiri yang ada di CNN.

Algoritma CNN selain memiliki kelebihan juga terdapat kekurangan, salah satunya adalah algoritma ini hanya dapat digunakan pada data dua dimensi karena memiliki sifat konvolusi[6]. Dalam implementasi algoritma CNN, penelitian ini menggunakan *Keras* yang merupakan *open source network library* dalam *python*. Tidak perlunya menuliskan kode pada *Keras* dapat mempermudah dalam melakukan penelitian. Dalam melakukan pengujian, penelitian ini akan mencari nilai akurasi yang didapatkan.

Pada penelitian yang dilakukan I Wayan Suartika E. P, dkk tentang klasifikasi citra menggunakan Convolutional Neural Network (CNN) pada Caltech 101 membuktikan bahwa algoritma CNN terbukti berhasil mengungguli metode Machine Learning lain seperti SVM pada kasus klasifikasi objek citra yang dilakukan pada tahun 2012 oleh Alex Krizhevsky yang menerapkan algoritma CNN dan menjuarai kompetisi ImageNet Large Scale Visual Recognition Challenge 2012[7].

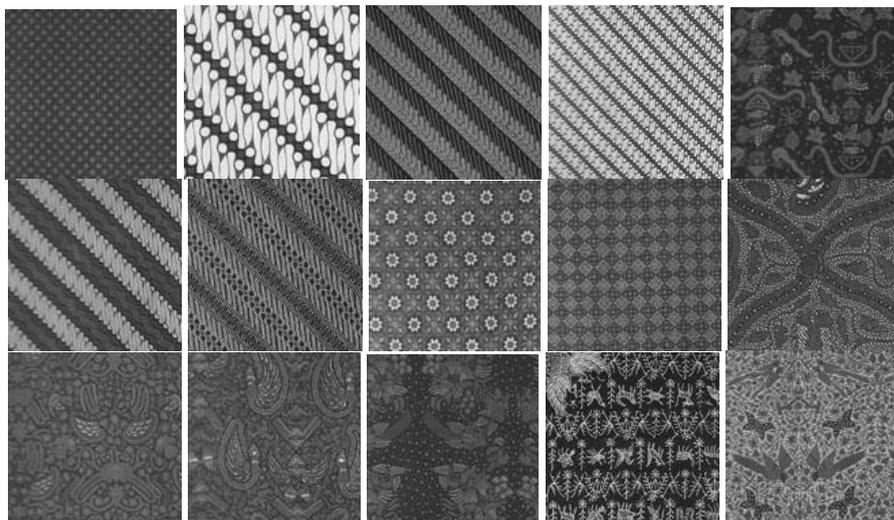
Selain model yang dibahas diatas, penelitian ini digunakan pada model VGG16. VGG16 merupakan model CNN yang diusulkan oleh K. Simonyan dan A. Zisserman dari Universitas Oxford[8]. Model ini dapat meningkatkan konvolusi yang sangat kecil dalam jaringan[9]. Terdapat penelitian klasifikasi batik menggunakan model VGG16 sebagai ekstraksi fitur. Akurasi yang didapatkan cukup tinggi sebesar 80%. Sehingga model VGG16 mampu mengklasifikasi citra batik dengan cukup baik[10].

Berdasarkan kelebihan dari algoritma CNN, adapun hipotesa dari penelitian ini adalah sistem klasifikasi batik yang dibangun menggunakan algoritma CNN berbasis *Keras* ini mampu menghasilkan sistem dengan nilai *precision*, *recall*, dan *f-measure* yang baik. Oleh karena itu dalam penelitian ini akan membuktikan model yang diusulkan mampu memperbaiki akurasi model VGG16.

2. Metode

2.1 Dataset

Dataset yang digunakan dalam penelitian ini adalah data citra batik yang berjumlah 300 data dengan 50 jenis batik dan setiap kategori terdapat 6 data. Dataset dibagi menjadi dua, yaitu data training sebanyak 250 data dan data testing sebanyak 50 data. Berikut beberapa contoh dataset, dapat dilihat pada Gambar 1.



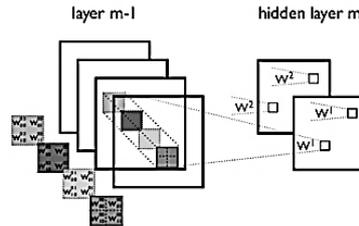
Gambar 1. Dataset Batik

2.2 Convolutional Neural Network

Convolutional Neural Network (CNN) merupakan salah satu metode *Deep Learning* dan pengembangan dari *Multilayer Perceptron* (MLP) yang didesain untuk mengolah data dua dimensi, sehingga bobot antar *neuron* berbeda[11]. Model standar dari CNN terdiri dari:

2.2.1 Convolutional Layer

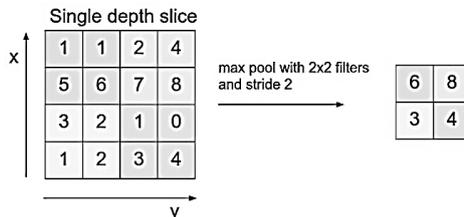
Proses utama CNN adalah konvolusi. Konvolusi merupakan pengaplikasian fungsi terhadap output fungsi lain secara berulang. *Convolutional layer* yang melakukan operasi konvolusi pada *output* dari *layer* sebelumnya[12]. Setiap lapisan yang dikonvolusi akan mengkonversi setiap *filter* ke bagian data masukan dan masukan tersebut akan menghasilkan *activation map* atau *feature map 2D*[7]. Proses konvolusi dapat dilihat pada ilustrasi Gambar 2.



Gambar 2. Convolutional Layer

2.2.2 Pooling Layer

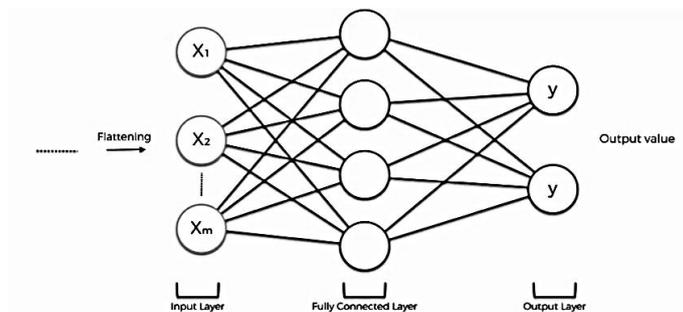
Tahap *pooling layer* merupakan tahapan setelah *convolutional layer* yang merupakan pengukuran matriks yang terdiri dari *filter* dengan ukuran *stride* tertentu[12]. Setiap adanya pergeseran akan ditentukan oleh jumlah dari *stride* yang digeser pada area *feature map* atau *activation map*[7]. Penggunaan fungsi *feature map* pada *pooling layer* merupakan tahapan yang penting dalam pembuatan model CNN. Bentuk dari lapisan *pooling* paling umum dengan menggunakan *filter* berukuran 2×2 yang diaplikasikan dengan langkah sebanyak 2 dan kemudian beroperasi pada setiap irisan dari *input*. Proses ini dapat dilihat pada ilustrasi Gambar 3.



Gambar 3. Pooling Layer

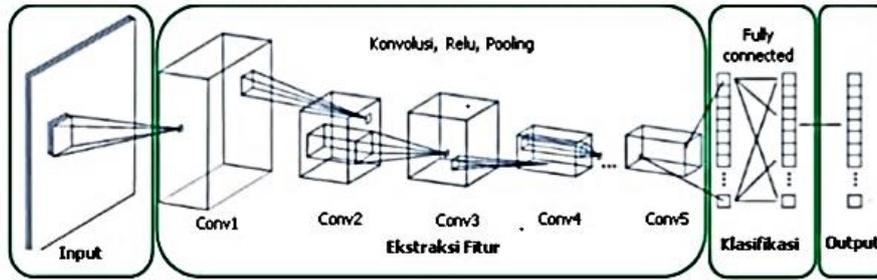
2.2.3 Fully Connected Layer

Tahap *fully connected layer* ini biasa digunakan untuk mengolah data, sehingga data tersebut dapat diklasifikasikan[13]. Output yang dihasilkan dari tahap *pooling* akan diproses melalui "*flatten*" atau *reshape* yang kemudian menghasilkan sebuah vektor yang akan digunakan sebagai inputan pada tahap *fully connected layer*[7]. Proses ini dapat dilihat pada ilustrasi Gambar 4.



Gambar 4. Fully Connected Layer

Selain tahapan yang terdapat dalam CNN, adapun arsitektur CNN yang dapat dilihat pada ilustrasi Gambar 5.



Gambar 5. Arsitektur CNN

3. Hasil dan Pembahasan

Terdapat 3 pengujian yang dilakukan pada penelitian ini. Pengujian menggunakan data test yang berbeda. Data test yang digunakan diambil satu citra dari setiap jenis batik. Pengujian 2 dilakukan dengan menggunakan model VGG16 dan pengujian 3 dilakukan dengan menghilangkan proses dropout.

3.1 Ekstraksi dan Pemodelan

Pada proses ini dilakukan penentuan nilai *filter*, *kernel*, fungsi aktivasi yang digunakan pada tahap ekstraksi dan pemodelan. Hasil yang diperoleh dari tahap ini dapat dilihat pada Gambar 6.

```

Model: "sequential_1"
-----
Layer (type)                Output Shape         Param #
-----
conv2d_1 (Conv2D)           (None, 32, 32, 32)  416
-----
max_pooling2d_1 (MaxPooling2 (None, 16, 16, 32)  0
-----
conv2d_2 (Conv2D)           (None, 16, 16, 50)  6450
-----
max_pooling2d_2 (MaxPooling2 (None, 8, 8, 50)  0
-----
conv2d_3 (Conv2D)           (None, 8, 8, 80)    16880
-----
max_pooling2d_3 (MaxPooling2 (None, 4, 4, 80)  0
-----
dropout_1 (Dropout)         (None, 4, 4, 80)    0
-----
flatten_1 (Flatten)         (None, 1280)         0
-----
dense_1 (Dense)              (None, 512)          655872
-----
dropout_2 (Dropout)         (None, 512)          0
-----
dense_2 (Dense)              (None, 50)           25650
-----
Total params: 704,468
Trainable params: 704,468
Non-trainable params: 0
-----
None
    
```

Gambar 6. Hasil Model

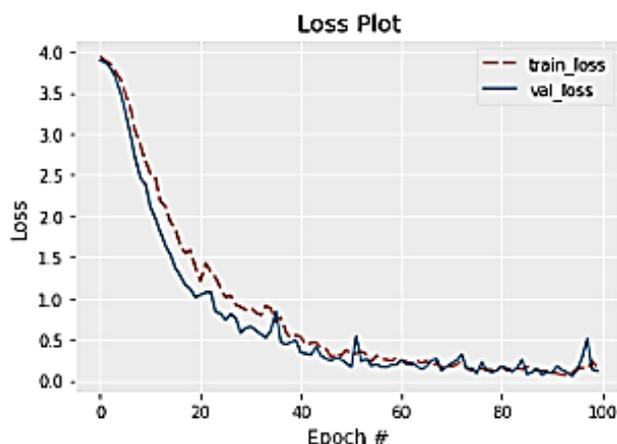
3.2 Pengujian 1

Pengujian 1 dilakukan untuk penggunaan data test yang berbeda. Pengujian ini dilakukan untuk menguji ketahanan sistem. Data test yang memiliki akurasi terbaik akan digunakan pada pengujian 2 dan 3. Berikut merupakan hasil dari pengujian 1, dapat dilihat pada Tabel 1.

Tabel 1 Hasil Pengujian Skenario 1

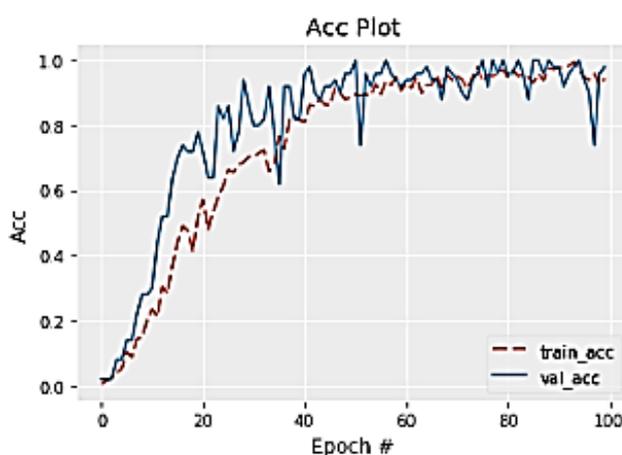
Uji	Train Acc	Test Acc	Test Loss	Train Time (s)
1	90%	89%	0.3494	20.3
2	94%	94%	0.2215	20.3
3	98%	98%	0.1157	20.3
4	96%	96%	0.1613	20.3
5	96%	96%	0.1097	20.3
6	98%	98%	0.1787	20.3

Tabel 1 menjelaskan tentang hasil pengujian yang telah dilakukan pada skenario pengujian 1. Waktu yang dibutuhkan pada pengujian data test yang berbeda tidak ada perubahan, dan akurasi yang didapatkan memiliki jarak yang tidak terlalu signifikan. Sehingga pada pengujian 1 dapat disimpulkan bahwa model CNN ini cukup tahan. Nilai akurasi tertinggi, yaitu uji 3 dan dapat dilihat nilai *loss* yang dihasilkan pada uji tersebut, dapat dilihat pada Gambar 7



Gambar 7. Nilai Loss Uji 3 Model CNN

Setelah mengetahui nilai *loss* yang dihasilkan pada uji 3. Berikut merupakan train akurasi dan val akurasi, dapat dilihat pada Gambar 8.



Gambar 8. Nilai Acc Uji 3 Model CNN

3.2 Pengujian 2

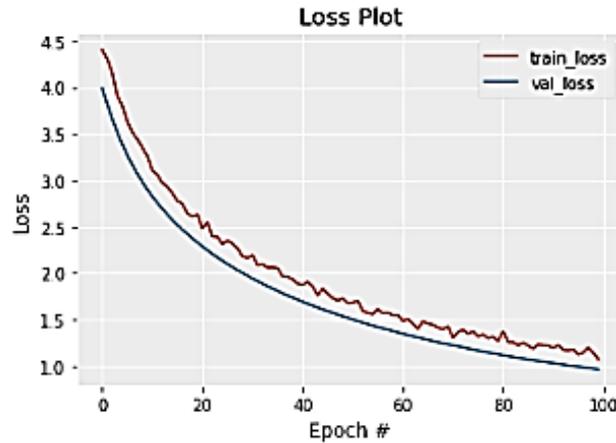
Pada pengujian 2 dilakukan pada model VGG16. Pengujian ini dilakukan untuk mengetahui apakah model yang diusulkan lebih baik dibandingkan model VGG16. Hasil pengujian 2 dapat dilihat pada Tabel 2.

Tabel 2. Hasil Pengujian Skenario 2

Uji	Train Acc	Test Acc	Test Loss	Train Time (s)
1	92%	91%	1.0176	90.2
2	98%	98%	0.9593	90.1
3	94%	94%	1.0087	130.1
4	96%	95%	1.0193	90.2
5	86%	86%	0.9967	90.2
6	94%	93%	0.9454	90.3

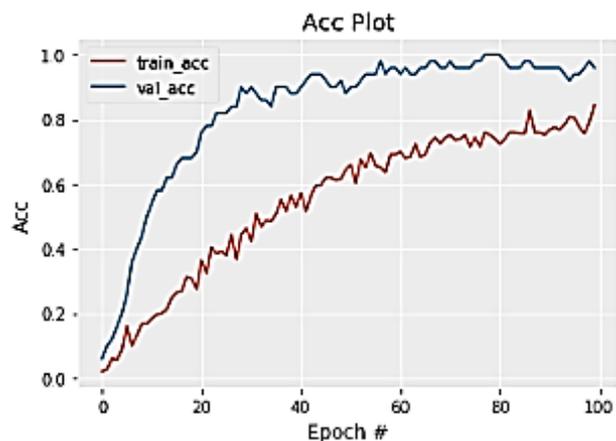
Tabel 2 menjelaskan tentang hasil pengujian 2. Pengujian dilakukan pada data test yang digunakan pada pengujian 1. Uji 2 memperoleh akurasi tertinggi, yaitu 98% dan membutuhkan

waktu paling cepat dibandingkan dengan data test yang lain. Nilai akurasi tertinggi, yaitu uji 2 dan dapat dilihat nilai *loss* yang dihasilkan pada uji tersebut, dapat dilihat pada Gambar 9.



Gambar 9. Nilai Loss Uji 2 Model VGG16

Setelah mengetahui nilai *loss* yang dihasilkan pada uji 2. Berikut merupakan train akurasi dan val akurasi, dapat dilihat pada Gambar 10.



Gambar 10. Nilai Acc Uji 2 Model VGG16

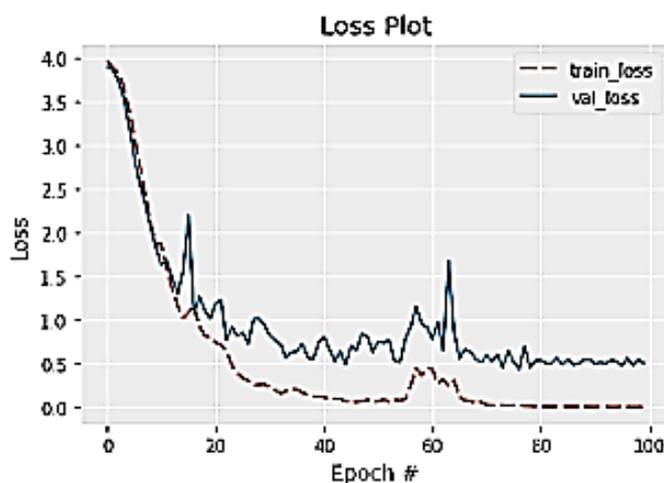
3.3 Pengujian 3

Pada pengujian 3 dilakukan dengan menghilangkan proses *dropout* yang terdapat pada tahap CNN. Pengujian ini dilakukan untuk mengetahui apakah proses *dropout* berpengaruh terhadap algoritma CNN. Hasil pengujian 3 dapat dilihat pada Tabel 3.

Tabel 3. Hasil Pengujian Skenario 3

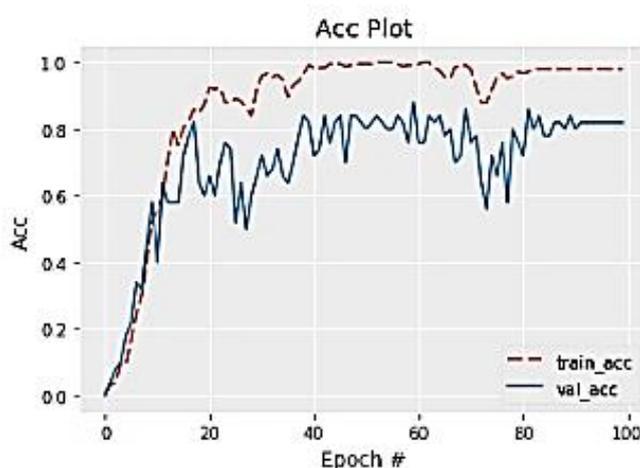
Uji	Train Acc	Test Acc	Test Loss	Train Time (s)
1	84%	83%	0.8337	20.3
2	98%	97%	0.1071	20.2
3	96%	96%	0.1611	20.3
4	80%	79%	0.5053	20.3
5	96%	96%	0.1330	20.3
6	96%	96%	0.1825	20.4

Tabel 3 menjelaskan tentang hasil pengujian 3. Pengujian dilakukan pada data test yang digunakan pada pengujian sebelumnya. Uji 4 memperoleh akurasi terendah, yaitu 80%. Dari hasil pengujian 3, akan menampilkan salah satu nilai *loss* yang dihasilkan pada uji 4 yang memiliki akurasi terendah. Berikut merupakan nilai *loss* uji 4 yang dapat dilihat pada Gambar 11.



Gambar 11. Nilai Loss Uji 4 Tanpa Dropout

Gambar 11 menjelaskan tentang hasil nilai *loss* uji 4 tanpa *dropout*. Dilihat dari *epoch* 10 sampai 65 nilai *loss* yang dihasilkan tidak stabil. Nilai test *loss* lebih tinggi daripada train *loss*. Hal ini memungkinkan adanya *overfitting*. Nilai *loss* telah diketahui, berikut merupakan nilai akurasi yang dihasilkan, dapat dilihat pada Gambar 12.



Gambar 12. Nilai Acc Uji 4 Tanpa Dropout

4. Kesimpulan

Berdasarkan pengujian yang telah dilakukan, model CNN terbukti mampu mengklasifikasi citra batik dengan baik dan membutuhkan waktu yang lebih singkat dibandingkan model lain. Hal ini dapat dibuktikan dengan menggunakan data test yang berbeda, nilai akurasi yang dihasilkan data test satu dengan data test yang lain memiliki jarak yang tidak terlalu signifikan dan membutuhkan waktu yang cukup stabil dari uji 1 sampai uji 6. Hal tersebut terbukti bahwa model CNN ini cukup tahan. Penggunaan model VGG16 menghasilkan akurasi yang cukup tinggi dan dapat mengimbangi dari model CNN. Akan tetapi model VGG16 membutuhkan waktu yang cukup lama dibandingkan model CNN. Penggunaan layer kompleks yang dimiliki oleh VGG16 tidak terbukti lebih baik dibandingkan model yang diusulkan. Pengujian tanpa *dropout* yang menghasilkan nilai *loss* tidak stabil, memungkinkan adanya *overfitting*. Sehingga untuk mencegah terjadinya *overfitting*, proses *dropout* perlu ditambahkan dalam model CNN. Akurasi yang didapatkan menggunakan CNN dengan menambahkan proses *dropout* mendapatkan nilai akurasi cukup tinggi sebesar 98%. Hal ini dapat disimpulkan bahwa model yang diusulkan sangat baik untuk mencari nilai performa dari sistem klasifikasi citra.

Referensi

- [1] J. W. Yodha and A. W. Kurniawan, "Pengenalan Motif Batik Menggunakan Deteksi Tepi Canny Dan K-Nearest Neighbor," *J. Techno.COM*, Vol. 13, No. 4, Novemb. 2014 251-262, vol. 13, no. 4, pp. 251–262, 2014.
- [2] I. Nurhaida, A. Noviyanto, R. Manurung, and A. M. Arymurthy, "Automatic Indonesian's Batik Pattern Recognition Using SIFT Approach," *Procedia Comput. Sci.*, vol. 59, no. Iccsci, pp. 567–576, 2015.
- [3] D. Yulianti, "Mengungkap Sejarah Dan Motif," *Paramita*, vol. 20, no. 1, pp. 11–20, 2010.
- [4] A. A. Utama, R. Efendi, D. Andreswari, and J. Rekursif, "Klasifikasi Motif Batik Besurek Menggunakan Metode Rotated Haar Wavelet Transformation Dan," *Rekursif*, pp. 161–175, 2016.
- [5] Y. Azhar, A. E. Minarno, and Y. Munarko, "Optimasi Deteksi Texton Pada Metode Multi Texton Co - Occurrence Descriptor Untuk Image Retrieval," pp. 34–38, 2018.
- [6] M. M. Susilo, D. M. Wonohadidjojo, and N. Sugianto, "Pengenalan Pola Karakter Bahasa Jepang Hiragana Menggunakan 2D Convolutional Neural Network," *J. Inform. dan Sist. Inf. Univ. Ciputra*, vol. 03, no. 02, pp. 28–36, 2017.
- [7] I. W. S. E. Putra, "Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) Pada Caltech 101," *J. Tek. ITS*, vol. 5, no. 1, p. 76, 2016.
- [8] Q. Guan *et al.*, "Deep convolutional neural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: A pilot study," *J. Cancer*, vol. 10, no. 20, pp. 4876–4882, 2019.
- [9] Z. Yang, J. Yue, Z. Li, and L. Zhu, "Vegetable Image Retrieval with Fine-tuning VGG Model and Image Hash," *IFAC-PapersOnLine*, vol. 51, no. 17, pp. 280–285, 2018.
- [10] Y. Gultom, A. M. Arymurthy, and R. J. Masikome, "Batik Classification using Deep Convolutional Network Transfer Learning," *J. Ilmu Komput. dan Inf.*, vol. 11, no. 2, p. 59, 2018.
- [11] Sam'ani and M. H. Qamaruzzaman, "Pengenalan Huruf Dan Angka Tulisan Tangan Menggunakan Metode Convolution Neural Network (CNN)," *J. Speed – Sentra Penelit. Eng. dan Edukasi*, vol. 9, no. 2, pp. 55–64, 2017.
- [12] G. Santoso, Aditya and Ariyanto, "Implementasi Deep Learning Berbasis Keras Untuk Pengenalan Wajah," *Emit. J. Tek. Elektro*, vol. 18, no. 01, pp. 15–21, 2018.
- [13] G. Lin and W. Shen, "Research on convolutional neural network based on improved Relu piecewise activation function," *Procedia Comput. Sci.*, vol. 131, pp. 977–984, 2018.