

Deteksi dan Pengenalan Jenis Corak Batik Nusantara Menggunakan Metode CNN Berbasis Android

Faiq Fahrian Khoirul Anam Al Aziz¹, Saefurrohman²

Prodi Teknik Informatika, Fakultas Teknologi Informasi & Industri Universitas Stikubank Semarang, Jl. Tri Lomba Juang No 1 Mugas Semarang, Indonesia, 50241

e-mail: ¹faiqfahriankhoirulanamalaziz@mhs.unisbank.ac.id, ²saefurr@edu.unisbank.ac.id

Submitted Date: April 14th, 2023
Revised Date: April 26th, 2023

Reviewed Date: April 21st, 2023
Accepted Date: April 30th, 2023

Abstract

This research aims to develop a system for the detection and recognition of traditional Indonesian batik patterns using Convolutional Neural Network (CNN) method based on the Android platform, utilizing TensorFlow Lite as the framework. The research is motivated by the importance of preserving and promoting the cultural heritage of batik patterns, which represent the distinctive and treasured cultural identity of Indonesia. The methodology involves training a CNN model using a dataset consisting of 500 images of various batik patterns from 10 different types of batik. The dataset is divided into training and testing data in an 80:20 ratio. The results of the research indicate that the developed model for the detection and classification of batik patterns achieves high accuracy, with a training data accuracy of 92.25% and a testing data accuracy of 94%. In conclusion, the research demonstrates that the developed model is capable of accurately recognizing and detecting various traditional batik patterns. The research has practical benefits for the public, as it enhances knowledge and understanding of different batik patterns. Furthermore, the research contributes to the advancement of knowledge and the researcher's proficiency in implementing the CNN method for real-time detection of batik patterns.

Keywords: Batik; Detection; CNN; Android

Abstrak

Penelitian ini bertujuan untuk mengembangkan sistem deteksi dan pengenalan jenis corak batik Nusantara menggunakan metode Convolutional Neural Network (CNN) berbasis Android dengan menggunakan TensorFlow Lite sebagai framework. Penelitian ini didasari oleh pentingnya pelestarian dan pengenalan warisan budaya corak batik yang merupakan ciri khas dan pusaka budaya Indonesia. Metode penelitian melibatkan penggunaan CNN yang telah dilatih menggunakan dataset berjumlah 500 citra corak batik dari 10 jenis batik yang berbeda. Dataset ini dibagi menjadi data training dan data testing dengan perbandingan 80:20. Hasil penelitian menunjukkan bahwa model deteksi dan klasifikasi corak batik memiliki akurasi tinggi, dengan akurasi data training sebesar 92,25% dan akurasi data testing sebesar 94%. Penelitian dapat disimpulkan bahwa model yang dikembangkan mampu mengenali dan mendeteksi jenis corak batik Nusantara dengan akurasi yang signifikan. Penelitian ini memiliki manfaat bagi masyarakat dalam meningkatkan pengetahuan dan pemahaman mengenai berbagai jenis corak batik. Selain itu, penelitian ini juga memberikan kontribusi dalam pengembangan pengetahuan dan kemampuan peneliti dalam mengimplementasikan metode CNN untuk deteksi corak batik secara real-time.

Kata kunci: Batik; Deteksi; CNN; Android

1 Pendahuluan

Batik memiliki nilai warisan budaya yang khas bagi Indonesia dan masyarakatnya. Batik (dari

kata "ambatik") memiliki asal-usul dari bahasa Jawa yang terdiri dari "amba" yang berarti menulis dan "nitik" yang berarti kain dengan noktah kecil.

Seni batik melibatkan proses melukis di atas kain dengan menggunakan lilin atau malam sebagai pelindung untuk menciptakan ragam hiasan di permukaan kain. Istilah "batik" juga mencerminkan nilai-nilai kearifan yang tercermin dalam tradisi dan budaya, dengan hiasan dan lukisan batik yang mengandung makna dan ornamen yang menarik sesuai dengan motif dan gaya yang berkembang di Indonesia. Batik telah diakui oleh UNESCO sebagai warisan dunia pada 2 Oktober 2009 (Saefurrohman & Ningsih, 2016).

Selain itu, kain batik sering digunakan dalam acara-acara pesta sehari-hari di Indonesia. Batik dianggap sebagai pakaian tradisional yang identik dengan masyarakat Indonesia. Pada dasarnya, seni membatik melibatkan proses menghias kain dengan menggunakan lilin sebagai media. Teknik ini melibatkan pengaplikasian lapisan lilin pada bagian-bagian tertentu kain, yang kemudian dicelupkan dalam pewarnaan. Inilah yang membuat batik begitu istimewa. Seni membatik melibatkan penggunaan lilin untuk menghasilkan desain kain dengan lapisan lilin yang ditutupi. Keunikan batik terletak pada setiap motif yang memiliki makna filosofis dan nilai sejarah. Panjang, corak, dan motif batik tidak terlepas dari daerah asalnya (Lestari, 2012).

Kemajuan teknologi telah memiliki dampak yang signifikan dalam kehidupan manusia, termasuk dalam bidang seni. Dalam konteks batik, teknologi dapat berperan penting dalam membantu manusia mengenali berbagai jenis corak batik tradisional di Nusantara. Motif-motif batik ini umumnya mencerminkan alam dan budaya dari daerah asalnya, dan sering kali mengandung makna filosofis yang melambangkan nilai-nilai masyarakat setempat. Sebagai hasilnya, setiap daerah memiliki variasi corak dan warna yang unik, dan batik menjadi warisan budaya yang berharga yang perlu dijaga kelestariannya (Putri & Rochmawati, 2019).

Pemanfaatan teknologi dalam pelestarian budaya dapat memainkan peran penting dalam menjembatani masa lalu dengan perkembangan zaman. Salah satu cara yang digunakan untuk mengenali jenis corak batik secara akurat adalah melalui penggunaan sistem deteksi menggunakan metode Convolutional Neural Network (CNN), yang merupakan salah satu metode terkini dalam bidang Machine Learning. Dengan memanfaatkan teknologi TensorFlow, sebuah *library* yang mampu

mendeteksi objek menggunakan kamera *smartphone*, manusia dapat dibantu dalam mengenali dan memahami corak batik secara lebih baik. CNN memiliki kemampuan untuk mengidentifikasi berbagai fitur dalam gambar digital, seperti perbedaan warna, tepi, dan sudut. Proses deteksi objek dalam metode ini memerlukan estimasi lokasi yang akurat agar dapat mengklasifikasikan objek dengan tingkat akurasi yang tinggi (Bariyah, Rasyidi & Ngatini, 2021).

Selain itu, dengan pesatnya perkembangan teknologi mobile, khususnya *smartphone*, telah membawa kemudahan dalam kehidupan sehari-hari. Sistem berbasis Android menjadi populer di kalangan masyarakat karena sifatnya yang *open source* dan memberikan kesempatan pengembangan. Dalam konteks ini, aplikasi berbasis Android dapat menjadi sarana yang mempermudah masyarakat dalam mengakses informasi secara luas (Alfikri, Utomo, Februriyanti, & Nurwahyudi, 2022)..

Berdasarkan uraian sebelumnya, peneliti bermaksud untuk mengembangkan sebuah "Sistem Deteksi dan Pengenalan Jenis Corak Batik Nusantara Dengan Menggunakan Metode *Convolutional Neural Network* (CNN) Berbasis Android". Tujuan dari sistem ini adalah untuk memudahkan masyarakat dalam mengenali berbagai jenis corak batik dengan akurasi yang tinggi.

2 Landasan Teori

2.1 Batik

Batik adalah seni tradisional dalam pembuatan kain yang dihasilkan melalui proses pewarnaan dengan menggunakan lilin untuk melindungi bagian-bagian tertentu dari kain agar tidak terkena pewarnaan. Teknik ini menghasilkan pola dan corak yang khas pada kain. Batik merupakan bagian penting dari warisan budaya Indonesia dan banyak dikenal sebagai karya seni yang indah dan bernilai tinggi. Motif-motif batik bervariasi dari daerah ke daerah, dengan setiap motif memiliki makna, filosofi, dan cerita yang terkait dengan budaya setempat. Batik juga sering digunakan sebagai simbol identitas nasional Indonesia (Agustin, 2014).

2.2 Artificial Intelligence

Kecerdasan Buatan atau *Artificial Intelligence* (AI) mengacu pada pengembangan

sistem komputer atau mesin yang dapat melakukan tugas-tugas yang biasanya membutuhkan kecerdasan manusia. AI melibatkan studi dan pembuatan agen cerdas yang dapat menafsirkan lingkungan mereka, berpikir secara logis, belajar, dan mengambil keputusan atau tindakan untuk mencapai tujuan tertentu. Teknologi AI mencakup berbagai bidang dan metode seperti *machine learning*, *deep learning*, dan banyak lagi (Savitri, 2019).

2.3 Machine Learning

Machine Learning, juga dikenal sebagai pembelajaran mesin, adalah pendekatan dalam bidang kecerdasan buatan yang digunakan secara luas untuk meniru atau menggantikan perilaku manusia dalam menyelesaikan masalah atau melakukan otomatisasi. Prinsip utama *Machine Learning* adalah meniru proses pembelajaran dan generalisasi yang dilakukan oleh manusia atau makhluk cerdas lainnya. Dalam *Machine Learning*, terdapat dua aplikasi utama yang sering digunakan, yaitu identifikasi dan prediksi (Zindani, Amalia, & Putro, 2020).

2.4 Deep Learning

Deep Learning adalah subbidang dari *Machine Learning* yang menggunakan jaringan saraf tiruan berstruktur untuk memproses informasi. Metode ini bertujuan untuk meniru fungsi dan proses yang terjadi di dalam otak manusia dengan menggunakan arsitektur jaringan saraf yang dalam (*deep neural networks*) (Shafira, 2018).

2.5 Citra Digital

Gambar digital direpresentasikan sebagai fungsi dua dimensi, di mana setiap piksel pada gambar memiliki koordinat spasial (x, y) yang menentukan posisinya dalam matriks. Amplitudo fungsi pada setiap pasangan titik (x, y) menggambarkan intensitas atau kecerahan gambar pada titik tersebut (Gonzales, 2004).

2.6 Convolution Neural Network

Convolutional Neural Network (CNN) adalah metode *neural network* yang populer untuk memproses data berdimensi tinggi, seperti gambar dan video. Perbedaannya dengan *neural network* lainnya terletak pada penggunaan kernel 2 dimensi atau dimensi tinggi yang melakukan konvolusi pada setiap unit dalam lapisan CNN. Kernel ini

menggabungkan fitur spasial dengan bentuk spasial yang mirip dengan media input. Dengan menggunakan parameter yang tepat, CNN dapat mengurangi kompleksitas data dan mengidentifikasi pola atau fitur yang relevan secara efisien. Hal ini memungkinkan CNN mengenali objek atau pola dalam gambar atau video dengan akurasi tinggi (Khan, 2018).

2.7 Google Colab

Google Colab adalah sebuah platform populer untuk pengembangan dan penelitian dalam komputasi dalam jaringan (*cloud computing*). Platform ini memungkinkan pengguna untuk menulis dan menjalankan kode *Python*, melakukan analisis data, serta menjalankan model *machine learning* dan *deep learning* dengan menggunakan sumber daya komputasi yang disediakan oleh *Google* (Arifianto, 2022).

2.8 Tensorflow

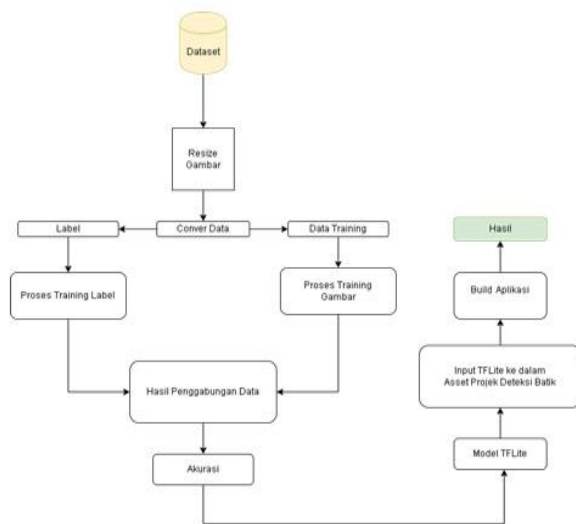
TensorFlow adalah sebuah *framework* sumber terbuka yang digunakan untuk kecerdasan buatan dan *machine learning*. *Framework* ini dikembangkan oleh *Google* dan banyak digunakan dalam pengembangan model dan aplikasi *machine learning*. *TensorFlow* menyediakan alat dan lingkungan yang kuat untuk membangun dan melatih model *machine learning* secara efisien. Dengan menggunakan *TensorFlow*, pengembang perangkat lunak dapat memanfaatkan berbagai teknik *deep learning*, seperti *neural networks*, untuk memproses dan menganalisis data dengan kemampuan yang sangat kuat (Abadi, 2016).

2.9 Android Studio

Android Studio adalah *Integrated Development Environment* (IDE) yang dirancang khusus untuk pengembangan aplikasi *Android*. Dengan *Android Studio*, pengembang perangkat lunak dapat membuat, menguji, dan mendistribusikan aplikasi *Android* dengan lebih efisien. IDE ini menyediakan berbagai fitur seperti editor kode yang canggih, pengujian aplikasi, pemecahan masalah, serta integrasi dengan *Android SDK* yang memungkinkan pengembang untuk mengakses berbagai alat dan sumber daya yang diperlukan dalam pengembangan aplikasi *Android* (Maiyana, 2018).

3 Metode Penelitian

3.1 Tahapan Penelitian



Gambar 1 Diagram alur tahap penelitian

Peneliti mengumpulkan dataset yang telah disiapkan untuk penelitian deteksi batik yang terdiri dari 10 jenis motif batik dengan jumlah 500 citra batik dengan perbandingan rasio 80% data *training* dan 20% data testing. Setelah itu, gambar-gambar dalam dataset akan melalui proses *resizing*, yaitu mengubah ukuran dataset menjadi 224x224 agar sesuai dengan kebutuhan model yang akan digunakan. Data yang telah diubah ukurannya kemudian dikonversi ke format yang sesuai untuk diproses lebih lanjut.

Selanjutnya, dataset akan melewati proses pelabelan (*labeling*). Hal ini dilakukan untuk memberikan informasi tentang kategori atau klasifikasi pada setiap gambar dalam dataset. Dataset kemudian dibagi menjadi dua bagian: data pelatihan (*training data*) dan data uji (*test data*). Data pelatihan akan digunakan dalam proses pelatihan model deteksi batik, sementara data uji digunakan untuk menguji performa model yang telah dilatih. Proses pelatihan pada data pelatihan melibatkan penggunaan metode *Convolutional Neural Network* (CNN).

CNN adalah jenis algoritma yang sering digunakan dalam pengolahan gambar dan pemrosesan citra. Setelah melalui proses pelatihan, data dari proses pelatihan dan hasil penggabungan data akan dievaluasi menggunakan metrik akurasi dan mungkin juga metrik lainnya untuk mengukur

sejauh mana model deteksi batik tersebut efektif dan akurat. Jika hasil evaluasi memuaskan, model yang telah dilatih akan diubah menjadi format *TFLite*.

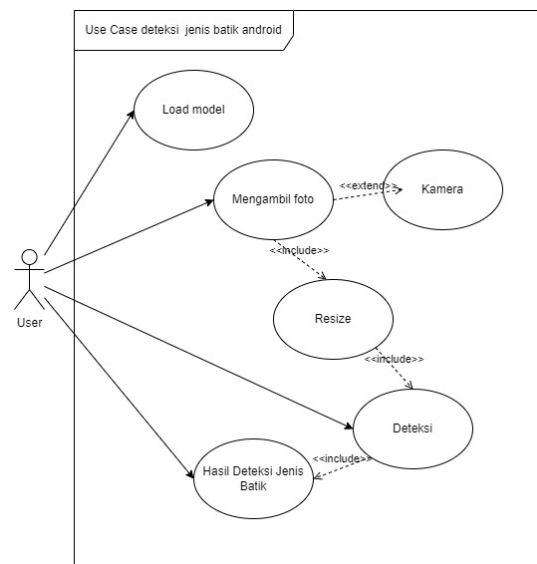
TFLite adalah format yang dioptimalkan khusus untuk menjalankan model *TensorFlow* pada perangkat mobile dan perangkat terbatas lainnya. Setelah proses input data selesai, peneliti akan membangun aplikasi dengan menggunakan *Android Studio*. Akhirnya, hasil dari aplikasi tersebut akan tersedia.

3.2 Perancangan Sistem

Tahap perancangan sistem memiliki peranan penting dalam pengembangan sistem aplikasi, dimana tujuannya adalah untuk merancang secara rinci arsitektur dan fungsionalitas sistem. Untuk membantu visualisasi interaksi antara pengguna dan sistem, serta struktur internal sistem, seringkali digunakan diagram seperti *use case*, *activity*, *sequence*, dan *class*. Dengan menggunakan diagram tersebut, dapat meminimalisir risiko plagiarisme dengan tetap memberikan pemahaman yang jelas tentang pentingnya tahap perancangan sistem dalam pengembangan aplikasi.

Rancangan sistem yang dibuat antara lain:

1. *Use Case Diagram*



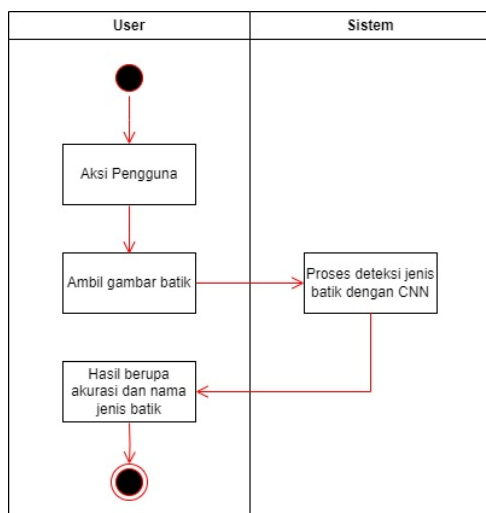
Gambar 2 Usecase diagram

Gambar 2 menjelaskan aplikasi deteksi dan pengenalan jenis corak batik nusantara berbasis *Android* menggunakan metode CNN. Diagram ini memberikan gambaran tentang interaksi antara

pengguna (*user*) dan sistem melalui serangkaian use case seperti user, loadmodel, mengambil foto, kamera, *resize*, deteksi, dan hasil deteksi jenis batik.

2. Activity Diagram

Activity diagram menggambarkan langkah-langkah yang dilakukan oleh pengguna dan sistem dalam proses deteksi dan pengenalan jenis corak batik nusantara menggunakan metode CNN. Diagram dimulai dengan pengguna mengambil gambar corak batik. Setelah itu, sistem melakukan proses klasifikasi menggunakan metode CNN untuk mengidentifikasi jenis batik yang ada. Hasil deteksi, termasuk jenis batik yang terdeteksi dan tingkat akurasi, dikirimkan kembali kepada pengguna. Diagram ini memberikan gambaran visual dan jelas mengenai alur proses dalam deteksi jenis corak batik nusantara dengan menggunakan metode CNN.

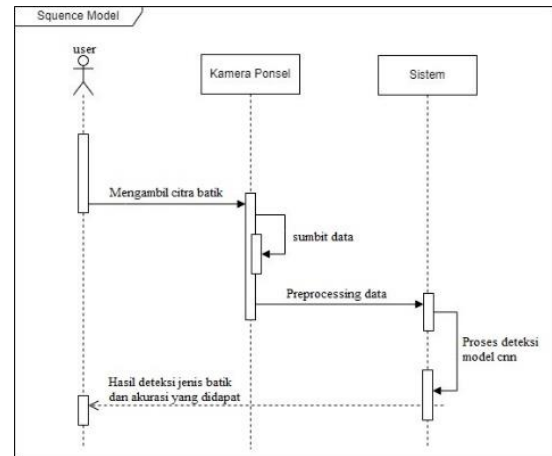


Gambar 3 Activity diagram

3. Sequence Diagram

Sequence diagram memberikan gambaran visual dan terstruktur mengenai interaksi antara aktor-aktor yang terlibat dalam proses deteksi dan pengenalan jenis corak batik menggunakan metode CNN. proses dimulai dengan pengguna mengambil foto corak batik. Sistem kemudian memproses data gambar tersebut dengan menggunakan metode CNN untuk mengidentifikasi jenis batik. Proses deteksi melibatkan tahapan pra-pemrosesan gambar dan penerapan model CNN. Hasil

identifikasi deteksi kemudian dikirimkan kembali ke aplikasi dan ditampilkan kepada pengguna.



Gambar 4 Sequence diagram deteksi corak batik

4 Hasil dan Pembahasan

Berikut merupakan hasil dari perancangan dan deteksi jenis batik:

a. Hasil Pelatihan Model

```

model.summary()

Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
conv2d (Conv2D)              (None, 222, 222, 32)        896
max_pooling2d (MaxPooling2D) (None, 111, 111, 32)        0
conv2d_1 (Conv2D)            (None, 109, 109, 64)        18496
max_pooling2d_1 (MaxPooling2D) (None, 54, 54, 64)        0
conv2d_2 (Conv2D)            (None, 52, 52, 128)        73856
max_pooling2d_2 (MaxPooling2D) (None, 26, 26, 128)        0
flatten (Flatten)            (None, 86528)               0
dense (Dense)                 (None, 256)                 22151424
dense_1 (Dense)              (None, 10)                  2570
-----
Total params: 22,247,242
Trainable params: 22,247,242
Non-trainable params: 0
    
```

Gambar 5 Model *sequential cnn*

Gambar 6 menunjukkan model *sequential*. Model ini terdiri dari beberapa lapisan yang berbeda.

1. Lapisan pertama, *Convolutional Layer 1* menggunakan operasi konvolusi dengan 32

filter berukuran 3x3. Dalam lapisan ini, Setiap filter memiliki $3 \times 3 \times 3 = 27$ parameter (3x3 untuk ukuran kernel dan 3 untuk saluran warna pada input). Selain itu, terdapat 32 parameter bias, satu untuk setiap filter. Oleh karena itu, total jumlah parameter yang dapat diubah dalam *Convolutional Layer 1* adalah $32 \times 27 + 32 = 896$. Inputnya adalah gambar dengan dimensi $224 \times 224 \times 3$, dan hasil konvolusinya menghasilkan gambar dengan dimensi $222 \times 222 \times 32$. Kemudian, *Max Pooling Layer 1* melakukan operasi *max pooling* dengan *pool size 2x2* pada hasil dari *Convolutional Layer 1*, menghasilkan gambar dengan dimensi $111 \times 111 \times 32$.

- Lapisan kedua, *Convolutional Layer 2* menggunakan operasi konvolusi dengan 64 filter berukuran 3x3. Setiap filter memiliki $3 \times 3 \times 32 = 288$ parameter (3x3 untuk ukuran kernel dan 32 untuk saluran warna pada input). Selain itu, terdapat 64 parameter bias, satu untuk setiap filter. Oleh karena itu, total jumlah parameter yang dapat diubah dalam *Convolutional Layer 2* adalah $64 \times 288 + 64 = 18,496$. Inputnya adalah gambar dengan dimensi $111 \times 111 \times 32$ (hasil dari *Max Pooling Layer 1*), dan hasil konvolusinya menghasilkan gambar dengan dimensi $109 \times 109 \times 64$. Setelah itu, *Max Pooling Layer 2* melakukan operasi *max pooling* dengan *pool size 2x2* pada hasil dari *Convolutional Layer 2*, menghasilkan gambar dengan dimensi $54 \times 54 \times 64$.
- Convolutional Layer 3* menggunakan operasi konvolusi dengan 128 filter berukuran 3x3. Setiap filter akan memiliki $3 \times 3 \times 64 = 576$ parameter (3x3 untuk ukuran kernel dan 64 untuk saluran warna pada input). Selain itu, terdapat 128 parameter bias, satu untuk setiap filter. Oleh karena itu, total jumlah parameter yang dapat diubah dalam *Convolutional Layer 3* adalah $128 \times 576 + 128 = 73,856$. Inputnya adalah gambar dengan dimensi $54 \times 54 \times 64$ (hasil dari *Max Pooling Layer 2*), dan hasil konvolusinya menghasilkan gambar dengan dimensi $52 \times 52 \times 128$. Dilanjutkan dengan *Max Pooling Layer 3* yang melakukan operasi *max pooling* dengan *pool size 2x2* pada hasil dari *Convolutional Layer 3*, menghasilkan gambar dengan dimensi $26 \times 26 \times 128$.
- Selanjutnya, *Flatten Layer* mengubah gambar 3D menjadi vektor 1D dengan panjang 86,528.

- Dense Layer 1* merupakan *fully connected layer* dengan 256 neuron, di mana setiap neuron terhubung dengan setiap neuron pada *Flatten Layer*. Jumlah parameter dalam *Dense Layer 1* adalah 22,151,424.
- Terakhir, *Dense Layer 2* juga merupakan *fully connected layer* dengan 10 neuron, di mana setiap neuron terhubung dengan setiap neuron pada *Dense Layer 1*. 10 neuron menunjukkan banyaknya kategori gambar yang digunakan yaitu Batik Betawi, Batik Ciamis, Batik Garutan, Batik Kawung, Batik Lasem, Batik Pekalongan, Batik Priangan, Batik Sidomukti, Batik Sogan, dan Batik Tambal. Jumlah parameter dalam *Dense Layer 2* adalah 2,570. Secara total, model ini memiliki 22,247,242 parameter.

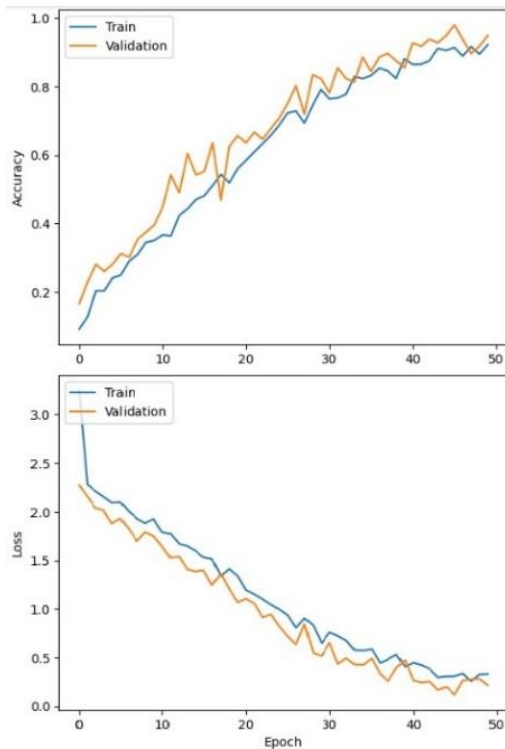
Peneliti membuat model dengan menggunakan *epoch 50*, *batch_size 32*, *learning rate 0,001* dengan menggunakan *validation_split* sebesar 0,2 artinya 80% data digunakan untuk pelatihan dan 20% data digunakan untuk validasi. Pembuatan model menggunakan *google colab*. Berikut hasil perhitungan *model fit*:

```
Epoch 1/50  
12/12 [=====] - 142s 11s/step - loss: 3.2457 - accuracy: 0.0934 - val_loss: 2.2734 - val_accuracy: 0.3667  
Epoch 2/50  
12/12 [=====] - 64s 5s/step - loss: 2.2798 - accuracy: 0.1277 - val_loss: 2.1564 - val_accuracy: 0.2292  
Epoch 3/50  
12/12 [=====] - 68s 6s/step - loss: 2.2133 - accuracy: 0.2038 - val_loss: 2.0491 - val_accuracy: 0.2882  
Epoch 4/50  
12/12 [=====] - 63s 5s/step - loss: 2.1564 - accuracy: 0.2038 - val_loss: 2.0148 - val_accuracy: 0.2684  
Epoch 5/50  
12/12 [=====] - 63s 5s/step - loss: 2.0944 - accuracy: 0.2418 - val_loss: 1.8796 - val_accuracy: 0.2882  
Epoch 46/50  
12/12 [=====] - 63s 5s/step - loss: 0.3084 - accuracy: 0.9130 - val_loss: 0.1196 - val_accuracy: 0.9792  
Epoch 47/50  
12/12 [=====] - 62s 5s/step - loss: 0.3386 - accuracy: 0.8886 - val_loss: 0.2636 - val_accuracy: 0.9375  
Epoch 48/50  
12/12 [=====] - 61s 5s/step - loss: 0.2574 - accuracy: 0.9158 - val_loss: 0.2768 - val_accuracy: 0.8958  
Epoch 49/50  
12/12 [=====] - 62s 5s/step - loss: 0.3271 - accuracy: 0.8940 - val_loss: 0.2819 - val_accuracy: 0.9167  
Epoch 50/50  
12/12 [=====] - 67s 5s/step - loss: 0.3380 - accuracy: 0.9212 - val_loss: 0.2169 - val_accuracy: 0.9479
```

Gambar 6 Hasil model fit

Pada gambar 7 menunjukkan nilai *epochs 50* nilai akurasi pelatihan mencapai 0.9212 dan nilai akurasi validasi mencapai 0.9792. Dan nilai *loss* pada saat proses pelatihan yaitu 0.2574, sedangkan nilai validasi *loss* yaitu sebesar 0.1167. Secara keseluruhan, melalui pelatihan data dengan 50 iterasi, menandakan peningkatan kemampuan model dalam meminimalkan kesalahan prediksi. Hasil validasi juga menunjukkan peningkatan akurasi seiring waktu. Hasil menunjukkan kemampuan model dalam menggeneralisasi pengetahuan dan memprediksi dengan akurasi tinggi pada data yang belum pernah dilihat sebelumnya. Potensi penggunaan model ini terletak

pada pengenalan pola dan klasifikasi gambar batik secara otomatis.



Gambar 7 Histogram hasil akurasi & loss model

Dalam gambar 8 terlihat adanya keterkaitan antara akurasi dan loss pada data pelatihan dengan jumlah *epoch* atau iterasi. Korelasi pada nilai akurasi menunjukkan hubungan positif, yang berarti semakin tinggi jumlah *epoch* yang digunakan, semakin tinggi juga nilai akurasi pada data pelatihan. Namun, hubungan antara jumlah *epoch* dan nilai *loss* memiliki korelasi negatif. Artinya, semakin besar jumlah *epoch* yang digunakan, semakin rendah nilai *loss* yang dihasilkan pada pelatihan data.

b. Hasil Akurasi Model

Tingkat keberhasilan model dalam melakukan klasifikasi dapat diidentifikasi melalui hasil akurasi yang diperoleh. Dalam rangka mengukur tingkat keberhasilan model dalam melakukan klasifikasi, peneliti menggunakan hasil akurasi sebagai indikator utama. Berikut hasil dari akurasi model pada data Pelatihan dan data Pengujian.

Data Pelatihan

Untuk menghitung akurasi secara spesifik, penulis menggunakan dataset citra sebanyak 400 dalam tahap pengujian menggunakan data latih. Hasil dari *Confusion Matrix* yang terperinci dapat dilihat pada Tabel 1

Tabel 1 *Confusion matrix* data pelatihan

Matrik		Actual class									
		Batik Betawi	Batik Ciamis	Batik Garutan	Batik Kawung	Batik Lasem	Batik Pekalongan	Batik Priangan	Batik Sidomukti	Batik Sogan	Batik Tambal
Predicted Class	Batik Betawi	39	0	0	0	0	0	0	0	1	0
	Batik Ciamis	0	32	6	0	0	0	0	0	0	0
	Batik Garutan	0	4	33	0	0	0	0	0	0	0
	Batik Kawung	0	0	0	40	0	0	0	0	0	0
	Batik Lasem	1	0	1	0	40	1	0	3	0	0
	Batik Pekalongan	0	3	0	0	0	36	0	1	0	0
	Batik Priangan	0	1	0	0	0	0	38	0	2	0
	Batik Sidomukti	0	0	0	0	0	1	1	34	0	0
	Batik Sogan	0	0	0	0	0	2	0	1	37	0
	Batik Tambal	0	0	0	0	0	0	1	0	0	40

Untuk mengukur keberhasilan model deteksi secara keseluruhan, digunakan rumus Akurasi sebagai berikut.

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% s$$

$$Akurasi = \frac{369}{400} \times 100\% \\ Akurasi = 92,25\%$$

Data Pengujian

Untuk menghitung akurasi secara spesifik, peneliti menggunakan dataset citra sebanyak 100 pada tahap pengujian menggunakan data uji.

Dalam proses *Confusion Matrix* yang terperinci telah dihasilkan dan informasinya dapat ditemukan pada Tabel 2

Tabel 2 *Confusion matrix* data pengujian

Matrik		Actual class									
		Batik Betawi	Batik Ciamis	Batik Garutan	Batik Kawung	Batik Lasem	Batik Pekalongan	Batik Priangan	Batik Sidomukti	Batik Sogan	Batik Tambal
Predicted Class	Batik Betawi	10	0	0	0	0	0	0	0	0	0
	Batik Ciamis	0	7	0	0	0	0	0	0	3	0
	Batik Garutan	0	2	10	0	0	0	0	0	0	0
	Batik Kawung	0	0	0	10	0	0	0	0	0	0
	Batik Lasem	0	0	0	0	10	0	0	0	0	0
	Batik Pekalongan	0	1	0	0	0	10	0	0	0	0
	Batik Priangan	0	0	0	0	0	0	10	0	0	0
	Batik Sidomukti	0	0	0	0	0	0	0	10	0	0
	Batik Sogan	0	0	0	0	0	0	0	0	7	0
	Batik Tambal	0	0	0	0	0	0	0	0	0	10

Akurasi model data pengujian dapat dihitung dengan menggunakan rumus sebagai berikut.

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \times 100\%$$

$$Akurasi = \frac{94}{100} \times 100\%$$

$$Akurasi = 94\%$$

c. Hasil Pengujian Aplikasi

Tabel 3 merupakan hasil dari rangkaian pengujian aplikasi yang dilakukan dengan menggunakan sejumlah set data uji yang dipilih secara acak. Pengujian aplikasi dilakukan untuk mengevaluasi kinerja dan fungsionalitas model yang telah dikembangkan dalam aplikasi tersebut. Aplikasi ini menggunakan berbagai input data yang beragam, sehingga dapat menguji berbagai aspek fungsionalitas dan performa dari model yang telah dirancang dan dikembangkan.

Tabel 3 Hasil pengujian model aplikasi

Hasil	Keterangan
	<p>Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Ciamis, dengan tingkat akurasi mencapai 96,61%.</p>

Hasil	Keterangan
 <p>Batik Ciamis 91,04%</p>	<p>Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Ciamis, dengan tingkat akurasi mencapai 91,04%.</p>
 <p>Batik Garutan 91,04%</p>	<p>Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Garutan, dengan tingkat akurasi mencapai 91,04%.</p>
 <p>Batik Kawung 98,27%</p>	<p>Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Kawung, dengan tingkat akurasi mencapai 98,27%.</p>

Hasil	Keterangan
 <p>Batik Lasem 91,53%</p>	<p>Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Lasem, dengan tingkat akurasi mencapai 91,53%.</p>
 <p>Batik Pekalongan 94,97%</p>	<p>Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Pekalongan, dengan tingkat akurasi mencapai 94,97%.</p>
 <p>Batik Pabangran 96,68%</p>	<p>Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Pabangran, dengan tingkat akurasi mencapai 96,68%.</p>

Hasil	Keterangan
	Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Sidomukti, dengan tingkat akurasi mencapai 99,89%.
	Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Sogan, dengan tingkat akurasi mencapai 99,37%.
	Dalam pengujian ini, terlampir tampilan aplikasi yang menunjukkan hasil sukses dalam mendeteksi corak batik Tambal, dengan tingkat akurasi mencapai 97,62%.

Dalam pengujian akurasi pada model berbasis Android, peneliti menjalankan aplikasi dengan menggunakan 30 sampel citra uji yang mencakup berbagai jenis corak batik. Setiap jenis corak batik diuji sebanyak 3 kali untuk mendapatkan hasil yang lebih reliable dan representatif. Hasil dari pengujian tersebut kemudian disajikan dalam Tabel 4, yang memberikan gambaran visual tentang tingkat

keakuratan deteksi corak batik pada setiap pengujian yang dilakukan.

Tabel 4 Hasil pengujian citra batik secara acak

Batik	Uji 1	Uji 2	Uji 3
Betawi	92,39	92,62	90,49
Ciamis	96,61	76,47	96,78
Garutan	91,04	89,67	99,21
Kawung	98,27	88,77	99,45
Lasem	91,53	94,64	98,08
Pekalongan	94,97	84,85	87,83
Priangan	96,68	85,12	96,68
Sidomukti	99,89	91,27	93,48
Sogan	99,37	84,31	91,73

5 Kesimpulan

Berdasarkan penelitian yang dilakukan, dapat ditarik beberapa kesimpulan sebagai Model deteksi dan klasifikasi corak batik menggunakan *Convolutional Neural Network (CNN)* telah berhasil dikembangkan dengan menggunakan dataset 500 citra corak batik yang terdiri dari 10 jenis batik yaitu Batik Betawi, Batik Ciamis, Batik Garutan, Batik Kawung, Batik Lasem, Batik Pekalongan, Batik Priangan, Batik Sidomukti, Batik Sogan, dan Batik Tambal. Pemodelan memiliki akurasi yang tinggi dengan data *training* mencapai 92,25% dan data *testing* mencapai 94%.

Daftar Pustaka

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Zheng, X. (2016, November). Tensorflow: a system for large-scale machine learning. In *Osdi* (Vol. 16, No. 2016, pp. 265-283).
- Agustin, A. (2014). Sejarah batik dan motif batik di Indonesia. In *Seminar Nasional RisetInovatif II* (No. 2339-1553, p. 541).
- Alfikri, R. H., Utomo, M. S., Februariyanti, H., & Nurwahyudi, E. (2022). Pembangunan Aplikasi Penerjemah Bahasa Isyarat Dengan Metode Cnn Berbasis Android. *Jurnal Teknoinfo*, 16(2), 183-197.
- Arifianto, J. (2022). Aplikasi Web Pendeteksi Jerawat Pada Wajah Menggunakan Model Deep Learning Dengan Tensorflow.
- Bariyah, T., Rasyidi, M. A., & Ngatini, N. (2021). Convolutional Neural Network untuk metode klasifikasi multi-label pada motif batik. *Techno. Com*, 20(1), 155-165.

- Gonzalez, R.C., Woods, R.E., dan Eddins, S.L. (2004). *Digital Image Processing Using MATLAB*. Pearson Education.
- Khan, S., H. Rahmani, S. Shah, and D.M Bennamoun. (2018). *A Guide to Convolutional Neural Networks for Computer Vision*. New York: Morgan & Claypool Publishers.
- Lestari, S. D. (2012). *Mengenal Aneka Batik*. PT Balai Pustaka (Persero).
- Maiyana, E. (2018). Pemanfaatan android dalam perancangan aplikasi kumpulan doa. *Jurnal Sains dan Informatika: Research of Science and Informatic*, 4(1), 54-65.
- Putri, R. A., & Rochmawati, N. (2019). Penerapan Algoritma Support Vector Machine untuk Klasifikasi Motif Citra Batik Solo Berdasarkan Fitur Multi-Autoencoders. *Journal of Informatics and Computer Science (JINACS)*, 1(01), 56-63
- Saefurrohman, S., & Ningsih, D. H. U. (2016). Desain Motif Batik Dengan Metode Fraktal Dan Algoritma L-System untuk Membangun Pustaka Batik Wali. *Dinamik*, 21(1), 42-51.
- Savitri, A. (2019). *Revolusi industri 4.0: mengubah tantangan menjadi peluang di era disrupsi 4.0*. Penerbit Genesis.
- Shafira, T. (2018). *Implementasi Convolutional Neural Networks Untuk Klasifikasi Citra Tomat Menggunakan Keras* (Doctoral dissertation, Universitas Islam Indonesia).
- Zindani, A. Y., Amalia, A., & Putro, F. W. (2020). Pendeteksi Kendaraan Untuk Keamanan Perlintasan Kereta Api. *Lomba Karya Tulis Ilmiah*, 1(1), 35-47.

