

# Penggunaan Data Augmentasi dan *Hyperparameter Tuning* dalam Klasifikasi Jenis Batik menggunakan Model CNN

## *Use of Augmentation Data and Hyperparameter Tuning in Batik Type Classification Using the CNN Model*

<sup>1</sup>Siti Auliaddina\*, <sup>2</sup>Toni Arifin

<sup>1,2</sup>Teknik Informatika, Teknologi Informasi, Universitas Adhirajasa Reswara Sanjaya, Antapani Jalan Terusan Sekolah No.1-2, Cicaheum, Kec. Kiaracandong, Kota Bandung, Jawa Barat, Indonesia 40282

\*e-mail: [siti.auliaddina@gmail.com](mailto:siti.auliaddina@gmail.com)

(*received*: 8 Agustus 2023, *revised*: 6 November 2023, *accepted*: 11 November 2023)

### Abstrak

Batik adalah salah satu budaya seni Indonesia yang paling dikenal di dunia dan memiliki motif serta jenis batik tradisional yang berbeda-beda dan memiliki keunikannya sendiri-sendiri. Namun sayangnya, masih begitu banyak masyarakat Indonesia yang belum dapat membedakan jenis-jenis batik berdasarkan motifnya. Karena itulah dibutuhkan sebuah cara untuk membantu masyarakat agar mudah untuk dapat membedakan jenis-jenis batik berdasarkan motifnya. Penelitian ini bertujuan untuk mengklasifikasikan jenis-jenis batik berdasarkan motifnya dengan menggunakan model *deep learning Convolutional Neural Network* dengan menggunakan *Data Augmentation* dan *Hyperparameter Tuning*. CNN termasuk dalam jenis *Deep Neural Network* karena kedalaman jaringannya yang tinggi dan banyak diterapkan pada data citra. Selain itu diterapkan pula *Data Augmentation* dan *Hyperparameter Tuning* untuk mengurangi *overfitting*. Hasil dari penelitian ini menunjukkan model CNN yang menggunakan optimasi *Data Augmentation* dan *Hyperparameter Tuning* mendapatkan nilai akurasi validasi, presisi dan *recall* jauh lebih tinggi yaitu sebesar 66,67% dibandingkan dengan mode CNN yang tidak menggunakan *Data Augmentation* dan *Hyperparameter Tuning* yang memiliki akurasi validasi, presisi, dan *recall* sebesar 28,15%. Selain itu diantara *Data Augmentation* dan *Hyperparameter Tuning*, *Data Augmentation* lah yang paling mempengaruhi peningkatan akurasi validasi, presisi, dan *recall* dibandingkan *Hyperparameter Tuning* dengan peningkatan akurasi validasi menjadi 64% dari akurasi validasi sebesar 28,15%.

**Kata kunci:** Batik, *Convolutional Neural Network*, *Data Augmentation*, *Deep Learning*, *Hyperparameter Tuning*.

### Abstract

*Batik is one of Indonesia's most recognized artistic cultures in the world and has different motifs and types of traditional batik and each has its own uniqueness. But unfortunately, there are still so many Indonesian people who cannot distinguish the types of batik based on their motifs. That's why we need a way to help people easily be able to distinguish the types of batik based on their motifs. This research was conducted to classify types of batik based on their motifs using the Convolutional Neural Network deep learning model using Data Augmentation and Hyperparameter Tuning. CNN is included in the type of Deep Neural Network because of its high network depth and is widely applied to image data. Besides that, Data Augmentation and Hyperparameter Tuning are also applied to reduce overfitting. The results of this study show that the CNN model that uses Data Augmentation optimization and Hyperparameter Tuning gets a much higher accuracy, precision and recall value of 66.67% compared to the CNN mode that does not use Data Augmentation and Hyperparameter Tuning which has validation accuracy, precision, and recall of 28.15%. Besides that, among Data Augmentation and Hyperparameter Tuning, Data Augmentation is the one that most influences the increase in validation accuracy, precision, and recall compared to Hyperparameter Tuning with an increase in validation accuracy to 64% from a validation accuracy of 28.15%.*

**Keywords:** Batik, Convolutional Neural Network, Data Augmentation, Deep Learning, Hyperparameter Tuning.

## 1 Pendahuluan

Batik merupakan salah satu dari seni budaya Indonesia yang sudah ada dari zaman dahulu yaitu dari zaman Kerajaan Majapahit dan masih tinggi tingkat perkembangannya bahkan sampai sekarang ini. Untuk melindungi hak kekayaan intelektual batik, pada tanggal 3 September 2008, pemerintah Indonesia menetapkan batik sebagai warisan budaya Indonesia di UNESCO. Dan pada tanggal 2 Oktober 2009, UNESCO secara resmi mengakui batik sebagai warisan budaya tak benda (*Masterpieces of the Oral and the Intangible Heritage of Humanity*) [1]. Awalnya budaya batik adalah suatu adat istiadat turun menurun, sehingga motif suatu batik berbeda-beda yang dapat dikenali dari asal keluarganya ataupun dari asal daerah batik tersebut. Budaya Indonesia yang begitu kaya dan beragam menyebabkan begitu banyak terciptanya berbagai motif dan jenis batik tradisional yang berbeda-beda dan memiliki keunikannya sendiri-sendiri [2]. Namun sayangnya, masih begitu banyak masyarakat Indonesia yang masih belum dapat membedakan jenis-jenis batik berdasarkan motifnya [3].

Karena masih begitu banyaknya masyarakat yang masih belum dapat membedakan jenis-jenis batik berdasarkan motifnya, maka dibutuhkan sebuah alat yang dapat digunakan untuk dapat mengidentifikasi jenis batik berdasarkan motifnya. Begitu pesatnya perkembangan teknologi di zaman sekarang ini, sebuah alat dapat digunakan untuk mengidentifikasi jenis batik berdasarkan motifnya menggunakan gambar batik yang ingin diketahui jenisnya dengan mengembangkan model *Deep Learning* pada alat tersebut. Model *Deep Learning* yang dapat digunakan untuk melakukan klasifikasi gambar adalah *Convolutional Neural Network (CNN)* [4]. *Convolutional Neural Network (CNN)* adalah salah satu metode *Deep Learning* yang dapat memberikan hasil terbaik dalam pengenalan citra atau gambar sampai saat ini [5].

Namun walaupun model *Convolutional Neural Network (CNN)*, merupakan model yang paling baik untuk klasifikasi citra yang sangat cocok untuk klasifikasi jenis batik yang datasetnya berupa citra, bukan berarti hasil akurasi dari model *CNN* tidak dapat ditingkatkan lagi. Salah satu caranya adalah dengan cara melakukan *Data Augmentation* dan *Hyperparameter Tuning*. *Data Augmentation* atau augmentasi data adalah cara yang dilakukan untuk secara signifikan menambah keragaman data gambar yang digunakan dengan beberapa Teknik, contohnya adalah *cropping*, *padding*, dan *flipping horizontal* [6]. Augmentasi data dilakukan untuk mengatasi *overfitting* yang terjadi di *training model* [7].

Sedangkan *Hyperparameter Tuning* adalah sebuah metode yang digunakan untuk mengoptimalkan parameter yang digunakan dalam algoritma *Machine Learning/Deep Learning* salah satunya adalah algoritma *Convolutional Neural Network (CNN)*, sehingga dapat meningkatkan kinerja model *CNN* dengan signifikan [8]. *Hyperparameter* memiliki nilai yang harus ditetapkan terlebih dahulu sebelum proses pembelajaran yang dilakukan oleh model algoritma dimulai. Salah satu teknik dari *Hyperparameter Tuning* adalah *Grid Search* [9].

Berdasarkan dari pemaparan dan penjelasan permasalahan di atas, maka pada kesempatan kali ini akan dilakukan penelitian yang bertujuan untuk klasifikasi jenis batik berdasarkan motifnya menggunakan algoritma *CNN*, serta mengetahui sejauh mana pengaruh dari augmentasi data dan *hyperparameter tuning* pada peningkatan akurasi algoritma *CNN*.

## 2 Tinjauan Literatur

*Convolutional Neural Networks (CNN)* adalah evolusi dari *Multilayer Perceptron (MLP)* yang dirancang untuk memproses data dua dimensi. *CNN* diklasifikasikan sebagai *Deep Neural Network* karena kedalaman jaringannya yang besar dan banyak diterapkan pada data gambar. Dari segi klasifikasi citra, *MLP* kurang cocok karena tidak menyimpan informasi spasial data citra dan menganggap setiap piksel sebagai fitur independen sehingga memberikan hasil yang buruk. Lapisan *convolutional* di *CNN* melakukan fungsi yang dilakukan oleh sel-sel di korteks *visual*. Pada umumnya *convolutional layer* akan mendeteksi fitur berupa *edge* pada citra, kemudian *subsampling layer* akan mereduksi dimensi dari fitur yang didapat dari *convolutional layer*, dan terakhir meneruskannya ke *output node* melalui proses *forward propagation*, dan prediksi kelas data akhirnya ditentukan oleh metode *Softmax* pada *dense layer* atau *fully connected layer* [10].

Penggunaan *CNN* untuk melakukan klasifikasi gambar telah banyak dilakukan seperti penelitian yang telah dilakukan oleh Sunu Ilham Pradika, Budi Nugroho, dan Eva Yulia Puspaningrum pada tahun 2020 yang menggunakan model *Deep Learning CNN (Convolutional Neural Network)* untuk membuat sistem yang dapat digunakan untuk identifikasi tulisan tangan huruf hijaiyah. Selain menggunakan algoritma *CNN* dalam penelitian ini dilakukan pula optimasi yaitu *Data Augmentation* atau augmentasi data untuk menambah variasi data pada data latih. Selain itu digunakan pula arsitektur *SIP-Net*, dan didapatkan hasil akurasi sebesar 99,7% [11].

Selain itu pada penelitian yang dilakukan oleh Rizki Mawan pada tahun 2020 yang melakukan penelitian untuk melakukan klasifikasi motif batik menggunakan model *CNN* dilakukan klasifikasi batik berdasarkan 3 jenis motif batik yang mencapai rata-rata akurasi sebesar 65% yang kemudian naik menjadi 70% setelah dilakukan kombinasi dengan *Grayscale* [12].

Terdapat pula penelitian yang telah dilakukan oleh Dandhi Trimakno dan Kusriani pada tahun 2021 untuk melakukan penelitian mengklasifikasikan 4 jenis batik berdasarkan motifnya menggunakan model *CNN* dan *K-NN* yang juga menggunakan optimasi augmentasi data yang memberikan pengaruh cukup besar pada peningkatan nilai akurasi dibandingkan tanpa menggunakan augmentasi data. Dalam penelitian ini augmentasi data dapat meningkatkan nilai akurasi bahkan bisa lebih dari 12%, dimana akurasi tertinggi untuk model *CNN* menjadi 96,92% dan model *K-NN* sebesar 81,36% [13].

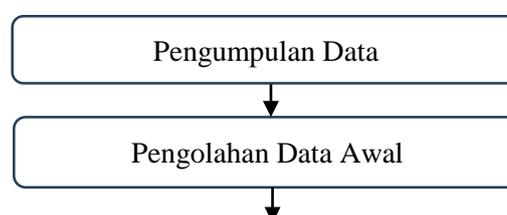
Dari tinjauan literatur di atas terlihat bahwa model *CNN* sangat cocok untuk digunakan dalam klasifikasi citra, bahkan dari beberapa pemaparan penelitian di atas terdapat penelitian yang sama-sama menggunakan batik sebagai objek klasifikasi yang juga akan dilakukan dalam penelitian kali ini. Selain itu terdapat juga penelitian yang menggunakan augmentasi data untuk meningkatkan akurasi. Augmentasi data merupakan salah cara yang dapat digunakan agar tidak terjadi *overfitting* dengan cara meningkatkan jumlah data. Teknik ini merupakan solusi bagi data yang memiliki jumlah data terbatas. Augmentasi data menggabungkan kumpulan metode yang meningkatkan atribut dan ukuran dataset pelatihan. Sehingga jaringan *neural network* dapat bekerja lebih baik ketika teknik ini digunakan [13].

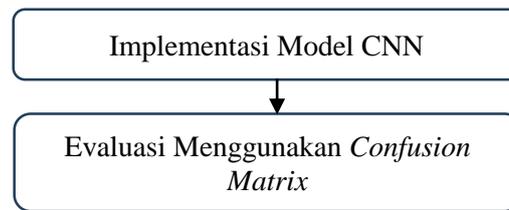
Selain augmentasi data, *hyperparameter tuning* juga dapat digunakan untuk meningkatkan nilai akurasi dari model *CNN*, *hyperparameter tuning* adalah salah satu metode optimasi dengan menyediakan metode pencarian parameter yang optimal untuk model *CNN*. *CNN* yang dikombinasikan dengan *Hyperparameter Tuning* memerlukan pengaturan parameter. *Hyperparameter tuning* memberikan kombinasi parameter terbaik dengan model yang menunjukkan hasil maksimal. Penggunaan *hyperparameter tuning* untuk dapat meningkatkan nilai akurasi pada model *CNN* telah dibuktikan pada penelitian yang dilakukan oleh Agus Eko Minarno, Mochammad Hazmi Cokro, dan Muhammad Rifal Alfarizy pada tahun 2021 pada penelitiannya membuktikan bahwa penggunaan *Filter Gabor* dan *Hyperparameter Tuning* pada model *CNN* dapat meningkatkan nilai akurasi dibanding dengan tidak menggunakan *Filter Gabor* dan *Hyperparameter Tuning*. Dan nilai akurasi yang didapat pada model *CNN* yang menggunakan *Filter Gabor* dan *Hyperparameter Tuning* adalah sebesar 97,9% dan nilai AUC adalah sebesar 99% [14].

Dikarenakan sudah banyak penelitian yang membuktikan bahwa augmentasi data dan *hyperparameter tuning* dapat meningkatkan nilai akurasi pada model *CNN*, maka pada penelitian kali ini akan menggunakan keduanya yaitu augmentasi data dan *hyperparameter tuning* untuk meningkatkan nilai akurasi pada model *CNN* dan juga untuk mengurangi *overfitting*.

### 3 Metode Penelitian

Dalam penelitian ini akan dibuat satu rangkaian penelitian untuk mendapatkan kesimpulan penelitian dengan menggunakan metode penelitian eksperimen yang menggunakan Bahasa pemrograman *python* menggunakan *IDE Google Colaboratory*, Adapun tahap penelitiannya adalah sebagai berikut:



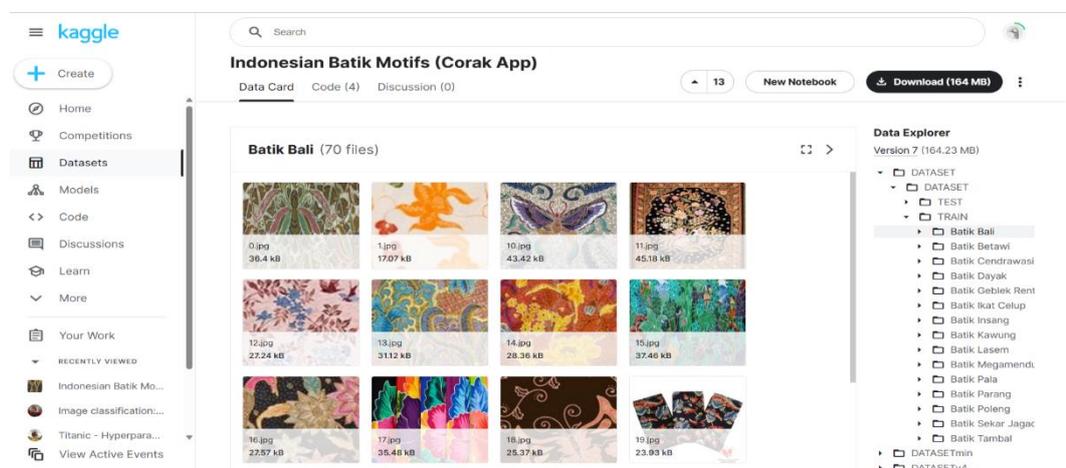


Gambar 1. Tahapan Penelitian[15]

Berdasarkan langkah-langkah penelitian yang diuraikan pada Gambar 1, akan dijelaskan lebih rinci lagi mengenai langkah-langkah penelitian yang akan dilakukan dalam penelitian ini. Berikut ini adalah penjelasannya:

### 3.1 Pengumpulan Data

Dalam penelitian ini digunakan dataset berupa kumpulan cita dan merupakan dataset publik di mana dataset ini merupakan dataset yang dapat diakses oleh siapapun secara gratis yang diperoleh dari *website Kaggle*. Dataset yang digunakan dalam penelitian ini memiliki judul *Indonesian Batik Motifs (Corak App)* dengan linknya adalah (<https://www.kaggle.com/datasets/alfanme/indonesian-batik-motifs-corak-app>). Di bawah ini adalah gambar dari beberapa data gambar batik dataset *Indonesian Batik Motifs (Corak App)*.



Gambar 2. Dataset Indonesian Batik Motifs (Corak App)[16]

Gambar 2 di atas merupakan hasil dari tangkapan layar dari sumber dataset yang akan digunakan. Dataset ini memiliki 1350 jumlah data gambar motif batik yang dibagi menjadi 15 kelas motif batik yaitu Batik Bali, Batik Lasem, Batik Betawi, Batik Megamendung, Batik Tali Celup, Batik Parang, Batik Cendrawasih, Batik Dayak, Batik Geblek Renteng, Batik Insang, Batik Pala, Batik Sekar Jagad, Batik Poleng, Batik Kawung, dan Batik Tambal.

### 3.2 Pengolahan Data Awal

Dataset yang berupa gambar citra dapat diolah dengan terlebih dahulu melakukan pemisahan dataset menjadi 2 bagian yaitu data latih dan data test. Dengan komposisi data latih sebesar 90% dan data test sebesar 10%. Setelah itu melakukan *resize* atau mengubah ukuran solusi awal dari gambar. Kemudian dilakukan normalisasi pada dataset, agar semua ukuran resolusi gambar memiliki ukuran resolusi yang sama. Selain itu untuk menambah varian dari data, dilakukanlah augmentasi data. Augmentasi data dilakukan untuk mengurangi resiko *overfitting* sehingga performa model yang digunakan dapat bekerja optimal. Augmentasi data yang dilakukan dalam penelitian ini adalah *flipping*, *rescale*, *rotate*, dan *zoom*. Setelah semua dilakukan diperlukan untuk memisahkan dataset menjadi 2 bagian yaitu data latih dan data validasi.

### 3.3 Implementasi Model CNN

Model CNN akan diujikan pada dataset Batik yang telah diolah. Model CNN yang dibangun akan menggunakan arsitektur *xception*. *Xtreme of Inception (Xception)* merupakan sebuah arsitektur model algoritma CNN dengan menggunakan metode konvolusi yang dapat dipisahkan secara mendalam. Arsitektur *xception* adalah hasil dari pengembangan salah satu arsitektur model CNN *Inception* yang memiliki 36 *convolutional layer* yang menjadi dasar dari jaringan ekstraksi fitur [17]. Selain itu ditambahkan pula 1 *flatten layer*, dan 1 *output layer* dalam model CNN yang akan diimplementasikan. Dalam proses pembuatan model CNN yang digunakan dalam penelitian ini, penambahan tahapan *hyperparameter tuning* dilakukan untuk menemukan model dengan parameter terbaik, sehingga mencapai kinerja yang unggul. *Hyperparameter tuning* digunakan untuk menentukan jumlah *neuron* yang akan digunakan pada *dense layer*, selain itu *hyperparameter tuning* digunakan untuk mencari *optimizer*, *learning rate* pada *optimizer*, *batch size*, dan jumlah *epoch* terbaik. Setelah *hyperparameter tuning* dilakukan, maka semua parameter terbaik dari hasil *hyperparameter tuning* akan dimasukkan ke dalam model CNN yang akan digunakan.

### 3.4 Evaluasi Menggunakan Confusion Matrix

Tahap yang terakhir adalah tahap evaluasi. Dimana model evaluasi yang dipakai dalam penelitian ini adalah *confusion matrix*. Metode *confusion matrix* metode evaluasi yang digunakan dalam model klasifikasi yang hasilnya merupakan *performance model* klasifikasi dan direpresentasikan dalam bentuk tabel. Tabel *performance model* klasifikasi terbentuk dari baris dan kolom yang banyaknya sama dengan total kelas nilai *false positives* (FP) yang artinya data yang sebenarnya bernilai negatif dideteksi sebagai data yang bernilai positif, *false negatives* (FN) merupakan data yang sebenarnya bernilai positif dideteksi sebagai data yang bernilai negatif, *true positives* (TP) merupakan data yang bernilai positif yang terdeteksi benar sesuai dengan nilainya, dan *true negatives* (TN) merupakan jumlah data yang bernilai negatif yang terdeteksi benar sesuai dengan nilainya[18].

## 4 Hasil dan Pembahasan

Pada bagian ini akan dijelaskan satu persatu tentang hasil penelitian dan pembahasan dari penelitian yang telah dilakukan yaitu klasifikasi jenis batik berdasarkan motifnya dengan menggunakan model algoritma *Convolutional Neural Network (CNN)*. Dan tahapan dari hasil penelitian adalah sebagai berikut.

### 4.1 Dataset

Dataset batik yang diambil dari *website Kaggle* ini terdiri dari 15 kelas yang setiap kelasnya terdiri dari 90 *file* gambar batik, sehingga total *file* citra batik adalah 1350 *file*, yang kemudian dijadikan *dataframe*. Kemudian dilakukan pemisahan dataset menjadi data latih dan data test. Pemisahan dataset menjadi data latih dan data test menggunakan fungsi *train\_test\_split* yang merupakan fungsi dari *library scikit-learn*. Penentuan komposisi data latih dan data test menggunakan parameter *train\_set*. Komposisi yang digunakan dalam penelitian ini adalah 90% untuk data latih dan 10% untuk data test, sehingga jika ditotalkan untuk data latih berjumlah 1215 data dan 315 data untuk data test. Selain itu digunakan pula parameter *random\_state* yang merupakan *random number generator* yang berguna untuk mengontrol proses *Shuffle*, dimana akan selalu didapatkan data-latih dan data test yang sama datanya walaupun di eksekusi yang berbeda-beda. Pemilihan komposisi data latih sebesar 90% dan data test sebesar 10% dikarenakan dengan memperbanyak komposisi train menjadi 90% dapat memperkecil hasil akurasi yang *overfitting*.

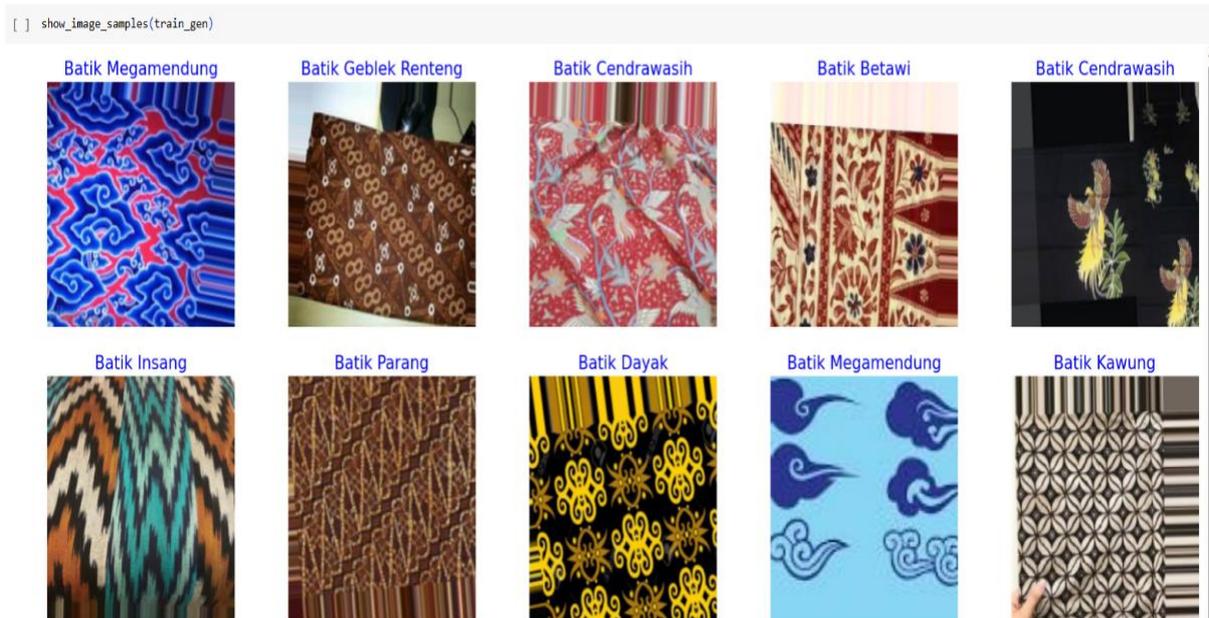
### 4.2 Pengolahan Data Awal

Setelah proses pemisahan dataset menjadi data latih dan data test, maka proses selanjutnya adalah prapemrosesan atau pengolahan data awal yaitu augmentasi data dan *hyperparameter tuning* agar data latih dapat digunakan untuk melakukan training model.

#### 4.2.1 Augmentasi Data

Pada penelitian ini augmentasi data menggunakan *libabry Keras ImageDataGenerator*. Pada proses augmentasi data ini semua data gambar akan dinormalisasi dengan menyamakan semua ukuran data gambar batik, dilanjutkan dengan melakukan *rescale*, pemberian parameter *rotation\_range*,

melakukan pergeseran *image* baik secara *horizontal* dan *vertical*. Setelah pemberian parameter pada *ImageDataGenerator*, maka untuk diimplementasikan pada data latih dan data test maka menggunakan fungsi *flow\_from\_dataframe*. Dan hasil dari augmentasi data dapat dilihat pada gambar di bawah ini.



**Gambar 3. Citra Batik Data Latih Hasil Augmentasi Data**

Gambar 3 merupakan hasil dari augmentasi data pada data latih dengan melakukan normalisasi semua ukuran data gambar baik pada data latih maupun data test menjadi 224x224, lalu data latih diberikan parameter *rescale* sebesar 1/255, *rotation\_range* sebesar 10 derajat, pergeseran *image* secara *horizontal* dan *vertical* sebesar 20%, dan membalikkan gambar secara *horizontal*.

#### 4.2.2 Hyperparameter tuning

Langkah selanjutnya adalah melakukan *hyperparameter tuning* yang dilakukan dengan menggunakan bahasa pemrograman *python* menggunakan *tool IDE Google Colaboratory*. dimana metode *hyperparameter tuning* yang digunakan adalah metode *grid search* yang menggunakan *library scikit-learn*. *Hyperparameter tuning* dilakukan untuk mencari beberapa parameter yang paling optimal untuk dimasukkan dalam *training model*. Parameter yang dicari pada proses *Hyperparameter Tuning* di penelitian ini adalah jumlah *layer* dan *neuron* pada *dense layer*, *optimizer*, *learning rate* pada *optimizer*, *batch size*, dan jumlah *epoch*. Didapatkan hasil bahwa tidak diperlukan *layer dense* tambahan, dengan optimizernya adalah 'Adam' dengan *learning rate* sebesar 0.001, dengan *batch size* sebesar 16, serta jumlah *epoch* sebanyak 100 *epoch* adalah parameter-parameter yang paling baik untuk digunakan dalam model algoritma *CNN* pada penelitian ini.

```
Best: 0.3580246865749359, using {'layers': []}
0.3580246865749359 (0.06741941628898697) with: {'layers': []}
0.2106995850801468 (0.0711541544656211) with: {'layers': [32]}
0.3152263402938843 (0.03722858634483223) with: {'layers': [64]}

Best: 0.46090534925460813, using {'optimizer': 'Adam'}
0.35390947461128236 with: {'optimizer': 'SGD'}
0.33827160596847533 with: {'optimizer': 'RMSprop'}
0.46090534925460813 with: {'optimizer': 'Adam'}

Best: 0.43456789255142214, using {'learn_rate': 0.001}
0.43456789255142214 with: {'learn_rate': 0.001}
0.17119341492652893 with: {'learn_rate': 0.01}
0.05432098805904388 with: {'learn_rate': 0.1}

Best: 0.3761316895484924, using {'batch_size': 16}
0.3761316895484924 with: {'batch_size': 16}
0.3736625611782074 with: {'batch_size': 32}
0.34485597014427183 with: {'batch_size': 64}

Best: 0.5448559761047364, using {'epochs': 100}
0.5390946388244628 with: {'epochs': 10}
0.5423868417739868 with: {'epochs': 50}
0.5448559761047364 with: {'epochs': 100}
```

**Gambar 4. Parameter-parameter Terbaik Hasil Hyperparameter Tuning**

### 4.3 Implementasi Model CNN

Tahap selanjutnya adalah implementasi model CNN yang menggunakan bahasa pemrograman *python* dengan menggunakan *IDE Google Colaboratory*, serta *library keras*. Model algoritma yang digunakan dalam penelitian ini adalah model *Convolutional Neural Network*, dan arsitektur yang digunakan adalah arsitektur *xception*. Arsitektur *xception* digunakan sebagai *base model* pada algoritma CNN yang dibangun pada penelitian ini. Model CNN yang diimplementasikan pada data *train*, merupakan gabungan dari arsitektur CNN *xception* dengan ditambahkan 1 *layer Conv2D*, 1 *layer MaxPool2D*, 1 *layer Flatten*, dan 1 *layer output*. Model yang telah dibangun tersebut kemudian di *compile* dengan *optimizer* dan *learning rate* yang merupakan hasil optimasi *hyperparameter tuning* yaitu '*Adam*', dengan *learning rate 0.001*. *Epoch* yang digunakan adalah 100 dengan *batch size* adalah 16. *Epoch* sebesar 100 artinya adalah proses *training* akan berulang-ulang sampai 100 kali, sehingga mendapatkan ekstraksi ciri dari fitur yang dibutuhkan. Untuk grafik *loss* dan akurasi berdasarkan *epoch* dapat dilihat pada Gambar 5 di bawah ini.

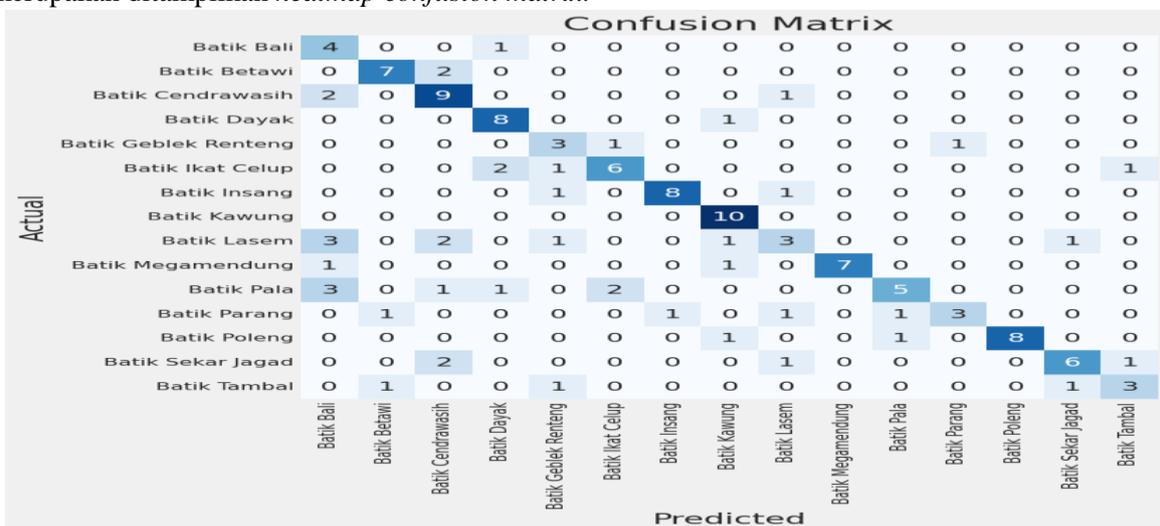


Gambar 5. Grafik Loss dan Akurasi Berdasarkan Epoch

Dari kedua gambar grafik di atas, terlihat pada grafik *loss epoch* terbaik adalah pada *epoch ke-7* dimana nilai *loss* dan nilai *loss validasi* mencapai yang paling rendah yaitu sebesar 1,4679 sedangkan pada grafik akurasi terlihat bahwa *epoch* terbaik adalah pada *epoch ke-20* dimana nilai akurasi validasi mencapai 70,37%.

### 4.4 Evaluasi Menggunakan Confusion Matrix

Pada tahap ini evaluasi akan menguji model yang sudah dibangun pada data test apakah kelas hasil prediksi sesuai dengan kelas yang sebenarnya menggunakan metode *confusion matrix*. Pada tahap evaluasi metode *confusion matrix* Bahasa pemrograman yang digunakan adalah *python* dan *IDE* yang digunakan adalah *Google Colaboratory*, serta *library* yang digunakan adalah *scikit-learn*. Berikut merupakan ditampilkan *heatmap confusion matrix*.



Gambar 6. Heatmap Hasil Evaluasi Confusion Matrix Pada Model CNN

Dapat dilihat dari Gambar 6 bahwa hasil prediksi yang tepat dengan *actual* kelas adalah berjumlah 90 hasil prediksi, yang terdiri dari 4 hasil prediksi kelas Batik Bali, 7 hasil prediksi kelas Batik Betawi, 9 hasil prediksi kelas Batik Cendrawasih, 8 hasil prediksi kelas Batik Dayak, 3 hasil prediksi kelas Batik Geblek Renteng, 6 hasil prediksi kelas Batik Ikat Celup, 8 hasil prediksi kelas Batik Insang, 10 hasil prediksi kelas Batik Kawung, 3 hasil prediksi kelas Batik Lasem, 7 hasil prediksi kelas Batik Megamendung, 5 hasil prediksi kelas Batik Pala, 3 hasil prediksi kelas Batik Parang, 8 hasil prediksi kelas Batik Poleng, 6 hasil prediksi kelas Batik SekarJagad, 3 hasil prediksi kelas Batik Tambal. Jika ditotalkan maka jumlah hasil prediksi yang tepat adalah 90 buah dari total data latih sebanyak 135. Sehingga nilai akurasi dapat dihitung dengan formula sebagai berikut.

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

$$Accuracy = \frac{90}{135} \times 100\%$$

$$Accuracy = 66,67\%$$

Hasil dari perhitungan menunjukkan bahwa nilai akurasi dari klasifikasi jenis batik berdasarkan motifnya menggunakan augmentasi data dan *hyperparameter tuning* adalah sebesar 66,67%. Selanjutnya untuk mengetahui nilai presisi dapat menggunakan formula seperti di bawah ini.

$$Precision = \frac{TP}{TP+FP} \times 100\%$$

$$Precision = \frac{90}{90+45} \times 100\%$$

$$Precision = 66,67\%$$

Hasil dari perhitungan menunjukkan bahwa nilai presisi dari klasifikasi jenis batik berdasarkan motifnya menggunakan augmentasi data dan *hyperparameter tuning* adalah sebesar 66,67%. Selanjutnya untuk mengetahui nilai *recall* dapat menggunakan formula seperti di bawah ini.

$$Recall = \frac{TP}{TP+FN} \times 100\%$$

$$Recall = \frac{90}{90+45} \times 100\%$$

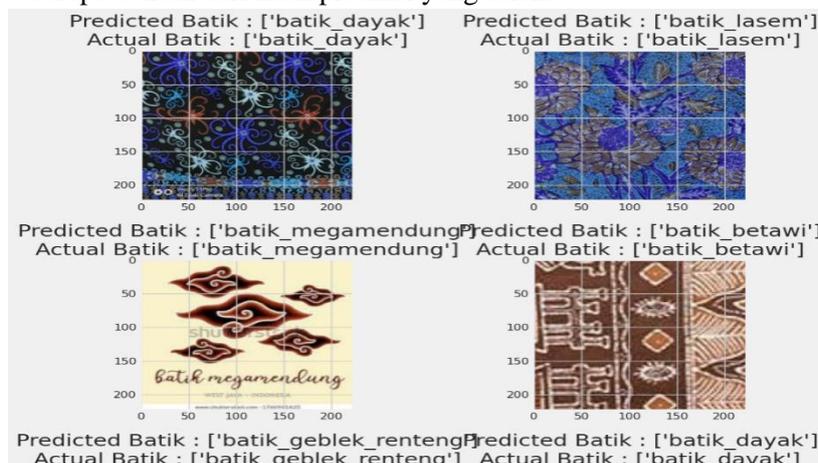
$$Recall = 66,67\%$$

Dari hasil perhitungan akurasi, presisi, dan *recall* di atas sehingga diperoleh hasil seperti pada Tabel 1 di bawah ini.

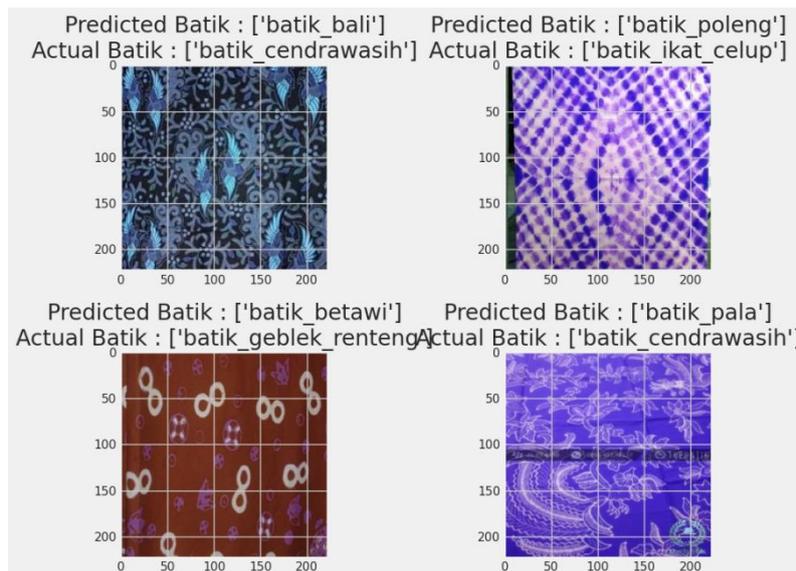
**Tabel 1. Hasil Akurasi, Akurasi Validasi, Presisi Dan Recall Model CNN Menggunakan Augmentasi Data dan Hyperparameter Tuning**

	Akurasi	Akurasi Validasi	Presisi	Recall
Hasil	99,48%	66,67%	66,67%	66,67%

Dan dibawah ini merupakan beberapa contoh dari hasil prediksi jenis batik berdasarkan motifnya yang benar dan beberapa contoh dari hasil prediksi yang salah.



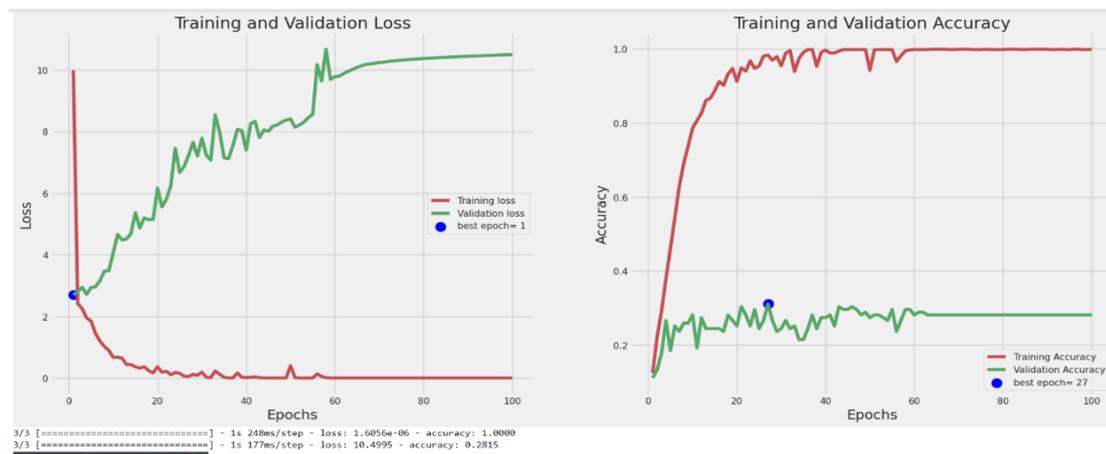
**Gambar 7. Hasil Predikis Jenis Batik Yang Tepat**



Gambar 8. Hasil Prediksi Jenis Batik Yang Tidak Tepat

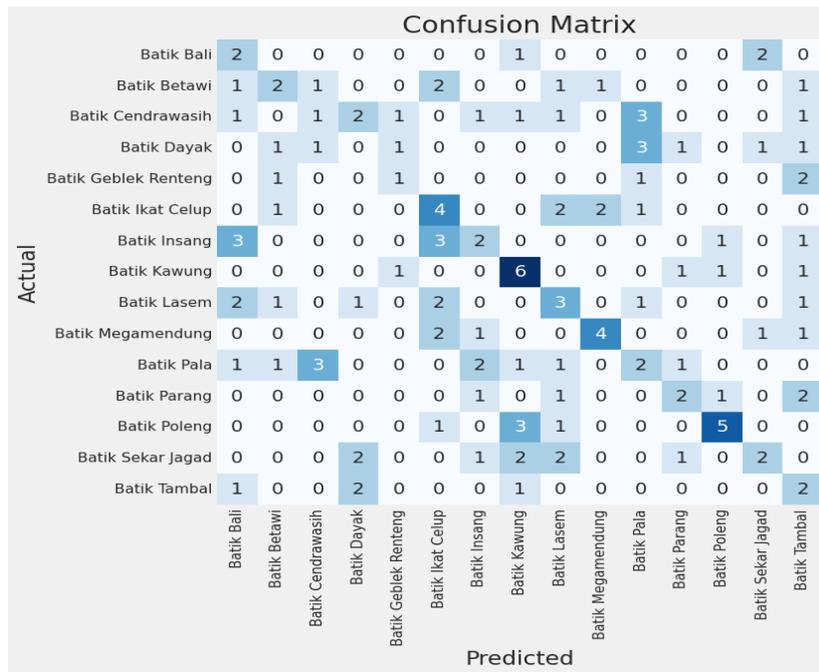
### Implementasi Algoritma CNN Tanpa Menggunakan Augmentasi Data dan Hyperparameter Tuning

Dalam penelitian ini dilakukan pula implementasi algoritma CNN pada dataset yang tidak menggunakan augmentasi data dan *hyperparameter tuning*, untuk membandingkan hasil akurasi dengan model algoritma CNN yang sama pada dataset yang menggunakan augmentasi data dan *hyperparameter tuning* dengan hasil akurasi pada dataset yang tidak menggunakan augmentasi data dan *hyperparameter tuning*. Dan hasil akurasi dari pelatihan model CNN tanpa Augmentasi data dan *hyperparameter tuning* pada epoch ke-100 adalah 94,32% sedangkan untuk nilai akurasi validasinya adalah 22,22%. Untuk gambar grafik akurasi dan loss berdasarkan epoch nya dapat dilihat di bawah ini.



Gambar 9. Grafik akurasi dan loss berdasarkan epoch pada model CNN tanpa augmentasi data dan hyperparameter tuning

Berdasarkan Gambar 9 di atas dapat diketahui bahwa untuk grafik loss posisi epoch terbaiknya adalah epoch ke-1 dimana nilai loss validasi mencapai loss terendah dengan nilai loss sebesar 2.7124, dan pada grafik akurasi posisi epoch terbaiknya ada pada epoch ke-27 dimana nilai akurasi validasi mencapai 31,85%. Dan untuk hasil dari evaluasi *confusion matrix* dapat dilihat di bawah ini.



**Gambar 10. Heatmap Hasil Evaluasi Confusion Matrix Pada Model CNN Tanpa Menggunakan Augmentasi Data dan Hyperparameter Tuning**

Terlihat pada Gambar 10 bahwa hasil prediksi yang tepat dengan *actual* kelas adalah berjumlah 38 hasil prediksi, yang terdiri dari 2 hasil prediksi kelas Batik Bali, 2 hasil prediksi kelas Batik Betawi, 1 hasil prediksi kelas Batik Cendrawasih, 0 hasil prediksi kelas Batik Dayak, 1 hasil prediksi kelas Batik Geblek Renteng, 4 hasil prediksi kelas Batik Ikat Celup, 2 hasil prediksi kelas Batik Insang, 6 hasil prediksi kelas Batik Kawung, 3 hasil prediksi kelas Batik Lasem, 4 hasil prediksi kelas Batik Megamendung, 2 hasil prediksi kelas Batik Pala, 2 hasil prediksi kelas Batik Parang, 5 hasil prediksi kelas Batik Poleng, 2 hasil prediksi kelas Batik SekarJagad, 2 hasil prediksi kelas Batik Tambal. Jika ditotalkan maka jumlah hasil prediksi yang tepat adalah 38 buah dari total data latih sebanyak 135. Sehingga nilai akurasi dapat dihitung dengan formula sebagai berikut.

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

$$Accuracy = \frac{38}{135} \times 100\%$$

$$Accuracy = 28,15\%$$

Hasil dari perhitungan menunjukkan bahwa nilai akurasi dari klasifikasi jenis batik berdasarkan motifnya tanpa menggunakan augmentasi data dan *hyperparamter tuning* adalah sebesar 28,15%. Selanjutnya untuk mengetahui nilai presisi dapat menggunakan formula seperti di bawah ini.

$$Precision = \frac{TP}{TP+FP} \times 100\%$$

$$Precision = \frac{38}{38+97} \times 100\%$$

$$Precision = 28,15\%$$

Hasil dari perhitungan menunjukkan bahwa nilai presisi dari klasifikasi jenis batik berdasarkan motifnya tanpa menggunakan augmentasi data dan *hyperparamter tuning* adalah sebesar 28,15%. Selanjutnya untuk mengetahui nilai *recall* dapat menggunakan formula seperti di bawah ini.

$$Recall = \frac{TP}{TP+FN} \times 100\%$$

$$Recall = \frac{38}{38+97} \times 100\%$$

$$Recall = 28,15\%$$

Dari hasil perhitungan akurasi, presisi, dan *recall* diatas sehingga dapat dibuat tabel seperti dibawah ini.

**Tabel 2. Hasil Akurasi, Akurasi Validasi, Presisi dan Recall Model CNN Tanpa Menggunakan Augmentasi Data dan Hyperparameter Tuning**

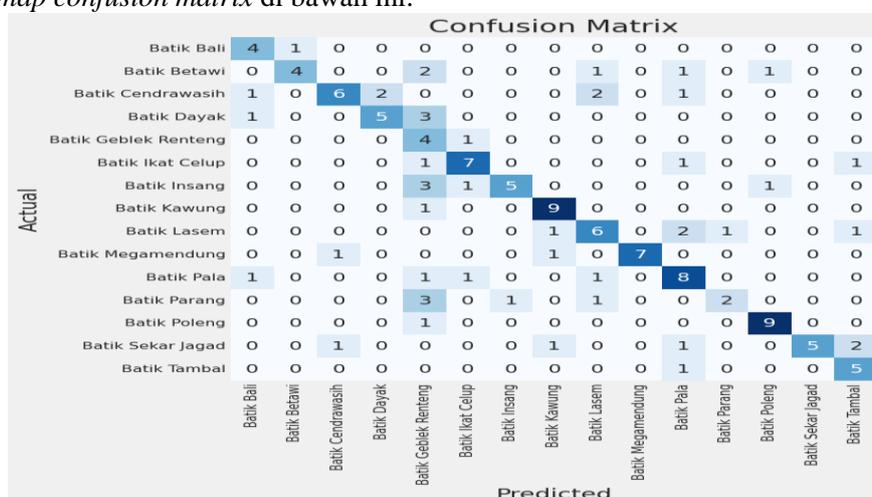
	Akurasi	Akurasi Validasi	Presisi	Recall
Hasil	100%	28,15%	28,15%	28,15%

Dari hasil penelitian yang telah dipaparkan diatas tentang apakah augmentasi data dan *hyperparameter tuning* dapat mempengaruhi akurasi model CNN untuk klasifikasi jenis batik berdasarkan motifnya dapat dilihat pada Tabel 3 di bawah ini.

**Tabel 3. Tabel perbandingan nilai akurasi model CNN menggunakan augmentasi data dan hyperparameter tuning dengan akurasi model CNN tanpa menggunakan augmentasi data dan hyperparameter tuning**

	Model CNN Menggunakan Augmentasi Data dan Hyperparameter Tuning	Model CNN Tanpa Menggunakan Augmentasi Data dan Hyperparameter Tuning
Akurasi	99,48	100%
Akurasi Validasi	66,67%	28,15%
Presisi	66,67%	28,15%
Recall	66,67%	28,15%

Terlihat dari Tabel 3 terlihat bahwa nilai akurasi validasi, presisi, dan *recall* model CNN menggunakan augmentasi data dan *hyperparameter tuning* jauh lebih besar dibandingkan dengan nilai akurasi validasi, presisi, dan *recall* model CNN tanpa menggunakan augmentasi data dan *hyperparameter tuning* namun walaupun begitu nilai akurasi data *training* pada keduanya sangatlah tinggi. Hal ini membuktikan bahwa augmentasi data dan *hyperparameter tuning* dapat memperkecil kesenjangan nilai akurasi data *train* dengan nilai akurasi data validasi, atau dapat disimpulkan pula augmentasi data dan *hyperparameter tuning* dapat memperkecil resiko *overfitting*. Namun perlu dikaji pula manakah antara augmentasi data dan *hyperparameter tuning* yang paling berpengaruh pada kenaikan nilai akurasi validasi. Sehingga dilakukan pulalah penelitian mencari nilai akurasi validasi pada model CNN yang menggunakan augmentasi data namun tanpa menggunakan *hyperparameter tuning* dan nilai akurasi validasi pada model CNN tanpa menggunakan augmentasi data namun menggunakan *hyperparameter tuning*. Untuk hasil evaluasi *confusion matrix* model CNN yang menggunakan augmentasi data namun tanpa menggunakan *hyperparameter tuning* dapat dilihat pada gambar *heatmap confusion matrix* di bawah ini.



**Gambar 11. Heatmap Confusion Matrix Model CNN Menggunakan Augmentasi Data Tanpa Hyperparameter Tuning**

Terlihat pada Gambar 11 di atas bahwa hasil prediksi yang tepat dengan *actual* kelas adalah berjumlah 86 hasil prediksi, yang terdiri dari 4 hasil prediksi kelas Batik Bali, 4 hasil prediksi kelas

Batik Betawi, 6 hasil prediksi kelas Batik Cendrawasih, 5 hasil prediksi kelas Batik Dayak, 4 hasil prediksi kelas Batik Geblek Renteng, 7 hasil prediksi kelas Batik Ikat Celup, 5 hasil prediksi kelas Batik Insang, 9 hasil prediksi kelas Batik Kawung, 6 hasil prediksi kelas Batik Lasem, 7 hasil prediksi kelas Batik Megamendung, 8 hasil prediksi kelas Batik Pala, 2 hasil prediksi kelas Batik Parang, 9 hasil prediksi kelas Batik Poleng, 5 hasil prediksi kelas Batik SekarJagad, 5 hasil prediksi kelas Batik Tambal. Jika ditotalkan maka jumlah hasil prediksi yang tepat adalah 86 buah dari total data latih sebanyak 135. Sehingga nilai akurasi dapat dihitung dengan formula sebagai berikut.

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

$$Accuracy = \frac{86}{135} \times 100\%$$

$$Accuracy = 64\%$$

Hasil dari perhitungan menunjukkan bahwa nilai akurasi dari klasifikasi jenis batik berdasarkan motifnya menggunakan augmentasi data tanpa *hyperparameter tuning* adalah sebesar 64%. Selanjutnya untuk mengetahui nilai presisi dapat menggunakan formula seperti di bawah ini.

$$Precision = \frac{TP}{TP+FP} \times 100\%$$

$$Precision = \frac{86}{86+49} \times 100\%$$

$$Precision = 64\%$$

Hasil dari perhitungan menunjukkan bahwa nilai presisi dari klasifikasi jenis batik berdasarkan motifnya menggunakan augmentasi data tanpa *hyperparameter tuning* adalah sebesar 64%. Selanjutnya untuk mengetahui nilai *recall* dapat menggunakan formula seperti di bawah ini.

$$Recall = \frac{TP}{TP+FN} \times 100\%$$

$$Recall = \frac{86}{86+49} \times 100\%$$

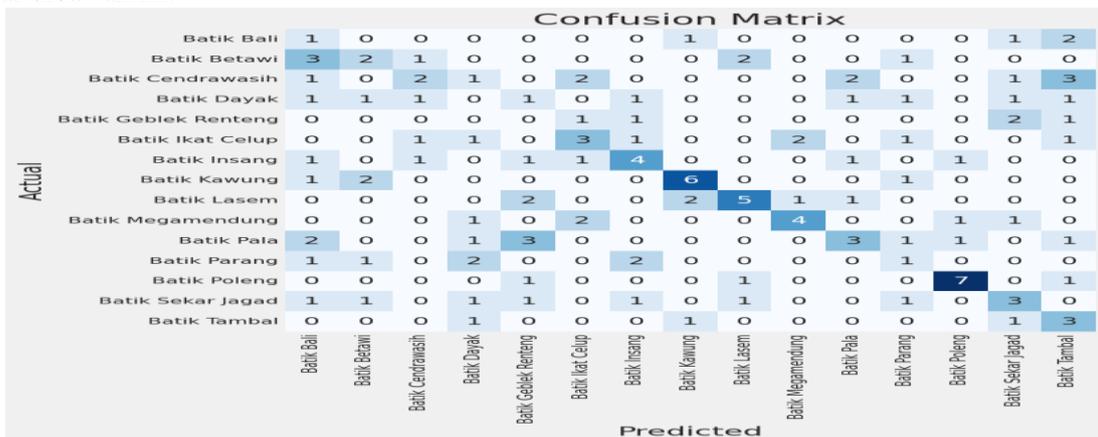
$$Recall = 64\%$$

Dari hasil perhitungan akurasi, presisi, dan *recall* diatas sehingga diperoleh Tabel 4 seperti dibawah ini.

**Tabel 4. Hasil Akurasi, Akurasi Validasi, Presisi dan Recall Model CNN menggunakan augmentasi tanpa hyperparameter tuning**

	Akurasi	Akurasi Validasi	Presisi	Recall
Hasil	95,31%	64%	64%	64%

Selanjutnya dilakukan penelitian untuk mencari nilai akurasi validasi, presisi, dan *recall* pada model CNN yang menggunakan *hyperparameter tuning* tanpa augmentasi data. Nilai akurasi validasi, presisi, dan *recall* didapatkan dari hasil evaluasi *confusion matrix* model CNN yang menggunakan *hyperparameter tuning* tanpa augmentasi data, dan *heatmap confusion matrix* nya dapat dilihat pada gambar dibawah ini.



**Gambar 12. Heatmap Confusion Matrix Model CNN Menggunakan Hyperparameter Tuning Tanpa Augmentasi Data**

Terlihat pada Gambar 2 bahwa hasil prediksi yang tepat dengan *actual* kelas adalah berjumlah 44 hasil prediksi, yang terdiri dari 1 hasil prediksi kelas Batik Bali, 2 hasil prediksi kelas Batik Betawi, 2 hasil prediksi kelas Batik Cendrawasih, 0 hasil prediksi kelas Batik Dayak, 0 hasil prediksi kelas Batik Geblek Renteng, 3 hasil prediksi kelas Batik Ikat Celup, 4 hasil prediksi kelas Batik Insang, 6 hasil prediksi kelas Batik Kawung, 5 hasil prediksi kelas Batik Lasem, 4 hasil prediksi kelas Batik Megamendung, 3 hasil prediksi kelas Batik Pala, 1 hasil prediksi kelas Batik Parang, 7 hasil prediksi kelas Batik Poleng, 3 hasil prediksi kelas Batik SekarJagad, 3 hasil prediksi kelas Batik Tambal. Jika ditotalkan maka jumlah hasil prediksi yang tepat adalah 44 buah dari total data latih sebanyak 135. Sehingga nilai akurasi dapat dihitung dengan formula sebagai berikut.

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

$$Accuracy = \frac{44}{135} \times 100\%$$

$$Accuracy = 33\%$$

Hasil dari perhitungan menunjukkan bahwa nilai akurasi dari klasifikasi jenis batik berdasarkan motifnya menggunakan *hyperparameter tuning* tanpa augmentasi data adalah sebesar 33%. Selanjutnya untuk mengetahui nilai presisi dapat menggunakan formula seperti di bawah ini.

$$Precision = \frac{TP}{TP+FP} \times 100\%$$

$$Precision = \frac{44}{44+91} \times 100\%$$

$$Precision = 33\%$$

Hasil dari perhitungan menunjukkan bahwa nilai presisi dari klasifikasi jenis batik berdasarkan motifnya menggunakan *hyperparameter tuning* tanpa augmentasi data adalah sebesar 33%. Selanjutnya untuk mengetahui nilai *recall* dapat menggunakan formula seperti di bawah ini.

$$Recall = \frac{TP}{TP+FN} \times 100\%$$

$$Recall = \frac{44}{44+91} \times 100\%$$

$$Recall = 33\%$$

Dari hasil perhitungan akurasi, presisi, dan *recall* di atas sehingga dapat dibuat Tabel 5 seperti dibawah ini.

**Tabel 5. Hasil Akurasi, Akurasi Validasi, Presisi Dan Recall Model CNN Menggunakan Hyperparameter Tuning Tanpa Augmentasi Data**

	Akurasi	Akurasi Validasi	Presisi	Recall
Hasil	99,48%	33%	33%	33%

Sehingga dari semua hasil di atas dapat dibuat tabel perbandingan nilai akurasi, akurasi validasi, presisi dan *recall* yang dapat dilihat pada Tabel 6 di bawah ini.

**Tabel 6. Tabel Perbandingan Nilai Akurasi Model CNN**

	Model CNN Menggunakan Augmentasi Data dan Hyperparameter Tuning	Model CNN Tanpa Augmentasi Data dan Hyperparameter Tuning	Model CNN Menggunakan Augmentasi Tanpa Hyperparameter Tuning	Model CNN Menggunakan Hyperparameter Tuning Tanpa Augmentasi Data
Akurasi	99,48	100%	95,31%	99,48%
Akurasi Validasi	66,67%	28,15%	64%	33%
Presisi	66,67%	28,15%	64%	33%
Recall	66,67%	28,15%	64%	33%

Dari tabel 6 di atas terlihat bahwa model *CNN* yang menggunakan augmentasi data dan *hyperparameter tuning* memiliki nilai akurasi validasi, presisi dan *recall* paling tinggi, sehingga memiliki kesenjangan nilai paling kecil dengan akurasi data latih. Namun terlihat pula dari tabel di atas bahwa augmentasi data lah yang memiliki pengaruh paling signifikan untuk menaikkan nilai akurasi validasi, presisi, dan *recall* disbanding dengan *hyperparameter tuning*. Walaupun begitu *hyperparameter tuning* memiliki pula peran dalam meningkatkan nilai akurasi validasi, presisi dan *recall* sehingga jika digabungkan antara augmentasi data dan *hyperparameter tuning* pada optimasi model *CNN* akan makin memperkecil resiko *overfitting*. Sayangnya, dapat terlihat pada tabel di atas walaupun sudah menggunakan augmentasi data dan *hyperparameter tuning* untuk optimasi model *CNN* tetap saja perbedaan nilai akurasi data latih dengan akurasi validasi cukup jauh yaitu hampir 33%.

## 5 Kesimpulan

Berdasarkan hasil penelitian klasifikasi jenis batik berdasarkan motifnya menggunakan model algoritma *CNN* dengan augmentasi data dan *hyperparameter tuning* dapat ditarik kesimpulan bahwa penelitian ini dapat dijadikan sebagai dasar pembuatan aplikasi klasifikasi jenis batik berdasarkan motifnya, sehingga dapat membantu banyak orang untuk lebih mengenal lagi jenis-jenis batik berdasarkan motifnya, walaupun begitu klasifikasi jenis batik berdasarkan motifnya dengan jumlah kelas atau jenis batik sebanyak 15 kelas dengan menggunakan algoritma *CNN* dengan augmentasi data dan *hyperparameter tuning* masih mengalami *overfitting*, dimana nilai akurasi data training sebesar 99,48% sedangkan nilai akurasi validasi, presisi dan *recall* hanya sebesar 66,67%. Hal ini bisa disebabkan karena 3 hal yaitu, langkah *preprocessing* yang kurang, struktur model *CNN* yang kurang tepat, dan kurangnya data pada dataset yang menyebabkan kurangnya data latih yang dapat dilatih pada model. Selain itu, hal yang paling berparuh signifikan pada kenaikan akurasi validasi, presisi dan *recall* adalah augmentasi data dibandingkan dengan *hyperparameter tuning*, dimana akurasi validasi, presisi dan *recall* klasifikasi batik menggunakan algoritma *CNN* tanpa augmentasi data dan *hyperparameter tuning* meningkat secara signifikan dari 28, 15% menjadi 64% ketika menggunakan augmentasi data tanpa *hyperparameter tuning*, namun ketika hanya menggunakan *hyperparameter tuning* tanpa augmentasi data peningkatan data nilai akurasi validasi, presisi dan *recall* hanya mencapai 33% saja.

## Referensi

- [1] A. P. Siregar *et al.*, “Upaya Pengembangan Industri Batik di Indonesia,” *Dinamika Kerajinan dan Batik: Majalah Ilmiah*, vol. 37, no. 1, 2020, doi: 10.22322/dkb.v37i1.5945.
- [2] A. Amaris Trixie, “Filosofi Motif Batik Sebagai Identitas Bangsa Indonesia,” *Journal of Design and Creative Industry*, vol. 1, no. 1, pp. 1–9, 2020, [Online]. Available: <https://journal.uc.ac.id/index.php/FOLIO/article/view/1380/1148>
- [3] D. G. T. Meranggi, N. Yudistira, and Y. A. Sari, “Batik Classification Using Convolutional Neural Network with Data Improvements,” *International Journal on Informatics Visualization*, vol. 6, no. 1, pp. 6–11, 2022, doi: 10.30630/joiv.6.1.716.
- [4] M. A. Rasyidi, R. Handayani, and F. Aziz, “Identification of batik making method from images using convolutional neural network with limited amount of data,” *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 3, pp. 1300–1307, 2021, doi: 10.11591/eei.v10i3.3035.
- [5] T. Arifin and N. H. Surya, “Klasifikasi X - Ray Pneumonia dengan Metode Convolutional Neural Network menggunakan Arsitektur VGG X-Ray,” 2022. [Online]. Available: <http://sistemasi.ftik.unisi.ac.id>
- [6] J. Sanjaya and M. Ayub, “Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 6, no. 2, pp. 311–323, 2020, doi: 10.28932/jutisi.v6i2.2688.
- [7] C. U. Khasanah, A. K. Pertiwi, and F. Witamajaya, “Implementasi Data Augmentation Random Erasing dan GridMask pada CNN untuk Klasifikasi Batik Implementation of Random Erasing and GridMask Data Augmentations on CNN for Batik Classification,” *Sisfotenika, Jurnal*, vol. 13, no. 1, pp. 16–28, 2023.
- [8] A. E. Minarno, M. H. C. MANDIRI, and M. R. ALFARIZY, “Klasifikasi COVID-19 menggunakan Filter Gabor dan CNN dengan Hyperparameter Tuning,” *ELKOMIKA: Jurnal Teknik Energi Elektrik*,

- Teknik Telekomunikasi, & Teknik Elektronika*, vol. 9, no. 3, p. 493, 2021, doi: 10.26760/elkomika.v9i3.493.
- [9] W. Nugraha and A. Sasongko, "Hyperparameter Tuning on Classification Algorithm with Grid Search," *SISTEMASI*, vol. 11, no. 2, p. 391, May 2022, doi: 10.32520/stmsi.v11i2.1750.
- [10] I. A. Anjani, Y. R. Pratiwi, and S. Norfa Bagas Nurhuda, "Implementation of Deep Learning Using Convolutional Neural Network Algorithm for Classification Rose Flower," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Mar. 2021. doi: 10.1088/1742-6596/1842/1/012002.
- [11] S. I. Pradika, B. Nugroho, and E. Y. Puspaningrum, "Pengenalan Tulisan Tangan Huruf Hijaiyah menggunakan Convolution Neural Network Dengan Augmentasi Data," *Prosiding Seminar Nasional Informatika Bela Negara*, vol. 1, pp. 129–136, 2020, doi: 10.33005/santika.v1i0.35.
- [12] R. Mawan, "Klasifikasi Motif Batik menggunakan Convolutional Neural Network," *Jnanaloka*, pp. 45–50, 2020, doi: 10.36802/jnanaloka.2020.v1-no1-45-50.
- [13] D. Trimakno and Kusriani, "Impact of Augmentation on Batik Classification using Convolution Neural Network and K-Nearest Neighbor," *ICOIACT 2021 - 4th International Conference on Information and Communications Technology: The Role of AI in Health and Social Revolution in Turbulence Era*, pp. 285–289, 2021, doi: 10.1109/ICOIACT53268.2021.9564000.
- [14] A. E. Minarno, M. H. C. Mandiri, and M. R. Alfarizy, "Klasifikasi COVID-19 menggunakan Filter Gabor dan CNN dengan Hyperparameter Tuning," *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 9, no. 3, p. 493, 2021, doi: 10.26760/elkomika.v9i3.493.
- [15] S. H. Sahir, *Metodologi Penelitian*. 2021. Accessed: Jul. 2, 2023. [Online]. Available: [www.penerbitbukumurah.com](http://www.penerbitbukumurah.com)
- [16] A. Alfian, "Indonesian Batik Motifs (Corak App)." Accessed: Jul. 22, 2023. [Online]. Available: <https://www.kaggle.com/datasets/alfanme/indonesian-batik-motifs-corak-app>
- [17] R. Kurniawan, P. B. Wintoro, Y. Mulyani, and M. Komarudin, "Implementasi Arsitektur Xception pada Model Machine Learning Klasifikasi Sampah Anorganik," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 11, no. 2, Apr. 2023, doi: 10.23960/jitet.v11i2.3034.
- [18] F. Parajun, R. Nur Aziza, and D. Kuswardani, "Implementasi Algoritma Convolutional Neural Network Dalam Mengklasifikasi Kesegaran Buah Berdasarkan Citra Buah," *KILAT: Jurnal Kajian Ilmu dan Teknologi*, vol. 11, no. 1, 2022, doi: 10.33322/kilat.v11i1.1458.