

KLASIFIKASI JENIS BATIK BOMBA MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK DENGAN ARSITEKTUR EFFICIENT-NET B2 (BATIK BOMBA SULAWESI TENGAH)

Julian Witjaksono^{*1)}, Mohammad Yazdi Pusadan²⁾, Yusuf Anshori³⁾, Rizka Ardiansyah⁴⁾, Ryfial Azhar⁵⁾

1. Teknik Informatika, Fakultas Teknik, Universitas Tadulako, Indonesia
2. Teknik Informatika, Fakultas Teknik, Universitas Tadulako, Indonesia
3. Teknik Informatika, Fakultas Teknik, Universitas Tadulako, Indonesia
4. Teknik Informatika, Fakultas Teknik, Universitas Tadulako, Indonesia
5. Teknik Informatika, Fakultas Teknik, Universitas Tadulako, Indonesia

Article Info

Kata Kunci: Batik, *Convolutional Neural Networks*, *EfficientNet-B2*, Augmentasi Data, Adamax, Klasifikasi Citra

Keywords: *Batik, Convolutional Neural Networks, EfficientNet-B2, Data Augmentation, Adamax, Image Classification*

Article history:

Received 19 Juli 2024

Revised 3 September 2024

Accepted 2 Oktober 2024

Available online 1 September 2025

DOI :

<https://doi.org/10.29100/jupi.v10i3.6191>

* Corresponding author.

Corresponding Author

E-mail address:

boiwick@gmail.com

ABSTRAK

Batik adalah salah satu warisan budaya Indonesia yang diakui oleh UNESCO sebagai warisan dunia. Keanekaragaman motif batik mencerminkan kekayaan budaya dan seni yang perlu dilestarikan. Salah satu motif batik yang unik adalah batik Bomba dari Kabupaten Donggala, Sulawesi Tengah. Untuk membantu mengklasifikasikan motif batik yang beragam, penelitian ini menggunakan metode *Convolutional Neural Network (CNN)* dengan arsitektur *EfficientNet-B2*. Penelitian ini melibatkan pengumpulan 21 citra batik Bomba dari berbagai sumber di Kota Palu, Sulawesi Tengah. Proses data preprocessing dilakukan melalui teknik augmentasi data, sementara model dikembangkan dengan menggunakan transfer learning dan beberapa teknik optimisasi seperti batch normalization, regulasi, dropout layer, dan fungsi aktivasi ReLU serta softmax. Model dilatih dengan optimizer Adamax dan early stopping untuk mencegah overfitting. Hasil pelatihan menunjukkan akurasi tinggi sebesar 100% pada data pelatihan dan 99.59% pada data validasi. Pengujian menggunakan *confusion matrix* menunjukkan akurasi total model sebesar 96%, dengan kesalahan klasifikasi minimal pada gambar "maleo". Model ini berhasil mengklasifikasikan motif batik Bomba dengan tingkat akurasi yang tinggi, menunjukkan potensi besar penggunaan teknologi kecerdasan buatan dalam pelestarian dan pengembangan warisan budaya batik.

ABSTRACT

Batik is one of Indonesia's cultural heritages recognized by UNESCO as a world heritage. The diversity of batik motifs reflects the rich culture and art that needs to be preserved. One unique batik motif is Bomba batik from Donggala Regency, Central Sulawesi. To help classify the diverse batik motifs, this research employs the *Convolutional Neural Network (CNN)* method with the *EfficientNet-B2* architecture. This study involves collecting 21 Bomba batik images from various sources in Palu City, Central Sulawesi. The data preprocessing process is carried out through data augmentation techniques, while the model is developed using transfer learning and several optimization techniques such as batch normalization, regulation, dropout layers, and ReLU and softmax activation functions. The model is trained with the Adamax optimizer and early stopping to prevent overfitting. Training results show a high accuracy of 100% on training data and 99.59% on validation data. Testing using a *confusion matrix* shows an overall model accuracy of 96%, with minimal misclassification errors on "maleo" images. This model successfully classifies Bomba batik motifs with a high degree of accuracy, demonstrating the great potential of using artificial intelligence technology in the preservation and development of batik cultural heritage.

I. PENDAHULUAN

Batik merupakan salah satu warisan negara Indonesia yang sangat berharga dan telah diakui oleh *UNESCO* (*The United Nations Educational, Scientific and Cultural Organization*) sebagai warisan dunia. Batik juga memiliki keanekaragaman yang kaya, seperti tercermin dalam perkembangan seni batik yang semakin meluas. Pakaian batik juga tidak jarang di jumpai pada masyarakat Indonesia, tidak hanya terbatas pada situasi formal melainkan juga merambah ke pakaian sehari-hari. Hal ini terlihat di masyarakat Indonesia dari peningkatan jumlah pemakaian batik [1].

Selain itu, Indonesia memiliki 38 motif batik yang tersebar di seluruh penjuru tanah air. Salah satunya ialah batik Bomba. Batik Bomba merupakan hasil inspirasi dari kekayaan kain tradisional Kabupaten Donggala, yaitu Tenun Donggala. Sejak tahun 2009, batik Bomba mulai dikembangkan menjadi industri kerajinan lokal dan telah menjadi salah satu ikon dari Sulawesi Tengah [2].

Keberagaman motif ini bukan hanya mencerminkan warisan kultural, tetapi juga mewakili kemajuan seni batik sebagai simbol kebanggaan nasional yang patut dilestarikan. Pertumbuhan dan evolusi batik tidak hanya terbatas sebagai bentuk seni tradisional semata, melainkan telah merambah sebagai simbol kebanggaan dan warisan berharga yang kini semakin mendapat pengakuan.

Dengan peningkatan penggunaan batik yang meluas, tampak jelas bahwa batik bukan sekadar pakaian tradisional, melainkan telah menjadi bagian integral dari gaya hidup modern. Semua ini mencerminkan betapa esensialnya pelestarian dan pengembangan seni batik di tengah dinamika budaya Indonesia yang terus berkembang.

Dikarenakan banyaknya keanekaragaman jenis batik dan motif batik, menjadi tantangan bagi warga Indonesia dan juga penulis dalam membedakan antara satu jenis motif dengan yang lainnya secara tepat. Hal ini menyebabkan hambatan dalam memahami serta menghargai kekayaan budaya yang terkandung dalam setiap motif batik.

Klasifikasi adalah suatu bentuk analisis data yang digunakan untuk menghasilkan model yang menjelaskan pengelompokan atau klasifikasi data. Dalam klasifikasi, dibangunlah klasifikasi atau model untuk memprediksi label kelas (kategori), seperti prediksi cuaca hujan atau cerah. Akurasi merupakan metrik yang digunakan untuk mengevaluasi hasil dari model klasifikasi. Akurasi dihitung dengan membagi jumlah prediksi model yang benar dengan total jumlah prediksi yang dilakukan [3].

AI atau kecerdasan buatan mengacu pada sistem komputer yang dirancang untuk melaksanakan tugas-tugas yang biasanya menuntut kecerdasan dan kognisi seperti manusia [4]. Pada teknologi AI atau sebuah kecerdasan buatan, di tuntut supaya dapat membuat sebuah keputusan dengan melalui beberapa proses di dalamnya. seperti menganalisis sebuah data yang telah di proses agar mesin atau kecerdasan buatan dapat mempelajari atau menganalisis data tersebut dan dapat di implementasikan terhadap data baru agar dapat di ambil sebuah kesimpulan dari apa yang di pelajari oleh model sebelumnya [4].

Teknologi kecerdasan buatan memiliki beberapa subdisiplin, termasuk *Machine Learning* (ML). *Machine Learning* memungkinkan sistem untuk belajar dan beradaptasi dari data yang diberikan (*model training*). Setelah proses pelatihan selesai, model yang dihasilkan dapat digunakan untuk memecahkan berbagai permasalahan. Dengan teknologi ini, komputer dapat menirukan dan menggantikan perilaku manusia dalam menyelesaikan tugas-tugas tertentu secara otomatis [5].

Dalam beberapa kasus, pendekatan *Machine Learning* dapat menjadi semakin kompleks tergantung pada tujuan dan permasalahan yang ingin dipecahkan. Salah satu teknik *Machine Learning* yang dapat digunakan untuk menangani masalah-masalah kompleks adalah *Deep Learning* (DL). *Deep Learning* sendiri merupakan cabang *Machine Learning* yang terinspirasi oleh struktur otak manusia. Dalam *Deep Learning*, diterapkan jaringan syaraf buatan dengan banyak lapisan tersembunyi (*hidden layers*) [6].

Deep learning juga merupakan sebuah algoritma jaringan saraf tiruan yang memanfaatkan metadata sebagai input dan mengolahnya melalui lapisan tersembunyi (*hidden layers*). Lapisan tersembunyi ini berfungsi untuk mempelajari fitur-fitur berdasarkan nilai output dari lapisan sebelumnya. Lapisan-lapisan tersebut memproses dan mentransformasikan data input secara bertahap sehingga dapat mengenali pola atau fitur yang lebih tinggi dan abstrak [7].

Artificial Neural Network (ANN) merupakan model komputasi yang terinspirasi dari oleh cara kerja sistem saraf biologis pada otak manusia [8]. Otak manusia bekerja seperti mesin yang melakukan pemrosesan informasi dengan operasi yang kompleks dengan itu setiap jaringan pada otak dapat berkoordinasi dengan sangat mudah dalam memproses informasi [8]. Otak memiliki desain unik yang memungkinkannya untuk memproses informasi. Struktur tersebut terdiri dari banyak *neuron* kompleks yang saling terhubung dan bekerja bersama untuk memecahkan berbagai masalah sehari-hari [8].

Artificial Neural Network (ANN) menerapkan konsep neuron pada model komputasi di mana neuron buatan dihubungkan dan disusun dalam lapisan-lapisan untuk membentuk jaringan yang kompleks. Fungsi dari jaringan tersebut ditentukan oleh pembelajaran dan koneksi antar neuron [9].

Convolutional Neural Network (CNN) adalah salah satu arsitektur jaringan saraf tiruan (Artificial Neural Network) yang dirancang secara khusus untuk melakukan pemrosesan dan ekstraksi informasi dari kumpulan data, terutama data berbentuk citra atau gambar. Salah satu keunggulan utama *CNN* adalah kemampuannya dalam memproses dan mengekstrak informasi dari data visual dengan efektif.

CNN dirancang untuk secara efektif mengenali pola dan fitur dalam data citra menggunakan lapisan-lapisan konvolusi dan pemilihan fitur yang dilakukan secara otomatis. Dengan struktur dan mekanisme khususnya, *CNN* telah menjadi metode yang sangat sukses dalam berbagai tugas pengolahan citra seperti deteksi objek, klasifikasi, segmentasi, dan rekonstruksi citra [10]. Model arsitektur pada metode *CNN* juga dapat dikembangkan dengan menggunakan *transfer learning*. *Transfer learning* pada *CNN* memanfaatkan model yang sudah dilatih sebelumnya oleh orang lain atau peneliti menggunakan dataset yang sangat besar, seperti dataset *ImageNet*. Dengan memanfaatkan pengetahuan yang telah dipelajari oleh model tersebut, *transfer learning* memungkinkan penggunaan model yang sudah terlatih untuk tugas-tugas pengolahan citra baru dengan dataset yang lebih kecil atau spesifik. Hal ini mempercepat proses pengembangan model *CNN* dan meningkatkan kinerja model dalam tugas-tugas pengenalan citra [10].

Efficient Net merupakan salah satu arsitektur *transfer learning* dan salah satu arsitektur yang memanfaatkan *Convolutional Neural Network (CNN)* yang diklaim memiliki efisiensi dan kinerja yang unggul dibandingkan dengan model-model sebelumnya. Keunggulan ini dapat dicapai melalui penerapan skala yang disesuaikan, di mana kedalaman, resolusi, dan lebar model diubah secara proporsional. Dalam praktiknya, penyesuaian ini menghasilkan penambahan parameter yang relatif kecil, yang pada gilirannya mengakibatkan waktu komputasi yang lebih cepat [11].

Dapat di simpulkan bahwa, *transfer learning* merupakan sebuah teknik yang menggunakan model yang sudah dilatih sebelumnya (*pre-trained*) pada suatu dataset untuk menyelesaikan masalah yang serupa [7]. Model tersebut dilatih dengan menggunakan dataset yang sangat besar, seperti dataset *ImageNet*. Teknik ini menggunakan model *pre-trained* sebagai titik awal, lalu memodifikasi dan memperbarui parameternya agar sesuai dengan dataset masalah baru [7]. Pada praktiknya model *transfer learning* (*pre-trained*) di load. Kemudian, di lakukan penyesuaian dan penyempurnaan pada parameter dari model *pre-trained* agar sesuai dengan dataset baru, dan tugas yang di berikan pada model. Lalu model *pre-trained* diterapkan pada dataset baru untuk mendapat output dari tugas yang di berikan [7].

Penerapan *transfer learning* pada penelitian ini memiliki kelebihan yaitu, secara signifikan dapat mengurangi waktu dan bobot komputasi yang diperlukan untuk melatih model, dan juga penerapan *transfer learning* dapat menghasilkan kinerja yang lebih baik dibandingkan dengan melatih model tanpa menggunakan *transfer learning* (pembuatan model secara manual).

Pada penelitian yang di lakukan oleh [12] dengan menggunakan metode *Machine Learning* tradisional yang bertujuan untuk melakukan klasifikasi Motif Batik Karawang menggunakan algoritma *K-Nearest Neighbor (KNN)* sebagai model machine learningnya hasil dari penelitian tersebut dapat dihasilkan akurasi terbaik, yaitu 95% dengan melakukan kombinasi metode untuk ekstraksi ciri citra dalam manipulasi warna RGB, HSV, GLCM, LBP (*Local Binary Pattern*) serta melakukan pembagian data sebesar 80% data latih dan 20% data test, [12]. Ini menunjukkan bahwa pemilihan fitur ekstraksi yang tepat sangat penting untuk meningkatkan akurasi model. Penelitian ini berhasil menunjukkan bahwa kombinasi *PCA* dan *KNN* adalah metode yang efektif untuk klasifikasi citra motif batik.

Penelitian lain yang berbeda yang di teliti oleh [1] dengan menggunakan algoritma *Support Vector Machine (SVM)* dapat menghasilkan nilai akurasi terbaik sebesar 96% dengan memanfaatkan ekstraksi fitur *GLCM*, pada prosesnya dilakukan serangkaian percobaan klasifikasi menggunakan algoritma Multiclass Support Vector Machine dan metode *GLCM* dan *DWT* sebagai teknik ekstraksi fitur. Penelitian ini terbagi menjadi tiga tahapan. Pada tahap pertama, data dibagi menjadi tiga kelas dan dilakukan pembagian data latih sebesar 70% dan data uji sebesar 30%, lalu 75% data latih dan 25% data uji, serta 80% data latih dan 20% data uji. Pada tahap ini, klasifikasi hanya dilakukan pada dua kelas batik. Pada tahap kedua, pembagian data tetap sama dengan tahap pertama, namun klasifikasi dilakukan pada tiga jenis batik. Pada tahap ketiga, pembagian data juga sama dengan tahap sebelumnya, namun klasifikasi dilakukan pada empat jenis batik [1].

Penggunaan *Gray Level Co-occurrence Matrix (GLCM)*, *Discrete Wavelet Transform (DWT)*, dan algoritma Support Vector Machine (SVM) dalam klasifikasi batik [1] memiliki alasan yang kuat dan saling melengkapi yang mana *GLCM* efektif dalam mengekstrak fitur tekstur dari citra batik dengan menghitung frekuensi pasangan piksel pada jarak dan orientasi tertentu. Sedangkan *DWT* menawarkan analisis multi-resolusi yang memungkinkan

ekstraksi informasi detail dan global secara bersamaan, serta mengurangi dimensi data tanpa kehilangan informasi penting [1]. Sementara itu, penggunaan algoritma SVM pada penelitian “*Batik Classification using Texture Analysis and Multiclass Support Vector Machine*” dikenal karena kinerja model yang tinggi dalam klasifikasi, terutama dalam masalah dengan dimensi tinggi [1].

Tidak hanya memanfaatkan *Machine Learning* tradisional metode *Deep Learning* juga implementasikan dalam beberapa penelitian untuk mendapatkan hasil serta proses pelatihan yang lebih kompleks, pada penelitian yang dilakukan oleh [11] pada pengklasifikasian jenis batik juga dilakukan dengan menggunakan metode *Deep Learning*. Dengan memanfaatkan arsitektur *Efficient Net-B2* dan *Transfer Learning* dapat dihasilkan akurasi sebesar 86.3% dengan menambahkan metode *fine-tuning* serta penambahan proses augmentasi pada data [11]. Penelitian lain yang diteliti oleh [13], dengan menggunakan arsitektur *VGG16* dapat menghasilkan nilai akurasi 95.83% tanpa menggunakan data *augmentasi* sedangkan dengan menggunakan data augmentasi nilai akurasinya naik sebanyak 3.13% menjadi 98.96% [13].

Pada penelitian lain, yang diteliti oleh [14] penelitian tersebut memiliki fokus untuk meneliti beberapa pola dari beberapa batik yaitu : Kawung, Mega Mendung, Parang, Sekar Jagat, Truntum, pada dasarnya data yang digunakan berupa gambar citra batik dengan 5 kelas yaitu : Kawung, Mega Mendung, Parang, Sekar Jagat, dan Truntum. Agar proses pelatihan model klasifikasi memiliki performa yang baik maka perlu juga dilakukan data processing, berikut beberapa step yang dilakukan dalam penelitian ini yaitu : *Image Filtering*, *Image Labeling*, *Image Resizing* dan *Image Normalization*. Agar dapat dilakukannya training model maka perlu dilakukan splitting data, untuk pembagian data yaitu : training, validation, and testing dengan komposisi 60% (1961 + augmentasi), 20% (487) dan 20% (487). Data augmentasi digunakan guna untuk menambahkan variasi data dan juga menghindari overfitting pada model. Metode klasifikasi yang digunakan dalam penelitian ini yaitu menyangkut beberapa metode yaitu: *Method 1 (VGG16 Modified: 11 Conv Layer + 4 MaxPool + Global Average Pooling)*, *Method 2 (VGG16 Modified: 8 Conv Layer + 3 MaxPool + Global Average Pooling)*. Untuk penelitian dengan menggunakan metode pertama dengan best model berada pada 26th epoch maka dapat dihasilkan nilai yaitu : *Training loss: 0.0479 – Training accuracy: 0.9857, Validation loss: 0.3097 – Validation accuracy: 0.9369* dan untuk besaran dari testing yaitu: *Testing loss = 0.187* dan *Testing accuracy = 0.963*. sedangkan pada metode kedua dengan best model berada pada 29th epoch maka dapat dihasilkan nilai yaitu : *Training loss: 0.1917 – Training accuracy: 0.9347, Validation loss: 0.4492 – Validation accuracy: 0.8635* dan untuk nilai dari data testing yaitu : *Testing loss = 0.206 and Testing accuracy = 0.926* [14] penelitian ini dapat dihasilkan kesimpulan bahwa nilai akurasi yang diperoleh mencapai 96,30% yang ada pada metode pertama [14]. pemilihan arsitektur *Vgg16* sebagai model *machine learning* dikarenakan *Vgg16* dapat menunjukkan kemampuan generalisasi yang baik, yang sangat penting mengingat variasi dalam gambar batik yang dapat muncul dari pencahayaan, sudut, dan latar belakang yang berbeda [14]. hasil penelitian tersebut menunjukkan bahwa *Vgg16* mampu memberikan akurasi yang tinggi dalam klasifikasi citra, menjadikannya pilihan yang tepat untuk meningkatkan akurasi klasifikasi batik [14].

Berdasarkan penjelasan tentang beberapa penelitian terkait tersebut, menunjukkan bahwa metode *deep learning* cenderung menghasilkan akurasi yang lebih tinggi dibandingkan tradisional machine learning untuk klasifikasi motif batik. Metode tradisional dalam klasifikasi batik, seperti yang digunakan dalam penelitian oleh [12] dan [1], melibatkan algoritma *Machine Learning* klasik seperti *K-Nearest Neighbor (KNN)* dan *Support Vector Machine (SVM)*. Sedangkan metode modern, seperti *Deep Learning*, menggunakan arsitektur jaringan saraf tiruan yang lebih kompleks seperti *Convolutional Neural Networks (CNN)*. Maka dari itu penggunaan metode machine learning tradisional tidak selalu efisien dikarenakan metode modern machine learning dapat memberikan hasil yang lebih akurat dan efisien dibandingkan metode tradisional machine learning, hal ini didukung oleh penelitian-penelitian terkait seperti Penelitian [12] yang menggunakan metode tradisional *Machine Learning (K-Nearest Neighbor)* dan mencapai akurasi 95% dalam klasifikasi motif batik Karawang, penelitian [1] menggunakan metode tradisional *Machine Learning (Support Vector Machine)* dan mencapai akurasi 96% dalam klasifikasi motif batik, dan Penelitian [11] dan [14] menggunakan metode *Deep Learning (Efficient Net-B2 dan VGG16)* dan mencapai akurasi yang lebih tinggi (86,3% dan 98,96% dengan data augmentasi) dalam klasifikasi motif batik.

Dengan memanfaatkan *CNN* dan metode transfer learning pada beberapa penelitian terdahulu dibuktikan bahwa metode *CNN* memiliki proses yang lebih kompleks dari pada tradisional *Machine Learning*, Metode *Convolutional Neural Network (CNN)* terbukti efektif dalam pengolahan citra digital. Banyak penelitian telah mengungkapkan penggunaan *CNN* dapat menghasilkan tingkat akurasi di atas 70% dalam berbagai aplikasi pengolahan citra [10]. Hal ini menunjukan, bahwa dengan menggunakan metode *CNN* dapat menghasilkan kinerja lebih baik dalam melakukan klasifikasi terhadap batik.

II. METODE PENELITIAN

A. Batik Dataset

Citra Batik di gunakan dalam penelitian ini, data citra batik ini di ambil secara langsung dari toko-toko kain batik Bomba di Kota Palu, Sulawesi Tengah, serta dari orang-orang dan warga lokal Kota Palu yang memiliki kain batik bomba. Secara keseluruhan, peneliti memperoleh 21 citra gambar. Dalam proses pengumpulan data, peneliti melakukan observasi dan mencatat informasi tentang motif batik yang tersedia, termasuk pola, warna, jenis kain, dan atribut lain yang membedakan satu motif batik dengan yang lainnya. Data yang diperoleh dari pengamatan ini menjadi sumber utama untuk menganalisis karakteristik dan variasi motif batik bomba yang ada. Contoh data batik dapat di lihat pada Gambar 1.



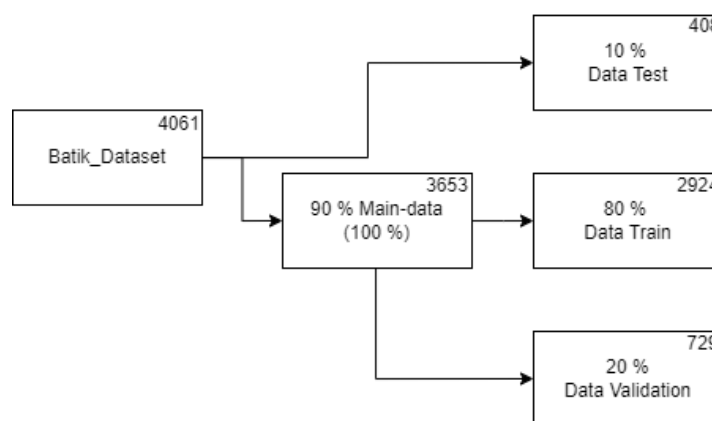
Maleo Mawar Kelor

Gambar. 1. Data Citra batik bomba motif maleo, mawar, kelor

B. Data Preprocessing

Data preprocessing merupakan sebuah teknik yang umum dimanfaatkan dalam (*Deep Learning*) [15]. Pada penelitian ini, digunakan teknik augmentasi data (*data augmentation*). Penambahan augmentasi data bertujuan untuk meningkatkan variasi pada data citra, dengan tujuan akhir untuk memperkaya data latih [13]. Dalam penelitian ini, akan diimplementasikan tujuh jenis teknik augmentasi data yaitu *rotation_range*, *Width_shift_range*, *height_sift_range*, *shear_range*, *zoom_range*, *horizontal_flip*, dan *fill_mode*. Pada *augmentasi rotation_range* data gambar dilakukan rotasi sudut secara acak dari 0° hingga 45° , *Width_shift_range* digunakan untuk mengatur posisi citra lebar gambar. Digunakan nilai 0.2 untuk menunjukkan bahwa gambar dapat secara acak bergeser hingga 20% dari lebar awal gambar, *height_sift_range* sama seperti fungsi *Width_shift_range* tetapi mengatur panjang dari citra, pada *shear_range* nilai dari parameternya adalah 0.2 Menunjukkan bahwa gambar dapat dikenakan perubahan sudut (*shear*) dengan maksimal 0.2 radian atau sekitar $11,5$ derajat dalam arah berlawanan dengan arah jarum jam. Selanjutnya gambar di lakukan perbesaran secara acak sebesar 0.2 dengan parameter *zoom_range*, Nilai 0.2 menunjukkan seberapa besar intensitas dari transformasi, kemudian gambar citra di lakukan *horizontal_flip* yang berguna untuk melakukan pembalikan gambar secar ahorizontal secar acak, terakhir untuk mengisi bagian citra yang kosong di gunakan parameter *fill_mode*.

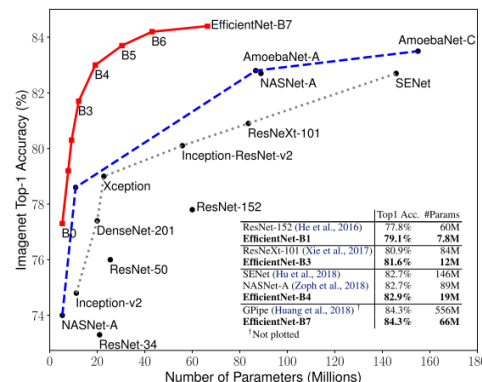
Setelah proses augmentasi selesai, selanjutnya dilakukan proses data splitting. Proses ini diterapkan pada data training, validation, dan testing. Pada awalnya, data yang telah di lakukan augmentasi di lakukan pembagian sebesar 90% untuk data pelatihan dan 10% untuk data *testing*, selanjutnya pada data pelatihan di bagi kembali menjadi 80% dan 20% untuk training dan validation. Pembagian data (*Data splitting*) dapat di lihat pada Gambar 2.



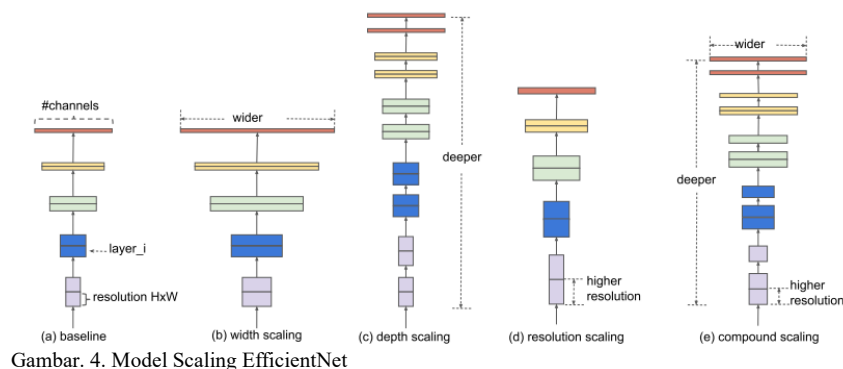
Gambar. 2. Data Splitting

C. Perancangan model

Penelitian ini menerapkan metode *Convolutional Neural Network (CNN)* dengan menggunakan arsitektur EfficientNet-B2. EfficientNet-B2 merupakan varian arsitektur EfficientNet, yang telah terbukti efektif dalam tugas pengenalan pola pada citra. Penelitian sebelumnya [16] menunjukkan bahwa EfficientNet mampu mencapai tingkat akurasi yang tinggi dengan menggunakan jumlah parameter yang lebih sedikit dibandingkan dengan model *ConvNet* lainnya. Selain memiliki akurasi yang tinggi, *EfficientNet* juga memiliki efisiensi pada proses komputasinya. Sebagai contoh, arsitektur *EfficientNet-B3* dapat mencapai akurasi yang lebih tinggi daripada *ResNeXt-101* dengan menggunakan 18 kali lebih sedikit *FLOPS* [16]. Dalam pengujian dengan transfer learning, arsitektur *EfficientNet* terbukti mampu mencapai akurasi yang lebih baik daripada model-model lain seperti *NASNet-A* dan *Inception-v4*, dengan mengurangi jumlah parameter hingga 21x [16]. Detail untuk Grafik model size dengan ImageNet dapat di lihat pada Gambar 3.



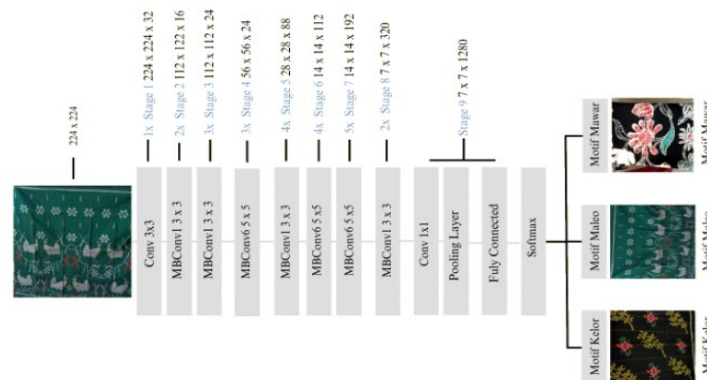
Gambar. 3. Grafik Model Size vs. ImageNet Accuracy



Gambar. 4. Model Scaling EfficientNet

EfficientNet menggunakan metode “*compound scaling*” yang secara bersamaan menyesuaikan tiga dimensi utama dari model: kedalaman (depth), lebar (width), dan resolusi (resolution). *EfficientNet* tidak hanya meningkatkan satu dimensi, pendekatan ini memastikan bahwa semua dimensi diskalakan dengan proporsi yang sama, yang membantu mencapai keseimbangan optimal antara akurasi dan efisiensi [16]. Arsitektur dari metode *compound scaling* dapat di lihat pada Gambar 4. Penggunaan algoritma *EfficientNet* dalam penelitian ini didasarkan pada keunggulan-keunggulan yang dimilikinya terkait kinerja, efisiensi, dan kemampuannya untuk mencapai keseimbangan yang baik antara akurasi dan penggunaan sumber daya. *EfficientNet-B2* dirancang untuk memberikan akurasi tinggi dengan jumlah parameter yang lebih sedikit dibandingkan arsitektur lain seperti *ResNet* atau *Inception*, memungkinkan model mencapai performa yang baik tanpa membutuhkan sumber daya komputasi besar. Dengan metode *compound scaling*, *EfficientNet-B2* dapat menyesuaikan kedalaman, lebar, dan resolusi secara bersamaan, membantu model belajar dari data secara lebih efektif dan meningkatkan akurasi tanpa menambah kompleksitas model secara berlebihan. Dari segi kinerja, EfficientNet-B2 memberikan hasil yang lebih baik dibandingkan model-model lain yang ditampilkan pada Gambar 3. Ini termasuk peningkatan akurasi yang signifikan dengan pengurangan jumlah parameter, menjadikannya lebih praktis untuk digunakan dalam aplikasi yang memerlukan efisiensi tinggi. Sebaliknya, model machine learning tradisional memiliki beberapa kekurangan, seperti algoritma KNN yang efisiensi pembelajarannya kurang optimal karena semua perhitungan dilakukan saat klasifikasi, bukan saat data pelatihan pertama kali diperoleh. Hal ini membuat KNN kurang efisien, terutama untuk repositori data besar atau aplikasi yang memerlukan klasifikasi dinamis. KNN juga memiliki waktu klasifikasi yang

lama dengan kompleksitas $O(n^2)$ untuk mengklasifikasikan instance baru, serta ketergantungan pada pemilihan nilai k yang tepat, di mana kesalahan dalam pemilihan k dapat mempengaruhi akurasi klasifikasi. Oleh karena itu, dalam konteks penelitian ini, pemilihan *EfficientNet* didasarkan pada efisiensi dan kinerja yang lebih baik dibandingkan dengan algoritma *machine learning* tradisional.



Gambar. 5. Transfer learning EfficientNet Model Architecture

Berdasarkan penelitian ini, model dasar dari *EfficientNet* dimuat sebagai transfer learning dengan menambahkan layer-layer tambahan untuk pengaturan output pengaturan tersebut dapat mencegah terjadinya overfitting yang disebabkan oleh data yang tidak sempurna. Beberapa pengaturan yang ditambahkan sebagai berikut : *Batch Normalization*, *Regularization learning kernel*, *dropout* layer serta fungsi aktivasi pada layer (*relu* dan *softmax*). Visualisasi arsitektur Model *EfficientNet* (*Transfer Learning*) dapat dilihat pada Gambar 5.

Pada klasifikasi batik bomba, dengan menambahkan layer *Batch Normalization* ke dalam model transfer learning, memiliki dampak signifikan dalam meningkatkan performa model dan mengurangi risiko *overfitting* [17]. *Batch Normalization* berfungsi untuk menormalisasi aktivasi dari layer-layer sebelumnya, yang tidak hanya mempercepat proses pelatihan tetapi juga menjaga stabilitas selama pelatihan [17]. Dengan menormalisasi distribusi aktivasi, model dapat menggunakan laju pembelajaran yang lebih tinggi tanpa risiko divergensi, sehingga mempercepat konvergensi dan memungkinkan model untuk mempelajari representasi data yang lebih kompleks [17]. Selain itu, *Batch Normalization* juga berperan sebagai regularisasi, yang dapat mengurangi kebutuhan akan teknik lain seperti *Dropout*, karena *Batch Normalization* membantu mencegah model terjebak dalam mode saturasi dan mengurangi *overfitting* yang disebabkan oleh data yang tidak sempurna. Dengan demikian, integrasi *Batch Normalization* dalam arsitektur model tidak hanya meningkatkan efisiensi pelatihan tetapi juga memperkuat kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya, sehingga dengan mengintegrasikan *Batch Normalization* pada klasifikasi batik dapat menjadi keunggulan tersendiri dalam pengembangan model deep learning yang lebih robust dan efektif [17].

Regularization memainkan peran penting dalam meningkatkan performa model dan mencegah terjadinya overfitting dan meningkatkan generalisasi terhadap model [18]. *Regularization* digunakan untuk mencegah model dari terlalu menyesuaikan diri dengan data pelatihan yang mungkin tidak sempurna. Dengan mengurangi kompleksitas model, *regularization* membantu meningkatkan generalisasi, sehingga model dapat beradaptasi dengan baik pada data yang belum terlihat [18]. Penambahan layer dropout dalam model neural network berfungsi sebagai teknik regularisasi yang efektif untuk mengatasi masalah overfitting, terutama ketika data pelatihan tidak sempurna atau terbatas. *Dropout* bekerja dengan cara secara acak menghilangkan sejumlah neuron selama proses pelatihan, sehingga mencegah neuron-neuron tersebut saling bergantung satu sama lain. Hal ini mendorong setiap neuron untuk belajar fitur yang lebih umum dan tidak hanya bergantung pada kehadiran neuron lain, yang sering kali menyebabkan co-adaptation yang kompleks. Dengan mengurangi kompleksitas model, *dropout* membantu meningkatkan kemampuan generalisasi model terhadap data yang belum terlihat, sehingga performa pada data uji dapat ditingkatkan [19]. Selain itu, dengan menggabungkan dropout dengan teknik lain seperti *Batch Normalization* dan *regularization learning kernel*, model dapat lebih stabil dan efisien dalam pelatihan, yang pada gilirannya meningkatkan kecepatan konvergensi dan akurasi prediksi [19]. Pada fungsi aktivasi *ReLU* (*Rectified Linear Unit*) ditambahkan pada lapisan *fully connected*. Keunggulan dari fungsi aktivasi ini yaitu dapat memberikan komputasi yang efisien, dapat Meningkatkan Kecepatan Konvergensi [20]. Sehingga aktivasi *ReLU* memungkinkan jaringan untuk belajar representasi yang lebih kompleks dan *non-linear* dari data, serta meningkatkan kecepatan pelatihan dan konvergensi model [20]. Pada layer output digunakan fungsi aktivasi *softmax*, aktivasi *softmax* sendiri

memberikan probabilitas untuk setiap kelas, yang memungkinkan model untuk memprediksi kelas mana yang paling mungkin untuk input tertentu (*multi-class-prediction*) [20]. Dengan demikian, penambahan Batch Normalization, Regularization learning kernel dan dropout tidak hanya berkontribusi pada pengurangan overfitting, tetapi juga meningkatkan performa keseluruhan model dalam tugas klasifikasi, citra batik bomba. Detail model Efficient net yang di gunakan dapat di lihat pada Tabel I.

TABEL I
DESAIN MODEL ARSITEKTUR TRANSFER LEARNING

Layers	Type	Filter / Kernel	Activation Function	Batch Size	Output Shape		
					Rows	Columns	Channels
Input	InputLayer	-	-	None	260	260	3
Blok konvolusi	EfficientNetB2	-	Swiss	None	9	9	1408
Pooling	Global Max-Pooling2D	-	-	None		1408	
Batch Normalization	BatchNormali-zation	-	-	None		1408	
Fully Connected	Dense	256	ReLu	None		256	
Dropout	Dropout	-	-	None		256	
Output	Dense	-	Softmax	None		Sesuai Kategori (3)	

D. Optimisasi dan pelatihan model

Sebelum menjalankan proses pelatihan model, penting untuk mempertimbangkan parameter-parameter penting yang dapat mempengaruhi jalannya pelatihan. Hal ini meliputi pemilihan optimizer yang tepat untuk klasifikasi citra batik, penentuan jumlah *epoch* pelatihan serta melakukan pengaturan terhadap *learning rate*. Dalam penelitian ini, jumlah *epoch* pelatihan yang diterapkan adalah 25 iterasi. *Epoch* berfungsi untuk menentukan berapa kali jaringan saraf akan dilatih pada keseluruhan dataset pelatihan [21]. Selanjutnya, nilai *learning rate* yang digunakan adalah 0,001, nilai *learning rate* yang kecil digunakan untuk membuat jalanya proses dari pelatihan berjalan dengan langkah yang cukup kecil hal itu di maksudkan agar model pelatihan dapat belajar secara perlahan dan agar dapat mempertahankan kan proses dari kestabilan pelatihan model, pemilihan nilai *learning rate* tidak bisa di tentukan secara acak, yang mana Jika *learning rate* terlalu besar, proses pelatihan bisa menjadi tidak stabil dan melompat-lompat di sekitar minimum fungsi loss, menyebabkan model gagal mencapai konvergensi. Dan apabila pemilihan nilai *learning rate* yang terlalu rendah maka proses pelatihan akan berjalan sangat lambat dan mungkin memerlukan banyak epoch untuk mencapai hasil yang maximal [22]. Maka dari itu *learning rate* yang dipilih pada penelitian ini adalah 0,001, yang umumnya merupakan nilai yang baik untuk banyak jenis model dan dataset [22]. Selain itu, model bisa terjebak pada minimum lokal dan tidak mencapai minimum global yang optimal. Maka dari itu Pemilihan fungsi *loss* dan metrik evaluasi juga dapat menentukan performa akhir pada model. *Optimizer* Adamax di gunakan pada penelitian ini, adamax sendiri merupakan varian lain dari *optimizer Adaptive moment optimization* (Adam) [23]. Adamax adalah algoritma optimasi yang dirancang untuk meminimalkan fungsi objektif yang bersifat stokastik (mengandung unsur probabilistik). Ini sangat berguna, di mana fungsi yang dioptimalkan sering kali memiliki banyak parameter dan dapat bervariasi secara signifikan. Dengan kata lain, Adamax adalah teknik optimasi yang efektif untuk menangani model yang kompleks dengan noise tinggi dan dataset yang besar [23]. Demi mendapatkan hasil yang optimal Pemilihan *optimizer* Adamax di maksudkan untuk menangani masalah dengan noise tinggi atau parameter yang sangat besar yang ada pada dataset [23]. Pengaruh Adamax dalam pelatihan model yaitu dapat menyesuaikan *learning rate* untuk setiap parameter secara individual, yang memungkinkan pelatihan yang lebih cepat dan stabil. Ini juga membantu dalam mencapai hasil maximal yang lebih baik, terutama pada dataset dengan variasi yang signifikan. Hal ini di karnakan Adamax menggabungkan keuntungan dari adaptive *learning rate* dengan stabilitas dalam pembaruan bobot [23]. Selanjutnya, pada proses pelatihan fungsi *earlystopping* di gunakan, *early stopping* adalah teknik penting dalam pelatihan model yang berfungsi untuk meningkatkan kemampuan generalisasi dan mencegah overfitting. Dalam konteks pelatihan model untuk penelitian ini, early stopping menghentikan proses pelatihan sebelum mencapai jumlah epoch yang telah ditentukan, berdasarkan kinerja model pada data validasi. Jika model menunjukkan penurunan kinerja, seperti peningkatan loss atau penurunan akurasi, pelatihan dihentikan untuk mencegah model dari memorisasi noise dalam data pelatihan. Dengan demikian, early stopping membantu menjaga keseimbangan antara belajar dari data pelatihan dan mempertahankan kemampuan untuk menggeneralisasi pada data baru [24]. Pendekatan ini sangat relevan ketika menggunakan optimizer seperti Adamax, yang dirancang untuk menangani fungsi objektif yang kompleks dan

noise tinggi, sehingga dapat meningkatkan efektivitas pelatihan. Dengan mengoptimalkan parameter seperti learning rate dan jumlah epoch, serta menerapkan early stopping, model dapat dilatih dengan lebih stabil dan efisien, [24]. Early stopping dapat mengurangi waktu pelatihan dan sumber daya komputasi yang diperlukan. Dengan menghentikan pelatihan ketika kinerja model pada data validasi mulai menurun, early stopping mencegah pelatihan yang berlebihan yang dapat mengakibatkan pemborosan sumber daya [24].

E. Pengujian dan evaluasi model

Pada proses pengujian dan evaluasi, kinerja dari model akan di lakukan pengujian dengan menggunakan *confusion matrix* dan *matrix evaluasi*. Pada penelitian ini, *confusion matrix* digunakan untuk mengevaluasi kinerja model klasifikasi yang memprediksi tiga kelas: Kelor, Maleo, dan Mawar. serta *Confusion matrix* mengukur performansi dari sebuah model klasifikasi, terdapat beberapa parameter performansi yang di gunakan untuk mengukur seberapa baik model klasifikasi antara lain precision, *recall*, f1-score dan akurasi. Untuk mencari nilai-nilai ini, digunakan beberapa istilah seperti *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, dan *False Negative (FN)*, yang kemudian dijelaskan dalam sebuah matriks ordo [25]. *Confusion matrix* pada penelitian ini yaitu *confusion matrix* ordo 3x3 yang dapat dilihat pada Tabel II.

TABEL II
CONFUSION MATRIX 3X3

Confusion matrix		Predicted		
		Kelor	Maleo	Mawar
Actual	Kelor	TP	FP / FN	FP / FN
	Maleo	FP / FN	TP / TN	FP / FN
	Mawar	FP / FN	FP / FN	TP / TN

Dapat di lihat pada Tabel II. *Confusion matrix* bekerja dengan menunjukkan jumlah prediksi yang benar dan salah, yang dibagi menjadi beberapa kategori: True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN). Pada kasus klasifikasi batik bomba nilai True Positive (TP) merupakan jumlah kasus di mana model memprediksi bahwa hasil klasifikasi positif dan hasilnya benar positif., sedangkan True Negative (TN) menampilkan jumlah kasus di mana model bahwa hasil klasifikasi negatif dan hasilnya benar negatif. Sedangkan pada False Positive (FP), dan False Negative (FN) di gunakan untuk menampilkan jumlah kasus di mana model memprediksi bahwa hasil klasifikasi positif tapi, hasilnya negative dan sebaliknya [26].

Akurasi pada *confusion matrix* adalah parameter yang digunakan untuk mengukur tingkat ketepatan sebuah model dalam mengenali data input dengan data output yang benar [26]. Selanjutnya, parameter presisi digunakan untuk mengukur tingkat ketepatan model dalam memberikan informasi sesuai dengan apa yang diminta. Parameter presisi juga akan menunjukkan seberapa sering prediksi positif model benar, pada konteks klasifikasi motif batik di mana salah klasifikasi dapat menurunkan kualitas prediksi. [26]. *Recall* digunakan untuk mengukur tingkat kemampuan model dalam menemukan informasi yang sesuai/relevan. Parameter *Recall* akan menunjukkan seberapa baik model menemukan semua sampel positif yang sebenarnya, penting dalam penelitian ini untuk memastikan semua motif batik dikenali dengan benar. Sedangkan nilai *F1-Score* adalah perhitungan evaluasi dalam information retrieval yang mengkombinasikan antara *recall* dan presisi [26]. F1-Score akan memberikan gambaran keseimbangan/harmonik tentang performa model, penting ketika kita ingin mengevaluasi kinerja model secara keseluruhan dengan mempertimbangkan baik presisi maupun recall [26]. Rumus perhitungan *Akurasi*, *Precision*, *Recall* dan *F1-Score* dapat dilihat pada persamaan (1), (2), (3) dan persamaan (4).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1-Score = 2x \frac{Recall * Precision}{Recall+ Precision} \quad (4)$$

TP = True Positive

FP = False Positive

FN = False Negative

III. HASIL DAN PEMBAHASAN

Hasil pelatihan model menunjukkan kinerja yang menjanjikan pada klasifikasi citra dengan nilai *epoch* yang di gunakan selama pelatihan yaitu 25 *epoch*, dengan memanfaatkan teknik *earlystopping* yang di gunakan untuk melakukan pemberhentian apabila pelatihan model memiliki nilai *loss* yang sama selama beberapa iterasi dan optimizer adamax di gunakan dengan nilai *learning_rate* = 0.001, nilai *learning rate* yang kecil digunakan untuk membuat jalanya proses dari pelatihan berjalan dengan langkah yang cukup kecil hal itu di maksudkan agar model pelatihan dapat belajar secara perlahan dan agar dapat mempertahankan kan proses dari kestabilan pelatihan. Pada pelatihan model awalnya model di latih hingga 25 epoch dengan kinerja model pada data validasi dipantau setelah setiap iterasi epoch. *Early stopping* diatur untuk memantau nilai validation loss dan menghentikan pelatihan jika tidak ada peningkatan kinerja setelah sejumlah epoch tertentu Hal ini bertujuan untuk menghindari overfitting dengan menghentikan pelatihan saat model mulai menghafal data pelatihan dan tidak lagi meningkatkan kemampuannya untuk menggeneralisasi pada data baru. Selama iterasi epoch 1 hingga 19, training loss terus menurun, dan training accuracy meningkat mencapai 100% mulai epoch ke-3.

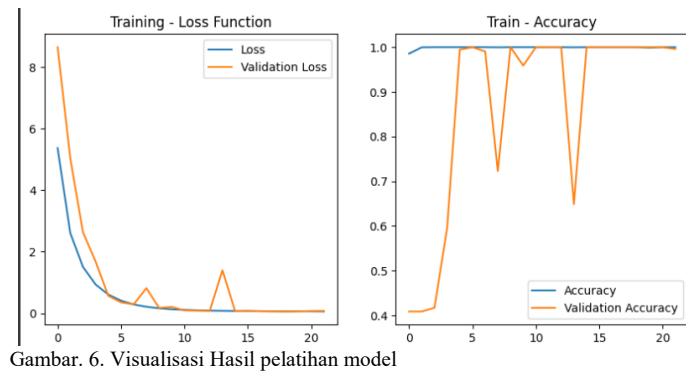
Pada saat yang sama, validation loss menurun secara signifikan dari 8.6450 di epoch pertama menjadi 0.0604 di epoch ke-19, menunjukkan bahwa model semakin baik dalam menggeneralisasi data yang belum dilihat selama pelatihan. Validation accuracy juga meningkat dan mencapai 100% pada beberapa epoch, menandakan bahwa model memiliki kinerja yang sangat baik pada data validasi. Namun, setelah mencapai epoch ke-19, validation loss mulai meningkat lagi (0.0638 pada epoch ke 20, 0.0757 pada epoch ke 21, dan 0.0789 pada epoch 22), sementara validation accuracy sedikit menurun setelah mencapai 100%. Ini menandakan bahwa model mulai overfitting pada data pelatihan dan kehilangan kemampuannya untuk menggeneralisasi pada data validasi.

Proses pelatihan dihentikan pada epoch ke-22 karena validation loss tidak lagi membaik dan malah menunjukkan peningkatan. Early stopping membantu menghentikan pelatihan pada titik optimal untuk menjaga kinerja yang baik pada data validasi dan menghemat waktu serta sumber daya komputasi dengan menghentikan pelatihan sebelum mencapai jumlah epoch maksimum yang telah ditentukan. Proses early stoping yang di tampilkan pada penelitian ini memberikan dampak yang sangat signifikan dalam menjaga keseimbangan antara kinerja model pada data pelatihan dan data validasi, serta efisiensi dalam penggunaan sumber daya komputasi. Dengan diterapkannya early stopping, model berhenti dilatih pada epoch ke-22, sebelum mencapai jumlah epoch maksimum yang ditentukan yaitu 25 epoch. Ini memastikan bahwa model tidak overfitting pada data pelatihan. Proses rincian pada pelatihan model dapat dilihat pada Tabel III.

TABEL III
PELATIHAN MODEL

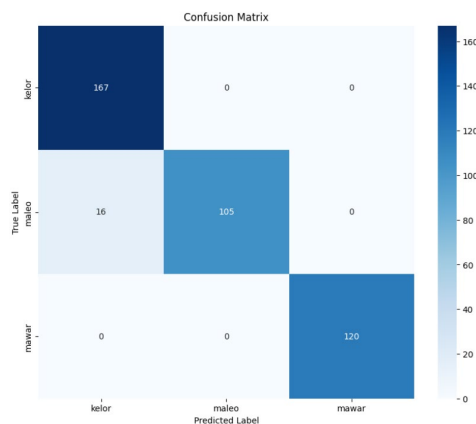
Epoch	Training Loss	Training Accuracy	Validation loss	Validation Accuracy
1	5.3686	0.9856	8.6450	0.4088
2	2.6072	0.9997	5.0370	0.4088
3	1.5082	1.0000	2.6369	0.4170
4	0.9368	1.0000	1.6788	0.5953
5	0.6102	1.0000	0.5706	0.9945
6	0.4129	1.0000	0.3572	1.0000
7	0.2885	1.0000	0.2913	0.9904
8	0.2139	0.9997	0.8189	0.7229
9	0.1644	1.0000	0.1768	1.0000
10	0.1315	1.0000	0.2102	0.9588
11	0.1134	1.0000	0.0956	1.0000
12	0.0983	1.0000	0.0862	1.0000
13	0.0893	1.0000	0.0799	1.0000
14	0.0823	0.9997	1.3991	0.6488
15	0.0766	1.0000	0.0684	1.0000
16	0.0744	1.0000	0.0798	1.0000
17	0.0710	1.0000	0.0683	1.0000
18	0.0684	1.0000	0.0634	1.0000
19	0.0655	1.0000	0.0604	1.0000
20	0.0682	0.9990	0.0638	1.0000
21	0.0653	1.0000	0.0757	1.0000
22	0.0613	1.0000	0.0789	0.9959

Visualisasi pebandingan antara model training *loss* dan validation *loss* serta train accuracy dan validation accuracy dapat di lihat pada Gambar 6 :



Gambar. 6. Visualisasi Hasil pelatihan model

Untuk mendapatkan hasil analisis yang lebih mendalam di lakukan proses evaluasi dengan menggunakan *confusion matrix* dengan menggunakan data pengujian. Hasil dari *confusion matrix* menunjukkan apakah model melakukan klasifikasi dengan baik pada data batik. Hasil pengujian *confusion matrix* model dapat di lihat pada Gambar 7 :

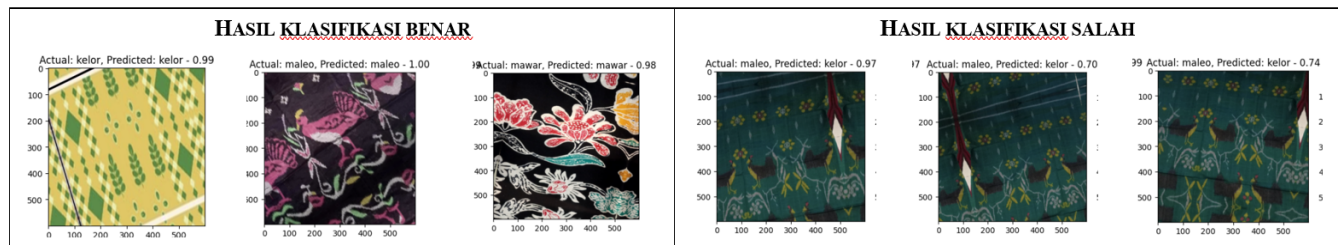


Gambar. 7. Pengujian *Confusion matrix*

Hasil dari pengujian *confusion matrix* pada Gambar 7, memberikan gambaran tentang performa model klasifikasi citra. Matriks ini menunjukkan bahwa model memiliki akurasi yang sangat tinggi dalam mengklasifikasikan gambar "kelor" dan "mawar", dengan 167 gambar "kelor" dan 120 gambar "mawar" diklasifikasikan dengan benar tanpa kesalahan. Namun, terdapat beberapa kesalahan dalam klasifikasi gambar "maleo", di mana 16 gambar "maleo" diklasifikasikan sebagai "kelor", sementara 105 gambar "maleo" diklasifikasikan dengan benar. Kesalahan ini menunjukkan bahwa model mengalami kesulitan dalam membedakan gambar "maleo" dari "kelor". Kesalahan prediksi bisa terjadi dikarenakan kemiripan visual antara gambar "maleo" dan "kelor", yang membuat model kesulitan dalam mengidentifikasi citra secara jelas. Faktor lainnya yang bisa menjadi alasan mengapa mesin mengalami kesalahan prediksi adalah adanya kualitas gambar yang rendah atau adanya noise yang mengganggu dalam gambar kelas "maleo" seperti gambar yang buram, memiliki resolusi rendah, atau mengandung gangguan visual bisa membuat model kesulitan dalam melakukan klasifikasi yang akurat. Kemungkinan lain, yang dapat menyebabkan terjadinya kesalahan dalam klasifikasi pada motif batik "maleo" adalah dikarenakan ketidakseimbangan dalam dataset di mana jumlah gambar "maleo" lebih sedikit dibandingkan dengan "kelor" dan "mawar" bisa menyebabkan model tidak mendapatkan cukup informasi untuk belajar mengenali "maleo" dengan baik, sehingga model cenderung belajar lebih baik pada kategori yang memiliki lebih banyak contoh. Overfitting juga dapat mengganggu system klasifikasi batik yang mana proses tersebut terjadi ketika model terlalu banyak belajar dari data pelatihan dan kurang mampu menggeneralisasi pada data baru. Jika data "maleo" dalam dataset pelatihan tidak cukup beragam, model mungkin gagal mengidentifikasi variasi gambar "maleo" dalam data pengujian. Untuk mengurangi kesalahan klasifikasi pada citra, berikut beberapa strategi dapat diterapkan,

Pertama, dengan melakukan pembersihan data yang di gunakan untuk menghapus noise dan meningkatkan kualitas gambar "maleo" dapat membantu model belajar dari data yang lebih bersih dan informatif, serta menggunakan teknik peningkatan resolusi atau filtering noise. Kedua, menyeimbangkan dataset dengan

menambahkan lebih banyak gambar "maleo" atau menggunakan teknik resampling (*oversampling* atau *undersampling*) dapat membantu model belajar dengan proporsi yang lebih seimbang dan mengurangi bias terhadap kategori yang lebih dominan. Ketiga, menggunakan teknik regularisasi seperti dropout, L2 regularization, dan batch normalization dapat membantu mencegah overfitting, membuat model lebih robust dan mampu menggeneralisasi dengan lebih baik. Berikut beberapa contoh data hasil pengujian model yang terklasifikasi dengan benar dan hasil pengujian model yang terklasifikasi dengan salah dapat yang dapat dilihat pada Gambar 8.



Gambar. 8. Hasil Pengujian Model Klasifikasi

Pada *Clasification report* menunjukkan bahwa model klasifikasi citra batik memiliki performa yang sangat baik secara keseluruhan. *Precision* untuk kategori "kelor" dicapai sebesar 0.91, dengan *recall* 1.00 dan *f1-score* 0.95, menunjukkan bahwa model sangat baik dalam mengidentifikasi gambar "kelor" tanpa kesalahan. Kategori "maleo" memiliki *precision* sempurna 1.00, tetapi *recall* lebih rendah sebesar 0.87, menghasilkan *f1-score* 0.93, yang menunjukkan bahwa beberapa gambar "maleo" tidak terdeteksi dengan baik.

Kesalahan prediksi ini dapat mempengaruhi kinerja model secara keseluruhan diantara lain untuk nilai *precision* dan *recall* pada kategori "maleo" akan lebih rendah dibandingkan dengan kategori lainnya. *Precision* yang lebih rendah menunjukkan bahwa dari semua prediksi "maleo" yang dilakukan oleh model, beberapa di antaranya salah, sedangkan *recall* yang lebih rendah menunjukkan bahwa model gagal mengidentifikasi semua gambar "maleo" yang ada dalam dataset. Hal ini dapat dilihat pada Tabel IV, di mana *precision* untuk kategori "maleo" adalah 1.00, tetapi *recall* lebih rendah sebesar 0.87, menghasilkan *f1-score* 0.93.

Ketidakseimbangan ini juga mempengaruhi *f1-score*, yang merupakan rata-rata harmonis atau nilai keseimbangan dari *precision* dan *recall*, menunjukkan bahwa model kurang mampu mengenali kategori "maleo" dibandingkan dengan kategori lainnya. Akurasi total model adalah 96%, yang berarti model berhasil mengklasifikasikan 96% dari total 408 gambar dengan benar. Alasan lain yang dapat mengakibatkan kesalahan klasifikasi pada maleo mungkin disebabkan oleh kemiripan visual antara kategori "maleo" dan "kelor" atau kurangnya representasi variasi pada data "maleo" dalam dataset. Kemudian, pada kategori "mawar" memiliki *precision*, *recall*, dan *f1-score* sempurna masing-masing 1.00, menunjukkan kinerja yang sempurna dalam mengklasifikasikan citra batik. Akurasi total model adalah 96%, yang berarti model berhasil mengklasifikasikan 96% dari total 408 gambar dengan benar. *Clasification Report* hasil pengujian model dapat dilihat pada Tabel IV.

TABEL IV
CLASIFICATION REPORT

	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Kelor	0.91	1.00	0.95
Maleo	1.00	0.87	0.93
Mawar	1.00	1.00	1.00
<i>Accuracy</i>			0.96

Hasil yang didapatkan dari penelitian ini menunjukkan bahwa penelitian ini memiliki keunggulan-keunggulan yang tidak dimiliki dari penelitian sebelumnya. Dibandingkan dengan penelitian menggunakan metode K-Nearest Neighbor (KNN) oleh [12] yang mencapai akurasi 95%, metode dalam penelitian ini menggunakan dapat menunjukkan akurasi yang lebih tinggi dengan pendekatan deep learning. Penelitian dengan KNN menggunakan ekstraksi fitur RGB, HSV, GLCM, dan LBP, dan pembagian data sebesar 80% latih dan 20% uji. Kelebihan dari penelitian ini adalah penggunaan deep learning yang lebih mampu menangkap fitur kompleks dari citra batik dibandingkan ekstraksi fitur manual yang digunakan dalam KNN. Penelitian lain yang disusun oleh [1] menggunakan Support Vector Machine (SVM) dengan ekstraksi fitur GLCM dan DWT, mencapai akurasi 96%. SVM dikenal efektif

dalam mengekstraksi fitur tekstur, tetapi metode deep learning yang di gunakan pada menunjukkan kemampuan generalisasi yang lebih baik tanpa memerlukan ekstraksi fitur manual, memungkinkan model belajar langsung dari data mentah. Pada penelitian yang memanfaatkan machine learning *modern (deep learning)* lainnya, [11] menggunakan arsitektur *Efficient Net-B2* dengan *transfer learning* dan *augmentasi data*, mencapai akurasi 86.3%. Sedangkan penelitian oleh [13] menggunakan VGG16, mencapai akurasi 95.83% tanpa augmentasi dan meningkat menjadi 98.96% dengan augmentasi. Dibandingkan dengan penelitian ini, memiliki keuntungan dengan mengimplementasikan early stopping dan optimizer Adamax yang membantu menghindari overfitting dan menjaga kestabilan pelatihan, yang tidak selalu diterapkan dalam penelitian lain. Pada penelitian yang disusun oleh [14] yang menggunakan VGG16 modifikasi dengan dua metode berbeda menunjukkan akurasi 96.30% yang mana pada metode pertama.

Perbandingan yang dapat diberikan dari penelitian ini adalah hasil yang kompetitif dengan pendekatan yang berbeda, menunjukkan bahwa early stopping dan optimisasi Adamax dapat menjaga kinerja yang baik dan stabilitas pelatihan. Maka dari itu keuntungan yang dapat diberikan dari penelitian ini adalah, penggunaan metode earllystopping pada penenelitian ini dapat membantu untuk mencegah overfitting dan menghentikan dapat menghentikan pelatihan saat pelatihan berada pada titik optimal, pemanfaatan optimizer adamax juga menjadi salah satu keunggulan yang dapat diberikan, dengan memberikan stabilitas pada pelatihan, Adamax memastikan bahwa model dapat belajar secara perlahan dan menghindari perubahan drastic pada nilai *loss*. Pemanfaatan machine learning *modern (deep learning)* dapat menghasilkan nilai akurasi sebsar 96%, hal itu menunjukan bahwa pengimplementasian metode deep learning pada peneliltian "*Klasifikasi batik bomba*" sangat efektif, lebih baik dibandingkan beberapa penelitian dengan metode menggunakan metode tradisional. Dengan hasil dari validation accuracy yang mencapai 100% pada beberapa epoch, hal itu menandakan bahwa model dapat secara baik melakukan generalisasi secara menyeluruh. Maka dari itu, pada penelitian ini dapat di tunjukkan bahwa memiliki beberapa keunggulan dalam hal akurasi, stabilitas, dan efisiensi pelatihan dibandingkan dengan metode machine learning tradisional dan beberapa metode deep learning lainnya.

IV. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa dari 408 data citra batik dari 3 jenis gambar ("kelor", "maleo", dan "mawar") yang telah dikumpulkan, dengan pembagian data latih dan data pengujian sebesar 80:20, arsitektur model yang menggunakan 25 *epoch*, *early stopping*, dan optimizer Adamax dengan nilai learning rate sebesar 0.001, berhasil diimplementasikan untuk membangun model yang mampu mencapai nilai akurasi sebesar 100% dan *loss* sebesar 0.0613 pada proses pelatihan, menunjukkan bahwa model yang dihasilkan memiliki kinerja yang baik. Hal ini didukung dengan hasil pengujian model menggunakan *confusion matrix* yang menunjukkan bahwa model memiliki akurasi yang sangat tinggi dalam mengklasifikasikan gambar "kelor" dan "mawar", dengan 167 gambar "kelor" dan 120 gambar "mawar" diklasifikasikan dengan benar tanpa kesalahan. Namun, terdapat beberapa kesalahan dalam klasifikasi gambar "maleo", di mana 16 gambar "maleo" diklasifikasikan sebagai "kelor", sementara 105 gambar "maleo" diklasifikasikan dengan benar. Laporan klasifikasi menunjukkan bahwa model memiliki precision, *recall*, dan *f1-score* yang sangat baik pada kategori "kelor" (precision sebesar 0.91, *recall* 1.00, dan *f1-score* 0.95) dan kategori "mawar" (precision, *recall*, dan *f1-score* masing-masing sebesar 1.00). Pada kategori "maleo", precision mencapai 1.00, namun *recall* lebih rendah sebesar 0.87, menghasilkan *f1-score* 0.93. Akurasi total model adalah 96%, yang berarti model berhasil mengklasifikasikan 96% dari total 408 gambar dengan benar, menunjukkan bahwa model telah mampu melakukan klasifikasi citra batik dengan sangat baik.

DAFTAR PUSTAKA

- [1] W. T. Puspitasari, D. C. R. Novitasari, and W. D. Utami, "Batik Classification using Texture Analysis and Multiclass Support Vector Machine," Scitepress, Jan. 2020, pp. 65–71. doi: 10.5220/00085173006500071.
- [2] D. Warli and S. Musa, "EKSPLOKASI ETNOMATEMATIKA DAN ETNOSAINS (ETNOMATHSAINS) PADA BATIK BOMBA," 2022.
- [3] N. Nurtiwi, R. Ruliana, and Z. Rais, "Convolutional Neural Network (CNN) Method for Classification of Images by Age," *JINAV J. Inf. Vis.*, vol. 3, no. 2, pp. 126–130, 2022, doi: 10.35877/454ri.jinav1481.
- [4] M. Sobron and Lubis, "Implementasi Artificial Intelligence Pada System Manufaktur Terpadu," *Semin. Nas. Tek. UISU*, vol. 4, no. 1, pp. 1–7, 2021, [Online]. Available: <https://jurnal.uisu.ac.id/index.php/semnastek/article/view/4134>
- [5] A. Nata and S. Suparmadi, "Analisis Sistem Pendukung Keputusan Dengan Model Klasifikasi Berbasis Machine Learning Dalam Penentuan Penerima Program Indonesia Pintar," *J. Sci. Soc. Res.*, vol. 5, no. 3, p. 697, 2022, doi: 10.54314/jssr.v5i3.1041.
- [6] A. Santoso and G. Ariyanto, "Implementasi Deep Learning berbasis Keras untuk Pengenalan Wajah," *Emit. J. Tek. Elektro*, vol. 18, no. 1, pp. 15–21, 2018, doi: 10.23917/emitor.v18i01.6235.
- [7] A. E. Wijaya, W. Swastika, and O. H. Kelana, "Implementasi Transfer Learning Pada Convolutional Neural Network Untuk Diagnosis Covid-19 Dan Pneumonia Pada Citra X-Ray," *Sainsbertek J. Ilm. Sains Teknol.*, vol. 2, no. 1, pp. 10–15, 2021, doi: 10.33479/sb.v2i1.125.
- [8] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. E. Mohamed, and H. Arshad, "State-of-the-art in artificial neural network applications: A survey," *Heliyon*, vol. 4, no. 11, p. e00938, 2018, doi: 10.1016/j.heliyon.2018.e00938.

- [9] J. L. Garcia Rosa, "Biologically Plausible Artificial Neural Networks," *Artif. Neural Networks - Archit. Appl.*, 2013, doi: 10.5772/54177.
- [10] A. Ibnul Rasidi, Y. A. H. Pasaribu, A. Ziqri, and F. D. Adhinata, "Klasifikasi Sampah Organik dan Non-Organik Menggunakan Convolutional Neural Network," *J. Tek. Inform. dan Sist. Inf.*, vol. 8, no. 1, pp. 72–81, 2022, doi: 10.28932/jutisi.v8i1.4314.
- [11] S. Aras and A. Setyanto, "Deep Learning Untuk Klasifikasi Motif Batik Papua Menggunakan EfficientNet dan Transfer Learning," vol. 8, no. 1, 2022.
- [12] A. P. B. Salsabila, C. Rozikin, and R. I. Adam, "Klasifikasi Motif Batik Karawang Berbasis Citra Digital dengan Principal Component Analysis dan K-Nearest Neighbor," *J. Sist. dan Teknol. Inf.*, vol. 11, no. 1, p. 20, Jan. 2023, doi: 10.26418/justin.v11i1.46936.
- [13] C. U. Khasanah, E. Utami, and S. Raharjo, "Implementation of Data Augmentation Using Convolutional Neural Network for Batik Classification," in *2020 8th International Conference on Cyber and IT Service Management, CITSM 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020. doi: 10.1109/CITSM50537.2020.9268890.
- [14] M. T. D. Putra* and G. P. Kusuma*, "Batik Classification using Deep Learning," *Int. J. Recent Technol. Eng.*, vol. 8, no. 4, pp. 11416–11421, Nov. 2019, doi: 10.35940/ijrte.D9039.118419.
- [15] W. Bismi and M. Qomaruddin, "Klasifikasi Citra Genus panthera Menggunakan Pendekatan Deep learning Berbasis Convolutional Neural network (CNN)," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 5, no. 2, p. 172, 2023, doi: 10.36499/jinrpl.v5i2.8931.
- [16] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.
- [17] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *32nd Int. Conf. Mach. Learn. ICML 2015*, vol. 1, pp. 448–456, 2015.
- [18] C. Cortes, G. Research, and N. York, "L2 Regularization for Learning Kernels," *Proc. Twenty-Fifth Conf. Uncertain. Artif. Intell.*, pp. 109–116, 2004.
- [19] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," pp. 1–18, 2012, [Online]. Available: <http://arxiv.org/abs/1207.0580>
- [20] A. F. Agarap, "Deep Learning using Rectified Linear Units (ReLU)," no. 1, pp. 2–8, 2018, [Online]. Available: <http://arxiv.org/abs/1803.08375>
- [21] N. Fadlia and R. Kosasih, "Klasifikasi Jenis Kendaraan Menggunakan Metode Convolutional Neural Network (Cnn)," *J. Ilm. Teknol. dan Rekayasa*, vol. 24, no. 3, pp. 207–215, 2019, doi: 10.35760/tr.2019.v24i3.2397.
- [22] J. Deng *et al.*, "RALR: Random amplify learning rates for training neural networks," *Appl. Sci.*, vol. 12, no. 1, 2022, doi: 10.3390/app12010268.
- [23] D. P. Kingma and J. L. Ba, "ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION," pp. 1–15, 2015.
- [24] Y. Bai, E. Yang, and B. Han, "Understanding and Improving Early Stopping for Learning with Noisy Labels," no. NeurIPS, pp. 1–15, 2021.
- [25] R. Magdalena, S. Saidah, N. K. C. Pratiwi, and A. T. Putra, "Klasifikasi Tutupan Lahan Melalui Citra Satelit SPOT-6 dengan Metode Convolutional Neural Network (CNN)," *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 3, p. 335, 2021, doi: 10.26418/jp.v7i3.48195.
- [26] N. A. Sundari, R. Magladena, and S. Saidah, "Klasifikasi Jenis Kulit Wajah Menggunakan Metode Covolutional Neural Network (CNN) Efficientnet-B0 Skin Classification System Using Convolutional Neural Network (CNN) EfficientNet-B0," 2022.