

Multi-architectural Transfer Learning CNN for Klowong Batik Fabric Defect Classification

Dhika Wahyu Pratama¹, Andi Sudiarso^{*2}, Denny Sukma Eka Atmaja³, Muhammad Kusumawan Herliansyah⁴

^{1,2,3,4}Department Mechanical and Industrial Engineering, Universitas Gadjah Mada, Indonesia

³School of Industrial and Systems Engineering, Telkom University, Indonesia

Email: ²a.sudiarso@ugm.ac.id

Received : Jun 2, 2025; Revised : Jun 26, 2025; Accepted : Jul 20, 2025; Published : Aug 19, 2025

Abstract

Klowong is a base cloth that has been given a hot wax pattern as the initial stage in the batik making process but has not yet become a finished batik. Nowadays, written batik machine are available but still limited and production defects still occur, reducing the value of batik. Manual QC makes subjective assessments, so an accurate and efficient automated inspection system is needed for SMEs. This study proposes a defect classification approach on batik klowong fabric based on transfer learning using deep convolutional neural networks (CNN) architecture that has been verified to be reliable in image classification schemes. The basic models used include VGG16, ResNet50V2, InceptionV3, and MobileNetV2, with modifications to the fully connected layers to reduce parameter complexity. The dataset consists of 1000 klowong fabric images with a resolution of 224×224 pixels, with a ratio of 80:10:10 for training, validation, and testing. Data augmentation was applied to improve the generalization of the model. Evaluation is performed based on accuracy, precision, recall, F1-score, and inference time. The experimental results show that VGG16 has the best performance in the testing stage with 92% accuracy. The combination of VGG16 with conventional classifiers (SVM and Random Forest) significantly speeds up the inference time (up to 0.0001 seconds per image) but with a decrease in accuracy to 81-83%. Therefore, the VGG16 model with the modified final layer is recommended as the optimal solution with the best trade-off between classification performance and computational efficiency, especially for application scenarios on low-resource devices such as batik SMEs.

Keywords : Batik, Classification, CNN, Klowong, VGG16.

This work is an open access article and licensed under a Creative Commons Attribution-Non Commercial 4.0 International License



1. PENDAHULUAN

Batik merupakan salah satu warisan budaya Indonesia yang diakui oleh UNESCO sebagai warisan budaya tak benda pada tahun 2009 [1], [2]. Batik mengandung nilai estetika dan historis bagi bangsa Indonesia. Saat ini terdapat beberapa teknologi yang dapat mendukung pelestarian dan membantu produksi pada pembatikan. Salah satu tantangan dalam pembatikan adalah mayoritas dari pembatikan masih menggunakan metode manual [3], [4]. Untuk membuat sebuah batik tulis, beberapa proses yang harus dilakukan dan harus ada dalam pembuatan batik tulis diantaranya adalah persiapan kain, pembuatan pola dengan pensil atau nyoret, pencantingan pola dasar atau nglowong, penutupan pola yang tidak ingin diwarnai atau nembok, pewarnaan atau nyelup, pengisian detail kecil pada batik atau isen-isen, peluruhan lilin atau nglorod, dan pengeringan. Salah satu proses manual yang memakan banyak waktu dalam proses pembatikan adalah proses nglowong. Proses nglowong dilakukan dengan menorehkan lilin batik panas kepada kain yang telah dipola dengan pensil untuk menghasilkan pola dasar pada kain batik, dimana keahlian dari pembatik merupakan kunci utama dalam menghasilkan kain klowong yang baik.

Saat ini terdapat mesin yang dapat mempercepat proses nglowong, namun masih terbatas ke beberapa UMKM saja. Dalam praktiknya, mesin ini menggantikan proses nglowong yang memakan sangat lama sehingga jumlah produksi batik dapat dipercepat. Mesin batik tulis dilengkapi dengan canting yang dipanaskan sehingga dalam prosesnya tidak menghilangkan esensi dari batik tulis itu sendiri yaitu menorehkan lilin panas. Desain CAD atau Computer Aided Design diperlukan sebagai gambaran mesin dalam membuat pola sesuai keinginan dari pengguna. Nantinya pengguna perlu melakukan penyesuaian parameter setiap melakukan produksi kain klowong [5]. Walaupun menggunakan mesin, namun kesalahan dari proses produksi masih terkadang terjadi. Ketidaksempurnaan atau kecacatan dari proses nglowong ini dapat menurunkan nilai artistik dan ekonomis dari batik. Proses quality control (QC) dari hasil pembatikan juga masih manual sehingga maningkatkan subjektifitas dari penilaian hasil produksi kain batik jadi maupun kain klowong [6]. Oleh karena itu, diperlukan sebuah sistem yang dapat menilai atau melakukan inspeksi kualitas hasil kain klowong secara lebih objektif, akurat dan efisien, khususnya untuk kemampuan komputasi komputer pada level UMKM yang termasuk low atau pada level komputasi rendah.

Pemanfaatan teknologi pengolahan citra dapat membantu dalam objektifitas penilaian hasil pada kain klowong. Deep Learning (DL) merupakan cabang dari kecerdasan buatan yang memanfaatkan jaringan saraf tiruan banyak lapisan yang tersembunyi sehingga secara otomatis mampu mempelajari fitur [7], [8]. Metode Convolutional Neural Networks (CNN) sebagai arsitektur yang sangat populer dalam deep learning baik untuk melakukan klasifikasi maupun deteksi dapat diimplementasikan pada permasalahan ini karena terbukti baik dalam beberapa kasus. Secara spesifik pemanfaatannya ada pada kasus seperti lane detection [9], kecacatan pada pola kain [10], pengenalan pola kain tertentu [11], memprediksi penggunaan energi sebuah gedung [12], pengenalan wajah [13] serta pada kecacatan kain batik [2] serta pengenalan pola pada kain batik [14], [15].

Penerapan pengolahan citra memungkinkan sistem mampu mengidentifikasi cacat produksi atau pola pola cacat yang muncul pada kain klowong. Dengan hasil yang lebih objektif dan lebih konsisten, maka pemanfaatannya sangat menarik jika diimplementasikan dalam permasalahan produksi kain klowong. Penggunaan CNN dalam melakukan evaluasi kualitas kain klowong berdampingan membantu pekerjaan dari operator mesin dalam mengidentifikasi adanya kesalahan produksi secara lebih dini. Akibatnya, kesalahan yang teridentifikasi lebih awal akan mengurangi kerugian dari segi material dan juga waktu pembuatan. Deep Learning memiliki keunggulan jika dibandingkan dengan metode konvensional yaitu kemampuannya dalam melakukan ekstraksi fitur secara otomatis tanpa memerlukan preprocessing yang kompleks. CNN dapat mengenali pola pola visual dan juga tekstur serta motif yang kompleks dengan baik dan telah dibuktikan dengan beberapa penelitian terkait pengenalan pola [16].

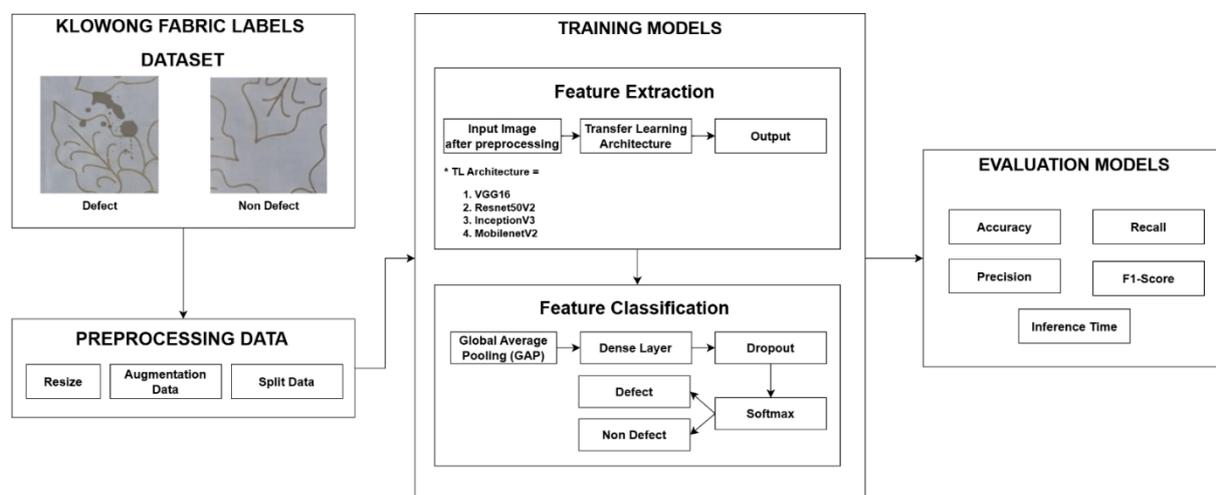
Beberapa penelitian yang dekat kepada kasus batik yang sudah memanfaatkan metode CNN adalah penelitian tentang klasifikasi pattern batik seperti : [6] yang melakukan klasifikasi 5 motif batik. Penelitiannya menggunakan metode Convolutional Neural Networks (CNN) yang menghasilkan 96% accuracy, 96.78% precision, 96.74% recall, dan 96.74%. Studi lainnya dilakukan [17] dengan mengklasifikasikan batik berdasarkan teknik pembuatannya yaitu Batik tulis dan Batik Cap. Penelitian ini menggunakan beberapa metode CNN seperti ResNet, DenseNet dan VGG dengan hasil terbaik pada DenseNet169 dengan akurasi testing sebesar 79.17% pada dataset awal namun VGG13 menunjukkan hasil terbaik pada dataset modifikasi sebesar 87.61%, namun hasil ini masih dapat dieksplorasi lebih jauh. [18] melakukan klasifikasi 3 jenis motif batik dengan menggunakan metode CNN dan menghasilkan rata-rata akurasi sebesar 65% dan metode CNN dikombinasikan dengan Grayscale menghasilkan rata-rata di 70% akurasi. [19] melakukan eksplorasi model CNN pada pengenalan pola untuk 7 jenis kain batik yang menghasilkan nilai akurasi sebesar 95% pada testingnya. Terdapat penelitian lain yang membahas defect namun pada permukaan logam yang mengimplementasikan metode CNN. Beberapa metode kemudian dibandingkan dengan hasil dari metode MobilenetV2 dan InceptionV3 yang

menghasilkan hasil terbaik pada CNN yang dibuat dengan hasil 100% testing pada model K3, 90,3% untuk MobilenetV2, dan 94,4% untuk InceptionV3 [20].

Pada studi ini, masalah akan difokuskan kepada klasifikasi secara spesifik pada kecacatan dari kain klowong. Arsitektur CNN yang sudah banyak digunakan seperti Resnet, VGG, Inception dan Mobilenet nantinya akan digunakan pada penelitian ini karena pada beberapa penelitian sebelumnya memiliki potensi dan menunjukkan hasil yang baik pada pengenalan pola kain. Pada penelitian ini metode klasifikasi Hybrid juga akan diuji dengan tujuan untuk mencari nilai akurasi, presisi, recall, dan nilai f1 dan kecepatan komputasi yang lebih cepat untuk mengevaluasi model. Penelitian ini bertujuan untuk menemukan model yang ringan dan cepat namun memiliki akurasi yang baik untuk dapat membantu UMKM batik dalam implementasinya agar tidak memerlukan daya komputasi yang tinggi.

2. METODE

Pada penelitian ini, metode transfer learning arsitektur CNN akan digunakan untuk mengenali pola defect pada proses klasifikasi defect pada kain klowong batik. Nantinya, beberapa metode yang telah dilatih sebelumnya atau transfer learning akan digunakan sebagai pembandingan dari metode yang dieksplorasi. Metode seperti VGG16, Resnet50V2, InceptionV3, dan MobilenetV2 akan digunakan sebagai pembandingan karena telah terbukti keandalannya dalam beberapa kasus klasifikasi citra [17], [20], [21], [22], [23], [24]. Cara dan tahapan yang dilakukan pada penelitian ini dapat dilihat pada Gambar 1 yang meliputi proses ekstraksi fitur dan tahapan klasifikasi akan diuji. Nantinya model yang memiliki hasil terbaik akan coba di kombinasikan dengan metode klasifikasi konvensional seperti SVM dan Random Forest untuk melihat pengaruh keandalan dan kecepatan prediksi model.

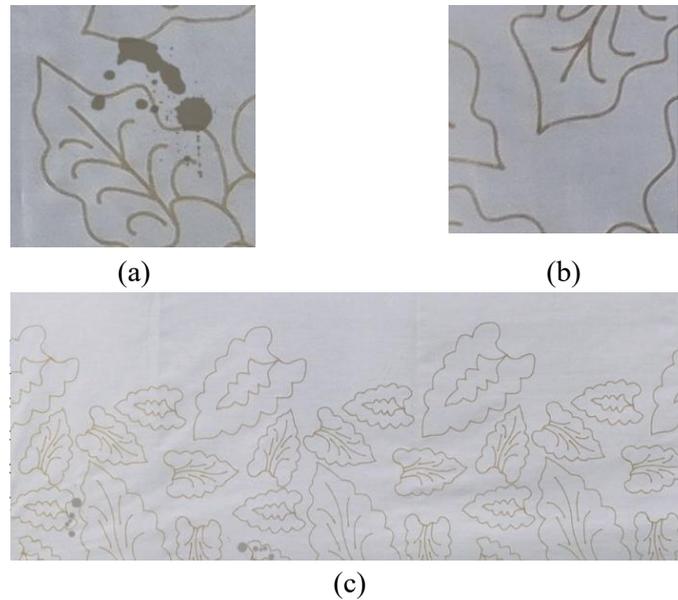


Gambar 1. Diagram Alir Penelitian

2.1. Akuisisi dan Augmentasi Data

Dataset yang digunakan pada penelitian ini adalah dataset kain klowong hasil produksi yang diambil menggunakan kamera yang hasilnya terdapat pada Gambar 1, dan dilakukan proses cropping dengan ukuran 224x224, hal ini dikarenakan metode CNN seperti VGG, Resnet, Inception, dan Mobilenet cukup kompatibel dengan ukuran tersebut. Dataset ini berjumlah 1000 dataset yang terdiri dari 500 potongan gambar kain yang tergolong sebagai defect dan 500 untuk kain klowong normal atau non defect. Gambar 2 menampilkan gambar potongan kain klowong defect dan juga gambar potongan yang normal. Dapat dilihat bahwa kain yang mengalami defect memiliki noda berupa lilin batik yang terlalu cair sehingga tumpah ke dalam kain. Sedangkan pada untuk gambar yang normal tidak terdapat noda atau spot pada kain klowong hasil produksi. Dataset ini kemudian dibagi lagi untuk training dan

testing dengan variasi proporsi train dan testing yaitu 80% untuk training, 10% validasi, dan 10% untuk testing.

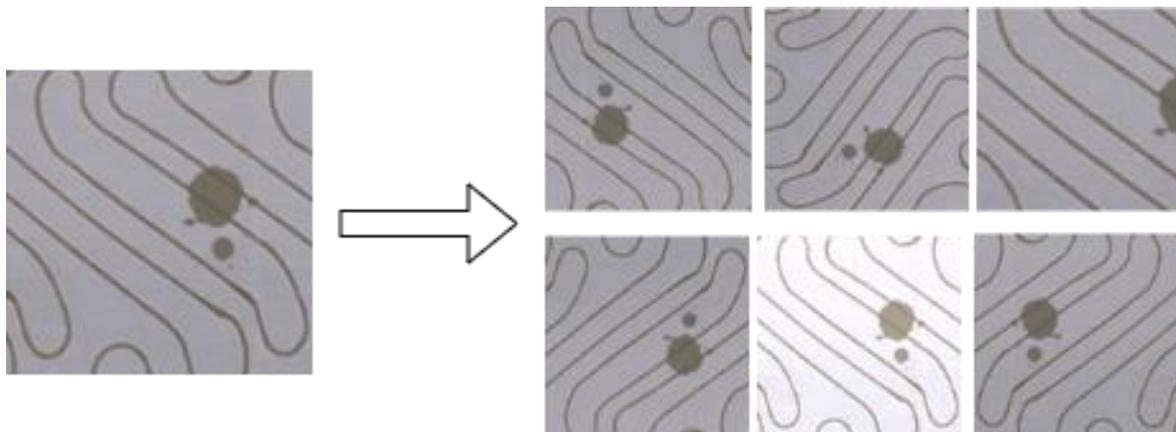


Gambar 2. Contoh Kain Klowong. (a) Defect, (b) Non-Defect, (c) Keseluruhan Kain

Pada proses training, akan dilakukan augmentasi data dengan tujuan memperbanyak data karena deep learning membutuhkan banyak dataset [25]. Proses augmentasi data dilalui dengan beberapa variasi seperti rescale, flipping, dan lainnya. Nantinya setelah melalui proses augmentasi, 1 gambar akan menghasilkan beberapa gambar sesuai dengan proses augmentasi yang dilakukan sehingga menambah jumlah data pada fase training. Tabel 1 menunjukkan beberapa parameter yang digunakan dalam melakukan augmentasi data

Tabel 1. Parameter Augmentasi

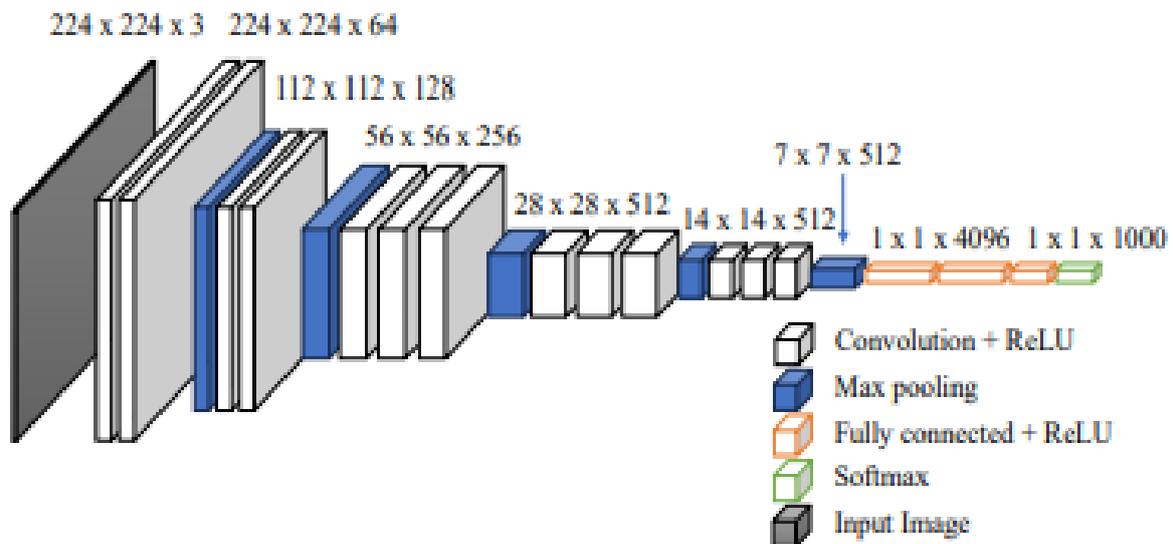
Perubahan	Nilai
Horizontal Flip	True
Vertical Flip	True
Rotation	90,180
Zoom	1.1-2.0
Contrast	0.5-1.5



Gambar 3. Contoh gambar yang melalui proses augmentasi

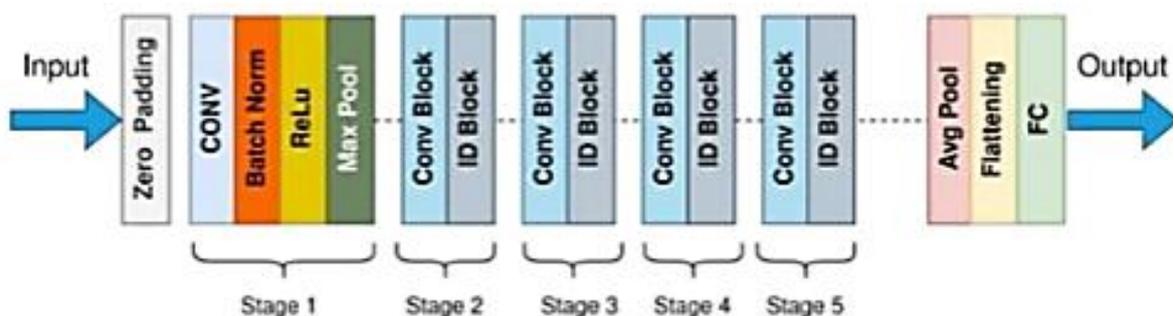
2.2. Model Arsitektur

Dataset yang sudah melalui proses augmentasi saat ini siap untuk digunakan terutama dalam melakukan training dengan model CNN. Visual Geometry Group (VGG) merupakan kelompok yang menemukan arsitektur CNN dengan nama populer VGGNet. VGG 16 merupakan salah satu arsitektur yang terkenal dengan 13 convolutional layers dan 3 fully connected layers [26]. Bentuk arsitekturnya dapat dilihat pada Gambar 4. Model ini merupakan salah satu arsitektur yang menghasilkan persentase 7.3% pada kesalahan yang termasuk kesalahan kecil [27]. Arsitektur VGG16 yang sebelumnya dilatih pada dataset imagenet dengan input $224 \times 224 \times 3$ untuk output 1000 kelas [28], namun nantinya akan coba diimplementasikan pada permasalahan ini dengan output 2 kelas. Nantinya bagian akhir dari VGG16 akan disesuaikan dengan kebutuhan dan dimodifikasi untuk mengurangi jumlah parameter.



Gambar 4. Model Arsitektur VGG16 [29]

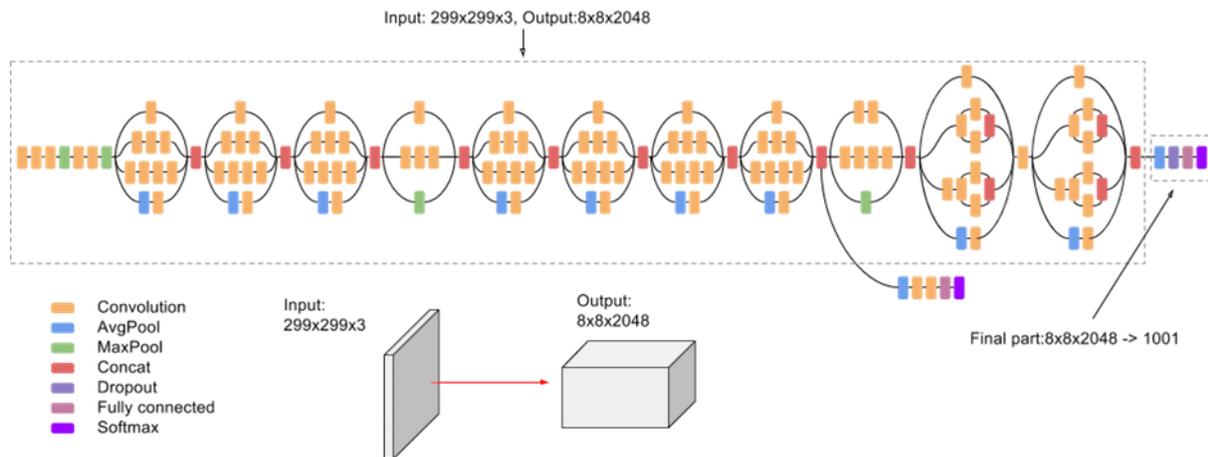
Arsitektur Resnet memiliki ciri berupa fungsi residual sehingga mudah untuk dioptimalkan. Hal ini dikarenakan adanya proses shortcut atau skip dari koneksi. Tujuan dari dilakukannya proses shortcut ini adalah untuk memastikan semua fitur tetap terdeteksi sehingga jaringan dapat dibangun dengan dalam dan baik [26]. Pada studi ini, ResNet50V2 akan digunakan karena termasuk salah satu versi updated dan cukup ringan. Gambar 5 memuat informasi bagaimana arsitektur ini dibentuk sebelum akhirnya akan dimodifikasi pada bagian head sampai output.



Gambar 5. Model Arsitektur ResNet50[30]

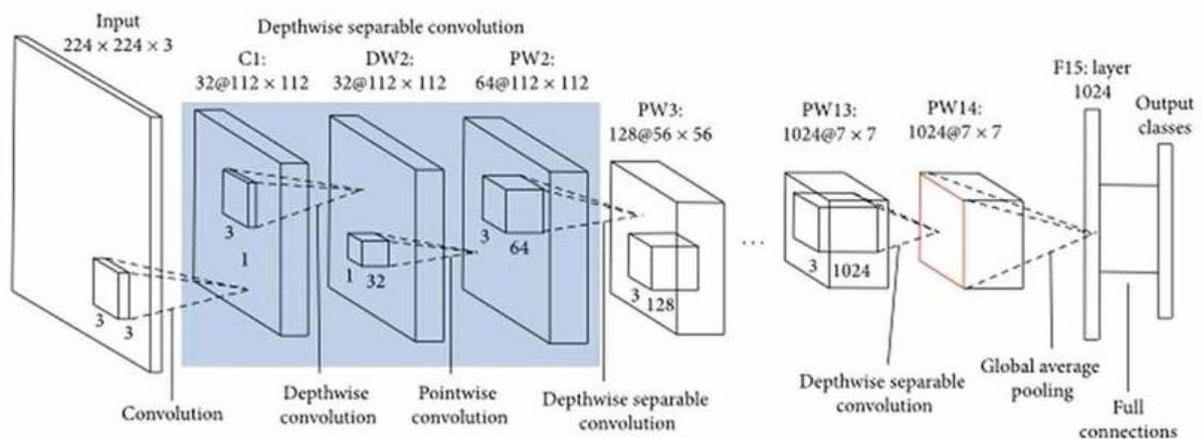
Metode pelatihan selanjutnya adalah menggunakan arsitektur InceptionV3. Arsitektur yang di develop langsung oleh Google pada tahun 2014 ini memenangkan posisi runner-up pada kejuaraan

Imagenet Large Scale Visual Recognition Competition atau ILSVRC dengan imagenet dataset. arsitektur ini memiliki ciri penambahan layer konvolusi 1x1 dengan tujuan mereduksi dimensi sehingga mengurangi computational cost. Selain itu, tujuannya juga mirip dengan ResNet yaitu memperdalam dan memperluas jaringan syaraf. Gambar 6 memuat informasi bagaimana arsitektur ini dibentuk sebelum akhirnya akan dimodifikasi pada bagian head sampai output.



Gambar 6. Model Arsitektur InceptionV3 [30]

Mobilenet merupakan salah satu arsitektur CNN ringan dan banyak digunakan pada beberapa aplikasi mobile karena ringan [31], [32]. arsitektur ini memiliki latensi yang rendah dan juga baik digunakan untuk deteksi objek, klasifikasi, dan beberapa tugas lainnya[33]. Model ini sangat baik dalam mengurangi kompleksitas model dan juga ukuran sehingga berpengaruh baik pada computational cost [33], [34]. Model yang ringan dan cepat juga menjadi salah satu tujuan utama pada studi ini sehingga arsitektur ini juga ikut dipertimbangkan. Gambar 7 Memuat informasi bagaimana arsitektur ini dibentuk sebelum akhirnya akan dimodifikasi pada bagian head sampai output.



Gambar 7. Model Arsitektur MobileNetV2 [35]

Bagian fully connected layers merupakan bagian yang ada setelah ekstraksi fitur dan langsung terkoneksi dengan layer sebelumnya pada model yang dilatih [6]. Arsitektur yang dipilih untuk training merupakan arsitektur yang telah jadi sebelumnya dan memiliki layer fully connected tersendiri. Namun kita dapat menggantinya dengan layer baru yang lebih efisien. Pada penelitian ini, untuk mengurangi jumlah parameter dan mengurangi waktu komputasi, maka layer pada Fully Connected Layers dimodifikasi menjadi seperti pada Tabel 2.

Tabel 2. Fully Connected Layers Parameters.

Layer	Parameter
Global Average Pooling2D	-
Dense	128, activation = 'relu'
Dropout	0.2
Dense Output	2, activation = 'softmax'

Setelah dilakukan penggantian layers pada FCL, maka Tabel 3 menunjukkan penurunan jumlah parameter dari masing model. Penggunaan global average pooling (GAP) dilakukan untuk merangkul secara efisien bagian dari spatial fatures dengan baik. Pemilihan ini juga didasari dengan pertimbangan pengurangan jumlah parameter yang memperkecil kemungkinan overfitting. Layers dense dengan 128 neurons digunakan karena dengannilai yang lebih besar seperti 256 atau 512 menghasilkan nilai akurasi yang sama namun jumlah parameter yang lebih banyak, sedangkan untuk jumlah neuron yang lebih sedikit menghasilkan nilai akurasi yang lebih rendah. Penggunaan 0.2 pada dropout untuk mencegah terjadinya overfitting. Sedangkan penggunaan softmax activation pada akhir layers untuk mengkonversi angka logit ke probabilitas untuk 2 kategori yaitu *defect* dan *non-defect*.

Tabel 3. Komparasi Parameter Model

Model Arsitektur	Jumlah Parameter	
	Sebelum	Sesudah
VGG16	138,357,544	14,780,610
Resnet50V2	25,613,800	23,827,330
InceptionV3	23,851,784	22,065,314
MobilenetV2	3,538,984	2,422,210

2.3. Pelatihan Model

Beberapa model akan dilatih memakai setup menggunakan Google Colab dengan GPU Tesla 4. Pelatihan model dilakukan sesuai dengan proses yang ditunjukkan pada Gambar 1. Nantinya model seperti VGG16, Resnet50V2, InceptionV3, dan MobilenetV2 akan secara bergantian dilatih dengan setting seperti pada Tabel 4. Pelatihan model ini dilakukan untuk melatih model terhadap pengenalan pola defect sehingga mampu memprediksi secara akurat [20]. Data pada pelatihan ini tentunya telah melalui augmentasi data seperti pada Tabel 1, dan beberapa hyperparameter yang digunakan lainnya adalah seperti "Adam" optimizer, batch size dengan jumlah 32, 50 epochs dan loss function berupa categorical crossentropy. Checkpoint digunakan pada proses pelatihan ini dengan evaluasi model berdasarkan akurasi validasi terbaik.

2.4. Pengujian Model dan Metrik Evaluasi

Pengujian model dilakukan dengan melakukan load pada model yang telah dilatih. Nantinya model akan memprediksi sejumlah data yang telah dibagi dan akan dibandingkan nilai akurasinya. Model terbaik nantinya akan dipilih dan akan dilakukan hibridisasi dengan metode konvensional untuk mengeksplor kemampuan model.

Evaluasi model dilakukan dengan menghitung nilai akurasi dari pengujian dan juga melakukan analisis pada confusion matrix. Confusion matrix dapat menunjukkan hasil perbandingan dari label asli dengan label prediksi dan dapat dilihat pada Gambar 4. Nantinya dapat dilihat kecenderungan kesalahan klasifikasi antar kelas. Berikut adalah beberapa komponen yang terdapat dalam confusion matrix seperti:

- True Positive (TP): Data positif yang diprediksi sebagai benar.

- True Negative (TN): Data negatif yang diprediksi sebagai benar.
- False Positive (FP): Data negatif yang diprediksi sebagai positif.
- False Negative (FN): Data positif yang diprediksi sebagai negatif.

Berikut adalah beberapa metrik perhitungan yang digunakan seperti akurasi, presisi, recall, dan f1-score.

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

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

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

$$F1 - Score = 2 \frac{(precision \times recall)}{(precision + recall)} \tag{4}$$

Actual Class	C1	TP_1	FP_1/FN_2
	C2	FP_2/FN_1	TP_2
		C1	C2

Prediction Class

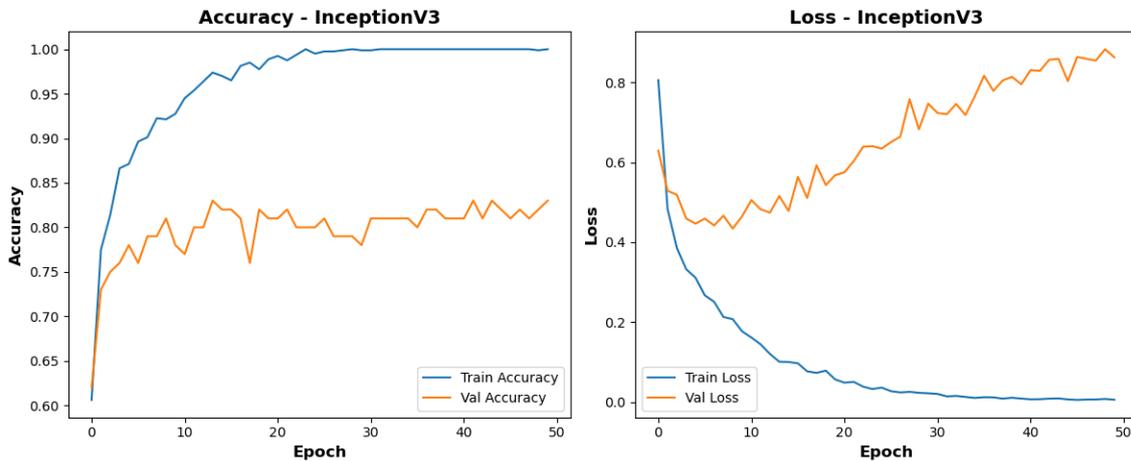
Gambar 8. Confusion Matrix dengan C1 dan C2 masing-masing mewakili Cacat dan Tidak Cacat.

3. HASIL

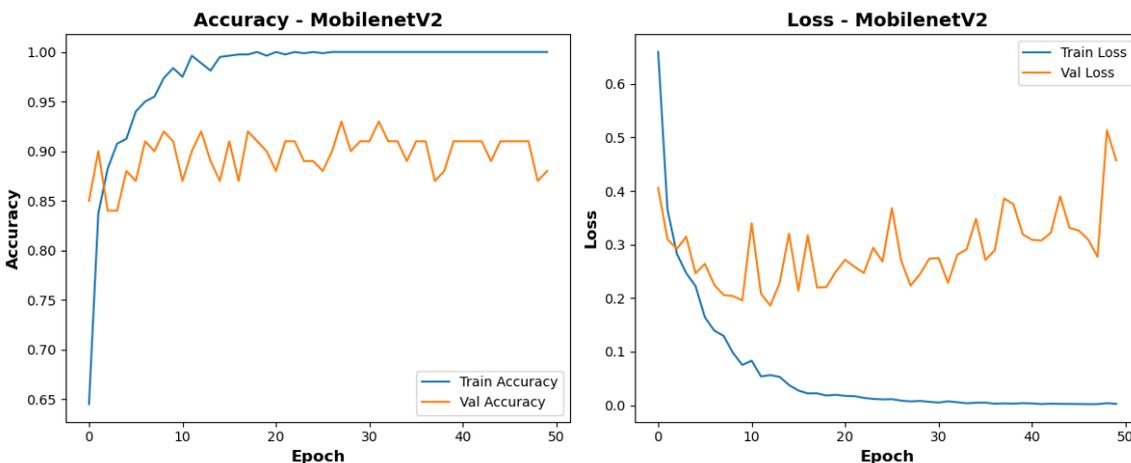
Training dan testing model dilakukan menggunakan Google Colab Pro dengan menggunakan Tesla 4 GPU serta menggunakan setting High Ram dengan library TensorFlow untuk modelnya. Penggunaan ini didasari dengan kemampuan komputasi yang lebih baik pada penggunaan GPU dibandingkan dengan CPU untuk melakukan pelatihan model gambar. Dalam pelatihannya dilakukan secara bertahap menggunakan parameter yang telah disesuaikan. Untuk melakukan pantauan secara lebih baik, digunakan grafik visualisasi metrik seperti grafik akurasi dan loss. Performa pelatihan dari model yang dilatih menggunakan dataset kain klowong dapat dilihat pada gambar 9 sampai 12.

Hasil training tiap iterasinya divisualisasikan melalui grafik train dan validation untuk masing model. Penggunaan grafik ini sangat penting untuk melakukan evaluasi secara visual terkait apakah model belajar secara optimal atau justru model mengalami overfitting atau underfitting. Overfitting sendiri adalah kecenderungan model untuk terlalu terpaku kepada data belajar atau training yang memungkinkan untuk mengingat seluruh detail pada data belajar, akibatnya model sulit untuk memahami pola dari gambar yang berbeda dari data pelatihan. Underfitting disini lain adalah kondisi dimana model belum cukup belajar untuk mengenali pola dalam data pelatihan yang diberikan, akibatnya model sangat buruk dalam melakukan prediksi data validasi atau data uji, bahkan dalam kasus underfitting model juga buruk dalam memahami data belajar. Grafik menunjukkan bahwa pada metode MobilenetV2 dan InceptionV3 memiliki kecenderungan bertambahnya validation loss semakin bertambahnya iterasi walaupun training lossnya semakin berkurang. Hal ini mengindikasikan secara kuat bahwa model memiliki kecenderungan untuk mengalami overfitting yang diakibatkan oleh model tidak dapat mengeneralisasi data validasi dengan baik. Tabel 4 menampilkan hasil checkpoint training

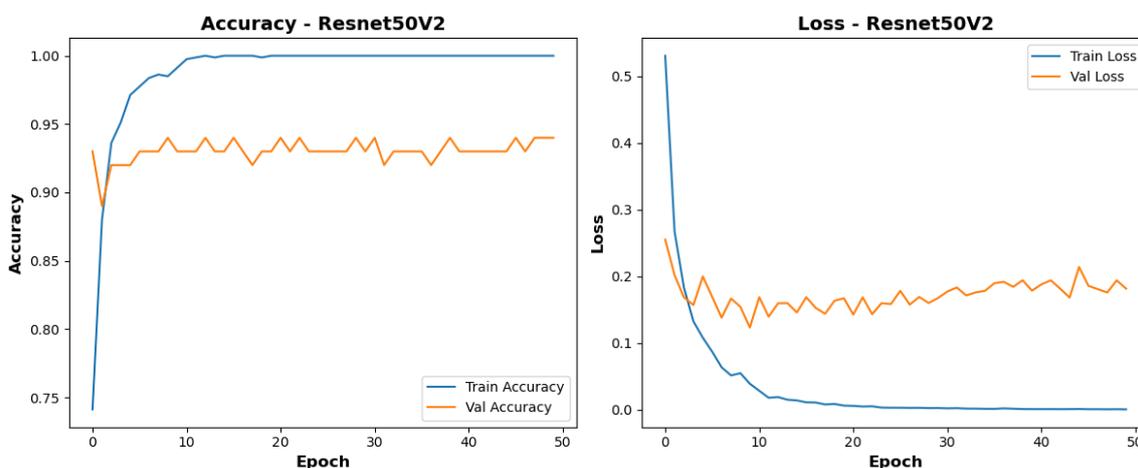
pada validasi test terbaik tiap model untuk disimpan dalam model “h5”. Hal ini dilakukan untuk mengantisipasi model yang kembali buruk saat training terlalu berlebihan sehingga performa terbaik dari model dapat tersimpan dan di load untuk melakukan prediksi pada data testing. Meskipun early stopping merupakan pendekatan yang banyak dilakukan karena baik dalam mencegah overfitting, namun pendekatan ini tidak dapat memastikan titik berhenti merupakan hasil terbaik [36].



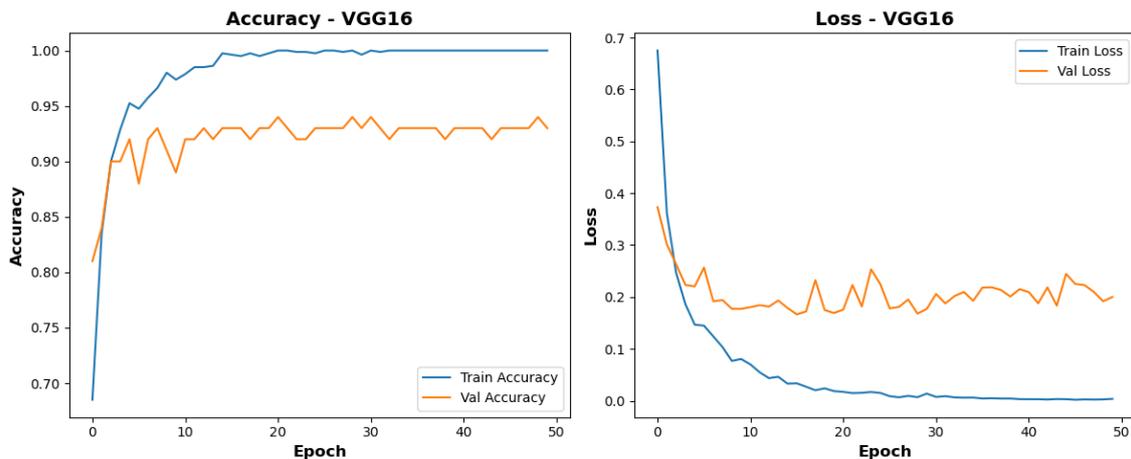
Gambar 9. Grafik Akurasi dan Loss Pelatihan InceptionV3



Gambar 10. Grafik Akurasi dan Loss Pelatihan MobilenetV2



Gambar 11. Grafik Akurasi dan Loss Pelatihan Resnet50V2



Gambar 12. Grafik Akurasi dan Loss Pelatihan VGG16

Tabel 4. Checkpoint Pelatihan Model

Parameter	Nilai
monitor	“val_accuracy”
save_best_only	True
mode	Max
Verbose	1

Tabel 5. Perbandingan Hasil Pelatihan Model

Parameter	Model Arsitektur			
	VGG16	Resnet50V2	InceptionV3	MobilenetV2
Training Time	3 min 15 sec	4 min 27 sec	3 min 48 sec	3 min 11 sec
Train Accuracy	1.0000	0.98	0.9824	1.0000
Train Loss	0.0168	0.0639	0.0885	0.0168
Val Accuracy	0.94	0.94	0.83	0.93
Val Loss	0.1757	0.1543	0.5158	0.2235

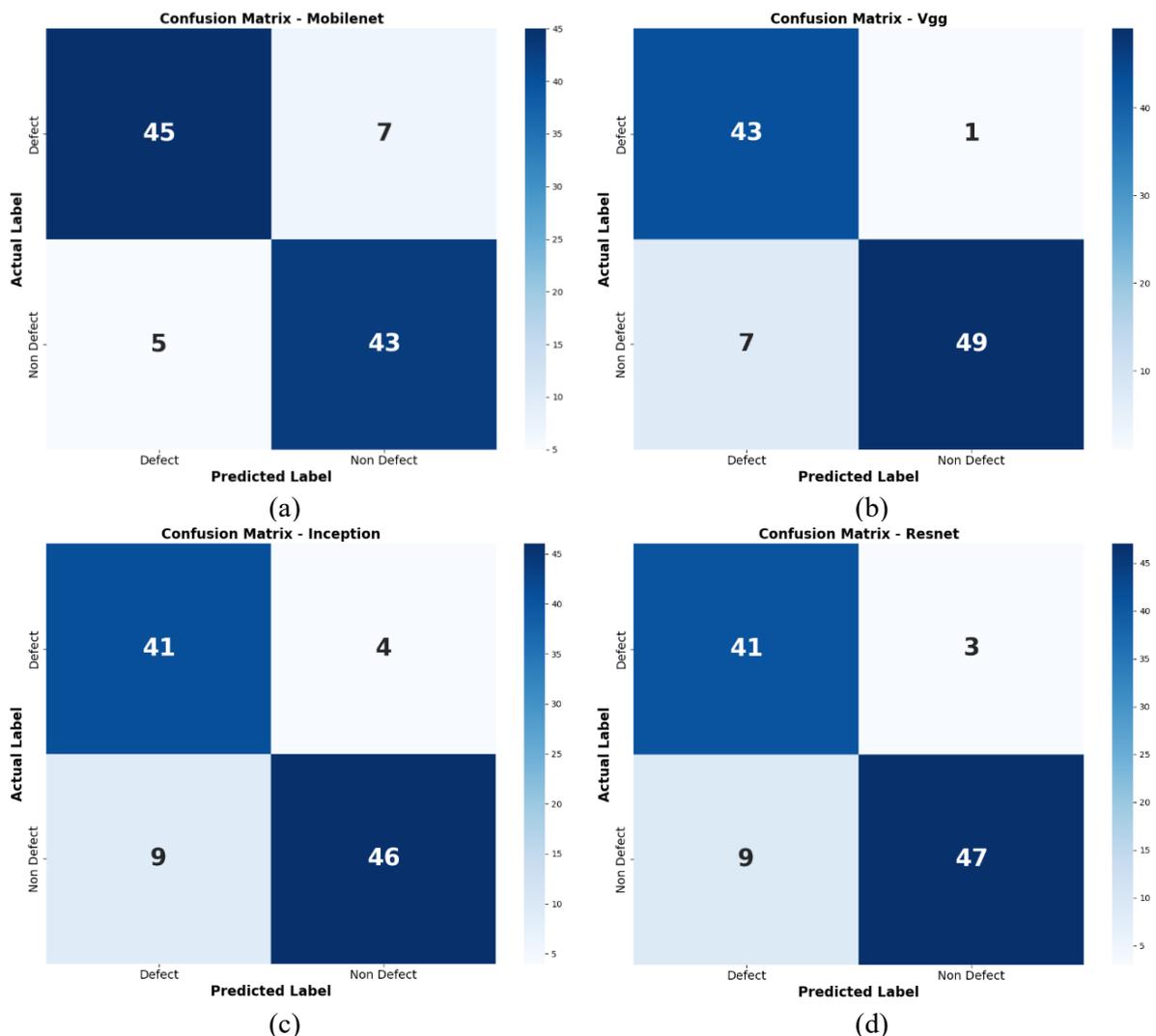
Hasil yang ditunjukkan pada tabel 5 bahwa waktu training tercepat diperoleh model MobilenetV2 dengan waktu 3 menit 11 detik diikuti oleh model VGG16 dengan waktu 3 menit 15 detik, lalu model InceptionV3 dengan waktu 3 menit 48 detik, kemudian waktu terlama diperoleh model Resnet50V2 dengan waktu 4 menit 27 detik. Hal ini menunjukkan bahwa model MobilenetV2 unggul dalam waktu komputasi. Pada hasil akurasi training, VGG16 dan MobilenetV2 memiliki hasil yang sama diangka 1 atau saat dikonversikan kedalam persentase adalah sebesar 100 % akurasi. Hal ini yang mengindikasikan performa terbaik pada saat training oleh kedua model, model InceptionV3 menempati posisi ketiga dengan nilai akurasi 0.9824 atau 98.24% akurasi, sedangkan Resnet50V2 menjadi paling bawah pada akurasi training dengan hasil 0.98 atau saat dikonversikan menjadi 98%. Meskipun demikian, hasil yang diperoleh cukup baik meskipun Resnet50V2 ada pada posisi terbawah, hal ini mengindikasikan model dapat menggeneralisir data dengan baik.

Hasil train loss terkecil sekaligus diperoleh VGG16 dan MobilenetV2 dengan nilai 0.0168 yang menunjukkan bahwa kecilnya tingkat kesalahan pada proses prediksi selama pelatihan. Hasil terbaik masing model berdasarkan hasil uji validasi yang secara tidak langsung sedikit merepresentasikan keadaan data testing jatuh kepada Resnet50V2, dengan nilai val akurasi tertinggi dan val loss terkecil pada proses training yaitu 0.94 dan 0.1543. Namun, berdasarkan keseluruhan hasil pada proses training,

model VGG16 dan MobilenetV2 menjadi model terbaik dengan posisi seri pada komparasi head-to-head berdasarkan nilai parameter.

Berdasarkan hasil yang diperoleh pada Gambar 9 sampai Gambar 12, dapat dilihat bahwa mulai pada epoch 20 tidak terjadi peningkatan yang signifikan pada nilai akurasi training. Namun, penggunaan earlystopping tidak digunakan namun digantikan dengan checkpoint model yang nantinya akan menyimpan weight terbaik dari model berdasarkan nilai akurasi validasinya. Mekanisme early stopping tentunya mampu mengefisienkan waktu training. Hal ini dapat ditinjau pada penelitian kedepannya dengan membandingkan hasil dari earlystopping dengan penggunaan model checkpoint.

Setelah melakukan proses pelatihan, model kemudian diuji menggunakan data uji yang telah dipisahkan yaitu 10% dari dataset atau sebanyak 100 gambar. Hasil visualisasi dari pengujian pada data uji kemudian ditampilkan pada gambar tabel confusion matriks dari hasil pengujian model tersajikan pada Gambar 8 dengan melakukan load best model yang telah disimpan melalui model *checkpoint* untuk melakukan prediksi pada data uji. Data yang ditampilkan oleh tabel confusion matrix nantinya dapat dihitung kembali untuk diketahui metrik evaluasi dari tiap kelas dan secara keseluruhan dari sail uji pada modelnya. Untuk mempermudah pengamatan, hasil uji dari keempat model, maka Tabel 6 menampilkan komparasi metrik evaluasi tiap model.



Gambar 13. Confusion Matrix untuk tiap model, (a) MobilenetV2, (b) VGG16, (c) InceptionV3 dan (d) Resnet50V2

Tabel 6. Testing Parameters Comparison.

Parameters	Architecture			
	VGG16	Resnet50V2	InceptionV3	MobilenetV2
Inference time	2s/step	2s/step	2s/step	699ms/step
Avg Accuracy	0.92	0.88	0.87	0.88
Avg Precision	0.93	0.89	0.87	0.88
Avg Recall	0.92	0.88	0.87	0.88
F1-score	0.92	0.88	0.87	0.88
Pred time	0.0893 s/img	0.0849 s/img	0.0887 s/img	0.0404 s/img

Hasil yang diperoleh pada Gambar 13 menunjukkan bahwa model VGG16 merupakan model terbaik dengan kesalahan yang lebih sedikit dibandingkan dengan model lainnya. Semua model memiliki kecenderungan, namun tetap memiliki beberapa kesalahan. Tabel 6 menunjukkan bahwa VGG16 merupakan model terbaik meskipun waktu prediksi tiap gambarnya menjadi yang terlama. Jika ditinjau kembali, perbedaan kecepatan antara VGG dan beberapa model lainnya tidak signifikan kecuali dengan MobilenetV2. Namun, karena kecepatan prediksi tiap gambar dibawah 0.1 detik, maka model ini dipilih karena untuk implementasi realtime angka tersebut masih termasuk kedalam spesifikasi *realtime*. Untuk meningkatkan waktu prediksi tiap gambarnya, maka model terbaik VGG16 dipilih yang kemudian bagian Fully Connected Layers (FCL) yang sebelumnya merupakan layers dense dan dropout digantikan dengan metode klasifikasi klasik berupa SVM (Support Vector Machine) dan RF (Random Forest) dengan parameter yang dituning.

Keduanya model kemudian ditraining dengan model VGG16 yang menjadi *feature extraction* dan hasilnya direpresentasikan pada Tabel 7. Pada proses training, dilakukan tuning parameter dari SVM dan RF. Untuk SVM, proses pencarian parameter dilakukan dengan teknik Grid Search dengan 5-fold cross validation dengan parameter yang terpilih adalah $C = 10$, $\gamma = 0.01$ dan kernel = "rbf". Sedangkan untuk RF juga dilakukan dengan metode Grid Search dan 5-fold cross validation dengan parameter yang terpilih berupa $n_estimator = 200$, $max_depth = none$ dan $min_sample_split = 5$. Hasil yang diperoleh menunjukkan adanya peningkatan waktu prediksi secara signifikan, dimana waktu prediksi tiap gambar untuk model VGG16-SVM menjadi 0.000171 detik dan model VGG16-RF menjadi 0.000115. Namun, hal ini berdampak pada akurasi dan metrik lainnya dikorbankan. Sehingga model VGG16 dengan FCL awal merupakan hasil terbaik untuk permasalahan ini.

Tabel 7. Perbandingan Model VGG16.

Parameters	Architecture		
	VGG16+FCL	VGG16-SVM	VGG16-RF
Avg Accuracy	0.92	0.83	0.81
Avg Precision	0.93	0.83	0.82
Avg Recall	0.92	0.83	0.81
F1-score	0.92	0.83	0.81
Pred time	0.0893 s/img	0.000171 s/img	0.000115 s/img

4. DISKUSI

Melihat hasil yang diperoleh dari pelatihan awal, terlihat bahwa metode MobilenetV2 memiliki potensi pada segi kecepatan, namun dalam implementasi secara langsung, ketelitian dan akurasi juga diperlukan. Kecepatan MobilenetV2 yang lebih cepat 50% jika dibandingkan dengan VGG16

menandakan bahwa model tersebut ringan dan cepat namun perlu di tuning lebih jauh, baik dari segi parameter dan maupun bagian Fully Connected Layers. Metode VGG16 sendiri akhirnya dipilih untuk kemudian dieksplorasi dengan menggunakannya sebagai feature extraction. Metode klasifikasi yang lebih tradisional menunjukkan potensi yang luar biasa dalam kecepatan dengan lebih cepat lebih dari 500 kali lipat dari model awal, namun sangat mengorbankan hasil pada akurasi dan metriks lainnya. Dalam kasus produksi secara real time, besarnya kesalahan prediksi nantinya akan mempersulit pengguna sehingga tidak praktis dalam implementasi. Model VGG16 sendiri akhirnya dipilih karena memiliki potensi yang lebih untuk dieksplorasi lebih jauh dengan kemampuannya yang cukup stabil di semua metriks penilaian.

Penelitian yang dilakukan sebelumnya [2] yang membahas defect pada kain klowong dengan menggunakan metode integral image memang saat ini memiliki beberapa keunggulan terutama dalam waktu komputasi sebesar 0.019 detik dan akurasi 96% jika dibandingkan dengan Model VGG16 dengan waktu komputasi dan akurasi yang terdapat pada Tabel 7. Namun, penelitian tersebut memiliki keterbatasan untuk mengenali pola atau jenis defect yang akan muncul jika kedepannya ditemukan lebih banyak variasi defect pada saat produksi kain klowong. Penelitian ini menunjukkan adanya potensi yang baik pada pengenalan jenis defect yang muncul dengan memanfaatkan metode klasifikasi. Setelah melakukan percobaan dengan mengganti FCL dengan beberapa metode konvensional, dapat dilihat bahwa penggantian FCL sangat berpengaruh terhadap waktu prediksi dan juga akurasi, sehingga untuk dapat meningkatkan akurasi pada kasus defect klowong, maka dapat dilakukan penelitian yang lebih mendalam mengenai struktur FCL yang cocok dengan kasus ini, dan juga menambahkan jumlah dataset yang digunakan. Untuk penelitian selanjutnya, model VGG16 dapat dimodifikasi dan dieksplorasi lebih lanjut untuk menaikkan nilai metriks akurasi dan juga mempercepat waktu komputasinya. Penelitian selanjutnya juga dapat menambahkan beberapa klasifikasi cacat yang lebih bervariasi yang kemungkinan akan muncul pada proses pembuatan kain klowong menggunakan mesin batik tulis.

5. KESIMPULAN

Berdasarkan hasil studi yang dilakukan, metode transfer learning terbukti mampu mengidentifikasi dan mengklasifikasikan defects pada kain klowong dengan hasil yang cukup baik. Untuk penelitian selanjutnya, kami ingin mengeksplorasi kembali metode yang lebih ringan dengan modifikasi CNN untuk kasus klasifikasi karena hasil saat ini masih memiliki potensi untuk ditingkatkan. Tantangan dalam penelitian ini adalah pola tiap kain yang berbeda sehingga model terkadang kesulitan untuk membedakan antara defect dan pattern klowong. Hal ini dikarenakan warna defect dengan warna dari pattern klowong hampir mirip dan terkadang sama. Melihat dari performa model, hasil pada training menunjukkan bahwa model VGG16 dan MobilenetV2 merupakan model terbaik dari segi akurasi dan loss, namun pada saat fase testing VGG16 sedikit lebih superior diandingkan dengan model lain seperti Resnet50V2, InceptionV3 dan MobilenetV2 dengan nilai akurasi prediksi berturut turut sebesar 92%, 88%, 87% dan 88%. Model VGG16 yang dikombinasikan dengan FCL juga memiliki hasil yang superior pada akurasi dibandingkan dengan VGG16-SVM dan VGG-RF dengan akurasi 92%, 83% dan 81%.

DAFTAR PUSTAKA

- [1] E. Sugiarto, F. Budiman, N. Hendriyanto, and A. Fahmi, "Optimization of Discrete Wavelet Transform Based on Gray Level Co-occurrence Matrix (GLCM) and Principal Component Analysis (PCA) to Improve Features Extraction on Batik Pattern Recognition," *IJETAE*, 2023, doi: 10.46338/ijetae0523_12.
- [2] D. S. E. Atmaja, S. Wibirama, M. K. Herliansyah, and A. Sudiarso, "Comparative study of integral image and normalized cross-correlation methods for defect detection on Batik klowong fabric," *Results in Engineering*, vol. 25, p. 104124, Mar. 2025, doi: 10.1016/j.rineng.2025.104124.

-
- [3] F. Abdullah, S. Bt. Basar, D. M. Sya'bany, and A. Mohd. Adnan, "Transformation from manual to digital: Artificial Intelligence in batik education for 5.0 society," *SHS Web Conf.*, vol. 197, p. 03001, 2024, doi: 10.1051/shsconf/202419703001.
- [4] S. B. Silah, B. Isa, Ponimin, and T. Karyono, "Malaysian and Indonesian Batik: The Best Practices," *KnE Social Sciences*, pp. 47–55, 2021, doi: 10.18502/kss.v5i6.9177.
- [5] M. N. D. Ulhaq and A. Sudiarso, "Parameter Optimization of Writing Batik Machine with Synthetic & Natural Fabric," *J. Ind. Eng. Edu.*, vol. 1, no. 2, pp. 1–35, 2023, [Online]. Available: <https://jieee.bksti.org/index.php/iee/article/view/20>
- [6] M. W. Ilahi, C. N. Apriyani, A. Desiani, N. Gofar, Y. Andriani, and M. R. Halim, "Classification of Geometric Batik Motif Typical of Indonesian Using Convolutional Neural Network," *j. Teknik inform.*, vol. 15, no. 1, pp. 91–100, 2022, doi: 10.15408/jti.v15i1.24968.
- [7] W. Chen, B. Yang, J. Li, and J. Wang, "An Approach to Detecting Diabetic Retinopathy Based on Integrated Shallow Convolutional Neural Networks," *IEEE Access*, vol. 8, pp. 178552–178562, 2020, doi: 10.1109/ACCESS.2020.3027794.
- [8] H. Song, X.-Y. Han, C. E. Montenegro-Marin, and S. Krishnamoorthy, "Retraction Note: Secure prediction and assessment of sports injuries using deep learning based convolutional neural network," *J Ambient Intell Human Comput*, vol. 15, no. 8, pp. 3273–3273, 2024, doi: 10.1007/s12652-024-04823-w.
- [9] F. A. A., A. Soleymani, and H. Khosravi, "A Method for Automatic Lane Detection using a Deep Network," *IJE*, vol. 35, no. 04, pp. 802–809, 2022, doi: 10.5829/IJE.2022.35.04A.20.
- [10] Z. Bai and J. Jing, "Mobile-Deeplab: a lightweight pixel segmentation-based method for fabric defect detection," *J Intell Manuf*, vol. 35, no. 7, pp. 3315–3330, Oct. 2024, doi: 10.1007/s10845-023-02205-1.
- [11] I. A. Dewi and M. A. N. E. Salawangi, "High performance of optimizers in deep learning for cloth patterns detection," *IJ-AI*, vol. 12, no. 3, p. 1407, Sep. 2023, doi: 10.11591/ijai.v12.i3.pp1407-1418.
- [12] M. Khodadadi, L. Riazi, and S. Yazdani, "A Novel Ensemble Deep Learning Model for Building Energy Consumption Forecast," *IJE*, vol. 37, no. 6, pp. 1067–1075, 2024, doi: 10.5829/IJE.2024.37.06C.03.
- [13] E. Charoqdouz and H. Hassanpour, "Feature Extraction from Several Angular Faces Using a Deep Learning Based Fusion Technique for Face Recognition," *IJE*, vol. 36, no. 8, pp. 1548–1555, 2023, doi: 10.5829/IJE.2023.36.08B.14.
- [14] K. Azmi, S. Defit, and S. Sumijan, "Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat," *JURNAL TEKNOLOGI UNIVERSAL*, vol. 16, no. 1, pp. 28–40, 2023, doi: 10.52072/unitek.v16i1.504.
- [15] H. Fonda, "Klasifikasi Batik Riau Menggunakan Convolutional Neural Networks (CNN)," *jik*, vol. 9, no. 1, pp. 7–10, 2020, doi: 10.33060/JIK/2020/Vol9.Iss1.144.
- [16] J. Fan, W. K. Wong, J. Wen, C. Gao, D. Mo, and Z. Lai, "Fabric Defect Detection Using Deep Convolution Neural Network," *AATCC Journal of Research*, vol. 8, no. 1_suppl, pp. 143–150, 2021, doi: 10.14504/ajr.8.S1.18.
- [17] 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 EEI*, vol. 10, no. 3, pp. 1300–1307, Jun. 2021, doi: 10.11591/eei.v10i3.3035.
- [18] R. Mawan, "Klasifikasi motif batik menggunakan Convolutional Neural Network," *JNNLK*, pp. 45–50, 2020, doi: 10.36802/jnنالoka.2020.v1-no1-45-50.
- [19] E. Sugiarto, F. Budiman, and A. Fahmi, "Implementation of Deep Learning Based on Convolution Neural Network for Batik Pattern Recognition," *KINETIK*, 2025, doi: 10.22219/kinetik.v10i1.2019.
- [20] D. W. Pratama, M. Ismail, R. Nurraudah, A. P. Rifai, and H. T. Nguyen, "Classification of Metal Surface Defects Using Convolutional Neural Networks (CNN)," *Green Intell. Syst. Appl.*, vol. 5, no. 1, pp. 93–105, 2025, doi: 10.53623/gisa.v5i1.581.
- [21] S. R. Arshad and M. K. Shahzad, "Deep Learning Based Fabric Defect Detection," *Res. Rep. Comput. Sci.*, vol. 3, no. 1, Mar. 2024, doi: 10.37256/rrcs.3120244156.
-

-
- [22] A. S. Paymode and V. B. Malode, "Transfer Learning for Multi-Crop Leaf Disease Image Classification using Convolutional Neural Network VGG," *Artificial Intelligence in Agriculture*, vol. 6, pp. 23–33, 2022, doi: 10.1016/j.aiaa.2021.12.002.
- [23] U. Sanath Rao *et al.*, "Deep Learning Precision Farming: Grapes and Mango Leaf Disease Detection by Transfer Learning," *Global Transitions Proceedings*, vol. 2, no. 2, pp. 535–544, 2021, doi: 10.1016/j.gltp.2021.08.002.
- [24] M. Mohsin, O. S. Balogun, K. Haataja, and P. Toivanen, "Real-time defect detection and classification on wood surfaces using deep learning," *ei*, vol. 34, no. 10, pp. 382-1-382–6, Jan. 2022, doi: 10.2352/EI.2022.34.10.IPAS-382.
- [25] M. D. Bloice, P. M. Roth, and A. Holzinger, "Biomedical image augmentation using Augmentor," *Bioinformatics*, vol. 35, no. 21, pp. 4522–4524, 2019, doi: 10.1093/bioinformatics/btz259.
- [26] P. A. Pramesti *et al.*, "Classification of Clove Leaf Blister Blight Disease Severity Using Pre-trained Model VGG16, InceptionV3, and ResNet," *Jurnal Ilmu Komputer dan Informasi*, vol. 17, no. 2, pp. 159–166, 2024, doi: 10.21609/jiki.v17i2.1237.
- [27] Z. Jiang, "A Novel Crop Weed Recognition Method Based on Transfer Learning from VGG16 Implemented by Keras," *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 677, no. 3, p. 032073, 2019, doi: 10.1088/1757-899X/677/3/032073.
- [28] I. Ramadhanti, A. Prasetiadi, and I. Kresna A, "CLOTHING RECOMMENDATION AND FACE SWAP MODEL BASED ON VGG16, AUTOENCODER, AND FACIAL LANDMARK POINTS," *J. Tek. Inform. (JUTIF)*, vol. 5, no. 1, pp. 19–29, 2024, doi: 10.52436/1.jutif.2024.5.1.1016.
- [29] Y. Pamungkas, M. R. N. Ramadani, and E. N. Njoto, "Effectiveness of CNN Architectures and SMOTE to Overcome Imbalanced X-Ray Data in Childhood Pneumonia Detection," *Journal of Robotics and Control*, vol. 5, no. 3, pp. 775–785, 2024, doi: 10.18196/jrc.v5i3.21494.
- [30] D. R. A. N. Pratama and Y. Azhar, "Convolutional Neural Network for COVID-19 Detection Using InceptionV3 Transfer Learning," *J. Tek. Inform. (JUTIF)*, vol. 6, no. 2, pp. 483–500, 2025, doi: 10.52436/1.jutif.2025.6.2.4094.
- [31] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," 2017, *arXiv*: arXiv:1704.04861. doi: 10.48550/arXiv.1704.04861.
- [32] K. R. Prilianti, V. V. Oktariyanto, and H. Setiawan, "Ornamental Plant Identification System Using Transfer Learning On Convolutional Neural Network," *J. Tek. Inform. (JUTIF)*, vol. 5, no. 4, pp. 1015–1023, 2024, doi: 10.52436/1.jutif.2024.5.4.1964.
- [33] T. T. Mengistie and D. Kumar, "Covid-19 Face Mask Detection Using Convolutional Neural Network and Image Processing," in *2021 2nd International Conference for Emerging Technology (INCET)*, Belagavi, India: IEEE, May 2021, pp. 1–7. doi: 10.1109/INCET51464.2021.9456288.
- [34] W. Wahyuningsih, G. S. Nugraha, and R. Dwiyanaputra, "Classification Of Dental Caries Disease In Tooth Images Using A Comparison Of Efficientnet-B0, Mobilenetv2, Resnet-50, Inceptionv3 Architectures," *J. Tek. Inform. (JUTIF)*, vol. 5, no. 4, pp. 177–185, 2024, doi: 10.52436/1.jutif.2024.5.4.2187.
- [35] A. H. I. Al-Rammahi, "Face mask recognition system using MobileNetV2 with optimization function," *Applied Artificial Intelligence*, vol. 36, no. 1, p. 2145638, 2022, doi: 10.1080/08839514.2022.2145638.
- [36] P. Yuan and C. Zhu, "Frequency Ensemble Based on Multi-Band Checkpoint Saving Mechanism for Time Series Forecasting," in *2024 9th International Conference on Intelligent Computing and Signal Processing (ICSP)*, Xian, China: IEEE, 2024, pp. 714–717. doi: 10.1109/ICSP62122.2024.10743879.
-

