(JUSS) Jurnal Sains dan Sistem Informasi ISSN 2614-8277
Vol.9 No.1, Juli 2026

PERBANDINGAN EFFICIENTNETV2 DAN MOBILENETV3 PADA CNN UNTUK KLASIFIKASI GAMBAR ASLI DAN GAMBAR AI

Danish Wiedi Marchello Sandi[1)], Pradita Eko Prasetyo Utomo[2)], Ulfa Khaira[3)]
[1]Fakultas Sains dan Teknologi, Universitas Jambi
email: wiedims@gmail.com
[2]Fakultas Sains dan Teknologi, Universitas Jambi
email: pradita.eko@unja.ac.id
[3]Fakultas Sains dan Teknologi, Universitas Jambi
email: ulfakhaira@unja.ac.id

Abstract

The rapid advancement of generative artificial intelligence, particularly Generative Adversarial Networks and Diffusion Models, has enabled the creation of synthetic images with a visual quality that is increasingly difficult to distinguish from authentic photographs, raising concerns over misinformation, media manipulation, and digital identity misuse. This study implements a Convolutional Neural Network (CNN) to classify real and AI-generated images and compares the performance of two transfer learning architectures, EfficientNetV2-B0 and MobileNetV3-Large, against a CNN trained from scratch. The dataset consists of 10,930 images collected from two Kaggle repositories, comprising 5,508 AI-generated images and 5,422 real images, which were split into 80% training, 10% validation, and 10% testing data, resized to 224x224 pixels, and augmented prior to training using a batch size of 32, a maximum of 20 epochs, a learning rate of 0.001, and the Adam optimizer. The experimental results show that MobileNetV3-Large achieved the best performance with a training accuracy of 96.84%, a validation accuracy of 96.25%, a testing accuracy of 96.71%, and a testing loss of 0.1031, outperforming EfficientNetV2-B0 (94.41% testing accuracy) and the CNN trained from scratch (91.58% testing accuracy). Hyperparameter experiments further confirm that a batch size of 32 combined with 20 training epochs produces the most stable convergence across all three architectures. The best-performing model was subsequently deployed as a REST API using the Flask framework to support real-time image classification. These findings indicate that transfer learning, particularly with the MobileNetV3-Large architecture, provides an effective and computationally efficient approach for detecting AI-generated images.

Keywords: *AI-Generated Image, Convolutional Neural Network, Transfer Learning, EfficientNetV2, MobileNetV3*

1. PENDAHULUAN

Perkembangan teknologi kecerdasan buatan (Artificial Intelligence/AI) yang berlangsung sangat pesat dalam beberapa tahun terakhir telah membawa perubahan signifikan dalam berbagai bidang, khususnya pada pengolahan citra digital. Kemajuan model generatif seperti Generative Adversarial Network (GAN) dan Diffusion Model memungkinkan pembuatan gambar sintetis yang memiliki kualitas visual sangat tinggi dan semakin sulit dibedakan dari gambar asli melalui pengamatan manusia (Nugroho, 2025). Teknologi ini telah dimanfaatkan pada berbagai sektor, mulai dari industri kreatif, hiburan, pendidikan, kesehatan, hingga pemasaran digital. Namun demikian, di sisi lain, kemampuannya dalam menghasilkan gambar yang tampak realistis juga menimbulkan berbagai tantangan baru terkait autentikasi dan verifikasi keaslian konten digital (Sa'adah et al., 2025).

Meningkatnya jumlah gambar yang dihasilkan oleh AI memunculkan kekhawatiran terhadap penyebaran informasi palsu, manipulasi media, serta penyalahgunaan identitas digital. Gambar sintetis yang dibuat menggunakan teknologi AI dapat dimanfaatkan untuk menyebarkan hoaks, membuat bukti visual palsu, maupun mendukung praktik penipuan digital (Sianipar, 2026). Kondisi ini menyebabkan kebutuhan akan sistem yang mampu membedakan antara gambar asli dan gambar buatan AI menjadi semakin penting, sehingga berbagai penelitian di bidang forensik digital dan computer vision terus dikembangkan untuk mendeteksi

karakteristik khas yang terdapat pada gambar sintetis (Roumpies et al., 2025).

Salah satu pendekatan yang banyak digunakan dalam klasifikasi gambar adalah metode Convolutional Neural Network (CNN). CNN memiliki kemampuan yang sangat baik dalam mengekstraksi fitur visual secara otomatis dan telah menunjukkan performa tinggi pada berbagai tugas klasifikasi citra (Wijaya et al., 2025). Beberapa penelitian menunjukkan bahwa arsitektur CNN mampu mengidentifikasi perbedaan antara gambar asli dan gambar yang dihasilkan AI dengan tingkat akurasi yang cukup tinggi melalui pembelajaran pola visual yang terdapat pada data citra (Mu et al., 2024). Meskipun demikian, pelatihan CNN dari awal (training from scratch) membutuhkan jumlah data yang sangat besar dan biaya komputasi yang tinggi, sehingga pendekatan ini tidak selalu efisien untuk diterapkan pada dataset yang berukuran terbatas (Simangunsong et al., 2025).

Sebagai solusi atas keterbatasan tersebut, teknik transfer learning hadir sebagai pendekatan yang memanfaatkan pengetahuan yang telah dipelajari oleh suatu model pada tugas atau dataset tertentu untuk membantu menyelesaikan tugas lain yang memiliki karakteristik serupa (Ali & Abdulazeez, 2024). Pada kondisi ketika jumlah data terbatas, penggunaan transfer learning dapat meningkatkan akurasi model sekaligus mengurangi risiko overfitting (Plested et al., 2026), baik melalui pendekatan feature extraction dengan membekukan lapisan awal model, maupun fine-tuning dengan memperbarui sebagian lapisan model pra-latih agar lebih sesuai dengan data target (Tan et al., 2018). Di antara berbagai arsitektur pre-trained yang tersedia, EfficientNetV2 dan MobileNetV3 merupakan arsitektur yang cukup sering dijadikan rujukan dalam penelitian berbasis transfer learning untuk klasifikasi citra. EfficientNetV2 dikembangkan sebagai penerus EfficientNet dengan peningkatan signifikan pada efisiensi pelatihan melalui kombinasi teknik compound scaling dan progressive learning (Tan & Le, 2021), serta telah terbukti efektif dalam berbagai tugas klasifikasi berbasis citra (Zhao et al., 2025). Sementara itu, MobileNetV3 dirancang khusus untuk perangkat mobile dan edge device dengan mengintegrasikan teknik Neural Architecture Search serta mekanisme squeeze-and-excitation yang memungkinkan model secara adaptif memperkuat fitur informatif dan menekan fitur yang kurang relevan (Howard et al., 2019).

Penelitian ini bertujuan untuk mengimplementasikan dan membandingkan efektivitas model transfer learning EfficientNetV2-B0 dan MobileNetV3-Large dengan CNN dari awal dalam mengklasifikasikan gambar asli dan gambar buatan AI, sekaligus mengevaluasi pengaruh hyperparameter epoch dan batch size terhadap performa model. Hasil penelitian ini diharapkan dapat memberikan kontribusi pada pengembangan sistem deteksi gambar sintetis yang lebih akurat dan efisien, memperkaya literatur di bidang computer vision dan forensik digital, serta menjadi acuan bagi pengembangan sistem serupa yang dapat diimplementasikan secara praktis melalui layanan REST API.

2. METODE PENELITIAN

A. Tempat dan Waktu Penelitian

Penelitian ini dilaksanakan di Kampus Universitas Jambi Mendalo yang beralamat di Jalan Jambi–Muara Bulian No. KM. 15, Mendalo Darat, Kecamatan Jambi Luar Kota, Kabupaten Muaro Jambi, Provinsi Jambi. Adapun waktu pelaksanaan penelitian berlangsung selama kurang lebih satu bulan.

B. Alat Penelitian

Perangkat keras yang digunakan dalam penelitian ini adalah laptop personal dengan prosesor AMD Ryzen 5 5600U, RAM sebesar 16 GB, penyimpanan 512 GB, serta kartu grafis Nvidia RTX 3050 Laptop yang berjalan pada sistem operasi Windows 11. Proses pelatihan model dilakukan pada platform komputasi awan Google Colab dengan Python versi 3.12.13, RAM sebesar 12,7 GB, kapasitas disk sekitar 112,6 GB, serta akselerator Tensor Processing Unit (TPU) Tesla T4. Google Colab dipilih karena telah terintegrasi dengan berbagai pustaka deep learning seperti TensorFlow dan Keras, sehingga memudahkan proses implementasi, pelatihan, validasi, dan evaluasi model CNN tanpa memerlukan instalasi perangkat lunak khusus pada komputer lokal.

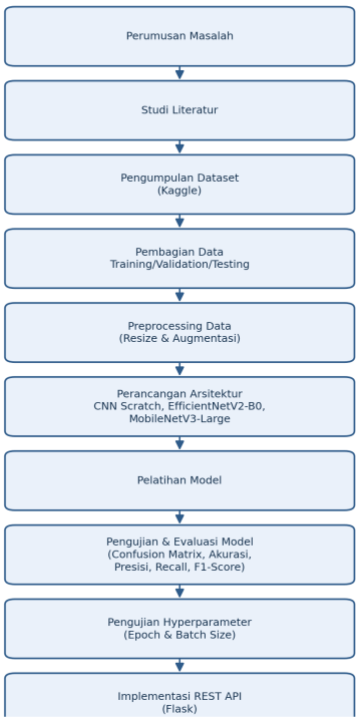
Dataset penelitian yang digunakan meliputi dataset citra gambar asli dan gambar

 ISSN 2614-8277

buatan AI yang diperoleh dari platform Kaggle, pustaka pemrograman TensorFlow 2.20.0 beserta modul pendukung seperti Keras, scikit-learn, NumPy, dan Pandas, serta framework Flask yang digunakan pada tahap implementasi REST API. Selain itu, digunakan pula studi literatur berupa jurnal, artikel, dan publikasi ilmiah yang relevan dengan metode transfer learning, arsitektur EfficientNetV2 dan MobileNetV3, serta teknik klasifikasi citra berbasis CNN.

C. Alur Penelitian

Penelitian ini dirancang melalui serangkaian tahapan sistematis guna membandingkan performa CNN dari awal dengan CNN yang menerapkan transfer learning EfficientNetV2-B0 dan MobileNetV3-Large dalam mengklasifikasikan gambar asli dan gambar buatan AI.



Gambar 1. Alur Penelitian

Tahapan pertama adalah perumusan masalah, yaitu proses identifikasi kebutuhan akan sistem yang mampu membedakan gambar asli dan gambar buatan AI secara otomatis. Tahapan kedua adalah studi literatur, yaitu penelusuran berbagai sumber rujukan ilmiah terkait CNN, transfer learning, EfficientNetV2, dan MobileNetV3 guna membangun landasan teoretis penelitian. Tahapan ketiga adalah pengumpulan dataset, yang dilakukan dengan mengunduh dan menggabungkan data dari dua sumber Kaggle. Tahapan keempat adalah pembagian data menjadi data training, validation, dan testing menggunakan teknik stratified split agar proporsi kelas tetap seimbang.

Tahapan kelima adalah preprocessing data, yang meliputi proses resize citra menjadi 224×224 piksel serta augmentasi data guna memperkaya variasi data pelatihan dan mengurangi risiko overfitting. Tahapan keenam adalah perancangan arsitektur, yaitu penyusunan tiga model CNN yang akan dibandingkan, yaitu CNN Scratch, CNN dengan transfer learning EfficientNetV2-B0, dan CNN dengan transfer learning MobileNetV3-Large. Tahapan ketujuh adalah pelatihan model menggunakan optimizer Adam dengan learning rate 0,001, dilengkapi mekanisme EarlyStopping dan ReduceLROnPlateau. Tahapan kedelapan adalah pengujian dan evaluasi model menggunakan confusion matrix serta metrik akurasi, presisi, recall, dan F1-score. Tahapan kesembilan adalah pengujian hyperparameter epoch dan batch size untuk memperoleh konfigurasi paling optimal. Tahapan terakhir adalah implementasi model terbaik ke dalam layanan REST API berbasis Flask agar dapat digunakan untuk klasifikasi gambar secara real-time.

D. Populasi dan Sampel Data

Populasi dalam penelitian ini merujuk pada keseluruhan citra gambar asli dan gambar buatan AI yang tersedia secara publik pada platform Kaggle. Sampel data yang digunakan pada penelitian ini berjumlah 10.930 citra yang diperoleh dari dua sumber, yaitu dataset "muhammadsaoodsarwar/ai-vs-real-192-class-scene-image-dataset" yang berisi 3.508 gambar AI dan 5.422 gambar asli, serta tambahan 2.000 gambar AI dari dataset "doctorstrange420/real-and-fake-ai-generated-art-images-dataset" guna menyeimbangkan proporsi kedua kelas, sehingga total gambar buatan AI menjadi 5.508 citra. Dataset yang telah digabungkan tersebut kemudian dibagi menjadi 80% data training, 10% data validation, dan 10% data testing menggunakan teknik stratified random sampling agar distribusi kelas pada setiap subset tetap proporsional.

 ISSN 2614-8277

E. Teknik Analisis Data

Analisis data pada penelitian ini dilakukan secara kuantitatif dengan mengimplementasikan algoritma Convolutional Neural Network (CNN) yang dikenal mampu mengekstraksi fitur visual dan mengklasifikasikan data citra dengan baik. Proses pelatihan model menggunakan fungsi loss Categorical Crossentropy, optimizer Adam dengan learning rate 0,001, batch size 32, serta maksimum 20 epoch yang dilengkapi EarlyStopping dengan patience 5 dan ReduceLROnPlateau untuk menjaga stabilitas konvergensi. Evaluasi performa model dilakukan menggunakan confusion matrix serta metrik akurasi, presisi, recall, dan F1-score, dengan persamaan sebagai berikut:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1\ Score = 2 \times \left(\frac{Precision \times Recall}{Precission + Recall}\right)$$

Selain pengujian arsitektur, dilakukan pula eksperimen pengaruh hyperparameter berupa variasi jumlah epoch (10, 15, dan 20) serta variasi batch size (16 dan 32) untuk memperoleh konfigurasi yang menghasilkan akurasi paling optimal.

3. HASIL DAN PEMBAHASAN

A. Hasil

Hasil Pengumpulan Dataset

Pada tahap awal penelitian, dikumpulkan dataset yang berisikan citra gambar asli dan gambar buatan AI. Jumlah keseluruhan data yang berhasil dikumpulkan adalah 10.930 citra yang diperoleh dari dua sumber Kaggle dan digabungkan menjadi satu dataset. Distribusi jumlah data pada masing-masing kelas ditunjukkan pada Tabel 1.

TABEL 1
Distribusi Dataset Berdasarkan Kelas

Kelas	Jumlah Citra	Persentase
Gambar Buatan AI	5.508	50,39%
Gambar Asli	5.422	49,61%
Total	**10.930**	**100%**

Merujuk pada Tabel 1, proporsi antara gambar buatan AI dan gambar asli relatif seimbang, yaitu masing-masing sebesar 50,39% dan 49,61%. Keseimbangan proporsi kelas ini penting untuk menghindari bias model terhadap salah satu kelas selama proses pelatihan, sehingga model dapat mempelajari karakteristik kedua kelas secara adil.

Hasil Pembagian Data Training, Validation, dan Testing

Dataset yang telah dikumpulkan selanjutnya dibagi menjadi tiga subset menggunakan teknik stratified random sampling dengan proporsi 80% data training, 10% data validation, dan 10% data testing. Hasil pembagian data ditunjukkan pada Tabel 2.

TABEL 2
Pembagian Data Training, Validation, dan Testing

Subset Data	Jumlah Citra	Proporsi
Training	8.744	80%
Validation	1.093	10%
Testing	1.093	10%
Total	**10.930**	**100%**

Hasil preprocessing Data

Seluruh citra pada dataset diubah ukurannya menjadi 224×224 piksel dengan input shape 224×224×3 agar sesuai dengan input standar arsitektur CNN yang digunakan. Sebelum diproses lebih lanjut, data training terlebih dahulu melalui tahap augmentasi berupa normalisasi piksel (rescale 1/255), rotation range 30°, width shift dan height shift sebesar 0,2, shear range 0,2, zoom range 0,2, serta horizontal flip. Khusus untuk model EfficientNetV2-B0, proses rescaling manual tidak diterapkan karena arsitektur tersebut telah memiliki lapisan preprocessing bawaan, sehingga penambahan rescaling secara manual berpotensi menyebabkan proses normalisasi dilakukan dua kali dan mengganggu performa model.

Hasil Perancangan Arsitektur

Arsitektur CNN Scratch yang dirancang terdiri atas tiga blok konvolusi dengan jumlah filter masing-masing 128, 64, dan 32, yang setiap blok diikuti oleh fungsi aktivasi ReLU, lapisan MaxPooling2D, dan Batch Normalization. Hasil dari tiga blok konvolusi tersebut kemudian diteruskan ke lapisan Flatten, Dense dengan 256 unit, Dropout

 ISSN 2614-8277

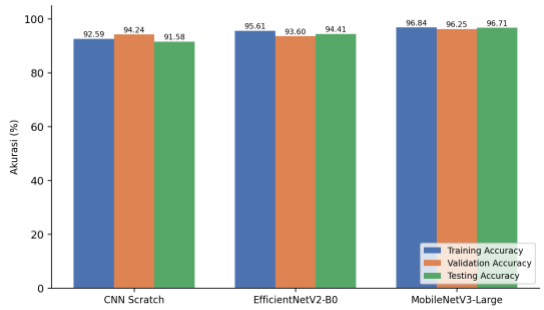
sebesar 0,5, serta lapisan output Dense dengan 2 neuron dan fungsi aktivasi Softmax. Pada model CNN dengan transfer learning EfficientNetV2-B0 dan MobileNetV3-Large, digunakan bobot pre-trained ImageNet yang dibekukan (frozen), sementara hanya lapisan Global Average Pooling, Dense 256 unit dengan aktivasi ReLU, Dropout 0,5, serta lapisan output Dense dengan aktivasi Softmax yang dilatih ulang. Pendekatan feature extraction ini dipilih untuk mempercepat proses pelatihan dan mengurangi risiko overfitting mengingat jumlah parameter yang dilatih menjadi jauh lebih sedikit dibandingkan melatih keseluruhan arsitektur dari awal.

Hasil Pelatihan dan Pengujian Model

Proses pelatihan dilakukan menggunakan batch size 32, maksimum 20 epoch, learning rate 0,001, dan optimizer Adam. Pengujian dilakukan terhadap 1.093 data uji yang terdiri atas 551 citra gambar buatan AI dan 542 citra gambar asli. Ringkasan hasil akurasi dan loss pada data training, validation, dan testing untuk ketiga model ditunjukkan pada Tabel 3 dan divisualisasikan pada Gambar 2.

TABEL 3
Ringkasan Hasil Pelatihan dan Pengujian Model

Model	Training Acc.	Validation Acc.	Testing Acc.	Testing Loss
CNN Scratch	92,59%	94,24%	91,58%	0,2355
Efficient NetV2-B0	95,61%	93,60%	94,41%	0,1531
MobileNetV3-Large	96,84%	96,25%	96,71%	0,1031

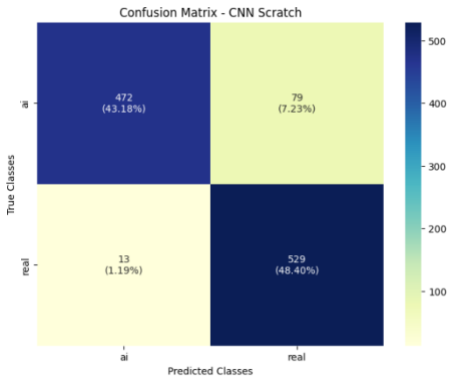


Gambar 2. Grafik Perbandingan Akurasi Training, Validation, dan Testing

Berdasarkan Tabel 3 dan Gambar 2, model CNN dengan transfer learning MobileNetV3-Large secara konsisten memperoleh nilai akurasi tertinggi pada ketiga subset data, yaitu 96,84% pada data training, 96,25% pada data validation, dan 96,71% pada data testing, dengan nilai loss testing terendah sebesar 0,1031. Model CNN dengan transfer learning EfficientNetV2-B0 menempati posisi kedua dengan akurasi testing sebesar 94,41% dan loss 0,1531, sementara model CNN Scratch memperoleh hasil paling rendah dengan akurasi testing sebesar 91,58% dan loss tertinggi sebesar 0,2355.

Hasil Confusion Matrix Model CNN Scratch

Guna memberikan gambaran lebih rinci mengenai distribusi kesalahan klasifikasi, dilakukan analisis confusion matrix terhadap model CNN Scratch sebagaimana ditunjukkan pada Gambar 3.



Gambar 3. Confusion Matrix Model CNN Scratch

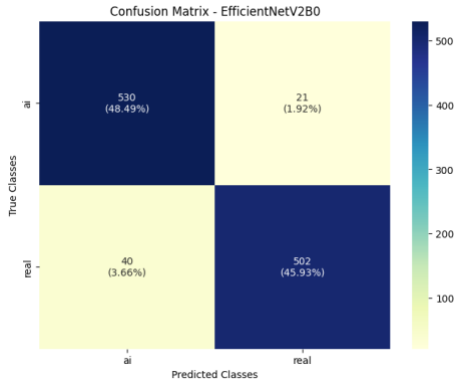
Berdasarkan Gambar 3, dari 551 citra AI yang diujikan, model CNN Scratch berhasil mengklasifikasikan 472 citra dengan benar sebagai citra AI, namun masih terdapat 79 citra AI yang salah diklasifikasikan sebagai citra asli. Sementara itu, dari 542 citra asli, model berhasil mengklasifikasikan 529 citra dengan benar, dan hanya 13 citra asli yang salah diprediksi sebagai citra AI. Kesalahan klasifikasi yang lebih tinggi pada kelas AI dibandingkan kelas Real mengindikasikan bahwa sebagian citra buatan AI memiliki karakteristik visual yang menyerupai citra asli, sehingga lebih sulit dibedakan oleh model dengan kapasitas representasi fitur yang terbatas seperti CNN Scratch.

Hasil Confusion Matrix Model CNN dengan EfficientNetV2-B0

Untuk memberikan gambaran lebih rinci mengenai distribusi kesalahan klasifikasi, dilakukan analisis confusion matrix terhadap

 ISSN 2614-8277

model CNN dengan EfficientNetV2-B0 yang ditunjukkan pada Gambar 4.

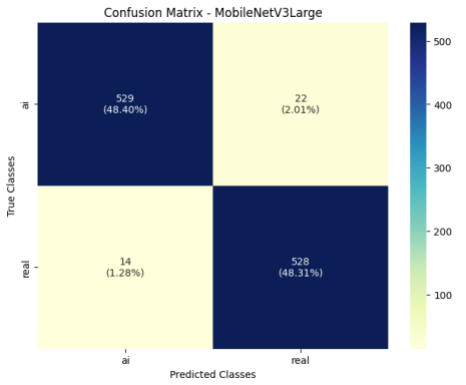


Gambar 4. Confusion Matrix Model CNN dengan EfficientNetV2-B0

Berdasarkan Gambar 4, dari 551 citra AI yang diujikan, model CNN Scratch berhasil mengklasifikasikan 530 citra dengan benar sebagai citra AI dan hanya 21 citra AI yang salah diklasifikasikan sebagai citra asli. Sementara itu, dari 542 citra asli, model berhasil mengklasifikasikan 502 citra dengan benar, namun masih terdapat 40 citra asli yang salah diprediksi sebagai citra AI. Kesalahan klasifikasi yang lebih tinggi pada kelas Real dibandingkan kelas AI mengindikasikan bahwa sebagian citra buatan Real memiliki karakteristik visual yang menyerupai citra AI, sehingga lebih sulit dibedakan oleh model dengan kapasitas representasi fitur CNN dengan EfficientNetV2-B0.

Hasil Confusion Matrix Model CNN dengan MobileNetV3-Large

Untuk memberikan gambaran lebih rinci mengenai distribusi kesalahan klasifikasi, dilakukan analisis confusion matrix terhadap model CNN dengan MobileNetV3-Large yang ditunjukkan pada Gambar 5.



Gambar 5. Confusion Matrix Model CNN dengan MobileNetV3-Large

Berdasarkan Gambar 5, dari 551 citra AI yang diujikan, model CNN Scratch berhasil mengklasifikasikan 529 citra dengan benar sebagai citra AI dan hanya 22 citra AI yang salah diklasifikasikan sebagai citra asli. Sementara itu, dari 542 citra asli, model berhasil mengklasifikasikan 528 citra dengan benar dan hanya 14 citra asli yang salah diprediksi sebagai citra AI. Kesalahan klasifikasi antara kelas Real dan kelas AI cenderung kecil yang mengidentifikasi model MobileNetV3-Large sudah cukup baik dalam membedakan fitur antara kelas Real dan kelas AI.

Hasil Pengujian Hyperparameter Epoch

Pengujian pengaruh jumlah epoch dilakukan dengan menggunakan nilai learning rate 0,001, dimensi citra 224×224 piksel, dan batch size 32 pada ketiga model. Hasil pengujian ditunjukkan pada Tabel 4, 5, dan 6.

TABEL 4
Pengaruh Jumlah Epoch pada Model CNN Scratch

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss	Time
10	86,87%	89,94%	31,62%	26,57%	26 menit
15	90,60%	92,41%	25,15%	22,01%	41 menit 54 detik
20	92,59%	94,24%	20,32%	20,77%	49 menit 45 detik

TABEL 5
Pengaruh Jumlah Epoch pada Model CNN dengan EfficientNetV2-B0

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss	Time
10	95,89%	93,96%	10,40%	15,26%	24 menit 15 detik
15	96,20%	94,88%	9,76%	13,42%	29 menit 47 detik
20	96,52%	95,06%	9,08%	12,16%	41 menit 52 detik

TABEL 6
Pengaruh Jumlah Epoch pada Model CNN dengan MobileNetV3-Large

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss	Time
10	96,57%	93,78%	9,07%	13,80%	27 menit 48 detik
15	96,52%	94,69%	9,21%	13,18%	28 menit 41 detik
20	96,84%	96,25%	8,64%	10,96%	36 menit 31 detik

Berdasarkan Tabel 4, 5, dan 6, penambahan jumlah epoch dari 10 hingga 20 secara konsisten meningkatkan akurasi validasi pada ketiga model, meskipun peningkatan tersebut mulai melambat setelah epoch ke-15. Kombinasi epoch 20 dengan learning rate 0,001 dan batch size 32 memberikan hasil akurasi validasi dan loss terbaik pada seluruh model, sehingga konfigurasi ini dipilih sebagai konfigurasi akhir pada penelitian ini.

Hasil Pengujian Hyperparameter Batch Size

Pengujian pengaruh batch size dilakukan dengan menggunakan nilai learning rate 0,001, dimensi citra 224×224 piksel, dan jumlah epoch 20 pada ketiga model. Hasil pengujian ditunjukkan pada Tabel 7, 8, dan 9.

TABEL 7
Pengaruh Batch Size pada Model CNN Scratch

Batch Size	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss	Time
16	85,74%	88,29%	34,80%	28,64%	39 menit 42 detik
32	92,59%	94,24%	20,32%	20,77%	49 menit 45 detik

TABEL 8
Pengaruh Batch Size pada Model CNN dengan EfficientNetV2-B0

Batch Size	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss	Time
16	96,28%	93,41%	9,14%	15,50%	28 menit 2 detik
32	96,52%	95,06%	9,08%	12,16%	41 menit 52 detik

TABEL 9
Pengaruh Batch Size pada Model CNN dengan MobileNetV3-Large

Batch Size	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss	Time
16	96,40%	93,14%	9,53%	17,44%	23 menit 40 detik
32	96,84%	96,25%	8,64%	10,96%	36 menit 31 detik

Tabel 7, 8, dan 9, batch size 32 secara konsisten menghasilkan akurasi validasi yang lebih tinggi dan loss yang lebih rendah dibandingkan batch size 16 pada ketiga model, meskipun membutuhkan waktu pelatihan yang sedikit lebih lama. Dengan demikian, kombinasi epoch 20 dan batch size 32 ditetapkan sebagai konfigurasi hyperparameter optimal pada penelitian ini.

Hasil Implementasi REST API

Model CNN dengan MobileNetV3-Large yang memperoleh performa terbaik selanjutnya diimplementasikan ke dalam layanan REST API menggunakan framework Flask. Layanan ini menyediakan dua endpoint utama, yaitu /api/health yang berfungsi untuk memeriksa kesiapan model, dan /api/predict yang menerima unggahan gambar berformat jpg, jpeg, atau png melalui form-data dengan key file, kemudian mengembalikan hasil klasifikasi dalam format JSON berupa label kelas beserta persentase probabilitasnya. Layanan REST API tersebut selanjutnya diintegrasikan ke dalam antarmuka website berbasis Flask, sehingga pengguna dapat mengunggah gambar secara langsung dan memperoleh hasil klasifikasi berupa label "asli" atau "ai" beserta tingkat kepercayaan (confidence) dari model.

B. Pembahasan

Penelitian ini dilaksanakan dengan tujuan membandingkan efektivitas CNN dari awal dengan CNN yang menerapkan transfer learning EfficientNetV2-B0 dan MobileNetV3-Large dalam mengklasifikasikan gambar asli dan gambar buatan AI. Hasil yang diperoleh menunjukkan bahwa kedua model transfer learning secara konsisten mengungguli CNN Scratch, yang membuktikan bahwa pemanfaatan bobot pre-trained ImageNet mampu memberikan representasi fitur awal yang lebih kaya dibandingkan model yang dilatih sepenuhnya dari awal dengan data yang relatif terbatas (Ali & Abdulazeez, 2024). Kondisi ini sejalan dengan pandangan bahwa transfer learning sangat bermanfaat pada kasus klasifikasi citra dengan ukuran dataset kecil hingga menengah, sebagaimana ditemukan pada penelitian sebelumnya (Tan et al., 2018).

Di antara kedua model transfer learning, MobileNetV3-Large secara konsisten menunjukkan performa lebih baik dibandingkan EfficientNetV2-B0 pada seluruh metrik yang diuji. Keunggulan ini dapat dikaitkan dengan mekanisme squeeze-and-excitation yang terdapat pada arsitektur MobileNetV3, yang memungkinkan model untuk secara adaptif memperkuat kanal fitur yang informatif dan menekan kanal yang kurang relevan terhadap tugas klasifikasi gambar asli dan gambar buatan AI (Howard et al., 2019). Selain itu, penggunaan fungsi aktivasi hard-swish pada MobileNetV3 turut memberikan efisiensi komputasi yang lebih baik dibandingkan fungsi aktivasi konvensional, tanpa mengorbankan akurasi model.

Hasil ini menarik untuk dibandingkan dengan temuan penelitian terdahulu yang menunjukkan hasil yang tidak selalu konsisten antara kedua arsitektur pada domain data yang berbeda. Wardhana & Nugroho (2025) melaporkan bahwa MobileNetV2 jauh lebih unggul dibandingkan MobileNetV3 pada klasifikasi jenis jeruk, sementara Rafliansyah

(2024) menemukan bahwa InceptionV3 lebih unggul dibandingkan EfficientNetV2-S pada klasifikasi gambar buah-buahan. Sebaliknya, temuan pada penelitian ini justru sejalan dengan hasil Wulandari (2026) yang juga melaporkan keunggulan MobileNetV3 dibandingkan EfficientNetV2 pada klasifikasi citra patung Bali, meskipun selisih akurasi pada penelitian tersebut relatif lebih kecil dibandingkan penelitian ini. Perbedaan-perbedaan tersebut menegaskan bahwa performa suatu arsitektur transfer learning sangat bergantung pada karakteristik domain data, kompleksitas fitur visual yang harus dipelajari, serta kesesuaian antara rancangan arsitektur dengan pola data yang dihadapi.

Rendahnya performa CNN Scratch dibandingkan kedua model transfer learning turut mengonfirmasi keterbatasan pelatihan model dari awal pada kondisi data yang relatif terbatas. Dengan hanya tiga blok konvolusi dan tanpa representasi fitur awal yang kaya, CNN Scratch cenderung lebih rentan terhadap overfitting maupun kesulitan dalam mempelajari pola visual yang kompleks, sebagaimana ditunjukkan oleh nilai loss testing yang jauh lebih tinggi (0,2355) dibandingkan kedua model transfer learning. Sebagaimana dijelaskan oleh Zhao (2025), regularisasi tambahan seperti augmentasi data adaptif dan dropout dapat membantu mencegah overfitting pada dataset berukuran kecil, namun tetap tidak dapat sepenuhnya menyamai keunggulan representasi fitur yang diperoleh melalui transfer learning dari model yang telah dilatih pada jutaan gambar.

Hasil confusion matrix pada model CNN Scratch turut memperlihatkan bahwa kesalahan klasifikasi lebih banyak terjadi pada citra AI yang diprediksi sebagai citra asli dibandingkan sebaliknya. Kondisi ini mengindikasikan bahwa gambar buatan AI generasi terbaru, khususnya yang dihasilkan oleh model Diffusion dan GAN modern, telah mencapai tingkat realisme visual yang sangat tinggi sehingga semakin sulit dibedakan dari gambar asli oleh model dengan kapasitas representasi fitur yang terbatas (Nugroho, 2025). Temuan ini memperkuat urgensi pengembangan sistem deteksi berbasis transfer learning yang mampu menangkap pola artefak halus pada gambar sintetis, sebagaimana yang menjadi fokus berbagai penelitian forensik digital terkini (Roumpies et al., 2025).

Hasil pengujian hyperparameter menunjukkan bahwa penggunaan batch size yang lebih besar (32) secara konsisten menghasilkan akurasi validasi yang lebih tinggi dan loss yang lebih rendah dibandingkan batch size yang lebih kecil (16) pada ketiga model. Temuan ini sejalan dengan pandangan bahwa batch size yang lebih besar mampu memanfaatkan kemampuan paralel perangkat keras secara lebih optimal dan menghasilkan estimasi gradien yang lebih stabil selama proses optimasi, meskipun pada beberapa kasus batch size yang terlalu besar berisiko menurunkan kemampuan generalisasi model (Keskar et al., 2017). Pada penelitian ini, batch size 32 belum menunjukkan indikasi penurunan generalisasi tersebut, sehingga dapat dianggap sebagai konfigurasi yang cukup seimbang antara efisiensi komputasi dan kualitas hasil pelatihan.

Implementasi model MobileNetV3-Large ke dalam layanan REST API berbasis Flask menunjukkan bahwa model yang dihasilkan tidak hanya unggul secara akurasi, tetapi juga cukup ringan untuk diintegrasikan ke dalam aplikasi web maupun sistem lain secara real-time. Hal ini sejalan dengan tujuan awal perancangan arsitektur MobileNetV3 yang memang ditujukan untuk perangkat dengan keterbatasan sumber daya komputasi (Howard et al., 2019). Kemampuan ini memberikan nilai tambah praktis bagi penelitian ini, mengingat kebutuhan akan sistem deteksi gambar buatan AI yang dapat diakses secara cepat dan mudah menjadi semakin mendesak di tengah maraknya penyebaran konten visual sintetis yang berpotensi disalahgunakan untuk tindak kejahatan siber maupun manipulasi informasi (Sianipar, 2026).

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa konfigurasi hyperparameter optimal untuk klasifikasi gambar asli dan gambar buatan AI menggunakan CNN pada penelitian ini adalah epoch sebanyak 20 dan batch size 32, dengan pembagian dataset 80% training, 10% validation, dan 10% testing pada dimensi citra 224×224 piksel serta learning rate 0,001. Dari ketiga model yang diuji, CNN dengan transfer learning MobileNetV3-Large menghasilkan performa terbaik dengan akurasi testing sebesar 96,71% dan loss 0,1031, mengungguli

EfficientNetV2-B0 (94,41%) dan CNN Scratch (91,58%). Keunggulan MobileNetV3-Large terutama disebabkan oleh mekanisme squeeze-and-excitation yang memungkinkan model memperkuat fitur informatif secara adaptif. Model terbaik tersebut telah berhasil diimplementasikan ke dalam REST API berbasis Flask sehingga dapat digunakan untuk klasifikasi gambar secara real-time. Penelitian ini memiliki keterbatasan berupa cakupan dataset yang berasal dari dua sumber publik dengan variasi domain gambar yang belum sepenuhnya representatif terhadap seluruh jenis gambar buatan AI yang beredar di internet, sehingga generalisasi model terhadap gambar sintetis dari model generatif terbaru masih perlu dievaluasi lebih lanjut. Untuk penelitian selanjutnya, disarankan untuk menambahkan variasi hyperparameter sebagai pembanding, memperbanyak dan mendiversifikasi dataset gambar asli maupun gambar buatan AI, serta menguji ketahanan model terhadap teknik kompresi maupun manipulasi gambar tambahan guna meningkatkan akurasi dan generalisasi model secara berkelanjutan.

5. REFERENSI

Ali, H. A., & Abdulazeez, A. M. (2024). Transfer Learning In Machine Learning: A Review Of Methods And Applications. *Indonesian Journal of Computer Science*.

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V., & Adam, H. (2019). *Searching for MobileNetV3*. http://arxiv.org/abs/1905.02244

Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., & Tang, P. T. P. (2017). *On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima*. http://arxiv.org/abs/1609.04836

Mu, J., Adrezo, M., & Haikal, A. N. (2024). Identifikasi Wajah Asli dan Buatan Deepfake Menggunakan Metode Convolutional Neural Network. *Teknika*, *13*(1), 45–50. https://doi.org/10.34148/teknika.v13i1.705

Nugroho, A. Y. (2025). *Deep Learning Algorithms in the Development of Generative AI Models for Automated Content Creation*. 111–122. https://doi.org/10.59059/mutiara.v3i4.2804

Plested, J., Phiri, M., & Gedeon, T. (2026). Deep transfer learning for image classification: a survey. *Artificial Intelligence Review*, *59*(3). https://doi.org/10.1007/s10462-026-11491-z

Rafliansyah, R. H., Rahmat, B., & Putra, C. A. (2024). Klasifikasi Suara Instrumen Musik Tiup Menggunakan Metode Convolutional Neural Network. *Merkurius : Jurnal Riset Sistem Informasi Dan Teknik Informatika*, *2*(4), 01–09. https://doi.org/10.61132/merkurius.v2i4.119

Roumpies, F., Spathoulas, G., & Kakarountas, A. (2025). Synthetically Generated Media, types, detection, and the threat they lurk for users, state and private actors. *Proceedings - 28th Pan-Hellenic Conference on Progress in Computing and Informatics with International Participation, PCI 2024*, 431–436. https://doi.org/10.1145/3716554.3716835

Sianipar, J. F. B. (2026). *Urgensi Pembaruan Hukum Pidana Indonesia Dalam Menghadapi Kejahatan Deepfake Artificial Intelligence Sebagai Ancaman Kejahatan Siber Generasi Baru*. https://doi.org/10.61104/jq.v4i3.6993

Simangunsong, J., Simanjuntak, N. D., & Matondang, A. A. (2025). Penerapan Transfer Learning untuk Klasifikasi Citra Bunga Berbasis Convolutional Neural Network. *Jurnal Minfo Polgan*, *14*(1), 1062–1067. https://doi.org/10.33395/jmp.v14i1.14980

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). *A Survey on Deep Transfer Learning*. http://arxiv.org/abs/1808.01974

Tan, M., & Le, Q. V. (2021). *EfficientNetV2: Smaller Models and Faster Training*. https://github.com/google/

Wardhana, S. F. D., & Nugroho, A. (2025). Perbandingan Arsitektur MobileNetV2 dan MobileNetV3 Dalam Klasifikasi Jenis Jeruk. *Jurnal Ilmu Komputer Dan*

Bisnis, *16*(1), 25–34. https://doi.org/10.47927/jikb.v16i1.916

Wijaya, A. C., Mahbubi, A. B., Januarta, M. F., Syabila, T., & Zakaria, D. (2025). *Analisis Sistematis Algoritma Convolutional Neural Network untuk Klasifikasi Gambar Bokeh dan Blur: Tinjauan Literatur*.

Wulandari, D., Iswara, I. B. A. I., Yudi Antara, I. G. M., Putra Asana, I. M. D., & Gandika Supartha, I. K. D. (2026). Balinese Statue Image Classification Using Transfer Learning: A Comparative Study of MobileNetV3 and EfficientNetV2. *Jurnal Teknik Informatika (Jutif)*, *7*(2), 1589–1608. https://doi.org/10.52436/1.jutif.2026.7.2.5556

Zhao, Z., Bakar, E. B. A., Razak, N. B. A., & Akhtar, M. N. (2025). Progressive optimization of EfficientNetV2 for classification based on images of corroded objects. *Computational and Applied Mathematics*, *44*(8). https://doi.org/10.1007/s40314-025-03382-w