

Analisis Performa CNN Berbasis MobileNetV2 pada Dataset Citra Dokumen Laporan Praktikum

Hafiz Aryanda^{1,*}, Lailan Sofinah Harahap², Dimas Aqila Aptanta³

^{1,2,3}) Universitas Islam Negeri Sumatera Utara, Jl. Lapangan Golf, Durin Jangak, Pancur Batu, Deli Serdang, Sumatera Utara

^{*}) Email corresponding author: hafizaryandaaa@gmail.com

Submitted: 21/12/2025

Accepted: 29/12/2025

Published: 31/12/2025

Abstrak

Penelitian ini bertujuan untuk menganalisis performa CNN berbasis MobileNetV2 dalam klasifikasi citra dokumen laporan praktikum. Pendekatan yang digunakan adalah transfer learning dengan bobot pra-latih dari ImageNet, serta fine-tuning pada lapisan atas untuk menyesuaikan model terhadap karakteristik dokumen akademik. Dataset terdiri dari ribuan citra dokumen yang telah melalui tahap pra-pemrosesan berupa resizing, normalisasi, dan augmentasi. Hasil eksperimen menunjukkan bahwa pada baseline, akurasi validasi mencapai 64,88% namun akurasi evaluasi manual hanya 10,54%, menandakan adanya masalah domain gap dan kesulitan separasi antar kelas. Fine-tuning meningkatkan akurasi validasi training menjadi 70,43%, tetapi akurasi evaluasi biner justru menurun menjadi 47,50%, sehingga performa model tetap rendah. Temuan ini menegaskan bahwa CNN kurang tepat untuk tugas validasi kepatuhan dokumen yang membutuhkan analisis semantik dan aturan tata letak yang kompleks serta dinamis. Sebagai solusi, integrasi OCR dengan rule-based validation direkomendasikan karena mampu memberikan akurasi lebih tinggi, interpretasi jelas, serta umpan balik detail sesuai pedoman penulisan akademik modern.

Kata Kunci: CNN, MobileNetV2, Pembelajaran Transfer, Klasifikasi Citra Dokumen, Validasi Berbasis Aturan dengan OCR

Abstract

This study aims to analyze the performance of a CNN based on MobileNetV2 in classifying images of practicum report documents. The approach employed is transfer learning with pre-trained weights from ImageNet, along with fine-tuning on the upper layers to adapt the model to the characteristics of academic documents. The dataset consists of thousands of document images that have undergone preprocessing steps such as resizing, normalization, and augmentation. Experimental results show that in the baseline, validation accuracy reached 64.88%, but manual evaluation accuracy was only 10.54%, indicating a domain gap problem and difficulty in class separation. Fine-tuning increased training validation accuracy to 70.43%, yet binary evaluation accuracy decreased to 47.50%, so the model performance remained low. These findings emphasize that CNNs are less suitable for compliance validation tasks requiring semantic analysis and complex, dynamic layout rules. As a solution, integrating OCR with rule-based validation is recommended, as it can provide higher accuracy, clearer interpretation, and more detailed feedback aligned with modern academic writing guidelines.

Keywords: CNN, MobileNetV2, Transfer Learning, Document Image Classification, OCR Rule-Based Validation

PENDAHULUAN

Dalam lingkungan akademik, laporan praktikum merupakan salah satu bentuk dokumen karya ilmiah yang wajib disusun oleh mahasiswa sebagai bagian dari kegiatan perkuliahan. Laporan ini tidak hanya berfungsi sebagai dokumentasi hasil percobaan, tetapi juga sebagai sarana untuk melatih keterampilan menulis ilmiah sesuai dengan pedoman kampus. Setelah

laporan selesai disusun, dokumen tersebut akan diperiksa oleh asisten laboratorium (aslab) sebagai salah satu bentuk evaluasi tugas mahasiswa. Proses pemeriksaan ini bertujuan untuk memastikan bahwa laporan telah mengikuti format dan tata penulisan yang ditetapkan, sehingga dapat diterima dan dinilai secara resmi. Namun proses pemeriksaan manual sering kali memakan waktu dan berpotensi menimbulkan kesalahan, sehingga diperlukan upaya pendekatan yang mampu melakukan analisis otomatis terhadap dokumen laporan praktikum agar evaluasi dapat berlangsung lebih cepat, konsisten, dan objektif.

Salah satu upaya pendekatan yang digunakan untuk analisis otomatis adalah pendekatan dengan komputasional. Dalam pendekatan ini dapat diupayakan dengan cara membuat dokumen tersebut menjadi citra-citra dokumen yang akan menjadi objek analisis dalam pendekatannya. Pendekatan ini bisa berupa pemodelan dari salah satu algoritma Jaringan Saraf Tiruan (JST) seperti *Convolutional Neural Network* (CNN). Dalam penelitian ini memiliki signifikansi dengan mengevaluasi performa CNN dan *transfer learning*. Performa CNN ini akan dianalisis dengan alasan apakah CNN mampu dalam memodelkan sistem otomatis untuk pemeriksaan citra dokumen. Karena seperti dokumen akademik juga memiliki karakteristik visual yang berbeda dengan citra umum seperti objek atau wajah.

Berbagai penelitian sebelumnya telah menunjukkan efektivitas CNN dan *transfer learning* dalam klasifikasi citra, terutama pada domain medis dan teknis. Seperti pada penelitian yang dilakukan Zaidan untuk membandingkan CNN dengan model *transfer learning* seperti MobileNetV2 dan VGG16 untuk klasifikasi kanker payudara, dan menemukan bahwa *transfer learning* dapat meningkatkan akurasi pada *dataset* terbatas (Zaidan et al., 2025). Lalu ada penelitian yang dilakukan Salehi untuk menegaskan bahwa CNN dan *transfer learning* mampu mengatasi keterbatasan data dalam analisis citra medis, meskipun tantangan seperti *overfitting* dan ketidak serasian dalam sistem masih menjadi masalah (Salehi et al., 2023). Sementara itu, dalam penelitian nurhaliza menerapkan metode OCR untuk pengenalan karakter dokumen administratif, dengan hasil akurasi tinggi, menunjukkan potensi pendekatan komputasional dalam otomatisasi analisis dokumen (Nurhaliza & ETP, 2022).

Sebagian besar penelitian sebelumnya memfokuskan pada penerapan CNN untuk citra medis maupun pengenalan karakter dokumen administratif. Kajian yang secara khusus menganalisis performa algoritma CNN pada citra dokumen, khususnya citra dokumen seperti laporan praktikum, masih sangat terbatas. Sehingga diperlukan evaluasi sistematis untuk mengetahui sejauh mana CNN mampu beradaptasi dalam kasus seperti ini. Penelitian ini hadir untuk mengisi celah tersebut dengan melakukan analisis performa CNN berbasis MobileNetV2 dan *transfer learning* pada *dataset* citra dokumen laporan praktikum, sehingga dapat memberikan gambaran mengenai potensi dan keterbatasan algoritma CNN dalam konteks dokumen pendidikan. Dengan demikian, penelitian ini dapat berkontribusi pada upaya memperluas pemahaman tentang penerapan CNN dalam konteks penelitian citra dokumen.

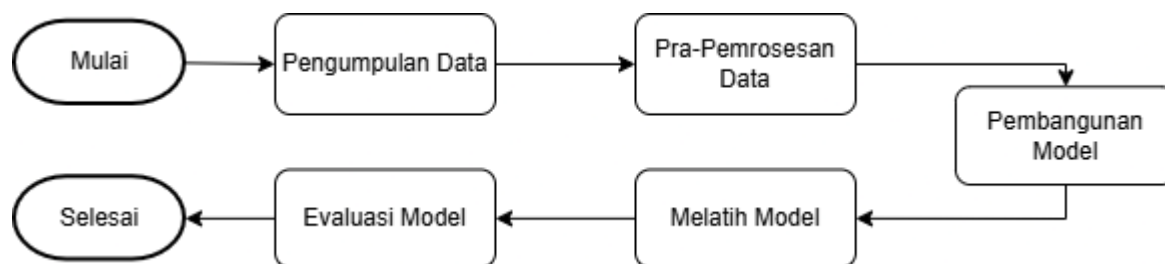
Seperti penelitian yang dilakukan oleh Zaidan adalah melakukan analisis komparatif antara CNN yang dibangun dari awal dengan model berbasis *transfer learning*. CNN adalah algoritma *deep learning* yang populer dalam melakukan pengolahan citra. Umumnya digunakan untuk melakukan pengenalan objek pada citra (Dacipta & Putra, 2022). *Deep learning* sendiri adalah subbidang pembelajaran mesin yang mensimulasikan proses berpikir manusia dengan memproses informasi dengan cara yang mirip dengan cara manusia berpikir (Sutriawan et al., 2023). CNN juga merupakan dari pengembangan dari JST, di mana CNN menambahkan lapisan konvolusi dan *pooling* untuk menangani data spasial seperti gambar, berbeda dengan JST klasik yang hanya menggunakan lapisan penuh (Krichen, 2023). JST sendiri adalah salah satu sistem pemrosesan informasi yang didesain dengan menirukan cara kerja otak manusia dalam menyelesaikan suatu masalah dengan melakukan proses belajar melalui perubahan bobot sinapsisnya (Panggabean & Sagala, 2021).

Dalam penelitian ini menggunakan salah satu metode yaitu *transfer learning* yang artinya adalah teknik pembelajaran mendalam dengan memanfaatkan model pra-latih untuk meningkatkan kinerja pada tugas baru yang terkait (Mufidatuzzainiya & Faisal, 2025). Konsep ini penting dalam penelitian berbasis CNN karena mampu mengatasi keterbatasan data dengan pengetahuan awal tentang fitur visual umum. Pada penelitian Zaidan, *transfer learning* diterapkan pada varian CNN MobileNetV2. MobileNetV2 sendiri adalah model CNN ringan yang mendukung *end-to-end deep learning* dengan efisiensi komputasi melalui *depthwise separable convolution*. Arsitektur ini menawarkan performa cepat dan efisien pada perangkat terbatas, serta banyak digunakan dalam penelitian dengan keterbatasan data karena mampu menyeimbangkan akurasi dan kecepatan (Hidayati et al., 2025).

Pada penelitian ini, citra digital digunakan sebagai media untuk menganalisis dokumen akademik yang dipindai atau difoto. Citra digital sendiri adalah citra yang dapat diolah komputer, biasanya berupa matriks piksel dengan nilai intensitas tertentu (Nasution et al., 2025). Penelitian Nurhaliza menunjukkan bahwa analisis citra dokumen dapat dilakukan secara otomatis, meskipun pendekatan berbasis CNN untuk evaluasi klasifikasi dokumen masih jarang dilakukan. Salah satu bentuk khusus dari citra digital adalah citra dokumen, yaitu gambar digital dari dokumen hasil scan, foto, atau rekaman video yang digunakan untuk berbagai tugas analisis (Arlazarov et al., 2022). Berbeda dengan citra objek umum, citra dokumen memiliki karakteristik visual lebih halus, seperti margin, tata letak, dan jenis *font*, sehingga menuntut metode analisis yang berbeda.

METODE PENELITIAN

Pendekatan penelitian yang digunakan dalam studi ini adalah pendekatan komputasional berbasis deep learning, khususnya JST CNN yang beroperasi pada data citra dokumen digital. Penelitian ini memanfaatkan teknik *transfer learning* dengan arsitektur MobileNetV2 sebagai *feature extractor* untuk melakukan klasifikasi citra dokumen laporan praktikum ke dalam kategori tertentu. Model CNN yang dibangun merupakan bagian dari JST yang bersifat supervised, di mana proses pelatihan dilakukan menggunakan *dataset* berlabel sehingga sistem mampu mempelajari pola visual yang relevan dengan kepatuhan format laporan. Berikut merupakan alur kerja dalam pembuatan model CNNnya:



Gambar 1. Alur Penelitian

1. Pengumpulan Data

Pada penelitian ini, data diperoleh dari kumpulan laporan praktikum mahasiswa yang digunakan sebagai objek analisis. Setiap laporan praktikum biasanya terdiri dari beberapa modul, di mana masing-masing modul memiliki antara 5 hingga 12 halaman. Laporan-laporan tersebut dikumpulkan dari berbagai sumber dan dijadikan data mentah untuk proses pelatihan dan pengujian model.

Tahapan pengumpulan data dilakukan dengan mengonversi setiap halaman laporan praktikum ke dalam format citra digital berupa file berformat JPG dan PNG. Format ini dipilih karena merupakan format standar citra raster yang umum digunakan dalam pengolahan citra

digital dan kompatibel dengan berbagai pustaka *deep learning* seperti TensorFlow dan Keras. Citra raster sendiri adalah gambar yang tersusun dari banyak titik kecil yang disebut piksel, seperti susunan kotak-kotak kecil yang membentuk sebuah gambar. Setiap kotak ini menyimpan informasi warna atau tingkat kecerahan dari bagian kecil di gambar tersebut (Irdiansyah 2025).

Setelah konversi dilakukan, setiap halaman laporan disimpan sebagai satu citra terpisah, sehingga satu modul laporan menghasilkan beberapa jenis pada bagian citra laporan praktikum. Setiap citra kemudian diberi label. Pelabelan ini dilakukan secara manual dengan mengacu pada pedoman penulisan laporan praktikum yang berlaku, seperti pengaturan margin, jarak antar paragraf, posisi gambar dan caption, serta tata letak teks. Dengan demikian, *dataset* yang dihasilkan terdiri dari ratusan citra dokumen terklasifikasi, yang nantinya digunakan untuk proses pelatihan model.

Tabel 1. Distribusi Dataset Citra Laporan Praktikum

| Jenis Bagian Laporan | Label | |
|-----------------------------|-------|-------|
| | Benar | Salah |
| Halaman Awal | 176 | 276 |
| Halaman Isi | 146 | 138 |
| Halaman Kesimpulan | 63 | 63 |
| Lembar Pengesahan (LP) | 63 | 62 |
| Total Jumlah Seluruh Citra: | 987 | |

2. Pra-Pemrosesan Data

Setelah pengumpulan data selesai, sebaiknya dilakukan prapemrosesan data agar data dapat dipersiapkan dengan lengkap dan formatnya sesuai dengan yang dibutuhkan untuk pemodelan (Abdiansah et al., 2025). Langkah pertama dalam pemrosesan untuk pembuatan model ini adalah *resizing* gambar. *Resizing* digunakan untuk mengubah ukuran gambar menjadi seragam dan konsisten, sehingga data *training*, validasi, dan *testing* menggunakan ukuran target yang sama (Nurlatifa et al., 2024). Pada *dataset* awal, setiap halaman laporan praktikum memiliki dimensi yang bervariasi tergantung pada perangkat pemindaian atau metode konversi yang digunakan. Variasi ukuran ini dapat menyebabkan ketidaksesuaian bentuk masukan pada arsitektur CNN. Contohnya, salah satu gambar pada *dataset* awal memiliki ukuran (1755, 1241, 3), dan setelah proses *resize* menjadi (224, 224, 3). Proses ini dilakukan pada semua gambar dalam *dataset* agar ukuran gambar menjadi konsisten.

Langkah kedua pada tahap preprocessing dalam model ini adalah melakukan normalisasi, yang maksudnya adalah mengubah nilai piksel gambar menjadi rentang antara 0 dan 1. Normalisasi penting dilakukan dalam CNN karena membantu model belajar lebih cepat dan stabil dengan menyamakan skala nilai input sehingga mengurangi ketidakseimbangan nilai yang dapat mempengaruhi proses pelatihan (Maulana et al., 2023). Contohnya salah satu gambar pada *dataset* awal memiliki nilai piksel (0 255) dan setelah proses normalisasi menjadi (0 0 1 0). Proses ini dilakukan pada semua gambar dalam *dataset* juga agar nilai piksel pada semua gambar menjadi konsisten.

Langkah ketiga adalah melakukan proses augmentasi, yaitu teknik memperbanyak dan memperkaya variasi data gambar dengan melakukan transformasi tertentu pada gambar asli (Rhamadiyanti & Kusriani, 2024). Pada model ini, digunakan augmentasi *offline*, di mana proses augmentasi menghasilkan file gambar baru yang disimpan dan dapat diakses secara terpisah, sehingga *dataset* menjadi lebih besar dan bervariasi saat pelatihan model. Dapat dilihat pada

tabel dibawah bahwasanya hasil augmentasi menunjukkan peningkatan nyata pada jumlah data, yaitu dari 987 citra awal menjadi 1797 citra setelah proses augmentasi selesai, yang berarti augmentasi secara efektif memperkaya *dataset*.

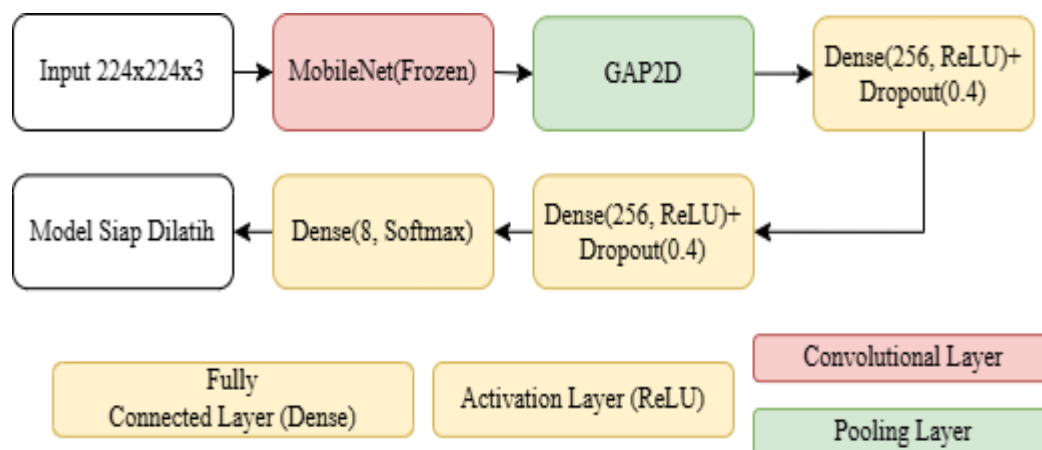
Tabel 2. Distribusi Dataset Citra Laporan Praktikum Setelah Preprocessing

| Jenis Bagian Laporan | Label | |
|-----------------------------|-------|-------|
| | Benar | Salah |
| Halaman Awal | 340 | 337 |
| Halaman Isi | 338 | 34 |
| Halaman Kesimpulan | 340 | 339 |
| Lembar Pengesahan (LP) | 340 | 340 |
| Total Jumlah Seluruh Citra: | 2714 | |

Langkah keempat pada proses ini adalah pembagian data. Pembagian data adalah proses memisahkan dataset menjadi beberapa bagian. Tujuannya adalah sebagai pelatihan dan pengujian model. Hal ini penting dilakukan agar model dapat dilatih pada sebagian data dan diuji pada data lain yang belum pernah dilihat sebelumnya, sehingga kinerja model dapat dievaluasi secara objektif (Ery, 2024). Dalam model ini, pembagian data dilakukan dengan rasio 8:2, artinya 80% data digunakan untuk pelatihan dan 20% sisanya digunakan untuk pengujian, guna memastikan model belajar dari data yang cukup dan diuji pada data yang representatif.

3. Perancangan Arsitektur Model

Pada tahap perancangan arsitektur, penelitian ini melakukan perancangan arsitektur model CNN yang digunakan untuk melakukan klasifikasi kepatuhan format pada citra halaman laporan praktikum. CNN dipilih karena kemampuannya dalam mengekstraksi fitur visual secara otomatis melalui proses konvolusi dan *pooling*, sehingga sangat efektif dalam mengenali pola tata letak dokumen seperti margin, posisi judul, jarak paragraf, dan struktur penulisan lainnya. Arsitektur CNN yang digunakan terdiri dari beberapa lapisan inti, yaitu *Convolutional Layer*, *Activation Layer* (ReLU), *Pooling Layer*, dan diakhiri dengan *Fully Connected Layer* sebagai pengambil keputusan (Dartiko et al., 2024).



Gambar 2. Arsitektur CNN

Pada penelitian ini, arsitektur CNN dirancang dengan memanfaatkan model MobileNetV2 sebagai *feature extractor* untuk menjaga keseimbangan antara kedalaman jaringan, akurasi, dan efisiensi komputasi. Pendekatan ini memungkinkan pemrosesan citra dokumen berukuran 224×224 piksel secara optimal tanpa menimbulkan *overfitting*, karena bobot pada lapisan dasar MobileNetV2 dibekukan (*frozen*) selama pelatihan.

Pada tahap awal arsitektur MobileNetV2, lapisan konvolusi (*Convolutional Layer*) berfungsi mengekstraksi fitur dasar dari citra dokumen. Setiap lapisan konvolusi dilengkapi dengan fungsi aktivasi ReLU6 untuk menambahkan sifat non-linear, sehingga jaringan dapat mempelajari pola yang lebih kompleks dan tidak terbatas pada hubungan linear saja. Proses penyederhanaan dimensi fitur dilakukan dengan memanfaatkan *strided convolution*, yang bekerja mirip dengan *pooling layer* dalam memperkecil ukuran fitur sambil tetap mempertahankan informasi penting. Pada tahap ekstraksi fitur, digunakan *Global Average Pooling* (GAP) untuk meratakan seluruh nilai fitur menjadi satu vektor representasi. Teknik ini efektif dalam mengurangi jumlah parameter model sekaligus menekan risiko *overfitting* (Saputra et al., 2023).

Setelah melalui seluruh tahapan ekstraksi fitur tersebut, keluaran jaringan konvolusi diteruskan ke bagian klasifikasi yang terdiri dari beberapa *fully connected layer* (*Dense Layer*). Lapisan ini menggunakan fungsi aktivasi ReLU untuk memperkuat kemampuan jaringan dalam mempelajari hubungan antar fitur, serta lapisan *softmax* di bagian *output* untuk menghasilkan probabilitas klasifikasi. Sistem kemudian mengelompokkan citra dokumen ke dalam dua kategori, yaitu format “benar” atau “salah”, berdasarkan parameter-parameter visual yang telah dipelajari selama proses pelatihan model. Arsitektur CNN ini dirancang menggunakan *framework* Keras dengan tipe model *Sequential*, yaitu model yang tersusun secara berurutan dari *layer* pertama hingga *layer* terakhir (Al-hafizh & Alamsyah, 2024). Model *Sequential* dipilih karena sangat sesuai untuk penelitian ini, mengingat struktur alurnya sederhana, linier, dan tidak memiliki percabangan sehingga mempermudah proses pembangunan, pelatihan, serta interpretasi jaringan. Struktur model CNN dapat dilihat pada tabel arsitektur dibawah ini.

Proses perancangan arsitektur ini menghasilkan model dengan kompleksitas yang tetap efisien. Total parameter yang terbentuk sebanyak 2.619.848 parameter, dengan 361.864 parameter *trainable* yang berada pada lapisan klasifikasi (*Dense Layer*), sedangkan 2.257.984 parameter non-*trainable* berasal dari lapisan MobileNetV2 yang dibekukan. Pengaturan ini memastikan bahwa proses pelatihan berlangsung lebih cepat, stabil, serta mampu mengurangi risiko *overfitting* karena sebagian besar parameter inti dari model pretrained tidak diubah selama pelatihan.

4. Latih Model

Setelah arsitektur CNN selesai dirancang, tahap berikutnya adalah melakukan proses pelatihan untuk mengajarkan model mengenali pola visual pada citra halaman laporan praktikum. Pelatihan dilakukan dengan menggunakan sekumpulan data citra yang telah melalui tahap pra-pemrosesan sebelumnya, yang data tersebut juga telah dibagi menjadi dua bagian yaitu data latih dan data validasi. Hasil training juga divisualisasikan melalui grafik perkembangan berikut:

Model dilatih menggunakan arsitektur MobileNetV2 dengan bobot pra-latih dari ImageNet sebagai *feature extractor*. Pada tahap awal, seluruh lapisan dasar dibekukan agar model hanya menyesuaikan bagian klasifikasi terhadap pola visual dokumen. Proses pelatihan dilakukan dengan *optimizer Adam* (*learning rate* $1e-4$), *loss function categorical crossentropy*, *batch size* 32, dan 25 epochs.

Selanjutnya dilakukan *fine-tuning* dengan membuka 30 lapisan terakhir MobileNetV2. Pada tahap ini digunakan *learning rate* lebih kecil ($1e-5$) agar penyesuaian bobot berlangsung

lebih halus. Model dilatih selama 20 epochs dengan total parameter sekitar 2,6 juta, di mana sebagian besar tetap dibekukan untuk menjaga efisiensi dan mengurangi risiko *overfitting*.

5. Evaluasi Model

Setelah model CNN selesai melalui proses pelatihan, tahap berikutnya adalah melakukan model evaluasi untuk menilai kemampuan model dalam mengenali citra halaman laporan praktikum yang tidak digunakan selama proses training. Beberapa metrik yang digunakan dalam penelitian ini antara lain accuracy, precision, recall, F1-score, dan confusion matrix. *Accuracy* menunjukkan persentase prediksi yang benar dari seluruh sampel, sedangkan *precision* mengukur ketepatan model dalam memprediksi suatu kelas tertentu. *Recall* digunakan untuk menilai kemampuan model mendeteksi seluruh sampel pada kelas tertentu, khususnya penting pada kasus ketidakseimbangan kelas. F1-score merupakan rata-rata harmonis antara *precision* dan *recall*, sehingga memberikan gambaran kinerja model secara lebih seimbang (Septiani et al., 2025).

Selain itu, ada sebuah *confusion matrix* (CM) yang digunakan untuk melihat performa model secara lebih detail per kelas, termasuk kelas mana yang paling sering salah diklasifikasikan dan pola kesalahan yang muncul. Dan diberlakukannya juga *Binary Accuracy* dengan cara mengonversi hasil klasifikasi ke dalam dua kategori sederhana, yaitu Benar dan Salah. Untuk memberikan gambaran lebih detail, metrik Precision, Recall, dan F1-Score digunakan dalam konteks klasifikasi biner, sehingga dapat menunjukkan keseimbangan antara ketepatan prediksi dan kemampuan model dalam mengenali kelas positif maupun negatif.

Untuk menilai performa model, digunakan beberapa metrik evaluasi untuk analisis yang lebih mendalam. Pertama, dilakukan evaluasi per jenis halaman untuk mengetahui performa model pada setiap kategori dokumen secara spesifik. Kedua, menggunakan evaluasi dengan CM untuk *Binary Accuracy*. Terakhir, menggunakan evaluasi *classification report* yang memuat metrik evaluasi per kelas, sehingga memberikan gambaran komprehensif mengenai distribusi performa model pada seluruh kategori dokumen.

HASIL DAN PEMBAHASAN

1. Analisis Performa model MobileNetV2

Tabel 3. Analisis Hasil Latihan Pada Model MobileNet V2

| Metriks | Nilai |
|------------------------------|-------------------------|
| Total Jumlah Sampel Latihan | 2173 Sampel |
| Total Jumlah Sampel Validasi | 541 Sampel |
| Akurasi Latihan Terakhir | 60.02% |
| Akurasi Latihan Terbaik | 64.88% pada epoch ke 15 |
| Langkah pada setiap epoch | 68 Langkah |

Pelatihan model dilakukan selama 25 epoch dengan konfigurasi *transfer learning* menggunakan MobileNetV2 sebagai *feature extractor*. Dari hasil tersebut, dapat diamati bahwa model tidak mengalami *overfitting*, ditunjukkan oleh selisih akurasi antara data latih dan validasi yang relatif kecil (sekitar 5%). Namun, akurasi validasi cenderung macet setelah epoch ke-15, menandakan bahwa model mencapai titik konvergensi pada tahap tersebut. Dari hasil latihan tersebut dapat dilihat hasil evaluasi model sebagai berikut:

Tabel 4. Laporan Klasifikasi Evaluasi Model MobileNetV2

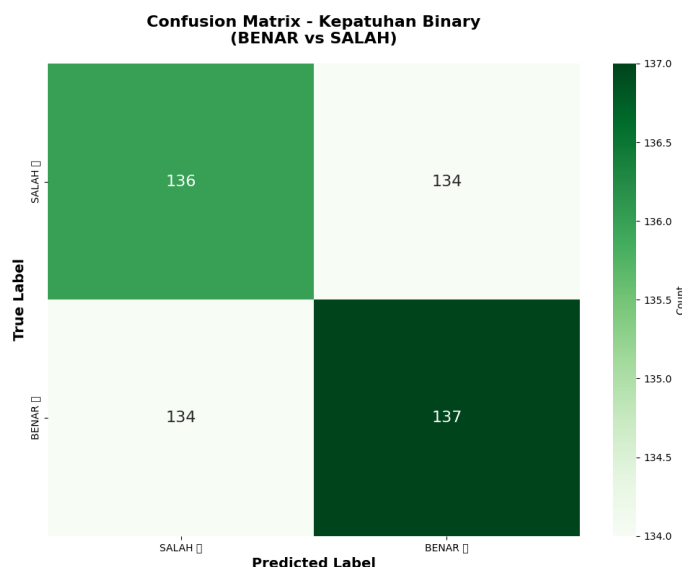
| Metriks | Precision | Recall | F1-Score |
|--------------------|--------------|--------|----------|
| Halaman Awal_Benar | 0.0517 | 0.0441 | 0.0476 |
| Halaman Awal_Salah | 0.1000 | 0.1194 | 0.1088 |
| Halaman Isi_Benar | 0.1343 | 0.1343 | 0.1343 |
| Halaman Isi_Salah | 0.1695 | 0.1471 | 0.1575 |
| Halaman | 0.0794 | 0.0735 | 0.0763 |
| Kesimpulan_Benar | | | |
| Halaman | 0.0641 | 0.0746 | 0.0690 |
| Kesimpulan_Salah | | | |
| Lembar | 0.1325 | 0.1618 | 0.1457 |
| Pengesahan_Benar | | | |
| Lembar | 0.1132 | 0.0882 | 0.0992 |
| Pengesahan_Salah | | | |
| Accuracy | 0.1054 = 10% | | |

Pada tahap pelatihan, model menunjukkan *validation accuracy* sebesar 64,88%. Namun, ketika dilakukan evaluasi manual terhadap data uji, akurasi validasi hanya mencapai 10,54%. Perbedaan yang sangat besar, yaitu selisih 54,34%, menjadi indikasi adanya masalah serius pada proses data generator atau ketidaksesuaian label yang digunakan dalam pelatihan. Hal ini menunjukkan bahwa meskipun model terlihat cukup baik saat *training*, performa sebenarnya pada data evaluasi tidak konsisten.

Lalu untuk mengetahui evaluasi terhadap model klasifikasi target biner terhadap kepatuhan dokumen, dengan dua kategori utama: “BENAR” dan “SALAH”. Dapat dilihat sebagai tabel dan gambar CM di bawah, hasil pengujian menunjukkan bahwa performa model masih berada pada tingkat yang relatif rendah. CM yang ditampilkan memperlihatkan distribusi kesalahan yang cukup merata antara kedua kelas, sehingga model cenderung kesulitan dalam mengenali pola kepatuhan dokumen secara konsisten. Evaluasi ini menegaskan bahwa meskipun arsitektur CNN berbasis MobileNetV2 mampu melakukan klasifikasi, namun performanya pada domain dokumen laporan praktikum masih terbatas.

Tabel 5. Laporan Klasifikasi Pada Kelas *Binary*

| Metriks | Skor |
|-----------|--------|
| Accuracy | 50.46% |
| Precision | 50.55% |
| Recall | 50.55% |
| F1-Score | 50.55% |



Gambar 3. CM Kelas Binary

2. Analisis Peforma model MobileNetV2 dengan Fine-Tuning

Tabel 6. Analisis Hasil Latihan Pada Model MobileNet V2 dengan *Fine-Tuning*

| Metriks | Nilai |
|------------------------------|-------------------------|
| Total Jumlah Sampel Latihan | 2173 Sampel |
| Total Jumlah Sampel Validasi | 541 Sampel |
| Akurasi Latihan Terakhir | 70.15% |
| Akurasi Latihan Terbaik | 70.43% pada epoch ke 16 |
| Langkah pada setiap epoch | 68 Langkah |

Pada pelatihan awal, model hanya menggunakan pengetahuan umum dari *ImageNet* tanpa banyak penyesuaian, sehingga akurasi validasi berhenti di sekitar 64,88%. Setelah dilakukan *fine-tuning*, yaitu membuka sebagian lapisan atas dari MobileNetV2 dan melatih ulang dengan kecepatan belajar yang lebih kecil, model bisa lebih menyesuaikan diri dengan pola khusus yang ada pada citra dokumen laporan praktikum. Hasilnya, akurasi validasi meningkat menjadi 70,43%, atau naik sekitar 5,55% dibanding sebelumnya. Artinya, dengan memberi kesempatan model untuk “belajar ulang” pada lapisan tertentu, ia menjadi lebih peka terhadap karakteristik dokumen, bukan hanya mengandalkan pengetahuan umum dari gambar-gambar di *ImageNet*. Dari hasil latihan tersebut dapat dilihat hasil evaluasi model sebagai berikut:

Tabel 7. Laporan Klasifikasi Evaluasi Model MobileNetV2 Dengan *Fine-Tuning*

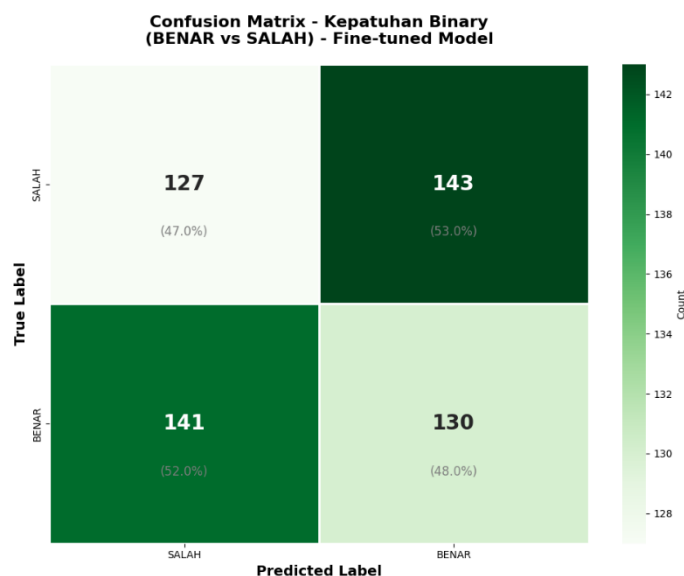
| Metriks | Precision | Recall | F1-Score |
|--------------------|-----------|--------|----------|
| Halaman Awal_Benar | 0.0882 | 0.0441 | 0.0588 |
| Halaman Awal_Salah | 0.1600 | 0.2388 | 0.1916 |
| Halaman Isi_Benar | 0.1687 | 0.2090 | 0.1867 |
| Halaman Isi_Salah | 0.0784 | 0.0588 | 0.0672 |

| | | | |
|-----------------------------|-------------|--------|--------|
| Halaman Kesimpulan_Benar | 0.0800 | 0.0882 | 0.0839 |
| Halaman Kesimpulan_Salah | 0.1452 | 0.1343 | 0.1395 |
| Lembar Pengesahan_Benar | 0.0864 | 0.1029 | 0.0940 |
| Lembar Pengesahan_Salah | 0.1636 | 0.1324 | 0.1463 |
| Accuracy | 0.1257= 12% | | |

Jika dibandingkan dengan model tanpa *fine-tuning* (*baseline*), perbedaan utama terletak pada konsistensi hasil. Pada *baseline*, terdapat gap yang sangat besar antara akurasi validasi saat *training* (64,88%) dan akurasi evaluasi manual (10,54%), yang mengindikasikan adanya masalah serius pada proses data generator atau pelabelan. Sedangkan pada *fine-tuning*, meskipun akurasi validasi meningkat hingga 70,43%, hasil evaluasi manual tetap menunjukkan akurasi rendah (12%) dengan distribusi kesalahan yang tinggi di semua kelas. Dengan demikian, *fine-tuning* memang memberikan perbaikan pada tahap pelatihan, tetapi belum sepenuhnya menyelesaikan masalah mendasar yang sudah terlihat pada *baseline*.

Tabel 8. Laporan Klasifikasi Pada Kelas *Binary* Setelah *Fine-Tuning*

| Metriks | Skor |
|-----------|--------|
| Accuracy | 47.50% |
| Precision | 47.62% |
| Recall | 47.97% |
| F1-Score | 47.79% |



Gambar 4. CM Kelas *Binary* Setelah *Fine-Tuning*

Evaluasi klasifikasi biner setelah *fine-tuning* menunjukkan bahwa performa model justru menurun. Akurasi hanya mencapai 47,50%, lebih rendah dibanding *baseline* sebelumnya yang

mencapai 50,46%. Meskipun akurasi validasi saat *training* meningkat, hasil evaluasi manual menunjukkan bahwa model masih kesulitan membedakan dokumen “BENAR” dan “SALAH” secara konsisten. Ini menandakan bahwa *fine-tuning* belum berhasil memperbaiki kinerja model pada data uji yang sebenarnya

3. Penyebab Performa Rendah

Hasil penelitian menunjukkan bahwa akurasi rendah pada model *baseline* maupun *fine-tuning* disebabkan oleh beberapa faktor utama. Pada model *baseline* dengan akurasi evaluasi manual hanya 10,54%, terdapat tiga masalah kritis. Yaitu:

1. Terjadi domain gap antara data pra-latih ImageNet dan citra dokumen. CNN yang dilatih pada gambar natural seperti anjing atau mobil cenderung mengekstraksi fitur berupa tepi, tekstur, dan warna, sementara dokumen membutuhkan fitur berbeda seperti margin, spasi, dan tata letak.
2. Terdapat masalah *low separability* karena kelas “Benar” dan “Salah” terlalu mirip secara visual. Analisis statistik menunjukkan bahwa 75% kategori halaman tidak memiliki perbedaan signifikan, sehingga sulit dipisahkan oleh model.

Tabel 9. Statistik Skor Kemiripan Antar Kelas

| Jenis Halaman | P-Value | Kesimpulan |
|--------------------|---------|---------------|
| Halaman Awal | 0.023 | Berbeda |
| Halaman Isi | 0.267 | Tidak berbeda |
| Halaman Kesimpulan | 0.693 | Tidak berbeda |
| Lembar Pengesahan | 0.786 | Tidak berbeda |

3. Terdapat *task mismatch* antara kemampuan CNN yang berfokus pada tampilan visual dengan kebutuhan validasi dokumen yang bersifat semantik dan berbasis aturan, sehingga model hanya menilai “terlihat seperti X” alih-alih “sesuai aturan Y”.

Pada tahap *fine-tuning*, akurasi biner justru menurun menjadi 47,50%, lebih rendah dibanding *baseline*. Tiga penyebab utama penurunan ini adalah:

1. *Atastrophic forgetting*, di mana fitur pra-latih dari *ImageNet* yang sebelumnya stabil menjadi rusak karena penyesuaian terhadap data dokumen yang terbatas dan *noisy*, sehingga model kehilangan pengetahuan awal namun belum sepenuhnya belajar fitur dokumen.
2. Terjadi *overfitting* pada *noise*, di mana model lebih banyak menangkap pola *spurious* seperti “dokumen blur = salah” daripada pola bermakna seperti “margin tidak sesuai = salah”. Hal ini terlihat dari meningkatnya akurasi training hingga 70% tetapi akurasi validasi turun ke 47,5%.

Dengan demikian, baik *baseline* maupun *fine-tuning* sama-sama menunjukkan keterbatasan serius, yang menegaskan bahwa masalah utama bukan hanya pada arsitektur model, tetapi juga pada kualitas data, kesamaan antar kelas, dan ketidaksesuaian antara tugas visual dengan kebutuhan semantik dokumen.

4. Solusi

Solusi yang dapat diterapkan untuk mengatasi rendahnya akurasi model adalah pendekatan OCR + *Rule-Based Validation*. Dengan metode ini, teks dan tata letak dokumen diekstrak menggunakan OCR, lalu diperiksa melalui aturan sederhana seperti margin, spasi, dan ukuran *font*. Sistem tidak hanya memberi status “BENAR” atau “SALAH”, tetapi juga menjelaskan

alasan kesalahan secara detail. Keunggulannya, akurasi bisa mencapai 85–95% jika OCR berjalan baik, hasil evaluasi lebih mudah dipahami, tidak memerlukan pelatihan ulang, serta aturan dapat diperbarui sesuai kebutuhan. Pendekatan ini lebih praktis dan relevan dibanding CNN karena langsung menilai dokumen berdasarkan aturan penulisan yang jelas.

KESIMPULAN

Penggunaan MobileNetV2 dengan *transfer learning* dan *fine-tuning* belum mampu memberikan hasil optimal dalam klasifikasi dokumen. *Baseline* menunjukkan akurasi validasi manual hanya 10,54% akibat domain gap antara ImageNet dan dokumen, rendahnya keterpisahan antar kelas, serta ketidaksesuaian tugas visual CNN dengan kebutuhan validasi semantik. *Fine-tuning* memang meningkatkan akurasi validasi *training* dari 64,88% menjadi 70,43%, tetapi akurasi evaluasi biner justru turun menjadi 47,50%, dipengaruhi oleh *catastrophic forgetting*, *overfitting* terhadap *noise*, dan keterbatasan jumlah data dibanding kompleksitas parameter. Pendekatan berbasis CNN terbukti kurang tepat untuk tugas validasi kepatuhan dokumen, sehingga solusi alternatif yang lebih efektif adalah integrasi OCR dengan *rule-based validation* yang mampu menilai teks dan tata letak secara langsung, memberikan akurasi lebih tinggi, interpretasi jelas, serta umpan balik detail sesuai aturan penulisan.

DAFTAR PUSTAKA

- Abdiansah, L., Sumarno, Eviyanti, A., & Azizah, N. L. (2025). Implementation of Convolutional Neural Networks Algorithm for Javanese Handwriting Recognition. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 5(April), 496–504. <https://doi.org/https://doi.org/10.57152/malcom.v5i2.1814>
- Al-hafizh, F., & Alamsyah. (2024). Indonesian Journal of Mathematics and Natural Sciences. *Indonesian Journal of Mathematics and Natural Sciences*, 47(1), 1–11. <https://doi.org/https://doi.org/10.15294/4x9r3p15>
- Arlazarov, V. V., Andreeva, E. I., Bulatov, K. B., Nikolaev, D. P., Petrova, O. O., Savelev, B. I., & Slavin, O. A. (2022). Document image analysis and recognition: a survey. *Computer Optics*, 46(4), 567–589. <https://doi.org/10.18287/2412-6179-CO-1020>
- Dacipta, P. N., & Putra, R. E. (2022). Sistem Klasifikasi Limbah Menggunakan Metode Convolutional Neural Network (CNN) Pada Webservice Berbasis Framework Flask. *Journal of Informatics and Computer Science (JINACS)*, 3(04), 394–402. <https://doi.org/10.26740/jinacs.v3n04.p394-402>
- Dartiko, F., Pradana, R. J., Sari, R. E., Syahputra, W., & Oktoeberza, W. K. Z. (2024). Klasifikasi Kanker Kulit Berbasis CNN dengan Metode Hybrid Preprocessing. *Jurnal Teknik Elektromedik Indonesia*, 5(2), 124–132. <https://doi.org/https://doi.org/10.18196/mt.v5i2.22675>
- Ery, Y. (2024). *Train Validate Test dan Cross Validation*. Binus University. <https://sis.binus.ac.id/2024/11/01/train-validate-test-dan-cross-validation/>
- Hidayati, U., Hardyan, F. W., Auliawati, F. R., Rafsanjani, A. A., Reza, F. A., & Siregar, M. M. (2025). Model coralnet, inceptionv3, dan mobilenetv2 untuk klasifikasi kondisi terumbu karang. *JITET (Jurnal Informatika Dan Teknik Elektro Terapan)*, 13(3), 1773–1783. <https://doi.org/http://dx.doi.org/10.23960/jitet.v13i3.6591>
- IRDIANSYAH, E. N. (2025). *Analisis ketelitian geometrik citra pleiades 1b dasar rencana*

- detail tata ruang (studi kasus : kota bandar lampung) lembar pengesahan analisis ketelitian geometrik citra pleiades 1b (studi kasus : kota bandar lampung) [Universitas Winaya Mukti]. <https://doi.org/https://repo.unwim.ac.id/id/eprint/1076>
- Krichen, M. (2023). Convolutional Neural Networks: A Survey. *MDPI: Computers*, 12(8), 1–41. <https://doi.org/10.3390/computers12080151>
- Maulana, S. A., Batubara, S. H., Amelia, T. A., & Pasaribu, Y. P. P. (2023). Penerapan Metode CNN (Convolutional Neural Network) Dalam Mengklasifikasi Jenis Ubur-Ubur. *Jurnal Penelitian Rumpun Ilmu Teknik (JUPRIT)*, 2(4), 122–130. <https://doi.org/https://doi.org/10.55606/juprit.v2i4.3084>
- Mufidatuzzainiya, A., & Faisal, M. (2025). Penggunaan Teknik Transfer Learning pada Metode CNN untuk Pengenalan Tanaman Bunga. *JISKA (Jurnal Informatika Sunan Kalijaga)*, 10(2), 195–206. <https://doi.org/10.14421/jiska.2025.10.2.195-206>
- Nasution, M. U., Harahap, L. S., & Syakbani, F. (2025). Tinjauan Metode Pengolahan Citra Digital untuk Deteksi Objek Otomatis. *Jurnal Informatika Dan Teknik Elektro Terapan*, 13(3), 1020–1028. <https://doi.org/http://dx.doi.org/10.23960/jitet.v13i3.7135>
- Nurhaliza, S. S., & ETP, L. (2022). Sistem Pengenalan Karakter Dokumen Secara Otomatis Menggunakan Metode Optical Character Recognition. *Petir: Jurnal Pengkajian Dan Penerapan Teknik Informatika*, 15(1), 166–175. <https://doi.org/10.33322/petir.v15i1.1610>
- Nurlatifa, Nurhaeni, Hidayat, A., & Prasetya, M. R. A. (2024). Metode Convolutional Neural Network (CNN) Untuk Klasifikasi Tingkat Kesehatan Tanaman Lidah Buaya Berbasis Web. *Jurnal Teknik Informatika Dan Sistem Informasi*, 11(4), 392–406. <https://doi.org/https://doi.org/10.35957/jatisi.v11i4.9003>
- Panggabean, E., & Sagala, J. R. (2021). Analisa Perbandingan Metode Jaringan Syaraf Tiruan Dengan Metode Sistem Pendukung Keputusan Untuk Penerimaan Tenaga Kerja. *Jurnal Media Informatika*, 2(2), 74–80. <https://doi.org/10.55338/jumin.v2i2.697>
- Rhamadiyanti, D. T., & Kusriani. (2024). Analisa Performa Convolutional Neural Network dalam Klasifikasi Citra Apel dengan Data Augmentasi. *KLIK: Kajian Ilmiah Informatika Dan Komputer*, 5(1), 154–162. <https://doi.org/10.30865/klik.v5i1.2023>
- Salehi, A. W., Khan, S., Gupta, G., Alabdullah, B. I., Almjally, A., Alsolai, H., Siddiqui, T., & Mellit, A. (2023). A Study of CNN and Transfer Learning in Medical Imaging: Advantages, Challenges, Future Scope. *MDPI: Sustainability*, 15(7), 1–28. <https://doi.org/10.3390/su15075930>
- Saputra, R. A., Rasendriya, D. R. P., & Asyrofi, M. A. (2023). Implementasi Neural Network (Cnn) Untuk Mendeteksi Penggunaan Masker Pada Gambar. *JITET (Jurnal Informatika Dan Teknik Elektro Terapan)*, 11(3), 710–714. <https://doi.org/https://doi.org/10.23960/jitet.v11i3.3286>
- Septiani, D., Fadhli, M., & Soim, S. (2025). Cnn Algorithm Optimization For Classifying Numbers In Handwriting. *TRANSMISI: Jurnal Ilmiah Teknik Elektro*, 27(1), 57–63. <https://doi.org/10.14710/transmisi.27.1.57-63>
- Sutriawan, S., Fanani, A. Z., Alzami, F., & Basuki, R. S. (2023). Deep Learning Jaringan Saraf Tiruan Untuk Pemecahan Masalah Deteksi Penyakit Daun Apel. *Jurnal Teknologi*

Informasi Dan Komunikasi (TIKomSiN), 11(1), 35–42.
<https://doi.org/10.30646/tikomsin.v11i1.729>

Zaidan, N., Seka, E. M., Axellee, R., Syahputra, M. H., Chrisnawati, G., & Hendrian, Y. (2025). Analisis Komparatif CNN dan Transfer Learning untuk Prediksi Dini Kanker Payudara Berbasis Deep Learning. *RIGGS: Journal of Artificial Intelligence and Digital Business*, 4(2), 5947–5955. <https://doi.org/10.31004/riggs.v4i2.1497>