

Implementasi Model EfficientNetB0 Berbasis *Deep learning* untuk Klasifikasi Penyakit Mata pada Citra Fundus

Esa Maidatussohiba¹, Yuni Yamasari²

^{1,2} Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya

¹ esa.22131@mhs.unesa.ac.id

² yuniyamasari@unesa.ac.id

Abstrak— Gangguan penglihatan merupakan permasalahan kesehatan global yang terus meningkat dan sering kali disebabkan oleh penyakit mata seperti katarak, glaukoma, dan retinopati diabetik. Keterbatasan tenaga medis serta fasilitas diagnostik menyebabkan banyak kasus penyakit mata terdeteksi pada tahap lanjut. Oleh karena itu, pemanfaatan kecerdasan buatan berbasis citra fundus retina menjadi salah satu pendekatan yang potensial untuk mendukung deteksi dini penyakit mata secara otomatis. Penelitian ini mengimplementasikan model *deep learning* berbasis *Convolutional Neural Network* (CNN) menggunakan arsitektur EfficientNetB0 untuk melakukan klasifikasi penyakit mata pada citra fundus retina.

Dataset yang digunakan adalah *Eye Disease Dataset* dari Kaggle yang terdiri dari empat kelas, yaitu katarak, retinopati diabetik, glaukoma, dan normal. Penelitian ini mengevaluasi pengaruh rasio pembagian dataset terhadap performa model dengan tiga skenario pembagian data pelatihan dan validasi, yaitu 95%:5%, 90%:10%, dan 85%:15%. Tahapan penelitian meliputi *preprocessing* citra, penerapan *data augmentation*, serta optimasi *pipeline* data menggunakan *caching*, *prefetching*, dan *shuffle*. Evaluasi model dilakukan menggunakan metrik *accuracy*, *F1-Score*, *precision*, *recall*, dan waktu komputasi. Hasil eksperimen menunjukkan bahwa rasio pembagian dataset 95%:5% memberikan performa terbaik dengan akurasi 96,19% dan waktu pelatihan 8 menit 27 detik. Hasil penelitian menunjukkan bahwa arsitektur EfficientNetB0 mampu mengekstraksi fitur citra fundus secara efektif dan efisien serta memberikan performa klasifikasi yang baik terutama pada pembagian data dengan jumlah data *training* lebih besar.

Kata Kunci— klasifikasi penyakit mata, citra fundus retina, *deep learning*, EfficientNetB0, *convolutional neural network*.

I. PENDAHULUAN

Gangguan penglihatan menjadi salah satu permasalahan kesehatan global yang signifikan. Laporan dari *World Health Organization* menyebutkan bahwa lebih dari 2,2 miliar orang di seluruh dunia mengalami berbagai bentuk gangguan penglihatan atau kebutaan, dan hampir satu miliar kasus di antaranya sebenarnya dapat dicegah apabila mendapatkan penanganan yang tepat [1]. Penyakit mata seperti katarak, glaukoma, dan retinopati diabetik merupakan beberapa penyebab utama kehilangan penglihatan yang dapat berkembang secara progresif apabila tidak ditangani sejak dini. Keterlambatan diagnosis serta keterbatasan tenaga medis dan fasilitas diagnostik sering menyebabkan banyak kasus penyakit mata baru terdeteksi pada tahap lanjut [2].

Salah satu metode yang dapat diterapkan untuk mendukung deteksi dini penyakit mata adalah dengan memanfaatkan citra fundus retina sebagai sarana diagnosis yang bersifat non-

invasif [3]. Citra fundus memungkinkan visualisasi struktur retina seperti pembuluh darah, makula, dan saraf optik secara detail sehingga dapat digunakan untuk mengidentifikasi berbagai kondisi penyakit mata. Dalam beberapa tahun terakhir, metode *deep learning* khususnya *Convolutional Neural Network* (CNN) telah banyak digunakan dalam klasifikasi citra medis karena memiliki kemampuan yang baik dalam mengekstraksi fitur citra secara otomatis [4]. Salah satu arsitektur CNN yang banyak digunakan adalah EfficientNetB0 yang memiliki jumlah parameter relatif efisien namun tetap mampu menghasilkan performa klasifikasi yang tinggi pada berbagai tugas pengenalan citra [5].

Selain pemilihan arsitektur model, strategi pengolahan data juga memiliki peran penting dalam meningkatkan performa model *deep learning*. Teknik *preprocessing*, augmentasi citra, serta pengaturan *pipeline* data dapat membantu meningkatkan variasi data pelatihan dan kemampuan generalisasi model [6][7]. Oleh karena itu, penelitian ini mengimplementasikan model CNN dengan arsitektur EfficientNetB0 untuk melakukan klasifikasi penyakit mata pada citra fundus retina menggunakan *Eye Disease Dataset* dari Kaggle yang terdiri dari empat kelas, yaitu katarak, retinopati diabetik, glaukoma, dan normal. Penelitian ini juga mengevaluasi pengaruh rasio pembagian dataset terhadap performa model dengan tiga skenario pembagian data pelatihan dan validasi, yaitu 95%:5%, 90%:10%, dan 85%:15%.

II. METODE PENELITIAN

Beberapa tahapan utama dalam pembangunan model klasifikasi citra penyakit mata berbasis *deep learning* dimulai dari persiapan dataset citra retina, dilanjutkan dengan tahap *preprocessing* untuk menyesuaikan ukuran citra dan meningkatkan kualitas data sebelum digunakan sebagai input model. Setelah itu dilakukan pembagian dataset menjadi data pelatihan dan data validasi dengan beberapa variasi rasio untuk menganalisis pengaruh proporsi data terhadap performa model klasifikasi.

Citra yang telah diproses kemudian digunakan dalam proses pelatihan model dengan memanfaatkan arsitektur EfficientNetB0 sebagai *backbone* ekstraksi fitur. Untuk meningkatkan kemampuan representasi fitur, model dilengkapi dengan mekanisme *attention* menggunakan *Squeeze-and-Excitation Block* serta beberapa layer tambahan seperti *pooling*, *batch normalization*, dan *fully connected layer*. Proses pelatihan juga memanfaatkan *pipeline* data dan strategi optimasi untuk meningkatkan efisiensi komputasi. Evaluasi kinerja model dilakukan dengan menggunakan

berbagai metrik seperti *accuracy*, *F1-Score*, *precision*, *recall*, serta analisis ROC dan AUC.

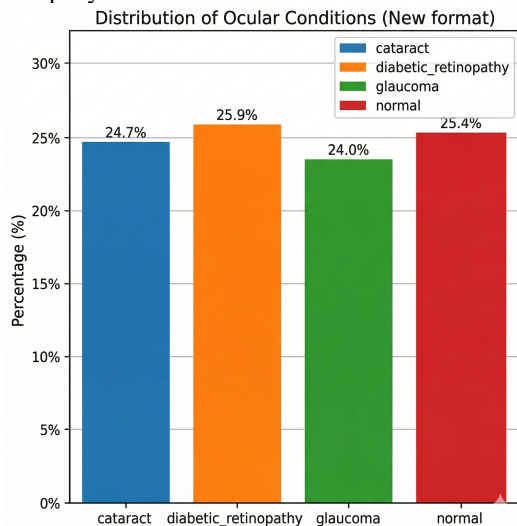
A. Lingkungan Komputasi dan Library yang Digunakan

Pemanfaatan lingkungan komputasi berbasis Kaggle Notebook yang mendukung akselerasi GPU dilakukan dengan menggunakan bahasa pemrograman Python, serta framework utama TensorFlow dan Keras dalam proses pembangunan dan pelatihan model klasifikasi citra. Beberapa library pendukung juga digunakan untuk mendukung proses pengolahan data dan evaluasi model, di antaranya NumPy, Pandas, Matplotlib, Seaborn, OpenCV, serta Scikit-learn untuk menghitung metrik evaluasi seperti *precision*, *recall*, dan *F1-Score*. Selain itu, library VisualKeras digunakan untuk memvisualisasikan arsitektur model *deep learning* yang dibangun pada penelitian ini.

B. Deskripsi Data

Dataset yang digunakan merupakan dataset publik dari platform Kaggle yang berisi citra fundus retina untuk klasifikasi penyakit mata berbasis citra medis. Dataset terdiri dari empat kelas, yaitu katarak, retinopati diabetik, glaukoma, dan mata normal dengan total 4.217 citra. Distribusi data meliputi 1.038 citra katarak, 1.098 citra retinopati diabetik, 1.007 citra glaukoma, dan 1.074 citra mata normal.

Data kemudian dibagi menjadi data pelatihan dan data validasi dengan beberapa variasi rasio, yaitu 95%:5%, 90%:10%, dan 85%:15%, untuk menganalisis pengaruh proporsi data terhadap performa model. Seluruh citra disesuaikan ke ukuran 224 × 224 piksel dengan tiga kanal warna (RGB) agar sesuai dengan kebutuhan input arsitektur CNN yang digunakan. Distribusi jumlah citra pada setiap kelas ditunjukkan pada Gambar 1 dalam bentuk diagram lingkaran. Proporsi data antar kelas relatif seimbang sehingga dataset dapat digunakan dengan baik untuk tugas klasifikasi multikelas penyakit mata.



Gbr. 1 Bar Chart Jumlah Dataset Setiap Kelas

Deskripsi dari tampilan citra mata normal menunjukkan retina yang jernih dan pembuluh darah yang jelas. Katarak tampak berkabut akibat kekeruhan pada lensa, retinopati diabetik ditandai dengan bercak perdarahan dan eksudat pada retina, sedangkan glaukoma menunjukkan perubahan pada area *optic cup*. Perbedaan karakteristik ini menjadi dasar untuk model *deep learning* dalam mempelajari pola klasifikasi pada citra retina.

C. Preprocessing Data

Tahap *preprocessing* dilakukan untuk mempersiapkan citra sebelum digunakan sebagai input pada model *deep learning*. Proses ini bertujuan untuk memastikan keseragaman format data, meningkatkan kualitas data, serta membantu model dalam mempelajari pola citra dengan lebih baik. Tahapan *preprocessing* yang dilakukan meliputi:

1. *Resizing* Citra

Seluruh citra pada dataset disesuaikan menjadi ukuran 224 × 224 piksel dengan tiga kanal warna (RGB). Ukuran ini dipilih karena sesuai dengan ukuran input standar yang digunakan oleh arsitektur EfficientNetB0.

2. *Data Augmentation*

Untuk meningkatkan variasi data pelatihan dan mengurangi risiko *overfitting*, diterapkan beberapa teknik augmentasi citra. Transformasi yang digunakan ditunjukkan pada Tabel 1.

TABEL I
TEKNIK AUGMENTASI PADA *PREPROCESSING* DATA

No.	Teknik Augmentasi	Deskripsi
1	<i>Random Crop</i>	Memotong area acak dari citra untuk meningkatkan variasi posisi objek
2	<i>Random Flip</i>	Membalik citra secara horizontal
3	<i>Random Rotation</i>	Memutar citra dengan sudut kecil
4	<i>Random Zoom</i>	Melakukan pembesaran atau pengecilan citra
5	<i>Random Contrast</i>	Mengubah tingkat kontras citra
6	<i>Random Brightness</i>	Mengubah tingkat kecerahan citra
7	<i>Random Translation</i>	Menggeser posisi citra secara horizontal atau vertikal
8	<i>Gaussian Noise</i>	Menambahkan noise acak pada citra

Teknik augmentasi ini membantu model menjadi lebih *robust* terhadap perubahan orientasi, pencahayaan, maupun variasi kondisi citra.

3. Penambahan Optimasi *Pipeline* Data (*Caching*, *Shuffle*, dan *Prefetch*)

Dataset diproses menggunakan *pipeline* data yang terdiri dari *caching*, *shuffle*, dan *prefetching* untuk meningkatkan efisiensi pemrosesan data selama pelatihan. *Caching* digunakan untuk menyimpan data yang telah diproses agar tidak perlu dimuat ulang pada setiap *epoch*, *shuffle* digunakan untuk mengacak urutan data pelatihan, sedangkan *prefetching* digunakan untuk mempersiapkan *batch* data berikutnya selama proses pelatihan berlangsung.

Mekanisme ini membantu meningkatkan efisiensi pelatihan karena *cache* menyimpan hasil *preprocessing* di memori sehingga data tidak perlu dibaca ulang pada setiap *epoch*, sementara *prefetch* memungkinkan data dipersiapkan lebih awal sebelum digunakan oleh model sehingga mampu mengurangi degradasi performa selama proses *training* dan meningkatkan efisiensi pelatihan secara keseluruhan [8]. Selain itu, dokumentasi resmi TensorFlow juga merekomendasikan penggunaan kombinasi *cache*, *prefetch*, dan *shuffle* karena dapat mengurangi overhead I/O serta meningkatkan *throughput pipeline* data dengan memisahkan waktu produksi dan konsumsi data [9].

4. One-Hot Encoding

Label kelas pada dataset diubah ke dalam format *one-hot encoding* agar sesuai dengan kebutuhan model klasifikasi multikelas. Dalam format ini, setiap kelas direpresentasikan sebagai vektor biner yang memungkinkan model menghitung kesalahan prediksi dengan lebih tepat. Dengan tahapan *preprocessing* tersebut, data citra yang digunakan pada proses pelatihan model menjadi lebih terstruktur, bervariasi, dan siap digunakan untuk proses klasifikasi menggunakan arsitektur *convolutional neural network*.

D. Pembagian Dataset

Setelah proses *preprocessing*, dataset kemudian dibagi menjadi data pelatihan (*training set*) dan data validasi (*validation set*). Pada penelitian ini dilakukan beberapa variasi rasio pembagian dataset untuk menganalisis pengaruh proporsi data terhadap performa model. Rasio pembagian dataset yang digunakan meliputi:

- 95% data pelatihan dan 5% data validasi
- 90% data pelatihan dan 10% data validasi
- 85% data pelatihan dan 15% data validasi

Pembagian dataset dilakukan menggunakan fungsi `image_dataset_from_directory` pada TensorFlow dengan parameter `validation_split` sehingga data dapat dipisahkan secara otomatis berdasarkan rasio yang ditentukan. Setiap konfigurasi rasio dataset diuji secara terpisah untuk melihat pengaruhnya terhadap akurasi dan stabilitas model.

E. Arsitektur Model

Model utama yang digunakan adalah EfficientNetB0, yaitu salah satu arsitektur *Convolutional Neural Network* (CNN) yang dikenal memiliki keseimbangan antara akurasi dan efisiensi komputasi. EfficientNetB0 dioptimalkan untuk efisiensi komputasi melalui pendekatan *compound scaling* yang secara bersamaan menyesuaikan kedalaman jaringan, lebar jaringan, dan resolusi input, sehingga mampu menekan biaya komputasi tanpa mengorbankan kemampuan representasi fitur [10].

Gbr. 2 Arsitektur EfficientNetB0

Pada implementasinya, EfficientNetB0 digunakan sebagai *backbone pretrained* yang telah dilatih sebelumnya menggunakan dataset ImageNet. Seluruh layer pada *backbone* diatur dalam kondisi *trainable* sehingga model dapat melakukan proses *fine-tuning* terhadap dataset penyakit mata yang digunakan. Proses ini memungkinkan model untuk menyesuaikan parameter jaringan dengan karakteristik citra fundus retina. Setelah proses ekstraksi fitur oleh EfficientNetB0, beberapa komponen tambahan diterapkan untuk meningkatkan kemampuan klasifikasi model, yaitu sebagai berikut. Arsitektur EfficientNetB0 ditunjukkan pada Gambar 2 [11].

1. Squeeze-and-Excitation (SE) Block

SE Block digunakan sebagai mekanisme *channel attention* yang bekerja dengan menyesuaikan bobot pada setiap kanal fitur sehingga model dapat lebih fokus pada fitur yang paling relevan yang mampu meningkatkan kemampuan representasi fitur dengan *overhead* komputasi yang relatif minimal [12].

2. MaxPooling2D

Setelah proses ekstraksi fitur, dilakukan reduksi dimensi menggunakan MaxPooling2D untuk mempertahankan fitur yang paling dominan dari area penting pada citra retina. Pendekatan ini membantu model mengekstraksi informasi visual yang lebih signifikan pada proses klasifikasi penyakit mata. Penggunaan MaxPooling diketahui mampu mempertahankan fitur paling dominan pada area penting citra [13].

3. Batch Normalization

Untuk meningkatkan stabilitas proses pelatihan, diterapkan *Batch Normalization* (BN) pada jaringan. Teknik ini berfungsi untuk menormalkan aktivasi pada setiap *mini-batch* sehingga distribusi data *antar-layer* menjadi lebih stabil dan dapat mempercepat proses konvergensi selama pelatihan model [14].

4. Dense Layer (128 Neuron)

Fitur yang telah diproses kemudian diteruskan ke *Dense Layer* dengan 128 neuron untuk mempelajari representasi fitur yang lebih kompleks sebelum dilakukan proses klasifikasi. Jumlah neuron sebanyak 128 dipilih untuk memberikan keseimbangan antara kemampuan pembelajaran fitur dan kompleksitas model sehingga tetap efisien secara komputasi.

5. Dropout (0,1)

Teknik Dropout digunakan sebagai metode regularisasi dengan nilai *dropout rate* sebesar 0,1 untuk mengurangi risiko

overfitting. *Dropout* bekerja dengan menonaktifkan sebagian neuron secara acak selama proses pelatihan sehingga jaringan tidak terlalu bergantung pada neuron tertentu. Penggunaan teknik ini mampu memutus *co-adaptation* antar neuron sehingga membantu meningkatkan kemampuan generalisasi model [15].

6. Output Layer (Softmax)

Pada tahap akhir, digunakan layer output dengan fungsi aktivasi softmax untuk menghasilkan probabilitas klasifikasi pada setiap kelas penyakit mata, yaitu katarak, retinopati diabetik, glaukoma, dan mata normal.

F. Konfigurasi Pelatihan dan Evaluasi Model

Konfigurasi pelatihan dilakukan untuk mengoptimalkan proses *training* model setelah arsitektur CNN dibangun. Beberapa komponen yang dikonfigurasi meliputi *optimizer*, strategi *learning rate*, *loss function*, *metrics*, *callback*, dan *class weighting*. Proses pelatihan dilakukan hingga maksimum 150 *epoch*, namun dapat dihentikan lebih awal menggunakan mekanisme *callback* untuk menjaga efisiensi pelatihan.

1. Optimizer

Model menggunakan Adam *optimizer*, yaitu algoritma optimasi adaptif yang mampu menyesuaikan *learning rate* untuk setiap parameter secara otomatis. Pendekatan ini banyak digunakan pada model *deep learning* karena mampu mempercepat proses konvergensi serta meningkatkan stabilitas pelatihan [16].

2. Learning Rate Strategy

Strategi *learning rate* yang digunakan adalah *Cyclical Learning Rate* (CLR). Metode ini mengatur nilai *learning rate* secara siklis selama proses pelatihan sehingga model dapat menghindari kondisi *local minima* dan mencapai konvergensi yang lebih baik [17].

Nilai *learning rate* berubah secara periodik antara batas minimum (*base_lr*) dan batas maksimum (*max_lr*). Parameter *stepsize* menentukan jumlah iterasi yang diperlukan untuk berpindah dari nilai minimum ke maksimum. Strategi ini memungkinkan model melakukan eksplorasi ruang parameter secara lebih luas sehingga membantu menghindari kondisi *local minima* dan meningkatkan stabilitas konvergensi selama proses pelatihan.

3. Loss Function

Model menggunakan *Categorical Crossentropy* karena tugas klasifikasi bersifat multi-class dengan label yang telah di-*one-hot encoding*. Fungsi ini mengukur perbedaan antara probabilitas prediksi model dan label sebenarnya sehingga model dapat belajar meminimalkan kesalahan prediksi.

4. Metrics

Selama proses pelatihan digunakan *accuracy* sebagai metrik utama untuk memantau performa model pada setiap *epoch*. Evaluasi akhir model kemudian dianalisis menggunakan metrik tambahan seperti *precision*, *recall*, dan *F1-Score*. Waktu komputasi digunakan untuk mengukur efisiensi model.

5. Callback

Beberapa *callback* diterapkan untuk meningkatkan efisiensi pelatihan, yaitu *EarlyStopping* untuk menghentikan pelatihan ketika *validation loss* tidak mengalami peningkatan, *ModelCheckpoint* untuk menyimpan bobot model terbaik berdasarkan nilai *validation loss* terendah, serta *LearningRateScheduler* untuk mengatur siklus *learning rate* menggunakan metode CLR.

6. Class Weighting

Class weighting diterapkan untuk mengatasi ketidakseimbangan jumlah data antar kelas. Dengan memberikan bobot lebih pada kelas minoritas, model dapat belajar secara lebih seimbang dan mengurangi bias terhadap kelas mayoritas.

III. HASIL DAN PEMBAHASAN

A. Hasil Evaluasi Model pada Berbagai Rasio Pembagian Dataset

Evaluasi model dilakukan menggunakan beberapa metrik performa yaitu *accuracy*, *precision*, *recall*, dan *F1-Score*, serta analisis waktu komputasi pelatihan. Pengujian dilakukan pada tiga skenario pembagian dataset yaitu 95:5, 90:10, dan 85:15 untuk menganalisis pengaruh proporsi data pelatihan terhadap performa model.

TABEL II
HASIL EVALUASI KINERJA MODEL PADA PEMBAGIAN DATASET

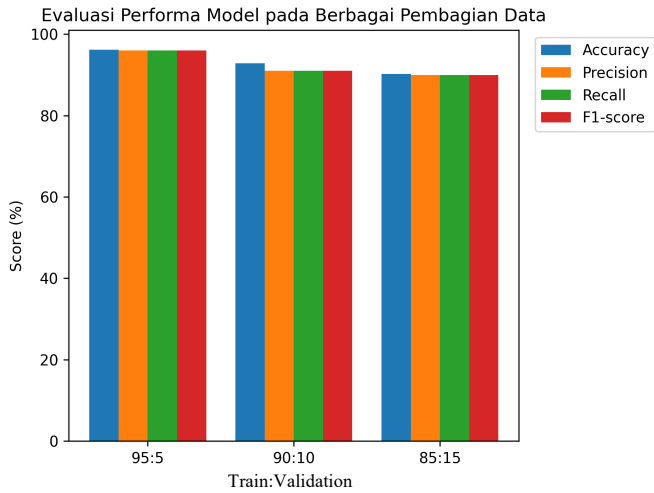
Rasio Dataset	Accuracy	Precision	Recall	F1-Score	Waktu Komputasi
95%:5%	96,19	96	96	96	8m 27s
90%:10%	92,87	91	91	91	16m 9s
85%:15%	90,19	90	90	90	22m 11s

Berdasarkan Tabel II, hasil evaluasi menunjukkan bahwa model memperoleh performa terbaik pada rasio pembagian dataset 95%:5% dengan nilai *accuracy* sebesar 96,19%, serta nilai *precision*, *recall*, dan *F1-Score* masing-masing sebesar 96%. Hasil ini menunjukkan bahwa proporsi data pelatihan yang lebih besar memberikan kesempatan bagi model untuk mempelajari pola dan karakteristik citra fundus retina secara lebih optimal, sehingga mampu meningkatkan kemampuan klasifikasi model. Pada rasio dataset 90%:10%, performa model mengalami penurunan dengan nilai *accuracy* sebesar 92,87%, serta *precision*, *recall*, dan *F1-Score* sebesar 91%. Penurunan performa juga terlihat pada rasio 85%:15%, dengan nilai *accuracy* sebesar 90,19% dan nilai *precision*, *recall*, serta *F1-Score* sebesar 90%. Hal ini menunjukkan bahwa semakin kecil proporsi data pelatihan yang digunakan, maka kemampuan model dalam mempelajari fitur penting dari citra retina juga cenderung menurun, sehingga berdampak pada performa klasifikasi yang dihasilkan.

Selain itu, perbedaan rasio dataset juga memengaruhi waktu komputasi selama proses pelatihan. Pada rasio 95%:5%, waktu komputasi yang dibutuhkan adalah 8m 27s, sedangkan pada rasio 90%:10% meningkat menjadi 16m 9s, dan pada rasio 85%:15% mencapai 22m 11s. Hasil ini menunjukkan bahwa perubahan komposisi data pada proses pelatihan dan validasi dapat memengaruhi durasi pelatihan model. Dalam

penulisan waktu komputasi tersebut, m menyatakan menit (minutes) dan s menyatakan detik (seconds).

B. Analisis Efektivitas Performa Model



Gbr. 3 Bar Chart Hasil Performa Model

Berdasarkan Gambar 3, terlihat bahwa performa model tertinggi diperoleh pada rasio pembagian dataset 95%:5% dengan nilai accuracy sebesar 96,19%, serta *precision*, *recall*, dan *F1-Score* yang berada pada kisaran 96%. Hasil ini menunjukkan bahwa proporsi data pelatihan yang lebih besar memberikan kesempatan bagi model untuk mempelajari pola fitur pada citra fundus retina secara lebih optimal. Dengan jumlah data pelatihan yang lebih banyak, model mampu mengenali karakteristik masing-masing kelas penyakit mata dengan lebih baik sehingga menghasilkan tingkat prediksi yang lebih akurat.

Sebaliknya, pada rasio dataset 90%:10% dan 85%:15%, seluruh metrik performa mengalami penurunan bertahap. Pada rasio 90%:10%, nilai accuracy menurun menjadi 92,87%, sedangkan pada rasio 85%:15% turun menjadi 90,19%, dengan nilai *precision*, *recall*, dan *F1-Score* yang juga berada pada kisaran 90–91%. Penurunan ini menunjukkan bahwa semakin kecil proporsi data pelatihan yang digunakan, kemampuan model dalam mempelajari variasi pola pada citra menjadi lebih terbatas. Meskipun demikian, nilai seluruh metrik evaluasi masih menunjukkan performa klasifikasi yang cukup baik dan stabil pada ketiga skenario pembagian dataset.

C. Analisis Efisiensi Model

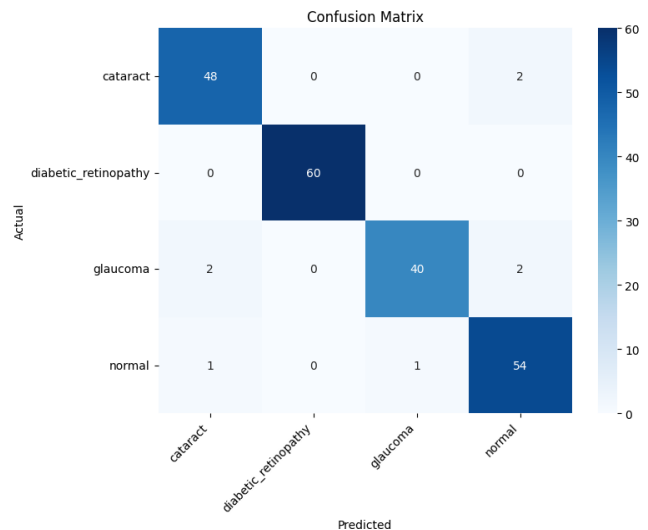


Gbr. 4 Line Chart Peningkatan Waktu Kompitasi pada Rasio Dataset

Berdasarkan Gambar 4, waktu komputasi pelatihan menunjukkan kecenderungan meningkat seiring dengan perubahan rasio pembagian dataset. Pada rasio 95%:5%, waktu komputasi yang dibutuhkan relatif paling rendah yaitu sekitar 8 menit 27 detik. Hal ini menunjukkan bahwa proses pelatihan dapat berlangsung lebih cepat ketika proporsi data validasi lebih kecil, sehingga sebagian besar data digunakan secara langsung dalam proses pembelajaran model tanpa menambah beban evaluasi validasi yang terlalu besar.

Selanjutnya, pada rasio 90%:10%, waktu komputasi meningkat menjadi sekitar 16 menit 9 detik, dan kembali meningkat pada rasio 85%:15% hingga mencapai sekitar 22 menit 11 detik. Peningkatan ini menunjukkan bahwa semakin besar proporsi data validasi yang digunakan, maka proses pelatihan membutuhkan waktu komputasi yang lebih lama karena model harus melakukan evaluasi performa pada data validasi dalam jumlah yang lebih besar pada setiap *epoch*.

D. Confusion Matrix



Gbr. 5 Confusion Matrix Ratio 95%:5%

Berdasarkan Gambar 5 yang menunjukkan hasil *confusion matrix* dari model pada rasio pembagian dataset 95%:5%,

yang merupakan konfigurasi dengan performa terbaik sekaligus waktu komputasi paling efisien. *confusion matrix* digunakan untuk menganalisis performa klasifikasi model secara lebih rinci pada setiap kelas penyakit mata. Hasil evaluasi menunjukkan bahwa model mampu mengklasifikasikan sebagian besar citra dengan benar. Pada kelas *diabetic retinopathy*, seluruh data berhasil diprediksi dengan tepat sebanyak 60 citra, sehingga tidak ditemukan kesalahan klasifikasi pada kelas ini. Sementara itu, pada kelas *cataract*, model berhasil mengklasifikasikan 48 citra dengan benar, dengan 2 citra yang salah diprediksi sebagai kelas normal. Hasil ini menunjukkan bahwa model memiliki kemampuan yang sangat baik dalam mengenali karakteristik utama pada sebagian besar kelas penyakit mata.

Pada kelas *glaucoma*, model berhasil memprediksi 40 citra dengan benar, namun terdapat 2 citra yang salah diklasifikasikan sebagai *cataract* dan 2 citra lainnya sebagai normal. Sementara itu, pada kelas normal, model berhasil mengklasifikasikan 54 citra dengan benar, dengan 1 citra salah diprediksi sebagai *cataract* dan 1 citra sebagai *glaucoma*. Secara keseluruhan, hasil *confusion matrix* menunjukkan bahwa model memiliki tingkat akurasi yang tinggi dengan jumlah kesalahan klasifikasi yang relatif kecil. Hal ini menunjukkan bahwa model mampu mempelajari perbedaan karakteristik visual pada citra fundus retina dengan baik sehingga dapat membedakan kondisi mata normal maupun berbagai jenis penyakit mata secara efektif.

E. Analisis Hasil Prediksi Model pada Sampel Citra

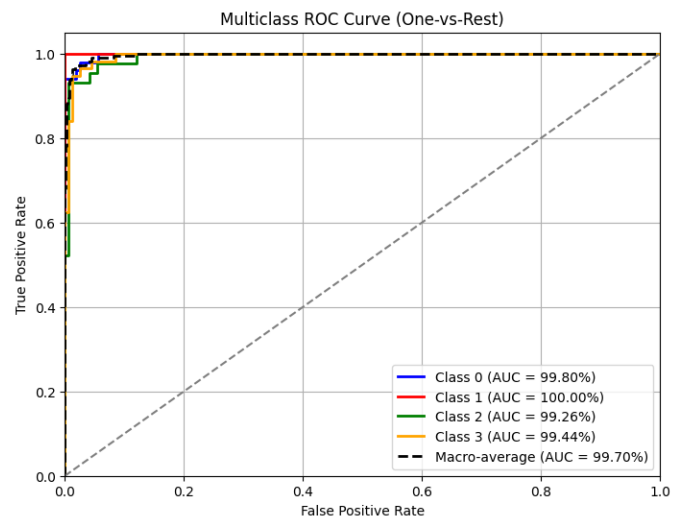
Untuk memberikan gambaran lebih jelas mengenai kemampuan model dalam melakukan klasifikasi citra fundus retina, dilakukan pengujian prediksi menggunakan beberapa sampel citra dari setiap kelas penyakit mata. Pengujian ini menggunakan model terbaik yang diperoleh pada rasio pembagian dataset 95%:5%, yang sebelumnya menunjukkan performa evaluasi tertinggi serta waktu komputasi yang paling efisien. Sampel citra dipilih dari beberapa kelas yaitu *cataract*, *diabetic retinopathy*, *glaucoma*, dan normal untuk melihat bagaimana model melakukan prediksi secara langsung terhadap citra yang diberikan.

Hasil menunjukkan seluruh sampel citra yang diuji, hanya terdapat satu citra yang mengalami kesalahan klasifikasi, sedangkan citra lainnya berhasil diprediksi sesuai dengan label sebenarnya. Kesalahan prediksi ini disebabkan oleh kemiripan pola visual antara beberapa kondisi retina, seperti kemiripan tekstur atau pencahayaan pada citra fundus yang dapat memengaruhi proses ekstraksi fitur oleh model. Meskipun demikian, secara keseluruhan hasil prediksi menunjukkan bahwa model memiliki kemampuan klasifikasi yang baik dalam membedakan kondisi mata normal maupun berbagai jenis penyakit mata.

F. Analisis Hasil ROC

Evaluasi performa model juga dilakukan menggunakan *Receiver Operating Characteristic* (ROC) untuk melihat kemampuan model dalam membedakan setiap kelas penyakit

pada berbagai nilai threshold. Analisis ROC pada penelitian ini difokuskan pada skenario dengan performa terbaik, yaitu model EfficientNetB0 dengan pembagian rasio dataset 95:5 (*train : validation*). ROC pada kasus klasifikasi multikelas dihitung menggunakan pendekatan *One-vs-Rest* (OvR), yaitu dengan membandingkan masing-masing kelas terhadap seluruh kelas lainnya. Kurva ROC menggambarkan hubungan antara *True Positive Rate* (TPR) dan *False Positive Rate* (FPR). Semakin mendekati sudut kiri atas, maka performa model semakin baik dalam membedakan kelas.



Gbr. 6 Hasil Kurva ROC

Berdasarkan hasil pengujian pada Gambar 6, diperoleh nilai *Area Under Curve* (AUC) yang sangat tinggi pada setiap kelas. Nilai AUC untuk Class 0 yaitu penyakit katarak memiliki nilai sebesar 99.80%, Class 1 yang berarti retinopati diabetik yang memiliki nilai sebesar 100.00%, Class 2 yang berarti glaukoma mempunyai nilai sebesar 99.26%, dan Class 3 yang berarti mata normal memiliki nilai sebesar 99.44%. Selain itu, nilai *macro-average* AUC yang diperoleh adalah 99.70%, yang menunjukkan bahwa model memiliki kemampuan klasifikasi yang sangat baik secara keseluruhan. Tingginya nilai AUC pada seluruh kelas menunjukkan bahwa model EfficientNetB0 mampu membedakan setiap kategori penyakit mata dengan tingkat akurasi yang sangat tinggi. Kurva ROC yang berada jauh di atas garis diagonal (*baseline classifier*) menandakan bahwa model memiliki kemampuan diskriminasi yang sangat baik dalam membedakan antara kelas positif dan negatif pada setiap kategori penyakit.

Dengan demikian, hasil evaluasi ROC ini memperkuat bahwa konfigurasi model EfficientNetB0 dengan rasio dataset 95:5 merupakan konfigurasi yang paling optimal dalam penelitian ini untuk tugas klasifikasi penyakit mata.

IV. KESIMPULAN

Penelitian ini mengimplementasikan model EfficientNetB0 berbasis *Deep learning* untuk melakukan klasifikasi penyakit mata pada citra fundus retina yang terdiri dari empat kelas, yaitu katarak, retinopati diabetik, glaukoma, dan mata normal.

Proses penelitian meliputi tahap *preprocessing* citra, penerapan teknik *data augmentation*, optimasi *pipeline* data menggunakan *caching*, *shuffle*, dan *prefetch*, serta pembangunan arsitektur CNN dengan penambahan komponen seperti *Squeeze-and-Excitation block*, *pooling*, *batch normalization*, *dense layer*, dan *dropout*. Model kemudian dilatih menggunakan strategi optimasi Adam *optimizer* dengan pengaturan *Cyclical Learning Rate* untuk meningkatkan stabilitas konvergensi selama proses pelatihan.

Hasil evaluasi menunjukkan bahwa rasio pembagian dataset 95%:5% memberikan performa terbaik dibandingkan rasio lainnya dengan nilai *accuracy* sebesar 96,19%, serta *precision*, *recall*, dan *F1-Score* masing-masing sebesar 96%, dengan waktu komputasi pelatihan 8 menit 27 detik. Analisis lebih lanjut melalui confusion matrix, prediksi sampel citra, dan kurva ROC menunjukkan bahwa model mampu membedakan setiap kelas penyakit mata dengan tingkat kesalahan yang relatif kecil dan nilai AUC rata-rata sebesar 99,70%. Secara keseluruhan, hasil penelitian ini menunjukkan bahwa arsitektur EfficientNetB0 mampu mengekstraksi fitur citra fundus retina secara efektif dan efisien sehingga berpotensi digunakan sebagai pendekatan berbasis kecerdasan buatan untuk mendukung deteksi dini penyakit mata.

UCAPAN TERIMA KASIH

Penulis mengucapkan puji syukur kepada Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga penelitian ini dapat diselesaikan dengan baik. Penulis juga menyampaikan terima kasih kepada orang tua yang senantiasa memberikan doa, dukungan, dan motivasi selama proses penelitian dan penulisan artikel ini. Selain itu, penulis mengucapkan terima kasih kepada dosen pembimbing serta seluruh tim penulis atas bimbingan, arahan, kerja sama, dan kontribusi yang diberikan selama proses penelitian hingga penyusunan artikel ini. Dukungan dari berbagai pihak tersebut sangat membantu dalam terselesaikannya penelitian ini dengan baik.

REFERENSI

- [1] World Health Organization, "Blindness and vision impairment," World Health Organization. Accessed: Sep. 04, 2025. [Online]. Available: <https://www.who.int/en/news-room/fact-sheets/detail/blindness-and-visual-impairment?>
- [2] A. Almazroa et al., "The Patients' Perspective for the Impact of Late Detection of Ocular Diseases on Quality of Life: A Cross-Sectional Study," *Clin. Optom.*, vol. 15, no. September, pp. 191–204, 2023, doi: 10.2147/OPTO.S422451.
- [3] Q. Yang et al., "Use of artificial intelligence with retinal imaging in screening for diabetes-associated complications: systematic review," *eClinicalMedicine*, vol. 81, no. February, p. 103089, 2025, doi: 10.1016/j.eclinm.2025.103089.
- [4] R. Alsohemi and S. Dardouri, "Fundus Image-Based Eye Disease Detection Using EfficientNetB3 Architecture," *J. Imaging*, vol. 11, no. 8, pp. 1–18, 2025, doi: 10.3390/jimaging11080279.
- [5] Mithunraj et al., "Optimizing CNN performance in medical imaging with cross-modality pre-training: A study using MobileNetV3," *Intell. Med.*, vol. 12, no. July, p. 100281, 2025, doi: 10.1016/j.ibmed.2025.100281.
- [6] A. O. Asia et al., "Detection of Diabetic Retinopathy in Retinal Fundus Images Using CNN Classification Models," *Electron.*, vol. 11, no. 17, 2022, doi: 10.3390/electronics11172740.
- [7] X. Sun et al., "Robust Retinal Vessel Segmentation from a Data Augmentation Perspective," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12970 LNCS, pp. 189–198, 2021, doi: 10.1007/978-3-030-87000-3_20.
- [8] N. Krichevsky, R. St Louis, and T. Guo, "Quantifying and Improving Performance of Distributed *Deep learning* with Cloud Storage," *Proc. - 2021 IEEE Int. Conf. Cloud Eng. IC2E 2021*, pp. 99–109, 2021, doi: 10.1109/IC2E52221.2021.00024.
- [9] DeepWiki, "Performance Optimization," 20/04/2025. [Online]. Available: <https://deepwiki.com/tensorflow/docs/5.3-performance-optimization>
- [10] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.
- [11] J. R. Riza, Z. Mahmud, S. A. Ridita, and A. Bhattacharya, "PhytoCare : A hybrid approach for identifying Rice , Potato and Corn diseases," no. February, pp. 1–59, 2024.
- [12] S. Narayanan, "Enhanced plant disease classification with attention-based convolutional neural network using squeeze and excitation mechanism," no. August, 2025, doi: 10.3389/frai.2025.1640549.
- [13] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8689 LNCS, no. PART 1, pp. 818–833, 2014, doi: 10.1007/978-3-319-10590-1_53.
- [14] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network *training* by reducing internal covariate shift," *32nd Int. Conf. Mach. Learn. ICML 2015*, vol. 1, pp. 448–456, 2015.
- [15] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.
- [16] E. M. Dogo, O. J. Afolabi, and B. Twala, "On the Relative Impact of Optimizers on Convolutional Neural Networks with Varying Depth and Width for Image Classification," *Appl. Sci.*, vol. 12, no. 23, 2022, doi: 10.3390/app122311976. of Massachusetts, Amherst, MA, CMPSCI Tech. hal. 99-02, 1999.
- [17] G. Atteia, A. A. Alhussan, and N. A. Samee, "BO-ALLCNN: Bayesian-Based Optimized CNN for Acute Lymphoblastic Leukemia Detection in Microscopic Blood Smear Images," *Sensors*, vol. 22, no. 15, pp. 1–20, 2022, doi: 10.3390/s22155520.