



# Perbandingan Kinerja Arsitektur MobileNetV2 dan MobileNetV3 Dalam Klasifikasi Penyakit Retina pada Citra Optical Coherence Tomography (OCT) Menggunakan Optimizer AdamW dan SGD

Ricko Andreas Kartono\*, Nur Rachmat

Fakultas Ilmu Komputer Dan Rekayasa, Program Studi Informatika, Universitas Multi Data Palembang, Palembang, Indonesia

Email: <sup>1,\*</sup>rickokartono05@gmail.com, <sup>2</sup>nur.rachmat@mdp.ac.id

Email Penulis Korespondensi: rickokartono05@gmail.com

**Abstrak**—Penyakit retina merupakan salah satu gangguan penglihatan serius yang dapat menyebabkan penurunan fungsi penglihatan hingga kebutaan. Proses diagnosis penyakit retina umumnya masih dilakukan secara manual oleh tenaga medis melalui pemeriksaan citra Optical Coherence Tomography (OCT), yang memerlukan waktu lama, ketelitian, serta berpotensi menimbulkan kesalahan diagnosis. Studi sebelumnya sebagian besar menggunakan arsitektur CNN yang lebih besar dan kompleks, dengan optimasi yang terbatas pada beberapa jenis *optimizer* yang umum. Penelitian ini bertujuan untuk mengembangkan model klasifikasi penyakit retina secara otomatis menggunakan metode *Convolutional Neural Network* (CNN) dengan memanfaatkan arsitektur MobileNetV2 dan MobileNetV3, yang lebih ringan dan efisien, sehingga memungkinkan aplikasi yang lebih cepat dan dapat diakses di perangkat sumber daya terbatas. Arsitektur yang digunakan mencakup MobileNetV2, MobileNetV3-Large dan MobileNetV3-Small, serta dilakukan perbandingan antara dua jenis optimiser, yaitu AdamW dan SGD. Dataset yang digunakan terdiri atas 4000 citra OCT, yang terbagi ke dalam empat kelas, yaitu Normal, *Choroidal Neovascularization* (CNV), *Diabetic Macular Edema* (DME), dan *Drusen*. Proses pelatihan dilakukan menggunakan pendekatan *transfer learning* dan hasil evaluasi diukur menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*. Hasil penelitian menunjukkan bahwa kombinasi arsitektur MobileNetV2 dengan *batch size* 16 serta *optimizer* AdamW atau SGD menghasilkan performa terbaik, dengan akurasi mencapai 85,75%, yang merupakan akurasi terbaik dari beberapa kombinasi yang diuji. Temuan ini menegaskan bahwa arsitektur ringan memiliki potensi besar untuk dikembangkan menjadi aplikasi diagnosis penyakit retina yang cepat, akurat, dan dapat digunakan langsung di lapangan melalui perangkat mobile berbasis *deep learning*.

**Kata Kunci:** Penyakit Retina; Optical Coherence Tomography (OCT); Optimizer ; MobileNetV2; MobileNetV3; Deep Learning

**Abstract**—Retinal diseases are serious visual disorders that can lead to decreased visual function and even blindness. The diagnosis of retinal diseases is generally still performed manually by medical professionals through the examination of Optical Coherence Tomography (OCT) images, a process that requires considerable time, high precision, and is prone to diagnostic errors. Previous studies have mostly employed larger and more complex CNN architectures, with optimization limited to a few commonly used optimizers. This study aims to develop an automatic retinal disease classification model using Convolutional Neural Network (CNN) methods by leveraging the lightweight and efficient MobileNetV2 and MobileNetV3 architectures, enabling faster applications that can be deployed on resource-constrained devices. The architectures evaluated include MobileNetV2, MobileNetV3-Large, and MobileNetV3-Small, along with a comparison of two optimizers, namely AdamW and Stochastic Gradient Descent (SGD). The dataset used consists of 4,000 OCT images divided into four classes: Normal, Choroidal Neovascularization (CNV), Diabetic Macular Edema (DME), and Drusen. The training process was conducted using a transfer learning approach, and model performance was evaluated using accuracy, precision, recall, and F1-score metrics. The experimental results indicate that the combination of the MobileNetV2 architecture with a batch size of 16 and either the AdamW or SGD optimizer achieved the best performance, reaching an accuracy of 85.75%, which is the highest among all tested configurations. These findings highlight the strong potential of lightweight architectures to be developed into fast, accurate, and field-deployable retinal disease diagnostic applications on mobile devices using deep learning.

**Keywords:** Retinal Disease; Optical Coherence Tomography (OCT); Optimizer; MobileNetV2; MobileNetV3; Deep Learning

## 1. PENDAHULUAN

Mata merupakan salah satu panca indera terpenting bagi manusia karena berfungsi sebagai organ penglihatan yang mendeteksi dan merespons cahaya [1]. Retina sebagai bagian yang paling sensitif terhadap cahaya, mengubah rangsangan cahaya menjadi sinyal saraf sehingga gangguan pada retina dapat berdampak langsung pada penglihatan [2].

Penyakit retina merupakan kondisi ketika jaringan retina mengalami kerusakan atau kelainan. Beberapa penyakit yang sering dijumpai antara lain Choroidal Neovascularization (CNV), Diabetic Macular Edema (DME), dan *Drusen*. CNV merupakan tahap lanjut dari *age-related macular degeneration* (AMD) yang ditandai pertumbuhan pembuluh darah abnormal dan berpotensi menyebabkan kebutaan. DME adalah komplikasi retinopati diabetik yang terjadi akibat penumpukan cairan pada makula. Sementara itu, *Drusen* merupakan endapan lipid dan protein yang terbentuk di bawah retina. Dalam jumlah kecil, drusen biasanya tidak berbahaya, tetapi jika banyak atau berukuran besar dapat menjadi indikator awal AMD [3]. Perbedaan karakteristik CNV, DME, dan *Drusen* menunjukkan bahwa ketiganya dapat berkembang menjadi kondisi berat. Karena risiko kebutaan yang dapat terjadi, deteksi dini menjadi langkah penting untuk mencegah kerusakan permanen pada penglihatan.[4].

Secara umum, identifikasi penyakit retina masih banyak dilakukan secara manual oleh tenaga medis dengan memeriksa kondisi retina pasien secara langsung. Metode manual memerlukan ketelitian tinggi dan waktu yang tidak sedikit, terutama ketika jumlah pasien banyak. Kondisi tersebut berpotensi meningkatkan risiko kesalahan diagnosis [5]. Untuk mengatasi kendala ini, teknologi pencitraan seperti Optical Coherence Tomography (OCT) mulai banyak digunakan karena mampu menghasilkan citra retina beresolusi tinggi [6].



Meskipun memiliki kualitas citra yang baik, hasil OCT sering dipengaruhi oleh pergerakan pasien, perbedaan perangkat, dan kondisi pemindaian, sehingga dapat muncul noise maupun kontras rendah. Variasi ini dapat mengganggu proses klasifikasi otomatis penyakit retina [5]. OCT menjadi pilihan utama karena mampu menampilkan penampang melintang retina dengan detail tinggi, sehingga informasi struktural yang kompleks dapat terlihat dengan jelas. Hal ini sulit dicapai dengan metode lain seperti *fundus photography* atau *fluorescein angiography* yang tidak menampilkan struktur lapisan retina secara mendalam [7].

Walaupun OCT menghasilkan citra tiga dimensi (3D), banyak penelitian menggunakan citra dua dimensi (2D) karena lebih efisien dan ringan secara komputasi. Citra 2D mengurangi kebutuhan sumber daya sehingga proses pelatihan model dapat dilakukan lebih cepat, meskipun informasi struktural tidak selengkap citra 3D. Citra 3D memberikan detail yang lebih komprehensif, tetapi memerlukan komputasi yang lebih besar [8], [9]. Oleh karena itu, penelitian ini menggunakan arsitektur yang ringan, yaitu MobileNetV2 dan MobileNetV3 yang dirancang untuk keterbatasan sumber daya komputasi, seperti perangkat *mobile* atau *low-resource-hardware*.

Seiring perkembangan zaman, teknologi juga mendorong pemanfaatan *Artificial Intelligence* (AI), cabang dari AI, yaitu *machine learning* dan bagiannya yaitu *deep learning* untuk dikembangkan untuk komputer belajar dari data dan mengenali pola secara otomatis berdasarkan data [10]. *Deep learning* saat ini banyak dimanfaatkan untuk analisis citra medis. *Convolutional Neural Network* (CNN) merupakan salah satu metode dari *deep learning* yang efektif dalam mengenali pola pada gambar dan mengklasifikasikan objek, termasuk penyakit retina [11], [12]. Berbagai penelitian sebelumnya menunjukkan CNN mampu memberikan hasil akurat pada citra OCT. Terdapat penelitian terdahulu yang menggabungkan dua jenis arsitektur yaitu *custom CNN* dan *Gated Recurrent Unit* (GRU) pada citra OCT yang mampu mendeteksi penyakit retina dengan mencapai akurasi sebesar 98,90% [13].

Selain itu, menggunakan arsitektur CNN lainnya, yaitu VGG16 dan Inception V3 dan memperoleh akurasi masing-masing 93,12% dan 91,37% [14]. Namun, model yang lebih ringan seperti MobileNetV2 dan MobileNetV3 ternyata mampu menghasilkan akurasi lebih tinggi sebesar 97,73% menggunakan MobileNetV2, sementara akurasi sebesar 97,68% dengan MobileNetV3 [15], [16].

Meskipun MobileNetV2 dan MobileNetV3 telah menunjukkan performa yang sangat tinggi, sebagian besar penelitian sebelumnya lebih menunjukkan pada pemilihan arsitektur model, sementara pemilihan optimasi yang tepat pada proses pelatihan kurang menunjukkan pada penelitian sebelumnya. Padahal, citra OCT memiliki karakteristik berupa *noise*, kontras rendah, serta variasi antar perangkat dan kondisi pemindaian yang berpotensi menyebabkan *overfitting*, terutama pada *dataset* medis yang umumnya terbatas dan tidak seimbang. Oleh karena itu, pemilihan *optimizer* menjadi faktor penting dalam meningkatkan stabilitas pelatihan serta kemampuan generalisasi model.

Perbedaan akurasi antar penelitian ini tidak hanya menunjukkan bahwa model ringan selalu lebih unggul dibanding model besar. Variasi akurasi dipengaruhi oleh perbedaan *dataset* yang digunakan, ukuran dan kualitas data, teknik augmentasi, preprocessing, serta metrik evaluasi yang dipakai. MobileNet memiliki performa lebih baik pada dataset kecil atau tidak seimbang karena arsitekturnya yang lebih ringan. Sebaliknya VGG16 memerlukan dataset yang lebih banyak karena arsitekturnya lebih berat. Oleh karena itu, meskipun lebih besar dan berat, VGG16 dapat menghasilkan akurasi yang lebih kecil apabila *dataset* yang digunakan kecil dibandingkan arsitektur MobileNet yang memakai *dataset* banyak [14]. MobileNetV3 sendiri menunjukkan performa yang baik pada berbagai jenis citra, tidak hanya citra medis. Adapun penelitian membuktikan bahwa MobileNetV3-Small dapat mencapai akurasi hingga 99,5% dalam klasifikasi penyakit daun jagung, lebih tinggi dibandingkan MobileNetV3-Large [17]. Keunggulan MobileNetV3 terletak pada efisiensinya dalam memproses citra dengan tetap menjaga akurasi tinggi. Model yang ringan seperti MobileNetV2 dan MobileNetV3 sangat cocok untuk *dataset* medis yang umumnya terbatas dan tidak seimbang. Oleh karena itu, pemilihan arsitektur MobileNetV2 dan MobileNetV3 dalam penelitian ini bertujuan untuk mendukung pengembangan sistem klasifikasi citra OCT yang efisien dan berpotensi diimplementasikan pada perangkat *mobile* atau lingkungan dengan sumber daya komputasi terbatas.

Selain arsitektur model, keberhasilan pelatihan CNN juga dipengaruhi oleh pemilihan *optimizer*. *Optimizer* berperan memperbarui bobot selama pelatihan agar model mencapai performa terbaik. SGD dikenal stabil dan memiliki kemampuan generalisasi yang baik, sedangkan AdamW mempercepat proses pembelajaran melalui teknik *decoupled weight decay* [18]. Perbandingan kedua *optimizer* dilakukan untuk melihat perbedaan karakteristik dan kemampuan dalam meningkatkan kinerja model pada data saat dilatih, terutama pada citra OCT yang memiliki *noise* dan potensi *overfitting* akibat keterbatasan data. Dengan demikian, penelitian ini menggunakan AdamW dan SGD untuk membandingkan efektivitas keduanya pada arsitektur MobileNet.

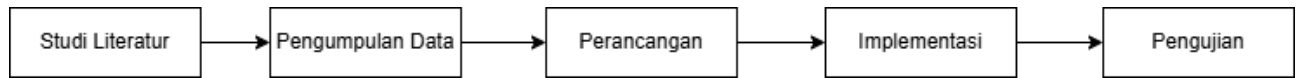
Berdasarkan berbagai penelitian terdahulu, penelitian ini bertujuan menerapkan model CNN dengan arsitektur MobileNetV2 dan MobileNetV3 untuk mengklasifikasikan penyakit retina pada citra OCT. Diharapkan penelitian ini menghasilkan model yang akurat, efisien, dan dapat mendukung proses identifikasi penyakit retina secara cepat serta membantu pelayanan kesehatan dalam mendeteksi gangguan penglihatan secara lebih dini.

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Penelitian ini dilakukan melalui beberapa tahapan yang dirancang secara sistematis yang bertujuan menghasilkan model klasifikasi penyakit retina pada citra OCT menggunakan arsitektur MobileNetV2 dan MobileNetV3. Proses penelitian

dilakukan melalui sejumlah tahapan yang disusun secara terstruktur hingga menghasilkan hasil penelitian. Tahapan penelitian tersebut dapat dilihat pada Gambar 1.



**Gambar 1.** Tahapan Penelitian

Gambar 1 menunjukkan tahap penelitian yang disusun secara sistematis untuk memastikan setiap proses terstruktur. Tahapan dimulai dari studi literatur sebagai dasar pemilihan metode dan arsitektur model, dilanjutkan dengan pengumpulan data yang berperan penting dalam menentukan kualitas pelatihan model. Tahap perancangan dan implementasi dilakukan untuk menerapkan model CNN sesuai rancangan, dan tahap pengujian bertujuan mengevaluasi performa model dalam mengklasifikasikan penyakit retina pada citra OCT.

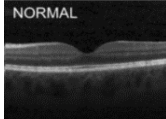
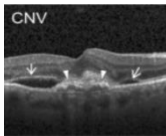
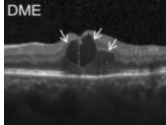
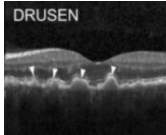
## 2.2 Studi Literatur

Studi literatur dilakukan dengan proses pengumpulan berbagai sumber referensi yang digunakan sebagai dasar pembelajaran serta pemahaman mengenai penggunaan metode CNN dan arsitektur MobileNetV2 dan MobileNetV3. Literatur yang digunakan mencakup jurnal penelitian yang memuat landasan teori, metodologi, serta langkah-langkah penelitian yang relevan untuk dijadikan acuan dalam penelitian ini.

## 2.3 Pengumpulan Data

Pada penelitian ini, digunakan dataset citra OCT yang diperoleh dari website *kaggle*. Dataset ini merupakan kumpulan citra retina pasien nyata yang sebelumnya dihimpun oleh berbagai institusi medis, seperti Shiley Eye Institute of UCSD, California Retinal Research Foundation, Medical Center Ophthalmology Associates, The Shanghai First People's Hospital, dan Beijing Tongren Eye Center. Secara keseluruhan, dataset asli berjumlah sekitar 84.495 citra yang terbagi ke dalam empat kategori utama, yaitu Normal, CNV, DME, dan Drusen. Namun, penelitian ini hanya memanfaatkan 4.000 citra yang diambil dari seluruh dataset untuk menjaga keseimbangan kelas dan efisiensi proses pelatihan sesuai kemampuan komputasi. Pemilihan subset dataset ini bertujuan untuk mengurangi redundansi data, dikarenakan pada citra medis yang memiliki kemiripan pola antar sampel. Kondisi tersebut dapat mempengaruhi proses pelatihan model dan meningkatkan risiko *overfitting*. Dataset sebesar 4.000 citra ini kemudian dibagi ke dalam tiga folder utama, yaitu data latih (80%), data validasi (10%), dan data uji (10%). Berdasarkan pembagian tersebut, data latih berisi 3.200 citra, data validasi 400 citra, dan data uji 400 citra. Kemudian setiap folder dipisahkan lagi ke dalam empat kelas subfolder, yaitu Normal, CNV, DME, dan *Drusen*, sehingga setiap kelas berisi 800 citra untuk pelatihan, 100 untuk validasi, dan 100 untuk pengujian. Rincian Dataset dapat dilihat pada Tabel 1.

**Tabel 1.** Rincian Dataset

No	Nama	Nomor	Penjelasan	Contoh
1	Normal	1000	Kondisi mata tanpa kelainan dan lapisan retina sehat	
2	CNV	1000	Pembentukan pembuluh darah baru di bawah retina	
3	DME	1000	Pembengkakan cairan pada bagian tengah retina	
4	Drusen	1000	Bintik kecil dibawah retina seperti lemak atau protein	

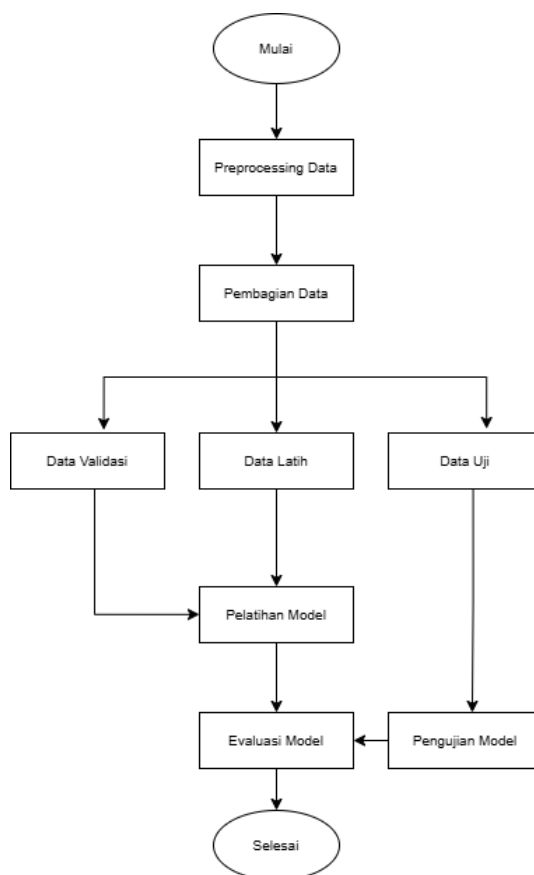
Pada Tabel 1, menunjukkan karakteristik dari empat kelas utama pada citra OCT. Kelas Normal memiliki struktur retina yang sehat, CNV ditandai dengan pembuluh darah abnormal atau terdapat pembuluh darah baru di bawah retina yang dapat menyebabkan kebutaan, DME menunjukkan pembengkakan akibat penumpukan cairan yang disebabkan oleh



penyakit diabetes, dan DRUSEN terlihat sebagai bintik kecil seperti lemak atau protein, jika semakin banyak dapat menyebabkan kebutaan.

### 2.3 Perancangan

Pada tahap ini peneliti melakukan beberapa tahapan perancangan model mulai dari proses pengumpulan data hingga tahap evaluasi kinerja model dengan menggunakan bahasa pemrograman *Python*, serta *framework deep learning PyTorch* dan *torchvision*. Skema perancangan dapat dilihat pada Gambar 2.



**Gambar 2.** Skema Perancangan

Berikut ini adalah penjelasan mengenai proses skema perancangan model pada Gambar 2.

a. *Preprocessing Data*

Pada tahap ini, *preprocessing* data dilakukan dengan beberapa proses, yaitu melakukan perubahan ukuran pada citra menjadi ukuran 224 x 224, kemudian dilakukan *bilateral filter* untuk mengurangi noise seperti bintik halus pada citra OCT tanpa merusak tepi, *Bilateral filter* menggunakan parameter diameter 5, *sigma color* 20, dan *sigma space* 20. Selanjutnya dilakukan augmentasi data untuk mencegah *overfitting*, seperti melakukan rotasi acak hingga 15 derajat, *flipping*, serta penyesuaian tingkat kecerahan, kontra dan saturasi dengan nilai 0,1, dan normalisasi piksel citra menggunakan nilai *mean* dan standar deviasi *ImageNet*, sehingga distribusi nilai piksel berada pada skala yang sesuai untuk proses pelatihan model.

b. *Pembagian Data*

Pembagian data dilakukan dengan menggunakan *random sampling*, dikarenakan data yang tidak seimbang dari 84,495 citra OCT, hanya diambil 4000 data, kemudian dibagi menjadi tiga bagian, yaitu data latih 80%, data uji 10%, dan data validasi 10%.

c. *Pelatihan Model*

Pada tahap ini, pelatihan model memanfaatkan teknik *feature extraction*, yang memanfaatkan nilai bobot yang sudah dilatih (*pretrained weights*) yang berasal dari *dataset ImageNet*. Bobot ini berfungsi sebagai pengekstrak fitur awal sehingga model tidak perlu belajar fitur dasar dari awal. Selanjutnya dilakukan proses *fine-tuning*, di mana lapisan akhir dilatih kembali agar sesuai dengan jumlah kelas pada dataset. Proses pelatihan menggunakan *learning rate* sebesar 0,001, dengan variasi *batch size* 16 dan 32, serta jumlah epoch maksimal sebanyak 50 *Epoch*. Untuk meningkatkan efektivitas pelatihan dan mencegah terjadinya *overfitting*, diterapkan teknik *early stopping* dengan parameter *patience* sebesar 8, yang menghentikan proses pelatihan secara otomatis ketika akurasi validasi tidak mengalami peningkatan dalam beberapa *epoch*. Selain itu, metode *ReduceLRonPlateau* dengan nilai *patience* yang sama untuk menurunkan nilai *learning rate* secara bertahap ketika nilai *loss* validasi semakin buruk, sehingga membantu model mencapai nilai akurasi model yang lebih optimal.



#### d. Pengujian Model

Pada tahap ini, model yang telah dilatih dilakukan pengujian menggunakan dataset *testing*, yang bertujuan untuk melihat apakah model mampu mengenali gambar baru dan tidak hanya mengenali data *training*.

#### e. Evaluasi Model

Evaluasi model dilakukan untuk mengetahui seberapa baik model bekerja selama proses pelatihan dan pengujian. Evaluasi tersebut menggunakan *confusion matrix*.

### 2.4 Implementasi

Implementasi dilakukan dari sistem yang telah dirancang pada tahap sebelumnya dengan menggunakan data *training* untuk melatih model agar mampu mengenali pola citra, serta data *validation* untuk mengevaluasi kinerja model selama proses pelatihan. Setelah model mencapai performa yang optimal, dilakukan pengujian menggunakan data *testing* yang telah dipisahkan sebelumnya. Implementasi ini menggunakan metode *Convolutional Neural Network* (CNN) dengan arsitektur MobileNetV2 dan MobileNetV3 varian *Large* dan *Small*, yang diterapkan menggunakan teknik *transfer learning* pada dataset citra retina OCT, serta membandingkan kedua jenis *optimizer*, yaitu AdamW dan SGD, sehingga sistem dapat mengenali serta mengklasifikasikan citra retina ke dalam kategori penyakit dengan tingkat akurasi yang baik.

### 2.5 Pengujian

Pada tahapan ini, dilakukan pengujian terhadap model sistem yang telah dibangun sebelumnya menggunakan data uji yang telah dipisahkan dari data pelatihan. Hasil dari proses uji coba ini kemudian dievaluasi menggunakan *confusion matrix*, yang berfungsi untuk menilai kinerja model melalui beberapa metrik evaluasi, yaitu, *accuracy*, *precision*, *recall*, dan *F1-Score* [19].

*Accuracy* merupakan rasio antara jumlah prediksi yang benar terhadap total keseluruhan data uji [20]. Metrik ini menunjukkan seberapa besar proporsi data yang berhasil diklasifikasikan dengan benar oleh model. Nilai *accuracy* dihitung dengan menggunakan persamaan (1).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (1)$$

*Precision* merupakan metrik yang digunakan untuk mengukur tingkat akurasi model dalam memprediksi data positif dengan benar, yaitu seberapa banyak prediksi positif yang sesuai dengan kondisi sebenarnya, sedangkan *recall* merupakan metrik yang menunjukkan kemampuan model mengenali seluruh data positif dengan benar. Kedua metrik ini digunakan secara bersamaan untuk menilai keseimbangan antara ketepatan dan kelengkapan dalam mendeteksi model [20]. Rumus masing-masing metrik ditunjukkan pada persamaan (2) dan (3).

$$Precision = \frac{TP}{TP+FP} \times 100\% \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \times 100\% \quad (3)$$

*F1-Score* (F1) merupakan rata-rata nilai harmonis antara nilai *precision* dan nilai *recall*, yang digunakan untuk menilai keseimbangan antara ketepatan prediksi dan kemampuan model dalam mengenali seluruh data positif secara benar [20]. Rumus perhitungannya ditunjukkan pada persamaan (4).

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

## 3. HASIL DAN PEMBAHASAN

### 3.1 Hasil Perbandingan Kinerja Model

Pada bagian ini, dilakukan analisis perbandingan terhadap kinerja model dengan arsitektur MobileNetV2, MobileNetV3 varian *Large* dan *Small*. Setiap model diuji menggunakan kombinasi *hyperparameter tuning*, meliputi variasi dua jenis *optimizer*, yaitu AdamW dan SGD, serta perbedaan *batch size*, yaitu 16 dan 32. Perbandingan dapat dilihat pada Tabel 2, Tabel 3, dan Tabel 4.

**Tabel 2.** Hasil MobileNetV2

Nomor	Optimizer	Batch Size	Akurasi	Precision	Recall	F1-Score
1	AdamW	16	85,75%	0,86	0,86	0,86
2	SGD	16	85,75%	0,86	0,86	0,86
3	AdamW	32	84,75%	0,85	0,85	0,85
4	SGD	32	83%	0,83	0,83	0,83

Berdasarkan hasil pengujian pada Tabel 2, model dengan arsitektur MobileNetV2 dengan kombinasi *optimizer* AdamW dan SGD dengan *batch size* 16 memberikan akurasi sebesar 85,75%, dengan nilai precision, recall, dan F1-score yang sama, yaitu 0,86. Sementara itu, penggunaan *batch size* 32 menghasilkan akurasi 84,75% pada AdamW dan 83% pada SGD, dengan nilai precision, recall, dan F1-score yang sedikit lebih rendah.

**Tabel 3.** Hasil MobileNetV3 Small

Nomor	Optimizer	Batch Size	Akurasi	Precision	Recall	F1-Score
1	AdamW	16	78,75%	0,79	0,79	0,79
2	SGD	16	77,25%	0,78	0,77	0,77
3	AdamW	32	81,5%	0,81	0,82	0,81
4	SGD	32	75%	0,75	0,75	0,75

Berdasarkan hasil pengujian pada Tabel 3, model dengan arsitektur MobileNetV3 *Small*, menunjukkan bahwa akurasi berada di rentang 75% hingga 81,5%, dengan nilai precision, recall, dan F1-score yang berubah mengikuti hasil akurasi. Nilai akurasi tertinggi terdapat pada *optimizer* AdamW dengan *batch size* 32 dengan akurasi 81,5%, sedangkan nilai terendah terdapat pada *optimizer* SGD dengan *batch size* yang sama menghasilkan akurasi 75%.

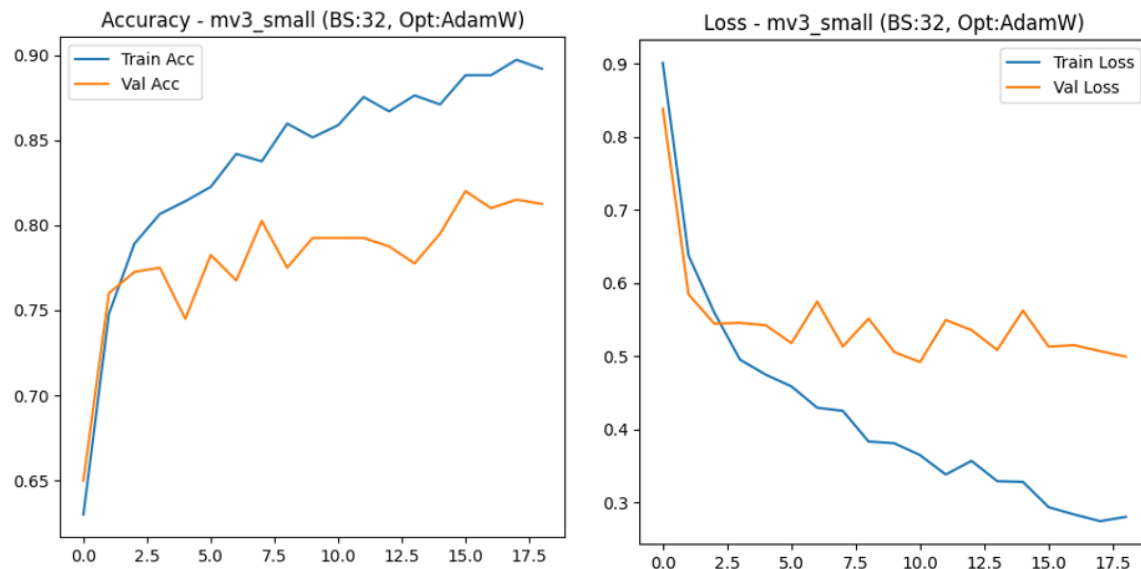
**Tabel 3.** Hasil MobileNetV3 Large

Nomor	Optimizer	Batch Size	Akurasi	Precision	Recall	F1-Score
1	AdamW	16	83,25%	0,83	0,83	0,83
2	SGD	16	78,75%	0,79	0,79	0,79
3	AdamW	32	83,75%	0,84	0,84	0,84
4	SGD	32	79,5%	0,81	0,80	0,79

Berdasarkan hasil pengujian pada Tabel 4, model dengan arsitektur MobileNetV3 *Large*, menunjukkan bahwa akurasi berada di rentang 78,75% hingga 83,75%. dengan nilai precision, recall, dan F1-score yang berubah mengikuti hasil akurasi. Nilai akurasi tertinggi terdapat pada *optimizer* AdamW dengan *batch size* 32 dengan akurasi 83,75%, sedangkan nilai terendah terdapat pada *optimizer* SGD dengan *batch size* 16 menghasilkan akurasi sebesar 78,75%.

### 3.2 Hasil Kinerja Model Terbaik

Pada bagian ini, akan ditampilkan hasil kinerja model terbaik yang diperoleh dari pengujian MobileNetV2 dan MobileNetV3 varian *Large* dan *Small*, dengan menampilkan grafik akurasi dan *loss*, serta menampilkan hasil *confusion matrix*.

**Gambar 3.** Grafik Akurasi dan *Loss* MobileNetV3 *Small* AdamW

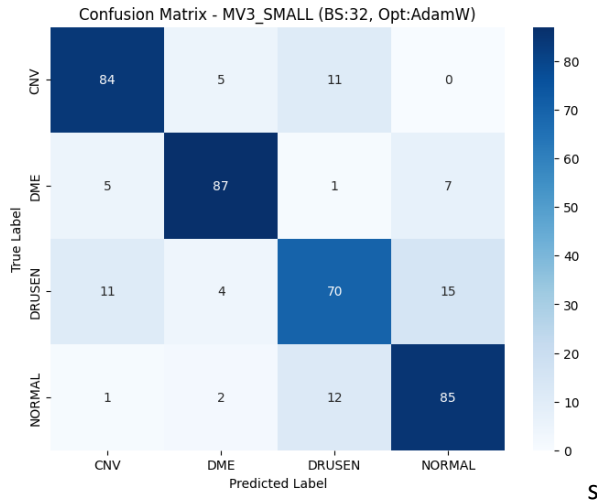
Pada Gambar 3, memperlihatkan grafik akurasi dan *loss* menggunakan arsitektur MobileNetV3 *Small*. Grafik kiri menunjukkan *training accuracy* (garis biru) yang terus meningkat secara konsisten sejak awal pelatihan, mencapai nilai tertinggi mendekati 0.90 pada akhir *epoch*, yang menunjukkan bahwa model semakin pintar dalam mengenali pola pada data latih. Sementara itu, *validation accuracy* (garis oranye) meningkat tajam pada awalnya, tetapi setelah sekitar *epoch* ke-5, peningkatannya mulai melambat di kisaran nilai 0.81. Adanya jarak yang semakin lebar antara garis biru dan oranye setelah *epoch* ke-5 menandakan bahwa model mulai mengalami *overfitting*.

Grafik kanan menunjukkan *training loss* (garis biru) yang menurun tajam di awal pelatihan dan terus turun hingga mencapai nilai rendah di bawah 0.3, menunjukkan bahwa model secara efektif mengurangi tingkat kesalahan pada data latih. Sebaliknya, *validation loss* (garis oranye) menurun dengan cepat hanya sampai *epoch* ke-2, namun setelah itu cenderung stagnan (*plateau*) di sekitar nilai 0.5 dan tidak menunjukkan penurunan lebih lanjut hingga pelatihan selesai.

Secara keseluruhan, grafik ini menunjukkan bahwa model memiliki kapasitas belajar yang sangat baik pada data latih, namun performanya pada data validasi sedikit menurun. Hal ini membuat proses pelatihan dihentikan atau



diterapkan *early stopping* lebih awal sekitar *epoch* ke-5 atau ke-6 saat *validation loss* mulai stabil, berguna untuk mencegah *overfitting* dan menjaga model tetap baik.



Gambar 4. Confusion Matrix MobileNetV3 Small AdamW

Pada Gambar 4, menampilkan *confusion matrix* yang memperlihatkan bahwa model mampu bekerja dengan cukup baik. Mayoritas prediksi sudah cukup tepat, di mana performa paling memuaskan terlihat pada kelas DME dan Normal, dengan masing-masing 87 dan 85 data yang berhasil diklasifikasikan dengan benar. Kelas CNV juga menunjukkan hasil yang cukup baik dengan 84 prediksi tepat. Meskipun demikian, tantangan terbesar model ini terletak pada kelas Drusen. Di sini, akurasi paling rendah hanya 70 data yang benar, dan sering terjadi kesalahan menebak atau *misclassification*.

Kesalahan yang cukup mencolok adalah 15 data Drusen yang justru diprediksi sebagai Normal, serta 11 data Drusen yang dianggap sebagai CNV. Sebaliknya, kebingungan juga terjadi pada 12 data Normal yang meleset diprediksi menjadi Drusen. Pola kesalahan ini menggambarkan bahwa model masih agak kesulitan membedakan fitur visual antara Drusen dan Normal, serta sebagian CNV, kemungkinan karena kemiripan tekstur atau bentuk pada citra tersebut.



Gambar 5. Grafik Akurasi Dan Loss MobileNetV3 Large AdamW

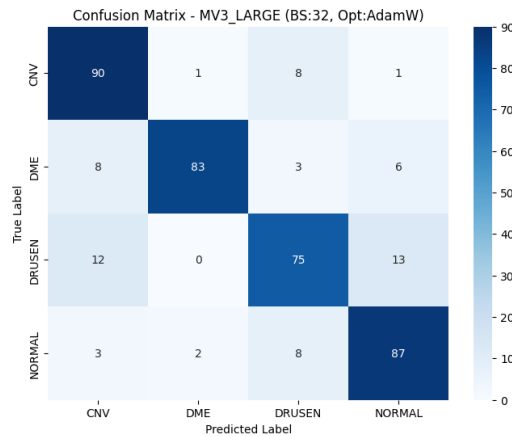
Pada Gambar 5, memperlihatkan grafik kiri menunjukkan *training accuracy* (garis biru) yang terus meningkat secara konsisten sejak awal pelatihan, mencapai nilai tertinggi di atas 0.95 pada akhir *epoch*, yang menunjukkan bahwa model semakin pintar dalam mengenali pola pada data latih. Sementara itu, *validation accuracy* (garis oranye) meningkat tajam pada awalnya, tetapi setelah sekitar *epoch* ke-9, peningkatannya mulai melambat di kisaran nilai 0.85. Adanya jarak yang semakin lebar antara garis biru dan oranye setelah *epoch* ke-9 menandakan bahwa model mulai mengalami *overfitting*.

Grafik kanan menunjukkan *training loss* (garis biru yang menurun tajam di awal pelatihan dan terus turun hingga mencapai nilai rendah di sekitar 0.1, menunjukkan bahwa model secara efektif mengurangi tingkat kesalahan pada data latih. Sebaliknya, *validation loss* (garis oranye) menurun dengan cepat pada fase awal hingga mencapai titik terendah di



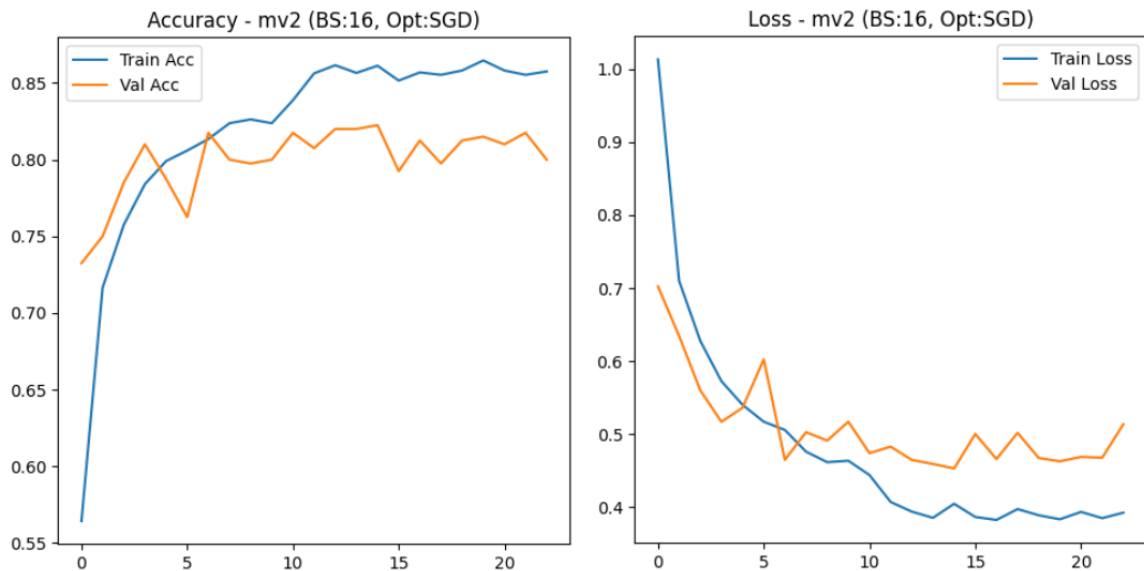
sekitar *epoch* ke-8, namun setelah itu justru menunjukkan tren kenaikan kembali dan tidak menunjukkan penurunan lebih lanjut hingga pelatihan selesai.

Secara keseluruhan, grafik ini menunjukkan bahwa model memiliki kapasitas belajar yang sangat baik pada data latih, namun performanya pada data validasi cenderung menurun di tahap akhir. Hal ini membuat proses pelatihan sebaiknya dihentikan atau diterapkan *early stopping* lebih awal sekitar *epoch* ke-8 atau ke-9 saat *validation loss* belum meningkat kembali, berguna untuk mencegah *overfitting* dan menjaga model tetap baik.



Gambar 6. Confusion Matrix MobileNetV3 Large AdamW

Pada Gambar 6, *confusion matrix* diatas menunjukkan hasil bahwa model ini melakukan kinerja yang cukup baik, dengan prediksi yang benar pada sebagian besar kelas, terutama pada CNV dan Normal, dengan 90 dan 87 data terklasifikasi dengan benar. Namun, kesalahan klasifikasi lebih banyak terjadi pada kelas Drusen, di mana beberapa data Drusen diprediksi sebagai Normal sebanyak 13 data dan CNV sebanyak 12 data, serta terdapat data DME yang terklasifikasikan sebagai CNV sebanyak 8 data. Hal ini menunjukkan bahwa model sedikit kesulitan dalam membedakan antara kelas tersebut, yang mungkin memiliki bentuk serupa. Meskipun demikian, model ini menunjukkan performa yang baik secara keseluruhan, namun masih memerlukan penyempurnaan lebih lanjut, terutama pada beberapa kelas yang sering mengalami *missclassification*.

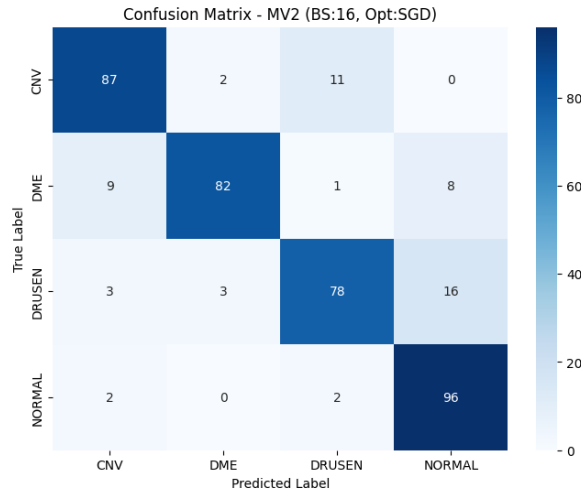


Gambar 7. Grafik Akurasi dan Loss MobileNetV2 SGD

Pada Gambar 7, memperlihatkan grafik akurasi dan *loss*, grafik kiri menunjukkan *training accuracy* (garis biru) yang terus meningkat sejak awal pelatihan, mencapai stabilitas di sekitar nilai 0.85, yang menunjukkan bahwa model semakin baik dalam mempelajari data latih. Sementara itu, *validation accuracy* (garis oranye) juga meningkat pada awalnya, tetapi setelah sekitar *epoch* ke-12 hingga ke-15, peningkatannya mulai melambat dan tidak terus meningkat, yang menandakan model mulai mengalami *overfitting*. Hal ini menunjukkan bahwa meskipun *training accuracy* terus meningkat, *validation accuracy* berhenti menunjukkan perbaikan signifikan, sehingga pelatihan dihentikan menggunakan teknik *early stopping* pada sekitar *epoch* ke-12 hingga ke-15. Grafik kanan menunjukkan *training loss* (garis biru) yang menurun tajam di awal pelatihan dan stabil setelah mencapai nilai rendah sekitar 0.4, menunjukkan bahwa model sudah mengurangi kesalahan pada data latih. Sementara itu, *validation loss* (garis oranye) menurun dengan cepat pada awalnya

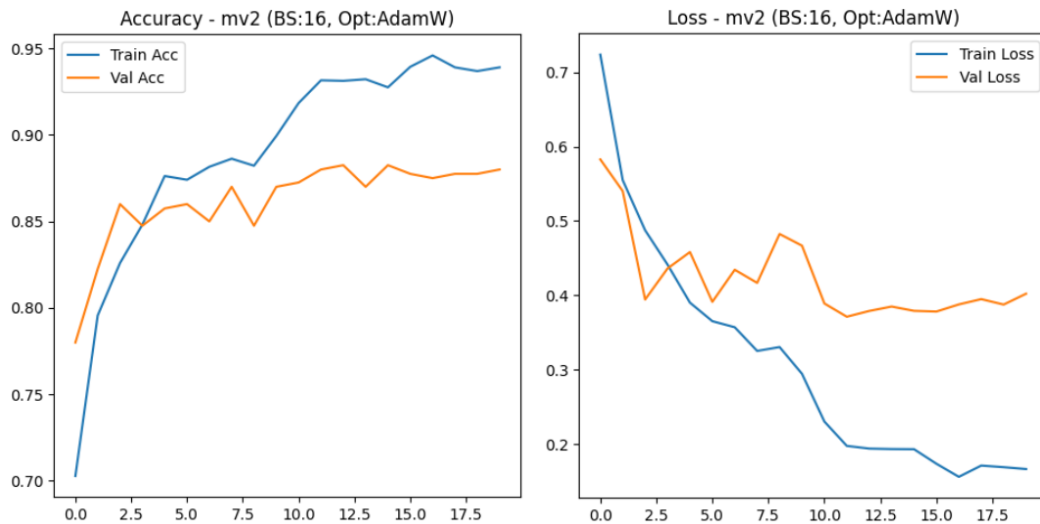
tetapi mulai tidak meningkat sekitar *epoch* ke12, menunjukkan bahwa model tidak lagi dapat memperbaiki kesalahan pada data validasi.

Secara keseluruhan grafik ini menunjukkan bahwa *early stopping* diterapkan untuk menghentikan pelatihan pada titik di mana model sudah mencapai performa terbaiknya pada data validasi, mencegah *overfitting*, dan memastikan model tidak terlalu terlatih pada data latih.



**Gambar 8.** Confusion Matrix MobileNetV2 SGD

Pada Gambar 8, *confusion matrix* diatas menunjukkan hasil bahwa model ini melakukan kinerja yang cukup baik, dengan prediksi yang benar pada sebagian besar kelas, terutama pada Normal dan CNV, dengan 96 dan 87 data terklasifikasi dengan benar. Namun, kesalahan klasifikasi lebih banyak terjadi pada kelas DME dan Drusen, di mana beberapa data DME diprediksi sebagai CNV sebanyak 9 data dan Normal sebanyak 8 data, serta beberapa data Drusen yang terklasifikasikan sebagai Normal sebanyak 16 data. Hal ini menunjukkan bahwa model sedikit kesulitan dalam membedakan antara kelas tersebut, yang mungkin memiliki bentuk serupa. Meskipun demikian, model ini menunjukkan performa yang baik secara keseluruhan, namun masih memerlukan penyempurnaan lebih lanjut, terutama pada beberapa kelas yang sering mengalami *missclassification*.

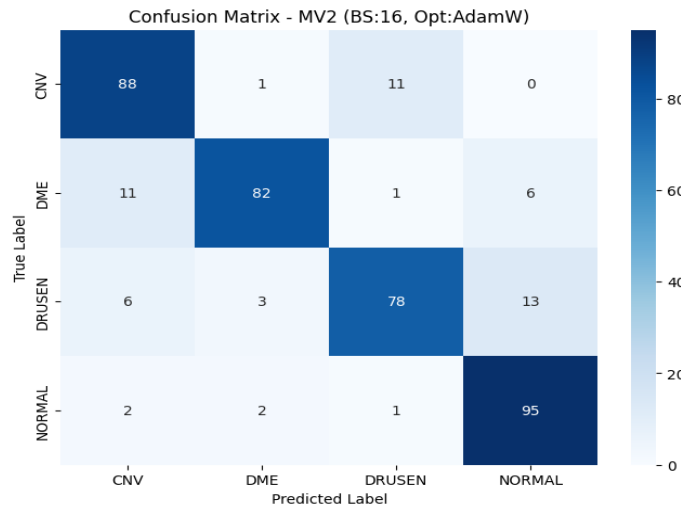


**Gambar 9.** Grafik Akurasi dan Loss MobileNetV2 AdamW

Pada Gambar 9, memperlihatkan grafik akurasi dan *loss*, grafik kiri menunjukkan *training accuracy* (garis biru) terus meningkat secara signifikan, mencapai stabilitas di sekitar 0.95, yang menunjukkan bahwa model semakin baik dalam mempelajari data latih. Sementara itu, *validation accuracy* (garis oranye) juga menunjukkan peningkatan pada awalnya, tetapi setelah *epoch* ke-12 hingga ke-15, peningkatannya mulai melambat dan tidak terus meningkat. Hal ini menunjukkan bahwa model mulai mengalami *overfitting*, meskipun *training accuracy* terus membaik, *validation accuracy* tidak menunjukkan peningkatan yang signifikan. Oleh karena itu, pelatihan dihentikan menggunakan teknik *early stopping* sekitar *epoch* ke-12 hingga ke-15. Pada grafik kanan, *training loss* (garis biru) menurun tajam pada awal pelatihan dan stabil di sekitar nilai 0.4, menunjukkan bahwa model berhasil mengurangi kesalahan pada data latih. *Validation loss* (garis oranye) menurun dengan cepat pada awalnya, tetapi kemudian tidak meningkat setelah *epoch* ke-

12, menandakan bahwa model tidak lagi mampu memperbaiki kesalahan pada data validasi. Hal ini memberitahukan bahwa model sudah mencapai titik optimal dan tidak bisa memperbaiki kesalahan lebih lanjut pada data validasi.

Secara keseluruhan grafik ini menunjukkan bahwa *early stopping* diterapkan untuk menghentikan pelatihan pada titik di mana model sudah mencapai performa terbaiknya pada data validasi, mencegah *overfitting*, dan memastikan model tidak terlalu terlatih pada data latih.



**Gambar 10.** Confusion Matrix MobileNetV2 AdamW

Pada Gambar 10, *confusion matrix* menunjukkan bahwa model ini melakukan kinerja yang cukup baik, dengan prediksi yang benar pada sebagian besar kelas, terutama pada kelas Normal dan CNV, dengan 95 dan 88 data terklasifikasi dengan benar. Namun, kesalahan klasifikasi lebih banyak terjadi pada kelas DME dan Drusen, di mana beberapa data DME sering diprediksi sebagai CNV sebanyak 11 data dan Normal sebanyak 6 data, serta beberapa data Drusen yang terklasifikasikan sebagai Normal sebanyak 13 data. Hal ini menunjukkan bahwa model sedikit kesulitan dalam membedakan antara kelas-kelas tersebut, yang mungkin memiliki ciri yang serupa. Meskipun demikian, model ini menunjukkan performa yang baik secara keseluruhan, namun masih memerlukan penyempurnaan lebih lanjut, terutama pada beberapa kelas yang sering mengalami *misclassification*.

Dari beberapa hasil kinerja model terbaik, model terbaik didapat pada MobileNetV2 dengan dua jenis *optimizer* AdamW dan SGD dengan *batch size* 16.

### 3.3 Pembahasan

Pada penelitian ini dilakukan pelatihan model menggunakan arsitektur MobileNetV2 dan MobileNetV3 varian *Large* dan *Small*. Dengan variasi *batch size* 16 dan 32, serta dua jenis *optimizer*, yaitu AdamW dan SGD.

Hasil penelitian menunjukkan bahwa model dengan arsitektur MobileNetV2, *batch size* 16, serta *optimizer* AdamW dan SGD, memberikan hasil akurasi yang lebih baik daripada kombinasi lainnya. Hal ini terlihat dari nilai akurasi keduanya mencapai sebesar 85,75%, yang merupakan akurasi tertinggi dalam seluruh eksperimen yang dilakukan. Nilai ini menunjukkan bahwa MobileNetV2 lebih mampu mengenali pola-pola penting pada citra OCT dibandingkan MobileNetV3, baik varian *Small* maupun *Large*.

Keunggulan MobileNetV2 dikarenakan menggunakan struktur, yaitu *depthwise separable convolution* yang lebih sederhana dan stabil. Pada dataset medis yang terbatas dan memiliki kemiripan pola antar kelas seperti citra OCT, arsitektur yang terlalu kompleks berpotensi mengalami *overfitting*. MobileNetV3, yang memiliki struktur *squeeze-and-excitation* dan fungsi aktivasi *hard-swish*, memerlukan jumlah dataset yang lebih banyak. Oleh karena itu, pada kondisi dataset yang relatif kecil dan seimbang, MobileNetV2 justru menunjukkan performa yang lebih baik dibandingkan MobileNetV3.

Dari sisi *optimizer*, baik AdamW maupun SGD pada konfigurasi MobileNetV2 dan *batch size* 16 menghasilkan nilai akurasi yang sama, yaitu sebesar 85,75%. Meskipun demikian, kedua *optimizer* memiliki karakteristik yang berbeda. AdamW menerapkan konsep *decoupled weight decay*, di mana regularisasi bobot dipisahkan dari pembaruan gradien, sehingga mampu mengurangi risiko *overfitting* dan menghasilkan proses *konvergensi* yang lebih stabil. Sehingga AdamW sangat cocok pada citra OCT yang umumnya memiliki *noise* serta variasi kualitas citra akibat yang berbeda. Sementara itu, SGD dikenal memiliki stabilitas dan kemampuan generalisasi yang baik, meskipun konvergensinya cenderung lebih lambat. Kesamaan nilai akurasi menunjukkan bahwa pada konfigurasi tertentu, kedua *optimizer* mampu mencapai solusi yang serupa, dengan AdamW unggul dari sisi kestabilan pembelajaran dan SGD unggul dari sisi konsistensi generalisasi.

Analisis grafik akurasi dan *loss* selama pelatihan juga memperkuat hasil tersebut. Model MobileNetV2 dengan *batch size* 16 menunjukkan peningkatan akurasi yang konsisten serta penurunan *loss* yang tidak terlalu menurun, menandakan proses pembelajaran yang stabil. Sebaliknya, pada *batch size* 32 dan arsitektur MobileNetV3, grafik cenderung menunjukkan *loss* semakin naik, hal ini mengindikasikan bahwa pembelajaran yang kurang stabil.



Berdasarkan hasil *confusion matrix*, sebagian besar kelas berhasil diklasifikasikan dengan benar. Namun, kesalahan klasifikasi paling sering terjadi antara kelas Drusen dan Normal. Hal ini dikarenakan tampilan drusen pada citra OCT memiliki bentuk dan pola visual yang mirip dengan jaringan retina normal, sehingga sulit dibedakan oleh model.

Jika dibandingkan dengan penelitian terdahulu, akurasi pada penelitian ini masih berada di bawah beberapa studi sebelumnya yang memiliki akurasi sangat tinggi. Namun, perbedaan ini tidak hanya menunjukkan bahwa kelemahan terdapat pada model, melainkan dipengaruhi oleh perbedaan tujuan penelitian, ukuran dataset, serta kompleksitas konfigurasi pelatihan. Penelitian ini lebih menekankan pada efisiensi model ringan dengan kebutuhan komputasi yang rendah, sehingga lebih relevan untuk skenario implementasi pada perangkat *mobile* atau lingkungan dengan sumber daya terbatas. Perbandingan hasil penelitian dapat dilihat pada Tabel 5.

**Tabel 5.** Perbandingan Hasil Penelitian

Nomor	Arsitektur	Dataset	Akurasi
1	Custom CNN + GRU	84,495	78,75%
2	VGG, InceptionV3	1000	93,12%, 91,37%
3	MobileNetV2	20,087	97,73%
4	MobileNetV3	84,495	97,68%
5	MobileNetV2 Dan MobileNetV3	4000	85,75%

Berdasarkan Tabel 5, dapat dilihat terdapat perbedaan performa akurasi dari berbagai arsitektur dan jumlah *dataset* yang digunakan. Model dengan arsitektur yang lebih berat seperti VGG dan InceptionV3 dan dataset berukuran besar pada arsitektur yang lebih ringan cenderung menghasilkan akurasi yang lebih tinggi. Sementara itu, penelitian ini dengan menggunakan arsitektur ringan, yaitu MobileNetV2 dan MobileNetV3 pada dataset yang lebih kecil dan seimbang, yaitu 4.000 citra menghasilkan akurasi 85,75%, yang lebih rendah dibandingkan beberapa model lain pada tabel, Hasil ini menunjukkan bahwa model ringan tetap mampu memberikan kinerja yang stabil dan efisien, sehingga dapat mengimplemntasikan pada perangkat dengan sumber daya komputasi terbatas.

#### 4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa proses evaluasi terhadap tiga arsitektur model, yaitu MobileNetV2, MobileNetV3 varian *Large dan Small*, menunjukkan bahwa adanya perbedaan signifikan dalam performa klasifikasi penyakit retina. Penelitian ini menguji variasi *optimizer* AdamW dan SGD, serta *batch size* 16 dan 32, dengan *learning rate* 0,001 dan jumlah *epoch* sebanyak 50. Hasil pengujian menunjukkan bahwa arsitektur MobileNetV2 memberikan performa paling konsisten dan lebih unggul dibandingkan kedua arsitektur lainnya. Konfigurasi terbaik ditemukan pada penggunaan MobileNetV2 dengan *optimizer* AdamW maupun SGD pada *batch size* 16, yang berhasil mencapai akurasi tertinggi sebesar 85,75%. Selain akurasi, nilai *precision*, *recall*, dan *F1-score* juga menunjukkan stabilitas kinerja model pada konfigurasi tersebut. Sebaliknya, sebagian besar kombinasi pada *batch size* 32 serta pada arsitektur MobileNetV3-Small dan MobileNetV3-Large menghasilkan performa yang lebih rendah dan kurang stabil. Jika dibandingkan dengan penelitian terdahulu, beberapa studi sebelumnya mampu mencapai akurasi yang jauh lebih tinggi, bahkan berada pada kisaran 97%–99% ketika menggunakan arsitektur MobileNet maupun model CNN yang lebih berat seperti VGG16 dan InceptionV3. Tingginya akurasi pada penelitian tersebut dipengaruhi oleh ukuran dataset yang lebih besar, serta teknik augmentasi atau *fine tuning* yang lebih besar. Meskipun akurasi yang diperoleh dalam penelitian ini lebih rendah, hasil tersebut menunjukkan bahwa model ringan seperti MobileNetV2 tetap mampu memberikan performa yang stabil dengan kebutuhan komputasi yang lebih rendah. Hal ini menjadikan model yang dikembangkan lebih sesuai untuk skenario dengan keterbatasan sumber daya, seperti sistem berbasis perangkat *mobile* atau lingkungan klinis dengan infrastruktur yang memiliki komputasi terbatas. Oleh karena itu, akurasi sebesar 85,75% pada penelitian ini dijadikan sebagai indikator kinerja awal model dalam mendukung proses klasifikasi citra OCT secara otomatis dengan kebutuhan komputasi yang lebih efisien. Secara keseluruhan, hasil penelitian ini menunjukkan bahwa MobileNetV2 dengan *optimizer* AdamW dan SGD, serta *batch size* 16 merupakan kombinasi terbaik untuk membantu mengklasifikasikan penyakit retina. Penelitian ini juga menunjukkan bahwa pemilihan arsitektur, teknik preprocessing seperti bilateral filter meningkatkan kualitas citra, dan *hyperparameter* yang tepat memiliki peran penting dalam meningkatkan efektivitas model *deep learning* pada bidang deteksi citra medis. Untuk penelitian selanjutnya, disarankan menguji model dengan menggunakan arsitektur CNN lebih berat daripada MobileNet yang ringan, seperti ResNet, DenseNet, Xception, dan banyak lainnya untuk melihat seberapa baik model dalam mengklasifikasikan penyakit retina pada citra OCT.

#### REFERENCES

- [1] A. E. Suwanda and D. Juniati, "Klasifikasi Penyakit Mata Berdasarkan Citra Fundus Retina Menggunakan Dimensi Fraktal Box Counting Dan Fuzzy K-Means," *Jurnal Penelitian Matematika dan Pendidikan matematika*, vol. 5, no. 1, pp. 10–18, 2022, doi: 10.30605/proximal.v5i1.1623.
- [2] P. N. Zakiya, L. Novamizanti, and S. Rizal, "Klasifikasi Patologi Makula Retina Melalui Citra Oct Menggunakan Convolutional Neural Network Dengan Arsitektur MobileNet," *e-Proceeding Eng.*, vol. 8, no. 5, pp. 5072–5082, 2021.



- [3] P. Riazi Esfahani *et al.*, “Deep Learning Classification of Drusen, Choroidal Neovascularization, and Diabetic Macular Edema in Optical Coherence Tomography (OCT) Images,” *Cureus*, vol. 15, no. 7, 2023, doi: 10.7759/cureus.41615.
- [4] A. P. Sunija, S. Kar, S. Gayathri, V. P. Gopi, and P. Palanisamy, “OctNET: A Lightweight CNN for Retinal Disease Classification on Optical Coherence Tomography Images,” *Comput. Methods Programs Biomed.*, p. 105877, 2020, doi: 10.1016/j.cmpb.2020.105877.
- [5] F. A. Shidik, K. Mustofa, A. P. Kartiningtyas, and T. Agustin, “Analisis Citra Medis Untuk Identifikasi Penyakit Mata Dengan Teknologi Convolutional Neural Networks,” *Semin. Nas. Amikom Surakarta 2024*, vol. 2, pp. 68–80, 2024, [Online]. Available: <https://ojs.amikomsolo.ac.id/index.php/semnasa/article/view/648>
- [6] F. M. Talaat, A. A. A. Ali, R. ElGendy, and M. A. ELShafie, “Deep attention for enhanced OCT image analysis in clinical retinal diagnosis,” *Neural Comput. Appl.*, vol. 37, no. 2, pp. 1105–1125, 2025, doi: 10.1007/s00521-024-10450-5.
- [7] A. Helene, K. N. Anna, and S. Vergmann, “Clinical Utilisation of Wide-Field Optical Coherence Tomography and Angiography : A Narrative Review,” *Ophthalmol. Ther.*, vol. 13, no. 4, pp. 903–915, 2024, doi: 10.1007/s40123-024-00905-2.
- [8] C. S. Langlo, A. Amin, and S. S. Park, “Optical coherence tomography retinal imaging : narrative review of technological advancements and clinical applications,” *Emerg. Med. Technol. Innov. Transl.*, vol. 13, no. 2, pp. 1–16, 2025, doi: 10.21037/atm-24-211.
- [9] R. K. Rasel, F. Wu, M. Chiariglione, S. S. Choi, N. Doble, and X. R. Gao, “Assessing the efficacy of 2D and 3D CNN algorithms in OCT - based glaucoma detection,” *Sci. Rep.*, vol. 14, pp. 1–9, 2024, doi: 10.1038/s41598-024-62411-6.
- [10] R. K. Dinata and N. Hasdyna, *Machine Learning. Lhokseumawe, Indonesia: Unimal Press*, 2020.
- [11] F. F. Maulana and N. Rochmawati, “Klasifikasi Citra Buah Menggunakan Convolutional Neural Network,” *J. Informatics Comput. Sci.*, vol. 1, no. 02, pp. 104–108, 2020, doi: 10.26740/jinacs.v1n02.p104-108.
- [12] D. Marcella, Y. Yohannes, and S. Devella, “Klasifikasi Penyakit Mata Menggunakan Convolutional Neural Network Dengan Arsitektur VGG-19,” *J. Algoritm.*, vol. 3, no. 1, pp. 60–70, 2022, doi: 10.35957/algorithm.v3i1.3331.
- [13] P. Powrozniak, M. Skublewska-paszkowska, R. Rejdak, and K. Nowomiejska, “Automatic Method Of Macular Diseases Detection Using Deep CNN-GRU Network in OCT Images,” *Acta Mech. Autom.*, vol. 18, no. 4, pp. 197–206, 2024, doi: 10.2478/ama-2024-0074.
- [14] A. Errabih *et al.*, “Identifying Retinal Diseases on OCT Image Based on Deep Learning,” *Int. J. online Biomed. Eng.*, vol. 18, no. 15, pp. 141–159, 2022, doi: 10.3991/ijoe.v18i15.33639.
- [15] D. Hanumanula *et al.*, “The development of an accurate and computationally feasible MobileNetV2 algorithm to diagnose retinal diseases,” *Int. J. Soc. Sci. Econ. Res.*, vol. 07, no. 09, pp. 3186–3191, 2022, doi: 10.46609/IJSSER.2022.v07i09.028.
- [16] S. Imran, A. L U, D. M S, T. Sharma, and T. Khanum, “Deep Learning-Based Classification of Retinal Diseases from OCT Images with LLM-Powered Patient Query Support,” *Int. J. Heal. Sci. Res.*, vol. 15, no. 7, pp. 314–323, 2025, doi: 10.52403/ijhsr.20250738.
- [17] W. Maximilliano and N. Rachmat, “Comparative Analysis of MobileNetV3-Large and Small for Corn Leaf Disease Classification,” *Brill. Res. Artif. Intell.*, vol. 5, no. 1, pp. 325–332, 2025, doi: 10.47709/brilliance.v5i1.6259.
- [18] S. Selvakumari and M. Durairaj, “A Comparative Study of Optimization Techniques in Deep Learning Using the MNIST Dataset,” *Indian J. Sci. Technol.*, vol. 18, no. 10, pp. 803–810, 2025, doi: 10.17485/IJST/v18i10.121.
- [19] L. Hakim, A. Sobri, L. Sunardi, and D. Nurdiansyah, “Prediksi penyakit jantung berbasis mesin learning dengan menggunakan metode K-NN,” *J. Digit. Teknol. Inf.*, vol. 07, no. 02, pp. 14–20, 2025.
- [20] S. A. Hicks *et al.*, “On evaluation metrics for medical applications of artificial intelligence,” *Sci. Rep.*, vol. 12, no. 1, pp. 1–9, 2022, doi: 10.1038/s41598-022-09954-8.