

Comparison of ResNet-50, EfficientNet-B1, and VGG-16 Algorithms for Cataract Eye Image Classification

Ilham Santoso¹, Ayub Michaelangelo Manurung², Egia Rosi Subhiyanto³

^{1,2,3}Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang

³Research Center of Intelligent Distributed Surveillance and Security (IDSS), Universitas Dian Nuswantoro, Semarang
111202113804@mhs.dinus.ac.id¹, 111202113797@mhs.dinus.ac.id², egia@dsn.dinus.ac.id³

Article Info

Article history:

Received 2024-11-26

Revised 2025-01-09

Accepted 2025-01-21

Keyword:

ResNet-50,

EfficientNet-B1,

Classification,

Eye,

Cataract.

ABSTRACT

Cataract is a leading cause of blindness worldwide, emphasizing the need for an effective early detection approach. This study evaluates the capabilities of three widely-used deep learning models—ResNet-50, EfficientNet-B1, and VGG-16—in classifying visual data. The analysis was conducted on a dataset of 2,112 images, comprising 1,074 normal cases and 1,038 cataract cases. The findings reveal that ResNet-50 achieved the best accuracy at 98.61%, followed by EfficientNet-B1 at 96.64% and VGG-16 at 93.82%. In comparison, previous research using Convolutional Neural Network (CNN) techniques reported an accuracy of 92.93%. These results highlight ResNet-50's superior potential for image classification tasks in this domain. This study contributes significantly to the selection of robust models for building an automated cataract detection framework.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Katarak merupakan salah satu penyebab kebutaan yang masih dapat dicegah, terutama pada kelompok usia lanjut. Sindrom ini disebabkan oleh kekeruhan lensa, yang pada dasarnya menghentikan cahaya mencapai retina. Organisasi Kesehatan Dunia (WHO) memperkirakan bahwa katarak menyebabkan sekitar setengah dari semua kasus kebutaan di seluruh dunia. Di Indonesia, penyebab kebutaan yang paling sering terjadi adalah perkembangan katarak. Menurut sebuah studi yang dilakukan di beberapa lokasi antara tahun 2014 dan 2016, katarak menyumbang sekitar dua pertiga dari semua kasus kebutaan dan gangguan penglihatan yang dilaporkan secara nasional. Biasanya, pengobatan kebutaan akibat katarak dimulai dengan kunjungan ke dokter untuk persiapan operasi. Gejala pasien membantu menentukan jenis katarak yang mereka miliki. Menurut Perhimpunan Dokter Spesialis Mata Indonesia, negara ini memiliki sekitar 3.000 dokter spesialis mata. Namun, jumlah ini tidak cukup untuk mengendalikan volume besar kasus katarak [1].

Berkat perkembangan teknologi, teknologi pemrosesan citra semakin banyak digunakan untuk mencapai berbagai tujuan. ResNet50 adalah metode pembelajaran mendalam yang digunakan dalam klasifikasi di mana setiap unit dan

output dihubungkan oleh jaringan dengan bobot yang disesuaikan pada setiap tingkat pemrosesan. Menggunakan shortcut [2], arsitektur CNN ResNet50 menyelesaikan masalah vanishing gradient dan memungkinkan jaringan yang lebih dalam tanpa mempengaruhi kinerja model. ResNet50 memperkenalkan konsep pembelajaran residual, yang mendukung proses pelatihan jaringan dengan kedalaman lebih besar tanpa penurunan performa [3].

Dikembangkan oleh Simonyan dan Zisserman, VGG-16 menekankan pentingnya jaringan dalam yang memiliki 16 lapisan menggunakan konvolusi 3x3 dan pooling 2x2 meskipun memiliki kekurangan dalam hal konsumsi memori dan jumlah parameter yang besar. (138 million) [4]. Arsitektur sederhana VGG-16 terkenal karena memungkinkan ekstraksi fitur yang efisien dan meningkatkan akurasi serta kinerja model [3]. Menurut Peneliti EfficientNet-B1 lebih efisien karena EfficientNetB1-nya dapat menganalisis gambar hingga 150x150 piksel dengan hanya 6,6 juta parameter dan menyeimbangkan ukuran jaringan melalui skala kompaun [5].

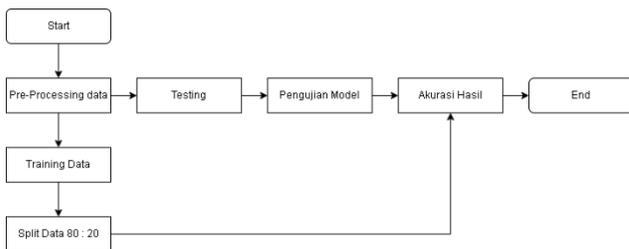
Penelitian ini terinspirasi dari penelitian sebelumnya dalam deteksi mata katarak menggunakan Convolutional Neural Network dengan dua dataset, Normal dan Katarak, penulis karya ini juga menggunakan dua dataset. Dalam

penelitian tersebut, [1] dicapai akurasi maksimum sebesar 92,93%. Banyak penelitian tentang klasifikasi mata katarak juga telah menggunakan sejumlah algoritma, seperti ResNet-50 [6], Convolutional Neural Network (CNN) [7], VGG-16 [8], Support Vector Machine (SVM) [9], [10] Inception-V3 [11], dan K-Nearest Neighbor (KNN) [12], juga telah digunakan dalam sejumlah studi tentang klasifikasi mata katarak.

Studi tambahan tentang identifikasi katarak termasuk [13], yang mencapai tingkat akurasi 96,63% menggunakan transfer learning dengan ResNet50 pada gambar fundus. Penelitian oleh [14] menggunakan model CNN untuk klasifikasi katarak dari citra fundus, dengan akurasi 97,66%. Penelitian oleh [15] menggunakan CNN untuk deteksi otomatis katarak pada citra fundus menggunakan Adam Optimizer mencapai Akurasi 99,13%. Penelitian oleh [16] menggunakan model Invariant of Inception V4 untuk klasifikasi empat jenis citra mencapai akurasi 96%. [17] mencapai tingkat akurasi 98,33% menggunakan Inception-V3 untuk identifikasi katarak menggunakan gambar fundus. [18]. Penelitian dari [19] mengembangkan skema klasifikasi katarak menjadi enam kategori menggunakan GLCM, ResNet, dan SVM, dengan akurasi rata-rata 91,5%. [20] mencapai tingkat akurasi 91,76% dari 170 data pasien dengan menggunakan K-Nearest Neighbor (KNN) untuk mengkategorikan jenis katarak berdasarkan gejala pasien.

Penelitian ini berfokus pada penerapan metode pembelajaran mendalam untuk mengklasifikasikan katarak dengan pendekatan yang lebih efisien, memanfaatkan arsitektur ResNet50, VGG-16, dan EfficientNetB1. Kontribusi penelitian ini terletak pada penggunaan dataset terkini dan optimasi teknik transfer learning untuk meningkatkan akurasi deteksi katarak.

II. METODE

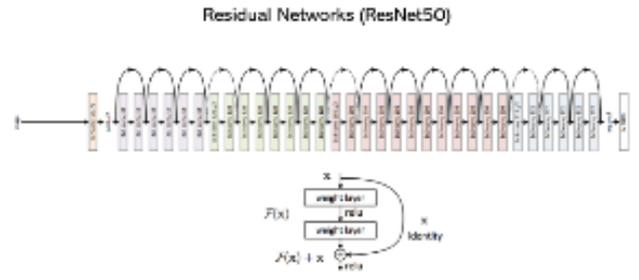


Gambar 1. FlowChart Diagram Pengembangan Sistem

A. ResNet-50

ResNet-50 adalah salah satu arsitektur jaringan saraf konvolusional (CNN) dengan 50 lapisan, yang pertama kali diperkenalkan oleh Kaiming He dan timnya pada tahun 2015 melalui makalah berjudul “Deep Residual Learning for Image Recognition”. Arsitektur ini menggunakan blok residual yang dirancang untuk memungkinkan aliran informasi melewati beberapa lapisan menggunakan koneksi shortcut. Pendekatan ini secara efektif mengatasi tantangan gradien yang menghilang, sehingga memungkinkan pelatihan model

dengan kedalaman jaringan yang sangat signifikan tanpa penurunan performa [21].



Gambar 2. ResNet-50

III. HASIL DAN PEMBAHASAN

Pada Tabel I, dijelaskan konfigurasi hyperparameter utama, meliputi jumlah lapisan konvolusi dan dense yang masing-masing terdiri dari dua lapisan, penggunaan fungsi loss Categorical Crossentropy untuk mengukur kesalahan klasifikasi, serta aktivasi ReLU di lapisan tersembunyi dan Softmax di lapisan luar untuk menghasilkan nilai probabilitas kelas. Optimizer yang digunakan adalah Adam, dengan proses pelatihan dilakukan selama 25 epoch menggunakan ukuran batch sebesar 64. Pengaturan ini dirancang untuk meningkatkan performa model dalam klasifikasi multi-kelas.

TABEL I
HYPERPARAMETER MODEL

Parameter	Value
Lapisan Konvolusi	2
Lapisan Dense	2
Loss Fungsi	Categorical Crosstropy
Aktivasi Fungsi	Untuk lapisan tersembunyi ReLU dan Softmax untuk lapisan luar
Optimizer	Adam
Jumlah Epoch	25
Ukuran Batch	64

TABEL II
HYPERPARAMETER MODEL

Layer(Type)	Output Shape	Param#
Input Layer	(None, 150, 150, 3)	0
ResNet-50 (Include_top=False)	(None, 5, 5, 2048)	23,587,712
BatchNormalization	(None, 2048)	8,192
Dense(128 units)	(None, 128)	262,272
Dropout	(None, 128)	0
Output Layer (Dense, 2 classes)	(None, 2)	258
Total Parameters	-	23,858,343
Trainable Parameters	-	23,824,386

Pada Tabel II memberikan rincian struktur arsitektur ResNet-50. Model ini memanfaatkan bobot pre-trained dengan menonaktifkan bagian atasnya (Include_top=False) agar sesuai dengan tugas klasifikasi dua kelas. Tabel tersebut

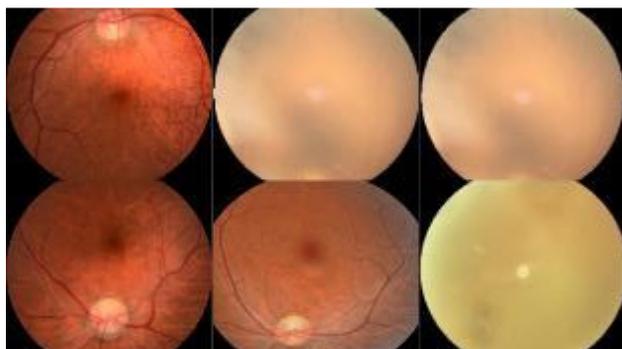
menjelaskan jumlah parameter pada setiap lapisan, termasuk pada lapisan ResNet-50, normalisasi batch, dense, dropout, dan output. Total parameter model berjumlah 23.858.434, di mana 23.856.434 di antaranya merupakan parameter yang dapat dilatih, sedangkan 2.000 lainnya tidak dapat dilatih. Kombinasi arsitektur dan konfigurasi hyperparameter ini dirancang untuk memastikan model bekerja secara optimal dalam tugas klasifikasi.

B. Dataset

Dataset yang digunakan dalam penelitian ini diambil dari Kaggle, yang terbagi menjadi dua kategori utama yaitu cataract dan normal. Subdirektori cataract memuat 1.038 gambar mata yang menunjukkan gejala katarak, sedangkan subdirektori normal memuat 1.074 gambar mata normal tanpa katarak. Total keseluruhan dataset berjumlah 2.112 gambar, yang kemudian dibagi menjadi dua bagian utama : data pelatihan sebanyak 80% (1.690 gambar) dan data validasi sebanyak 20% (422 gambar). Meskipun dataset mencakup data yang cukup beragam, penggunaannya tetap terbatas pada dataset yang tersedia secara public, sehingga tidak dapat mewakili sepenuhnya variabilitas kondisi dunia nyata. Oleh karena itu, diperlukan evaluasi lebih lanjut pada dataset eksternal untuk menguji generalisasi model.

TABEL III
SUMBER DATASET DAN DETAIL

No	Detail Dataset
1	Nama : eye_diseases_classification Banyak data : 2.112 Citra Class : Normal (1.038), Cataract (1.074) URL : eye_diseases_classification



Gambar 2. Contoh Dataset Normal dan Katarak

C. Pre-processing

Preprocessing data merupakan langkah penting dalam penelitian ini untuk memastikan bahwa data siap digunakan dalam pelatihan model deep learning[22]. Dataset yang terdiri dari dua kategori, yaitu cataract dan normal, diproses melalui beberapa tahapan untuk meningkatkan kualitas data dan efisiensi pelatihan. Tahap pertama melibatkan penyesuaian

ukuran gambar menjadi 150x150 piksel menggunakan parameter `image_size=(150, 150)` pada fungsi `image_dataset_from_directory`. Penyesuaian ini bertujuan untuk menyamakan dimensi input sesuai kebutuhan model yang digunakan, sekaligus mengurangi beban komputasi selama pelatihan.

Tahapan berikutnya adalah normalisasi data, di mana nilai piksel pada setiap gambar diskalakan ke rentang 0 hingga 1 dengan menggunakan fungsi `rescale=1./255`. Normalisasi ini membantu model mempelajari pola data lebih stabil dan efisien dengan menghindari masalah seperti exploding atau vanishing gradient selama proses optimasi. Gambar yang telah dinormalisasi juga memastikan model tidak terpengaruh oleh nilai piksel yang bervariasi secara ekstrem.

Selanjutnya, dataset dibagi menjadi dua bagian, yaitu **training set** dan **validation set**, dengan perbandingan 80:20, tergantung pada jumlah total data yang tersedia. Pembagian ini bertujuan agar model dapat dilatih menggunakan training set dan dievaluasi pada validation set untuk mengukur kinerjanya sebelum diterapkan pada data baru. Data training digunakan untuk melatih model, sementara data validasi digunakan untuk mengevaluasi kinerja model terhadap data yang tidak dilibatkan dalam proses pelatihan[22]. Pembagian ini dilakukan secara acak untuk memastikan kedua subset memiliki distribusi data yang representatif.

Proses preprocessing ini dirancang untuk memberikan data yang konsisten dari segi ukuran, skala, dan distribusi. Dengan tahapan ini, model dapat lebih mudah mempelajari pola yang relevan tanpa terganggu oleh ketidakkonsistenan dalam data. Implementasi preprocessing dilakukan bertujuan meningkatkan kemampuan model dalam mengklasifikasikan gambar. Berikut pseudocode untuk pre-processing data :

```
BEGIN
// Inisialisasi ImageDataGenerator untuk
normalisasi
SET train_datagen =
ImageDataGenerator(rescale=1./255)
SET validation_datagen =
ImageDataGenerator(rescale=1./255)
// Muat data training dengan augmentasi dari
direktori
SET train_generator =
train_datagen.flow_from_directory(
    directory='dataset2',
    target_size=(150, 150), // Ukuran gambar
menjadi 150x150
    batch_size=64, // Ukuran batch
    class_mode='categorical' // Mode
klasifikasi kategori
)
// Muat data validasi dengan augmentasi dari
direktori
SET validation_generator =
validation_datagen.flow_from_directory(
    directory='dataset2',
    target_size=(150, 150), // Ukuran gambar
menjadi150x150
```

```

batch_size=64, // Ukuran batch
class_mode='categorical' // Mode
klasifikasi kategori
) END

```

```

// Visualisasi matriks kebingungan
PLOT confusion_matrix(conf_matrix_test)
END

```

D. Evaluasi Model ResNet-50

Evaluasi model ResNet-50 dalam penelitian ini dilakukan untuk mengukur kinerjanya dalam tugas klasifikasi serta kemampuannya untuk menggeneralisasi data baru yang belum pernah dilihat sebelumnya. Untuk keperluan ini, data uji yang berasal dari dataset pelatihan digunakan dalam proses evaluasi akhir. Confusion matrix digunakan sebagai alat utama untuk menganalisis performa model, karena memungkinkan visualisasi keberhasilan dan kesalahan klasifikasi dengan cara membandingkan prediksi model terhadap label sebenarnya. Pseudocode berikut menggambarkan langkah-langkah evaluasi model :

```

BEGIN
// Evaluasi model menggunakan data validasi
SET validation_loss, validation_accuracy =
model.evaluate(
validation_generator
)
// Cetak hasil evaluasi
PRINT "Validation Loss:", validation_loss
PRINT "Validation Accuracy:",
validation_accuracy
// Evaluasi model menggunakan data testing
SET test_generator =
ImageDataGenerator(rescale=1./255).flow_from_directory(
directory='dataset2',
target_size=(150, 150), // Ukuran gambar
menjadi 150x150
batch_size=64, // Ukuran batch
class_mode='categorical',
shuffle=False
)
// Prediksi data testing
SET Y_pred_test = model.predict(test_generator)
SET y_pred_test = np.argmax(Y_pred_test,
axis=1)
// Dapatkan label sebenarnya dari data testing
SET y_true_test = test_generator.classes
// Buat laporan klasifikasi
PRINT "Classification Report:"
PRINT classification_report(y_true_test,
y_pred_test,

target_names=test_generator.class_indices.keys(
))
// Hitung matriks kebingungan
SET conf_matrix_test =
confusion_matrix(y_true_test, y_pred_test)
PRINT "Confusion Matrix:"
PRINT conf_matrix_test

```

Evaluasi efektivitas model sangat bergantung pada penggunaan metrik seperti akurasi, presisi, recall, dan F1-score. Berdasarkan hasil evaluasi, model ResNet-50 berhasil mencapai akurasi tinggi atas 98%, dengan presisi, recall, dan F1-score yang juga konsisten pada level tinggi. Confusion matrix mengungkapkan bahwa sebagian besar data diuji dengan benar, meskipun terdapat beberapa kesalahan klasifikasi. Pada kelas cataract, sebanyak 37 gambar salah diklasifikasikan sebagai normal, sementara pada kelas normal, hanya 2 gambar yang salah diklasifikasikan sebagai cataract.

Secara keseluruhan, model ResNet-50 menunjukkan performa yang sangat baik dalam mengklasifikasikan data dengan akurasi tinggi. Hal ini mencerminkan kemampuan model dalam meminimalkan kesalahan prediksi, sehingga layak untuk digunakan sebagai alat pendukung pengambilan keputusan dalam aplikasi deteksi katarak. Namun, ResNet-50 adalah model deep learning yang cukup kompleks, yang berarti ia memiliki banyak lapisan dan parameter, sehingga membutuhkan lebih banyak waktu untuk melakukan inferensi dibandingkan dengan model yang lebih sederhana.

III. HASIL DAN PEMBAHASAN

A. Pembagian Dataset

Proses pembagian dataset dilakukan menggunakan fungsi ImageDataGenerator dari Keras, yang memungkinkan preprocessing gambar sekaligus membagi data menjadi subset pelatihan dan validasi secara otomatis. Normalisasi gambar dilakukan dengan parameter rescale=1./255, yang mengonversi nilai piksel dari rentang [0, 255] menjadi [0, 1]. Dataset dipecah menjadi dua bagian utama, yaitu 80% dialokasikan untuk data pelatihan, sedangkan 20% sisanya untuk validasi.

Metode flow_from_directory digunakan untuk membaca gambar secara langsung dari direktori utama (dataset2) dan mengelompokkan data berdasarkan subdirektori yang mencerminkan label kelas. Subset data pelatihan dan validasi ditentukan melalui parameter subset='training' dan subset='validation', tanpa perlu memisahkan data secara manual. Setiap gambar diubah ukurannya menjadi 150x150 piksel, dengan ukuran batch sebanyak 64 gambar per iterasi. Mode klasifikasi yang digunakan adalah kategori (class_mode='categorical'), dan data diacak secara otomatis (shuffle=True) untuk mengurangi bias selama proses.

Pendekatan ini memberikan efisiensi dalam pembagian dataset serta mempermudah integrasi dengan pipeline pelatihan model. Untuk memastikan pembagian dataset dilakukan dengan benar, jumlah data pada subset pelatihan dan validasi juga diperiksa dan dicetak

B. Dataset Pelatihan dan Validasi

Proses pelatihan dan validasi dimulai dengan mendesain model menggunakan arsitektur Sequential, yang mencakup lapisan penting seperti konvolusi, pooling, Flatten, dan Dense. Model ini kemudian dikompilasi menggunakan fungsi loss categorical_crossentropy, optimizer Adam, serta metrik akurasi untuk mengevaluasi performanya. Selama pelatihan, data disediakan melalui generator, yang memastikan data diolah dalam batch dan urutannya diacak untuk mengurangi bias. Untuk mencegah overfitting, digunakan callback EarlyStopping, yang secara otomatis menghentikan pelatihan jika akurasi validasi tidak mengalami peningkatan setelah lima epoch berturut-turut.

Langkah-langkah teknis dari proses pelatihan dan validasi ini dapat digambarkan dalam bentuk pseudocode berikut :

```
BEGIN
// Inisialisasi dataset utama
LOAD main_dataset = 'dataset2'
// Tentukan parameter pembagian dataset
SET validation_split = 0.2 // 20% untuk
validasi
SET batch_size = 64
SET image_size = (150, 150) // Ukuran gambar
// Muat data training dan validasi dari
direktori utama
SET train_dataset =
LOAD_IMAGE_DATASET_FROM_DIRECTORY(
main_dataset,
SUBSET='training',
validation_split=validation_split,
shuffle=True,
seed=123,
image_size=image_size,
batch_size=batch_size
)
SET validation_dataset =
LOAD_IMAGE_DATASET_FROM_DIRECTORY(
main_dataset,
SUBSET='validation',
validation_split=validation_split,
shuffle=True,
seed=123,
image_size=image_size,
batch_size=batch_size
)
// Hitung jumlah total gambar
SET total_images = COUNT_IMAGES(main_dataset)
PRINT "Total images in dataset:",
total_images
// Hitung jumlah gambar dalam training dan
validasi
SET train_images =
COUNT_IMAGES(train_dataset)
SET validation_images =
COUNT_IMAGES(validation_dataset)
PRINT "Training images:", train_images
PRINT "Validation images:", validation_images
RETURN train_dataset, validation_dataset
END
```

Hasil pelatihan menunjukkan bahwa model mampu mencapai akurasi validasi yang optimal tanpa indikasi overfitting. Penggunaan callback EarlyStopping membantu menghentikan pelatihan sebelum performa model mulai menurun. Selain itu, riwayat pelatihan disimpan dalam objek history, memungkinkan analisis lebih lanjut seperti pembuatan grafik akurasi dan loss pada setiap epoch. Dengan pendekatan ini, model dapat belajar secara optimal sambil mempertahankan kemampuan generalisasi yang baik.

C. Model ResNet-50

Evaluasi kinerja model merupakan langkah penting dalam menentukan keberhasilan model klasifikasi. Dalam studi ini, digunakan arsitektur ResNet-50 sebelumnya melalui proses pelatihan dataset ImageNet sebagai fonasi model. Lapisan tambahan ditambahkan di atas ResNet-50 untuk menyesuaikan tugas klasifikasi dua kelas, yaitu cataract dan normal. Selama pelatihan, performa model dipantau melalui metrik akurasi dan loss, yang divisualisasikan dalam bentuk grafik untuk mendapatkan gambaran lebih jelas tentang proses pembelajaran model.

Berikut merupakan pseudocode untuk melakukan evaluasi terhadap model ResNet-50:

```
BEGIN

// Inisialisasi ResNet-50 tanpa lapisan
atas
(include_top=False)
SET base_model =
ResNet50(weights='imagenet',
include_top=False, input_shape=(150, 150, 3))

// Tambahkan lapisan kustom di atas model
dasar ResNet-50
SET x = base_model.output
x = BatchNormalization(axis=-1,
momentum=0.99, epsilon=0.001)(x)
x = Dense(128,
kernel_regularizer=l2(0.070),
activity_regularizer=l1(0.008),
bias_regularizer=l1(0.008),
activation='relu')(x)
x = Dropout(rate=0.3)(x)
output = Dense(2,
activation='softmax')(x)

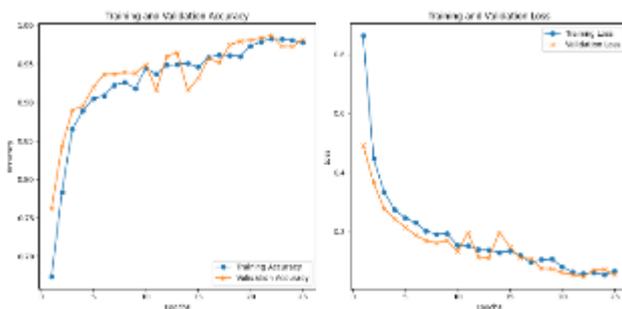
// Definisikan model akhir dengan input
dari ResNet-50 dan output yang telah
dimodifikasi
SET model =
Model(inputs=base_model.input,
outputs=output)

// Kompilasi model
```

```
CALL model.compile(
    loss='categorical_crossentropy',
    optimizer=Adam(learning_rate=0.0001),
    metrics=['accuracy']
)
END
```

Dengan memanfaatkan kemampuan transfer learning dari ResNet-50, model diharapkan dapat mencapai tingkat akurasi yang lebih tinggi dalam tugas klasifikasi, meskipun dengan jumlah data yang terbatas. Transfer learning memungkinkan model memanfaatkan fitur-fitur penting yang telah dipelajari dari dataset ImageNet, kemudian menyesuaikannya dengan karakteristik dataset spesifik. Selama proses pelatihan, metrik akurasi dan loss dipantau untuk memastikan model berjalan secara optimal dan menghindari overfitting. Visualisasi hasil pelatihan, seperti grafik akurasi dan loss, memberikan pemahaman yang lebih mendalam mengenai kinerja model, sehingga proses klasifikasi dapat dioptimalkan secara lebih baik.

1) Model ResNet-50



Gambar 3. Grafik Validasi Akurasi dan Loss Model ResNet-50

Pada Gambar 3, hasil pelatihan model deep learning menunjukkan peningkatan akurasi dan penurunan loss secara konsisten selama 25 epoch. Pada epoch pertama, akurasi pelatihan tercatat sebesar 67,38% dengan nilai loss 0,86, sementara akurasi validasi mencapai 76,18% dengan loss 0,49. Di akhir pelatihan (epoch 25), akurasi pelatihan meningkat menjadi 97,77% dengan loss 0,06, sedangkan akurasi validasi mencapai 98,15% dengan loss 0,05. Grafik menunjukkan pola serupa antara nilai akurasi dan loss pada data pelatihan dan validasi, yang menunjukkan bahwa model belajar secara efektif serta memiliki kemampuan generalisasi yang baik.. Model ini terbukti akurat dalam mengklasifikasikan gambar dengan performa yang stabil.

D. Hasil Analisis Evaluasi Model ResNet-50

Pada data uji, model menghasilkan akurasi **98.15%**, dengan presisi, tingkat pengembalian kembali, dan score F1 dengan rata-rata sebesar **98%**. Hasil ini mengindikasikan model mampu mengenali kelas dengan baik. Precision mencapai **100%** untuk kelas cataract dan **97%** untuk kelas normal. Recall masing-masing berada di **96%** untuk cataract

dan **100%** untuk normal, menandakan kemampuan model dalam mendeteksi hampir seluruh data dengan akurat.

Untuk mengurangi risiko overfitting, ada beberapa strategi yang dapat diterapkan. Regularisasi, seperti L2 regularization atau Dropout, membantu mencegah model terlalu menyesuaikan diri dengan data pelatihan. Augmentasi data juga dapat meningkatkan keberagaman data yang digunakan, sementara early stopping menghentikan pelatihan saat kinerja pada data validasi menurun. Selain itu, penggunaan k-fold cross-validation dapat memastikan model lebih mampu menggeneralisasi pada berbagai subnet data. Dengan langkah-langkah ini, model dapat dioptimalkan untuk performa yang lebih baik pada data baru.

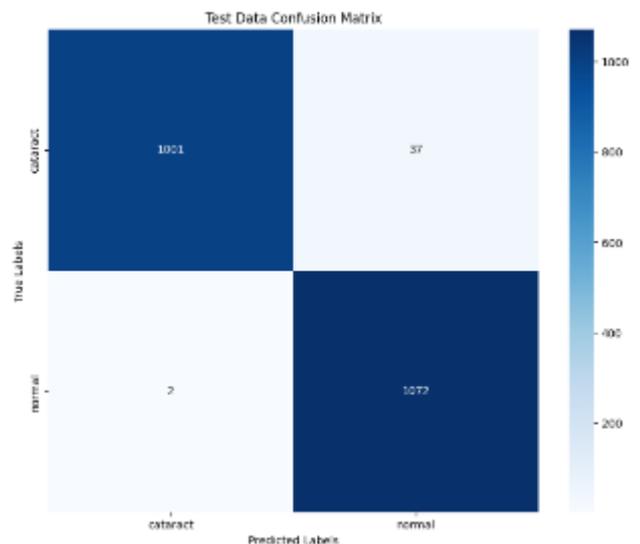
```
Found 7117 images belonging to 2 classes.
33/33 [-----] - 12s 364ms/step
Classification Report
precision    recall  f1-score   support

   cataract    1.00    0.96    0.98    1038
   normal     0.97    1.00    0.98    1074

 accuracy          0.98
 macro avg         0.98
weighted avg         0.98
```

```
Confusion Matrix
[[1001  37]
 [    2 1072]]
```

Gambar 4. Classification Report Metode ResNet-50



Gambar 5. Confusion Matrix Metode ResNet-50

Hasil confusion matrix memperlihatkan bahwa model ResNet-50 mampu secara akurat mengidentifikasi sebagian besar gambar. Sebanyak 1,001 gambar cataract dan 1,072 gambar normal teridentifikasi dengan akurat. Kesalahan klasifikasi yang terjadi sangat kecil, yaitu hanya 37 gambar cataract salah diklasifikasikan sebagai normal dan 2 gambar normal salah diprediksi sebagai cataract. Confusion matrix yang divisualisasikan melalui heatmap memberikan gambaran distribusi prediksi secara intuitif, di mana prediksi

yang benar dominan berada pada diagonal utama. Hal ini menunjukkan bahwa model mampu mengenali kedua kelas dengan konsistensi dan tingkat kesalahan yang rendah.

E. Model EfficientNet-B1

Penggunaan model EfficientNet-B1 dalam penelitian ini didasarkan pada kemampuan model sebagai arsitektur deep learning modern yang sebelumnya telah dilatih menggunakan dataset ImageNet. Model ini diprioritaskan karena kelebihanannya dalam mengelola efisiensi melalui teknik penskalaan kompaun, yang dapat menyeimbangkan lebar, kedalaman, dan resolusi jaringan. Dengan pendekatan ini, EfficientNet-B1 dapat mencapai kinerja tinggi dengan jumlah parameter yang relatif lebih sedikit. Untuk memenuhi tugas klasifikasi dua kelas, yaitu Katarak dan Normal, studi ini menambahkan beberapa lapisan tambahan di atas arsitektur dasar EfficientNet-B1.

Berikut Pseudocode untuk pengujian Model EfficientNet-B1:

```

BEGIN
// Load and preprocess test data
LOAD test_dataset, resize to (150, 150),
normalize pixel
normalize pixel values to [0, 1]

// Load trained EfficientNet-B1 model
LOAD pre-trained model with custom layers

// Predict and evaluate
PREDICT class labels on test_dataset
COMPARE predictions with true labels
CALCULATE accuracy, precision, recall, F1-
score, and confusion matrix

// Visualize results
PLOT confusion matrix
DISPLAY classification report

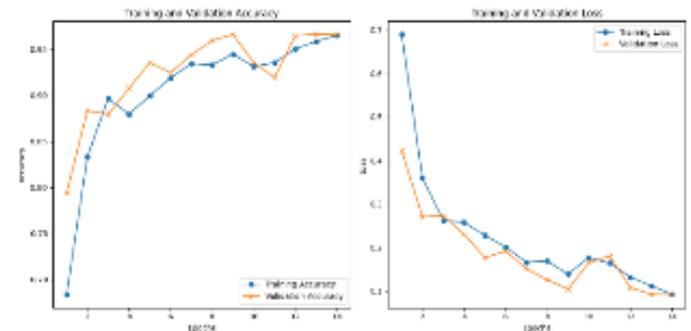
END
    
```

Strategi transfer learning diterapkan untuk mempercepat pelatihan dan mengatasi keterbatasan jumlah data. Model ini memanfaatkan fitur-fitur penting yang dipelajari dari dataset ImageNet untuk menyesuaikan dengan karakteristik dataset spesifik dari studi ini, tanpa perlu merancang arsitektur baru dari awal. Penambahan lapisan tambahan, seperti BatchNormalization, Dense Layer, dan Dropout, dirancang untuk mengurangi risiko overfitting sambil sekaligus meningkatkan kemampuan generalisasi model.

1) Model EfficientNet-B1

Pada Gambar 7 proses pelatihan model deep learning memperlihatkan peningkatan akurasi yang konsisten disertai dengan penurunan nilai loss selama 14 epoch. Pada epoch pertama, akurasi pelatihan mencapai 68,32% dengan nilai loss sebesar 0,69, sedangkan akurasi validasi tercatat sebesar

79,40% dengan loss 0,42. Di akhir pelatihan (epoch ke-14), akurasi pelatihan meningkat hingga 96,58% dengan nilai loss 0,09, sementara akurasi validasi mencapai 96,64% dengan loss yang sama, yaitu 0,09.



Gambar 7. Grafik Validasi Akurasi dan Loss Model EfficientNet-B1

Grafik menunjukkan pola akurasi dan loss yang serupa antara data pelatihan dan validasi, yang menandakan bahwa model mampu belajar secara efektif dan memiliki generalisasi yang baik. Dengan performa yang stabil, model ini dapat diandalkan untuk tugas klasifikasi data secara akurat.

F. Hasil Analisis Evaluasi Model EfficientNet-B1

Model yang diuji pada data pengujian menunjukkan kinerja yang sangat baik, seperti yang terlihat pada Classification Report dan Confusion Matrix. Secara keseluruhan, model sukses mencapai tingkat akurasi 97%, dengan precision, recall, dan f1-score masing-masing bernilai 0,97 untuk rata-rata makro maupun rata rata berbobot. Sedangkan untuk kategori normal, precision mencapai 0,96 , recall 0,98 , dan f1-score 0,97 dari total 1.074 gambar.

```

Found 2112 images belonging to 2 classes.
33/33 [=====] - 17s 509ms/step
Classification Report
      precision    recall  f1-score   support

   cataract      0.98      0.95      0.97      1038
    normal      0.96      0.98      0.97      1074

   accuracy                   0.97      2112
  macro avg      0.97      0.97      0.97      2112
weighted avg      0.97      0.97      0.97      2112

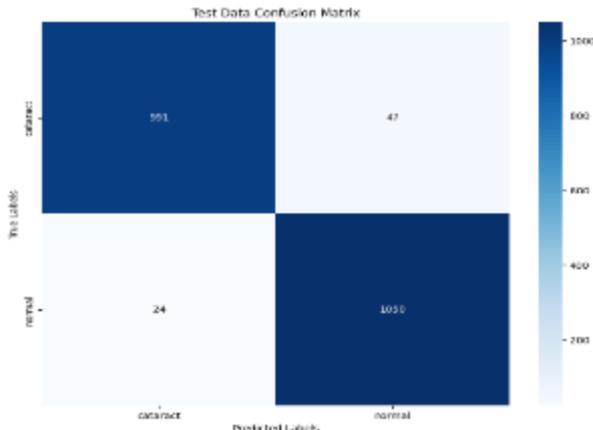
Confusion Matrix
[[ 991  47]
 [ 24 1050]]

Text(95.7222222222221, 0.5, 'True Labels')
    
```

Gambar 8. Classification Report Metode EfficientNet-B1

Berdasarkan Confusion Matrix, sebanyak 991 gambar cataract berhasil diklasifikasikan dengan benar, sementara 47 gambar salah diklasifikasikan sebagai normal. Sebaliknya, sebanyak 1.050 gambar normal diprediksi dengan benar, dengan hanya 24 gambar yang salah diklasifikasikan sebagai cataract. Hasil ini mengindikasikan bahwa model mampu

melakukan generalisasi secara efektif dalam membedakan kedua kategori dengan tingkat kesalahan prediksi yang minimal. Dengan akurasi yang tinggi, model ini dapat diandalkan untuk tugas klasifikasi serupa secara konsisten dan efektif.



Gambar 9. Confusion Matrix Metode EfficientNet-B1

G. Model VGG-16

Pekerjaan ini menggunakan model VGG-16 karena merupakan salah satu arsitektur pembelajaran mendalam yang telah menunjukkan kinerja luar biasa dalam beberapa tantangan kategorisasi gambar. Setelah dilatih sebelumnya pada dataset ImageNet, model ini dapat menggunakan karakteristik yang telah dipelajari sebelumnya untuk meningkatkan kinerja klasifikasi. Beberapa lapisan tambahan ditumpuk di atas desain dasar VGG-16 untuk memenuhi kebutuhan klasifikasi dua kelas: Katarak dan Normal.

Berikut pseudocode untuk pengujian Model VGG-16:

```

BEGIN
// Preprocess dataset
LOAD dataset, resize to (150, 150), normalize
pixel values to [0, 1]
SPLIT dataset into training (80%) and
validation (20%)

// Build and configure VGG-16 model
LOAD pretrained VGG-16 (ImageNet), freeze
base layers
ADD Flatten, Dense (128 units, ReLU), Dropout
(rate=0.3), Output (2 units, Softmax)

// Compile and train model
COMPILE with categorical crossentropy loss,
Adam
optimizer, and accuracy metric
TRAIN model with early stopping (patience=5)
on training and validation data

// Evaluate and test model
EVALUATE model on validation data
PREDICT and compare test results to true
labels
    
```

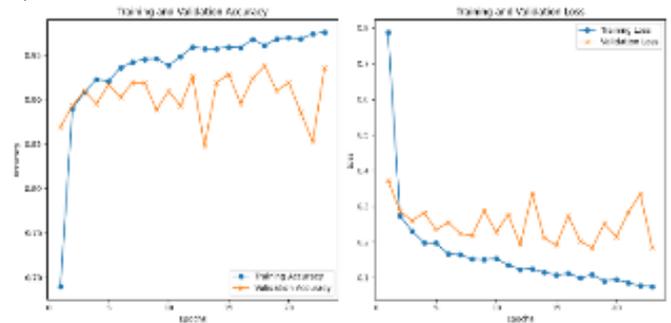
```

CALCULATE classification metrics and
confusion matrix

// Visualize results
PLOT training/validation accuracy and loss
PLOT confusion matrix
END
    
```

Menggunakan arsitektur dasar VGG-16 sebagai ekstraktor fitur, parameter di lapisan dasarnya dibekukan untuk menghentikan modifikasi selama proses pelatihan. Untuk menyesuaikan data yang digunakan dalam penelitian ini, beberapa lapisan tambahan Flatten, Dense, dan Dropout ditambahkan untuk Terutama dalam kasus dengan dataset yang cukup kecil, penambahan lapisan-lapisan ini bertujuan untuk meminimalkan potensi overfitting sekaligus memperkuat kemampuan model dalam melakukan generalisasi.

1) Model VGG-16



Gambar 10. Grafik Validasi Akurasi dan Loss Model VGG-16

Pada Gambar 3, proses pelatihan model deep learning menunjukkan peningkatan akurasi dan penurunan nilai loss secara bertahap selama 25 epoch. Pada epoch pertama, akurasi pelatihan tercatat sebesar **68,95%** dengan loss sebesar **0,78**, sementara akurasi validasi mencapai **86,94%** dengan nilai loss **0,37**. Di akhir pelatihan (epoch ke-25), akurasi pelatihan meningkat menjadi **97,58%** dengan loss **0,07**, sedangkan akurasi validasi mencapai **93,59%** dengan loss **0,18**. Grafik memperlihatkan pola yang konsisten antara nilai akurasi dan loss pada data pelatihan serta validasi, yang mengindikasikan bahwa model dapat mempelajari pola secara efisien dan memiliki kemampuan generalisasi yang optimal. Hasil ini membuktikan bahwa model mampu mengklasifikasikan data dengan akurasi tinggi dan performa yang stabil.

H. Hasil Analisis Evaluasi Model VGG-16

Model yang diuji pada data pengujian menunjukkan kinerja yang sangat baik, seperti yang ditampilkan pada Classification Report dan Confusion Matrix. Model ini berhasil memperoleh akurasi total sebesar 97%, dengan rata-rata precision, recall, dan f1-score sebesar 97%. Pada kategori cataract, model menunjukkan precision 0,96, recall 0,98, serta

f1-score 0,97 dari total 1.038 gambar. Sementara itu, untuk kategori normal, precision yang dihasilkan adalah 0,98, recall 0,96, dan f1-score 0,97 dari total 1.074 gambar.

```

Found 2112 images belonging to 2 classes.
33/33 [-----] - 88s 3s/step
Classification Report
      precision    recall  f1-score   support

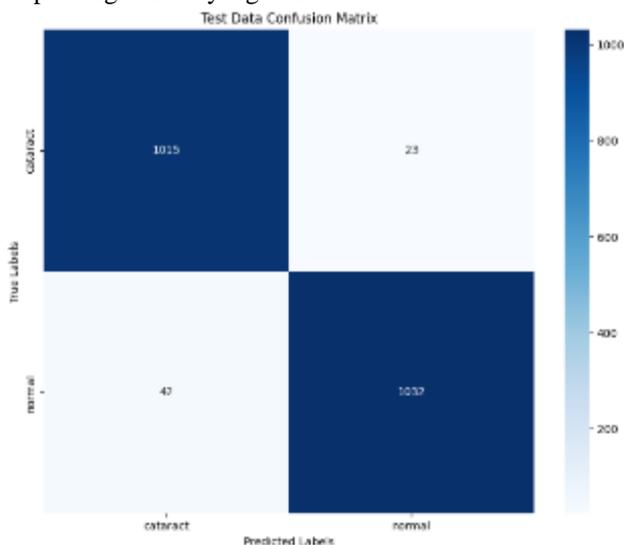
   cataract      0.96      0.98      0.97      1038
    normal      0.98      0.96      0.97      1074

 accuracy      0.97
 macro avg      0.97      0.97      0.97      2112
weighted avg      0.97      0.97      0.97      2112

Confusion Matrix
[[ 1015   23]
 [   42 1032]]
    
```

Gambar 11. Classification Report Metode VGG-16

Berdasarkan Confusion Matrix, model berhasil memprediksi 1.015 gambar cataract dengan benar, sementara 23 gambar salah diklasifikasikan sebagai normal. Selain itu, 1.032 gambar normal diprediksi dengan benar, dan 42 gambar salah diklasifikasikan sebagai cataract. Hasil ini menunjukkan bahwa model memiliki kemampuan klasifikasi yang akurat, dengan tingkat kesalahan prediksi yang rendah serta performa yang stabil. Model ini dapat diandalkan untuk klasifikasi data serupa dengan hasil yang konsisten.



Gambar 12. Confusion Matrix Metode VGG-16

I. Perbandingan Antara Ketiga Model

Perbandingan kinerja model ResNet-50, EfficientNet-B1, dan VGG-16 menunjukkan hasil yang bervariasi berdasarkan metrik evaluasi seperti akurasi, precision, recall, f1-score, dan loss. ResNet-50 menunjukkan kinerja terbaik dengan akurasi 97%, serta precision, recall, dengan f1-score yang juga mencapai 0,97, serta nilai loss validasi terendah, yaitu 0,18. Model ini turut memperlihatkan kemampuan generalisasi yang tinggi dengan pola akurasi dan loss yang stabil baik pada data pelatihan maupun validasi. Namun, ResNet-50

membutuhkan lebih banyak sumber daya komputasi dibandingkan dengan VGG-16 atau EfficientNet-B1, yang bisa menjadi kendala dalam implementasi pada perangkat dengan keterbatasan sumber daya.

TABEL IV
PERNANDINGAN ANTARA KETIGA MODEL

Model	Akurasi	Presisi	Recall	F1-Score
ResNet-50	98,15%	97%	97%	97%
EfficientNet-B1	96,64%	96%	95%	96%
VGG-16	93,82%	94%	93%	94%

EfficientNet-B1 menempati posisi kedua dengan akurasi sebesar 96% dan nilai loss validasi sebesar 0,21. Precision, recall, dan f1-score masing-masing berada di angka 0,96, yang sedikit lebih rendah dibandingkan ResNet-50. Meskipun begitu, model ini lebih efisien secara komputasi model ini bisa saja memiliki sedikit keterbatasan dalam hal kemampuan untuk mengatasi fitur yang sangat kompleks dalam dataset dibandingkan dengan ResNet-50..

Sementara itu, VGG-16 mencatat akurasi terendah sebesar 94%, dengan precision, recall, dan f1-score masing-masing 0,94, serta loss validasi mencapai 0,30. Model ini lebih sederhana dan membutuhkan lebih sedikit sumber daya komputasi, tetapi kinerjanya yang lebih rendah dibandingkan dengan dua model lainnya menunjukkan bahwa VGG-16 mungkin tidak seoptimal dalam menangani tugas klasifikasi yang lebih kompleks.

Secara keseluruhan, ResNet-50 merupakan model yang paling unggul untuk tugas klasifikasi ini, diikuti oleh EfficientNet-B1 yang menawarkan keseimbangan antara efisiensi dan performa. VGG-16, meskipun tertinggal, tetap menjadi opsi yang layak dengan performa yang memadai untuk klasifikasi gambar.

J. Perbandingan Antara Ketiga Model dengan Penelitian Sebelumnya

Perbandingan kinerja antara model ResNet-50, EfficientNet-B1, dan VGG-16 yang diklasifikasikan dalam studi ini dibandingkan dengan model CNN dari penelitian terdahulu. Model ResNet-50 mencapai performa terbaik dengan akurasi sebesar 98,15%, recall 98%, precision 98%, dan F1-score 98%, menunjukkan keunggulannya dalam mengklasifikasikan data secara akurat.

TABEL V
PERBANDINGAN MODEL PENELITIAN SEBELUMNYA

Model	Akurasi	Presisi	Recall	F1-Score
ResNet-50	98,15%	98%	98%	98%
EfficientNet-B1	93,82%	97%	97%	97%
VGG-16	96,64%	97%	97%	97%
CNN	92,93%	-	-	-

EfficientNet-B1 berada di posisi kedua dengan akurasi 96,64%, recall 97%, precision 97%, dan F1-score 97%, yang masih mendekati performa ResNet-50. VGG-16 menghasilkan akurasi 93,82%, yang memiliki nilai recall, precision, dan f1-score sebesar 97%, yang meskipun lebih

rendah dibandingkan dua model lainnya, tetap menunjukkan performa yang baik.

Sebagai perbandingan, model CNN dari penelitian sebelumnya hanya menggunakan dataset yang sama yang terdiri dari 2 jenis mata yaitu Cataract dan Normal, dan menggunakan Optimizer Adam dan mencapai akurasi 92,93%, dengan nilai recall, precision, dan F1-score yang lebih rendah. Hal ini menunjukkan peningkatan signifikan yang dicapai oleh model ResNet-50, EfficientNet-B1, dan VGG-16 dalam penelitian ini. Secara keseluruhan, ResNet-50 terbukti sebagai model terbaik dengan kemampuan klasifikasi yang sangat baik, sementara EfficientNet-B1 menawarkan performa yang hampir setara dengan efisiensi yang lebih tinggi. VGG-16 tetap relevan sebagai model klasifikasi yang andal, meskipun kinerjanya lebih rendah dibandingkan dua model lainnya. Dengan demikian, model yang digunakan dalam penelitian ini berhasil meningkatkan akurasi dan kemampuan generalisasi dibandingkan metode dari penelitian sebelumnya.

IV. KESIMPULAN

Penelitian ini mengklasifikasikan citra mata katarak menjadi dua jenis: normal dan katarak menggunakan tiga model deep learning: ResNet-50, EfficientNet-B1, dan VGG-16. ResNet-50 mencapai kinerja terbaik dengan akurasi 98,15%, serta presisi, recall, dan F1-score masing-masing 98% menurut hasil analisis. Dengan Loss Validation terendah sebesar 0,18, model ini juga menunjukkan kapasitas generalisasi yang luar biasa. Diikuti, EfficientNet-B1 adalah alternatif efektif dalam tugas klasifikasi dengan tingkat akurasi sebesar 96,64%, serta nilai precision, recall, dan F1-score masing-masing mencapai 97%. Meskipun kinerjanya lebih buruk dibandingkan dengan dua model lainnya, VGG-16 berada diposisi ketiga dengan tingkat akurasi 93,82%, serta nilai precision, recall, dan F1-score masing-masing sebesar 94%. Tetap berguna untuk tugas klasifikasi, ini Menekankan manfaat dari transfer learning dan arsitektur deep learning baru dalam meningkatkan akurasi dan generalisasi, ketiga model tersebut melampaui pendekatan CNN dari penelitian sebelumnya [1], yang hanya mencapai akurasi 92,93% dengan menggunakan dataset yang sama berupa Cataract dan Normal dan menggunakan Optimizer Adam. ResNet-50 secara keseluruhan ditemukan sebagai model dengan kinerja terbaik untuk klasifikasi katarak, EfficientNet-B1 berada di urutan kedua dengan efisiensi komputasi yang lebih tinggi. Dengan kemungkinan besar untuk digunakan dalam praktik klinis dunia nyata, model yang dibuat sangat membantu dalam memajukan sistem deteksi katarak otomatis yang lebih akurat dan efektif.

DAFTAR PUSTAKA

- [1] G. J. Bu'ulolo, A. Jacobus, and F. D. Kambey, "Identifikasi Citra Penyakit Mata Katarak Menggunakan Convolutional Neural Network," *Jurnal Teknik Informatika*, vol. 16, no. 4, pp. 375–382, 2021.
- [2] J. He, C. Li, J. Ye, Y. Qiao, and L. Gu, "Multi-label ocular disease classification with a dense correlation deep neural network," *Biomed Signal Process Control*, vol. 63, p. 102167, Jan. 2021, doi: 10.1016/j.bspc.2020.102167.
- [3] S. Astiti, N. Nopriadi, W. Novrian, and Y. P. Putra, "Penerapan Deep Learning pada Pengolahan Data Citra dan Klasifikasi Ulang Vaname Menggunakan Algoritma Convolutional Neural Network," *Building of Informatics, Technology and Science (BITS)*, vol. 6, no. 1, pp. 490–498, 2024.
- [4] R. J. Gunawan, B. Irawan, and C. Setianingsih, "Pengenalan Ekspresi Wajah Berbasis Convolutional Neural Network Dengan Model Arsitektur VGG16," *eProceedings of Engineering*, vol. 8, no. 5, 2021.
- [5] S.-Y. Lin and C.-L. Lin, "Brain tumor segmentation using U-Net in conjunction with EfficientNet," *PeerJ Comput Sci*, vol. 10, p. e1754, Jan. 2024, doi: 10.7717/peerj-cs.1754.
- [6] M. Singla, K. S. Gill, R. Chauhan, and H. S. Pokhariya, "ResNet50 Utilization for Bag Classification: A CNN Model Visualization Approach in Deep Learning," in *2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS)*, IEEE, Jun. 2024, pp. 1–4, doi: 10.1109/ICITEICS61368.2024.10624847.
- [7] R. B. J. Simanjuntak, Y. Fuâ€™TMadah, R. Magdalena, S. Saidah, A. B. Wiratama, and I. D. S. Ubaidah, "Cataract Classification Based on Fundus Images Using Convolutional Neural Network," *JOIV: International Journal on Informatics Visualization*, vol. 6, no. 1, p. 33, Mar. 2022, doi: 10.30630/joiv.6.1.856.
- [8] W. William and C. Lubis, "Klasifikasi Penyakit Mata Menggunakan CNN," *Jurnal Ilmu Komputer dan Sistem Informasi*, vol. 10, no. 1, Mar. 2022, doi: 10.24912/jiksi.v10i1.17834.
- [9] R. Z. I. Ramli, R. D. Atmaja, and I. Widjayanto, "Deteksi dan klasifikasi stadium katarak senilis berdasarkan citra mata menggunakan metode support vector machine (SVM)," *eProceedings of Engineering*, vol. 5, no. 2, 2018.
- [10] A. Febriyanti and A. Alamsyah, "Peningkatan Akurasi Algoritma Support Vector Machine Menggunakan Dual-Tree Complex Wavelet Transform Pada Klasifikasi Citra Katarak dan Normal," *Indonesian Journal of Mathematics and Natural Sciences*, vol. 45, no. 2, pp. 51–58, Oct. 2022, doi: 10.15294/ijmns.v45i2.39723.
- [11] S. Faizal, C. A. Rajput, R. Tripathi, B. Verma, M. R. Prusty, and S. S. Korade, "Automated cataract disease detection on anterior segment eye images using adaptive thresholding and fine tuned inception-v3 model," *Biomed Signal Process Control*, vol. 82, p. 104550, 2023.
- [12] Y. N. Fuadah, R. Magdalena, S. Palondongan, and N. Kumalasari, "Optimasi K-Nearest Neighbor Untuk Sistem Klasifikasi Kondisi Katarak," *TEKTRIKA-Jurnal Penelitian dan Pengembangan Telekomunikasi, Kendali, Komputer, Elektrik, dan Elektronika*, vol. 4, no. 1, pp. 16–25, 2019.
- [13] S. S. Mahmood, S. Chaabouni, and A. Fakhfakh, "Improving Automated Detection of Cataract Disease through Transfer Learning using ResNet50," *Engineering, Technology & Applied Science Research*, vol. 14, no. 5, pp. 17541–17547, Oct. 2024, doi: 10.48084/etasr.8530.
- [14] S. Gayathri, S. Saran, P. S. Kumar, and I. P. Singh, "Cataract Disease Classification using Convolutional Neural Network Architectures," in *2023 Second International Conference on Electronics and Renewable Systems (ICEARS)*, IEEE, Mar. 2023, pp. 992–998, doi: 10.1109/ICEARS56392.2023.10085502.
- [15] M. S. Junayed, M. B. Islam, A. Sadeghzadeh, and S. Rahman, "CataractNet: An Automated Cataract Detection System Using Deep Learning for Fundus Images," *IEEE Access*, vol. 9, pp. 128799–128808, 2021, doi: 10.1109/ACCESS.2021.3112938.
- [16] A. Raza, M. U. Khan, Z. Saeed, S. Samer, A. Mobeen, and A. Samer, "Classification of Eye Diseases and Detection of Cataract

- using Digital Fundus Imaging (DFI) and Inception-V4 Deep Learning Model,” in *2021 International Conference on Frontiers of Information Technology (FIT)*, IEEE, Dec. 2021, pp. 137–142. doi: 10.1109/FIT53504.2021.00034.
- [17] M. A. Amrullah and M. I. Irawan, “Implementasi Jaringan Saraf Konvolusional dengan Inception-V3 untuk Deteksi Katarak Menggunakan Gambar Digital Funduskopi,” *Jurnal Sains dan Seni ITS*, vol. 12, no. 1, pp. A27–A33, 2023.
- [18] A. Rusdy Prasetyo, Sussi, and B. Aditya, “Analisis Perbandingan Algoritma Support Vector Machine (SVM) Dan Convolutional Neural Network (CNN) Untuk Sistem Deteksi Katarak,” *Jurnal Ilmiah Teknik Mesin, Elektro dan Komputer*, vol. 3, no. 1, pp. 1–10, Mar. 2023, doi: 10.51903/juritek.v3i1.604.
- [19] Y. Xiong, Z. He, K. Niu, H. Zhang, and H. Song, “Automatic Cataract Classification Based on Multi-feature Fusion and SVM,” in *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, IEEE, Dec. 2018, pp. 1557–1561. doi: 10.1109/CompComm.2018.8780617.
- [20] M. Safaat, A. Sahari, and D. Lusiyantri, “Implementasi Metode K-Nearest Neighbor Untuk Mengklasifikasi Jenis Penyakit Katarak,” *Jurnal Ilmiah Matematika Dan Terapan*, vol. 17, no. 1, pp. 92–99, Jun. 2020, doi: 10.22487/2540766X.2020.v17.i1.15184.
- [21] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.
- [22] F. Ramadhani, D. Septiana, S. N. Amalia, P. M. Fadilah, and A. Satria, “Klasifikasi Risiko Gizi Buruk Pada Ibu Hamil Menggunakan Metode Random Forest,” *Djtechno: Jurnal Teknologi Informasi*, vol. 5, no. 2, pp. 370–380, Aug. 2024, doi: 10.46576/djtechno.v5i2.4815.