

RANCANG BANGUN SISTEM DETEKSI KATARAK MENGUNAKAN METODE *CONVOLUTIONAL NEURAL NETWORK*

Widyawati¹, Rafli Sidik², Ely Nuryani³, Persis Haryo Winasis⁴

^{1,3}Komputerisasi Akuntansi, Universitas Banten Jaya

²Teknik Informatika, Universitas Banten Jaya

Jl. Syekh Moh. Nawawi Albantani No.2, Kp. Baru, Kec. Curug, Kota Serang, Banten

⁴Rekayasa Perangkat Lunak, IPWIJA University Jakarta

Jl. H. Baping No.17, Susukan, Kec. Ciracas, Kota Jakarta Timur, DKI Jakarta

e-mail: *¹Widyawati@unbaja.ac.id, ²hallo.raflisidik@gmail.com, ³elynuryani@unbaja.ac.id,
⁴persisharyo@ipwija.ac.id

Abstract

Cataract is a condition in which the lens inside the eye becomes cloudy, resulting in blurred or hazy vision. RSAW treats around 800 cataract patients every month, served by seven cataract ophthalmologists. The limited number of doctors and different levels of expertise can affect the duration of the initial screening time. Therefore, a system is needed that can support doctors in the cataract diagnosis process. Convolutional Neural Network (CNN) is a type of neural network specifically designed to process image or video data. CNN is a type of deep learning model that can train systems using large amounts of data and integrate the feature extraction process with classification. This study aims to develop and evaluate the performance of a CNN-based cataract detection system as a tool for early diagnosis in cataract patients at RSAW. The CNN model was trained using an eye image dataset consisting of 1120 images of cataract and non-cataract patients. The CNN architecture used was VGG16, chosen for its ability to extract relevant features. The evaluation results show that the system is able to detect cataracts with an accuracy of 96.43%. This system has the potential to increase the efficiency of the screening process and reduce the workload of doctors, thereby improving the quality of eye health services.

Keyword: *Cataract, CNN, Deep Learning, Detection, VGG16*

PENDAHULUAN

Katarak merupakan salah satu penyebab utama gangguan penglihatan di dunia. Berdasarkan laporan Organisasi Kesehatan Dunia (WHO), lebih dari 2,2 miliar orang di dunia mengalami gangguan penglihatan, dan sekitar 94 juta di antaranya disebabkan oleh katarak. Di Indonesia, sekitar 2,4 juta orang menderita katarak, menjadikan negara ini sebagai salah satu dengan jumlah penderita katarak tertinggi di Asia Tenggara. Faktor utama penyebab katarak adalah penuaan, namun kondisi ini juga dapat dipicu oleh cedera mata, penggunaan obat-obatan tertentu, atau faktor genetik.

Gangguan penglihatan akibat katarak memiliki dampak signifikan terhadap produktivitas individu dan perekonomian nasional. Penurunan kualitas penglihatan dapat menghambat aktivitas sehari-hari, menurunkan kualitas hidup, dan membatasi kemampuan seseorang dalam bekerja. Oleh karena itu, deteksi dini katarak menjadi sangat penting untuk mencegah dampak lebih lanjut, terutama kebutaan yang dapat terjadi akibat keterlambatan diagnosis dan penanganan. Proses diagnosis masih bergantung pada pemeriksaan manual yang membutuhkan keahlian dan waktu yang tidak singkat, sehingga efisiensi dalam deteksi dini menjadi tantangan tersendiri.

Metode *Convolutional Neural Network* (CNN) dalam bidang kecerdasan buatan telah banyak digunakan untuk analisis citra medis, termasuk pendeteksian penyakit mata. CNN bekerja dengan meniru sistem pengenalan pola pada *visual cortex* manusia, sehingga mampu mengidentifikasi dan mengklasifikasikan gambar dengan tingkat akurasi yang tinggi. Dengan

perkembangan teknologi, sistem deteksi katarak berbasis web yang menggunakan metode CNN berpotensi menjadi solusi inovatif dalam membantu dokter melakukan deteksi dini secara lebih efisien dan akurat.

Dalam beberapa tahun terakhir, penelitian telah menunjukkan bahwa model *Convolutional Neural Network* (CNN) efektif dalam mendeteksi penyakit. Misalnya, sebuah studi pada tahun 2020 menunjukkan bahwa CNN dapat digunakan sebagai alat yang efektif dalam klasifikasi prediksi diabetes, yang nantinya dapat mendeteksi pasien yang kemungkinan menderita diabetes (Zahir & Adi Saputra, 2024). Penelitian lain pada tahun 2022 berhasil membangun sistem deteksi penyakit mata katarak menggunakan CNN, yang diharapkan dapat membantu memudahkan proses diagnosis dini penyakit mata katarak untuk masyarakat yang mengalami gangguan penglihatan (Ardana, 2024). Beberapa penelitian sebelumnya telah mengeksplorasi penggunaan CNN dalam bidang oftalmologi. Misalnya, penelitian oleh (Ting et al., 2017) menunjukkan bahwa model *deep learning* dapat mendeteksi retinopati diabetik dengan akurasi tinggi, bahkan sebanding dengan dokter spesialis mata. Penelitian lain oleh (Pratama & Utamingrum, 2017; Syahrul & Sasongko, 2022) juga membuktikan bahwa CNN dapat diterapkan dalam pendeteksian katarak dengan hasil yang menjanjikan. Namun, sebagian besar studi masih terbatas pada implementasi berbasis *desktop* atau memerlukan perangkat dengan spesifikasi tinggi, sehingga aksesibilitasnya masih terbatas.

Penelitian ini berupaya mengisi gap tersebut dengan mengembangkan sistem deteksi katarak berbasis *website* menggunakan metode CNN dengan arsitektur model VGG16. Keunggulan pendekatan ini adalah memungkinkan akses yang lebih luas tanpa terikat oleh batasan geografis dan perangkat keras tertentu. Selain itu, sistem ini dirancang untuk memberikan hasil deteksi yang cepat dan akurat, sehingga dapat membantu dokter dalam melakukan skrining awal terhadap pasien katarak.

Penelitian ini bertujuan untuk merancang dan membangun sistem deteksi katarak berbasis *website* menggunakan metode CNN dengan arsitektur VGG16. Berbeda dari penelitian sebelumnya, sistem ini menawarkan pendekatan yang lebih praktis dengan memanfaatkan platform berbasis web, sehingga mempermudah akses bagi tenaga medis maupun pasien. Melalui penelitian ini, diharapkan sistem yang dikembangkan dapat meningkatkan efisiensi proses diagnosis, mempercepat waktu deteksi, serta mengurangi risiko keterlambatan penanganan katarak yang dapat berujung pada kebutaan total. Dengan demikian, penelitian ini memberikan kontribusi terhadap inovasi teknologi dalam bidang kesehatan, khususnya dalam peningkatan layanan oftalmologi di Indonesia.

METODE PENELITIAN

Metode yang digunakan dalam penelitian ini mencakup berbagai pendekatan untuk memastikan sistem deteksi katarak dapat berfungsi secara optimal. Pendekatan utama yang diterapkan adalah penggunaan *Convolutional Neural Network* (CNN) dengan arsitektur VGG16, yang telah terbukti efektif dalam mengenali dan mengklasifikasikan gambar dengan tingkat akurasi yang tinggi (Azizah, 2023; Tilasefana & Putra, 2023). Dalam pengembangan sistem, metode Waterfall (I. A. G. S. Putra & Trisnawati, 2025; Widyawati et al., 2024) digunakan sebagai model pengembangan perangkat lunak yang sistematis dan terstruktur, memungkinkan setiap tahap proses dikembangkan secara bertahap mulai dari analisis, desain, implementasi, hingga pengujian. Untuk pemodelan sistem, penelitian ini menerapkan *Unified Modeling Language* (UML) guna memvisualisasikan arsitektur dan alur kerja sistem secara lebih jelas. Penyimpanan data dilakukan menggunakan MySQL, yang dipilih karena kemampuannya dalam menangani basis data yang besar serta keandalannya dalam proses transaksi data. Selanjutnya, sistem deteksi katarak ini dirancang dalam bentuk *website* (Nuryani et al., 2024), sehingga pengguna dapat mengakses informasi hasil screening mata dengan lebih mudah, tanpa keterbatasan perangkat atau lokasi geografis. Kombinasi metode ini diharapkan dapat menciptakan sistem yang lebih akurat, efisien, dan mudah diakses bagi pasien maupun tenaga medis, meskipun terdapat banyak deteksi penyakit mata katarak dengan metode lainnya

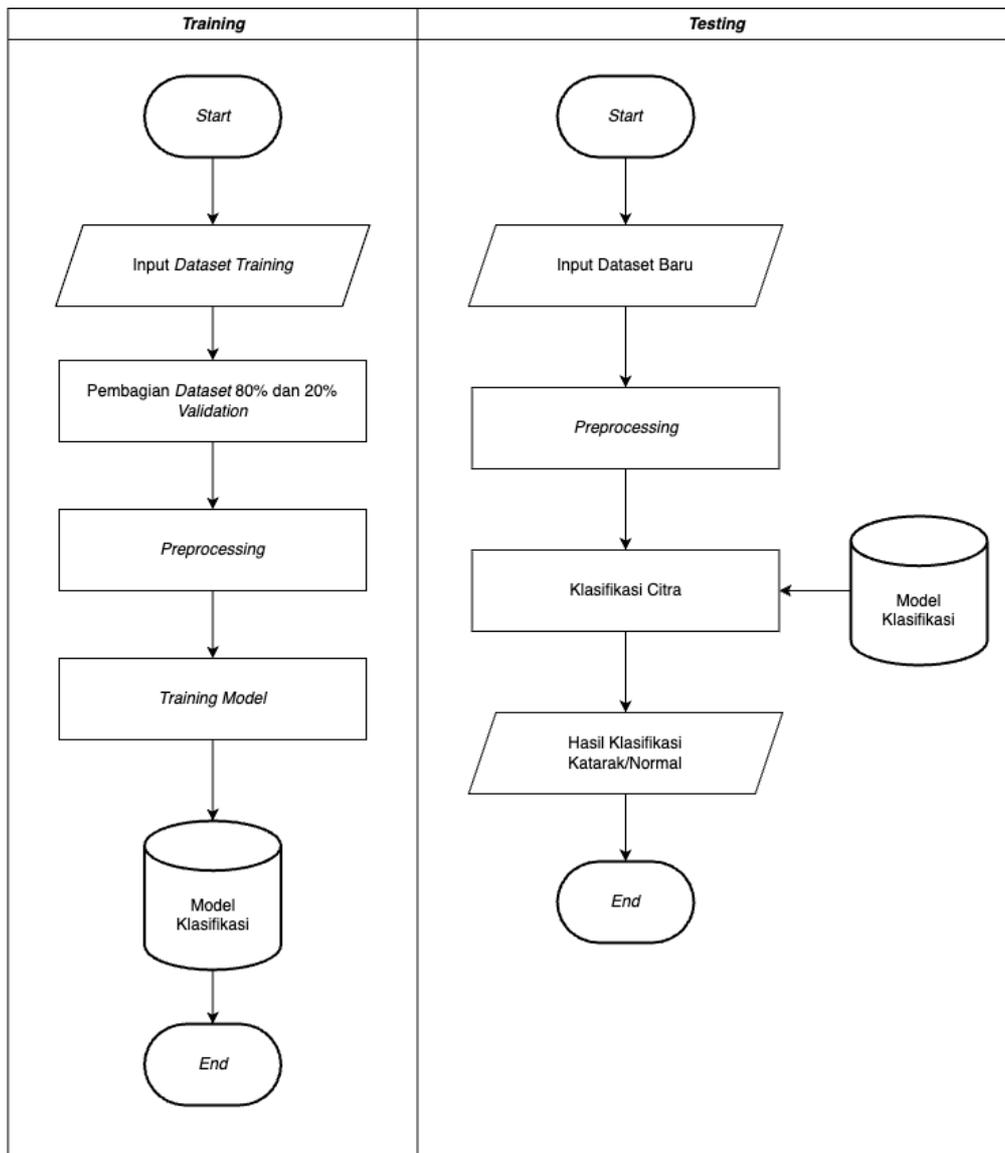
selain menggunakan CNN (Marcello et al., 2021).

Convolutional Neural Network (CNN) adalah jenis jaringan saraf tiruan yang dirancang khusus untuk memproses data berbentuk grid, seperti citra. CNN terdiri dari beberapa lapisan, termasuk lapisan konvolusi, *pooling*, dan *fully connected*, yang bekerja sama untuk mengekstraksi fitur dari input dan melakukan klasifikasi atau deteksi. Lapisan konvolusi berfungsi untuk mendeteksi fitur lokal dengan menerapkan filter pada input, sementara lapisan *pooling* mengurangi dimensi data untuk mengurangi kompleksitas komputasi dan mencegah *overfitting*. Lapisan *fully connected* kemudian menggabungkan fitur-fitur yang telah diekstraksi untuk menghasilkan output akhir (Setiawan, 2019).

Salah satu arsitektur CNN yang populer adalah VGG16 (Handhika, 2023; R et al., 2023; Ushuludin, 2023), yang diperkenalkan oleh *Visual Geometry Group* dari University of Oxford. VGG16 terdiri dari 16 lapisan yang dapat dilatih, termasuk 13 lapisan konvolusi dan 3 lapisan *fully connected*. Arsitektur ini menggunakan *filter* berukuran kecil (3x3) dalam setiap lapisan konvolusi dan menerapkan fungsi aktivasi ReLU untuk memperkenalkan *non-linearitas*. Setelah beberapa lapisan konvolusi, lapisan *pooling* dengan ukuran 2x2 digunakan untuk mengurangi dimensi data. Pada akhirnya, data yang telah diproses melalui lapisan-lapisan ini dilewatkan ke lapisan *fully connected* untuk menghasilkan output klasifikasi. VGG16 dikenal karena kedalamannya dan kemampuannya dalam mengekstraksi fitur yang kompleks dari citra, sehingga sering digunakan sebagai dasar dalam berbagai aplikasi pengenalan citra.

Dalam konteks deteksi katarak, CNN dengan arsitektur VGG16 dapat digunakan untuk menganalisis citra mata dan mengidentifikasi adanya katarak berdasarkan pola-pola yang terdeteksi dalam citra tersebut. Dengan melatih model pada dataset yang relevan, sistem dapat belajar membedakan antara mata yang sehat dan yang terkena katarak, sehingga membantu dalam proses deteksi dini dan penanganan yang lebih efektif.

Berikut ini merupakan *flowchart* yang digunakan dalam melaksanakan penelitian yang tertera pada gambar 1 di bawah ini:

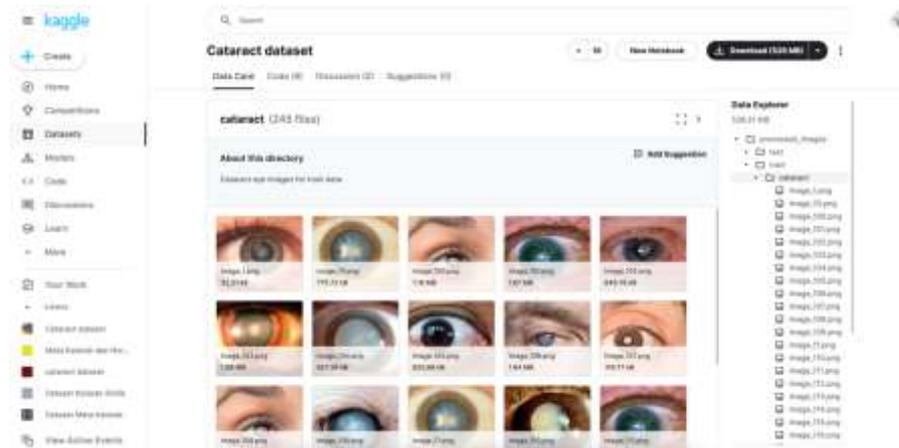


Gambar 1. Flowchart penyelesaian masalah.

Proses deteksi mata dengan *CNN* meliputi pengumpulan dan persiapan data berlabel. Model dilatih untuk mengenali fitur khas mata pada gambar. Setelah pelatihan, model diuji pada data baru untuk mengukur akurasi deteksinya. Tahap praproses data (misal: normalisasi, augmentasi) dan pemilihan parameter model yang tepat sangat penting untuk meningkatkan kinerja deteksi. Pemodelan deteksi mata menggunakan *CNN* dapat dilakukan melalui tahapan-tahapan berikut:

1) *Data Preparation*

Tahap awal dalam pengembangan model deteksi mata memerlukan kumpulan data yang terdiri dari citra-citra mata yang telah dianotasi secara akurat, mengindikasikan keberadaan atau ketiadaan mata pada setiap citra. Gambar 2 mengilustrasikan contoh kumpulan data citra mata yang telah dianotasi dimana *dataset* yang berasal dari *website* Kaggle.com (Nuralia et al., 2023).



Gambar 2. Dataset dari website Kaggle.com

2) Memuat *Dataset* dan *Dataflow*

Pada langkah ini, *dataset* yang telah diunggah ke *Google Drive* diunduh atau dipanggil. Tahap ini akan dilakukan pemuatan *dataset* ke dalam sistem yang akan dibangun menggunakan *Python GoogleColab*, dan mendefinisikan darimana sumber datanya berasal (*dictionary*).

```
# Persiapan data
from google.colab import drive
drive.mount('/content/drive')

sdir=r'/content/drive/MyDrive/Skripsi/Dataset/'
```

3) Pembagian *Dataset*

Kumpulan data (*dataset*) umumnya dipartisi menjadi tiga himpunan bagian yang berbeda (Azizah, 2023; F. A. Putra et al., 2024):

- a. *Training data*: Berfungsi sebagai landasan utama dalam proses pembelajaran model, di mana bobot-bobot jaringan saraf dioptimalkan berdasarkan pola-pola yang teridentifikasi pada data latih.
- b. *Validation data*: Digunakan secara berkala selama pelatihan untuk mengevaluasi performa model terhadap data yang belum pernah dilihat sebelumnya, membantu dalam penentuan parameter-parameter optimal dan pencegahan *overfitting*.
- c. *Testing data*: Berperan sebagai tolok ukur akhir untuk menilai kinerja generalisasi model setelah seluruh proses pelatihan selesai, memberikan estimasi yang lebih realistis mengenai kemampuan model dalam menghadapi data baru yang belum pernah ditemui.

```
# Pembagian Dataset
train_split=.8
test_split=.1
dummy_split=test_split/(1-train_split)
train_df, dummy_df=train_test_split(df, train_size=train_split, shuffle=True,
random_state=123)
test_df, valid_df=train_test_split(dummy_df, train_size=dummy_split, shuffle=True,
random_state=123)
print('train_df length: ', len(train_df), ' test_df length: ', len(test_df), ' valid_df length: ', len(valid_df))
```

Secara keseluruhan, kode ini membagi data menjadi tiga *subset*: 80% untuk *training*, 10% untuk testing, dan 10% untuk *validation*.

4) *Preprocessing*

Data yang telah dikumpulkan selanjutnya mengalami serangkaian *preprocessing* sebelum digunakan untuk melatih model *CNN*. Tahapan-tahapan *preprocessing* yang lazim dilakukan meliputi:

- a. Penyesuaian Ukuran (*Resizing*): Menyeragamkan dimensi citra-citra masukan agar sesuai dengan kebutuhan arsitektur *CNN*.
- b. Normalisasi (*Normalization*): Transformasi intensitas piksel ke dalam rentang numerik yang lebih terkendali, seperti 0-1 atau -1 hingga 1, untuk meningkatkan stabilitas dan kinerja pelatihan model.
- c. Augmentasi Data (*Data Augmentation*): Opsional, namun seringkali bermanfaat untuk memperkaya variasi data latih dengan menerapkan transformasi-transformasi seperti rotasi, translasi, atau pencerminan horizontal pada citra-citra asli.

```
height=128
width=128
channels=3
batch_size=64

img_shape=(height, width, channels)
img_size=(height, width)
length=len(test_df)
test_batch_size=sorted([int(length/n) for n in range(1,length+1) if length % n ==0 and
length/n<=80],reverse=True)[0]
test_steps=int(length/test_batch_size)
print ( 'test batch size: ',test_batch_size, ' test steps: ', test_steps)
```

Kode ini melakukan *preprocessing* data citra untuk pelatihan model *machine learning*. Pertama, dimensi citra (tinggi, lebar, *channel*) dan *batch size training* ditentukan. Kemudian, *batch size* dan *step count* untuk data uji dihitung berdasarkan jumlah sampel.

5) Pembuatan Model *CNN*

Pada langkah ini, dilakukan proses pembuatan model *CNN* pada sistem yang sedang dikembangkan.

```
model_name='Rafli'
# print("Building model with", base_model)

model = tf.keras.Sequential([
    tf.keras.layers.Conv2D(filters=32, padding='same',input_shape=img_shape,
kernel_size=(3,3), activation='relu', strides=1),
    tf.keras.layers.MaxPooling2D(pool_size=(2,2)),
    tf.keras.layers.Dropout(rate=0.6),

    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(class_count, activation='softmax')
])

model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=.001),
loss='categorical_crossentropy', metrics='accuracy')
```

Kode ini membangun sebuah model *CNN* untuk mengolah gambar, model ini terdiri dari beberapa lapisan yang bekerja sama untuk mengekstrak fitur-fitur penting dari gambar

dan melakukan klasifikasi. Model ini dapat disesuaikan dengan berbagai tugas pengenalan gambar dengan mengubah parameter-parameternya.

Model ini menggunakan arsitektur berurutan (*Sequential*) dengan lapisan-lapisan:

- a. *Conv2D*: Untuk ekstraksi fitur dari gambar.
- b. *MaxPooling2D*: Untuk mengurangi ukuran fitur dan meningkatkan invarian translasi.
- c. *Dropout*: Untuk regularisasi dan mengurangi *overfitting*.
- d. *Flatten*: Untuk mengubah data 2D dari lapisan pooling menjadi vektor 1D.
- e. *Dense*: Lapisan output yang memberikan prediksi akhir dengan aktivasi softmax.
- f. *Optimizer*: Menggunakan *Adam* dengan *learning rate* 0.001 untuk mengoptimalkan bobot model.
- g. *Loss Function*: Menggunakan *categorical_crossentropy* untuk mengukur kesalahan pada masalah klasifikasi multi-class.
- h. *Metrics*: Mengukur akurasi selama pelatihan dan evaluasi.

6) Pelatihan dan Pengujian Model

Selanjutnya adalah melakukan proses pelatihan menggunakan *training data* dan diuji menggunakan *testing data* yang telah dipersiapkan sebelumnya setelah model sudah dibuat. Proses deteksi mata katarak menggunakan algoritma *CNN* dipengaruhi oleh beberapa hal antara lain pembagian *training data* dan *testing*, *dropout*, dan *batchsize*.

HASIL DAN PEMBAHASAN

Pada penelitian ini uji coba yang dilakukan dengan pembagian data, *dropout*, dan *batchsize* untuk mendapatkan model klasifikasi yang terbaik. Pembagian data yang diuji coba yaitu *training data* 80%, *testing data* 10% dan *validation data* 10%. Dimana akan diatur *dropout layer* yang dilakukan pada uji coba nilai probabilitas (0.5, 0.6) dan pada *batchsize* yang diujikan bernilai (32, 64) dan *epoch* 10. Model terbaik dipilih berdasarkan evaluasi sistem menggunakan *confusion matrix* dengan nilai *accuracy*, *precision* dan *recall*.

Tabel 1. Pembagian data kelas.

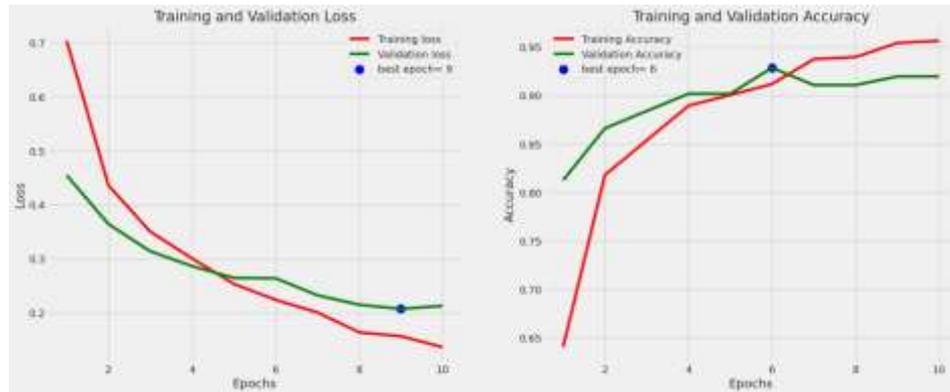
Nama Kelas	<i>Training Data</i> (80%)	<i>Testing Data</i> (10%)	<i>Validation Data</i> (10%)
Normal	448	57	57
Katarak	448	55	55
Jumlah	896	112	112

Uji coba dengan pembagian data ditunjukkan pada Tabel 1.

Tabel 2. Hasil evaluasi percobaan perbandingan.

<i>Epoch</i>	<i>Dropout</i>	<i>Batch Size</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>
10	0.5	32	95.24%	92.72%	98,07%
		64	94.64%	90.90%	98,03%
	0.6	32	95.53%	92,72%	98,11%
		64	96,43%	94,55%	98,11%

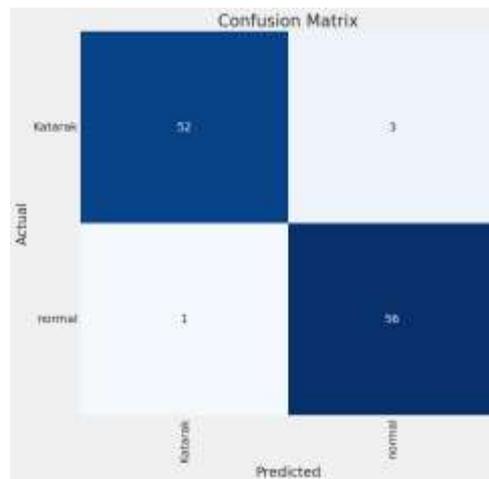
Berdasarkan Tabel 2 dapat diketahui akurasi terbaik dari pembagian *training data* 80% yaitu 96,43% dengan inisialisasi nilai probabilitas pada *dropout layer* 0.6 dan nilai batchsize 64. Setiap percobaan *batchsize* dan *dropout* menunjukkan rata-rata nilai akurasi tidak stabil. Hasil grafik pada proses evaluasi ditunjukkan pada gambar 2.



Gambar 3. Grafik hasil training.

Pada grafik diatas, bahwa akurasi pada training data adalah 0.9643 dan nilai lossnya adalah 0.1643, Kemudian untuk akurasi pada validation data adalah 0.9394 dan nilai lossnya adalah 0.2294. Akurasi model dari training data dan testing data jika dilihat dari epoch 1 sampai 10 dominan naik, namun juga ada penurunan pada epoch tertentu, kenaikan dan penurunan akurasi model terjadi karena data yang diprediksi benar setiap iterasi selalu berbeda atau naik turun. Pergerakan akurasi yang dihasilkan pada testing data maupun training data hampir stabil hal ini kemungkinan disebabkan oleh jumlah data yang sama pada tiap kelompok.

Dari proses training tersebut dilakukan pengujian dari data testing yang menghasilkan confusion matrix pada Gambar 4.



Gambar 4. Confusion matrix hasil pengujian.

Tabel 3. Hasil Pengujian Confusion Matrix.

Prediksi	Aktual	
	Katarak	Normal
Katarak	52	3
Normal	1	56

Berikut adalah uraian nilai-nilai dari matrix hasil pengujian:

- a) *True Positive (TP)*: 52 (Benar diprediksi sebagai Katarak)
- b) *True Negative (TN)*: 56 (Benar diprediksi sebagai Normal)
- c) *False Positive (FP)*: 1 (Salah, diprediksi sebagai Katarak, padahal Normal)
- d) *False Negative (FN)*: 3 (Salah, diprediksi sebagai Normal, padahal Katarak)

Dari uraian di atas dapat dilakukan perhitungan accuracy, precision dan recall . Berikut

perhitungan untuk mendapatkan nilai accuracy, precision dan recall untuk hasil pengujian:

$$\begin{aligned}
 Accuracy &= \frac{(TP+TN)}{(TP+FP+FN+TN)} & (1) \\
 &= \frac{(52 + 56)}{(52 + 1 + 3 + 56)} \\
 &= \frac{108}{112} \\
 &= 0.9643 * 100\% \\
 &= 96,43\%
 \end{aligned}$$

$$\begin{aligned}
 Precision &= \frac{TP}{(FP+TP)} & (2) \\
 &= \frac{52}{(3 + 52)} \\
 &= \frac{52}{55} \\
 &= 0.9455 * 100\% \\
 &= 94,55\%
 \end{aligned}$$

$$\begin{aligned}
 Recall &= \frac{TP}{(TP+FN)} & (3) \\
 &= \frac{52}{(52 + 1)} \\
 &= \frac{52}{53} \\
 &= 0.9811 * 100\% \\
 &= 98.11\%
 \end{aligned}$$

Berdasarkan hasil perhitungan metrik di atas, model yang digunakan memiliki kinerja yang sangat baik dalam mengklasifikasikan gambar mata antara yang terkena katarak dan yang normal. Model ini mampu mendeteksi sebagian besar kasus katarak dengan akurasi yang tinggi, dan juga memiliki tingkat kesalahan yang rendah.

KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa sistem deteksi katarak menggunakan metode *Convolutional Neural Network* (CNN) yang telah dirancang berhasil memberikan dukungan yang signifikan kepada dokter dalam proses skrining awal katarak. Sistem ini mampu menganalisis citra katarak dengan akurasi 96% dalam mendeteksi keberadaan katarak. Proses deteksi juga jauh lebih cepat, hanya membutuhkan waktu 5 detik per gambar, sehingga meningkatkan efisiensi pemeriksaan.

SARAN

Berdasarkan hasil penelitian ini, sistem deteksi katarak berbasis *Convolutional Neural Network* (CNN) telah menunjukkan tingkat akurasi yang cukup baik. Namun, akurasi yang diperoleh masih belum mencapai tingkat maksimal, sehingga diperlukan upaya lebih lanjut untuk meningkatkan performa model agar mendekati tingkat akurasi yang optimal. Salah satu aspek yang dapat diperbaiki dalam penelitian mendatang adalah penggunaan dataset yang lebih berkualitas. Dataset yang digunakan dalam penelitian ini masih memiliki keterbatasan dalam hal variasi dan jumlah data, yang dapat mempengaruhi kemampuan model dalam mengenali pola dengan lebih akurat. Oleh karena itu, pemilihan dataset yang lebih representatif dan berkualitas menjadi langkah penting dalam penelitian selanjutnya. Jika penelitian mendatang tetap menggunakan dataset yang sama, disarankan untuk melakukan seleksi data yang lebih ketat dan tidak menggunakan metode pemilihan secara acak, sehingga model dapat dilatih dengan data yang lebih relevan dan bervariasi. Selain itu, penting untuk menambahkan dataset testing yang

lebih luas guna mengukur performa model pada data yang benar-benar baru. Dengan demikian, penelitian di masa depan diharapkan dapat meningkatkan efektivitas sistem dalam mendeteksi katarak secara lebih akurat dan efisien, sehingga dapat berkontribusi dalam upaya deteksi dini dan penanganan katarak yang lebih baik di Indonesia.

DAFTAR PUSTAKA

- Ardana, A. V. P. (2024). *Sistem Deteksi Penyakit Mata Katarak Dengan Menggunakan Convolutional Neural Network (CNN)*.
- Azizah, Q. N. (2023). Klasifikasi Penyakit Daun Jagung Menggunakan Metode Convolutional Neural Network AlexNet. *Sudo Jurnal Teknik Informatika*, 2(1), 28–33. <https://doi.org/10.56211/sudo.v2i1.227>
- Handhika, T. (2023). *Implementasi Algoritma Convolutional Neural Network Dengan Arsitektur Vgg-16 Untuk Data Tak Seimbang Dalam Pengendalian Kualitas Hasil Produk Pengecoran*.
- Marcello, L. M., Oey, E., Lie, S., & Astuti, W. (2021). Automatic Cataract Detection System based on Support Vector Machine (SVM). *Proceedings of the Second Asia Pacific International Conference on Industrial Engineering and Operations Management*, 959–965.
- Nuralia, S., Harliana, H., & Prabowo, T. (2023). Implementasi Naive Bayes Classifier Dalam Memprediksi Kelulusan Mahasiswa Implementation of Naive Bayes Classifier in Predicting Student Graduation. *JACIS: Journal Automation Computer Information System*, 3(01), 63–72. <https://bit.ly/45fwDP4>
- Nuryani, E., Hasanah, H., Amiruddin, D., Saputri Pratama, S., Ilmu Komputer, F., Banten Jaya Jl Syekh Moh Nawawi Albantani Kp Boru Kecamatan Curug, U., Jaya, C., & Serang, K. (2024). Perancangan Sistem Informasi Pelayanan Pasien Berbasis Web Pada Praktik Mandiri Bidan Lia Yulianingsih. *Journal of Innovation and Future Technology (IFTECH)*, 6(1), 1–14.
- Pratama, A. A., & Utamingrum, F. (2017). Sistem Pendeteksi Tingkat Keparahan Katarak Berdasarkan Citra Digital Menggunakan Metode U-Net dan CNN. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 1(1), 1–9. <http://j-ptiik.ub.ac.id>
- Putra, F. A., Irawan, D., & Wulandari, C. (2024). Penerapan Metode CNN (Convolution Neural Network) Dalam Klasifikasi Buah. *Journal of Information System Research (JOSH)*, 6(1), 733–740. <https://doi.org/10.47065/josh.v6i1.6121>
- Putra, I. A. G. S., & Trisnawati, N. L. P. (2025). Sistem Pendeteksi Kesehatan Mental Remaja Menggunakan Metode Forward Chaining Dan Naive Bayes. *Jurnal Sistem Informasi Dan Informatika (Simika)*, 8(1), 212–222.
- R, I. M., Johan, T. M., & Luthfi, L. (2023). Klasifikasi Citra Ikan Menggunakan Algoritma Convolutional Neural Network dengan Arsitektur VGG-16. *KLIK: Kajian Ilmiah Informatika Dan Komputer*, 4(2), 978–985. <https://doi.org/10.30865/klik.v4i2.1209>
- Setiawan, W. (2019). Perbandingan Arsitektur Convolutional Neural Network Untuk Klasifikasi Fundus. *Jurnal SimanteC*, 7(2), 49–54.
- Syahrul, F. H., & Sasongko, P. S. (2022). Penerapan Convolutional Neural Network Untuk Klasifikasi Tingkat Keparahan Retinopati Diabetik Pada Penderita Diabetes Melitus. *Jurnal Masyarakat Informatika*, 13(1), 1–14.
- Tilasefana, R. A., & Putra, R. E. (2023). Penerapan Metode Deep Learning Menggunakan Algoritma CNN Dengan Arsitektur VGG NET Untuk Pengenalan Cuaca. *JINACS (Journal of Informatics and Computer Science)*, 05(1), 48–57.
- Ting, D. S. W., Cheung, C. Y.-L., Lim, G., Tan, G. S. W., Quang, N. D., Gan, A., Hamzah, H., Garcia-Franco, R., Yeo, I. Y. S., Lee, S. Y., Wong, E. Y. M., Sabanayagam, C., Baskaran, M., Ibrahim, F., Tan, N. C., Finkelstein, E. A., Lamoureux, E. L., Wong, I. Y., Bressler, N. M., ... Wong, T. Y. (2017). Development and Validation of a Deep Learning System for Diabetic Retinopathy and Related Eye Diseases Using Retinal Images From Multiethnic

- Populations With Diabetes. *JAMA Network*, 318(22), 2155–2265. <https://doi.org/10.1001/jama.2017.18152>
- Ushuludin, M. (2023). *Sistem Deteksi Masker pada Wajah Menggunakan Convolutional Neural Network Arsitektur VGG16*.
- Widyawati, W., Oki Astrabuwono, M., Surahmat, A., & Kadun, K. (2024). Sistem Pendukung Keputusan Pemberian Penghargaan Karyawan Menggunakan Metode Multi-Attribute Utility Theory (MAUT) Di PT Nikomas Gemilang. *Jurnal Innovation and Future Technology (IFTECH) P-ISSN*, 6(1), 144–152.
- Zahir, M., & Adi Saputra, R. (2024). Deteksi Penyakit Retinopati Diabetes Menggunakan Citra Mata Dengan Implementasi Deep Learning CNN. *JURNAL TEKNOINFO*, 18(1), 121–132. <https://www.kaggle.com/datasets/gunavenkatdoddi/eye-diseases-classification>