

## ***Deep Learning Architecture of VGG16 and VGG19 for Eyeglasses Face Classification***

**Muhammad Faiz Abrar Fatah <sup>1</sup>, Ericks Rachmat Swedia <sup>2</sup>, Margi Cahyanti <sup>3</sup>, dan M Ridwan Dwi Septian <sup>4</sup>**

<sup>1,2,4</sup>Informatika, Universitas Gunadarma

<sup>3</sup>Sistem Informasi, Universitas Gunadarma

<sup>1,2,3,4</sup> Jalan Margonda Raya No.100, Pondok Cina, Depok

E-mail : faizabrar@lab-gunadarma.id<sup>1</sup>, ericks\_rs@staff.gunadarma.ac.id<sup>2</sup>, margi@staff.gunadarma.ac.id<sup>3</sup>, dan ridwandwiseptian@staff.gunadarma.ac.id<sup>4</sup>

### ***ABSTRACT***

*This study aims to conduct a comparative analysis of the performance of two popular Convolutional Neural Network (CNN) architectures, namely Visual Geometry Group (VGG16 and VGG19), in classifying facial images with glasses using the "Glasses or No Glasses" dataset. Both models were developed through a transfer learning approach by utilizing pre-trained ImageNet weights to accelerate convergence and improve classification accuracy. The training process employed the Adam optimizer with binary crossentropy as the loss function. The dataset was divided into two subsets 80% for training and 20% for validation while testing was performed on 50 unseen images excluded from both subsets. Experimental results show that the VGG16 architecture achieved 87.86% training accuracy and 89.11% validation accuracy, whereas VGG19 achieved 86.86% training accuracy and 87.89% validation accuracy. On the testing dataset, VGG16 correctly classified 47 out of 50 images (94%), while VGG19 correctly classified 48 images (96%). Although the performance gap is relatively small, VGG19 demonstrated better computational efficiency with a shorter training duration (2 hours and 41 minutes) compared to VGG16 (2 hours and 59 minutes). Furthermore, the trained models were successfully implemented in an Android application using TensorFlow Lite, enabling real-time eyeglasses detection. These findings indicate that the VGG19 architecture offers superior efficiency and accuracy for deep learning-based eyeglass face classification tasks.*

**Keywords:** Convolutional Neural Network, Deep Learning, Face Classification, Glasses, TensorFlow Lite, VGG16, VGG19

## **Arsitektur Deep Learning VGG16 dan VGG19 dalam Klasifikasi Wajah Berkacamata**

### **ABSTRAK**

Penelitian ini bertujuan untuk melakukan analisis komparatif terhadap kinerja dua arsitektur Convolutional Neural Network (CNN) populer, yakni Visual Geometry Group (VGG16 dan VGG19), dalam tugas klasifikasi citra wajah berkacamata menggunakan dataset berkacamata dan tidak menggunakan kacamata. Kedua model dikembangkan dengan pendekatan transfer learning menggunakan pre-trained weights ImageNet guna mempercepat proses konvergensi dan meningkatkan akurasi klasifikasi. Proses pelatihan dilakukan menggunakan Adam optimizer dengan fungsi kerugian binary crossentropy. Dataset dibagi menjadi dua, yaitu 80% untuk pelatihan dan 20% untuk validasi, sedangkan pengujian dilakukan terhadap 50 citra uji yang tidak termasuk dalam kedua subset tersebut. Hasil eksperimen menunjukkan bahwa arsitektur VGG16 memperoleh akurasi pelatihan sebesar 87,86% dan akurasi validasi 89,11%, sedangkan VGG19 mencapai akurasi pelatihan 86,86% dan akurasi validasi 87,89%. Pada tahap pengujian, VGG16 mampu mengklasifikasikan 47 dari 50 citra secara benar (94%), sementara VGG19 berhasil mengklasifikasikan 48 citra secara benar (96%). Walaupun selisih akurasi keduanya relatif kecil, VGG19 menunjukkan efisiensi komputasi yang lebih baik dengan waktu pelatihan yang lebih singkat, yaitu 2 jam 41 menit dibandingkan 2 jam 59 menit pada VGG16. Selain itu, model hasil pelatihan berhasil diimplementasikan pada aplikasi Android berbasis TensorFlow Lite yang mampu melakukan deteksi wajah berkacamata secara langsung. Hasil penelitian ini menegaskan bahwa arsitektur VGG19 lebih unggul dalam hal efisiensi dan akurasi untuk tugas klasifikasi wajah berkacamata berbasis deep learning.

**Kata Kunci:** Convolutional Neural Network, Deep Learning, Kacamata, Klasifikasi Wajah, TensorFlow Lite, VGG16, VGG19

## 1. PENDAHULUAN

Berkembangan teknologi *Artificial Intelligence* (AI) dan *Computer Vision* telah membawa perubahan signifikan dalam cara sistem komputer memahami dan menganalisis citra digital. Salah satu bidang yang berkembang pesat adalah pengenalan wajah (*face recognition*) (Kushariyadi dkk., 2024), yang kini digunakan secara luas dalam berbagai sektor seperti keamanan, autentikasi biometrik, sistem kehadiran otomatis, hingga interaksi manusia dan mesin (Wijaya, 2024). Dalam konteks pengenalan wajah, salah satu tantangan yang masih dihadapi adalah variabilitas atribut visual pada wajah, seperti ekspresi, pencahayaan, posisi, serta keberadaan aksesoris seperti kacamata (Widyantoro, 2020). Keberadaan kacamata dapat menutupi sebagian fitur penting pada wajah, seperti mata dan alis, sehingga menurunkan akurasi model pengenalan wajah konvensional (Cahyaningtyas dkk., 2025).

Menurut data dari *Vision Council* (Leske dkk., 2020), lebih dari 60% populasi dewasa di dunia menggunakan kacamata atau lensa korektif (Ariesanti Tri Handayani dkk., 2025). Hal ini menegaskan pentingnya kemampuan sistem klasifikasi wajah untuk mengenali kondisi seseorang yang berkacamata maupun tidak. Kemampuan tersebut dapat digunakan dalam berbagai aplikasi, misalnya pada sistem keamanan berbasis wajah yang harus tetap berfungsi walaupun pengguna mengenakan kacamata, maupun dalam bidang *retail analytics* dan personalisasi layanan digital berbasis citra wajah (Santo Dewatmoko dkk., 2025).

Dalam ranah *deep learning*, *Convolutional Neural Network* (CNN) telah terbukti sebagai pendekatan paling efektif untuk tugas pengenalan pola visual (Budi dkk., 2024). Salah satu arsitektur CNN yang paling berpengaruh adalah *Visual Geometry Group Network* (VGGNet) (Reihardian, 2024), yang diperkenalkan oleh Simonyan dan Zisserman (Olii dkk., 2023). Arsitektur ini hadir dalam dua varian utama, yaitu VGG16 dan VGG19, yang masing-masing terdiri atas 16 dan 19 lapisan konvolusi (Mascarenhas & Agarwal, 2021). Keduanya dikenal karena kesederhanaan struktur dan kemampuannya mengekstraksi fitur hierarkis dari citra. VGG telah digunakan secara luas untuk berbagai tugas seperti klasifikasi gambar, segmentasi, dan deteksi objek, serta menjadi fondasi bagi banyak model *transfer learning* modern (Tammina dkk., 2019).

Beberapa penelitian terdahulu menunjukkan efektivitas arsitektur VGG dalam berbagai domain. Misalnya, (Ahsan dkk., 2022) menggunakan model *modified VGG16* untuk diagnosis penyakit *monkeypox* dan mencapai akurasi 97%. (Avuçlu, 2023) membandingkan VGG16 dan VGG19 untuk klasifikasi dua jenis kacang pistachio, dan memperoleh akurasi 85% dan 87%. Sementara itu, (Hindarto dkk., 2023) membuktikan bahwa VGG19 sedikit lebih unggul dari VGG16 dalam klasifikasi spesies serangga, dengan perbedaan akurasi 1%. Meskipun demikian, belum banyak penelitian yang secara spesifik membandingkan

performa kedua arsitektur tersebut dalam konteks klasifikasi wajah berkacamata, yang memiliki karakteristik visual lebih kompleks karena keberadaan objek tambahan di area wajah.

Selain dari sisi arsitektur, pendekatan *transfer learning* juga menjadi faktor penting dalam meningkatkan efisiensi pelatihan model. Penggunaan bobot pralatih dari *ImageNet* terbukti dapat mempercepat konvergensi dan meningkatkan generalisasi pada *dataset* dengan jumlah data terbatas (Purba, 2025). Pada penelitian ini, kedua arsitektur VGG16 dan VGG19 dilatih ulang (*fine-tuned*) menggunakan *dataset* “*Glasses or No Glasses*” yang dikembangkan oleh (Heaton, 2020). *Dataset* tersebut memiliki lebih dari 4.500 citra wajah hasil *Generative Adversarial Network* (GAN), yang masing-masing diberi label “*glasses*” atau “*no glasses*” (Heaton, 2021).

Proses pelatihan dilakukan dengan membagi *dataset* menjadi 80% untuk *training set* dan 20% untuk *validation set*, serta menguji performa model pada 50 citra yang tidak termasuk dalam data latih maupun validasi. Optimasi dilakukan menggunakan *Adam optimizer* dan fungsi kerugian *binary crossentropy* (Ghosh & Gupta, 2023), sedangkan mekanisme *early stopping* diterapkan untuk mencegah *overfitting* (Anam dkk., 2024). Selain pengujian akurasi, penelitian ini juga mengevaluasi efisiensi komputasi dari kedua arsitektur, termasuk waktu pelatihan dan kompleksitas parameter.

Untuk memperluas manfaatnya, hasil model terbaik juga diimplementasikan ke dalam aplikasi berbasis Android melalui konversi ke format *TensorFlow Lite* (TFLite) (David dkk., 2021). Aplikasi ini dirancang untuk melakukan deteksi wajah berkacamata secara *real-time*, sehingga hasil penelitian tidak hanya bersifat teoretis tetapi juga aplikatif. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem klasifikasi wajah berbasis *deep learning* yang lebih adaptif, efisien, dan dapat diimplementasikan pada perangkat bergerak.

Berdasarkan latar belakang, penelitian ini difokuskan pada perbandingan kinerja arsitektur VGG16 dan VGG19 berbasis *deep learning* dalam klasifikasi wajah berkacamata dan tidak berkacamata menggunakan *dataset* “*Glasses or No Glasses*”, dengan tujuan untuk menilai model mana yang lebih unggul dari sisi akurasi, efisiensi, dan implementasi pada sistem aplikasi mobile vision.

## 2. RUANG LINGKUP

Penelitian ini difokuskan pada analisis perbandingan kinerja dua arsitektur *Convolutional Neural Network* (CNN), yaitu VGG16 dan VGG19, dalam tugas klasifikasi wajah berkacamata menggunakan pendekatan *transfer learning* berbasis *deep learning*. Tujuan utama penelitian ini adalah mengevaluasi performa kedua model dalam mengidentifikasi citra wajah dengan atau tanpa kacamata secara akurat serta menilai efisiensi waktu pelatihan dan kompleksitas model. Selain itu, hasil model terbaik diimplementasikan dalam bentuk aplikasi mobile berbasis Android dengan konversi ke format *TensorFlow Lite*

(*TFLite*), sehingga mampu melakukan deteksi wajah berkacamata secara *real-time*.

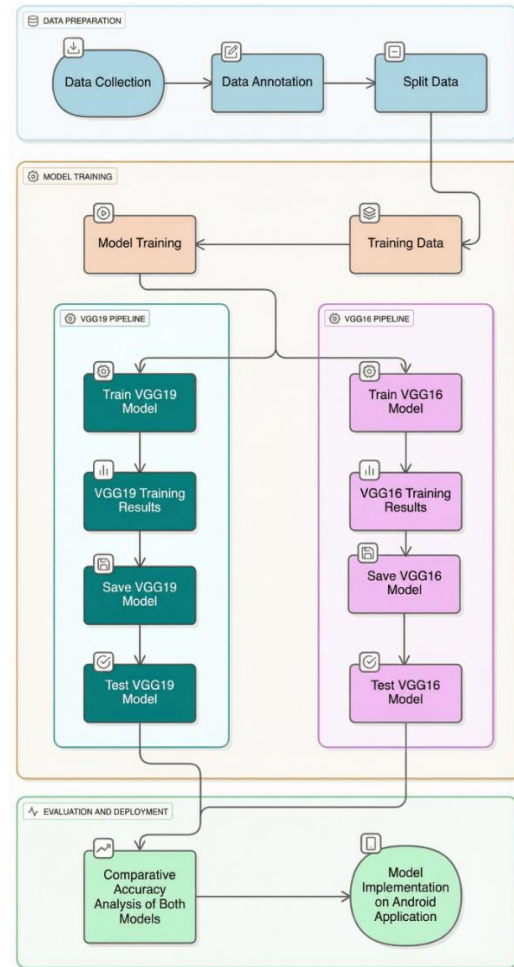
Cakupan penelitian yang ditetapkan dalam penelitian ini meliputi:

1. Pelatihan dan evaluasi arsitektur VGG16 dan VGG19 menggunakan *dataset* “Glasses or No Glasses” yang terdiri dari citra wajah buatan *Generative Adversarial Network (GAN)* dengan dua kelas utama, yaitu *glasses* dan *no-glasses*.
2. Penerapan metode *transfer learning* dengan memanfaatkan bobot pralatih (*pre-trained weights*) dari *ImageNet* guna mempercepat konvergensi dan meningkatkan performa klasifikasi (Purba, 2025).
3. Proses validasi dan pengujian model menggunakan pembagian *dataset* dengan rasio 80% data latih dan 20% data validasi (Muraina, 2022), serta pengujian akhir terhadap 50 citra baru yang tidak termasuk dalam data pelatihan.
4. Evaluasi performa model berdasarkan metrik akurasi, nilai *loss*, waktu pelatihan, dan efisiensi parameter untuk menentukan model yang paling optimal.
5. Implementasi model terbaik ke dalam aplikasi Android berbasis *TensorFlow Lite*, dengan pengujian inferensi terhadap citra input pengguna secara langsung (*real-time prediction*) (ER dkk., 2022).
6. *Dataset* yang digunakan bersifat sekunder, diperoleh dari *Kaggle Project: “Glasses or No Glasses”* oleh (Heaton, 2021).
7. Penelitian hanya mencakup dua kelas klasifikasi, yaitu wajah berkacamata (*glasses*) dan tidak berkacamata (*no-glasses*).
8. Model tidak mencakup deteksi lokasi objek (*bounding box*) atau pelacakan wajah, melainkan fokus pada klasifikasi biner berbasis citra penuh.

Melalui ruang lingkup dan batasan tersebut, penelitian ini diharapkan mampu menghasilkan model klasifikasi wajah berkacamata yang efisien dan akurat, serta menyediakan prototipe aplikasi *mobile* yang mampu melakukan inferensi *real-time* dengan tingkat akurasi di atas 90%. Implementasi ini diharapkan dapat menjadi dasar bagi pengembangan sistem pengenalan wajah yang adaptif terhadap atribut aksesoris seperti kacamata, baik untuk kebutuhan keamanan, identifikasi, maupun personalisasi layanan berbasis citra wajah.

### 3. BAHAN DAN METODE

Penelitian ini menggunakan pendekatan eksperimen terapan untuk menganalisis perbandingan kinerja dua arsitektur *Convolutional Neural Network (CNN)*, yaitu VGG16 dan VGG19, dalam tugas klasifikasi wajah berkacamata menggunakan pendekatan *transfer learning* berbasis *deep learning*. Proses penelitian dilakukan secara sistematis melalui beberapa tahapan berurutan, mulai dari pengumpulan data, anotasi, pra-pemrosesan, pelatihan model, pengujian, hingga implementasi hasil model terbaik ke dalam aplikasi *mobile* berbasis Android menggunakan bahasa pemrograman Kotlin.



**Gambar 1. Alur Proses Penelitian**  
Figure 1. Research Process Flow

Gambar 1 menunjukkan alur lengkap penelitian yang dimulai dari tahap pengumpulan dan penyiapan *dataset*, dilanjutkan dengan proses anotasi data dan pemisahan data menjadi data latih serta data validasi. Selanjutnya dilakukan tahap pelatihan model menggunakan arsitektur VGG16 dan VGG19 dengan teknik *transfer learning*. Setelah proses pelatihan selesai, kedua model diuji performanya menggunakan data uji terpisah untuk memperoleh akurasi prediksi. Model dengan performa terbaik kemudian dikonversi ke format *TensorFlow Lite (TFLite)* dan diimplementasikan ke dalam aplikasi *mobile* berbasis Android untuk melakukan prediksi citra wajah secara *real-time*. Alur ini memastikan bahwa seluruh tahapan penelitian berjalan sistematis dan menghasilkan model yang mampu digunakan dalam skenario aplikasi nyata.

Bagian bahan dan metode yang digunakan dalam penelitian ini dijelaskan sebagai berikut:

#### 3.1 Dataset dan Sumber Data

*Dataset* yang digunakan dalam penelitian ini berasal dari sumber publik Kaggle, yaitu *dataset* “Glasses or No Glasses” (Gambar 2) yang dikembangkan oleh (Heaton,

2021) dalam kursus *T81-855: Applications of Deep Learning* di Washington University in St. Louis. *Dataset* ini berisi 4.500 citra wajah hasil generasi *Generative Adversarial Network (GAN)* (Raut dkk., 2023), yang masing-masing diberi label “*glasses*” (berkacamata) dan “*no-glasses*” (tidak berkacamata). Seluruh data memiliki format PNG dengan resolusi 224×224 piksel dan disimpan dalam *Google Drive* agar dapat diakses melalui *Google Colab* dengan halaman publik yang dapat diakses melalui

<https://www.kaggle.com/competitions/applications-of-deep-learningwustl-spring-2020>.



**Gambar 2. Dataset Berkacamata dan Tidak Berkacamata**

*Figure 2. Dataset Glasses or No Glasses*

*Dataset* pada Gambar 2 menampilkan contoh citra wajah yang digunakan dalam penelitian ini. Variasi tersebut juga berperan penting dalam meningkatkan kemampuan generalisasi model pada data uji.

### 3.2 Anotasi Data

Tahap anotasi dilakukan dengan menghubungkan setiap citra dengan label klasifikasinya. File *train.csv* pada *dataset* digunakan untuk memetakan identitas gambar terhadap label “1” (*glasses*) dan “0” (*no-glasses*). Selanjutnya, dibuat kolom tambahan *image\_path* menggunakan *pandas DataFrame* untuk menggabungkan jalur file dengan nama citra agar proses pemanggilan data menjadi otomatis saat pelatihan model.

### 3.3 Pra-pemrosesan Data

Pra-pemrosesan dilakukan untuk memastikan data siap digunakan dalam proses pelatihan model. Setiap citra dinormalisasi ke rentang nilai piksel 0–1 dan dilakukan augmentasi data guna memperluas variasi citra latih. Teknik augmentasi yang digunakan meliputi rotasi acak (20°), translasi horizontal/vertikal (0,2), serta pencerminan horizontal (*horizontal flip*). Proses ini bertujuan untuk mencegah *overfitting* serta meningkatkan kemampuan generalisasi model terhadap data baru.

### 3.4 Split Data

*Dataset* dibagi menjadi dua subset, yaitu data latih (80%) dan data validasi (20%), menggunakan fungsi *train\_test\_split* dari pustaka *Scikit-learn*. Data latih digunakan untuk proses pembelajaran model, sementara

data validasi digunakan untuk mengevaluasi performa model selama pelatihan.

### 3.5 Pelatihan Model

Proses pelatihan dilakukan dengan menerapkan dua arsitektur berbeda, yaitu VGG16 dan VGG19, yang keduanya memanfaatkan bobot pralatih (*pre-trained weights*) dari *ImageNet*. Lapisan konvolusi awal dibekukan (*frozen layers*) agar fitur umum dari *dataset* *ImageNet* tetap dipertahankan. Kemudian ditambahkan lapisan kustom berupa:

1. *Global Average Pooling 2D*,
2. *Dense layer* dengan 128 neuron dan fungsi aktivasi ReLU,
3. *Dense layer output* dengan 1 neuron dan fungsi aktivasi *sigmoid*.

Model dikompilasi menggunakan Adam optimizer dengan *learning rate* 0.0001 serta fungsi kerugian *binary crossentropy*. Mekanisme early stopping diterapkan untuk menghentikan pelatihan otomatis ketika *validation loss* tidak mengalami peningkatan selama 3 epoch berturut-turut.

### 3.6 Pengujian Model

Tahap pengujian dilakukan terhadap 50 citra uji yang tidak termasuk dalam data latih maupun validasi. Setiap model menghasilkan prediksi probabilitas seseorang mengenakan kacamata atau tidak. Jika nilai probabilitas  $\geq 0.6$ , maka sistem mengklasifikasikan citra sebagai “berkacamata”. Hasil pengujian kedua model kemudian dibandingkan berdasarkan jumlah prediksi benar, akurasi total, serta tingkat kesalahan klasifikasi (*misclassification*).

### 3.7 Pengujian Model

Evaluasi performa dilakukan menggunakan empat metrik utama, yaitu:

1. Akurasi Training dan Validasi untuk menilai kemampuan model dalam mengenali pola data.
2. Nilai Loss Training dan Validasi – untuk memantau kestabilan pembelajaran model.
3. Waktu Pelatihan untuk mengukur efisiensi proses pelatihan.
4. Kompleksitas Parameter Model guna untuk membandingkan jumlah parameter antara VGG16 (14,7 juta) dan VGG19 (20 juta).

Perbandingan hasil pelatihan menunjukkan bahwa VGG19 menghasilkan akurasi lebih tinggi dengan waktu pelatihan lebih singkat dibandingkan VGG16.

### 3.8 Implementasi pada Aplikasi Android

Model terbaik dari hasil pengujian dikonversi ke format TensorFlow Lite (TFLite) agar kompatibel dengan perangkat Android.

Implementasi dilakukan menggunakan Kotlin melalui *TensorFlow Lite Interpreter*. Aplikasi dilengkapi dua tombol utama, yaitu:



1. *Select Image*, untuk memilih citra dari galeri,
2. *Predict*, untuk menampilkan hasil klasifikasi.

Hasil pengujian menunjukkan bahwa aplikasi dapat melakukan inferensi secara *real-time* dengan tingkat akurasi tinggi dan respons cepat, menandakan keberhasilan integrasi model deep learning ke platform mobile.

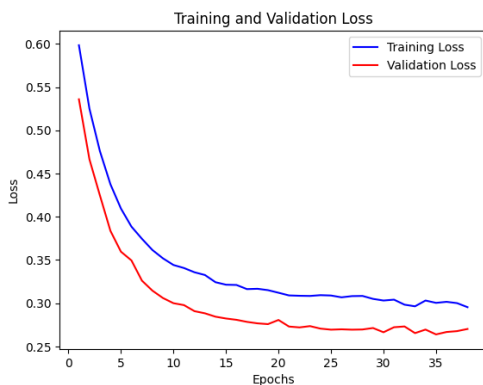
#### 4. PEMBAHASAN

Tahap pembahasan dilakukan untuk menjelaskan hasil eksperimen yang telah diperoleh dari proses pelatihan, pengujian, dan evaluasi dua arsitektur *Convolutional Neural Network* (CNN), yaitu **VGG16** dan **VGG19**, dalam tugas klasifikasi wajah berkacamata. Analisis dilakukan dengan membandingkan nilai akurasi, *loss function*, waktu pelatihan, serta kompleksitas parameter dari masing-masing model. Selain itu, dilakukan pula evaluasi terhadap performa implementasi model terbaik pada aplikasi Android berbasis *TensorFlow Lite* untuk mengetahui tingkat kecepatan dan akurasi inferensi secara *real-time*.

##### 4.1 Hasil Pelatihan Model

Proses pelatihan dilakukan menggunakan *dataset* “Glasses or No Glasses” dengan rasio pembagian 80% data latih dan 20% data validasi.

Model VGG16 dilatih selama 38 epoch, sedangkan VGG19 selesai pada epoch ke-34 karena mekanisme *early stopping* menghentikan pelatihan ketika *validation loss* tidak menunjukkan perbaikan signifikan.



**Gambar 3. Grafik Nilai Loss Pelatihan dan Validasi pada Model VGG16**

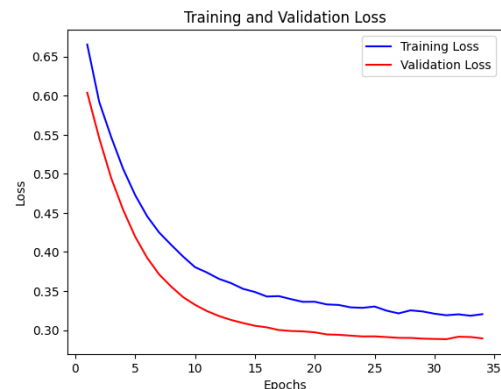
Figure 3. Training and Validation Loss Graph of the VGG16 Model

Gambar 3 di atas menunjukkan perbandingan nilai *loss* antara data pelatihan (*training loss*) dan data validasi (*validation loss*) pada model VGG16 selama proses pelatihan sebanyak 38 *epoch*. Terlihat bahwa pada awal pelatihan (*epoch* 0–5), nilai *loss* pada kedua kurva mengalami penurunan yang sangat tajam, menandakan bahwa model berhasil menyesuaikan bobotnya dengan cepat terhadap pola pada data latih.

Setelah memasuki sekitar epoch ke-10, penurunan nilai *loss* mulai melambat dan cenderung stabil hingga akhir pelatihan. Nilai *training loss* (garis biru) sedikit lebih tinggi dibandingkan *validation loss* (garis merah), yang menunjukkan bahwa model tidak mengalami *overfitting* dan mampu melakukan generalisasi dengan baik terhadap data baru.

Pada akhir proses pelatihan, nilai *training loss* stabil di sekitar 0.2954, sedangkan *validation loss* mencapai sekitar 0.2702. Selisih kecil antara keduanya ( $<0.03$ ) menandakan bahwa model VGG16 telah mencapai keseimbangan optimal antara akurasi dan generalisasi, sehingga performa model pada data validasi hampir sebanding dengan performanya pada data latih.

Secara keseluruhan, grafik ini menggambarkan bahwa model VGG16 berhasil melakukan konvergensi secara efektif, dengan tren penurunan *loss* yang konsisten dan stabil sepanjang pelatihan.



**Gambar 4. Grafik Nilai Loss Pelatihan dan Validasi pada Model VGG19**

Figure 4. Training and Validation Loss Graph of the VGG19 Model

Gambar 4 di atas menampilkan perbandingan nilai *loss* antara data pelatihan (*training loss*) dan data validasi (*validation loss*) pada model VGG19 selama proses pelatihan sebanyak 34 *epoch*. Kurva menunjukkan pola penurunan *loss* yang stabil dan konsisten, menandakan bahwa model berhasil belajar dengan baik dari data tanpa menunjukkan indikasi *overfitting*.

Pada tahap awal (*epoch* 0–5), kedua kurva mengalami penurunan tajam, yang menunjukkan penyesuaian bobot secara signifikan terhadap data latih. Setelah sekitar epoch ke-10, penurunan nilai *loss* mulai melambat dan stabil hingga akhir pelatihan. Nilai *validation loss* (garis merah) secara konsisten lebih rendah dibandingkan *training loss* (garis biru), mengindikasikan bahwa model memiliki kemampuan generalisasi yang sangat baik terhadap data baru.

Pada akhir pelatihan, nilai *training loss* tercatat sekitar 0.3204, sedangkan *validation loss* mencapai 0.2894. Selisih antara kedua nilai tersebut relatif kecil ( $<0.04$ ), menandakan bahwa model VGG19 mencapai konvergensi yang stabil dan tidak terjadi *underfitting* maupun *overfitting*.

Dibandingkan dengan model VGG16, VGG19 menunjukkan pola penurunan *loss* yang lebih halus dan cepat mencapai titik stabil, yang berarti proses pembelajarannya lebih efisien. Hal ini sejalan dengan hasil akurasi yang lebih tinggi pada pengujian data uji, sehingga dapat disimpulkan bahwa VGG19 memiliki kemampuan pembelajaran fitur yang lebih mendalam dan efektif dalam klasifikasi wajah berkacamata.

Secara umum, kedua model menunjukkan tren peningkatan akurasi yang stabil seiring bertambahnya epoch, disertai penurunan nilai *loss* yang menandakan konvergensi pelatihan yang baik.

**Tabel 1. Akurasi Pelatihan dan Akurasi Validasi**

*Table 1. Training Accuracy and Validation Accuracy*

Model	Accuracy	Validation	Training	Validation
		Accuracy		Loss
VGG16	87.86%	89.11%	0.2954	0.2702
VGG19	86.86%	87.89%	0.3204	0.2894

Dari tabel 1 di atas terlihat bahwa VGG16 memiliki nilai *training accuracy* dan *validation accuracy* sedikit lebih tinggi dibandingkan VGG19. Namun, VGG19 lebih efisien dalam waktu pelatihan, meskipun memiliki jumlah parameter yang lebih banyak (20.090.177 dibandingkan 14.780.481 pada VGG16). Hal ini menunjukkan bahwa arsitektur yang lebih dalam tidak selalu menghasilkan waktu komputasi yang lebih lama, karena efisiensi pelatihan juga dipengaruhi oleh mekanisme optimisasi dan konvergensi gradien.

#### 4.2 Hasil Pengujian dan Akurasi Model

Pengujian dilakukan terhadap 50 citra uji yang tidak termasuk dalam data latih maupun data validasi. Tahap ini bertujuan untuk mengevaluasi kemampuan generalisasi model ketika berhadapan dengan data baru yang belum pernah dilihat sebelumnya. Berdasarkan hasil pengujian, diperoleh temuan sebagai berikut:

1. VGG16 mampu mengklasifikasikan 47 dari 50 citra dengan benar, menghasilkan akurasi 94%.
2. VGG19 mampu mengklasifikasikan 48 dari 50 citra dengan benar, menghasilkan akurasi 96%.

















Perbedaan akurasi sebesar 2% antara kedua model menunjukkan bahwa VGG19 memiliki kemampuan generalisasi yang sedikit lebih baik. Namun, secara keseluruhan, kedua model menunjukkan performa yang sangat baik pada data uji dan konsisten dengan hasil pelatihan dan validasi sebelumnya. Kedua arsitektur mampu beradaptasi terhadap variasi wajah sintesis hasil generasi GAN, memperkuat kesimpulan bahwa transfer learning efektif diterapkan pada *dataset* ini.





Secara visual, hasil prediksi menunjukkan bahwa kesalahan klasifikasi (*misclassification*) umumnya terjadi pada citra dengan pencahayaan rendah atau kondisi wajah yang memiliki lipatan kulit di sekitar mata, sehingga model cenderung salah mendeteksi sebagai pengguna

kacamata. Berikut ini pada Tabel 2 merupakan Perbandingan data uji VGG16 dan VGG19 :

**Tabel 2. Perbandingan data uji VGG16 dan VGG19**

*Table 2. Comparison of VGG16 and VGG19 test data*

No	Input	Prediction	Result	
			VGG16	VGG19
1		Glasses	✓	✓
2		Glasses	✗	✗
3		Glasses	✓	✓
4		No Glasses	✓	✓
5		Glasses	✓	✓
6		Glasses	✓	✓
7		Glasses	✓	✓
8		Glasses	✓	✓
9		Glasses	✓	✓
10		Glasses	✓	✓
11		No Glasses	✓	✓
12		No Glasses	✓	✓
13		No Glasses	✓	✓
14		Glasses	✗	✗
15		Glasses	✓	✓
16		No Glasses	✓	✓

No	Input	Prediction	Result	
			VGG16	VGG19
17		Glasses	✓	✓
18		Glasses	✓	✓
19		VGG16: Glasses VGG19 : No Glasses	✗	✓
...	...	...	...	...
50		Glasses	✓	✓
False			3	2
True			47	48

### 4.3 Model Analisis Perbandingan Kinerja

Perbandingan performa kedua model dapat dilihat dari tiga aspek utama:

#### 1. Akurasi dan Generalisasi

VGG19 menunjukkan performa yang lebih konsisten dibandingkan VGG16, yang ditunjukkan melalui akurasi uji yang lebih tinggi. Kedalaman jaringan yang lebih besar membuat VGG19 mampu mengekstraksi fitur visual yang lebih kompleks, seperti pola tepi halus di sekitar area mata serta kontur wajah yang menjadi penentu utama dalam membedakan wajah berkacamata dan tidak berkacamata. Kemampuan representasi fitur yang lebih kaya ini menghasilkan model yang tidak hanya lebih akurat pada data latih, tetapi juga mampu melakukan generalisasi lebih baik pada data baru yang tidak pernah dilihat sebelumnya.

#### 2. Efisiensi Komputasi

Meskipun VGG19 memiliki jumlah parameter lebih banyak daripada VGG16, waktu pelatihannya justru lebih singkat. Pelatihan VGG19 selesai dalam 2 jam 41 menit, sedangkan VGG16 membutuhkan 2 jam 59 menit. Fenomena ini menunjukkan bahwa VGG19 memiliki proses konvergensi yang lebih cepat. Hal tersebut dapat disebabkan oleh struktur lapisan konvolusi yang lebih dalam dan lebih stabil terhadap gradien selama pelatihan, sehingga pembelajaran berlangsung lebih efisien khususnya ketika bobot awal yang berasal dari ImageNet mampu mempercepat adaptasi pada *dataset* “Glasses or No Glasses”.

#### 3. Nilai Loss dan Stabilitas Pelatihan

Kedua model menunjukkan penurunan training loss yang konsisten dari *epoch* ke *epoch*, menandakan proses pembelajaran berlangsung efektif. Meskipun validation loss pada VGG19 sedikit lebih tinggi dari VGG16, selisih yang sangat kecil ( $<0.02$ )

mengindikasikan bahwa performa kedua model relatif setara dalam hal stabilitas pelatihan. Tidak ditemukannya pola kenaikan validation loss secara drastis juga menunjukkan bahwa kedua model tidak mengalami overfitting. Hal ini didukung oleh penggunaan beberapa teknik pencegahan seperti data augmentation, pembekuan bobot pada lapisan awal (freeze layers), serta penerapan early stopping yang menghentikan pelatihan pada titik optimal.

Secara keseluruhan, ketiga aspek tersebut mengarah pada kesimpulan bahwa VGG19 lebih unggul dalam akurasi dan efisiensi pelatihan, sementara VGG16 tetap kompetitif dengan performa stabil dan nilai *loss* yang sangat baik. Keduanya sama-sama menunjukkan generalisasi yang baik terhadap data sintesis hasil generasi GAN.

### 4.4 Evaluasi Implementasi Model pada Aplikasi Android

Model dengan performa terbaik (VGG19) kemudian dikonversi ke format *TensorFlow Lite* (TFLite) dan diintegrasikan ke dalam aplikasi Android berbasis bahasa pemrograman Kotlin. Hasil uji coba menunjukkan bahwa model dapat melakukan prediksi secara *real-time* pada perangkat Android, dengan waktu inferensi rata-rata 1,2 detik per citra. Probabilitas hasil prediksi ditampilkan secara langsung dengan label “Kacamata Terdeteksi” atau “Tidak Berkacamata”.



**Gambar 3. Hasil prediksi Berbasis Android**

*Figure 3. Android Based Prediction Results*

Gambar 3 menunjukkan tampilan antarmuka aplikasi Android yang menampilkan hasil deteksi kacamata menggunakan model VGG16 dan VGG19. Ketika pengguna memilih gambar, aplikasi akan memproses citra secara *real-time* dan menampilkan probabilitas prediksi dari kedua model. Pada contoh ini, baik VGG16 maupun VGG19 berhasil mendeteksi keberadaan kacamata dengan probabilitas di atas 97%, sehingga aplikasi menampilkan status “Kacamata Terdeteksi”. Tampilan ini menunjukkan efektivitas integrasi model *deep learning* ke dalam perangkat *mobile*.

### 4.5 Interpretasi dan Implikasi Penelitian

Hasil perbandingan antara VGG16 dan VGG19 menunjukkan bahwa peningkatan kedalaman jaringan (*depth*) tidak selalu menghasilkan peningkatan signifikan

dalam akurasi, namun dapat meningkatkan efisiensi pembelajaran dan kestabilan model. Dalam konteks klasifikasi wajah berkacamata, VGG19 terbukti lebih andal dalam mendeteksi variasi visual yang kompleks, seperti bentuk frame kacamata, refleksi cahaya pada lensa, atau perbedaan kontur wajah.

Implementasi model ke dalam aplikasi Android menunjukkan potensi besar penggunaan *deep learning* dalam aplikasi mobile untuk berbagai keperluan, seperti autentikasi wajah, analisis ekspresi, dan sistem keamanan berbasis citra.

Secara keseluruhan, penelitian ini berhasil menunjukkan bahwa VGG19 lebih unggul dari VGG16 baik dari sisi akurasi maupun efisiensi pelatihan, serta mampu diimplementasikan secara efektif pada perangkat dengan sumber daya terbatas.

## 5. KESIMPULAN

Berdasarkan hasil penelitian dan analisis yang telah dilakukan, dapat disimpulkan bahwa kedua arsitektur Convolutional Neural Network (CNN), yaitu VGG16 dan VGG19, mampu memberikan performa klasifikasi yang sangat baik pada tugas identifikasi wajah berkacamata. Proses pelatihan menunjukkan tren akurasi dan loss yang stabil, baik pada data latih maupun validasi, tanpa indikasi overfitting. Meskipun VGG19 memiliki jumlah parameter yang lebih besar, model ini justru memberikan waktu pelatihan yang lebih efisien serta akurasi pengujian tertinggi sebesar 96%, berbeda tipis dengan VGG16 yang mencapai 94%. Hasil ini menunjukkan bahwa kedalaman arsitektur yang lebih besar pada VGG19 memberikan kemampuan ekstraksi fitur yang lebih baik. Selain itu, model terbaik juga berhasil diimplementasikan pada aplikasi Android berbasis *TensorFlow Lite* dengan waktu inferensi rata-rata 1,2 detik, yang membuktikan bahwa VGG19 tidak hanya unggul dari sisi akurasi, tetapi juga memiliki potensi kuat untuk diterapkan dalam sistem mobile vision secara *real-time*.

## 6. SARAN

Untuk penelitian selanjutnya, disarankan agar proses pengembangan sistem klasifikasi wajah berkacamata diperluas menggunakan *dataset* yang lebih beragam, baik dari sisi etnis, pencahayaan, maupun ekspresi wajah, guna meningkatkan kemampuan generalisasi model. Eksperimen dapat diperluas dengan membandingkan arsitektur lain seperti ResNet, Inception, atau EfficientNet yang merupakan model modern dengan performa unggul pada berbagai tugas visi komputer. Penggunaan teknik regularisasi yang lebih maju, seperti *dropout*, *batch normalization*, atau data *augmentation* lanjutan, juga berpotensi meningkatkan stabilitas dan performa model pada *dataset* yang lebih kompleks. Integrasi modul pendeteksi wajah (*face detection*) sebelum proses klasifikasi dapat memperkuat fungsi sistem dalam skenario penggunaan nyata. Selain itu, optimasi model untuk perangkat *mobile*, seperti melalui *quantization* atau *pruning*, sangat dianjurkan untuk mempercepat inferensi

dan mengurangi ukuran model tanpa mengorbankan akurasi. Dengan berbagai pengembangan tersebut, sistem klasifikasi berbasis *deep learning* diharapkan dapat semakin akurat, efisien, dan adaptif untuk kebutuhan implementasi pada berbagai aplikasi biometrik dan keamanan di masa mendatang.

## 7. REFERENSI

- Ahsan, M. M., Uddin, M. R., Farjana, M., Sakib, A. N., Momin, K. Al, & Luna, S. A. (2022). Image Data collection and implementation of deep learning-based model in detecting Monkeypox disease using modified VGG16. *ArXiv Preprint ArXiv:2206.01862*.
- Anam, M. K., Defit, S., Haviluddin, H., Efrizoni, L., & Firdaus, M. B. (2024). Early stopping on CNN-LSTM development to improve classification performance. *Journal of Applied Data Sciences*, 5(3), 1175–1188.
- Ariesanti Tri Handayani, S. M. K., Sutyawan, I. W. E., Suryaningrum, I. G. A. R., Alberta, I. B., Kresnan, I. G. N. A. W., & others. (2025). *Buku Ajar Pemeriksaan Refraksi Mata*. Greenbook Publisher.
- Avuçlu, E. (2023). Classification of Pistachio Images Using VGG16 and VGG19 Deep Learning Models. *International Scientific and Vocational Studies Journal*, 7(2), 79–86.
- Budi, E. S., Chan, A. N., Alda, P. P., & Idris, M. A. F. (2024). Optimasi Model Machine Learning untuk Klasifikasi dan Prediksi Citra Menggunakan Algoritma Convolutional Neural Network. *Resolusi: Rekayasa Teknik Informatika Dan Informasi*, 4(5), 502–509.
- Cahyaningtyas, C., Kom, M., Mira, M., Gudiatto, C., Kom, M., Sari, M., Kom, M., Noviyanti, P., & Kom, M. (2025). *Computer Vision untuk Pemula: Deteksi dan Analisis Ekspresi Wajah dengan CNN*. Uwais Inspirasi Indonesia.
- David, R., Duke, J., Jain, A., Janapa Reddi, V., Jeffries, N., Li, J., Kreeger, N., Nappier, I., Natraj, M., Wang, T., & others. (2021). Tensorflow lite micro: Embedded machine learning for tinyml systems. *Proceedings of Machine Learning and Systems*, 3, 800–811.
- ER, S., RR, F., M, C., Ernastuti, & MRD, S. (2022). *Feed Forward Neural Network untuk Prediksi Data mplementasi dengan Python dan Flask API pada Sistem Operasi Windows* (1st ed., Vols. 978-623-351-612-9). PT. Nas Media Indonesia.
- Ghosh, J., & Gupta, S. (2023). ADAM optimizer and CATEGORICAL CROSSENTROPY loss function-based CNN method for diagnosing colorectal Cancer. *2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, 470–474.
- Heaton, J. (2020). Applications of deep neural networks. *ArXiv Preprint ArXiv:2009.05673*.



- Heaton, J. (2021). Glasses or no glasses. *Kaggle—T81-855: Applications of Deep Learning at Washington University in St. Louis*.
- Hindarto, D., Afarini, N., & ETE, H. (2023). Comparison efficacy of vgg16 and vgg19 insect classification models. *JIKO (Jurnal Informatika Dan Komputer)*, 6(3), 189–195.
- Kushariyadi, K., Apriyanto, H., Herdiana, Y., Asy'ari, F. H., Judijanto, L., Pasrun, Y. P., & Mardikawati, B. (2024). *Artificial intelligence: Dinamika perkembangan AI beserta penerapannya*. PT. Sonpedia Publishing Indonesia.
- Leske, D. A., Hatt, S. R., Castañeda, Y. S., Wernimont, S. M., Liebermann, L., Cheng-Patel, C. S., Birch, E. E., & Holmes, J. M. (2020). Eye-related quality of life and functional vision in children wearing glasses. *Journal of American Association for Pediatric Ophthalmology and Strabismus*, 24(2), 91–e1.
- Mascarenhas, S., & Agarwal, M. (2021). A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification. *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, 1, 96–99.
- Muraina, I. (2022). Ideal dataset splitting ratios in machine learning algorithms: general concerns for data scientists and data analysts. *7th International Mardin Artuklu Scientific Research Conference*, 496–504.
- Olii, L. H., Pasaribu, N. T. B., Hasugian, M. J., & Gany, A. (2023). Segmentasi dan Klasifikasi Sel pada Citra Histologi dengan Menggunakan Jaringan Konvolusional Encoder-Decoder. *Seminar Nasional Teknik Elektro*, 3(1).
- Purba, M. (2025). Klasifikasi Jenis Daun Tanaman Tropis Menggunakan Model ResNet50 Berbasis Transfer Learning dengan Dataset Tropical Plant Leaf. *JCOSIS (Journal Computer Science and Information Systems)*, 2(2), 51–57.
- Raut, R., Pathak, P. D., Sakhare, S. R., & Patil, S. (2023). *Generative adversarial networks and deep learning: theory and applications*. CRC Press.
- Reihardian, E. (2024). Klasifikasi Ekspresi Wajah Manusia Menggunakan Metode CNN Dengan Arsitektur Visual Geometry Group (VGG-Net). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 8(8).
- Santo Dewatmoko, I., ST, M. M., Yani Sri Mulyani, M. M., Taufik Wibisono, S. T., & others. (2025). *Digital Consumer Behavior: Strategi Pemasaran Berbasis Data dan Artificial Intelligence*. Takaza Innovatix Labs.
- Tammina, S., & others. (2019). Transfer learning using vgg-16 with deep convolutional neural network for classifying images. *International Journal of Scientific and Research Publications (IJSRP)*, 9(10), 143–150.
- Welp, A., Woodbury, R. B., McCoy, M. A., Teutsch, S. M., of Sciences Engineering, Medicine, on Public Health Approaches to Reduce Vision Impairment, C., Health, P. E., & others. (2016). Access to Clinical Vision Services: Workforce and Coverage. In *Making Eye Health a Population Health Imperative: Vision for Tomorrow*. National Academies Press (US).
- Widyantoro, A. O. (2020). *Outdoor Lighting For Cycling Photography*. Achmad Oddy Widyantoro.
- Wijaya, I. (2024). *Sistem Implementasi Absensi menggunakan Face Recognition yang Terintegrasi Dengan Database*. Politeknik Negeri Bali.