

Klasifikasi Bentuk Bingkai (Frame) Kacamata Menggunakan CNN dengan Arsitektur Inception V3 dan Augmented Reality Berbasis Android

Valdy Ramadhan¹, Mochamad Wisuda Sardjono², Margi Cahyanti³, Ericks Rachmat Swedia⁴, M. Ridwan Dwi Septian⁵

^{1,4,5} Program Studi Informatika, Fakultas Teknologi Industri, Universitas Gunadarma

^{2,3} Program Studi Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Gunadarma

Email: valdyramadhann@gmail.com, moch_wisuda@staff.gunadarma.ac.id,
margi@staff.gunadarma.ac.id, ericks_rs@staff.gunadarma.ac.id,
ridwandwiseptian@staff.gunadarma.ac.id

ABSTRAK

Kacamata tidak hanya menjadi salah satu jenis alat bantu penglihatan bagi penderita penyakit mata, tetapi juga menjadi salah satu bagian pada dunia fashion yang semakin populer. Pemilihan desain bingkai kacamata dapat mempengaruhi penampilan seseorang dalam berbusana sehingga untuk menentukan pilihan harus memperhatikan dua aspek penting, yaitu gaya dan kenyamanan, serta dapat mengubah kesan pada wajah seseorang. Dalam merancang bingkai kacamata perlu berdasarkan ilmu pengukuran tubuh manusia, karena masing-masing ukuran dan bentuk pada organ manusia berbeda satu sama lain. Sehingga dengan keberagaman bentuk wajah manusia tersebut menjadi sangat penting dalam membuat pemilihan bingkai kacamata dan tantangan dalam melakukan penelitian untuk membangun aplikasi untuk merekomendasikan bingkai kacamata sesuai dengan bentuk wajah. Dengan era teknologi saat ini, dimungkinkan menerapkan kecerdasan buatan (AI) dan *Machine Learning* (ML) menjadi solusi terbaik menjawab tantangan tersebut. Beberapa penelitian telah mencoba mengklasifikasikan bentuk wajah dengan menggunakan ML, dengan hasil terbaik menggunakan arsitektur *Inception V3*. Dalam penelitian ini dikembangkan aplikasi berbasis Unity 3D yang menggabungkan *Augmented Reality* (AR) dengan ML untuk merekomendasikan bentuk bingkai kacamata berdasarkan bentuk wajah. Hasil pelatihan model *Inception V3* menunjukkan peningkatan kinerja seiring waktu. Namun, perlu mengatasi *overfitting* pada data validasi. Pada pengujian data test, model mencapai akurasi sekitar 78.6%, menunjukkan kemampuan prediksi yang baik. Teknologi ini memiliki potensi untuk membantu konsumen membuat keputusan yang lebih tepat dalam pemilihan kacamata.

Kata Kunci : *Bentuk Bingkai Kacamata, Bentuk Wajah, Inception V3, Augmented Reality, C#, Python, CNN*

ABSTRACT

Glasses are not only a type of vision aid for people with eye diseases, but they are also an increasingly popular part of the fashion world. The choice of eyeglass frame design can influence a person's appearance in clothing, so when making a choice you must pay attention to two important aspects, namely style and comfort, and can change the impression on a person's face. When designing eyeglass frames, it is necessary to use the science of measuring the human body, because each human organ's size and shape are different from each other. So, with the diversity of

human facial shapes, it becomes very important in making the choice of eyeglass frames and the challenge in conducting research to build an application to recommend eyeglass frames according to face shape. With the current technological era, it is possible to apply Artificial Intelligence (AI) and Machine Learning (ML) to be the best solution to answer these challenges. Several studies have tried to classify facial shape using ML, with the best results using the Inception V3 architecture. In this research, a Unity 3D-based application was developed that combines Augmented Reality (AR) with ML to recommend eyeglass frame shapes based on face shape. Inception V3 model training results show performance improvements over time. However, it is necessary to overcome overfitting in validation data. In testing the test data, the model achieved an accuracy of around 78.6%, indicating good prediction ability. This technology has the potential to help consumers make more informed decisions when selecting glasses

Keywords: *Eyeglasses Frame Shape, Face Shape, Inception V3, Augmented Reality, C#, Python, CNN*

1. PENDAHULUAN

Pemilihan bingkaiacamata menjadi salah satu bagian penting dalam berbusana, saat iniacamata tidak sekadar sebagai alat bantu penglihatan bagi penderita penyakit mata (Zhao, Zhou, & Zhang, 2020). Dalamacamata terdapat 2 bagian, diantaranya: lensa dan bingkai. Dari 2 bagian ini, bingkaiacamata menjadi berpengaruh dalam dunia fasion atau berbusana. Seseorang ingin berpenampilan menarik dengan berkacamata tidak perlu memiliki masalah pada mata, cukup memilih bingkaiacamata. Dalam memilih perlu diperhatikan bentuk bingkai, warna bingkai, dan bahan bingkai. Hal tersebut akan mempengaruhi gaya dan kenyamanan seseorang dalam menggunakanacamata (Graham & Ritchie, 2019).

Dalam membentuk bingkaiacamata perlu diperhatikan pengukuran dan seni penerapan yang menetapkan geometri fisik, sifat massa, dan kemampuan kekuatan tubuh manusia (Krzyszowski, et al., 2023). Dengan menggunakan antropometri, dimensi suatu produk harus dapat dicari dari dimensi tubuh manusia. Antropometri telah diakui sebagai hal penting dalam keberhasilan merancang dan mengukur helm pelindung kepala,acamata, dan respirator (Krzyszowski, et al., 2023). Menentukan pilihan bingkaiacamata salah satunya dipengaruhi dengan memperhatikan bentuk wajah setiap individu, disebabkan oleh keragaman bentuk wajah yang dimiliki oleh setiap individu (Young, Natalia, Sudirman, & Seong Ko, 2019). Sehingga ketika seseorang ingin menggunakanacamata, hal pertama yang dilakukan adalah mencoba semua bingkaiacamata yang terdapat pada optikacamata. Mengakibatkan butuh waktu lama dalam menentukan pilihan karena banyaknya bingkai yang dicoba. Salah satu tantangan menemukan bingkaiacamata yang cocok untuk seseorang, karena pilihan yang salah dapat mengurangi kenyamanan saat digunakan dan estetika penampilan. Keberagaman bentuk dan dimensi bingkaiacamata seringkali membuat konsumen merasa kesulitan dalam menentukan pilihan yang sesuai dengan bentuk wajah konsumen. Sebagian besar konsumen tidak memiliki pengetahuan tentang bentuk wajah dan tipe wajah yang seharusnya dipertimbangkan dalam memilih atau menentukan bingkaiacamata (Hapsari, Gernowo, & Widodo, 2020). Setiap individu mengharapkan bantuan untuk penilaian kesesuaianacamata untuk ukuran dan juga aspek estetika dari bingkaiacamata (Brooks & Borish, 2007). Penelitian sebelumnya mengenai pemilihan bingkaiacamata telah dilakukan

dengan menggunakan algoritma CART dengan penerapan model morfologi indeks wajah dalam identifikasi bentuk wajah (Hapsari, Gernowo, & Widodo, 2020). Algoritma CNN digunakan dalam memilih bentuk bingkai kacamata juga telah diterapkan dengan klasifikasi bentuk wajah pengguna yang merupakan oval atau bukan oval, menggunakan kacamata yang telah direkomendasikan (Young, Natalia, Sudirman, & Seong Ko, 2019).

Semakin banyak teknologi canggih yang dapat memudahkan kegiatan di era digital, pemanfaatan *Artificial Intelligence* (AI) sebagai sebuah teknik untuk meniru kemampuan intelektual yang dimiliki oleh manusia atau bahkan benda mati, dengan tujuan untuk menyelesaikan berbagai macam masalah (Ismawati & Ramadhanti, 2022). *Machine Learning* (ML) adalah subbidang dari *Artificial Intelligence* (AI) yang fokus pada pengembangan algoritma dan model komputer yang memungkinkan sistem untuk belajar dan meningkatkan kinerjanya secara otomatis dari pengalaman atau data. Dalam ML, sistem komputer dapat mengenali pola dalam data, membuat prediksi, dan mengambil keputusan tanpa perlu diprogram secara eksplisit untuk melakukan tugas tertentu. Kemampuan sistem untuk belajar dan beradaptasi dengan data baru menjadikan ML sangat relevan dalam berbagai aplikasi, termasuk pengenalan wajah, klasifikasi teks, prediksi peramalan cuaca, mobil otonom, dan banyak lagi. Perkembangan dalam bidang *machine learning* telah membuka peluang untuk mengembangkan algoritma yang mampu merekomendasikan bingkai kacamata berdasarkan bentuk wajah (Mehta & Mahmoud, 2022). Dalam situasi ini, *machine learning* dapat menjadi solusi yang sangat berguna untuk membantu konsumen dalam membuat keputusan yang lebih tepat.

Penelitian sebelumnya menyatakan bahwa ada berbagai metode yang dapat digunakan untuk mengklasifikasikan bentuk wajah dengan menggunakan *machine learning*. Beberapa metode yang telah diuji meliputi *K-Nearest Neighbors* (KNN), *Linear Discriminant Analysis* (LDA), *Support Vector Machines with Linear Kernel* (SVM-LIN), *Support Vector Machines with Radial Basis Function Kernel* (SVM-RBF), dan *Multi-Layer Perceptron* (MLP). Hasil dari penelitian ini menunjukkan bahwa metode yang memiliki akurasi paling baik dalam melakukan klasifikasi bentuk wajah adalah menggunakan arsitektur *Inception V3* (Emmanuel & Tio, 2019). Metode *Support Machine Vector* (SVM) menjadi salah satu metode yang sering digunakan dalam klasifikasi wajah, dimana proses tersebut dilakukan dengan mengklasifikasikan sebuah gambar sebelumnya melalui sebuah data *training* (Rizal, Girsang, & Prasetyo, 2019). Sistem identifikasi bentuk wajah untuk pemilihan frame kacamata menggunakan metode *transfer learning* telah dilakukan dengan arsitektur MobileNet dan divisualisasikan dengan web melalui *browser* (Hidayatillah, Mardiyantoro, & Hidayat, 2022).

2. METODE PENELITIAN

2.1. *Augmented Reality*

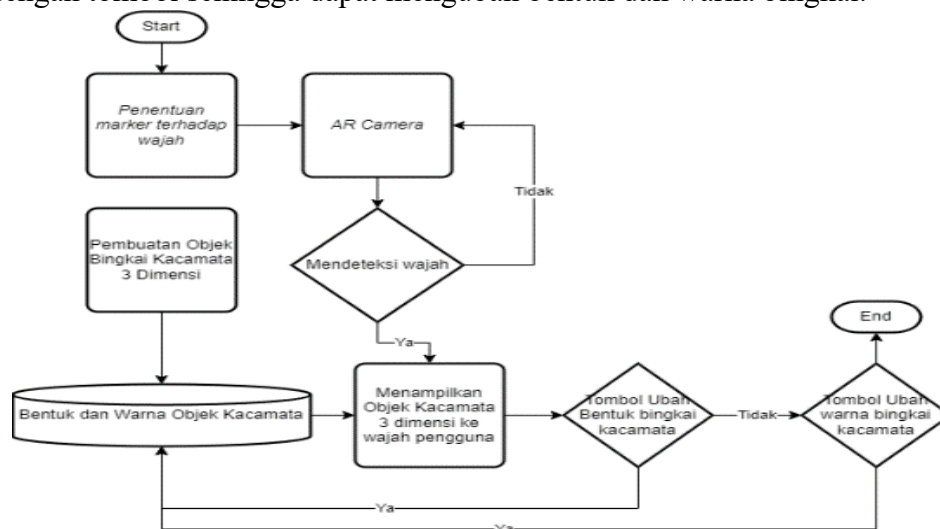
Augmented Reality merupakan sebuah teknologi yang dapat menampilkan objek secara virtual dengan bentuk 2 atau 3 dimensi dalam lingkungan nyata. Teknologi ini banyak dimanfaatkan dalam bidang yang membutuhkan visualisasi secara nyata melalui perangkat komputer (Firmantara, Mudakir, & Nuriman, 2023).

Pada penelitian ini dimanfaatkan teknologi *Augmented Reality* (AR) bingkai kaca mata menggunakan Unity 3D berbasis Android untuk memunculkan objek 3 dimensi bingkai kaca mata ke wajah pengguna agar dapat mencoba kaca mata tanpa menyentuh secara fisik.

Penelitian sebelumnya mengenai pembuatan objek 3 dimensi bingkai kaca mata menggunakan perangkat lunak Blender 3D, warna bingkai kaca mata dapat diubah, serta wajah pengguna aplikasi sebagai *marker*. Sehingga ketika kamera pada *smartphone* mengarah ke wajah pengguna maka akan tampil bingkai kaca mata berbentuk 3 dimensi dengan berbagai pilihan warna yang telah dibuat menggunakan perangkat lunak Blender 3D.

Dapat dilihat pada Gambar 1, mengenai bagan proses dari aplikasi AR. Kemudian teknologi tersebut dikembangkan pada penelitian ini dengan menerapkan model *machine learning* menggunakan arsitektur *Inception V3* untuk memindai bentuk wajah sehingga dapat diakses oleh pengguna melalui aplikasi yang dibuat dengan Unity 3D. Aplikasi Unity 3D tidak dapat menjalankan *script* python untuk mengeksekusi *machine learning* yang telah dibuat, sehingga dalam penelitian ini digunakan layanan *cloud* agar dapat menerapkan *machine learning* ke aplikasi sehingga dapat mengintegrasikan keduanya.

Pembuatan aplikasi dimulai dari penentuan *marker* terhadap wajah. *Marker* digunakan agar dapat menandakan program untuk menampilkan objek dalam aplikasi yaitu wajah. Dengan *marker*, kamera AR akan mencari wajah hingga terdeteksi. Setelah wajah terdeteksi maka aplikasi akan menampilkan objek kaca mata dalam bentuk 3 dimensi yang diambil dari database kemudian objek kaca mata diposisikan pada area mata. Tersedia fitur ubah bentuk bingkai dan warna bingkai kaca mata dengan tombol sehingga dapat mengubah bentuk dan warna bingkai.



Gambar 1. Bagan Proses Penelitian Sebelumnya

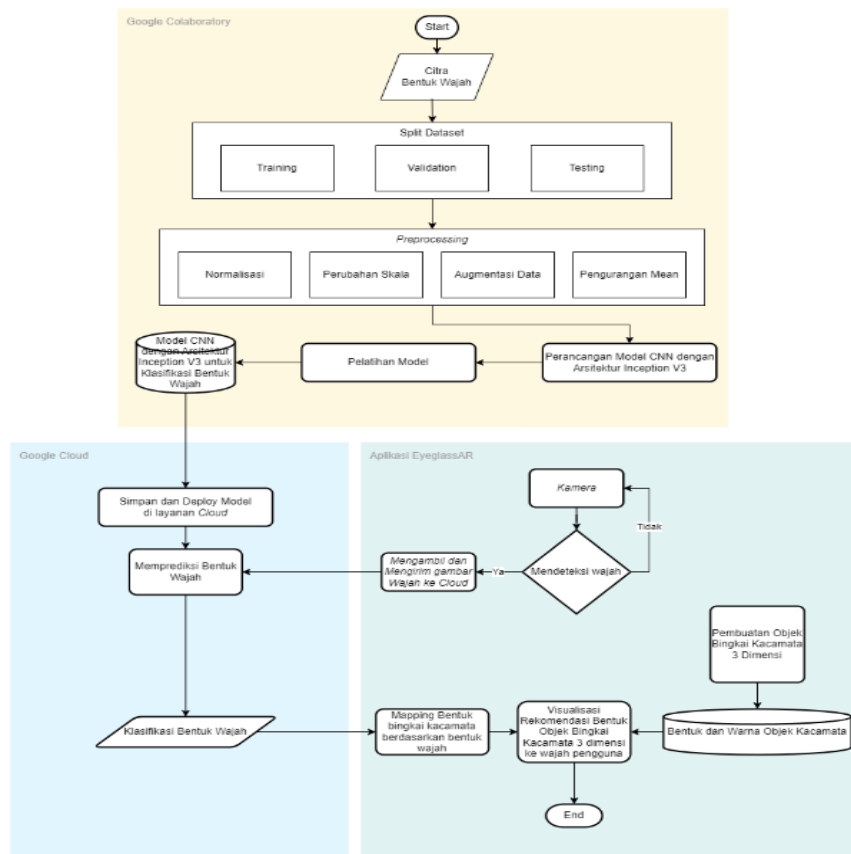
2.2. *Inception V3*

Dengan arsitektur ini dapat memberikan klasifikasi bentuk wajah dengan baik (Pratiwi & Pardede, 2023) (Hidayat, Astuti, Yaqin, Tjilen, & Arifianto, 2023).

Dalam penelitian ini peneliti menerapkan model *machine learning* menggunakan CNN dengan arsitektur *Inception V3* untuk memindai bentuk wajah seseorang. Sehingga pada penelitian ini arsitektur tersebut digunakan untuk memberikan rekomendasi bentuk bingkai kaca mata

berdasarkan bentuk wajah sehingga dapat memberikan visualisasi melalui AR dari hasil prediksi *machine learning* tersebut.

Pada Gambar 2. menampilkan aliran proses dalam penelitian ini, yang merupakan pengembangan dari pemanfaatan AR untuk menampilkan jenis bingkai kacamata yang sesuai dengan klasifikasi bentuk wajah melalui perangkat Android. Perangkat *smartphone* digunakan untuk mempermudah dalam mencoba atau menampilkan bentuk bingkai kacamata tanpa secara fisik, kapanpun dan dimanapun dapat dilakukan. Dalam penelitian ini tahap pertama yang dilakukan adalah pengumpulan data citra wajah sebanyak 5000 citra gambar bentuk wajah orang berasal dari Kaggle “*Face Shape dataset*”, terdiri dari 5 kelas: *heart*, *oblong*, *oval*, *round*, dan *square* (Lama, 2019). Kumpulan gambar ini dikelompokkan menjadi tiga bagian: *training set*, *validation set*, dan *testing set*. Selanjutnya dilakukan *preprocessing* data dengan melakukan augmentasi data, perancangan model CNN menggunakan arsitektur *Inception V3* dan bahasa pemrograman python diimplementasikan di *Google Colaboratory*. *Training set* dan *validation set* dilakukan *preprocessing* menggunakan CNN dengan arsitektur *Inception V3*, serta *confusion matrix* untuk menghasilkan model yang optimal. *Testing set* digunakan untuk melakukan pengujian model klasifikasi bentuk wajah. *Testing set* ini merupakan data citra gambar bentuk wajah yang tidak terdapat pada *training* dan *validation set*.



Gambar 2. Bagan Proses Penelitian Pengembangan Aplikasi

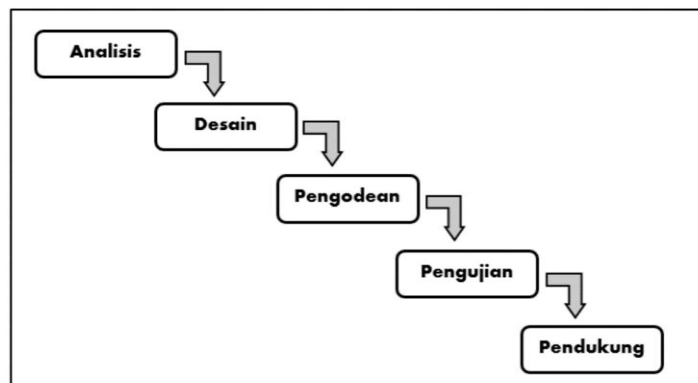
Setelah menguji model klasifikasi bentuk wajah yang optimal tersebut, kemudian model di-*deploy* menggunakan layanan *cloud*. *Google Cloud Platform* digunakan dalam penelitian ini sebagai layanan *cloud* untuk menyimpan model melalui *Google Cloud Storage* dan untuk menjalankan model menggunakan *Google Cloud Functions* agar aplikasi dapat melakukan *post request*.

Penelitian ini menghasilkan aplikasi yang dapat memindai dan mengirim wajah pengguna untuk dikirim dan menerima output dari *cloud functions*. *Output* tersebut berisi hasil prediksi model ML yang memprediksi apakah gambar yang dikirim dari Aplikasi adalah citra wajah berbentuk *heart*, *oblong*, *oval*, *round* atau *square*. Kemudian aplikasi akan menampilkan wajah pengguna beserta bingkai kacamata yang disajikan dengan *Augmented Reality* berdasarkan rekomendasi. Bentuk bingkai kacamata yang akan digunakan dalam aplikasi rekomendasi ini dibatasi 10 bingkai: *aviator*, *browline*, *butterfly*, *cat eye*, *geometry*, *oval*, *rectangle*, *round*, *sport*, dan *wayfarer*.

2.3. Metode *Waterfall*

Seiring dengan pertumbuhan sistem perangkat yang semakin besar, berbagai kendala mulai muncul. Perkembangan dalam pengembangan perangkat lunak, yang dikenal sebagai *Software Development Life Cycle* (SDLC) atau *System Development Life Cycle*, dimulai sejak tahun 1960-an. Inisialnya, SDLC digunakan untuk mengembangkan sistem skala besar yang memiliki fungsi kompleks untuk perusahaan-perusahaan besar pada masa itu. Sistem-sistem yang dikembangkan dalam konteks ini bertujuan untuk mengelola informasi mengenai kegiatan dan rutinitas yang dilakukan oleh perusahaan-perusahaan, yang seringkali melibatkan data dalam jumlah yang besar.

Software Development Life Cycle (SDLC) adalah suatu proses yang digunakan untuk mengembangkan atau mengubah sistem perangkat lunak. Proses ini melibatkan penggunaan model-model dan metodologi yang telah terbukti efektif berdasarkan praktik terbaik yang telah ada sebelumnya. Salah satu model yang digunakan dalam SDLC adalah Model *Waterfall*, yang juga dikenal sebagai model sekuensial linier atau alur hidup klasik (Ningsih & Nurfauziah, 2023). Model *Waterfall* mengadopsi pendekatan sekuensial dalam pengembangan perangkat lunak, dimulai dari tahap analisis, desain, pengodean, pengujian, hingga tahap pendukung seperti terlihat pada Gambar 3.



Gambar 3. Tahapan model SDLC *waterfall* (Rosa & Salahuddin, 2019:29)

3. PEMBAHASAN

3.1. Kumpulan Data Citra Bentuk Wajah

Untuk melatih *Machine Learning* dibutuhkan kumpulan data atau dataset yang akurat dan jumlah yang cukup agar dapat mencapai akurasi yang baik. Dataset yang digunakan dalam makalah ini adalah Dataset Bentuk Wajah yang tersedia secara gratis (Lama, 2019). Kumpulan data ini terdiri dari total 5000 gambar selebriti wanita yang dikategorikan ke dalam bentuk wajah *aviator, browline, butterfly, cat eye, geometry, oval, rectangle, round, sport, dan wayfarer*.

3.2. *Preprocessing*

Dibutuhkan kumpulan data atau *dataset* untuk melatih *machine learning* dalam menghasilkan klasifikasi dengan tingkat akurasi yang baik sehingga membutuhkan jumlah *dataset* yang cukup banyak. Dalam penelitian ini digunakan 5000 *dataset* dari Kaggle "*Face Shape dataset*" dengan lima kategori bentuk wajah yaitu hati, oblong, oval, bulat dan persegi. Masing-masing bentuk wajah berjumlah 1000 *dataset* (Lama, 2019).

Tahap *preprocessing* ini dilakukan bertujuan untuk menghasilkan citra yang lebih baik untuk diolah ke tahap selanjutnya. Tahap ini melakukan augmentasi data pada citra bentuk wajah. Augmentasi data pada pelatihan termasuk perubahan warna dan *flipping horizontal*, yang dapat membantu meningkatkan keragaman data pelatihan. Lalu memastikan bahwa data gambar telah diubah ukurannya menjadi seragam agar siap digunakan dalam pelatihan model.

```
TrainDatagen = ImageDataGenerator(  
    preprocessing_function= preprocess_input,  
    horizontal_flip = True  
)  
  
ValDatagen = ImageDataGenerator(  
    preprocessing_function= preprocess_input  
)  
  
TestDatagen = ImageDataGenerator(  
    preprocessing_function= preprocess_input  
)  
  
train_data = TrainDatagen.flow_from_directory(  
    train_images,  
    target_size = (299,299),  
    batch_size = 16,  
    class_mode = 'categorical'  
)  
  
val_data = ValDatagen.flow_from_directory(  
    val_images,  
    target_size = (299,299),  
    batch_size = 16,  
    class_mode = 'categorical'  
)  
  
test_data = TestDatagen.flow_from_directory(  
    test_images,
```

```
target_size = (299,299),
batch_size = 16,
class_mode = 'categorical'
)
```

Dataset yang digunakan dalam penelitian ini terbagi menjadi tiga bagian utama, yaitu data pelatihan, data validasi, dan data pengujian. Data pelatihan menggunakan 800 data untuk setiap bentuk wajah, data validasi dan data pengujian masing-masing menggunakan 100 data dari setiap bentuk wajah. Data pelatihan berisi gambar-gambar berbagai bentuk wajah manusia yang telah disiapkan dalam folder "*training_set*." Data validasi dan data pengujian masing-masing berada dalam folder "*ValidationSet*" dan "*TestingSet*." Setiap bagian dataset telah diproses dengan menggunakan alat bantu ImageDataGenerator untuk melakukan augmentasi data dan *preprocessing* yang diperlukan sebelum dimasukkan ke dalam model. Data ini kemudian dipisahkan menjadi *batch-batch* dengan ukuran 16 gambar per batch untuk proses pelatihan dan pengujian model. Selain itu, dataset ini diberikan label kelas berdasarkan struktur folder yang ada dalam dataset tersebut. Semua langkah persiapan data ini penting untuk memastikan kualitas data yang baik sebelum melanjutkan ke tahap pelatihan dan pengujian model klasifikasi bentuk wajah dengan menggunakan arsitektur *Inception V3*.

3.3. Perancangan Model

Dalam penelitian ini dilakukan telaah materi untuk model *Machine Learning* yang melakukan klasifikasi dengan bentuk wajah dari 2 sumber penelitian sebelumnya pengklasifikasian bentuk wajah berdasarkan analisis diskriminan Linear Discriminant *Analysis*, *Support Vector Machines*, *Multilayer Perceptron*, dan *K-Nearest Neighbors* (Emmanuel & Tio, 2019) dan klasifikasi bentuk wajah memiliki akurasi yang baik jika menggunakan *Inception V3* (Mehta & Mahmoud, 2022). Setelah menelaah materi tersebut dilanjutkan dengan melakukan analisis terhadapnya. Dari hasil kajian literatur ini, diperoleh pemahaman mendalam tentang landasan teori serta metode yang akan digunakan dalam penelitian ini. Tabel 1 dan Tabel 2 menjadi acuan dalam penelitian ini.

Tabel 1. Akurasi keseluruhan dari *500 Training Size* (Emmanuel & Tio, 2019)

Training Size	500
LDA	61.6%
SVM-LIN	55.2%
SVM-RBF	50.6%
MLP	54.0%
KNN	64.6%
IV3	97.8%

Tabel 2. Perbandingan Akurasi dari Metode ML dengan *dataset* (Mehta & Mahmoud, 2022)

Metode	Akurasi
Inception V3	84.4%
Region Similarity	80%

3D face data and SVM	73.68%
Active Appearance Model (AAM), segmentation and SVM	72%
Hybrid approach VGG and SVM	70.3%

Pada penelitian sebelumnya menyatakan untuk melakukan klasifikasi menggunakan Model *Inception v3* memiliki akurasi keseluruhan 84,4 % dan 97,8%, mengungguli pengklasifikasian berdasarkan analisis *Linear Discriminant Analysis*, *Support Vector Machines*, *Multilayer Perceptron*, dan *K-Nearest Neighbors*. Dari hasil tersebut maka pada penelitian ini *Convolutional Neural Network* (CNN) dengan arsitektur *Inception V3* digunakan sebagai *transfer learning*, model tersebut sudah dilatih terhadap suatu *dataset* untuk menyelesaikan masalah yang serupa dengan cara mengubah parameternya sehingga dapat sesuai dengan dataset baru untuk melakukan 5 klasifikasi bentuk wajah.

```
def build_model():
    base_model = InceptionV3(include_top=False, weights='imagenet', input_shape=(299, 299,
3))
    x = base_model.output
    x = GlobalAveragePooling2D()(x)
    x = Dense(1024, activation='relu')(x)
    x = Dropout(0.7)(x)
    x = Dense(512, activation='relu')(x)
    x = Dropout(0.5)(x)
    predictions = Dense(5, activation='softmax')(x)

    model = Model(inputs=base_model.input, outputs=predictions)
    return model
```

Model *Inception V3* menggunakan bobot yang berasal dari *dataset* ImageNet. Model dimodifikasi dengan menambahkan beberapa lapisan: *GlobalAveragePooling2D*, *Dense Layer* menggunakan aktivasi ReLU dengan 1024 unit, lapisan ini bertindak sebagai lapisan tersembunyi yang memungkinkan model untuk mempelajari pola-pola yang lebih kompleks dalam fitur-fitur citra yang telah dihasilkan. Aktivasi ReLU digunakan untuk memperkenalkan non-linearitas ke dalam model, yang memungkinkan model untuk memodelkan hubungan yang lebih kompleks dalam data. *Dropout* digunakan untuk mencegah *overfitting* dengan tingkat regularisasi sebesar 0,7.

Kemudian, ada *Dense Layer* lainnya dengan 512 unit dan aktivasi ReLU, yang berfungsi untuk melakukan pembelajaran lebih lanjut terhadap fitur-fitur yang telah diabstraksi. Terakhir, lapisan *Dense* output memiliki 5 unit (sesuai dengan jumlah kelas yang akan dihasilkan) dan aktivasi softmax. Lapisan ini menghasilkan probabilitas distribusi kelas untuk setiap citra, yang akan digunakan untuk klasifikasi akhir.

3.4. Pelatihan Model

Model yang telah dibangun akan dilatih untuk mempelajari fitur-fitur yang didapatkan dari masing-masing input kelas.

```
history = model.fit(
    train_data,
    validation_data = test_data,
    epochs = 50,
```

```
callbacks=[es,chkpt])
```

Proses pelatihan dimulai dengan memanggil metode `fit` pada objek model yang telah dibangun sebelumnya. Pada tahap ini, diberikan dua set data: `train_data` yang digunakan sebagai data pelatihan dan `val_data` yang digunakan sebagai data validasi. Data pelatihan berisi informasi mengenai citra wajah yang telah dikumpulkan dan diolah sebelumnya. Sedangkan data validasi digunakan untuk menguji performa model di setiap epoch selama proses pelatihan.

Selanjutnya, parameter `epochs` diberikan dengan nilai 50. Ini menentukan berapa kali model akan melihat seluruh data pelatihan dalam proses pelatihan. Pada setiap epoch, model akan melakukan perhitungan dan penyesuaian berdasarkan data pelatihan, dan kemudian mengukur performanya pada data validasi.

Pelatihan model menggunakan fungsi `callback`. `Callbacks` yang diberikan dalam parameter `callbacks` adalah `es` dan `chkpt`. `Callbacks` adalah objek yang memungkinkan kita untuk mengendalikan perilaku pelatihan. `es` adalah objek `EarlyStopping` yang akan menghentikan pelatihan jika tidak ada peningkatan yang signifikan pada performa model di atas data validasi setelah beberapa epoch. `chkpt` adalah objek `ModelCheckpoint` yang akan menyimpan model terbaik yang telah ditemukan selama pelatihan.

3.5. Pengujian Model

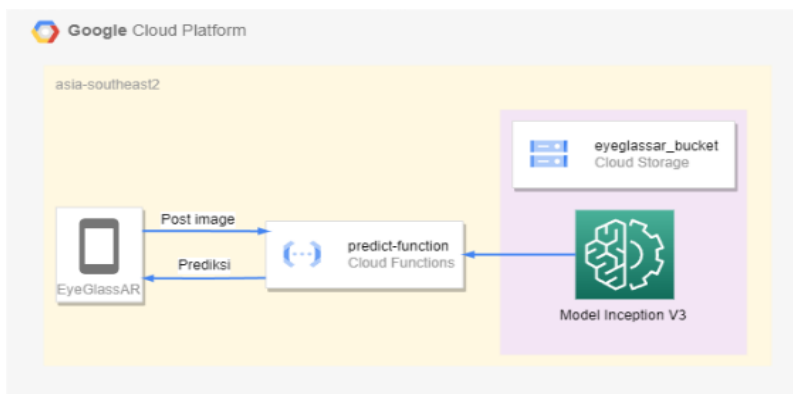
Pengujian model adalah tahap yang bertujuan untuk mengukur performa dan keandalan model terhadap data yang belum pernah dilihat sebelumnya. Proses pengujian ini penting untuk memastikan bahwa model tidak hanya mampu mempelajari data pelatihan dengan baik tetapi juga mampu menggeneralisasi dengan baik pada data yang tidak digunakan selama pelatihan.

Data yang digunakan dalam uji coba adalah data `testing`. Hasil uji coba terhadap data `testing` berupa nilai akurasi yang dihitung secara otomatis dengan kode berikut:

```
score = model.evaluate(test_data, verbose=1)
```

3.6. Konfigurasi di Layanan Cloud

Model yang telah dilatih digunakan untuk memprediksi bentuk wajah berdasarkan gambar wajah pengguna yang diambil dari aplikasi. Aplikasi Unity 3D tidak dapat menjalankan kode python, oleh karena itu digunakan layanan `cloud` agar aplikasi dapat mengirim `post request` gambar wajah pengguna ke model ML yang di `deploy` di `cloud`, dapat dilihat pada Gambar 3. Penyedia layanan `cloud` yang digunakan adalah `Google Cloud Platform`.













Gambar 4. Konfigurasi Layanan Cloud untuk Model Inception V3

Model disimpan menggunakan layanan *Google Cloud Storage* dan akan dijalankan menggunakan layanan *Google Cloud Functions*. Aplikasi melakukan *post request* ke *endpoint* Cloud Function, kemudian *Cloud Function* mengambil model dari *Cloud Storage* dan melakukan klasifikasi bentuk wajah. Prediksi dari klasifikasi bentuk wajah akan dikirim kembali ke aplikasi untuk acuan sebagai rekomendasi bentuk bingkai kacamata.

3.7. Penentuan Bentuk Bingkai Kacamata

Bentuk bingkai kacamata yang digunakan ada 10: *Aviator*, *Browline*, *Butterfly*, *Cat Eye*, *Geometric*, *Oval*, *Rectangle*, *Round*, *Sport*, dan *Wayfarer*. Rekomendasi bentuk bingkai mata berdasarkan wajah dengan pemetaan berikut:

Tabel 3. Kecocokan Bentuk Bingkai Kacamata

Bentuk Kacamata	Nama	Heart	Oblong	Oval	Round	Square
	Aviator	1	0	0	1	0
	Browline	0	1	1	0	1
	Butterfly	1	1	0	0	1
	Cat Eye	1	1	0	1	0
	Geometric	0	0	0	1	0
	Oval	1	1	1	0	1
	Rectangle	0	0	1	1	0
	Round	1	1	1	0	1
	Sport	0	0	1	1	0
	Wayfarer	1	0	1	0	0

3.8. Perbandingan Arsitektur VGG-16 dan *Inception V3*

Selain menggunakan *Inception V3*, dalam penelitian ini juga membandingkan algoritma CNN dengan arsitektur berbeda yaitu VGG-16. VGG-16 (Visual Geometry Group 16) adalah salah satu arsitektur *Convolutional Neural Network* (CNN) yang terkenal dalam dunia *computer vision*.

Arsitektur ini dikembangkan oleh grup penelitian *Visual Geometry Group* di University of Oxford. VGG-16 dikenal karena memiliki kedalaman yang relatif besar dan menjadi salah satu model yang sangat efektif dalam memproses citra. Tampak pada Tabel 4 adalah hasil perbandingannya:

Tabel 4. Perbandingan Akurasi Algoritma CNN Arsitektur VGG-16 dan Inception V3

	Arsitektur VGG-16	Arsitektur Inception V3
Hasil	loss: 0.3397	loss: 0.1732
Percobaan	accuracy: 0.8717	accuracy: 0.9435
Pelatihan	val_loss: 1.3284	val_loss: 1.0839
	val_accuracy:0.6079	val_accuracy:0.7460

4. HASIL DAN DISKUSI

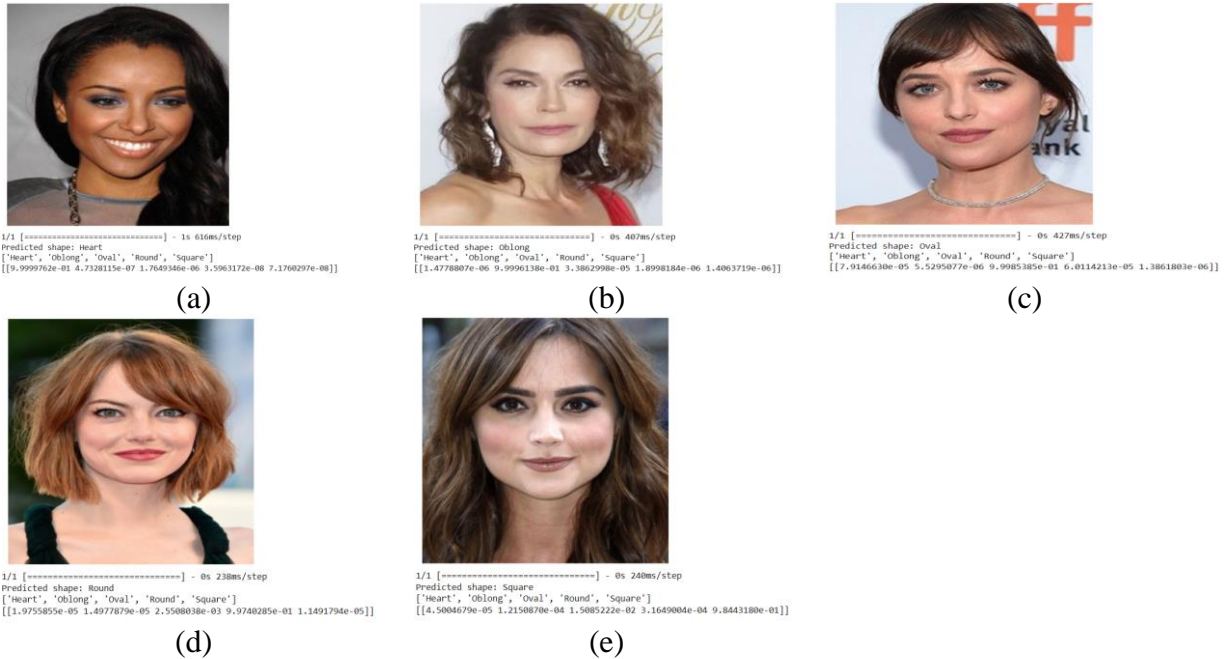
4.1. HASIL PENGUJIAN *TRAINING SET* DAN *VALIDATION SET*

Pengujian dilakukan pada *training set* dan *validation set* untuk menghasilkan model klasifikasi bentuk wajah yang optimal. Gambar 5 menampilkan hasil pengujian dengan jumlah *epoch* sebanyak 50, pada *epoch* ke-13 dihasilkan tingkat akurasi *training set* sebesar 0,9603 atau 96% dan tingkat akurasi *validation set* sebesar 0,7860 atau 78,6%. Setelah dilakukan pengujian sampai *epoch* ke-50, didapatkan hasil maksimum tingkat akurasi pada *epoch* ke-13. Sehingga dalam penelitian ini dihasilkan model optimal dengan jumlah *epoch* sebanyak 13 pada arsitektur *Inception V3*.

```
Epoch 1/50
250/250 [=====] - 93s 285ms/step - loss: 1.5824 - accuracy: 0.2965 - val_loss: 1.2302 - val_accuracy: 0.4800
Epoch 2/50
250/250 [=====] - 69s 276ms/step - loss: 1.1150 - accuracy: 0.5583 - val_loss: 1.2818 - val_accuracy: 0.5490
Epoch 3/50
250/250 [=====] - 70s 278ms/step - loss: 0.8028 - accuracy: 0.7072 - val_loss: 0.8195 - val_accuracy: 0.6920
Epoch 4/50
250/250 [=====] - 71s 283ms/step - loss: 0.5819 - accuracy: 0.7945 - val_loss: 0.7318 - val_accuracy: 0.7330
Epoch 5/50
250/250 [=====] - 70s 279ms/step - loss: 0.4311 - accuracy: 0.8545 - val_loss: 0.7986 - val_accuracy: 0.7460
Epoch 6/50
250/250 [=====] - 71s 284ms/step - loss: 0.3634 - accuracy: 0.8773 - val_loss: 1.1349 - val_accuracy: 0.7010
Epoch 7/50
250/250 [=====] - 73s 292ms/step - loss: 0.2805 - accuracy: 0.9137 - val_loss: 0.7382 - val_accuracy: 0.7830
Epoch 8/50
250/250 [=====] - 78s 312ms/step - loss: 0.2394 - accuracy: 0.9222 - val_loss: 0.6271 - val_accuracy: 0.8170
Epoch 9/50
250/250 [=====] - 73s 292ms/step - loss: 0.1953 - accuracy: 0.9398 - val_loss: 0.7735 - val_accuracy: 0.7700
Epoch 10/50
250/250 [=====] - 71s 283ms/step - loss: 0.1336 - accuracy: 0.9563 - val_loss: 0.8030 - val_accuracy: 0.7810
Epoch 11/50
250/250 [=====] - 71s 284ms/step - loss: 0.1728 - accuracy: 0.9460 - val_loss: 0.7356 - val_accuracy: 0.7840
Epoch 12/50
250/250 [=====] - 70s 279ms/step - loss: 0.1198 - accuracy: 0.9643 - val_loss: 0.9649 - val_accuracy: 0.7620
Epoch 13/50
250/250 [=====] - 70s 281ms/step - loss: 0.1276 - accuracy: 0.9603 - val_loss: 0.8483 - val_accuracy: 0.7860
```

Gambar 5. Akurasi *Training set* dan *Validation set*

Model optimal dengan *epoch* sama dengan 13, diuji menggunakan citra gambar yang terdapat pada *validation set*, terlihat pada Gambar 6 hasil dari pengujian menghasilkan tingkat prediksi bentuk wajah *Heart* dengan akurasi sebesar 99.9% (a), tingkat prediksi bentuk wajah *Oblong* sebesar 99.9% (b), tingkat prediksi bentuk wajah *Oval* sebesar 99.8% (c), tingkat prediksi bentuk wajah *Round* sebesar 99.7% (d) dan tingkat prediksi bentuk wajah *Square* sebesar 98.4% (e).



Gambar 6. Hasil Pengujian *Validation set*

4.2. HASIL PENGUJIAN *TESTING SET*

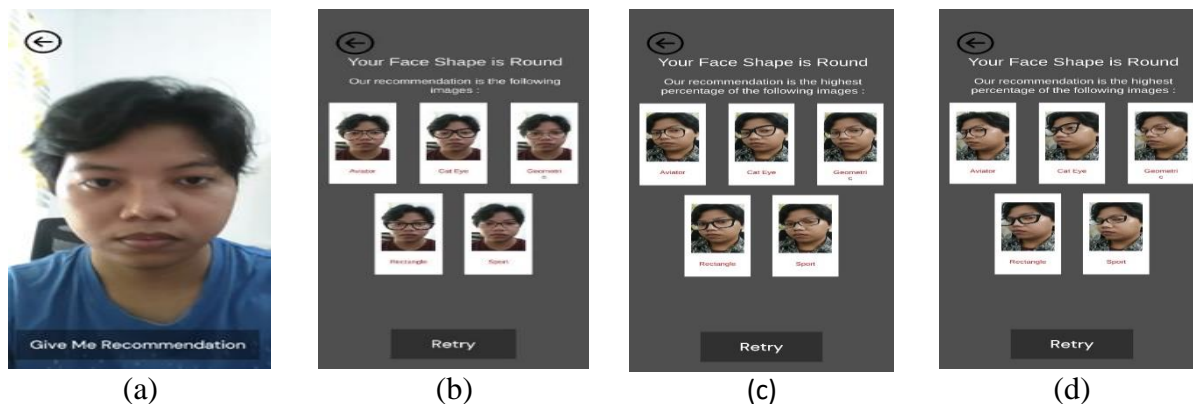
Pengujian model pada data test adalah langkah penting dalam mengevaluasi kinerja model klasifikasi. Data test adalah dataset yang tidak pernah dilihat oleh model selama proses pelatihan dan validasi. Tujuan utama dari pengujian pada data test adalah untuk mengukur sejauh mana model mampu menggeneralisasi dan melakukan prediksi yang baik pada data yang belum pernah dilihat sebelumnya.

63/63 [=====] - 9s 135ms/step - loss: 0.8483 - accuracy: 0.7860
Loss pada data pengujian: 0.8482540845870972
Akurasi pada data pengujian: 0.7860000133514404

Gambar 7. Akurasi *Testing set*

Hasil pengujian pada data test menunjukkan bahwa model yang telah dilatih memiliki tingkat akurasi sekitar 78.6% dan nilai loss sebesar 0.8483. Ini berarti model mampu melakukan prediksi dengan tingkat keakuratan yang baik pada data test yang belum pernah dilihat sebelumnya. Akurasi sebesar 78.6% mengindikasikan bahwa sekitar 78.6% dari data test telah diprediksi dengan benar oleh model.

Berikut adalah tampilan menu rekomendasi dari pengembangan Aplikasi, terlihat pada Gambar 8(a) dan 8(b).



Gambar 8. (a) Menu Rekomendasi, (b) Visualisasi bingkaiacamata yang direkomendasikan

Tombol “*Give Me Recommendation*” pada Gambar 8a ketika disentuh atau diklik aplikasi akan mengklasifikasikan bentuk wajah lalu akan menampilkan objek 3 Dimensi *Augmented Reality* Kacamata yang direkomendasikan sebanyak 5 bentuk frame kacamata, sehingga pengguna sudah mendapatkan visualisasi citra gambar wajah yang telah menggunakan bingkai kacamata seperti pada Gambar 8b. Dengan hasil ini maka dapat merekomendasikan beberapa frame kacamata bagi seseorang tanpa mencoba secara fisik untuk melihat kecocokan bentuk frame kacamata tersebut menurut bentuk wajahnya dari berbagai sudut kemiringan dapat dilihat pada Gambar 8(c) dan 8(d), dengan sudut kemiringan 30 dan 60 derajat.

5. KESIMPULAN

Dari hasil ujicoba menyatakan bahwa arsitektur *Inception V3* pada CNN dapat diimplementasikan pada *Augmented Reality* untuk mengklasifikasikan bentuk frame kacamata berdasarkan bentuk wajah dengan memiliki tingkat akurasi 78,6%. Hal ini dimungkinkan masyarakat dapat dengan mudah mengetahui bentuk frame kacamata yang sesuai dengan bentuk wajahnya tanpa harus membeli terlebih dahulu melalui perangkat *smartphone*.

Adapun saran bagi pengembangan penelitian ini dapat dilakukan dengan menambahkan fitur informasi rekomendasi bentuk *frame* kacamata yang trend berdasarkan bentuk wajah seseorang. Sehingga seseorang dapat menentukan pilihan frame kacamata sesuai dengan bentuk wajahnya

DAFTAR PUSTAKA

- Brooks, C. W., & Borish, I. M. (2007). *System For Ophthalmic Dispensing Third Edition*. Philadelphia: Elsevier Butterworth-Heinemann.
- Emmanuel, A., & Tio, D. (2019). Face shape classification using Inception v3. https://www.researchgate.net/publication/337386649_Face_shape_classification_using_Inception_v3.
- Firmantara, M. R., Mudakir, I., & Nuriman. (2023). Augmented Reality Research Trends in Indonesia: A Systematic Literature. *Journal of Science Education Research*, 71-81.

- Graham, D. L., & Ritchie, K. L. (2019). Making a Spectacle of Yourself: The Effect of Glasses and Sunglasses on Face Perception. *Sage*, 1-10.
- Hapsari, A. A., Gernowo, R., & Widodo, C. E. (2020). Penggunaan Algoritma CART untuk Pemilihan Bingkai Kacamata dengan Penerapan Model Morfologi Indeks Wajah untuk Identifikasi Bentuk Wajah. *Jurnal Sistem Informasi Bisnis*, 1-8.
- Hidayat, T., Astuti, I. A., Yaqin, A., Tjilen, A. P., & Arifianto, T. (2023). Grouping of Image Patterns Using Inception V3. *INTERNATIONAL JOURNAL*, 2336-2343.
- Hidayatillah, M. T., Mardiyantoro, N., & Hidayat, M. (2022). SISTEM IDENTIFIKASI BENTUK WAJAH UNTUK PEMILIHAN FRAME KACAMATA MENGGUNAKAN METODE TRANSFER LEARNING. *Jurnal Ilmiah Informatika dan Komputer Universitas Al-Qur'an*, 70-78.
- Ismawati, N. A., & Ramadhanti, S. (2022). Penerapan Artificial Intelligence Dalam Mendukung Pembelajaran Di Era Digita. *Seminar Nasional Prosiding Amal Insani Foundation* (pp. 158-166). Amal Insani Foundation.
- Krzyszowski, T., Dziadek, B., Franca, C., Martins, F., Gouveia, E. R., & Przednowek, K. (2023, March 2023 Saturday). System for Estimation of Human Anthropometric Parameters Based on Data from Kinect v2 Depth Camera. *Sensors - MDPI*, pp. 1-12.
- Lama, N. (2019). *Face Shape Dataset*. Retrieved from <https://www.kaggle.com/https://www.kaggle.com/datasets/niten19/face-shape-dataset>
- Mehta, A., & Mahmoud, T. (2022). Human Face Shape Classification with Machine Learning. https://www.researchgate.net/publication/362903132_Human_Face_Shape_Classification_with_Machine_Learning. doi:<https://doi.org/10.13140/RG.2.2.28282.57282>
- Ningsih, W., & Nurfauziah, H. (2023). Perbandingan Model Waterfall dan Metode Prototype Untuk Pengembangan Aplikasi Pada Sistem Informasi. *Jurnal Ilmiah Metadata*, 84-95.
- Pratiwi, V. R., & Pardede, J. (2023). Image Captioning Menggunakan Metode Inception-V3 dan Transformer. *Prosiding Diseminasi Fakultas Teknologi Industri - Institut Teknologi Nasional* (pp. 1-14). Bandung: Itenas.
- Rizal, R. A., Girsang, I. S., & Prasetyo, S. A. (2019). Klasifikasi Wajah Menggunakan Support Vector Machine (SVM). *Riset dan E-Jurnal Manajemen Informatika Komputer*, 1-4.
- Young, S., Natalia, F., Sudirman, S., & Seong Ko, C. (2019). EYEGASSES FRAME SELECTION BASED ON OVAL FACE SHAPE USING CONVOLUTIONAL NEURAL NETWORK. *ICIC International*, 707-715.
- Zhao, Z., Zhou, L., & Zhang, T. (2020). Intelligent Recommendation System for Eyeglass Design. *Springer Nature Switzerland AG*, 402-411. doi:https://doi.org/10.1007/978-3-030-20441-9_42