

KLASIFIKASI DETEKSI PENGGUNAAN MASKER MENGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK

Nalda Kresimo Negoro*¹⁾, Ema Utami²⁾, Ainul Yaqin³⁾

1. Universitas Amikom Yogyakarta, Indonesia
2. Universitas Amikom Yogyakarta, Indonesia
3. Universitas Amikom Yogyakarta, Indonesia

Article Info

Kata Kunci: Kecerdasan Buatan, Komputer Visi, Convolutional Neural Network, Era Industri 5.0, Face Mask Detection, Pembelajaran Mesin, Deteksi Objek

Keywords: Artificial Intelligence, CNN, Computer Vision, Convolutional Neural Network, Era Society 5.0, Face Mask Detection, Machine Learning, Object Detection

Article history:

Received 28 January 2023
Revised 4 February 2023
Accepted 2 March 2023
Available online 1 June 2023

DOI :

<https://doi.org/10.29100/jipi.v8i2.3748>

* Corresponding author.

Corresponding Author

E-mail address:

naldakresimonegoro@gmail.com

ABSTRAK

Berhubungan dengan era revolusi industri 5.0 kita perlu bersyukur semua pekerjaan menjadi dimudahkan dengan terdigitalisasi. Berbagai pekerjaan dapat diselesaikan jauh lebih mudah, cepat dan secara otomatis. Konsep era industri 5.0 memiliki fokus pendayagunaan aspek dari manusia, data dan teknologi berbasis modern. Manusia dan sistem saling terhubung dan mendapatkan hasil maksimal dengan bantuan AI. Konsep ini memberikan dampak positif untuk menghadapi perubahan besar pada transformasi digital. Perkembangan pesat dari transformasi digital saat ini ada pada pendeteksian objek menggunakan machine learning. Deteksi objek adalah teknik dari computer vision dalam pembacaan pengenalan objek pada gambar ataupun video. Pada penelitian ini diterapkan klasifikasi deteksi objek dengan algoritma Convolutional Neural Network (CNN) menggunakan arsitektur VGG16Net dengan mengklasifikasikan wajah bermasker dan tidak bermasker. Dataset yang digunakan untuk proses training diperoleh dari kaggle berjumlah 3.725 menggunakan masker, 3.828 tidak menggunakan masker dan dataset untuk proses testing menggunakan dataset personal berjumlah 16 dataset. Evaluasi jaringan pelatihan model menggunakan confusion matrix sedangkan tahap pengujian menggunakan SSD ResNet10. Hasil evaluasi dari rancangan implementasi pelatihan model didapatkan nilai akurasi 0,992, presisi 1.000, dan recall 0,984. Kemudian hasil pengujian testing mendapatkan nilai tertinggi dengan akurasi 100%.

ABSTRACT

In connection with "Industrial Revolution 5.0", we need to be grateful that all work has been made easy by being digitized. Various jobs can be completed much easier, faster, and automatically. The concept of "industrial era 5.0" has a focus on utilizing aspects of humans, data, and modern-based technology. Humans and systems are interconnected, and we get the most out of it with the help of AI. This concept has a positive impact on facing major changes in digital transformation. The rapid development of digital transformation today is in object detection using machine learning. Object detection is a technique from computer vision for reading objects in images or videos. In this study, object detection classification was applied using the Convolutional Neural Network (CNN) algorithm using the VGG16Net architecture by classifying masked and non-masked faces. The dataset for the training process came from Kaggle, with 3,725 using masks and 3,828 not using masks, and the dataset for the testing process came from personal datasets, with a total of 16 datasets. The evaluation of the model training network uses the confusion matrix, while the testing phase uses SSD ResNet 10. The results of the evaluation of the model training implementation design obtained an accuracy value of 0.992, a precision of 1,000, and a recall of 0.984. Then the test results get the highest value with 100% accuracy.

I. PENDAHULUAN

BERHUBUNGAN dengan masuknya revolusi era industri 5.0 kita perlu bersyukur membuat semua pekerjaan menjadi lebih dimudahkan dengan hadirnya kemudahan yang diberikan dengan serba terdigitalisasi. Berbagai hal yang dikerjakan dapat diselesaikan dengan jauh lebih mudah, lebih cepat dan dapat diselesaikan secara otomatis. Revolusi era industri 5.0 atau bisa disebut dengan *era society 5.0* telah di sampaikan oleh negara Jepang. Konsep dari era industri 5.0 memiliki fokus terhadap kombinasi antara penggunaan sumber daya dari aspek manusia, data serta kemajuan dari teknologi yang ada sampai saat ini dengan menggunakan teknologi berbasis modern seperti *AI*, *IoT* dan Robot [1]. Walaupun tidak memiliki perbedaan yang jauh dengan era industri terdahulu yaitu era industri 4.0, namun keduanya memiliki fokus yang berbeda. Secara sederhana pada era industri 4.0 berfokus pada pengumpulan informasi menggunakan jaringan dan di lakukan analisa, berbeda dengan fokus era industri 5.0 yaitu data yang didapatkan dari manusia akan dihubungkan dan diolah dengan sistem untuk mendapatkan hasil yang diinginkan dengan maksimal dibantu dengan bantuan dari *Artificial Intelligence* (AI). Hasil yang telah didapatkan kemudian dikembalikan kepada ruang fisik. Dengan hadirnya konsep berikut memberikan dampak positif untuk menghadapi perubahan pada transformasi digital dengan masuknya revolusi era industri 5.0 seiring kemunculan teknologi seperti *IoT*, *Big Data*, *Data Science*, *Computer Vision*, *Robotika*, *AI*, *Machine Learning*, *Deep Learning*, sampai pada pengolahan data dapat memberikan kemudahan dalam kehidupan manusia [2].

Perkembangan pesat dari transformasi digital saat ini ada pada pendeteksian objek menggunakan *machine learning*. Deteksi objek merupakan salah satu teknik *computer vision* untuk melakukan pembacaan atau pengenalan objek pada gambar ataupun video. Penerapan dari *computer vision* yang dapat digunakan pada berbagai bidang dapat memberikan kemudahan bagi manusia dalam membantu pekerjaan. *Computer vision* termasuk bagian dari *artificial intelligence* yang memprogram komputer agar dapat melihat dan melakukan pengenalan hingga deteksi objek yang ada disekitarnya layaknya cara kerja pada mata manusia dan mamalia [3]. Untuk dapat mengenali objek pada suatu gambar maupun video, sebuah sistem memerlukan dataset yang perlu dilatih terlebih dahulu terkait dengan kelas objek tersebut. Sistem akan mempelajari dataset tersebut untuk menemukan ciri khusus dari suatu objek tersebut yang dapat dijadikan pola oleh sistem dalam mengenali objek yang sama tetapi dikondisi yang berbeda. Proses dari pembelajaran sistem dengan dataset tersebut termasuk pada bidang keilmuan *machine learning* [4].

Terdapat banyak algoritma yang dapat digunakan dalam pembelajaran sistem terhadap objek citra seperti pada penelitian sebelumnya yaitu melakukan analisis komparasi klasifikasi citra pengguna masker dengan membandingkan kinerja dari metode CNN dibandingkan dengan kombinasi metode CNN dan *Support Vector Machine* (SVM). Pada penelitian berikut disebutkan arsitektur terbaik yang digunakan yaitu menggunakan salah satu dari tiga arsitektur yaitu MobileNet, ResNet50, dan VGG-16. Kesimpulan yang didapatkan pada penelitian berikut didapatkan nilai kinerja tertinggi yaitu dengan mengkombinasikan CNN dan ResNet50 didapatkan nilai kinerja tertinggi dengan *accuracy* 99,41% [5]. Maka dari itu, tidak begitu mengherankan jika algoritma CNN banyak digunakan pada proses klasifikasi citra. Selain itu penelitian kedua membahas identifikasi individu dalam penggunaan masker secara benar atau tidak benar serta memverifikasi identitas menggunakan layanan aplikasi mobile dan menggunakan metode CNN. Pada penelitian ini tidak menampilkan hasil setiap validasi akurasi pada 12 sampel wajah yang digunakan sebagai pelatihan model [6]. Penelitian ketiga yaitu memiliki tujuan yang sama melakukan identifikasi deteksi penggunaan masker tetapi menggunakan metode R-CNN dan model deteksi objek YOLOv3. Pada penelitian ini tidak ditunjukkan hasil tingkat akurasi yang didapatkan pada pelatihan model dan tidak menunjukkan berapa jumlah sampel wajah yang digunakan untuk pelatihan model [7]. Pada penelitian keempat yaitu pendeteksian citra wajah bermasker menggunakan CNN dan transfer learning dengan melakukan kombinasi analisa dari arsitektur VGG16, MobileNetV2, Xception, dan DenseNet201. Dari arsitektur VGG16 didapatkan akurasi 92,7% dan akurasi terbaik didapatkan oleh arsitektur Xception dengan nilai 0,988. Pada penelitian ini sayangnya hasil sampel yang diujikan tidak ditunjukkan hasil output menggunakan labelling pada objek yang terdeteksi [8].

Dikarenakan keingintahuan dari penulis untuk melakukan penelitian terkait klasifikasi dan pendeteksian penggunaan masker berbekal dengan membaca beberapa penelitian sebelumnya maka penulis ingin melakukan penelitian dengan metode CNN dan menggunakan arsitektur VGG16Net. Metode CNN dengan arsitektur VGG16Net tersebut akan diterapkan supaya dapat mengukur nilai kinerja *accuracy* tertinggi yang didapatkan dari metode tersebut. Selain itu penelitian ini juga akan mengukur kinerja algoritma dari nilai kinerja *precision* dan *recall*.

II. METODE PENELITIAN

2.1. Computer Vision

Komputer visi merupakan kemajuan teknologi yang mampu memproses, menganalisa, dan mendeteksi suatu citra kemudian merubahnya menjadi suatu informasi bagi sistem untuk mengambil suatu keputusan. Dengan kata lain teknologi *computer vision* terinspirasi dari fungsi mata manusia untuk mengenali suatu objek untuk dijadikan suatu informasi yang dapat diolah pada sistem komputer [9]. Pada penerapannya tujuan utama dari *computer vision* yaitu untuk dapat melihat dan melakukan analisa pada gambar digital secara otomatis kemudian memberikan informasi dari gambar tersebut. Dengan kata lain komputer akan menafsirkan dan melakukan analisa pada visual dengan simulasi seperti cara kerja manusia dalam melihat dan memahami lingkungan sekitar [10]. Dalam hal ini menerapkan model *machine learning* (ML) untuk melakukan identifikasi dan klasifikasi objek pada gambar dan video digital kemudian memungkinkan komputer untuk menganalisa terhadap apa yang dilihat.

2.2. Object detection

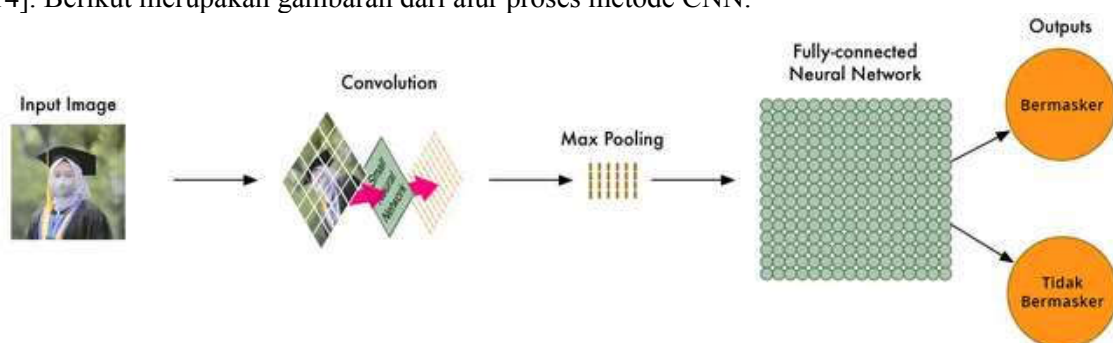
Object detection atau deteksi objek merupakan bagian dari pengenalan visual dalam *computer vision* yang sedang berkembang pesat pada beberapa tahun terakhir. Deteksi objek visual memiliki tujuan agar dapat menemukan objek dari kelas yang sudah ditargetkan tertentu dengan lokalisasi yang tepat dalam gambar yang diberikan dan menetapkan setiap instance objek label pada kelas yang sesuai. Maka, teknik ini dapat memungkinkan sistem untuk melakukan identifikasi serta menemukan objek pada gambar ataupun video. Identifikasi deteksi objek dapat digunakan dalam melakukan perhitungan jumlah objek dalam melihat dan menentukan keakuratannya dengan memberikan label secara akurat [11].

2.3. Deep Learning dan Machine Learning

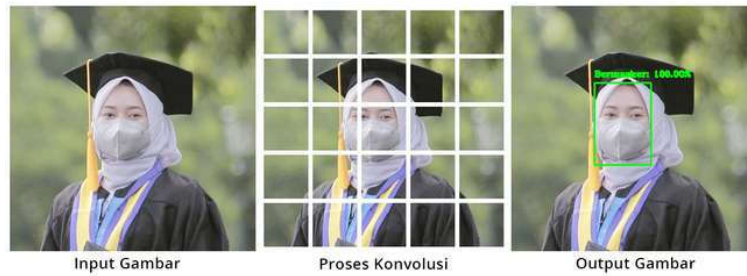
Penjelasan tentang apa itu *deep learning*, *deep learning* memiliki akar yang sama dengan *machine learning* (bagian dari *machine learning*). *Machine learning* secara umum mengacu pada kategori algoritma yang secara otomatis dapat menghasilkan model prediksi dengan mendeteksi pola dalam data. *Machine learning* telah berhasil diterapkan untuk melakukan tugas-tugas seperti klasifikasi. Apa yang membuat algoritma *deep learning* berbeda yaitu terletak pada cara mesin mempelajari fitur dari kumpulan data serta berapa banyak data yang digunakan untuk training model. Pembelajaran dapat terjadi di mana komputer secara otomatis menemukan pola dan kesamaan dalam data yang tidak berlabel. Dengan metode ini, digunakan sebagai alat eksplorasi untuk mendeteksi fitur dalam data, mengurangi dimensinya atau mengelompokkan data pada kelompok data yang serupa. Pengguna hanya perlu memberi tahu algoritma *deep learning* apakah dalam satu gambar ada objek hewan/manusia dalam gambar serta dengan memberikan contoh yang cukup dengan data, algoritma tersebut dapat mengetahui sendiri seperti apa bentuk objek tersebut [12].

2.4. Convolutional Neural Network (CNN)

CNN yaitu kombinasi dari pembelajaran mesin dan jaringan syaraf tiruan serta metode ini diperlukan pada penelitian klasifikasi gambar [13]. Metode CNN mampu memproses dengan sendirinya klasifikasi, ekstraksi objek dan pengenalan objek hingga bisa digunakan pada penggunaan gambar beresolusi tinggi yang memiliki model distribusi non parametrik. CNN akan melakukan pemrosesan lapisan konvolusional dengan penggabungan beberapa lapisan dalam prosesnya, metode ini dioperasikan secara paralel menggunakan beberapa elemen serta metode ini mencontoh proses dari sistem jaringan syaraf biologis. Metode CNN dapat mempresentasikan kedalaman 2D pada setiap neuron, maka dari itu tahap *processing* dari metode ini bisa diproses menggunakan penginputan gambar [14]. Berikut merupakan gambaran dari alur proses metode CNN.

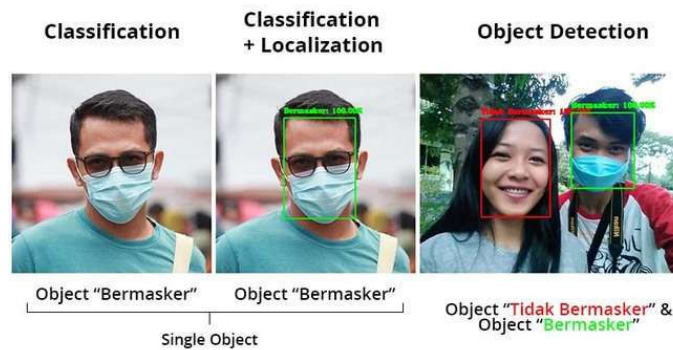


Gambar 1. Alur Proses Metode CNN



Gambar 2. Contoh Sampel Gambar Dari Alur Proses Metode CNN

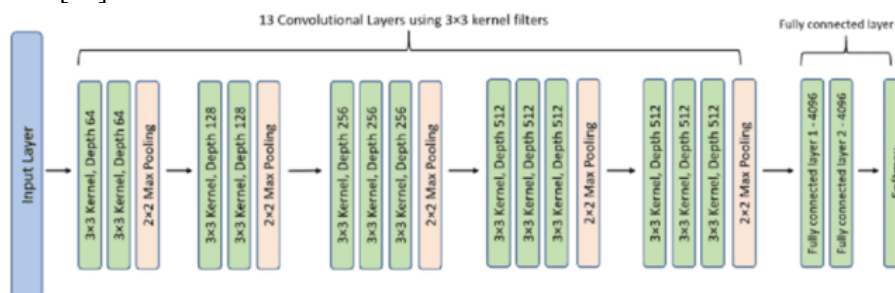
Sampai saat ini CNN sering digunakan dan diterapkan salah satunya pada klasifikasi dan deteksi gambar. Image classification merupakan teknik mengkategorikan suatu gambar kedalam suatu kategori tertentu sedangkan image localization merupakan bagian dari pengembangan dari teknik tersebut. Ketika output yang dapat dihasilkan terdapat letak dari objek kelas pada gambar tersebut bisa disebut dengan *bounding box*. *Object detection* merupakan klasifikasi dan lokalisasi pada beberapa objek pada suatu gambar [15].



Gambar 3. Contoh Deteksi Objek Pada Metode CNN

2.5. Arsitektur VGG16Net

Arsitektur ini dikembangkan oleh Alexnet dimana VGG16Net berfokus pada pemrosesan *feature extraction* pada convolusi layer hingga dapat memecah banyak gambar agar bisa diklasifikasikan. VGG16Net memiliki sekitar 16 bagian layer. Terdiri dari 13 lapisan layer konvolusi, 2 lapisan layer digunakan untuk layer *fully connected*, dan 1 layer untuk klasifikasi [16].



Gambar 4. Model Arsitektur VGG16Net

2.6. Jenis, Pendekatan, dan Sifat Penelitian

Pendekatan penelitian pada tahap ini bersifat kuantitatif dengan jenis penelitian eksperimental, pada penelitian ini dimana akan dilakukan skenario pengujian dalam klasifikasi deteksi penggunaan masker yang menggunakan metode algoritma CNN dengan arsitektur VGG16Net. Selanjutnya, skenario dari percobaan pengujian tersebut akan dievaluasi hasil untuk dapat menilai keakuratan dalam melakukan klasifikasi dan deteksi citra dari skenario yang dibuat agar dapat mengetahui fakta apa saja yang didapatkan dari hasil penelitian.

2.7. Metode Pengumpulan Data

Langkah berikut dilakukannya tahap mengumpulkan dataset dengan menggunakan dua jenis yaitu menggunakan dataset dari media internet serta menggunakan dataset pribadi. Hasil dari pengumpulan data didapatkan gambar dengan beberapa objek menggunakan masker dan beberapa objek tidak menggunakan masker. Dari dataset tersebut yang dikumpulkan dan diperoleh menampilkan banyak objek hingga diperlukannya *preprocessing* data yaitu dengan cara dilakukannya *cropping* pada objek gambar yang tidak diperlukan dan menggunakan objek gambar berupa wajah beberapa objek bermasker dan beberapa objek tidak bermasker. Selanjutnya, semua gambar tersebut

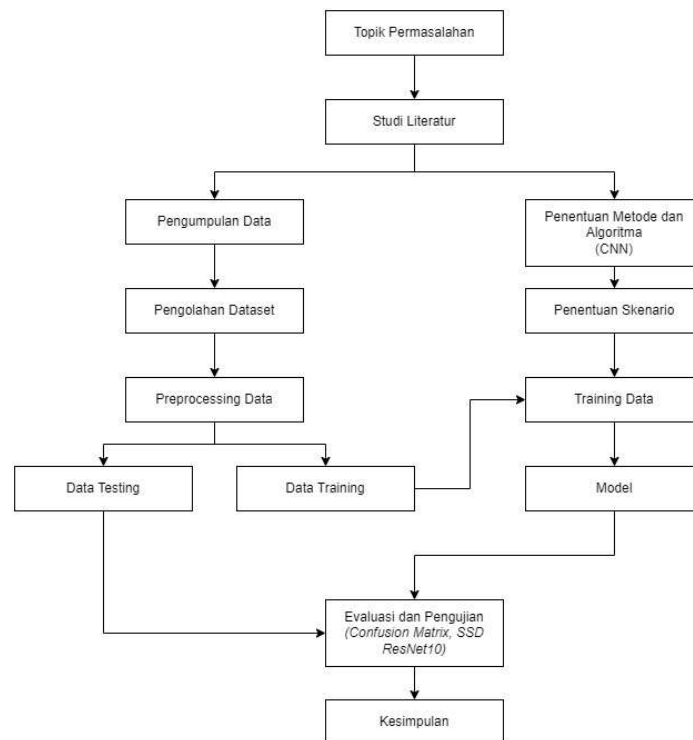
akan dipisah menjadi 2 folder yaitu folder pertama berisikan gambar orang-orang yang menggunakan masker dan folder kedua berisikan gambar orang-orang yang tidak menggunakan masker yang selanjutnya dapat digunakan sebagai dataset. Dataset yang didapatkan tersebut akan diolah hingga berfungsi dapat digunakan sebagai pembangunan model pembelajaran mesin.

2.8. Metode Analisis Data

Langkah ini bertujuan untuk memberikan pemahaman dari alur proses dan sistem kerja model yang hendak dibangun menggunakan metode yang sudah ditetapkan. Selain itu juga dapat memberikan pemahaman terhadap langkah-langkah pada proses pembangunan model menggunakan metode algoritma CNN dengan arsitektur VGG16Net. Dataset yang didapatkan dari pengumpulan data akan dilakukan preprocessing terlebih dahulu seperti melakukan partisi data dalam pelatihan dan pengujian dan membentuk *training image generator* sebagai *data augmentation* sebelum tahap ekstraksi fitur dan pembuatan tahap model. Dalam proses percobaan ini menggunakan bahasa pemrograman python serta dengan metode CNN dan model arsitektur VGG16Net.

2.9. Alur Penelitian

Bagian ini berisi diagram alur dari setiap proses dari penelitian berikut.



Gambar 5. Diagram Alur Penelitian

III. HASIL DAN PEMBAHASAN

Penelitian ini memberikan pembahasan tentang rancangan sistem secara menyeluruh, dengan mendapat data hasil dari implementasi dan pengujian tahap selanjutnya adalah menganalisa pada setiap proses tahap dari pengujian. Bagian ini akan menjelaskan proses pengumpulan dataset dari *face mask detection*, training dan hasil dari sistem *face mask detection*.

3.1. Dataset Mask Detection

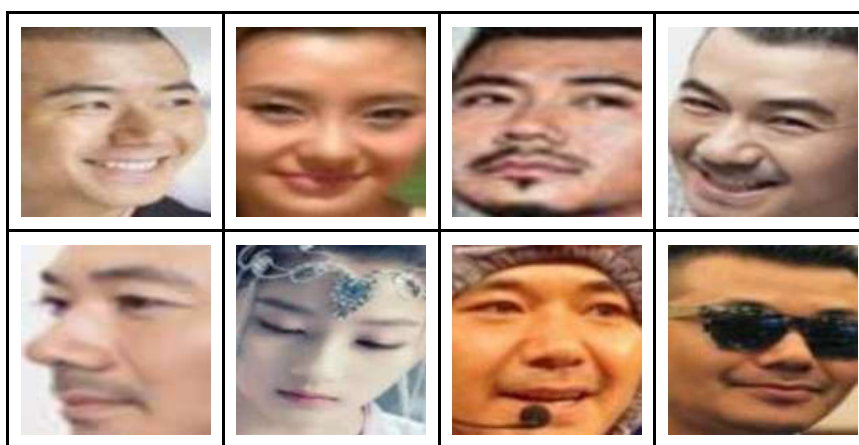
Penelitian berikut menggunakan dataset berupa data training dilakukan pemrosesan implementasi pada sistem *face mask detection* dan data testing sebagai data uji. Data training yang di dapatkan dari website kaggle dengan link sebagai berikut : www.kaggle.com/omkargurav/face-mask-dataset

Dataset training berupa file gambar yang didapatkan total 7.553 file yang terdiri dari 3.725 file gambar yang berisikan wajah orang-orang yang menggunakan masker serta 3.828 file gambar yang berisikan wajah orang-orang yang tidak menggunakan masker.

Tabel 1. Contoh Data Training Gambar Objek Bermasker



Tabel 2. Contoh Data Training Gambar Objek Tidak Bermasker



Selanjutnya pengambilan dataset untuk pengujian (data testing) yang di implementasikan ke sistem berjumlah 16 file yang terdiri dari 8 file gambar tidak memakai masker dan 8 file gambar yang memakai masker. File dataset bervariasi mulai dari gambar wajah laki-laki dan perempuan serta file gambar yang menggunakan kacamata dan tidak.

Tabel 3. Data Testing Gambar Objek Bermasker



Tabel 4. Data testing gambar objek tidak bermasker





3.2. Tahap Pembuatan Model

Tahap pembuatan model dengan membentuk bagian head dari model kemudian ditempatkan pada base model dan dilakukan perulangan pada seluruh base model. Kemudian selanjutnya dikompilasikan.

```

# Membentuk bagian head dari model yang akan ditempatkan pada base model
headModel = baseModel.output
headModel = AveragePooling2D(pool_size=(7, 7))(headModel)
headModel = Flatten(name="flatten")(headModel)
headModel = Dense(128, activation="relu")(headModel)
headModel = Dropout(0.5)(headModel)
headModel = Dense(2, activation="softmax")(headModel)

# Menempatkan head model pada base model
model = Model(inputs=baseModel.input, outputs=headModel)

# Perulangan pada seluruh base model
for layer in baseModel.layers:
    layer.trainable = False

# Persiapan kompilasi model
print("Mengkompilasi model...")
opt = Adam(lr=INIT_LR, decay=INIT_LR / EPOCHS)
model.compile(loss="binary_crossentropy", optimizer=opt,
              metrics=["accuracy"])

model.summary()
    
```

Gambar 6. Source Code Tahap Pembuatan Model

```

Mengkompilasi model...
Model: "model"
    
```

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0

Gambar 7. Potongan Gambar Hasil Kompilasi Dari Pembuatan Model

3.3. Pelatihan Model

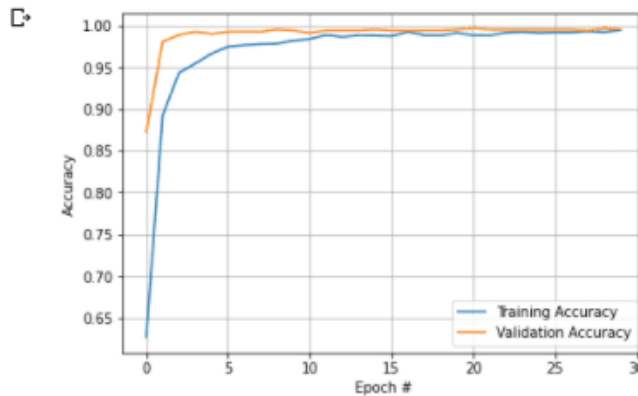
Tahap ini dataset yang dikumpulkan akan dilakukan training model dengan menggunakan metode CNN dan arsitektur VGG16Net. Data training yang terdiri dari 7.553 file gambar sampel dimana akan digunakan untuk pelatihan model.

```

# Pelatihan model
print("Training head model...")
H = model.fit(
    aug.flow(trainX, trainY, batch_size=BS),
    steps_per_epoch=len(trainX) // BS,
    validation_data=(testX, testY),
    validation_steps=len(testX) // BS,
    epochs=EPOCHS)
    
```

Gambar 8. Source Code Pelatihan Model

Arsitektur dari pelatihan model yang dikembangkan ini dilatih selama 30 epoch (iterations) serta batch size hingga 95. Batch size adalah seberapa banyak sampel data yang disebar pada jaringan syaraf [17]. Gambar dibawah ini merupakan grafik untuk menunjukkan hasil training accuracy dan validation accuracy.



time: 322 ms (started: 2022-01-02 10:24:13 +00:00)

Gambar 9. Hasil Training dan Validation Accuracy

3.4. Confusion Matrix

Dibawah ini gambar 8 merupakan source code untuk menampilkan prediksi label evaluasi menggunakan confusion matrix. Sedangkan gambar 9 merupakan tampilan hasil prediksi label evaluasi menggunakan confusion matrix. Label ini ditampilkan untuk memunculkan nilai prediksi dari pengukuran kinerja pelatihan model.

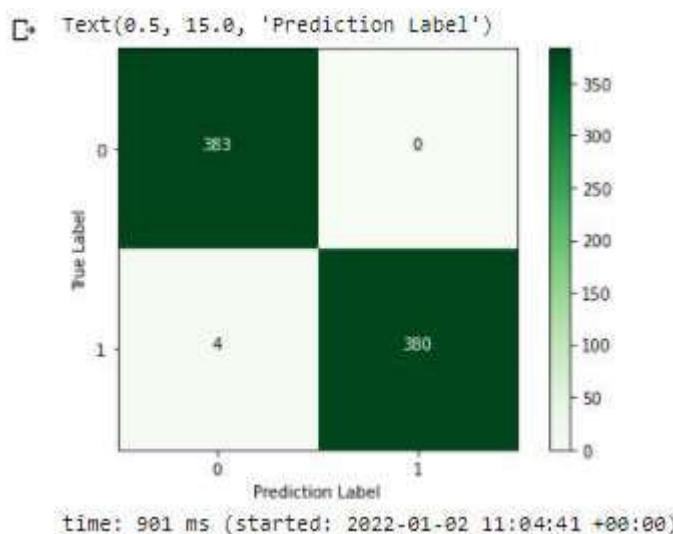
```

classes = [0, 1]
# Plot confusion matrix
plt.imshow(conf, interpolation='nearest', cmap=plt.cm.Greens)
plt.title("Confusion Matrix")
plt.colorbar()
tick_marks = np.arange(len(classes))
plt.xticks(tick_marks, classes)
plt.yticks(tick_marks, classes)

fmt = 'd'
thresh = conf.max() / 2.
for i, j in itertools.product(range(conf.shape[0]), range(conf.shape[1])):
    plt.text(j, i, format(conf[i, j], fmt),
            horizontalalignment="center",
            color="white" if conf[i, j] > thresh else "black")

plt.tight_layout()
plt.ylabel('True Label')
plt.xlabel('Prediction Label')
    
```

Gambar 10. Source Code Menampilkan Label Evaluasi Confusion Matrix



Gambar 11. Hasil Evaluasi Confusion Matrix

Confusion matrix umumnya dapat digunakan sebagai pengukuran kinerja dari proses pelatihan model terutama pada image classification. Pelatihan model klasifikasi deteksi objek dapat diukur dengan mengukur performa dari

output yang terdapat 2 kelas bahkan lebih. Pengukuran ini menggunakan label kombinasi berisikan 4 kolom yang memiliki perbedaan nilai pada setiap kolom dari segi nilai aktual maupun prediksi. Hasil output dari pengukuran kinerja ini memiliki 4 kombinasi. *True Positif (TP)*, *True Negatif (TN)*, *False Positif (FP)*, dan *False Negatif (FN)* [18]. Secara garis besar *confusion matrix* adalah tools analitik untuk dapat memprediksi dengan memberikan perbandingan nilai aktual dan nilai dari prediksi hal tersebut dilakukan sebagai evaluasi kinerja untuk pelatihan model. Pengukuran kinerja ini diantaranya dapat menghasilkan nilai akurasi, presisi, recall, dan F1. Berikut merupakan rumus perhitungan *confusion matrix*.

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

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

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

$$F1 = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (4)$$

```
results = {}  
  
# Akurasi  
metric = "Akurasi"  
results[metric] = (TP + TN) / (TP + TN + FP + FN)  
print(f"{metric} = {results[metric]: .3f}")  
  
# Recall  
metric = "Recall"  
results[metric] = TP / (TP + FN)  
print(f"{metric} = {results[metric]: .3f}")  
  
# Presisi  
metric = "Presisi"  
results[metric] = TP / (TP + FP)  
print(f"{metric} = {results[metric]: .3f}")  
  
# Nilai F1  
metric = "F1"  
results[metric] = 2 / (1 / results["Presisi"] + 1 / results["Recall"])  
print(f"{metric} = {results[metric]: .3f}")
```

Gambar 12. Implementasi Rumus Confusion Matrix

```
↳ Akurasi = 0.992  
Recall = 0.984  
Presisi = 1.000  
F1 = 0.992  
time: 21.7 ms (started: 2022-01-02 10:26:14 +00:00)
```

Gambar 13. Hasil Output Pengukuran Kinerja Menggunakan Confusion Matrix

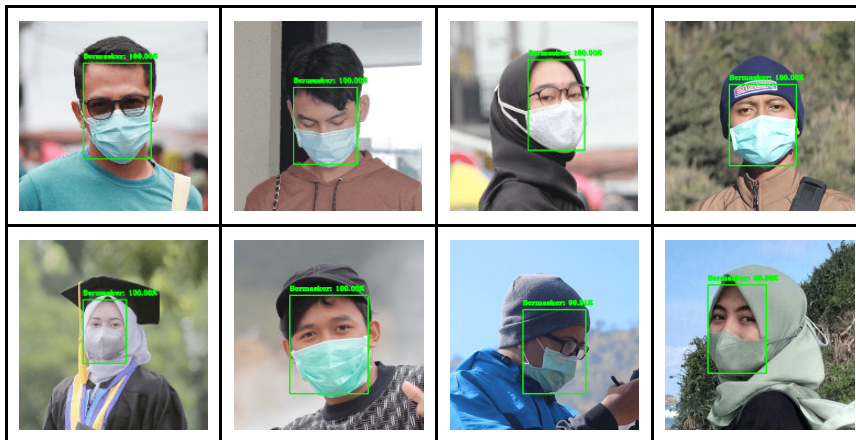
Pada penelitian terdahulu yaitu pendeteksian citra wajah bermasker menggunakan CNN dan transfer learning. Arsitektur VGG16 didapatkan akurasi 92,7% dan akurasi terbaik didapatkan oleh arsitektur Xception dengan nilai 0,988 [8]. Sedangkan pada hasil penelitian berikut membuktikan hasil output pengukuran kinerja menggunakan *confusion matrix* mendapatkan nilai akurasi lebih tinggi dengan 0.992. Hasil nilai evaluasi ini membuktikan dapat melampaui nilai akurasi dari penelitian sebelumnya dengan arsitektur VGG16 dengan nilai 92,7% dan arsitektur Xception dengan nilai 0,988.

3.5. Tahap Pengujian

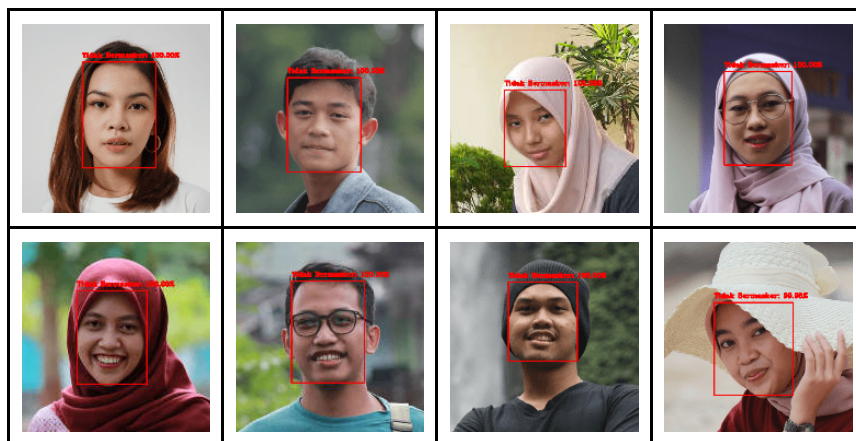
Pengujian pada sistem menggunakan SSD ResNet10. SSD ResNet (*Residual Neural network*) adalah suatu arsitektur yang begitu populer dan digunakan dalam melakukan deteksi objek pada gambar [19]. Pada tahap pengujian ini digunakan data testing yaitu sampel foto objek wajah bermasker dan tidak bermasker total berjumlah 16 data testing. Pada sistem ini terdapat 2 kelas dengan pemberian nilai 0 untuk dataset citra objek wajah tidak bermasker dan diberikan tanda warna merah. Sedangkan memberikan warna 1 untuk dataset wajah yang

menggunakan masker dan diberikan tanda dengan warna hijau. Tahap dari pengujian ini yaitu memasukan satu per satu sampel dataset berupa gambar dengan foto objek wajah bermasker dan tidak bermasker. Berikut merupakan tabel dari hasil tahap pengujiannya.

Tabel 5. Hasil Testing Dataset Gambar Menggunakan Masker



Tabel 6. Hasil Testing Dataset Gambar Tidak Menggunakan Masker



IV. KESIMPULAN

Dengan dataset yang dikumpulkan berjumlah total 7.553 file yang terdiri dari 3.725 file gambar yang berisikan wajah orang-orang yang menggunakan masker serta 3.828 file gambar yang berisikan wajah orang-orang yang tidak menggunakan masker. Pengambilan dataset yang di implementasikan ke sistem berjumlah 16 file yang terdiri dari 8 file gambar tidak memakai masker dan 8 file gambar yang memakai masker. Dari penelitian ini pada tahap evaluasi jaringan menggunakan *confusion matrix* didapatkan nilai *accuracy* 0.992, *precision* 1.000, dan *recall* 0.984. Kemudian untuk tahap pengujian dataset testing yaitu didapatkan nilai tertinggi dengan akurasi 100%.

DAFTAR PUSTAKA

- [1] Universitas Medan Area, “Apa Itu Era Society 5.0 dan Apa Perbedaannya dengan Era Industri 4.0?,” 11 01 2022. [Online]. Available: <https://barki.uma.ac.id/2022/01/11/apa-itu-era-society-5-0-dan-apa-perbedaannya-dengan-era-industri-4-0/>.
- [2] Nisa, “Mengenal Apa Itu Society 5.0 dan Contoh Penerapannya dalam Berbagai Bidang,” 04 04 2022. [Online]. Available: <https://inmarketing.id/society-5-0-adalah.html>.
- [3] G. D. Nursyafitri, “Kenali Penggunaan Computer Vision dalam Data Science,” 12 04 2022. [Online]. Available: <https://dqlab.id/kenali-penggunaan-computer-vision-dalam-data-science>.
- [4] M. R. A. Yudianto, *KLASIFIKASI CITRA WAYANG PUNAKAWAN MENGGUNAKAN METODE GAUSSIAN FILTER DAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK*, 2021.
- [5] A. Rahim, *ANALISIS KOMPARASI CONVOLUTIONAL NEURAL NETWORK (CNN) DENGAN SUPPORT VECTOR MACHINE (SVM) UNTUK KLASIFIKASI CITRA PENGGUNA MASKER*, 2021.
- [6] B. Kocacinar, B. Tas, F. P. Akbulut, C. Catal dan D. Mishra, *A Real-Time CNN-Based Lightweight Mobile Masked Face Recognition System*, 2022.
- [7] S. Singh, U. Ahuja, M. Kumar, K. Kumar dan M. Sachdeva, *Face mask detection using YOLOv3 and faster R-CNN models: COVID-19 environment*, 2021.

- [8] M. F. Naufal dan S. F. Kusuma, *PENDETEKSI CITRA MASKER WAJAH MENGGUNAKAN CNN DAN TRANSFER LEARNING*, 2021.
- [9] R. Firdaus, "Computer Vision," *Rancang Bangun Sistem Navigasi Multirotor Berbasis Waypoint Dan Computer Vision*, 2020.
- [10] D. Agustiani, *Implementasi Machine Learning dan Computer Vision pada Pengembangan Sistem Otomasi Klasifikasi dan Perhitungan Kendaraan*, 2019.
- [11] X. Wu, D. Sahoo dan S. C.H.Hoi, *Recent advances in deep learning for object detection*, pp. 39-64, 2020.
- [12] S. Christin, É. Hervet dan N. Lecomte, "Applications for deep learning in ecology," dalam *Methods in Ecology and Evolution*, BRITISH ECOLOGICAL SOCIETY, 2019, pp. 1632-1644.
- [13] H. Fonda, *KLASIFIKASI BATIK RIAU DENGAN MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORKS (CNN)*, 2020.
- [14] E. N. Arrofiqoh dan Harintaka, *IMPLEMENTASI METODE CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI TANAMAN PADA CITRA RESOLUSI TINGGI*, 2018.
- [15] R. Dharmadi, "CNN: Beyond Image Classification," 09 05 2018. [Online]. Available: <https://medium.com/nodeflux/cnn-beyond-image-classification-3b9b0af021a9>.
- [16] A. Saputro, S. Mu'min, M. Lutfi dan H. Putri, "VGGNet," *DEEP TRANSFER LEARNING DENGAN MODEL ARSITEKTUR VGG16 UNTUK KLASIFIKASI JENIS VARIETAS TANAMAN LENGKENG BERDASARKAN CITRA DAUN*, vol. 6, p. 612, 2022.
- [17] T. S. N. P. Putri, M. A. Fikih dan N. Setyawan, "Training Convolutional Neural Networks," *FACE MASK DETECTION COVID-19 USING CONVOLUTIONAL NEURAL NETWORK (CNN)*, 2020.
- [18] BINUS University School of Computer Science, "Confusion Matrix," 01 11 2020. [Online]. Available: <https://socs.binus.ac.id/2020/11/01/confusion-matrix/>. [Diakses 26 12 2022].
- [19] F. O. Reynaldi, O. Pahlevi dan I. Suryani, "Residual Neural Network (ResNet)," *ANALISA PERFORMA ARSITEKTUR MOBILENETV1 DAN RESNET MENGGUNAKAN META-LEARNING DALAM MENDETEKSI OBJEK HEWAN KUCING*, vol. 4, no. 1, 2021.