

Analisis Akurasi Perbandingan Jumlah Layer Deteksi Warna Objek Menggunakan Algoritma Convolutional Neural Network

Dio Prasetyo^{a,1,*}, Ninuk Wiliani^{a,2}^{a,b} Teknik Informatika, Universitas Pancasila, Jakarta, Indonesia¹4522210148@univpancasila.ac.id, ²Ninuk.wiliani@univpancasila.ac.id

*Penulis koresponden

Diterima	Direvisi	Disetujui	Dipublikasikan
06/01/2025	26/06/2025	27/06/2025	28/06/2025

ABSTRACT

This study evaluates the impact of variations in the number of layers on the implementation of the Convolutional Neural Network (CNN) algorithm in a color-based object identification and categorization system, using python language supported by the TensorFlow/Keras framework. The data used is a collection of visual data in the form of red and white cups divided into a proportion of 90% training data and 10% testing data in the dataset in this study which amounted to 62 red cup data and 59 white cup data. Testing was carried out by comparing three different convolution layer configurations of 1, 2, and 3 layers, where each configuration was integrated with a max pooling and fully connected layer. The results of the study showed an accuracy of 92%, precision of 93%, recall of 92%, and f1-score of 92%. On the other hand, the application of two and three convolution layers actually showed a significant decline with an accuracy of only 46%.

ABSTRAK

Penelitian ini mengevaluasi dampak variasi jumlah lapisan layer pada implementasi algoritma Convolutional Neural Network (CNN) dalam sistem identifikasi dan kategorisasi objek berbasis warna, menggunakan bahasa python dengan dukungan framework TensorFlow/Keras. Data yang digunakan adalah kumpulan data visual berupa cangkir merah dan putih yang terbagi dalam proporsi 90% data training dan 10% data testing pada dataset dalam penelitian ini yang berjumlah 62 data cangkir berwarna merah dan 59 data cangkir berwarna putih. Pengujian dilakukan dengan membandingkan tiga konfigurasi lapisan konvolusi berbeda 1, 2, dan 3 lapisan layer, dimana setiap konfigurasi diintegrasikan dengan lapisan max pooling dan fully connected. Hasil penelitian menunjukkan akurasi 92%, precision 93%, recall 92%, dan f1-score 92%. Di sisi lain, penerapan dua dan tiga lapisan konvolusi justru menunjukkan kemunduran signifikan dengan akurasi hanya 46%.

KEYWORDS

*Convolutional
Neural Network
Object Detection
Color
Classification
Transfer Learning
Deep Learning*

KATA KUNCI

*Convolutional
Neural Network
Deteksi Objek
Klasifikasi Warna
Transfer Learning
Deep Learning*

This is an open access article under the CC-BY-SA license.



1 PENDAHULUAN

Pada saat sekarang ini diperlukan komputer yang mampu mengenali objek secara visual dan real-time untuk mempermudah pekerjaan manusia di berbagai bidang, termasuk keamanan siber dalam pengoperasian keamanan komputer serta bidang medis, seperti mendeteksi dan mengklasifikasikan jenis penyakit secara otomatis berdasarkan gambar yang diinput secara visual dan real-time. Kemampuan ini dapat membantu mempercepat pekerjaan serta menghemat waktu secara signifikan. Salah satu algoritma populer dalam penelitian pengolahan data citra adalah Convolutional Neural Network (CNN), yaitu arsitektur jaringan saraf tiruan yang menyerupai cara kerja jaringan saraf manusia dalam mendeteksi dan mengklasifikasikan objek berdasarkan warna. CNN memiliki kemampuan untuk mengekstraksi fitur hierarkis dari data gambar, sehingga menjadi alat yang efektif untuk mengenali pola visual kompleks secara real-time. CNN dapat diterapkan untuk deteksi objek daun semanggi secara real-time menggunakan metode Single Shot Multibox Detector (SSD), yang menghasilkan akurasi signifikan dalam klasifikasi objek [1].

Salah satu kendala utama dalam penerapan CNN untuk deteksi warna objek adalah menentukan jumlah lapisan (layer) yang optimal. Lapisan yang terlalu sedikit mungkin tidak cukup untuk menangkap kompleksitas fitur warna, sedangkan jumlah lapisan yang terlalu banyak dapat menyebabkan overfitting dan menambah beban komputasi. Selain itu, variasi jumlah dataset juga berpengaruh terhadap akurasi pengenalan. Widystuti dan Darmawan (2018) menyebutkan bahwa jumlah dataset yang lebih besar memberikan pengaruh positif terhadap akurasi pengenalan dalam deep convolutional network [2].

Penelitian sebelumnya telah mengkaji pengaruh jumlah layer pada CNN terhadap akurasi klasifikasi gambar secara umum. Ramadhan et al. (2024) melakukan perbandingan jumlah layer pada CNN untuk meningkatkan akurasi klasifikasi gambar dan menemukan bahwa semakin banyak jumlah layer, semakin tinggi tingkat akurasi yang diperoleh [3]. Namun, penelitian yang secara spesifik menganalisis pengaruh jumlah layer terhadap akurasi dalam konteks deteksi warna objek masih sangat terbatas. Oleh karena itu, studi lanjutan diperlukan untuk memahami bagaimana konfigurasi layer pada algoritma CNN memengaruhi akurasi dalam deteksi warna objek.

Penelitian ini bertujuan untuk menganalisis pengaruh jumlah layer pada CNN terhadap akurasi deteksi warna objek. Dengan memahami hubungan ini, diharapkan konfigurasi CNN yang optimal dapat ditentukan untuk tugas deteksi warna, sehingga meningkatkan kinerja sistem pengenalan visual. Penelitian ini juga bermanfaat dalam pengembangan model CNN yang lebih efisien dan akurat untuk aplikasi pengolahan citra, seperti sistem pengenalan objek otomatis dan aplikasi lain yang membutuhkan deteksi warna yang presisi.

2 METODE PENELITIAN

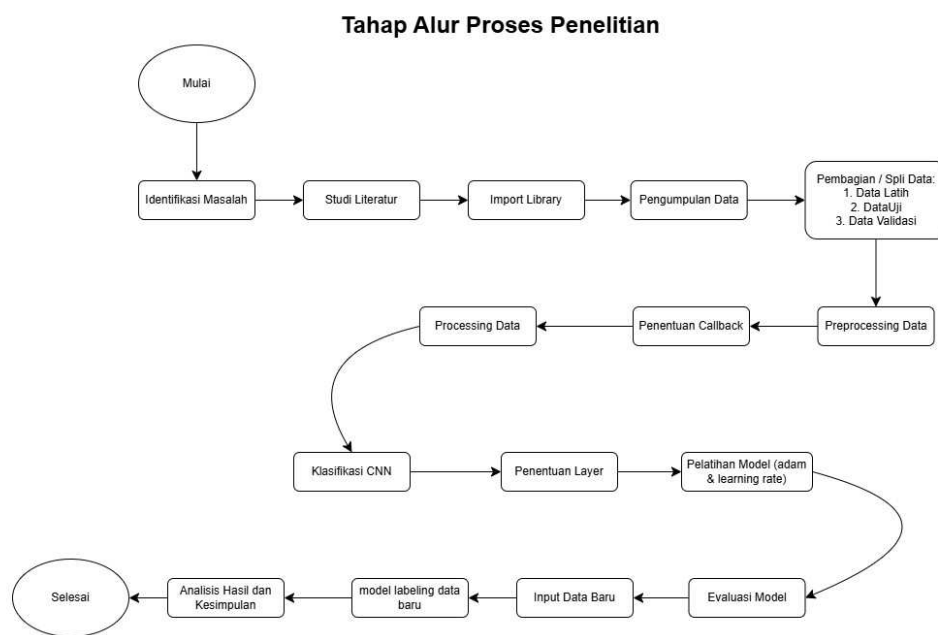
Pada bagian ini dijelaskan secara rinci langkah-langkah yang diambil dalam penelitian untuk menganalisis pengaruh jumlah lapisan (layer) pada algoritma Convolutional Neural Network (CNN) dalam mendeteksi dan mengklasifikasikan objek berdasarkan warna. Tujuan dari penelitian ini adalah untuk menemukan konfigurasi layer CNN yang optimal guna meningkatkan akurasi pengenalan objek secara visual dan real-time.

Penelitian ini menggunakan platform Google Colab sebagai alat utama untuk menjalankan kode program, mengolah data, dan membangun model algoritma CNN. Bahasa pemrograman Python digunakan dalam implementasi algoritma CNN, dengan memanfaatkan framework TensorFlow dan Keras yang tersedia di Google Colab. Dataset yang digunakan terdiri dari gambar cangkik berwarna merah dan putih, yang dikumpulkan melalui library Python yang secara otomatis mencari dan menyimpan gambar, baik di memori lokal komputer maupun di penyimpanan data lain seperti Google Drive. Library Python, seperti pip install `bing-image-downloader`, mencari gambar sesuai dengan kriteria atau judul yang ditentukan oleh peneliti dari berbagai sumber di internet. Setelah itu, data gambar tersebut diproses melalui tahap preprocessing, termasuk normalisasi data dan augmentasi gambar.

Proses penelitian ini melibatkan pembagian dataset menjadi data pelatihan, data uji, dan data validasi. Selanjutnya, dilakukan konfigurasi model CNN dengan variasi jumlah layer untuk

mengevaluasi pengaruhnya terhadap akurasi prediksi. Hasil akhir dari penelitian ini adalah analisis akurasi model CNN dalam mendeteksi warna objek dan rekomendasi konfigurasi layer yang optimal berdasarkan eksperimen yang dilakukan.

Alur penelitian ini dapat dilihat pada diagram flowchart yang disertakan dalam sub-bab ini. Diagram tersebut menggambarkan langkah-langkah penelitian secara sistematis, mulai dari identifikasi masalah hingga analisis hasil dan kesimpulan yang menentukan konfigurasi layer yang tepat untuk digunakan dalam algoritma CNN dalam mengklasifikasikan objek berdasarkan ekstraksi fitur warna dan bentuk objek. Diagram alir atau flowchart ini disajikan pada Gambar 1.



Gambar 1. Alur Penelitian

2.1 Identifikasi Masalah

Penelitian dimulai dengan peneliti meneliti permasalahan utama pada penelitian sebelumnya dalam pengklasifikasian objek berdasarkan warna. Permasalahan ini mencakup kurangnya pemahaman tentang pengaruh jumlah layer CNN terhadap akurasi dan hasil prediksi dan klasifikasi data baru pada pemodelan algoritma CNN yang dapat dilihat secara visual dan real-time. Menurut (Zhang et al, 2021) menyatakan bahwa konfigurasi layer pada CNN memainkan peran penting dalam menentukan kinerja model dalam klasifikasi gambar [4].

2.2 Studi Literatur

Dalam tahap ini peneliti melakukan studi literatur dari berbagai sumber informasi seperti jurnal ilmiah, buku, atau sumber informasi lainnya untuk memahami konsep CNN dan implementasinya dalam mengklasifikasikan objek berdasarkan ekstraksi fitur warna dan untuk menemukan jawaban sementara atau hipotesis sebelum melakukan percobaan eksperimen lebih lanjut. Dalam beberapa kajian literatur yang dibaca dan dipahami oleh peneliti ditemukan bahwa penentuan layer yang optimal dalam pemodelan algoritma CNN dalam mendeteksi objek berbasis warna sangat menentukan akurasi dan hasil kualitas model yang didapatkan dalam memprediksi objek berdasarkan ekstraksi fitur warna. Menurut (Krizhevsky et al, 2012), CNN telah terbukti efektif dalam mendeteksi pola visual kompleks melalui proses ekstraksi fitur hierarkis, yang membuatnya menjadi algoritma yang sangat populer dalam klasifikasi gambar.

2.3 Import Library

Library python seperti TensorFlow, Keras, NumPy, dan Matplotlib diimpor untuk membantu proses pengolahan data, pembuatan model algoritma CNN, dan visualisasi hasil prediksi pemodelan. Adapun keterangan dan kegunaan masing-masing library python akan diuraikan yaitu sebagai berikut ini:

1. Library `bing_image_downloader` digunakan untuk mengunduh data gambar dari Bing Image Search secara otomatis yang dicari dari berbagai sumber di internet berdasarkan kata kunci yang diberikan oleh peneliti
2. Pandas digunakan untuk manipulasi, meyimpan, dan analisis data berbasis tabular (DataFrame) [5].
3. Numpy digunakan untuk komputasi numerik dan operasi matematik dalam program terutama operasi array multidimensi [6].
4. Matplotlib dan seaborn digunakan untuk membuat plotting grafik dan visualisasi hasil analisis data [7].
5. sklearn (scikit-learn) digunakan untuk menyediakan alat tau tools bantu yang digunakan pada pemodelan machine laearning seperti regresi, klasifikasi dalam mengklasifikan data objek atau mengumpulkan data objek berdasarkan kriteria tertentu [8].
6. Tensorflow, Keras, os adalah framework yang melatih deep learning dan mendukung pembuatan model neural network dalam melatih, mendeteksi dan mengklasifikasikan objek, keras menyediakan API tingkat tinggi untuk membangun model dengan lebih penting dalam CNN aritektur layer seperti Conv2D, Dense, Dropout sangat menentukan akurasi pemodelan dan keakuratan prediksi kalsifikasi data baru sesuai dengan kelas data sebenarnya dalam pemodelan CNN, untuk optimizer Adam digunakan untuk optimasi training [9].
7. Callback seperti early stopping, `reducelronplateau`, `modelcheckpoint` yang fungsinya digunakan sebagai pengontrol pemodelan selama proses pelatihan data untuk menghindari overfitting dan mempercepat proses pelatihan yang masing-masing memiliki fungsi yaitu seperti early stopping digunakan untuk menghentikan proses pelatihan pemodelan ketika akurasi tidak meningkat, `reducelronplateau` digunakan untuk mengurangi learning rate secara adaptif, serta model checkpoint digunakan untuk menyimpan model dengan performa terbaik selama pelatihan [10].
8. Struktur Model CNN terdapat beberapa yaitu sebagai berikut Conv2D digunakan sebagai layer konvolusi untuk mengekstrak fitur lokal dari gambar, BatchNormalization digunakan untuk menormalkan output layer untuk stabilitas pelatihan, Max Pooling 2D digunakan untuk mengurangi dimensi spasial dari objek untuk mengurangi kompleksitas dalam mendeteksi objek, Dense digunakan sebagai fully connected layer untuk melakukan klasifikasi akhir, dan Dropout digunakan untuk mencegah overfitting dengan mematikan beberapa neuron selama pelatihan [11].
9. Image Data Generator Library yang berasal dari framework keras untuk modifikasi dan augmentasi data gambar yang fungsinya untuk membantu mencegah overfitting dan menghasilkan variasi gambar training untuk meningkatkan keragaman dataset tanpa perlu menambah data baru untuk mencegah terjadinya overfitting [12].

2.4 Pengumpulan Data

Dataset yang digunakan dalam penelitian ini berupa gambar cangkir berwarna merah dan putih yang dikumpulkan menggunakan library python seperti OpenCV dan pip install `bing_image_downloader` kemudian dataset disimpan dalam format yang sesuai untuk pelatihan model.

2.5 Pembagian Dataset

Setelah data dikumpulkan kemudian dataset dibagi menjadi tiga bagian yaitu sebagai berikut:

1. Data Latih dengan kuantittas data 90% dari total data keseluruhan adalah data yang digunakan untuk melatih model algoritma selama proses pelatihan
2. Data Uji dengan kuantitas data 5% dari total data keseluruhan adalah data yang digunakan untuk mengevaluasi performa model pada data yang belum pernah diklasifikasikan

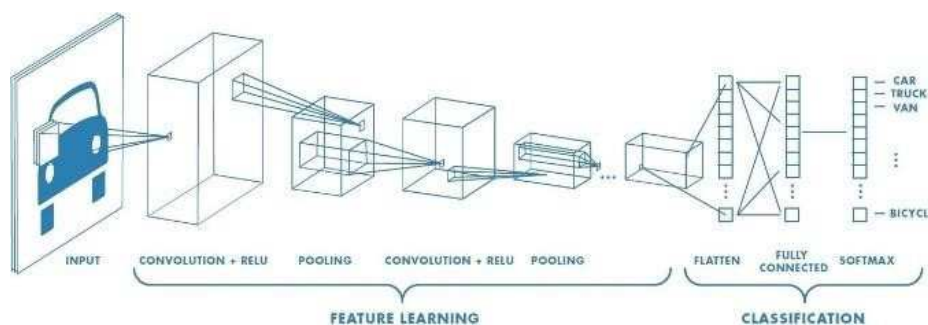
3. Data Validasi dengan kuantitas 5 % dari total data keseluruhan adalah data yang digunakan untuk mencegah overfitting selama pelatihan, pembagian dataset yang seimbang adalah kunci untuk menghindari bias model [10].

2.6 Preprocessing Data

Pada tahap ini dilakukan preprocessing atau persiapan data sebelum melakukan proses pemodelan atau training data gambar yang meliputi beberapa tahap yaitu seperti normalisasi data yang digunakan agar pixel gambar dinormalisasi menjadi rentang nilai $[0,1]$ untuk mempercepat proses pelatihan data, Augmentasi data yang digunakan untuk rotasi, flipping, dan scaling data yang fungsinya untuk meningkatkan keragaman data latih, menurut (Shorten dan Khoshgoftaar, 2019) mengemukakan pendapat bahwa augmentasi data dapat meningkatkan generalisasi model deep learning [13]. Callback diterapkan untuk mengontrol proses pelatihan model, seperti early stopping dan learning rate scheduler yang diterapkan untuk menghentikan pelatihan ketika akurasi berhenti meningkat dan menyesuaikan learning rate selama pelatihan [10].

2.7 Pelatihan model

Model CNN dilatih menggunakan optimizer adam dengan berbagai variasi jumlah layer yang ditentukan oleh peneliti misalnya 1, 2, 3, 5, dan 7 layer. Proses ini melibatkan evaluasi akurasi pemodelan dan kualitas generalisasi prediksi data baru agar sesuai dengan kelas sebenarnya pada objek data citra berdasarkan jumlah layer untuk menentukan konfigurasi yang optimal dalam mendeteksi dan mengklasifikasikan objek. Optimizer adam digunakan penelitian ini untuk mengoptimalkan model deep learning, contoh pelatihan model dengan menggunakan algoritma CNN dapat dilihat seperti pada Gambar 2 [13].



Gambar 2. Pemodelan CNN

2.8 Evaluasi Model

Evaluasi model dilakukan dengan menggunakan metrik evaluasi seperti classification report yang digunakan untuk melihat akurasi yang fungsinya untuk menilai keakuratan prediksi pemodelan dalam mendeteksi objek yang sesuai dengan kelas sebenarnya pada objek, precision dan recall yang digunakan untuk mengukur kualitas klasifikasi dalam mendeteksi objek [14]. Kombinasi precision dan recall memberikan pandangan yang lebih kompherensif tentang performa model [15].

3 HASIL DAN PEMBAHASAN

Berikut dalam bab ini akan dijelaskan hasil dan pembahasan penelitian berdasarkan dengan alur metode penelitian diatas dalam variasi beberapa layer dalam mendeteksi dan mengklasifikasikan objek berdasarkan ekstraksi fitur warna seperti contohnya pada penelitian ini dilakukan dengan menggunakan 3 kali percobaan dengan masing-masing percobaan dilakukan dengan jumlah layer yang berbeda seperti pada percobaan pertama dilakukan dengan menggunakan 1 layer konvolusi, 1 layer max pooling, dan 1 fully connected layer dalam mendeteksi objek berdasarkan ekstraksi fitur warna yang ada dan begitupun pada percobaan-percobaan selanjutnya dilakukan dengan jumlah layer yang berbeda dalam metode algoritma CNN. Hal ini digunakan untuk menentukan tingkat akurasi dan kualitas regenerasi prediksi data baru dalam pemodelan algoritma CNN dan untuk menganalisis konfigurasi dan jumlah layer mana yang terbaik dalam mendeteksi dan

mengklasifikasikan objek, tingkat akurasi diukur menggunakan metrik evaluasi seperti *classification report* yang terbukti sangat valid dalam menentukan akurasi pada pemodelan.

3.1 Pengujian pada algoritma CNN

Pada sub bab ini akan dijabarkan hasil penelitian berupa persentase tingkat akurasi pemodelan dengan variasi jumlah layer yang berbeda dalam 3 kali percobaan berturut-turut yang setiap iterasi percobaan jumlah layer mengikuti jumlah iterasi dalam percobaan dalam pemodelan algoritma *CNN* dengan mengukur tingkat akurasi menggunakan metrik evaluasi .

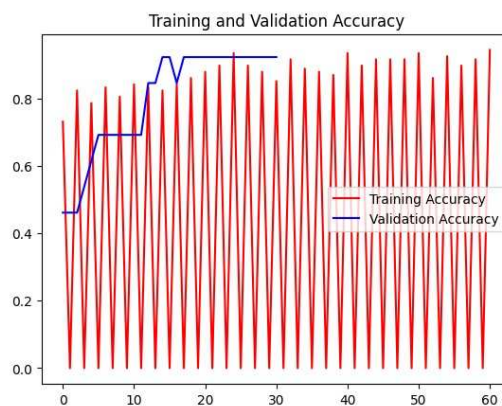
Tabel 1. Accuracy Pemodelan *CNN*

Range Data	Jumlah Layer	Accuracy Data	
		Train	Test
90% Train : 10% Test	1 Layer	82.41%	92.31%
90% Train : 10% Test	2 Layer	49.07%	46.15%
90% Train : 10% Test	3 Layer	53.70%	46.15%

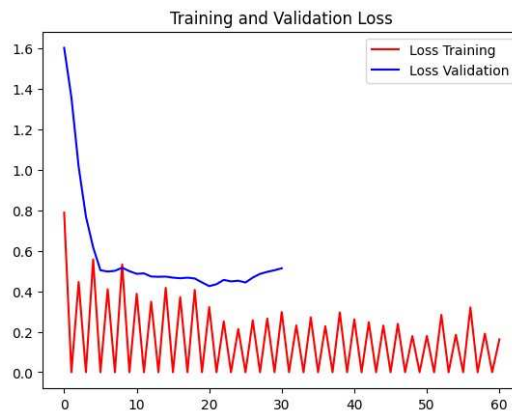
Dari hasil tabel hasil akurasi tersebut yang diukur menggunakan metrik evaluasi seperti *classification report* diatas dapat diketahui bahwa akurasi data testing dan training data yang dilakukan oleh pemodelan algoritma *CNN* dengan menggunakan 1 layer konvolusi menunjukkan persentase akurasi terbaik dalam proses pelatihan dan pengujian atau validasi data sehingga pemodelan algoritma ini sangat cocok dalam menggunakan 1 layer konvolusi, 1 layer max pooling, dan 1 fully connected layer dalam mendeteksi dan mengklasifikasikan objek dengan benar.

3.2 Hasil Grafik Akurasi dan Loss Pada Algoritma *CNN*

Percobaan dilakukan sebanyak tiga kali berturut-turut untuk menentukan grafik terbaik dalam pemodelan yang mencapai nilai tertinggi pada akurasi, mendekati atau mencapai 1.0, dan nilai terendah pada loss, mendekati atau mencapai 0.0. Hal ini sangat penting untuk menilai kualitas variasi jumlah layer dalam pemodelan algoritma *CNN* untuk mendeteksi dan mengklasifikasikan objek. Berikut dijabarkan hasil grafik training dan validation accuracy serta grafik training dan validation loss pemodelan *CNN* dengan menggunakan jumlah 1 layer konvolusi, 1 layer max pooling, dan 1 fully connected layer yang tertera pada Gambar 4 dan Gambar 5.



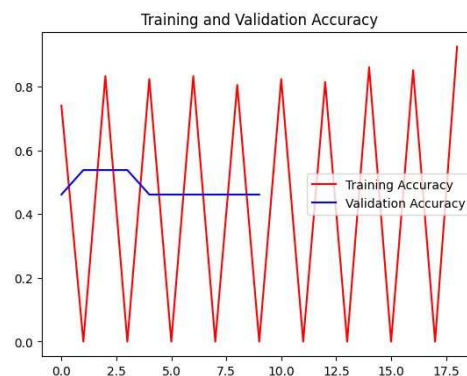
Gambar 4. Grafik Train dan Validation Accuracy Pemodelan *CNN* 1 layer



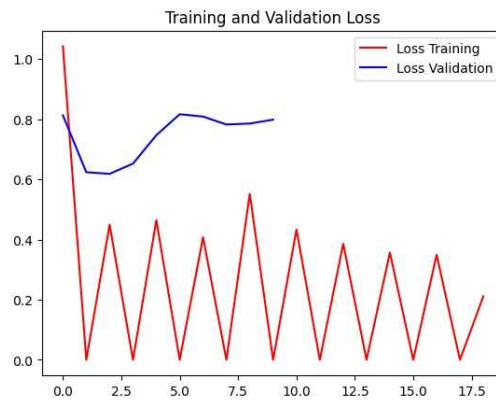
Gambar 5. Grafik Train dan Validation Loss Pemodelan CNN 1 layer

Pada hasil grafik tersebut menunjukkan bahwa tingkat akurasi data training dan data validasi dengan menggunakan 1 layer konvolusi menunjukkan hasil yang terbaik dengan grafik yang cenderung naik menjulang tinggi ke atas mendekati atau mencapai nilai 1.0 dan grafik training dan validation loss yang cenderung menurun ke bawah mendekati atau mencapai 0.0 hal ini memberikan keterangan bahwa pemodelan algoritma *CNN* dengan menggunakan 1 layer konvolusi, 1 layer max pooling, dan 1 fully connected layer sangat baik dan sangat bagus dalam melakukan proses pelatihan dan pengujian data sehingga dapat disimpulkan dari grafik *accuracy* dan *loss* dengan pemodelan algoritma *CNN* dengan menggunakan 1 layer konvolusi dapat dengan baik mengenali dan mengklasifikasikan objek data dengan memberikan label yang tepat pada data tersebut.

Sementara itu untuk grafik *training* and *validation accuracy* menggunakan 2 layer konvolusi, 1 layer max pooling, dan 1 fully connected layer menghasilkan grafik training dan validation accuracy serta grafik training dan validation loss pemodelan *CNN* dengan menggunakan jumlah 2 layer konvolusi, 2 layer max pooling, dan 1 fully connected layer yang tertera pada Gambar 6 dan Gambar 7.



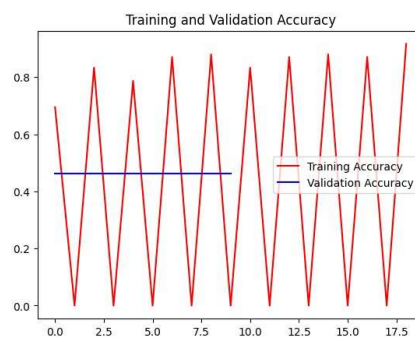
Gambar 6. Grafik Train dan Validation Accuracy Pemodelan *CNN* 2 layer



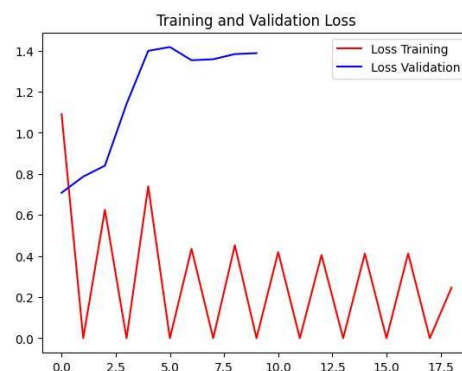
Gambar 7. Grafik Train dan Validation Loss Pemodelan *CNN* 2 layer

Pada hasil grafik tersebut menunjukkan bahwa tingkat akurasi data training dan data validasi dengan menggunakan 2 layer konvolusi menunjukkan hasil buruk dengan grafik yang cenderung turun dan stabil di nilai mendekati atau mencapai nilai 0.4 dan grafik training dan validation loss yang cenderung naik menjulang tinggi ke atas mendekati atau mencapai 1.0 hal ini memberikan keterangan bahwa pemodelan algoritma *CNN* dengan menggunakan 2 layer konvolusi, 2 layer max pooling, dan 1 fully connected layer tidak baik dan tidak bagus dalam melakukan proses pelatihan dan pengujian data sehingga dapat disimpulkan dari grafik *accuracy* dan *loss* dengan pemodelan algoritma *CNN* dengan menggunakan 2 layer konvolusi tidak dapat mengenali dan mengklasifikasikan objek data dengan benar.

Sementara itu untuk hasil grafik training dan validation accuracy serta grafik training dan validation loss pemodelan *CNN* dengan menggunakan jumlah 3 layer konvolusi, 3 layer max pooling, dan 1 fully connected layer yang tertera pada Gambar 8 dan Gambar 9.



Gambar 8. Grafik Train dan Validation Accuracy Pemodelan *CNN* 3 layer

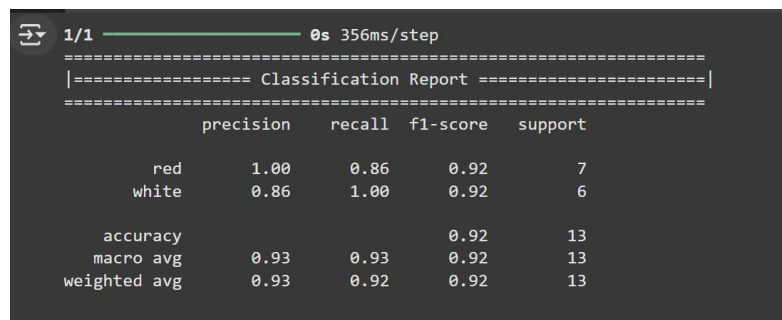


Gambar 9. Grafik Train dan Validation Loss Pemodelan *CNN* 3 layer

Pada hasil grafik tersebut menunjukkan bahwa tingkat akurasi data training dan data validasi dengan menggunakan 3 layer konvolusi menunjukkan hasil yang sangat buruk dengan grafik yang cenderung stabil di nilai mendekati atau mencapai nilai 0.4 dan grafik training dan validation loss yang cenderung naik menjulang tinggi ke atas mencapai 1.0 hal ini memberikan keterangan bahwa pemodelan algoritma *CNN* dengan menggunakan 3 layer konvolusi, 3 layer max pooling, dan 1 fully connected layer sangat tidak baik dan tidak bagus dalam melakukan proses pelatihan dan pengujian data sehingga dapat disimpulkan dari grafik *accuracy* dan *loss* dengan pemodelan algoritma *CNN* dengan menggunakan 3 layer konvolusi tidak dapat mengenali dan mengklasifikasikan objek data dengan benar.

3.3 Hasil Classification Report Algoritma CNN

Dijabarkan hasil *classification report* dari beberapa penentuan variasi jumlah layer pada algoritma *CNN* didalam penelitian ini variasi penentuan jumlah layer berbeda pada setiap percobaan yang dilakukan untuk melihat hasil *accuracy*, *precision*, dan *recall* pada masing-masing layer yang akan digunakan untuk menentukan layer yang optimal pada pemodelan algoritma *CNN* dalam mendeteksi dan mengklasifikasikan objek dengan memberikan label atau kelas yang tepat pada objek tersebut. Hasil *classification report* algoritma *CNN* menggunakan 1 layer konvolusi, 1 layer max pooling, dan 1 fully connected layer dalam mendeteksi objek berdasarkan warna dalam format *RGB* dan bentuk objek yang dapat dilihat pada Gambar 10 berikut ini.



```

1/1 ----- 0s 356ms/step
=====
|===== Classification Report =====|
=====
      precision    recall  f1-score   support

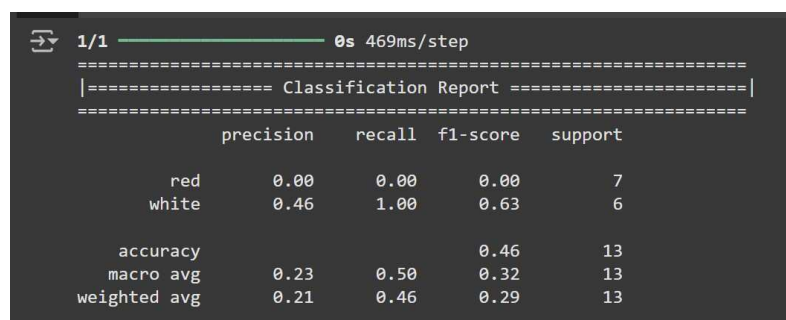
    red           1.00      0.86      0.92         7
    white          0.86      1.00      0.92         6

 accuracy          0.93      0.93      0.92        13
 macro avg          0.93      0.93      0.92        13
 weighted avg       0.93      0.92      0.92        13

```

Gambar 10. Hasil Classification Report Pemodelan *CNN* 1 layer konvolusi

Berdasarkan hasil *classification report* yang didapatkan dapat dilihat pada Gambar 10 didapatkan hasil bahwa akurasi menunjukkan tingkat presentase sebesar 92%, rata-rata tertimbang precision sebesar 93%, rata-rata tertimbang recall sebesar 92%, dan rata-rata tertimbang f1 score 92% hal ini menunjukkan keberhasilan pemodelan algoritma *CNN* dengan 1 layer konvolusi dalam mendeteksi dan mengklasifikasikan objek dengan benar dan mampu melakukan proses pelatihan dan pengujian data dengan cukup baik dan benar sehingga hasil yang diharapkan mampu memprediksi data baru dengan benar. Sementara itu hasil *classification report* algoritma *CNN* menggunakan 2 layer konvolusi 2 layer max pooling, dan 1 fully connected layer dalam mendeteksi objek berdasarkan warna dalam format *RGB* dan bentuk objek yang dapat dilihat pada Gambar 11 berikut ini.



```

1/1 ----- 0s 469ms/step
=====
|===== Classification Report =====|
=====
      precision    recall  f1-score   support

    red           0.00      0.00      0.00         7
    white          0.46      1.00      0.63         6

 accuracy          0.23      0.50      0.32        13
 macro avg          0.23      0.50      0.32        13
 weighted avg       0.21      0.46      0.29        13

```

Gambar 11. Hasil Classification Report Pemodelan *CNN* 2 layer konvolusi

Berdasarkan hasil *classification report* yang didapatkan dapat dilihat pada Gambar 11 didapatkan hasil bahwa akurasi menunjukkan tingkat presentase sebesar 46%, rata-rata tertimbang

precision sebesar 21%, rata-rata tertimbang recall sebesar 46%, dan rata-rata tertimbang f1 score 29% hal ini menunjukkan kegagalan pemodelan algoritma *CNN* dengan 2 layer konvolusi dalam mendeteksi dan mengklasifikasikan objek dan tidak mampu melakukan proses pelatihan dan pengujian data dengan cukup baik dan benar sehingga hasil yang dihasilkan oleh pemodelan dengan 2 layer konvolusi nantinya tidak akan benar dalam memprediksi data dan tidak memberikan label data yang tepat pada data baru.

Hasil classification report algoritma *CNN* menggunakan 3 layer konvolusi, 3 layer max pooling, dan 1 fully connected layer dalam mendeteksi objek berdasarkan warna dalam format *RGB* dan bentuk objek yang dapat dilihat pada Gambar 12 berikut ini.

```
1/1 ————— 1s 521ms/step
=====
|===== Classification Report =====|
=====
precision    recall  f1-score   support

   red         0.00     0.00     0.00         7
  white         0.46     1.00     0.63         6

 accuracy              0.46         13
  macro avg           0.23     0.50     0.32         13
 weighted avg           0.21     0.46     0.29         13
```

Gambar 12. Hasil Classification Report Pemodelan *CNN* 3 layer konvolusi

Berdasarkan hasil *classification report* yang didapatkan dapat dilihat pada Gambar 12 didapatkan hasil bahwa akurasi menunjukkan tingkat presentase sebesar 46%, rata-rata tertimbang precision sebesar 21%, rata-rata tertimbang recall sebesar 46%, dan rata-rata tertimbang f1 score 29% hal ini menunjukkan kegagalan pemodelan algoritma *CNN* dengan 3 layer konvolusi dalam mendeteksi dan mengklasifikasikan objek dan tidak mampu melakukan proses pelatihan dan pengujian data dengan cukup baik dan benar sehingga hasil yang dihasilkan oleh pemodelan dengan 3 layer konvolusi nantinya tidak akan benar dalam memprediksi data dan tidak memberikan label data yang tepat pada data baru.

Selain itu hasil pemodelan algoritma dengan menggunakan 1 layer konvolusi, 1 layer max pooling, dan 1 fully connected layer dalam mendeteksi dan mengklasifikasi objek baru dengan harapan berdasarkan grafik *accuracy* dan *loss* serta *classification report* yang dilampirkan dapat menunjukkan prediksi data yang akurat dalam mendeteksi dan mengklasifikasi kelas atau label yang tepat pada objek atau data baru, berikut ini dilampirkan hasil klasifikasinya yang terdapat pada gambar 13.

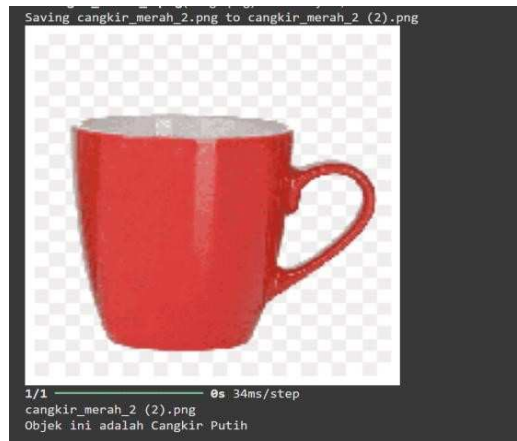


Gambar 13. Hasil Classification Report Pemodelan *CNN* 1 layer konvolusi

Dapat disimpulkan pada hasil klasifikasi data tersebut dengan menggunakan 1 layer konvolusi, 1 layer max pooling, dan 1 fully connected layer dapat dengan baik dalam mendeteksi

dan mengklasifikan objek dengan benar dan memberikan label atau kelas yang tepat pada objek data tersebut, hal ini membuktikan bahwa algoritma dengan 1 layer konvolusi, dan 1 layer max pooling dapat dengan baik dan akurat dalam mendeteksi objek dibandingkan dengan variasi jumlah layer lainnya.

Sebaliknya hasil pada pemodelan algoritma dengan menggunakan 2 layer konvolusi, 2 layer max pooling, dan 1 fully connected layer dalam mendeteksi dan mengklasifikan objek baru dengan data berdasarkan grafik *accuracy* dan *loss* serta *classification report* yang dilampirkan peneliti dapat memprediksi hasil yang ditentukan oleh pemodelan jenis ini tidak akan akurat dalam mendeteksi dan mengklasifikasikan objek data baru dengan benar karena nilai *accuracy* yang terdapat dalam *classification report* dalam proses pelatihan dan pengujian data menunjukkan performa yang buruk.



Gambar 13. Hasil Classification Report Pemodelan *CNN* 2 layer konvolusi

Dapat disimpulkan pada hasil klasifikasi data tersebut dengan menggunakan 2 layer konvolusi, 2 layer max pooling, dan 1 fully connected layer tidak akurat dalam mendeteksi dan mengklasifikan objek dengan benar dan memberikan label atau kelas yang tidak tepat pada objek data tersebut karena pemodelan tidak bisa membedakan antara warna pada background gambar dengan warna pada objek sebenarnya, hal ini membuktikan bahwa algoritma dengan 2 layer konvolusi, dan 2 layer max pooling tidak baik dan akurat dalam mendeteksi objek.

Sementara itu hasil pemodelan algoritma dengan menggunakan 3 layer konvolusi, 3 layer max pooling, dan 1 fully connected layer dalam mendeteksi dan mengklasifikan objek baru dengan data berdasarkan grafik *accuracy* dan *loss* serta *classification report* yang dilampirkan peneliti dapat memprediksi hasil yang ditentukan oleh pemodelan jenis ini sangat tidak akan akurat dalam mendeteksi dan mengklasifikasikan objek data baru dengan benar karena nilai *accuracy* yang terdapat dalam *classification report* dalam proses pelatihan dan pengujian data menunjukkan performa yang buruk.



Gambar 13. Hasil Classification Report Pemodelan *CNN* 2 layer konvolusi

Dapat disimpulkan pada hasil klasifikasi data tersebut dengan menggunakan 3 layer konvolusi, 3 layer max pooling, dan 1 fully connected layer tidak akurat dalam mendeteksi dan mengklasifikasi objek dengan benar dan memberikan label atau kelas yang tidak tepat pada objek data tersebut karena pemodelan tidak bisa membedakan antara warna pada background gambar dengan warna pada objek sebenarnya, hal ini membuktikan bahwa algoritma dengan 3 layer konvolusi, dan 3 layer max pooling tidak baik dan akurat dalam mendeteksi objek.

4 KESIMPULAN

Berdasarkan hasil penelitian ini, dapat disimpulkan bahwa pemodelan algoritma CNN dengan menggunakan 1 layer konvolusi, 1 layer max pooling, dan 1 fully connected layer memberikan performa terbaik dalam mendeteksi dan mengklasifikasi objek berdasarkan fitur warna. Evaluasi menunjukkan bahwa akurasi mencapai 92%, precision sebesar 93%, recall sebesar 92%, dan f1 score sebesar 92%. Sebaliknya, penambahan 2 atau 3 layer konvolusi menyebabkan penurunan performa yang signifikan, dengan akurasi hanya mencapai 46%. Grafik akurasi dan loss juga mendukung temuan ini, di mana konfigurasi dengan 1 layer konvolusi menghasilkan hasil yang lebih stabil dan mendekati nilai optimal dibandingkan dengan konfigurasi lainnya.

Untuk pengembangan penelitian selanjutnya disarankan untuk menggunakan Transfer Learning dengan arsitektur pretrained seperti VGG, ResNet, atau MobileNet, yang dapat meningkatkan akurasi pada dataset terbatas, seperti yang digunakan dalam penelitian ini dengan hanya 121 data, yang termasuk dalam kategori dataset kecil dan terbatas. Selain itu, eksperimen dengan algoritma optimasi seperti AdamW atau RMSprop, serta tuning hyperparameter menggunakan metode grid search atau Bayesian optimization, dapat menghasilkan konfigurasi yang lebih optimal. Penerapan mekanisme perhatian seperti Squeeze-and-Excitation (SE) block atau selfattention juga dapat meningkatkan kemampuan model dalam menangkap fitur-fitur penting dalam gambar. Penggunaan metode Explainable AI (XAI) juga dapat membantu memahami bagian-bagian gambar yang diperhatikan oleh model, memberikan wawasan untuk meningkatkan hasil prediksi dan membuat model lebih mudah diinterpretasikan.

5 KONTRIBUSI PENELITIAN

Penelitian ini dapat berkontribusi pada pengembangan pengetahuan lebih lanjut dalam mengidentifikasi pemodelan algoritma yang optimal untuk pengolahan data citra, sehingga dapat menjadi dasar bagi penemuan dan inovasi di bidang pengolahan data citra di masa mendatang.

UCAPAN TERIMA KASIH

Ucapan terima kasih disampaikan kepada semua pihak dan rekan-rekan peneliti yang telah memberikan kontribusi dalam penelitian ini, yang telah menjadi langkah penting dalam pengembangan serta pemilihan algoritma terbaik untuk pengolahan data citra. Terima kasih juga disampaikan kepada para dosen yang telah memberikan bimbingan, bantuan, dan informasi berharga selama proses penelitian. Selain itu, penghargaan yang sebesar-besarnya diberikan kepada orang tua dan keluarga atas dukungan dan motivasi yang tiada henti, yang memungkinkan penulis untuk menyelesaikan laporan penelitian ini dengan baik. Diharapkan, penelitian ini dapat memberikan manfaat berupa peningkatan wawasan dan pengetahuan bagi pembaca dalam memilih algoritma terbaik untuk pengolahan data citra.

DAFTAR PUSTAKA

- [1] I. Astuti, W. W. Ariestya, and B. Solehudin, "Deteksi Objek Daun Semanggi secara Real TIME Menggunakan CNN-Single Shot Multibox Detector (SSD)," *J. Ilm. Finfo*, vol. 14, no. 1, pp. 47–58, 2022, doi: 10.22441/fifo.2022.v14i1.005.
- [2] W. Silvani, S. Aurelia, N. Zulatifa, T. Agustin, and S. Indonesia, "Pengaruh Jumlah Epoch Terhadap Akurasi Model Convolutional Neural Network dalam Klasifikasi Penyakit pada Tanaman Padi," in *Seminar Nasional Amikom Surakarta (SEMNAS) 2024*, 2024, pp. 179–

190.

- [3] M. A. R. Ramadhan, T. Apriliyan, N. Ananta, and A. A. Zakkyfriza, "Perbandingan Jumlah Layer Pada Convolutional Neural Network Untuk Meningkatkan Akurasi Dalam Klasifikasi Gambar," *Merkurius J. Ris. Sist. Inf. dan Tek. Inform.*, vol. 2, no. 5, pp. 211–217, 2024.
- [4] M. Krichen, "Convolutional Neural Networks: A Survey," *Computers*, vol. 12, no. 8, pp. 1–41, 2023, doi: 10.3390/computers12080151.
- [5] W. McKinney, "Data Structures for Statistical Computing in Python," in *Proceedings of the 9th Python in Science Conference*, S. van der Walt and J. Millman, Eds., 2010, pp. 56–61. doi: 10.25080/Majora-92bf1922-00a.
- [6] C. R. Harris *et al.*, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, 2020, doi: 10.1038/s41586-020-2649-2.
- [7] A. H. Sial, S. Yahya, and S. Rashdi, "Comparative Analysis of Data Visualization Libraries Matplotlib and Seaborn in Python," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 10, no. 1, pp. 277–281, 2021, doi: 10.30534/ijatcse/2021/391012021.
- [8] D. K. Barupal and O. Fiehn, "Generating the blood exposome database using a comprehensive text mining and database fusion approach," *Environ. Health Perspect.*, vol. 127, no. 9, pp. 1–10, 2019, doi: 10.1289/EHP4713.
- [9] M. Abadi *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv Prepr. arXiv1603.04467*, 2016.
- [10] J. Heaton, "Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning," *Genet. Program. Evolvable Mach.*, vol. 19, no. 1–2, pp. 305–307, 2018, doi: 10.1007/s10710-017-9314-z.
- [11] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.
- [12] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J. Big Data*, vol. 6, no. 1, p. 60, 2019, doi: 10.1186/s40537-019-0197-0.
- [13] A. Khan, A. Sohail, U. Zahoor, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, 2020, doi: 10.1007/s10462-020-09825-6.
- [14] P. W. Cahyo and U. S. Aesyi, "Perbandingan LSTM dengan Support Vector Machine dan Multinomial Na ve Bayes pada Klasifikasi Kategori Hoax," *J. Transform.*, vol. 20, no. 2, p. 23, 2023, doi: 10.26623/transformatika.v20i2.5880.
- [15] A. Arum Sari and P. Pramono, "Prediksi Serangan Sql Injection Pada Jaringan Komputer Menggunakan Metode Support Vector Machine (SVM)," *J. Tecnoscienza*, vol. 8, no. 2, pp. 317–326, 2024, [Online]. Available: <https://www.kaggle.com/datasets/syedsaqlainhussain/sql-injection-dataset>