

KLASIFIKASI BUAH SEGAR DAN BUSUK MENGGUNAKAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK (CNN)

Maesha Ayu Syaharani, Theresia Aurelly Claudia Budianto, Riza Ibnu Adam

Informatika, Universitas Singaperbangsa Karawang

Jalan H.S Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Indonesia

maesyaayu62@gmail.com

ABSTRAK

Dalam sektor pertanian, mempertahankan kualitas produk merupakan kunci untuk kepuasan pelanggan dan keunggulan di pasar. Namun, klasifikasi buah segar dan busuk sering menjadi tantangan karena metode konvensional yang bergantung pada inspeksi visual pekerja cenderung tidak efisien, memakan waktu, dan rentan terhadap kesalahan. Penelitian ini bertujuan mengembangkan sistem klasifikasi otomatis menggunakan Convolutional Neural Network (CNN) dengan memanfaatkan dataset gambar buah yang mencakup berbagai kondisi kesegaran, berjumlah total 13.619 gambar. Teknik computer vision dan pengolahan citra diterapkan agar CNN dapat mengekstraksi dan menganalisis fitur visual kompleks dari buah-buahan. Hasil penelitian menunjukkan bahwa model CNN yang dikembangkan berhasil mencapai tingkat akurasi klasifikasi sebesar 96,67% pada data uji, yang mencerminkan kemampuan model dalam mengidentifikasi buah segar dan busuk secara efektif. Hal ini menegaskan bahwa pendekatan berbasis CNN merupakan solusi yang sangat efektif untuk mengotomatisasi proses penilaian kesegaran buah, yang diharapkan dapat meningkatkan efisiensi dan efektivitas dalam menjaga kualitas produk pertanian.

Kata kunci : *klasifikasi buah, convolutional neural network, kualitas produk*

1. PENDAHULUAN

Dalam sektor pertanian, mempertahankan kualitas produk merupakan aspek penting untuk memastikan kepuasan pelanggan dan juga keunggulan pasar [1].

Kualitas produk, khususnya buah-buahan, sangat bergantung pada tingkat konvensional kesegarannya. yang digunakan Metode untuk mengklasifikasikan buah segar dan busuk sering kali bergantung pada inspeksi visual oleh pekerja [2].

Meskipun cara ini sudah lama diterapkan, namun metode ini memiliki beberapa kelemahan, termasuk efisiensi yang rendah, waktu yang lama, dan potensi kesalahan manusia yang cukup tinggi.

Namun, seiring dengan kemajuan teknologi, berbagai metode otomatis berbasis teknologi telah mulai diterapkan untuk mengatasi masalah ini. Salah satunya adalah computer vision, bidang dalam kecerdasan buatan yang memungkinkan komputer untuk memperoleh, memproses, dan menganalisis data dari dunia nyata dalam bentuk gambar atau video. Dengan menggunakan teknik computer vision, komputer mampu melihat dan memahami apa yang dilihatnya, mirip dengan cara manusia melihat dan memahami sesuatu [3].

Dalam konteks pengolahan citra pertanian, computer vision dapat digunakan untuk mengotomatisasi proses klasifikasi dan pengenalan kualitas buah dengan cepat dan akurat, menggantikan kebutuhan akan inspeksi manual. CNN adalah salah satu pendekatan paling populer dalam computer vision.

CNN adalah jenis jaringan saraf tiruan yang telah terbukti efektif dalam mengenali pola dan fitur dalam

gambar, yang membuatnya sangat cocok untuk tugas-tugas klasifikasi gambar [4].

Dalam aplikasi klasifikasi buah, CNN dapat dilatih untuk mengenali perbedaan antara buah segar dan busuk berdasarkan karakteristik visual seperti warna, tekstur, dan bentuk. Pada penelitian sebelumnya yang dilakukan oleh Herlangga, dkk. (2024) telah menunjukkan potensi besar penggunaan CNN dalam pengenalan senjata tradisional Sumatera Barat dengan dataset citra sebanyak 819 data, mencapai akurasi yang sangat tinggi yaitu 100% [5].

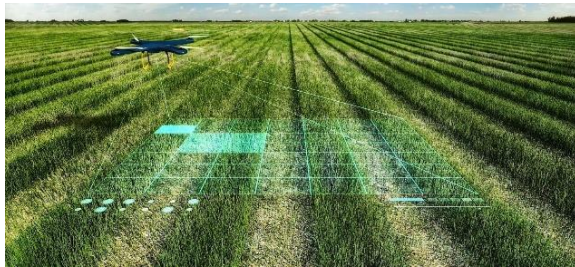
Hal ini menunjukkan bahwa CNN memiliki kemampuan yang signifikan dalam mengenali pola dan fitur spesifik dari gambar yang kompleks. Oleh karena itu, penelitian ini bertujuan untuk mengeksplorasi penerapan CNN dalam klasifikasi buah segar dan busuk guna meningkatkan efisiensi dibandingkan metode konvensional. Dengan demikian, diharapkan hasil penelitian ini tidak hanya memberikan kontribusi teoritis dalam bidang teknologi pengolahan citra dan machine learning, tetapi juga menawarkan solusi praktis yang dapat diterapkan dalam industri pertanian untuk meningkatkan kualitas dan efisiensi proses klasifikasi buah.

2. TINJAUAN PUSTAKA

2.1. Computer Vision

Computer Vision menggunakan kecerdasan buatan untuk memungkinkan komputer mengekstrak informasi dari input visual seperti gambar dan video. Dalam sektor pertanian, computer vision telah diterapkan dalam berbagai aplikasi, termasuk pemantauan pertumbuhan tanaman, deteksi hama, dan klasifikasi kualitas produk pertanian. Penggunaan

computer vision dalam pertanian bertujuan untuk meningkatkan efisiensi dan akurasi dalam proses-proses yang sebelumnya bergantung pada inspeksi manual [6].

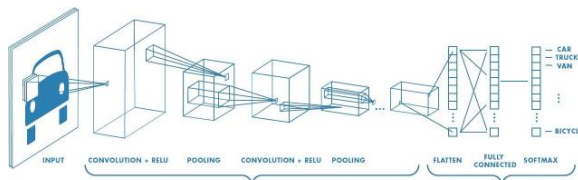


Gambar 1. Aplikasi Computer Vision di Pertanian

2.2. Convolutional Neural Network (CNN)

Convolutional Neural Network adalah jenis jaringan saraf tiruan yang dirancang khusus untuk pengolahan dan analisis data berbentuk grid, seperti gambar [7].

CNN menggunakan serangkaian lapisan konvolusi dan pooling untuk memproses gambar atau video. Lapisan konvolusi mengekstrak ciri-ciri dari masukan dengan cara menyilangkan filter kecil di atas gambar atau video, lalu menghitung perkalian titik antara filter dan masukan. Setelah itu, lapisan pooling mengambil sampel output dari lapisan konvolusi untuk mengurangi dimensi data dan meningkatkan efisiensi komputasinya [8]



Gambar 2. Arsitektur CNN

2.3. Tensorflow

TensorFlow adalah sebuah library open source yang dikembangkan oleh tim Google Brain, digunakan untuk komputasi numerik dan machine learning berskala besar. TensorFlow mendukung banyak bahasa pemrograman seperti Python, R, Swift, dan JavaScript. Aplikasinya mencakup pengenalan suara, klasifikasi gambar, penemuan obat, dan generasi gambar serta bahasa [9].

2.4. Klasifikasi Citra

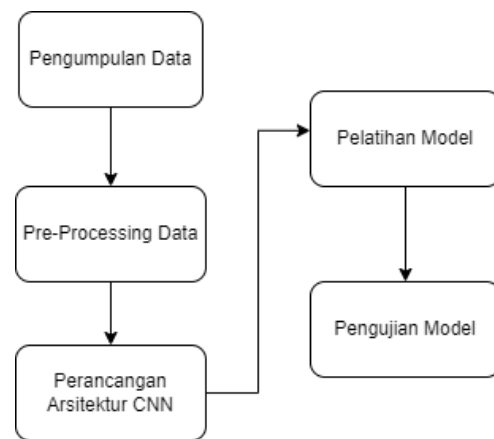
Klasifikasi citra adalah suatu metode dalam pengolahan citra yang bertujuan untuk mengelompokkan gambar ke dalam kategori atau label yang telah ditentukan sebelumnya [10].

Proses ini melibatkan identifikasi pola atau fitur visual yang membedakan antara kelas yang berbeda. Langkah-langkah utama dalam klasifikasi citra mencakup pengumpulan data gambar, pra-pemrosesan data untuk meningkatkan kualitas representasi, pembagian data menjadi pelatihan dan pengujian, serta

evaluasi kinerja model menggunakan data pengujian [11].

3. METODE PENELITIAN

Proses pengaplikasian model CNN untuk klasifikasi jenis buah segar dan busuk dilakukan dalam beberapa tahap, mulai dari pengumpulan data, preprocessing data, perancangan arsitektur CNN, pelatihan model, serta pengujian model. Tahap penelitian ditunjukkan pada Gambar 3.



Gambar 3. Tahap Penelitian

3.1. Pengumpulan Data

Penelitian ini menggunakan data sekunder yang diperoleh dari Kaggle. Dataset yang digunakan adalah "Fruits Fresh and Rotten for Classification" yang disusun oleh Sriram Reddy Kalluri pada tahun 2018. Dataset ini terdiri dari data latih dan data uji citra buah segar dan busuk, dengan total 13.619 citra yang mencakup tiga jenis buah yaitu apel, jeruk, dan pisang.

3.2. Pre-Processing Data

Pre-processing data dilakukan dengan mengubah nilai citra dari 0-255 menjadi 0-1 menggunakan proses normalisasi. Selanjutnya, dilakukan augmentasi data yang mencakup horizontal flip. Proses ini membantu meningkatkan variasi dalam data pelatihan.

3.3. Perancangan Arsitektur CNN

Perancangan dilakukan dengan membangun model Sequential yang terdiri dari beberapa lapisan Conv2D, MaxPooling2D, Dropout, dan Dense. Model ini dioptimalkan menggunakan optimizer Adam dengan loss function categorical_crossentropy.

3.4. Pelatihan Model

Pelatihan model dilakukan dengan memanfaatkan generator data yang telah di-augmentasi sebelumnya. Selama proses pelatihan, digunakan fungsi callback yang berfungsi untuk menghentikan pelatihan secara otomatis ketika akurasi yang diinginkan tercapai

3.5. Pengujian Model

Citra buah pada data uji digunakan untuk pengujian model. Hasil uji ditampilkan dalam bentuk grafik akurasi dan loss selama proses pelatihan dan validasi, yang memberikan gambaran mengenai kinerja model dalam mempelajari dan mengenali pola dari data. Pada tahap ini, mendapatkan tingkat keberhasilan dari metode yang sudah ditentukan, dimana akan menghitung nilai akurasi [12]

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

4. HASIL DAN PEMBAHASAN

Klasifikasi buah segar dan busuk menggunakan CNN dilakukan di Google Collaboratory menggunakan bahasa Python dengan beberapa library, salah satunya adalah TensorFlow yang berperan penting dalam membangun dan melatih model.

4.1. Normalisasi Citra

Mengatur augmentasi dan pemrosesan data citra, menskalakan nilai piksel untuk pelatihan dan validasi. Data generator memuat citra dari folder pelatihan dan validasi, mengubah ukurannya menjadi 150 x 150 piksel, dan mengelompokkan citra dalam batch (32 untuk pelatihan dan 8 untuk validasi) dengan mode kelas kategori.

```
train_datagen = ImageDataGenerator(
    rescale=1. / 255,
    horizontal_flip=True)

validation_datagen = ImageDataGenerator (rescale=1. / 255)

train_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=(150, 150),
    batch_size=32,
    class_mode='categorical')

validation_generator = validation_datagen.flow_from_directory(
    validation_dir,
    target_size=(150, 150),
    batch_size=8,
    class_mode='categorical')
```

Gambar 4. Pemrosesan Citra

Terdapat 10.911 citra data latih dan 2.708 citra data uji yang terbagi ke dalam 6 kelas yang sudah diterapkan skala dan ukurannya.

```
Found 10911 images belonging to 6 classes.
Found 2708 images belonging to 6 classes.
```

Gambar 5. Hasil Pemrosesan Citra

4.2. Model Arsitektur CNN

Pembuatan model arsitektur CNN pada penelitian ini terdiri dari beberapa lapisan Conv2D untuk mengekstraksi fitur gambar, diikuti oleh lapisan MaxPooling2D dan lapisan dropout untuk mengurangi overfitting. Fitur yang diekstraksi diubah menjadi vektor satu dimensi dan lapisan Dense untuk klasifikasi. Model menggunakan aktivasi softmax di lapisan output untuk probabilitas kelas output. Model

dikompilasi dengan optimizer Adam, loss function categorical_crossentropy, dan metrik akurasi.

```
model = tf.keras.models.Sequential([
    Conv2D(32, (3,3), padding='same', activation='relu', input_shape=(150, 150, 3)),
    MaxPooling2D(2,2),
    Dropout(0.25),
    Conv2D(64, (3,3), padding='same', activation='relu'),
    Conv2D(64, (3,3), padding='same', activation='relu'),
    MaxPooling2D(2,2),
    Dropout(0.25),
    Conv2D(128, (3,3), padding='same', activation='relu'),
    Conv2D(128, (3,3), padding='same', activation='relu'),
    MaxPooling2D(2,2),
    Dropout(0.25),
    Flatten(),
    Dense(512, activation='relu'),
    Dropout(0.5),
    Dense(6, activation='softmax'),
])

model.summary()
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

Gambar 6. Rancangan Arsitektur CNN

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 150, 150, 32)	896
max_pooling2d (MaxPooling2D)	(None, 75, 75, 32)	0
dropout (Dropout)	(None, 75, 75, 32)	0
conv2d_1 (Conv2D)	(None, 75, 75, 64)	18496
conv2d_2 (Conv2D)	(None, 75, 75, 64)	36928
max_pooling2d_1 (MaxPooling2D)	(None, 37, 37, 64)	0
dropout_1 (Dropout)	(None, 37, 37, 64)	0
conv2d_3 (Conv2D)	(None, 37, 37, 128)	73856
conv2d_4 (Conv2D)	(None, 37, 37, 128)	147584
max_pooling2d_2 (MaxPooling2D)	(None, 18, 18, 128)	0
dropout_2 (Dropout)	(None, 18, 18, 128)	0
flatten (Flatten)	(None, 41472)	0
dense (Dense)	(None, 512)	21234176
dropout_3 (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 6)	3078
Total params: 21515014 (82.07 MB)		
Trainable params: 21515014 (82.07 MB)		
Non-trainable params: 0 (0.00 Byte)		

Gambar 7. Hasil Rancangan Arsitektur CNN

4.3. Pelatihan Model

Mendefinisikan kelas callback untuk memantau akurasi selama pelatihan. Callback ini akan menghentikan pelatihan secara otomatis jika akurasi model melebihi ambang batas yang ditentukan (0.95 atau 95%) untuk menghindari overfitting dan menghemat sumber daya.

```
DESIRED_ACCURACY = 0.95

class myCallback(tf.keras.callbacks.Callback):
    def on_epoch_end(self, epoch, logs={}):
        if(logs.get('accuracy')>DESIRED_ACCURACY):
            self.model.stop_training = True

callbacks = myCallback()
```

Gambar 8. Batas Akurasi Pelatihan

Model CNN dilatih dengan menggunakan 2 generator, `train generator` untuk data pelatihan dan `validation generator` untuk data validasi. Generator ini memastikan data diproses dalam batch dan memungkinkan augmentasi data. Proses pelatihan terdiri dari 25 epoch dan model dilatih dan divalidasi dengan 150 langkah per epoch, namun akan berhenti secara otomatis jika akurasi mencapai 0.95 (95%).

```
history = model.fit(
    train_generator,
    steps_per_epoch=150,
    epochs=25,
    validation_data=validation_generator,
    validation_steps=150,
    shuffle=True,
    callbacks=[callbacks])
```

Gambar 9. Pelatihan Model

Selama pelatihan, model menunjukkan peningkatan bertahap dalam akurasi dan penurunan loss, menandakan perbaikan kinerja dari epoch ke epoch. Pada akhir pelatihan, model mencapai akurasi pelatihan sebesar 95.33% pada epoch ke-18. Ini menunjukkan bahwa model memiliki kemampuan yang sangat baik dalam mengklasifikasikan data baru.

```
epoch 1/25 [-----] - 1473s 18b/step - loss: 1.4817 - accuracy: 0.4188 - val_loss: 0.9348 - val_accuracy: 0.6092
epoch 2/25 [-----] - 1480s 18b/step - loss: 0.7338 - accuracy: 0.7283 - val_loss: 0.4696 - val_accuracy: 0.8383
epoch 3/25 [-----] - 1528s 18b/step - loss: 0.4778 - accuracy: 0.8342 - val_loss: 0.3228 - val_accuracy: 0.8980
epoch 4/25 [-----] - 1593s 18b/step - loss: 0.3998 - accuracy: 0.8583 - val_loss: 0.3248 - val_accuracy: 0.8825
epoch 5/25 [-----] - 1658s 18b/step - loss: 0.3226 - accuracy: 0.8821 - val_loss: 0.2845 - val_accuracy: 0.8925
epoch 6/25 [-----] - 1723s 18b/step - loss: 0.2982 - accuracy: 0.8958 - val_loss: 0.1928 - val_accuracy: 0.9358
epoch 7/25 [-----] - 1788s 18b/step - loss: 0.2515 - accuracy: 0.9119 - val_loss: 0.1991 - val_accuracy: 0.9325
epoch 8/25 [-----] - 1853s 18b/step - loss: 0.2088 - accuracy: 0.9223 - val_loss: 0.1689 - val_accuracy: 0.9480
epoch 9/25 [-----] - 1918s 18b/step - loss: 0.2547 - accuracy: 0.9135 - val_loss: 0.1708 - val_accuracy: 0.9380
epoch 10/25 [-----] - 1983s 18b/step - loss: 0.1525 - accuracy: 0.9315 - val_loss: 0.1149 - val_accuracy: 0.9558
epoch 11/25 [-----] - 2048s 18b/step - loss: 0.1087 - accuracy: 0.8935 - val_loss: 0.2088 - val_accuracy: 0.9325
epoch 12/25 [-----] - 2113s 18b/step - loss: 0.2142 - accuracy: 0.9221 - val_loss: 0.1548 - val_accuracy: 0.9475
epoch 13/25 [-----] - 2178s 18b/step - loss: 0.2117 - accuracy: 0.9275 - val_loss: 0.1796 - val_accuracy: 0.9313
epoch 14/25 [-----] - 2243s 18b/step - loss: 0.1788 - accuracy: 0.9367 - val_loss: 0.1676 - val_accuracy: 0.9467
epoch 15/25 [-----] - 2308s 18b/step - loss: 0.1888 - accuracy: 0.9438 - val_loss: 0.2005 - val_accuracy: 0.9317
epoch 16/25 [-----] - 2373s 18b/step - loss: 0.1878 - accuracy: 0.9358 - val_loss: 0.1564 - val_accuracy: 0.9392
epoch 17/25 [-----] - 2438s 18b/step - loss: 0.1470 - accuracy: 0.9487 - val_loss: 0.0838 - val_accuracy: 0.9725
epoch 18/25 [-----] - 2503s 18b/step - loss: 0.1311 - accuracy: 0.9533 - val_loss: 0.0982 - val_accuracy: 0.9667
```

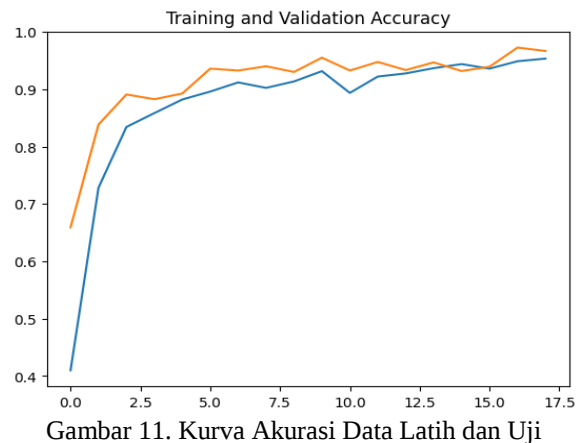
Gambar 10. Hasil Pelatihan Model

4.4. Pengujian Model

Pada proses pelatihan, model dilatih menggunakan data latih untuk mempelajari pola-pola yang ada dalam gambar. Setelah model selesai dilatih, langkah selanjutnya adalah menguji kinerja model menggunakan data uji yang belum pernah dilihat sebelumnya. Penerapan model CNN dalam klasifikasi buah segar dan busuk pada data uji mencapai akurasi sekitar 96.67%. Hasil ini menunjukkan bahwa model mampu menggeneralisasikan pola-pola yang telah dipelajari dari data pelatihan dengan baik ke data uji yang belum pernah dilihat sebelumnya.

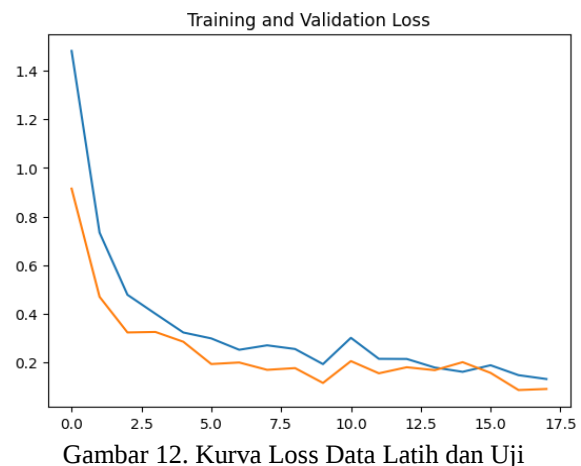
Gambar dibawah menunjukkan kurva akurasi pelatihan dan validasi model. Nilai akurasi bertahap mulai dari rendah menjadi tinggi. Garis akurasi pelatihan dan validasi yang hampir sejajar menunjukkan model tidak overfitting dan mampu menggeneralisasi data baru dengan baik, mendukung

akurasi 96.67% pada data uji dalam klasifikasi buah segar dan busuk.

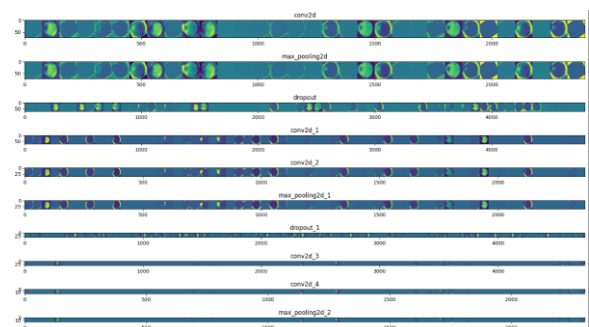


Gambar 11. Kurva Akurasi Data Latih dan Uji

Pada kurva di bawah menunjukkan tingkat loss pada pelatihan dan validasi data. Pada awalnya, loss sangat tinggi yang menandakan bahwa model masih belum mampu memprediksi dengan baik. Seiring bertambahnya epoch, loss menurun secara signifikan, menandakan bahwa model semakin baik dalam memahami pola-pola pada data. Penurunan loss ini menunjukkan bahwa model berhasil meminimalkan kesalahan prediksi. Selain itu, loss pada data pelatihan dan validasi tetap konsisten dan stabil pada nilai rendah, yang berarti model tidak overfitting dan mampu menggeneralisasi data baru dengan baik.



Gambar 12. Kurva Loss Data Latih dan Uji



Gambar 13. Hasil Visualisasi Lapisan CNN

Visualisasi pada gambar 13 menunjukkan bagaimana CNN belajar mengenali objek dalam gambar secara bertahap. Setiap baris dalam gambar mewakili satu lapisan dalam CNN. Lapisan-lapisan ini secara berturut-turut mengekstrak fitur-fitur yang semakin kompleks, mulai dari tepi sederhana hingga bentuk objek yang lebih abstrak.

Berikut adalah ringkasan hasil dari citra yang sudah diklasifikasi. Pada setiap citra mempunyai keterangan mengenai label yang benar dan label prediksi yang diklasifikasikan oleh model.



Gambar 14. Hasil Klasifikasi Citra

5. KESIMPULAN DAN SARAN

Berdasarkan penelitian dan analisis yang telah dilakukan, dapat disimpulkan bahwa klasifikasi buah segar dan busuk menggunakan model CNN yang dilatih dengan 10.911 citra data latih dan 2.708 citra data uji menghasilkan akurasi pelatihan sekitar 95,33% dan akurasi validasi sebesar 96,67%, dengan loss validasi sebesar 9,02%. Hal ini menunjukkan bahwa model CNN yang dikembangkan sangat efektif dalam mengidentifikasi dan membedakan buah segar dari busuk. Dengan akurasi tinggi dan loss yang relatif rendah, model CNN terbukti mampu meningkatkan kinerja selama pelatihan dan mempertahankan kinerja yang baik pada data validasi, ini menjadi solusi efisien dan efektif untuk klasifikasi objek. Saran untuk pengembangan selanjutnya adalah mengurangi tingkat loss pada data latih dan uji serta meningkatkan desain arsitektur model CNN untuk hasil yang lebih optimal.

DAFTAR PUSTAKA

- [1] "Meningkatkan Kualitas Hasil Panen: Kunci Meningkatkan Nilai Jual dan Daya Saing Produk Pertanian," panda, Jan 2024. [Online]. Available: <https://www.panda.id/meningkatkan-kualitas-hasil-panen-meningkatkan-nilai-jual-dan-daya-saing-produk-pertanian/>. [Accessed: May 27, 2024].
- [2] L. Magwaza dan S. Tesfay, "Tinjauan Metode Destruktif dan Non-destruktif dalam Penentuan Kematangan Buah Alpukat," *Teknologi Pangan dan Bioproses*, vol. 8, no. 10, Sep. 2015, doi: 10.1007/s11947-015-1568-y.
- [3] C. Juliandy and Darwin, "Pengenalan Computer Vision Berbasis Convolutional Neural Network melalui DewataTalks," **Sarwahita: Jurnal Pengabdian Kepada Masyarakat**, vol. 21, no. 1, 30 Apr. 2024. DOI: <https://doi.org/10.21009/sarwahita.211.4>
- [4] A. H. Makarim, "Menggunakan Convolutional Neural Network (CNN) untuk Klasifikasi Gambar Produk," *Medium*, May, 2024. [Online]. Available: <https://medium.com/@21611129/menggunakan-convolutional-neural-network-cnn-untuk-klasifikasi-gambar-produk-c9f75c2446cb>. [Accessed: May 27, 2024].
- [5] A. Herlangga, R. R. Karim, dan M. K. Nurwijaya, "Penerapan Transfer Learning EfficientNetB3 untuk Pengenalan Senjata Tradisional Sumatera Barat Menggunakan Convolutional Neural Network (CNN)," *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, vol. 12, no. 2, 2024. doi: 10.23960/jitet.v12i2.4256.
- [6] A. P. Hadi, "COMPUTER VISION ? PENGERTIAN, CONTOH, DAN APLIKASINYA," Universitas STEKOM, 2023. [Online]. Available: <https://komputer-grafis-d3.stekom.ac.id/index.php/informasi/baca/COMPUTER-VISION-PENGERTIAN-CONTOH-DAN-APLIKASINYA/5dbdf66075ce5e47cb39584288bc8b7dc3333b1b>. [Accessed: May 28 2024].
- [7] M. M. Anwar, "Jaringan Saraf Tiruan: Kekuatan Dari Kecerdasan Buatan," *HMPs Fisika*, 2023. [Online]. Available: <https://fisika.uin-malang.ac.id/hmj/index.php/2023/07/04/jaringan-saraf-tiruan-kekuatan-dari-kecerdasan-buatan/> [Accessed: May 28 2024].
- [8] GeekforGeeks, "Introduction to Convolution Neural Network," Mar, 2024. [Online]. Available: <https://geeksforgeeks.org/introduction-convolution-neural-network/>. [Accessed: May 28, 2024].
- [9] P. Arfienda, "MATERI PENDAMPING MEMAHAMI CONVNET DENGAN TENSORFLOW," *Algoritma Data Science School*, 2019. [Online]. Available: <https://algoritma.blog/convolutional-neural-networks-tensorflow/> [Accessed: May 29 2024].
- [10] Saprudina, R. Amaliaa, dan P. Rosyana, "Klasifikasi Citra Menggunakan Metode Random Forest dan Sequential Minimal Optimization (SMO)," *JUSTIN (Jurnal Sistem dan Teknologi Informasi)*, vol. 09, no. 2, 2021. doi: 10.26418/justin.v9i2.44120.
- [11] A. Pamungkas, "Penerapan Convolutional Neural Network untuk Klasifikasi Citra," *Pemrograman MATLAB*, Juli 2022, <https://pemrogramanmatlab.com/2023/07/22/penerapan-convolutional-neural-network-untuk-klasifikasi-citra/>.
- [12] F. Santoso, dan E. Hartati, "Penggunaan Algoritma Random Forest Dalam Klasifikasi Buah Segar dan Busuk," *Jurnal Algoritme*, Vol.3 No.1, 2022