

## Klasifikasi Buah dan Sayuran Segar atau Busuk Menggunakan *Convolutional Neural Network*

Eka Aenun Nisa Munfaati <sup>(1)\*</sup>, Arita Witanti <sup>(2)</sup>

Informatika, Fakultas Teknologi Informasi, Universitas Mercu Buana, Yogyakarta  
e-mail : aennunisaeka@gmail.com, arita@mercubuana-yogya.ac.id.

\* Penulis korespondensi.

Artikel ini diajukan 25 Juli 2023, direvisi 19 Desember 2023, diterima 19 Desember 2023, dan dipublikasikan 25 Januari 2024.

### Abstract

Fresh fruits and vegetables contain many nutrients, such as minerals, vitamins, antioxidants, and beneficial fiber, superior to those found in rotten or almost rotten produce. On the other hand, fruits and vegetables that are nearly spoiled or already rotten have significantly lost their nutritional value. Rotten produce also harbors bacteria and fungi that can lead to infections and food poisoning when consumed. Convolutional Neural Network (CNN) offers a programmable solution for classifying fresh and rotten fruits and vegetables. Image processing using the TensorFlow library is employed in this classification process. During testing on the training data, the CNN achieved an accuracy of 90.42%. In comparison, the validation accuracy reached 94.21% when using the SGD optimizer, 20 epochs, a batch size 16, and a learning rate of 0.01. For the testing data, the accuracy obtained was 80.83%.

**Keywords:** Convolutional Neural Network, Tensorflow, Classification, Fruits, Vegetables

### Abstrak

Buah dan sayuran segar mengandung banyak nutrisi seperti mineral, vitamin, antioksidan, serta serat yang baik bagi tubuh dibandingkan buah dan sayuran busuk. Di sisi lain, buah dan sayuran yang hampir busuk atau yang sudah busuk telah kehilangan sebagian besar nilai gizi yang dimiliki. Buah dan sayuran yang busuk mengandung bakteri dan jamur yang dapat menyebabkan infeksi dan keracunan makanan apabila dikonsumsi oleh tubuh. Convolutional Neural Network (CNN) dapat menjadi solusi untuk mengklasifikasikan buah dan sayuran yang segar dengan busuk secara terprogram. Proses pengolahan citra dengan CNN juga menggunakan library Tensorflow. Dalam proses klasifikasi ini menggunakan data citra berjumlah 12.220, hasil pengujian pada data *training* menghasilkan akurasi sebesar 90,42% dan akurasi validasi sebesar 94,21% dengan menggunakan optimizer SGD (*Stochastic Gradient Descent*), *epochs* 20, 16 *batch size*, dan *learning rate* 0,01. Sedangkan untuk data *testing* memperoleh akurasi sebesar 80,83%.

**Kata Kunci:** Convolutional Neural Network, Tensorflow, Klasifikasi, Buah, Sayuran

## 1. PENDAHULUAN

Buah dan sayuran dapat menjadi sumber nutrisi yang berguna dalam menjaga kesehatan tubuh manusia dikarenakan kandungan yang dimiliki oleh buah dan sayuran sangat banyak seperti mineral, antioksidan, vitamin, dan serat yang tinggi. Buah dan sayuran sangat bermanfaat bagi tubuh dalam menjaga kepadatan tulang dan jantung. Selain itu, buah dan sayuran juga dapat meminimalisir risiko penyakit *stroke*, jantung koroner, kanker, hipertensi, dan diabetes. *Phytoingredients* yang terkandung dalam buah dan sayuran berfungsi untuk mencegah terjadinya *stress* oksidatif (Rarastiti, 2022). Dalam penelitian Jafaruddin (2023), mengonsumsi buah dan sayuran segar dapat meningkatkan daya ingat. Hal ini dikarenakan buah dan sayuran segar mengandung zat antioksidan yang berguna untuk melindungi sel-sel pada otak. Kandungan antioksidan dalam buah dan sayuran segar juga dapat mengobati penyakit kanker dan jenis penyakit lainnya.



Dalam *The World Health Organization* (WHO) oleh Magalhães et al. (2022) menganjurkan untuk mengonsumsi setidaknya 400 gram buah dan sayuran segar. Mengonsumsi buah dan sayuran segar mengandung gizi yang sangat tinggi sehingga dalam menjaga kesehatan tubuh, sedangkan mengonsumsi buah dan sayuran busuk sangat berbahaya karena dapat menyebabkan keracunan. Meskipun demikian, dalam penelitian oleh Gregori et al. (2019) mengatakan bahwa minat akan buah dan sayuran pada tingkat dunia masih terbilang rendah jika didasarkan pada anjuran WHO. Pada tahun 2007, rata-rata konsumsi harian per kapita buah dan sayuran di Amerika Serikat (USA) adalah 325 g dan 505 g (830 g); Jerman 624 g dan 106 g (730 g); Finlandia 429 g dan 130 g (559 g). Angka-angka ini menunjukkan adanya kesenjangan antara anjuran WHO dengan minat konsumsi buah dan sayuran di berbagai negara.

Upaya dalam meningkatkan konsumsi buah dan sayuran setiap harinya dapat dimulai dengan meningkatkan kualitas buah dan sayuran yang segar. Salah satu upaya dalam meningkatkan kualitas buah dan sayuran segar adalah dengan memisahkan antara buah dan sayuran segar dengan buah dan sayuran busuk. Buah dan sayuran segar yang tercampur dengan buah dan sayuran busuk dapat mengakibatkan buah dan sayuran segar menjadi busuk karena terkontaminasi oleh buah dan sayuran busuk. Untuk memisahkan buah dan sayuran segar dengan busuk dapat dilakukan secara otomatis dengan memanfaatkan teknologi masa kini. Perkembangan teknologi memberikan kemudahan dalam melakukan pemisahan terhadap buah dan sayuran segar atau busuk dengan lebih cepat dibandingkan dilakukan secara manual.

*Convolutional Neural Network* (CNN) termasuk algoritma *deep learning* yang dapat digunakan dalam klasifikasi objek Iswantoro & UN (2022). Dengan menggunakan teknologi masa kini maka dapat melakukan proses pengolahan gambar untuk diklasifikasikan dengan hanya menggunakan fitur penangkap gambar berupa kamera atau galeri dari sistem. Menurut Sya'ban et al. (2022) dalam penelitiannya mengatakan bahwa CNN bekerja sama halnya dengan cara kerja *neuron* yang terdapat pada otak manusia berdasarkan persepsi visual.

Pengembangan model CNN untuk klasifikasi buah segar atau busuk pernah dilakukan oleh Sya'ban et al. (2022) dengan cara melakukan perancangan terhadap arsitektur CNN dengan tujuan agar model mampu mengklasifikasikan jenis buah beserta tingkat kesegarannya. Penelitian ini terdiri dari buah apel, pisang, dan jeruk mandarin dari yang segar hingga busuk di mana *dataset* dalam penelitian ini diperoleh dari Kaggle. *Dataset* dalam penelitian ini terdiri dari 10,900 citra buah yang terbagi menjadi 3 data di antaranya 8.720 pada data *train*, 2.180 pada data validasi, dan 1.090 pada data *testing*. Hasil riset memperoleh akurasi *training* mencapai 98,20%, 99,22% pada akurasi validasi, dan 91,65% pada akurasi *testing* dengan menggunakan 50 *epochs*.

Dalam penelitian "Klasifikasi *Chest X-Ray Images* Berdasarkan Kriteria Gejala Covid-19 Menggunakan *Convolutional Neural Network*" yang dilakukan oleh Ayumi & Nurhaida (2021) mengusulkan implementasi metode CNN dengan menggunakan 2 lapisan *layer convolution*, *maxpooling*, dan *fully connected layer* untuk melakukan klasifikasi terhadap citra *chest X-ray* pada pasien yang diduga terkena Covid-19. Penelitian ini menggunakan *dataset Covid-19 Radiography Database* yang terdiri dari 1.200 citra pada *class* kasus positif Covid-19 dan 1.341 citra pada *class* normal. Riset ini memperoleh hasil akurasi *training* mencapai 94,05% dengan *loss* 25,58% dan akurasi validasi mencapai 96,06% dengan *loss* 14,71%.

Penelitian oleh Hawari et al. (2022) membahas mengenai implementasi metode CNN dalam melakukan klasifikasi terhadap 4 jenis tanaman padi di antaranya yaitu: *Leaf Brown*, *Brown Spot*, *Daun Sehat*, dan *Hawar*. Dalam risetnya, ia melakukan *training* model dengan menggunakan 10 *epochs* sehingga memperoleh akurasi *training* sebesar 85% dengan *loss* mencapai 38% dan akurasi validasi sebesar 95% dengan *loss* 30%.

Penelitian berikutnya oleh Qotrunnada et al. (2022) tentang klasifikasi terhadap wajah bermasker atau tidak bermasker dengan *convolutional neural network*. Penelitian yang ia lakukan



menggunakan citra wajah dengan ukuran 150x150 piksel. Pada penelitiannya ia memperoleh akurasi sebesar 82,35% pada *validation* dan 98,20% pada *training*.

Berdasarkan penelitian di atas, dapat disimpulkan bahwa CNN mampu melakukan klasifikasi terhadap suatu *object* dengan tingkat akurasi yang tinggi dan mudah digunakan. Sehingga dalam penelitian ini penulis memutuskan untuk menggunakan metode CNN dalam melakukan klasifikasi terhadap buah segar dan busuk. Dengan digunakannya metode ini diharapkan dapat menghasilkan akurasi yang baik.

## 2. METODE PENELITIAN

### 2.1 Konsep Dasar

#### 2.1.1 Convolutional Neural Network (CNN)

*Convolutional Neural Network* (CNN) yaitu *deep learning* untuk mengidentifikasi objek yang terdapat pada sebuah gambar. Hasil identifikasi objek tersebut dimanfaatkan untuk menganalisa pada citra yang ada (Iswantoro & UN, 2022). Dalam penelitian Mulyanto et al. (2021) mengatakan bahwasannya CNN dapat digunakan untuk mengklasifikasikan citra atau gambar dengan menggunakan pendekatan *deep learning*. *Pooling*, *convolution*, dan *fully connected* merupakan *layer* pada CNN. Selain itu juga terdapat *feature learning* dan *classification*.

*Feature learning* yaitu suatu proses yang melakukan “*encoding*” pada citra menjadi sebuah fitur yang berbentuk numerik dengan tujuan untuk menampilkan citra di dalamnya (Mahaputri et al., 2022). Menurut Wulandari et al. (2020) *feature learning* termasuk bagian dari arsitektur *Convolutional Neural Network* yang berperan dalam mengkonversi matriks input menjadi *feature maps*. *Convolutional layer* dan *pooling layer* merupakan *layer* yang dimiliki *feature learning*.

*Classification* bertujuan untuk mengidentifikasi *neuron* yang diekstrak yang terdapat pada *feature learning*. Dalam *classification* terdapat *flatten* dan *fully connected layer*. Menurut Mahaputri et al. (2022) *flatten* dapat melakukan konversi antara *feature map* yang berbentuk *multidimensional array* menjadi vektor.

#### 2.1.2 Confusion Matrix

*Confusion matrix* yaitu suatu proses untuk mempresentasikan dan mencocokkan antara nilai sebenarnya dengan nilai perkiraan pada model untuk mengevaluasi nilai akurasi, *precision*, *recall*, dan *F1-score* (Mahaputri et al., 2022). Istilah-istilah yang digunakan dalam *confusion matrix* di antaranya yaitu: *False Negative* (FN), *True Negative* (TN), *False Positive* (FP), dan *True Positive* (TP).

#### 2.1.3 Python dan Tensorflow

Bahasa pemrograman Python termasuk jenis bahasa pemrograman tingkat tinggi dikarenakan memiliki kemampuan yang lebih tinggi dalam abstraksi dan pemrograman yang mudah dipahami oleh manusia dibandingkan bahasa lainnya. Python memiliki *library* yang lengkap dan bersifat *open source*. Oleh sebab itu bahasa pemrograman Python sangat cocok jika digunakan dalam pengembangan *deep learning* dan juga *machine learning* (Alfarizi et al., 2023).

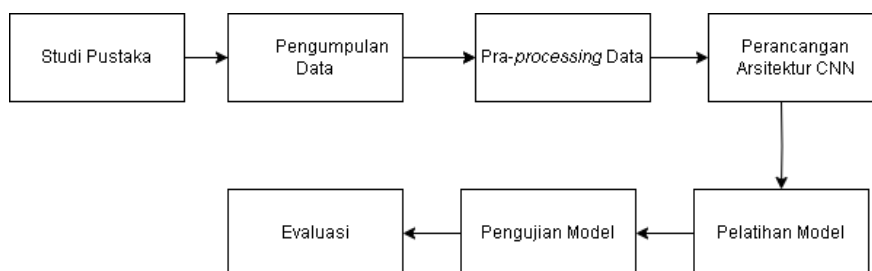
Tensorflow merupakan sebuah *open-source library* untuk mengembangkan dan melatih model *machine learning* atau *deep learning*. Tensorflow mengintegrasikan model algoritma *deep learning* dan jaringan saraf ke dalam satu set *library* untuk melakukan tugas-tugas di bidang pembelajaran mesin dan jaringan saraf (Apendi et al., 2023).

## 2.2 Tahap Penelitian

Proses pengembangan model CNN untuk klasifikasi buah dan sayuran segar atau busuk dilakukan melalui 6 tahap. Keenam tahapan tersebut adalah studi pustaka, pengumpulan data,



*preprocessing* data, perancangan arsitektur CNN, pelatihan model, serta pengujian dan evaluasi model. Alur penelitian ditunjukkan seperti pada Gambar 1.



**Gambar 1 Tahap Penelitian**

### 2.2.1 Studi Pustaka

Tahapan awal yang dilakukan dalam penelitian ini adalah melakukan studi pustaka. Studi pustaka dilakukan dengan mempelajari berbagai buku dan jurnal ilmiah. Hal ini dilakukan untuk mempelajari algoritma klasifikasi gambar atau objek dengan menggunakan *convolutional neural network*.

### 2.2.2 Pengumpulan Data

**Tabel 1 Data Primer Buah dan Sayuran**

No.	Class	Data Testing
1	Apel Segar	10
2	Apel Busuk	10
3	Pisang Segar	10
4	Pisang Busuk	10
5	Timun Segar	10
6	Timun Busuk	10
7	Tomat Segar	10
8	Tomat Busuk	10
9	Kentang Segar	10
10	Kentang Busuk	10
11	Okra Segar	10
12	Okra Busuk	10
<b>Total</b>		<b>120</b>

**Tabel 2 Data Sekunder Buah dan Sayuran**

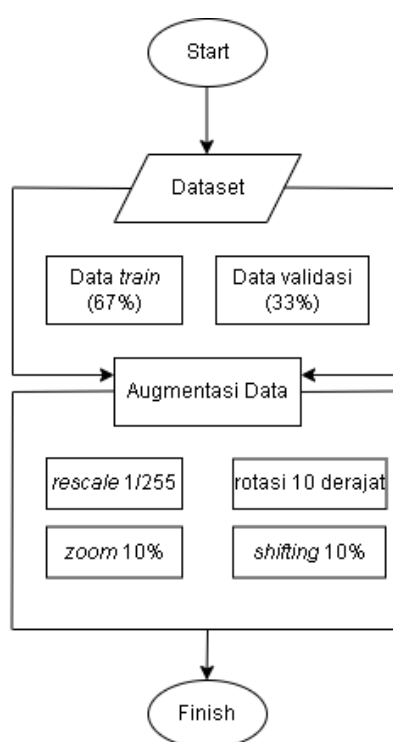
No.	Class	Data Train	Data Validasi
1	Apel Segar	731	396
2	Apel Busuk	906	387
3	Pisang Segar	887	511
4	Pisang Busuk	708	370
5	Timun Segar	496	279
6	Timun Busuk	421	255
7	Tomat Segar	877	255
8	Tomat Busuk	843	353
9	Kentang Segar	536	270
10	Kentang Busuk	802	370
11	Okra Segar	635	370
12	Okra Busuk	338	224
<b>Total</b>		<b>8180</b>	<b>4040</b>



Penelitian ini menggunakan data primer dan data sekunder. Data primer diperoleh dari kamera peneliti dan Google Photo, sedangkan data sekunder menggunakan *dataset Vegetable and Fruits Fresh and Stale* pada Kaggle (Baloch & Khaliq, 2022). Data primer dan sekunder terdiri dari buah dan sayuran segar hingga busuk dengan total citra 120 citra dan 12.220 citra. Data primer dan data sekunder seperti pada Tabel 1 dan Tabel 2. Data primer digunakan dalam pengujian atau evaluasi model CNN, sedangkan data sekunder digunakan untuk tahap pelatihan model CNN.

### 2.2.3 Preprocessing Data

*Pre-processing* data dilakukan dengan menggunakan proses *thresholding*, yaitu mengubah nilai citra dari 0-255 menjadi 0-1. Selanjutnya dilakukan augmentasi data yang terdiri dari melakukan rotasi pada gambar sebesar 10 derajat, *zoom* pada gambar sebesar 10%, dan mengatur *shifting* sebesar 10%. Tahapan *preprocessing* seperti pada Gambar 2.



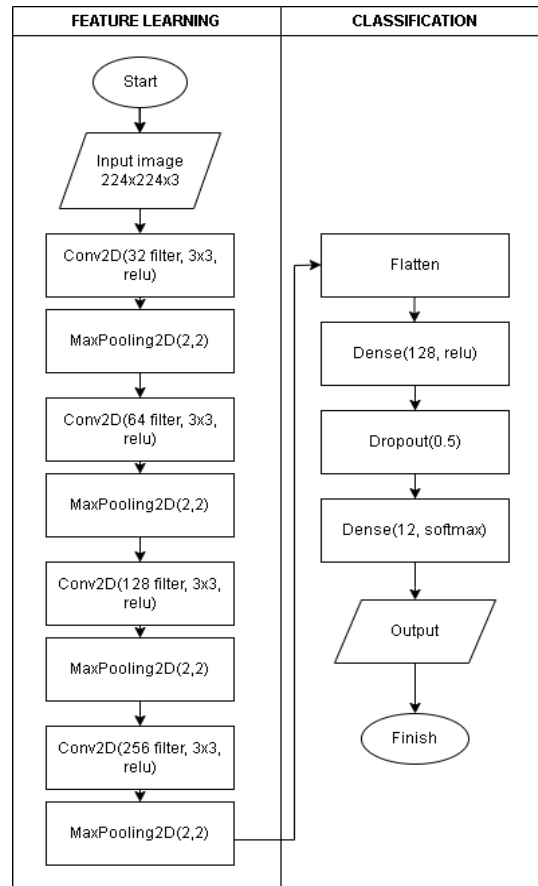
Gambar 2 Tahap *Preprocessing*

### 2.2.4 Perancangan Arsitektur CNN

Dalam tahap perancangan arsitektur CNN, citra pada *dataset* di-*resize* menjadi 224x224x3 piksel. Perancangan arsitektur CNN terdiri dari *feature learning* dan *classification*. Pada tahap *feature learning*, terdapat proses *convolution* dan *pooling* dalam citra *input*. *Convolution* pertama menggunakan 32 filter dan 3x3 kernel matriks dengan aktivasi ReLu. Setelah itu, terdapat proses *Max pooling2D* dengan ukuran filter 2x2. Setelah itu, dilakukan proses konvolusi kedua dengan 64 filter dan 3x3 kernel matriks dengan aktivasi ReLu. Kemudian, dilakukan kembali proses *Max pooling2D* dengan ukuran filter 2x2 dan dilanjutkan dengan proses konvolusi ketiga dan keempat, masing-masing dengan jumlah filter 128 dan 256, kernel matriks 3x3 dengan aktivasi ReLu serta *Max pooling2D* dengan ukuran filter 2x2. Setelah tahap *feature learning* selesai, kemudian akan dilakukan proses *flatten* untuk mengubah *output* dari proses *convolution* yang berupa *matrix* menjadi vektor dan kemudian dilanjutkan dengan tahap *classification*.



Pada tahap *classification*, lapisan *dense* pertama terdiri dari 128 *neuron* dengan aktivasi ReLu dan dilanjutkan dengan teknik *dropout* sebesar 0,5. Lapisan *dense* kedua terdiri dari 12 *neuron* dengan aktivasi *softmax* tanpa menggunakan teknik *dropout*. Arsitektur CNN pada penelitian ini seperti pada Gambar 3.



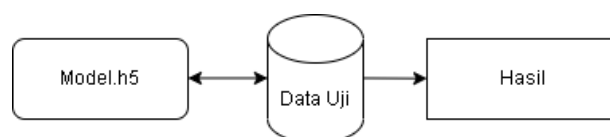
Gambar 3 Arsitektur CNN

### 2.2.5 Pelatihan Model

Prose pelatihan model dilakukan dengan data sekunder yang telah dilakukan teknik *pre-processing*. Pada pelatihan model dilakukan scenario uji coba terhadap 4 *hyperparameter* yaitu: *optimizer* (ADAM, SGD, dan RMSprop), *learning rate* (0,01, 0,001, dan 0,0001), *batch size* (16, 32, dan 64), dan *epochs* (10, 15, dan 20). Hal ini bertujuan untuk memperoleh hasil akhir yang optimal pada model CNN.

### 2.2.6 Pengujian dan Evaluasi Model

Citra buah dan sayuran pada data uji digunakan untuk proses pengujian dan evaluasi model. Hasil dari evaluasi model akan direpresentasikan dalam *confusion matrix* untuk memberikan gambaran yang lebih komprehensif tentang performa model. Tahap evaluasi model seperti pada Gambar 4.



Gambar 4 Tahap Evaluasi Model CNN



### 3. HASIL DAN PEMBAHASAN

#### 3.1 Skenario Uji Coba

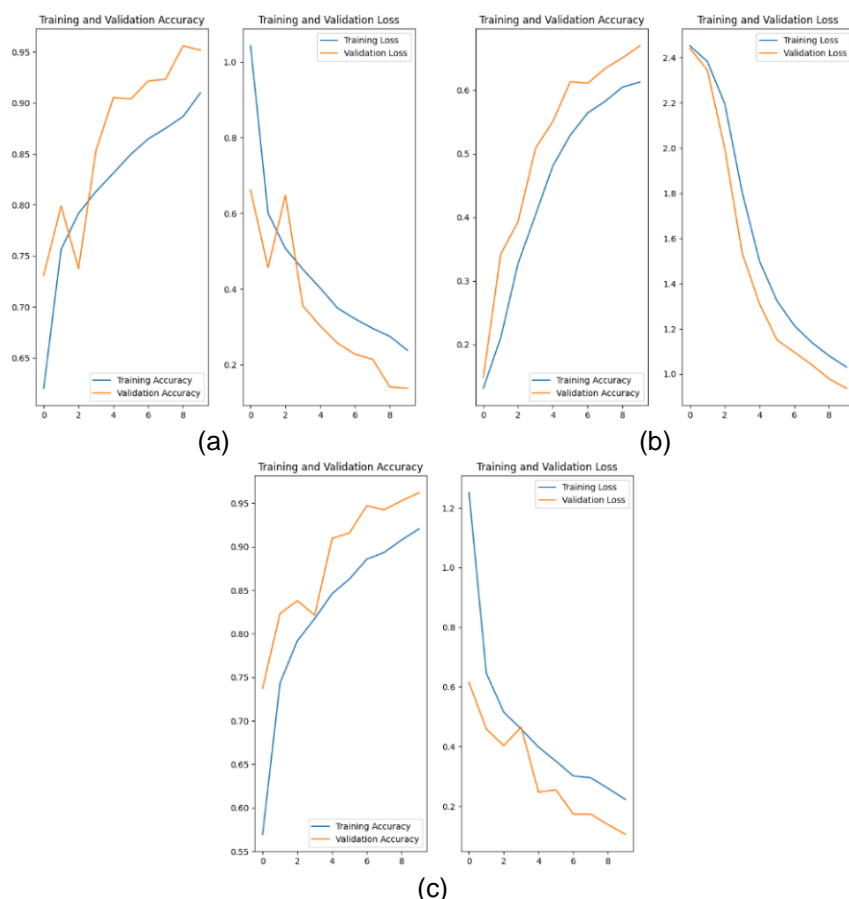
Dilakukan skenario uji coba terhadap *hyperparameter* untuk melihat performa model terbaik pada setiap skenario uji coba. Skenario uji coba terdiri dari skenario *optimizer* (ADAM, SGD, dan RMSprop), *learning rate* (0,01, 0,001, dan 0,0001), *batch size* (16, 32, dan 64), dan *epochs* (10, 15, dan 20). Setiap citra input akan di-*resize* menjadi 224x224x3 piksel. Hasil dari skenario uji coba adalah sebagai berikut.

##### 3.1.1 Skenario Uji Coba *Optimizer*

Skenario pertama yaitu melakukan pengujian terhadap 3 jenis *optimizer* yaitu Adam, SGD, dan RMSprop. Pada pengujian *optimizer hyperparameter* seperti *epochs* diatur dengan jumlah 10 *epochs*, 32 *batch size*, dan 0,001 *learning rate*. Hasil dari skenario *optimizer* seperti pada Tabel 3.

Tabel 3 Perbandingan Akurasi dan Loss pada *Optimizer*

<i>Optimizer</i>	Validation Acc	Loss
Adam	0,9559	0,1413
SGD	0,6696	0,9368
RMSprop	0,9619	0,1053



Gambar 5 Grafik Pengujian Optimizer (a) Adam, (b) SGD, dan (c) RMSProp





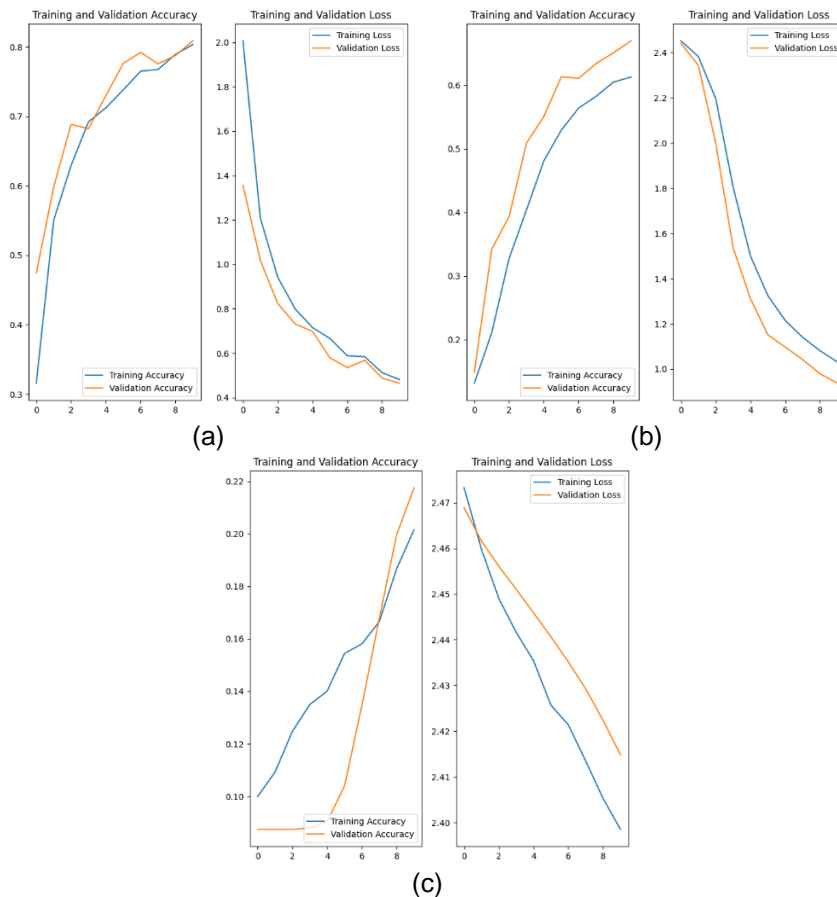
Berdasarkan Tabel 3, *optimizer* RMSprop memperoleh akurasi tertinggi disertai *loss* terendah. Sedangkan untuk *optimizer* Adam memperoleh akurasi dan *loss* sedikit lebih rendah dibandingkan dengan *optimizer* RMSprop. Namun, hasil terbaik yang dipilih untuk skenario *optimizer* adalah SGD, karena pada setiap iterasi (*epochs*) mengalami peningkatan nilai akurasi. Sedangkan *optimizer* Adam dan RMSprop masih muncul beberapa *spikes* atau lonjakan seperti pada Gambar 5.

### 3.1.2 Skenario Uji Coba *Learning Rate*

Skenario kedua yaitu pengujian terhadap pengaruh besaran nilai *learning rate*. Nilai *learning rate* yang digunakan terdiri dari nilai 0,01, 0,001, dan 0,0001. Sedangkan untuk *hyperparameter* lainnya seperti *optimizer* menggunakan RMSprop, 10 epochs, dan 32 *batch size*. Hasil skenario uji coba *learning rate* seperti pada Tabel 4.

**Tabel 4 Perbandingan Akurasi dan Loss pada *Learning Rate***

<i>Learning Rate</i>	Validation Acc	Loss
0,01	0,8089	0,4644
0,001	0,6696	0,9368
0,0001	0,2176	2.4148



**Gambar 6 Grafik Pengujian *Learning Rate* (a) 0,01, (b) 0,001, dan (c) 0,0001**

Berdasarkan Tabel 4, *learning rate* dengan nilai yang besar memperoleh akurasi yang sangat rendah disertai *loss* yang sangat tinggi. Hal ini disebabkan semakin besar nilai *learning rate* maka akan sangat berdampak pada ketelitian *neural network* pada saat proses pelatihan model dikarenakan model mengalami proses pelatihan yang terbilang lebih cepat. Sedangkan, semakin





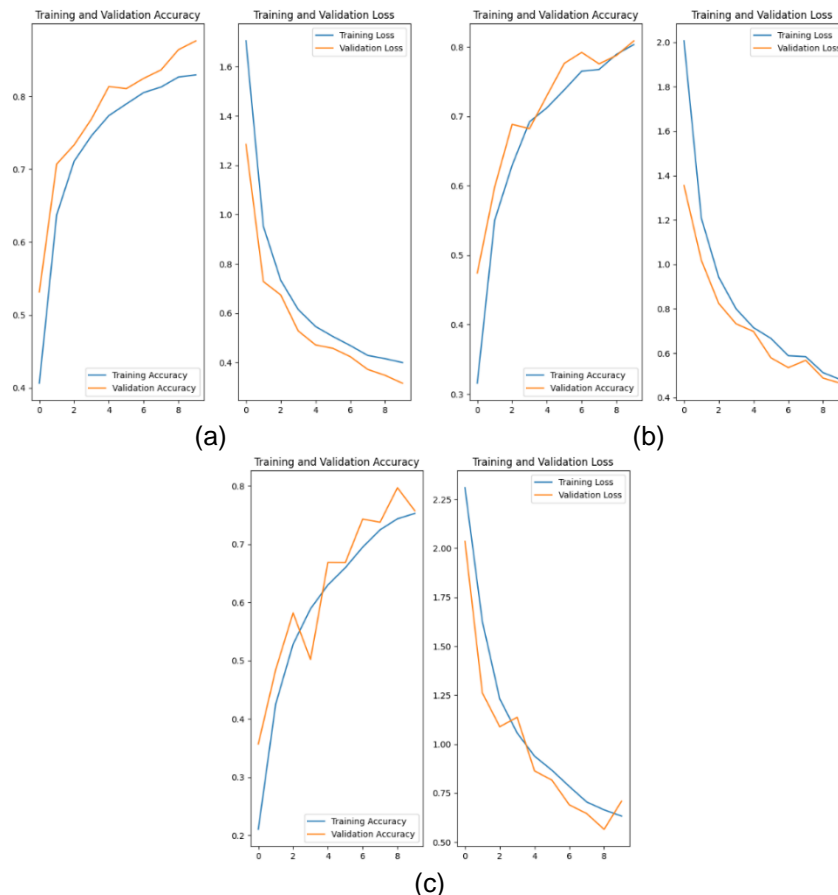
kecil nilai *learning rate* mengakibatkan *neural network* memiliki tingkat ketelitian yang cenderung lebih tinggi namun membutuhkan waktu *running* yang cenderung lebih lama. Sehingga pada skenario ini, nilai *learning rate* terbaik adalah 0,01. Grafik pengujian terhadap pengaruh nilai *learning rate* dapat dilihat pada Gambar 6.

### 3.1.3 Skenario Uji Coba *Batch Size*

Skenario ketiga yaitu pengujian terhadap pengaruh besaran *batch size*. Nilai *batch size* yang sesuai akan menghasilkan model dengan performa yang baik. Skenario *batch size* dilakukan dengan menggunakan besaran *batch size* di antaranya yaitu 16, 32, dan 64. *Hyperparameter* lainnya menggunakan *hyperparameter* terbaik yang diperoleh pada tahap sebelumnya yaitu *optimizer* RMSprop, 0,01 *learning rate*, dan 10 *epochs*. Hasil skenario *batch size* dapat dilihat pada Tabel 5.

Tabel 5 Perbandingan Akurasi dan Loss Terhadap Penggunaan Nilai *Batch Size*

<i>Batch Size</i>	Validation Acc	Loss
16	0,8762	0,3162
32	0,8089	0,4644
64	0,7968	0,5651



Gambar 7 Grafik Pengujian *Batch Size* (a) 16, (b) 32, dan (c) 64

Berdasarkan Tabel 5, besaran *batch size* yang lebih kecil memperoleh akurasi yang cenderung lebih tinggi dibanding *batch size* yang lebih tinggi. *Batch size* yang lebih kecil memperoleh *loss* terendah dibandingkan dengan besaran *batch size* lainnya. Grafik pengujian terhadap pengaruh nilai *batch size* seperti pada Gambar 7.

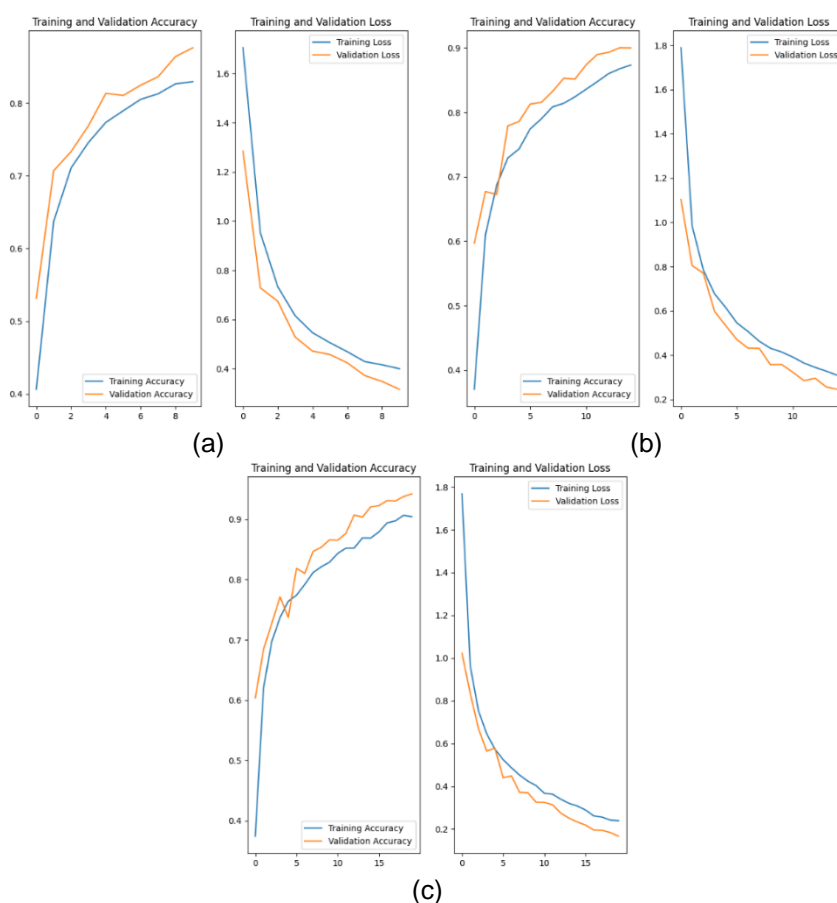


### 3.1.4 Skenario Uji Coba Epochs

Skenario keempat yaitu pengujian terhadap penggunaan jumlah *epochs* pada saat pelatihan model dengan tujuan untuk memperoleh jumlah *epochs* yang optimal sehingga menghasilkan akurasi terbaik. Pengujian dilakukan dengan menggunakan beberapa jumlah *epoch* antara lain 10, 15, dan 20 *epochs*. *Hyperparameter optimizer* menggunakan RMSprop, 0,01 *learning rate*, dan 16 *batch size*. Hasil skenario *epochs* seperti pada Tabel 6.

Tabel 6 Perbandingan Akurasi dan Loss pada Epochs

Epochs	Validation Acc	Loss
10	0,8762	0,3162
15	0,8998	0,2454
20	0,9421	0,1661



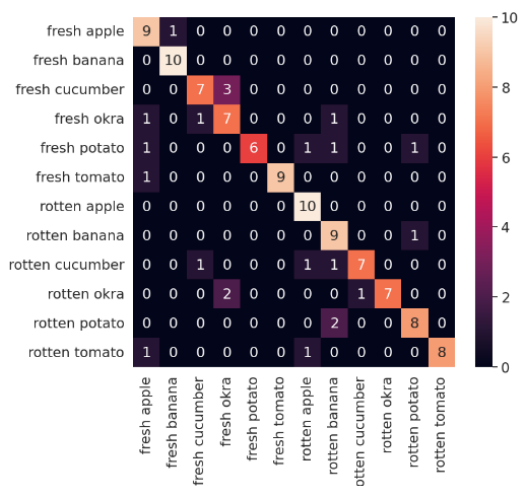
Gambar 8 Grafik Pengujian Jumlah Epochs (a) 10, (b) 15, dan (c) 20

Berdasarkan Tabel 6, bertambahnya jumlah *epochs* sangat berpengaruh terhadap tingkat akurasi dan *loss* yang diperoleh. *Epochs* dengan jumlah yang lebih banyak memperoleh akurasi yang semakin tinggi dan *loss* yang lebih rendah. Jumlah *epochs* terbaik dalam uji coba ini adalah 20 *epochs* dengan perolehan akurasi validasi mencapai 94,21%. Pada *epochs* yang berjumlah 20 memperlihatkan bahwasannya akurasi validasi memperoleh akurasi yang lebih tinggi dibandingkan akurasi *training* hampir pada setiap *epochs*. Hal ini menunjukkan bahwasannya model mampu menggeneralisasi dengan baik. Masing-masing akurasi pada *training* dan validasi mengalami peningkatan yang fluktuatif. Hal ini berarti model belum mencapai konvergen, namun hal tersebut tidak selalu mengindikasikan adanya *overfitting* atau *underfitting* pada model. Grafik pengujian terhadap pengaruh jumlah *epochs* seperti pada Gambar 8.



### 3.2 Hasil Evaluasi Model

Berdasarkan skenario uji coba terhadap *hyperparameter* yang telah dilakukan maka diperoleh *hyperparameter* yaitu: *optimizer* SGD, 0,01 *learning rate*, 16 *batch size*, dan 20 *epochs*. Model CNN dengan akurasi tertinggi yang diperoleh akan dilakukan evaluasi menggunakan data uji/*testing* sehingga memperoleh *confusion matrix* seperti pada Gambar 9. Berdasarkan *confusion matrix* pada Gambar 9. maka diperoleh nilai *precision*, *recall*, *f1-score*, dan akurasi pada data uji seperti pada Tabel 7.



Gambar 9 Confusion Matrix pada Data Uji

Tabel 7 Hasil Evaluasi Model CNN pada Data Uji

	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Fresh Apple	69,23%	90%	78,26%
Fresh Banana	90,91%	100%	95,24%
Fresh cucumber	77,78%	70%	73,68%
Fresh Okra	58,33%	70%	63,64%
Fresh Potato	100%	60%	75%
Fresh Tomato	100%	90%	94,74%
Rotten Apple	76,92%	100%	86,96%
Rotten Banana	64,29%	90%	75%
Rotten cucumber	87,5%	70%	77,78%
Rotten Okra	100%	70%	82,35%
Rotten Potato	80%	80%	80%
Rotten Tomato	100%	80%	88,89%
<b>Rata-rata</b>	<b>83,75%</b>	<b>80,83%</b>	<b>80,96%</b>
<b>Akurasi</b>		<b>80,83%</b>	

## 4. KESIMPULAN

Memisahkan antara buah dan sayuran yang segar dengan busuk dapat dilakukan dengan menggunakan teknologi *deep learning*. Salah satu teknologi *deep learning* adalah *Convolutional Neural Network* (CNN). Dalam penelitian ini, klasifikasi dengan CNN memperoleh akurasi pelatihan sebesar 90,42%, akurasi validasi mencapai 94,21%, dan akurasi pengujian mencapai 80,83%. Akurasi ini diperoleh dengan menggunakan *optimizer* SGD, 0,01 *learning rate*, 16 *batch size*, dan 20 *epochs*. Dengan demikian, penggunaan teknologi *deep learning*, khususnya CNN, dapat menjadi solusi yang andal dan efisien dalam melakukan klasifikasi objek visual, seperti membedakan antara buah dan sayuran yang segar dengan yang busuk untuk tujuan pemilihan dan manajemen kualitas produk.



## DAFTAR PUSTAKA

- Alfarizi, M. R. S., Al-farish, M. Z., Taufiqurrahman, M., Ardiansah, G., & Elgar, M. (2023). Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning. *KARIMAH TAUHID*, 2(1), 1–6. <https://doi.org/10.30997/KARIMAHTAUHID.V2I1.7518>
- Apendi, S., Setianingsih, C., & Paryasto, M. W. (2023). Deteksi Bahasa Isyarat Sistem Isyarat Bahasa Indonesia Menggunakan Metode Single Shot Multibox Detector. *EProceedings of Engineering*, 10(1), 249–255. <https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/19322>
- Ayumi, V. (Vina), & Nurhaida, I. (Ida). (2021). Klasifikasi Chest X-Ray Images Berdasarkan Kriteria Gejala Covid-19 Menggunakan Convolutional Neural Network. *Journal Scientific and Applied Informatics*, 4(2), 147–153. <https://doi.org/10.36085/JSAL.V4I2.1513>
- Baloch, A., & Khalique, A. (2022). *Vegetables & Fruits fresh and Stale*. Kaggle. <https://www.kaggle.com/datasets/alibaloch/vegetables-fruits-fresh-and-stale>
- Gregori, D., French, M., Gallipoli, S., Lorenzoni, G., & Ghidina, M. (2019). Global, Regional, and National Levels of Fruit and Vegetable Consumption from the ROUND (WoRld Map of COnsUmption of Fruit and Vegetables and Nutrient Deficits) Project (P18-067-19). *Current Developments in Nutrition*, 3(Suppl 1), nzz039.P18-067-19. <https://doi.org/10.1093/CDN/NZZ039.P18-067-19>
- Hawari, F. H., Fadillah, F., Alviandi, M. R., & Arifin, T. (2022). Klasifikasi Penyakit Tanaman Padi Menggunakan Algoritma CNN (Convolutional Neural Network). *Jurnal Responsif: Riset Sains Dan Informatika*, 4(2), 184–189. <https://doi.org/10.51977/JTI.V4I2.856>
- Iswantoro, D., & UN, D. H. (2022). Klasifikasi Penyakit Tanaman Jagung Menggunakan Metode Convolutional Neural Network (CNN). *Jurnal Ilmiah Universitas Batanghari Jambi*, 22(2), 900. <https://doi.org/10.33087/jiubj.v22i2.2065>
- Jafaruddin, N. (2023). Pola Hidup Sehat dengan Konsumsi Buah dan Sayur. *Abdimas Galuh*, 5(1), 569–577. <https://doi.org/10.25157/AG.V5I1.9919>
- Magalhães, B., Gaspar, P. D., Corceiro, A., João, L., & Bumba, C. (2022). Fuzzy Logic Decision Support System to Predict Peaches Marketable Period at Highest Quality. *Climate 2022, Vol. 10, Page 29, 10(3)*, 29. <https://doi.org/10.3390/CLI10030029>
- Mahaputri, C., Kristian, Y., & Setyati, E. (2022). Pengenalan Makanan Tradisional Indonesia Beserta Bahan-bahannya dengan Memanfaatkan DCNN Transfer Learning. *INSYST: Journal of Intelligent System and Computation*, 4(2), 61–68. <https://doi.org/10.52985/INSYST.V4I2.252>
- Mulyanto, A., Susanti, E., Rossi, F., Wajiran, W., & Borman, R. I. (2021). Penerapan Convolutional Neural Network (CNN) pada Pengenalan Aksara Lampung Berbasis Optical Character Recognition (OCR). *JEPIN (Jurnal Edukasi Dan Penelitian Informatika)*, 7(1), 52–57. <https://doi.org/10.26418/JP.V7I1.44133>
- Qotrunnada, Mufida, F., & Utomo, P. H. (2022). Metode Convolutional Neural Network untuk Klasifikasi Wajah Bermasker. *PRISMA, Prosiding Seminar Nasional Matematika*, 5, 799–807. <https://journal.unnes.ac.id/sju/index.php/prisma/article/view/54602>
- Rarastiti, C. N. (2022). Hubungan Pengetahuan Gizi Seimbang dengan Konsumsi Buah dan Sayur pada Remaja. *Jurnal Penelitian Inovatif*, 2(2), 281–288. <https://doi.org/10.54082/JUPIN.80>
- Sya'ban, D. R., Hamzah, A., & Susanti, E. (2022). Klasifikasi Buah Segar dan Busuk Menggunakan Algoritma Convolutional Neural Network Dengan TFLite Sebagai Media Penerapan Model Machine Learning. *PROSIDING SNAST*, F7-16. <https://doi.org/10.34151/PROSIDINGSNAST.V8I1.4180>
- Wulandari, I., Yasin, H., & Widiharih, T. (2020). Klasifikasi Citra Digital Bumbu dan Rempah dengan Algoritma Convolutional Neural Network (CNN). *Jurnal Gaussian*, 9(3), 273–282. <https://doi.org/10.14710/J.GAUSS.9.3.273-282>

