Convolutional Neural Network Implementation Using the TensorFlow Library for Freshness Detection in Apples
[Implementasi Convolutional Neural Network Menggunakan Library TensorFlow untuk Deteksi Kesegaran pada Apel]

Diana Cindy Agustin *[1)], Mochamad Alfan Rosid [2)]

[1),2)] Program Studi Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia
*Email Penulis Korespondensi: diandystin@umsida.ac.id

Abstract. *Artificial intelligence products such as computer vision have begun to replace the role of humans in several fields of work. Based on BPS data, the percentage of Indonesia's fruit exports grew 10.36% in February 2022 with a value of US$ 340 million. This causes the need for time efficiency when carrying out the export process for determining the quality of the fruit. The amount of time, human resources, and the lack of understanding in determining the quality of fruit are things that need to be considered. Based on these problems and questions, this research was conducted to create a project that proves computer vision can also determine fruit's quality by applying the Convolutional Neural Network algorithm with LeNet-5 architecture. This project is an android mobile-based application that can detect the condition of the fruit whether it is suitable (fresh) or not suitable (rotten) for consumption*.

Keywords – *android; artificial intelligence; convolutional neural network; fruit freshness detection; LeNet-5*

Abstrak. *Produk kecerdasan buatan seperti visi telah mulai menggantikan peran manusia dalam beberapa bidang pekerjaan. Berdasarkan data BPS, persentase ekspor buah Indonesia tumbuh 10,36% pada Februari 2022 dengan nilai US$ 340 juta. Hal ini menyebabkan perlunya efisiensi waktu saat melakukan proses ekspor untuk menentukan kualitas buah. Jumlah waktu, sumber daya manusia, dan kurangnya pemahaman dalam menentukan kualitas buah adalah hal yang perlu diperhatikan. Berdasarkan permasalahan dan pertanyaan tersebut, penelitian ini dilakukan untuk membuat sebuah proyek yang membuktikan bahwa visi computer juga dapat menentukan kualitas buah dengan menerapkan algoritma Convolutional Neural Network dengan struktur LeNet-5. Proyek ini merupakan aplikasi berbasis mobile android yang dapat mendeteksi kondisi buah apakah cocok (segar) atau tidak cocok (busuk) untuk dikonsumsi.*

Kata Kunci – *android; convolutional neural network; deteksi kesegaran buah; kecerdasan buatan; LeNet-5*

I. PENDAHULUAN

Meski sudah dipetik dari pohonnya, apel pada umumnya akan tetap mengalami proses pematangan terutama jika dibiarkan dalam suhu ruangan. Proses impor dan ekspor membutuhkan waktu yang sangat lama terlebih dalam proses pengirimannya, oleh karena itu dilakukan banyak inovasi untuk meningkatkan umur simpan dari produk buah maupun komoditas hortikultura lainnya. Namun walaupun sudah dilakukan inovasi untuk meningkatkan umur simpannya, beberapa buah akan tetap ada yang mengalami pembusukan, hal ini menyebabkan proses pemilihan buah yang masih segar atau busuk harus dilakukan. Proses pemilahan buah untuk memisahkan yang segar dan busuk tentu akan membutuhkan waktu yang lama dan sumber daya manusia yang besar. Dengan perkembangan teknologi yang semakin cepat, sangat memungkinkan bahwa suatu saat nanti teknologi juga bisa menggantikan beberapa peran manusia dalam pekerjaan.

Di tahun 2019, Indonesia berhasil mengekspor komoditi apel sebanyak 149.074,12 kg dengan total nilai sebesar 180.012,87 dolar AS [1]. Hal ini menyebabkan perlunya efisiensi waktu khususnya ketika melakukan proses ekspor dan impor terutama dalam menentukan kualitas buah.

Banyaknya waktu yang terbuang untuk memisahkan buah yang baik atau busuk secara manual dan kurangnya pemahaman konsumen dalam menentukan kualitas buah juga bisa menyebabkan bertambahnya masalah lingkungan. Secara global, diperkirakan 1,3 miliar ton makanan untuk manusia hilang dan terbuang per tahun [2].

Suatu sistem perlu dikembangkan untuk membantu memilah buah yang segar dan busuk. Salah satu teknologi yang dapat digunakan untuk membuat *computer vision* adalah bidang pembelajaran mesin (*machine learning*), yaitu aplikasi komputer dan algoritma matematika yang diambil dengan cara pembelajaran yang berasal dari data dan menghasilkan prediksi di masa yang akan datang [3]. *Machine learning* memiliki cabang ilmu berbasis Jaringan Saraf Tiruan (JST) atau bisa juga disebut sebagai perkembangan dari JST yakni deep learning. *Convolutional Neural Network* (CNN/ConvNet) adalah salah satu algoritma dari deep learning yang merupakan pengembangan dari *Multilayer Perceptron* (MLP) yang dirancang untuk mengolah data dalam bentuk dua dimensi, misalnya gambar atau suara [4].

Copyright © Universitas Muhammadiyah Sidoarjo. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Beberapa penelit telah melakukan penelitian terhadap system deteksi kesegaran apel dengan menggunakan beberapa struktur CNN yang berbeda. Berdasarkan penelitian yang dilakukan oleh Y. Li, X. Feng, Y. Liu, and X. Han dengan judul "*Apple quality identification and classification by image processing based on convolutional neural networks*" [5] menggunakan struktur model dari Google Inceptionv3 dengan 28.800 training data dan 7200 validation data, mendapatkan akurasi untuk pelatihan sebesar 92% dan validasi sebesar 91,2%.

Pada penelitian lain yang dilakukan oleh A. B. Kuriakose, F. Siju, M. K. Aji, T. Rahim, and C. Rani, dengan judul "*Automatic Fruit Classification and Freshness Detection*" [6] menggunakan struktur model VGGnet dan sistem mendapatkan hasil yang sesuai, namun penelitian tersebut menghasilkan sistem yang mungkin akan sulit untuk digunakan oleh masyarakat yang kurang pengetahuan akan teknologi sehingga pengembangan sistem yang lebih user friendly masih bisa dilakukan.

Berdasarkan permasalah dan pertanyaan tersebut, penelitian ini dilakukan untuk membuat sebuah proyek yang memudahkan pengguna dalam menentukan kualitas buah dalam waktu singkat dengan menerapkan algoritma *Convolutional Neural Network* dengan bantuan menggunakan struktur LeNet-5 yang selanjutnya akan dikelola menggunakan bantuan google colab.

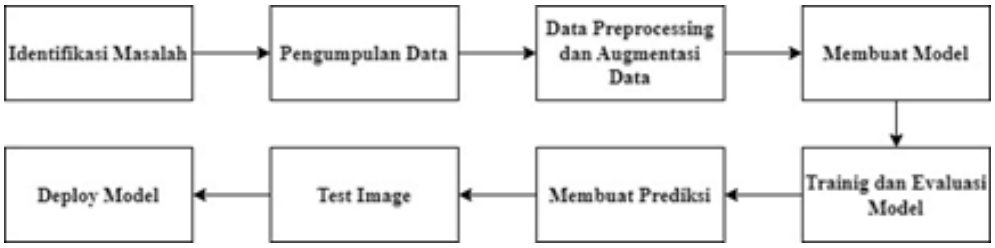
Proyek ini adalah sebuah aplikasi berbasis mobile android yang dapat mendeteksi kondisi buah apakah layak untuk dikonsumsi (masih dalam kondisi baik) atau tidak layak untuk dikonsumsi (busuk) sehingga dapat menciptakan gaya hidup masyarakat lebih sehat dan cara memilih buah yang tepat.

II. METODE

Pada penelitian ini dilakukan beberapa tahap proses penelitian untuk mendapatkan program yang dapat mengenali kesegaran buah apel berdasarkan foto atau gambar yang diberikan.

A. Sistematis Penelitian

Pada Gambar 1 berikut adalah sistematis penelitian yang dilakukan beserta penjelasan untuk setiap proses penelitian ini.



Gambar 1. Sistematis Penelitian

B. Identifikasi Masalah

Penelitian dilakukan berdasarkan permasalahan yang ada dan sudah dijelaskan sebelumnya, yaitu :

1. Penerapan metode Convolutional Neural Network untuk membangun aplikasi deteksi kesegaran pada buah apel.
2. Aplikasi yang dikembangkan mampu mendeteksi buah apel yang segar atau busuk.

C. Pengumpulan Data

Langkah pertama dari penelitian ini adalah pengumpulan data yang diambil melalui Kaggle [7] yaitu komunitas online untuk data scientist dan praktisi machine learning yang memungkinkan pengguna untuk menemukan dan menerbitkan kumpulan data, menjelajahi dan membuat model dalam lingkungan ilmu data berbasis web, bekerja dengan data scientist dan machine learning engineers untuk memecahkan tantangan ilmu data.

Dataset yang digunakan memiliki hingga 4.259 data gambar terdiri dari data *train* dan data *test*, masing-masing data *train* dan *test* memiliki dua kelas yaitu *fresh apples* dan *rotten apples*. Gambar 2 adalah *data test* untuk *fresh apples* dan *rotten apples*.

Copyright © Universitas Muhammadiyah Sidoarjo. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Gambar 2. Dataset Test Fresh Apples dan Rotten Apples [7]

Gambar 3 berikut merupakan dataset train untuk *fresh apples* dan *rotten apples*.

Gambar 3. Dataset Train Fresh Apples dan Rotten Apples [7]

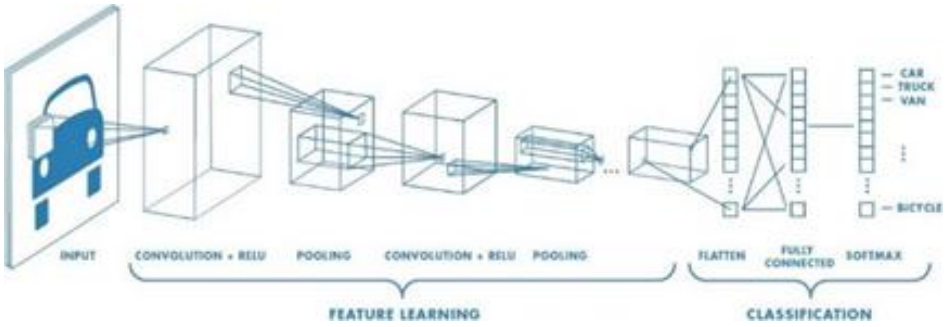
D. Data Preprocessing dan Augmentasi Data

Data preprocessing dilakukan dengan memisahkan data menjadi dua set data yaitu *training set* yang akan dipelajari oleh model dan *testing set* untuk memeriksa keakuratan model setelah *training*. Augmentasi data dilakukan untuk membantu mengatasi masalah umum pada *deep learning* yaitu *data hungry* (membutuhkan banyak data) juga *data overfitting* karena setelah dilakukan augmentasi maka setiap data akan mendapatkan data baru, mengkombinasikan beberapa augmentasi bisa dilakukan sehingga jumlah data juga akan semakin banyak [8] Pada penelitian kali ini menggunakan augmentasi data *flip, random crop, random rotation, dan shear* sehingga menghasilkan 652 citra baru .

E. Membuat Model

Sebuah model *machine learning* akan menentukan keluaran yang didapatkan setelah menjalankan algoritma pembelajaran *machine learning* pada data yang dikumpulkan. Penting untuk memilih model yang relevan dengan tugas yang dihadapi. *Convolutional Neural Network* hanya bisa digunakan pada dataset dua dimensi seperti objek gambar karena *Convolutional Neural Network* memiliki sifat proses konvolusi [9].

Convolutional Neural Network memiliki cara kerja yang sama dengan tradisional *Artificial Neural Network*, mulai dari masukan awal sampai hasil akhir, setiap jaringan akan tetap menampilkan sebuah skor atau yang disebut dengan *weight* dan *loss functions* pada *layer* terakhir. CNN memiliki empat macam *layer* utama menurut LeNet5 yaitu *convolutional layer*, *relu layer*, *pooling/subsampling layer, dan fully connected layer* seperti yang dapat dilihat pada Gambar 4 [10].



Gambar 4. Arsitektur CNN [10]

Copyright © Universitas Muhammadiyah Sidoarjo. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

1. Convolutional Layer

Convolutional layer melakukan operasi dot antara matriks input dengan filter untuk mengubah input menjadi feature maps [11].

2. ReLu Layer (fungsi aktivasi ReLu)

ReLu (Rectification Linear Unit) adalah operasi peningkatan representasi dan pengenalan nonlinearitas dari model [10].

3. Pooling (Subsampling) Layer

Pooling dilakukan bertujuan untuk mengurangi banyaknya parameter atau spasial, ada dua macam *pooling* yang umum digunakan yaitu *average pooling* dan *max pooling* [12].

4. Fully Connected Layer

Terdapat beberapa *fully connected layer* pada bagian klasifikasi, salah satunya yaitu *flatten layer* yang digunakan untuk *flattening* data yang berbentuk multidimensional *array* menjadi satu dimensi. *Layer* lainnya merupakan *hidden layer* dan *output layer* atau *layer dense* [13].

TensorFlow merupakan salah satu *library* yang bersifat *open source* untuk *data science* yang dikembangkan oleh Google untuk penelirian jaringan syaraf dalam dan pembelajaran mesin [8].

Untuk mencetak ringkasan model yang mencakup nama dan jenis semua lapisan dalam model, bentuk keluaran untuk setiap lapisan, dan jumlah parameter bobot tiap layer, menggunakan *model summary* seperti pada Tabel 1.

Tabel 1. Hasil Model Summary

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 98, 98, 32)	896
batch_normalization (BatchNormalization)	(None, 98, 98, 32)	128
max_pooling2d (MaxPooling2D)	(None, 49, 49, 32)	0
batch_normalization_1 (BatchNormalization)	(None, 49, 49, 32)	128
dropout (Dropout)	(None, 49, 49, 32)	0

F. Training dan Evaluasi Model

Training merupakan proses paling penting dalam *machine learning*. Saat *training*, data yang sudah disiapkan akan diteruskan ke model *machine learning* untuk menemukan pola dan membuat prediksi. Hal ini menghasilkan model *learning* dari data sehingga dapat menyelesaikan tugas yang ditetapkan.

Setelah melakukan *training* pada model, selanjutnya adalah memeriksa atau mengevaluasi model dengan menguji model pada *data testing* yang sudah dipisahkan pada *data preprocessing* sebelumnya [14].

G. Membuat Prediksi dan Test Image

Setelah melakukan evaluasi pada model, tahap selanjutnya adalah membuat prediksi dan mencoba model dengan memasukkan data gambar di luar dataset yang sudah digunakan dalam tahap *training* sampai evaluasi karena jika tetap menggunakan data yang sama, hasil yang didapatkan menjadi kurang akurat karena model sudah terbiasa dengan data tersebut dan menemukan pola yang sama di dalamnya seperti sebelumnya.

H. Deploy Model

Deploy model dilakukan untuk memasukkan model *machine learning* ke *device* yang diinginkan. Dalam penelitian kali ini model akan *di-deploy* ke android. Setelah model sudah berada pada performa terbaik maka selanjutnya model akan disimpan alam format .h5 kemudian model yang dihasilkan dapat diekspor dengan format Flatbuffer (.tflite) untuk dijalankan pada perangkat mobile dengan menggunakan *TensorFlow Lite*.

III. HASIL DAN PEMBAHASAN

Deteksi kesegaran buah ini dilakukan dengan memasukkan inputan berupa gambar atau foto melalui kamera pada aplikasi yang kemudian akan memberikan hasil klasifikasi buah tersebut termasuk dalam buah yang segar atau busuk. Proses pembuatan model dilakukan menggunakan bahasa pemrograman python dengan metode *convolutional neural network* dan sturuktur model LeNet-5. Masing-masing data *train* dan *test* memiliki dua kelas yaitu *fresh apples* dan *rotten apples*. Kemudian akan dilakukan *pre-processing* data dengan membagi data menjadi data *train*, data *validation*, dan data *test* yang disajikan pada Tabel 2.

Copyright © Universitas Muhammadiyah Sidoarjo. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Tabel 2. Partisi Data

Bagian	Jumlah Data	Kelas	Jumlah
Data *train*	2611	*Fresh Apples*	1355
		Rotten Apples	1256
Data *validation*	652	*Fresh Apples*	338
		Rotten Apples	314
Data *test*	996	*Fresh Apples*	395
		Rotten Apples	601

Model yang digunakan dalam penelitian ini terdiri dari *convolutional layer, pooling layer, dropout layer, flatten layer,* dan *fully connected layer.* Setelah model terbentuk kemudian akan dilakukan *training* citra menggunakan model fit dengan 15 kali iterasi atau epoch. Proses training dengan menunjukkan perubahan *accuracy* dan *loss* ditunjukkan pada Gambar 5.

```
Epoch 1/15
/usr/local/lib/python3.8/dist-packages/keras/backend.py:5585: UserWarning: "`sparse_categorical_crossentropy` received `from_logits
  output, from_logits = _get_logits(
131/131 [==============================] - 45s 245ms/step - loss: 1.1987 - accuracy: 0.8035 - val_loss: 3.7096 - val_accuracy: 0.67
Epoch 2/15
131/131 [==============================] - 27s 210ms/step - loss: 0.3062 - accuracy: 0.8870 - val_loss: 1.8754 - val_accuracy: 0.86
Epoch 3/15
131/131 [==============================] - 27s 208ms/step - loss: 0.2765 - accuracy: 0.8809 - val_loss: 1.4554 - val_accuracy: 0.85
Epoch 4/15
131/131 [==============================] - 27s 210ms/step - loss: 0.2638 - accuracy: 0.8970 - val_loss: 0.9090 - val_accuracy: 0.88
Epoch 5/15
131/131 [==============================] - 27s 209ms/step - loss: 0.2634 - accuracy: 0.8958 - val_loss: 0.6390 - val_accuracy: 0.94
Epoch 6/15
131/131 [==============================] - 28s 211ms/step - loss: 0.2305 - accuracy: 0.9004 - val_loss: 0.5078 - val_accuracy: 0.93
Epoch 7/15
131/131 [==============================] - 28s 211ms/step - loss: 0.2638 - accuracy: 0.9004 - val_loss: 0.5560 - val_accuracy: 0.93
Epoch 8/15
131/131 [==============================] - 28s 211ms/step - loss: 0.2262 - accuracy: 0.9108 - val_loss: 1.0620 - val_accuracy: 0.85
Epoch 9/15
131/131 [==============================] - 28s 211ms/step - loss: 0.2239 - accuracy: 0.9134 - val_loss: 0.3951 - val_accuracy: 0.85
Epoch 10/15
131/131 [==============================] - 27s 209ms/step - loss: 0.2218 - accuracy: 0.9192 - val_loss: 1.2602 - val_accuracy: 0.83
Epoch 11/15
131/131 [==============================] - 32s 243ms/step - loss: 0.2169 - accuracy: 0.9119 - val_loss: 0.4587 - val_accuracy: 0.85
Epoch 12/15
131/131 [==============================] - 27s 208ms/step - loss: 0.2253 - accuracy: 0.9119 - val_loss: 6.1676 - val_accuracy: 0.66
Epoch 13/15
131/131 [==============================] - 27s 206ms/step - loss: 0.2226 - accuracy: 0.9203 - val_loss: 0.2365 - val_accuracy: 0.92
Epoch 14/15
131/131 [==============================] - 27s 208ms/step - loss: 0.2187 - accuracy: 0.9184 - val_loss: 0.5563 - val_accuracy: 0.83
Epoch 15/15
131/131 [==============================] - 32s 243ms/step - loss: 0.2027 - accuracy: 0.9230 - val_loss: 0.3722 - val_accuracy: 0.94
```

Gambar 5. Proses Training

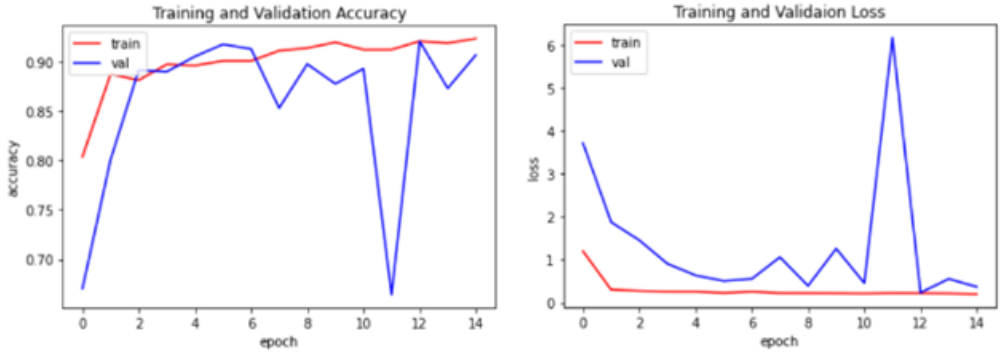
Setelah dilakukan *training* maka model akan melakukan evaluasi pada 996 data *test* untuk melihat tingkat akurasi dari model yang sudah dibuat dalam membaca dan mengklasifikasikan data seperti pada Gambar 6.

```
Found 996 images belonging to 2 classes.
996/996 [==============================] - 10s 10ms/step - loss: 0.1555 - accuracy: 0.9357
accuracy test:  0.935742974281311
```

Gambar 6. Hasil Evaluasi Data

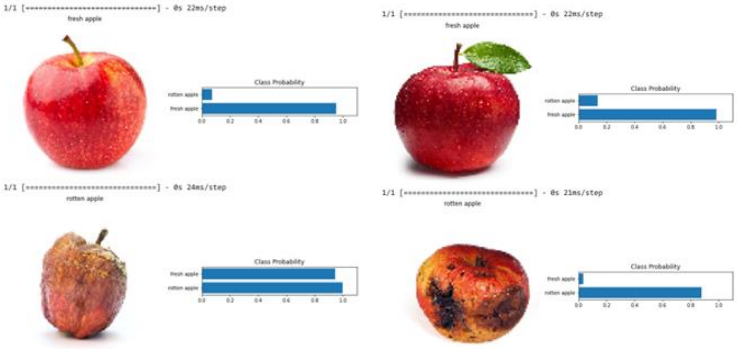
Seperti yang dapat dilihat pada Gambar 6, akurasi yang ditampilkan mencapai angka 0.93 atau 93% dari maksimal 100% tingkat akurasi yang bisa didapat oleh sebuah model. Proses training yang dilakukan oleh model menunjukkan *validation accuracy* yang didapat relatif stabil pada iterasi 0 sampai iterasi 11 kemudian ada perubahan cukup signifikan di satu titik dari iterasi 11 ke 12 dan cenderung stabil kembali pada iterasi 12 sampai selesai. Validation loss juga menunjukkan hasil yang sama, ada perubahan signifikan pada satu titik saja yakni iterasi ke 12.

Copyright © Universitas Muhammadiyah Sidoarjo. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Gambar 7. Grafik Model dengan Epoch 15

Berdasarkan Gambar 7 dapat dilihat bahwa semakin bertambahnya jumlah epoch maka semakin naik grafik *accuracy* dan semakin turun grafik *loss* yang didapat. Hal ini menjadi baik karena itu artinya *validation loss* semakin rendah dan tingkat *validation accuracy* semakin naik. Model dianggap sudah cukup baik karena memiliki akurasi tinggi dan loss rendah.

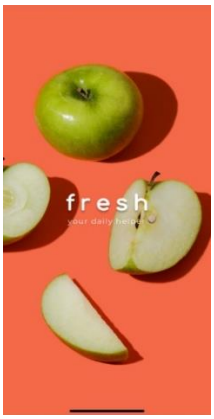


Gambar 8. Hasil Tes Model

Pada Gambar 8 dapat dilihat bahwa hasil tes model menggunakan data yang berbeda dari dataset *train* dan *test*, menunjukkan hasil yang sesuai dan model dapat mengklasifikasikan buah tersebut segar atau busuk berdasarkan kelasnya. Pada penelitian kali ini model diekstrak menjadi format Flatbuffer (.tflite). Setelah diekstrak, model kemudian di-deploy ke dalam android untuk memudahkan visualisasi dari program yang sudah dibuat. Aplikasi android ini memiliki tiga tahapan proses yang akan dilalui oleh pengguna yakni halaman splash screen, halaman input gambar atau foto, dan halaman hasil prediksi.

A. Halaman Splash Screen

Pada halaman ini pengguna akan disajikan tampilan awal dari aplikasi ketika akan digunakan

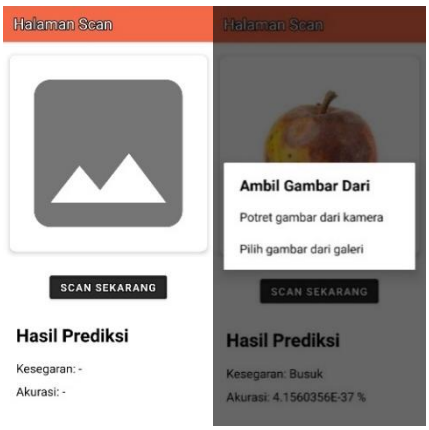


Gambar 9. Halaman Splash Screen

Copyright © Universitas Muhammadiyah Sidoarjo. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

B. Halaman Input Gambar atau Foto

Pada halaman ini pengguna diminta untuk memasukkan gambar yang sudah berada dalam perangkat seluler mereka atau menggambil foto apel dari kamera perangkat seluler secara *real time*.



Gambar 10. Halaman Input Gambar

C. Halaman Hasil Prediksi

Setelah memasukkan gambar atau foto dari pengguna, selanjutnya aplikasi akan memberikan hasil prediksi apel tersebut masuk ke dalam kelas apel segar atau busuk.

Gambar 11. Halaman Hasil Prediksi Kesegaran Apel

IV. SIMPULAN

Penelitian kali ini dilakukan dengan tujuan untuk membuat sistem atau aplikasi yang dapat mengenali kesegaran buah apel kemudian mengelompokkannya ke dalam kelas yakni *fresh apple* atau *rotten apple*. Kesimpulan yang didapatkan pada penelitian ini adalah model yang sudah dibuat menggunakan empat macam *layer* utama menurut LeNet 5 yakni *convolutional layer*, *relu layer*, *pooling/subsampling layer, dan fully connected layer* dapat mengenali kesegaran pada buah apel dan menentukan kelas dari setiap apel dengan benar.

Penggunaan layer yang tepat juga dapat mempengaruhi kinerja model yang dibuat, dalam penelitian kali ini penggunaan kombinasi *convolutional layer*, *relu layer*, *pooling/subsampling layer*, *dan fully connected layer* mampu memberikan hasil akurasi yang cukup baik yakni 93%. Adapun penerapan *TensorFlow Lite* berjalan baik dan mampu memberikan hasil yang sesuai setelah model dimasukkan ke dalam android.

Adapun saran yang dapat digunakan untuk penelitian selanjutnya yakni pengembangan kemampuan model untuk melihat kesegaran apel tidak hanya dari visual saja melainkan bisa dari sisi lain seperti bagian dalam apel itu sendiri atau memasukkan model ke perangkat hardware lainnya selain android untuk penggunaan dalam skala yang lebih besar. Model yang sudah dibuat menggunakan struktur LeNet-5 berjalan dengan cukup baik, namun akurasinya bisa ditingkatkan dengan mencoba implementasi struktur lain atau *transfer learning* sehingga bisa mencapai akurasi 100% untuk mendapatkan kinerja model dan keakuratan yang lebih baik.

Copyright © Universitas Muhammadiyah Sidoarjo. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

UCAPAN TERIMA KASIH

Terimakasih kepada Laboratorium Informatika Universitas Muhammadiyah Sidoarjo yang telah memberikan dukungan fasilitas.

REFERENSI

[1] Cintia Alifta Riyanisa, "RANTAI NILAI GLOBAL (GLOBAL VALUE CHAINS) PADA MASA PANDEMI TERKAIT POSISI BUAH JAMBU DAN APEL," *Indonesian Journal of International Relations*, vol. 6, no. 2, pp. 307–325, Aug. 2022, doi: 10.32787/ijir.v6i2.349.

[2] Z. Dou, J. D. Toth, and M. L. Westendorf, "Food waste for livestock feeding: Feasibility, safety, and sustainability implications," *Global Food Security*, vol. 17. Elsevier B.V., pp. 154–161, Jun. 01, 2018. doi: 10.1016/j.gfs.2017.12.003.

[3] J. Homepage, A. Roihan, P. Abas Sunarya, and A. S. Rafika, "IJCIT (Indonesian Journal on Computer and Information Technology) Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper," 2019.

[4] S. Ilahiyah and A. Nilogiri, "Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network".

[5] Y. Li, X. Feng, Y. Liu, and X. Han, "Apple quality identification and classification by image processing based on convolutional neural networks," *Sci Rep*, vol. 11, no. 1, Dec. 2021, doi: 10.1038/s41598-021-96103-2.

[6] A. B. Kuriakose, F. Siju, M. K. Aji, T. Rahim, and C. Rani, "Automatic Fruit Classification and Freshness Detection." [Online]. Available: www.ijert.org

[7] SRIRAM REDDY KALLURI, "Fruits fresh and rotten for classification," *Kaggle*, 2018.

[8] M. Faturrachman and I. Yustiana, "DISEASE DETECTION SYSTEM IN CASSAVA LEAVES USING ANDROID-BASED DEEP LEARNING AND TENSORFLOW."

[9] I. W. Prastika, E. Zuliarso, J. T. Lomba, J. No, and S. 50241, "DETEKSI PENYAKIT KULIT WAJAH MENGGUNAKAN TENSORFLOW DENGAN METODE CONVOLUTIONAL NEURAL NETWORK," *Jurnal Manajemen informatika & Sistem Informasi)*, vol. 4, no. 2, 2021, [Online]. Available: http://e-journal.stmiklombok.ac.id/index.php/misi

[10] S. Yuliany and A. Nur Rachman, "Implementasi Deep Learning pada Sistem Klasifikasi Hama Tanaman Padi Menggunakan Metode Convolutional Neural Network (CNN)," 2022.

[11] I. Wulandari, H. Yasin, and T. Widiharih, "KLASIFIKASI CITRA DIGITAL BUMBU DAN REMPAH DENGAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK (CNN)", [Online]. Available: https://ejournal3.undip.ac.id/index.php/gaussian/

[12] S. Juliansyah and A. D. Laksito, "Klasifikasi Citra Buah Pir Menggunakan Convolutional Neural Networks," *Jurnal Telekomunikasi dan Komputer*, vol. 11, no. 1, pp. 65–72, 2021, doi: 10.22441/incomtech.v10i2.10185.

[13] B. Nugroho and E. Yulia, "KINERJA METODE CNN UNTUK KLASIFIKASI PNEUMONIA DENGAN VARIASI UKURAN CITRA INPUT," vol. 8, no. 3, pp. 533–538, 2021, doi: 10.25126/jtiik.202184515.

[14] C. A. Lorentius, R. Adipranata, and A. Tjondrowiguno, "Pengenalan Aksara Jawa dengan Menggunakan Metode Convolutional Neural Network."

Conflict of Interest Statement:
The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © Universitas Muhammadiyah Sidoarjo. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.