

KLASIFIKASI BUAH MANGGIS LOKAL DAN EKSPOR BERDASARKAN WARNA DENGAN ARSITEKTUR *MOBILENET V2*

Tedi Hadi Mursid¹, M. Rafi Muttaqin², Agus Sunandar³

Teknik Informatika, Sekolah Tinggi Teknologi Wastukencana
Purwakarta, Jawa Barat

E-mail: tedihadi98@wastukencana.ac.id¹, rafi@wastukencana.ac.id², agoes.61@wastukencana.ac.id³

Abstrak - Manggis (*Garcinia mangostana L.*) merupakan tumbuhan yang berasal dari daerah Asia Tenggara meliputi Indonesia, Malaysia, Thailand dan Myanmar. Buah manggis memiliki rasa yang khas, banyak digemari sampai keluar negeri hingga menjadi salah satu komoditas ekspor yang sangat prospektif, dengan pangsa pasar yang tak pernah jenuh. Meskipun demikian, dalam konteks produksi buah manggis di Indonesia, terdapat permasalahan yang signifikan terkait penyortiran buah manggis setelah panen khususnya dalam hal analisis warna kulit buah secara visual oleh manusia, yang jelas memiliki keterbatasan. Oleh karena itu, mengingat urgensi permasalahan ini, penelitian lebih lanjut diperlukan sebagai langkah menuju pengembangan sistem klasifikasi yang dapat memberikan akurasi tinggi dalam menentukan tingkat kematangan buah manggis untuk produksi lokal dan ekspor. Hal ini diharapkan dapat mengoptimalkan proses penyeleksian hasil panen buah manggis. Untuk mengolah data menggunakan metode *Convolutional Neural Network (CNN)*. CNN sendiri terdiri dari beberapa arsitektur yang salah satunya yaitu *MobileNet V2*, yang mana merupakan salah satu arsitektur CNN yang memiliki kemampuan yang sangat baik dalam mendeteksi objek pada gambar secara akurat dan cepat. Arsitektur ini dapat bekerja dengan menggunakan perangkat yang memiliki sumber daya yang ringan. Berdasarkan penelitian yang dilakukan dengan jumlah dataset sebesar 384 citra untuk 2 jenis manggis yaitu ekspor dan lokal. Dimana 307 citra digunakan untuk *training* dan 39 citra untuk *validation* dataset. Kemudian pada proses pengujian digunakan 38 citra yang diambil secara acak sebagai data *testing* untuk masing-masing citra buah manggis. Setelah dilakukan proses pelatihan diperoleh nilai akurasi terbaik dengan 307 citra untuk data pelatihan dan 39 citra untuk data pengujian sebesar 95%.

Kata Kunci: *Convolutional Neural Network*, Klasifikasi, Manggis, *MobileNet V2*

I. PENDAHULUAN

Manggis merupakan tumbuhan fungsional karena sebagian besar dari tumbuhan tersebut dapat dimanfaatkan sebagai obat. Buah manggis adalah salah satu buah yang dikenal sangat bermanfaat bagi kesehatan. Kandungan zat dalam buah manggis diyakini dapat mencegah timbulnya penyakit dalam. Buah manggis mengandung aktivitas *antiinflamasi* dan *antioksidan*, sehingga di luar negeri buah manggis dikenal sebagai buah yang memiliki kadar *antioksidan* tertinggi didunia (Tampati et al., 2023). Menurut data yang tercantum pada *website* trademap.org, pada tahun 2022 nilai ekspor manggis Indonesia mencapai \$76.256, dengan pasar utama adalah China, Arab, dan Malaysia (Nurarsyika Hesti, 2023).

Meskipun demikian, dalam konteks produksi buah di Indonesia, terdapat permasalahan yang signifikan terkait penyortiran buah setelah panen khususnya dalam hal analisis warna kulit buah secara visual oleh manusia, yang jelas memiliki keterbatasan (Saptana et al., 2019). Metode penyeleksian ini memerlukan tenaga kerja yang cukup besar, dan tingkat konsistensi dalam menilai tingkat kematangan buah tidak selalu terjamin karena manusia dapat mengalami kelelahan. Oleh karena itu, mengingat urgensi permasalahan ini, penelitian lebih lanjut diperlukan sebagai langkah

menuju pengembangan sistem klasifikasi yang dapat memberikan akurasi tinggi dalam menentukan tingkat kematangan buah manggis untuk produksi lokal dan ekspor. Hal ini diharapkan dapat mengoptimalkan proses penyeleksian hasil panen buah manggis.

Petunjuk penting dalam mendeteksi *level* buah manggis adalah warna dari buah manggis tersebut (Narakusuma et al., 2013). Untuk mengolah data menggunakan metode *Convolutional Neural Network (CNN)* bagian dari algoritma *deep learning* dikembangkan dengan *Multy Layer Perceptron (MLP)* dirancang sebagai mengolah data dalam bentukan dua dimensi, misalnya gambar (Pratama et al., 2023). CNN digunakan untuk mengklasifikasi data yang terlabel dengan memfungsikan metode *supervised learning*. Pada tingkat kedalaman jaringan CNN termasuk jenis *Deep Neural Network* dan sering digunakan dalam data citra *image* (Luqman Hakim et al., 2021).

Pada penelitian-penelitian sebelumnya dalam pendeteksian objek menggunakan metode *Convolutional Neural Network* bertujuan sebagai pengganti kinerja otak manusia dalam mengenali kelas tingkat kematangan pada buah (Hidayat, 2022). Pada penelitian-penelitian tersebut menjelaskan *neural network* akan menerima informasi dari *nodes* atau informasi pada titik-titik yang terkumpul pada satu *layer* lalu diteruskan

pada *hidden layer*. Dimana pada metode *convolutional neural network*, *nodes* akan saling terhubung membuat metode *convolutional neural network* lebih hemat daya dalam komputasi atau pemrosesan informasi. CNN sendiri terdiri dari beberapa arsitektur yang salah satunya yaitu *MobileNet V2* yang mana merupakan salah satu arsitektur CNN yang memiliki kemampuan yang sangat baik dalam mendeteksi objek pada gambar secara akurat dan cepat. Arsitektur ini dapat bekerja dengan menggunakan perangkat yang memiliki sumber daya yang ringan (Prima, 2023).

Penelitian ini akan dilihat bagaimana mengklasifikasikan buah kematangan buah manggis ekspor dan lokal berdasarkan warna kulit buah manggis, sehingga diharapkan dapat membantu para petani manggis dalam penyortiran buah manggis.

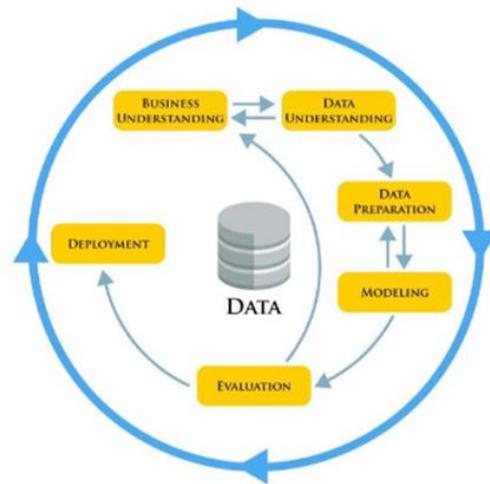
II. TINJAUAN PUSTAKA

Klasifikasi Citra

Pengelompokan piksel atau elemen gambar dalam suatu citra ke dalam kategori-kategori tertentu disebut klasifikasi citra. Setiap kategori mewakili entitas dengan ciri khas yang dapat diidentifikasi. Salah satu metode yang bisa digunakan untuk melakukan klasifikasi citra adalah jaringan saraf tiruan. Metode ini terinspirasi dari cara kerja sistem saraf manusia dan merupakan pendekatan komputasional untuk memproses informasi (Herdiansah et al., 2022).

CRISP-DM

CRISP-DM (*Cross-Industry Standard Process for Data Mining*) adalah metodologi untuk menganalisis dataset yang sangat besar dari berbagai sudut pandang, dengan tujuan menghasilkan wawasan yang bermanfaat. Proses ini juga dikenal sebagai *Knowledge Discovery in Database* (KDD). Metode CRISP-DM memiliki kemampuan untuk mengidentifikasi dan membentuk pola-pola baru dari kumpulan data dan informasi yang masif. Hasil analisis berupa informasi dan pola baru ini dapat dimanfaatkan sebagai alternatif pemecahan masalah dalam berbagai konteks (Iswavigra et al., 2023).



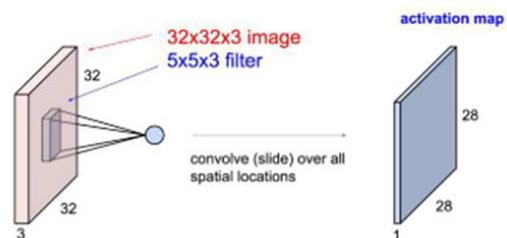
Gambar 1. Metode CRISP-DM

Convolutional Neural Network

Convolutional Neural Network (CNN) umumnya diaplikasikan untuk tugas klasifikasi dan identifikasi gambar. CNN memiliki beragam arsitektur yang populer di komunitas peneliti karena kekhasan dan kemampuannya menghasilkan akurasi tinggi. Sesuai namanya, lapisan konvolusi merupakan komponen kunci dalam CNN. Fungsi utama lapisan ini adalah mengekstraksi ciri-ciri dari gambar menggunakan *filter* atau *kernel* tertentu (Hermana et al., 2024). Pada proses training terdapat 3 tahapan yaitu *convolutional layer*, *pooling layer*, dan *fully connected layer* (Charli et al., 2020). Berikut penjelasan dari ketiga *layer* tersebut:

1. Convolutional Layer

Struktur ini terdiri dari unit-unit yang disebut neuron, yang disusun secara khusus untuk membentuk sebuah filter dengan dimensi tertentu (diukur dalam piksel). Setiap filter ini memiliki nilai-nilai tersendiri, dan nilai-nilai inilah yang berperan sebagai parameter yang akan diperbarui selama proses pembelajaran berlangsung.

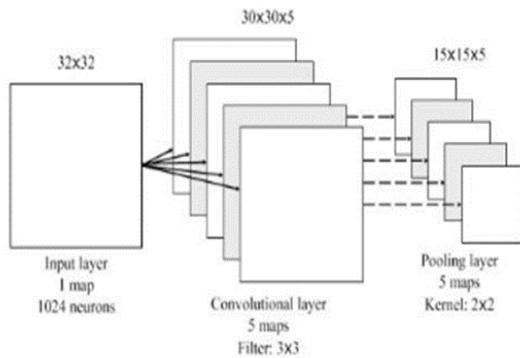


Gambar 2. Convolutional Layer

2. Pooling Layer

Lapisan *pooling* berperan dalam mempertahankan dimensi data selama proses konvolusi dengan cara melakukan pengurangan sampel. Seperti yang diilustrasikan dalam gambar 3, penggunaan *pooling* memungkinkan kita untuk

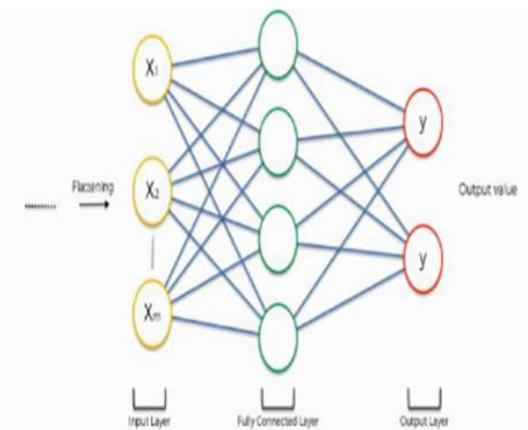
menyajikan data dalam bentuk yang lebih ringkas. Hal ini menjadikan data lebih mudah dikelola dan membantu mencegah overfitting pada model.



Gambar 3. Pooling Layer

3. Fully Connected Layer

Dalam lapisan yang sepenuhnya terhubung, tiap unit neuron memiliki hubungan dengan seluruh aktivasi dari lapisan sebelumnya. Konsep ini identik dengan cara kerja *Multy Layer Perceptron* (MLP), di mana perhitungan dilakukan melalui operasi perkalian matriks yang kemudian ditambahkan dengan nilai bias.

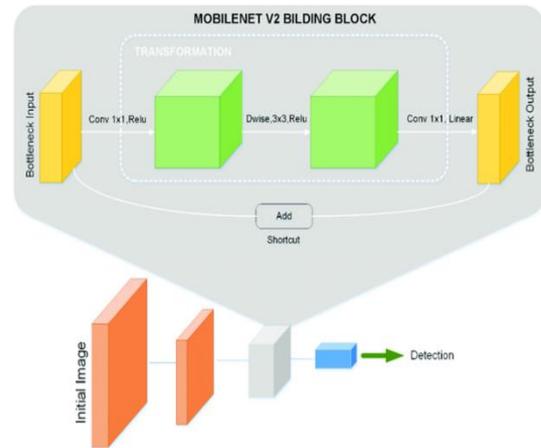


Gambar 4. Fully Connected Layer

MobileNet V2

MobileNet V2 merupakan arsitektur CNN yang dirancang untuk efisiensi, dengan jumlah parameter yang lebih sedikit dibandingkan model arsitektur CNN lainnya. Desain yang dioptimalkan ini memungkinkan implementasi pada perangkat dengan kemampuan komputasi terbatas, seperti ponsel pintar atau *Raspberry Pi*, untuk aplikasi yang membutuhkan respons cepat atau pengolahan secara *real-time* (Anditto & Roestam, 2022). Dalam versi keduanya, *MobileNet V2* memperkenalkan dua fitur baru: *bottleneck linier* dan *shortcut connection* antar *bottleneck*. Arsitektur *MobileNet V2* diilustrasikan pada gambar 5. Bagian *bottleneck* berfungsi sebagai pintu masuk dan keluar data

model. Sementara itu, *shortcut connection* pada *bottleneck* berkontribusi pada peningkatan kecepatan pelatihan dan peningkatan akurasi model (Nyoman & Putu Kusuma Negara, 2021).



Gambar 5. Arsitektur *MobileNet V2*

Confusion Matrix

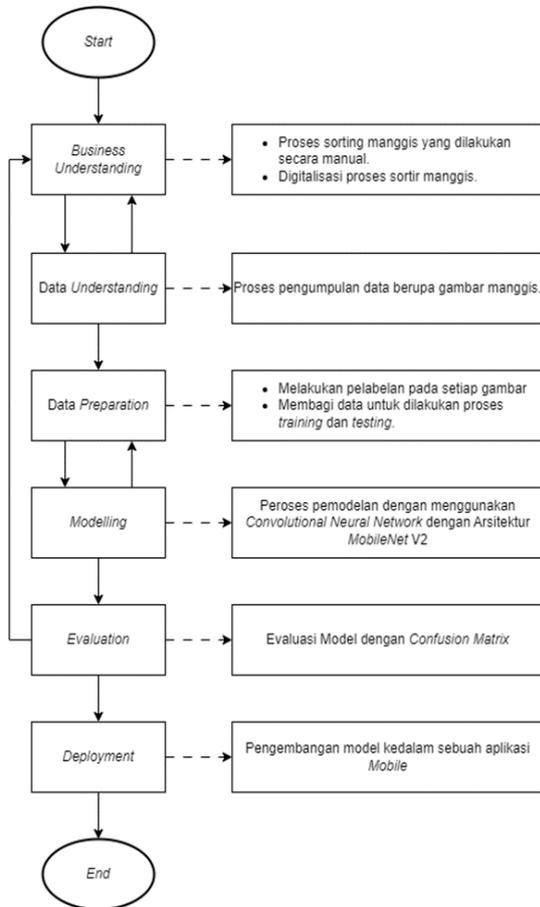
Confusion Matrix merupakan teknik yang umum digunakan dalam konsep data *mining*, untuk mengevaluasi ketepatan klasifikasi. Teknik ini menggunakan tabel untuk menampilkan perbandingan antara jumlah data uji yang berhasil diklasifikasikan dengan benar dan yang salah diklasifikasikan (Utami et al., 2021).

Tabel 1. *Confusion Matrix*

Predicted Class	Actual Class	
	+	-
+	True Positives (TP)	False Positives (FP)
-	False Negatives (FN)	True Negatives (TN)

III. METODE PENELITIAN

Metode penelitian yang digunakan dalam proses tahapan penelitian ini menggunakan *Cross Industry Standart Process for Data Mining* (CRISP-DM) yang terdiri dari beberapa tahapan yang dilakukan diantaranya *Bussines Understanding*, *Data Understanding*, *Data Preparation*, *Modelling*, *Evaluation* dan *Deployment*.



Gambar 6. Metode Penelitian

1. Business Understanding

Proses klafikasi buah manggis untuk tujuan ekspor dan lokal menggunakan metode *deeplearning* dengan algoritma *Convolutional Neural Network* menggunakan arsitektur *MobileNet V2*. Model akan masukkan berupa citra gambar buah manggis yang akan diproses pada jaringan syaraf tiruan, menghasilkan *output* ke dalam kategori Lokal dan Ekspor dalam bentuk aplikasi *realtime*.

2. Data Understanding

Proses pengumpulan data dilakukan secara langsung di lokasi penyortiran buah manggis. Data mentah berupa gambar buah manggis yang memperlihatkan objek secara jelas dan fokus, dengan cara mengambil gambar buah manggis menggunakan kamera digital.

3. Data Preparation

Pada proses tahapan data preparation dilakukan persiapan data citra gambar manggis yang telah dikumpulkan di dalam tahapan sebelumnya, untuk selanjutnya data dapat diproses di dalam tahapan pemodelan. Di dalam data preparation dilakukan 3 persiapan diantaranya *labelling* data, *split* data dan data Augmentasi.

4. Modelling

Model yang digunakan pada proses klasifikasi pengolahan data citra gambar buah manggis adalah *Convolutional Neural Network* dengan arsitektur *MobileNet V2*. Input yang diterima model berupa citra gambar buah manggis dengan *size* 244 x 244 piksel dengan 3 *channel* yaitu RGB (*Red, Green dan Blue*).

5. Evaluation

Proses evaluasi pada penelitian ini menggunakan *Confusion Matrix*, yang bertujuan untuk mengukur seberapa baik performa tingkat akurasi model yang sebelumnya telah dilakukan pelatihan dengan menggunakan dataset citra gambar buah manggis.

6. Deployment

Pada tahapapan ini merupakan proses pengaplikasian model yang telah dilatih dan di evaluasi performa tingkat akurasi kedalam sebuah aplikasi *realtime* untuk mendeteksi objek buah manggis yang bertujuan untuk mengklasifikasikan manggis yang akan diekspor dan dipasarkan di lokal.

IV. HASIL DAN PEMBAHASAN

Data Understanding

Data yang diperoleh berupa citra gambar dengan format *.jpg* yang diambil menggunakan kamera digital *Canon 1200D* beresolusi 18 MP. Dataset citra tersebut berisi 240 data dengan 2 katagori yang mana terdiri dari 120 untuk dataset gambar buah manggis yang bertujuan untuk diekspor, dan 120 untuk dataset gambar buah manggis yang bertujuan pemasaran lokal.



Gambar 7. Dataset Buah Manggis

Pada gambar 7 diatas merupakan beberapa sampel yang diambil dari dataset penelitian ini, 2 sampel di bagian atas menunjukkan sampel dengan kategori “Lokal”, sedangkan 2 sampel yang bawah menunjukkan sampel dengan kategori “Ekspor”.

Data Preparation

Dalam proses persiapan data dilakukan beberapa tahapan diantaranya pelabelan data, *Split* data dan augmentasi data. Sebelum dilakukan ketiga tahapan berikut dalam proses data preparation data akan terlebih dahulu dikumpulkan dalam sebuah folder, yang nantinya akan dipanggil dan dimasukan kedalam *tols jupiter notebook* dengan script perintah sebagai berikut.

```
[4]: sdir=r'C:/Users/user/SKRIPSI/Dataset/Skripsi'
```

Gambar 8. Script Perintah Upload Dataset

1. Labelling Data

Label ini akan terdiri dari dua kelas, "Lokal" dan "Ekspor", yang masing-masing akan dimasukkan ke dalam *variable array* yang disebut "*label*", seperti yang ditunjukkan pada gambar 9. *Script* yang digunakan untuk melakukan proses pemberian *label* pada masing - masing setiap dataset gambar buah manggis.

```
filepaths=[]
labels=[]
classlist=os.listdir(sdir)
for klass in classlist:
    classpath=os.path.join(sdir,klass)
    if os.path.isdir(classpath):
        flist=os.listdir(classpath)
        for f in flist:
            fpath=os.path.join(classpath,f)
            filepaths.append(fpath)
            labels.append(klass)
Fseries= pd.Series(filepaths, name='filepaths')
Lseries=pd.Series(labels, name='labels')
df=pd.concat([Fseries, Lseries], axis=1)
print (df.head())
print (df['labels'].value_counts())
```

Gambar 9. Script Proses Labelling Data

Output atau hasil dari *script* proses pemberian label pada dataset gambar ditunjukkan pada gambar 10.



Gambar 10. Dataset yang sudah diberi Label

2. Split Data

Proses tahapan persiapan data selanjutnya adalah pembagian data (*Split* Data), yang bertujuan untuk membagi data menjadi tiga bagian. Pertama, data digunakan untuk pelatihan sebesar 0,8 atau rasio 80% dari seluruh dataset yang ditampung ke dalam variabel *train_split* untuk melakukan pelatihan model. Kedua, data digunakan untuk pengujian sebesar 0,1 atau 10% yang ditampung ke dalam variabel *test_split* untuk melakukan pengujian model. Terakhir, data digunakan untuk validasi sebesar 0,1 atau 10% yang ditampung kedalam variabel *dummy_split*, untuk memvalidasi hasil model. *script* yang digunakan untuk membagi data, seperti yang ditunjukkan pada gambar 11.

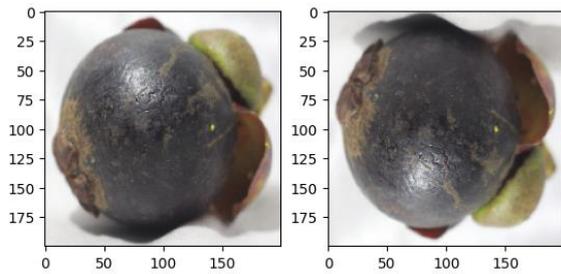
```
train_split=.8
test_split=.1
dummy_split=test_split/(1-train_split)
train_df, dummy_df=train_test_split(df, train_size=train_split, shuffle=True)
test_df, valid_df=train_test_split(dummy_df, train_size=dummy_split, shuffle=True)
print ('train df length: ', len(train_df), ' test df length: ', len(test_df), ' valid df length: ', len(valid_df))
```

Gambar 11. Script Proses Data Split

Dari 384 jumlah *dataset* yang yang telah dilakukan *split* data, sebanyak 307 dataset digunakan untuk pelatihan, 38 dataset untuk pengujian, dan 39 dataset untuk validasi.

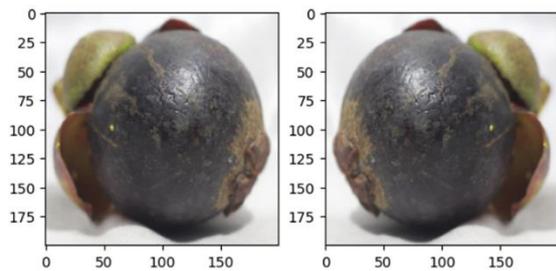
3. Data Augmentasi

Augmentasi data digunakan untuk memperbesar jumlah dataset asli dengan cara menduplikasi dataset asli kemudian mengubah tiruan dari dataset asli tersebut ke bentuk atau posisi yang berbeda dengan cara memutar gambar secara "*horizontal flip*" dan "*vertical flip*" serta pembesaran gambar dengan ukuran rasio *zoom* 0,2 atau 20%.



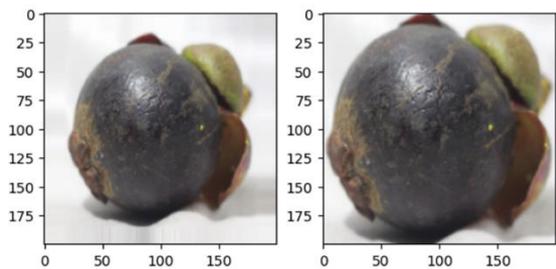
Gambar 12. Augmentasi Data dengan *Vertical Flip*

Pada gambar 12 menunjukkan hasil *output* dari proses augmentasi data dengan perintah *vertical_flip* atau merubah posisi asli menjadi posisi vertikal.



Gambar 13. Augmentasi Data dengan *Horizontal Flip*

Pada gambar 13 menunjukkan hasil *output* dari proses augmentasi data dengan perintah *horizontal_flip* atau merubah posisi asli menjadi posisi *horizontal*.



Gambar 14. Augmentasi Data dengan *Zoom Range*

Pada gambar 14 menunjukkan hasil *output* dari proses augmentasi data dengan perintah *zoom_range* atau merubah bentuk dataset asli atau memperbesar *object* dengan raisio 0 sampai dengan 0.2 atau 0-20%.

Modelling

Selanjutnya melakukan pelatihan terhadap model, yang dilakukan dengan menggunakan metode *transfer learning* dengan arsitektur *MobileNet V2* yang diperoleh dari library "*tensorflow.keras*".

1. Transfer Learning MobileNet V2

Proses pertama yang dilakukan pada tahapan *transfer learning* adalah men-*download* model arsitektur *MobileNet V2* yang terdapat pada library "*keras*". Pada gambar 15 merupakan *script* untuk melakukan proses *download* arsitektur *MobileNet V2*.

```
base_model_MNetV2 = tf.keras.applications.MobileNetV2(include_top=False, weights='imagenet', input_shape=img_shape)
base_model_MNetV2.trainable = False
```

Gambar 15. *Script* Proses *Download* Arsitektur *MobileNet V2*

Pada gambar 15 merupakan *script* proses *transfer learning* model arsitektur *MobileNet V2* yang *download* dari library "*keras*", yang ditampung kedalam variabel *base_model_MNetV2* dengan masukan *include_top = false* yang bertujuan untuk memodifikasi beberapa *layer* pada arsitektur *MobileNet V2*.

```
model_MobileNetV2 = tf.keras.Sequential([
    base_model_MNetV2,
    tf.keras.layers.Conv2D(filters=32, padding='same', kernel_size=3, activation='relu', strides=1),
    tf.keras.layers.MaxPool2D(pool_size=2, strides=2),
    tf.keras.layers.Dropout(rate=0.5),

    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(class_count, activation = 'softmax')
])
model_MobileNetV2.compile(optimizer=Adam(learning_rate=0.001), loss='categorical_crossentropy', metrics=['accuracy'])
```

Gambar 16. *Script* Proses Modifikasi *Layer* Arsitektur *MobileNet V2*

Pada gambar 16 merupakan proses untuk memodifikasi *layer* yang terdapat pada arsitektur *MobileNet V2*, dapat dilihat pada *layer Conv 2D* diberikan sebuah masukan dengan *filters* sebanyak 32 dengan *kernel size* sebanyak 3 dengan tujuan masukan *chanel* warna yaitu *Red*, *Green* dan *Blue* (RGB), *layer dense* dengan masukan hasil *output* dari *class_count* atau objek yang dideteksi yaitu sebanyak 2 *output*. optimasi yang digunakan adalah *Adam optimizer* dengan *learning rate* sebesar 0.001.

```

Model: "sequential"
Layer (type)                Output Shape                Param #
-----
mobilenetv2_1.00_160 (Func  (None, 5, 5, 1280)        2257984
conv2d (Conv2D)             (None, 5, 5, 32)          368672
max_pooling2d (MaxPooling2D) (None, 2, 2, 32)          0
dropout (Dropout)          (None, 2, 2, 32)          0
flatten (Flatten)           (None, 128)                0
dense (Dense)               (None, 2)                  258
-----
Total params: 2,626,914
Trainable params: 368,930
Non-trainable params: 2,257,984
    
```

Gambar 17. Output Modifikasi Arsitektur *MobileNet V2*

Pada gambar 17, modifikasi model arsitektur arsitektur *MobileNet V2* menghasilkan 2.626.914 total parameter, dengan 368.930 parameter yang dapat dilatih dan 2.257.984 parameter yang tidak dapat dilatih.

2. Training Model

Proses training model arsitektur *MobileNet V2* diberikan masukan *epoch* = 100 atau dilakukan perulangan pelatihan sebanyak 100 kali. Berikut pada gambar 18 merupakan hasil dari proses *training* model.



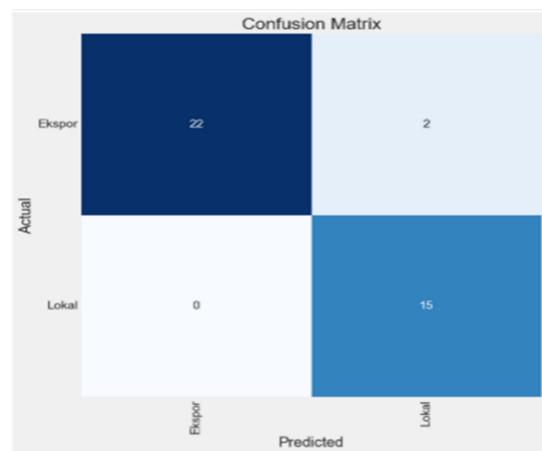
Gambar 18. Grafik *Training* Validasi *Loss* dan *Accuracy*

Hasil grafik yang terlihat di sebelah kiri pada Gambar 18 menunjukkan hasil *training* dan *validation loss*, dimana garis merah menunjukkan hasil nilai *training loss* setiap *epoch* atau iterasi pelatihan, dan garis hijau menunjukkan nilai *validation loss* setiap *epoch* atau iterasi. Titik biru mewakili *epoch point* terbaik dengan nilai penurunan *loss* terbaik, terdapat pada *epoch* ke-87 dengan nilai *val_loss* sebesar 0,0056. Hasil grafik sebelah kanan merupakan hasil *training* dan validasi *accuracy*, dimana garis merah menunjukkan hasil nilai akurasi *training accuracy* setiap *epoch* atau iterasi pelatihan dan garis hijau

mewakili nilai validasi *accuracy* setiap *epoch* atau iterasi, kemudian titik biru menunjukkan nilai terbaik pada setiap *epoch*. *Epoch point* terbaik dengan nilai validasi *accuracy* terbaik untuk setiap *epoch*, terletak pada *epoch* ke-2. *Epochs* dengan *val_accuracy* sebesar 1.0000. Nilai rata-rata akurasi hasil pelatihan model adalah 0,9500 yaitu. 95%.

Evaluation

Confusion matrix adalah persebaran dari data validasi(uji) ke dalam bentuk tabel matriks. Pada penelitian ini, *confusion matrix* ditampilkan sebagai 2 kelas dengan label prediksi pada sumbu X dan label yang benar pada sumbu Y. Sebuah gambar atau citra dapat disebut memprediksi dengan benar jika sumbu X dan Y bertemu pada tempat yang sama.

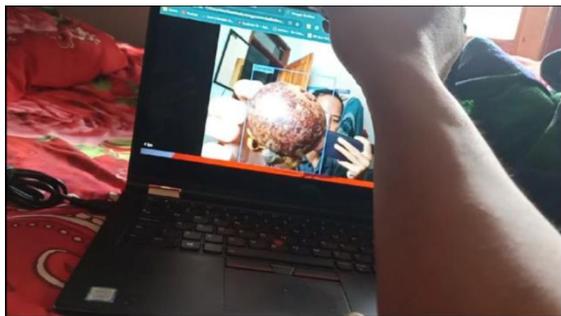


Gambar 19. *Confusion Matrix* Arsitektur *MobileNet V2*

Dapat dilihat pada gambar 19 diatas dari 24 dataset gambar buah manggis dengan label “Ekspor”, 22 dataset dengan label “Ekspor” dengan hasil prediksi *true* “Ekspor” dan 2 dataset dengan label “Ekspor” dengan prediksi *false* “Ekspor” atau “Lokal”. Selanjutnya dari 15 dataset gambar buah manggis dengan label “Lokal”, 15 dataset dengan label “Lokal” dengan hasil prediksi *true* “Lokal” dan 0 dataset dengan label “Lokal” dengan prediksi *false* “Lokal”.

Deployment

Model yang sudah berhasil dibuat dan mendapatkan nilai akurasi tinggi dan nilai *loss* rendah, tahapan selanjutnya model akan *diconvert* ke dalam format “.*pt*” untuk selanjutnya akan diimplementasikan ke dalam aplikasi. Aplikasi yang berhasil dibuat akan dilakukan uji coba menggunakan sample lain dengan mengambil gambar secara langsung atau secara *realtime*.



Gambar 20. Aplikasi Klasifikasi Buah Manggis

V. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan penelitian yang dilakukan dengan jumlah dataset sebesar 384 citra untuk 2 jenis manggis yaitu ekspor dan lokal. Dimana 307 citra digunakan untuk *training* dan 39 citra untuk *validation* dataset. Kemudian pada proses pengujian digunakan 38 citra yang diambil secara acak sebagai data *testing* untuk masing-masing citra buah manggis. Setelah dilakukan proses pelatihan diperoleh nilai akurasi terbaik dengan 307 citra untuk data pelatihan dan 39 citra untuk data pengujian sebesar 95%. Proses pelatihan dataset dilakukan dengan model *deep learning* menggunakan algoritma *convolutional neural network* dengan arsitektur yang digunakan adalah *MobileNet V2*, hasil dari proses *transfer learning* dari *library keras* dengan *framework Tensor flow*. Berdasarkan hasil dari pendeteksian jenis buah manggis ekspor dan lokal dengan metode *Convolutional Neural Network* menggunakan arsitektur *MobileNet V2* dapat dinilai bekerja dengan baik.

Saran

Untuk penelitian yang akan dilakukan selanjutnya, saran yang diajukan penulis yaitu melakukan pengembangan aplikasi dengan memanfaatkan sistem otomatisasi robotika.

DAFTAR PUSTAKA

- Anditto, R., & Roestam, R. (2022). Security Monitoring Using Improved Mobilenet V2 With Fine-Tuning To Prevent Theft in Residential Areas During the Covid-19 Pandemic. *SINTECH (Science and Information Technology) Journal*, 5(1), 87–94. <https://doi.org/10.31598/sintechjournal.v5i1.1023>
- Charli, F., Syaputra, H., Akbar, M., Sauda, S., & Panjaitan, F. (2020). Implementasi Metode Faster Region Convolutional Neural Network (Faster R-CNN) Untuk Pengenalan Jenis Burung Lovebird. *Journal of Information Technology Ampera*, 1(3), 185–197. <https://doi.org/10.51519/journalita.volume1.issue3.year2020.page185-197>
- Herdiansah, A., Borman, R. I., Nurnaningsih, D., Sinlae, A. A. J., & Al Hakim, R. R. (2022). Klasifikasi Citra Daun Herbal Dengan Menggunakan Backpropagation Neural Networks Berdasarkan Ekstraksi Ciri Bentuk. *JURIKOM (Jurnal Riset Komputer)*, 9(2), 388. <https://doi.org/10.30865/jurikom.v9i2.4066>
- Hermana, A. N., Husada, M. G., & Kurniawan, O. (2024). Penerapan SMOTE Untuk Mengatasi Data Imbalance pada Identifikasi Originalitas Sepatu Converse Menggunakan CNN Arsitektur VGG-16. 8, 10710–10722.
- Hidayat, D. (2022). Klasifikasi Jenis Mangga Berdasarkan Bentuk Dan Tekstur Daun Menggunakan Metode Convolutional neural Network (CNN). 5, 98–103.
- Iswavigra, D. U., Zen, L. E., & Hanim, H. (2023). Marketing Strategy UMKM Dengan CRISP-DM Clustering & Promotion Mix Menggunakan Metode K-Medoids. *Jurnal Informasi Dan Teknologi (JIDT)*, 5, 45–54. <https://doi.org/10.37034/jidt.v5i1.260>
- Luqman Hakim, Sari, Z., & Handhajani, H. (2021). Klasifikasi Citra Pigmen Kanker Kulit Menggunakan Convolutional Neural Network. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(2), 379–385. <https://doi.org/10.29207/resti.v5i2.3001>
- Narakusuma, M. A., Fauzi, A. M., & Firdaus, M. (2013). Rantai Nilai Produk Olahan Manggis. *Jurnal Manajemen & Agribisnis*, 10(1), 11–21.
- Nurarsyika Hesti, S. (2023). Perluasan Pasar Ekspor Produk Manggis Ke Korea Selatan Dengan Memanfaatkan IK-CEPA Pada CV A&H. *Jurnal Multidisiplin Indonesia*, 2(8), 2510–2521. <https://doi.org/10.58344/jmi.v2i9.471>
- Nyoman, P., & Putu Kusuma Negara. (2021). Deteksi Masker Pencegahan Covid19 Menggunakan Convolutional Neural Network Berbasis Android. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(3), 576–583. <https://doi.org/10.29207/resti.v5i3.3103>
- Pratama, A., Nurcahyo, A. C., & Firgia, L. (2023). Penerapan Machine Learning dengan Algoritma Logistik Regresi untuk Memprediksi Diabetes. *Prosiding CORISINDO 2023*, 116–121. <https://stmikpontianak.org/ojs/index.php/corisindo/article/view/30%0Ahttps://stmikpontianak.org/ojs/index.php/corisindo/article/download/30/22>
- Prima, A. (2023). Rancang Bangun Sistem Pendeteksi Aneka Ragam Buah Menggunakan MobileNetv2. *Jurnal Sistim*

- Informasi Dan Teknologi*, 5(2), 208–215.
<https://doi.org/10.60083/jsisfotek.v5i2.217>
- Saptana, N., Perwita, A. D., Darwis, V., & Suhartini, S. H. (2019). Dinamika Kelembagaan Kemitraan Usaha Rantai Pasok Buah Tropika Berorientasi Ekspor. *Forum Penelitian Agro Ekonomi*, 36(1), 45. <https://doi.org/10.21082/fae.v36n1.2018.45-61>
- Tampati, L. R., Ariessaputra, S., Yadnya, M. S., Elektro, J. T., Teknik, F., & Mataram, U. (2023). *Klasifikasi Kematangan Buah Manggis Berdasarkan Warna Dan Tekstur Menggunakan K-Nearest Neighbor [Classification Of Mangosteen Ripeness Based On Color And Texture Using K-Nearest Neighbor]*. 1–13.
- Utami, D. Y., Nurlalah, E., & Hasan, F. N. (2021). Comparison of Neural Network Algorithms, Naive Bayes and Logistic Regression to predict diabetes. *Journal of Informatics and Telecommunication Engineering*, 5(1), 53–64. <https://doi.org/10.31289/jite.v5i1.5201>