

Transfer Learning Model Pralatih MobileNetV2 dan DenseNet121 untuk Klasifikasi Tanaman Rempah

Yuditha Hatur Puspita dan Ahmad Sabri

Universitas Gunadarma

Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
E-mail: yhpuspita@gmail.com, sabri@staff.gunadarma.ac.id*

Abstrak

Tanaman rempah merupakan jenis rimpang (tanaman dengan batang menjalar yang terdapat di bawah tanah) dengan rasa dan aroma kuat yang digunakan dalam makanan sebagai bumbu, penambah rasa, dan bahan baku obat herbal. Rempah memiliki beragam bentuk, warna dan tekstur. Hal ini seringkali menyebabkan orang awam mengalami kesulitan dalam mengidentifikasi jenisnya. Dengan bantuan kecerdasan buatan, masalah tersebut diatasi dengan menggunakan model *Convolution Neural Network* (CNN) yang dilatih untuk mengidentifikasi jenis rempah berdasarkan citra. Untuk tujuan tersebut, penelitian ini menerapkan *transfer learning* terhadap dua model CNN pralatih (*pretrained CNN*) MobileNetV2 dan DenseNet121, yang dilatih kembali dengan dataset citra 5 kelas empon-empon (jahe, kencur, kunyit, lengkuas, temulawak). Kinerja kedua model tersebut diperbandingkan pada epoch ke 5, 10, 20 dan 30. Metrik evaluasi mencakup akurasi, *precision*, *recall*, dan *F1-score*. Hasil evaluasi menunjukkan bahwa dengan pelatihan sebanyak 30 *epoch*, kedua model memiliki *precision* yang tidak jauh berbeda, sedangkan DenseNet121 secara umum lebih unggul dibandingkan MobileNetV2 dalam hal akurasi, *precision* dan *F1-score*. Dalam progres pelatihan, MobileNetV2 mengalami peningkatan akurasi secara cepat pada *epoch* awal, namun pada *epoch* selanjutnya mengalami *overfitting*. Hal ini tidak terjadi pada DenseNet121, di mana akurasinya senantiasa meningkat secara konsisten sampai *epoch* ke 30. Secara umum, DenseNet121 memiliki kinerja yang lebih baik dibandingkan MobileNetV2 dalam mengenali kelima jenis rempah pada dataset empon-empon.

Kata kunci : Convolutional Neural Network, DenseNet121, Klasifikasi, MobileNet V2.

Pendahuluan

Indonesia merupakan negara yang memiliki tanah yang luas dan subur sehingga dapat ditumbuhi berbagai macam tanaman. Salah satu jenis tanaman yang tumbuh subur di Indonesia adalah tanaman rempah-rempah. Bahkan, kebun dan pekarangan rumah masyarakat di kepulauan nusantara sangat kaya dengan potensi rempah. Rempah-rempah adalah sejenis tanaman dengan rasa dan aroma kuat yang digunakan dalam makanan sebagai bumbu dan penambah rasa. Selain digunakan dalam masakan, rempah-rempah juga dapat digunakan sebagai bahan baku obat herbal. Rempah-rempah merupakan bagian tanaman yang berasal dari bagian batang, daun, kulit kayu, umbi, rimpang, akar, biji, bunga atau bagian-bagian tubuh tumbuhan lainnya.

Salah satu bagian tanaman rempah yang banyak digunakan adalah rimpang. Rimpang adalah modifikasi batang yang tertanam dalam tanah. Pada

bagian pucuk rimpang tanaman terdapat kuncup dan tunas-tunas tanaman. Rimpang adalah organ tumbuhan yang berfungsi dalam perbanyakan vegetatif tanaman. Rempah dan herba yang bermanfaat dalam bentuk rimpang dikenal sebagai kelompok empon-empon, yang meliputi antara lain kunyit, kencur, jahe, lengkuas, laos dan lainnya [1]. Karena manfaatnya, rempah-rempah merupakan komoditas yang memiliki nilai ekonomi tinggi [2].

Namun, banyaknya manfaat yang diberikan oleh rempah-rempah berbanding terbalik dengan pengetahuan masyarakat terhadap rempah-rempah itu sendiri. Hilangnya pengetahuan tradisional ini dipengaruhi oleh modernisasi yang menyebabkan penurunan minat generasi muda untuk mengenali berbagai jenis rempah-rempah [3].

Salah satu cara untuk mengatasi keadaan ini adalah dengan menerapkan model kecerdasan buatan yang dilatih untuk mengenali rempah berdasarkan input visual (melalui citra ataupun kamera video). Penerapan model kecerdasan bu-

DOI : <http://dx.doi.org/10.32409/jikstik.23.1.3502>,

*)Penulis Korespondensi

atan ini berguna untuk membantu seseorang yang awam dalam mengidentifikasi jenis rempah.

Model kecerdasan buatan untuk menjawab permasalahan ini adalah neural network. Model ini terinspirasi dari struktur otak manusia yang terdiri dari simpul saraf dan jaringan yang menghubungkannya. Model *neural network* terdiri dari lapisan simpul-simpul dan ruas-ruas. Simpul-simpul pada sebuah lapisan terhubung oleh ruas-ruas tersebut dengan simpul-simpul pada lapisan berikutnya. Pada model *neural network* dilakukan pelatihan berdasarkan sebuah dataset untuk mengenali pola pada dataset tersebut. Model yang sudah terlatih kemudian digunakan untuk keperluan pengambilan keputusan [4], antara lain untuk melakukan klasifikasi. *Neural network* juga memiliki keunggulan dalam melakukan ekstraksi fitur dibandingkan model yang tidak berbasis *neural network* [5].

Untuk melakukan klasifikasi berbasis citra, digunakan *Convolutional Neural Network* (CNN). Model ini memiliki ribuan sampai jutaan parameter, yang nilainya diperoleh dari proses pelatihan model [6]. Oleh karena jumlah parameter yang begitu masif, maka melatih CNN membutuhkan sumber daya komputasi yang besar dan waktu yang lama. Solusi dari masalah ini adalah dengan menggunakan model CNN yang sudah terlatih (*pre-trained CNN*), yaitu model CNN yang sudah dilatih sebelumnya dengan menggunakan komputer dengan kinerja tinggi. Model ini dapat dilatih kembali dengan dataset yang diinginkan. Proses melatih kembali model CNN pralatih dengan dataset lain disebut *transfer learning*. Proses *transfer learning* hanya mengubah parameter pada lapisan akhir (*top layer*) dari model CNN tersebut, sehingga biaya komputasi dan waktu menjadi jauh lebih rendah. Model CNN pralatih adalah antara lain InceptionV3 [7], VGG16 [8], AlexNet [9], MobileNetV2, [10], DenseNet121 [11].

Penelitian terkait klasifikasi rempah ataupun tanaman dengan menggunakan model kecerdasan buatan telah dilakukan sebelumnya. Pada penelitian [12] digunakan algoritma *Naïve Bayes* dengan *10-fold cross validation* untuk klasifikasi rempah jenis kunyit, temulawak, jahe dan lengkuas, dengan hasil akurasi sebesar 52%. Penelitian [13] menggunakan algoritma *Support Vector Machine* (SVM) untuk klasifikasi rempah. Model dilatih dengan 150 citra latih dan diuji dengan 30 citra uji, dan memberikan akurasi sebesar 78%. Penelitian [14] menggunakan model CNN untuk klasifikasi tingkat kematangan buah kurma. Model ini dilatih dengan 30.688 citra latih dan diuji dengan 6.368 citra uji. Hasil akurasi secara keseluruhan sebesar 96,98%.

Penelitian [15] melakukan transfer learning terhadap CNN pralatih AlexNet dan VGG16 untuk klasifikasi rempah. Dalam penelitian ini, VGG16 menghasilkan akurasi sebesar 81%, sedangkan AlexNet mengalami *overfitting*. Berikutnya, penelitian [16] melakukan klasifikasi melanoma (salah

satu jenis kanker kulit) dengan *transfer learning* CNN pralatih MobileNetV2, ResNet50V2, InceptionV3 dan InceptionResNetV2. Dalam penelitian ini, MobileNetV2 memberikan akurasi 85%, tertinggi di antara model lainnya yang diuji. Pada penelitian [17] dilakukan perbandingan kinerja MobileNet, VGG-19, Resnet50V2, DenseNet121, dan NASNetMobile. Model-model tersebut dilatih dengan 3 dataset berbeda, yaitu Wayang Indonesia, Drone-Bird dan daun anggur. Dari lima arsitektur tersebut diperoleh bahwa bahwa arsitektur DenseNet121 memberikan skor evaluasi tertinggi.

Berdasarkan uraian latar belakang di atas, maka untuk menjawab permasalahan pengenalan jenis rempah, penelitian ini mengusulkan penggunaan model kecerdasan buatan, dengan melakukan *transfer learning* terhadap MobileNetV2 dan DenseNet121. Kedua model tersebut dipilih karena beberapa hasil penelitian terdahulu mencatat keunggulan kinerja kedua model ini terhadap model lainnya [16, 17]. Selain itu, penelitian ini juga mengusulkan perbandingan kinerja MobileNetV2 dan DenseNet121 dalam melakukan klasifikasi rempah.

Metode Penelitian

Tahapan dalam penerapan *transfer learning* terhadap model pralatih MobileNetV2 dan DenseNet121 serta evaluasi terhadap kinerja kedua model tersebut untuk pengenalan jenis rempah mencakup akuisisi, praproses, splitting, transfer learning, dan evaluasi. Berikut uraian detail dari tahapan-tahapan tersebut.

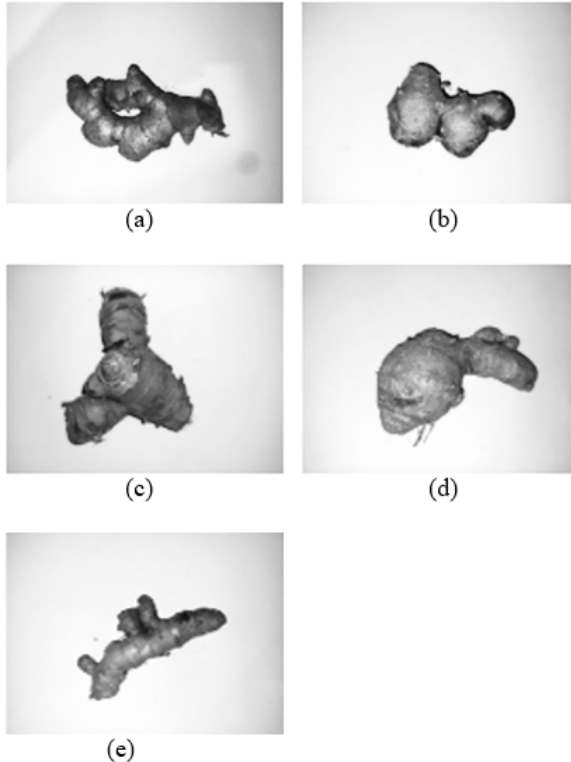
1. Akuisisi dataset

Dataset pada penelitian ini adalah dataset dari 5 kelas empon-empon, yaitu jahe, kencur, kunyit, lengkuas, temulawak. Penelitian ini menggunakan dataset yang tersedia pada laman <https://www.kaggle.com/datasets/owenlie/empon-dataset>. Setiap instance pada dataset tersebut merupakan citra RGB salah satu dari kelima jenis empon-empon di atas, dengan dimensi $3472 \times 4624 \times 3$. Secara keseluruhan, dataset memuat 638 citra empon-empon-empon. Beberapa sampel citra pada dataset diberikan pada Gambar 1.

2. Praproses dataset

Pada tahap ini dilakukan *image resizing* dan *dataset augmenting*. *Image resizing* dilakukan untuk mereduksi dimensi matriks citra, dari yang semula berukuran $3472 \times 4624 \times 3$ menjadi $244 \times 244 \times 3$. Hal ini dilakukan agar dimensi citra sama dengan dimensi input layer dari MobileNetV2 dan DenseNet121, yang berukuran $244 \times 244 \times 3$. Reduksi dimensi ini juga berguna untuk mengurangi penggunaan memori dan mempercepat proses pelatihan.

Dataset augmenting dilakukan untuk menambah *instance* pada dataset agar diperoleh variasi yang lebih beragam, sehingga meningkatkan akurasi model. Metode augmenting yang digunakan pada penelitian ini adalah random rotation untuk memutar gambar dengan sudut tertentu secara acak, dan random flip untuk membalik orientasi gambar, baik secara horizontal maupun secara vertikal.



Gambar 1: Sampel dataset empon-empon: jahe (a), kencur (b), lengkuas (c), temulawak (d), kunyit (e).

3. Dataset Splitting

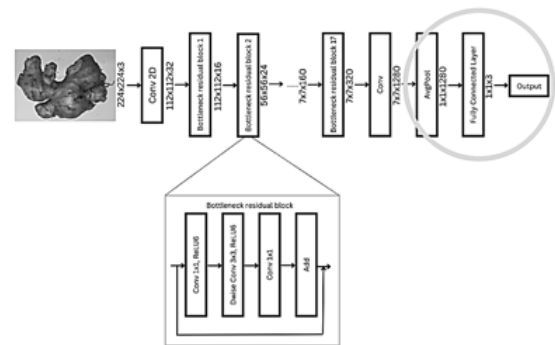
Pada tahapan ini, dataset dialokasikan menjadi 3 peruntukan, yaitu pelatihan, validasi dan pengujian. Penelitian ini menerapkan rasio 80% untuk data latih, 10% untuk data validasi dan 10% untuk data uji.

4. Transfer learning model CNN pralatih

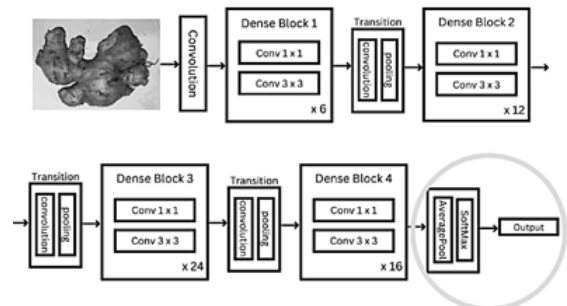
Model pralatih MobileNetV2 dan DenseNet121 memiliki arsitektur sebagaimana ditunjukkan pada Gambar 2 dan Gambar 3. Secara keseluruhan, kedua model tersebut tersusun dari 2 bagian *layer* yaitu *feature learning* dan *classification layer* atau *top layer* (bagian yang diberi lingkaran pada Gambar 2 dan Gambar 3). Pada *transfer learning*, modifikasi dilakukan pada *top layer*. Pada penelitian ini, *top layer* didefinisikan sebagai *average pooling layer*, dirangkai dengan 5 *dense layer* dengan koefisien *dropout* 0.2 dan fungsi aktivasi *softmax*. Penetapan

5 *dense layer* merujuk pada 5 kelas pada *dataset* empon-empon. Sampel kode bahasa pemrograman *Python* untuk modifikasi *top layer* diberikan pada Gambar 4.

Transfer learning dilakukan dengan menggunakan perangkat komputer dengan CPU Intel i7-8550 dengan RAM 8GB, sistem operasi Windows 10 Pro 64 bit, bahasa pemrograman *Python* versi 3.11 yang berjalan pada Google Colab Pro. Hiperparameter yang digunakan adalah optimizer Adam dengan *batch size* 64, *learning rate* 10^{-3} dan dengan 30 *epoch*. Untuk loss function yang digunakan adalah *categorical crossentropy*.



Gambar 2: Arsitektur MobileNetV2.



Gambar 3: Arsitektur DenseNet121.

```
global_average_layer = tf.keras.layers.GlobalAveragePooling2D()
feature_batch_average = global_average_layer(feature_batch)
prediction_layer = tf.keras.layers.Dense(num_classes, activation='softmax')
prediction_batch = prediction_layer(feature_batch_average)

inputs = tf.keras.Input(shape=(224, 224, 3))
x = data_augmentation(inputs)
x = preprocess_input(x)
x = base_model(x, training=False)
x = global_average_layer(x)
x = tf.keras.layers.Dropout(0.2)(x)
outputs = prediction_layer(x)
model = tf.keras.Model(inputs, outputs)

base_learning_rate = 0.001
model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=base_learning_rate)
              loss=tf.keras.losses.CategoricalCrossentropy(),
              metrics=['accuracy'])

history = model.fit(train_dataset,
                    epochs=5,
                    validation_data=validation_dataset,
                    )
```

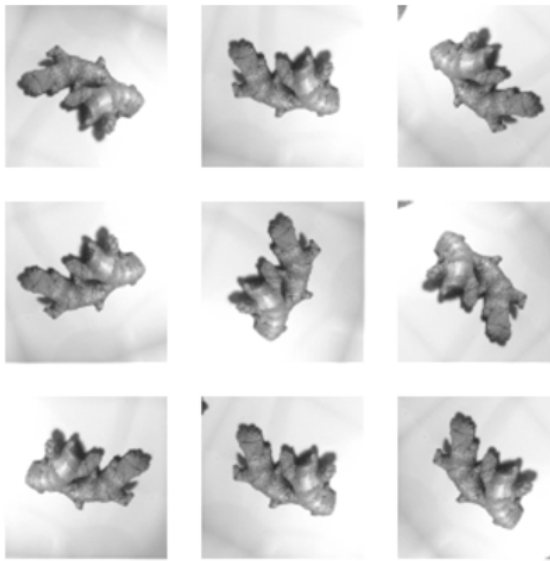
Gambar 4: Kode python untuk modifikasi top layer.

5. Evaluasi model

Evaluasi dilakukan berdasarkan kinerja model pada dataset uji. Metrik pengukuran kinerja mencakup akurasi, *precision*, *precision* dan *F1-score*. Evaluasi mencakup perbandingan nilai metrik dari pengujian kedua model.

Hasil dan Pembahasan

Setelah akuisisi data, dilakukan *image resizing* dan *dataset augmenting*. Gambar 5 menampilkan sampel *dataset augmenting* dengan transformasi *random rotation* dan *random flip*.



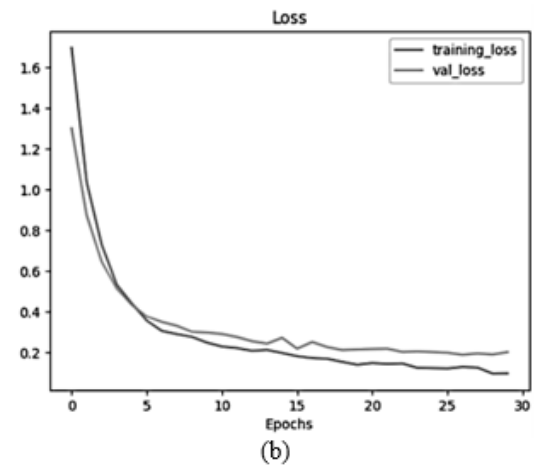
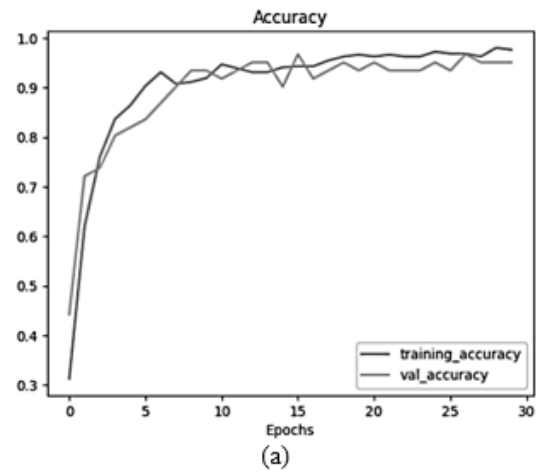
Gambar 5: Sampel hasil augmentasi sebuah citra jahe dengan transformasi *random flip* dan *random rotation*.

Setelah praproses data, dilakukan *transfer learning* pada model MobileNetV2 dan DenseNet121. Perbandingan kinerja *transfer learning* kedua model dilakukan untuk 5, 10, 20, dan 30 *epoch*.

Model MobileNetV2 memiliki trainable parameter sebanyak 2.223.872, sedangkan untuk model DenseNet121 memiliki trainable parameter sebanyak 6.953.856. Jumlah parameter tersebut berbanding lurus terhadap total durasi pelatihan model. Hasil dari pelatihan model disajikan secara grafis pada Gambar 6 dan Gambar 7, sedangkan Rekapitulasi hasil pelatihan diberikan pada Tabel 1.

Pada pelatihan MobileNetV2, akurasi pelatihan dan validasi meningkat dengan cepat pada iterasi awal, untuk kemudian cenderung melandai pada iterasi selanjutnya (Gambar 6a). Demikian pula sebaliknya dengan *training loss* dan *validation loss*, di mana besarnya menurun dengan cepat pada iterasi awal. Untuk kemudian melandai pada iterasi berikutnya. Terdapat kecenderungan *overfitting* karena selisih antara *training loss* dengan *validation loss*

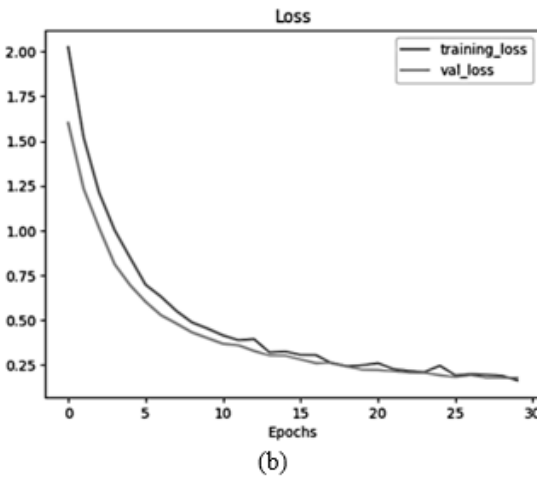
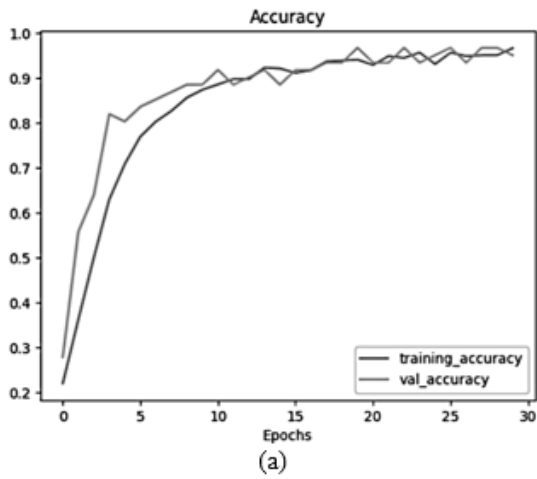
loss yang semakin besar seiring dengan penambahan *epoch* (Gambar 6b).



Gambar 6: Grafik akurasi (a) dan *loss* (b) dari *transfer learning* MobileNetV2 dengan 30 *epoch*.

Pada pelatihan DenseNet121, besaran akurasi juga meningkat dengan cepat, namun tidak secepat pada MobileNetV2 (Gambar 7a). Demikian pula dalam penurunan nilai *training loss* dan *validation loss* yang tidak secepat MobileNet2 (Gambar 7b). Berbeda dengan MobileNetV2, pada validasi tidak terlihat kecenderungan *overfitting*. Dalam hal ini bahan *training loss* dan *validation loss* cenderung konvergen pada angka yang sama.

Secara umum, kedua model memperlihatkan progres pelatihan dan validasi secara normal, di mana akurasi meningkat seiring dengan penambahan *epoch*. Fenomena *overfitting* terjadi pada MobileNetV2.



Gambar 7: Grafik akurasi (a) dan *loss* (b) dari *transfer learning* DenseNet121 dengan 30 *epoch*.

Rekapitulasi hasil *transfer learning* pada Tabel 1 memberikan hasil yang lebih detail secara kuantitatif. Dalam hal akurasi pelatihan, MobileNetV2 unggul terhadap DenseNet121 pada keempat *checkpoint* pengamatan, yaitu *epoch* ke 5, ke 10, ke 20, dan ke 30. Namun dalam hal akurasi validasi, MobileNetV2 hanya unggul pada *epoch* ke 5 dan ke 20.

Tabel 1: Rekapitulasi Hasil *Transfer Learning* MobileNetV2 dan DenseNet121.

<i>Epoch</i>	Akurasi Pelatihan	Akurasi Validasi	Training Loss	Val. Loss	Durasi (detik)
<i>MobileNetV2</i>					
5	90,75%	86,89%	0,3727	0,4335	51,381
10	93,70%	90,16%	0,2096	0,2710	100,933
20	96,85%	96,72%	0,1216	0,1574	206,074
30	97,64%	95,08%	0,0987	0,2031	1999,205
<i>DenseNet121</i>					
5	84,25%	83,61%	0,6949	0,6168	191,334
10	92,52%	90,16%	0,3902	0,3736	200,177
20	95,87%	95,08%	0,1939	0,2178	357,027
30	96,65%	95,08%	0,1639	0,1783	2155,791

Dalam hal *training loss*, MobileNetV2 unggul jauh terhadap DenseNet121 pada semua *epoch checkpoint*, karena nilainya selalu lebih kecil. Na-

mun dalam *validation loss*, pada MobileNetV2 terjadi *bouncing value* pada *epoch* ke 20, di mana nilai *loss* kembali naik. Hal ini mengindikasikan terjadinya *overfitting* pada MobileNetV2. Di sisi lain, hal ini tidak terjadi pada *validation loss* dari DenseNet 121, yang nilainya semakin kecil secara konsisten.

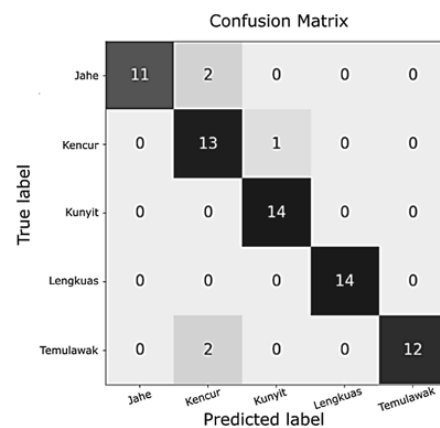
Dalam hal durasi pelatihan, MobileNetV2 selalu lebih cepat dibandingkan DenseNet121 untuk semua *epoch*.

Akurasi hasil pengujian diberikan oleh Tabel 2. Dalam pengujian model dengan data uji, MobileNetV2 memberikan akurasi yang lebih baik dibandingkan DenseNet121 pada tiga *epoch checkpoint* pertama (5, 10 dan 20), namun terjadi penurunan akurasi akibat *overfitting* pada *epoch* ke 30.

Tabel 2: Akurasi Pengujian.

<i>Epoch</i>	MobileNetV2	DenseNet121
5	86%	83%
10	90%	88%
20	96%	93%
30	93%	94%

Selanjutnya dilakukan evaluasi berdasarkan metrik presisi, *precision* dan skor F1. Untuk mendapatkan nilai-nilai metrik tersebut diperlukan nilai-nilai matriks konfusi dari hasil pengujian model, sebagaimana diberikan oleh Gambar 8. Dari matriks konfusi tersebut dapat diketahui bahwa dari pengujian model MobileNetV2 berdasarkan 69 citra empon-empon, terdapat 5 citra yang diklasifikasikan secara keliru. Dalam hal ini, dua citra jahe diklasifikasikan sebagai kencur, satu citra kencur sebagai kunyit, dan dua citra temulawak sebagai kencur. Nilai untuk metrik presisi, *precision* dan skor F1 untuk mode MobileNetV2 diberikan pada Tabel 3.

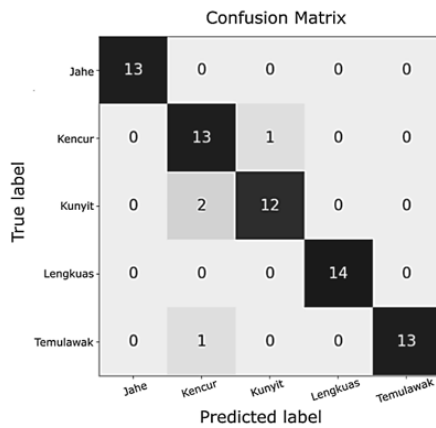


Gambar 8: Matriks konfusi dari model MobileNetV2 dengan 30 *Epoch*.

Tabel 3: Metrik *precision*, *precision*, F1-score dari model MobileNetV2 dengan 30 *epoch*.

	<i>Precision</i>	<i>Precision</i>	F1-score
Jahe	1,00	0,85	0,92
Kencur	0,76	0,93	0,84
Kunyit	0,93	1,00	0,97
Lengkuas	1,00	1,00	1,00
Temulawak	1,00	0,86	0,92
Akurasi		0,93	

Untuk pengujian menggunakan model DenseNet121 dengan 30 *epoch*, matriks konfusi yang diperoleh diberikan oleh Gambar 9.



Gambar 9: Matriks konfusi dari model DenseNet 121 dengan 30 *Epoch*.

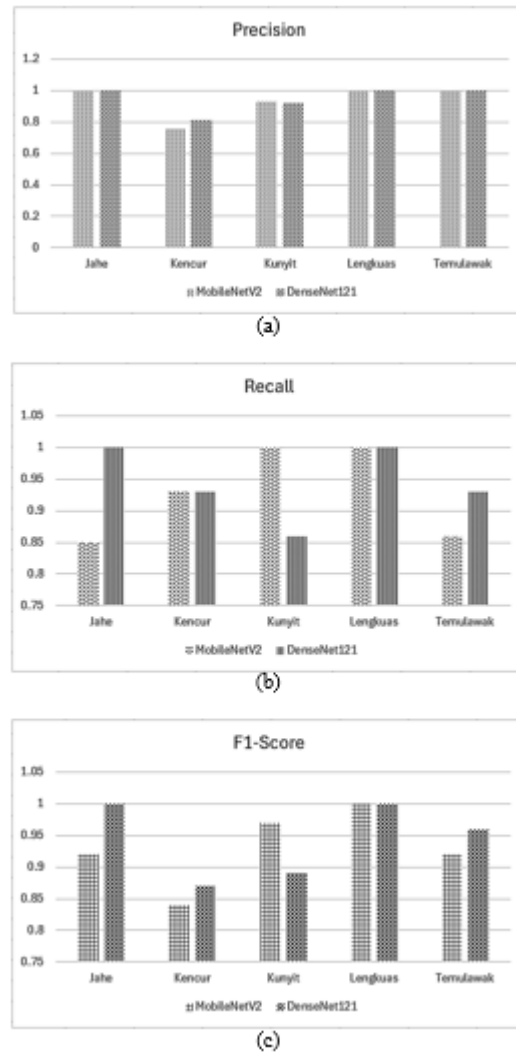
Dari matriks konfusi tersebut dapat diketahui bahwa dari pengujian model DenseNet121 berdasarkan 69 citra empon-empon, terdapat 4 citra yang diklasifikasikan secara keliru. Dalam hal ini, satu citra kencur diklasifikasikan sebagai kunyit, dua citra kunyit sebagai kencur, dan satu citra temulawak sebagai kunyit. Nilai untuk metrik presisi, *precision* dan F1-score untuk mode DenseNet121 diberikan pada Tabel 4.

Tabel 4: Metrik *precision*, *precision*, F1-score dari model DenseNet121 dengan 30 *epoch*.

	<i>Precision</i>	<i>Precision</i>	F1-score
Jahe	1,00	1,00	1,00
Kencur	0,81	0,93	0,87
Kunyit	0,92	0,86	0,89
Lengkuas	1,00	1,00	1,00
Temulawak	1,00	0,93	0,96
Akurasi		0,94	

Rekapitulasi perolehan metrik secara grafis diberikan oleh Gambar 10. Untuk metrik *precision* (Gambar 10a), secara umum kedua model menunjukkan kinerja yang hampir sama. Seluruh citra yang diprediksi sebagai jahe, lengkuas, dan temulawak, memiliki ketepatan 100%. Hal ini ditunjukkan oleh nilai *precision* untuk ketiga

jenis rempah tersebut sebesar 1. Prediksi model DenseNet121 untuk citra kencur memiliki presisi yang lebih baik daripada MobileNetV2. Dalam hal memprediksi citra kunyit, kedua model memiliki kinerja yang hampir sama.



Gambar 10: Grafik rekapitulasi perolehan metrik *precision* (a), *recall* (b), F1-score (c) untuk MobileNetV2 dan DenseNet121.

Untuk metrik *precision*, kedua model memiliki keunggulannya masing-masing. Berdasarkan Gambar 10b, DenseNet121 dapat mengenali jahe dan temu lawak lebih baik dari MobileNetV2. Sebaliknya, MobileNetV2 mengenali kunyit lebih baik dari Dense121. Namun dalam mengenali kencur dan lengkuas, kedua model memiliki kemampuan yang setara.

Untuk metrik F1-score (Gambar 10c), model DenseNet121 memiliki nilai 1 untuk kelas jahe dan lengkuas. Hal ini menunjukkan bahwa model mampu mengenali dan memprediksi dengan ketepatan 100% untuk kedua kelas tersebut. DenseNet121 juga lebih mampu mengenali dan memprediksi dengan akurasi yang lebih baik dibandingkan MobileNetV2 untuk kelas kencur dan

temulawak. Untuk *F1-score*, MobileNetV2 hanya unggul dalam mengenali dan memprediksi dengan akurasi yang lebih baik untuk kelas kencur.

Penutup

Penelitian ini melakukan *transfer learning* untuk model pralatih MobileNetV2 dan DenseNet121 untuk klasifikasi rempah jenis empon-empon. Berdasarkan evaluasi kinerja kedua model tersebut, *transfer learning* DenseNet121 berlangsung lebih lama dibandingkan MobileNetV2. Namun secara umum, untuk pelatihan sebanyak 30 *epoch*, DenseNet121 lebih unggul dibandingkan MobileNetV2 dalam hal akurasi, *precision* dan *F1-score*. Sedangkan dalam hal *precision*, kedua model menunjukkan kinerja yang tidak jauh berbeda. Dalam progres pelatihan, MobileNetV2 mengalami peningkatan akurasi secara cepat pada *epoch* awal, namun menunjukkan gejala *overfitting* pada *epoch* selanjutnya. Hal ini tidak terjadi pada DenseNet121, di mana akurasinya senantiasa meningkat secara konsisten sampai *epoch* ke 30. Secara umum, DenseNet121 memiliki kinerja yang lebih baik dibandingkan MobileNetV2 dalam mengenali kelima jenis rempah pada dataset empon-empon.

Untuk penelitian selanjutnya disarankan untuk melakukan *fine-tuning* terhadap hiperparameter agar mendapatkan hasil yang lebih baik, menambah jumlah dataset dan jenis rempah-rempah, menggunakan model pralatih lainnya dan mengimplementasikan model yang sudah dilatih pada suatu aplikasi *desktop/mobile* sehingga dapat digunakan untuk berbagai keperluan nyata.

Daftar Pustaka

- [1] Luchman Hakim, "Rempah dan Herba Kebun-Pekarangan Rumah Masyarakat: Keragaman, Sumber Fitofarmaka dan Wisata Kesehatan-kebugaran", Diandra, Yogyakarta, 2015.
- [2] Intan Sari Tanjung, Samsi Efendi dan Dini Andriani Nasution, "Strategi Kementerian Perdagangan Dalam Meningkatkan Ekspor Rempah Indonesia Dalam Upaya Meningkatkan Perekonomian Indonesia", Jurnal Ilmu Komputer, Ekonomi dan Manajemen (JIKEM), Vol. 2 No. 2, pp. 3651-3656, 2022.
- [3] Adnan, Zidni Ilman Navia, Mega Silvia, Maulida Antika dan Adi Bejo Suwardi, "Diversity of herbs and spices plants and their importance in traditional medicine in the South Aceh District, Indonesia", Biodiversitas Journal of Biological Diversity, Vol. 23 No.7, pp. 3836-3843. 2022.
- [4] Belal A. M. Ashqar, Bassem S. Abu-Nasser and Samy S. Abu-Naser, "Plant Seedlings Classification Using Deep Learning", International Journal of Academic Information Systems Research (IJAIRS), Vol. 3 No. 1, pp. 7-14, 2019.
- [5] Jing Wei Tan, Siow-Wee Chang, Sameem Abdul-Kareem, Hwa Jen Yap and Kien-Thai Yong, "Deep Learning for Plant Species Classification using Leaf Vein Morphometric", IEEE/ACM Transactions on Computational Biology and Bioinformatics, Vol. 17 No.1, pp. 82-90, 2018.
- [6] David Argüeso, Artzai Picon, Unai Irusta, Alfonso Medela, Miguel G San-Emeterio, Arantza Bereciartua and Aitor Alvarez-Gila, "Few-Shot Learning approach for plant disease classification using images taken in the field", Computers and Electronics in Agriculture, Vol.175 No.1, pp. 1-8, 2020.
- [7] C. Szegedy et al., "Going Deeper with Convolutions," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9, DOI: 10.1109/CVPR.2015.7298594, 2015.
- [8] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in 3rd International Conference on Learning Representations (ICLR) 2015, pp. 1–14, Sep. 2015.
- [9] Krizhevsky, Alex; Sutskever, Ilya; Hinton, Geoffrey E., "ImageNet classification with deep convolutional neural networks" (PDF). Communications of the ACM. 60 (6): 84–90. doi:10.1145/3065386. ISSN 0001-0782. S2CID 195908774, 2017.
- [10] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, pp. 4510–4520. doi: 10.1109/CVPR.2018.00474, Jun. 2018.
- [11] G. Huang, Z. Liu, L. Van Der Maaten and K. Weinberger, "Densely Connected Convolutional Networks," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017 pp. 2261-2269. doi: 10.1109/CVPR.2017.243
- [12] Nadya P. Batubara, Didit Widiyanto dan Nurul Chamidah, "Klasifikasi Rempah Rimpang Berdasarkan Ciri Warna RGB Dan Tekstur GLCM Menggunakan Algoritma Naive Bayes", Jurnal Informatik, Vol. 16 No. 3, pp. 156-163, 2020.
- [13] Saniyatul Mawaddah, Mohammad Robihul Mufid, Arif Basofi and Agung Fiyanto, "Rhizome Image Classification Using Support Vector Machine", In International Conference on

- Applied Science and Technology on Social Science 2021 (iCAST-SS 2021), Vol. 647 No.1, pp. 990-993, 2021.
- [14] Amin Nasiria, Amin Taheri-Garavandb and Yu-Dong Zhang, "Image-based deep learning automated sorting of date fruit". *Postharvest Biology and Technology*, Vol. 153 No. 1, pp. 133-141, 2019.
- [15] Evan Tanuwijaya and Angelica Roseanne, "Classification of Indonesian Spices Digital Image using Modified VGG 16 Architecture", *Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, Vol. 21 No. 1, pp. 189-196, 2021.
- [16] Rarasmaya Indraswaria, Rika Rokhanab and Wiwiet Herulambang, "Melanoma image classification based on MobileNetV2 network". *Procedia Computer Science*, Vol. 197, pp. 198-207, 2022.
- [17] Muhammad Daffa Arviano Putra, Tawang Sahro Winanto, Retno Hendrowati, Aji Primajaya and Faisal Dharma Adhinata, "A Comparative Analysis of Transfer Learning Architecture Performance on Convolutional Neural Network Models with Diverse Datasets", *Komputika: Jurnal Sistem Komputer*, Vol. 12 No. 1, pp. 1-11, 2023.