

Implementasi Transfer Learning Dalam Klasifikasi Penyakit Pada Daun Teh Menggunakan MobileNetV2

I Gede Diva Dwijayana^{a1}, I Gede Arta Wibawa^{a2}

^aProgram Studi Informatika, Fakultas Matematika dan Ilmu
Pengetahuan Alam Universitas Udayana
Badung, Bali, Indonesia
1dwijayanaigede@gmail.com
2gede.arta@unud.ac.id

Abstract

Indonesia is currently the seventh largest tea producer in the world. However, tea farmers in Indonesia still use old technology and simple farming methods. Limited knowledge of small farmers about diseases that can attack their tea leaves. a tool is needed to classify the types of tea leaf diseases using digital images. This research uses a deep learning approach with Convolutional Neural Network using MobileNetV2 architecture for tea leaf disease classification with digital image data. With the transfer learning method, the MobileNetV2 model is trained with 3 different epochs. The best accuracy is obtained from the model trained with epoch 20 with an accuracy value of 94.6% with a loss value of 0.287. The MobileNetV2 model that has been trained shows good results in classifying tea leaf diseases.

Keywords: MobileNetV2, Klasifikasi Citra, Daun Teh, Deep Learning, Transfer Learning

1. Pendahuluan

Indonesia saat ini adalah produsen teh terbesar ketujuh di dunia. Teh Indonesia yang diekspor terutama berasal dari perkebunan-perkebunan besar di negara ini, baik yang dimiliki negara maupun swasta yang biasanya menghasilkan teh bermutu tinggi atau premium, sementara mayoritas petani kecil lebih berorientasi kepada pasar domestik, karena teh yang dihasilkan berkualitas lebih rendah dan karenanya memiliki harga penjualan yang lebih murah. Petani-petani kecil juga masih menggunakan teknologi lama dan metode-metode pertanian yang sederhana, biasanya tidak memiliki fasilitas pengolahan[1]. Untuk meningkatkan kualitas dan kuantitas teh yang dihasilkan, petani-petani kecil ini harus dapat mengawasi kesehatan teh mereka dari serangan penyakit yang menjangkit daun teh. Beberapa hama dan penyakit mengganggu pertumbuhan dan kinerja teh budidaya[2]. Jamur merupakan kelompok utama patogen penyebab penyakit pada tanaman teh. Umumnya penyakit daun mempengaruhi hasil panen secara langsung, seperti *grey blight*, *brown blight* dan banyak penyakit daun lainnya [2]. Para petani kecil memiliki sedikit pengetahuan untuk mengenali penyakit yang dapat menyerang daun teh.

Teknologi informasi telah banyak berkembang, khususnya dalam pengolahan data citra. Klasifikasi citra merupakan salah satu implementasi dari pengolahan data citra. Klasifikasi citra merupakan proses pengelompokan *pixel-pixel* dan *element* data citra digital menjadi kategori-kategori. Setiap citra digital dikenali melalui fitur yang ada pada citra tersebut sehingga bisa dikelompokkan menjadi kategori tertentu. Teknologi klasifikasi citra dapat dimanfaatkan untuk mengenali penyakit daun teh dengan data citra digital.

Penelitian-penelitian terkait tentang klasifikasi citra yang pernah dilakukan sebelumnya antara lain : Penelitian yang dilakukan oleh Nur Ibrahim, dkk pada tahun 2022 yang melakukan klasifikasi tingkat kematangan pucuk daun teh dengan pendekatan *deep learning* arsitektur VGG19 dan ResNet50 yang mendapatkan akurasi terbaik dari model VGG19 yaitu sebesar 97,5%[3];

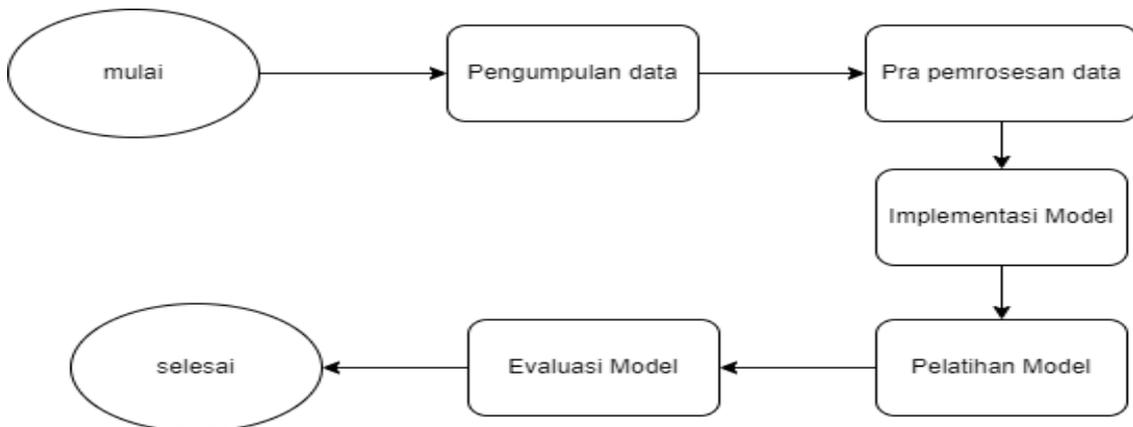
penelitian yang dilakukan oleh Faisal Mashuri dan Ultach Enri pada tahun 2022 yaitu melakukan deteksi penyakit daun gandum yang diujikan dengan model InceptionV3, MobileNetV2, VGG16, ResNet101V2, DenseNet201 dengan hasil akurasi dari InceptionV3 yaitu 100%, akurasi dari MobileNetV2 sebesar 100%, VGG16 sebesar 76%, ResNet100 dengan hasil akurasi 100%, dan DenseNet201 dengan akurasi sebesar 95%[4]; penelitian yang dilakukan oleh Herwina, dkk pada tahun 2022 yang membuat deteksi penyakit tanaman padi menggunakan MobileNet berbasis aplikasi *android* yang mendapat akurasi 95%[5];.

Berdasarkan permaparan dan penelitian yang telah dilakukan sebelumnya , maka penulis melakukan penelitian terhadap klasifikasi penyakit pada daun teh dengan pendekatan *deep learning* menggunakan *Convolutional Neural Network* arsitektur MobileNetV2. Pada penelitian ini dipilih arsitektur MobileNetV2 karena berdasarkan penelitian sebelumnya hasil akurasi dari arsitektur MobileNetV2 lebih tinggi dibandingkan dengan beberapa arsitektur lain.

2. Metode Penelitian

Metode yang akan digunakan dalam penelitian ini adalah metode *transfer learning* untuk mengklasifikasi penyakit pada daun teh menggunakan arsitektur MobileNetV2. *Transfer learning* adalah menggunakan model yang sudah dilatih (*pre-trained*) dengan data latih yang besar lalu digunakan kembali untuk ekstraksi fitur pada sebuah *dataset* baru dengan kelas yang baru[5].

2.1. Alur Penelitian

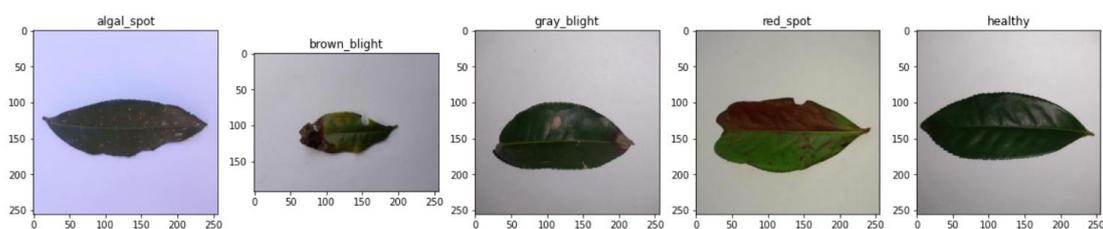


Gambar 1. Alur Penelitian

Pelaksanaan penelitian dimulai dengan pengumpulan data yaitu data gambar daun teh sesuai dengan jenis penyakitnya dan gambar daun teh yang sehat. Selanjutnya data tersebut akan melalui pra pemrosesan untuk normalisasi data dan menyesuaikan ukuran gambar agar dapat digunakan untuk melatih Model. Kemudian implementasi model dengan mengkonfigurasi model CNN arsitektur MobileNetV2 dengan beberapa parameter. Selanjutnya pelatihan model untuk dapat mengklasifikasikan penyakit daun teh dengan data gambar. Setelah model tersebut dilatih, akan dilakukan evaluasi pada model tersebut untuk mengetahui tingkat akurasi dari hasil prediksi model terhadap data gambar daun teh.

2.2. Pengumpulan Data

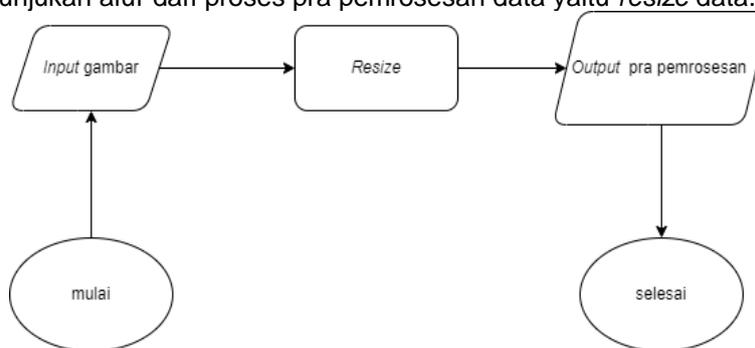
Data yang digunakan pada penelitian ini adalah data *open source* dengan 5867 data gambar yang diambil dari portal Kaggle (Tea_Leaf_Disease) [6]. Data yang diambil dari sumber data terbagi menjadi 6 kategori tetapi dalam penelitian ini hanya 5 kategori yang akan digunakan sehingga jumlah data berkurang menjadi 4867 data. Kategori yang digunakan yaitu *algal spot*, *brown blight*, *gray blight*, *red spot* dan *healthy*. Pada kategori *algal spot*, *gray blight*, *red spot* dan *healthy* terdapat 1000 data. Pada kategori *brown blight* terdapat 867 data.



Gambar 2. Contoh data gambar dari masing-masing kategori

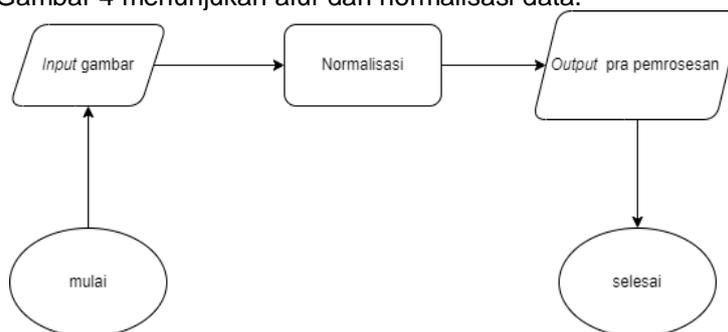
2.3. Pra Pemrosesan Data

Pra pemrosesan data yang dilakukan adalah mengatur ulang ukuran atau dimensi dari data gambar yang didapat dari sumber data, agar data bisa digunakan untuk tahap pelatihan model. Pada arsitektur model MobileNetV2 dimensi data yang bisa diinputkan pada model adalah data gambar berdimensi (224, 224). Ukuran dari data yang telah didapat tidak merata. Terdapat data yang ukurannya lebih besar atau lebih kecil dari standard input MobileNetV2 yaitu (224, 224) Gambar 3 menunjukkan alur dari proses pra pemrosesan data yaitu *resize* data.



Gambar 3. Alur *Resize* data

Selain melakukan *resize* data, langkah lainnya dari pra pemrosesan data adalah normalisasi data. Normalisasi adalah proses untuk membuat beberapa variabel memiliki rentang nilai yang sama, tidak ada yang terlalu besar maupun terlalu kecil sehingga dapat membantu dalam pelatihan model agar hasil pelatihan dari model menjadi lebih baik. Normalisasi data pada data gambar salah satunya adalah mengalikan setiap *pixel* pada *image* dengan nilai 1/255 (255 adalah nilai *max* dari *pixel*). Gambar 4 menunjukkan alur dari normalisasi data.



Gambar 4. Alur Normalisasi data

2.4. Implementasi Model

Model yang digunakan untuk *transfer learning* pada penelitian ini adalah MobileNetV2 yang merupakan arsitektur *deep learning Convolutional Neural Network*. Inti dari *Convolutional Neural Networks (CNN)* adalah lapisan *convolutional* yang dapat mengekstrak fitur lokal (misalnya *edge*) melintasi gambar input melalui konvolusi. Setiap *node* dalam lapisan *convolutional* terhubung ke *subset* kecil dari *neuron* yang terhubung secara spasial[4]. MobileNet memang diciptakan untuk aplikasi *mobile* dan *embedded system* yang menggunakan *depthwise separable convolutions*.

DSC terdiri dari dua lapisan, pertama adalah *depthwise convolution* yang menerapkan filter tunggal pada setiap saluran *input*, sedangkan lapisan kedua adalah *pointwise convolution* yang akan membuat kombinasi linier dari *output* lapisan *depthwise*[5]. MobileNetV2 adalah modifikasi pertama dari MobileNet. Perbedaan MobileNetV1 dengan MobileNetV2 adalah penggunaan *inverted residual blocks* dan *linear bottleneck*. Dalam *bottleneck* terdapat input dan output antara model sementara lapisan dalam merangkum kemampuan model untuk mengubah input dari tingkat yang lebih rendah (yaitu piksel) ke deskriptor dengan tingkat yang lebih tinggi[5].

2.5. Pelatihan Model

Model yang dilatih dengan tambahan parameter sebagai berikut :

1. Jumlah *epoch* adalah 20, 10 dan 5
2. *Batch size* 32
3. Menggunakan *dropout* sebesar 0.5
4. Menambahkan 1024 *Fully-Connected Layer* dibawah *layer pre-trained* model MobileNetV2
5. *Layer* untuk fungsi aktivasi yang digunakan adalah *rectified linear unit*(RELU)
6. *Optimizer* yang digunakan adalah Adam
7. *Loss function* yang digunakan adalah *categorical crossentropy*
8. *Metrics* yang digunakan adalah *accuracy*
9. Menggunakan *softmax* sebagai fungsi aktivasi dari *layer* terakhir.

2.6. Evaluasi Model

Evaluasi model bertujuan untuk mengukur tingkat hasil kinerja dari model yang sudah dilatih. Model hasil latih diuji menggunakan *metrics* evaluasi *accuracy*, dan *loss*. *Accuracy* didefinisikan sebagai tingkat kebenaran dari suatu model untuk mengklasifikasikan data dengan benar. *Loss* adalah nilai dari ketidaktahuan model mengenali data [3].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Loss = \frac{FP + FN}{TP + TN + FP + FN} \quad (2)$$

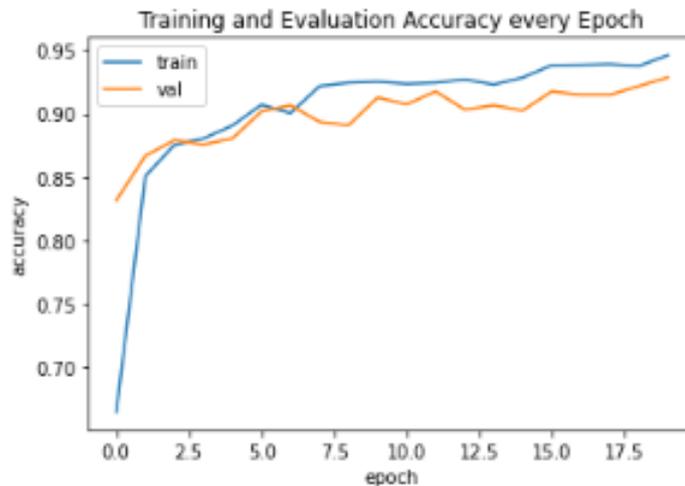
Keterangan :

1. *True Positive*(TP) adalah data yang tepat diklasifikasikan oleh model sebagai nilai positif atau benar.
2. *True Negative* adalah data yang tepat diklasifikasi oleh model sebagai nilai negative atau salah.
3. *False positive* adalah data salah namun diklasifikasikan sebagai data yang benar
4. *False Negative* adalah data yang benar namun diklasifikasikan salah sebagai data salah[3].

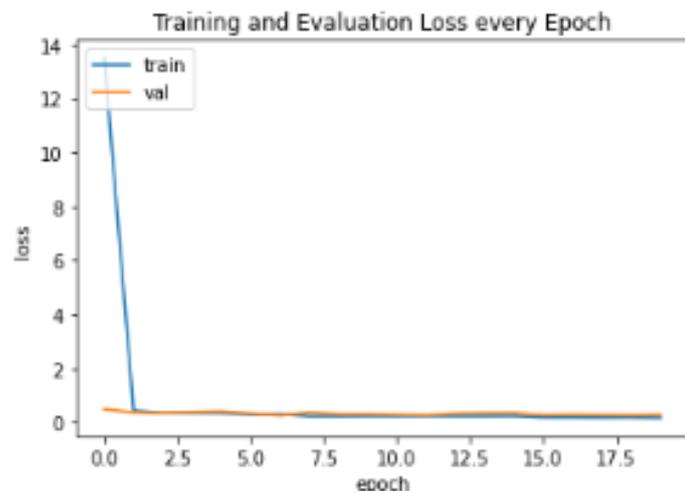
3. Hasil dan Diskusi

Pelatihan model deep learning arsitektur MobileNetV2 dengan data gambar yang telah dikumpulkan. Tahap pertama adalah membagi data menjadi data *training* dan data validasi. Perbandingan dari data *training* dan data validasi adalah 0,7 dari keseluruhan data untuk data *training* dan 0.3 dari keseluruhan data untuk data validasi. Pada setiap direktori dari data *training* dan data validasi data tersebut sudah dikelompokan berdasarkan kategorinya. Setelah data dibagi lalu data tersebut akan melalui proses normalisasi dan juga *resize* menjadi (224,224). Proses normalisasi bertujuan untuk membuat beberapa variabel memiliki rentang nilai yang sama, tidak ada yang terlalu besar maupun terlalu kecil sehingga dapat membuat pelatihan model menjadi lebih baik. Setiap piksel pada data dikalikan dengan 1./255. Pelatihan model dilakukan 3 kali dengan *epoch* 20, 10 dan 5. Hasil dari pelatihan model dengan *epoch* 20 yaitu nilai akurasi

pada iterasi pertama sebesar 66,6% dan untuk nilai *loss* sebesar 13,506. Pada iterasi kedua nilai akurasi naik menjadi 85% dan nilai *loss* turun secara signifikan menjadi 0,4387. Pada iterasi selanjutnya nilai akurasi bertambah perlahan dan begitu juga dengan nilai *loss* yang berkurang secara perlahan. Pada iterasi terakhir atau iterasi ke-20 didapatkan nilai *loss* sebesar 0,1683 dan nilai akurasi sebesar 94,6%.



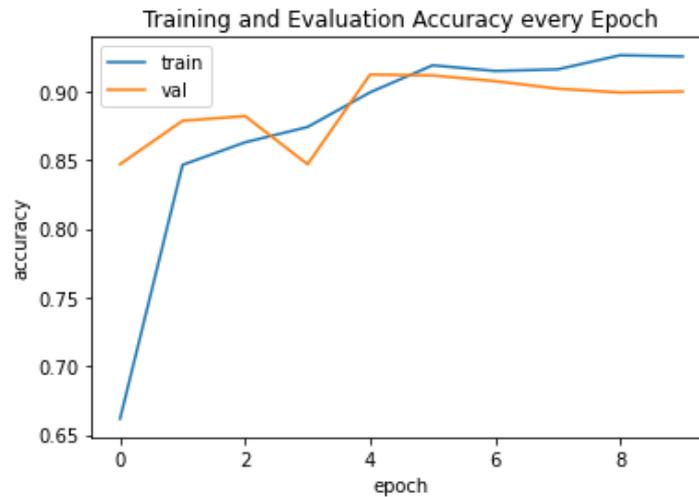
Gambar 5. Diagram *accuracy epoch 20*



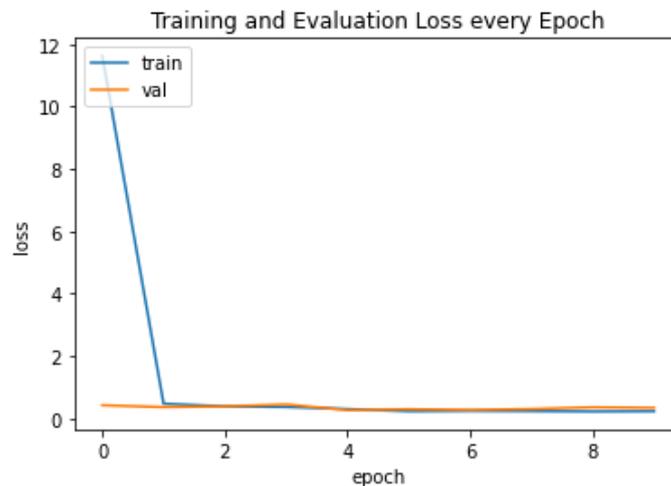
Gambar 6. Diagram *loss epoch 20*

Hasil akurasi yang didapat dari pelatihan model dengan *epoch 20* jika dievaluasi dengan data validasi adalah 92,8% dan nilai *loss* adalah 0,287. Dapat dilihat dari Gambar 9, garis biru menunjukan data *train* dan garis orange menunjukan data validasi. *Accuracy* naik secara signifikan pada *epoch* awal dan naik perlahan pada *epoch* selanjutnya. Nilai *loss* turun drastis pada *epoch* awal dan turun sangat perlahan sampai *epoch* pelatihan selesai.

Selanjutnya hasil dari pelatihan model dengan *epoch 10* yaitu nilai akurasi pada iterasi pertama sebesar 66,9% dan untuk nilai *loss* sebesar 11,3130. Pada iterasi kedua nilai akurasi naik menjadi 83,7% dan nilai *loss* turun secara signifikan menjadi 0,4863. Pada iterasi selanjutnya nilai akurasi bertambah perlahan dan begitu juga dengan nilai *loss* yang berkurang secara perlahan. Pada iterasi ke-10 didapatkan nilai *loss* sebesar 0,2612 dan nilai akurasi sebesar 91,3%.



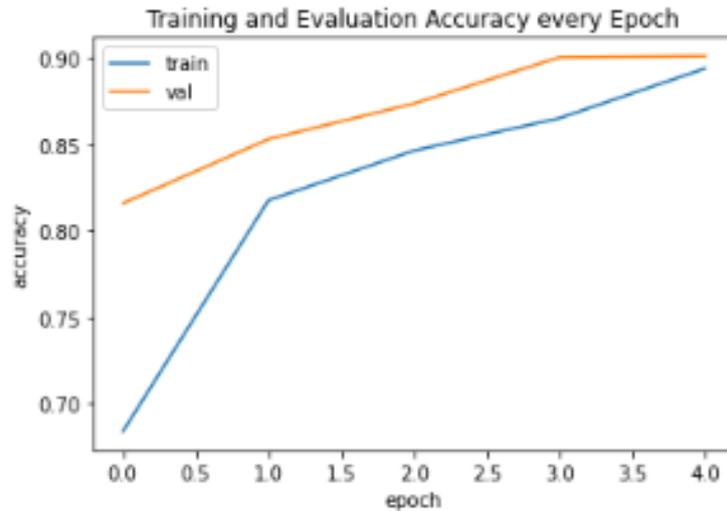
Gambar 7. Diagram accuracy epoch 10



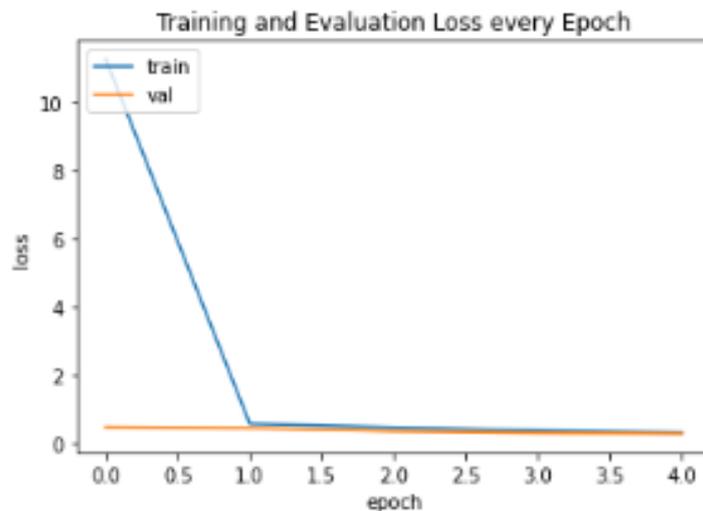
Gambar 8. Diagram loss epoch 10

Hasil akurasi yang didapat dari pelatihan model dengan *epoch* 10 jika dievaluasi dengan data validasi adalah 91,4% dengan nilai *loss* sebesar 0,319. Dibandingkan dengan model sebelumnya yang dilatih dengan *epoch* 20, hasil latihan model dengan *epoch* 10 menunjukkan sedikit penurunan nilai akurasi dan sedikit kenaikan nilai *loss*, tetapi model dengan *epoch* 10 masih memiliki akurasi yang tinggi dan nilai *loss* yang sedikit. Pada diagram menunjukkan bahwa kenaikan akurasi secara signifikan terjadi di *epoch* awal. Penurunan nilai *loss* juga terjadi secara signifikan pada *epoch* awal.

Hasil dari pelatihan model dengan *epoch* 5 yaitu nilai akurasi pada iterasi pertama sebesar 68,4% dan untuk nilai *loss* sebesar 11,2515. Pada iterasi kedua nilai akurasi naik menjadi 81,7% dan nilai *loss* turun secara signifikan menjadi 0,5622. Pada iterasi selanjutnya nilai akurasi bertambah perlahan dan begitu juga dengan nilai *loss* yang berkurang secara perlahan. Pada iterasi ke-5 didapatkan nilai *loss* sebesar 0,3025 dan nilai akurasi sebesar 89,3%.



Gambar 9. Diagram accuracy epoch 5



Gambar 10. Diagram loss epoch 5

Dari hasil evaluasi model dengan data validasi didapatkan akurasi 90% dan nilai *loss* 0,279 pada epoch 5. Jika dievaluasi dengan data validasi, model dengan epoch 5 memiliki akurasi paling sedikit dibandingkan dengan model dengan epoch 20 dan 10. Nilai *loss* pada model dengan epoch 5 justru lebih kecil dibandingkan dengan model dengan epoch 10.

Nilai akurasi dari model MobileNetV2 yang dilatih menggunakan epoch 20, 10 dan 5 dengan pembagian data *training* 70% dan data validasi adalah 30% dari total data dapat dilihat pada tabel dibawah ini. Dari hasil analisis dari tabel, dapat diketahui bahwa jumlah epoch berpengaruh pada tingkat akurasi. Semakin banyak jumlah epoch maka semakin tinggi akurasi yang didapat.

Tabel 1. Nilai Akurasi dan Loss Data Training dan Validasi dari Setiap Percobaan Epoch

Epoch	Akurasi Data Training (%)	Loss Data Training	Akurasi Data Validasi (%)	Loss Data Validasi
20	94,6%	0,1683	92,8%	0,2870
10	91,3%	0,2512	91,4%	0,3199
5	89,3%	0,3025	90%	0,2792

4. Kesimpulan

Berdasarkan hasil evaluasi dari pelatihan model MobileNetV2 dengan 3 percobaan *epoch* yang berbeda, menunjukkan bahwa akurasi tertinggi didapat dari model dengan *epoch* 20 dengan nilai akurasi 94,6% dan nilai *loss* terendah yaitu 0,2870. Penggunaan model arsitektur MobileNetV2 dalam mengklasifikasikan penyakit daun teh dengan data gambar, memberikan hasil yang baik dengan penambahan beberapa parameter dan *convolutional layer* yang telah digunakan. Adapun perbandingan hasil akurasi penggunaan arsitektur ini dengan lainnya seperti VGG16 yang terdapat pada penelitian serupa[4] mengindikasikan bahwa mesin klasifikasi penyakit daun teh dengan menggunakan pendekatan *deep learning* dengan arsitektur MobileNetV2 sudah cukup baik dalam melakukan klasifikasi. Hal ini dibuktikan dengan nilai akurasi model yang dibangun yang telah mendekati model dengan akurasi tertinggi yang dibangun dengan arsitektur berbeda pada penelitian sejenis.[3][4][5]

References

- [1] Indonesia Investments, "Teh", 22 November 2015.[Online]. Available: <https://www.indonesia-investments.com/id/bisnis/komoditas/teh/item240>. [Access on 27 September 2022]
- [2] A.K. Pandey, G.D. Sinniah, A. Babu and A. Tanti. "How The Global Industry Copes With Fungal Diseases - Challenges and Opportunities". 2021 The American Phytopathological Society, vol. 105, no. 7, p 1868-1879, 2021
- [3] N. Ibrahim, et al. "Klasifikasi Tingkat Kematangan Pucuk Daun Teh Menggunakan Metode Convolutional Neural Network". ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika, vol. 10, no. 1, p. 162-176, 2022
- [4] F. Mashuri and U. Enri. "Implementasi Transfer Learning Dalam Mendeteksi Penyakit Pada Gandum". Jurnal Nuansa Informatika, vol. 16, no. 1, p 66-77, 2022
- [5] Herwina, Darmatasia, A. K. A. Shiddiq and T. D. Syahputra. "Deteksi Penyakit pada Tanaman Padi Menggunakan MobileNet Transfer Learning Berbasis Android". AGENTS: Journal of Artificial Intelligence & Data Science, vol. 2, no. 2, p. 1-8, 2022
- [6] S. Datta, "Tea_Leaf_Disease", 22 May 2020. [Online]. Available: <https://www.kaggle.com/datasets/saikatdatta1994/tea-leaf-disease> . [Accessed on 25 September 2022]
- [7] R. O. Ekoputris, "MobileNet: Deteksi Objek pada Platform Mobile", 9 May 2018.[Online]. Available: <https://medium.com/nodeflux/mobilenet-deteksi-objek-pada-platform-mobile-bbbf3806e4b3> . [Access on 27 September 2022]
- [8] N. Rochmawati, dkk. "Analisa Learning rate dan Batch size Pada Klasifikasi Covid Menggunakan Deep learning dengan Optimizer Adam". Journal Information Engineering and Educational Technology, vol. 05, no. 02, p. 44-48, 2021
- [9] Q. Lina, "Apa itu Convolutional Neural Network?", 2 Januari 2019.[Online]. Available: <https://medium.com/@16611110/apa-itu-convolutional-neural-network-836f70b193a4> . [Access on 28 September 2022]
- [10] R.R. Allaam and A.T. Wibowo. "Klasifikasi Genus Tanaman Anggrek Menggunakan Metode Convolutional Neural Network (CNN)". e-Proceeding of Engineering, vol. 8, no. 2, p. 1153-1189, 2021
- [11] A. Fuadi and A. Suharso. "Perbandingan Arsitektur Mobilenet Dan Nasnetmodile Untuk Klasifikasi Penyakit Pada Citra Daun Kentang". JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika), vol. 07, no. 03, 2022
- [12] M.F. Supriadi, E. Rachmawati and A. Arifianto. "Pembangunan Aplikasi Mobile Pengenalan Objek Untuk Pendidikan Anak Usia Dini". Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK), vol. 8, no. 2, p. 357-364, 2019
- [13] M. Basorudin, A. Rizqi, S. Murdaningrum and W. Maharani. "Kajian Persebaran Komoditas Teh: Pengembangan Kawasan Perkebunan Teh di Provinsi Jawa Barat Tahun 2015". Jurnal Sosial Ekonomi Pertanian, vol. 15, no. 5, p. 205-214, 2019
- [14] E.I. Haksoro and A. Setiawan. "Pengenalan Jamur Yang Dapat Dikonsumsi Menggunakan Metode Transfer Learning Pada Convolutional Neural Network". Jurnal ELTIKOM : Jurnal Teknik Elektro, Teknologi Informasi dan Komputer, vol. 5, no. 2, p. 81-91, 2021
- [15] N. Hardi. "Komparasi Algoritma MobileNet Dan Nasnet Mobile Pada Klasifikasi Penyakit Daun Teh". Reputasi: Jurnal Rekayasa Perangkat Lunak, vol. 3, no. 1, p. 50-56, 2022

Halaman ini sengaja dikosongkan