

Klasifikasi Biji Kopi Arabika Menggunakan *Convolutional Neural Network*

1st Nailul Fikri Nasution
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

nailulfikrinst@student.telkomuniversity.ac.id

2nd Casi Setianingsih
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

setiacasie@telkomuniversity.ac.id

3rd Randy Erfa Saputra
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

resaputra@telkomuniversity.ac.id

Abstrak — Biji kopi Arabika memiliki rasa dan bentuk yang berbeda, menjadikannya salah satu komoditas kopi paling berharga di dunia. Proses memilih biji kopi dikenal sebagai *sortasi* dalam industri kopi. Identifikasi dan klasifikasi biji kopi dapat menjadi tugas yang sulit dan memakan waktu untuk dilakukan secara manual. Tugas akhir ini menggunakan *Convolutional Neural Network* (CNN) dengan arsitektur *MobileNetV2*. Dengan kinerja yang baik, biji kopi di klasifikasikan menggunakan arsitektur *MobileNetV2*. *Hyperparameter* yang terdiri dari *epoch*, *batch size*, dan *learning rate* akan di optimalkan untuk meningkatkan kinerja model CNN. Tujuan dari tugas akhir ini adalah untuk menemukan *hyperparameter* yang ideal yang akan memberikan kinerja yang baik pada model CNN. Digunakan empat kelas biji kopi di antaranya *Premium*, *Longberry*, *Peaberry*, dan Cacat yang akan diklasifikasi berdasarkan bentuk dan warna dari masing-masing biji kopi. Percobaan sudah dilakukan, menunjukkan bahwa penentuan *hyperparameter* sangat memengaruhi kinerja model. Dengan *learning rate* 0.0001, *batchsize* 16, dan *epoch* 30 mendapatkan nilai akurasi 88,19%, presisi 96,74%, *recall* 89%, dan *f1-score* 92,71%.

Kata kunci— Biji Kopi, Arabika, Klasifikasi, Sortasi, CNN, *MobileNetV2*

I. PENDAHULUAN

Karena kopi adalah buah yang menyegarkan, orang mengonsumsinya untuk memenuhi kebutuhan hidup mereka. Produksi kopi global telah meningkat dengan cepat dari abad ke-20 hingga saat ini. Beberapa wilayah Asia dan Afrika, serta Amerika Tengah dan Amerika Selatan adalah tempat asal kopi. Indonesia juga adalah negara penghasil kopi terbaik, menghasilkan 3–4% dari produksi kopi global. Sebagian besar perkebunan kopi di Indonesia masih dimiliki oleh perkebunan rakyat [1].

Menurut data terbaru yang dikumpulkan oleh Gabungan Eksportir Kopi Indonesia (GAEKI), perkebunan kecil atau kopi milik rakyat menyumbang 121 juta hektar dari 1,26 juta hektar kopi yang diproduksi di Indonesia. Selain itu, pengelolaan pasca panen yang konvensional dan keterbatasan teknologi adalah masalah besar. Akibatnya, sangat sulit untuk mengharapkan kualitas kopi yang sama, dan kopi Indonesia memiliki kualitas yang cukup rendah [1].

Salah satu proses yang diperlukan untuk membuat kopi siap minum adalah *sortasi* biji kopi. Proses *sortasi* biji kopi untuk memenuhi standar berdasarkan ukuran, bentuk, berat jenis, sifat permukaan, dan warna. *Sortasi* manual memakan waktu yang lebih lama dan menyebabkan karyawan lelah lebih banyak, sehingga mengurangi produktivitas. Bagi pembeli, keseragaman ukuran, bebas dari unsur pengotor, dan bebas dari cacat kecil adalah kualitas biji kopi yang paling penting [2]. Proses ini dilakukan setelah proses penjemuran dan proses *Hulling*.

Deep learning adalah bagian dari pembelajaran mesin yang berkaitan dengan algoritma yang meniru struktur dan fungsi otak, yang disebut jaringan saraf tiruan, yang dapat digunakan untuk mengolah data ini. Berbagai bidang telah melakukan banyak penelitian tentang *deep learning*. Musik juga terkait dengan sosial media, ekonomi, dan bahkan emosi [3].

Deep learning memiliki banyak *Framework* salah satu yang sering digunakan yaitu Pytorch [4]. Dalam *Framework* Pytorch, terdapat beberapa model yang bisa digunakan, salah satunya adalah *MobileNetV2*. *MobileNetV2* adalah arsitektur yang dikembangkan berdasarkan *Convolutional Neural Network* (CNN) [5]. Model CNN menunjukkan kinerja yang baik, namun peningkatan lebih lanjut dapat dicapai dengan menyesuaikan *hyperparameter* untuk mengoptimalkan model. *Hyperparameter* seperti *epoch*, *Batch Size*, *Learning Rate*, dan *Optimizer* biasanya disesuaikan untuk meningkatkan performa model CNN. [6].

Dalam tugas akhir ini, penulis berkonsentrasi pada klasifikasi biji kopi hijau dengan menggunakan metode *Convolutional Neural Network* (CNN) dengan arsitektur *MobileNetV2*. Selama proses pelatihan model, penulis mencoba mengubah *hyperparameter* seperti *epoch*, *batch size*, dan *learning rate*. Untuk mengetahui tingkat akurasi, presisi, *recall*, dan *f1-score*, penulis menggunakan *confusion matrix*.

Data yang digunakan dalam Tugas Akhir ini ialah data yang di dapat dari Universitas Syiah Kuala Aceh, merupakan *dataset* publik yang di beri nama USK-Coffee. Dalam *dataset* tersebut terdapat empat kelas biji kopi hijau jenis arabika yang memiliki bentuk yang berbeda di antaranya *Premium*, *Longberry*, *Peaberry*, dan Cacat [7].

Implementasi dan manfaat dari identifikasi biji kopi berdasarkan bentuk di harapkan dapat mempermudah identifikasi biji kopi arabika melalui *sortasi* otomatis menggunakan metode klasifikasi CNN dengan arsitektur yang digunakan yaitu *MobileNetV2*.

II. DASAR TEORI

A. Biji Kopi Arabika

Kopi Arabika, yang asal-usulnya dapat ditelusuri Kembali ke hutan-hutan dataran tinggi Ethiopia, telah berkembang menjadi varietas kopi yang paling banyak diminati di dunia, terutama karena profil rasa yang superior dan aroma yang khas. Proses domestiknya dimulai di Ethiopia, kemudian penyebarannya melintasi semenanjung Arab ke Yaman, membuka pintu bagi ekspansi global melalui jalur perdagangan dan ekspedisi penjelajahan [8].

Varietas Arabika, dengan keragaman genetiknya yang luas, telah mengadaptasi karakteristik unik di berbagai mikroklimat, menghasilkan spektrum rasa yang luas dan memperkaya palet kopi dunia. Namun, varietas ini menghadapi tantangan signifikan dari perubahan iklim yang mengancam habitat alaminya, penyakit seperti karat daun kopi, dan masalah keberlanjutan yang mempengaruhi produksi [8].

B. Klasifikasi Biji Kopi Berdasarkan Bentuk

Klasifikasi biji kopi berdasarkan bentuk adalah salah satu metode untuk menilai kualitas dan karakteristik biji kopi. Berbagai bentuk biji kopi menunjukkan asal-usul, varietas, dan proses pasca panen yang berbeda, yang semua ini berpengaruh terhadap rasa akhir dari kopi. Dalam tugas akhir ini, terdapat 4 kelas biji kopi antara lain:

1. Premium

Biji Kopi *Semi washed* adalah istilah populer untuk biji kopi *premium*. Proses pencucian setengah setelah panen adalah teknik yang paling sering digunakan di kalangan petani kopi di Sumatra, Indonesia.

biji kopi *premium* memiliki warna biru-hijau dan rasa nikmat yang mengingatkan tentang rumput segar dan cendawan gula. Karakteristik ini juga menunjukkan hasil pemrosesan yang lebih baik. Biji kopi berkualitas *premium* memiliki bentuk yang lebih besar dan bulat [9].



GAMBAR 1
Biji Kopi Premium

2. Longberry

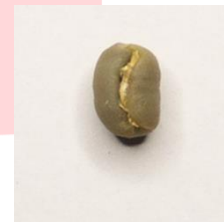
Kopi Arabika *Longberry* merupakan salah satu varietas biji kopi spesifik yang dibedakan dari bentuk fisiknya yang memanjang, berbeda dengan bentuk fisik kopi pada umumnya. Variasi khusus ini memberikan rasa pedas yang khas dan aroma rempah yang nyata. Meskipun secara keseluruhan karakternya lebih ringan, kopi Arabika *Longberry* menunjukkan profil yang seimbang dan mengingatkan pada rasa herbal yang biasa ditemukan pada kopi Aceh. [10].



GAMBAR 2
Biji Kopi Longberry

3. Peaberry

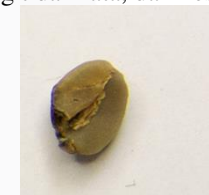
Dibandingkan dengan biji kopi lainnya, biji *Peaberry* berukuran lebih kecil, bulat, dan padat. Karena hanya 3% hingga 5% biji kopi pada pohon saat dipetik, dan biji kopi *Peaberry* sangat dicari. Hanya 5-kilogram biji kopi dipanen dari setiap 200-kilogram biji kopi. Petani sering memilih *Peaberry* langsung dari seluruh produksi mereka karena harga yang ditawarkan lumayan. *Peaberry* juga disebut kopi *lanang*. [11].



GAMBAR 3
Biji Kopi Peaberry

4. Cacat

Disebabkan oleh kesalahan dalam pengolahan atau dimakan oleh serangga, biji kopi yang rusak dapat dikenali oleh bentuknya yang tidak utuh, potongannya yang pecah-pecah, ukurannya yang tidak rata, dan kesan lubang [12].



GAMBAR 4
Biji Kopi Cacat

C. Deep Learning

Deep learning adalah algoritma pembelajaran mesin yang terinspirasi oleh struktur otak manusia dan memiliki kemampuan *feature engineering* untuk merekayasa fitur secara otomatis sehingga tidak perlu membangun model ekstraksi fitur yang seringkali sangat rumit [15]. Untuk menghasilkan keluaran yang diinginkan saat menggunakan masukan baru, Algoritma *Deep Learning* dilatih untuk mengenali pola dan mengkategorikan berbagai jenis data.

Machine Learning membutuhkan banyak waktu dan tenaga untuk membuat ekstraktor fitur, tetapi *Deep Learning* secara otomatis mengekstraksi fitur untuk tujuan klasifikasi. Ini adalah perbedaan utama antara keduanya. Untuk melatih algoritmanya, *Deep Learning* membutuhkan banyak data. Salah satu pengaplikasian *Deep Learning* adalah klasifikasi gambar [16].

D. Convolutional Neural Network (CNN)

Berbeda dengan metode pada umumnya yang memerlukan desain *filter* manual, *Convolutional Neural Network* (CNN) adalah model *Deep Learning* yang digunakan untuk pemrosesan gambar. CNN mempermudah

pra-pemrosesan dengan belajar bobot dan bias untuk membedakan fitur dalam gambar [12].

CNN menggunakan bidang reseptif yang tumpang tindih untuk menganalisis seluruh gambar, seperti korteks visual manusia, di mana *neuron* merespon pada area tertentu dari bidang visual. Ini memungkinkan CNN secara otomatis belajar dan mengidentifikasi berbagai aspek gambar [12].

Sebuah CNN terdiri dari tiga lapisan *Convolutional Layer*, *Pooling Layer*, dan *Fully Connected Layer*. Melalui *Convolutional Layer*, CNN pertama kali dapat menangkap atribut dasar seperti warna dan tepi. Untuk mengurangi dimensi, lapisan berikutnya dapat dicampur dengan *Pooling Layer*, yang memungkinkan CNN untuk mengenali bentuk dan komponen yang lebih kompleks [13].

Fully Connected Layer menyelesaikan proses dengan membuat klasifikasi akhir, yang memungkinkan untuk mengidentifikasi jenis kendaraan dalam gambar. Arsitektur ini memungkinkan CNN untuk meningkatkan kemampuan pengenalannya dengan mempelajari fitur gambar yang lebih kompleks secara bertahap [13].

1. Convolutional Layer

Convolutional Layer adalah komponen mendasar dari *Convolutional Neural Network* (CNN), yang terutama bertugas memproses gambar masukan. Ini beroperasi menggunakan filter atau kernel, matriks bobot kecil, untuk memindai seluruh gambar. Dengan melakukan perkalian titik antara filter dan wilayah lokal gambar, fitur penting seperti tepi dan bentuk akan diekstraksi [13].

Prosesnya melibatkan menggeser filter melintasi lebar, tinggi, dan kedalaman gambar (untuk gambar berwarna), menghasilkan peta fitur yang mewakili keberadaan fitur tertentu di lokasi berbeda dalam masukan. Mekanisme ini memungkinkan mempelajari pola kompleks dalam data visual secara efisien [13].

2. Pooling Layer

Pooling Layer memainkan peran penting dalam mengurangi dimensi data masukan. Ini beroperasi dengan menerapkan filter tanpa bobot terkait di seluruh masukan, sehingga menghasilkan pembuatan keluaran. Ada dua jenis metode *Pooling Layer* yang digunakan, yaitu *Max Pooling* dan *Average Pooling* [13].

Max Pooling melibatkan pemilihan piksel dengan nilai tertinggi dalam bidang reseptif sebagai keluaran, sedangkan *Average Pooling* menghitung nilai rata-rata dalam bidang reseptif untuk menentukan keluaran. *Pooling Layer* digunakan untuk meningkatkan efisiensi komputasi, mengurangi kompleksitas, dan mengurangi kemungkinan *overfitting* [13].

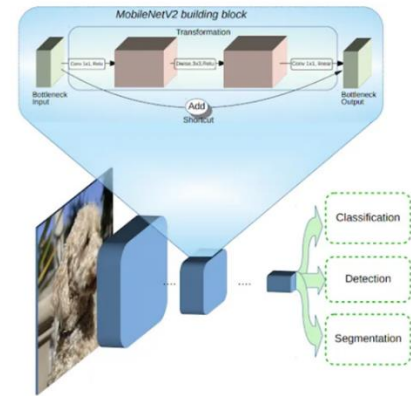
3. Fully Connected Layer

Lapisan ini bertanggung jawab untuk melaksanakan klasifikasi tugas dengan memanfaatkan fitur-fitur yang diekstraksi pada lapisan sebelumnya. Meskipun *Convolutional Layer* dan *Pooling Layers* menggunakan fungsi *ReLU*, *Fully Connected Layer* sepenuhnya menggunakan fungsi aktivasi *Softmax* untuk mengklasifikasikan, menghasilkan probabilitas mulai dari 0 hingga 1 [13].

E. MobileNetV2

MobileNets adalah salah satu *Convolutional Neural Network* (CNN) yang dirancang untuk mengatasi masalah sumber daya komputasi yang berlebihan. Dikembangkan oleh peneliti Google, *MobileNets* dirancang untuk telepon

selular. Tidak seperti arsitektur CNN pada umumnya, *MobileNets* membedakan dirinya dengan memanfaatkan lapisan konvolusi dengan *filter* yang sesuai dengan ketebalan gambar masukan. Konvolusi pada *MobileNets* terbagi menjadi *depthwise convolution* dan *pointwise convolution* [17]. Pada April 2017, *MobileNets* merilis versi kedua yaitu *MobileNetV2* yang masih menggunakan *depthwise convolution* dan *pointwise convolution* dengan menambahkan fitur baru yaitu *linier bottleneck* dan *shortcut connections* antar *bottlenecks* [17].



GAMBAR 5
MobileNetV2 Building Block

III. PERANCANGAN SISTEM

A. Desain Sistem

Desain sistem pada penelitian ini ialah klasifikasi biji kopi jenis arabika menggunakan model *Convolutional Neural Network* dengan arsitektur *MobileNetV2*. Setiap tahap dari sistem yang dibangun pada penelitian ini adalah Langkah citra digital, *preprocessing*, ekstraksi ciri, klasifikasi, dan evaluasi model.



GAMBAR 6
Diagram Blok Klasifikasi Biji Kopi

Pada gambar 6, menunjukkan bagaimana sistem bekerja dari tahap akuisisi citra yang akan di klasifikasi hingga tahap evaluasi. Sebelum sistem bekerja, pertama yang harus dilakukan yaitu memasukkan citra digital yang akan diklasifikasi.

Setelah itu melakukan *preprocessing*, ditahap ini adalah tahap mentransformasi data yang akan diolah dan dilanjutkan dengan *preprocessing* data. Lanjut ke tahap ekstraksi ciri dan ketahap klasifikasi. Pada tahap evaluasi model dilakukan untuk mengetahui performansi dari model.

B. Akuisisi Citra

Dataset seluruh gambar merupakan gambar biji kopi hijau. Isi dari *dataset* tersebut gambar biji kopi hijau dengan empat kelas yang kemudian akan dilakukan normalisasi, sehingga akan mudah terbaca oleh model *MobileNetV2*.

Proses pembagian *dataset* terbagi tiga yakni: data *train* (pelatihan), data *Validasi* (validasi), dan data *Testing*

(pengujian). Proses selanjutnya yaitu tranformasi citra untuk meningkatkan performasi dari model *MobileNetV2* yang akan diuji karena model tersebut bisa mempelajari citra dengan baik.

C. Dataset

Data yang digunakan pada tugas akhir ini adalah USK-Coffee *Dataset* yang didapat dari situs comvis.unsyiah.ac.id. USK-Coffee *Dataset* merupakan Kumpulan dataset publik yang memiliki empat kelas biji kopi hijau dan memiliki total 8.000 gambar biji kopi hijau [7].

TABEL 1
Dataset Citra Biji Kopi

Kategori	Total	Ukuran
Peaberry	2.000	256x256 piksel
Longberry	2.000	256x256 piksel
Premium	2.000	256x256 piksel
Cacat	2.000	256x256 piksel

Setiap kategori dibagi ke dalam tiga kelompok data yaitu:

1. *Training*, data pelatihan untuk melatih model.
 2. *Validasi*, data validasi digunakan untuk mengkonfirmasi atau memvalidasi kinerja klasifikasi selama pelatihan.
 3. *Testing*, data yang digunakan untuk menguji kinerja model yang telah dibuat.
- Adapun pembagian jumlah gambar pada setiap kelompok dapat dilihat pada tabel 2.

TABEL 2
Jumlah Citra Setiap Kelompok

Kelas	Train	Validasi	Test
Premium	1.200	400	400
Peaberry	1.200	400	400
Longberry	1.200	400	400
Defect	1.200	400	400
Total	4800	1600	1600

D. Pembagian Dataset

Seluruh gambar pada USK-Coffee *dataset* yang digunakan untuk melakukan proses klasifikasi biji kopi pada tugas akhir ini dibagi menjadi tiga kelompok yakni: *training* (pelatihan), *validasi* (validasi), *testing* (tes). Rasio yang digunakan pada tugas akhir ini adalah 4.800 gambar untuk *training*, 1.600 gambar untuk *validasi* dan 1.600 untuk *testing*. Rasio dataset yang digunakan mengikuti besaran nilai *batchsize* yang digunakan.

Rangkaian pembagian *dataset* ini digunakan untuk melatih dan menguji model *MobileNetV2*. Data *training* digunakan melatih model untuk mempelajari gambar sedangkan data *validasi* adalah sampel data pada model ketika latih saat melakukan *hyperparameter* untuk mencapai optimum. Data *testing* merupakan data yang digunakan untuk uji akhir dalam memprediksi gambar melihat seberapa baik dalam mendeteksi biji kopi.

E. Preprocessing

Tahap persiapan, juga dikenal sebagai *preprocessing* data, melibatkan pemrosesan data asli untuk memastikan data tersebut sesuai untuk digunakan dalam model klasifikasi.

Dalam penelitian ini, *preprocessing* data melibatkan penyesuaian ukuran gambar menjadi 256x256 piksel, sehingga memastikan semua gambar memiliki dimensi yang konsisten dan mengandung warna RGB. Kemudian melakukan tranformasi data terhadap citra dengan menggunakan *Flip Horizontal* dan kemudian menggunakan normalisasi citra.

Dataset dibagi menjadi tiga bagian untuk dimasukkan dalam model klasifikasi, dengan 4800 data pelatihan digunakan untuk pelatihan, 1600 data validasi digunakan untuk menghindari *overfitting*, dan 1600 data pengujian digunakan untuk menilai keakuratan model.

F. Klasifikasi

1. Pelatihan Model

Selama pelatihan, nilai parameter ditetapkan menggunakan *categorical crossentropy* untuk klasifikasi multikelas dengan *Adam Optimizer* yang bertujuan untuk mengoptimalkan model. Dengan nilai *Batch Size* adalah 16, nilai *Epoch* adalah 30 dan nilai *Learning Rate* 0.0001. nilai *Batch size*, *Epoch*, dan *Learning Rate* yang ideal untuk mencapai akurasi tertinggi ditentukan melalui eksperimen. Hasilnya, berbagai nilai *Batch Size*, *Epoch*, dan *Learning Rate* diuji untuk mendapatkan hasil yang paling akurat.

2. Pengujian

Setelah prosedur pelatihan selesai, fase selanjutnya melibatkan evaluasi kemampuan model untuk mengkategorikan gambar berdasarkan kelasnya masing-masing. Hasil evaluasi ini akan menampilkan pilihan sampel gambar, yang menampilkan hasil klasifikasi yang dilakukan sebelumnya.

3. Evaluasi

Penilaian kinerja suatu model klasifikasi dapat ditentukan melalui berbagai parameter pengukuran kinerja, antara lain akurasi, sensitivitas, dan presisi. Faktor-faktor ini biasanya dihitung menggunakan *Confusion Matrix*, yang terdiri dari nilai *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN).

Confusion Matrix mewakili terjadinya peristiwa Positif (P) dan Negatif (N). Selain itu, laporan klasifikasi memberikan tampilan langsung nilai akurasi, presisi, *recall*, dan *f1-score* untuk tujuan evaluasi. Secara sistematis, rumus dapat ditulis sebagai berikut.

Rumus akurasi:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Rumus presisi:

$$precision = \frac{TP}{TP + FP}$$

Rumus *recall*:

$$recall = \frac{TP}{TP + FN}$$

Rumus *f1-score*:

$$f1\ score = 2 \times \frac{recall \times precision}{recall + precision}$$

G. Komparasi

Banyak hal mempengaruhi hasil akhir dalam klasifikasi, bukan hanya *learning rate*, *batch size*, dan *epoch*. Namun, penelitian ini hanya berfokus pada menganalisis hasil komparasi dari beberapa *learning rate*, *batch size*, dan *epoch*.

Oleh karena itu, akan dibagi menjadi tiga bagian komparasi pada tahap ini yaitu:

1. Membandingkan *Learning rate*: *Learning rate* yang akan dibandingkan adalah 0.01, 0.001, dan 0.0001.
2. Membandingkan *Batch Size*: *batch size* yang digunakan ada tiga yaitu 8, 16, dan 32.
3. Membandingkan *Epoch*: *epoch* yang digunakan sebanyak tiga nilai yaitu 10, 20, dan 30.

Untuk pengukuran hasil komparasi akan menggunakan pengukuran *accuracy* dan *loss*. *Loss* yang digunakan adalah *binary cross entropy*. Secara matematis, rumus dapat ditulis sebagai berikut.

$$\text{LogLos} = \frac{-1}{N} + \sum_{i=1}^N y_i * \log(p(y_i)) + 1(1 - y_i) * \log(1 - p(y_i)) \quad (3.1)$$

Dimana:

N = jumlah sampel

y_i = label

$p(y_i)$ = prediksi

Untuk pengukuran *loss*, semakin kecil hasilnya semakin baik. Berbeda dengan akurasi, bila nilai semakin besar maka semakin bagus.

H. Struktur MobileNetV2

Arsitektur *MobileNetV2* dirancang untuk mengurangi *complexity cost* dan ukuran model *network* [21]. *MobileNetV2* memiliki dua jenis blok, yang pertama digunakan untuk *residual block* dan yang kedua untuk *downsizing*, sehingga sangat cocok untuk media dengan daya komputasi rendah seperti smartphone atau raspberry [22].

TABEL 3
Arsitektur MobileNetV2

Input	Operator	<i>t</i>	<i>c</i>	<i>n</i>	<i>s</i>
$224^3 \times 3$	Conv2d	-	32	1	2
$112^2 \times 32$	Bottleneck	1	16	1	1
$112^2 \times 16$	Bottleneck	6	24	2	2
$56^2 \times 24$	Bottleneck	6	32	3	2
$28^2 \times 32$	Bottleneck	6	64	4	2
$14^2 \times 64$	Bottleneck	6	96	3	1
$14^2 \times 96$	Bottleneck	6	160	3	2
$7^2 \times 160$	Bottleneck	6	320	1	1
$7^2 \times 320$	Conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	Avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	Conv2d 1x1	-	k	-	-

Setiap baris memberikan informasi sequence dari satu atau lebih modul stride layers dengan jumlah pengulangan yang dilambangkan dengan *n*. Pada layer di sequence yang sama memiliki nilai *c* sebagai keluaran. Pada layer pertama setiap sequence berisi nilai stride *s* dan stride lainnya menggunakan stride 1. Semua spatial convolution menggunakan kernel berukuran 3x3 [22].

I. Analisis Kebutuhan Sistem

1. Kebutuhan Perangkat Lunak

Berikut merupakan kebutuhan perangkat lunak yang digunakan dalam mengklasifikasi biji kopi:

- a. *Anaconda Navigator* digunakan untuk mengakses aplikasi *Jupyter Notebook*.
- b. *Jupyter Notebook*, aplikasi yang digunakan dalam proses klasifikasi dengan menggunakan Bahasa pemrograman Python.

- c. Windows 11 Home, digunakan sebagai sistem operasi pada laptop.
- d. *Library*, *library* yang digunakan ada beberapa diantaranya *numpy*, *pytorch*, *summary*, dan *matplotlib*.

2. Kebutuhan Perangkat Keras

Kebutuhan perangkat keras untuk pengujian tugas akhir yang digunakan ialah laptop, sebagai penunjang perangkat keras dan pengolahan program. Spesifikasi laptop yang digunakan ialah sebagai berikut:

1. *Processor* yang digunakan, *Intel Core i7 Gen9*.
2. *GPU* yang digunakan, *Nvidia GeForce GTX 1650*.
3. *RAM* yang digunakan, kapasitas 24 *GigaByte*.
4. *Internal Storage*, SSD 240 *Gigabyte*.

IV. SKENARIO PENGUJIAN

A. Skenario Pengujian

Skenario pengujian dilakukan dengan melatih data *training* dan data *validation* menggunakan CNN dengan arsitektur *MobileNetV2*. Pada tahap pelatihan model akan dilakukan dengan beberapa skenario yang bertujuan untuk mengoptimalkan model dalam mengklasifikasi biji kopi.

Pada skenario pengujian ini, terdapat tiga skenario dalam mengoptimalkan model dengan penyesuaian *hyperparameter* yang digunakan pada tugas akhir ini. Skenario pengujian yang pertama melakukan pengujian nilai *learning rate*, skenario kedua melakukan pengujian terhadap nilai *batch size*, dan skenario ketiga dilakukan pengujian terhadap nilai *epoch*.

Kemudian model akan dilakukan pengujian dengan menggunakan data *testing* yang terdiri dari 400 gambar setiap kelas yang dilakukan secara acak, untuk mengetahui performa yang di hasilkan. Dalam skenario ini, jenis *optimizer* yang digunakan ialah *Optimizer Adam*.

1. Pengujian Learning Rate

Dalam Skenario pertama penulis melakukan pengujian jumlah nilai *learning rate* pada saat pelatihan model yang bertujuan untuk mendapatkan nilai *learning rate* yang optimal sehingga dapat memberikan hasil performansi yang baik. *Learning rate* adalah salah satu parameter yang berpengaruh terhadap performansi model [20].

Pada pengujian ini menggunakan 300 data *training*, 100 data *validasi*, dan 100 data *testing*. Tujuan pengujian ini untuk mencari hasil yang optimal dengan beberapa perubahan parameter berikut:

- a. *Batchsize* = 16
- b. *Learning rate* = 0.01, 0.001, dan 0.0001
- c. *Epoch* = 30

TABEL 4 Pengujian Learning Rate

<i>Learning rate</i>	Akurasi	Presisi	<i>Recall</i>	<i>F1-score</i>
0,01	81,56	78,81	76,25	77,51
0,001	86	87,76	93,25	90,42
0,0001	88,19	96,74	89	92,71

2. Pengujian Batch Size

Skenario kedua yaitu Pengujian *Batch size* bertujuan untuk mengetahui nilai yang optimal sehingga hasil performansi yang didapat baik. Berikut nilai parameter yang digunakan:

- a. *Batch size* = 8, 16, dan 32
- b. *Epoch* = 30

c. *Learning rate* = 0.0001

Pada pengujian ini dilakukan secara bertahap, dan terdapat pembagian data yang disesuaikan dengan besaran nilai *batch size*. Untuk *batch size* 8, 600 data *training*, 200 data *validasi*, dan 200 data *testing*. *Batch size* 16, data *training* 300, data *validasi* 100, dan data *testing* 100. *Batch size* 32, data *training* 150, data *validasi* 50, dan data *testing* 50.

Nilai data tersebut didapat dari hasil pembagian dari total data *training*, *validasi*, dan *testing* dengan besaran nilai *batch size*. Hasil pengujian *batch size* dapat dilihat pada tabel 5.

TABEL 5 Pengujian *Batch Size*

<i>Batch size</i>	Akurasi	Presisi	Recall	<i>F1-score</i>
8	88,12	93,99	90	91,95
16	88,19	96,74	89	92,71
32	86,44	95,08	91,75	93,38

3. Pengujian *Epoch*

Pada skenario ketiga, dilakukan pengujian terhadap *epoch* yang bertujuan untuk mengetahui jumlah *epoch* yang optimal sehingga dapat memberikan hasil performa yang baik. Adapun nilai parameter yang digunakan sebagai berikut:

- Epoch* = 10, 20, dan 30
- Batch size* = 16
- Learning rate* = 0,0001

Pada skenario pengujian ketiga, menggunakan dataset dengan pembagian 300 data *training*, 100 data *validasi*, dan 100 data *testing*. Hasil pengujian *epoch* dapat dilihat pada tabel 6.

TABEL 6 Pengujian *Epoch*

<i>Epoch</i>	Akurasi	Presisi	Recall	<i>F1-score</i>
10	85,06	96,51	83	89,25
20	86	94,37	88	91,07
30	88,19	96,74	89	92,71

B. Pengujian Akurasi

Pengujian akurasi dilakukan untuk mengetahui ketepatan nilai dari hasil pelatihan yang telah dilakukan. Terdapat empat kelas biji kopi yang akan diuji akurasinya, yaitu kelas *premium*, *peaberry*, *longberry*, dan cacat.

C. Analisis Pengujian

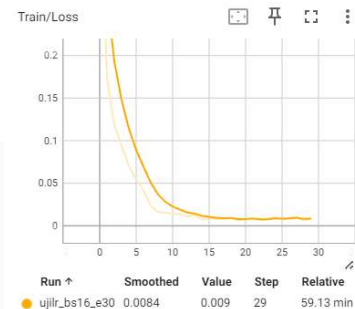
Pada pengujian menggunakan *MobileNetV2*, didapatkan parameter yang optimal yang berdampak terhadap performansi model. Berikut grafik *training* dan *validasi* dari hasil pengujian.



GAMBAR 7
Grafik *training* akurasi



GAMBAR 8
Grafik *validasi* akurasi



GAMBAR 9
Grafik *training* loss



GAMBAR 10
Grafik *validasi* loss

Pada hasil pengujian ditemukan yang optimal dari yang sudah diuji coba dari pengujian *hyperparameter*. Hasil yang optimal adalah model dengan *hyperparameter* nilai yaitu *learning rate* 0.0001, *batch size* 16, dan *epoch* 30. Terlihat grafik *training* dan *validasi* cukup optimal karena tidak mengalami *overfitting* maupun *underfitting* yakni jarak akurasi *training* dan *validasi* tidak terlalu jauh.



GAMBAR 11
Confusion Matrix

Pada gambar 11 menunjukkan label kelas biji kopi yang dimana terdiri dari empat kelas yaitu *Premium*, *Peaberry*, *Longberry*, dan Cacat. Menunjukkan bahwa kelas *premium* mendapatkan nilai yang cukup tinggi yakni model mampu mendeteksi biji kopi *premium* dengan baik, disusul dengan kelas *longberry* dan cacat.

Sedangkan terendah pada kelas *peaberry*, disebabkan biji kopi *peaberry* dan biji kopi *premium* memiliki warna yang hampir serupa dan beberapa bentuk biji kopi *peaberry* menyerupai bentuk *premium*. Kekurangan variasi data yang digunakan juga berdampak pada proses pelatihan.

V. KESIMPULAN

Berdasarkan hasil dari pengujian dan analisis yang telah dilakukan pada Tugas Akhir ini, maka dapat disimpulkan sebagai berikut:

1. Sistem klasifikasi empat kelas biji kopi cukup optimal sehingga performa yang didapatkan pun baik dengan menggunakan model *MobileNetV2*.
2. Dengan melakukan pengujian *hyperparameter* pada *MobileNetV2*, didapatkan performansi model yang cukup optimal dengan *learning rate* 0.0001, *batch size* 16, dan *epoch* 30. Menghasilkan nilai Akurasi 88,19%, Presisi 96,74%, *Recall* 89%, dan *F1-score* 92,71%.

REFERENSI

- [1] K. D. P. R. I. IV, "Parlementaria Terkini - Dewan Perwakilan Rakyat," 2020. <https://www.dpr.go.id/berita/detail/id/28110/t/Petani+Kopi+Hadapi+Tiga+Masalah+Besar> (accessed Aug. 09, 2021).
- [2] Mawardi, I., Hanif, H., Jennifar, J., & Safaruddin, S. (2021, January 4). PENERAPAN MESIN SORTASI DALAM UPAYA EFESIENSI PROSES PRODUKSI KOPI GAYO SEBAGAI PRODUK UNGGULAN DAERAH ACEH TENGAH. *Jurnal Bakti Masyarakat Indonesia*, 3(2). <https://doi.org/10.24912/jbmi.v3i2.9400>.
- [3] Rochmawati, N., Hidayati, H. B., Yamasari, Y., Tjahyaningtijas, H. P. A., Yustanti, W., & Prihanto, A. (2021, December 31). Analisa Learning Rate dan Batch Size pada Klasifikasi Covid Menggunakan Deep Learning dengan Optimizer Adam. *Journal of Information Engineering and Educational Technology*, 5(2), 44–48. <https://doi.org/10.26740/jieet.v5n2.p44-48>.

- [4] sname. (2020, November 2). Maruti Techlabs. <https://marutitech.com/top-8-deep-learning-frameworks/>.
- [5] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, 2018, doi: 10.1109/CVPR.2018.00474.
- [6] F. Hutter, L. Kotthoff, and J. Vanschoren, *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019.
- [7] Febriana, A., Muchtar, K., Dawood, R., Lin, CY., (2022, June). "USK-COFFEE Dataset: A Multi-class Green Arabica Coffee Bean Dataset for Deep Learning." In 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom). IEEE.
- [8] "History of Arabica," World Coffee Research, Jun. 09, 2023. <https://varieties.worldcoffeeresearch.org/arabica-2/history-of-arabica>.
- [9] S. H. Sinaga and E. Julianti, "Physical Characteristics of Gayo Arabica Coffee with Semi-Washed Processing," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 782, no. 3, 2021, doi: 10.1088/1755-1315/782/3/032093.
- [10] N. Anhofiah, "KARAKTERISTIK MUTU KOPI ARABIKA LONGBERRY DI KBQ BABURRAYAN HASIL PENGOLAHAN SEMI WASH DENGAN PENGARUH PERLAKUAN LAMA FERMENTASI DAN LAMA PENYANGRAIAN," Anhofiah | Jurnal Pertanian Agros, 2023. <https://doi.org/10.37159/j.pagros.v25i1.2407>.
- [11] D. Suhandy, M. Yulia, and Kusumiyati, "Chemometric Quantification of Peaberry Coffee in Blends Using UV–Visible Spectroscopy and Partial Least Squares Regression," *AIP Conf. Proc.* 2021, 2018, doi: 10.1063/1.5062774.
- [12] "Mengenal Karakteristik Kopi Arabika" [Get to Know the Characteristics of Arabica Coffee] 2020. <https://www.dolce-gusto.co.id/mengenal-kopi-arabika> (accessed Nov. 03, 2021).
- [13] L. F. Torres, "Convolutional Neural Network From Scratch - LatinXinAI - Medium," Medium, Nov. 23, 2023. <https://medium.com/latinxinai/convolutional-neural-network-from-scratch-6b1c856e1c07>.
- [14] Febriana, A., Muchtar, K., Dawood, R., Lin, CY., (2022, June). "USK-COFFEE Dataset: A Multi-class Green Arabica Coffee Bean Dataset for Deep Learning." In 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom). IEEE.
- [15] Suyanto, *Machine Learning Tingkat Dasar dan Lanjut*. 2018.
- [16] M. R. R. Allaam and A. T. Wibowo, "KLASIFIKASI GENUS TANAMAN ANGGREK MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN) Program Studi Sarjana Informatika Fakultas Informatika Universitas Telkom Bandung," *e-Proceeding Eng.*, vol. 8, no. 2, pp. 1153–1189, 2021.
- [17] R. O. Ekoputris, "MobileNet: Deteksi Objek pada Platform Mobile - Nodeflux - Medium," *Medium*, May 31, 2018. <https://medium.com/nodeflux/mobilenet-deteksi-objek-pada-platform-mobile-bbbf3806e4b3>
- [18] J. Sanjaya and M. Ayub, "Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup," *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 2, pp. 311–323, 2020, doi: 10.28932/jutisi.v6i2.2688.

- [19] F. Zaelani and Y. Miftahuddin, "Perbandingan Metode EfficientNetB3 dan MobileNetV2 Untuk Identifikasi Jenis Buah-buahan Menggunakan Fitur Daun," *Jurnal Ilmiah Teknologi Infomasi Terapan*, vol. 9, no. 1, Dec. 2022, doi: 10.33197/jitter.vol9.iss1.2022.911.
- [20] N. Rochmawati, H. B. Hidayati, Y. Yamasari, H. P. A. Tjahyaningtjas, W. Yustanti, and A. Prihanto, "Analisa Learning Rate dan Batch Size pada Klasifikasi Covid Menggunakan Deep Learning dengan Optimizer Adam," *J. Inf. Eng. Educ. Technol.*, vol. 5, no. 2, pp. 44–48, 2021, doi: 10.26740/jieet.v5n2.p44-48.
- [21] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, 2018, doi: 10.1109/CVPR.2018.00474.
- [22] Afis Julianto, Andi Sunyoto, and Ferry Wahyu Wibowo, "OPTIMASI HYPERPARAMETER CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI PENYAKIT TANAMAN PADI," *TEKNIMEDIA: Teknologi Informasi dan Multimedia*, vol. 3, no. 2, pp. 98–105, Dec. 2022, doi: 10.46764/teknimedia v3i2.77.