

**KLASIFIKASI MUTU GREENBEAN *COFFEE*
MENGUNAKAN METODE *CONVOLUTIONAL*
*NEURAL NETWORK***

Oleh :

AZHAR MUHAMAD

T3118209

SKRIPSI

**Untuk memenuhi salah satu syarat ujian
Guna memperoleh gelar Sarjana**



**PROGRAM SARJANA
TEKNIK INFORMATIKA
UNIVERSITAS ICHSAN GORONTALO
GORONTALO
2023**

PERSETUJUAN SKRIPSI

KLASIFIKASI MUTU *GREENBEAN COFFEE* MENGUNAKAN METODE *CONVOLUTIONAL NEURAL NETWORK*

Oleh

AZHAR MUHAMMAD

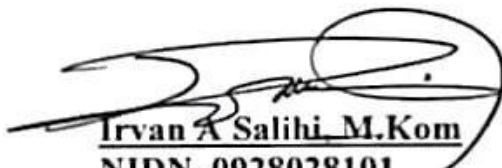
T3118209

SKRIPSI

Untuk memenuhi salah satu syarat ujian
Guna memperoleh gelar Sarjana.
Program Studi Teknik Informatika
ini telah disetujui oleh Tim Pembimbing

Gorontalo, Februari 2023

Pembimbing I



Irvan A Salihi, M.Kom
NIDN. 0928028101

Pembimbing II



Aprivanto Alhamad, M.Kom
NIDN. 0924048601

PENGESAHAN SKRIPSI

KLASIFIKASI MUTU *GREENBEAN COFFEE* MENGUNAKAN METODE *CONVOLUTIONAL* *NEURAL NETWORK*

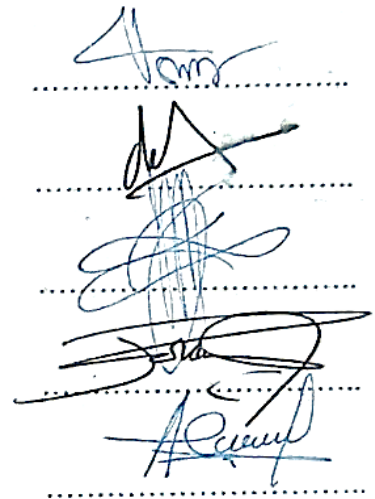
Oleh

AZHAR MUHAMMAD

T3118209

Diperiksa oleh Panitia ujian Strata Satu (S1)
Universitas Ichsan gorontalo

1. Ketua Penguji
Asmaul Husna N, M.Kom
2. Anggota
Sudirman S. Panna, M.Kom
3. Anggota
Zulfrianto Y. Lamasigi, M.Kom
4. Pembimbing I
Irvan Abraham Salihi, M.Kom
5. Pembimbing II
Apriyanto Alhamad, M.Kom



Mengetahui :



Dekan Fakultas Ilmu Komputer

Irvan Abraham Salihi, M.Kom
NIDN. 0928028101



Ketua Program Studi

Sudirman S. Panna, M.Kom
NIDN. 0924038205

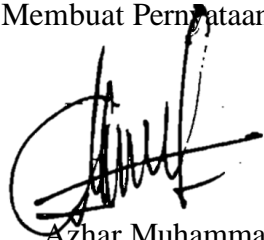
PERNYATAAN SKRIPSI

Dengan ini saya menyatakan bahwa :

1. Karya tulis (Skripsi) saya ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (Sarjana) baik di Universitas Ichsan Gorontalo maupun di perguruan tinggi lainnya.
2. Karya tulis (Skripsi) saya ini adalah murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan dari Tim Pembimbing.
3. Karya tulis (Skripsi) saya tidak terdapat karya atau pendapat yang telah dipublikasikan orang lain, kecuali secara tertulis dicantumkan sebagai acuan/sitasi dalam naskah dan dicantumkan pula dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini maka saya bersedia menerima sanksi akademik berupa pencabutan gelar, yang telah diperoleh karena karya tulis ini, serta sanksi lainnya sesuai dengan norma-norma yang berlaku di Universitas Ichsan Gorontalo.

Gorontalo, Februari 2023

Yang Membuat Pernyataan,



Azhar Muhammad

ABSTRACT

AZHAR MUHAMAD. T3118209. *QUALITY CLASSIFICATION OF GREEN BEAN COFFEE USING CONVOLUTIONAL NEURAL NETWORK METHOD*

Coffee is one of Indonesia's foreign exchange sources and plays an important role in the development of the plantation industry. In the commercial process, a product must have advantages, especially in terms of quality to survive in world market competition. The Convolutional Neural Network (CNN) method is a Deep Learning method that can identify and classify an object in a digital image. The training process is carried out by looking for a model structure that matches the training data and validation data so that overfitting does not occur in the CNN network. The experimental results in this study indicate that the Convolutional Neural Network method can classify the quality of green bean coffee with an accuracy rate of 90%, recall of 92%, precision of 86%, and F1-Score of 88% from 30 images by taking 15 sample images from each class using confusion matrix testing.



Keywords: Classification, Green bean Coffee, Convolutional Neural Network

ABSTRAK

AZHAR MUHAMAD T3118209. KLASIFIKASI MUTU GREENBEAN COFFEE MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK

Kopi adalah salah satu penghasil devisa Indonesia dan memegang peran penting dalam pengembangan industri perkebunan. Dalam proses komersial, suatu produk harus memiliki keunggulan terutama dalam hal kualitas agar dapat bertahan dalam persaingan dunia. Metode *Convolutional Neural Network* (CNN) merupakan metode *Deep Learning* yang dapat mengidentifikasi dan mengklasifikasi sebuah objek dalam sebuah citra digital. Proses *training* dilakukan dengan mencari bentuk model yang sesuai dengan data *training* dan data validasi sehingga tidak terjadi *overfitting* pada jaringan CNN. Hasil percobaan pada penelitian ini menunjukkan bahwa metode *Convolutional Neural Network* dapat mengklasifikasikan mutu *greenbean coffee* dengan tingkat akurasi sebesar 90%, *recall* 92%, *precision* 86%, dan *F1-Score* sebesar 88% dari 30 gambar dengan mengambil 15 sampel gambar dari setiap kelas dengan menggunakan pengujian confusion matrix.



Kata Kunci : Klasifikasi, *Greenbean Coffee*, *Convolutional Neural Network*

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT atas segala rahmatnya sehingga dapat menyelesaikan skripsi ini yang berjudul **“Klasifikasi mutu *Greenbean Coffee* menggunakan metode *Convolutional Neural Network*”**. Sebagai salah satu syarat Ujian Akhir guna memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Ichsan Gorontalo.

Penulis menyadari bahwa skripsi ini tidak akan terwujud tanpa bantuan dan dorongan dari berbagai pihak. Untuk itu, penulis dengan tulus dan rendah hati mengucapkan terima kasih kepada:

1. Ibu Dr. Juriko Abdusamat, M.Si., selaku ketua Yayasan Pengembangan Ilmu Pengetahuan dan Teknologi (YPIPT) Ichsan Gorontalo;
2. Bapak Dr. Abdul Gaffar La Tjokke, M.Si, selaku Rektor Universitas Ichsan Gorontalo;
3. Bapak Irvan Abraham Salihi, M.Kom, Selaku Dekan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
4. Bapak Sudirman Melangi, M.Kom, selaku Wakil Dekan I Bidang Akademik dan Bidang Kemahasiswaan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
5. Ibu Irma Surya Kumala Idris, M.Kom, selaku Wakil Dekan II Bidang Administrasi Umum dan Keuangan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
6. Bapak Sudirman S. Panna, S.Kom, M.Kom, selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
7. Bapak Irvan A Salihi , M.Kom, selaku Pembimbing I, yang selalu mendukung dan membimbing penulis dalam penyusunan skripsi,
8. Bapak Aprianto Alahmad, M.Kom, selaku Pembimbing II, yang selalu mendukung dan membimbing penulis dalam penyusunan skripsi,

9. Bapak dan Ibu Dosen Universitas Ichsan Gorontalo yang telah mendidik dan mengajarkan berbagai disiplin ilmu kepada penulis;
10. Teristimewa kepada keluarga saya, umi dan ma'nila tercinta , yang sudah memberikan kasih dan sayang kepada penulis mulai dari kecil sampai sekarang dan Ibu, Abah yang selalu memberikan dorongan moral maupun materi yang sangat besar kepada saya. Tanpa cinta dari keluarga mungkin skripsi ini tidak dapat diselesaikan;
11. Rekan-rekan angkatan 2018 khususnya Rizky Abd Karim S. Nur, S.Kom, Rahmat Hidayat, S.Kom, Nur Aiffah Thiara Papunas, S.Kom, Dwiki Kurniawan, Faizal Rahman dan senior-senior saya, yang telah banyak membantu dan mendukung saya;
12. Kepada semua pihak yang ikut membantu dalam menyelesaikan skripsi ini yang tak sempat penulis sebutkan satu-persatu;

Penulis mengucapkan terima kasih banyak kepada semua pihak yang telah membantu dalam menyelesaikan skripsi ini sehingga usulan skripsi ini dapat terselesaikan. Penulis mengharapkan saran dan kritik sehingga skripsi ini dapat memberikan manfaat bagi bidang pendidikan serta bisa dikembangkan lagi lebih lanjut.

Gorontalo, Februari 2023

Penulis

DAFTAR ISI

	Halaman
PERSETUJUAN SKRIPSI.....	i
PENGESAHAN SKRIPSI.....	ii
PERNYATAAN SKRIPSI	iii
ABSTRACT	iv
ABSTRAK	v
KATA PENGANTAR.....	vi
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
DAFTAR LAMPIRAN	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah	3
1.3 Rumusan Masalah	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
BAB II LANDASAN TEORI	4
2.1 Tinjauan Studi	4
2.2 Tinjauan Pustaka	6
2.2.1 Kopi	6
2.2.2 <i>Greenbean Coffee</i>	7
2.2.3 Citra (image).....	7
2.2.4 Pengolahan Citra.....	7
2.2.5 <i>Convolutional neural network (CNN)</i>	7
2.3 Kerangka Pikir.....	15
BAB III METODE PENELITIAN.....	16

3.1 Jenis , Metode , Subjek , Objek dan Waktu Penelitian.....	16
3.2 Pengumpulan Data.....	16
3.3 Pemodelan	17
3.3.1 <i>Pra-Processig</i>	17
3.3.2 Data Training	18
3.3.3 Training CNN	18
3.3.4 Data Testing.....	19
3.4 Evaluasi Model.....	19
BAB IV HASIL PENELITIAN.....	20
4.1 Hasil Pengumpulan data	20
4.2 Hasil Pemodelan	21
4.2.1 Pra Pengolahan Data.....	21
4.2.2 Data Traning	23
4.2.3 Traning CNN	23
4.2.4 Data Testing.....	23
4.2.5 <i>Convolutional Neural Network (CNN)</i>	24
4.3 Evaluasi Model.....	31
4.3.1 <i>Confusion Matrix</i>	31
BAB V PEMBAHASAN	35
5.1 Pembahasan Model.....	35
BAB VI PENUTUP.....	41
6.1 Kesimpulan.....	41
6.2 Saran	41
DAFTAR PUSTAKA.....	42

DAFTAR GAMBAR

Gambar 2.1 Contoh gambar kopi.....	6
Gambar 2.2 Arsitektur CNN.....	8
Gambar 2.3 Proses Convolution	8
Gambar 2.4 Proses polling layer.....	9
Gambar 2.5 Max polling	10
Gambar 2.6 Flatten	11
Gambar 2.7 Fully connected layer	12
Gambar 2.8 Dropout	13
Gambar 3.1 Pemodelan.....	17
Gambar 4.1 Hasil Encoder.....	22
Gambar 4.2 Tahapan Pra-pengolahan.....	22
Gambar 4.3 Model Arsitektur	23
Gambar 4.4 Operasi Konvolusi.....	23
Gambar 4.5 Sampel Gambar dan Input	23
Gambar 4.6 Proses Konvolusi Kernel 3x3.....	24
Gambar 4.7 proses Pergeseran dengan Stride=1 kernel 3x3.....	24
Gambar 4.8 Proses Konvolusi pada pergeseran pertama.....	25
Gambar 4.9 Proses konvolusi pada pergeseran terakhir	25
Gambar 4.10 Feature Map	26
Gambar 4.11 MaxPooling	26
Gambar 4.12 Matriks dan vektor	27
Gambar 4.13 Fully Connected layer	27
Gambar 5.1 Proses Resize Gambar	33
Gambar 5.2 Label Encoder dan hasil encoder	34
Gambar 5.3 Coding Augmentasi.....	34
Gambar 5.4 Coding Model Fitting.....	35
Gambar 5.5 Hasil Model Fitting	36
Gambar 5.6 Coding Akurasi training dan validasi.....	37
Gambar 5.7 Coding untuk grafik loss.....	38
Gambar 5.7 Hasil Grafik Loss training dan validasi.....	38

DAFTAR TABEL

Tabel 2.1 Penelitian terkait	4
Tabel 2.2 Hyperparameter Pada <i>Convolutional Layer</i>	9
Tabel 2.3 Confusion Matrix.....	13
Tabel 4.1 Kopi buruk dan baik	20
Tabel 4.2 Hasil Data Testing	30
Tabel 4.3 Hasil Evaluasi Confusion Matrix	30

DAFTAR LAMPIRAN

Lampiran 1 : Kode Program	42
Lampiran 2 : Daftar Riwayat Hidup	43
Lampiran 3 : Hasil Turniti	44
Lampiran 4 : Surat Keterangan Bebas Pustaka	45
Lampiran 5 : Surat Penelitian	46
Lampiran 6 : Surat Rekomendasi Penelitian	47
Lampiran 7 : Lembar Revisi	48

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kopi merupakan salah satu komoditas perkebunan unggulan tanah air. Adapun jenis kopi yang banyak dibudidayakan di Indonesia yaitu jenis kopi Robusta dan kopi Arabica, karena dinilai memiliki cita rasa yang tinggi[1]. Bukan hanya untuk menikmati rasa kopinya sendiri, tetapi untuk negara seperti Indonesia, terdapat sumber nilai ekonomi dan devisa yang cukup besar[2]. Ada banyak varietas kopi, tetapi hanya dua yang memiliki nilai jual yang signifikan, kombinasi antara arabika sangrai dan robusta. Perbedaan utama antara kedua kopi adalah rasa, warna, bentuk, dan kandungan kafeinnya.

Biji arabika memiliki rasa yang lebih ringan dan kandungan kafein 70% lebih sedikit dibandingkan biji kopi robusta. Karakteristik biji kopi arabika memiliki bentuk yang lebih oval, dan ukurannya lebih besar dengan garis lipatan tengahnya lebih tegas sedangkan robusta memiliki ukuran lebih kecil dan lebih bundar, dan juga garis tengahnya terlihat kurang jelas serta warnanya pun lebih pucat.

Untuk meningkatkan daya saing pasar, kualitas biji kopi harus memenuhi standar kualitas agar dapat diterima oleh sebagian besar konsumen. Biji kopi terdaftar dalam Standar Nasional Indonesia (SNI) (2008), yang menetapkan definisi dan penentuan nilai cacat biji kopi dan penilaian kualitas berdasarkan harga nilai default yang diperoleh. Klasifikasi mutu biji kopi berdasarkan SNI 01-2907-2008[3].

Teknologi pengolahan citra digital dapat digunakan untuk mengklasifikasikan mutu biji kopi yang sesuai standar SNI. Pengolahan citra digital dapat mendeteksi warna dan tekstur yang terdapat pada biji kopi. Hal itu dilakukan dengan melakukan perbandingan komposisi warna RGB (Red, Green, Blue), nilai GLCM, dan deteksi cacat yang diperoleh dari citra biji kopi. Fitur RGB dan GLCM dipilih karena kematangan buah yang ditandai

dengan perubahan warna dan tekstur pada permukaan biji kopi.[4] Fitur RGB dan ekstraksi fitur tekstur dipilih karena untuk menentukan perubahan warna dalam mendeteksi cacat warna pada biji kopi. Studi terkait dalam pengklasifikasian K-Nearest Neighbor (K-NN) telah dilakukan beberapa peneliti sebelumnya[5].

Penelitian sebelumnya memakai deteksi Tepi Canny menggunakan pembagian terstruktur mengenai K-Nearest Neighbor pada mengidentifikasi jenis kopi arabika dan robusta. Sehingga K-Nearest Neighbor bisa membedakan gambaran biji kopi Arabika dan Robusta memakai 7 fitur momen invarian yang sebagai tujuan primer penelitian. Persentase gambaran tertinggi buat biji Arabika dan Robusta diklasifikasikan dalam K=1 dan K=7, dengan akurasi 67,5% [6].

Pada Penelitian (Arboleda, Fajardo & Medina, 2018), dengan menerapkan Metode Klasifikasi Mutu Greenbean Kopi Arabika Lanang Dan Biasa Menggunakan K-Nearest Neighbor Berdasarkan Bentuk. Hasil pengujian menggunakan metode K-Nearest Neighbor dalam klasifikasi menggunakan semua karakteristik perimeter, rasio kebulatan, luas, dan keliling akurasi 63,5 % tercapai. Sistem ini dapat mendeteksi biji kopi arabika utuh dengan akurasi 97%. Akurasi 82,56 dicapai dengan akurasi robusta 78,46% berbeda dengan penelitian sebelumnya yang menggunakan tiga fitur morfologi seperti rasio kebulatan, luas, dan keliling [3].

Pada Penelitian (Rizky Abd Karim S. Nur) dengan menerapkan metode *Convolutional Neural Network* untuk Pengenalan obat herbal pada penelitian ini, metode jaringan saraf convolutional. Metode CNN merupakan metode *deep learning* yang mengidentifikasi dan mengklasifikasikan suatu objek dalam sebuah citra digital. Proses pelatihan dilakukan dengan mencari pola yang sesuai dengan data yang akan diolah untuk mendapatkan hasil yang terbaik. Argumentasi juga digunakan pada data training dan validasi agar tidak terjadi *overfitting* pada jaringan CNN. Hasil percobaan dari penelitian ini menunjukkan bahwa metode *convolutional neural network* dapat mengidentifikasi tanaman obat dengan tingkat akurasi 85%, kemampuan

pemulihan 87%, akurasi 89% dan skor 89% F1 adalah 88.51 keluar dari 40 gambar menggunakan alat *Confusion Matrix Test* [7].

Berdasarkan penelitian diatas, dapat dilakukan penelitian untuk menguji coba algoritma Convolutional Neural Netwrok (CNN) untuk klasifikasi Mutu Biji Kopi dengan mengangkat judul penelitian **“Klasifikasi Mutu *Greenbean* Kopi Menggunakan Metode *Convolutional Neural Network* (CNN) “**

1.2 Identifikasi Masalah

Banyaknya jenis biji kopi yang ada membuat masyarakat sulit dalam memilih biji kopi yang berkualitas

1.3 Rumusan Masalah

Berdasarkan latar belakang diatas maka dapat dirumuskan masalah Bagaimana hasil akurasi penggunaan metode Convolutional Neural Network pada klasifikasi mutu Greenbean Coffee?

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah Untuk mengetahui tingkat akurasi penggunaan algoritma *Convolutional Neural Network* pada klasifikasi mutu *Greaanbean Coffee*.

1.5 Manfaat Penelitian

Mengetahui penerapan metode *Convolutional Neural Network* (CNN) untuk klasifikasi gambar.

BAB II

LANDASAN TEORI

2.1 Tinjauan Studi

Penelitian tentang Klasifikasi *Greenbean* kopi Menggunakan Metode *Convolutional Neural Network* (CNN). Dan metode yg digunakan telah dilakukan oleh beberapa peneliti sebelumnya. Berikut beberapa penelitian terkait

Tabel 2.1 Penelitian Terkait

No	Peneliti	Judul	Tahun	Metode	Hasil
1.	Dedy Ikhsan	Metode Klasifikasi Mutu <i>Greenbean</i> Kopi Arabika Lanang Dan Biasa Menggunakan <i>K-Nearest Neighbor</i> Berdasarkan Bentuk	2020	<i>K-Nearest Neighbor</i> (K-NN)	Klasifikasi mutu biji kopi hijau dalam empat karakteristik dengan K-NN mencapai akurasi 63.5 dengan nilai K3.
2.	Rizky abd karim s. nur	Klasifikasi jenis tumbuhan obat herbal menggunakan <i>Convolutional neural network</i>	2022	<i>Convolutional Neural Network</i>	Metode CNN yang diterapkan mampu mengenali jenis tumbuhan obat herbal dengan benar dapat dilihat pada tabel hasil klasifikasi data uji dengan mendapatkan akurasi hingga 85%, nilai precision 89%,

					nilai <i>recall</i> 87% dan <i>F1-score</i> 88.51%, dengan data yang sama dan 40 sampel jenis daun dengan masing-masing kelas 4 sampel jenis daun untuk data uji [8]
3.	Fitrianingsih and Rodiah	Klasifikasi jenis citra daun mengga menggunakan <i>Convolutional Neural Network</i>	2020	<i>Convolutional Neural Network (CNN)</i>	Penelitian ini menerapkan CNN dan mengukur setiap gambar untuk mendeteksi 44 struktur daun dari dataset daun MK. Modul yang digunakan dalam penelitian ini terdiri dari tiga jenis yaitu basicCNN, AlexNet dan GoogleNet, dengan akurasi 95% [9].

2.2 Tinjauan Pustaka

2.2.1 Kopi

Kopi adalah salah satu tanaman semak yang dapat tumbuh di daerah tropis dengan ketinggian 700 – 1600 mdpl. Pohon kopi dipangkas pendek untuk menghemat energi dan bantuan panen, namun bisa tumbuh lebih dari 30 kaki (9 meter) tinggi. Setiap pohon ditutupi daun hijau dan ranting yang saling bertautan saling berpasangan. Ceri kopi tumbuh di sepanjang cabang. Karena tumbuh dalam siklus yang terus menerus, tidak biasa melihat bunga, buah hijau dan buah matang bersamaan pada satu pohon. Dibutuhkan hampir setahun untuk ceri yang matang setelah berbunga pertama, dan sekitar 5 tahun pertumbuhan mencapai produksi buah penuh. Sementara tanaman kopi bisa hidup sampai 100 tahun, mereka umumnya paling produktif antara usia 7 dan 20. Perawatan yang tepat dapat mempertahankan dan bahkan meningkatkan hasilnya selama bertahun-tahun, tergantung varietasnya. Rata-rata pohon kopi menghasilkan 10 pon ceri kopi per tahun, atau 2 pon *green bean coffee*. Semua kopi komersial berasal dari wilayah dunia disebut *Coffee Belt*. Pohon tumbuh paling baik di tanah yang subur, dengan suhu ringan, hujan deras dan matahari yang teduh.

Kopi berasal dari genus tanaman yang dikenal dengan nama *Coffea*. Dalam genus ada lebih dari 500 marga dan 6.000 spesies pohon tropis dan semak belukar. Para ahli memperkirakan bahwa ada sekitar 25 sampai 100 spesies tanaman kopi. Genus ini pertama kali dijelaskan pada abad ke-18 oleh ahli botani Swedia, Carolus Linneaus, yang juga menggambarkan *Coffea Arabica* di *Species Plantarum* pada tahun 1753. Ahli botani tidak setuju sejak klasifikasi yang tepat, karena tanaman kopi dapat berkisar secara luas. Mereka bisa menjadi semak-semak kecil ke pohon tinggi, dengan daun berukuran satu sampai 16 inci, dan warnanya ungu atau kuning ke hijau gelap yang dominan.



(a)



(b)

Gambar 2. 1 Contoh Gambar Kopi

2.2.2 *Greenbean Coffee*

Greenbean coffee adalah biji kopi mentah dari tanaman *coffea*. Biji kopi pada dasarnya adalah biji kopi yang awalnya hijau, yang ada di lapisan dalam buah kopi berwarna merah. Setelah melalui beberapa proses, *Green Bean Coffee* siap untuk disangrai sesuai dengan keinginan Anda. Standar cafe dalam meroast *Greenbean coffee* adalah medium. Standar umumnya digunakan oleh warga negara Australia dan AS. Kopi hijau ini melewati 11 langkah proses kualitas pasca panen. Setelah buah panen, batas maksimal pengupasan adalah 8 jam. selama proses fermentasi 16 jam. Ini membantu menjaga kualitas. Proses penjemuran Tidak langsung di lantai ataupun di tanah

2.2.3 **Citra (image)**

Citra adalah kombinasi titik, garis, kotak, dan warna yang menciptakan suatu objek biasanya objek fisik atau manusia. Gambar dapata berupa gambar 2D seperti lukisan, foto dan video.

2.2.4 **Pengolahan Citra**

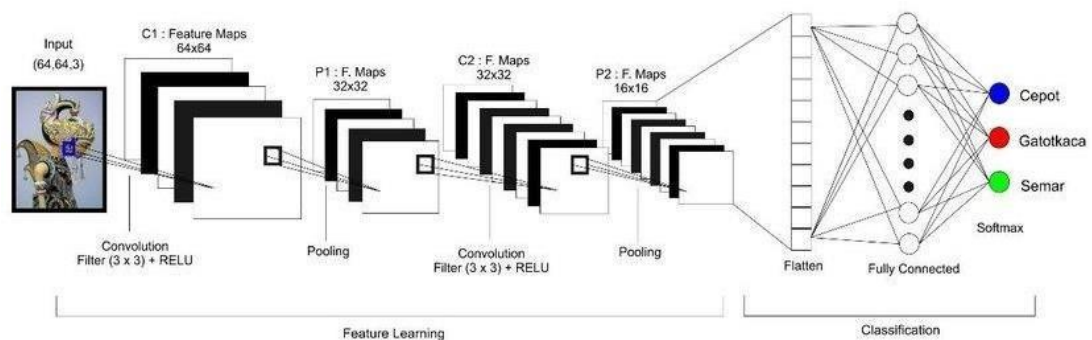
Pengolahan citra (image processing) merupakan proses mengolah pixel-pixel di dalam citra digital untuk tujuan tertentu. Pengolahan citra dilakukan karena beberapa alasan yaitu untuk mendapatkan citra asli dari suatu citra yang mengalami penurunan kualitas karena pengaruh derau atau untuk memperoleh citra dengan karakteristik dan cocok secara visual yang dibutuhkan untuk tahap lebih lanjut dalam proses analisis citra. Pengelompokan data numerik dan simbolik (termasuk citra) dilakukan secara otomatis oleh komputer. Tujuan dari pemrosesan ini adalah memperbaiki kualitas citra agar suatu objek dalam citra dapat dikenali dan diinterpretasi oleh manusia atau mesin komputer[10]

2.2.5 *Convolutional neural network (CNN)*

Convolutional Neural Network (CNN) adalah metode klasifikasi yang termasuk dalam kelompok *deep learning* yang menggunakan lapisan konvolusi untuk mengonvolusikan sebuah input dengan sebuah filter. *Convolutional Neural Network (CNN)* terdiri dari lapis masukan (input layer), lapis keluaran (output layer) dan sejumlah lapisan tersembunyi (Hidden Layers), lapisan ini terdiri dari dua tahapan utama yaitu fase pembelajaran fitur terdiri dari lapisan konvolusional,

ReLU(fungsi aktivasi) dan lapisan penyatuan, sedangkan fase klasifikasi terdiri dari lapisan dan prediksi yang datar dan terhubung sepenuhnya. Setiap bagian CNN memiliki dua proses utama yaitu *feedforward* dan *backpropagation* [11]. *Convolutional Neural Network* (CNN) merupakan pengembangan dari *Multilayer Perceptron* (MLP) yang termasuk dalam neural network bertipe *feed forward* (bukan berulang). *Convolutional Neural Network* sama seperti *neural network* yang didesain untuk mengolah data dua dimensi. CNN termasuk dalam jenis Deep Neural Network karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra. (Suartika, et.al, 2016).

Secara garis besar, *CNN* tidak terlalu jauh berbeda dengan neural network biasanya. *CNN* terdiri dari neuron yang memiliki weight, bias dan activation function. Arsitektur *Convolutional Neural Network* dapat dilihat pada gambar 2.1

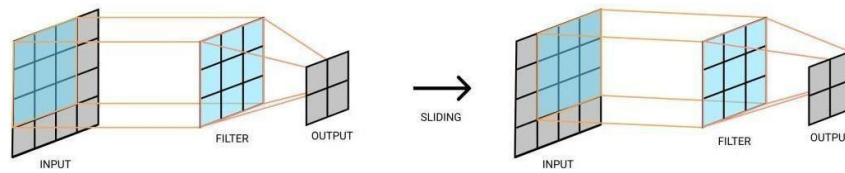


Gambar 2.2 Arsitektur Convolutional Neural Network (Nurhikmat, 2018)

Dari gambar 2.1 dapat dilihat ada 3 layer utama dalam *Convolutional Neural Network*, yaitu :

1. *Convolutional layer*

Convolution Layer merupakan hasil dari proses *convolution* dimana proses tersebut melakukan pengenalan pola, pola yang dimaksud meliputi tepi, bentuk, warna, tekstur pada suatu citra. Pola yang dihasilkan tergantung pada filternya, filternya berupa matriks dimana nilai awal adalah nilai random, kemudian ukuran filter menentukan nilai baris dan kolom.



Gambar 2.3 Proses Convolution [12]

filter akan memproses *dot product* pada neuron yang berada dalam ruang lingkup filter, kemudian melakukan proses *sliding* untuk menghitung nilai selanjutnya, hasil dari proses ini akan menjadi keluaran dari proses konvolusi, proses ini akan berlanjut sampai sebagai masukan yang diterima.[12] *Convolutional layer* memiliki *hyperparameter* dan parameter. *Hyperparameter* pada *layer* ini menjadi acuan untuk menentukan jumlah dan ukuran hasil ekstraksi *layer*, dapat dilihat dibawah.

Tabel 2.2 *Hyperparameter* Pada *Convolutional Layer*

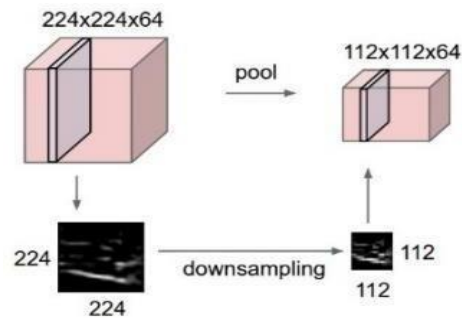
No	<i>Hyperparameter</i>	Keterangan
1	<i>Depth</i>	Kedalaman <i>layer</i> atau jumlah <i>layer</i> Konvolusi
2	<i>Stride</i>	Jumlah pergeseran <i>filter</i> pada proses Konvolusi
3	<i>Zero-padding</i>	Jumlah penambahan nilai intensitas nol di daerah sekitar input gambar

2. *Rectified Linear Units* (RELU)

Aktivasi ReLu (Rectified Linear Unit) adalah lapisan aktivasi pada model CNN yang mengaplikasikan fungsi $f(x) = \max(0, x)$ yang artinya fungsi ini menconvert semua nilai negatif menjadi nilai nol terhadap nilai piksel pada input citra. Maka aktivasi ini menormalisasikan nilai sehingga tidak ada nilai yang di bawah nol. [13]

3. *Pooling Layer*

Pooling Layer biasanya muncul setelah lapisan konvolusi. *pooling Layer* pada dasarnya terdiri dari filter ukuran dan kenaikan yang diterapkan ke semua area objek peta. *Pooling Layer* juga akan mereduksi ukuran spasial dan jumlah parameter dalam jaringan serta mempercepat komputasi dan mengontrol terjadinya overfitting (Samuel Sena, 2017)



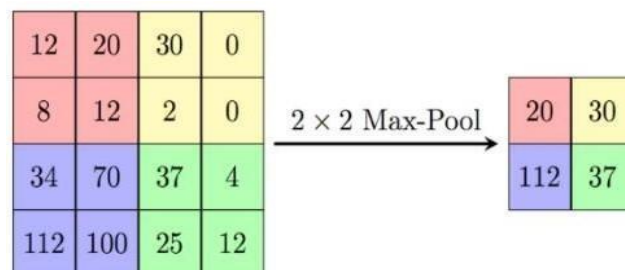
Gambar 2.4 Proses *Pooling Layer*

Pooling Layer bekerja dengan blok spasial yang bergerak sepanjang ukuran *feature pattern*. Ukuran pergeseran blok pada umumnya adalah ukuran pada dimensi blok ($H \times H$) itu sendiri sehingga tidak ada overlapping seperti pada *Convolutional layer*. Pergerakan blok diikuti dengan perhitungan *pooling* pada masukan pola fitur (u). Pada layer ini tidak memiliki parameter karena parameter sudah ditentukan sebelumnya (fixed). *Pooling layer* memiliki beberapa macam tipe antara lain sebagai berikut:

a. *Max Pooling*

Max Pooling output pola fitur (z) didapatkan dengan mencari nilai yang paling besar pada blok spasial pada setiap pergerakan. Metode ini merupakan metode standar yang digunakan dalam penelitian.

$$u_{i,j}^{(k)} = \max_{p,q \in P_{i,j}} z_{p,q}^{(k)} \quad (2-1)$$



Gambar 2.5 *Maxpooling* [13]

b. *Average Pooling*

Average Pooling Mendapatkan pola fitur keluaran dengan menghitung nilai rata-rata masukan pola fitur pada blok spasial $H \times H$ piksel di setiap wilayah pergerakan. *Average pooling* tetap menyimpan informasi spasial pada pola fitur sebelumnya.

$$u_{i,j}^{(k)} = \frac{1}{H^2} \sum_{(p,q) \in P_{ij}} z_{p,q}^{(k)} \quad (2-2)$$

c. *Lp Pooling*

Lp Pooling Adalah tipe *pooling Layer* generalisasi dari *Average pooling*.

$$u_{i,j}^{(k)} = \left(\frac{1}{H^2} \sum_{(p,q) \in P_{ij}} (z_{p,q}^{(k)})^P \right)^{\frac{1}{P}} \quad (2-3)$$

Ketika nilai $P = 1$, *pooling Layer* bekerja sebagai *average pooling* dan jika nilai $P = \infty$ *pooling Layer* bekerja sebagai *Max Pooling*. Sebagai contoh, input ke *pooling Layer* dengan ukuran $W1 \times H1 \times D1$. Untuk mendapatkan ukuran output dari *pooling Layer* dapat menggunakan persamaan 2-4.

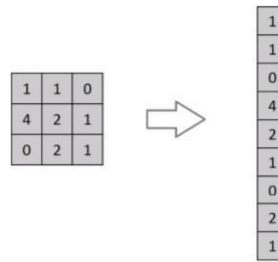
$$W2 = \frac{W1 - F}{S} + 1 = H2 \quad (2-4)$$

$$D2 = D1$$

**padding* biasanya dilakukan pada *pooling Layer*, namun mungkin ditemukan. Dimensi keluaran dari *Pooling Layer* juga menggunakan rumus yang sama seperti pada *convolution layer*. Tujuan penggunaan *pooling Layer* adalah untuk mengurangi ukuran *feature map* (downsampling), sehingga mempercepat proses komputasi, karena lebih sedikit parameter yang perlu diperbarui dan mengatasi *overfitting*. (Samuel Sena, 2017)

4. *Flatten*

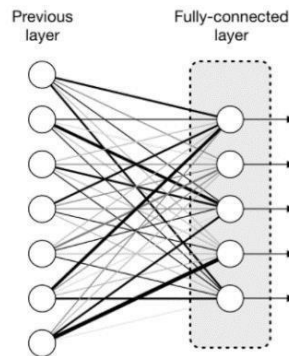
Flattening adalah operasi yang mengubah matriks menjadi vektor satu dimensi. Proses *flattening* mengubah *feature map* yang telah diperoleh dari layer sebelumnya menjadi vektor satu dimensi agar *feature map* tersebut dapat diklasifikasikan dengan *fully-connected layer* dan *softmax*. [13]



Gambar 2.6 Flatten [13]

5. Fully connected layer

Fully connected layer mengambil input dari hasil output *pooling layer* yang berupa *feature map*. *Feature map* dalam bentuk *multidimensional array*. Per lapisan ini akan membentuk kembali *feature map* dan menghasilkan vektor sebanyak n -dimensi di mana n adalah jumlah kelas output yang harus dipilih program. Misalnya, jika lapisan terdiri dari 500 neuron, fungsi aktivasi *softmax* akan diterapkan untuk mengembalikan daftar probabilitas terbesar untuk setiap 10 label kelas sebagai klasifikasi akhir jaringan.[14]



Gambar 2.7 Fully Connected Layer (KIDS, 2019)

6. Softmax

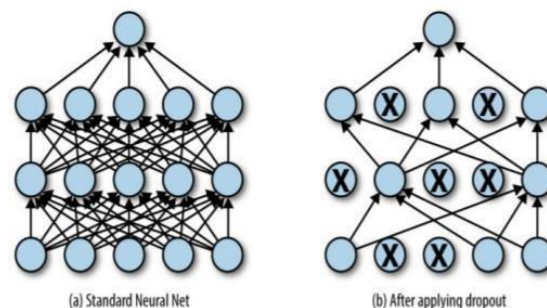
Softmax merupakan fungsi aktivasi yang digunakan pada layer output. Layer output memiliki banyak kesamaan dengan *fully-connected layer*, yang membedakan kedua layer ini adalah penggunaan fungsi aktivasi *softmax* pada layer output dan fungsi aktivasi ReLU pada *fully-connected layer*. [13]

$$f(x)_i = \frac{e^{x_i}}{\sum_{j=1}^k e^{x_j}} \text{ for } i = 1, 2, 3, \dots, k \quad (2-5)$$

Keuntungan utama menggunakan Softmax adalah rentang probabilitas output. Rentang output adalah 0 sampai 1, dan jumlah semua probabilitas dalam satu objek output adalah 1. Keunggulan lain dari softmax adalah dapat digunakan untuk *multiple classification logistic regression model* [15].

7. Dropout

Kemungkinan model *fully connected layer* akan menimbulkan *overfitting*. Oleh karena itu, diterapkan dropout pada model ini untuk mengurangi hal tersebut.



Gambar 2.8 Dropout [13]

Dropout akan memilih secara acak neuron mana yang akan di non-aktifkan sehingga tidak terlalu banyak weight dan neuron yang terlibat di dalam perhitungan layer neural network. Probabilitas neuron yang dilibatkan dalam perhitungan akan di tetapkan misalnya antara 50% sampai 70%, neuron yang tidak aktif ditandai dengan tanda silang.[13]

1. Confusion Matrix

Confusion matrix adalah metode untuk melakukan perhitungan yang akurat untuk konsep data mining. Evaluasi dengan matriks kebingungan memberikan nilai presisi, akurasi, dan *recall*. *Confusion matrix* memberikan keputusan yang diperoleh dari penilaian *performance* klasifikasi berdasarkan objek dengan benar atau salah (*gurunescu*). *Confusion matrix* berisikan informasi yang berisi informasi *actual* dan prediksi pada sistem klasifikasi.[12]

Tabel 2.3 *Confusion Matrix*

		Actual Class	
		Positive (P)	Negative(N)
Predicted Class	Positive(P)	True Positive(TP)	False Positive(FP)
	Negative(N)	False Negative(FN)	True Negative(TN)

Nilai akurasi menunjukan seberapa akurat sistem dapat mengklasifikasikan data dengan benar. Dengan kata lain skor akurasi adalah perbandingan antar dua yang terklasifikasi benar dengan total data dan dapat diperoleh dari skor akurasi, presisi, dan recall.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} * 100\%$$

$$Presisi = \frac{TP}{TP + FB} * 100\%$$

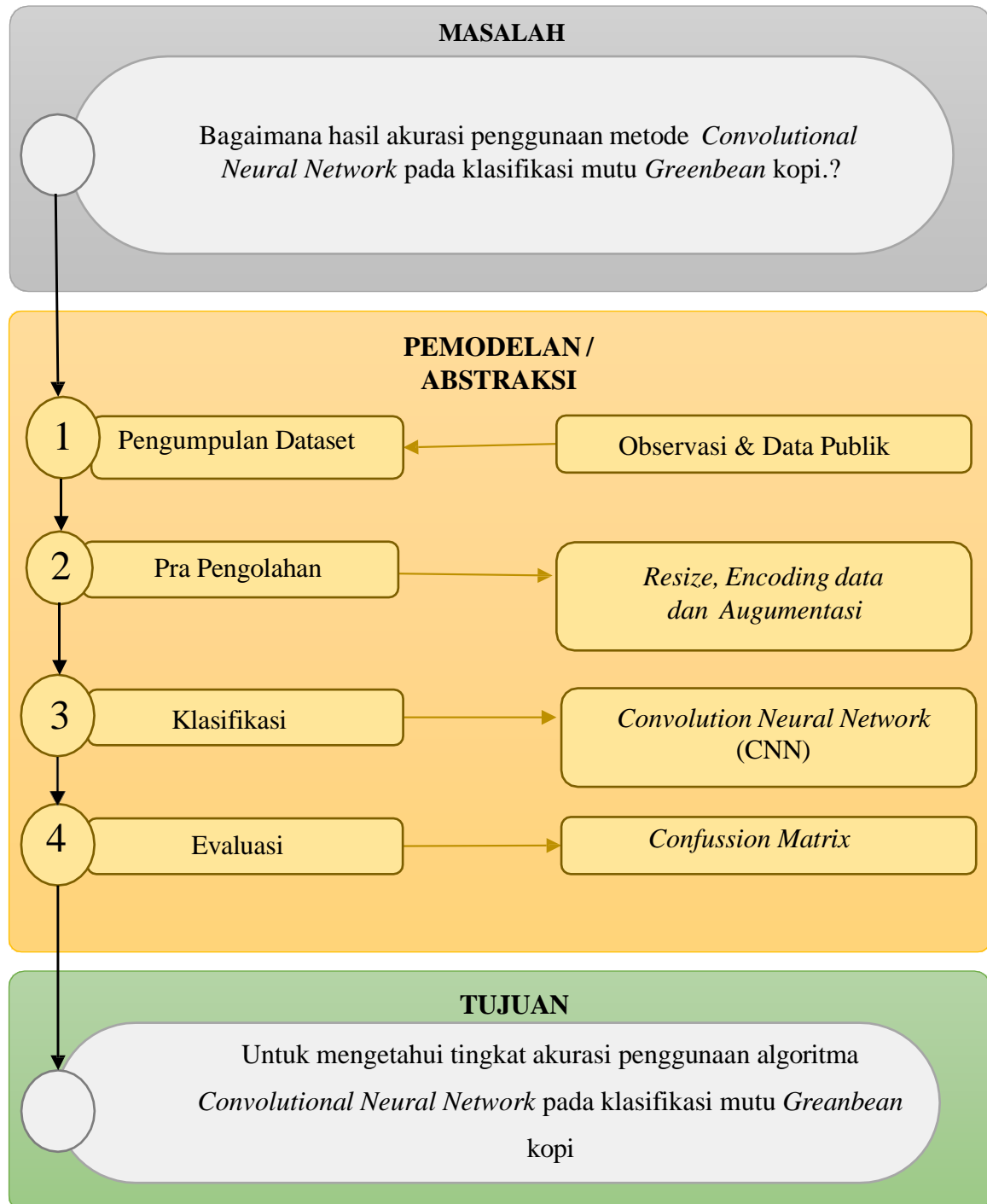
$$Recall = \frac{TP}{FN + TP} * 100\%$$

Keterangan :

- TP (True Positive) – Jumlah data dengan nilai sebenarnya positif dan nilai prediksi positif
- FP (False Positive) – Jumlah data dengan nilai sebenarnya negatif dan nilai prediksi positif
- FN (False Negative) – Jumlah data dengan nilai sebenarnya positif dan nilai prediksi negative
- TN (True Negative) – Jumlah data dengan nilai sebenarnya negatif dan nilai prediksi negative

Sumber : *The Science Of Machine Learning* (2018)

2.3 Kerangka Pikir



BAB III

METODE PENELITIAN

3.1 Jenis , Metode , Subjek , Objek dan Waktu Penelitian

Pada level aplikasi, penelitian ini merupakan penelitian terapan. Karena sifat data yang ditangani, maka penelitian ini merupakan penelitian kuantitatif. Dalam hal pengolahan data, penelitian ini merupakan penelitian konfirmatori.

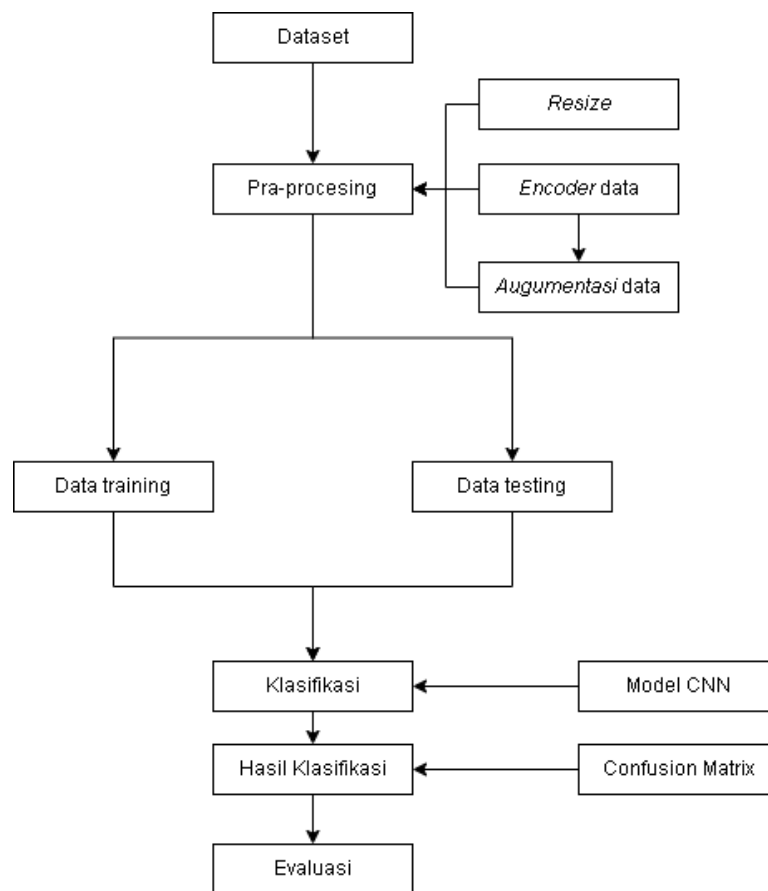
Penelitian ini menggunakan metode penelitian *eksperimen*. Dengan demikian jenis penelitian ini adalah penelitian *eksperimental*. Subjek penelitian ini adalah klasifikasi, pada objek kopi. Penelitian ini dimulai dari Februari 2020

3.2 Pengumpulan Data

Data set yang digunakan dalam penelitian ini adalah dataset public dengan nama *Multi Class Dataset Composed Of The Various Greenbean Coffee*. Data Set yang di unduh dari <https://comvis.unsyiah.ac.id/usk-coffee/>, Dataset ini merupakan data yang pada umumnya digunakan dalam eksperimen untuk klasifikasi greenbeancoffee. Dataset ini akan dibagi menjadi dua bagian yaitu data training dan data testing. Berikut pendelasan mengenai *Multi Class Dataset Composed Of The Various Greenbean Coffee* :

Jumlah Gambar	: 3.200
Format Gambar	: JPG
Resolusi	: High Resolution
URL	: https://comvis.unsyiah.ac.id/usk-coffee/ ,

3.3 Pemodelan



Gambar 3.1 Pemodelan

3.3.1 *Pra-Processig*

Dari data set atau data publik akan dilakukan proses *grayscale* pada gambar dari citra berwarna di konversi ke citra *grayscale*. kemudian akan transformasi atau encoding data, selanjutnya augumentasi gambar atau memanipulasi gambar sedemikian rupa sehingga komputer akan mendeteksi bahwagambar yang diubah adalah gambar yang berbeda. Agar memperbanyak jumlah data yang akan diolah agar mendapatkan model yang terbaik. Berikut adalah tahapan Pra-pengolahan :

1. *Resize*

Resize merupakan tahapan awal dari *pre-processing*, bertujuan untuk mengubah ukuran citra asli dari ukuran 256x256 piksel menjadi 48x48 piksel lebih kecil dari skala citra asli agar mempercepat proses latih.

2. *Encoding Data (Label Encoder)*

Pada proses ini akan dilakukan tranformasi label menjadi bentuk numerik. Agar dapat diolah oleh sistem.

3. *Augumentasi Data*

Augumentasi yaitu suatu teknik manipulasi pada data yang ada tanpa kehilangan inti dari data tersebut. Pada penelitian ini dilakukan teknik augumentasi terhadap data training dengan mengatur *rescale* 1./255, *rotation range* , *horizontal flip*, *with shif range* 0.1, *height shif range* 0.1, dan *fill mode*. Beberapa cara tersebut membantu model yang akan disusun serta mempermudah untuk melatih model tersebut.

3.3.2 Data Training

Data training berupa data yang telah diseimbangkan jumlahnya untuk di olah menggunakan proses *resampling* data yang nantinya akan dilatih menggunakan algoritma *Convolutional Neural Network* (CNN), jumlah data training yang digunakan berupa 80% yang akan di proses di training CNN.

3.3.3 Training CNN

Training menggunakan *Convolutional Neural Network* (CNN) yaitu menjadikan data training menjadi data inputan. Algoritma *Convolutional Neural Network* (CNN) bekerja dengan proses konvolusi dengan menerapkan karnel konvolusi (filter) dengan ukuran tertentu pada gambar, dalam hal ini komputer memperoleh informasi perwakilan baru dengan mengalikan bagian gambar dengan filter yang digunakan, setelah itu nilai matrixnya akan di normalisasikan dengan fungsi aktivasi RELU, kemudian akan dilakukan proses *Pooling Layer* untuk mengecilkan ukuran gambar, 3 proses diatas sebagai proses fitur ekstrasi dan selanjutnya akan di lakukan proses klasifikasi, pada proses klasifikasi ada yang namanya Flatten atau pengubahan dari bentuk matriks ke vektor atau bentuk satu dimensi tetapi tetap mempertahankan informasi spasial dari tiap gambar, setelah itu akan dilakukan proses *fully connected layer* untuk menentukan *class* nya. Dan dari

proses training ini menghasilkan model yang sudah dilatih dari proses training ini dan akan menjadi ketentuan untuk dijadikan pengenalan untuk data testing.

3.3.4 Data Testing

Data testing merupakan data yang telah diseimbangkan jumlahnya yang digunakan untuk menguji data training. Jumlah data yang digunakan 20%. Data testing digunakan untuk mengetahui berapa tingkat keberhasilan untuk melakukan klasifikasi mutu biji kopi.

3.4 Evaluasi Model

Proses evaluasi bertujuan untuk mengetahui hasil kerja dari metode yang digunakan, evaluasi dilakukan pada semua data uji kemudian target kelusssaran dari hasil tersebut dihasilkan akan dipetakan ke dalam *Confussion Matrix* untuk dievaluasi keakuratannya.


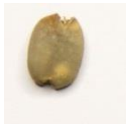


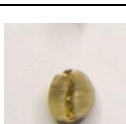
BAB IV

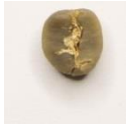

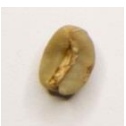
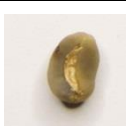
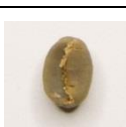
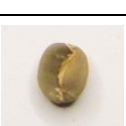
HASIL PENELITIAN

4.1 Hasil Pengumpulan data

Data set yang digunakan dalam penelitian ini adalah dataset *public* dengan nama multi class dataset composed of the various greenbean coffee. Data Set yang di unduh dari <https://comvis.unsyiah.ac.id/usk-coffee/>. Dataset ini merupakan data yang pada umumnya digunakan dalam eksperiment untuk klasifikasi greenbeancoffee. Dataset ini akan dibagi menjadi dua bagian yaitu data traning dan data testing, Dataset ini merupakan data yang pada umumnya digunakan dalam eksperiment untuk klasifikasi jenis greenbean coffee Dengan jumlah data training 2400 image dan data testing 800 image

Tabel 4.1 Kopi Buruk dan Baik

No	Gambar	Hasil
1.		Buruk
2.		Buruk
3.		Buruk
4.		Buruk
5.		Buruk

No	Gambar	Hasil
6.		Baik
7.		Baik
8.		Baik
9.		Baik
10.		Baik
.
.
.
.
3200.		Baik

4.2 Hasil Pemodelan

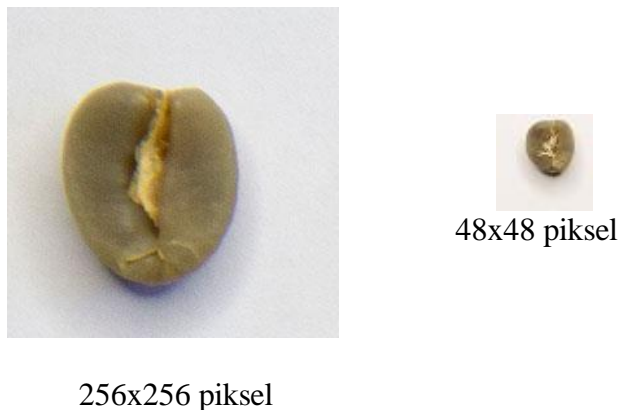
4.2.1 Pra Pengolahan Data

Pra-pengolahan citra diartikan sebagai pemrosesan awal dalam proses pengolahan citra sebelum proses utama dilakukan. Pra-pengolahan bertujuan untuk mempermudah proses dalam melakukan klasifikasi citra. Pada tahapan ini citra dari *Greenbean coffee* yang sudah dikonversikan agar didapatkan data yang sesuai dengan kebutuhan. Tahapan ini berfungsi untuk menormalisasikan data citra *Greenbean coffee*, kemudian akan dilakukan encoding data atau perubahan data

label dari kata menjadi numerik. Selanjutnya dilakukan augmentasi data agar dapat menambah jumlah data menjadi lebih banyak. Pra-pengolahan dalam penelitian ini dilakukan dengan tahapan sebagai berikut :

1. *Resize*

Resize merupakan tahapan awal dari *pre-processing*. Bertujuan untuk mengubah ukuran citra asli dari ukuran 256x256 piksel menjadi 48x48 piksel lebih kecil dari skala citra asli agar mempercepat proses latih.



2. *Encoding Data (Label Encoder)*

Pada proses ini akan dilakukan transformasi label kata menjadi bentuk numerik. Agar dapat diolah oleh program.

```
baik > 0
rusak > 1
```

Gambar 4.1 Hasil *Encoder*

3. *Augumentasi Data*

Augumentasi yaitu suatu teknik manipulasi pada data yang ada tanpa kehilangan inti dari data tersebut. Pada penelitian ini dilakukan teknik *augumentasi* terhadap data training dengan mengatur *rescale* 1./255, *rotation range* , *horizontal flip*, *with shif range* 0.1, *height shif range* 0.1, dan *fill mode*. Beberapa cara tersebut membantu model yang akan disusun serta mempermudah untuk melatih model tersebut.

4.2.2 Data Training

Data training berupa data yang telah diseimbangkan jumlahnya untuk diolah menggunakan proses *resampling* data yang nantinya akan dilatih menggunakan algoritma *Convolutional Neural Network* (CNN), jumlah data training yang digunakan berupa 80% yang akan di proses di training CNN.

4.2.3 Training CNN

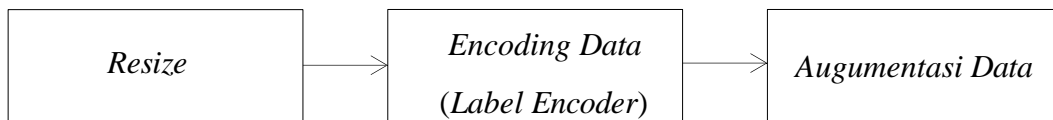
Training menggunakan *Convolutional Neural Network* (CNN) yaitu menjadikan data training menjadi data inputan. Algoritma *Convolutional Neural Network* (CNN) bekerja dengan proses konvolusi dengan menerapkan karnel konvolusi (filter) dengan ukuran tertentu pada gambar, dalam hal ini komputer memperoleh informasi perwakilan baru dengan mengalikan bagian gambar dengan filter yang digunakan, setelah itu nilai matrixnya akan di normalisasikan dengan fungsi aktivasi RELU, kemudian akan dilakukan proses *pooling Layer* untuk mengecilkan ukuran gambar, 3 proses diatas sebagai proses fitur ekstrasi dan selanjutnya akan di lakukan proses klasifikasi, pada proses klasifikasi ada yang namanya *Flatten* atau pengubahan dari bentuk matriks ke vektor atau bentuk satu dimensi tetapi tetap mempertahankan informasi spasial dari tiap gambar, setelah itu akan dilakukan proses *fully connected layer* untuk menentukan *class* nya. Dan dari proses training ini menghasilkan model yang sudah dilatih dari proses training ini dan akan menjadi ketentuan untuk dijadikan pengenalan untuk data testing.

4.2.4 Data Testing

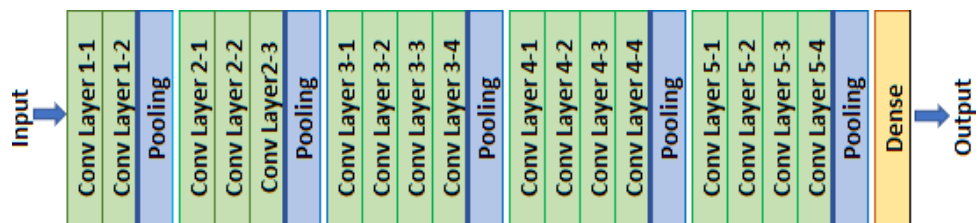
Data testing merupakan data yang telah diseimbangkan jumlahnya yang digunakan untuk menguji data training. Jumlah data yang digunakan 20%. Data testing digunakan untuk mengetahui berapa tingkat keberhasilan untuk melakukan klasifikasi mutu biji kopi.

4.2.5 Convolutional Neural Network (CNN)

Implementasi metode CNN terdapat beberapa tahapan mulai dari input data, menentukan pengaturan untuk model yang akan di terapkan sampai pada output dari kelas yang diklasifikasi. Serta mengatur nilai epoch untuk melatih data agar menghasilkan hasil yang sesuai. Model arsitektur CNN yang digunakan pada penelitian ini adalah model arsitektur VGG16. Dan berikut ini adalah tahapan implementasi model CNN yang peneliti buat:



Gambar 4.2 Tahapan Pra-pengolahan

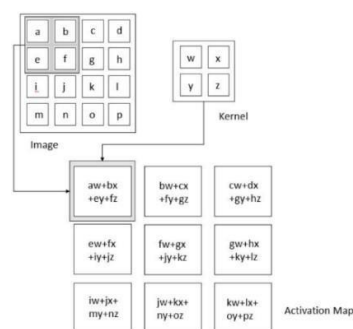


Gambar 4.3 Model Arsitektur CNN

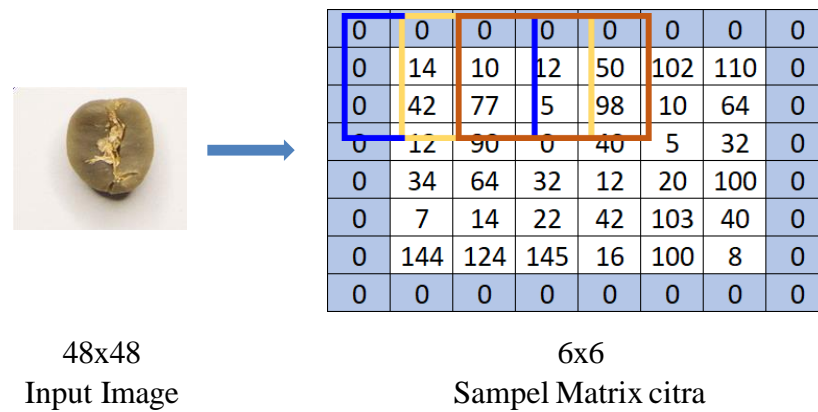
Proses pertama metode *Convolutional Neural Network* :

1. Convolutional Layer

Pada layer ini akan dilakukan perhitungan antara dua matriks, dimana satu matriks adalah parameter yang dipelajari atau kernel, dan matriks lainnya adalah pixel citra.

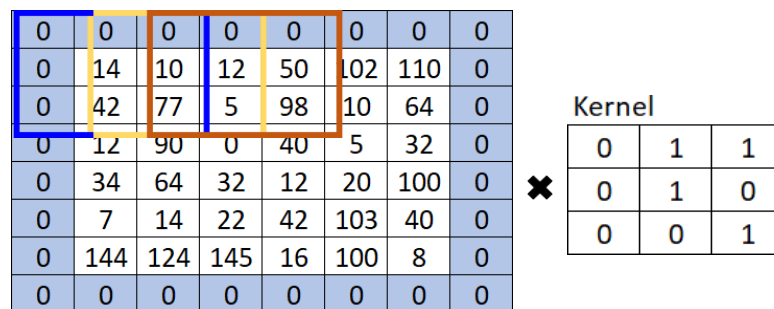


Gambar 4.4 Operasi Konvolusi



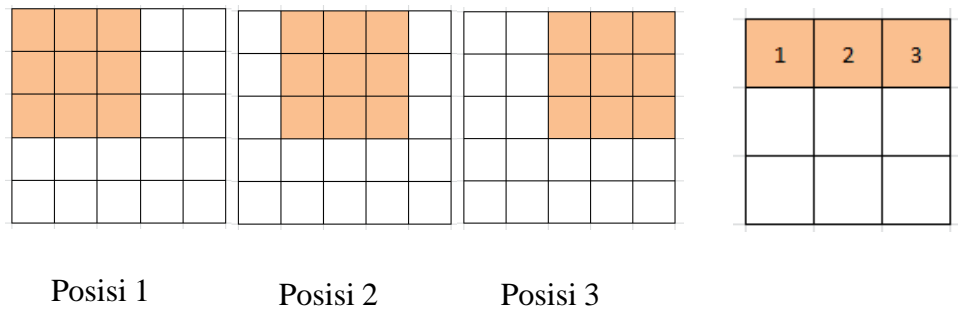
Gambar 4.5 Sampel Gambar dan Input

Gambar 4.5 adalah gambar yang menunjukkan dataset yang tersedia. Ukuran piksel dari citra masukan adalah 48x48, sehingga penulis hanya mengambil sebagian dari piksel tersebut atau 6x6. Gambar 4.6 menunjukkan proses konvolusi menggunakan karnel.



Gambar 4.6 Proses Konvolusi dengan Karnel 3x3

Gambar 4.6, Menunjukkan perhitungan pada proses konvolusi dimana sebuah karnel ukuran 3x3 yang dimulai pada sisi bagian kiri dapat dilihat pada gambar 4.7. proses ini disebut jendela geser. Pada penelitian ini dilaporkan nilai padding 1. Menambahkan nilai nol disekitar nilai matriks input sehingga input dan output memiliki nilai matriks yang sama dan informasi gambar tidak berkurang.



Gambar 4.7 Proses pergeseran dengan *Stride* = 1 dan Kernal 3x3

Perhitungan manual metode *convolutional neural network* untuk proses konvolusi. Dari pixel matriks 48 x 48 x 1 diambil contoh dengan ukuran matriks citra = 6 x 6 x 1, kernel = 3 x 3 x 1, S = 1, P = 1.

Awalnya akan dihitung output volume-nya : $V = \frac{W - F + 2P}{S} + 1$

W = input size

P = padding

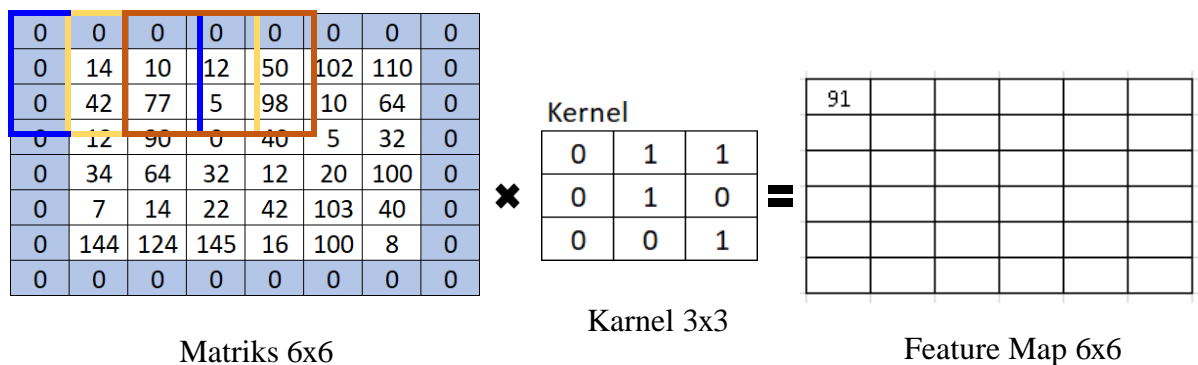
V = Volume

F = kernel size

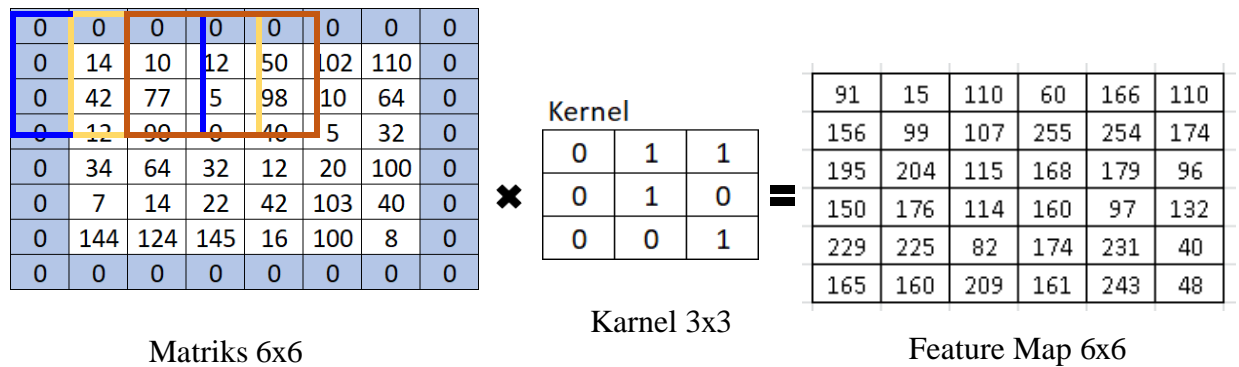
S = Stride

Masukkan nilai-nya $V = \frac{6 - 3 + 2(1)}{1} + 1 = 6$

jadi Output volumenya 6 x 6 x 1.



Gambar 4.8 Proses konvolusi pada pergeseran pertama



Gambar 4.9 Proses konvolusi pada pergeseran Terakhir

Adapun perhitungan yang dilakukan untuk mendapat hasil pada Gambar 4.8 dan gambar 4.9 adalah sebagai berikut :

Posisi 1 :

$$= 0 \times 0 + 0 \times 1 + 0 \times 1 + 0 \times 0 + 14 \times 1 + 10 \times 0 + 0 \times 0 + 42 \times 0 + 77 \times 1$$

$$= \mathbf{91}$$

Posisi 2 :

$$= 0 \times 0 + 0 \times 1 + 0 \times 1 + 14 \times 0 + 10 \times 1 + 12 \times 0 + 42 \times 0 + 77 \times 0 + 5 \times 1$$

$$= \mathbf{15}$$

Posisi 3 :

$$= 0 \times 0 + 0 \times 1 + 0 \times 1 + 10 \times 0 + 12 \times 1 + 50 \times 0 + 77 \times 0 + 5 \times 0 + 98 \times 1$$

$$= \mathbf{110}$$

Seterusnya cara perhitungannya sama kemudian dari perhitungan diatas didapatkan output konvolusi atau *feature map* dengan 6x6x1

91	15	110	60	166	110
156	99	107	255	254	174
195	204	115	168	179	96
150	176	114	160	97	132
229	225	82	174	231	40
165	160	209	161	243	48

Gambar 4.10 *Feature Map*

Setelah mendapatkan output konvolusi atau *feature map*, maka akan dilakukan *pooling layer* atau mengurangi ukuran dimensi *feature map* sehingga mempercepat komputasi. *Feature map* merupakan pemetaan citra atau intisari citra dipetakan dari hasil konvolusi.

2. *Pooling layer*

Dari hasil proses konvolusi di dapatkan featur map selanjutnya akan di lakukan pooling layer menggunakan ukuran filter 2 x 2. Menggunakan max pooling maka tiap filter matriks akan di ambil nilai paling besar dan akan dimasukkan ke dalam feature map. Pada matriks di bawah tiap blok warna menandakan tiap pergeseran window dengan matriks 2 x 2 masing-masing window akan di ambil nilai terbesar, kemudian nilai yang paling besar akan menjadi *feature map* baru.

91	15	110	60	166	110
156	99	107	255	254	174
195	204	115	168	179	96
150	176	114	160	97	132
229	225	82	174	231	40
165	160	209	161	243	48

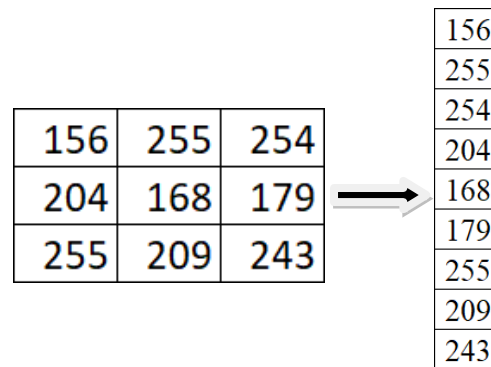
156	255	254
204	168	179
255	209	243

Gambar 4.11 *Max Pooling*

Dari hasil pooling layer didapatkan feature map dengan ukuran 3 x 3. Proses konvolusi dan pooling layer ini akan terus di ulang sesuai dengan arsitektur model CNN yang digunakan. Setelah pooling layer citra baru akan dilakukan proses *flattening*.

3. *Flatten*

Pada tahap ini feature map akan di lakukan proses flattening atau mengubah matriks menjadi bentuk vektor atau satu dimensi.

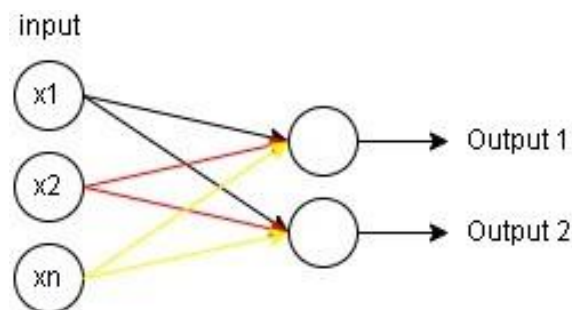


Gambar 4.12 Matriks dan Vektor

4. *Fully Connected Layer*

Fully connected layer berperan untuk mengklasifikasikan data masukan.

Padaproses training ini berguna untuk mencari nilai bobot terbaik.



Gambar 4.13 *Fully Connected Layer*

Setelah mendapatkan output konvolusi atau feature map, maka akan dilakukan pooling Layer atau mengurangi ukuran dimensi feature map sehingga mempercepat komputasi. Featur map merupakan pemetaan citra atau intisari citra yang dipetakan dari hasil konvolusi.

1. Inisialisasi semua bobot,

Bobot awal atau $w = 0$, untuk semua inputan.

2. Fungsi aktivasi yang digunakan

Aktivasi yang digunakan pada penelitian ini adalah $\sigma = \text{softmax}$

3. Laju Pembelajaran atau *Learning rate*

Pada penelitian ini ditentukan nilai lerning rate nya $\alpha = 0.1$

4. Menghitung persamaan nilai output :

$$f(x) = \sum_{x=1}^n (X1W1 + X2W2 + \dots + XnWn)$$

$$\sigma y_j(P) = \frac{e^{x_i}}{\sum_{j=i} e^{x_j}}$$

$$f(x) = \sum_{x=i}^9 (x1w11 + x2w22 + \dots + x9w91)$$

$$f(x) = \sum_{x=i}^9 (156.0 + 255.0 + 254.0 + 204.0 + 168.0 + 179.0 + 255.0 + 209.0 + 243.0 = 0)$$

$$f(x) = \sum_{x=i}^9 (156.0 + 255.0 + 254.0 + 204.0 + 168.0 + 179.0 + 255.0 + 209.0 + 243.0 = 0)$$

$$f(x) = \sum_{x=i}^9 (156.0 + 255.0 + 254.0 + 204.0 + 168.0 + 179.0 + 255.0 + 209.0 + 243.0 = 0)$$

Softmax:

$$\sigma y(1) = \frac{e^{x_i}}{\sum_{j=i} e^{x_j}} = \frac{2.71828^0}{2.71828^0 + 2.71828^0 + 2.71828^0} = \frac{1}{3} = 0.33$$

Nilai sama dengan nilai output lainnya dikarenakan menggunakan nilai $w = 0$

1. Hitung gradien *error* untuk *layer output*

$$e_k(p) = y_{dk}(p) - y_k(p)$$

$$\delta_k(p) = y_k(p) * [1 - y_k(p)] * e_k(P)$$

Untuk data pertama, nilai keluaran yang didapatkan adalah $y_{dk} = 0.33$, nilai yang diharapkan $y_k = 2$ nilai neuron output semuanya sama karena nilai $w = 0$.

$$e_2(1) = y_d - y_1(1) = 2 - 0.33 = -1.67$$

$$\delta_2(1) = y_2(1) * [1 - y_2(1)] * e_1(1) = 2 * [1 - 2] * -1.67 = 3.34$$

2. Menghitung Koreksi Bobot

Untuk Δw_{12} , Δw_{22} , dan Δw_{32} :

$$\Delta w_{jk}(p) = \alpha \cdot y_j(p) * \delta_k(p)$$

$$\Delta w_{12} = \alpha * y_1(1) * \delta_1(1) = 0.002 * 156 * 3.34 = 1.04208$$

$$\Delta w_{22} = \alpha * y_2(1) * \delta_1(1) = 0.002 * 255 * 3.34 = 1.7034$$

$$\Delta w_{32} = \alpha * y_3(1) * \delta_1(1) = 0.002 * 254 * 3.34 = 1.69672$$

3. Perbaharui Bobot

Dengan diperolehnya nilai pengkoreksi bobot maka proses memperbaharui nilai bobot dengan menggunakan persamaan :

$$w_{jk}(p + 1) = w_{jk}(p) + \Delta w_{jk}(p)$$

$$(p + 1) = w_{12}(p) + \Delta w_{12}(p) = 0 + 0.52104 = 1.04208$$

$$(p + 1) = w_{22}(p) + \Delta w_{22}(p) = 0 + 0.8517 = 1.04208$$

Setelah itu untuk iterasi selanjutnya, kembali ke langka 5 dan diulangi sampai mendapatkan hasil yang sesuai. Setelah itu jika didapatkan hasil yang sesuai, lalu di klasifikasikan citra sesuai dengan kelasnya.



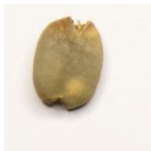













4.3 Evaluasi Model














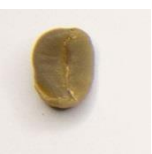
4.3.1 Confusion Matrix

Pada penelitian ini menggunakan confusion matrix sebagai metode. Dalam perhitungan akurasi untuk klasifikasi mutu *Greenbeen Coffee*. Evaluasi kinerja pengenalan mutu *Greenbeen Coffee* didasarkan dari jumlah pengujian objek salah dan benar yang dideteksi dapat dilihat dari tabel berikut:

\

Tabel 4.2 Hasil Data Testing

No	Image testing	Kelas	Hasil Klasifikasi	Image testing	Kelas	Hasil Klasifikasi
1.		Buruk	Buruk		Buruk	Buruk
2.		Buruk	Buruk		Buruk	Buruk
3.		Buruk	Baik		Buruk	Buruk
4.		Buruk	Buruk		Buruk	Buruk
5.		Buruk	Buruk		Buruk	Buruk
6.		Buruk	Buruk		Buruk	Buruk
7.		Buruk	Buruk		Buruk	Buruk
8.		Buruk	Buruk		Baik	Buruk

No	Image testing	Kelas	Hasil Klasifikasi	Image testing	Kelas	Hasil Klasifikasi
9.		Baik	Buruk		Baik	Baik
10.		Baik	Baik		Baik	Baik
11.		Baik	Baik		Baik	Baik
12.		Baik	Baik		Baik	Baik
13.		Baik	Baik		Baik	Baik
14.		Baik	Baik		Baik	Baik
15.		Baik	Baik		Baik	Baik

Tabel 4.3 Hasil Evaluasi Confusion Matrix

Prediksi	Aktual	
	Baik	Buruk
Baik	13	2
Buruk	1	14

$$\begin{aligned}
 Accuracy &= \frac{TP + TN}{TP + FP + FN + TN} = \frac{13 + 14}{13 + 2 + 1 + 14} \\
 &= \frac{27}{30} = 0.9 * 100 = 90\%
 \end{aligned}$$

$$Precision = \frac{TP}{TP + FP} = \frac{13}{13 + 2} = 0.86 = 0.86 * 100\% = 86\%$$

$$Recall = \frac{TP}{TP + FN} = \frac{13}{13 + 1} = 0.92 = 0.92 * 100\% = 92\%$$

$$\begin{aligned}
 f1 - score &= 2x \frac{(recall * precision)}{(recall + precision)} = 2 * \frac{(92 * 86)}{(92 + 86)} = 2 * \frac{7.912}{178} \\
 &= 2 * 44.4 = 88
 \end{aligned}$$

BAB V

PEMBAHASAN

5.1 Pembahasan Model

Berdasarkan perolehan hasil penelitian yang telah dilakukan pada bab sebelumnya, maka dapat dijelaskan beberapa hal sebagai berikut :

- Data gambar yang berhasil dikumpulkan pada penelitian ini adalah sebanyak 3200 gambar *Greenbean Coffee*, yang dibagi kedalam dua kelas yaitu baik dan buruk. Berdasarkan visualisasi citra bahwa kedua jenis kelas *Greenbean Coffee* tersebut dapat dibedakan pada bentuknya.
- Kemudian data diproses menggunakan 3 tahapan pra-pengolahan sebelum data pada penelitian ini diproses menggunakan metode *Convolutional Neural Network*. Adapun proses pra-pengolah sebagai berikut :

1) *Resize*

Resize merupakan tahapan awal dari *pre-processing*, bertujuan untuk mengubah ukuran citra asli dari ukuran 256x256 piksel menjadi 48x48 piksel lebih kecil dari skala citra asli agar mempercepat proses latih.

```
def generate_image(image_source_path):  
    im_list = []  
    labels = []  
    for path in paths.list_images(image_source_path):  
        img = cv2.imread(path)  
        #img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)  
        img = cv2.resize(img, (48, 48))  
        im_list.append(img)  
        labels.append(os.path.split(os.path.dirname(path))[-1])  
        df_label = pd.DataFrame(labels)  
    return im_list, df_label
```

Gambar 5.1 Proses Resize Gambar

2) Encoding Data(Label Encoder)

Pada proses ini akan dilakukan tranformasi label menjadi bentuk numerik. Agar dapat diolah oleh sistem.

```
from sklearn.preprocessing import LabelEncoder
encoder = LabelEncoder()
encoder.fit(y_train)
y_train = encoder.transform(y_train)

y_test = encoder.transform(y_test)
```

```
for i, item in enumerate(encoder.classes_):
    print(item, '>', i)
```

```
baik > 0
rusak > 1
```

Gambar 5.2 Coding Label Encoder dan Hasil Encoder

3) Augumentasi Data

Augumentasi yaitu suatu teknik manipulasi pada data yang ada tanpa kehilangan inti dari data tersebut. Pada penelitian ini dilakukan teknik augumentasi terhadap data training dengan mengatur *rescale* 1./255, *rotation range* , *horizontal flip*, *with shif range* 0.1, *height shif range* 0.1, dan *fill mode*. Beberapa cara tersebut membantu model yang akan disusun serta mempermudah untuk melatih model tersebut.

```
from keras.preprocessing.image import ImageDataGenerator
datagen = ImageDataGenerator(
    rescale=1./255,
    rotation_range = 10,
    horizontal_flip = True,
    width_shift_range=0.1,
    height_shift_range=0.1,
    fill_mode = 'nearest')

testgen = ImageDataGenerator(rescale=1./255)
datagen.fit(x_train)
```

Gambar 5.3 Coding Augumentasi

Dari data augmentasi tersebut dapat dijelaskan sebagai berikut :

- Pada percobaan yang dilakukan pada penelitian ini fitur citra di ekstraksi menggunakan metode CNN karena metode ini tidak perlu menggunakan metode ekstraksi tambahan. fitur ekstraksi CNN melakukan proses konvolusi dan pooling layer, sesuai dengan model yang dibuat. *Pooling Layer* dipakai ditiap blok agar mengurangi ukuran pixel yang akan diolah sehingga dapat mengurangi waktu komputasi dari algoritma CNN yang digunakan pada penelitian ini sehingga penggunaan algoritma CNN ini menjadi lebih efisien untuk diterapkan pada aplikasi yang akan dikembangkan kedepannya.
- Arsitektur CNN dibuat dengan 5 blok, blok pertama dua kali konvolusi kemudian dilakukan maxpooling selanjutnya untuk blok ke dua tiga kali konvolusi kemudian di maxpooling, selanjutnya untuk blok ke tigasampai lima dilakukan konvolusi sebanyak empat kali lalu di maxpooling, kemudian dilakukan proses dense dengan 3 output atau keluaran menggunakan fungsi aktivasi softmax.
- Dari proses konvolusi dan pooling didapatkan matriks citra untuk dijadikan variabel input pada algoritma klasifikasi yang digunakan.
- Pada penelitian yang dilakukan, algoritma CNN yang digunakan sebagai fitur ekstraksi dan juga sebagai pengklasifikasi data citra digunakan sebagaimana yang telah digambarkan pada bab sebelumnya.
- Dalam proses training data menggunakan algoritma CNN dengan nilai learning rate sebesar 0.1, batchsize dengan nilai 7 dan jumlah epochs sebanyak 150.

```
history = model.fit(train_flow,
                    steps_per_epoch=len(x_train)/batch_sizes,
                    epochs =150,
                    verbose=1,
                    validation_data = test_flow)
```

Gambar 5.4 Coding Model fitting

```

Epoch 142/150
342/342 [=====] - 10s 29ms/step - loss: 0.3573 - accuracy: 0.8662 - val_loss: 0.6455 - val_accuracy: 0.7200
Epoch 143/150
342/342 [=====] - 11s 31ms/step - loss: 0.4462 - accuracy: 0.8142 - val_loss: 0.5012 - val_accuracy: 0.7875
Epoch 144/150
342/342 [=====] - 11s 32ms/step - loss: 0.3313 - accuracy: 0.8875 - val_loss: 0.4288 - val_accuracy: 0.8100
Epoch 145/150
342/342 [=====] - 11s 31ms/step - loss: 0.3578 - accuracy: 0.8596 - val_loss: 0.4682 - val_accuracy: 0.8125
Epoch 146/150
342/342 [=====] - 11s 32ms/step - loss: 0.3046 - accuracy: 0.8921 - val_loss: 0.5091 - val_accuracy: 0.8012
Epoch 147/150
342/342 [=====] - 10s 30ms/step - loss: 0.3253 - accuracy: 0.8850 - val_loss: 0.5179 - val_accuracy: 0.8012
Epoch 148/150
342/342 [=====] - 11s 32ms/step - loss: 0.3000 - accuracy: 0.8888 - val_loss: 0.7971 - val_accuracy: 0.7250
Epoch 149/150
342/342 [=====] - 11s 32ms/step - loss: 0.3094 - accuracy: 0.8842 - val_loss: 0.7757 - val_accuracy: 0.6100
Epoch 150/150
342/342 [=====] - 11s 31ms/step - loss: 0.3216 - accuracy: 0.8879 - val_loss: 0.4532 - val_accuracy: 0.7862

```

Gambar 5.5 Hasil Model Fiting

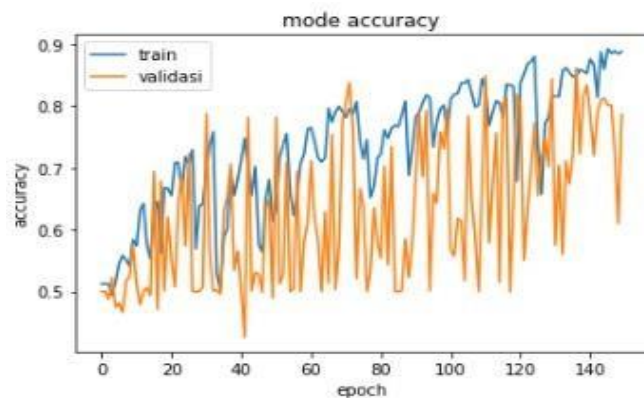
Berdasarkan hasil data learning yang dilakukan, nilai akurasi data training yang diperoleh berubah dari 88.4 %, nilai akurasi data validasi berubah dari 78.6 %, sedangkan nilai loss training sebesar 32.1 %. Nilai data validasi 45.3 %. Seperti pada grafik dibawah ini.

```

import matplotlib.pyplot as plt
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('mode accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'validasi'], loc='upper left')
plt.show()

```

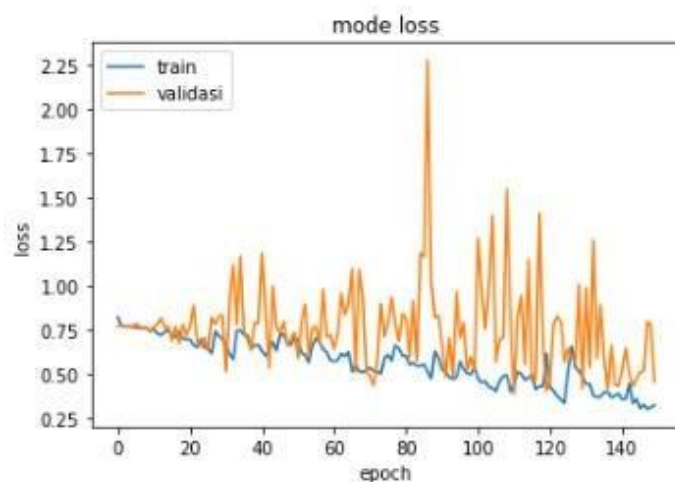
Gambar 5.6 Coding Untuk Grafik Akurasi



Gambar 5.6 Hasil Grafik akurasi training dan validasi

```
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('mode loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'validasi'], loc='upper left')
plt.show()
```

Gambar 5.7 Coding Untuk Grafik Loss



Gambar 5.8 Hasil Grafik Loss training dan validasi

- Berdasarkan hasil proses klasifikasi yang dilakukan terhadap data testing yang terbentuk dalam *Confusion Matrix* dengan jumlah 3.200 gambar dengan mengambil 15 sampel jenis *greenbeen coffee* dari setiap kelas dengan data yang sama, dapat diperoleh nilai kinerja model dari algoritma *Convolutional Neural Network* (CNN) yang digunakan yakni nilai *recall* sebesar 92%, nilai *precission* sebesar 86%, nilai *accuracy* sebesar 90% dan nilai f1-Score sebesar 88%. Dengan hasil tersebut maka dapat dikatakan bahwa penggunaan metode *Convolutional Neural Netwrok* dapat bekerja dengan baik, sehingga penggunaan metode tersebut dapat digunakan dalam mengembangkan sebuah aplikasi yang dapat dimanfaatkan dibidang yang membutuhkan.

- Pergerakan nilai loss function mendekati nol dan menghasilkan peningkatan nilai akurasi bersamaan dengan jumlah epoch yang digunakan menunjukkan hasil yang baik. Grafik yang ditunjukkan pada Gambar 5.6 dan Gambar 5.8 menunjukkan korelasi antara fungsi loss dan akurasi nilai data training. Pada nilai akurasi menunjukkan korelasi positif atau hubungan searah. Penggunaan jumlah epoch yang besar dapat menghasilkan nilai akurasi yang tinggi terhadap data training. Tetapi, hal tersebut berbanding terbalik dengan nilai pada loss function, karena nilai loss merupakan korelasi negatif. Semakin tinggi jumlah epoch yang digunakan maka nilai loss terhadap data training dapat menjadi rendah. Untuk memperoleh nilai loss yang rendah, dapat dilakukan dengan cara menambah jumlah epoch yang akan digunakan untuk data training Penelitian ini menggunakan jumlah epoch sebanyak 150.

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan dan pembahasn yanga telah di uraikan maka dapat di tarik suatu kesimpulan bahwa, kinerja model klasifikasi mutu green bean coffee menggunakan metode *Convolutional Neural Network* setelah dihitung menggunakan *Convoution Matriks* dengan jumlah 15 gambar pada tiap kelas menghasilkan nilai akurasi sebesar 90%, nilai precission 86%, nilai recall 92% dan F1-score 88%.

6.2 Saran

Setelah melakukan penelitian untuk klasifikasi mutu *greenbeen coffee* ada beberapa saran yang perlu diperhatikan :

1. Diharapkan pada peneliti selanjutnya untuk melakukan klasifikasi dengan menggunakan *dataset* yang lebih banyak lagi , agar proses pembelajaran dari metode tersebut semakin baik.
2. Diharapkan juga penelitian selanjutnya dapat mengklasifikasikan gambar menggunakan hardware yang mendukung untuk mempercepat proses training.

DAFTAR PUSTAKA

- [1] M. A. Rizal, “Klasifikasi Mutu Biji Kopi Menggunakan Metode K – Nearest Neighbor Berdasarkan Warna Dan Tekstur,” *Univ. Teknol. Yogyakarta*, vol. 1, no. 1, pp. 1–8, 2019.
- [2] P. Thanthirige *et al.*, “No Title,” no. August, pp. 1–63, 2016.
- [3] M. K. Neighbor, B. Bentuk, D. Ikhsan, E. Utami, and F. W. Wibowo, “Metode Klasifikasi Mutu Greenbean Kopi Arabika Lanang Dan Biasa,” no. 2, pp. 1–8, 2020.
- [4] T. Y. Prahudaya and A. Harjoko, “Metode klasifikasi mutu jambu biji menggunakan knn berdasarkan fitur warna dan tekstur,” vol. 6, 2017, doi: 10.22146/teknosains.26972.
- [5] 2 Rodiah 1 Fitrianiingsih, “KLASIFIKASI JENIS CITRA DAUN MANGGA MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK,” 2020, https://www.researchgate.net/publication/350431243_KLASIFIKASI_JENIS_CITRA_DAUN_MANGGA_MENGGUNAKAN_CONVOLUTIONAL_NEURAL_NETWORK
- [6] L. Vinet and A. Zhedanov, “A ‘missing’ family of classical orthogonal polynomials,” *J. Phys. A Math. Theor.*, vol. 44, no. 8, pp. 1–183, 2011, doi: 10.1088/1751-8113/44/8/085201.
- [7] R. Abd and K. S. Nur, “Klasifikasi Jenis Tumbuhan Obat Herbal Menggunakan Convolutional Neural Network,” vol. x, no. X.
- [8] R. Abd and K. S. Nur, *KLASIFIKASI JENIS TUMBUHAN OBAT HERBAL MENGGUNAKAN CONVOLUTIONAL NEURAL Oleh PROGRAM SARJANA Guna memperoleh gelar Sarjana . Program Studi Teknik Informatika ini telah disetujui oleh Tim Pembimbing. 2022.*
- [9] Fitrianiingsih and Rodiah, “Klasifikasi Jenis Citra Daun Mangga Menggunakan Convolutional Neural Network,” *J. Ilm. Teknol. dan Rekayasa*, vol. 25, no. 3, pp. 223–238, 2020, doi: 10.35760/tr.2020.v25i3.3519.
- [10] Liana Sudjarwadi, “Pengolahan Citra Digital,” pp. 5–47, 2017.

Lampiran 1 : Kode Program

```
from google.colab import drive
drive.mount('/content/drive')
def generate_image(image_source_path):
    im_list = []
    labels = []
    for path in paths.list_images(image_source_path):
        img = cv2.imread(path)
        #img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        img = cv2.resize(img, (48, 48))
        im_list.append(img)
        labels.append(os.path.split(os.path.dirname(path))[-1])
    df_label = pd.DataFrame(labels)
    return im_list, df_label
from imutils import paths
import os
import cv2
import pandas as pd
x_train, y_train =
generate_image('/content/drive/MyDrive/dataset2/newdatakopi/train
')
x_test, y_test =
generate_image('/content/drive/MyDrive/dataset2/newdatakopi/test'
)
import pickle

with open('x_train.pkl', 'wb') as file_train:
    pickle.dump(x_train, file_train)

with open('x_test.pkl', 'wb') as file_test:
    pickle.dump(x_test, file_test)

with open('y_train.pkl', 'wb') as file_ytrain:
    pickle.dump(y_train, file_ytrain)

with open('y_test.pkl', 'wb') as file_ytest:
    pickle.dump(y_test, file_ytest)
import pickle
x_train = pickle.load(open('x_train.pkl', 'rb'))
x_test = pickle.load(open('x_test.pkl', 'rb'))
y_train = pickle.load(open('y_train.pkl', 'rb'))
y_test = pickle.load(open('y_test.pkl', 'rb'))
y_train.columns = ['y']
y_test.columns = ['y']
```



```

y_train = y_train['y'].values.ravel()
y_test = y_test['y'].values.ravel()
from sklearn.preprocessing import LabelEncoder
encoder = LabelEncoder()
encoder.fit(y_train)
y_train = encoder.transform(y_train)

y_test = encoder.transform(y_test)
for i, item in enumerate(encoder.classes_):
    print(item, '>', i)
from keras import backend as K
import numpy as np

x_train = np.array(x_train, dtype='uint8')
x_test = np.array(x_test, dtype='uint8')

y_train = np.array(y_train, dtype='uint8')
y_test = np.array(y_test, dtype='uint8')

x_train = x_train.reshape(x_train.shape[0], 48, 48, 3)
x_test = x_test.reshape(x_test.shape[0], 48, 48, 3)
from keras.preprocessing.image import ImageDataGenerator
datagen = ImageDataGenerator(
    rescale=1./255,
    rotation_range = 10,
    horizontal_flip = True,
    width_shift_range=0.1,
    height_shift_range=0.1,
    fill_mode = 'nearest')

testgen = ImageDataGenerator(rescale=1./255)
datagen.fit(x_train)
batch_sizes = 7

train_flow = datagen.flow(x_train, y_train,
batch_size=batch_sizes)
test_flow = testgen.flow(x_test, y_test, batch_size=batch_sizes)
from keras.models import Sequential
from keras.layers import Conv2D, MaxPool2D, Flatten, Dense,
BatchNormalization, Dropout
from tensorflow.keras.optimizers import SGD, Adam
from keras import regularizers
from keras.regularizers import l1, l2
from matplotlib import pyplot as plt
import tensorflow as tf

```

```
model = Sequential()

#blok-A
model.add(Conv2D(64, kernel_size=(3,3), activation='relu',
padding='same', input_shape=(48,48,3)))
model.add(BatchNormalization())
model.add(Conv2D(64, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(MaxPool2D(pool_size=(2,2)))
model.add(Dropout(0.3))

#blok-B
model.add(Conv2D(128, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(128, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(128, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(MaxPool2D(pool_size=(2,2)))
model.add(Dropout(0.3))

#blok-C
model.add(Conv2D(256, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(256, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(256, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(256, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(MaxPool2D(pool_size=(2,2)))
model.add(Dropout(0.3))

#blok-D
model.add(Conv2D(256, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
```

```

model.add(Conv2D(256, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(256, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(256, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(MaxPool2D(pool_size=(2,2)))
model.add(Dropout(0.3))

#blok-E
model.add(Conv2D(512, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(512, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(512, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(512, kernel_size=(3,3), activation='relu',
padding='same'))
model.add(BatchNormalization())
model.add(MaxPool2D(pool_size=(2,2)))
model.add(Dropout(0.3))

model.add(Flatten())
#model.add(Dense(64, activation='relu',
kernel_regularizer=regularizers.l2(0.0001)))
model.add(Dropout(0.25))
model.add(Dense(64, activation='relu',
kernel_regularizer=regularizers.l2(0.0001)))
model.add(Dropout(0.25))
model.add(Dense(512, activation='relu',
kernel_regularizer=regularizers.l2(0.0001)))
model.add(Dropout(0.25))
model.add(Dense(512, activation='relu',
kernel_regularizer=regularizers.l2(0.0001)))
model.add(Dropout(0.25))
model.add(Dense(1, activation='sigmoid'))

sgd_opt = SGD(learning_rate=0.1)

```

```
model.compile(optimizer=sgd_opt, loss='binary_crossentropy',
metrics=['accuracy'])

model.summary()
history = model.fit(train_flow,
                    steps_per_epoch=len(x_train)/batch_sizes,
                    epochs =150,
                    verbose=1,
                    validation_data = test_flow)
import matplotlib.pyplot as plt
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('mode accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'validasi'], loc='upper left')
plt.show()
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('mode loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'validasi'], loc='upper left')
plt.show()
```

Lampiran 2 : Daftar Riwayat Hidup



Nama : Azhar Muhammad
Nim : T3118209
Tempat, Tanggal Lahir : Tompaso Baru, 19 Februari 2001
Agama : Islam
Email : azharkining@gmail.com

Riwayat Pendidikan :

1. Tahun 2012, Menyelesaikan Pendidikan di Sekolah Dasar Negeri 1 Tompaso baru, Kecamatan Tompaso Baru, Kabupaten Minahasa Selatan, Provinsi Sulawesi Utara.
2. Tahun 2015, Menyelesaikan Pendidikan di Madrasah Tsanawiyah Assalam Manado, Kecamatan Bunaken, Kota Manado, Provinsi Sulawesi Utara.
3. Tahun 2018, Menyelesaikan Pendidikan di Man Model 1 Manado, Kec. Tuminting Kota Manado, Provinsi Sulawesi Utara.
4. Tahun 2018, Telah di terima menjadi Mahasiswa di Perguruan Tinggi Swasta Universitas Ichsan Gorontalo.

Lampiran 3 : Hasil Turniti



Similarity Report ID: oid:25211:18733436

PAPER NAME

PROPOSAL_T3118209_AZHAR MUHAM
MAD.docx

AUTHOR

T3118209_AZHAR MUHAMMAD azharki
ning@gmail.com

WORD COUNT

2935 Words

CHARACTER COUNT

17284 Characters

PAGE COUNT

21 Pages

FILE SIZE

230.8KB

SUBMISSION DATE

Jun 13, 2022 3:04 PM GMT+8

REPORT DATE

Jun 13, 2022 3:05 PM GMT+8

● 30% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 30% Internet database
- 12% Publications database
- Crossref database
- Crossref Posted Content database
- 0% Submitted Works database

● Excluded from Similarity Report

- Bibliographic material
- Small Matches (Less than 25 words)

Lampiran 4 : Surat Keterangan Bebas Pustaka

 KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER
UPT. PERPUSTAKAAN FAKULTAS
SK. MENDIKNAS RI NO. 64/D/0/2001
Jl. Achmad Nadjamuddin No.17 Telp.(0435) 829975 Fax. (0435) 829976 Gorontalo

SURAT KETERANGAN BEBAS PUSTAKA
No : 001/Perpustakaan-Fikom/II/2023

Perpustakaan Fakultas Ilmu Komputer (FIKOM) Universitas Ichsan Gorontalo dengan ini menerangkan bahwa :

Nama Anggota : Azhar Muhammad
No. Induk : T3118209
No. Anggota : M20235

Terhitung mulai hari, tanggal : Sabtu, 11 Februari 2023, dinyatakan telah bebas pinjam buku dan koleksi perpustakaan lainnya.

Demikian keterangan ini di buat untuk di gunakan sebagaimana mestinya.



Gorontalo, 11 Februari 2023
Mengetahui,
Kepala Perpustakaan

Apriyanto Alhamad, M.Kom
NIDN : 0924048601

Lampiran 5 : Surat Penelitian

 **KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI**
UNIVERSITAS IHSAN GORONTALO
FAKULTAS ILMU KOMPUTER
SURAT KEPUTUSAN MENDIKNAS RI NOMOR 54/D/O/2001
Jl. Achmad Nadjamuddin No. 17 Telp (0435) 829975 Fax (0435) 829976 Gorontalo

SURAT KETERANGAN PENELITIAN
Nomor : 06/FIKOM-L/KANKPI/2023

Yang bertanda tangan di bawah ini

Nama	: Irvan Abraham Salibi, M.Kom
NIDN	: 0928028101
Jabatan	: Dekan Fakultas Ilmu Komputer

Dengan ini Menegaskan bahwa:

Nama Mahasiswa	: Azhar Muhammad
NIM	: 13118209
Program Studi	: Teknik Informatika

Bahwa yang bersangkutan benar-benar telah melakukan penelitian tentang "Klasifikasi Muka Green Bean Coffee Menggunakan Convolutional Neural Network" guna untuk menyelesaikan Studi pada Program Studi Teknik Informatika Fakultas Ilmu Komputer, dan bersangkutan telah menyelesaikan penelitian tersebut pada TGL 09 Januari 2023 sesuai dengan waktu yang telah di tentukan.

Demikian Surat Keterangan ini dibuat dan digunakan untuk sepeertanya.


Gorontalo, 09 Januari 2023
Irvan Abraham Salibi, M.Kom
NIDN 0928028101

Lampiran 6 : Surat Rekomendasi Penelitian

 **KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI**
UNIVERSITAS IHSAN GORONTALO
FAKULTAS ILMU KOMPUTER
SURAT KEPUTUSAN MENDIKNAS RI NOMOR 84/D/O/2001
Jl. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SURAT KETERANGAN PENELITIAN
Nomor : DS/FIKOM-UNIKOMPI/2023

Yang bertanda tangan di bawah ini

Nama	: Irvan Abraham Salibi, M.Kom
NIDN	: 0928028101
Jabatan	: Dekan Fakultas Ilmu Komputer

Dengan ini Menertangkan bahwa:


Nama Mahasiswa	: Azhar Muhammad
NIM	: T3118299
Program Studi	: Teknik Informatika

Bahwa yang bersangkutan benar-benar telah melakukan penelitian tentang "Klasifikasi Muts Green Bean Coffee Menggunakan Convolutional Neural Network" Guru untuk menyelesaikan Studi pada Program Studi Teknik Informatika Fakultas Ilmu Komputer, dan bersangkutan telah menyelesaikan penelitian tersebut pada TGL 09 Januari 2023 sesuai dengan waktu yang telah di tentukan.

Demiikian Surat Keterangan ini dibuat dan digunakan untuk sepetarnya.


Gorontalo, 09 Januari 2023
Irvan Abraham Salibi, M.Kom
NIDN 0928028101

Lampiran 7 : Lembar Revisi



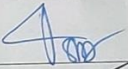


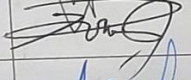
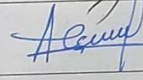
KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER
SK MENDIKNAS NOMOR 84/D/O/2001
JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

Berita Acara Perbaikan/Revisi Ujian SKRIPSI

Pada hari ini, Senin 3 April 2023, Pukul 12.30-14.30 Wita. Telah dilaksanakan Ujian SKRIPSI mahasiswa Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.

Nama : Azhar Muhamad
Nim : T3118209
Pembimbing I : Irvan Abraham Salihi, M.Kom
Pembimbing II : Apriyanto Alhamad, M.Kom
Judul SKRIPSI : Klasifikasi Mutu GreenBean Coffee Menggunakan Metode Convolutional Neural Network

Oleh Komite Seminar sebagai berikut :

No	Komite Seminar	Status	Tanda Tangan
1	Asmaul Husna N, M.Kom	Ketua	
2	Sudirman S. Panna, M.Kom	Anggota	
3	Zulfrianto Y. Lamasigi, M.Kom	Anggota	
4	Irvan Abraham Salihi, M.Kom	Anggota	
5	Apriyanto Alhamad, M.Kom	Anggota	

Program Studi Teknik Informatika Fakultas Ilmu Komputer Unisan Gorontalo