

**KLASIFIKASI KUALITAS BIJI JAGUNG MENGGUNAKAN
METODE FASTER REGION CONVOLUTIONAL
NEURAL NETWORK**

SKRIPSI

TIMOTHY AGALLIASIS

191402084



**PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024**

**KLASIFIKASI KUALITAS BIJI JAGUNG MENGGUNAKAN METODE
FASTER REGION CONVOLUTIONAL
NEURAL NETWORK**

SKRIPSI

**TIMOTHY AGALLIASIS
191402084**



**PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024**

PERSETUJUAN

Judul : KLASIFIKASI KUALITAS BIJI JAGUNG
MENGUNAKAN METODE FASTER
REGION CONVOLUTIONAL NEURAL
NETWORK

Kategori : SKRIPSI

Nama : TIMOTHY AGALLIASIS

Nomor Induk Mahasiswa : 191402084

Program Studi : SARJANA (S1) TEKNOLOGI INFORMASI

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

Medan, 14 Juni 2024

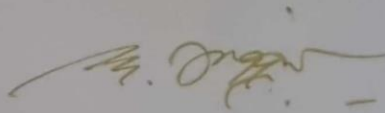
Komisi Pembimbing :

Pembimbing 2

Pembimbing 1



Umayu Ramadhani Putri Nasution
S.TI., M.Kom.
199104112021022001



Dr. Muhammad Anggia Muchtar S.T.,
MM.IT.
198001102008011010

Diketahui/disetujui oleh

Program Studi S1 Teknologi Informasi

Ketua,



Timothy Agalliasis S.T., M.Kom.
NIP. 197908312009121002

PERNYATAAN

**KLASIFIKASI KUALITAS BIJI JAGUNG MENGGUNAKAN
METODE FASTER REGION CONVOLUTIONAL NEURAL
NETWORK**

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, 14 Juni 2024

A handwritten signature in black ink, appearing to be 'A. H. ... c'.

NIM. 191402084

UCAPAN TERIMAKASIH

Puji Syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena atas kasih dan penyertaannya penulis bisa menyelesaikan skripsi sebagai syarat untuk menyelesaikan perkuliaha dan memperoleh gelar Sarjana Komputer pada program studi S1 Teknologi Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.

Penulis juga dapat menyelesaikan skripsi ini karena bantuan, dukungan, serta doa yang telah diberikan kepada penulis dari semua pihak. Penulis ingin menyampaikan terima kasih kepada:

1. Tuhan Yesus Kristus yang selalu memberikan kemudahan, kekuatan serta berkat yang melimpah sehingga penulis dapat menyelesaikan skripsi ini dengan baik.
2. Orang tua penulis. Bapak Hepata KLS Ginting dan Mamak Rismawati Br Sitepu yang selalu mendukung, memberi semangat, dan mendokan penulis. Begitu juga dengan kedua saudara penulis Yeshika Regina Br Ginting dan Hineni Fortuna Br Ginting yang selalu memberi dukungan, semangat, dan doa kepada penulis. Dan juga abang Andika Keliat selaku abang ipar dari penulis yang mendukung penulis dalam penyelesaian skripsi.
3. Ibu Dr. Maya Silvi Lydia, M.Sc selaku Dekan Fasilkom-TI Universitas Sumatera Utara.
4. Bapak Dedy Arisandi, S.Kom., M.Kom. Selaku Ketua program studi Teknologi Informasi Universitas Sumatera Utara.
5. Bapak Dr. Muhammad Anggia Muchtar S.T., MM.IT. dan Ibu Umayya Ramadhani Putri Nasution S.TI., M.Kom. selaku dosen pembimbing 1 dan 2 penulis yang telah bersedia menjadi membimbing penulis serta senantiasa memberikan kritik, saran, dan motivasi kepada penulis.
6. Seluruh Dosen, Staff dan Pegawai Program Studi S1 Teknologi Informasi yang telah memberikan manfaat bagi penulis serta membantu proses perkuliahan penulis.
7. Sheren Alvionita Siahaan dan Grace Ogestin Pasaribu yang telah memberikan waktu untuk berdiskusi dalam proses pengerjaan skripsi.

8. Senior penulis, Novalina Gabriella yang selalu memberikan masukan, dukungan, semangat dan mengingatkan penulis dalam proses pengerjaan skripsi.
9. Teman-teman angkatan 2019 selaku teman seperjuangan penulis dalam menyelesaikan perkuliahan.
10. Kepada senior dan junior lainnya yang namanya tidak dapat disebutkan satu persatu yang telah memberikan semangat dan saran dalam masa kuliah dan dalam menyelesaikan penulisan tugas akhir.
11. Kepada semua teman-teman penulis.

Penulis menyadari bahwa masih terdapat kekurangan dalam penulisan skripsi ini, oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan untuk penyempurnaan skripsi ini.

Medan, 14 Juni 2024

Penulis,

A handwritten signature in black ink, appearing to read 'Timothy Agalliasis' with a stylized flourish at the end.

Timothy Agalliasis

191402084

ABSTRAK

Jagung merupakan komoditas pangan kedua setelah padi. Jagung juga memiliki kualitas yang bervariasi. Penentuan kualitas biji jagung menjadi peran penting dalam menentukan harga jual biji jagung, dalam penentuan kualitas biji jagung masih menggunakan tenaga manual yang menyebabkan terjadinya subjectivitas dalam penilaian, sehingga penilaian kualitas biji jagung tidak sepenuhnya berdasarkan objek dari biji jagung tersebut dan akan menyebabkan terjadinya perbedaan pendapat antara pengamat dengan pengamat yang lain. Kualitas biji jagung dapat dilihat secara fisik biji jagung berdasarkan tekstur biji jagung yang dilihat dari kekerutan germ dan warna biji jagung. Penelitian ini menghasilkan sebuah sistem yang dapat mengklasifikasikan kualitas bijijagung dengan melihat warna, tekstur dan bentuk kekerutan pada *germ* biji jagung untuk meminimalisir terjadinya subjectivitas dalam penentuan kualitas biji jagung. Kualitas jagung tersebut kemudian di bagi atas lima tingkatan yaitu Kualitas A, Kualitas B, Kualitas C, Kualitas D dan Kualitas E. Pada penelitian ini menggunakan algoritma *Faster Region Convolutional Neural Network* dan menggunakan ResNet50 sebagai ekstraksi fitur. Adapun data yang digunakan pada penelitian ini berjumlah 2500 data yang kemudian dibagi menjadi 2000 data *training*, 250 data validasi serta 250 data *testing*. Setelah dilakukan pengujian, penelitian ini menghasilkan akurasi sebesar 95,2%.

Kata Kunci : Jagung, Kualitas Jagung, Citra Digital, Faster RCNN

*Classification of Corn Quality Using Faster Region Convolutional Neural Network***ABSTRACT**

Maize is the second food commodity after rice. Corn also has various qualities. Determining the quality of corn kernels plays an important role in determining the selling price of corn kernels, in determining the quality of corn kernels still uses manual labor which causes subjectivity in the assessment, so that the assessment of the quality of corn kernels is not fully based on the object of the corn kernels and will cause differences in opinion between observers and other observers. The quality of corn kernels can be seen physically based on the texture of the corn kernels, which is seen from the crunchiness of the germ and the color of the corn kernels. This research produces a system that can classify the quality of corn kernels by looking at the color, texture and shape of the germ of corn kernels to minimize subjectivity in determining the quality of corn kernels. The quality of corn is then divided into five levels, namely Quality A, Quality B, Quality C, Quality D and Quality E. This research uses the Faster Region Convolutional Neural Network algorithm and uses ResNet50 as feature extraction. The data used in this study amounted to 2500 data which were then divided into 2000 training data, 250 validation data and 250 testing data. After testing, this research resulted in an accuracy of 95.2%.

Keywords: Maize, Maize Quality, Digital Image, Faster RCNN

DAFTAR ISI

UCAPAN TERIMAKASIH	III
ABSTRAK	VII
ABSTRACT	VIII
DAFTAR ISI	IX
DAFTAR TABEL	XII
DAFTAR GAMBAR	XIII
BAB I PENDAHULUAN	15
1.1 Latar Belakang	15
1.2 Rumusan Masalah	17
1.3 Batasan Penelitian	18
1.4 Tujuan Penelitian	18
1.5 Manfaat Penelitian	18
1.6 Metodologi Penelitian	18
1.7 Sistematika Penulisan	19
BAB II LANDASAN TEORI	21
2.1 Jagung	21
2.2 Kualitas Biji Jagung	21
2.3 Pengolahan Citra Digital (Digital Image Processing)	23
2.4 Citra Digital	23
2.1.1 <i>Citra RGB</i>	24
2.1.2 <i>Citra Grayscale</i>	24
2.1.3 <i>Citra Biner</i>	25
2.5 Augmentasi	25
2.6 <i>Residual Network (ResNet)</i>	26
2.7 <i>Convolutional Neural Network (CNN)</i>	26
2.8 <i>Region Convolutional Neural Network (R-CNN)</i>	28
2.9 <i>Fast Region Convolutional Neural Network (Fast R-CNN)</i>	28

2.10 <i>Faster Region Convolutional Neural Network (Faster R-CNN)</i>	29
2.11 <i>Android Studio</i>	33
2.12 <i>Python</i>	33
2.13 <i>Tensorflow Lite</i>	34
2.14 <i>Confussion Matrix</i>	34
2.15 Penelitian Terdahulu	35
2.16 Perbedaan Penelitian	37
BAB III ANALISIS DAN PERANCANGAN SISTEM	39
3.1 Data yang Digunakan	39
3.2 Analisis Sistem	41
3.2.1 <i>Image Acquisition</i>	44
3.2.2 <i>Image Pre-Processing</i>	44
3.2.2.1 <i>Labelling</i>	44
3.2.2.2 <i>Resizing</i>	45
3.2.2.3 <i>Augmentasi</i>	46
3.2.3 <i>Feature Extraction</i>	48
3.2.4 <i>Image Classifier</i>	51
3.2.4.1 <i>Feature Map</i>	52
3.2.4.2 <i>Region Proposal Network (RPN)</i>	52
3.2.4.3 <i>Region of Interest Pooling</i>	53
3.2.5 <i>Learned Model</i>	55
3.2.6 <i>TFLite Model</i>	56
3.2.7 <i>Training Data</i>	56
3.2.8 <i>Output</i>	56
3.3 PERANCANGAN ANTARMUKA SISTEM	56
3.3.1 Rancangan Splashscreen	57
3.3.2 Rancangan Tampilan Halaman Utama	57
3.3.3 Rancangan Tampilan Halaman Cek Kualitas Jagung	57
3.3.4 Rancangan Tampilan Hasil Klasifikasi	58
3.3.5 Rancangan Tampilan Tutorial	58
3.3.6 Rancangan Tampilan Informasi Kualitas Jagung	59
BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM	60

4.1 Implementasi Sistem	60
4.1.1 Perangkat Keras dan Perangkat Lunak	60
4.1.2 Implementasi Data	60
4.1.3 Implementasi Aplikasi	63
4.1.3.1 Tampilan Halaman <i>Splash Screen</i>	63
4.1.3.2 Tampilan Halaman Utama	64
4.1.3.3 Tampilan Halaman <i>Testing</i>	65
4.1.3.4 Tampilan Halaman Hasil	66
4.1.3.5 Tampilan Halaman Tutorial Aplikasi	67
4.1.3.6 Tampilan Halaman Informasi Kualitas Biji Jagung	68
4.2 Prosedur Operasional	69
4.3 Pelatihan Sistem	69
4.4 Pengujian Sistem	71
BAB V KESIMPULAN DAN SARAN	80
5.1 Kesimpulan	80
5.2 Saran	80
DAFTAR PUSTAKA	82

DAFTAR TABEL

Tabel 2. 1 Definisi fisik biji jagung menurut SNI 4483-2013	22
Tabel 2. 2 Syarat Mutu sebagai Acuan Harga Pembelian Jagung Tingkat Petani	22
Tabel 2. 3 Penelitian Terdahulu	36
Tabel 3. 1 Banyak Data	41
Tabel 3. 2 Kategori Jagung Berdasarkan Klasifikasi	44
Tabel 4. 3 Hasil Evaluasi Model	75
Tabel 4. 4 TP, FP, FN	76
Tabel 4. 5 Nilai Precision, Recall, dan F1-Score	78

DAFTAR GAMBAR

Gambar 2. 1 Citra RGB	24
Gambar 2. 2 Citra Grayscale	25
Gambar 2. 3 Citra Biner	25
Gambar 2. 4 ResNet (He et al, 2015)	26
Gambar 2. 5 Arsitektur Umum Convolutional Neural Network (Sumit Saha, 2018)	27
Gambar 2. 6 Arsitektur R-CNN	28
Gambar 2. 7 Arsitektur Fast R-CNN	29
Gambar 2. 8 Arsitektur Umum Faster R-CNN (Nguyen et al., 2018)	31
Gambar 3. 1 Pengukuran Kadar Air Biji Jagung Menggunakan Grain Moisture Meter	39
Gambar 3. 2 Citra data yang digunakan	40
Gambar 3. 3 Arsitektur Umum	43
Gambar 3. 4 Labelling	45
Gambar 3. 5 Proses Perhitungan Resizing	45
Gambar 3. 6 Citra Jagung sebelum (kiri) dan sesudah di-resize (kanan)	46
Gambar 3. 7 Pseudocode Proses Resizing	46
Gambar 3. 8 Sebelum dan Sesudah Rotasi 90 Derajat	47
Gambar 3. 9 Sebelum dan Sesudah Flip Kiri Kanan	47
Gambar 3. 10 Sebelum dan Sesudah Flip Atas Bawah	48
Gambar 3. 11 Pseudocode Proses Augmentasi	48
Gambar 3. 12 Matriks Input Gambar	49
Gambar 3. 13 Perhitungan Filter Matriks	49
Gambar 3. 14 Pergeseran Matriks	50
Gambar 3. 15 Pseudocode Proses Klasifikasi	51
Gambar 3. 16 Feature Map	52
Gambar 3. 17 Target <i>Region Proposal Network</i>	53
Gambar 3. 18 Target <i>Region of Interest Pooling</i>	54
Gambar 3. 19 Proses ROI Pooling	54
Gambar 3. 20 Proses Penggabungan ROI Pooling	55
Gambar 3. 21 Hasil Penggabungan RoI Pooling	55

Gambar 3. 22	Rancangan Tampilan Halaman Utama	57
Gambar 3. 23	Rancangan Tampilan Cek Kualitas Jagung	58
Gambar 3. 24	Rancangan Tampilan Hasil Klasifikasi	58
Gambar 3. 25	Rancangan Tampilan Tutorial	59
Gambar 3. 26	Rancangan Tampilan Kualitas Biji Jagung	59
Gambar 4. 1	Data Jagung Kualitas A	61
Gambar 4. 2	Data Jagung Kualitas B	61
Gambar 4. 3	Data Jagung Kualitas C	62
Gambar 4. 4	Data Jagung Kualitas D	62
Gambar 4. 5	Data Jagung Kualitas E	63
Gambar 4. 6	Tampilan Halaman Splash Screen	64
Gambar 4. 7	Tampilan Halaman Utama	65
Gambar 4. 8	Tampilan Halaman Testing	66
Gambar 4. 9	Tampilan Halaman Hasil	67
Gambar 4. 10	Tampilan Halaman Tutorial Aplikasi	68
Gambar 4. 11	Tampilan Halaman Informasi Kualitas Jagung	69

BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan negara agraris karena mengandalkan hidup dari sektor pertanian. Salah satu hasil dari pertanian yang memengaruhi perkembangan sektor pertanian di Indonesia ialah jagung. Jagung merupakan makanan kedua setelah padi dan merupakan sumber kalori atau pengganti beras serta digunakan sebagai pakan ternak. Perkembangan tanaman jagung akan meningkat dari tahun ke tahun dan berbanding lurus dengan kemajuan industri pakan ternak sehingga kualitas jagung harus lebih diperhatikan. Berdasarkan Data Badan Litbang Pertanian (2015) menyebutkan bahwa pemanfaatan jagung tertinggi digunakan sebagai pakan ternak yaitu 22% pakan ternak langsung dan 44% bahan baku industri pakan, 25% bahan baku industri pangan dan 9% konsumsi rumah tangga secara langsung. Dalam bidang industri pertanian kualitas biji jagung menjadi aspek penting karena faktor dari kualitas biji jagung yang kurang baik akan berpengaruh besar terhadap produk akhir dari industri pakan.

Kualitas jagung yang rendah di sektor pertanian disebabkan oleh tingginya kadar air jagung dan berbagai faktor lain. Industri pakan ternak tidak dapat mengambil biji jagung langsung dari petani jagung karena terdapat standar yang sudah ditetapkan oleh SNI terkait perbedaan kadar air yang jauh sehingga bidang industri memasok dari pengepul yang menyerap jagung lokal dan sampai ke industri. Kondisi tersebut menyebabkan ketidakstabilan harga jagung di tingkat pertanian karena adanya peran pihak ketiga dalam penentuan harga. Disebabkan kondisi tersebut Kementerian Perdagangan RI mengeluarkan Peraturan Menteri Perdagangan RI No.27/M-DAG/PER/5/2017 mengenai penentuan harga jagung berdasarkan kadar air biji jagung.

Di Tingkat petani, harga yang sudah ditetapkan Menteri Perdagangan tidak sepenuhnya dilakukan, karena pengepul mempertimbangkan dari lamanya proses pengeringan sehingga berdampak pada keputusan harga beli. Pengambilan Keputusan untuk menentukan harga beli jagung ini akan membuat adanya keberpihakan dan merugikan salah satu pihak sehingga bisa menimbulkan konflik karena terdapat pengamatan yang berbeda karena rendahnya objektivitas dan konsistensi dalam penentuan kualitas biji jagung. Selain itu ada juga permasalahan mengenai kualitas biji jagung, yaitu permasalahan kualitas gabah, khususnya jagung, masih menjadi

permasalahan. Kerusakan biji jagung yang serius biasanya disebabkan oleh penanganan yang buruk. Misalnya, saat biji jagung dipilpil dengan mesin, terdapat biji jagung yang tidak baik, seperti biji jagung rusak, patah, kulit beberapa biji jagung terkelupas, sehingga mempercepat pertumbuhan jamur. Selain itu, evaluasi penentuan kualitas biji jagung pada proses penentuan kualitas selama ini dilakukan dengan inspeksi visual yang dilakukan oleh pengamat secara manual. Penentuan kualitas dengan cara tersebut mempunyai beberapa kelemahan yaitu terjadi persepsi yang berbeda antara pengamat dalam penentuan kualitas biji jagung, memerlukan waktu yang lama, kelelahan, keterbatasan visual masing masing pengamat dan menghasilkan keputusan yang tidak konsisten ini disebabkan oleh perbedaan hasil pengamatan dari masing-masing pengamat. Sehingga akan terjadi negosiasi yang bisa merugikan beberapa pihak dan penilaian kualitas biji jagung tidak berdasarkan objektivitas dari jagung tersebut. Dalam penentuan kualitas biji jagung, terdapat juga alat untuk mengukur kualitas biji jagung berdasarkan kadar air biji jagung tersebut dinamakan Grain Moisture Meter. Tetapi alat tersebut memiliki kekurangan pada saat penggunaan yang kurang efisien dan juga harga yang cukup mahal sehingga di tingkat petani masih jarang menggunakan alat tersebut.

Perkembangan teknologi yang semakin pesat dan canggih pada masa sekarang ini telah banyak membantu mempercepat pekerjaan manusia. Hal ini dapat dibuktikan dengan adanya pengolahan citra digital untuk mengelompokkan kualitas biji jagung berdasarkan tekstur dan warna dari biji jagung, menggunakan jaringan syaraf tiruan (Soemantri, Agus Supriatna Abubakar, 2020) dengan penelitian yang berjudul “Sistem Penunjang Keputusan Untuk Menentukan Kelas Mutu Jagung Dengan Menggunakan Teknologi Pengolahan Citra Digital dan Jaringan Syaraf Tiruan”. Penelitian ini memperlihatkan bahwa jaringan syaraf tiruan yang dibuat ternyata memiliki tingkat akurasi sebesar 98%. Selain itu terdapat penelitian lain mengenai klasifikasi kualitas biji jagung berdasarkan deteksi warna dan bentuk oleh (Ulla Delfana Rosiani, Mustika Mentari, Andi Novan Prastya, 2019) dengan judul “Kualitas Biji Jagung Berdasarkan Deteksi Warna dan Bentuk Menggunakan Metode K-Nearest Neighbor”. Penelitian ini menggunakan metode K- Nearest Neighbour (K-NN) dan mendapatkan akurasi tertinggi sebesar 76,67%. Selanjutnya penelitian lain mengenai jagung yang dilakukan oleh (MIFTAHUL JANNAH, 2011) dengan judul penelitian “IDENTIFIKASI MUTU JAGUNG MENGGUNAKAN FITUR WARNA DAN TEKSTUR BERBASIS

PENGOLAHAN CITRA DIGITAL DAN ALGORITMA K-NEAREST NEIGHBOR (K-NN)”.

Penelitian ini mendapatkan akurasi sebesar 90%. Penelitian yang menggunakan algoritma *Faster R-CNN* dengan judul “Klasifikasi Mutu Buah Pala (*Myristica Fragrans* Houtt) Berbasis Pengolahan Citra Menggunakan Metode Deep Learning Arsitektur *Faster R-CNN*” oleh (Muh. Subhan, Hasan Basri, 2019) menghasilkan tingkat akurasi 87%. Dan penelitian yang berjudul “Klasifikasi Pola Kain Tenun Melayu Menggunakan *Faster R-CNN*” oleh (Yoze Rizky, Reny medikawaty Taufiq, Dinia Putri, Harun Mukhtar, 2021) mendapatkan tingkat akurasi 82,14%.

Dengan didasari oleh latar belakang tersebut serta penelitian-penelitian terdahulu, penulis mengajukan sebuah penelitian yang akan menghasilkan sebuah aplikasi mobile dengan metode klasifikasi *Faster Region Convolutional Neural Network (Faster R-CNN)* untuk membantu masyarakat baik penjual maupun pembeli dalam menentukan kualitas Jagung berdasarkan warna dan ciri fisik Jagung.. Penelitian ini diberi judul **KLASIFIKASI KUALITAS BIJI JAGUNG MENGGUNAKAN FASTER REGION CONVOLUTIONAL NEURAL NETWORK (FASTER R-CNN)**.

1.2 Rumusan Masalah

Penilaian kualitas biji jagung merupakan kegiatan yang dilakukan sebelum jagung di jual oleh petani, penilaian biji jagung masih menggunakan pengamatan secara manual dengan penglihatan dari manusia, yaitu dengan melihat beberapa biji jagung dan menentukan kualitasnya berdasarkan tekstur dan warna. Kegiatan ini mempunyai kelemahan pada subjektifitas penilaian kualitas biji jagung antara pengamat satu dengan yang lainnya dan akan dapat merugikan pihak-pihak tertentu selain itu waktu yang akan lebih lama karena perbedaan penilaian antara pengamat. Selain cara manual penilaian kualitas biji jagung juga sudah ada yang menggunakan alat yang Grain Moisture Meter untuk mengukur kualitas biji jagung berdasarkan kadar air yang terkandung di biji jagung, tetapi memiliki kelemahan yaitu ketidakpraktisan dalam penggunaan alat dan harga alat tersebut. Dengan pemanfaatan ilmu citra digital akan membantu dalam menentukan kualitas biji jagung, dibutuhkan sistem yang dapat mengklasifikasi kualitas biji jagung berdasarkan citra biji jagung dengan cara yang lebih praktis dan efisien.

1.3 Batasan Penelitian

Agar penelitian lebih terarah dan untuk menghindari penyimpangan dan meluasnya lingkup permasalahan, maka peneliti membuat beberapa batasan masalah, yaitu:

1. Data yang digunakan berekstensi .JPG/.JPEG.
2. Sampel biji jagung yang digunakan sebanyak segenggam tangan dan diletakkan di atas alas berwarna hitam.
3. Sampel biji jagung yang diteliti berjenis pionner berasal dari Kecamatan Munthe, Kabupaten Karo, Sumatera Utara
4. *Output* sistem yang dihasilkan pada penelitian ini adalah klasifikasi kualitas biji jagung dan tebagi menjadi 5 tingkatan kualitas biji jagung yaitu: Kualitas A, Kualitas B, Kualitas C, Kualitas D, dan Kualitas E.
5. Pengambilan gambar dilakukan dengan kamera Handphone berjarak kurang lebih 10 cm dari objek.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah membantu petani dan pengepul biji jagung dalam menilai atau menentukan kualitas biji jagung dengan sistem yang dapat menklasifikasikan kualitas biji jagung menggunakan metode *Faster Region Convolutional Neural Network (Faster R-CNN)* sehingga tidak terjadi perbedaan pendapat pada saat penilaian kualitas biji jagung.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah:

1. Membantu menentukan atau mengklasifikasikan kualitas biji jagung dengan pengolahan citra.
2. Membantu petani jagung untuk menentukan kualitas biji jagung berdasarkan citra biji jagung sehingga tidak terjadi perbedaan pendapat dalam menentukan kualitas biji jagung.
3. Implementasi metode *Faster Region Convolutional Neural Network* dalam mengklasifikasi jagung berdasarkan citra jagung.

1.6 Metodologi Penelitian

Tahapan-tahapan yang akan dilakukan dalam penelitian ini adalah sebagai berikut:

1. Studi Literatur

Studi literatur merupakan proses gumpulkan data biji jagung dan referensi yang berkaitan dengan penelitian klasifikasi kualitas jagung, *image processing*, dan *Faster Region Convolutional Neural Network* yang didapat dari berbagai jurnal, artikel, buku, skripsi, dan sumber referensi lain yang berkaitan dengan penelitian.

2. Analisis Permasalahan

Pada langkah ini, referensi dan data yang dikumpulkan sebelumnya dianalisis untuk mendapatkan pemahaman tentang metode *Faster Region Convolutional Neural Network*, yang kemudian digunakan dalam studi klasifikasi kualitas jagung.

3. Perancangan Sistem

Berdasarkan analisis tahap sebelumnya, maka dibuatlah rancangan sistem berupa rancangan arsitektur, penentuan *training* dan *testing* data, desain antarmuka sistem.

4. Implementasi

Perencanaan sistem yang telah dilakukan sebelumnya kemudian diimplementasikan sehingga tercipta sistem yang sesuai dengan tujuan penelitian.

5. Pengujian Sistem

Pada tahap ini, sistem yang dibangun diuji untuk mengetahui seberapa akurat metode *Faster Region Convolutional Neural Network* digunakan dalam klasifikasi kualitas jagung.

6. Dokumentasi Penyusunan Laporan

Pada tahap ini dilakukan dokumentasi dan penyusunan laporan yang menjabarkan hasil dari keseluruhan penelitian yang telah dilakukan.

1.7 Sistematika Penulisan

Sistematika yang digunakan pada skripsi ini terdiri dari lima bab dan masing-masing bab disertai dengan penjelasan secara singkat sebagai berikut:

BAB I : PENDAHULUAN

Bab ini berisikan latar belakang penelitian ini, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

BAB II : LANDASAN TEORI

Bab ini berisikan teori-teori yang berhubungan dengan kualitas biji jagung, pengolahan citra digital, ekstraksi warna, dan metode Faster R-CNN yang digunakan pada penelitian ini.

BAB III : ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisikan tentang arsitektur umum dari metode Faster R-CNN serta langkah- langkah pada tahap *pre-processing*, proses *training*, *testing*, dan perancangan antar mukasistem.

BAB IV : IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini berisikan implementasi dari hasil analisis dan perancangan sistem yang telah dibahas pada bab sebelumnya dan menampilkan hasil pengujian terhadap sistem yang telah dibangun.

BAB V: KESIMPULAN DAN SARAN

Bab ini berisikan tentang kesimpulan dari keseluruhan penelitian yang telah dilakukan dan saran yang diajukan untuk pengembangan penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Jagung

Jagung (*Zea mays* L) adalah biji-bijian populer di seluruh dunia yang termasuk dalam keluarga rumput (Graminaceae). Jagung merupakan bahan baku utama pakan burung sekitar 50% dari ransum. Kandungan kimia jagung terdiri dari karbohidrat 61%, air 13,5%, protein kasar 10%, lemak 4%, gula 1,4%, pentosan 6%, serat kasar 2,3%, abu 1,4% dan zat lainnya 0,4% (Rukmana, 1997). Jagung kuning mengandung 6,4 - 11,3 µg/g karotenoid, dimana 22% adalah beta-karoten dan 51% adalah xantofil (Koswara dalam Aini, 2013). Sifat kimia jagung menunjukkan bahwa nilai gizi jagung cukup tinggi untuk digunakan sebagai bahan pangan. Pemanfaatan jagung sebagai bahan pakan dan makanan. Ekstraksi minyak pada biji jagung akan dibuat menjadi tepung sebagai bahan baku industri. Produksi tepung yang disebut pati jagung sebagai bahan baku dan sumber energi alternatif dari tongkol (Respati et al, 2013). Jagung yang digunakan untuk pakan ternak dibagi menjadi dua kelompok yaitu ruminansia dan non ruminansia. Pada umumnya ternak ruminansia memanfaatkan limbah jagung berupa jerami dan jagung muda berusia 60 hari sebagai pakannya. Biji jagung hampir seluruhnya dimanfaatkan untuk sapi non-ruminansia dan sebagian kecil untuk sapi perah. Salah satu keunggulan jagung sebagai bahan pakan unggas khususnya ayam petelur adalah mengandung *xanthophyll* yang berguna untuk mencerahkan warna kuning telur. Zat ini tidak ditemukan pada biji-bijian lain (Tangendjaja et al, 2005).

2.2 Kualitas Biji Jagung

Kualitas jagung secara fisik ditentukan dari warna, bentuk dan tekstur. sedangkan secara kimia ditentukan dari berbagai analisis, seperti kadar air dan kadar aflatoksin. Penurunan kualitas jagung umumnya terjadi karena kadar air yang tinggi, biji jagung rusak, warna biji tidak merata, biji pecah, dan kontaminan lain yang mengakibatkan mutu jagung buruk (BPTP, 2009). Standar mutu yang harus dipenuhi dari jagung sebagai bahan baku pangan diperlukan untuk menjamin produsen dan menjamin mutu

pangan yang akan menggunakan jagung. Definisi kerusakan fisik dari biji jagung bersumber dari BSN pada tahun 2013 dalam kategori fisik biji jagung dapat dilihat pada Tabel 2.1.

Tabel 2. 1 Definisi fisik biji jagung menurut SNI 4483-2013

No.	Fisik Biji Jagung	Defenisi
1.	Biji Utuh	Biji jagung kering yang secara fisik keseluruhannya utuh tanpa ada nya bercak, cacat ataupun jamur.
2.	Biji Rusak	Biji jagung yang pecah yang mengalami kerusakan karena pengaruh mekanis, biologis, fisik dan enzimatis.
3.	Biji Pecah	Biji jagung yang pecah selama proses pengolahan yang memiliki ukuran minimal 0,6 bagian jagung yang utuh.
4.	Biji Berjamur	Biji jagung yang sudah terserang jamur yang warnanya sudah berubah dari warna aslinya.
5.	Biji Asing	Semua benda selain jagung.

Sumber: BSN,2013

Adapun mutu jagung di tingkat petani dikategorikan berdasarkan PERMENDAGRI No.27/M-DAG/PER/5/2017 ditunjukkan pada Tabel 2.2. Kualitas ini didasarkan pada kadar air dan dijadikan tolak ukur untuk menentukan harga beli jagung di tingkat petani jagung.

Tabel 2. 2 Syarat Kualitas sebagai Acuan Harga Pembelian Jagung Tingkat Petani

No	Kualitas	Kriteria
1	Kualitas A	Kadar air 15%, Aflatoksin 100 ppb, biji rusak, biji pecah 2%, biji berjamur 2%, benda asing 2%
2	Kualitas B	Kadar air 20%, Aflatoksin 100 ppb, biji rusak, biji pecah 2%, biji berjamur 2%, benda asing 2%

No	Kualitas	Kriteria
3	Kualitas C	Kadar air 25%, Aflatoksin 100 ppb, biji rusak, biji pecah 2%, biji berjamur 2%, benda asing 2%
4	Kualitas D	Kadar air 30%, Aflatoksin 100 ppb, biji rusak, biji pecah 2%, biji berjamur 2%, benda asing 2%
5	Kualitas E	Kadar air 35%, Aflatoksin 100 ppb, biji rusak, biji pecah 2%, biji berjamur 2%, benda asing 2%

Pada Penelitian ini, Kualitas biji jagung yang diteliti dibagi menjadi 5, yakni Kualitas A, Kualitas B, Kualitas C, Kualitas D, dan Kualitas E. Kualitas-kualitas biji jagung di bagi berdasarkan ciri fisik dari warna, bentuk, dan tekstur yang terkandung didalam biji jagung.

2.3 Pengolahan Citra Digital (Digital Image Processing)

Pengolahan citra digital merupakan suatu bentuk pengolahan data yang mengambil masukan berupa gambar dan keluaran berupa gambar atau bagian dari gambar. Tujuan dari pengolahan ini adalah untuk meningkatkan kualitas citra agar dapat dengan mudah diinterpretasikan oleh manusia atau komputer. Pengolahan citra adalah metode pengolahan atau manipulasi citra digital untuk menghasilkan citra baru (Gonzales et al., 2002).

2.4 Citra Digital

Citra digital adalah representasi gambar, kemiripan, atau tiruan suatu objek. Gambar dibagi menjadi dua bagian, yaitu gambar analog dan gambar digital. Gambar analog adalah gambar yang berkesinambungan, seperti gambar dari layar televisi, foto *rontgen*, hasil CT scan, dan lain-lain.

Citra digital saat ini merupakan proses pengolahan citra yang dilakukan pada komputer (Sutoyo et al, 2009). Suatu citra digital dapat direpresentasikan dengan sebuah matriks yang terdiri dari M kolom dan N baris dan perpotongan antara kolom dan baris tersebut adalah piksel (*pixel = picture element*), yang mewakili elemen terkecil dari citra tersebut. Piksel memiliki koordinat dan intensitas atau warna yang menjadi parameter. Nilai pada koordinat (x,y) adalah $f(x,y)$ yang merupakan intensitas atau warna piksel pada titik tersebut. Berdasarkan kombinasi warna pada piksel,

umumnya citra *digital* dibagi menjadi tiga yaitu citra RGB, citra grayscale, dan citra biner.

2.4.1 Citra RGB

Citra berwarna, atau biasa dinamakan citra RGB, merupakan jenis citra yang menyajikan warna dalam bentuk komponen R (merah), G (hijau), dan B (biru). Setiap komponen warna menggunakan 8 bit (nilainya berkisar antara 0 sampai dengan 255). Dengan demikian, kemungkinan warna yang bisa disajikan mencapai $255 \times 255 \times 255$ atau 16.581.375 warna.



Gambar 2. 1 Citra RGB (Hestiningsih, Idhawati, 2008)

2.4.2 Citra Grayscale

Citra *grayscale* adalah suatu citra yang hanya memiliki warna tingkat keabuan. Warna abu-abu pada citra *grayscale* adalah warna R (*Red*), G (*Green*), B (*Blue*) yang memiliki intensitas yang sama. Sehingga dalam *grayscale image* hanya membutuhkan nilai intensitas tunggal dibandingkan dengan citra berwarna membutuhkan tiga intensitas untuk tiap pikselnya. Intensitas dari citra grayscale disimpan dalam 8 bit integer yang memberikan 256 kemungkinan yang mana dimulai dari level 0 sampai dengan 255 yang berarti 0 untuk hitam dan 255 untuk putih dan nilai di antaranya adalah derajat keabuan.



Gambar 2. 2 Citra Grayscale (Hestinationsih, Idhawati, 2008)

2.4.3 Citra Biner

Citra biner (monokromatik) merupakan citra yang sudah melalui proses pemisahan piksel berdasarkan skala keabuannya. Citra biner merupakan citra yang hanya mewakili nilai setiap piksel dalam satu bit (satu nilai biner). Citra biner mempunyai dua warna yaitu hitam dan putih. Diperlukan sepotong memori untuk menyimpan dua warna ini. Setiap piksel pada citra mempunyai nilai 0 untuk warna hitam dan 1 untuk warna putih (Hestinationsih, Idhawati, 2008).



Gambar 2. 3 Citra Biner (Hestinationsih, Idhawati, 2008)

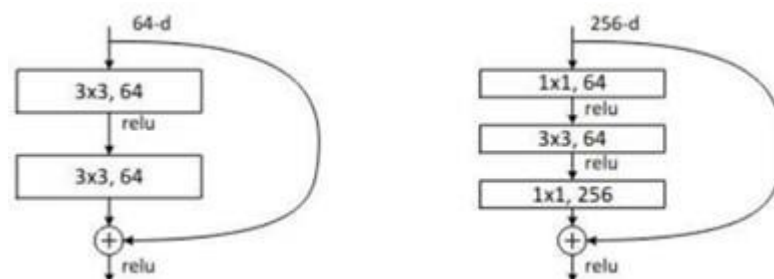
2.5 Augmentasi

Augmentasi yaitu memanipulasi atau memodifikasi citra sehingga citra yang asli akan berubah bentuk dan posisi. Augmentasi dilakukan agar mesin dapat mengenali citra objek dalam bentuk dan posisi yang berbeda-beda. Augmentasi dapat meningkatkan akurasi model yang dilatih, karena augmentasi memberikan informasi tambahan pada

model, yang dapat berguna dalam menciptakan model generalisasi yang lebih baik (Mahmud et al, 2019). Selain itu, augmentasi juga dapat memperbanyak jumlah data yang akan diproses. Proses augmentasi pada penelitian ini adalah *flip* dan *rotate*. *Flip* pada penelitian ini merupakan flip horizontal dan vertikal dan *rotate* pada penelitian ini merupakan rotasi sebesar 90 derajat.

2.6 Residual Network (ResNet)

Residual Network (ResNet) merupakan *artificial neural network* yang dikembangkan oleh (He et al). *ResNet* memanfaatkan *skip connection* untuk melompati beberapa *layer* atau lapisan sebagai jalan pintas. *Deep network* sering kali mengalami penurunan *gradient* saat model melakukan *back propagation*, *gradient* yang mengecil tersebut membuat pembelajaran lebih sulit. *Skip connection* ini memungkinkan jaringan untuk mempelajari *identity function* dengan melewati *input* dengan melewati satu *convolutional layer* ke *convolutional layer* lainnya (He et al, 2015). *ResNet* memiliki beberapa jenis diantaranya adalah *Resnet-34*, *ResNet-50*, *ResNet-101*, *ResNet-V2*, *ResNet-50V2*, dan lain sebagainya. Berbeda dengan *ResNet-34* yang melewati 2 blok, *ResNet-50* melewati 3 blok dan memiliki 1x1, 3x3, dan 1x1 *convolutional layer*. *Convolutional layer* pertama mengurangi dimensi, kemudian fitur di kalkulasikan pada *layer bottleneck* 3x3, lalu kemudian dimensi ditingkatkan lagi pada *layer* 1x1. Adapun visualisasi dari *ResNet* terdapat pada Gambar 2.4.



Gambar 2. 4 ResNet (He et al, 2015)

2.7 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah kelas pembelajaran mendalam yang dapat mengenali dan mengklasifikasikan gambar. Metode CNN adalah kelas jaringan saraf yang dikhususkan untuk memproses data dengan topologi mirip grid, seperti gambar. Metode CNN juga dapat digunakan dalam klasifikasi gambar, pengenalan wajah, analisis dokumen, klasifikasi video, dll.

Metode CNN mengklasifikasikan gambar dengan mengolah suatu gambar masukan kemudian mengklasifikasikannya ke dalam kategori tertentu, misalnya gambar manusia memiliki wajah, mata, bibir, hidung, tangan, dll. Gambar disajikan dalam matriks yang berisi nilai setiap piksel dengan dimensi resolusi*panjang* tinggi yang disebut saluran. Saluran ini biasanya terdiri dari tiga, artinya gambar tersebut merupakan gambar RGB, dimana setiap lapisan (saluran) mewakili warna *Red-Green-Blue*.

CNN memiliki 3 tipe *layer* yaitu *convolutional*, *pooling* dan *fully connected*.

Berikut penjelasan terkait lapisan yang ada pada CNN :

1. *Convolutional Layer*

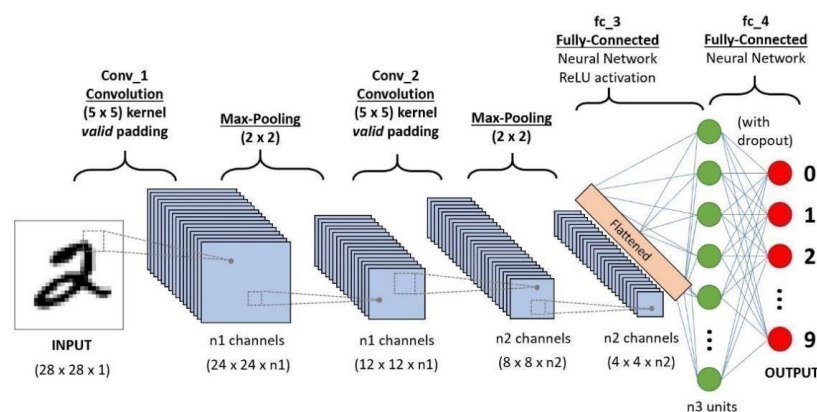
Lapisan yang melakukan ekstraksi fitur dalam piksel dari bagian terpenting wilayah tertentu.

2. *Pooling Layer*

Tujuannya adalah untuk menangkap informasi yang dihasilkan dengan mengekstraksi fitur langkah demi langkah untuk mengurangi ukuran representasi. Selain itu, beberapa parameter dikurangi untuk mengurangi kompleksitas komputasi model. Metode *pooling* yang umum digunakan adalah *Max Pooling* yang mencari nilai maksimum dan *Average Pooling* yang mencari nilai rata-rata.

3. *Fully Connected Layer* .

Lapisan di mana semua aktivitas dari *neuron* atau *layer* dikumpulkan untuk membuat perhitungan untuk menentukan distribusi kelas. Lapisan ini menyimpan kemungkinan gambar input dari kelas yang berbeda sesuai dengan pelatihan yang dilakukan.



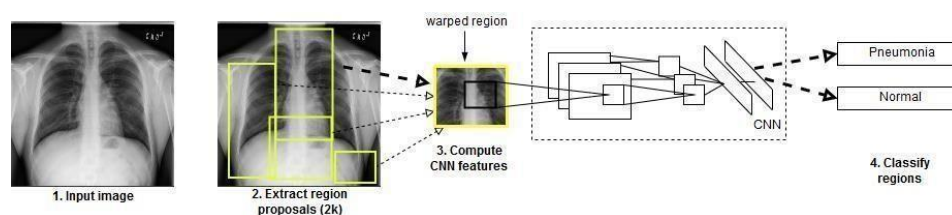
Gambar 2. 5 Arsitektur Umum Convolutional Neural Nework (Sumit Saha, 2018)

Seperti terlihat pada Gambar 2.7, arsitektur CNN dibagi menjadi dua bagian utama, yaitu *Feature Extraction Layer* dan *Fully Connected Layer*. Pada lapisan ekstraksi fitur, citra “*encoding*” menjadi peta fitur berupa angka-angka yang mewakili citra tersebut (*Feature Extraction*). *Feature Extraction Layer* terdiri dari *convolution layer* dan *pooling layer*. *Convolution layer* adalah bagian penting dari metode CNN yang membedakannya dari jaringan saraf lainnya. *Convolution layer* adalah lapisan pertama yang mengekstrak fitur dari gambar masukan. Proses konvolusi menjaga hubungan antar piksel dengan mempelajari fitur gambar menggunakan operasi matematika antara matriks gambar dan filter atau *kernel*. *Kernel* adalah operator yang digunakan di semua gambar untuk mendapatkan nilai *array* dari suatu gambar. *Kernel* biasanya berupa matriks 3×3 atau 5×5 yang berisi nilai acak antara -1 dan 1.

Secara singkat, metode CNN bekerja dengan melatih dan menguji setiap gambar dengan urutan proses. Dimulai dengan membagi gambar menjadi gambar yang lebih kecil, kemudian memasukkan setiap gambar yang lebih kecil ke dalam jaringan saraf yang lebih kecil, menyimpan hasil dari setiap gambar kecil ke dalam tabel baru, mengambil sampel atau memperkecil ukuran ruang untuk mengurangi jumlah parameter dan perhitungan saat ukuran gambar terlalu besar, besar dan dapat diprediksi.

2.8 Region Convolutional Neural Network (R-CNN)

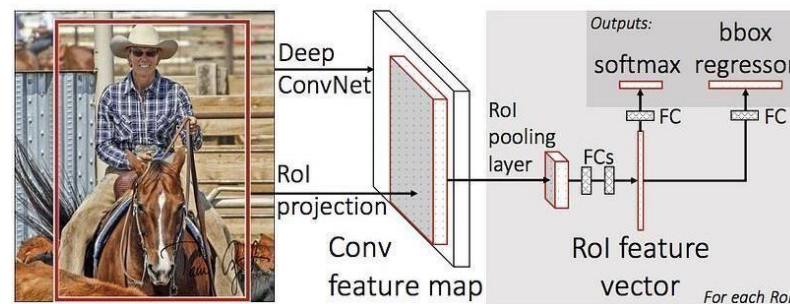
Region convolutional neural network merupakan metode untuk mendeteksi objek pada *image processing* yang berbasis *deep learning*. Perbaikan metode ini dari metode CNN adalah bahwa citra yang akan diinput akan dikelompokkan menjadi 2000 *region* dan dipilih berdasarkan tekstur, warna, dan intensitas. Pada metode CNN pembagian *region* citra dilakukan dengan skala *region* besar sehingga pemrosesan citranya lambat. R-CNN memiliki algoritma yang menemukan dua wilayah termirip dan kemudian digabungkan secara bersama. Algoritma tersebut digunakan pada *selective search*. Arsitektur *region convolutional neural network* bisa dilihat pada Gambar 2.6.



Gambar 2. 6 Arsitektur R-CNN (unicom.ac.id)

2.9 Fast Region Convolutional Neural Network (Fast R-CNN)

Metode dikembangkan oleh Ross Gershick pada tahun 2015 ini merupakan metode lanjutan dari R-CNN. Fast R-CNN melakukan proses *training* dan *testing* data lebih cepat dibandingkan R-CNN. Fast R-CNN hanya memiliki 1 CNN. Fungsi R-CNN sebagai *classifier* dengan *ROI Pooling* dan *fully-connected layer* digantikan oleh SVM. Hasil *feature map* dicocokkan pada ROI yang kemudian dilakukan proses pengklasifikasian kelas. Pendekatan dilakukan 1 CNN pada *feed forward network* dan *ROI pooling layer* berfungsi sebagai penambahan kapabilitas R-CNN menjadi *end-to-end differentiable* yang memudahkan proses pada saat *training*. Adapun arsitektur umum dari *Fast Region Convolutional Neural Network* berikut pada Gambar 2.7 dibawah ini :



Gambar 2. 7 Arsitektur Fast R-CNN (Diwakar Diwakar, 2022)

2.10 Faster Region Convolutional Neural Network (Faster R-CNN)

Faster R-CNN merupakan penyempurnaan dari algoritma Fast R-CNN. Algoritma R-CNN dan Fast R-CNN memerlukan waktu untuk mengetahui *region* proposal dengan menggunakan *selective search*, maka dari itu muncul metode Faster R-CNN yang dapat menghilangkan algoritma selektif serta jaringan dapat mempelajari proposal wilayahnya.

Metode Faster R-CNN terdiri dari dua langkah yaitu ekstraksi dan pelatihan menggunakan *Region Proposal Network* (RPN) dan klasifikasi fitur yang dihasilkan. Terdapat empat langkah dalam proses pelatihan, yaitu:

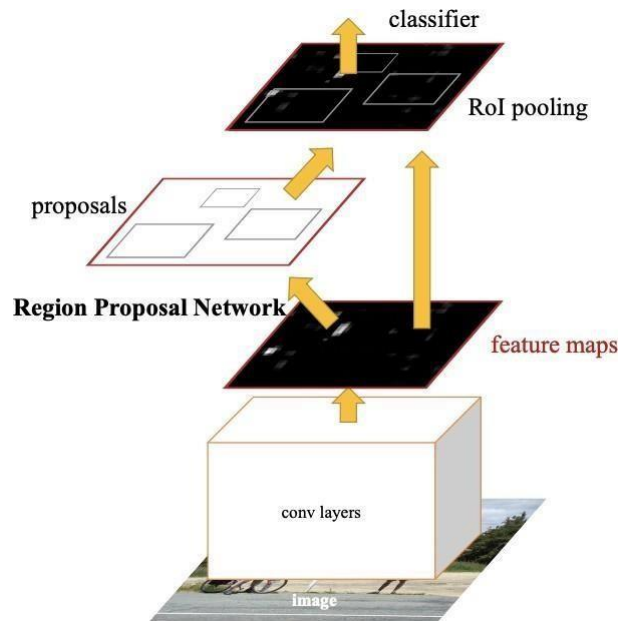
- Pelatihan R-CNN
- Pelatihan R-CNN lebih cepat dengan RPN
- Pelatihan ulang RPN dengan R-CNN lebih cepat
- R-CNN lebih cepat dan pelatihan ulang RPN diperbarui (Hermawati, Tjandrasa, & Suciati, 2018).

Pada Faster R-CNN, citra diinput dan diproses pada *layer* CNN. Proses konvolusi tersebut menggunakan pre-trained model Resnet50. Kemudian output dari proses tersebut merupakan *feature map* yang menyediakan peta fitur konvolusional. Proposal wilayah kemudian diproses dengan *Region Proposal Network* yang memiliki 9 *anchor* yang berfungsi untuk menilai mana yang bukan objek dan mana yang objek, kemudian diprediksi lalu dibentuk ulang menggunakan lapisan penggabungan ROI yang digunakan untuk mengklasifikasikan gambar dalam *region proposal* dan memprediksi nilai *offset* pada *bounding box*.

Faster R-CNN memiliki metode RPN yang dapat mendeteksi objek dengan kecepatan 0,2 detik per gambar. Pendeteksian dilakukan dengan menelusuri ciri objek pada citra melalui proses konvolusi atau CNN. Faster R-CNN memiliki tiga fungsi penting yaitu untuk mengklasifikasikan dan menghasilkan *feature map* dengan *deep neural network*, menghasilkan proposal wilayah oleh *Regional Proposal Network* (RPN), dan menemukan regresi dan menambah lapisan konvolusional oleh *regressor* (Nguyen et al., 2018). Adapun bagian-bagian pada Faster R-CNN adalah :

- *Convolutional layer*, terdiri dari lapisan yang membentuk suatu filter dua dimensi dengan panjang dan tinggi. Bagian ini berfungsi untuk mempelajari bagian yang akan dijadikan ciri dari sebuah objek dan membuat *feature map* dari objek yang telah diambil.
- *Feature map*, yang memberi informasi tentang representasi vektor dari citra yang diambil. Bagian ini terbentuk oleh *convolutional layer*.
- *Region proposal network*, yang berfungsi untuk mengolah *feature map* yang telah tercipta dan memprediksi *bounding box* objek dan bagian yang dianggap sebagai objek tersebut.
- *ROI pooling*, berfungsi untuk mengekstrak *feature map* yang sebelumnya diproses *Region Proposal Network* (RPN) untuk kemudian diklasifikasikan pada *classification layer*.
- *Classification layer*, yaitu proses terakhir yang dilakukan pada pengklasifikasian objek yang sebelumnya dideteksi oleh RPN dan membuat label objek yang telah diklasifikasikan.

Adapun arsitektur umum dari Faster R-CNN berikut pada Gambar 2.8 dibawah ini :



Gambar 2. 8 Arsitektur Umum Faster R-CNN (Nguyen et al., 2018)

Faster R-CNN memiliki beberapa kelebihan utama yaitu:

1. Menghasilkan performa yang lebih cepat dan akurat
2. Proposal region tidak sebesar R-CNN sehingga mempercepat dan mempermudah proses kerja
3. Bisa digunakan pada proses pengklasifikasian yang bersifat *real time* karena pemrosesannya hanya sekitar 1-2 detik. Adapun persamaan untuk menghitung *loss function* dari *Faster R-CNN* adalah sebagai berikut :

$$L(\{P\}).\{t\} = \frac{1}{N_{cls}} \sum iL_{cls}(P_i, P_i + \frac{1}{N_{reg}} : \sum iL_{reg}(t_i, t_i) \quad (2,2)$$

Keterangan :

P_i = Prediksi anchor pada gambar dan persamaan ground-truth label

$$P_i^* = \begin{cases} 0 & \text{negative label} \\ 1 & \text{positive label} \end{cases}$$

$t_i = \{p_x, p_y, p_w, p_h\}$ adalah representasi vector dari 4 parameter pada bounding box dimana didalamnya terdapat sebuah objek yang terprediksi.

Ada beberapa parameter yang berpengaruh untuk metode *Faster Region Convolutional Neural Network* :

1. Distribusi Dataset

Distribusi dataset mengacu pada bagaimana objek yang ada dalam dataset

didistribusikan dalam berbagai kelas (kategori) dan dalam berbagai tingkat kesulitan.

2. *Learning Rate*

Learning rate digunakan mengontrol sejauh mana bobot dari model diperbarui selama proses pelatihan. Ini adalah faktor yang menentukan seberapa besar atau kecil langkah-langkah perubahan dilakukan pada bobot model selama setiap iterasi pelatihan. *Learning rate* memiliki pengaruh besar terhadap keberhasilan pelatihan model. Semakin kecil *learning rate* maka akan semakin besar akurasi yang didapat.

3. Epochs

Epochs merujuk pada satu siklus lengkap pelatihan model terhadap seluruh dataset pelatihan. *Epochs* menggambarkan satu kali proses di mana seluruh dataset pelatihan digunakan untuk melatih model. Selama satu epoch, setiap sampel dalam dataset digunakan satu kali dalam proses pelatihan. Namun jika terlalu banyak epochs dapat mengakibatkan *overfitting*, yaitu kondisi di mana model menjadi terlalu spesifik terhadap data pelatihan dan kehilangan kemampuan untuk menggeneralisasi dengan baik pada data uji atau data yang belum pernah dilihat.

4. Steps

Steps merujuk pada satu iterasi di mana model menghitung gradien (perubahan parameter) dan memperbarui bobotnya.

Adapun cara yang dapat dilakukan untuk menentukan parameter terbaik model *Faster Region Convolutional Neural Network* adalah :

1. Pemilihan Dataset

Dapat memilih dataset yang sesuai untuk tugas deteksi objek. Pastikan dataset tersebut mencerminkan variasi yang ada di dunia nyata dalam hal kelas objek, tingkat kesulitan, lokasi, geometri, dan lainnya.

2. Arsitektur Model

Pilih arsitektur model yang akan digunakan sebagai dasar Faster R-CNN, Adapun beberapa arsitektur yang sering digunakan adalah *VGG16*, *ResNet*, atau arsitektur lainnya. Pemilihan arsitektur ini akan memengaruhi kapasitas dan akurasi model.

3. Hyperparameter

Menggunakan beberapa hyperparameter *learning rate*, *batch size*, jumlah *epochs*, dan mulai melakukan eksperimen untuk menentukan *hyperparameter* terbaik.

4. *Data Augmentation*

Melakukan pertimbangan dalam penggunaan augmentasi data, seperti rotasi, pergeseran, perbesaran, atau pencerahan, untuk meningkatkan kemampuan model dalam mengatasi variasi data.

5. *Optimizer*

Menggunakan optimizer yang sesuai, seperti SGD atau Adam, dan mengatur parameter optimizer.

6. Validasi dan *Monitoring*

Menggunakan dataset validasi untuk memonitor kinerja model selama pelatihan. Memantau metrik seperti akurasi, presisi, *recall*, dan *F1-score* untuk menilai kinerja model.

7. Evaluasi Hasil

Melakukan evaluasi berbagai konfigurasi, pilih konfigurasi yang memberikan kinerja terbaik pada dataset validasi atau dataset uji yang terpisah.

2.11 Android Studio

Android Studio adalah *Integrated Development Environment (IDE)* resmi untuk pengembangan aplikasi Android. *Android studio* menyajikan fitur utama *Interface* pengembangan untuk menata letak *interface* dan mengelola aplikasi. Adapun fitur- fitur dari *Android studio* yaitu:

- Menggunakan *Google Cloud Platform*.
- Memiliki *framework* dan alat *testing* lengkap.
- Menggunakan sistem pembangunan berbasis *gradle*.
- Menggunakan *Emulator* untuk memastikan kinerja aplikasi.
- Terintegrasi dengan *GitHub*.

2.12 Python

Python mewakili satu *framework* atau *library* yang digunakan dalam pengembangan pembelajaran *deep learning*. Keras dirancang untuk menyederhanakan penggunaan dan mengurangi kompleksitas dalam pengembangan model pembelajaran mendalam. Keras didefinisikan dengan tujuan meningkatkan algoritma RNN dan CNN, atau bahkan pengaturan dual-link yang berfungsi dengan baik pada GPU lebih dari CPU

Sejak *TensorFlow* versi 2.0, Keras telah menjadi API tingkat 1. Resmi tinggi *TensorFlow*, yang berarti Keras dapat digunakan sebagai bagian penting dari lingkungan pengembangan *TensorFlow*.

2.13 Tensorflow Lite

Tensorflow lite merupakan sebuah *library open-source* yang digunakan untuk pembelajaran mesin (*machine learning*) untuk menjalankan model *tensorflow* pada perangkat *mobile*.

2.14 Confussion Matrix

Confussion Matrix merupakan pengukuran performa untuk suatu permasalahan klasifikasi dan nilai output yang asli dibandingkan dengan output nilai yang diinginkan untuk menghasilkan nilai akurasi (Suyanto, 2018:331). *Confussion matrix* membentuk empat nilai, diantaranya adalah:

- True Positive (TP), merupakan nilai positif dengan prediksi bernilai positif
- True Negative (TN), merupakan nilai negatif dengan prediksi bernilai negatif
- False Positive (FP), merupakan nilai negatif dan prediksi bernilai positif
- False Negative (FN), merupakan nilai positif dengan prediksi negative

Metode perhitungan *confussion matrix* untuk menghitung *performance matrix* adalah:

1. Accuracy

Accuracy merupakan nilai yang di dapat dari nilai *output* yang asli bernilai positif dengan prediksi bernilai positif dan nilai *output* bernilai *negative* dengan prediksi yang bernilai *negative*. *Accuracy* berikut dari kedekatan nilai *output* asli dengan nilai prediksi benar untuk menghitung berapa akurasi model mengklasifikasikan secara benar. Semakin banyak nilai dengan diprediksi benar maka semakin tinggi nilai akurasi dari suatu penelitian.

$$Akurasi = \frac{\text{Jumlah data dengan prediksi benar}}{\text{Jumlah keseluruhan data}} \quad (2.3)$$

2. Precision

Precision merupakan nilai prediksi positif dibanding dengan nilai keseluruhan hasil prediksi positif yang didapat oleh sistem dengan jumlah data yang terbaca oleh sistem dengan sesuai ataupun tidak sesuai.

$$Precision = \frac{TP}{TP+FP} \quad (2.4)$$

3. Recall

Recall merupakan nilai peluang prediksi positif dibandingkan dengan nilai keseluruhan bernilai positif. *Recall* digunakan untuk menghitung perbandingan dari jumlah data yang sesuai berdasarkan hasil dari sistem pada kumpulan data.

$$Recall = \frac{TP}{TP+FN} \quad (2.5)$$

4. *F1-Score* merupakan metode yang digunakan untuk menghitung perbandingan dari rata-rata *precision* dan *recall* yang dibobotkan. Adapun persamaan dari *F1 Score* terdapat pada Persamaan 2.6.

$$F1 - Score = \frac{Recall \times Precision}{Recall + Precision} \quad (2.6)$$

2.15 Penelitian Terdahulu

Sebelum dilakukannya penelitian ini, sudah pernah dilakukan penelitian untuk mengidentifikasi kualitas jagung berdasarkan fitur warna HSV dan tekstur GLCM yang dilakukan oleh Miftahul Jannah pada tahun 2018 dengan berjudul “Identifikasi Mutu Jagung Menggunakan Fitur Warna Dan Tekstur Berbasis Pengolahan Citra Digital Dan Algoritma K-Nearest neighbor (KNN)” pada penelitian ini digunakan fitur HSV karena warna mirip dengan system penglihatan manusia, sedangkan fitur tekstur digunakan untuk pendugaan kadar air, karena tinggi rendahnya kadar air dapat mempengaruhi tekstur jagung. pada penelitian ini digunakan data ratio 70% dan 30%, jumlah data uji latih yang digunakan sebanyak 350 citra dan data uji sebanyak 150 citra. Akurasi tertinggi diperoleh sebesar 90%.

Penelitian yang dilakukan Ulla Delfana Rosiani, Mustika Mentari, dan Andi Novan Prastya, penelitian ini dilakukan untuk mengklasifikasi kualitas biji jagung berdasarkan deteksi warna dan bentuk menggunakan metode *k-nearest neighbor* dengan masukan berupa nilai warna dan bentuk dari hasil deteksi pada biji jagung.. Pengujian pada penelitian ini menggunakan 30 citra input dengan rincian 10 citra input uji coba 1 menggunakan 10 citra input, uji coba 2 menggunakan 10 citra input, uji coba 3. Dengan hasil akurasi sebesar 76,67% dan presentase error atau kesalahan program sebesar 23,33%. Penelitian ini berjudul “Klasifikasi Kualitas Biji Jagung Berdasarkan Deteksi Warna dan Bentuk Menggunakan Metode K-Nearest Neighbor”.

Penelitian yang dijadikan acuan dalam memahami algoritma Faster R-CNN

adalah penelitian yang dilakukan oleh Subhan & Basri (2019) dengan judul “Klasifikasi Mutu Buah Pala (*Myristica Fragrans Houtt*) Berbasis Pengolahan Citra Menggunakan Metode *Deep Learning* Arsitektur Faster R-CNN”. Peneliti memiliki tujuan untuk mengembangkan metode yang mengklasifikasikan biji buah pala, dengan menggunakan metode pengolahan citra dengan parameter warna dan bentuk memakai metode Faster R- CNN. Penelitian ini memperoleh hasil akurasi terbaik sebesar 95% dengan *learning rate* 4000 dengan waktu proses 0.04 menit per detik.

Dan selain itu ada juga penelitian menggunakan metode Faster RCNN yang dilakukan oleh Yoze Rizky, Reny medikawaty Taufiq, Dinia Putri, dan Harun Mukhtar yang berjudul “Klasifikasi Pola Kain Tenun Melayu Menggunakan Faster R-CNN” dengan tingkat akurasi sebesar 82,14% dengan model VGG yang dipilih untuk merancang sebuah sistem untuk mengklasifikasikan motif tenun melayu.

Tabel 2. 1 Penelitian Terdahulu

No.	Penulis	Tahun	Metode	Keterangan
1.	MIFTAHUL JANNAH	2018	<i>K-NEAREST NEIGHBOR (K- NN)</i>	Penelitian ini menghasilkan sistem yang dapat mengidentifikasi mutu jagung menggunakan fitur warna dan tekstur dengan memperoleh tingkat akurasi sebesar 90%.
2.	Ulla Delfana Rosiani., Mustika Mentari, Andi Novan Prastya.	2019	<i>K-Nearest Neighbor</i>	Penelitian ini menghasilkan sistem yang dapat mengidentifikasi mutu jagung menggunakan deteksi warna dan bentuk dengan memperoleh tingkat akurasi sebesar 76,67%.

No.	Penulis	Tahun	Metode	Keterangan
3.	Jihad Ardiansyah, Rita Purnamasari, Bambang Hidayat	2020	<i>DISCRET WAVELET TRANSFORM</i>	Penelitian ini menghasilkan sistem yang dapat mengklasifikasikan kualitas jagung menggunakan deteksi tekstur dengan memperoleh tingkat akurasi sebesar 93,33%.
4.	Muh. Subhan, Hasan Basri	2019	<i>Faster R-CNN</i>	Penelitian ini bertujuan untuk mengembangkan metode pengolahan citra dengan parameter warna dan bentuk dan memperoleh hasil akurasi terbaik sebesar 95% dengan learning rate 4000 dengan waktu proses 0.04 menit/detik.
5.	Yoze Rizky, Reny medikawaty Taufiq, Dinia Putri, dan Harun Mukhtar	2021	<i>Faster R-CNN</i>	Penelitian ini mengembangkan metode Faster RCNN menggunakan arsitektur VGG didapatkan akurasi 82.14%.
6.	MUHAMMAD FAJAR AL HANIEF	2020	<i>Artificial Neural Network (ANN)</i>	Penelitian ini mengklasifikasikan kualitas biji jagung dengan menggunakan metode <i>Artificial Neural Network</i> dan mendapat akurasi terbaik sebesar 95 persen.

2.16 Perbedaan Penelitian

Perbedaan penelitian ini dengan penelitian sebelumnya ialah, pada penelitian ini data yang digunakan untuk penelitian ini berbeda dengan penelitian yang telah ada karena pengambilan data dilakukan oleh penulis, dan langsung dari hasil panen jagung di daerah sumatera utara. Pada penelitian sebelumnya belum ada yang melakukan penelitian klasifikasi kualitas jagung dengan metode Faster Regional Convolutional Network, metode ini memiliki kecepatan deteksi yang lebih cepat dan ekstraksi fitur yang lebih kompleks. Sampel data yang digunakan pada penelitian ini diambil dengan menggunakan alat pengukur kelembapan biji-bijian yaitu Grain Moisture Meter LDS-1G sebagai acuan dalam pembagian kelas kualitas biji jagung.

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Data yang Digunakan

Data yang digunakan dalam penelitian ini berupa gambar biji jagung varietas *pionner*, data citra Biji Jagung diambil dari Kecamatan Munte, Kabupaten Karo, Sumatera Utara. Data citra Biji Jagung diambil langsung dari petani setelah proses pemipilan yang digunakan dengan cara mekanis (menggunakan mesin pemipil jagung). Pada penelitian ini, sampel biji jagung yang digunakan akan diukur kadar airnya terlebih dahulu menggunakan alat *Grain Moisture Meter LDS-1G* sebagai acuan dalam menentukan kelas dari biji jagung tersebut. *Grain Moisture Meter* merupakan alat pengukur kelembaban biji-bijian yang dirancang untuk mengukur kadar air yang terkandung dalam biji-bijian seperti biji jagung, alat ini menggunakan sensor atau probe yang ditempatkan dalam sampel biji-bijian. Alat ini digunakan untuk memastikan kualitas biji-bijian seperti dalam penelitian ini alat ini digunakan untuk membantu dalam menentukan kualitas biji jagung dan membagi kualitas menjadi beberapa kelas yang akan di klasifikasikan.



Gambar 3. 1 Pengukuran Kadar Air Biji Jagung Menggunakan Grain Moisture Meter

Biji jagung yang digunakan untuk menjadi sampel merupakan kumpulan biji jagung sebanyak segenggam tangan yang memiliki kualitas sama karena biji-biji jagung yang digunakan berasal dari lahan yang sama perawatan jagung mulai dari benih menjadi jagung yang siap di panen dengan proses yang sama, dan cuaca yang sama juga.

Pada saat proses panen, jagung di panen dengan dengan mengupas kulit jagung

dan menyatukan jagung yang sudah di kupas kulitnya ke dalam karung setelah itu karung yang berisi jagung tersebut akan di bawa ke gudang penyimpanan jagung, setelah itu jagung masuk ke dalam proses pengeringan dengan mengeringkan secara manual dengan memanfaatkan cahaya matahari, proses pengeringan dilakukan dengan mengeringkan jagung yang memiliki waktu panen yang sama, sehingga jagung memiliki kategori kualitas yang sama dan tidak tercampur, karena apabila tercampur akan mempersulit proses penjualan kepada pengepul karena pengepul tidak akan mau menerima jagung yang memiliki kualitas berbeda. Setelah proses pengeringan selanjutnya jagung akan dipilpil untuk memisahkan biji jagung dengan tongkol jagung, untuk memudahkan pengepul jagung dalam membawa biji jagung tersebut dan untuk mengetahui berat biji jagung tersebut tanpa tongkol jagung. Citra biji jagung yang di gunakan dalam data pada penelitian ini.

Jumlah total data sebanyak 2500 citra Biji Jagung, dan setiap kelas atau kualitas berjumlah 500 data per atau kualitas. Data diambil menggunakan kamera *smartphone* berektensi *jpg*. Contoh data citra biji jagung yang akan digunakan pada penelitian ini dapat di lihat pada Gambar 3.2 dan tabel pembagian data biji jagung berdasarkan kualitas berikut pada Tabel 3.1



Gambar 3. 2 Citra data yang digunakan

Pengklasifikasian kualitas biji jagung berdasarkan tingkat kualitas biji jagung di bagi menjadi 5 kelas atau kualitas, Seluruh data di bagi menjadi data *training*, data *testing*, dan data *validation*. Data Training merupakan data yang digunakan untuk proses melatih dan membangun model. Data Testing merupakan data yang digunakan untuk

menguji model setelah proses pelatihan. Data Validation merupakan data yang digunakan untuk mengoptimalkan pada saat melatih model, dan juga berguna untuk mem-validasi model mencegah terjadinya *overfitting*.

Tabel 3. 1 Banyak Data

Dataset	Jumlah Dataset
A	500
B	500
C	500
D	500
E	500
Jumlah Seluruh Data	2500

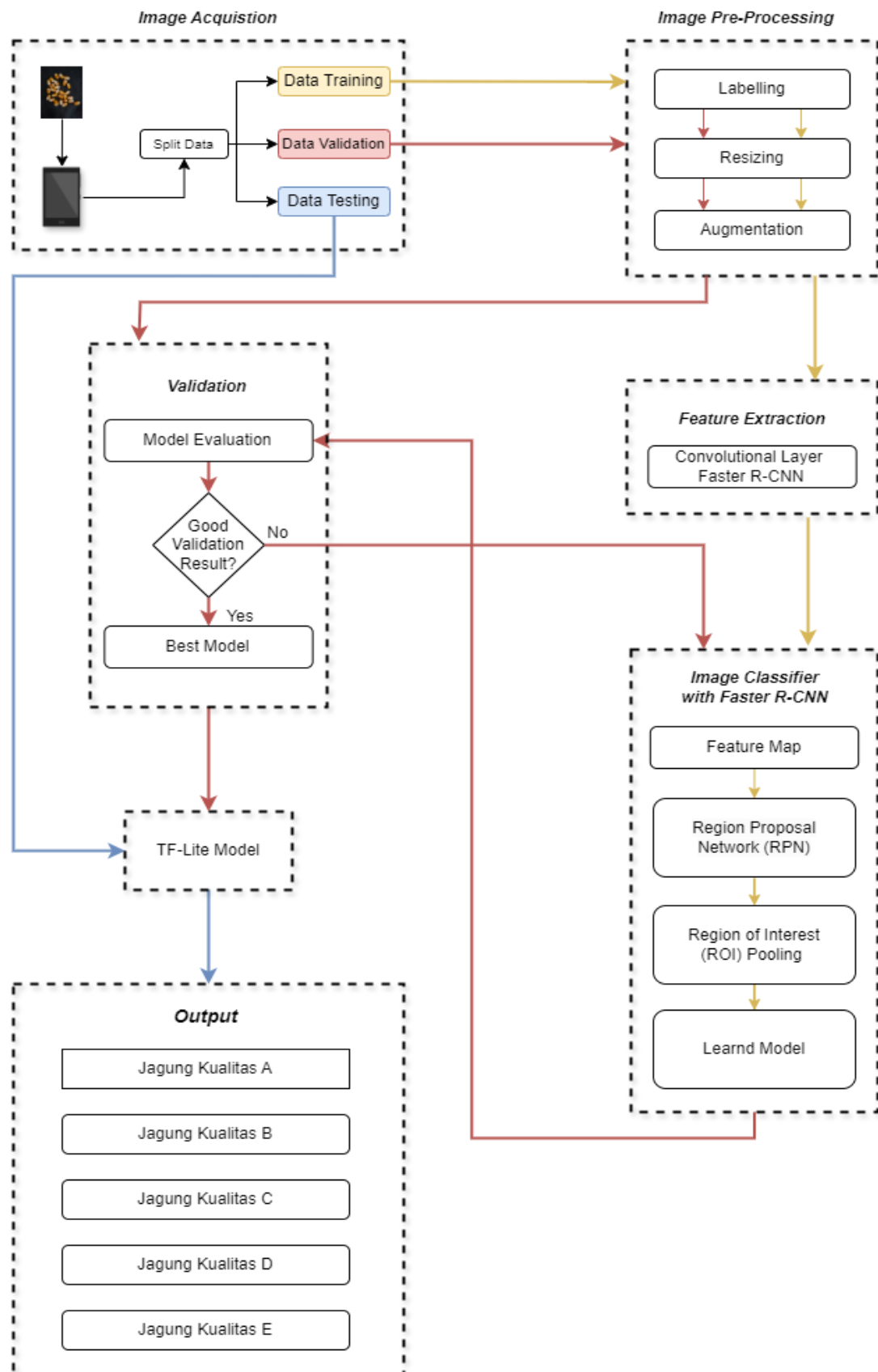
Pembagian jumlah data dilakukan dalam bentuk persentase dengan data *training* sebesar 80%, data *testing* sebesar 10%, dan data *validation* sebesar 10%.

3.2 Analisis Sistem

Penelitian ini memerlukan beberapa langkah atau tahapan dalam proses analisis, tahap pertama pada penelitian ini dilakukan pengumpulan data citra biji jagung. Data citra biji jagung tersebut di data dengan membagi kategori kualitas jagung yang akan diklasifikasikan, biji jagung dengan tingkat kualitas A, yaitu kualitas jagung dengan kualitas terbaik yang memiliki kadar air 15% - 20%. Kualitas B, yaitu kualitas jagung yang memiliki kadar air 21% - 25%. kualitas C, yaitu kualitas jagung yang memiliki kadar air 26% - 30%. kualitas D, yaitu kualitas jagung yang memiliki kadar air 31% - 35%. Dan kualitas E, yaitu kualitas jagung yang memiliki kadar air 36% - 40%. Seluruh data citra biji jagung diambil dengan menggunakan kamera *smartphone*.

Tahap berikutnya merupakan tahap *pre-processing*, Pada tahap *pre-processing*, tahap pertama yang akan dilakukan adalah proses *labeling* untuk memberikan penamaan pada citra biji jagung dan proses *resizing* untuk merubah ukuran *pixel* setiap gambar biji jagung mejadi ukuran 256 x 256 *pixel*. Berikutnya citra melewati proses augmentasi yaitu dengan melakukan *flip* gambar secara vertikal dan horizontal serta rotasi gambar sebesar 90 derajat.

Tahap selanjutnya merupakan tahap pengklasifikasian data citra, penelitian ini menggunakan fitur ekstraksi yang terdapat pada algoritma *Faster Region Convolutional Neural Network* dengan bantuan *pre-trained* model ResNet50 sebagai *backbone*. *Output* yang dihasilkan setelah tahap klasifikasi berupa 5 kategori kualitas berdasarkan kadar air dari biji jagung, kategori kualitas biji jagung yaitu kategori A, kategori B, kategori C, kategori D, kategori E. Keseluruhan tahapan pada penelitian ini berikut pada arsitektur umum pada gambar 3.3 berikut.



Gambar 3. 3 Arsitektur Umum

Kualitas jagung di petani diklasifikasikan berdasarkan PERMENDAGRI No. 27/M-DAG/PER/5/2017 sesuai tabel. Kualitas ini didasarkan pada kadar air dan dijadikan tolak ukur untuk menentukan harga beli jagung di tingkat petani.

Tabel 3. 2 Kategori Jagung Berdasarkan Klasifikasi

Jenis Uji	Satuan	Kualitas A	Kualitas B	Kualitas C	Kualitas D	Kualitas E
Kadar Air	%	15-19	20-24	25-59	30-34	35-39
Butir Rusak	%	2	2	2	2	2
Butir Berjamur	%	2	2	2	2	2
Butir Pecah	%	2	2	2	2	2

Berikut ini merupakan penjelasan terkait proses arsitektur:

3.2.1 *Image Acquisition*

Image Acquisition adalah proses tahapan awal sistem berupa pengumpulan data citra biji jagung sehingga diperoleh citra digital. Pada penelitian ini penulis menggunakan citra biji jagung sebagai input pada aplikasi. Citra biji jagung yang digunakan dibagi menjadi 5 kategori berdasarkan kualitas biji jagung yaitu, kualitas A, kualitas B, kualitas C, kualitas D, kualitas E. Data merupakan dasar utama dalam analisis dan implementasi metode *Faster R-CNN* yang digunakan penulis untuk mengklasifikasikan kualitas biji jagung. Data Citra diambil dengan *smartphone* dan berekstensi *.jpeg/.jpg*. Keseluruhan data citra biji jagung dibagi menjadi 3 bagian yaitu, 80% data *training*, 10% data *testing*, dan 10% data *validasi*.

3.2.2 *Image Pre-Processing*

Proses adalah proses pengolahan gambar biji jagung agar mendapat hasil lebih baik untuk diproses ke tahap selanjutnya. Tahapan *preprocessing* yang dilakukan pada penelitian ini yaitu *Labelling*, *Resizing*, dan Augmentasi data.

3.2.2.1 *Labelling*

Proses *labelling* merupakan proses menamai atau memberikan label pada data yang dapat digunakan untuk melatih sistem. Pelabelan dilakukan dengan pengelompokan data

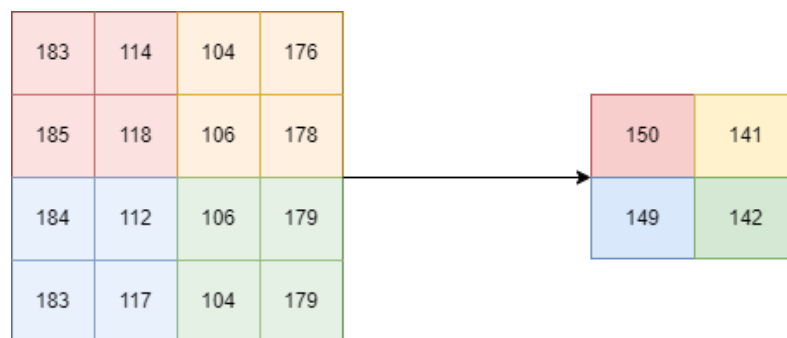
biji jagung berdasarkan kualitas biji jagung, proses pelabelan citra biji jagung menggunakan *Google Drive* dan dikelompokkan kualitas biji jagung yang sama ke dalam satu *folder* dimana *folder* memiliki nama sesuai dengan tingkatan klasifikasinya, terdapat 5 folder, yaitu kategori kualitas biji jagung kualitas “A”, kualitas”B”, kualitas”C”, kualitas”D”, dan kualitas “E” dan masing masing folder merupakan tingkatan kualitas biji jagung yang akan diklasifikasikan. Proses *labelling* berikut pada Gambar 3.4



Gambar 3. 4 Labelling

3.2.2.2 Resizing

Proses *Resizing* adalah proses pengubah ukuran citra yang memiliki satuan *pixel*. Pada penelitian ini proses *resizing* digunakan untuk mengubah ukuran gambar biji jagung menjadi lebih kecil karena semakin besar ukuran *pixel* pada gambar akan membuat pemrosesan menjadi semakin lama, oleh karena itu dilakukan proses *resizing* agar ukuran *pixel* pada citra lebih sedikit dan untuk menyesuaikan ukuran citra biji jagung sehingga semua gambar berukuran yang sama. Pada penelitian ini, data citra yang dikumpulkan memiliki ukuran 3000 x 4000 *pixel* dengan ukuran *file* kurang lebih 3 MB, pada proses ini ukuran citra diubah menjadi 256 x 256 *pixel* sehingga memudahkan proses pelatihan pada sistem dan menghemat sumber daya komputasi. Proses *resizing* berikut pada Gambar 3.5



Gambar 3. 5 Proses Perhitungan Resizing

Proses perhitungan nilai piksel dari 4 x 4 menjadi 2 x 2 adalah sebagai berikut:

$$P1 = (183+114+185+118) : 4 = 150$$

$$P2 = (104+176+106+178) : 4 = 141$$

$$P3 = (184+112+183+117) : 4 = 149$$

$$P4 = (106+179+104+179) : 4 = 142$$

P1, P2, P3, dan P4 merupakan hasil *resizing* dari piksel 4 x 4 menjadi 2 x 2. Berikut merupakan citra data biji jagung sebelum dan setelah di *resizing*.



Gambar 3. 6 Citra Jagung sebelum (kiri) dan sesudah di-resize (kanan)

Tahap untuk menjalankan proses *resizing* data gambar dilakukan melalui *Google Colaboratory* yang tertera pada *pseudocode* berikut.

```
folder_hasil <-- "path/to/your/output_folder"
nama_folder <-- "your_folder_name"
target_size <-- (256, 256)
```

Gambar 3. 7 Pseudocode Proses Resizing

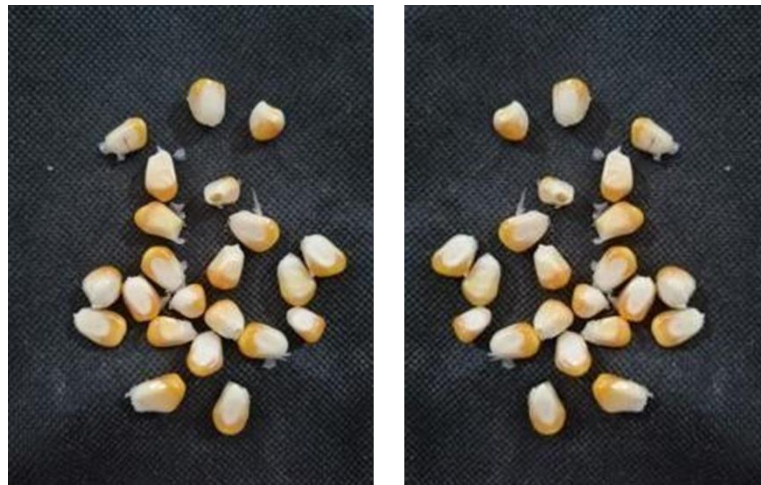
3.2.2.3 Augmentasi

Augmentasi merupakan proses citra dimanipulasi dan dimodifikasi sehingga citra asli akan diubah bentuk dan posisinya. Tujuan augmentasi agar mesin dapat belajar mengenali citra yang berbeda-beda dan untuk memperbanyak data sehingga data menjadi lebih bervariasi. Proses augmentasi telah digunakan dalam banyak kasus dan berhasil meningkatkan performa dari sistem sehingga sistem berhasil mengenali lebih banyak objek dari bentuk dan pola yang beragam. Pada penelitian ini, proses augmentasi

yang dilakukan adalah *flip* citra biji jagung secara *vertical* dan *horizontal* serta rotasi gambarsebesar 90 derajat yang akan membantu peningkatan kuantitas dataset. Proses augmentasi berikut pada Gambar 3.8, Gambar 3.9, dan Gambar 3.10.



Gambar 3. 8 Sebelum dan Sesudah Rotasi 90 Derajat



Gambar 3. 9 Sebelum dan Sesudah Flip Kiri Kanan



Gambar 3. 10 Sebelum dan Sesudah Flip Atas Bawah

Adapun *pseudocode* augmentasi ditunjukkan pada Gambar 3.11 di bawah ini:

```
transform_and_save(data, folder_hasil, nama_folder, index, "flip_left_right")
index += 1

transform_and_save(data, folder_hasil, nama_folder, index, "rotate_90")
index += 1

transform_and_save(data, folder_hasil, nama_folder, index, "flip_top_bottom")
index += 1
```

Gambar 3. 11 *Pseudocode Proses Augmentasi*

3.2.3 Feature Extraction

Feature Extraction merupakan proses dimana *convolutional layer* digunakan sebagai *feature map* dari layer setiap objek yang terdapat pada algoritma *Faster R-CNN* digunakan untuk mengekstraksi ciri pada gambar dan pelajari bagian-bagian penting. akan menjadi ciri/*feature* dari suatu bentuk yang didapatkan akan dianalisis untuk pengenalan gambar. *Convolution layer* merupakan neuron yang disusun membentuk filter panjang dan tinggi dalam piksel.

Pada penelitian ini, *convolution layer* yang digunakan adalah *ResNet50*. Pada layer ini *filter* dilatih untuk melakukan ekstraksi fitur yang sesuai dengan gambar. Kovolusi dihitung dengan menggeser *filter* yang sejajar dengan gambar *input* dan menghasilkan *matrix* dua dimensi yang disebut dengan *feature map*. *Activation map* dibuat dengan menggeser *filter* dari satu titik ke setiap kemungkinan lokasi *filter* pada gambar

dengan menggunakan *filter* yang berbeda sehingga menghasilkan gambar yang baru. Proses *convolution layer* berikut pada Gambar 3.12.

182	114	103	176	95
185	117	106	178	100
184	113	106	179	101
183	117	105	179	101
183	116	107	178	100

Gambar 3. 12 Matriks Input Gambar

Kemudian matriks dari input gambar dihitung dengan menggunakan dengan *filter* yang terdapat pada *convolutional layer*. *Activation map* dihasilkan dari pengulangan proses perhitungan ini untuk setiap elemen dari gambar *input*. Adapun contoh perhitungan *filter* pada matriks gambar pada penelitian adalah :

182	114	103	176	95								
185	117	106	178	100	0	1	0	A1	A2	A3		
184	113	106	179	101	×	[1	0	1]	=	A4	A5	A6
183	117	105	179	101	0	1	0	A7	A8	A9		
183	116	107	178	100								

Gambar 3. 13 Perhitungan Filter Matriks

Pada *Convolution Layer* nilai matriks gambar *input* pada penelitian ini sebesar 5×5 *pixel*. Kemudian nilai tersebut akan dikalikan dengan nilai *filter* yaitu sebesar 3×3 . Matriks gambar input akan bergeser sebesar 3×3 dari kiri atas ke kanan dengan pergeseran 1 *pixel* sampai melewati semua *pixel* dari baris pertama. Pergeseran matriks tersebut berikut dibawah ini :

182	114	103	176	95	182	114	103	176	95	182	114	103	176	95
185	117	106	178	100	185	117	106	178	100	185	117	106	178	100
184	113	106	179	101	184	113	106	179	101	184	113	106	179	101
183	117	105	179	101	183	117	105	179	101	183	117	105	179	101
183	116	107	178	100	183	116	107	178	100	183	116	107	178	100
182	114	103	176	95	182	114	103	176	95	182	114	103	176	95
185	117	106	178	100	185	117	106	178	100	185	117	106	178	100
184	113	106	179	101	184	113	106	179	101	184	113	106	179	101
183	117	105	179	101	183	117	105	179	101	183	117	105	179	101
183	116	107	178	100	183	116	107	178	100	183	116	107	178	100
182	114	103	176	95	182	114	103	176	95	182	114	103	176	95
185	117	106	178	100	185	117	106	178	100	185	117	106	178	100
184	113	106	179	101	184	113	106	179	101	184	113	106	179	101
183	117	105	179	101	183	117	105	179	101	183	117	105	179	101
183	116	107	178	100	183	116	107	178	100	183	116	107	178	100
182	114	103	176	95	182	114	103	176	95	182	114	103	176	95
185	117	106	178	100	185	117	106	178	100	185	117	106	178	100
184	113	106	179	101	184	113	106	179	101	184	113	106	179	101
183	117	105	179	101	183	117	105	179	101	183	117	105	179	101
183	116	107	178	100	183	116	107	178	100	183	116	107	178	100

Gambar 3. 14 Pergeseran Matriks

Hasil dari perkalian antara matriks deteksi fitur atau *filter* dan matriks pada *input* menjadi nilai dari *feature map*. Berikut perhitungan dari *convolutional layer*:

$$A1 = (182 \times 0) + (114 \times 1) + (103 \times 0) + (185 \times 1) + (117 \times 0) + (106 \times 1) + (184 \times 0) + (113 \times 1) + (106 \times 0)$$

$$= 114 + 185 + 106 + 113 = 518$$

$$A2 = (114 \times 0) + (103 \times 1) + (176 \times 0) + (117 \times 1) + (106 \times 0) + (178 \times 1) + (113 \times 0) + (106 \times 1) + (179 \times 0)$$

$$= 103 + 117 + 178 + 106 = 504$$

$$A3 = (103 \times 0) + (176 \times 1) + (95 \times 0) + (106 \times 1) + (178 \times 0) + (100 \times 1) + (106 \times 0) + (179 \times 1) + (101 \times 0)$$

$$= 176 + 106 + 100 + 179 = 561$$

$$A4 = (185 \times 0) + (117 \times 1) + (106 \times 0) + (184 \times 1) + (113 \times 0) + (106 \times 1) + (183 \times 0) + (117 \times 1) + (105 \times 0)$$

$$= 117 + 184 + 106 + 117 = 524$$

$$A5 = (117 \times 0) + (106 \times 1) + (178 \times 0) + (113 \times 1) + (106 \times 0) + (179 \times 1) + (117 \times 0) + (105 \times 1) + (179 \times 0)$$

$$= 106 + 113 + 179 + 105 = 503$$

$$A6 = (106 \times 0) + (178 \times 1) + (100 \times 0) + (106 \times 1) + (179 \times 0) + (101 \times 1) + (105 \times 0) + (179 \times 1) + (101 \times 0)$$

$$= 178 + 106 + 101 + 179 = 564$$

$$A7 = (184 \times 0) + (113 \times 1) + (106 \times 0) + (183 \times 1) + (117 \times 0) + (105 \times 1) + (183 \times 0) + (116 \times 1) + (107 \times 0)$$

$$= 113 + 183 + 105 + 116 = 517$$

$$A8 = (113 \times 0) + (106 \times 1) + (179 \times 0) + (117 \times 1) + (105 \times 0) + (179 \times 1) + (116 \times 0) + (107 \times 1) + (178 \times 0)$$

$$= 106 + 117 + 179 + 107 = 509$$

$$A9 = (106 \times 0) + (179 \times 1) + (101 \times 0) + (105 \times 1) + (179 \times 0) + (101 \times 1) + (107 \times 0) + (178 \times 1) + (100 \times 0)$$

$$= 179 + 105 + 101 + 178 = 563$$

Dengan perhitungan di atas, maka didapatkan contoh hasil matriks penelitian ini yang sudah diproses pada *convolutional layer* yaitu :

```
518  504  561
[524  503  564]
517   509  563
```

3.2.4 Image Classifier

Pada penelitian ini, proses klasifikasi dilakukan dengan menggunakan algoritma. *Faster Region Convolutional Neural Network* dengan *tools open source* yaitu *google collab* dengan menggunakan *TensorFlow*. Pada proses ini sistem akan mengenali objek dan mengklasifikasikan citra biji jagung ke dalam kategori kualitas biji jagung. Berikut *pseudocode proses* klasifikasi citra gambar berikut pada Gambar 3.15.

```
EarlyStopping(monitor <- 'val_loss',
  mode <- 'min',
  verbose <- 1,
  patience <- 8)

model.fit(data_train,
  batch_size <- 64,
  epochs <- 100,
  validation_data <- data_val,
  callbacks=[es])
```

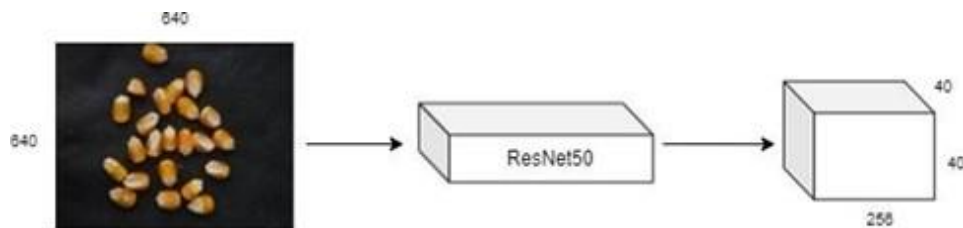
Gambar 3. 15 Pseudocode Proses Klasifikasi

Pseudocode diatas merupakan pemanggilan model untuk *image classification*, *Data_train* dan *validation_data* merupakan perintah untuk melatih data serta

mengevaluasi model. *Batch_size* merupakan penentuan ukuran *batch* data yang digunakan dalam pelatihan model. *Epoch* merupakan jumlah iterasi yang dilakukan dari seluruh data pelatihan. *Model summary* merupakan pemanggilan untuk menampilkan arsitektur dari model yang telah dibuat. *Train_data* dan *validation_data* akan mempelajari gambar dan label yang sesuai dengan gambar sehingga model bisa mengklasifikasikan gambar dengan labelnya.

3.2.4.1 Feature Map

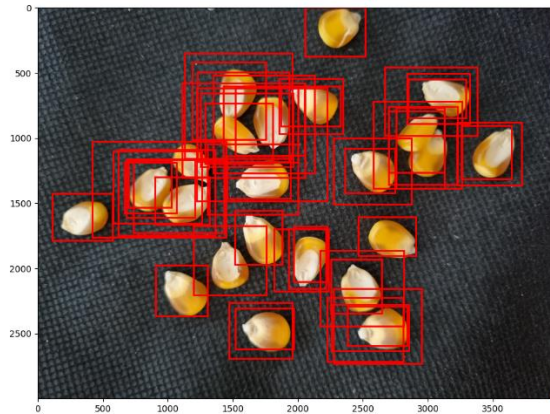
Feature Map merupakan *map* yang dibuat oleh *convolutional layer* yang berisi informasi tentang representasi vektor dari *image*. *Convolution layer* menghasilkan dua *feature map* yang sama. Kemudian *Feature map* diolah di RPN dan menghasilkan *region proposal*, *feature map* lainnya langsung dikirim ke *pooling layer*. Gambar ditampilkan sebagai *height*, *width*, dan *depth* diteruskan ke lapisan tengah menggunakan *CNN* yang sebelumnya sudah dilatih dan berakhir sebagai *convolutional feature map*.



Gambar 3. 16 Feature Map

3.2.4.2 Region Proposal Network (RPN)

Setelah melewati *feature map*, kemudian citra diproses oleh RPN dan dibuat dalam *convolutional layer* untuk memprediksi apa yang dianggap sebagai objek dan membuat prediksi *bounding box* objek. RPN dibagi menjadi 2 *convolutional layer* dimana 1 *layer* bertanggung jawab untuk menemukan letak objek dan 1 *layer* memprediksi *bounding box*. RPN akan menilai objek pada suatu citra dan membuat nilai objek menjadi kotak pembatas *output*.



Gambar 3. 17 Target Region Proposal Network

3.2.4.3 Region of Interest Pooling

Region of Interest (ROI) merupakan wilayah yang direkomendasikan dari gambar asli. Sebelum diklasifikasi, objek harus melewati proses *ROI pooling*. *ROI pooling* menyamakan ukuran dari *feature map* dan *region proposal* yang sudah diolah oleh RPN. *ROI pooling* kemudian informasi *feature map* dan proposal dikirimkan untuk diklasifikasi pada *classification layer*. Pada tahap *fully connected layer* setiap neurons memiliki koneksi penuh ke semua aktivasi dalam lapisan sebelumnya. Hal ini sama persis dengan Multi Layer Perceptron (MLP), yaitu komputasi menggunakan suatu perkalian matriks yang diikuti dengan bias offset. *Fully connected layer* terdiri dari input layer, hidden layer, activation layer, dan output layer. Algoritma yang digunakan pada layer ini yakni feed forward dan backpropagation. Pada tahap feedforward, informasi mengalir dari lapisan input melalui lapisan-lapisan tersembunyi hingga mencapai lapisan output. Setelah output dihasilkan, perbandingan dilakukan dengan target yang diinginkan untuk mengukur seberapa baik jaringan bekerja. Selanjutnya, pada tahap backpropagation, kesalahan ini dikoreksi dengan memperbarui bobot-bobot jaringan melalui optimisasi berbasis gradien. Pada layer ini dibagi menjadi 2 layer, yakni flatten layer dan softmax layer. Flatten layer berfungsi untuk mengubah image value menjadi satu kolom vektor, sementara softmax later berfungsi untuk mengetahui nilai dominan dari masing-masing kelas. Dengan iterasi melalui proses ini, Multi Layer Perceptron (MLP) belajar untuk menyesuaikan bobotnya sehingga dapat menghasilkan output yang semakin akurat sesuai dengan target yang diinginkan.



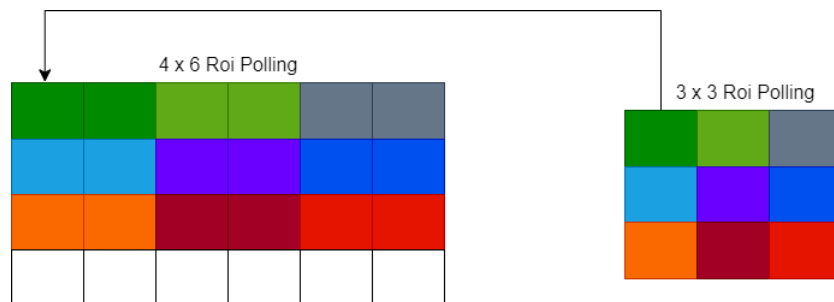
Gambar 3. 18 Target *Region of Interest Pooling*

Pada gambar 3.18 objek yang ditargetkan oleh *region of interest pooling* berukuran 200 x 145 yang akan dibagi dengan 32 (faktor skala). Adapun perhitungannya adalah sebagai berikut.

Lebar : $200/32 = 6,25$ dibulatkan menjadi 6

Tinggi : $145/32 = 4,5$ dibulatkan menjadi 4

Pada contoh target *region of interest pooling* ukuran *RoI* adalah 4 x 6 x 512, sehingga harus diubah menjadi ukuran 3 x 3 x 512. Untuk itu ukuran tersebut harus diubah menjadi ukuran tetap *fully connected layer*. Gambar 3.18 dibawah ini merupakan proses *region of interest pooling*.



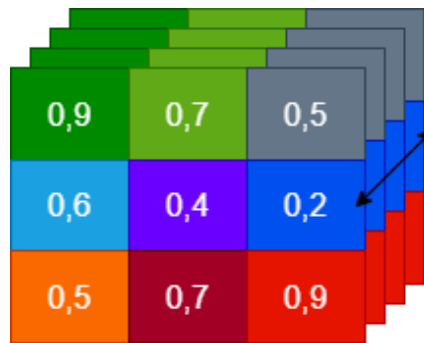
Gambar 3. 19 Proses *Region of Interest Pooling*

Pada gambar 3.19 ukuran RoI yang sebelumnya 4 x 6 akan melakukan tahap pooling hingga ukurannya berubah menjadi 3 x 3. Untuk bisa memenuhi tahap *fully connected layer* memerlukan RoI dengan ukuran 3 x 3. Panjang dan lebar yang tadinya berukuran 6 dan 4 akan dibagi 3, sehingga terdapat nilai Panjang dan lebar 2 dan 1. Setelah proses ini selesai, maka data akan digabungkan menjadi sebuah matriks berukuran 3 x 3 x 512. Dibawah ini merupakan proses penggabungan *data pooling*.



Gambar 3. 20 Proses Penggabungan *Region of Interest Pooling*

Proses ini menunjukkan bahwa nilai yang diambil adalah nilai tertinggi pada setiap matriks, sehingga data berubah menjadi satu ukuran saja. Proses penggabungan data ini dilakukan pada seluruh matriks *region of interest pooling*. Adapun hasil akhir proses penggabungan ini berikut pada gambar di bawah ini.



Gambar 3. 21 Hasil Penggabungan *Region of Interest Pooling*

3.2.5 *Learned Model*

Proses klasifikasi citra yang dilakukan saat *training* data menghasilkan sebuah *learned model* yang kemudian akan digunakan pada aplikasi. *Learned model* atau yang dapat disebut juga dengan *saved model* berisi definisi grafik serta bobot model hasil *data training* di mana terdapat 5 kategori kualitas biji jagung “A”, ”B”, ”C”, ”D”, dan “E”. Selanjutnya, *learned model* ini kemudian di-convert ke dalam *Tensorflow Lite* supaya digunakan pada *android*.

3.2.6 TFLite Model

Learned model yang telah terbentuk kemudian di ubah menjadi *format tensorflowlite* agar bisa dijalankan pada aplikasi berbasis *android*. *Tensorflow Lite* menyediakan *framework* model *Tensorflow* yang sudah dilatih. *Learned model* yang berukuran besar tersebut di-*convert* ke *TFLite* model sehingga ukuran model menjadi ukuran biner yang lebih kecil dan lebih ringan untuk dijalankan pada perangkat berbasis *android*.

3.2.7 Training Data

Training Data merupakan proses pelatihan untuk mempelajari sistem pada saat input data dapat mengklasifikasikan sesuai dengan kategori yang di harapkan menggunakan metode dari *mechine learning*. Model yang sudah di *training* akan digunakan pada proses *testing data*. Proses *training data* menggunakan data *training* dan data *validation*. Proses *training* ini dibagi menjadi 3 data *input*, yaitu data *training*, data *validation*, dan data *testing*.

- *Data Training*

Data Training merupakan data digunakan untuk melatih model.

- *Data Validation*

Data Validation digunakan pada saat mengvalidasi data pada saat pemrosesan pelatihan. *Data validation* juga dapat digunakan untuk memastikan bahwa model tidak *overfitting*.

- *Data Testing*

Data Testing adalah data untuk mengevaluasi kinerja model, dikarenakan data ini belum pernah diproses oleh model.

3.2.8 Output

Setelah melakukan semua proses pada citra, maka hasil *outputnya* menjadi 3 yaitu Kualitas biji jagung yaitu kategori A, kategori B, kategori C, kategori D, dan kategori E.

3.3 Perancangan Antarmuka Sistem

Tahap ini merupakan tahapan yang menunjukkan dan menjelaskan mengenai perancangan antarmuka sistem aplikasi “*Corn Quality*” agar para pengguna mendapatkan gambaran bagaimana mengoperasikan aplikasi ini. Terdapat 3 menu yaitu:

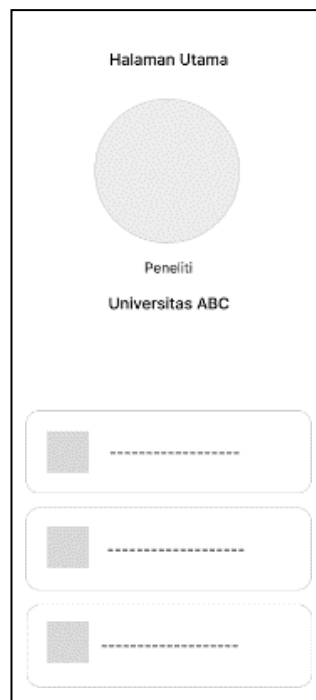
Halaman Utama, *Testing* dan *Result*.

3.3.1 Rancangan *Splashscreen*

Splash Screen adalah tampilan awal yang dilihat oleh pengguna saat membuka aplikasi. Pada tahap ini, akan menampilkan logo aplikasi dan beberapa informasi mengenai aplikasi yang sedang berjalan.

3.3.2 Rancangan Tampilan Halaman Utama

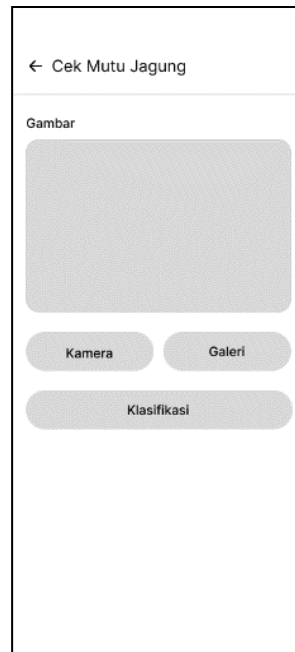
Halaman ini merupakan halaman untuk mengklasifikasikan kualitas jagung dimana nantinya pengguna akan mengambil foto jagung yang akan diklasifikasikan kualitasnya. Halaman utama akan terbuka jika pengguna menekan button “Mulai Klasifikasi” pada halaman *splashscreen*. Pada halaman ini pengguna akan diminta untuk memilih gambar jagung dari galeri atau mengambil langsung dari kamera untuk diklasifikasikan kualitasnya. Rancangan halaman utama terdapat pada gambar 3.22 berikut.



Gambar 3. 22 Rancangan Tampilan Halaman Utama

3.3.3 Rancangan Tampilan Halaman Cek Kualitas Jagung

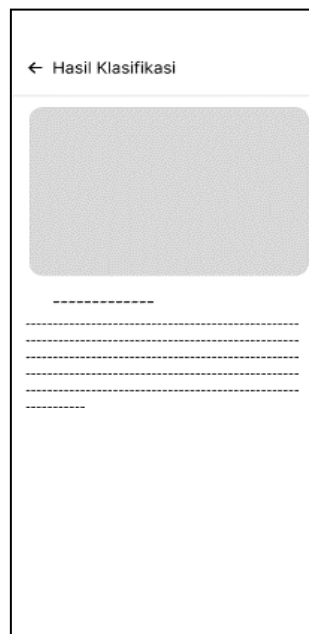
Halaman ini berisikan informasi yang memudahkan pengguna untuk mengetahui tata cara penggunaan aplikasi. Tata cara ini dituliskan dalam bentuk kalimat. Rancangan tampilan tutorial penggunaan aplikasi ini, berikut pada gambar 3.23 berikut.



Gambar 3. 23 Rancangan Tampilan Cek Kualitas Jagung

3.3.4 *Rancangan Tampilan Hasil Klasifikasi*

Halaman ini akan menampilkan informasi kualitas biji jagung dari hasil gambar yang di proses pada halaman utama.

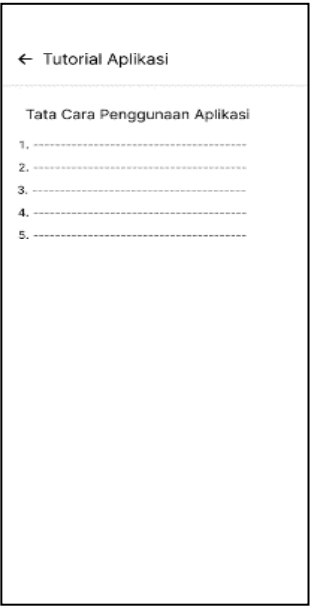


Gambar 3. 24 Rancangan Tampilan Hasil Klasifikasi

3.3.5 *Rancangan Tampilan Tutorial*

Halaman ini berisikan informasi yang memudahkan pengguna untuk mengetahui tata cara penggunaan aplikasi. Tata cara ini dituliskan dalam bentuk kalimat. Rancangan

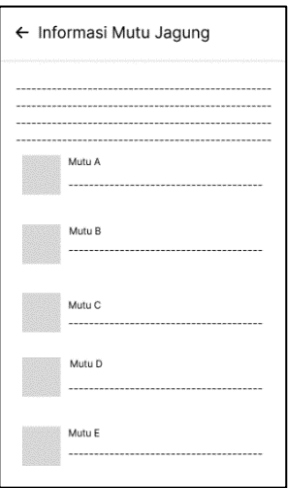
tampilan tutorial penggunaan aplikasi ini, berikut pada gambar 3.25 berikut.



Gambar 3. 25 Rancangan Tampilan Tutorial

3.3.6 Rancangan Tampilan Informasi Kualitas Jagung

Halaman ini berisikan informasi kualitas biji jagung yang akan diklasifikasikan berdasarkan Standar Nasional Indonesia, standardinas pertanian dan dirampungkan menjadi 5 tingkatan. Rancangan halaman Informasi Kualitas Jagung, berikut pada gambar 3.26.



Gambar 3. 26 Rancangan Tampilan Kualitas Biji Jagung

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Klasifikasi kualitas buah jagung menggunakan metode *Faster Region Convolutional Neural Neural Network* dibuat dan diimplementasikan ke dalam sistem dan membutuhkan perangkat keras, perangkat lunak, serta perangkat lain adalah sebagai berikut.

4.1.1 Perangkat Keras dan Perangkat Lunak

Perangkat keras dan perangkat lunak yang penulis pakai pada penelitian ini memiliki dibangun aplikasi dengan spesifikasi :

1. Laptop ASUS Vivo Book X409FJ
2. Processor Core I5
3. Storage SSD 256GB
4. RAM 4 GB
5. Sistem Operasi Windows 11 pro 64-bit
6. *Android Studio*
7. *Google Collab Notebooks*
8. *Tensorflow lite*

4.1.2 Implementasi Data

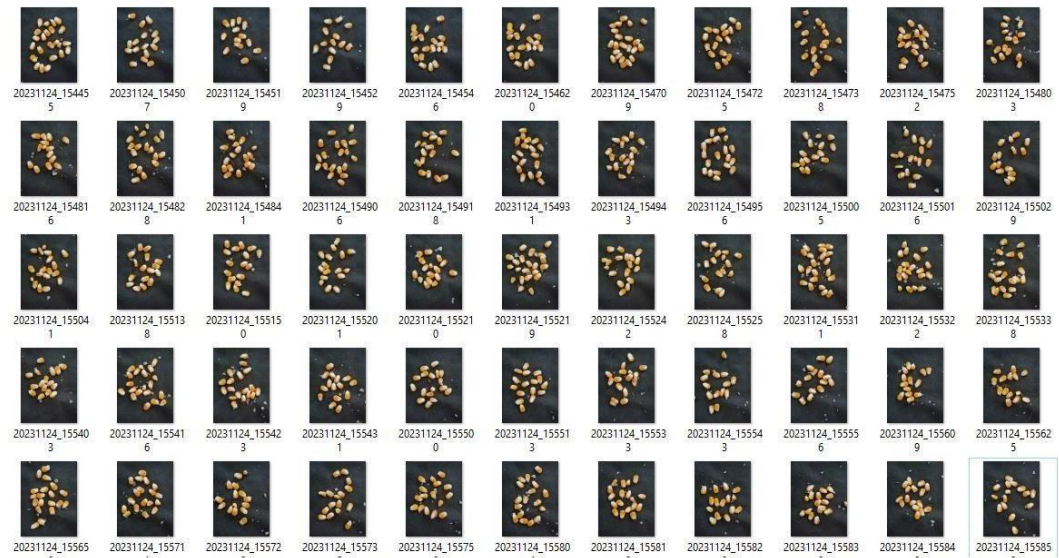
Data yang digunakan pada penelitian ini diambil langsung dari Tanah Karo dengan pengelompokan data berdasarkan kadar air yaitu kualitas A, B, C, D, dan E. Jumlah data yang diambil sebanyak 2.500 data dengan pembagian data 500 citra per kualitas.



Gambar 4. 1 Data Jagung Kualitas A



Gambar 4. 2 Data Jagung Kualitas B



Gambar 4.3 Data Jagung Kualitas C



Gambar 4.4 Data Jagung Kualitas D



Gambar 4. 5 Data Jagung Kualitas E

4.1.3 Implementasi Aplikasi

Sistem diimplementasikan berbasis *mobile* menggunakan Android Studio. Setelah model dilatih, model akan disimpan dalam format *TensorFlow Lite*. Berikut merupakan hasil implementasi desain antarmuka yang telah di rancang pada bab 3, yaitu :

4.1.3.1 Tampilan Halaman Splash Screen

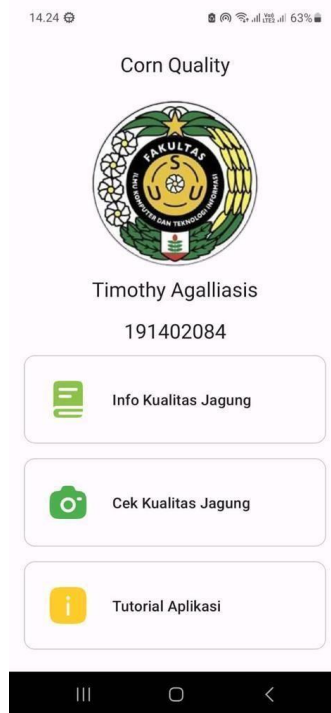
Halaman *splash screen* merupakan tampilan awal yang muncul ketika membuka aplikasi. Halaman *splash screen* pada aplikasi berisi logo aplikasi yang ada ditengah halaman *splash screen*. Halaman *splash screen* berikut pada gambar 4.6.



Gambar 4. 6 Tampilan Halaman Splash Screen

4.1.3.2 Tampilan Halaman Utama

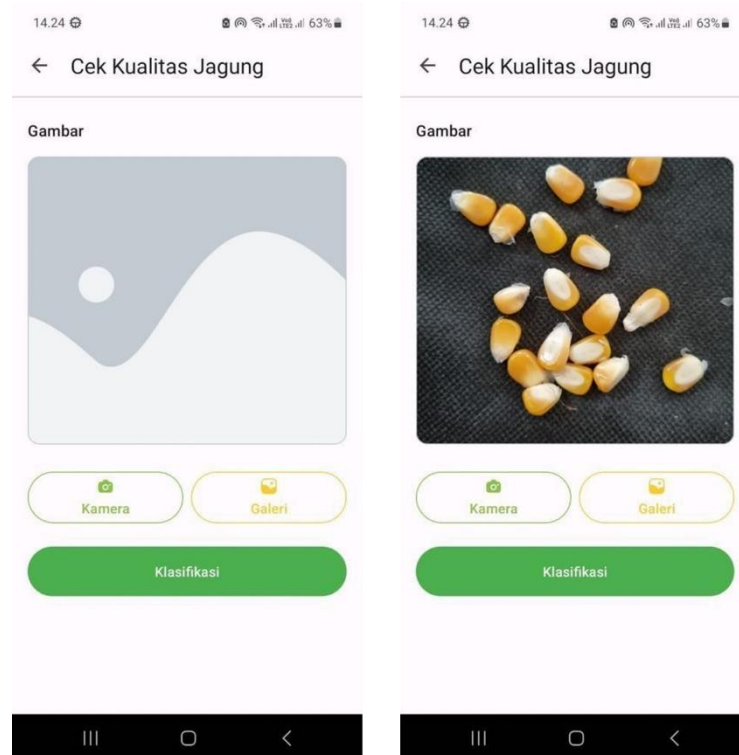
Tampilan halaman utama merupakan tampilan awal ketika aplikasi dibuka. Pada tampilan ini terdapat judul aplikasi (penelitian), logo, data peneliti, instansi peneliti, tombol “Cek Kualitas Jagung” yang diklik akan mengarahkan pengguna ke halaman klasifikasi jagung, tombol “Info Kualitas Jagung” yang jika diklik akan terlihat oleh pengguna halaman informasi kualitas jagung, dan tombol “Tutorial Aplikasi” akan mengarahkan pengguna ke halaman tata cara penggunaan aplikasi.



Gambar 4. 7 Tampilan Halaman Utama

4.1.3.3 Tampilan Halaman Testing

Tampilan halaman *testing* merupakan tampilan yang terbuka ketika pengguna mengklik tombol “Cek Kualitas Jagung”. Pada *testing* terdapat *button* untuk “Klasifikasi” agar kemudian diklasifikasikan, dan 2 *button* untuk memilih metode pengambilan gambar. Untuk pengambilan gambar dari galeri, terdapat *button* “Galeri” dan untuk pengambilan gambar dari kamera terdapat *button* “Kamera”. Setelah gambar dipilih, maka gambar akan muncul pada halaman ini di bagian form untuk menampilkan *preview* gambar. Desain halaman *testing* setelah *image* diinput. Berikut Gambar 4.8.



Gambar 4. 8 Tampilan Halaman Testing

4.1.3.4 Tampilan Halaman Hasil

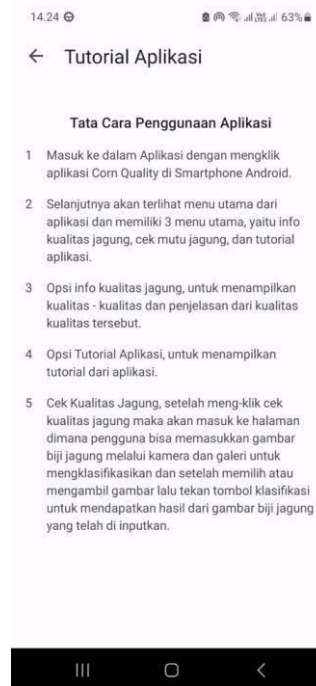
Tampilan *result* merupakan tampilan ketika pengguna mengambil foto dan melanjutkan ke tahap selanjutnya. Tampilan ini menampilkan hasil klasifikasi dari hasil foto yang diambil beserta keterangan klasifikasinya.



Gambar 4. 9 Tampilan Halaman Hasil

4.1.3.5 Tampilan Halaman Tutorial Aplikasi

Halaman informasi penggunaan aplikasi berisi tata cara penggunaan aplikasi yang ditulis ke dalam kalimat. Halaman ini didapat setelah pengguna mengklik tombol “Tutorial Aplikasi” halaman utama. Tampilan halaman tutorial aplikasi berikut pada Gambar 4.10.



Gambar 4. 10 Tampilan Halaman Tutorial Aplikasi

4.1.3.6 Tampilan Halaman Informasi Kualitas Biji Jagung

Halaman informasi tingkat kematangan ini berisi informasi mengenai buah kualitas biji jagung. Halaman ini didapat setelah pengguna mengklik tombol “Informasi Kualitas Biji Jagung” pada halaman utama. Tampilan halaman informasi penggunaan aplikasi terdapat pada gambar 4.11.



Gambar 4. 11 Tampilan Halaman Informasi Kualitas Jagung

4.2 Prosedur Operasional

Prosedur operasional membahas tentang bagaimana *user* mengoperasikan aplikasi yang sudah dibuat. Pertama ketika pengguna mengklik icon aplikasi, akan muncul halaman *splash screen* yang menampilkan logo dan nama aplikasi. Setelah itu akan langsung memasuki halaman utama yang memiliki tiga *button*. Pada halaman ini pengguna dapat memilih tombol tutorial aplikasi untuk dapat mengetahui cara menggunakan aplikasi. Selain itu, tombol info kualitas jagung berisikan informasi mengenai kualitas jagung yang dapat di klasifikasikan di aplikasi ini. Tombol “Cek Kualitas Jagung” merupakan halaman untuk mengklasifikasikan kualitas jagung. Apabila pengguna sudah memiliki gambar, maka tinggal menekan tombol galeri. Lalu dapat mengklasifikasikan kualitas jagung. Selain itu, pengguna juga dapat menggunakan tombol kamera untuk mengklasifikasikan gambar jagung yang di ambil secara langsung. Setelah gambar jagung sudah di masukkan, pengguna dapat langsung mengklasifikasikan jagung.

4.3 Pelatihan Sistem

Pelatihan sistem pada penelitian ini menggunakan data sebanyak 7500 data jagung yang terdiri dari 5 kelas setelah melewati 3 proses augmentasi. Pelatihan sistem ini dilakukan sebanyak 4 kali proses pelatihan sistem dengan jumlah *epoch* dan *batchsize*

yang berbeda. Pelatihan sistem pertama menggunakan *batchsize* 16 dengan epoch 30 mendapatkan akurasi sebesar 95% pada epoch yang ke 16. Setelah itu dilakukan pelatihan dengan *batchsize* 32 dengan epoch 100 mendapatkan akurasi sebesar 97% pada epoch yang ke 12. Setelah itu dilakukan pelatihan dengan *batchsize* 64 dengan epoch 60 mendapatkan akurasi sebesar 97% pada epoch ke 12. Setelah itu dilakukan pelatihan sistem dengan *batchsize* 8 dengan epoch 100 didapatkanlah hasil yang lebih tinggi yaitu sebesar 98% pada epoch ke 19. Hal ini didukung dengan penelitian yang dilakukan oleh (Pembudi, Rio et al, 2023) menyatakan bahwa nilai *epoch* sangat mempengaruhi akurasi. Selain itu (Alvera Kusuma, 2022) menyatakan bahwa semakin besar nilai *epoch* semakin besar tingkat akurasi. Pada penelitian ini pelatihan menggunakan *early stopping* untuk mencegah terjadinya overfitting maka penulis hanya menampilkan sampai epoch ke 19. Adapun hasil pengujian yang didapatkan dengan menggunakan *batchsize* 8 terdapat pada tabel 4.1 berikut.

Tabel 4. 1 Tabel Pelatihan Sistem

Epoch	Loss	Accuracy	Val_Loss	Val_Accuracy
1	1.1402	0.4834	0.5359	0,7609
2	0.6918	0.6661	0.3813	0.8062
3	0.4974	0.7678	0.2956	0.8719
4	0.3684	0.8412	0.2300	0.9094
5	0.2563	0.8942	0.1365	0.9500
6	0.1778	0.9271	0.1157	0.9484
7	0.1229	0.9519	0.1102	0.9625
8	0.0857	0.9662	0.1005	0.9703
9	0.0868	0.9642	0.0837	0.9766
10	0.0703	0.9728	0.1650	0.9468
11	0.0671	0.9737	0.0732	0.9766
12	0.0545	0.9781	0.1005	0.9812
13	0.0582	0.9764	0.2313	0.9656
14	0.0752	0.9706	0.1026	0.9703
15	0.0719	0.9752	0.1130	0.9719
16	0.0370	0.9242	0.0995	0.9750
17	0.0317	0.9856	0.1863	0.9641
18	0.0398	0.9844	0.1034	0.9734
19	0.0404	0.9868	0.0811	0.9812




Tabel 4.1 yang merupakan hasil uji model yang menggunakan *batchsize* 8 dengan epoch sebanyak 100 dengan menggunakan *optimizer* adam, pelatihan model menggunakan *initializer uniform* dan juga *activation ReLu* guna mendapatkan nilai terbaik dalam menentukan *Loss*, *Accuracy*, *Val_Loss*, dan *Val_Accuracy*. Selain itu,






pelatihan ini menggunakan *early stopping* untuk menghentikan pelatihan pada titik di mana model memiliki performa terbaik pada dataset validasi, *early stopping* membantu model untuk lebih baik dalam melakukan generalisasi pada data baru.





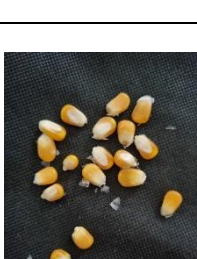
4.4 Pengujian Sistem






Pengujian sistem menggunakan data sebanyak 250 data jagung, yang merupakan 10% dari jumlah data total. Tahap ini dilakukan untuk melihat kinerja dari algoritma *Faster R-CNN* yang sebelumnya sudah dilatih dalam mengklasifikasikan kualitas jagung pada 5 tingkatan kualitas. Pengujian dilakukan dengan membandingkan keakuratan sistem dengan alat Grain Moisture Meter sebagai acuan dalam menentukan kualitas biji jagung. Adapun kualitas yang dapat diklasifikasikan yaitu kualitas A, B, C, D dan E. Tabel berikut akan menampilkan hasil uji klasifikasi data pada sistem.



Tabel 4. 2 Hasil Pengujian Sistem

No.	Gambar	Grain Moisture Meter	Sistem	Status
1.		A	A	BENAR
2.		A	A	BENAR
3.		D	D	BENAR

No.	Gambar	Grain Moisture Meter	Sistem	Status
4.		B	B	BENAR
5.		E	D	SALAH
6.		C	C	BENAR
7.		A	A	BENAR
8.		C	C	BENAR

No.	Gambar	Grain Moisture Meter	Sistem	Status
9.		D	C	SALAH
10.		D	D	BENAR
11.		B	B	BENAR
12.		E	E	BENAR
13.		C	D	SALAH

No.	Gambar	Grain Moisture Meter	Sistem	Status
14.		E	E	BENAR
15.		B	B	BENAR
16.		E	E	BENAR
17.		A	A	BENAR
18.		D	C	SALAH

No.	Gambar	Grain Moisture Meter	Sistem	Status
19.		B	B	BENAR
20		C	D	SALAH

Pada tabel di atas, terdapat 20 contoh pengujian pada sistem dari keseluruhan hasil uji. Pengujian gambar jagung berikut di lampiran 1 penelitian. Setelah dilakukannya proses pengujian pada sistem maka dilakukan proses evaluasi pada model menggunakan *Confussion Matrix* berikut pada tabel 4.3 berikut.

Tabel 4. 3 Hasil Evaluasi Model Menggunakan Confussion Matrix

		Aktual					Total
		Jagung Kualitas A	Jagung Kualitas B	Jagung Kualitas C	Jagung Kualitas D	Jagung Kualitas E	
Prediksi	Jagung Kualitas A	50	0	0	0	0	50
	Jagung Kualitas B	1	48	1	0	0	50
	Jagung Kualitas C	0	0	46	4	0	50
	Jagung Kualitas D	0	0	4	46	0	50
	Jagung Kualitas E	0	0	1	1	48	50
Total		51	48	52	51	48	250

Data uji pada tabel 4.3 diatas menunjukkan hasil evaluasi *confussion matrix* menampilkan 250 citra. Citra pada setiap kelas dibagi menjadi 50 citra. Pada kelas

jagung kualitas A tidak didapati kesalahan pada sistem. Pengujian sistem pada jagung kualitas B terdapat kesalahan sebanyak dua kali. Selanjutnya pada pengujian jagung kualitas C dan D terdapat 4 kali kesalahan pengujian sistem dikarenakan pencahayaan, adanya kemiripan citra pada mutu C dan D. Selanjutnya pada pengujian klasifikasi mutu jagung dengan mutu E terdapat kesalahan sebanyak 2 kali dikarenakan fokus pada pengambilan citra.

Setelah dilakukannya evaluasi *confussion matrix* dihasilkanlah nilai *true positive* (TP), *false positive* (FP), dan *false negative* (FN). Tabel nilai ini berikut pada tabel 4.7 berikut ini.

Tabel 4. 4 TP, FP, FN

	True Positive (TP)	False Positive (FP)	False Negatif (FN)
Kualitas A	50	0	1
Kualitas B	48	2	0
Kualitas C	46	4	6
Kualitas D	46	4	5
Kualitas E	48	2	0
Total	238	12	12

Tabel di atas menampilkan 238 data yang masuk ke dalam nilai *true positif* yang berarti sistem tersebut dapat mengklasifikasikan kualitas jagung dengan benar sebanyak 238 kali. Terdapat pula 12 data pada *false positif* yang berarti sistem melakukan prediksi benar padahal data adalah data yang salah. *False negative* terdapat 12 data yang artinya sistem menghasilkan data salah padahal sebenarnya data tersebut adalah data benar.

Setelah nilai tersebut didapatkan, dihasilkanlah nilai *precision*, *recall*, dan *F1-Score*. Perhitungan nilai dari tingkatan kualitas adalah sebagai berikut.

1. Precision

Rumus perhitungan *Precision*. Berikut Persamaan 4.1

$$Precision \text{ Kualitas A} = \frac{50}{50} \times 100\% = 100\%$$

$$\text{Precision Kualitas A} = \frac{48}{50} \times 100\% = 96,66\%$$

$$\text{Precision Kualitas C} = \frac{46}{50} \times 100\% = 92\%$$

$$\text{Precision Kualitas D} = \frac{46}{50} \times 100\% = 92\%$$

$$\text{Precision Kualitas E} = \frac{48}{50} \times 100\% = 96,66\%$$

2. Recall

Rumus perhitungan *Recall*. Berikut Persamaan 4.2

$$\text{Recall Kualitas A} = \frac{50}{51} \times 100\% = 96,66\%$$

$$\text{Recall Kualitas B} = \frac{48}{48} \times 100\% = 100\%$$

$$\text{Recall Kualitas C} = \frac{46}{52} \times 100\% = 88,46\%$$

$$\text{Recall Kualitas D} = \frac{46}{51} \times 100\% = 90,19\%$$

$$\text{Recall Kualitas E} = \frac{48}{50} \times 100\% = 96\%$$

3. F1 – Score

Rumus perhitungan *F1 - Score*. Berikut Persamaan 4.3

$$\begin{aligned} \text{F1-Score Kualitas A} &= 2 \times \frac{(1 \times 0.96)}{(1 + 0.96)} \times 100\% \\ &= 2 \times \frac{(0,96)}{(1.96)} \times 100\% \\ &= 97\% \end{aligned}$$

$$\begin{aligned} \text{F1-Score Kualitas B} &= 2 \times \frac{(1 \times 0.96)}{(1 + 0.96)} \times 100\% \\ &= 2 \times \frac{(0.96)}{(1.96)} \times 100\% \\ &= 97\% \end{aligned}$$

$$\text{F1-Score Kualitas C} = 2 \times \frac{(0.92 \times 0.88)}{(0.92 + 0.88)} \times 100\%$$

$$= 2 \times \frac{(0.8096)}{(1.8)} \times 100\%$$

$$= 89,95\%$$

$$F1\text{-Score Kualitas D} = 2 \times \frac{(0.92 \times 0.90)}{(0.92 + 0.90)} \times 100\%$$

$$= 2 \times \frac{(0.8288)}{(1.82)} \times 100\%$$

$$= 91,07\%$$

$$F1\text{-Score Kualitas E} = 2 \times \frac{(0.96 \times 0.96)}{(0.96 + 0.96)} \times 100\%$$

$$= 2 \times \frac{(0.9216)}{(1.92)} \times 100\%$$

$$= 96\%$$

Dari perhitungan di atas dapat ditampilkan nilai-nilai di dalam tabel berikut.

Tabel 4. 5 Nilai Precision, Recall, dan F1-Score

	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Kualitas A	100%	96,66%	97%
Kualitas B	96,66%	100%	97%
Kualitas C	92%	88,46%	89,95%
Kualitas D	92%	90,19%	91,07%
Kualitas E	96,66%	96%	96%

Berdasarkan uji sistem yang dilakukan pada aplikasi klasifikasi mutu jagung menggunakan metode *Faster Region Convolutional Neural Network*, dihasilkan nilai akurasi sistem melalui perhitungan berikut.

$$Accuracy = \frac{N_{Benar}}{N} \times 100\%$$

$$Accuracy = \frac{238}{250} \times 100\%$$

$$= 95,2\%$$

Berdasarkan perhitungan yang telah dilakukan di atas penelitian ini mendapatkan nilai akurasi sebesar 95,2%. Adapun beberapa faktor kesalahan pada sistem aplikasi pada penelitian ini yaitu sebagai berikut :

1. Pencahayaan merupakan faktor yang mempengaruhi kesalahan sistem dalam mengklasifikasikan jagung. Apabila pengambilan gambar dengan intensitas cahaya yang terlalu terang, seperti di bawah sinar matahari, maka sistem tidak dapat mengklasifikasikan jagung dengan baik.
2. Jarak pengambilan gambar sebaiknya 10 cm. Pengambilan gambar dengan jarak yang terlalu jauh maupun terlalu dekat dapat memengaruhi kualitas hasil foto dan akurasi sistem klasifikasi. Dengan menjaga jarak yang optimal, informasi yang diperoleh dapat lebih tepat dan memberikan kontribusi positif pada proses pengklasifikasian jagung.
3. Kamera yang tidak fokus, menyebabkan gambar menjadi blur atau kabur juga merupakan salah satu faktor yang mempengaruhi kesalahan sistem dalam mengklasifikasikan jagung, hal ini mengakibatkan kemampuan sistem untuk melakukan klasifikasi mutu jagung menjadi tidak akurat.

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang sudah dilakukan didapatkan kesimpulan sebagai berikut.

1. Penelitian ini bertujuan untuk mengklasifikasikan kualitas biji jagung dengan menggunakan algoritma Faster R-CNN dengan pre-trained model ResNet-50 berdasarkan kadar air yang dilihat dari warna dan tekstur dari biji jagung menggunakan data latih sebanyak 2000 data dan data uji sebanyak 250 data dan mendapatkan tingkat akurasi sebesar 95,2 %.
2. Penelitian ini memiliki tujuan untuk mengurangi perbedaan pendapat pada saat penilaian kualitas biji jagung dan mengurangi terjadinya subjectivitas dalam penilaian kualitas biji jagung.
3. Mempermudah dalam penentuan kualitas biji jagung karena menggunakan aplikasi di dalam *smartphone*
4. *Parameter batch size, epoch*, sangat mempengaruhi nilai akurasi pada model yang dilatih. Ukuran *batch size* yang lebih kecil dari ukuran *epoch* pada penelitian ini meningkatkan akurasi.
5. Penyebab kesalahan sistem dalam mengklasifikasikan kualitas biji jagung dipengaruhi oleh fokus kamera, pencahayaan, jarak pengambilan sampel serta posisi sampel.

5.2 Saran

Penulis menyadari bahwa penelitian ini masih perlu dikembangkan. Untuk itu adapun beberapa saran guna meningkatkan penelitian selanjutnya yaitu:

1. Diharapkan untuk membuat sebuah sistem dengan varietas jagung yang berbeda selain jenis *pionner*.
2. Diharapkan untuk dapat menambahkan variasi data training serta jumlah data yang

lebih banyak guna menambahkan ciri dan perbedaan yang lebih bervariasi.

3. Diharapkan pada penelitian berikutnya menggunakan algoritma selain Faster R-CNN dalam proses klasifikasi biji jagung, sehingga nilai akurasi klasifikasi dengan algoritma baru dapat dibandingkan.
4. Diharapkan penelitian kedepannya bisa membangun penentuan kualitas biji jagung dengan pendeteksian biji jagung dengan menggunakan aplikasi secara *realtime*.
5. Diharapkan kedepannya terdapat sistem yang dapat mengklasifikasikan kualitas biji jagung tanpa memiliki kesalahan pada pencahayaan dan jarak pengambilan gambar.

DAFTAR PUSTAKA

- Adrizal, Anggraini D., N. Novita, Santosa, Andasuryani. 2011. Pendugaan Kualitas Fisik Biji Jagung untuk Bahan Pakan Menggunakan Jaringan Syaraf Tiruan Berdasarkan Data Citra Digital. *Jurnal Peternakan Indonesia* 13(3): 183-190
- Adrizal1 , D. Anggraini , N. Novita , Santosa , Andasuryani. Pendugaan Kualitas Fisik Biji Jagung untuk Bahan Pakan Menggunakan Jaringan Syaraf Tiruan Berdasarkan Data Citra Digital. 2011
- Anggraini, 2008. Pendugaan Mutu Fisik Jagung dengan Jaringan Saraf Tiruan menggunakan Parameter Warna Pada Pengolahan Citra Digital. *Jurnal Volume ke-1. Padang (ID) : UNAND Padang*.
- Bilaut, CA. 2015. penentuan kualitas biji jagung untuk pembenihan berdasarkan tingkat kecerahan warna menggunakan support vector machine. Skripsi. Kupang (ID) : Stikom Uyelindo Kupang.
- Bonita Indah Nuryanti, Dr. Ir. Nursigit Bintoro, M. Sc.. Analisis Matematis Perubahan Kualitas Sifat Fisik Biji Jagung Pada Beberapa Cara Penyimpanan Dengan Kadar Air Bervariasi. 2014
- BSN. 2013. SNI 4483-1998 Tentang Jagung Bahan Pakan Ternak. Badan Standarisasi Nasional. Jakarta
- Bustomi. Analisis Distribusi Intensitas RGB Citra Digital untuk Klasifikasi Kualitas Biji Jagung menggunakan Jaringan Syaraf Tiruan. 2014
- Cahyono, R. W., & Sari, R. F. (2016). Klasifikasi Kualitas Biji Jagung Menggunakan Jaringan Syaraf Tiruan Backpropagation. *Jurnal Ilmiah Teknologi Informasi Asia*, 10(2), 93-100.
- Debby Permatasari. Sistem Klasifikasi Kualitas Biji Jagung Berdasarkan Tekstur Berbasis Pengolahan Citra Digital. 2012
- Isfarayani, E., & Supriana, T. (2018). Klasifikasi Kualitas Jagung dengan Metode K-Nearest Neighbor (KNN) Berdasarkan Fitur Citra. *Jurnal Rekayasa Elektrika*, 15(2), 82-87.
- Kementrian Perdagangan RI. 2017. Peraturan Menteri Perdagangan Republik Indonesia No.27/MDAG/PER/5/2017 tentang Penetapan Harga Acuan

- Pembelian Jagung di Tingkat Petani. Menteri Perdagangan RI. Jakarta
- Maulidiansyah. Isnaini Abdillah, Klasifikasi Jenis Jagung Berdasarkan Bentuk Biji Menggunakan Metode You Only Look Once (YOLO). 2023
- Muh. Subhan, Hasan Basri. 2019. Klasifikasi Mutu Buah Pala (*Myristica Fragrans* Houtt) Berbasis Pengolahan Citra Menggunakan Metode Deep Learning Arsitektur Faster R-CNN. INTEK Jurnal Penelitian. 2019, Volume 6 (2): 106-113
- Muhammad Fajar Al Hanief. Klasifikasi Kualitas Biji Jagung Berdasarkan Warna dengan Menggunakan Pengolahan Citra Digital. 2020
- Mulyanto, Ferdinan. 2020. Aplikasi Pendeteksian Objek Buah-Buahan yang Memiliki Kemiripan Menggunakan Algoritma *Faster R-CNN* Berbasis Android. Skripsi. Medan. Universitas Sumatera Utara
- Munarto, Ri, Endi P., Rensi S. 2014. Klasifikasi Kualitas Biji Jagung Manis Berdasarkan Fitur Warna Menggunakan Fuzzy Logic. Simposium Nasional RAPI XIII : 5-12
- Ren, S., He, K., Girshick, R., & Sun, J. (2017). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 1137-1149.
- Sari, N. I., & Tirtadji, P. (2017). Pengembangan Sistem Inspeksi Nondestruktif untuk Menentukan Kualitas Biji Jagung. *Jurnal Teknologi Pertanian*, 18(3), 167-174.
- Zahroh, F., Suryaningrat, W., & Fanany, M. I. (2019). Klasifikasi Kualitas Biji Jagung Menggunakan Metode K-Means Clustering dan Naive Bayes Classifier. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 6(2), 183-190.
- Jihad Ardiansyah, Rita Purnamasari, Bambang Hidayat. 2020. Sistem Klasifikasi Kualitas Biji Jagung Berdasarkan Tekstur Dengan Metode Discret Wavelet Transform Dan Klasifikasi Support Vector Machine Berbasis Pengolahan Citra Digital.
- Debby Permatasari. 2012. Sistem Klasifikasi Kualitas Biji Jagung Berdasarkan Tekstur Berbasis Pengolahan Citra Digital.
- Muhammad Fajar Al Hanief. 2021. Klasifikasi Tingkat Kematangan Buah Kopi Menggunakan Algoritma Fuzzy C- Means.



**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI**
UNIVERSITAS SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

Jalan Universitas No. 9A Gedung A, Kampus USU Medan 20155, Telepon: (061) 821007
Laman: <http://Fasilkomti.usu.ac.id>

KEPUTUSAN
DEKAN FAKULTAS ILMU KOMPUTER
DAN TEKNOLOGI INFORMASI
NOMOR : 1934/UN5.2.14.D/SK/SPB/2024
DEKAN FAKULTAS ILMU KOMPUTER
DAN TEKNOLOGI INFORMASI UNIVERSITAS SUMATERA UTARA

- Membaca** : Surat Permohonan Mahasiswa Fasilkom-TI USU tanggal 13 Juni 2024 perihal permohonan ujian skripsi:
Nama : TIMOTHY AGALLIASIS
NIM : 191402084
Program Studi : Sarjana (S-1) Teknologi Informasi
Judul Skripsi : Klasifikasi Kualitas Biji Jagung Menggunakan Metode Faster Region Convolutional Neural Network
- Memperhatikan** : Bahwa Mahasiswa tersebut telah memenuhi kewajiban untuk ikut dalam pelaksanaan Meja Hijau Skripsi Mahasiswa pada Program Studi Sarjana (S-1) Teknologi Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara TA 2023/2024.
- Menimbang** : Bahwa permohonan tersebut diatas dapat disetujui dan perlu ditetapkan dengan surat keputusan
- Mengingat** : 1. Undang-undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional.
2. Peraturan Pemerintah Nomor 17 tahun 2010 tentang pengelolaan dan penyelenggara pendidikan.
3. Keputusan Rektor USU Nomor 03/UN5.1.R/SK/SPB/2021 tentang Peraturan Akademik Program Sarjana Universitas Sumatera Utara.
4. Surat Keputusan Rektor USU Nomor 1876/UN5.1.R/SK/SDM/2021 tentang pengangkatan Dekan Fasilkom-TI USU Periode 2021-2026
- MEMUTUSKAN**
- Menetapkan** :
Pertama : Membentuk dan mengangkat Tim Penguji Skripsi mahasiswa sebagai berikut:
Ketua : Dedy Arisandi ST., M.Kom.
NIP: 197908312009121002
Sekretaris : Fanindia Purnamasari S.TI,M.IT
NIP: 198908172019032023
Anggota Penguji : Dr. Muhammad Anggia Muchtar ST., MMIT.
NIP: 198001102008011010
Anggota Penguji : Umayya Ramadhani Putri Nasution S.TI, M.Kom.
NIP: 199104112021022001
Moderator : -
Panitera : -
- Kedua** : Segala biaya yang diperlukan untuk pelaksanaan kegiatan ini dibebankan pada Dana Penerimaan Bukan Pajak (PNPB) Fasilkom-TI USU Tahun 2024.
- Ketiga** : Keputusan ini berlaku sejak tanggal ditetapkan dengan ketentuan bahwa segala sesuatunya akan diperbaiki sebagaimana mestinya apabila dikemudian hari terdapat kekeliruan dalam surat keputusan ini.

Tenbusan :
1. Ketua Program Studi Sarjana (S-1) Teknologi Informasi
2. Yang bersangkutan
3. Arsip

Medan, 14 Juni 2024
Ditandatangani secara elektronik oleh:
Dekan



Maya Silvi Lydia
NIP 197401272002122001