

**KLASIFIKASI CITRA REMPAH-REMPAH DENGAN
MENGUNAKAN ALGORITMA *CONVOLUTIONAL NEURAL
NETWORK* (CNN)**

SKRIPSI

OLEH:

**LOUDJI ZAMICO PRATAMA
178160010**



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MEDAN AREA
2023**

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah
3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area

Document Accepted 13/7/23

Access From (repository.uma.ac.id)13/7/23

**KLASIFIKASI CITRA REMPAH-REMPAH DENGAN
MENGUNAKAN ALGORITMA *CONVOLUTIONAL NEURAL
NETWORK* (CNN)**

SKRIPSI

Diajukan sebagai Salah Satu Syarat untuk Memperoleh
Gelar Sarjana di Fakultas Teknik
Universitas Medan Area

Oleh:

**LOUDJI ZAMICO PRATAMA
178160010**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MEDAN AREA MEDAN
2023**

LEMBAR PENGESAHAN

Judul Skripsi : Klasifikasi Citra Rempah-Rempah dengan Menggunakan Algoritma *Convolutional Neural Network* (CNN)


Nama : Loudji Zamico Pratama

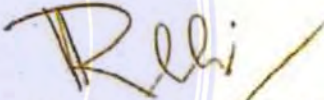
NPM : 178160010

Fakultas : Teknik


Prodi : Informatika

Disetujui oleh
Komisi Pembimbing


Muhathir, S.T, M.Kom
Pembimbing I


Rizki Muliono, S.Kom, M.Kom
Pembimbing II


Dr. Rahmat Syah, S.Kom, M.Kom
Dekan Fakultas Teknik


Rizki Muliono, S.Kom, M.Kom
Ka. Prodi

Tanggal Lulus : 4 Mei 2023

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah
3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area

Document Accepted 13/7/23

Access From (repository.uma.ac.id)13/7/23

HALAMAN PERNYATAAN

Saya menyatakan bahwa skripsi yang saya susun, sebagai syarat memperoleh gelar sarjana merupakan hasil karya tulis saya sendiri. Adapun bagian-bagian tertentu dalam penulisan skripsi ini yang saya kutip dari hasil karya orang lain telah dituliskan sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan ilmiah.

Saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi-sanksi lainnya dengan peraturan yang berlaku, apabila di kemudian hari ditemukan adanya plagiat dalam skripsi ini.



HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR/SKRIPSI/TESIS UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Medan Area, saya yang bertanda tangan di bawah ini:

Nama : Loudji Zamico Pratama

NPM : 178160010

Program Studi : Teknik Informatika

Fakultas : Teknik

Jenis karya : Tugas Akhir/Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Medan Area **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

Klasifikasi Citra Rempah-Rempah Dengan Menggunakan Algoritma *Convolutional Neural Netwrok* (CNN).

Dengan Hak Bebas Royalti Noneksklusif ini Universitas Medan Area berhak menyimpan, mengalihmedia/format-kan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir/skripsi/tesis saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Medan

Pada tanggal : 4 Mei 2023

Yang menyatakan



(Loudji Zamico Pratama)

ABSTRAK

Indonesia merupakan salah satu negara beriklim tropis yang memiliki kekayaan alam yang sangat besar kedua di dunia setelah Brazil. Salah satu kekayaan alam terkenal yang dimiliki oleh Indonesia berasal dari sektor agraris atau pertanian yaitu berupa kekayaan rempah-rempah. Rempah-rempah merupakan jenis tumbuhan yang sangat mudah ditemukan di wilayah Indonesia dan manfaatnya sangat banyak digunakan oleh masyarakat. Pada Penelitian ini dilakukan uji coba pada rempah-rempah dengan menggunakan Algoritma *Convolutional Neural Network* untuk mempermudah masyarakat dalam mengenali atau membedakan jenis rempah-rempah terutama pada rempah-rempah yang memiliki kemiripan baik dari segi bentuk dan warna. *Convolutional Neural Network* adalah jenis dari algoritma *deep learning* yang merupakan pengembangan dari *Multilayer Perceptron*. CNN sering digunakan untuk menyelesaikan permasalahan *computer vision* terkhususnya pada identifikasi citra dan pengolahan citra. Pada penelitian ini hasil pengujian yang dilakukan pada rempah-rempah menunjukkan bahwa Model 12 dengan menggunakan CNN pada penelitian ini menjadi model terbaik dengan tingkat akurasi mencapai 100%, *Precision* 100%, *Recall* 100%, dan *F1-Score* 100% dalam klasifikasi 10 jenis rempah-rempah.

Kata Kunci: Rempah-rempah; *Convolutional Neural Network*; Kekayaan Alam.

ABSTRACT

Indonesia is a tropical country which has the second largest natural wealth in the world after Brazil. One of the well-known natural resources owned by Indonesia comes from the agricultural or agricultural sector, namely in the form of a wealth of spices. Spices are a type of plant that is very easy to find in Indonesia and its benefits are widely used by the community. In this research, experiments were carried out on spices using the Convolutional Neural Network Algorithm to make it easier for the public to recognize or differentiate types of spices, especially spices that have similarities both in terms of shape and color. CNN is a type of deep learning algorithm which is a development of the Multilayer Perceptron. CNN is often used to solve computer vision problems, especially in image identification and image processing. In this study, the results of tests conducted on spices showed that Model 12 using CNN in this study was the best model with an accuracy rate of 100%, Precision 100%, Recall 100%, and F1-Score 100% in the classification of 10 types of spices.

Keywords: Spices; Convolutional Neural Networks; Natural wealth.

RIWAYAT HIDUP

Penulis dilahirkan di Pariaman Pada tanggal 15 Januari 1999 dari ayah Berlizami dan ibu Hermawati. Penulis merupakan putra pertama dari tiga bersaudara. Tahun 2017 Penulis lulus dari YPT Teladan Medan dan pada tahun 2017 terdaftar sebagai mahasiswa Fakultas Teknik Universitas Medan Area. Pada tahun 2020, Penulis telah selesai melaksanakan praktek kerja lapangan (PKL) di PT. Kawasan Industri Medan (Persero).



KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Allah Subhanahu Wa Ta'ala, Allah Yang Maha Pengasih lagi Maha Penyayang sehingga penulis bisa menyelesaikan skripsi dengan judul **“Klasifikasi Citra Rempah-Rempah Dengan Menggunakan Algoritma Convolutional Neural Network (CNN)”**.

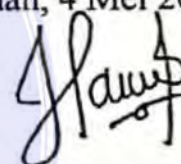
Dalam penyelesaian penulisan tugas akhir/skripsi ini, penulis menyadari bahwa masih banyak keterbatasan penulis dalam menyusun dan menyelesaikan tugas akhir/skripsi ini. Skripsi ini tidak dapat tersusun dengan baik tanpa adanya dukungan dan bantuan dari berbagai pihak. Selama dalam menyelesaikan tugas akhir/skripsi ini, penulis mendapatkan banyak bimbingan, bantuan dan arahan dari berbagai pihak. Oleh karena itu dengan kesempatan ini, penulis menyampaikan rasa hormat dan terimakasih kepada:

1. Tuhan Yang Maha Kuasa yang telah memberikan rahmat-Nya, kesehatan dan juga karunia-Nya, sehingga saya sebagai penulis dapat menyelesaikan Skripsi ini dengan sebaik-baiknya.
2. Orang Tua Bapak dan Ibu Saya yang telah mendukung, memberikan semangat, motivasi, dan banyak perhatian serta memenuhi segala kebutuhan saya selama masa penyusunan tugas akhir/skripsi ini.
3. Yayasan Haji Agus Salim Universitas Medan Area.
4. Bapak Prof. Dr. Dadan Ramdan, M.Eng., M.Sc selaku Rektor Universitas Medan Area
5. Bapak Dr. Rahmad Syah, S.Kom., M.Kom selaku Dekan Fakultas Teknik Universitas Medan Area.
6. Ibu Susilawati S.Kom, M.Kom Selaku Wakil Dekan.
7. Bapak Rizki Muliono, S.Kom, M.Kom selaku Kepala Program Studi Teknik Informatika dan Dosen Pembimbing II Tugas Akhir yang telah memberikan arahan, bimbingan, semangat, dan motivasi kepada penulis hingga penyusunan tugas akhir/skripsi.
8. Bapak Muhathir S.T, M.Kom., selaku Dosen Pembimbing I yang telah memberikan arahan, bimbingan, semangat, dan motivasi kepada penulis hingga penyusunan tugas akhir/skripsi.

9. Seluruh Dosen Teknik Informatika Universitas Medan Area yang telah memberikan ilmunya dan saran kepada penulis selama menjalani Studi.
10. Seluruh pegawai Universitas Medan Area yang telah membantu dalam proses administrasi.
11. Teman-teman kelas saya terkhususnya untuk Teknik Informatika angkatan 2017.

Penulis menyadari bahwa di dalam tugas akhir/skripsi ini masih terdapat kekurangan didalam penulisannya. Oleh karena itu, dengan kerendahan hati penulis menerima kritikan dan saran yang membangun demi tercapainya kesempurnaan tugas akhir/skripsi ini nantinya. Semoga tugas akhir/skripsi ini bermanfaat bagi pengembangan Ilmu pengetahuan.

Medan, 4 Mei 2023



Loudji Zamico Pratama

178160010

DAFTAR ISI

LEMBAR PENGESAHAN	ii
HALAMAN PERNYATAAN	iii
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	iv
ABSTRAK	v
ABSTRACT	vi
RIWAYAT HIDUP.....	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	5
1.3. Tujuan Penelitian	5
1.4. Batasan Masalah	5
1.5. Manfaat Penelitian	5
BAB II TINJAUAN PUSTAKA.....	6
2.1. Rempah-rempah.....	6
2.2. Klasifikasi	7
2.3. Citra Digital	7
2.3.1. Citra Warna.....	9
2.3.2. Citra <i>Grayscale</i>	9
2.3.3. Citra Biner	10
2.4. <i>Deep Learning</i>	11
2.5. <i>Convolutional Neural Network</i>	11
2.5.1. <i>Convolution Layer</i>	13
2.5.2. <i>Subsampling Layer</i>	13
2.5.3. <i>Flatten Layer</i>	14
2.5.4. <i>Fully Connected Layer</i>	15
2.5.5. Fungsi Aktivasi	15
2.6. <i>Dropout</i>	17
2.7. <i>Confusion Matrix</i>	18
2.7. Penelitian Terdahulu	19
BAB III METODOLOGI PENELITIAN.....	22
3.1. Alat dan Bahan Penelitian.....	22
3.1.1. Perangkat Keras (<i>Hardware</i>).....	22
3.1.2. Perangkat Lunak (<i>Software</i>).....	22
3.2. Alur Metodologi Penelitian.....	23
3.1. Pengumpulan <i>Dataset</i>	24

3.2. <i>Preprocessing Dataset</i>	25
3.2.1. <i>Resize</i>	25
3.2.2. <i>Pembagian Dataset</i>	25
3.3. <i>Klasifikasi CNN</i>	26
3.3.1. <i>Convolution Layer</i>	26
3.3.2. <i>Subsampling Layer</i>	29
3.3.3. <i>Flatten Layer</i>	31
3.3.4. <i>Fully Connected Layer</i>	31
3.4. <i>Inialisasi Hyperparameter</i>	32
3.5. <i>Metode Evaluasi</i>	32
3.5.1. <i>Confusion Matrix</i>	33
3.5.2. <i>Pengukuran Kinerja</i>	33
BAB IV HASIL DAN PEMBAHASAN	35
4.1. <i>Hasil</i>	35
4.1.1. <i>Implementasi CNN</i>	35
4.1.2. <i>Confusion Matrix</i>	40
4.1.3. <i>Evaluasi Kinerja Algoritma</i>	56
4.2. <i>Pembahasan</i>	72
BAB V KESIMPULAN DAN SARAN	74
5.1 <i>Kesimpulan</i>	74
5.1 <i>Saran</i>	74
DAFTAR PUSTAKA	75
LAMPIRAN	78

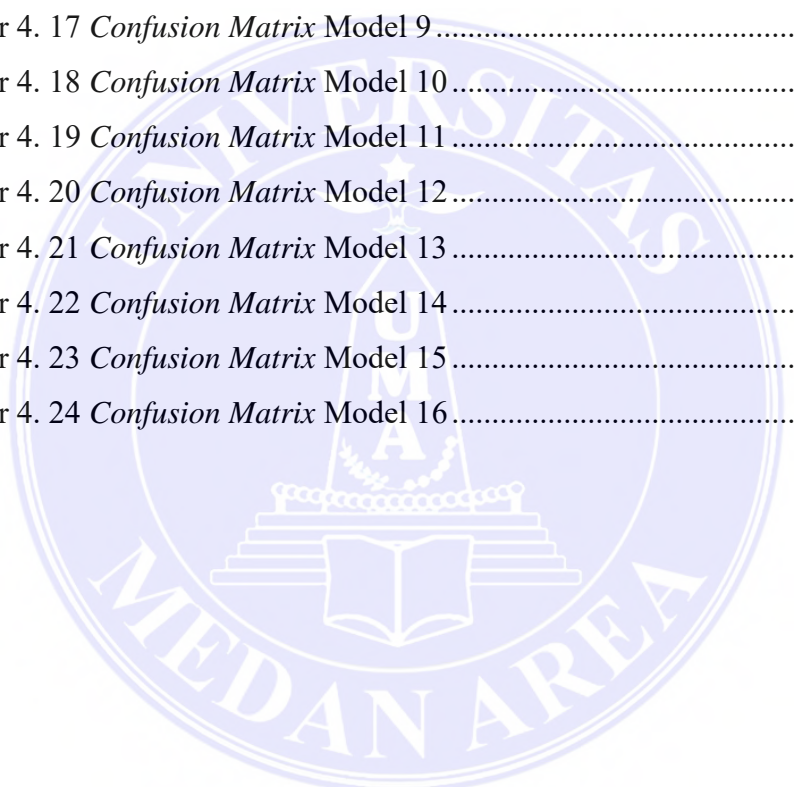
DAFTAR TABEL

Tabel 3. 1 Perangkat Keras (<i>Hardware</i>)	22
Tabel 3. 2 Perangkat Lunak (<i>Software</i>).....	22
Tabel 3. 3 <i>Resize</i>	25
Tabel 3. 4 Pembagian <i>Dataset</i>	26
Tabel 3. 5 <i>Hyperparameter</i>	32
Tabel 3. 6 <i>Confusion Matrix</i>	33
Tabel 4. 1 Model Pengujian.....	35
Tabel 4. 2 Hasil Evaluasi Model 1	56
Tabel 4. 3 Hasil Evaluasi Model 2	57
Tabel 4. 4 Hasil Evaluasi Model 3	58
Tabel 4. 5 Hasil Evaluasi Model 4	59
Tabel 4. 6 Hasil Evaluasi Model 5	60
Tabel 4. 7 Hasil Evaluasi Model 6	61
Tabel 4. 8 Hasil Evaluasi Model 7	62
Tabel 4. 9 Hasil Evaluasi Model 8	63
Tabel 4. 10 Hasil Evaluasi Model 9	64
Tabel 4. 11 Hasil Evaluasi Model 10	65
Tabel 4. 12 Hasil Evaluasi Model 11	66
Tabel 4. 13 Hasil Evaluasi Model 12	67
Tabel 4. 14 Hasil Evaluasi Model 13	68
Tabel 4. 15 Hasil Evaluasi Model 14	69
Tabel 4. 16 Hasil Evaluasi Model 15	70
Tabel 4. 17 Hasil Evaluasi Model 16	71
Tabel 4. 18 Hasil Pengujian Model.....	72
Tabel 4. 19 Perbandingan Kinerja Model Penelitian	73

DAFTAR GAMBAR

Gambar 2. 1 Jenis Rempah-rempah	6
Gambar 2. 2 Citra digital dan Nilai piksel	8
Gambar 2. 3 Citra Warna (RGB)	9
Gambar 2. 4 Citra <i>Grayscale</i>	10
Gambar 2. 5 Citra Biner	10
Gambar 2. 6 Arsitektur CNN	12
Gambar 2. 7 Proses <i>Convolution Layer</i>	13
Gambar 2. 8 Proses <i>Max Pooling</i>	14
Gambar 2. 9 Proses <i>Flatten Layer</i>	14
Gambar 2. 10 <i>Fully Connected Layer</i>	15
Gambar 2. 11 Fungsi Aktivasi Relu.....	16
Gambar 2. 12 Cara Kerja Dropout	17
Gambar 2. 13 <i>Confusion Matrix</i>	18
Gambar 3. 1 Alur Metodologi Penelitian.....	23
Gambar 3. 2 Contoh <i>Dataset</i> Rempah	24
Gambar 3. 3 Sampel Data Input.....	26
Gambar 3. 4 Proses Konvolusi Dengan Kernel 3 x 3	27
Gambar 3. 5 Pergeseran Matrik	28
Gambar 3. 6 Hasil Perhitungan <i>Convolution Layer</i>	29
Gambar 3. 7 Proses <i>Max Pooling</i>	30
Gambar 3. 8 Hasil <i>Max Pooling</i>	30
Gambar 3. 9 Hasil Dari <i>Flatten Layer</i>	31
Gambar 3. 10 <i>Fully Connected Layer</i>	31
Gambar 4. 1 <i>Training Accuracy Epoch 15</i>	36
Gambar 4. 2 <i>Loss Training Accuracy Epoch 15</i>	36
Gambar 4. 3 <i>Accuracy Validation Epoch 15</i>	37
Gambar 4. 4 <i>Loss Validation Accuracy Epoch 15</i>	37
Gambar 4. 5 <i>Training Accuracy Epoch 25</i>	38
Gambar 4. 6 <i>Loss Training Accuracy Epoch 25</i>	38
Gambar 4. 7 <i>Validation Accuracy Epoch 25</i>	39

Gambar 4. 8 <i>Loss Validation Accuracy Epoch 25</i>	39
Gambar 4. 9 <i>Confusion Matrix Model 1</i>	40
Gambar 4. 10 <i>Confusion Matrix Model 2</i>	41
Gambar 4. 11 <i>Confusion Matrix Model 3</i>	42
Gambar 4. 12 <i>Confusion Matrix Model 4</i>	43
Gambar 4. 13 <i>Confusion Matrix Model 5</i>	44
Gambar 4. 14 <i>Confusion Matrix Model 6</i>	45
Gambar 4. 15 <i>Confusion Matrix Model 7</i>	46
Gambar 4. 16 <i>Confusion Matrix Model 8</i>	47
Gambar 4. 17 <i>Confusion Matrix Model 9</i>	48
Gambar 4. 18 <i>Confusion Matrix Model 10</i>	49
Gambar 4. 19 <i>Confusion Matrix Model 11</i>	50
Gambar 4. 20 <i>Confusion Matrix Model 12</i>	51
Gambar 4. 21 <i>Confusion Matrix Model 13</i>	52
Gambar 4. 22 <i>Confusion Matrix Model 14</i>	53
Gambar 4. 23 <i>Confusion Matrix Model 15</i>	54
Gambar 4. 24 <i>Confusion Matrix Model 16</i>	55



BAB I

PENDAHULUAN

1.1. Latar Belakang

Indonesia merupakan salah satu negara beriklim tropis yang memiliki kekayaan alam yang sangat besar kedua di dunia setelah Brazil. Salah satu kekayaan alam terkenal yang dimiliki oleh Indonesia berasal dari sektor agraris atau pertanian yaitu berupa kekayaan rempah-rempah. (Wulandari dkk., 2020).

Rempah-rempah merupakan jenis tumbuhan yang sangat mudah ditemukan di wilayah Indonesia dan manfaatnya sangat banyak digunakan oleh masyarakat. Rempah-rempah terdapat pada bagian-bagian tumbuhan yang berasal dari bagian batang, daun, kulit kayu, umbi, rimpang, akar, biji, bunga dan bagian tubuh tumbuhan lainnya. Aroma yang dihasilkan rempah-rempah sangat kuat, sehingga sering digunakan dalam jumlah kecil maupun dalam jumlah yang besar sebagai perisa, pewarna, pengawet makanan dan untuk obat tradisional. (Kaharuddin dkk., 2019)

Kekayaan jenis rempah-rempah yang dimiliki Indonesia seperti kemiri, cabai, temu kunci, lengkuas, jahe, kencur, temulawak, kunyit, cengkeh, lada, pala, ketumbar, kayu manis, kapulaga, vanili, andaliman, serai, asam jawa, jintan, adas, daun salam, bawang merah dan bawang putih. (Fitriyan & Magriza, 2022).

Rempah-rempah memiliki keunikan jenis dan bentuknya masing-masing. Seperti lengkuas yang memiliki rimpang besar dan tebal, berdaging, berbentuk silindris, berdiameter sekitar 2-4 cm, dan bercabang-cabang. Bagian luar lengkuas berwarna coklat sedikit kemerahan atau kuning kehijauan pucat, mempunyai sisik berwarna putih atau kemerahan, keras mengkilap, sedangkan pada bagian dalam berwarna putih dan daging rimpang yang sudah tua berserat kasar. Kunyit memiliki rimpang bercabang banyak, bagian luar tampak berwarna jingga kecoklatan, dan bagian dalamnya berwarna jingga terang atau kuning. Temulawak memiliki rimpang yang besar, yang terdiri dari rimpang induk berbentuk bulat seperti telur, dengan anakan-anakan rimpang yang panjang langsing, berjumlah 3-4. Bagian luar rimpang temulawak berwarna kuning muda, dan bagian dalamnya kuning jingga. Kemudian pada rempah kencur memiliki rimpang sedikit pendek berwarna coklat, berbentuk jari dan tumpul. Bagian luarnya seperti bersisik.

Daging rimpang tidak keras, rapuh, mudah patah dan bergetah. Sedangkan jahe memiliki ciri-ciri yaitu rimpangnya bercabang - cabang, tebal dan agak melebar (tidak silindris), berwarna kuning pucat. Pada bagian dalam rimpangnya berserat sedikit kasar, dan berwarna kuning muda dengan ujung merah muda. (Matondang dkk., 2005).

Pada Penelitian terdahulu yang dilakukan Kaharuddin dkk (2019). Dilakukan ujicoba dengan menggunakan 4 jenis rempah-rempah yaitu: kunyit, jahe, kencur, dan lengkuas. Jumlah data yang digunakan sebanyak 160 citra dari 4 jenis rempah-rempah. Penelitian ini memanfaatkan fitur warna RGB dan tekstur pada citra yang di klasifikasi dengan algoritma *K-Nearest Neighbor* dan menggunakan pengukuran jarak menggunakan *Euclidean Distance*. Hasil akurasi yang didapat yaitu 73%. (Kaharuddin dkk., 2019)

Penelitian selanjutnya yang dilakukan Jatmoko dan Sinaga (2020). Penelitian dilakukan dengan menggunakan 2 jenis dari rempah-rempah yaitu: kunyit dan temulawak. Data citra yang digunakan pada penelitian berjumlah sebanyak 180 citra, terbagi menjadi data latih dan data uji. Sebelum dilakukan klasifikasi dengan algoritma *K-Nearest Neighbor* citra terlebih dahulu di ekstraksi fitur warna untuk memperoleh nilai RGB yang selanjutnya dihitung *euclidean distance* pada algoritma *K-Nearest Neighbor*. Hasil yang diperoleh dapat membedakan citra kunyit dan citra temulawak dengan baik dengan nilai akurasi tertinggi $K1=95\%$ (Jatmoko & Sinaga, 2020).

Penelitian selanjutnya dilakukan Batubara dkk (2020). Penelitian dilakukan dengan menggunakan 4 jenis rempah-rempah yaitu: jahe, kunyit, lengkuas dan temulawak. Jumlah data citra yang digunakan sebanyak 80 citra dari 4 jenis rempah-rempah. Pertama Citra rempah dikonversi menjadi citra *Grayscale* dan dilakukan langkah *segmentasi* dengan *thresholding* menggunakan metode *Otsu*. Setelah didapatkan hasil dari *segmentasi*, kemudian akan dilakukan *ROI (Region of Interest)* yang hasilnya berupa perkalian pada piksel. Selanjutnya akan dilakukan tahapan ekstraksi ciri dengan memanfaatkan *GLCM (Grey Level Co-occurrence Matrix)* dan ekstraksi fitur RGB (*Red, Green, Blue*). Setelah hasil dari tahapan ekstraksi ciri didapatkan, maka akan dilakukan klasifikasi dengan algoritma *Naïve Bayes*. Tahapan terakhir adalah uji performa dari hasil klasifikasi

dengan menggunakan *K-fold cross validation* yang menghasilkan akurasi sebesar 52%. (Batubara dkk., 2020)

Berdasarkan dari beberapa penelitian terdahulu yang dipaparkan belum ditemukan klasifikasi dari jenis rempah-rempah dengan menggunakan algoritma *Convolutional Neural Network* (CNN). CNN adalah jenis dari algoritma *deep learning* yang merupakan pengembangan dari *Multilayer Perceptron* (MLP). CNN sering digunakan untuk menyelesaikan permasalahan *computer vision* terkhususnya pada identifikasi citra dan pengolahan citra. CNN biasanya digunakan untuk melakukan klasifikasi gambar, mengelompokkannya berdasarkan kesamaan, dan melakukan pengenalan objek. CNN terdiri dari beberapa layer dan dirancang untuk pengenalan citra kompleks yang efektif. Kelebihan utama yang dimiliki CNN dibandingkan dengan pendahulunya yaitu dapat secara otomatis mendeteksi fitur-fitur penting dengan cara mempelajari atau mengekstrak fitur dari kumpulan data seperti gambar, video, dan teks. Algoritma CNN mempunyai performansi yang baik dalam ekstraksi fitur untuk mendapatkan akurasi yang tinggi layaknya cara kerja pada otak manusia. (Baihaqy, Wibowo, & Utama, 2022)

Algoritma CNN pernah dilakukan pada beberapa kasus seperti klasifikasi kendaraan pada jalan raya, klasifikasi penyakit kulit wajah, dan klasifikasi identitas dengan citra telapak tangan. Pada klasifikasi kendaraan pada jalan raya yang dilakukan Iskandar Mulyana dkk (2022). Pada Penelitian ini citra yang akan diklasifikasi yaitu: Sepeda, Sepeda Motor, Becak, Bajaj, Mobil, Mobil Pick up, Mobil Molen, Bus, dan Truk. Jumlah data citra sebanyak 1700 citra yang dibagi menjadi data latih dan data uji. 1406 citra sebagai data latih dan 274 citra sebagai data uji. Implementasi pengenalan citra kendaraan dilakukan dengan menggunakan 2 model uji, model *Sequential* dan model VGG16 tingkat atas yang dijalankan dengan memanfaatkan Google Collaboratory dan Keras. Model *Sequential* mendapatkan hasil berupa nilai evaluasi akurasi sebesar 98,18%, dan nilai *loss* 0,103. Sedangkan pada model *on top* VGG16 menghasilkan tingkat akurasi 99,64% dan tingkat *loss* 0,014. (Iskandar Mulyana dkk., 2022).

Selanjutnya CNN pada kasus klasifikasi penyakit kulit wajah yang dilakukan Nurkhasanah & Murinto (2021). Pada penelitian ini akan diklasifikasikan

5 jenis penyakit kulit wajar yang terjadi pada kulit manusia yaitu jerawat, eksim, milia, herpes, dan melasma. Arsitektur CNN yang dipakai yaitu VGG16. Dalam arsitektur VGG16 terdapat dua lapisan yaitu *feature extraction* dan *classification*. Lapisan *feature extraction* terdiri dari *convolution layer*, Fungsi aktivasi ReLU dan *average pooling* yang digunakan untuk mengurangi *feature map*. Sedangkan Lapisan *classification* terdiri atas *fully connected layer* dan Fungsi Aktivasi *softmax* digunakan untuk proses klasifikasi. Hasil CNN yang diperoleh sangat baik, dimana pelatihan sebesar 98% dan hasil dari validasi sebesar 88% dari 325 citra data pelatihan dan 125 data validasi. Pada pengujian terhadap data baru hasil akurasi yang diperoleh sebesar 90% dari 50 citra data uji. (Nurkhasanah & Murinto, 2021)

Selanjutnya Algoritma CNN pernah dilakukan penelitian pada kasus Klasifikasi Identitas Dengan Citra Telapak Tangan yang dilakukan Wita & Liliana (2022). Telapak tangan menjadi objek pada penelitian ini, karena pada telapak tangan memiliki fitur/pola yang sangat unik serta berbeda pada setiap individu. Selain fitur/pola unik, luas yang dimiliki permukaan telapak tangan juga menjadi salah satu pertimbangan dalam menentukan objek penelitian. Luas permukaan telapak tangan lebih besar dibandingkan dengan luas permukaan jari. 26 label digunakan sebagai *dataset* dimana label tersebut dimulai dari label A sampai pada label Z. Klasifikasi masing-masing identitas pada citra telapak tangan memanfaatkan algoritma CNN. Hasil klasifikasi dari identitas telapak tangan ini menunjukkan masing-masing label dapat diklasifikasikan dengan akurasi 98%. (Wita & Liliana, 2022)

Berdasarkan kelebihan dari algoritma CNN yang dinilai dapat menghasilkan akurasi paling signifikan dalam pengolahan dan pengenalan citra. Maka dalam penelitian ini akan dilakukan uji coba model dari *deep learning* yaitu CNN dalam mengklasifikasi jenis rempah-rempah dengan tujuan untuk mempermudah masyarakat dalam mengenali atau membedakan jenis rempah-rempah terutama pada rempah-rempah yang memiliki kemiripan baik dari segi bentuk dan warna.

1.2. Rumusan Masalah

Adapun rumusan masalah pada penelitian ini yaitu, Bagaimana kinerja dari algoritma *Convolutional Neural Network* (CNN) dalam mengklasifikasi jenis rempah-rempah?

1.3. Tujuan Penelitian

Tujuan dari penelitian ini untuk mengukur kinerja dari CNN dalam mengklasifikasi rempah-rempah.

1.4. Batasan Masalah

Adapun batasan masalah pada penelitian ini yaitu sebagai berikut:

1. Citra yang digunakan yaitu berupa citra rempah-rempah dengan format .Jpg.
2. Citra rempah-rempah diambil dengan memanfaatkan kamera smartphone Samsung A32 berbasis *Android*.
3. Jenis rempah-rempah yang digunakan sebanyak 10 jenis rempah, yaitu Jahe, Kunyit, Lengkuas, Temulawak, Kencur, Pala, Kemiri, Jintan, Adas Manis, dan Ketumbar.
4. Jumlah keseluruhan data gambar yaitu berjumlah 1500 gambar, dimana masing-masing dari jenis rempah sebanyak 150 gambar.
5. Jarak pengambilan objek citra tidak lebih dari 20 cm.
6. Menggunakan 2 optimizer yaitu, Adam dan RMSprop
7. Fungsi aktivasi yang digunakan yaitu, Relu dan *Softmax*.
8. *Performance Measure* yang digunakan yaitu, *Accuracy*, *Precision*, *Recall* dan *F1-Score*.

1.5. Manfaat Penelitian

1. Membantu masyarakat dalam klasifikasi citra rempah-rempah, sehingga mampu mempermudah dalam mengenali jenis rempah dengan benar.
2. Meningkatkan wawasan sebagai bahan pembelajaran serta referensi untuk penelitian mengenai *image processing*.
3. Sebagai kebaruan ilmu komputer khususnya pada bidang *deep learning* dalam pengenalan rempah.

BAB II

TINJAUAN PUSTAKA

2.1. Rempah-rempah

Rempah-rempah merupakan salah satu dari sumber daya hayati yang sejak lama memiliki peran sangat penting dalam sejarah kehidupan manusia. Dimana rempah-rempah memiliki banyak manfaat, diantaranya sebagai sumber makanan dan pengobatan. (Robi, Masitoh, & Muflihati, 2019)

Rempah-rempah sangat banyak dijumpai pada negara-negara tropis, termasuk Indonesia yang memiliki kekayaan dari sektor agraris atau pertanian. Banyak dari jenis rempah-rempah ini dapat tumbuh dengan mudah di kebun dan pekarangan rumah. Kegunaan dari rempah-rempah juga sudah banyak dimanfaatkan oleh masyarakat dalam kepentingan industri farmasi, industri pangan, dan industri lainnya. (Jatmoko & Sinaga, 2020)

Rempah-rempah sendiri dapat ditemukan pada bagian tumbuhan yang terdapat pada bagian batang, daun, kulit kayu, umbi, rimpang (*rhizome*), akar, biji, bunga, dan bagian lainnya. Bagian-bagian tumbuhan tersebut mempunyai kandungan dari senyawa fitokimia yang merupakan hasil dari tumbuhan sebagai bagian dari proses metabolisme tumbuhan (Kaharuddin dkk., 2019)

Saat ini, rempah-rempah telah menjadi salah satu komoditi utama dari banyak negara di dunia. Rempah-rempah juga memiliki nilai ekonomi yang sangat penting dan merupakan sumber pendapatan dari banyak negara di dunia.



Gambar 2. 1 Jenis Rempah-rempah

2.2. Klasifikasi

Klasifikasi merupakan sebuah langkah atau proses dalam data mining untuk membentuk suatu model ataupun fungsi yang digunakan untuk menjelaskan serta memisahkan kelas data yang satu dengan kelas data yang lainnya. Dimana pemisahan kelas data ini bertujuan untuk menyatakan objek tersebut masuk pada kategori tertentu dengan melihat karakteristik dan atribut data berdasarkan dengan hasil dari model yang telah didefinisikan sebelumnya. (Nursyahfitri dkk., 2021)

Pada proses untuk melakukan klasifikasi terbagi menjadi 2 tahapan yaitu tahap pelatihan (*training*) yang merupakan tahap untuk melakukan pelatihan atau pembelajaran terhadap suatu data yang bertujuan untuk membentuk suatu model atau fungsi yang dapat digunakan dalam klasifikasi. Dan tahap pengujian (*testing*) merupakan tahap untuk melakukan evaluasi tingkat kinerja yang diperoleh dari hasil model tahap pelatihan (*training*). Keluaran (*output*) tahap pengujian berupa hasil tingkat akurasi model atau fungsi untuk memprediksi data yang tidak dikenali kelasnya dengan menggunakan data pengujian.

Hubungan klasifikasi sangat erat kaitan dengan algoritma-algoritma yang memiliki fungsi mengelompokkan data sesuai dengan kelasnya. Terdapat beberapa algoritma yang bisa digunakan untuk klasifikasi diantaranya adalah Algoritma KNN (*K-Nearest Neighbor*), *Naïve Bayes*, SVM, Algoritma CNN (*Convolutional Neural Network*), dan lain-lainya. Dimana masing-masing dari algoritma memiliki langkah-langkah kerja yang berbeda dalam melakukan klasifikasi terhadap suatu kasus atau data. (Yudianto, Kusriani, & Al Fatta, 2020)

2.3. Citra Digital

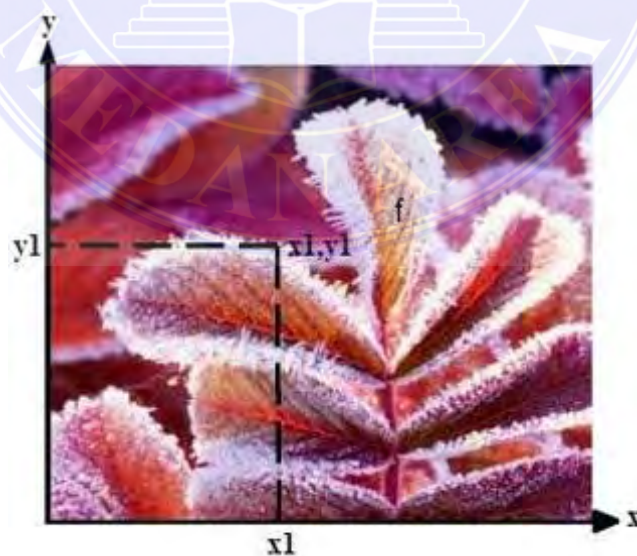
Citra digital merupakan gambar yang direpresentasikan secara digital dalam bentuk 2 dimensi. Citra digital secara umum tersimpan pada suatu larik (*array*) yang terdiri atas nilai-nilai real serta kompleks yang direpresentasikan dengan deret bit-bit tertentu. (Zakti, Fauzi, & Sihombing, 2022)

Citra digital yang tersimpan pada larik (*array*) 2 dimensi tersebut terdiri atas unsur-unsur kecil yang disebut sebagai piksel. Dimana masing-masing dari piksel memiliki satu warna yang menyatu dengan piksel-piksel yang lainnya sehingga membentuk suatu pola dan menghasilkan gambar.

Larik memiliki struktur yang terdiri dari baris horizontal yang disebut dengan baris (*lines*) dan kolom vertical (*samples*). Susunan dari piksel dan struktur larik (*array*) dari citra digital tersebut disebut dengan data raster. (Prabowo, Abdullah, & Manik, 2018)

Umumnya saat melakukan representasi, sebuah citra digital memerlukan penggunaan *memory* dalam ukuran besar. Penggunaan *memory* yang besar tergantung juga pada ukuran citra digital. Pada sisi lainnya, citra digital banyak mengandung duplikasi pada data. Duplikasi data ini bisa diartikan dalam dua hal. Pertama, kemungkinan besar suatu piksel dengan piksel lainnya pada citra digital mempunyai intensitas yang sama, sehingga setiap piksel memakan tempat pada penyimpanan *memory*. Kedua, banyak citra digital yang mengandung *region* yang sama, untuk hal itu bagian yang sama tersebut tidak butuh dikodekan secara berulang-ulang kali disebabkan redundansi. (Ndruru, 2020)

Representasi citra digital dan pikselnya dapat ditunjukkan pada gambar 2.2 berikut.



Gambar 2. 2 Citra digital dan Nilai piksel
Sumber : (Robianto, Sitorus, Ristian, 2021)

Citra digital dapat dibagi menjadi 3 jenis citra yaitu citra warna, citra *grayscale*, dan citra biner.

3.3.1 Citra Warna

Citra warna merupakan sebuah sistem grafik yang mempunyai satu set nilai-nilai yang tersusun untuk menyatakan tingkat dari warna. Setiap piksel yang ada pada citra mewakili warna yang merupakan kombinasi warna RGB (*Red, Green, Blue*). Dimana setiap warna tersebut membutuhkan penyimpanan *memory* sebesar 8 bit atau 1 *byte*, yang artinya tiap warna memiliki gradasi warna sebanyak 256. Sehingga setiap piksel dalam citra memiliki kombinasi warna yaitu $2^8 \times 2^8 \times 2^8 = 2^{24} = 16.777.216$ warna. Hal inilah yang menyebabkan citra warna disebut juga dengan *true color* karena memiliki jumlah warna yang banyak. (Robianto, Sitorus, & Ristian, 2021)



Gambar 2.3 Citra Warna (RGB)

2.3.2. Citra *Grayscale*

Citra *grayscale* merupakan satu dari jenis citra digital yang hanya memiliki satu nilai kanal yang ada pada setiap piksel. Dimana bisa dikatakan nilai warna dari R (*Red*) = G (*Green*) = B (*Blue*). Nilai tersebut digunakan untuk menunjukkan tingkat dari intensitas. (Gultom, Sianturi, & Ndruru, 2021)

Pada umumnya kebanyakan dari citra *grayscale* menggunakan penyimpanan sebesar 8 bit atau 1 *byte* yang dapat menampung 256 tingkat skala abu-abu. Dimana pada setiap piksel memiliki nilai intensitas mulai dari 0 sebagai

nilai minimum hingga 255 sebagai nilai maksimal, dengan warna hitam yang memiliki nilai intensitas yang paling kecil yaitu 0 dan warna putih yang memiliki nilai intensitas yang paling besar yaitu 255. (Ellif, Sitorus, & Hidayati, 2021)



Gambar 2. 4 Citra *Grayscale*

2.3.3. Citra Biner

Citra biner merupakan citra yang hanya memiliki 2 nilai piksel yaitu 0 dan 1. Nilai piksel 0 digunakan untuk merepresentasikan warna hitam dan nilai piksel 1 digunakan untuk merepresentasikan warna putih.

Karena citra biner hanya memiliki dua warna pada setiap nilai piksel, sehingga penyimpanan *memory* yang diperlukan hanya 1 bit untuk mewakili setiap piksel citra biner. Citra biner ini sering juga disebut dengan Citra B&W atau citra monokrom. (Arkani, Magdalena, & Patmasari, 2020)



Gambar 2. 5 Citra Biner

2.4. Deep Learning

Deep learning merupakan cabang *machine learning* yang terdiri dari algoritma dengan pemodelan abstraksi tingkat tinggi pada data dengan menggunakan kumpulan dari fungsi transformasi *non-linear* yang berlapis-lapis dan sangat mendalam bahkan dapat mencapai ratusan lapisan. (Adrianto, Wahyuddin, & Winarsih, 2021)

Pada saat ini, telah banyak yang memanfaatkan *deep learning* dalam berbagai macam pekerjaan. Dimana *deep learning* ini digunakan untuk mengenali objek, memprediksi kejadian atau peluang, dan mendiagnosa berbagai macam penyakit. Salah satu di dalam pemanfaatan pada *deep learning* yaitu pada pengolahan citra digital. Dengan adanya *deep learning* ini dapat mempermudah manusia dalam mengenali atau mengklasifikasi objek dengan baik dan efisien. (Maulana & Rochmawati, 2019)

Yang menjadi satu tantangan untuk keberhasilan implementasi dengan *deep learning* yaitu dalam pengaturan pada nilai untuk *hyperparameter*, salah satunya yaitu topologi jaringan, yang kaitan sangat erat dengan jumlah *hidden layer* dan jumlah *hidden neuron*. Penentuan untuk jumlah *hidden layer* dan jumlah untuk *neuron* sangat penting dan mempunyai pengaruh yang besar pada kinerja yang dilakukan *deep learning*. Metode *deep learning* ini juga secara signifikan meningkatkan kerja pada pengenalan percakapan, pengenalan dari objek visual, deteksi obyek dan lain sebagainya. (Choldun R. & Surendro, 2018)

2.5. Convolutional Neural Network

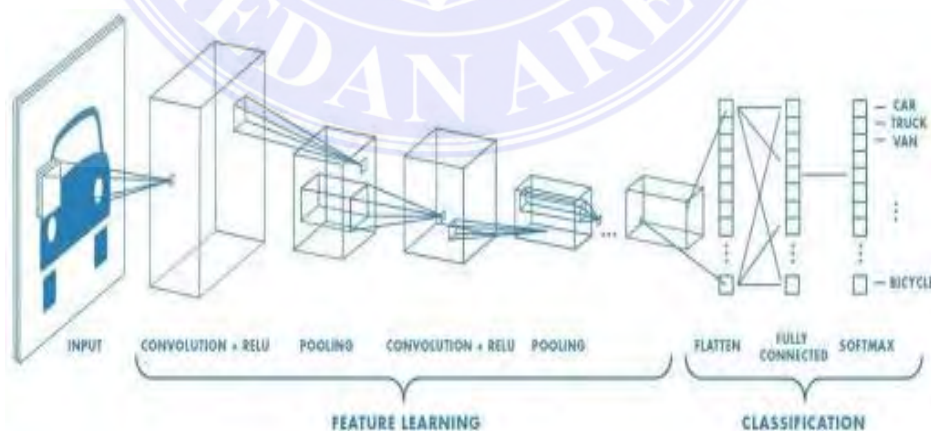
Convolutional Neural Network (CNN) adalah satu dari beberapa jenis algoritma *Deep Learning* yang merupakan pengembangan dari *Multilayer Perceptron* (MLP) yang bertujuan untuk melakukan pengolahan data dalam bentuk dua dimensi. Algoritma CNN termasuk jenis *deep neural network* karena memiliki kedalaman jaringan yang tinggi dan sering diaplikasikan untuk data pada citra. Algoritma CNN sangat banyak digunakan pada saat melakukan analisis gambar visual, mendeteksi maupun mengenali objek yang ada pada citra atau *image*. Secara garis besar, CNN tidak jauh perbedaannya dengan *neural network* lainnya. Dimana algoritma CNN ini terdiri dari *neuron* yang mempunyai nilai

bobot (*weight*), bias dan *activation function*. (Nugroho, Fenriana, & Arijanto, 2020)

Pada dasarnya, konsep yang ada pada algoritma CNN memiliki kesamaan atau kemiripan dengan cara kerja yang terdapat pada MLP. Tetapi pada CNN, setiap *neuron-neuron* nya dipresentasikan atau ditampilkan ke dalam bentuk dua dimensi. Sedangkan pada MLP, tiap-tiap *neuron* nya di presentasikan atau ditampilkan hanya dalam bentuk satu dimensi. Sehingga dengan ada perbedaan tersebut menyebabkan parameter bobot dari CNN dan MLP berbeda. (Wairata, Swedia, & Cahyanti, 2021)

Pada CNN, data yang dipropagasikan pada jaringan adalah data dalam bentuk dua dimensi. Sehingga pada CNN operasi linear dan parameter bobot berbeda. Dalam CNN operasi linear menggunakan operasi konvolusi, sedangkan bobot tidak lagi dalam bentuk satu dimensi, melainkan berbentuk empat dimensi yang merupakan kumpulan dari kernel konvolusi. (Septian dkk., 2020).

Salah satu kelebihan yang dimiliki pada algoritma CNN yaitu algoritma ini tidak membutuhkan ekstraksi ciri tertentu pada prosesnya untuk menghasilkan klasifikasi. Namun algoritma CNN memerlukan jumlah data yang lebih banyak untuk proses training sehingga secara komputasi membutuhkan unit pemrosesan grafis yang mahal dari segi biaya untuk pelatihan model (Nashr, Fachrurrozi, Triningsih, & Miraswan, 2020).



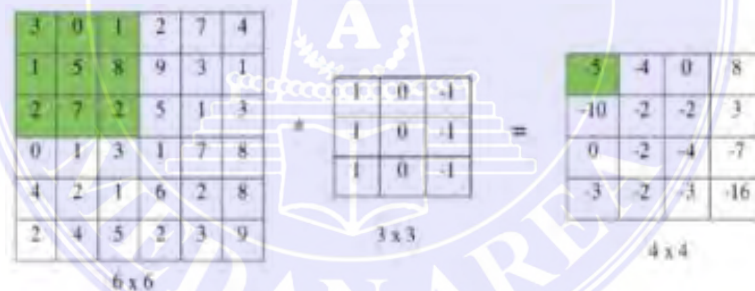
Gambar 2. 6 Arsitektur CNN
Sumber : (Nugroho, Fenriana, & Arijanto, 2020)

Algoritma CNN memiliki beberapa layer-layer utama sebagai berikut :

2.5.1. Convolution Layer

Convolution Layer merupakan proses *convolution* yang menjadi bagian yang penting pada arsitektur algoritma CNN, karena *convolution* menjadi lapisan pertama atau utama untuk memproses sebuah citra. *Convolution* adalah sebuah istilah dalam matematis yang memiliki makna mengaplikasikan suatu fungsi yang ada pada *output* fungsi lainnya secara berulang. Dalam melakukan pengolahan citra, *convolution* berarti mengaplikasikan kernel dari suatu citra disemua *offset* yang memungkinkan. (Pintanarum, Prasetiadi, & Ramdani, 2021)

Convolution data citra bertujuan untuk mengekstraksi fitur-fitur pada citra yang dijadikan masukan (*input*). Dimana *convolution* menghasilkan transformasi linear dari data *input* sesuai dengan informasi spasial yang ada pada data citra. Bobot pada layer menspesifikasikan *kernel convolution* yang digunakan, sehingga *kernel* pada *convolution* dapat dilatih dengan *input* yang ada pada algoritma CNN. (Alwanda, Ramadhan, & Alamsyah, 2020).

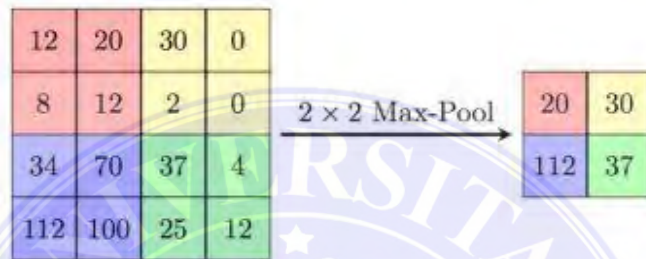


Gambar 2. 7 Proses *Convolution Layer*
Sumber : (Yudianto, Kusri, & Al Fatta, 2020)

2.5.2. Subsampling Layer

Subsampling merupakan tahapan yang berfungsi untuk melakukan reduksi terhadap ukuran dari sebuah data citra. Pada pengolahan citra, *subsampling* juga digunakan sebagai peningkatkan *invariance* posisi dari fitur. Algoritma CNN sebagian besar menggunakan metode *subsampling layer* yaitu *max pooling*. *max pooling* adalah proses membagi *output* yang dihasilkan dari *convolution layer* menjadi beberapa *grid*/tabel kecil. Dimana dari *grid*/tabel tersebut akan diambil

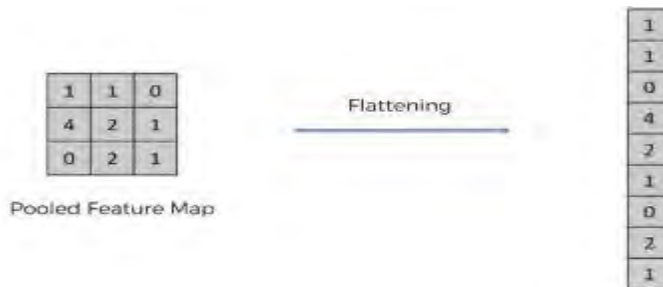
nilai-nilai maksimal setiap *grid*/tabel untuk disusun nilai matriks pada citra yang telah di reduksi. Proses reduksi ini memastikan bahwa fitur yang didapatkan akan sama, meskipun pergeseran (*translasi*) terjadi pada objek citra. Tujuan dilakukan tahapan *subsampling* atau *pooling layer* ini ialah untuk mengurangi dimensi yang terdapat pada *feature map* (*downsampling*), sehingga dapat mempercepat proses dari komputasi karena jumlah parameter yang akan diupdate semakin sedikit dan mengatasi *overfitting*. (Pratiwi, Cahyanti, & Lamsani, 2021).



Gambar 2. 8 Proses *Max Pooling*
Sumber : (Pratiwi, Cahyanti, & Lamsani, 2021)

2.5.3. Flatten Layer

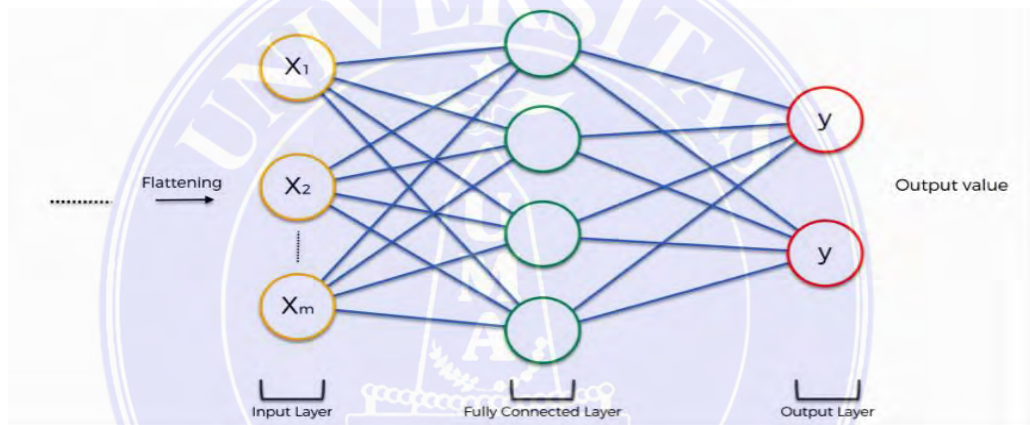
Flatten mempunyai arti berupa perataan. Perataan disini dimaksudkan pada suatu matriks atau *feature maps* yang diperoleh dari hasil *output* proses *convolution layer* dan *subsampling layer* untuk dilakukan perubahan ukuran menjadi bentuk larik (*array*) satu dimensi. *Flatten layer* ini juga merupakan bagian penting dalam model untuk melakukan tahapan klasifikasi. Dimana hasil dari proses *convolution layer* dan *subsampling layer* tidak dapat dilakukan oleh *fully connected layer*, tanpa melalui proses bernama *flatten layer*. (Antoko, Ridani, & Minarno, 2021)



Gambar 2. 9 Proses *Flatten Layer*
Sumber : (Antoko, Ridani, & Minarno, 2021)

2.5.4. Fully Connected Layer

Fully Connected Layer merupakan lapisan yang terdiri dari beberapa *neuron-neuron* yang saling tersambung satu dengan *neuron* yang lain seperti halnya pada jaringan saraf manusia. Nilai-nilai matriks yang sebelumnya telah diperoleh dari proses *convolution layer*, dan *subsampling layer* akan dimasukkan kedalam *fully connected layer* sebagai pengelanaan *input* pada layer ini. *Fully connected layer* pada dasarnya digunakan pada MLP yang bertujuan dalam melakukan transformasi data menjadi data tunggal atau satu dimensi agar data tersebut dapat diklasifikasi secara linear. Dimana transformasi data tersebut menggunakan proses *flatten layer*. (Ayu, Dwi, & Minarno, 2021).



Gambar 2. 10 *Fully Connected Layer*
Sumber : (Sulistyo Budi, Patmasari, & Saidah, 2021)

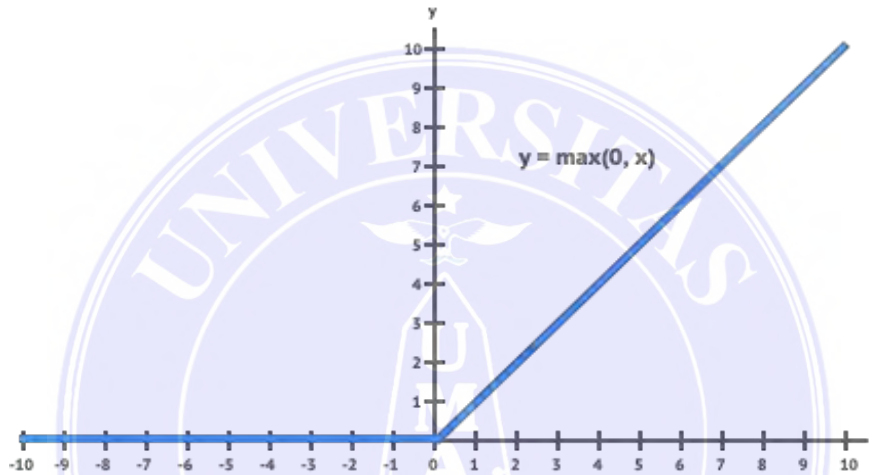
2.5.5. Fungsi Aktivasi

Fungsi aktivasi merupakan sebuah fungsi *non linear* yang memungkinkan suatu Jaringan syaraf tiruan untuk dapat mentransformasi data dari input menjadi dimensi yang lebih tinggi sehingga dapat dilakukan pemotongan *hyperlane* sederhana yang memungkinkan untuk dilakukannya klasifikasi. Dalam arsitektur CNN, penerapan dari fungsi aktivasi dilakukan setelah proses *convolution layer* dan sebelum terjadi proses *subsampling layer* untuk menghasilkan suatu pola fitur.

Terdapat beberapa fungsi aktivasi yang sering digunakan pada CNN, antara lain Relu dan *Softmax*.

a. *Rectified Linear Units* (RELU)

Rectified Linear Unit (RELU) merupakan fungsi aktivasi yang mempunyai kelebihan yaitu mampu memproses data dalam ukuran besar dengan cepat, yang dilakukan di antara *convolutional layer* dan *subsampling layer*. RELU menjaga hasil citra konvolusi pada domain definit positif, sehingga setiap nilai negatif yang berasal dari proses konvolusi akan melalui proses fungsi aktivasi RELU, dan mengubah nilai negatif tersebut sama dengan 0. (Sulistyo Budi, Patmasari, & Saidah, 2021)



Gambar 2. 11 Fungsi Aktivasi Relu

Sumber : (Antoko, Ridani, & Minarno, 2021)

b. *Softmax*

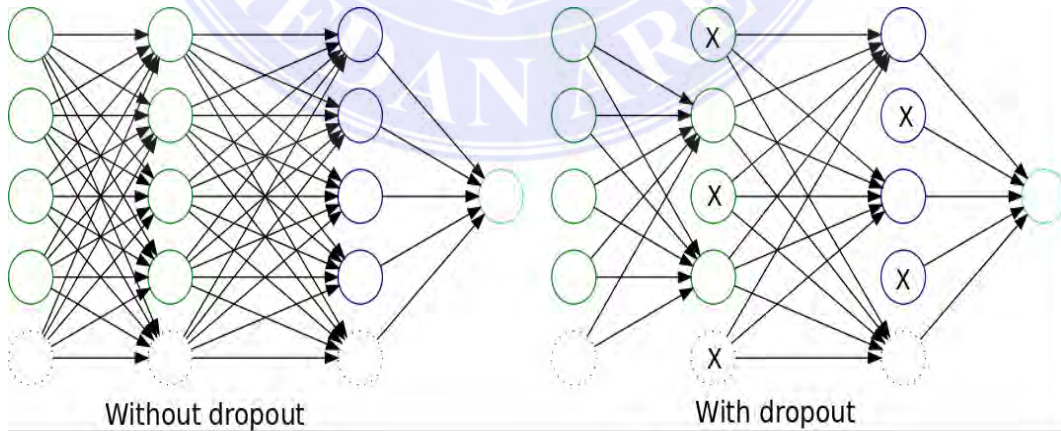
Softmax merupakan fungsi aktivasi yang mengubah nilai angka menjadi nilai *probability* dengan skala tertentu yang proporsional. Fungsi ini sering digunakan pada neural network yang memiliki kategori *output* yang banyak (*multi-class*). Hal tersebut dikarenakan fungsi aktivasi ini dapat mengubah nilai perhitungan menjadi nilai *probability* sehingga nilai hasil perhitungan bisa saling dibandingkan. Dengan menggunakan fungsi *softmax* dapat terlihat kelas mana yang mempunyai nilai kemungkinan terbesar. Dimana, kemungkinan terbesar tersebut akan menjadi kelas terpilih dan masukkan selanjutnya akan terklasifikasi menjadi kelas tersebut. (Dzaky, 2021)

2.6. Dropout

Dropout merupakan teknik *regularisasi* jaringan syaraf tiruan dimana beberapa neuron akan dipilih secara acak dan tidak akan digunakan selama proses pelatihan model jaringan. Neuron-neuron tersebut akan dihilangkan secara acak. Hal tersebut akan membuat neuron yang dihilangkan akan dihentikan sementara oleh model jaringan. *Dropout* bertujuan untuk mencegah terjadinya *overfitting* dan mengurangi kompleksitas atau jumlah parameter yang digunakan dalam suatu model sehingga akan mempercepat proses kerja dari pelatihan model. *Overfitting* adalah keadaan dimana hampir seluruh data yang telah melalui proses training mencapai persentase yang baik, tetapi terjadi ketidakselarasan pada saat melakukan proses prediksi.

Menghilangkan suatu neuron artinya akan menghilangkannya untuk sementara dari model jaringan yang ada. Tiap-tiap neuron akan diberikan probabilitas yang memiliki nilai antara 0 dan 1. Nilai tersebut dapat ditentukan sebelumnya ketika memberikan *dropout* pada *hidden layer*. (Antoko, Ridani, & Minarno, 2021)

Pada Gambar 2.12 dibawah ini, dapat dilihat perbedaan antara model jaringan syaraf biasa tanpa menggunakan *dropout* dengan model jaringan syaraf yang menggunakan *dropout* dimana terdapat beberapa neuron yang dilewati atau tidak digunakan lagi.



Gambar 2. 12 Cara Kerja Dropout
Sumber : (Antoko, Ridani, & Minarno, 2021)

2.7. Confusion Matrix

Confusion matrix merupakan metode yang dapat digunakan untuk melakukan pengukuran terhadap kinerja yang dimiliki oleh suatu metode klasifikasi dalam mengestimasi objek yang bernilai benar (*true*) atau yang bernilai salah (*false*). *Confusion matrix* pada umumnya mengandung informasi yang bisa dipakai untuk membandingkan hasil klasifikasi yang telah di ujicoba kan oleh sistem dengan hasil dari klasifikasi yang sebenarnya. (Astiningrum, Syulistyo, & Caesarla, 2020).

Confusion Matrix sangat berguna untuk mengevaluasi kinerja klasifikasi karena memberikan gambaran yang jelas tentang jumlah data yang diklasifikasikan dengan benar dan salah. Dari *Confusion Matrix*, beberapa metrik dapat dihitung seperti Akurasi (*Accuracy*), Presisi (*Precision*), *Recall* (Sensitivitas), dan *F1-Score*. (Uly, Hendry, & Iriani, 2023).

Pada *confusion matrix* terdapat beberapa istilah yang mendasar untuk melakukan penentuan dari nilai akurasi yaitu *true positive* (TP) adalah kondisi dimana model mengklasifikasikan data sebagai *true* dan jawaban aktualnya adalah *true*, *true negative* (TN) adalah kondisi dimana model mengklasifikasikan data sebagai *false* dan jawaban aktualnya adalah *false*, *false positive* (FP) adalah kondisi dimana model mengklasifikasikan data sebagai *true* dan jawaban aktualnya adalah *false*, dan *false negative* (FN) adalah kondisi dimana model mengklasifikasikan data sebagai *false* dan jawaban aktualnya adalah *true*. (Wulandari et al., 2020).

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 2.13 *Confusion Matrix*

2.7. Penelitian Terdahulu

Beberapa penelitian yang berhubungan mengenai klasifikasi rempah-rempah sebelumnya pernah dilakukan dengan menggunakan beragam metode. Detail dari penelitian terdahulu dapat dilihat pada Tabel 2.1.

Tabel 2.1 Penelitian Terdahulu

No	Peneliti	Judul	Keterangan	Hasil / Akurasi
1.	Kaharuddin et al (2019).	Klasifikasi Jenis Rempah-Rempah Berdasarkan Fitur Warna RGB Dan Tekstur Menggunakan Algoritma <i>K-Nearest Neighbor</i>	Pada penelitian ini proses pertama yang dilakukan yaitu melakukan <i>cropping</i> dan <i>cleansing</i> background pada citra. Selanjutnya citra rempah dikonversi menjadi <i>grayscale</i> dan dilakukan ekstraksi fitur tekstur dengan menghitung <i>energy</i> dan <i>homogeneity</i> . Hasil dari ekstraksi fitur tekstur yang didapatkan akan dilakukan tahap pengujian klasifikasi menggunakan Algoritma <i>K-Nearest Neighbor</i> dengan nilai $K=1$, $K=3$, dan $K=5$, serta memanfaatkan metode pengukuran jarak yaitu <i>Euclidean Distance</i> .	73%
2.	Jatmoko dan Sinaga (2020)	<i>K-Nearest Neighbor</i> Dan Ekstraksi Warna Mean RGB Untuk Identifikasi Kunyit Atau Temulawak	Tahap pertama citra terlebih dahulu di ekstraksi fitur warna untuk memperoleh nilai RGB yang kemudian dilakukan <i>Euclidean Distance</i> terhadap algoritma klasifikasi <i>K-Nearest Neighbor</i> . Hasil akurasi yang diperoleh 95% pada $K=1$	95%
3.	Batubara et al (2020).	Klasifikasi Rempah Rimpang Berdasarkan Ciri Warna RGB Dan Tekstur GLCM Menggunakan Algoritma <i>Naive Bayes</i>	dataset dari citra rempah rimpang diubah/dikonversi menjadi <i>Grayscale</i> dan selanjutnya akan dilakukan <i>segmentasi</i> dengan <i>thresholding</i> menggunakan metode <i>Otsu</i> . Kemudian hasil yang diperoleh dari dari proses <i>segmentasi</i> dilakukan RoI (<i>Region of Interest</i>) yang menghasilkan	52%

			perkalian nilai piksel. Dari perkalian nilai piksel akan dilakukan ekstraksi ciri dengan menggunakan GLCM dan ekstraksi fitur RGB. Setelah itu dilakukan klasifikasi menggunakan algoritma <i>Naïve Bayes</i> . Tahap terakhir yaitu uji performa dengan menggunakan <i>K-fold cross validation</i>	
4	(Windha & Agita, 2022)	Analisis Pengaruh Data Augmentasi Pada Klasifikasi Bumbu Dapur Menggunakan <i>Convolutional Neural Network</i>	Pengujian dilakukan dengan menggunakan 80 citra data <i>testing</i> yang terdiri dari citra yang diambil sendiri dan citra search internet.. Hasil dari penelitian ini adalah didapatkan model untuk mengklasifikasi bumbu dapur. Hasil pengujian menunjukkan CNN yang tidak menerapkan augmentasi dari model tersebut hanya mendapatkan akurasi sebesar 54% sedangkan CNN yang menerapkan augmentasi dari model tersebut mendapatkan akurasi sebesar 80%.	80%
5	(Tanuwijaya & Roseanne, 2021)	Modifikasi Arsitektur VGG16 untuk Klasifikasi Citra Digital Rempah-Rempah Indonesia	penelitian ini menggunakan sebuah model klasifikasi dengan <i>convolution neural network</i> seta arsitektur VGG 16 yang dimodifikasi. Untuk tahap latih model modifikasi VGG 16 ini menggunakan <i>dataset</i> rempah yang diperoleh dari Kaggle. Sedangkan validasi model yang digunakan yaitu berupa akurasi, <i>loss</i> , <i>precision</i> , dan <i>recall</i> untuk membandingkan model yang memiliki nilai yang terbaik. Hasil klasifikasi yang didapatkan dari VGG 16 rata-rata akurasi sebesar 81%, nilai <i>recall</i> sebesar 76%, dan nilai <i>precision</i> sebesar 81% untuk fase <i>training</i> dan untuk fase validasi, akurasi sebesar 85%, nilai <i>recall</i> sebesar 80%, dan nilai <i>precision</i> sebesar 84%	85%

6	(Wulandari et al., 2020)	Klasifikasi Citra Digital Bumbu Dan Rempah Dengan Algoritma <i>Convolutional Neural Network</i> (CNN)	Penelitian ini menggunakan <i>Convolutional Neural Network</i> dengan jumlah data citra sebanyak 300 citra. proses klasifikasi melalui tahapan <i>preprocessing</i> citra, perhitungan dengan CNN, dan perhitungan akurasi dengan <i>confusion matrik</i> . Pada model ini didapatkan nilai akurasi data <i>training</i> sebesar 98% dan nilai <i>loss</i> 0,0769. Nilai akurasi data <i>testing</i> sebesar 0,85 dan nilai <i>loss</i> 0,4773.	98%
---	--------------------------	---	---	-----



BAB III

METODOLOGI PENELITIAN

3.1 Alat dan Bahan Penelitian

Dalam penelitian ini, dibutuhkan peralatan dan bahan pendukung agar penelitian dapat berjalan dengan lancar. Adapun peralatan dan bahan pendukung dibagi menjadi perangkat keras dan perangkat lunak. Berikut merupakan perangkat keras dan perangkat lunak yang digunakan selama penelitian :

3.1.1 Perangkat Keras (*Hardware*)

Tabel 3. 1 Perangkat Keras (*Hardware*)

No	<i>Hardware</i>	Spesifikasi
1	Laptop	Samsung NP275E4V-K01ID
2	Processor	AMD E2-2000 APU 1.8 GHz
3	RAM	4 GB DDR3
4	Kamera	Smartphone Samsung A32

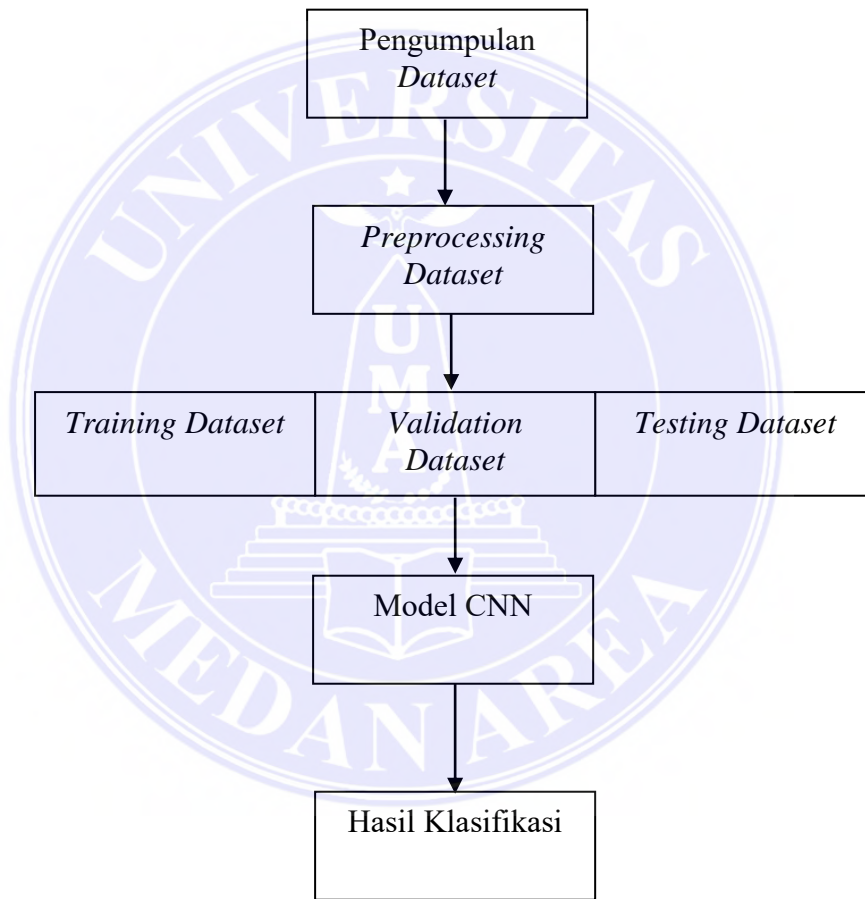
3.1.2 Perangkat Lunak (*Software*)

Tabel 3. 2 Perangkat Lunak (*Software*)

No	Software	Spesifikasi
1	Sistem Operasi	Windows 10 Pro 64-bit
2	Tools Pemograman	Google Colab - RAM : 12 GB - Disk : 107 GB
3	Bahasa Pemrograman	Python

3.2 Alur Metodologi Penelitian

Metodologi penelitian menggunakan algoritma *Convolutional Neural Network* yang terdiri dari pengumpulan *dataset*, *preprocessing dataset*, *training dataset* pada CNN, testing dataset pada CNN, dan perhitungan Hasil CNN yang bertujuan untuk mendapatkan hasil klasifikasi jenis rempah-rempah. Adapun tahapan atau langkah-langkah pada metodologi penelitian ini disajikan dalam bentuk alur penelitian sebagai berikut.

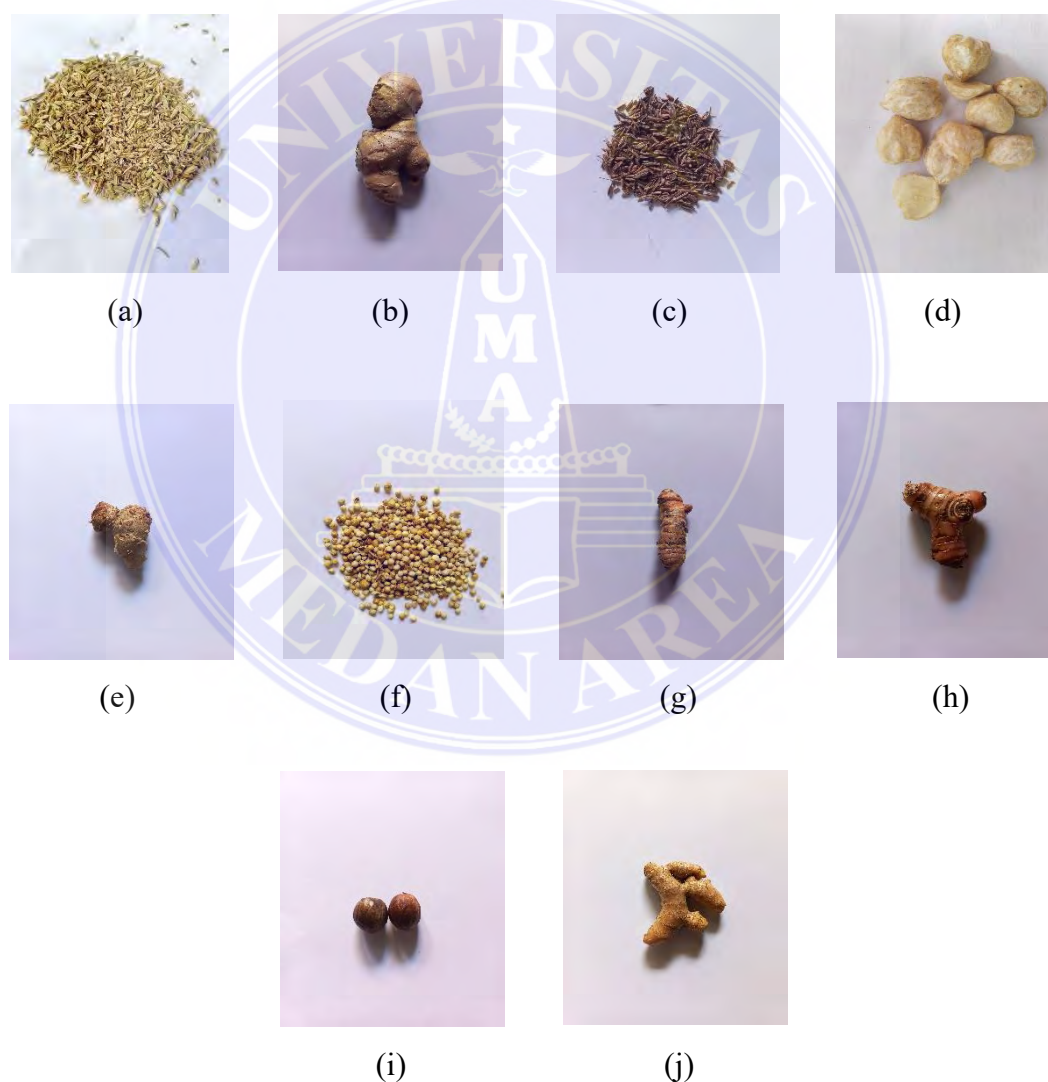


Gambar 3. 1 Alur Metodologi Penelitian

3.1. Pengumpulan *Dataset*

Dataset yang digunakan dalam penelitian ini adalah citra dari 10 jenis rempah-rempah yang diambil secara langsung dengan menggunakan bantuan dari kamera smartphone Samsung A32 dengan ekstensi Jpg. Keseluruhan citra rempah-rempah yang dijadikan dataset berjumlah 1500 citra atau gambar. Dimana citra rempah-rempah tersebut nantinya akan digunakan sebagai data *training*, data *validation* dan data *testing* pada proses klasifikasi.

Adapun sampel citra dataset dari 10 jenis rempah-rempah, disajikan pada Gambar 3.2 sebagai berikut.



Gambar 3. 2 Contoh *Dataset* Rempah (a) Adas Manis, (b) Jahe, (c) Jintan, (d) Kemiri, (e) Kencur, (f) Ketumbar, (g) Kunyit, (h) Lengkuas, (i) Pala, (j) Temulawak



3.2. Preprocessing Dataset

Sebelum melakukan proses klasifikasi menggunakan algoritma CNN, citra rempah-rempah terlebih dahulu akan dilakukan proses *preprocessing dataset*. *Preprocessing dataset* ini merupakan langkah awal yang memungkinkan untuk mendapatkan hasil citra yang lebih bagus dan maksimal. Berikut ini adalah proses *preprocessing dataset* yang dilakukan pada penelitian ini.

3.2.1. Resize

Resize merupakan tahap awal pada *preprocessing*, dimana citra akan dilakukan perubahan ukuran menjadi ukuran yang lebih kecil ataupun ukuran yang lebih besar dari ukuran citra yang awal atau sebenarnya. Tahap *resize* ini bertujuan untuk mempercepat proses klasifikasi dan menyeragamkan ukuran dari citra. Pada penelitian ini ukuran citra yang digunakan adalah 224x224 piksel. Pada Tabel 3.3 dapat dilihat perbedaan ukuran citra sebelum dilakukan *resize* dengan ukuran citra sesudah dilakukan *resize*, yaitu sebagai berikut.

Tabel 3. 3 *Resize*

Sebelum	Sesudah
	

3.2.2. Pembagian Dataset

Dataset yang telah diambil dan dikumpulkan, kemudian akan dilakukan pembagian data menjadi *training dataset*, *Validation Dataset*, dan *testing dataset* dengan perbandingan 60%:20%:20% dari data keseluruhan. Dimana 60% dari *dataset* akan digunakan untuk keperluan data *training*, 20% digunakan untuk data

testing dan 20% akan digunakan untuk keperluan data *validation*. Berikut pembagian data yang ditunjukkan pada Tabel 3.4.

Tabel 3. 4 Pembagian *Dataset*

Jenis Data	Jumlah Data
<i>Data Training</i>	960
<i>Data Testing</i>	300
<i>Data Validation</i>	240

3.3. Klasifikasi CNN

Setelah melakukan pengumpulan *dataset* dan *preprocessing* data, langkah selanjutnya adalah melakukan perhitungan dengan menggunakan algoritma CNN. Pada tahapan ini *training dataset* akan dilakukan pelatihan dengan menggunakan algoritma CNN yang bertujuan agar sistem dapat mengenali jenis dari rempah-rempah dalam pengujian sistem nantinya. Adapun tahap yang dilakukan adalah sebagai berikut.

3.3.1. Convolution Layer

Convolution layer merupakan proses pertama yang ada pada algoritma CNN yang bertujuan untuk menghasilkan *feature map*. Berikut ini adalah pembahasan untuk lapisan konvolusi.



Input Image



136	133	128	145	121
136	134	121	123	125
129	131	165	203	118
119	125	170	190	128
110	122	124	122	130

Sampel Matrik Citra

Gambar 3. 3 Sampel Data Input

Pada Gambar 3.3. merupakan *input image* yang digunakan sebagai *input* pada proses *convolution layer*. *Input image* tersebut diwakili oleh nilai-nilai piksel yang memiliki ukuran piksel 224 x 224. sehingga penulis hanya mengambil sebagian piksel saja, dengan ukuran 5 x 5 sebagai sampel. Pada penelitian ini juga digunakan matrik kernel 3 x 3. Pemilihan matrik kernel 3x3 dikarena terdapat pusat kernel yang akan diplot pada tiap index matriks citra masukkan. untuk proses perhitungan *convolution layer*. Adapun proses konvolusi dengan kernel dapat dilihat pada Gambar 3.4.

136	133	128	145	121
136	134	121	123	125
129	131	165	203	118
119	125	170	190	128
110	122	124	122	130

X

1	0	1
0	1	0
1	0	1

Gambar 3. 4 Proses Konvolusi Dengan Kernel 3 x 3

Matrik kernel 3 x 3 akan melakukan pergerakan dimulai dari sudut kiri atas dari bagian *input image* dengan menggunakan *stride* 1. Pergerakan akan berjalan terus dilakukan hingga seluruh piksel pada baris pertama selesai diproses. Selesai pada baris pertama pergerakan atau pergeseran tersebut akan dilanjutkan pada baris berikutnya hingga seluruh piksel citra diproses dan menjadi matrik baru. Pergeseran matrik dapat dilihat pada Gambar 3.5.

136	133	128	145	121
136	134	121	123	125
129	131	165	203	118
119	125	170	190	128
110	122	124	122	130

136	133	128	145	121
136	134	121	123	125
129	131	165	203	118
119	125	170	190	128
110	122	124	122	130



Gambar 3.5 Pergeseran Matrik

Pada setiap pergeseran piksel akan dilakukan perhitungan antara piksel dari *input image* yang dikalikan dengan matrik kernel. Dimana hasil dari perkalian tersebut akan menghasilkan *feature map*. Perhitungan dari pergeseran matrik Gambar 3.5 adalah sebagai berikut.

$$C_{1,1} = (136*1) + (133*0) + (128*1) + (136*0) + (134*1) + (121*0) + (129*1) + (131*0) + (165*1) = 692$$

$$C1,2 = (133*1) + (128*0) + (145*1) + (134*0) + (121*1) + (123*0) + (131*1) + (165*0) + (203*1) = 733$$

$$C1,3 = (128*1) + (145*0) + (121*1) + (121*0) + (123*1) + (125*0) + (165*1) + (203*0) + (118*1) = 655$$

$$C2,1 = (136*1) + (134*0) + (121*1) + (129*0) + (131*1) + (165*0) + (119*1) + (125*0) + (170*1) = 677$$

$$C2,2 = (134*1) + (121*0) + (123*1) + (131*0) + (165*1) + (203*0) + (125*1) + (170*0) + (190*1) = 737$$

$$C2,3 = (121*1) + (123*0) + (125*1) + (165*0) + (203*1) + (118*0) + (170*1) + (190*0) + (128*1) = 747$$

$$C3,1 = (129*1) + (131*0) + (165*1) + (119*0) + (125*1) + (170*0) + (110*1) + (122*0) + (124*1) = 653$$

$$C3,2 = (131*1) + (165*0) + (203*1) + (125*0) + (170*1) + (190*0) + (122*1) + (124*0) + (122*1) = 748$$

$$C3,3 = (165*1) + (203*0) + (118*1) + (170*0) + (190*1) + (128*0) + (124*1) + (122*0) + (130*1) = 727$$

Hasil perhitungan input image pada *convolution layer* terlihat pada Gambar 3.6.

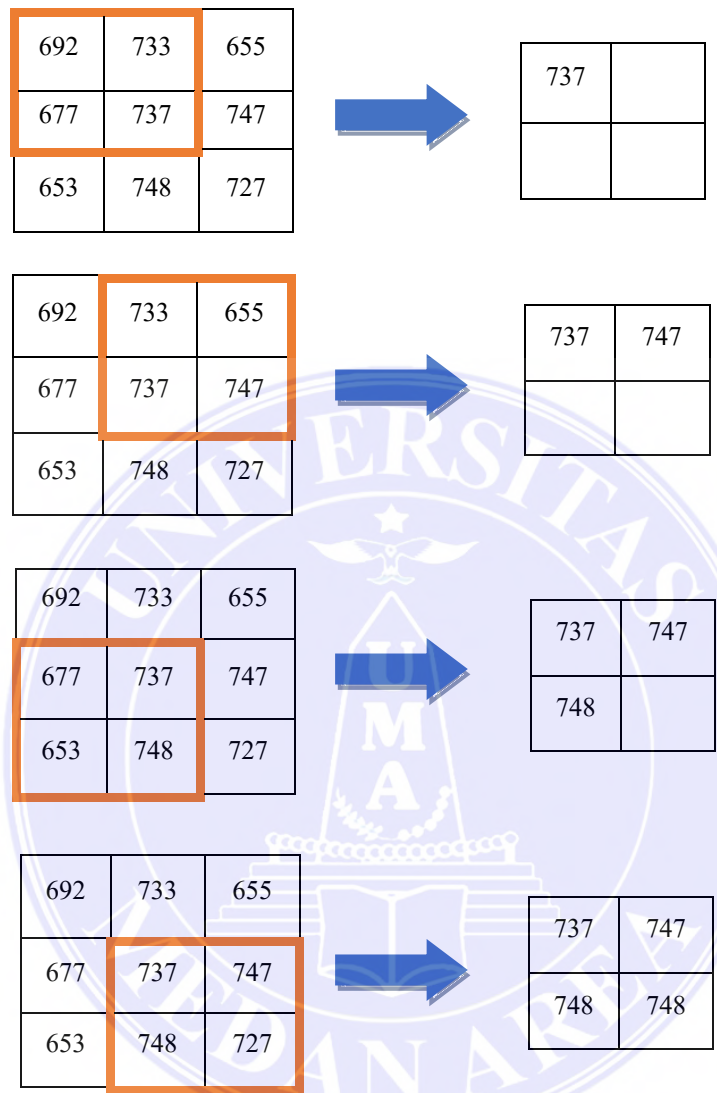
692	733	655
677	737	747
653	748	727

Gambar 3. 6 Hasil Perhitungan *Convolution Layer*

3.3.2. *Subsampling Layer*

Setelah melakukan proses *convolution layer*, selanjutnya akan dilakukan proses yang kedua yaitu *subsampling layer*. Pada *subsampling layer* hasil akhir dari proses *convolution layer* akan digunakan sebagai *input*. Pada lapisan ini, penulis menggunakan *stride* 1 dengan matrik kernel 2 x 2 dan juga menggunakan

metode *subsampling layer* yaitu *max pooling*. Adapun proses *max pooling* yang akan dilakukan dapat dilihat pada Gambar 3.6.



Gambar 3. 7 Proses *Max Pooling*

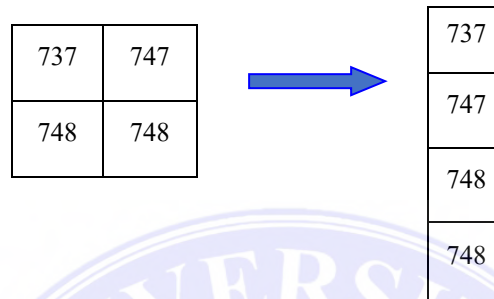
Dimana *Max pooling* berfungsi untuk mengambil nilai maksimal dari kernel 2 x 2, setelah berhasil mendapatkan nilai maksimal, maka kita akan mendapatkan matrik baru. Hasil akhir dapat dilihat pada Gambar 3.7 berikut ini.

737	747
748	748

Gambar 3. 8 Hasil *Max Pooling*

3.3.3. Flatten Layer

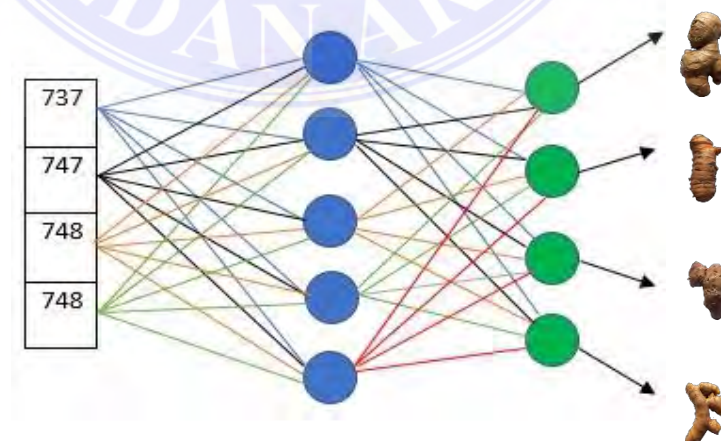
Hasil akhir dari *max pooling* akan diubah menjadi 1 vektor, dimana hasil dari *flatten layer* ini selanjutnya akan digunakan sebagai input pada proses berikutnya yaitu *fully connected layer*. Hasil dari *flatten layer* dapat dilihat pada Gambar 3.8.



Gambar 3. 9 Hasil Dari Flatten Layer

3.3.4. Fully Connected Layer

Setelah mendapatkan hasil *flatten layer*, selanjutnya akan dilakukan proses *fully connected layer*. Dimana *fully connected layer* disini akan berperan sebagai *hidden layer* yang merupakan lapisan yang dapat menentukan ke *output* manakah citra akan dikelompokkan. Gambar 3.9 merupakan tampilan umum dari *fully connected layer*



Gambar 3. 10 Fully Connected Layer

3.4. Inisialisasi Hyperparameter

Berikut ini adalah parameter yang digunakan pada penelitian kali ini.

Tabel 3. 5 Hyperparameter

Parameter	Value
<i>Epoch</i>	(15), (25)
<i>Batch Size</i>	(32), (64)
<i>Optimizer</i>	(RMSprop), (Adam)
<i>Learning Rate</i>	(0.001), (0.0001)

Tabel 4.1 diatas dapat dilihat *hyperparameter* yang digunakan pada proses penelitian yaitu *Epoch*, *Batch Size*, *Optimizer*, *Dropout* dan *Learning rate*. Pada parameter *Epoch* yang digunakan adalah *epoch* dengan nilai 15 dan 25. *Batch size* yang digunakan adalah 32 dan 64. *Dropout* yang digunakan adalah 0.2. Untuk *Learning rate* yang digunakan adalah 0,001 dan 0,009. Sedangkan untuk *Optimizer* dipilih *Optimizer* Adam dan RMSprop. Pemilihan *Optimizer* Adam dan *Optimizer* RMSprop ini didasarkan pada penelitian yang dilakukan oleh (Julianto, Sunyoto, & Wibowo, 2022) mendapatkan hasil yang sangat baik dengan nilai akurasi sebesar 97,56% untuk RMSprop dan 96,87% untuk Adam.

3.5. Metode Evaluasi

Setelah melakukan perhitungan dengan Algoritma *Convolutional Neural Network* (CNN). Selanjutnya hasil yang di dapatkan akan dilakukan metode evaluasi. Metode evaluasi bertujuan untuk mengetahui kinerja dari Algoritma *Convolutional Neural Network* (CNN). Metode evaluasi yang digunakan pada penelitian ini adalah *Confusion Matrix*.

3.5.1. Confusion Matrix

Confusion Matrix digunakan sebagai metode evaluasi yang digunakan untuk menghitung nilai kinerja algoritma CNN setelah didapatkan hasil pada perhitungan atau model dari CNN. Berikut adalah table dari Confusion Matrix yang ditampilkan pada Tabel dibawah ini.

Tabel 3. 6 Confusion Matrix

		Predicted	
		Negative (-)	Positive (+)
Actual	Negative (-)	True Negative (TN)	False Positive (FP)
	Positive (+)	False Negative (FN)	True Positive (TP)

Keterangan :

1. True Positive (TP), adalah kondisi dimana model mengklasifikasikan data sebagai TRUE dan jawaban aktualnya adalah TRUE.
2. True Negative (TN), adalah kondisi dimana model mengklasifikasikan data sebagai FALSE dan jawaban aktualnya adalah FALSE.
3. False Positive (FP), kondisi dimana model mengklasifikasikan data sebagai TRUE dan jawaban aktualnya adalah FALSE.
4. False Negative (FN), kondisi dimana model mengklasifikasikan data sebagai FALSE dan jawaban aktualnya adalah TRUE.

3.5.2. Pengukuran Kinerja

Pengukuran Kinerja (*Perfomance Measure*) merupakan langkah yang digunakan untuk menghitung kinerja algoritma. Dimana pengukuran kinerja ini dilakukan setelah mendapatkan hasil dari confusion matrix yang berupa nilai TP, TN, FP, dan FN. Pengukuran kinerja ini berisi *accuracy*, *precision*, *recall*, *f1-score*, dan *f2-score*. Berikut Rumus yang digunakan pada pengukuran kinerja.

A. Accuracy

Merupakan rasio prediksi Benar (positif dan negatif) dengan keseluruhan data. Akurasi menjawab pertanyaan “Berapa persen jahe yang benar diprediksi DO dan Tidak DO dari keseluruhan data jahe”

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} \dots\dots\dots(3.1)$$

B. Precision

Merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. Precision menjawab pertanyaan “Berapa persen jahe yang benar DO dari keseluruhan jahe yang diprediksi DO?”

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \dots\dots\dots(3.2)$$

C. Recall

Merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Recall menjawab pertanyaan “Berapa persen jahe yang diprediksi DO dibandingkan keseluruhan jahe yang sebenarnya DO”.

$$\text{Recall / Sensitivity} = \frac{\text{TP}}{\text{TP}+\text{FN}} \dots\dots\dots(3.3)$$

D. F1-Score

F1 Score merupakan perbandingan rata-rata presisi dan recall

$$\text{F1-Score} = 2x \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \dots\dots\dots(3.4)$$

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil penelitian ini dapat disimpulkan bahwa algoritma *Convolutional Neural Network* dalam mengklasifikasi citra rempah-rempah memperoleh hasil terbaik yang di dapatkan pada Model 12 dengan menggunakan *Epoch* 25, *Optimizer* RMSprop, *Batch Size* 32 dan *Learning Rate* 0.0001. Dimana hasil dari pengujian pada Model 12 ini mendapatkan nilai *Precision* 100%, *Recall* 100%, *F1-Score* 100%, *Accuracy* 100%, dan *Loss Accuracy* 0.0152. Sedangkan pada model 1 dan model 2 terdapat jenis rempah yang tidak terdeteksi sesuai kelas nya yaitu adas manis yang masuk pada kelas rempah jintan.

5.1 Saran

Adapun Saran dari penulis untuk pengembangan dalam penelitian yaitu sebagai berikut.

1. Membangun implementasi dari hasil model yang didapat ke dalam aplikasi, baik berbasis android ataupun berbasis website untuk mempermudah proses klasifikasi.
2. Menambahkan lebih banyak jumlah rempah-rempah yang dapat di klasifikasi.
3. Meningkatkan kualitas citra sehingga dapat dikenali lebih baik sesuai dengan kelas nya.
4. Pada objek rempah berukuran kecil sebaiknya dilakukan augmentasi pada citra seperti, zoom, cropping, rotasi, horizontal flip, vertikal flip, menambah noise dan gaussian blur

DAFTAR PUSTAKA

- Adrianto, L. B., Wahyuddin, M. I., & Winarsih, W. (2021). Implementasi *Deep Learning* Untuk Sistem Keamanan Data Pribadi Menggunakan Pengenalan Wajah dengan Metode *Eigenface* Berbasis Android. *Jurnal Informasi dan Komunikasi*, 89-96.
- Alwanda, M. R., Ramadhan, R. P., & Alamsyah, D. (2020). Implementasi Metode *Convolutional Neural Network* Menggunakan Arsitektur LeNet-5 untuk Pengenalan *Doodle*. *Algoritme*, 1, 45-56.
- Antoko, T. D., Ridani, M. A., & Minarno, A. E. (2021). Klasifikasi Buah Zaitun Menggunakan *Convolution Neural Network*. *Sistem Komputer*, 10, 119-126.
- Arkani, T. A., Magdalena, R., & Patmasari, R. (2020). Analisis Kepadatan Tulang Manusia Menggunakan Metode *Support Vector Machine*. *e-Proceeding of Engineering*, 7, 333-339.
- Astiningrum, M., Syulistyo, A. R., & Caesarla, A. K. (2020). Identifikasi Komponen GUI Pada Prototipe Aplikasi Mobile. *Jurnal Informatika Polinema*, 6, 51-56.
- Ayu, T., Dwi, V., & Minarno, A. E. (2021). Pendiagnosa Daun Mangga Dengan Model *Convolutional Neural Network*. *CESS*, 6, 230-235.
- Baihaqy, M., Wibowo, A. T., & Utama, D. Q. (2022). Klasifikasi Tanaman Anggrek Jenis *Phalaenopsis* Berdasarkan Labellum Bunga Menggunakan Metode *Convolutional Neural Network* (CNN). *e-Proceeding of Engineering*, 9, 1942-1951.
- Batubara, N. P., Widiyanto, D., & Chamidah, N. (2020). Klasifikasi rempah rimpang berdasarkan ciri warna rgb dan tekstur glcm menggunakan algoritma *naive bayes*. *Informatik: Jurnal Ilmu Komputer*, 16(3), 156. <https://doi.org/10.52958/iftk.v16i3.2196>
- Choldun R., M. I., & Surendro, K. (2018). Klasifikasi Penelitian Dalam *Deep Learning*. *Jurnal Ilmiah Manajemen Informatika*, 10, 25-33.
- Dzaky, A. T. (2021). Deteksi Penyakit Tanaman Cabai Menggunakan Metode *Convolutional Neural Network*. *e-Proceeding of Engineering*, 8, 3039-3055.
- Ellif, Sitorus, S. H., & Hidayati, R. (2021). Klasifikasi Kematangan Pepaya Menggunakan Ruang Warna Hsv Dan Metode *Naive Bayes Classifier*. *Jurnal Komputer dan Aplikasi*, 09, 66-75.
- Fitriyan, M., & Magriza, I. A. (2022). Rancang Bangun Aplikasi Penjualan Bahan Rempah-Rempah. 2(3), 1–13.
- Gultom, H., Sianturi, L. T., & Ndruru, E. (2021). Analisa Metode Mash-1 Untuk Mendeteksi Orisinalitas Citra Digital. *BULLETIN OF COMPUTER SCIENCE RESEARCH*, 1, 42-47.

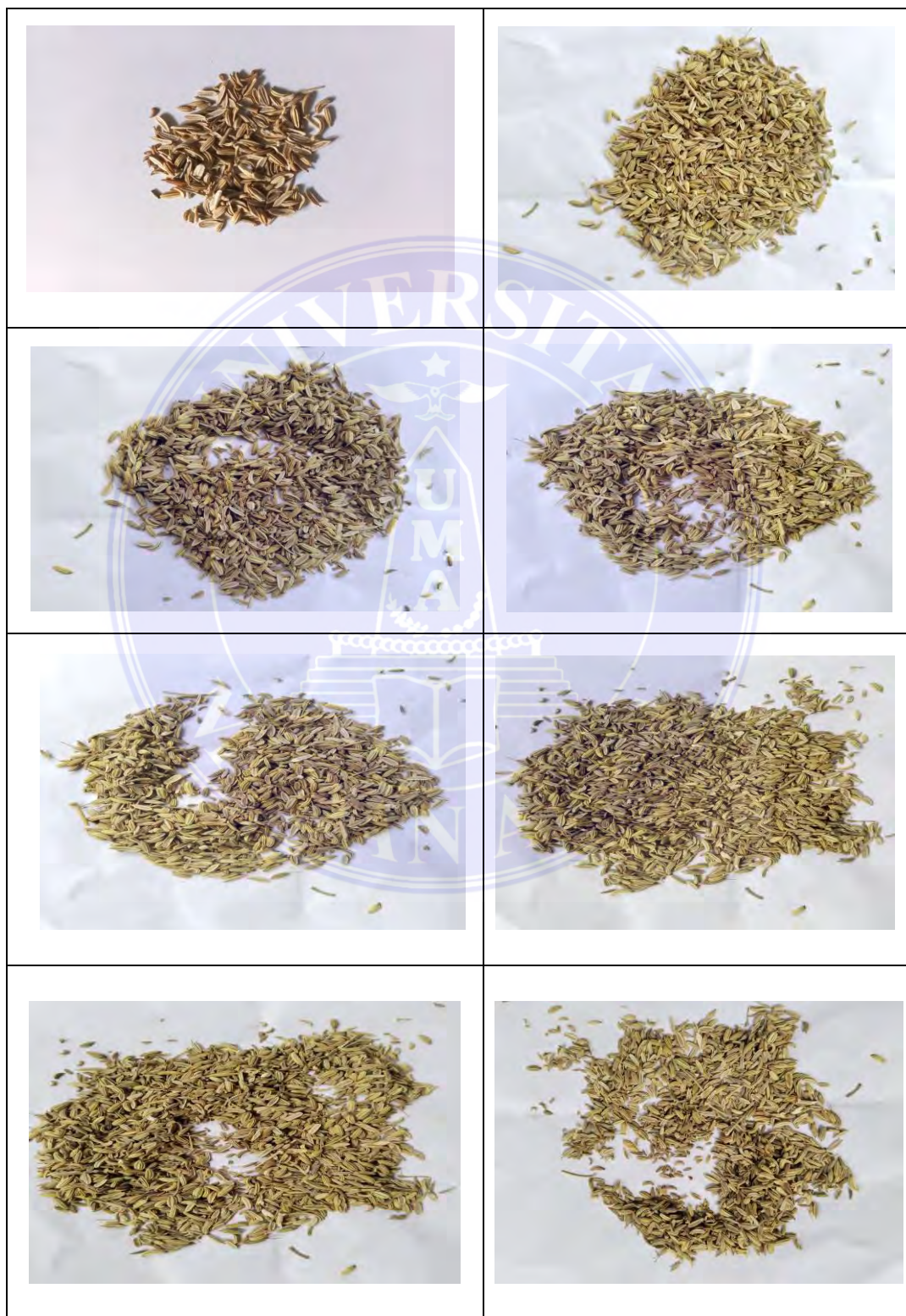
- Iskandar Mulyana, D., Ainur Rofik, M., & Ohan Zoharuddin Zakaria, M. (2022). Klasifikasi Kendaraan pada Jalan Raya menggunakan Algoritma *Convolutional Neural Network* (CNN). *Jurnal Pendidikan Tambusai*, 6(1), 1668–1679.
- Jatmoko, C., & Sinaga, D. (2020). *K-Nearest Neighbor* Dan Ekstraksi Warna Mean RGB Untuk Identifikasi Kunyit Atau Temulawak. 564–570.
- Kaharuddin, Kusri, & Luthfi, E. T. (2019). Klasifikasi Jenis Rempah-Rempah Berdasarkan Fitur Warna Rgb Dan Tekstur Menggunakan Algoritma *K-Nearest Neighbor*. *Jurnal Informasi Interaktif*, 4(1), 17–22.
- Matondang, I., Wahyuningsih, E., Rahayu, S. E., & Sinaga, E. (2005). *Katalog Tumbuhan Obat di Indonesia Zingiberaceae*. Jakarta.
- Nashr, M. H., Fachrurrozi, M., Triningsih, E., & Miraswan, K. J. (2020). Pengenalan Motif Kain Songket Pada Citra Kamera Smartphone Dengan Beragam Sudut Pandang Menggunakan Cnn. *Jurnal Ilmu Komputer Dan Teknologi Informasi*, 22-26.
- Ndruru, E. (2020). Kompresi File Citra Dengan Algoritma *Transformasi Walsh-Hadamard*. *Journal of Information Technology and Accounting*, IV(1), 1–7.
- Nugroho, P. A., Fenriana, I., & Arijanto, R. (2020). Implementasi *Deep Learning* Menggunakan *Convolutional Neural Network* (CNN) Pada Ekspresi Manusia. *Jurnal ALGOR*, 2, 12-21.
- Nurkhasanah, & Murinto. (2021). Klasifikasi Penyakit Kulit Wajah Menggunakan Metode *Convolutional Neural Network* Classification of Facial Skin Diseases Using the Method of the Convolutional Neural Network. *Sainteks*, 18(2), 183–190. <https://www.kaggle.com/datasets>.
- Nursyahfitri, R., Maharadja, A. N., Farissa, R. A., & Umaidah, Y. (2021). Klasifikasi Penentuan Jenis Obat Menggunakan Algoritma *Decision Tree*. *Jurnal Informatika Polinema*, 7(3), 53–60. <https://doi.org/10.33795/jip.v7i3.629>
- Pintanarum, R., Prasetiadi, A., & Ramdani, C. (2021). Klasifikasi Rasa Berdasarkan Citra Buah Menggunakan Algoritma *Convolutional Neural Network* Dengan Teknik Identitas Ganda. *IJIS*, 6, 79-88.
- Prabowo, D. A., Abdullah, D., & Manik, A. (2018). Deteksi Dan Perhitungan Objek Berdasarkan Warna Menggunakan *Color Objek Tracking*. *Jurnal Pseudocode*, V, 85-91.
- Pratiwi, H. A., Cahyanti, M., & Lamsani, M. (2021). Implementasi *Deep Learning Flower Scanner* Menggunakan Metode *Convolutional Neural Network*. *Sebatik*, 25, 124-130.
- Robi, Y., Masitoh, S., & Muflihati. (2019). Etnobotani Rempah Tradisional Di Desa Empoto Kabupaten Sanggau Kalimantan Barat. *Jurnal Hutan Lestari*, 130-142.

- Robianto, Sitorus, S. H., & Ristian, U. (2021). Penerapan Metode *Decision Tree* Untuk Mengklasifikasikan Mutu Buah Jeruk Berdasarkan Fitur Warna Dan Ukuran. *Jurnal Komputer dan Aplikasi*, 09, 76-86.
- R. Sulistyono Budi, R. Patmasari, & S. Saidah. (2021). Klasifikasi Cuaca Menggunakan Metode Convolutional Neural Network (CNN). *e-Proceeding of Engineering*, 8(5), 5047–5052.
- Septian, M. R. D., Paliwang, A. A. A., Cahyanti, M., & Swedia, E. R. (2020). Penyakit Tanaman Apel Dari Citra Daun Dengan *Convolutional Neural Network*. *Sebatik*, 24(2), 207–212.
<https://doi.org/10.46984/sebatik.v24i2.1060>
- Tanuwijaya, E., & Roseanne, A. (2021). Modifikasi Arsitektur VGG16 untuk Klasifikasi Citra Digital Rempah-Rempah Indonesia. *Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, 21, 189-196. doi: 10.30812/matrik.v21i1.1492.
- Wairata, C. R., Swedia, E. R., & Cahyanti, M. (2021). Pengklasifikasian Genre Musik Indonesia Menggunakan *Convolutional Neural Network*. *Sebatik*, 25, 255-261.
- Windha, M. P., & Agita, P. K. (2022). Analisis Pengaruh Data Augmentasi Pada Klasifikasi Bumbu Dapur Menggunakan Convolutional Neural Network. *Jurnal Media Informatika Budidarma*, 6, 2022-2031. doi:10.30865/mib.v6i4.4201
- Wita, D. S., & Liliana, D. Y. (2022). Klasifikasi Identitas Dengan Citra Telapak Tangan Menggunakan *Convolutional Neural Network* (CNN). 6(1), 1–7.
- Wulandari, I., Yasin, H., & Widiharih, T. (2020). Klasifikasi Citra Digital Bumbu Dan Rempah Dengan Algoritma *Convolutional Neural Network* (CNN). *Jurnal Gaussian*, 9(3), 273–282. <https://doi.org/10.14710/j.gauss.v9i3.27416>
- Yudianto, M. R., Kusriani, & Al Fatta, H. (2020). Analisis Pengaruh Tingkat Akurasi Klasifikasi Citra Wayang Dengan Algoritma *Convolutional Neural Network*. *Jurnal Teknologi Informasi*, 4, 182-190.
- Zakti, I. W., Fauzi, A., & Sihombing, A. (2022). Pengacakan Citra Digital Dengan Metode *Vertical Bit Rotation* (VBR) Memanfaatkan Algoritma *One Time Pad* (OTP) Sebagai Keamanan Biner. *Jurnal Teknik Informatika Kaputama (JTik)*, 6(1), 182-190.

LAMPIRAN

Lampiran 1. Dataset Rempah

A. Adas Manis



B. Jahe



C. Jintan



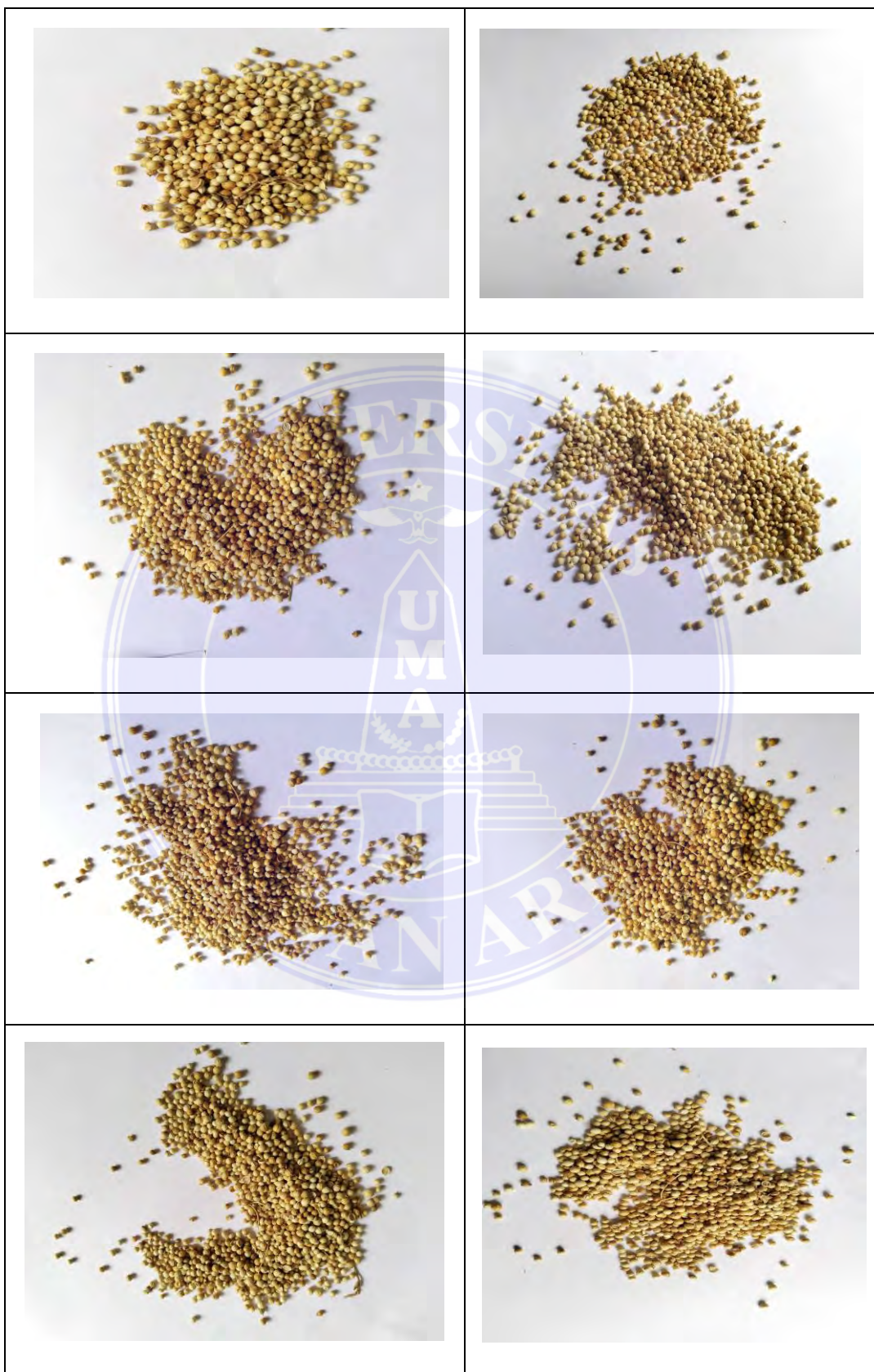
D. Kemiri



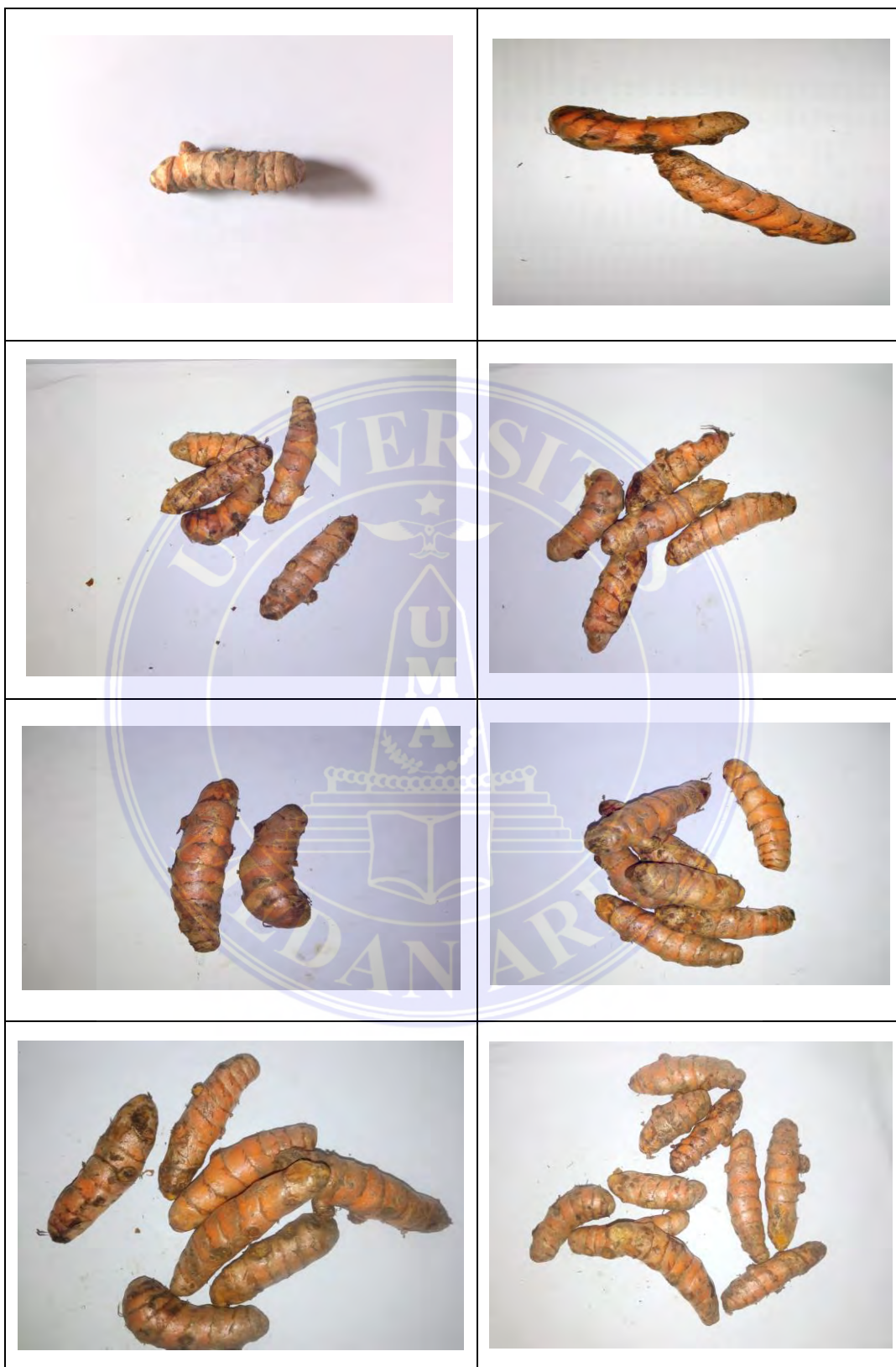
E. Kencur



F. Ketumbar



G. Kunyit



H. Lengkuas



I. Pala



J. Temulawak



Lampiran 2. Source Code Program

A. Import Library

```
import warnings
warnings.filterwarnings('always')
warnings.filterwarnings('ignore')
from tensorflow import keras
import tensorflow as tf
import pathlib
import PIL
from tensorflow.keras.preprocessing.image import load_img
from tensorflow.keras.preprocessing.image import
ImageDataGenerator
from tensorflow.keras.preprocessing import image
from tensorflow.keras.models import Sequential, Model
from tensorflow.keras.layers import GlobalAveragePooling2D, Dense,
BatchNormalization, Dropout, Flatten, Conv2D, MaxPooling2D
from tensorflow.keras.utils import img_to_array
from tensorflow.keras.optimizers import RMSprop
from tensorflow.keras.optimizers import Adam
from sklearn.model_selection import train_test_split
from skimage.io import imsave, imread
import numpy as np
import random
import matplotlib.pyplot as plt
%matplotlib inline
```

B. Menghubungkan Ke Gdrive

```
from google.colab import drive
drive.mount('/content/gdrive')
```

C. Memanggil directory dataset

```
im_shape = (224,224)
train_dir = '/content/gdrive/MyDrive/Rempah/Train'
test_dir = '/content/gdrive/MyDrive/Rempah/Test'
seed = 10
BATCH_SIZE = 32
BATCH_SIZE_1 = 64
```


D. Membagi jumlah dataset validasi dan test

```
data_generator = ImageDataGenerator(rescale=1./255,
validation_split=0.2)
val_data_generator = ImageDataGenerator(rescale=1./255,
validation_split=0.2)
```

E. Membagi Dataset sesuai kelas

```
train_generator = data_generator.flow_from_directory(train_dir,
target_size=im_shape, shuffle=True, seed=seed,
class_mode='categorical', batch_size=BATCH_SIZE,
subset="training")

validation_generator =
val_data_generator.flow_from_directory(train_dir,
target_size=im_shape, shuffle=False, seed=seed,
class_mode='categorical', batch_size=BATCH_SIZE,
subset="validation")

test_generator = ImageDataGenerator(rescale=1./255)
test_generator = test_generator.flow_from_directory
(test_dir, target_size=im_shape, shuffle=False, seed=seed,
class_mode='categorical', batch_size=BATCH_SIZE)

nb_train_samples = train_generator.samples
nb_validation_samples = validation_generator.samples
nb_test_samples = test_generator.samples
classes = list(train_generator.class_indices.keys())
print('Classes: '+str(classes))
print("Ukuran citra : {}".format(im_shape))
num_classes = len(classes)
```

F. Menampilkan sampel dataset rempah

```
image = imread
('/content/gdrive/MyDrive/Rempah/Train/AdasManis/Adas
Manis_001.jpg')
plt.figure(figsize=(5,5))
plt.imshow(image)
```

G. Mengatur Model Pengujian

```

model = Sequential()
model.add(Conv2D(20, kernel_size=(3, 3),
                activation='relu',
                input_shape=(im_shape[0],im_shape[1],3)))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Conv2D(40, kernel_size=(3,3), activation='relu'))
model.add(Flatten())
model.add(Dense(100, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(num_classes, activation='softmax'))
model.summary()

```

H. Model Compile

```

model.compile(loss='categorical_crossentropy',
              optimizer=Adam(learning_rate=0.001),
              metrics=['accuracy'])

```

I. Pengujian Model & Save Model terbaik

```

epochs = 15

#Callback to save the best model
callbacks_list = [
    keras.callbacks.ModelCheckpoint(
        filepath='model.h5',
        monitor='val_loss', save_best_only=True, verbose=1),
    keras.callbacks.EarlyStopping(monitor='val_loss',
patience=10,verbose=1)
]

#Training
cnn_history = model.fit(
    train_generator,
    steps_per_epoch=nb_train_samples // BATCH_SIZE,
    epochs=epochs,
    callbacks = callbacks_list,
    validation_data=validation_generator,
    verbose = 1,
    validation_steps=nb_validation_samples // BATCH_SIZE)

```

J. Menampilkan Grafik

```
def show_plots(cnn_history):
    loss_vals = cnn_history['loss']
    val_loss_vals = cnn_history['val_loss']
    epochs = range(1, len(cnn_history['accuracy'])+1)

    f, ax = plt.subplots(nrows=1,ncols=2,figsize=(15,6))

    # plot loss
    ax[0].plot(epochs, loss_vals, color='red',marker='o',
label='Training Loss')
    ax[0].plot(epochs, val_loss_vals, color='black', marker='*',
label='Validation Loss')
    ax[0].set_title('Training & Validation Loss')
    ax[0].set_xlabel('Epochs')
    ax[0].set_ylabel('Loss')
    ax[0].legend(loc='best')
    ax[0].grid(True)

    # plot accuracy
    acc_vals = cnn_history['accuracy']
    val_acc_vals = cnn_history['val_accuracy']

    ax[1].plot(epochs, acc_vals, color='red', marker='o',
label='Training Accuracy')
    ax[1].plot(epochs, val_acc_vals, color='black', marker='*',
label='Validation Accuracy')
    ax[1].set_title('Training & Validation Accuracy')
    ax[1].set_xlabel('Epochs')
    ax[1].set_ylabel('Accuracy')
    ax[1].legend(loc='best')
    ax[1].grid(True)

    plt.show()
    plt.close()

show_plots(cnn_history.history)
```

K. Memanggil Model

```
from tensorflow.keras.models import load_model

model = load_model('model.h5')
```

L. Menampilkan Hasil Validasi

```
score = model.evaluate_generator(validation_generator)
print('Val loss:', score[0])
print('Val accuracy:', score[1])
```

M. Menampilkan Hasil Test

```
score = model.evaluate_generator(test_generator)
print('Test loss:', score[0])
print('Test accuracy:', score[1])
```

N. Menampilkan Hasil Training

```
Train_loss, Train_accuracy = model.evaluate(train_generator)
print("Accuracy : {:.2f}%".format(Train_accuracy*100))
print("Loss      : {:.3f}".format(Train_loss))
```

O. Confusion Matrix

```
import itertools

#Plot the confusion matrix. Set Normalize = True/False
def plot_confusion_matrix(cm, classes, normalize=True,
title='Confusion matrix', cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """
    plt.figure(figsize=(10,10))
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
```

```

plt.xticks(tick_marks, classes, rotation=45)
plt.yticks(tick_marks, classes)
if normalize:
    cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
    cm = np.around(cm, decimals=2)
    cm[np.isnan(cm)] = 0.0
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]),
range(cm.shape[1]))):
        plt.text(j, i, cm[i, j],
                horizontalalignment="center",
                color="white" if cm[i, j] > thresh else "black")
plt.tight_layout()
plt.ylabel('True label')
plt.xlabel('Predicted label')

```

P. Classification Report

```

# Some reports
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
import sklearn.metrics
import numpy as np



#On test dataset
Y_pred = model.predict_generator(test_generator)
y_pred = np.argmax(Y_pred, axis=1)
target_names = classes

#Confution Matrix
cm = confusion_matrix(test_generator.classes, y_pred)
plot_confusion_matrix(cm, target_names, normalize=False,
title='Confusion Matrix')
print('Classification Report')
print(classification_report(test_generator.classes, y_pred,
target_names=target_names))


confusion_array =
sklearn.metrics.confusion_matrix(test_generator.classes, y_pred)

```

Lampiran 3. SK Pembimbing Tugas Akhir

	UNIVERSITAS MEDAN AREA FAKULTAS TEKNIK
Kampus I : Jalan Kolam Nomor 1 Medan Estate/Jalan PBSI Nomor 1 ☎ (061) 7366878, 7360168, 7364348, 7366781, Fax.(061) 7366998 Medan 20223	
Kampus II : Jalan Setiabudi Nomor 79 / Jalan Sei Serayu Nomor 70 A, ☎ (061) 8225602, Fax. (061) 8226331 Medan 20122	
Website: www.teknik.uma.ac.id E-mail: univ_medanarea@uma.ac.id	
Nomor : 319/FT.6/01.10/XI/2022	22 Nopember 2022
Lamp : -	
H a l : Perpanjang SK Pembimbing Tugas Akhir	
Yth. Pembimbing Tugas Akhir Muhathir, ST, M.Kom Rizki Muliono, S.Kom, M.Kom di Tempat	
Dengan hormat, Sehubungan telah berakhirnya waktu masa berlaku SK pembimbing nomor 113/FT.6/01.10/V/2022 tertanggal 9 Mei 2022 maka perlu diterbitkan kembali SK Pembimbing Skripsi baru atas nama mahasiswa berikut :	
N a m a : Loudji Zamico Pratama	
N P M : 178160010	
Jurusan : Informatika	
Oleh karena itu kami mengharapkan kesediaan saudara :	
1. Muhathir, ST, M.Kom	(Sebagai Pembimbing I)
2. Rizki Muliono, S.Kom, M.Kom	(Sebagai Pembimbing II)
Adapun Tugas Akhir Skripsi berjudul :	
"Klasifikasi Citra Rempah - Rempah dengan Menggunakan Algoritma <i>Convolutional Neural Network</i> (CNN)"	
SK Pembimbing ini berlaku selama enam bulan terhitung sejak SK ini diterbitkan. Jika proses pembimbing melebihi batas waktu yang telah ditetapkan, SK ini dapat ditinjau ulang.	
Demikian kami sampaikan, atas kesediaan saudara diucapkan terima kasih.	
 Dr. Rahmad Syah, S. Kom, M. Kom	

Lampiran 4. Surat Penelitian dan Pengambilan Data Tugas Akhir

 **UNIVERSITAS MEDAN AREA**
FAKULTAS TEKNIK

Kampus I : Jalan Kolam Nomor 1 Medan Estate/Jalan PBSI Nomor 1 ☎(061) 7366878, 7360168, 7364348, 7366781. Fax.(061) 7366998 Medan 20223
Kampus II : Jalan Setiabudi Nomor 79 / Jalan Sei Serayu Nomor 70 A, ☎ (061) 8225602, Fax. (061) 8226331 Medan 20122
Website: www.teknik.uma.ac.id E-mail: univ_medanarea@uma.ac.id

Nomor : 356 /FT.6/01.10/XII/2022 5 Desember 2022
Lamp : -
Hal : **Penelitian Dan Pengambilan Data Tugas Akhir**

Yth. Pimpinan UD. Sumber Rezeki
Pajak Pasar 3, Jalan Datuk Kabu Psr. 3 Tembung
Di
Deli Serdang

Dengan hormat,
Kami mohon kesediaan Bapak/Ibu berkenan untuk memberikan izin dan kesempatan kepada mahasiswa kami tersebut dibawah ini :



NO	N A M A	N P M	PRODI
1	Loudji Zamico Pratama	178160010	Teknik Informatika

Untuk melaksanakan Penelitian dan Pengambilan Data Tugas Akhir pada perusahaan/Instansi yang Bapak/Ibu Pimpin.

Perlu kami jelaskan bahwa Pengambilan Data tersebut adalah semata-mata untuk tujuan ilmiah dan Skripsi yang merupakan salah satu syarat bagi mahasiswa tersebut untuk mengikuti ujian sarjana lengkap pada Fakultas Teknik Universitas Medan Area dan tidak untuk dipublikasikan, dengan judul penelitian :


Klasifikasi Citra Rempah Rempah dengan Menggunakan Algoritma *Convolutional Neural Network* (CNN)

Atas perhatian dan kerja sama yang baik diucapkan terima kasih.


Dekan,

Dr. Rahmad Syah, S. Kom, M. Kom

Tembusan :
1. Ka. BAMAI
2. Mahasiswa
3. File

Lampiran 5. Surat Selesai Riset/ Penelitian

 **UD. SUMBER REZEKI**
Jl. Datuk Kabu, Pasar 3 Tembung
Deli Serdang

Deli Serdang, 16 Januari 2023

Hal : **Selesai Penelitian dan Pengambilan Data Tugas Akhir**

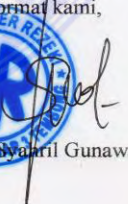
Kepada Yth. Dekan Fakultas Teknik
Universitas Medan Area
Di
Tempat


Dengan hormat,
Berdasarkan surat Dekan Fakultas Teknik Universitas Medan Area Nomor 356/FT.6/01.10/XII/2022 tanggal 5 Desember 2022, Perihal Penelitian dan Pengambilan Data Tugas Akhir di UD. Sumber Rezeki, maka bersama ini kami nyatakan bahwa mahasiswa di bawah ini:

Nama	: Loudji Zamico Pratama
NPM	: 178160010
Fakultas	: Teknik
Program Studi	: Teknik Informatika

Dinyatakan benar telah selesai melakukan penelitian dan pengambilan data di UD. Sumber Rezeki, sehubungan dengan penyusunan skripsi "**Klasifikasi Citra Rempah Rempah dengan Menggunakan Algoritma Convolutional Neural Network (CNN)**".

Demikian surat keterangan ini dibuat dengan sebenarnya dan dapat dipergunakan sebagaimana mestinya.

Deli Serdang, 16 Januari 2023
Hormat kami,

Syahril Gunawan



Lampiran 6. Plagiarism Tugas Akhir



Similarity Report ID: oid:29477:37360153

PAPER NAME	AUTHOR
Bab1-5_178160010_Loudji Zamico Prata ma.pdf	Loudji Zamico Pratama

WORD COUNT	CHARACTER COUNT
13152 Words	72279 Characters

PAGE COUNT	FILE SIZE
74 Pages	1.8MB

SUBMISSION DATE	REPORT DATE
Jun 12, 2023 8:50 AM GMT+7	Jun 12, 2023 8:52 AM GMT+7

● 25% Overall Similarity

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

- 19% Internet database
- 5% Publications database
- Crossref database
- Crossref Posted Content database
- 16% Submitted Works database

● Excluded from Similarity Report

- Small Matches (Less than 10 words)

Summary