

**KONVERSI LEMBAR JAWABAN UJIAN MAHASISWA
MENJADI DOKUMEN DIGITAL MENGGUNAKAN
CONVOLUTIONAL NEURAL NETWORK (CNN)
DAN LONG SHORT-TERM MEMORY (LSTM)**

SKRIPSI

**ALLIA RANIA
171402151**



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

**KONVERSI LEMBAR JAWABAN UJIAN MAHASISWA
MENJADI DOKUMEN DIGITAL MENGGUNAKAN
CONVOLUTIONAL NEURAL NETWORK (CNN)
DAN LONG SHORT-TERM MEMORY (LSTM)**

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat
memperoleh ijazah Sarjana Teknologi Informasi

ALLIA RANIA

171402151



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

PERSetujuan

Judul : KONVERSI LEMBAR JAWABAN UJIAN
MAHASISWA MENJADI DOKUMEN DIGITAL
MENGGUNAKAN *CONVOLUTIONAL NEURAL
NETWORK (CNN)* DAN *LONG SHORT-TERM MEMORY
(LSTM)*

Kategori : SKRIPSI

Nama : ALLIA RANIA

Nomor Induk Mahasiswa : 171402151

Program Studi : SARJANA (S1) TEKNOLOGI INFORMASI

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
SUMATERA UTARA

Medan, 04 Juli 2024

Komisi Pembimbing :

Pembimbing 2



Ivan Jaya, S.Si., M.Kom.
NIP. 198407072015041001

Pembimbing 1



Dr. Erna Budhiarti Nababan, M.IT.
NIP. 196210262017042001

Diketahui/Disetujui oleh

Program Studi S1 Teknologi Informasi

Ketua,




Dedy Arisandi, S.Si., M.Kom.
NIP. 197908312002121002

PERNYATAAN

KONVERSI LEMBAR JAWABAN UJIAN MAHASISWA MENJADI DOKUMEN
DIGITAL MENGGUNAKAN *CONVOLUTIONAL NEURAL NETWORK (CNN)* DAN
LONG SHORT-TERM MEMORY (LSTM)
SKRIPSI

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

Medan, 04 Juli 2024

Allia Rania

171402151

UCAPAN TERIMA KASIH

Puji syukur penulis kepada Allah S.W.T, atas izin dari-Nya penulis menyelesaikan skripsi sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer pada Program Studi S1 Teknologi Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.

Pada kesempatan ini, penulis ingin berterima kasih kepada:

1. Keluarga penulis, Alm.Baba Umar Ali dan Mama Rilly Yahwita Siregar yang telah membesarkan serta memberikan kasih sayang, doa dan dukungan selama pengerjaan skripsi, begitu juga kepada abang dan kakak penulis yaitu Bang Andre Rezky dan Kak Meliana Oktaviana serta ponakan tersayang Armo yang selalu memberikan dukungan terhadap penulis.
2. Bapak Dr. Muryanto Amin, S.Sos., M.Si selaku Rektor Universitas Sumatera Utara.
3. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
4. Bapak Dedy Arisandi, ST., M.Kom. selaku Ketua Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.
5. Ibu Dr. Erna Budhiarti Nababan, M.IT selaku Dosen Pembimbing I dan Pak Ivan Jaya, S.Si., M.Kom. selaku Dosen Pembimbing II yang telah banyak memberikan bimbingan, saran dan kritik yang memotivasi serta dukungan dalam proses penyempurnaan skripsi ini.
6. Dosen Penguji.
7. Seluruh Dosen Program Studi S1 Teknologi Informasi yang telah memberikan ilmu yang bermanfaat kepada penulis hingga dapat menunjang penyelesaian skripsi ini.
8. Staf dan pegawai Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara yang membantu urusan administrasi dalam menyelesaikan skripsi.
9. Sahabat penulis Ucca Artanti, Nurul Andini, Nadia Siti Namira dan Tria Riskiani yang selalu memberikan motivasi dan dukungan dalam penyelesaian skripsi ini.

10. Teman penulis Fani Theresa Hutabarat, Destri Celcylia Silitonga, Melati Yulvira Salsabila, Muhammad Bayhaqi Daulay.
11. Teman-teman angkatan 2017 Teknologi Informasi lainnya yang telah berjuang bersama penulis.
12. Kepada Widias yang telah memberikan bantuan, dukungan, dan semangat dalam proses penyelesaian penulisan skripsi ini.
13. Semua pihak yang terlibat langsung maupun tidak langsung yang tidak dapat penulis ucapkan satu persatu yang telah membantu penyelesaian skripsi ini.

Medan, 04 Juli 2024

Allia Rania

171402151

ABSTRAK

Aplikasi ini menyajikan proses mengenal kata yang dapat dilakukan dengan cara mengenali dan konversi kata dalam bentuk lembar jawaban dari dokumen teks (*hardcopy*) ke dalam bentuk dokumen file (*softcopy*) secara otomatis. Selanjutnya *preprocessing* dilakukan untuk mempermudah tahap segmentasi. Segmentasi citra merupakan salah satu tahapan *image processing* untuk membagi beberapa objek sesuai dengan kriteria awal yang ditentukan. Dengan metode *Optical Character Recognition* segmentasi citra dapat dilakukan. *Optical Character Recognition (OCR)* bertujuan mengidentifikasi suatu titik pada gambar berdasarkan tingkat kecerahan. Diterapkan *Convolutional Neural Network* dan *Long Short-Term Memory (LSTM)* yang merupakan metode untuk memproses informasi-informasi *template* gambar yang mencocokkan citra gambar. Pendeteksi citra yang masuk akan diuji apakah aplikasi dapat mengenal dengan jelas atau tidak. Hasil pelatihan akan dilakukan dengan validasi pada setiap masukan lembar jawaban. Berdasarkan hasil *output* dari aplikasi, dilakukan pengujian untuk melihat kecocokan antara tulisan di *hardcopy* dengan *output* tersebut. Hasilnya, kecocokan rata-rata sebesar 80% (dari total 90 data uji) sudah *match* dengan tulisan.

Kata Kunci: *Hardcopy, Softcopy, Optical Character Recognition (OCR), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), Tesseract Engine.*

**CONVERT STUDENT EXAM ANSWER SHEETS INTO DIGITAL DOCUMENTS
USING CONVOLUTIONAL NEURAL NETWORK (CNN) AND LONG SHORT-
TERM MEMORY (LSTM)**

ABSTRACT

This application presents the process of recognizing words that can be done by automatically recognizing and converting words in the form of answer sheets from text documents (hardcopy) into document files (softcopy). Furthermore, preprocessing is done to simplify the segmentation stage. Image segmentation is one of the stages of image processing to divide several objects according to predetermined criteria. With the Optical Character Recognition method, image segmentation can be performed. Optical Character Recognition (OCR) aims to identify a point in the image based on the brightness level. The application of Convolutional Neural Network and Long Short-Term Memory (LSTM) method is applied to process small information from images that match the image templates. The application will detect whether the entered image can be tested clearly or not. The results of the training will be carried out by validating each answer sheet input. Based on the tests that have been carried out, the results obtained from this application are an average of 80% of 90 data.

Keywords: *Hardcopy, Softcopy, Optical Character Recognition (OCR), Long Short-Term Memory (LSTM), Tesseract Engine.*

DAFTAR ISI

PERSETUJUAN	ii
PERNYATAAN	iii
UCAPAN TERIMA KASIH	iv
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian	4
1.7 Sistematika Penulisan	4
BAB 2 LANDASAN TEORI	6
2.1 Citra Digital	6
2.1.1 Citra Biner	6
2.1.2 Citra Grayscale	7
2.1.3 Citra Warna	7
2.2 Optical Character Recognition (OCR)	8
2.3 Tesseract	8
2.4 Piksel	9
2.5 Format Citra Digital	9

2.6	<i>Cropping</i>	10
2.7	<i>Resize</i>	10
2.8	<i>Thresholding</i>	11
2.9	<i>Convolutional Neural Network (CNN)</i>	11
2.9.1	<i>Convolutional Layer</i>	12
2.9.2	<i>Pooling Layer</i>	12
2.9.3	<i>Relu Layer</i>	13
2.9.4	<i>Fully Connected Layer</i>	14
2.10	<i>Long Short-Term Memory (LSTM)</i>	14
2.11	Penelitian Terdahulu	17
BAB 3 ANALISIS DAN PERANCANGAN SISTEM		20
3.1	<i>Dataset yang digunakan</i>	20
3.2	Analisis Sistem	22
3.3	<i>Preprocessing</i>	24
3.4	CNN	27
3.5	LSTM	32
3.6	Perancangan Antarmuka Sistem	33
3.7	<i>Confusion Matrix</i>	37
BAB 4 IMPLEMENTASI DAN PENGUJIAN SISTEM		39
4.1	Implementasi Sistem	39
4.1.1	<i>Spesifikasi Perangkat Keras dan Lunak</i>	39
4.1.2	<i>Perancangan Antarmuka</i>	40
4.2	Prosedur Operasional	43
4.3	Implementasi Model	46
4.4	Pengujian Sistem	47
4.5	Evaluasi	57
BAB 5 KESIMPULAN DAN SARAN		58
5.1	Kesimpulan	58

5.2	Saran	58
	DAFTAR PUSTAKA	59

DAFTAR TABEL

Tabel 2.1. Penelitian Terdahulu	18
Tabel 2.2. Penelitian Terdahulu	19
Tabel 3.1. Pembagian <i>Dataset</i>	21
Tabel 3.2. Model LSTM	32
Tabel 4.1. Epoch Penelitian	47
Tabel 4.2. Pengujian Sistem	48
Tabel 4.3. Pengujian Sistem (Lanjutan)	49
Tabel 4.4. Pengujian Sistem (Lanjutan)	50
Tabel 4.5. Pengujian Sistem (Lanjutan)	51
Tabel 4.6. Pengujian Sistem (Lanjutan)	52
Tabel 4.7. Pengujian Sistem (Lanjutan)	53
Tabel 4.8. Pengujian Sistem (Lanjutan)	54
Tabel 4.9. Pengujian Sistem (Lanjutan)	55
Tabel 4.10. Pengujian Sistem (Lanjutan)	56

DAFTAR GAMBAR

Gambar 2.1 Citra Biner	6
Gambar 2.2 Citra Grayscale	7
Gambar 2.3 Citra Warna	7
Gambar 2.4 Arsitektur Convolutional Neural Network	12
Gambar 2.5 Pooling	13
Gambar 2.6 Struktur Jaringan LSTM	14
Gambar 3.1. Contoh Lembar Jawaban	20
Gambar 3.2. Arsitektur Umum	23
Gambar 3.3. Bagian Cropping	24
Gambar 3.4. Perhitungan Threshold	25
Gambar 3.5. Nilai Threshold	27
Gambar 3.6. Nilai Citra Masukan	27
Gambar 3.7. Penggunaan padding dan <i>kernel</i> 3x3	28
Gambar 3.8. Pergeseran <i>Kernel</i>	29
Gambar 3.9. Hasil Perhitungan	31
Gambar 3.10. Hasil Perhitungan <i>Max Pooling</i>	31
Gambar 3.11. Hasil Proses <i>Flattening</i>	32
Gambar 3.12. Menu Utama	34
Gambar 3.13. Proses Nilai	35
Gambar 3.14. Data Mahasiswa	36
Gambar 3.15. Train Model	37
Gambar 3.16. Confusion Matrix	38
Gambar 4.1. Fitur Home	40
Gambar 4.2. Mulai Proses	41
Gambar 4.3. Daftar Mahasiswa	42
Gambar 4.4. Hasil <i>Export</i>	42
Gambar 4.5. <i>Train Model Preprocessing</i>	43
Gambar 4.6. Proses Nilai Individu	44

Gambar 4.7. Input Lembar Jawaban	44
Gambar 4.8. Proses Individu Lembar Jawaban	45
Gambar 4.9. Proses Keseluruhan Lembar Jawaban	46
Gambar 4.10. Contoh Lembar Jawaban Gagal	57

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Teknologi berkembang secara cepat di era modern saat ini. Salah satu bukti berkembangnya teknologi yaitu dirancangnya komputer. Sama seperti manusia, komputer juga memiliki kecerdasan sehingga komputer sangat berguna untuk membantu sebagian pekerjaan manusia dan mempermudah urusan manusia. Dengan demikian, komputer bermanfaat untuk melakukan efisiensi dalam pekerjaan.

Akan tetapi, masifnya penggunaan teknologi dapat menjadi kendala tersendiri apabila tidak dilakukan juga dengan beberapa pendekatan sistematis yang meningkatkan efisiensi. *Output* dari penggunaan teknologi ini masih banyak yang menggunakan kertas/*hardcopy* sebagai media penyimpanan dan pencetakan. Melihat perkembangan tersebut, semakin banyak dokumen digital maupun *hardcopy* yang beredar baik secara daring maupun luring. Hasil akhir berupa *file softcopy* untuk mempermudah proses rekapitulasi.

Untuk memudahkan rekapitulasi dan efisiensi pembacaan pada dokumen *hardcopy*, perlu suatu metode pembacaan yang dapat menerjemahkan hasil pembacaan yang nantinya akan disimpan ke dalam *database*. Terdapat banyak keuntungan dalam kecanggihan teknologi yang berkembang dengan cepat. Manfaat yang dapat diambil salah satunya untuk mengurangi biaya dan mempermudah akses informasi. Disebutkan perangkat keras yang biasa digunakan adalah kamera yang ada pada *smartphone* dapat menggantikan pengguna *scanner* yang ukurannya jauh lebih besar. Fungsi *scanner* memiliki kegunaan yang sama dengan kamera seperti mendeteksi dan melakukan pembacaan secara menyeluruh pada sebuah dokumen dalam bentuk citra digital.

Penelitian seperti pada Firdaus *et al.* (2021) yang melakukan penelitian tentang pembacaan citra digital menyampaikan bahwa berada di masa pandemi memakan banyak waktu dikarenakan banyak dokumen fisik yang harus diproses dengan cepat dan diolah dengan aplikasi secara berulang apabila dokumen tersebut ternyata hilang, rusak bahkan tidak memiliki salinan lainnya. Dengan adanya *Optical Character Recognition* (OCR) hal

ini teratasi baik karena OCR dapat konversi huruf atau gambar menjadi karakter ASCII yang mudah dikenal komputer.

Penulis mencoba mendapatkan solusi atas permasalahan yang sudah diidentifikasi melalui penyajian program OCR agar mempermudah proses pembacaan dokumen. Dokumen yang hilang, rusak atau bahkan tidak ada salinan lainnya dapat berhasil dipindai oleh OCR sehingga mempercepat proses ketikan juga pengolahan dokumen.

Penelitian dilakukan dengan OCR untuk memproses pengenalan karakter. Mesin OCR ini akan diterapkan pada pembacaan dokumen *hardcopy* untuk mendapatkan dokumen *softcopy* dari dokumen *hardcopy* yang tersedia lalu memasukkannya ke dalam *database*.

Pada penelitian ini, penulis menggunakan dua arsitektur utama dalam bidang kecerdasan buatan / artificial intelligence (AI) yang telah merevolusi berbagai aplikasi seperti pengenalan gambar, pemrosesan bahasa alami, dan analisis waktu yaitu *Convolutional Neural Networks* (CNN) dan *Long Short-Term Memory* (LSTM).

CNN dirancang khusus untuk memproses data grid seperti gambar dengan memanfaatkan operasi konvolusi untuk mengekstraksi fitur-fitur penting secara hierarkis. Kemampuannya dalam menangkap pola spasial menjadikan CNN ideal untuk tugas pengenalan gambar. Sementara itu, LSTM adalah jenis jaringan saraf rekuren (RNN) yang memungkinkan pemrosesan data urutan dengan mengatasi tantangan seperti vanishing gradient dan long-term dependencies.

Dengan mempertahankan memori internal yang kuat, LSTM telah mengungguli RNN tradisional dalam pemrosesan teks, prediksi waktu, dan tugas-tugas lain yang melibatkan data urutan.

Kombinasi dari kedua arsitektur ini, seperti CNN-LSTM, telah digunakan secara luas untuk menggabungkan keunggulan pengenalan pola spasial dan temporal dalam berbagai aplikasi, membuka peluang baru dalam pemodelan dan analisis data kompleks.

Berdasarkan permasalahan dan latar belakang yang terkait, maka penulis memberikan judul penelitian yaitu “Konversi Lembar Jawaban Ujian Mahasiswa Menjadi Dokumen Digital Menggunakan *Convolutional Neural Network* (CNN) dan *Long Short-Term Memory* (LSTM)”.

1.2 Rumusan Masalah

Banyak dokumen hilang dan rusak yang tidak berhasil dipindai sehingga dibutuhkan proses otomatisasi untuk menjadikan dokumen digital.

1.3 Tujuan Penelitian

Mengkonversi lembar jawaban ujian dengan menggunakan algoritma *Convolutional Neural Network* dan *Long Short-Term Memory*.

1.4 Batasan Masalah

Beberapa batasan permasalahan terhadap penelitian ini adalah sebagaimana berikut:

1. Pengenalan tulisan yang diuji adalah huruf A-Z dan a-z yang merupakan huruf latin dan angka (0-9).
2. Kertas yang digunakan untuk penulisan tangan dilakukan diatas kertas yang berwarna putih dengan tulisan berwarna hitam.
3. Citra tulisan yang dipakai dalam bentuk format *.jpg*, *.jpeg* atau *.png*.
4. *Template file*, ukuran dan posisi cetakan lembar jawaban sudah ditentukan.
5. Hanya mengambil bagian header kertas yang ada Nama, NIM dan Nilai.
6. Data gambar berukuran maksimal 1 Mb (Megabyte).
7. Data nama-nama mahasiswa harus disimpan dalam *database*.
8. *Output* berupa Nama, NIM, Mata Kuliah dan Nilai yang akan disimpan dalam bentuk *.csv*.
9. Penulisan nilai harus kapital dan NIM harus jelas serta tidak boleh tulisan sambung pada bagian Nama dan NIM.

1.5 Manfaat Penelitian

Adapun pada penelitian ini penulis berharap dapat memberikan keuntungan dalam bentuk:

1. Meningkatkan efisiensi dalam melakukan rekapitulasi informasi terhadap dokumen *hardcopy* menjadi satu kesatuan informasi.
2. Meningkatkan efektivitas proses *scanning* dokumen *hardcopy*.

1.6 Metodologi Penelitian

Berikut ini merupakan prosedur yang perlu diikuti dalam melaksanakan penelitian:

1. Studi Literatur

Pengumpulan informasi ini dilakukan dengan kaitan *Optical Character Recognition* (OCR), *Convolutional Neural Network*, *Long Short-Term Memory (LSTM)* seperti jurnal, skripsi dan sebagainya.

2. Analisis Permasalahan

Di dalam tahap ini, kami mencoba melakukan \ analisis terhadap informasi yang didapatkan dalam rangka melakukan penyelesaian terhadap masalah konversi informasi di dalam dokumen *hardcopy* menjadi bentuk *softcopy*.

3. Perancangan Sistem

Perancangan terhadap tahapan ini berupa arsitektur sistem, pengumpulan data terhadap permasalahan yang sudah dianalisis.

4. Implementasi Sistem

Tahap ini permasalahan dan perancangan dilakukan agar solusi dapat ditemukan berdasarkan hasil yang didapatkan. Rancangan tersebut menjadi sistem yang siap dipakai.

5. Pengujian Sistem

Sistem akan diuji dan dieksperimenkan untuk mengevaluasi sejauh mana tingkat akurasi yang telah dicapai sebagaimana yang diekspektasikan.

6. Dokumentasi serta Pembentukan Laporan

Dokumentasi menyeluruh terhadap pengerjaan serta penulisan laporan didasarkan pada hasil penelitian berupa program yang sudah dibentuk dan dijalankan.

1.7 Sistematika Penulisan

Pada bagian ini, penulis mencoba menjabarkan struktur di dalam skripsi ini yang terdiri dari 5 (lima) bagian yaitu:

Bab 1 Pendahuluan

Bab ini mencakup latar belakang penelitian, perumusan masalah, tujuan penelitian, batasan

masalah, manfaat penelitian, metodologi penelitian dan struktur penulisan.

Bab 2 Landasan Teori

Bab ini mengulas teori-teori yang mendukung skripsi, yang digunakan untuk menganalisis dan memahami permasalahan yang akan dipecahkan. Teori-teori yang diterapkan dalam penelitian ini meliputi *Optical Character Recognition (OCR)*, *Convolutional Neural Network (CNN)* dan *Long Short-Term Memory (LSTM)*.

Bab 3 Analisis dan Perancangan Sistem

Bab ini memuat informasi tentang perancangan sistem/model berdasarkan algoritma untuk mengenali tulisan tangan yang ditulis dalam kertas. Analisis dilakukan berdasarkan input huruf/angka yang digunakan untuk membangun model/sistem.

Bab 4 Implementasi dan Pengujian Sistem

Bab ini mencakup pelaksanaan implementasi sistem yang telah dirancang berdasarkan hasil yang telah dibahas pada Bab 3. Selain itu, sistem juga diuji untuk mengevaluasi kelebihan dan kekurangan yang ada pada sistem yang telah dibangun.

Bab 5 Kesimpulan dan Saran

Bab ini memuat ringkasan hasil penelitian dari Bab 3 dan 4. Selain itu, terdapat juga rekomendasi untuk penelitian lanjutan.

BAB 2

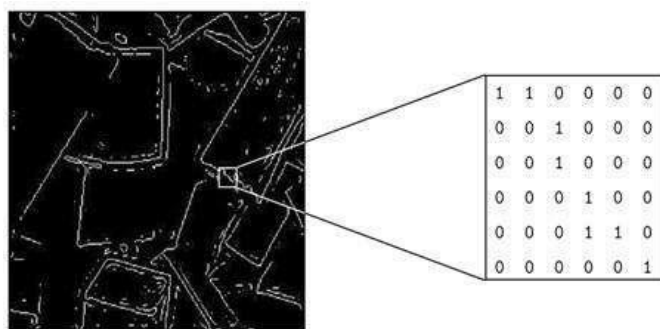
LANDASAN TEORI

2.1 Citra Digital

Image atau citra secara harfiah yaitu gambar yang memiliki dua dimensi. Pada saat yang sama, dari sudut pandang sistematis, gambar dua dimensi memiliki intensitas cahaya yang berupa fungsi kontinu dari gambar atau citra. Sehingga citra data tersimpan ke dalam memori perangkat komputer yang mengubah menjadi format citra digital. Mengkonversi citra ke format digital dapat berupa *scanner*, kamera digital dan *handycam*. Berbagai proses pengolahan citra yang dapat dilakukan setelah citra diubah menjadi citra digital.

2.1.1 Citra Biner

Setiap piksel membutuhkan sekitar 1bit memori yang menghasilkan nilai intensitas menjadi dua, yaitu 0 atau 1 seperti pada Gambar 2.1.

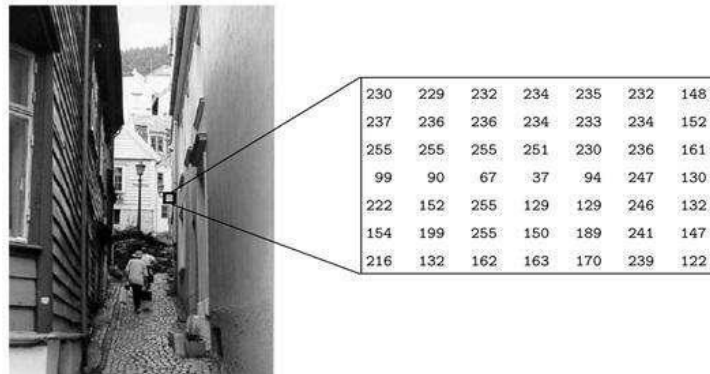


Gambar 2.1 Citra Biner

(<https://catatanpeneliti.wordpress.com/>)

2.1.2 Citra Grayscale

Nilai-nilai mewakili setiap matriks citra data yang intensitas setiap piksel berupa 0 sampai dengan 255. Dibutuhkan 8bit memori pada setiap piksel seperti diberikan dapat dilihat Gambar 2.2.

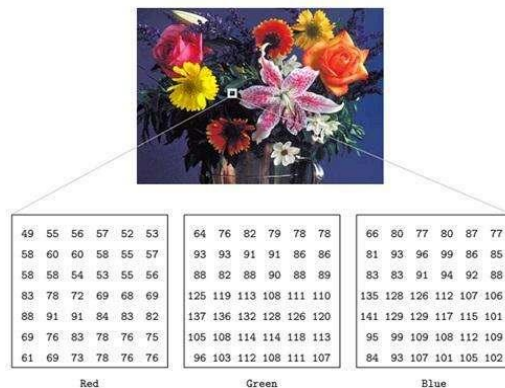


Gambar 2.2 Citra Grayscale

(<https://catatanpeneliti.wordpress.com/>)

2.1.3 Citra Warna

Piksel mempunyai 3 (tiga) komponen utama warna pada citra yang spesifik, yaitu merah – *red*, hijau – *green* dan biru – *blue*. 3 (tiga) komponen warna sebelumnya masing-masing memiliki 8 bit juga dapat menyimpan citra warna menjadi 24 bit seperti diberikan pada Gambar 2.3.



Gambar 2.3 Citra Warna

(<https://catatanpeneliti.wordpress.com/>)

2.2 Optical Character Recognition (OCR)

Citra data huruf dapat diidentifikasi menjadi sebuah *file* tertulis pada sebuah aplikasi yang disebut sebagai *Character Pattern Recognition* atau yang lebih dikenal *Optical Character Recognition (OCR)*. Sistem komputer ini dapat memahami kata melalui tulisan tangan ataupun *printer*. Dengan pengenalan huruf ini, kecerdasan komputer dapat ditingkatkan. OCR menghasilkan proses teks yang berupa citra gambar masukan dengan akurasi yang bergantung pada kejelasan gambar atau metode yang digunakan.

Cara kerja OCR (*Optical Character Recognition*) terdiri dari:

1. *Preprocessing*

Menghapus bagian yang diidentifikasi tidak signifikan pada *input* untuk melakukan tahapan selanjutnya.

2. *Clusterization*

Clusterization menghitung urutan karakter yang tepat dalam sebuah teks juga berfungsi menemukan karakter dalam setiap gambar individu.

3. *Feature Extraction*

Tujuan dari *Feature Extraction* adalah memperoleh karakter unik melalui karakter gambar individu sehingga mengenali klasifikasi tersebut sendiri.

4. *Classification*

Pada tahap terakhir dari OCR (*Optical Character Recognition*) untuk mengenali setiap karakter yang berasal dari informasi tentang *Feature Extraction* sebelumnya.

2.3 Tesseract

Menurut Smith, *Tesseract Optical Character Recognition (OCR)* adalah mesin optik yang mengenal sebuah karakter. *Tesseract Optical Character Recognition (OCR)* adalah mesin yang saat ini sedang dikembangkan oleh Google. Tesseract pertama dikembangkan oleh Hewlett Packard (HP) diantara tahun 1985 sampai 1995 (Smith, 2007). Melewati tahun 2006, HP merilis Tesseract dapat digunakan secara bebas. Mulai saat itu, Tesseract diambil

alih oleh Google untuk dikembangkan secara ekstensif dan merilis lisensi di bawah Apache 2.0 (Smith, 2007) pada aplikasi *software*.

Tesseract (OCR) mengubah suatu proyek menjadi *software* yang menggunakan tambahan aplikasi lainnya yang memiliki fungsi sebagai pengenalan. Salah satu contoh adalah *software* Android Studio yang tidak memiliki OCR berkembang dalam pembuatan aplikasi tersebut. Jadi jika seseorang mengandalkan OCR untuk mengutamakan pengembangan aplikasinya, aplikasi tersebut perlu menambahkan *library* OCR baru, salah satunya adalah mesin OCR Tesseract. Tesseract OCR mendukung banyak bahasa untuk pengenalan atau pendeteksian teks.

2.4 Piksel

Elemen gambar dari representasi titik kecil per inci yang diambil melalui grafik disebut piksel. Piksel juga merupakan singkatan dari bahasa Inggris *Picture Element*. Piksel ditempatkan secara teratur berdekatan untuk membentuk gambar. Pada skala resolusi tertinggi, printer warna berhasil mengolah cetakan melebihi 2.500 titik per inci yang lebih dari 16 juta warna per inci, artinya satu inci persegi menampilkan tingkatan resolusi gambar tersebut.

Sesuai dengan 150 juta bit data. Seringkali layar datar yang dijumpai memiliki ribuan piksel yang dibagi menjadi beberapa kolom dan baris. Piksel yang terkandung pada layar dapat diketahui resolusinya. Resolusi maksimum yang ditawarkan adalah 1024 x 728, kemudian jumlah piksel pada layar adalah 786432 piksel. Ketajaman yang ditampilkan pada layar mempengaruhi tingkatan jumlah piksel yang dimiliki (Putra & Wirdiani 2014).

2.5 Format Citra Digital

Berbagai format gambar digital yang umum digunakan antara berikut:

1. BMP

Format gambar digital ini adalah sebuah metode untuk menyimpan data gambar digital tanpa memodifikasi gambar itu sendiri. Data citra yang terdapat dalam format ini memiliki

representasi yang serupa dengan data citra yang digunakan untuk menggambarkan citra dalam memori komputer.

2. GIF

Format gambar digital ini memiliki kemampuan untuk melakukan kompresi *lossless*, yang berarti kompresi tersebut tidak menghilangkan data dalam gambar digital. Dalam format ini, kualitas gambar digital tetap sama dengan gambar aslinya, tanpa mengalami penurunan. Namun, kelemahan utama dari format ini adalah keterbatasan dalam mendukung kedalaman warna, yaitu hanya hingga 256 warna.

3. JPEG

Ini adalah format gambar digital yang menggunakan kompresi *lossy*. Kualitas gambar digital yang disimpan dalam format ini berkurang dibandingkan dengan gambar aslinya. Format JPEG sejauh ini merupakan format yang paling populer. Hal ini dikarenakan ukurannya yang kecil dan portabel. Format ini sangat cocok untuk menyimpan foto karena ukuran data yang dihasilkan kecil dan kualitasnya bagus untuk mata manusia. Karena ukurannya yang kecil, *file* ini sering digunakan jaringan (internet).

4. PNG

Format PNG merupakan format yang memiliki kesamaan dengan format GIF, bahkan dirancang sebagai pengganti format GIF dengan beberapa perbaikan. Format PNG menggunakan metode kompresi *lossless* untuk mengompresi gambar digital, seperti yang disebutkan sendiri oleh Hendri (2014).

2.6 Cropping

Cropping adalah teknik proses seleksi sub piksel dari citra menjadi citra baru.

2.7 Resize

Resize adalah teknik pengolahan citra yang mengubah citra secara proporsional tanpa memotong bagian citra dengan mengubah jumlah piksel atau resolusi citra. Dalam

pengolahan citra, ukuran citra diperbesar atau diperkecil sesuai dengan faktor skala yang diinginkan.

2.8 Thresholding

Thresholding adalah teknik proses pengolahan citra yang dilakukan dalam rangka melakukan pemisahan objek dari *background* di dalam suatu citra. Caranya adalah dengan mengubah nilai piksel menjadi dua nilai. Nilai tersebut adalah biner 0 (nol) atau 1 (satu). Teknik ini memutuskan suatu piksel pada suatu citra merupakan bagian dari suatu objek atau latar belakang berdasarkan *threshold* yang ditentukan.

Nilai piksel yang lebih besar dari ambang diganti menjadi 1, sedangkan nilai yang terlihat sedikit atau kurang dari atau sama dengan ambang diubah menjadi 0. Kondisi ini dapat dilihat pada persamaan 2.1.

$$T(x,y) = \begin{cases} 1, & \text{jika } f(x,y) > T \\ 0, & \text{jika } f(x,y) \leq T \end{cases} \quad (2.1)$$

Keterangan:

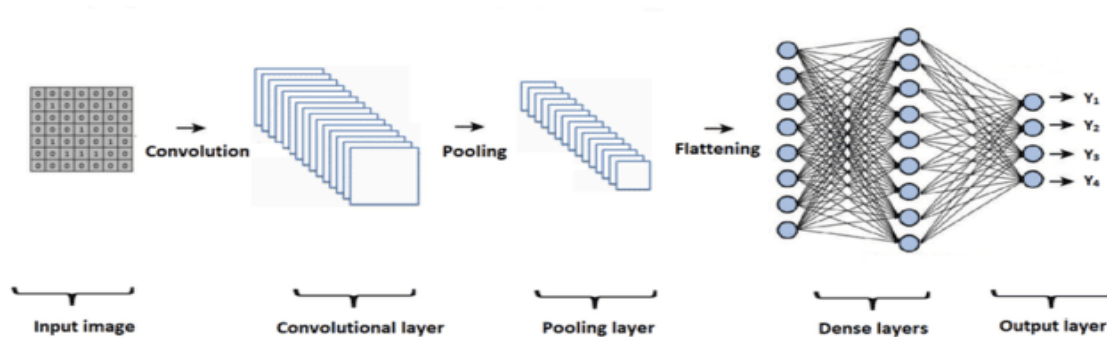
$T(x,y)$ adalah hasil *thresholding* di piksel (x,y)

$f(x,y)$ adalah nilai piksel di koordinat (x,y)

T adalah batas ambang (*threshold*)

2.9 Convolutional Neural Network (CNN)

Adapun implementasi *Deep Learning* dapat dikategorikan melalui jaringannya yang dikenal dengan *Convolutional Neural Network (CNN)*. Sementara itu, *Machine Learning* adalah *part of deep learning* yang dapat difungsikan sebagaimana *human function*, misalkan manusia melakukan pengajaran kepada komputer untuk mengerjakan pekerjaan dengan benar merupakan bagian dari proses *training* (Deng & Yu 2013) seperti pada Gambar 2.4.



Gambar 2.4 Arsitektur Convolutional Neural Network

(<https://www.researchgate.net/>)

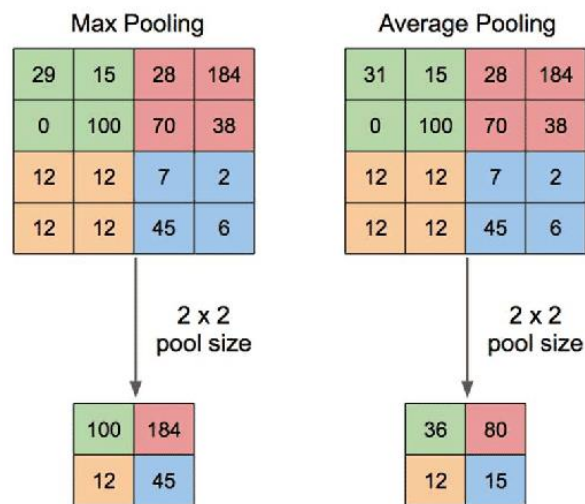
2.9.1 Convolutional Layer

Konvolusi (*convolution*) menghasilkan citra baru dengan cara memanipulasi citra melalui sebuah proses dimana menggunakan eksternal *mask* atau *subwindows*. Setiap elemen merupakan hasil perkalian dari hasil jumlah total dalam matriks atau dua vektor (Madenda, 2015). Konvolusi merupakan salah satu operasi dasar pada pengolahan citra dan pengenalan pola, khususnya pada *Convolutional Neural Network (CNN)*. Konvolusi dilakukan dengan menggeser sebuah filter konvolusi atau kernel pada sebuah matriks. Filter konvolusi terdiri dari sejumlah bobot yang dipelajari oleh jaringan secara otomatis selama proses *training*. Proses konvolusi mengidentifikasi jaringan untuk mengekstraksi fitur-fitur penting dari gambar, seperti tepian, garis, tekstur dan pola-pola kompleks.

2.9.2 Pooling Layer

Pooling adalah salah satu langkah dari *Convolutional Neural Network (CNN)* yang dilakukan setelah langkah konvolusi. *Pooling* bertujuan untuk memperkecil dimensi citra (spasial) dan mengurangi *overfitting* pada model CNN. Pengelompokan (*subregion*) pada citra dilakukan dengan pemilihan nilai terbesar atau terkecil dari beberapa nilai. Dalam pemrosesan *pooling*, ada dua bentuk yang sering digunakan, yaitu *max pooling* dan *average*

pooling. Angka maksimum dari *sub region* yang telah dikelompokkan diambil sebagai angka yang berfungsi sebagai *max pooling*. Di sisi lain, angka *average pooling* (nilai rata-rata) dianggap sebagai nilai representatif. *Pooling* dilakukan untuk mempercepat pembelajaran pada model CNN dengan cara memperkecil dimensi citra. Pengurangan ukuran gambar dapat mengurangi jumlah parameter pada model CNN juga menghilangkan *noise* atau fitur yang tidak relevan pada citra dapat dilihat pada Gambar 2.5.



Gambar 2.5 Pooling

(<https://www.researchgate.net/>)

2.9.3 Relu Layer

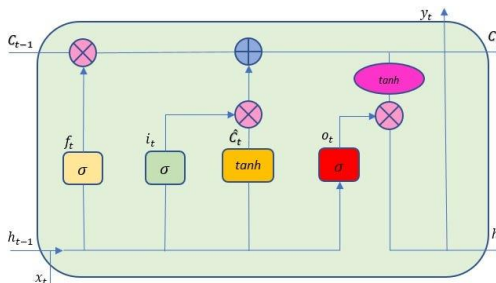
Selain itu, terdapat indikator ReLU (*Rectification Linear Unit*) yang diartikan sebagai suatu operasi yang digunakan untuk mengenalkan nonlinieritas dan meningkatkan visualisasi model. Fungsi aktivasi ReLU didefinisikan sebagai $f(x) = \max(0, x)$ (Heaton, 2015). Ketika input bernilai negatif, nilai *output* neuron akan menjadi 0. Sedangkan jika input neuron positif, *output*-nya akan sesuai dengan input aktivitas tersebut (Kim et al., 2016).

2.9.4 Fully Connected Layer

Hijazi et al. (2015) menyatakan bahwa lapisan *fully connected layer* terdiri dari serangkaian operasi konvolusi. Tujuan utama dari lapisan ini adalah menerima masukan input dari tahap sebelumnya yang memiliki korelasi yang signifikan dengan kelas yang spesifik. Lapisan ini bertujuan dengan menghubungkan *node* secara linear ke dalam satu dimensi seperti yang dijelaskan (Albelwi & Mahmood 2017).

2.10 Long Short-Term Memory (LSTM)

Algoritma *LSTM* memiliki kemampuan untuk menyimpan informasi dalam waktu yang lama. Terdiri dari 4 komponen sel utama yaitu *input gate*, *hidden state* dan *forget gate*, serta *output gate*. Dengan 4 komponen sel utama yang ada pada LSTM sebelumnya terdapat sebuah *cell state* yang berfungsi mengalirkan informasi dari satu *hidden state* ke *hidden state* selanjutnya tanpa melakukan perubahan signifikan. Adapun fungsi *gate input* adalah untuk mengizinkan atau menghentikan aliran sinyal masuk yang dapat mengganti kondisi sel. Oleh karena itu, *output gate* mengizinkan atau mengontrol dampak dari kondisi sel dengan unit lain. Dengan bantuan *forget gate*, sel dapat mengontrol pengingatan atau pengabaian terhadap kondisi koneksi berulang-ulang sel yang diatur seperti pada Gambar 2.6.



Gambar 2.6 Struktur Jaringan LSTM

(<https://www.medium.com/>)

Dalam arsitektur LSTM, fungsi aktivasi f dapat menggunakan fungsi tanh atau ReLU. Hipotesis h dalam LSTM dapat dijelaskan pada persamaan 2.2. h_t menggambarkan informasi yang diarsipkan oleh model pada titik waktu t baik dalam urutan ataupun tersembunyi sedangkan softmax aktivasi untuk menerapkan output. A_t melakukan prediksi urutan keluaran dan V merupakan matiks parameter penghubung model A_t dan h_t .

$$h_t = \text{softmax}(A_t, V) \quad (2.2)$$

Langkah awal dalam operasi LSTM adalah menghapus informasi dari *cell state* menggunakan *forget gate* f . *Forget gate* menerima vektor h_{t-1} sebagai input pertama dan kemudian memprosesnya melalui *forget gate* dengan menambahkan nilai bias b dapat dilihat pada persamaan 2.3. W_f menghubungkan f_t dengan h_{t-1} dan x_t input saat ini sesuai urutan data.

$$f_t = \sigma(W_f [h_{t-1}, x_t] + b_f) \quad (2.3)$$

Langkah berikutnya, jaringan LSTM melakukan pembaruan informasi ke dalam *cell state* menggunakan *input gate* i yang dapat dilihat pada persamaan 2.4 untuk menentukan informasi mana yang akan diperbarui dalam *cell state*. Sigmoid σ yang diterapkan untuk mengontrol seberapa banyak informasi yang akan dimasukkan. W_i adalah parameter yang menghubungkan i_t dengan h_{t-1} dan x_t input saat ini sesuai urutan data. Terakhir b_i bias digunakan untuk menambah pergeseran.

$$i_t = \sigma(W_i [h_{t-1}, x_t] + b_i) \quad (2.4)$$

Pada waktu yang sama, vektor yang sama juga diteruskan ke lapis tanh untuk mengubah rentang nilai vektor menjadi nilai nyata antara -1 hingga 1. Ini menghasilkan kandidat *cell state* yang dapat dilihat pada persamaan 2.5. \hat{C}_t untuk memuat informasi x_t input saat ini

dengan h_{t-1} . W_c bentuk representasi input saat ini x_t dan sebelumnya h_{t-1} untuk vektor baru \hat{C}_t dengan b_c bias

$$\hat{C}_t = \tanh (W_c \cdot [h_{t-1}, x_t] + b_c) \quad (2.5)$$

Vektor yang telah melewati *input gate* i_t dan lapisan tanh menghasilkan hanya output yang diinginkan yang akan dimasukkan ke dalam *cell state* C_t . Dengan demikian menghasilkan nilai *cell state* yang dapat dilihat pada persamaan 2.6.

$$C_t = (f_t \cdot C_{t-1} + i_t \cdot \hat{C}_t) \quad (2.6)$$

Langkah terakhir adalah menentukan *output gate* yang nantinya vektor h_{t-1} diteruskan ke *output gate* yang dapat dilihat pada persamaan 2.7.

$$o_t = \sigma (W_o \cdot [h_{t-1}, x_t] + b_o) \quad (2.7)$$

Pada saat yang sama, C_t (*cell state* sebelumnya) diteruskan ke lapisan tanh dan bertemu dengan o_t (*output gate*) menghasilkan nilai output dari *cell* yang dapat dilihat pada persamaan 2.8.

$$h_t = o_t * \tanh (C_t) \quad (2.8)$$

LSTM menggunakan variasi yang berbeda dimana *input gate* bergantung pada *forget gate* yang dapat dilihat pada persamaan 2.9.

$$C_t = f_t * C_{t-1} + (1 - f_t) * \hat{C}_t \quad (2.9)$$

2.11 Penelitian Terdahulu

Berdasarkan beberapa penelitian terdahulu yaitu *Penerapan Metode CNN-LSTM Dalam Memprediksi Hujan Pada Wilayah Medan*. Penelitian ini membahas *CNN-LSTM* yang memakai arsitektur 2 *hidden layer* yang terdiri dari 400 epoch dan 6 batch size yang mampu memprediksi hujan. Input data yang berjumlah banyak menghasilkan nilai Root Mean Square Error (RMSE).

Penelitian selanjutnya yaitu *Optical Character Recognition Menggunakan Algoritma Template Matching Correlation*. (*Optical Character Recognition*) OCR digunakan untuk mengidentifikasi angka maupun *image* alfabet/huruf untuk mengkonversi menjadi *file* tulisan. Dilakukan (*thresholding*) untuk mengubah citra *black and white* ke dalam *binary image*. Dengan algoritma *Template Matching Correlation* menghasilkan akurasi pengenalan karakter yang cukup efektif sehingga mendapatkan sebesar 92.90%.

Penelitian selanjutnya yaitu *Implementasi Metode Template Matching Untuk Media Pembelajaran Pengenalan Karakter Huruf Alphabet*. Hasil implementasi program memiliki dampak yang relatif signifikan pada *size*, *thickness*, dan mode mempengaruhi terhadap hasil akurasi. Dengan data uji (testing) sebanyak 312 buah dengan data training sebanyak 104 buah. Nilai akurasi didapatkan dari jumlah nilai benar pembagian dengan testing dikali 100% yang menghasilkan tingkat akurasi yang rendah berada di bawah 80% setelah 3 kali uji coba.

Penelitian selanjutnya yaitu *Convolutional Neural Network (CNN) Untuk Pengklasifikasian Citra Makanan Tradisional*. Pengujian CNN dilakukan dengan konvolusi, 4 layer max pooling, ReLu aktivasi dan 2 layer fully connected. Data citra menggunakan dua bagian yaitu data latih 280 citra dan data uji sebanyak 100 citra. Akurasi menggunakan Confusion Matrix yang menghasilkan presisi 73%, recall 69% dan Fscore 69%.

Penelitian selanjutnya yaitu *Perancangan Aplikasi Konversi File Image Hasil Scan Menjadi File Text Dengan Metode Feature Extraction*. Mengkonversi sebuah file dokumen atau gambar yang akan di *scan* menjadi sebuah *file softcopy*.

Penelitian selanjutnya *Klasifikasi Ekspresi Teks Berbahasa Jawa Menggunakan Algoritma Long Short Term Memory*. Klasifikasi pada penelitian ini memiliki 3 proses yaitu, proses LSTM, proses SVM dan proses RF. Untuk uji coba digunakan confusion matrix terhadap hasil klasifikasi yang sebenarnya. Dari training confusion matrix didapatkan algoritma yang paling tinggi yaitu LSTM sebesar 92% dan yang rendah yaitu RF sebesar 27%. SVM mendapat nilai akurasi sebesar 50% dengan presisi, recall dan Fscore yang sempurna yaitu 100%.

Tabel 2.1. Penelitian Terdahulu

No.	Peneliti	Metode	Keterangan
1.	Mhd. Alfandi et al. (2022)	<i>CNN-LSTM</i>	Pengujian pada penelitian ini memiliki tingkat akurasi mencapai sebesar 0.08.
2.	Suryo Hartanto et al. (2015)	<i>Template Matching</i> <i>Correlation</i>	Penelitian ini menghitung nilai korelasi (<i>correlation</i>) pada dua buah matriks.
3.	M Maskuri (2017)	<i>Template Matching</i>	Pengenalan alphabet sebanyak 312 buah (data testing) dan 104 buah (data training).
4.	Akhmad Rohim et al. (2019)	<i>CNN</i>	Data citra dibagi menjadi dua, 280 citra untuk data latih dan 100 citra untuk data uji didapatkan 73% presisi, 69% recall dan 69% Fscore.

Tabel 2.2. Penelitian Terdahulu

5.	Kadri Yusuf (2018)	<i>Feature Extraction</i>	Angka 1 akan mewakili pixel berwarna hitam, sementara angka 0 akan mewakili pixel berwarna putih.
6.	Oddy Virgantara Putra (2021)	<i>Long Short Term Memory</i>	Dari ketiga hasil LSTM, SVM dan RF didapatkan dengan LSTM menghasilkan akurasi paling baik yaitu 92%.

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi rancangan sistem/model untuk metode CNN dan LSTM serta prosedur-prosedur yang dilakukan dalam tahap *preprocessing*, analisis dan perancangan sistem. Tahap analisis berupa langkah-langkah yang dilakukan dalam menganalisis data yang digunakan untuk proses pengolahan data. Tahap perancangan membahas rencana untuk menciptakan sebuah sistem.

3.1 *Dataset* yang digunakan

Dataset diambil dari *kaggle* dan MNIST (*Modified National Institute of Standards and Technology database*) dengan menggunakan 90 lembar jawaban mahasiswa untuk diuji yang berisi nama, NIM, mata kuliah dan nilai mahasiswa. Data yang dipakai sudah dalam bentuk ekstensi .png. Contoh data yang digunakan dapat dilihat pada Gambar 3.1.

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI			
UNIVERSITAS SUMATERA UTARA			
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI			
Jl. Universitas No. 9A Gedung A, Kampus USU Padang Bulan, Medan 20155			
Telp/Fax : 061-4210077 Email : fasit@unissu.ac.id Lembar : http://fasit.unissu.ac.id			
Nama :	Alfa Rizki	Mata Ujian :	Peterungan Bakuin
NIM :	17020121	Dosen :	Dr. Irw. Sudharto Marawan, M.IT
Program Studi :	Teknologi Informatika	Nilai :	
Hari / Tanggal :	Senin / 26 Maret 2018		75
LEMBAR JAWABAN			

Gambar 3.1. Contoh Lembar Jawaban

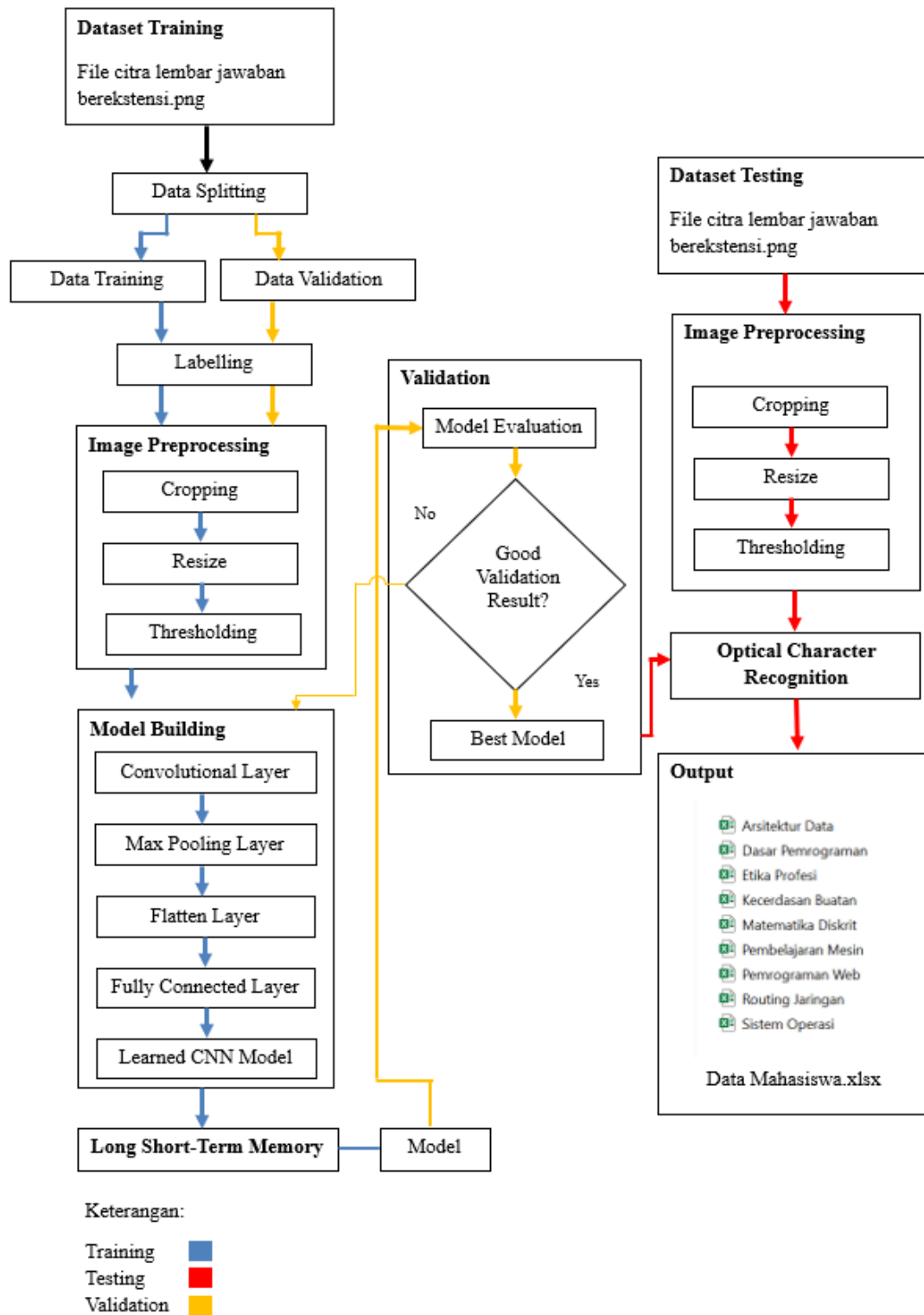
Dataset dibagi menjadi *training dataset* dengan 1600 data dan validasi 200. *Training dataset* akan digunakan untuk melatih model dalam mengenali pola pada data. Pada sisi lain *dataset* pengujian digunakan untuk menguji kemampuan model dalam rangka melakukan *value prediction* pada data yang *unobserved* sebelumnya dapat dilihat pada Tabel 3.1.

Tabel 3.1. Pembagian *Dataset*

<i>Dataset Training</i>		<i>Dataset Testing</i>
<i>Training</i>	<i>Validation</i>	90
1600	200	

3.2 Analisis Sistem

Pada skripsi ini menggunakan metode yang dapat menghasilkan suatu sistem berupa pengenalan tulisan tangan atau *Character Recognition* berdasarkan data mahasiswa baik nama, NIM, mata kuliah dan nilai. Sistem ini akan dikembangkan dengan metode yang digunakan *Optical Character Recognition*, *Convolutional Neural Network* dan *Long Short-Term Memory*. Sehingga penelitian ini akan meliputi beberapa tahapan. Tahap-tahap yang dilakukan adalah *data training* dan *data testing*. Pada tahap awal, lembar jawaban discan dan diubah menjadi gambar digital dalam ekstensi PNG. Selanjutnya citra diproses menggunakan OCR untuk mengenali dan memisahkan karakter yang ada pada citra. Proses OCR meliputi langkah-langkah *preprocessing* untuk meningkatkan kualitas gambar untuk membedakan setiap karakter pada gambar. Setelah karakter pada citra dipisahkan oleh OCR, sistem menggunakan CNN untuk mengenali nilai dari masing-masing karakter pada citra. Langkah-langkah dari CNN berupa konvolusi pada citra karakter, mengidentifikasi fitur-fitur penting dari gambar, seperti tepian, garis, tekstur dan pola-pola kompleks dari setiap karakter pada citra. Setelah itu *pooling* berfungsi untuk memperkecil ukuran citra dan mempermudah proses yang menghasilkan ringkasan lebih sederhana dan mudah dicerna. Selanjutnya LSTM proses mengenali karakter dalam gambar yang sudah dilakukan *training* pada model CNN. LSTM dapat memperhitungkan informasi jangka panjang dalam mengenal karakter dengan arsitektur umum sebagaimana pada Gambar 3.2.



Gambar 3.2. Arsitektur Umum

3.3 Preprocessing

a Cropping

Pada proses ini bertujuan untuk mengambil bagian penting yaitu nama, nim, mata kuliah dan nilai dan membuang bagian yang tidak diperlukan sehingga proses selanjutnya dapat dilakukan lebih optimal. Contoh data lembar jawaban setelah melalui proses *cropping* pada Gambar 3.3.

Nama	: Allia Rania	Mata Ujian	: Kecerdasan Buatan
NIM	: 171402151	Dosen	: Dr. Etna Budhiarti Nababan, M.IT
Program Studi	: Teknologi Informasi	Nilai	: 75
Hari / Tanggal	: Selasa / 26 Maret 2024		

Gambar 3.3. Bagian Cropping

b Resize

Langkah awal dalam tahap pra pemrosesan adalah mengubah ukuran (*resize*). Langkah ini bertujuan untuk menyesuaikan ukuran yang berbeda-beda menjadi ukuran yang sama. Penulis menggunakan *size* 300 x 300 piksel.

c Thresholding

Tahapan berikutnya yaitu *thresholding*. Tahapan ini dilakukan untuk menyelaraskan pewarnaan citra yang jika awalnya r, g, b menjadi warna keabuan terhadap citra asli. Dan jika warna citra sudah keabuan maka hasil akan lebih maksimal untuk lebih merata. Dalam penelitian ini citra data sedari awal sudah *grayscale* maka tidak diperlukan proses *grayscale* dan lanjut ke tahap *thresholding*. Telah dijelaskan sebelumnya bahwa jika citra masukan lebih besar atau sama maka nilai citra lebih rendah dari *threshold* (T) maka piksel akan diubah menjadi nilai 0 (hitam).

Salah satu lembar jawaban pada Gambar 3.3 diambil untuk di lakukan pengecekan angka grayscale dengan hasil selanjutnya yang akan digunakan untuk menjadi input dalam menghitung nilai threshold dapat dilihat pada Gambar 3.4.

```
[[ 3]
 [ 0]
 [ 0]
 [ 3]
 [ 7]
 [ 3]
 [ 0]
 [ 3]
 [ 0]
 [11]
 [ 0]
 [ 0]
 [ 3]
 [ 0]
 [ 0]
 [ 3]
 [ 8]
 [ 0]
 [ 0]
 [ 3]
 [ 0]
 [ 0]
 [ 0]
 [ 2]
 [ 0]
 [ 0]
 [ 0]
 [ 0]]
```

Gambar 3.4. Perhitungan Threshold

Citra data tersebut memiliki sebanyak 28 nilai, maka dalam penelitian ini yang diambil hanya 25 nilai sebagaimana mengikuti dari sumber penelitian terdahulu juga untuk memudahkan penulis dalam melakukan perhitungan berikutnya.

Selanjutnya pada tahapan ini, nilai citra data yang sudah *grayscale* akan dilakukan perhitungan pada setiap piksel dan dicari nilai yang maksimum dan minimum pada *threshold* (T) diketahui bahwa:

Nilai maksimum: 11

Nilai minimum : 0

Maka , $T = 11 + 0/2 = 5.5$

$3 < 5.5 = 0$

$$0 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$3 < 5.5 = 0$$

$$7 > 5.5 = 1$$

$$3 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$3 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$11 > 5.5 = 1$$

$$0 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$3 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$3 < 5.5 = 0$$

$$8 > 5.5 = 1$$

$$0 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$3 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$0 < 5.5 = 0$$

$$2 < 5.5 = 0$$

$$0 < 5.5 = 0$$

Telah ditentukan nilai 0 dan nilai 1 pada setiap piksel maka nilai *thresholding* yang berjumlah 25 nilai dibagi menjadi ukuran matriks 5 x 5 piksel seperti pada Gambar 3.5 dengan pembacaan dilakukan sesuai urutan dari kiri ke kanan.

0	0	0	0	1
0	0	0	0	1
0	0	0	0	0
0	1	0	0	0
0	0	0	0	0

Gambar 3.5. Nilai Threshold

3.4 CNN

Pada tahapan ini penulis menggunakan CNN untuk klasifikasi terhadap nilai *threshold* pada lembar jawaban yang telah dijelaskan sebelumnya pada bagian *threshold*. Piksel yang diambil berukuran 5 x 5 dikarenakan jumlah data yang awalnya 28 diambil 25 nilai sesuai dari citra data sebesar 300 x 300 piksel seperti pada Gambar 3.6.

3	0	0	3	7
3	0	3	0	11
0	0	3	0	0
3	8	0	0	3
0	0	0	2	0

Gambar 3.6. Nilai Citra Masukan

Dalam penelitian ini menggambarkan cara penggunaan padding=*'type'* terhadap konvolusi layer seperti pada Gambar 3.7.

0	0	0	0	0	0	0
0	3	0	0	3	7	0
0	3	0	3	0	11	0
0	0	0	3	0	0	0
0	3	8	0	0	3	0
0	0	0	0	2	0	0
0	0	0	0	0	0	0

1	1	1
1	1	1
1	1	1

Gambar 3.7. Penggunaan padding dan *kernel* 3x3

Proses layer konvolusi *input* dengan piksel 7 x 7 piksel yang telah ditambah *padding* sebelumnya menggunakan *kernel* yang berukuran 3 x 3 yang berperan sebagai proses konvolusi. Ukuran kernel 3 x 3 juga diambil nilai secara bebas untuk menentukan proses selanjutnya. Pergerakan *kernel* dimulai pada sudut kiri atas *kernel* 3 x 3, pergeseran akan memengaruhi baris pertama dan kolom ketiga dari *kernel* yang mengalami pergeseran ke arah kanan sejauh 1 piksel. Proses tersebut berulang terus untuk mencapai dan tersimpan citra baru seperti pada Gambar 3.8.

0	0	0	0	0	0	0
0	3	0	0	3	7	0
0	3	0	3	0	11	0
0	0	0	3	0	0	0
0	3	8	0	0	3	0
0	0	0	0	2	0	0
0	0	0	0	0	0	0

0	0	0	0	0	0	0
0	3	0	0	3	7	0
0	3	0	3	0	11	0
0	0	0	3	0	0	0
0	3	8	0	0	3	0
0	0	0	0	2	0	0
0	0	0	0	0	0	0

Gambar 3.8. Pergeseran *Kernel*

Masing-masing posisi menghitung jumlah elemen perkalian antara *kernel* pada citra masukan. *Feature map* akan digunakan dalam jumlah perhitungan pada pergerakan nilai *kernel* dapat dilihat sebagai berikut:

$$C1,1 = (0*1) + (0*1) + (0*1) + (0*1) + (3*1) + (0*1) + (0*1) + (3*1) + (0*1) \\ = 6$$

$$C1,2 = (0*1) + (0*1) + (0*1) + (3*1) + (0*1) + (0*1) + (3*1) + (0*1) + (3*1) \\ = 9$$

$$C1,3 = (0*1) + (0*1) + (0*1) + (0*1) + (0*1) + (3*1) + (0*1) + (3*1) + (0*1) \\ = 6$$

$$C1,4 = (0*1) + (0*1) + (0*1) + (0*1) + (3*1) + (7*1) + (3*1) + (0*1) + (11*1) \\ = 24$$

$$C1,5 = (0*1) + (0*1) + (0*1) + (3*1) + (7*1) + (0*1) + (0*1) + (11*1) + (0*1) \\ = 21$$

$$C2,1 = (0*1) + (3*1) + (0*1) + (0*1) + (3*1) + (0*1) + (0*1) + (0*1) + (0*1) \\ = 6$$

$$C2,2 = (3*1) + (0*1) + (0*1) + (3*1) + (0*1) + (3*1) + (0*1) + (0*1) + (3*1) \\ = 12$$

$$C2,3 = (0*1) + (0*1) + (3*1) + (0*1) + (3*1) + (0*1) + (0*1) + (3*1) + (0*1)$$

$$= 9$$

$$C2,4 = (0*1) + (3*1) + (7*1) + (3*1) + (0*1) + (11*1) + (3*1) + (0*1) + (0*1)$$

$$= 24$$

$$C2,5 = (3*1) + (7*1) + (0*1) + (0*1) + (11*1) + (0*1) + (0*1) + (0*1) + (0*1)$$

$$= 21$$

$$C3,1 = (0*1) + (3*1) + (0*1) + (0*1) + (0*1) + (0*1) + (0*1) + (3*1) + (8*1)$$

$$= 14$$

$$C3,2 = (3*1) + (0*1) + (3*1) + (0*1) + (0*1) + (3*1) + (3*1) + (8*1) + (0*1)$$

$$= 20$$

$$C3,3 = (0*1) + (3*1) + (0*1) + (0*1) + (3*1) + (0*1) + (8*1) + (0*1) + (0*1)$$

$$= 14$$

$$C3,4 = (3*1) + (0*1) + (11*1) + (3*1) + (0*1) + (0*1) + (0*1) + (0*1) + (3*1)$$

$$= 20$$

$$C3,5 = (0*1) + (11*1) + (0*1) + (0*1) + (0*1) + (0*1) + (0*1) + (3*1) + (0*1)$$

$$= 14$$

$$C4,1 = (0*1) + (0*1) + (0*1) + (0*1) + (3*1) + (8*1) + (0*1) + (0*1) + (0*1)$$

$$= 11$$

$$C4,2 = (0*1) + (0*1) + (3*1) + (3*1) + (8*1) + (0*1) + (0*1) + (0*1) + (0*1)$$

$$= 14$$

$$C4,3 = (0*1) + (3*1) + (0*1) + (8*1) + (0*1) + (0*1) + (0*1) + (0*1) + (2*1)$$

$$= 13$$

$$C4,4 = (3*1) + (0*1) + (0*1) + (0*1) + (0*1) + (3*1) + (0*1) + (2*1) + (0*1)$$

$$= 8$$

$$C4,5 = (0*1) + (0*1) + (0*1) + (0*1) + (3*1) + (0*1) + (2*1) + (0*1) + (0*1)$$

$$= 5$$

$$C5,1 = (0*1) + (3*1) + (8*1) + (0*1) + (0*1) + (0*1) + (0*1) + (0*1) + (0*1)$$

$$= 11$$

$$C5,2 = (3*1) + (8*1) + (0*1) + (0*1) + (0*1) + (0*1) + (0*1) + (0*1) + (0*1)$$

$$= 11$$

$$C5,3 = (8*1) + (0*1) + (0*1) + (0*1) + (0*1) + (2*1) + (0*1) + (0*1) + (0*1)$$

$$= 10$$

$$C5,4 = (0*1) + (0*1) + (3*1) + (0*1) + (2*1) + (0*1) + (0*1) + (0*1) + (0*1) \\ = 5$$

$$C5,5 = (0*1) + (3*1) + (0*1) + (2*1) + (0*1) + (0*1) + (0*1) + (0*1) + (0*1) \\ = 5$$

Hasil dari perkalian dan penjumlahan setiap posisi *kernel* seperti pada Gambar 3.9.

6	9	6	24	21
6	12	9	24	21
14	20	14	20	14
11	14	13	8	5
11	11	10	5	5

Gambar 3.9. Hasil Perhitungan

Langkah berikutnya adalah *max pooling* yang melibatkan pembagian *grid* yang lebih kecil dari lapisan konvolusi menggunakan ukuran matriks 2 x 2 piksel mengambil nilai maksimum dari matriks tersebut. Hasil dari proses tersebut dikenal sebagai peta fitur yang telah di *pooled* seperti pada Gambar 3.10.

12	12	24	24
20	20	24	24
20	20	20	20
14	14	13	8

Gambar 3.10. Hasil Perhitungan *Max Pooling*

Sebelum *input* dapat digunakan pada layer *fully connected*, maka nilai pada *pooled feature map* akan digunakan pada proses *flattening* seperti pada Gambar 3.11.

12
12
24
24
20
20
24
24
20
20
20
20
14
14
13
8

Gambar 3.11. Hasil Proses *Flattening*

Terakhir dilakukan mekanisme terhubungnya setiap neuron pada *fully connected layer*. Pada layer ini setiap potongan informasi akan menjadi masukan terhubung satu sama lain.

3.5 LSTM

Tabel 3.2. Model LSTM

<i>Layer</i>	<i>Output Shape</i>	Param
Embedding (Embedding)	(None, 500, 32)	192000
lstm (LSTM)	(None, 100)	53200
dense (Dense)	(None, 1)	101

Model tersebut adalah *training* model dari LSTM yang memiliki tiga lapisan, yakni *Embedding Layer*, *LSTM Layer* dan *Dense Layer* dapat dilihat sebagai berikut:

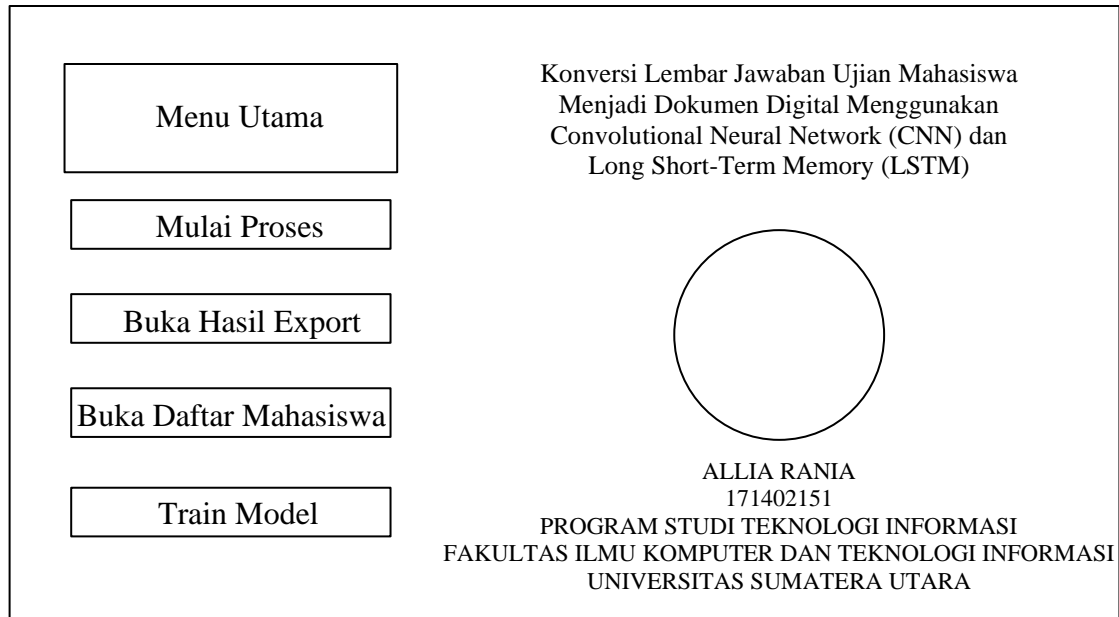
1. *Embedding Layer*: Layer ini memproses konversi kata-kata terhadap *dataset* menjadi vektor numerik. Terlihat pada tabel berikut Layer *Embedding* menerima masukan *input* sepanjang 500 kata dengan dimensi 32 kata masukan dengan total 192.000 parameter untuk dilatih.
2. *LSTM Layer*: Layer ini memiliki peran terpenting dari model LSTM, berguna untuk memeriksa informasi temporal pada *dataset*. Pada layer ini LSTM terdiri dari 100 neuron dan memiliki 53.200 parameter untuk dilatih.
3. *Dense Layer*: Layer ini menghasilkan hasil prediksi dari model. Pada model ini, *Dense Layer* terdiri dari 1 neuron dan memiliki 101 parameter untuk dilatih.

Model tersebut memiliki total 245.301 parameter yang dapat dilatih dan diatur terhadap kemampuan model.

3.6 Perancangan Antarmuka Sistem

3.7.1. Halaman *Home*

Halaman Beranda menampilkan tampilan awal ketika aplikasi mulai dijalankan pertama kali. Adapun berupa pada sisi kiri terdiri dari *train model*, buka hasil *export*, buka daftar mahasiswa dan mulai proses. Pada sisi kanan terdapat judul, nama, NIM serta institusi terlihat pada Gambar 3.12.



Gambar 3.12. Menu Utama

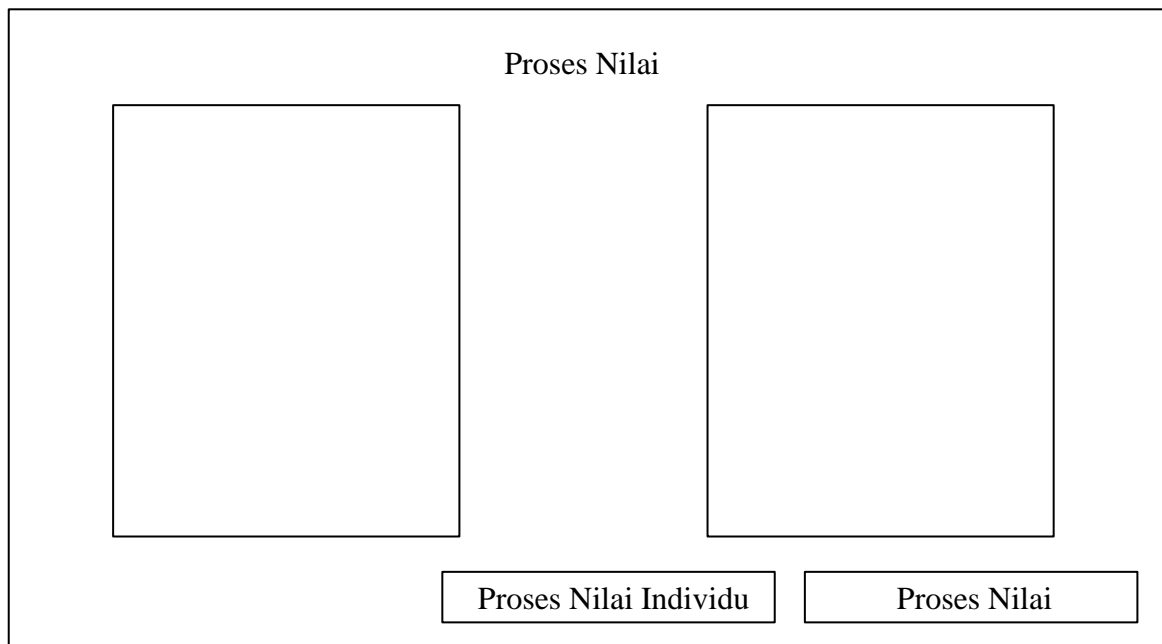
Keterangan:

1. *Train Model*, tombol untuk memulai *data training*
2. *Buka Hasil Export*, tombol untuk membuka hasil *export* dalam bentuk .csv yang sudah otomatis tersimpan
3. *Buka Daftar Mahasiswa*, tombol untuk membuka kumpulan daftar mahasiswa
4. *Mulai Proses*, tombol untuk memulai data testing
5. Judul
6. Logo Universitas
7. Nama, NIM, program studi, fakultas

3.7.1. Halaman Mulai Proses

Halaman Mulai Proses menampilkan proses menguji data dengan menggunakan metode *Convolutional Neural Network* serta *Long Short-Term Memory*. Pada halaman ini terdapat

dua bagian yang dapat dibedakan, antara lain proses nilai individu dan proses nilai (secara keseluruhan) yang bisa dilihat pada Gambar 3.13.



Gambar 3.13. Proses Nilai

Keterangan:

1. Hasil Proses Nilai Individu
2. Hasil Proses Nilai
3. Proses Nilai Individu, tombol untuk memulai proses terhadap lembar jawaban individu
4. Proses Nilai, tombol untuk memulai proses dalam jumlah banyak

3.7.1. Halaman Data Mahasiswa

Halaman Daftar Mahasiswa menampilkan daftar mahasiswa yang sudah tercatat untuk hasil yang lebih optimal saat *Optical Character Recognition* membaca lembar jawaban mahasiswa. Halaman ini juga terdiri dari 3 tombol, yaitu edit mahasiswa, tambah mahasiswa dan hapus mahasiswa yang terdapat pada Gambar 3.14.

Data Mahasiswa

Edit Mahasiswa

Tambah Mahasiswa

Hapus Mahasiswa

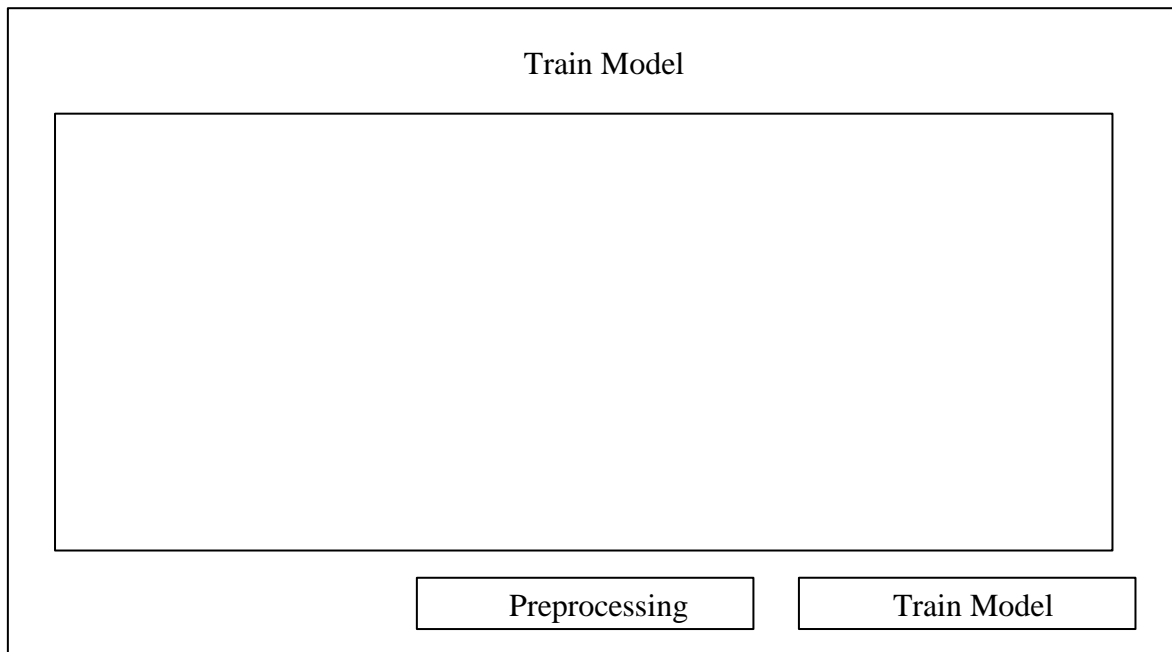
Gambar 3.14. Data Mahasiswa

Keterangan:

1. Hasil Daftar Mahasiswa
2. Edit Mahasiswa, tombol untuk mengedit nama atau NIM mahasiswa
3. Tambah Mahasiswa, tombol untuk menambah mahasiswa
4. Hapus Mahasiswa, tombol untuk menghapus salah satu mahasiswa dari daftar

3.7.1. Halaman *Train Model*

Halaman *Train Model* menampilkan dua tampilan berupa *preprocessing* dan *train model*. Tampilan *Threshold* pada *preprocessing* yang memakai dataset MNIST juga Confussion Matrix yang dievaluasi menggunakan *CNN* dapat dilihat pada Gambar 3.15.



Gambar 3.15. Train Model

Keterangan:

1. Hasil *Preprocessing* dan hasil *Train Model*
2. *Preprocessing*, tombol untuk memulai kerja *preprocessing* dari kumpulan beberapa angka
3. *Train Model*, tombol untuk memulai *train* dan menghasilkan hasil grafik

3.7 Confusion Matrix

Confusion Matrix adalah sebuah teknik evaluasi yang diimplementasikan dalam metode klasifikasi. Gambar 3.16. merupakan deskripsi sederhana untuk membantu memahami *Confusion Matrix* dalam hasil klasifikasi.

		Actual Values	
		1 (Positive)	0 (Negative)
Predicted Values	1 (Positive)	TP (True Positive)	FP (False Positive) <i>Type I Error</i>
	0 (Negative)	FN (False Negative) <i>Type II Error</i>	TN (True Negative)

Gambar 3.16. Confusion Matrix

(<https://medium.com/>)

Nilai *True Positive* (TP) merupakan data positif yang diklasifikasi benar. *False Positive* (FP) merupakan data negatif yang diklasifikasi benar. *False Negative* merupakan data negatif yang diklasifikasi salah. *True Negative* merupakan data negatif yang diklasifikasi benar. Pada penelitian ini penggunaan confusion matrix hanya untuk model training.

BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini memuat implementasi dari program yang telah dilakukan perancangan seperti yang dijelaskan pada Bab 3, serta mengulas *result* pengujian terhadap sistem yang telah dikembangkan.

4.1 Implementasi Sistem

4.1.1 Spesifikasi Perangkat Keras dan Lunak

Adapun spesifikasi perangkat keras (*hardware*) dan perangkat lunak (*software*) yang digunakan untuk menjalankan aplikasi.

Spesifikasi Perangkat Keras

1. Perangkat laptop yang digunakan ASUS VivoBook S14
2. Processor Intel(R) Core (TM) i5-8250U CPU @ 1.60GHz 1.80 GHz
3. Ram laptop 8GB
4. Hardisk 1TB

Spesifikasi Perangkat Lunak

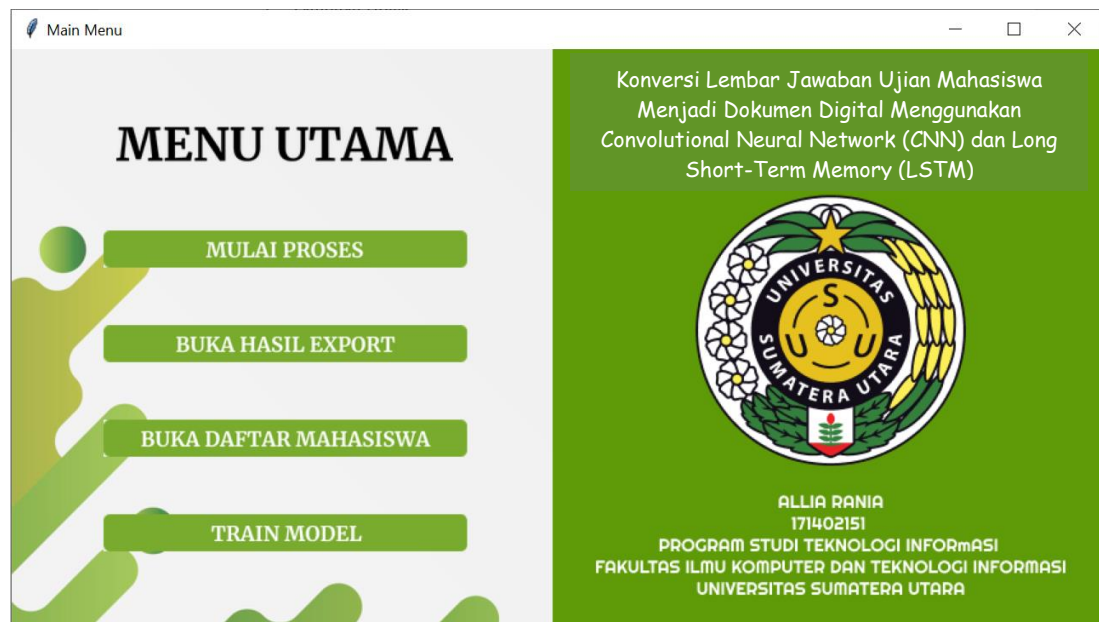
1. Sistem operasi Windows 10 Home 64 bit
2. *Python Version 3.10.6*
3. *Library* yang digunakan: *pytesseract, skimage, pandas, xlswriter, cv2.*
4. Untuk membangun *interface* menggunakan *tkinter python.*
5. *Visual Studio Code (VSC)*

4.1.2 Perancangan Antarmuka

Perancangan antarmuka pengguna sistem didasarkan pada hasil analisis yang dijelaskan dalam Bab 3, sebagai berikut:

1. Tampilan *Home*

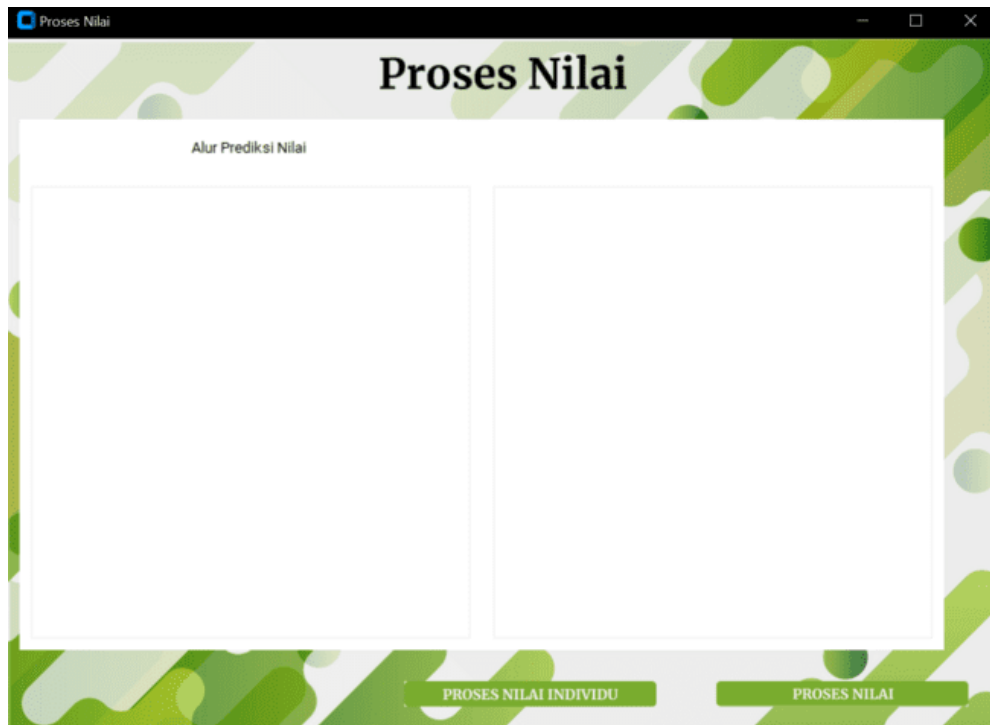
Rancangan tampilan yang sudah diperlihatkan pada Bab 3 sebelumnya, tampilan halaman *Home* menampilkan tampilan awal yang akan terhubung ke halaman lainnya. Fitur yang berisi terkait informasi penulis dan beberapa fitur lainnya seperti diberikan pada Gambar 4.1.



Gambar 4.1. Fitur Home

2. Tampilan Mulai Proses

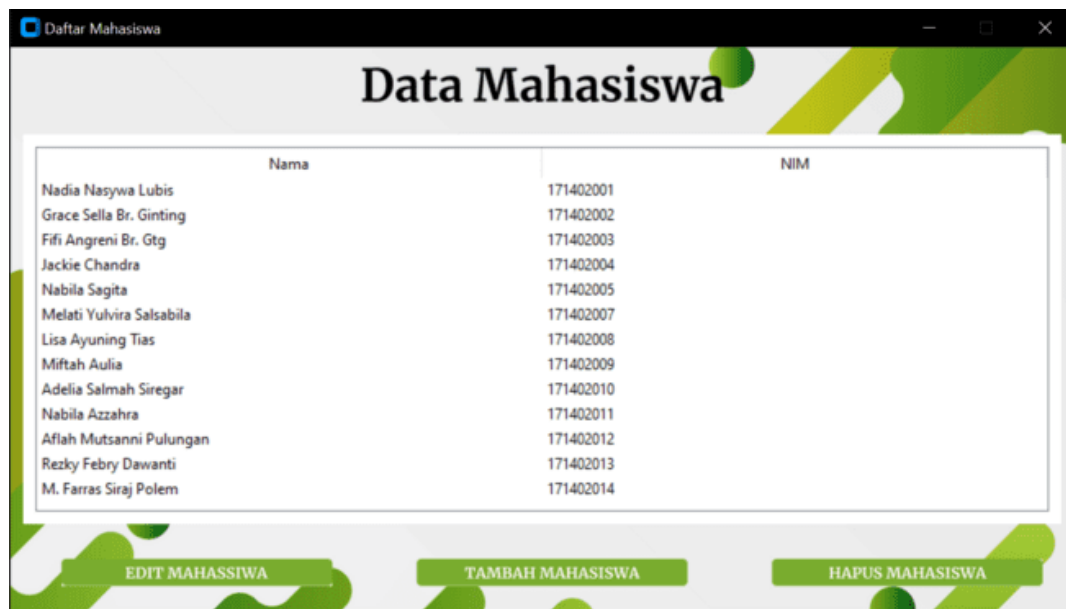
Pada halaman Mulai Proses dapat kita lihat proses menguji nilai data baik dengan cara individu ataupun dalam bentuk 9 folder (secara keseluruhan) yang berekstensi *.png* seperti pada Gambar 4.2.



Gambar 4.2. Mulai Proses

3. Tampilan Buka Daftar Mahasiswa

Halaman Daftar Mahasiswa menampilkan daftar mahasiswa yang sudah tercatat juga bisa ditambah untuk memaksimalkan hasil nilai lebih optimal dalam membaca lembar jawaban mahasiswa dapat dilihat Gambar 4.3.













Nama	NIM
Nadia Nasywa Lubis	171402001
Grace Sella Br. Ginting	171402002
Fifi Angreni Br. Gtg	171402003
Jackie Chandra	171402004
Nabila Sagita	171402005
Melati Yulvira Salsabila	171402007
Lisa Ayuning Tias	171402008
Miftah Aulia	171402009
Adelia Salmah Siregar	171402010
Nabila Azzahra	171402011
Aflah Mutsanni Pulungan	171402012
Rezky Febry Dawanti	171402013
M. Farras Siraj Polem	171402014

Gambar 4.3. Daftar Mahasiswa

4. Tampilan Buka Hasil *Export*

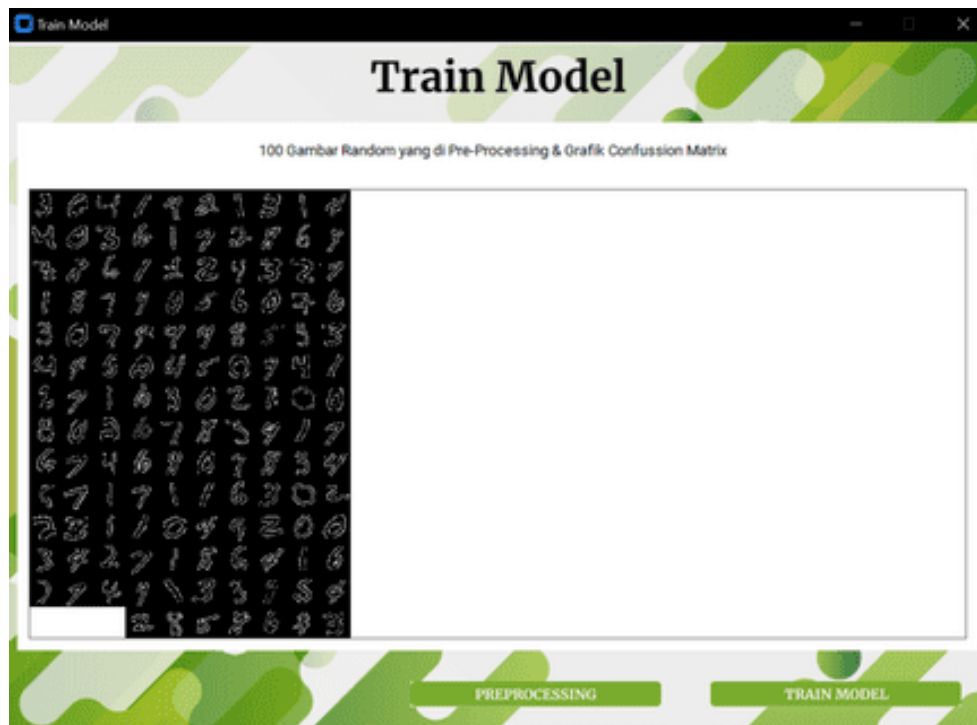
Halaman Hasil *Export* merupakan hasil otomatis setelah melakukan proses masukan input nilai yang telah tersedia pada Gambar 4.4.

Name	Date modified	Type	Size
 Arsitektur Data	6/12/2023 10:47 AM	Microsoft Excel Work...	6 KB
 Dasar Pemrograman	6/12/2023 10:47 AM	Microsoft Excel Work...	6 KB
 Etika Profesi	6/12/2023 10:47 AM	Microsoft Excel Work...	6 KB
 hasil	1/25/2023 6:15 PM	Text Document	0 KB
 Kecerdasan Buatan	6/12/2023 10:47 AM	Microsoft Excel Work...	6 KB
 Matematika Diskrit	6/12/2023 10:47 AM	Microsoft Excel Work...	6 KB
 Pembelajaran Mesin	6/12/2023 10:47 AM	Microsoft Excel Work...	6 KB
 Pemrograman Web	6/12/2023 10:47 AM	Microsoft Excel Work...	6 KB
 Routing Jaringan	6/12/2023 10:47 AM	Microsoft Excel Work...	6 KB
 Sistem Operasi	6/12/2023 10:47 AM	Microsoft Excel Work...	6 KB

Gambar 4.4. Hasil *Export*

5. Tampilan *Train Model*

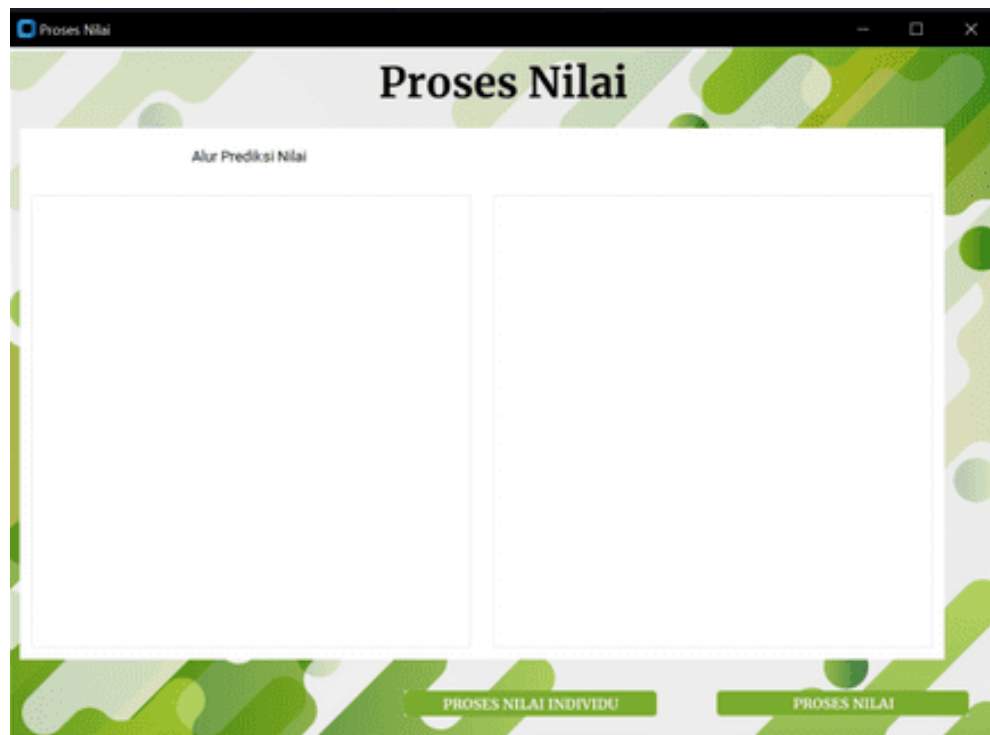
Halaman *Train Model* menampilkan dua tampilan berupa *preprocessing* dan *train model*. *Preprocessing* yang memakai dataset MNIST acak secara random dan *train model* dapat dilihat pada Gambar 4.5.



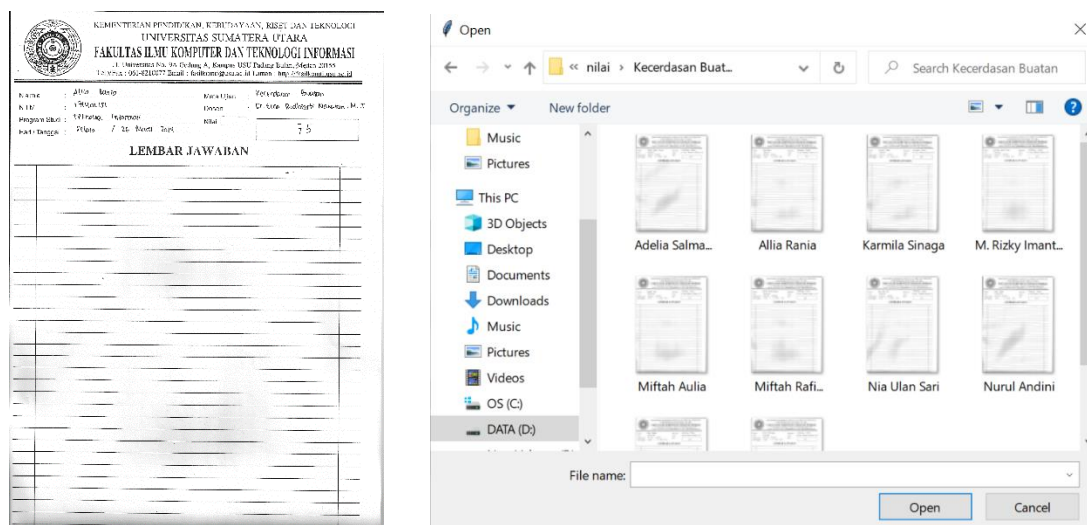
Gambar 4.5. *Train Model Preprocessing*

4.2 Prosedur Operasional

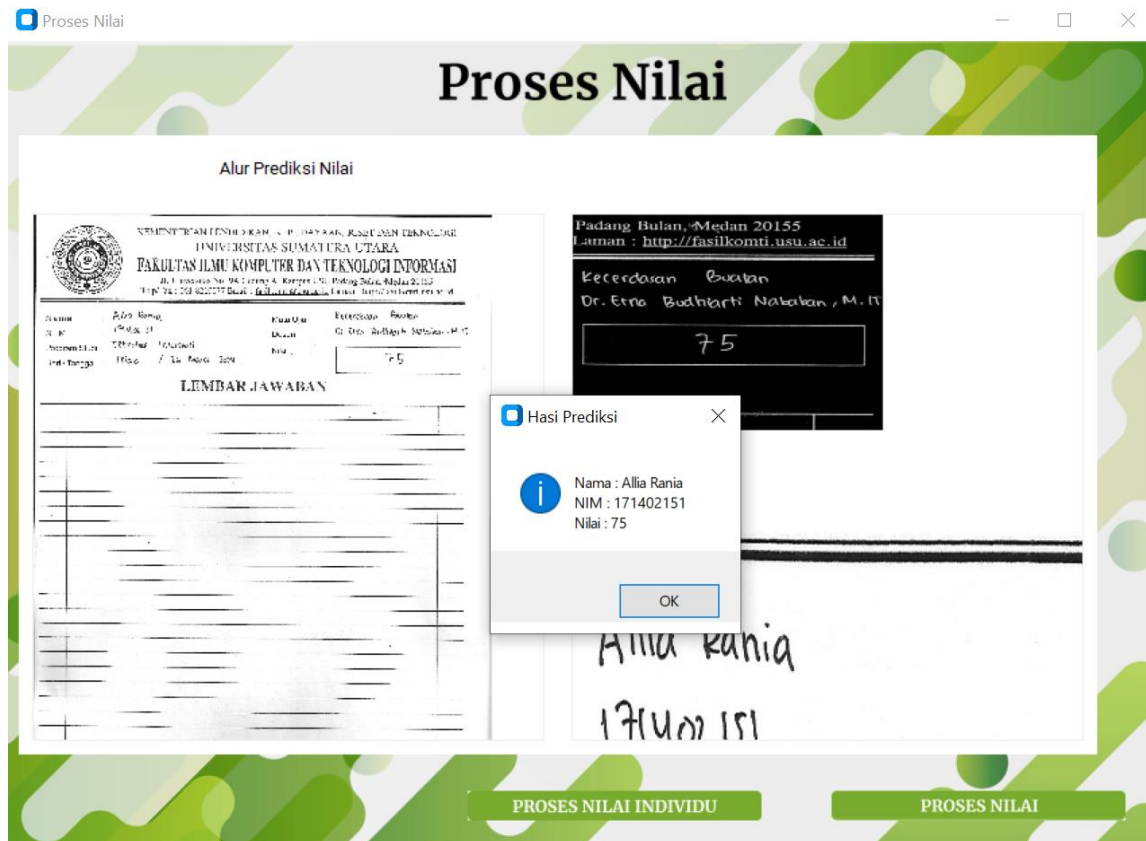
Pada bagian halaman berikut *user* memilih proses nilai individu yang dapat dilihat pada Gambar 4.6 untuk melakukan identifikasi terhadap satu lembar jawaban citra data dari folder (mata kuliah) yang diinginkan seperti pada Gambar 4.7 Setelah dilakukan identifikasi terhadap salah satu lembar jawaban selanjutnya proses nilai menghasilkan nama (Allia Rania), nim (171402151) dan nilai (71) untuk mengetahui akurat aplikasi dapat dilihat pada Gambar 4.8.



Gambar 4.6. Proses Nilai Individu



Gambar 4.7. Input Lembar Jawaban



Gambar 4.8. Proses Individu Lembar Jawaban

Kemudian pada bagian halaman yang masih sama, berikutnya *user* memilih proses nilai yang dapat dilihat pada Gambar 4.6 untuk melakukan identifikasi terhadap keseluruhan lembar jawaban citra data. Identifikasi dilakukan kembali untuk mengetahui akurasi aplikasi seperti pada Gambar 4.9.

Proses Nilai

Alur Prediksi Nilai

UNIVERSITAS SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

Padang Bulan, Medan 20155
Laman : <http://fasilkomti.usu.ac.id>

DASAR PEMROGRAMAN
Prof. Dr. Drs. Opim Salim Sitompal, M.S.

62

Mata Kuliah Proses Dasar Pemrograman 2/9

Karmila Sinaga
17/10/2026

PROSES NILAI INDIVIDU PROSES NILAI

Gambar 4.9. Proses Keseluruhan Lembar Jawaban

4.3 Implementasi Model

Dilakukan beberapa percobaan proses *training* dengan mengubah jumlah *batch size* dan *epoch* untuk mendapatkan hasil yang maksimal. Model dengan akurasi tertinggi adalah model dengan performa terbaik. Penulis mendapatkan kesimpulan bahwa dengan *batch size* sebesar 32 dan epoch sebanyak 10 sudah mencapai hasil akurasi terbaik dapat dilihat pada Tabel 4.1.

Tabel 4.1. Epoch Penelitian

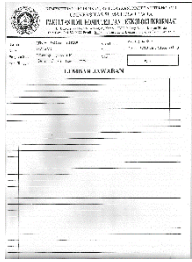
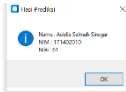
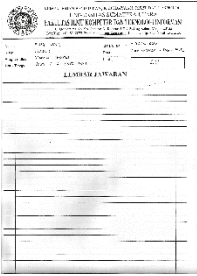
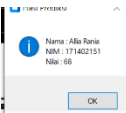
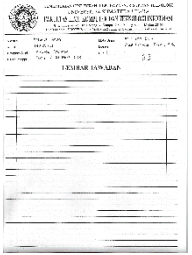
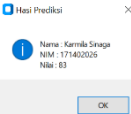
NO	epoch	batch	loss	accuracy
1.	1	32	4.818	98.42
2.	2	32	7.395	97.54
3.	3	32	7.021	97.78
4.	4	32	4.729	98.56
5.	5	32	5.262	98.35
6.	6	32	5.064	98.27
7.	7	32	8.572	97.03
8.	8	32	5.162	98.41
9.	9	32	4.407	98.59
10.	10	32	4.788	98.58

Dalam tabel 4.1. setiap baris mewakili satu *epoch* dan *batch* tertentu dalam proses pelatihan model. Pada baris pertama dilakukan penelitian pada *epoch* 1 dan *batch* 32 dengan nilai loss sebesar 4.818 dan akurasi sebesar 98.42%. *Epoch* dan *batch* merupakan parameter penting dalam pelatihan model. *Epoch* menentukan berapa kali model akan melihat seluruh dataset training sedangkan *batch* menentukan seberapa banyak sampel yang akan digunakan dalam satu kali perhitungan. Penelitian ini melakukan percobaan dengan menggunakan beberapa *epoch* dengan *batch size* 32 dan sudah mendapatkan rata-rata akurasi sebesar 90%.

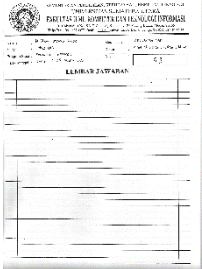
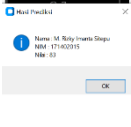
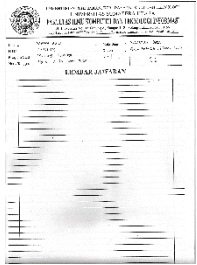
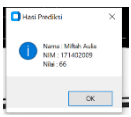
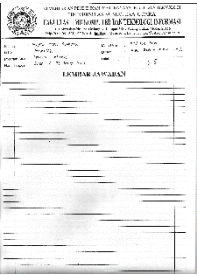
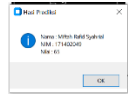
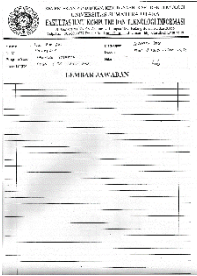
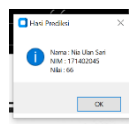
4.4 Pengujian Sistem

Tahap pengujian dilakukan setelah tahapan implementasi model dilewati sesuai dengan prosedur yang telah dijelaskan sebelumnya. Pengujian data dilakukan pada 90 lembar jawaban dalam bentuk kata dan kalimat yang diambil yaitu nama, nim dan nilai. Adapun hasil pengujian sistem identifikasi dapat dilihat pada Tabel 4.2.

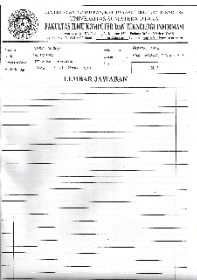
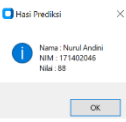
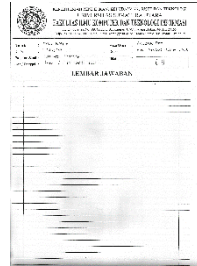
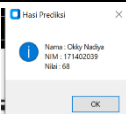

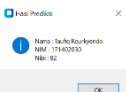
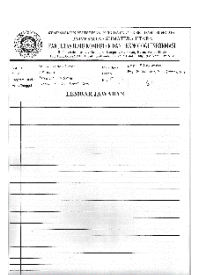
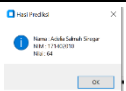
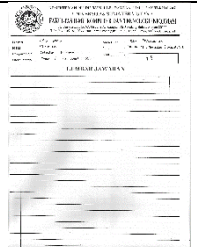
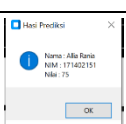
Tabel 4.2. Pengujian Sistem

NO	Lembar Jawaban	Teks	Citra Hasil	Nama Nim	Nilai	Kecocokan
1.		ADELIA SALMAH SIREGAR 171402010 61		ADELIA SALMAH SIREGAR 171402010	61	3/3
2.		ALLIA RANIA 171402151 68		ALLIA RANIA 171402151	68	3/3
3.		Karmila Sinaga 171402026 83		Karmila Sinaga 171402026	83	3/3

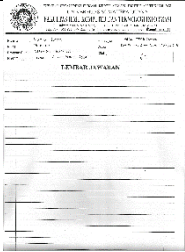
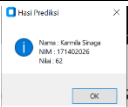
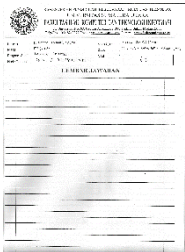
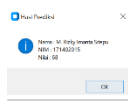
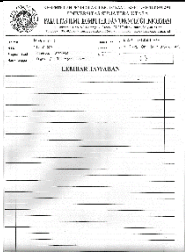
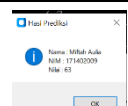
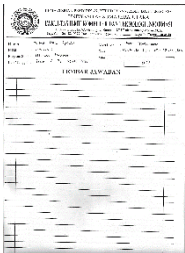
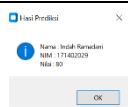
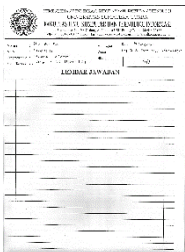
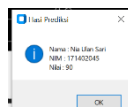
Tabel 4.3. Pengujian Sistem (Lanjutan)

4.		M. Rizky Imanta Sitepu 171402015 83		M. Rizky Imanta Sitepu 171402015	83	3/3
5.		MIFTAH AULIA 171402009 66		MIFTAH AULIA 171402009	66	3/3
6.		MIFTAH RAFID SYAHRIAL 171402049 65		MIFTAH RAFID SYAHRIAL 171402049	65	3/3
7.		Nia Ulan Sari 171402045 66		Nia Ulan Sari 171402045	66	3/3

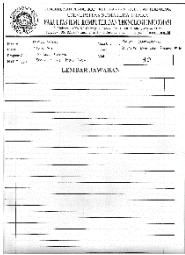
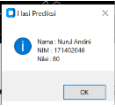
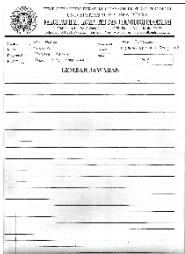
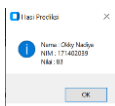
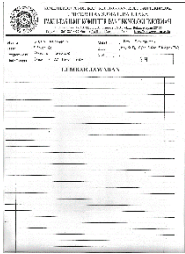
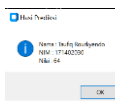
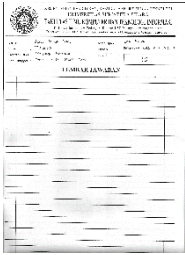
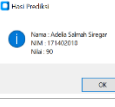
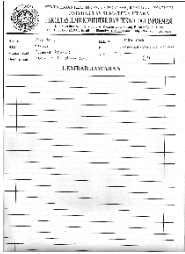
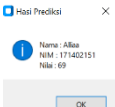
Tabel 4.4. Pengujian Sistem (Lanjutan)

8.		NURUL ANDINI 171402046 87		NURUL ANDINI 171402046	88	2/3
9.		OKKY NADIYA 171402049 68		OKKY NADIYA 171402039	68	3/3
10.		TAUFIQ ROURKYEN DO 171402030 82		TAUFIQ ROURKYEND O 171402030	82	3/3
11.		Adelia Salmah Siregar 171402010 64		Adelia Salmah Siregar 171402010	64	3/3
12.		Allia Rania 171402151 75		Allia Rania 171402151	75	3/3

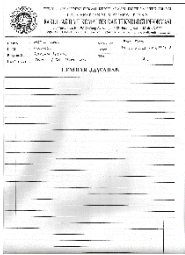
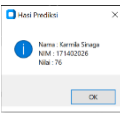
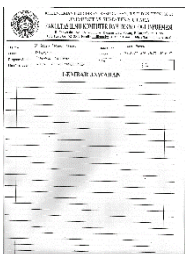
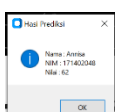
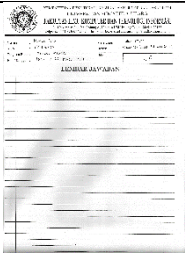

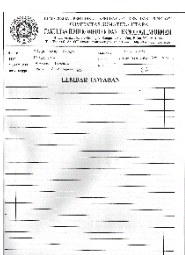
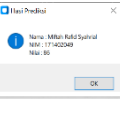
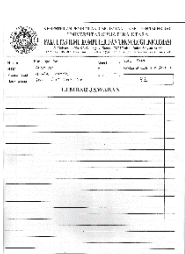
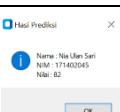
Tabel 4.5. Pengujian Sistem (Lanjutan)

13.		Karmila Sinaga 171402026 62		Karmila Sinaga 171402026	62	3/3
14.		M. RIZKY IMANTA SITEPU 171402015 68		M. RIZKY IMANTA SITEPU 171402015	68	3/3
15.		Miftah Aulia 171402009 63		Miftah Aulia 171402009	63	3/3
16.		Miftah Rafid Syahril 171402049 80		Indah Ramadhani 171402029	80	1/3
17.		Nia Ulan Sari 171402045 90		Nia Ulan Sari 171402045	90	3/3

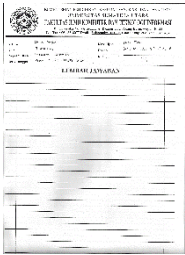
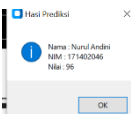
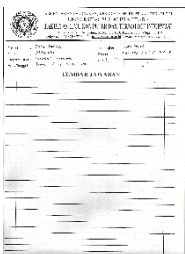
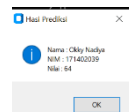
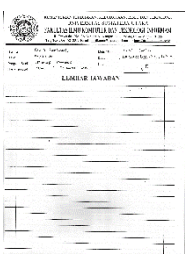
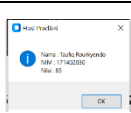
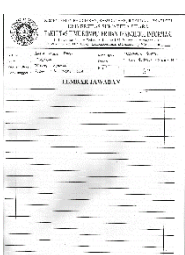
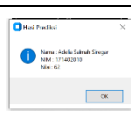
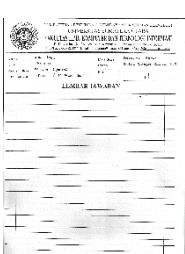
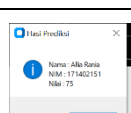
Tabel 4.6. Pengujian Sistem (Lanjutan)

18.		NURUL ANDINI 171402046 80		NURUL ANDINI 171402046	80	3/3
19.		Okky Nadiya 171402039 83		Okky Nadiya 171402039	83	3/3
20.		Taufiq Rourkyendo 171402030 64		Taufiq Rourkyendo 171402030	64	3/3
21.		Adelia Salmah Siregar 171402010 90		Adelia Salmah Siregar 171402010	90	3/3
22.		Allia Rania 171402151 69		Alliaa 171402151	69	2/3

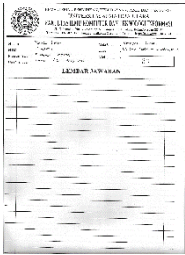
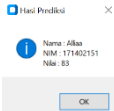
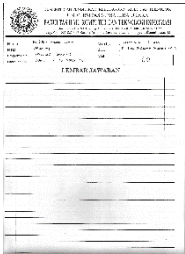
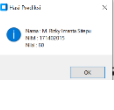
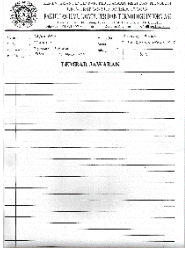
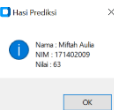
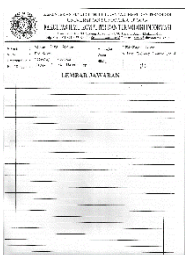
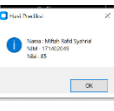
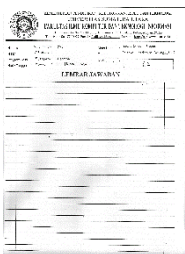
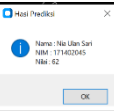
Tabel 4.7. Pengujian Sistem (Lanjutan)

23.		Karmila Sinaga 171402026 76		Karmila Sinaga 171402026	76	3/3
24.		M. Rizky Imanta Sitepu 171402015 62		Annisa 171402048	62	1/3
25.		Miftah Aulia 171402009 65		Miftah Aulia 171402009	65	3/3
26.		Miftah Rafid Syahril 171402049 86		Miftah Rafid Syahril 171402049	86	3/3
27.		Nia Ulan Sari 171402045 82		Nia Ulan Sari 171402045	82	3/3

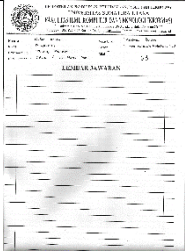
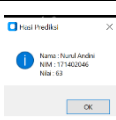
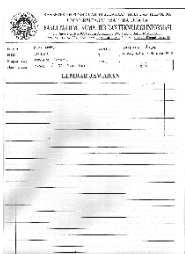
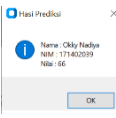
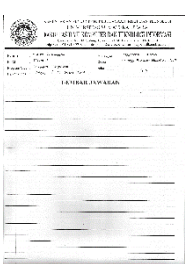
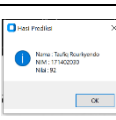
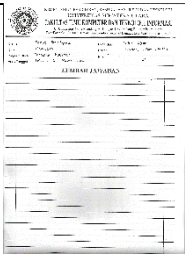
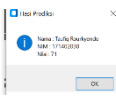
Tabel 4.8. Pengujian Sistem (Lanjutan)

28.		Nurul Andini 171402046 96		Okky Nadiya 171402039	96	1/3
29.		Okky Nadiya 171402039 64		Okky Nadiya 171402039	64	3/3
30.		Taufiq Rourkyendo 171402030 85		Taufiq Rourkyendo 171402030	85	3/3
31.		Adelia Salmah Siregar 171402010 62		Adelia Salmah Siregar 171402010	62	3/3
32.		Allia Rania 171402151 75		Allia Rania 171402151	75	3/3

Tabel 4.9. Pengujian Sistem (Lanjutan)

33.		Karmila Sinaga 171402026 83		Alliaa 171402151 83	1/3
34.		M. Rizky Imanta Sitepu 171402015 60		M. Rizky Imanta Sitepu 171402015 60	3/3
35.		Miftah Aulia 171402009 63		Miftah Aulia 171402009 63	3/3
36.		Miftah Rafid Syahril 171402049 85		Miftah Rafid Syahril 171402049 85	3/3
37.		NIA ULAN SARI 171402045 62		NIA ULAN SARI 171402045 62	3/3

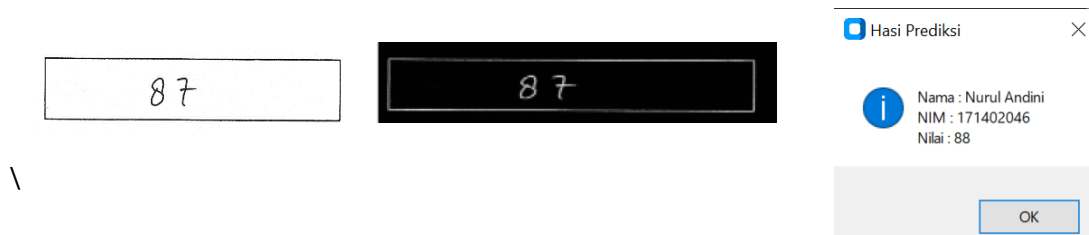
Tabel 4.10. Pengujian Sistem (Lanjutan)

38.		NURUL ANDINI 171402046 63		Allia Rania 171402151	63	1/3
39.		Okky Nadiya 171402039 66		Okky Nadiya 171402039	66	3/3
40.		Taufiq Rourkyendo 171402030 92		Taufiq Rourkyendo 171402030	92	3/3
.						
90.		Taufiq Rourkyendo 171402030 71		Taufiq Rourkyendo 171402030	71	3/3

4.5 Evaluasi

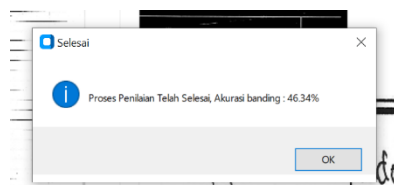
Pada bagian Evaluasi diketahui bahwa tingkat akurasi yang diperoleh mencapai 84.44% yang dimana merupakan tingkat akurasi yang sudah baik.

Adapun beberapa faktor yang menyebabkan tidak sempurnanya sistem ini, dikarenakan pada saat proses scanning tulisan angka 7 tidak jelas sehingga thresholding tidak dapat mendeteksi dan terbaca menjadi angka 8. Adapun dikarenakan tulisan yang mempunyai kemiripan angka lain sehingga Convolutional Neural Network menyebabkan adanya salah identifikasi. Contoh lembar jawaban yang gagal dapat dilihat pada Gambar 4.10.



Gambar 4.10. Contoh Lembar Jawaban Gagal

Adapun alasan mengapa saya memakai dua algoritma dikarenakan jika saya hanya memakai satu seperti Long Short-Term Memory saja maka hasil akan menjadi rendah seperti pada Gambar 4.11.



Gambar 4.11. Contoh Hasil Akurasi hanya menggunakan LSTM

BAB 5

KESIMPULAN DAN SARAN

Bab ini berisikan mengenai kesimpulan atas seluruh bagian skripsi serta saran pengembangan ke depan, khususnya sebagai referensi untuk penelitian selanjutnya.

5.1 Kesimpulan

Kesimpulan yang dapat penulis tarik dari seluruh hasil identifikasi, pengembangan, dan penelitian, termasuk pengujian sistem, adalah sebagaimana berikut:

1. Kualitas tulisan data citra uji sangat berpengaruh terhadap hasil validasi akurasi Nilai.
2. Algoritma *Convolutional Neural Network* dan *Long Short-Term Memory (LSTM)* memiliki akurasi yang cukup baik dalam mengklasifikasikan Nama, NIM dan nilai sebesar 80% dengan memakai 90 lembar jawaban.
3. OCR Tesseract Engine dan CNN sudah cukup baik untuk mengenali Nama, NIM dan Nilai dari lembar jawaban mahasiswa.

5.2 Saran

Penulis juga menyajikan beberapa rekomendasi yang dapat dipertimbangkan oleh penulis dalam rangka pengembangan dalam penelitian mendatang:

1. Perbanyak jumlah *dataset* agar dapat meningkatkan akurasi pengenalan Nama, NIM dan Nilai mahasiswa.
2. Dilakukan pengembangan lebih lanjut diharapkan sistem ini dapat berkembang menjadi platform *mobile* untuk kemudahan penggunaan.

DAFTAR PUSTAKA

- Alim, M. N. (2023). Pemodelan Time Series Data Saham LQ45 dengan Algoritma LSTM, RNN, dan Arima. *PRISMA, Prosiding Seminar Nasional Matematika*.
- Carbune, V., & dkk. (2019). Fast multi-language LSTM-based online handwriting recognition. *International Journal on Document Analysis and Recognition*.
- Kurniawan, A. A., & Mustikasari, M. (2021). Implementasi Deep Learning Menggunakan Metode CNN dan LSTM untuk Menentukan Berita Palsu dalam Bahasa Indonesia. *Jurnal Informatika Universitas Pamulang*.
- Lestari, S., & Pratama, M. F. (2022). Penerapan Metode Long Short-Term Memory Pada Pendataan Warga Berbasis Android. *Journal of Computer System and Informatics (JoSYC)*.
- Naseer, A., & Zafar, K. (2018). Comparative Analysis of Raw Images and Meta Feature based Urdu OCR using CNN and LSTM. *International Journal of Advanced Computer Science and Applications*.
- Putra, O. V., Musthafa, A., & Wibowo, K. P. (2019). Klasifikasi Ekspresi Teks Berbahasa Jawa Menggunakan Algoritma Long Short Term Memory. *Komputika: Jurnal Sistem Komputer*.
- Rachman, F. P., & Santoso, H. (2021). Perbandingan Model Deep Learning untuk Klasifikasi Sentiment Analysis dengan Teknik Natural Language Processing. *Jurnal Teknologi dan Manajemen Informatika*.
- Rohim, A. A., & dkk. (2020). Implementasi Algorithm CNN dan SLTM untuk Translator Things. *Jurnal METHODIKA*.
- Rohim, A., Sari, Y. A., & Tibyani. (2019). Convolution Neural Network (CNN) Untuk Pengklasifikasian Citra Makanan Tradisional. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Rosita, Y. D., & Sukmaningtyas, Y. N. (2020). LSTM Network and OCR Performance for Classification of Decimal Dewey Classification Code. *Record and Library Journal*.

- Rybalkin, V., & dkk. (2021). Efficient Hardware Architectures for 1D- and MD-LSTM Networks. *Journal of Signal Processing Systems*.
- Utomo, V. G., Pinem, A. P., & Christoko, B. V. (2019). Pengenalan Karakter Optis untuk Pencatatan Meter Air dengan Long Short Term Memory Recurrent Neural Network. *Jurnal Rekayasa Sistem dan Teknologi Informasi*.
- Wijaya, I., & Lubis, C. (2020). Pengimplementasian OCR menggunakan CNN untuk Ekstraksi Teks pada Gambar. *Jurnal Ilmu Komputer dan Sistem Informasi*.



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 : 2363/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 3 Juli 2024 perihal permohonan ujian skripsi:
 Nama : ALLIA RANIA
 NIM : 171402151
 Program Studi : Sarjana (S-1) Teknologi Informasi
 Judul Skripsi : Konversi Lembar Jawaban Ujian Mahasiswa Menjadi Dokumen Digital Menggunakan Convolutional Neural Network (CNN) dan Long Short-Term Memory
- 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 : Rossy Nurhasanah S.Kom., M.Kom
 NIP: 198707012019032016
- Anggota Penguji : Dr. Erna Budhiarti Nababan M.IT
 NIP: 196210262017042001
- Anggota Penguji : Ivan Jaya S.Si., M.Kom.
 NIP: 198407072015041001
- 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.

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

Medan, 04 Juli 2024
 Ditandatangani secara elektronik oleh:
 Dekan



Maya Silvi Lydia
 NIP 197401272002122001