

**EKSTRAKSI DATA E-KTP UNTUK PENCATATAN PESERTA
VAKSINASI COVID-19 DENGAN METODE YOLO V7
DAN PADDLE OCR BERBASIS ANDROID**

SKRIPSI

MUHAMMAD ELDWIN PASARIBU

181402086



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

EKSTRAKSI DATA E-KTP UNTUK PENCATATAN PESERTA
VAKSINASI COVID-19 DENGAN METODE YOLO V7
DAN PADDLE OCR BERBASIS ANDROID

SKRIPSI

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

MUHAMMAD ELDWIN PASARIBU
181402086



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

PERSETUJUAN

Judul : EKSTRAKSI DATA E-KTP UNTUK PENCATATAN PESERTA VAKSINASI COVID-19 DENGAN METODE YOLO V7 DAN PADDLE OCR BERBASIS ANDROID
Kategori : SKRIPSI
Nama : MUHAMMAD ELDWIN PASARIBU
Nomor Induk Mahasiswa : 181402086
Program Studi : SI TEKNOLOGI INFORMASI
Fakultas : ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA

Medan, 14 Juni 2024

Komisi Pembimbing :

Pembimbing II,

Annisa Fadhillah Pulungan, S.Kom., M.Kom.

NIP. 199308092020012001

Pembimbing I,

Ulfi Andayani, S.Kom., M.Kom.

NIP. 198604192015042004

Diketahui/disetujui oleh

Program Studi S1 Teknologi Informasi

Ketua,

Dedy Arisandi, S.T., M.Kom.
NIP. 1979060912009121002

PERNYATAAN

EKSTRAKSI DATA E-KTP UNTUK PENCATATAN PESERTA
VAKSINASI COVID-19 DENGAN METODE YOLO V7
DAN PADDLE OCR BERBASIS ANDROID

SKRIPSI

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

Medan, 14 Juni 2024

MUHAMMAD ELDWIN PASARIBU
181402086

UCAPAN TERIMA KASIH

Dengan penuh rasa syukur kepada Allah SWT., skripsi ini berhasil diselesaikan sebagai salah satu persyaratan untuk meraih gelar Sarjana Komputer di Program Studi S1 Teknologi Informasi, Fakultas Ilmu Komputer Dan Teknologi Informasi, Universitas Sumatera Utara..

Rasa terima kasih ini disampaikan kepada berbagai pihak yang telah memberi bimbingan serta doa kepada penulis, antara lain:

1. Ayahanda Ir. Achmad Syafwan dan Ibunda Henny Sopiani yang telah memberikan dukungan & doa, mulai dari awal kuliah hingga terselesaiannya pendidikan sarjana ini.
2. Bapak Prof. Dr. Muryanto Amin, S.Sos., M.Si. selaku Rektor Universitas Sumatera Utara.
3. Ibu Dr. Maya Silvi Lydia, M.Sc. Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
4. Bapak Dedy Arisandi, S.T., M.Kom. Ketua Program Studi S1 Teknologi Informasi sekaligus Dosen Pembanding I dan Bapak Ivan Jaya, S.Si., M.Kom. selaku serkretaris Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.
5. Ibu Ulfie Andayani, S.Kom., M.Kom selaku Dosen Pembimbing I dan Ibu Annisa Fadhillah Pulungan, S.Kom., M.Kom. selaku Dosen Pembimbing II yang telah membimbing, memberikan saran, motivasi serta semangat kepada penulis selama pengerjaan skripsi.
6. Ibu Ade Sarah Huzaifah, S.Kom., M.Kom selaku Dosen Pembanding II yang telah memberikan saran dan masukan sehingga skripsi ini bisa menjadi lebih baik.
7. Seluruh Dosen, pegawai dan staff Program Studi S1 Teknologi Informasi yang telah memberikan ilmu yang bermanfaat bagi penulis selama masa perkuliahan serta membantu segala pengurusan administrasi dan akademik selama perkuliahan.

8. Kawan-kawan Mahasiswa Teknologi Informasi USU 2018 yang telah membantu berjuang bersama penulis sejak maba sampai dengan penggerjaan tugas akhir, khususnya dari KOM B TI angkatan 2018.
9. Sobat Tembak Studio, Abhi R. P., Eric S. Simbolon, Farhan Alzuhri Nst., Choco, Hari D., Helmut S. Pakpahan, M. Arif P. Lubis, M. Alief Nuari, M. Daifullah, M. Khaffi Irwan, M. Luthfi Y., M. Raihansyah Lubis, M. Ridho, Rasyid Hafiz, Ruth C. Sianipar, Y. Felix Purba, Yedija P. Sipayung dan Zaki Afifi yang selalu membantu penulis mulai dari masa perkuliahan hingga penggerjaan skripsi ini.
10. Sahabat penulis Bang Andika Prayoga S.Kom, Celine Limois, Koh Ferdinand S.Psi., Faza Hakimul, Hazerd, Kevin Prayogo, Bang Jihat Medbagga A.Md., Bang Ronald, Bang Ridho A.Md., Rayhan Aulia S.E. & Shena yang selalu ada untuk penulis di waktu-waktu senggang, selalu memberikan semangat dan dukungan kepada penulis sehingga skripsi ini terselesaikan.
11. Kepada senior, junior dan teman-teman lainnya yang tidak dapat disebutkan satu persatu yang telah membantu dalam penyelesaian tugas akhir ini.

Dalam penulisannya, penulis sadar bahwa kekurangan dan kesalahan dalam penyusunannya. Oleh sebab itu diharapkan kritik, masukan maupun saran dapat menjadikan skripsi ini menjadi lebih baik.

Medan, 14 Juni 2024

Penulis

ABSTRAK

Kemajuan teknologi informasi telah memberikan kontribusi penting dalam membantu pekerjaan manusia. Misalnya, proses pendataan data yang masih dilakukan secara manual di lingkungan puskesmas menunjukkan bahwa penggunaan teknologi di Indonesia belum merata, terutama dalam proses vaksinasi COVID-19 beberapa waktu lalu. Akibatnya, proses pendataan memakan waktu lebih lama, menyebabkan keterlambatan dalam proses administrasi vaksinasi yang seharusnya cepat. Oleh sebab itu penelitian ini bertujuan untuk mendeteksi dan mengekstraksi data pada e-KTP untuk pada proses pencatatan peserta vaksinasi COVID-19 di puskesmas dengan menerapkan metode *You Only Look Once version 7* (YOLOv7) dan *library Paddle OCR* yang diimplementasikan secara mobile dengan sistem Android. Pada penelitian ini menggunakan 800 data citra e-KTP, dengan terdapat dua jenis pengujian yang dilakukan. Pertama, pendekripsi seluruh atribut data pada e-KTP mencapai akurasi yang sangat baik yaitu sebesar 98,19%. Kedua, pengujian kebenaran ekstraksi data teks e-KTP oleh *library Paddle OCR* menghasilkan akurasi yang juga sangat baik sebesar 94,45%. Sehingga penggunaan *Paddle OCR* dengan pendekatan deteksi YOLOv7 yang diimplementasikan dengan sistem Android dapat menjadi upaya solutif untuk memodernisasi tata kelola administratif di berbagai sektor pelayanan publik, terutama di lembaga-lembaga kesehatan.

Kata kunci : *Computer Vision, Deteksi e-KTP, You Only Look Once (YOLO) v7, Optical Character Recognition (OCR), Paddle OCR.*

E-KTP DATA EXTRACTION FOR RECORDING COVID-19

PARTICIPANTS WITH YOLO V7 AND PADDLE OCR

METHODS BASED ON ANDROID

ABSTRACT

The advancement of information technology has significantly contributed to assisting human tasks. For instance, the manual data recording process in health centers indicates that the use of technology in Indonesia is not evenly distributed, especially during the COVID-19 vaccination process some time ago. As a result, the data recording process takes longer, causing delays in the vaccination administration process that should be quick. Therefore, this research has an aim for to detected and extracted data from the electronic Kartu Tanda Penduduk (e-KTP) for the COVID-19 vaccination participant recording process at health centers by implementing the You Only Look Once version 7 (YOLOv7) method and the Paddle OCR library, implemented on the Android system. In this study, 800 e-KTP image data were used, and two types of tests were conducted. First, the detection of all attributes on the e-KTP achieved excellent accuracy, specifically 98.19%. Second, the truth extraction test of e-KTP attribute data by the PaddleOCR library resulted in equally excellent accuracy, reaching 94.45%. Thus, the use of PaddleOCR with the YOLOv7 detection approach implemented on the Android system can be a solutive effort to modernize administrative governance in various public service sectors, especially in healthcare institutions.

Keywords: Computer Vision, e-KTP Detection, You Only Look Once (YOLO) v7, Optical Character Recognition (OCR), Paddle OCR.

DAFTAR ISI

PERSETUJUAN	iii
PERNYATAAN.....	iv
UCAPAN TERIMA KASIH	v
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI.....	ix
DAFTAR TABEL	xiii
DAFTAR GAMBAR.....	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian.....	4
1.4 Batasan Masalah.....	4
1.5 Manfaat Penelitian.....	4
1.6 Metode Penelitian.....	5
1.6.1 <i>Studi Literatur</i>	5
1.6.2 <i>Pengumpulan Data</i>	5
1.6.3 <i>Analisis Permasalahan</i>	5
1.6.4 <i>Perancangan Sistem</i>	5
1.6.5 <i>Implementasi Sistem</i>	5
1.6.6 <i>Pengujian Sistem</i>	5
1.6.7 <i>Dokumentasi</i>	5
1.7 Sistematika Penulisan.....	6
BAB 2 LANDASAN TEORI	7
2.1 <i>Electronic - Kartu Tanda Penduduk (E-KTP)</i>	7
2.2 Program Vaksinasi COVID-19	8
2.3 <i>Computer Vision</i>	8
2.4 <i>Pre-processing</i>	9
2.4.1 <i>Background Removal</i>	9
2.4.2 <i>Image Resizing</i>	10
2.4.3 <i>Image Labelling</i>	10

2.5	<i>Convolutional Neural Network (CNN)</i>	11
2.6	<i>You Only Look Once (YOLO)</i>	13
2.6.1	<i>YOLO Versi 1</i>	15
2.6.2	<i>YOLO Versi 2</i>	16
2.6.3	<i>YOLO Versi 3</i>	16
2.6.4	<i>YOLO Versi 4</i>	16
2.6.5	<i>YOLO Versi 5</i>	17
2.6.6	<i>YOLO Versi 6</i>	17
2.6.7	<i>YOLO Versi 7</i>	17
2.7	Arsitektur YOLOv7.....	20
2.7.1	<i>Backbone</i>	21
2.7.2	<i>Neck</i>	22
2.7.3	<i>Head</i>	22
2.8	<i>Optical Character Recognition (OCR)</i>	23
2.9	<i>Parallel Distributed Deep Learning (Paddle)</i>	24
2.10	<i>Open Neural Network Exchange (ONNX)</i>	25
2.11	<i>React Native</i>	26
2.12	REST API.....	26
2.13	<i>Firebase</i>	27
2.14	<i>Confusion Matrix</i>	27
2.14.1	<i>Akurasi</i>	28
2.14.2.	<i>Precision</i>	28
2.14.3.	<i>Recall</i>	28
2.14.4.	<i>F1-Score</i>	29
2.15	Penelitian Terdahulu	29
2.15.1	<i>Perbedaan Penelitian</i>	31
BAB 3 ANALISIS DAN PERANCANGAN SISTEM	33
3.1	Data	33
3.2	Arsitektur Umum.....	34
3.3	<i>Image Acuisation</i>	35
3.4	<i>Image Pre-Processing</i>	35
3.4.1	<i>Background Removal</i>	36
3.4.3	<i>Image Resizing</i>	36
3.4.4	<i>Image Labeling</i>	37
3.5	YOLO V7	38

<i>3.5.1</i>	<i>Clone Repository</i>	39
<i>3.5.2</i>	<i>Konfigurasi Model</i>	39
<i>3.5.3</i>	<i>Proses Training</i>	39
<i>3.5.4</i>	<i>YOLOv7 Model</i>	40
<i>3.5.5.</i>	<i>Penghitungan Confidence Score</i>	41
<i>3.6</i>	<i>Paddle OCR</i>	42
<i>3.7</i>	<i>Open Neural Network Exchange (ONNX Model)</i>	43
<i>3.7.1</i>	<i>Proses Testing</i>	43
<i>3.8</i>	<i>Output</i>	44
<i>3.9</i>	<i>Activity Diagram</i>	45
<i>3.9.1</i>	<i>Proses Catat Peserta</i>	45
<i>3.9.2</i>	<i>Proses Menu Daftar Peserta</i>	46
<i>3.9.3</i>	<i>Proses Eksport Data</i>	48
<i>3.9.4</i>	<i>Proses Cara Penggunaan</i>	49
<i>3.9.5</i>	<i>Proses Tentang Aplikasi</i>	49
<i>3.10</i>	<i>Perancangan Antarmuka Sistem</i>	50
<i>3.10.1</i>	<i>Rancangan Tampilan Splashscreen</i>	50
<i>3.10.2</i>	<i>Rancangan Tampilan Home</i>	51
<i>3.10.3</i>	<i>Rancangan Tampilan Scan E-KTP</i>	51
<i>3.10.4</i>	<i>Rancangan Tampilan Daftar Peserta</i>	52
<i>3.10.5</i>	<i>Rancangan Tampilan Detail Peserta</i>	52
<i>3.10.6</i>	<i>Rancangan Tampilan Petunjuk Penggunaan Aplikasi</i>	53
<i>3.10.7</i>	<i>Rancangan Tampilan Tentang</i>	54
BAB 4 IMPLEMENTASI DAN PENGUJIAN SISTEM	55
<i>4.1</i>	<i>Implementasi Sistem</i>	55
<i>4.1.1</i>	<i>Lingkungan Implementasi</i>	55
<i>4.1.2</i>	<i>Implementasi Data</i>	56
<i>4.1.3</i>	<i>Pelatihan Sistem</i>	57
<i>4.2</i>	<i>Implementasi Perancangan Antarmuka</i>	59
<i>4.2.1</i>	<i>Tampilan Halaman Splashscreen</i>	59
<i>4.2.2</i>	<i>Tampilan Halaman Home</i>	60
<i>4.2.3</i>	<i>Tampilan Halaman Scan</i>	61
<i>4.2.4</i>	<i>Tampilan Halaman Daftar Peserta</i>	62
<i>4.2.5</i>	<i>Tampilan Detail Peserta</i>	63
<i>4.2.6</i>	<i>Tampilan Halaman Cara Penggunaan</i>	64

4.2.7	<i>Tampilan Halaman Tentang</i>	65
4.3	Pengujian Sistem	66
4.4	Penghitungan Hasil Uji Deteksi Data E-KTP	77
4.5	Penghitungan Hasil Uji Ekstraksi	79
BAB 5 KESIMPULAN DAN SARAN	83
5.1	Kesimpulan.....	83
5.2	Saran.....	84
DAFTAR PUSTAKA	85
LAMPIRAN	88
	Lampiran 1 : Hasil Deteksi Atribut Data E-KTP	88
	Lampiran 2. Hasil Kebenaran Ekstraksi Data Teks E-KTP	92

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	30
Tabel 3.1 Jumlah Dataset Penelitian	33
Tabel 3.2 Pembagian Dataset	35
Tabel 4.1 <i>Training</i> Model	58
Tabel 4.2 Hasil Pengujian	67
Tabel 4.3 Hasil TP,FP dan FN Deteksi Data E-KTP	77
Tabel 4.4 Hasil Penghitungan <i>Performance Matrix</i> Deteksi Data	78
Tabel 4.5 Hasil TP,FP dan FN Ekstraksi Data E-KTP	80
Tabel 4.6 Hasil Penghitungan <i>Performance Matrix</i> Deteksi Data	81
Tabel 4.7 Hasil <i>Performance Matrix</i> Pengujian	82

DAFTAR GAMBAR

Gambar 2.1 Electronic-Kartu Tanda Penduduk	7
Gambar 2.2 Arsitektur CNN (medium.com)	12
Gambar 2.3 Pooling Layer Operation (Bui & Chang, 2016)	12
Gambar 2.4 Cara Kerja Deteksi YOLO <i>method</i> (Redmon et al., 2016)	14
Gambar 2.5 Perkembangan YOLO (Ivanov, 2022)	15
Gambar 2.6 Translasi citra menjadi vector (Reddy, 2021)	18
Gambar 2.7 Objek dideteksi (Reddy, 2021)	18
Gambar 2.8 Objek tak terdeteksi (Reddy, 2021)	18
Gambar 2.9 Prediksi <i>Bounding Box</i> (Reddy, 2021)	19
Gambar 2.10 Arsitektur YOLO v7(roboflow.com)	20
Gambar 2.11 Arsitektur E-ELAN(Wang et al., 2022)	20
Gambar 2.12 Head YOLOv7 (Wang et al., 2022)	22
Gambar 2.13 Tahapan Sistem OCR (Jain et al., 2013)	22
Gambar 2.14 Kombinasi YOLO dan Tesseract (Purohit, 2019)	23
Gambar 2.15 <i>Confusion Matrix</i> Dua Kelas (towardsdatascience.com)	27
 Gambar 3.1 Arsitektur Umum	34
Gambar 3.2 E-KTP yang terkumpul	35
Gambar 3.3 <i>Background Removal</i> e-KTP	36
Gambar 3.4 <i>Resizing</i> 323x204 pixel	37
Gambar 3.5 Proses di <i>Labelling</i>	38
Gambar 3.6 Hasil <i>Labeling</i>	38
Gambar 3.7 <i>Clone repository</i> YOLOv7	39
Gambar 3.8 Konfigurasi YOLOv7 untuk Deteksi e-KTP	39
Gambar 3.9 <i>Training</i> YOLOv7	40
Gambar 3.10 <i>Bounding Boxes</i> YOLOv7	41
Gambar 3.11 Ekstraksi <i>Paddle OCR</i>	43
Gambar 3.12 <i>Database Firebase</i>	44

Gambar 3.13 Hasil Ekstraksi Data Peserta di Excel	45
Gambar 3.14 <i>Activity Diagram</i> Proses Catat Peserta	46
Gambar 3.15 <i>Activity Diagram</i> Proses Daftar Peserta	47
Gambar 3.16 <i>Activity Diagram</i> Proses Eksport Data	48
Gambar 3.17 <i>Activity Diagram</i> Proses Cara Penggunaan	49
Gambar 3.18 <i>Activity Diagram</i> Proses Tentang Aplikasi	50
Gambar 3.19 Rancangan Splashscreen Aplikasi	51
Gambar 3.20 Rancangan Tampilan <i>Home</i>	51
Gambar 3.21 Rancangan Tampilan <i>Scan e-KTP</i>	52
Gambar 3.22 Rancangan Tampilan Daftar Peserta	52
Gambar 3.33 Rancangan Detail Peserta	53
Gambar 3.24 Rancangan Tampilan Cara Penggunaan	53
Gambar 3.25 Rancangan Tampilan Tentang	54
Gambar 4.1 Grafik Pembagian Dataset Penelitian	56
Gambar 4.2 Dataset YOLO Untuk Proses <i>Training</i>	57
Gambar 4.3 Proses <i>Training</i> YOLOv7 di Colab	57
Gambar 4.4 Grafik hasil <i>training batch 8</i> dan <i>epoch 200</i>	59
Gambar 4.5 Kurva <i>Precision-Recall</i>	59
Gambar 4.6 Halaman <i>Splashscreen</i>	60
Gambar 4.7 Halaman <i>HomeScreen</i>	61
Gambar 4.8 Halaman <i>Scan</i>	62
Gambar 4.9 Halaman Daftar Peserta	63
Gambar 4.10 Halaman Hasil Detail Peserta	64
Gambar 4.11 Halaman Cara Penggunaan	65
Gambar 4.12 Halaman Tentang Aplikasi	66
Gambar 4.13 Grafik Persentase Hasil Deteksi E-KTP	75
Gambar 4.14 Grafik Persentase Hasil Ekstraksi E-KTP	76

BAB I

PENDAHULUAN

1.1 Latar Belakang

Wabah Penyakit *Coronavirus Disease* (COVID-19) telah diakui oleh Organisasi Kesehatan Dunia (WHO) sebagai Kedaruratan Kesehatan Masyarakat yang Meresahkan Dunia pada 30 Januari 2020 (Harapan et al, 2020). Untuk mengendalikan penyebaran virus ini, negara-negara di seluruh dunia telah melaksanakan program vaksinasi sebagai langkah pencegahan untuk mengurangi laju peningkatan kasus positif COVID-19. Di Indonesia, per tanggal 6 Juni 2022, vaksinasi telah diberikan kepada 96% dari target vaksinasi Indonesia untuk vaksin dosis pertama, 80% untuk dosis kedua, dan masih 22% untuk dosis ketiga atau penguat (*booster*).

Dalam proses vaksinasi di Indonesia terutama di puskesmas, peserta terlebih dahulu datang, mengambil nomor antrian, mengantri begitu lama hingga saatnya nomor antrian dipanggil kemudian menuju ke meja pendataan untuk dilakukan pendataan dengan menunjukkan e-KTP. Petugas menulis data dari e-KTP peserta ke berkas formulir pendaftaran vaksinasi secara manual. Setelah itu petugas kembali melakukan penginputan data peserta kembali ke aplikasi pendataan Kemenkes. Hal ini membuat terjadi 2x input data peserta yang tentunya memerlukan waktu yang seharusnya proses ini bisa lebih cepat dikarenakan penumpukan peserta di bagian pendaftaran tidaklah baik dan dapat dikhawatirkan terjadi penyebaran virus saat itu.

Karena pencatatan masih dilakukan secara manual, maka diperlukan suatu sistem yang dapat memperoleh data dari e-KTP peserta untuk administrasi data peserta vaksinasi secara otomatis, tanpa menunggu petugas menuliskan satu per satu data ke formular administrasi sehingga membantu petugas kesehatan dalam proses pencatatan peserta vaksinasi.

E-KTP (*electronic* Kartu Tanda Penduduk) merupakan kartu identitas yang dimiliki oleh masyarakat yang memuat identitas lengkap mengenai data pribadi individu, mulai dari Nomor Induk Kependudukan (NIK), nama, pekerjaan hingga alamat, pekerjaan serta masa berlaku kartu. E-KTP sering dibawa untuk melakukan pendaftaran ataupun pendaftaran salah satunya ketika pencatatan data peserta vaksinasi COVID-19. Adapun e-KTP itu sendiri sudah disebut menggunakan teknologi elektronik dalam pembuatannya namun elektronik yang dimaksud hanya sekedar penyimpanan data fisik individu seperti bola mata, sidik jari dan tanda tangan. Sehingga dalam penerapannya, pemerintah belum 100% menggunakan teknologi elektronik pada kartu e-KTP yang kita miliki untuk dimanfaatkan dalam administrasi penduduk pada instansi pemerintahan . Dengan *computer vision* data e-KTP dapat diekstrak menjadi data digital seperti penelitian yang (Nguyen et al., 2018), membangun sistem ekstraksi informasi dari kartu identitas data penduduk di Vietnam. Sehingga dapat membantu mempercepat proses administrasi penduduk.

Adapun teknologi *Optical Character Recognition* (OCR) dapat mengekstraksi dan merekognisi data dari suatu citra gambar kedalam bentuk teks, dapat dikombinasikan dengan metode YOLO. Sebagai contoh penelitian oleh Aprilino dan Amin pada tahun 2022 telah membuat suatu sistem yang dapat mendekripsi dan mengekstraksi data teks dari plat kendaraan yang ada di jalanan raya yang memiliki akurasi sebesar 100% untuk YOLO dengan *threshold* 0,5 serta akurasi sebesar 92% untuk *library* Tesseract OCR. (Aprilino, 2022)

Metode *You Only Look Once* (YOLO) merupakan suatu metode yang dapat melakukan pendekripsi objek yang terdapat pada citra gambar secara cepat dan tepat termasuk teks, terutama untuk YOLOv7. Dengan YOLOv7 proses deteksi objek dapat dilakukan dengan efektif dan efisien sehingga banyak dimanfaatkan untuk teknologi pengembangan deteksi objek. Sebagai contoh penelitian oleh Do dkk yang melakukan deteksi atribut data pada kartu tanda penduduk Vietnam dengan YOLOv7 dan VietOCR dapat mendekripsi teks serta mengekstraksi data secara otomatis dengan akurasi sebesar 98,19% (Do et al., 2023).

Penelitian lainnya dilakukan oleh Syahputra pada tahun 2019 juga melakukan ekstraksi data dari e-KTP yang kemudian data tersebut disimpan menjadi pencatatan daftar rekrutmen anggota partai politik. Kombinasi antara metode OCR dan LSTM

dinilai sangat baik untuk melakukan pemindaian data secara akurat pada tiap-tiap variable data sehingga menghasilkan akurasi sebesar 91,42% (Syahputra, 2019).

Kemudian ada penelitian dengan judul oleh Kim dkk yang Mendeteksi tanggal kadaluarsa pada produk yang kemudian memberikan notifikasi kepada para penyandang tunanetra sehingga mengetahui kapan produk mereka kadaluarsa. Dengan metode OCR dan YOLO, sistem mendapatkan akurasi sebesar 86% dengan waktu yaitu 22,20 detik (Kim et al., 2021).

Dan penelitian oleh Zhang dan Li pada tahun 2023 menerapkan teknologi *Paddle* OCR yang menjadi referensi penulis dalam penelitian ini. Adapun penelitian mereka ini mengusulkan metode pengenalan kode kesehatan yakni untuk pengenalan teks dalam gambar menggunakan algoritma lokasi target untuk menghasilkan teks yang dibutuhkan dan menerapkan hasil penelitian kedalam sistem pencegahan dan pengendalian epidemi di lingkungan kampus di Tiongkok. Hasil akurasi dari *Paddle* OCR dalam merekognisi kode adalah sebesar 97,34% (Li & Zhang, 2023).

Berdasarkan yang telah dipaparkan diatas diperlukan suatu solusi yang dapat membuat pelayanan vaksinasi COVID-19 di lingkungan puskesmas menjadi lebih baik, efektif dan efisien yaitu dengan membuat sistem pencatatan peserta vaksinasi COVID-19 secara digital berbasis Android guna menciptakan ekosistem pelayanan publik yang cepat dan tepat bagi instansi kesehatan di tanah air. Pada penelitian ini membahas mengenai teknologi ekstraksi dan rekognisi informasi dari suatu citra menjadi teks data dengan metode *Paddle Optical Character Recognition* (OCR) dan *You Only Look Once* (YOLO), sehingga judul “EKTRAKSI DATA E-KTP UNTUK PENCATATAN PESERTA VAKSINASI COVID-19 DENGAN METODE YOLO V7 DAN PADDLE OCR BERBASIS ANDROID” penulis angkat sebagai penelitian ini.

1.2 Rumusan Masalah

Proses manual dalam proses administrasi pasien di lingkungan instansi kesehatan yang salah satunya adalah pencatatan peserta vaksinasi COVID-19 di puskesmas yang lamban telah menyebabkan penumpukan antrian peserta di bagian pendataan. Selain itu, pemanfaatan teknologi e-KTP dalam birokrasi pelayanan publik di Indonesia khususnya instansi kesehatan belum optimal. Oleh karena itu, diperlukan sistem yang

dapat memberikan bantuan kepada petugas kesehatan puskesmas secara mudah, cepat, dan otomatis. Sistem ini akan menggunakan teknologi aplikasi Android untuk pencatatan digital yang lebih efisien.

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah untuk mengembangkan sebuah sistem berbasis android yang dapat digunakan untuk memperoleh data, mendekripsi dan mengekstraksi data pada e-KTP untuk pada proses pencatatan peserta vaksinasi COVID-19 di puskesmas dengan menerapkan metode *You Only Look Once version 7* (YOLOv7) dan *PaddleOCR*

1.4 Batasan Masalah

Adapun batasan masalah dalam penelitian ini yang antara lain:

1. Pengambilan data diambil dengan menggunakan kamera *smartphone* dengan resolusi minimal 12 *megapixel* dengan jarak pengambilan memiliki *range* 9-12 cm dan *angle* 90°.
2. Data citra yang digunakan berekstensi .jpg.
3. Citra yang diperoleh mendapatkan intensitas cahaya yang cukup dengan intensitas cahaya ruangan sekitar 560-1200 lumen tanpa membelakangi sumber cahaya.
4. Sistem tidak terhubung dengan data kependudukan di Ditjen Catpil Kemendagri.

1.5 Manfaat Penelitian

1. Mempermudah petugas kesehatan untuk melakukan pencatatan dan penyimpanan informasi data peserta.
2. Mempersingkat waktu yang diperlukan dalam proses pencatatan data peserta vaksinasi COVID-19 di lingkungan puskesmas.
3. Sebagai digitalisasi dan pemanfaatan teknologi informasi untuk pelayanan publik pemerintah khususnya di bidang kesehatan.
4. Sebagai referensi untuk penelitian selanjutnya mengenai kombinasi metode *PaddleOCR* dan YOLOv7 .

1.6 Metode Penelitian

Langkah-langkah dilakukan pada penelitian ini antara lain :

1.6.1 *Studi Literatur*

Tahap ini mengumpulkan informasi dan referensi yang ada terkait dari e-book, jurnal publikasi, karya ilmiah, berita online serta sumber informasi lain yang kredibel, memiliki hubungan dengan materi *Computer Vision*, *Optical Character Recognition* (OCR) dan *You Only Look Once* (YOLO).

1.6.2 *Pengumpulan Data*

Data yang dikumpulkan adalah data citra e-KTP yang dikumpulkan dari berbagai masyarakat secara manual baik langsung maupun tidak serta dari data penelitian terdahulu.

1.6.3 *Analisis Permasalahan*

Tahapan ini dilakukan evaluasi terhadap informasi dan referensi dari studi literatur. Tujuannya adalah untuk memperoleh pemahaman yang mendalam mengenai metode yang hendak diterapkan dalam penelitian ini, yaitu *You Only Look Once version 7* (YOLOv7) dan *Paddle Optical Character Recognition* (OCR). Metode ini akan digunakan untuk menyelesaikan tantangan ekstraksi data informasi dari e-KTP guna pencatatan peserta vaksinasi COVID-19 berbasis Android.

1.6.4 *Perancangan Sistem*

Proses ini bertujuan untuk menanggulangi permasalahan yang diidentifikasi dalam tahap analisis. Setelah itu, dilakukan perancangan sistem yang akan dibangun. Struktur umum dari perancangan solusi sistem penelitian.

1.6.5 *Implementasi Sistem*

Pada fase ini, sistem diwujudkan ke dalam kode pemrograman dan terintegrasi dengan berbagai perangkat yang diperlukan, memastikan bahwa implementasi sistem dapat diselesaikan dengan baik.

1.6.6 *Pengujian Sistem*

Setelah menyelesaikan konstruksi sistem, tahap berikutnya adalah pengujian sistem untuk memverifikasi kinerja dan keberfungian sistem dalam menangani permasalahan yang telah diteliti.

1.6.7 *Dokumentasi*

Pada langkah ini, laporan dibuat dan disusun berdasarkan hasil analisis, pelaksanaan, implementasi, dan pengujian sistem.

1.7 Sistematika Penulisan

Terdapat bab-bab dalam sistematika penulisan penelitian ini, antara lain :

BAB 1: Pendahuluan

Mencakup pendahuluan, perumusan permasalahan, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian, dan tata cara penulisan.

BAB 2: Landasan Teori

Mencakup penjelasan tentang materi pendukung yang bersumber dari buku, jurnal publikasi, *e-book*, artikel internasional dan informasi referensi lainnya untuk memperkuat penelitian ini. Teori mengenai metode yang digunakan tersebut antara lain *Optical Character Recognition (OCR)*, *You Only Look Once version 7 (YOLOv7)*, dan Sistem Android.

BAB 3: Analisis dan Perancangan Sistem

Bagian ini menjelaskan penjabaran dari arsitektur umum penelitian yakni sistem ekstraksi data e-KTP untuk pencatatan peserta vaksin COVID-19 puskesmas dengan metode YOLOv7 dan *PaddleOCR*.

BAB 4: Implementasi dan Pengujian Sistem

Menjabarkan penerapan dari analisis penelitian serta bagaimana sistem dirancang, bersama dengan pengujian sistem yang dikembangkan.

BAB 5: Kesimpulan dan Saran

Menjabarkan hasil penelitian dan memberikan rekomendasi kepada pembaca, terutama bagi peneliti yang akan datang.

BAB 2

LANDASAN TEORI

2.1 Electronic - Kartu Tanda Penduduk (E-KTP)

E-KTP merupakan kartu identitas yang dikeluarkan secara elektronik dengan menggunakan sistem komputarisasi. Didasarkan kepada dengan Undang-Undang (UU) tentang administrasi penduduk, No. 24 tahun 2013, e-KTP adalah bukti resmi kependudukan, diterbitkan disdukcapil sebagai kartu identitas penduduk resmi negara yang dilengkapi dengan *chip*. Gambar 2.1 merupakan contoh e-KTP.



Gambar 2.1 *Electronic-Kartu Tanda Penduduk*

Penetapan hak akses dan penggunaan data kependudukan diatur secara teknis melalui ketentuan Permendagri Nomor 102 Tahun 2019. Pemanfaatan data diri penduduk yang termuat dalam e-KTP oleh lembaga pemerintah, sebagai alat identifikasi resmi, memberikan kemudahan akses layanan publik, termasuk di antaranya di sektor pendidikan, kesehatan, perbankan, subsidi, asuransi, bantuan sosial dan lain-lain.

2.2 Program Vaksinasi COVID-19

Program ini adalah inisiatif massal yang diimplementasikan oleh pemerintah di seluruh dunia sebagai langkah untuk mengakhiri pandemi virus *corona* (COVID-19) dan meningkatkan kekebalan kelompok masyarakat (Chen et al., 2022). Vaksin adalah zat biologis mengandung *antigen*, seperti mikroorganisme lemah ataupun sudah mati, bagian dari mikroorganisme, atau toksin mikroorganisme yang telah diolah menjadi protein rekombinan atau toksoid. Ketika diberikan kepada individu, vaksin mendorong penciptaan kekebalan imuniti yang spesifik terhadap penyakit virus *sarscov-2*.

Per tanggal 9 Juni 2022, program ini telah diberikan kepada masyarakat sebanyak 200.754.718 dosis atau sekitar 96,39% dari sasaran vaksinasi untuk vaksin dosis, sebanyak 167.948.670 dosis atau 80,64% dari sasaran vaksinasi untuk dosis 2 serta masih sebanyak 47.306.542 dosis atau hanya 22,71% dari sasaran vaksinasi untuk dosis 3. Program vaksinasi ini dianggarkan oleh pemerintah pusat melalui Kementerian Kesehatan Republik Indonesia yang kemudian disebarluaskan ke seluruh penjuru tanah air sehingga masyarakat sebagaimana berdasarkan pada ayat 3 pada pasal 3 peraturan Menteri Kesehatan RI tidak dipungut bayaran atau gratis.

Pelayanan program vaksinasi dilaksanakan di fasilitas pelayanan kesehatan milik pemerintah, baik pusat ataupun daerah serta dari pihak swasta yang memenuhi syarat. Fasilitas tersebut berdasarkan PermenkesRI 2021 antara lain :

- a) Puskesmas dan puskesmas pembantu.
- b) Rumah sakit.
- c) Klinik.
- d) Unit pelayanan kesehatan pada kantor kesehatan di bandara dan pelabuhan.
- e) Pos pelayanan vaksinasi COVID-19 yang telah bekerjasama dengan puskesmas, dinas kesehatan kab/kota dan provinsi setempat.

2.3 Computer Vision

Computer vision adalah teknologi untuk memungkinkan mesin untuk melihat dan memahami gambar dan video. *Computer vision* bertujuan untuk mengembangkan algoritma dan model *machine learning* yang dapat memproses gambar dan video untuk menghasilkan informasi yang dapat diinterpretasikan oleh mesin.

Berikut merupakan bagian dari *computer vision* meliputi :

- Pengolahan Citra: yaitu suatu teknik untuk memproses gambar agar dapat diinterpretasikan oleh mesin. Teknik pengolahan citra meliputi segmentasi citra, ekstraksi fitur, dan klasifikasi citra (Szeliski, 2022).
- Pemodelan Geometrik: yakni suatu teknik untuk memodelkan objek di dalam gambar dan video. Pemodelan geometrik meliputi pengenalan objek, deteksi objek, pelacakan objek, dan rekonstruksi 3D (Hartley & Zisserman, 2003).
- Pembelajaran Mesin, adalah metode untuk mengembangkan model *machine learning* yang mampu melakukan pemrosesan terhadap gambar dan video. Jenis pembelajaran mesin mencakup terawasi, tanpa pengawasan, dan penguatan (Forsyth & Ponce, 2002).

2.4 *Pre-processing*

Preprocessing merupakan serangkaian teknik dan langkah-langkah yang diterapkan pada citra atau data visual sebelum diinput ke dalam model *computer vision*. Tujuan dari *preprocessing* adalah untuk meningkatkan kualitas data, memudahkan ekstraksi fitur, dan memungkinkan model untuk menghasilkan hasil yang lebih baik. Berikut adalah beberapa konsep teoritis terkait *preprocessing* dalam penelitian ini:

2.4.1 *Background Removal*

Background removal merupakan proses segmentasi citra dengan menghilangkan bagian latar belakang dari suatu citra untuk meninggalkan hanya objek yang diinginkan saja yang akan diproses kedepan. Tujuannya adalah untuk membuat objek menjadi fokus utama tanpa gangguan latar belakang. Adapun Langkah-langkah dalam prosesnya antara lain (Kumar, 2022):

- *Read the image*, yaitu menggunakan *image processing library* untuk nantinya dapat membaca citra dan melakukan proses *background removal*, contohnya opencv.
- Selanjutnya segmentasi, yaitu mengidentifikasi rentang warna background pada citra dalam ruang warna RGB menggunakan metode cv2.inRange di OpenCV.

- *Create Mask*, yaitu membuat *mask* berdasarkan segmentasi warna. Hal ini harus menonjolkan latar belakang dan membiarkan latar depan citra berwarna hitam.
- *Invert Mask*, yaitu melakukan pembalikan *mask* biner sehingga latar belakang menjadi hitam dan citra yang ingin diproses menjadi putih
- *Apply Mask to the original image*, yaitu menggunakan *inverted mask* sebelumnya untuk memperbanyak citra, menghilangkan latar belakang secara efektif. Hal ini dapat dilakukan dengan operasi *bitwise* seperti cv2.bitwise_
- *Save the result*. Yakni menyimpan citra yang telah diproses dan melihat hasil penghapusan latar belakang.

2.4.2 Image Resizing

Proses ini menyamakan ukuran setiap gambar sehingga masing-masing data citra memiliki ukuran yang sama besar. Selain itu, proses ini juga dilakukan untuk mengurangi waktu eksekusi citra pada saat proses *training*. Berikut merupakan persamaan untuk proses *resizing* citra seperti tampak pada persamaan 2.1 untuk lebar dan persamaan 2.2 untuk panjang.

$$\text{width} = \frac{\text{original width}}{\text{original height}} \times \text{new height} \quad (2.1)$$

$$\text{height} = \frac{\text{original height}}{\text{original width}} \times \text{new width} \quad (2.2)$$

Berikut merupakan langkah melakukan image resizing:

- *Read the image* dengan *library computer vision* seperti OpenCV lalu menentukan direktoriya
- Menentukan ukuran Panjang dan lebar tujuan *resiizing* lalu *resize* sesuai ukuran targetnya.
- Menyimpan citra yang sudah di-*resize* dengan cv2.imwrite.

2.4.3 Image Labelling

Image labeling dalam konteks *preprocessing* mengacu pada proses menandai atau memberikan label pada objek atau *area-of-interest* (AOI) tertentu dalam sebuah citra. Tujuan utama dari *image labeling* adalah untuk memberikan informasi tentang lokasi atau keberadaan objek tertentu dalam suatu gambar. *Image labeling* seringkali

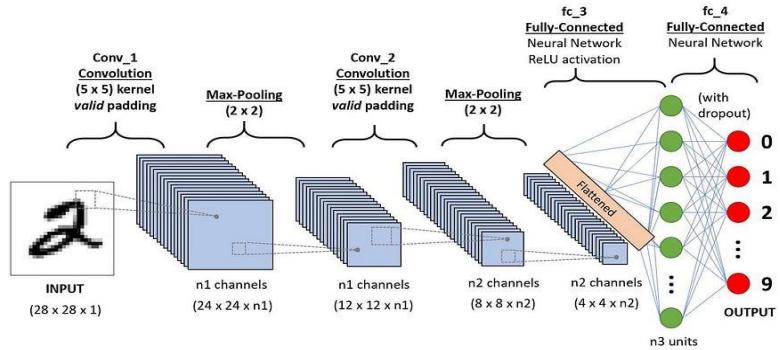
merupakan langkah penting dalam mempersiapkan data untuk tugas-tugas *computer vision* seperti deteksi objek atau klasifikasi. Beberapa langkah yang umumnya terkait dengan *image labeling* termasuk:

- *Bounding Box Labeling*: Pada deteksi objek, *image labeling* dapat dilakukan dengan menentukan batasan (*bounding box*) yang mengelilingi objek tersebut. *Bounding box* biasanya didefinisikan oleh empat koordinat (x min, y min, x max, y max) yang menunjukkan batas area objek.
- *Semantic Segmentation*: *Image labeling* juga dapat berarti memberikan label pada setiap piksel dalam citra, menunjukkan kelas atau kategori objek yang diwakili oleh piksel tersebut. Ini sering disebut sebagai *semantic segmentation*.
- *Instance Segmentation*: Jika ada lebih dari satu objek dari kelas yang sama dalam citra, *instance segmentation* memberikan label unik pada setiap objek tersebut. Setiap objek memiliki label yang berbeda, bahkan jika mereka termasuk dalam kelas yang sama.
- *Landmark Labeling*: Pada beberapa kasus, *image labeling* juga dapat mencakup menandai *landmark* atau titik-titik penting pada objek, seperti sudut wajah atau ujung objek tertentu.

Pemberian label pada citra dapat dilakukan secara manual atau menggunakan metode otomatis melalui algoritma komputer *vision*. *Image labeling* membantu model *machine learning* untuk memahami dan mempelajari karakteristik objek dalam citra, yang diperlukan untuk berbagai tugas seperti deteksi objek, segmentasi, atau klasifikasi.

2.5 Convolutional Neural Network (CNN)

CNN ialah algoritma objek deteksi dalam citra. Dimana merupakan perkembangan dari *Multi Layer Perceptron* (MLP). Algoritma ini umumnya digunakan untuk memproses data berdimensi dua. CNN adalah algoritma jaringan saraf yang memiliki sejumlah lapisan yang mencakup lapisan *Convolutional*, *pooling*, dan *fully connected* (Hu et al., 2015).

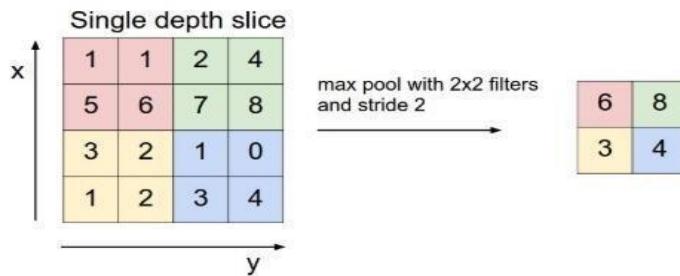


Gambar 2.2 Arsitektur CNN (medium.com)

Convolutional layer adalah lapisan inti dalam *Convolutional Neural Network*, terdiri dari serangkaian filter yang dapat disesuaikan. Pada tahap ini, gambar diekstraksi dari *input volume* dan kemudian disampaikan ke lapisan berikutnya untuk pengekstraksian fitur yang lebih rumit (Bui & Chang, 2016).

Ada 4 tahapan dalam fungsi CNN yang dapat dibagi menjadi:

1. *Layer Input*, tahap ini melibatkan penyimpanan *pixel* dari citra yang akan digunakan sebagai input.
2. *Layer Konvolusi*, pada tahap ini CNN menghitung hasil neuron yang akan terhubung dengan wilayah lokal pada citra input dengan memperhitungkan bobot dan area tertentu.
3. *Layer Pooling*, tahap ini mengurangi dimensi dengan melakukan *downsampling*, meningkatkan efisiensi komputasi dengan mengurangi jumlah parameter, dan membantu mengatasi *overfitting*, di mana model dapat melakukan prediksi yang baik pada data pelatihan tetapi tidak pada data baru.
4. *Layer Fully Connected*, pada tahap ini setiap neuron dalam lapisan ini terhubung dengan setiap terkoneksi neuron dalam lapisan sebelumnya.



Gambar 2.3 Pooling Layer Operation (Bui & Chang, 2016)

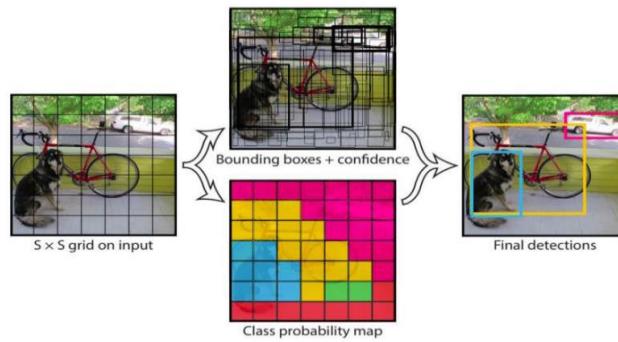
4. *Layer Fully Connected*, pada tahap ini setiap neuron dalam lapisan ini terhubung dengan setiap terkoneksi neuron dalam lapisan sebelumnya.

2.6 You Only Look Once (YOLO)

Algoritma YOLO adalah algoritma *objek detection* yang melakukan pengklasifikasian ulang untuk melakukan sebuah pendekatan pada sebuah objek, diperkenalkan oleh Joseph Redmon dan tim pada 2016 (Redmon et al., 2016). Algoritma ini memanfaatkan pendekatan dari jaringan syaraf tiruan untuk mengidentifikasi objek yang ada dalam citra. Pendekatan ini melibatkan pembagian citra ke dalam beberapa wilayah yang memfasilitasi proses klasifikasi deteksi objek .Kecepatan deteksinya sangat tinggi dan telah diperbarui dan terus hingga sekarang dengan kecepatan yang semakin meningkat (Redmon et al., 2016). YOLO membagi citra input menjadi kotak-kotak berukuran $S \times S$. Setiap kotak prediksi bertanggung jawab untuk mendeteksi objek yang mungkin terdapat di dalamnya (Chablani, 2017). Berikut adalah tahapan cara kerja dari metode YOLO: (Rahyagara, 2018)

1. Penggunaan dataset sebagai penyimpan informasi yang berkaitan dengan objek yang akan dideteksi.
2. Proses annotasi citra melibatkan *labelling* dengan menentukan kotak pembatas dan memberikan nama kelas untuk setiap objek dalam citra.
3. Pada tahap pelatihan, beberapa konsep yang perlu dipahami termasuk:
 - *Batch size*, yang merupakan jumlah data yang digunakan pada tiap iterasi *training*;
 - *Epoch*, yang menunjukkan jumlah iterasi dalam proses pelatihan.
 - *Learning rate*, yang mengatur seberapa cepat proses pelatihan berlangsung; learning rate mempengaruhi cepat model akan belajar mendeteksi objek

Proses pendekatan oleh YOLO yakni menggunakan *network* untuk membagi citra ke tiap-tiap kisi box dengan ukuran $S \times S$ pada citra seperti tampak pada gambar 2.4. Apabila target mengenai kotak maka akan mendekati target tersebut dengan memperkirakan *bounding box* + *confidence scores* nya beserta dengan peta probabilitasnya (*class probability map*) (Redmon et al., 2016).



Gambar 2.4 Cara Kerja Deteksi YOLO method (Redmon et al., 2016).

Berdasarkan Gambar 2.4, terdapat beberapa alur YOLO untuk melakukan pendektsian objek. yakni:

- Pendeteksian objek dilakukan dengan cara membagi *image* menjadi kotak-kotak berukuran $s \times s$. Setiap kotak-kotak ini disebut sebagai *bounding box*. *Bounding box* akan memprediksi apakah kotak tersebut berisi objek atau tidak. Prediksi ini dinyatakan sebagai nilai *Confidence* yang merupakan ukuran keyakinan bahwa *bounding box* berisi objek sesuai dengan rencana dan akurasi prediksi. Persamaan nilai *Confidence* dapat dinyatakan pada persamaan 2.3..

$$\text{Confidence} = \Pr(\text{class}) \times \text{IoU}_{\text{Pred}}^{\text{Truth}} \quad (2.3)$$

$\Pr(\text{Class})$ adalah probabilitas objek muncul di suatu wilayah kotak. IOU (*Intersection of Union*) adalah rasio tumpang tindih antara kotak prediksi dan kotak *Ground Truth*. Semakin besar nilai IOUPredTruth , maka semakin tinggi tingkat akurasi deteksi objek..

Persamaan IOU yaitu seperti pada persamaan 2.4.

$$\text{IoU}_{\text{Truth}}^{\text{Pred}} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \quad (2.4)$$

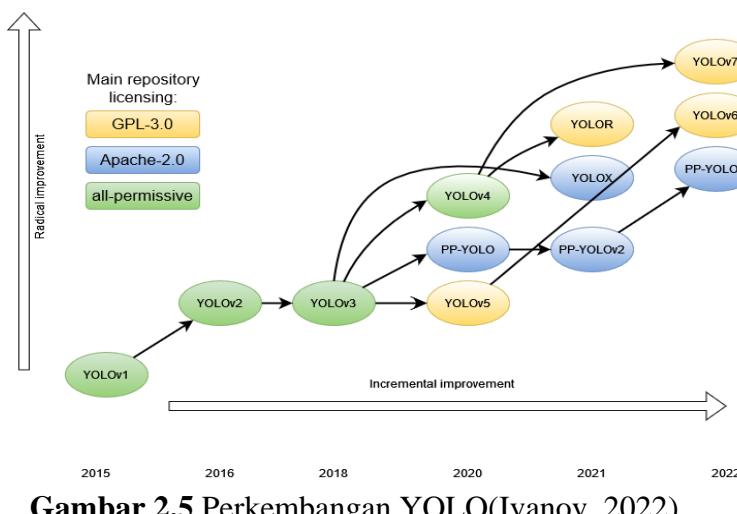
- Terdapat lima variable pada Bounding box objek yaitu x , y , w , h , dan c . Adapun berikut merupakan keterangannya:
 - x = titik tengah dari koordinat x
 - y = titik tengah dari koordinat y
 - w = lebar *bounding box*
 - h = tinggi *bounding box*
 - c = *confidence score*

- Jika terdapat objek di dalam grid, grid tersebut akan memprediksi nilai probabilitas kelas objek tersebut. Nilai probabilitas kelas dan nilai Confidence dari bounding box dikalikan untuk menghasilkan nilai Confidence pada bounding box tersebut untuk kelas objek tertentu. Nilai Confidence ini dapat digunakan untuk menentukan kelas objek yang paling mungkin berada di bounding box tersebut. Rumusnya seperti pada persamaan 2.5.

$$\Pr(\text{Class}_i | \text{Object}) \times \Pr(\text{object}) \times IoU_{\text{Truth}}^{\text{Pred}} = \Pr(\text{Class}_i) \times IoU_{\text{Truth}}^{\text{Pred}} \quad (2.5)$$

2.6.1 Perkembangan YOLO

Seiring waktu, YOLO terus berkembang dan Gambar 2.5 menunjukkan perkembangan YOLO *family* dari 2015 hingga sekarang



2.6.1.1 YOLO Versi 1

YOLOv1 adalah versi pertama dari YOLO yang diperkenalkan pada tahun 2015. Pada saat itu, YOLOv1 sangat revolusioner karena mampu mendeteksi objek secara *real-time* dengan kecepatan 45 *frame* per detik. YOLOv1 menggunakan satu *layer convolutional neural network* (CNN) untuk memprediksi kelas objek dengan menghasilkan *bounding box* secara bersamaan. Namun, performanya kurang baik dalam mendeteksi objek kecil dan objek yang saling tumpang tindih.

2.6.1.2 YOLO Versi 2

YOLOv2 dirilis pada tahun 2016 dan merupakan pembaruan besar dari YOLOv1. Pada YOLOv2, jumlah layer CNN yang digunakan untuk deteksi objek ditingkatkan menjadi 19, dan model ini juga menggunakan teknik seperti *batch normalization* dan *anchor boxes* untuk meningkatkan akurasi deteksi objek. Selain itu, YOLOv2 mampu mendeteksi objek dengan ukuran yang lebih kecil dan lebih baik dalam menangani objek yang tumpang tindih.

2.6.1.3 YOLO Versi 3

YOLOv3 dirilis pada tahun 2018 dan memperkenalkan beberapa inovasi, seperti penggunaan *feature pyramid network* (FPN) dan teknik seperti *multi-scale training* dan *test-time augmentation*. Dengan demikian, YOLOv3 mampu mendeteksi objek dengan akurasi yang lebih cepat dan lebih tinggi dari YOLOv2. YOLOv3 memanfaatkan struktur Darknet-53 yang terdiri dari 53 lapisan *convolutional* sehingga v3 ini memiliki kompleksitas yang lebih dibandingkan dengan YOLOv2 dan mencakup koneksi pintasan (*shortcut connections*) yang terdiri dari filter 3x3 dan 1x1 (Jonnalagadda, 2019).

2.6.1.4 YOLO Versi 4

YOLOv4, yang dirilis pada 2020, membawa berbagai inovasi dengan menggunakan Darknet sebagai *backbone*, penggunaan data augmentation yang lebih canggih, serta penerapan teknik *bag of freebies* dan *bag of specials* untuk meningkatkan performa. Pengembangan ini menghasilkan peningkatan akurasi deteksi objek dan kecepatan yang lebih tinggi. diciptakan oleh Alexey Bochkovsky, algoritma YOLOv4 dirancang untuk deteksi objek yang cepat dan efisien dalam sistem produksi, dengan fokus pada komputasi paralel. Sistem deteksi YOLOv4 menggunakan CSPDarknet53 sebagai *backbone* dan *Path Aggregation Network*(PAN) serta *Spatial Pyramid Pooling* (SPP) *Layer* pada bagian *Neck*. Penyesuaian pada SAM (*Spatial Attention Module*) dan PAN, serta penggunaan *Cross mini-Batch Normalization* (CmBN), juga menjadi bagian dari peningkatan yang dilakukan pada algoritma YOLOv4.

Beberapa peningkatan dilakukan pada algoritma YOLOv4 untuk menciptakan detektor yang dirancang untuk dijalankan pada GPU tunggal, antara lain:

- Penggunaan metode baru pada augmentasi berupa *Mosaic* dan *Self-Adversial Training* (SAT)

- Pemilihan *hyperparameter* yang optimal diterapkan saat menerapkan algoritma *genetic*
- Modifikasi pada SAM dan PAN serta penggunaan *Cross mini-Batch Normalization* (CmBN)

2.6.1.5 YOLO Versi 5

Dikembangkan oleh *ultralytics*, terdiri dari tiga pilar utama, yaitu, tulang punggung untuk ekstraksi fitur, leher yang difokuskan pada agregasi fitur, dan kepala untuk mengonsumsi fitur keluaran dari leher sebagai input dan menghasilkan deteksi. YOLO-v5, mirip dengan YOLO-v4, sehubungan dengan kontribusi, fokus pada penggabungan dan penyempurnaan berbagai teknik visi komputer untuk meningkatkan kinerja. Arsitektur menggunakan pytorch yang memiliki kelebihan yaitu menyediakan ekosistem yang mapan dan luas antar visi komputer dan menyediakan infrastruktur pendukung untuk perangkat seluler.

2.6.1.6 YOLO Versi 6

Diekmbangkan oleh Meituan *Technical Team* dari Tiongkok. Peningkatan tambahan yang difokuskan pada aplikasi industri meliputi penggunaan distilasi pengetahuan, melibatkan model guru yang digunakan untuk melatih model siswa, di mana prediksi dari guru digunakan sebagai label lembut bersama dengan kebenaran yang ada untuk melatih siswa. Ini dilakukan tanpa meningkatkan biaya komputasi karena pada dasarnya tujuannya adalah melatih model yang lebih kecil (siswa) untuk mereplikasi kinerja tinggi dari model yang lebih besar (guru).

2.6.1.7 YOLO Versi 7

YOLOv7, memiliki beberapa perbaikan dibandingkan dengan versi sebelumnya. Salah satu perbaikan utamanya adalah penggunaan kotak *anchor*. Kotak *anchor* adalah sekelompok kotak yang sebelumnya telah ditentukan dengan rasio aspek yang berbeda yang digunakan untuk mendeteksi objek. YOLOv7 menggunakan sembilan kotak *anchor*, yang memungkinkannya mendeteksi rentang bentuk dan ukuran objek yang lebih luas dibandingkan dengan versi sebelumnya, sehingga membantu mengurangi jumlah *false positives*.

2.6.1.8 YOLO Versi 8

Dikembangkan pada Januari 2023 oleh *ultralytics* sebagai model mutakhir dan tercanggih SOTA (*State of the art*), YOLOv8 membangun kesuksesan versi sebelumnya, memperkenalkan fitur-fitur baru dan peningkatan untuk meningkatkan kinerja, fleksibilitas, dan efisiensi. YOLOv8 mendukung berbagai tugas vision AI, termasuk deteksi, segmentasi, estimasi pose, pelacakan, dan klasifikasi. Fleksibilitas ini memungkinkan pengguna untuk memanfaatkan kemampuan YOLOv8 di beragam aplikasi dan domain.

2.6.1.9 YOLO Versi 9

YOLOv9 memperkenalkan *Programmable Gradient Information* (PGI) dan arsitektur jaringan baru yang disebut *Generalized Efficient Layer Aggregation Network* (GELAN) untuk mengatasi masalah kehilangan data dalam jaringan pembelajaran mendalam selama ekstraksi dan transformasi fitur. PGI dirancang untuk menyimpan informasi masukan yang lengkap untuk penghitungan gradien yang andal, sehingga meningkatkan pembaruan bobot. GELAN menggunakan konvolusi konvensional dibandingkan konvolusi berdasarkan kedalaman untuk efisiensi parameter yang lebih baik.

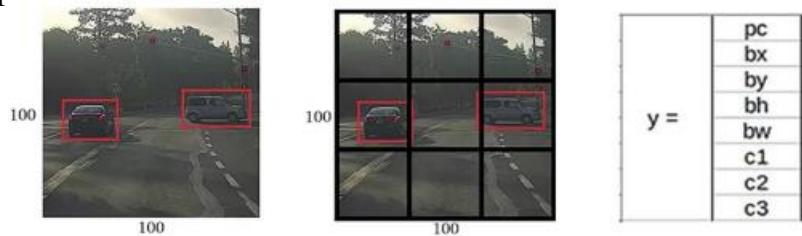
2.6.1.10 YOLO Versi 10

YOLOv10, dibangun pada paket *Ultralytics Python* oleh para peneliti di Universitas Tsinghua, memperkenalkan pendekatan baru untuk deteksi objek *real-time*, mengatasi kekurangan pasca-pemrosesan dan arsitektur model yang ditemukan pada versi YOLO sebelumnya. Dengan menghilangkan *non-maximum suppression* (NMS) dan mengoptimalkan berbagai komponen model, YOLOv10 mencapai kinerja tercanggih dengan pengurangan *overhead* komputasi secara signifikan. Eksperimen ekstensif menunjukkan trade-off akurasi-latensi yang unggul di berbagai skala model.

2.6.2 Proses Pembagian Citra dalam Grid

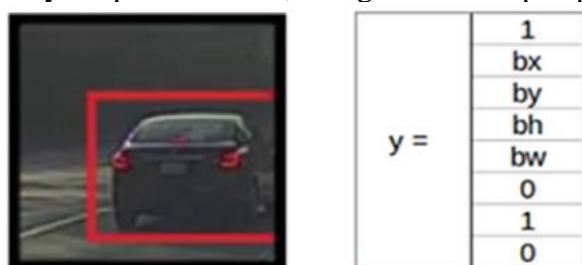
Masing-masing grid *image* terdiri dari *localization* serta *classification*. Terdapat lima *anchor box* dan satu atau lebih *class* dengan nilai (bx, by, bh, bw, c) pada tiap grid. YOLO membagi menjadi *grids* ukuran 3X3 dengan contoh tiga *class* atau objek yang

ingin diklasifikasikan lalu dilakukan perubahan pada setiap grid menjadi *vector* berlabel y dengan delapan dimensi.

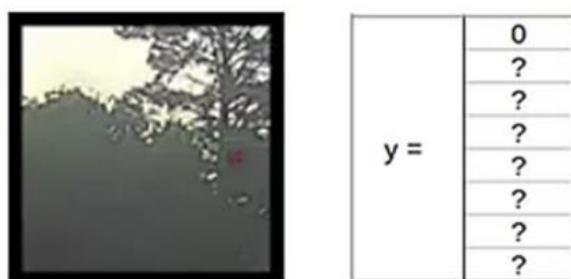


Gambar 2.6. Translasi citra menjadi vector (Reddy, 2021)

Proses ini mengidentifikasi keberadaan objek pada suatu citra dengan memprediksi *anchor box* atau kotak perkiraan bersama dengan probabilitasnya. Citra yang sedang diproses menghasilkan nilai-nilai untuk *anchor box*, seperti pc (probabilitas), bx, by, bh, dan bw (koordinat pembatas). Nilai pc didefinisikan sebagai satu jika objek terdeteksi dalam gambar dan nol jika tidak. Koordinat bx, by, bh, dan bw menentukan letak dan ukuran *anchor box* di dalam gambar. Selain itu, nilai c1, c2, dan c3 merepresentasikan kelas objek, di mana jika objeknya adalah mobil, maka nilai c2 akan menjadi satu, sedangkan c1 dan c3 akan nol, mengingat pada Gambar 2.6 tidak terdapat manusia dan motor. Proses ini melibatkan pembentukan vektor yang diulang-ulang, menghasilkan vektor seperti yang terlihat pada Gambar 2.7. Jika tidak ada objek yang terdeteksi pada satu grid tertentu, nilai vektor pc akan menjadi nol, dan nilai-nilai lainnya dapat diabaikan, sebagaimana tampak pada gambar 2.8.



Gambar 2.7 Objek dideteksi (Reddy, 2021).



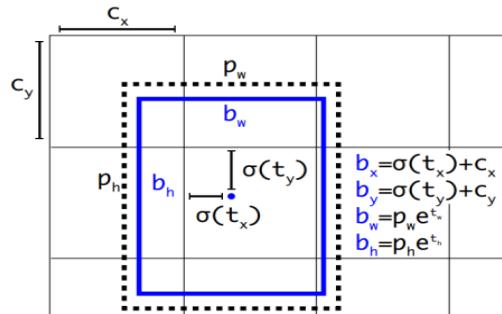
Gambar 2.8. Objek tak terdeteksi (Reddy, 2021).

Keseluruhan proses ini memiliki kesamaan seperti *Convolutional Layer* pada CNN, karena citra diinput menjadi 3×3 yang awalnya mempunyai 100×100 telah dibagi dengan total sembilan grid, masing-masing grid mempunyai satu *vector* yang memiliki delapan nilai dan menghasilkan output menjadi $3 \times 3 \times 8$.

2.6.3 Prediksi Bounding Box

Dalam upaya untuk memperkirakan *bounding box*, metode YOLO menghasilkan secara relatif terhadap lokasi grid prediksi koordinat titik tengah *bounding box*. Proses prediksi ini melibatkan penggunaan metode *k-means clustering* dengan nilai k yang ditetapkan sebanyak 5. Penetapan nilai k ini bertujuan untuk mencapai keseimbangan antara nilai kompleksitas model dan *recall*. Dalam perhitungan *Euclidean*, YOLO menerapkan IOU yaitu *Intersection Over Union* menggunakan Persamaan 2.6.

$$d(\text{box}, \text{centeroid}) = 1 - \text{IoU}(\text{box}, \text{centeroid}) \quad (2.6)$$



Gambar 2.9. Prediksi *Bounding Box* (Reddy, 2021)

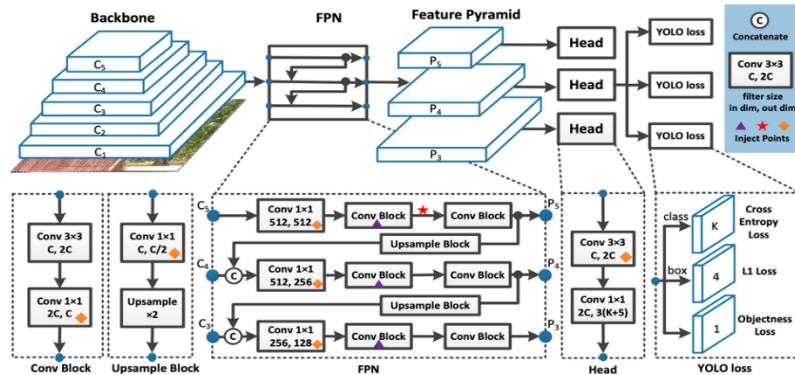
Gambar 2.9 menampilkan hasil prediksi *bounding box* yang berasal dari *anchor box* yang telah terbentuk. Untuk mendapatkan nilai b_x , digunakan nilai koordinat prediksi b_x (σt_x) yang dinormalisasi dengan panjang sel x (σc_x). Sebagai tambahan, nilai b_y dihitung dari koordinat prediksi b_y (σt_y) yang dinormalisasi dengan panjang sel y (σc_y). Sementara itu, nilai b_h dan b_w diperoleh dari tinggi dan lebar *bounding box*. Dalam menentukan area pada YOLO maka digunakan rumus berikut:

$$\text{Area} = (x \max - x \min) * (y \max - y \min) \quad (2.7)$$

2.7 Arsitektur YOLOv7

YOLOv7 berfokus pada contoh-contoh sulit yaitu objek-objek yang sulit dideteksi. YOLOv7 juga memiliki resolusi yang lebih tinggi daripada versi-versi sebelumnya. Ia memproses gambar pada resolusi 608×608 piksel, yang lebih tinggi daripada resolusi 416×416 yang digunakan dalam YOLOv3. Resolusi yang lebih tinggi ini memungkinkan YOLOv7 mendeteksi objek yang lebih kecil dan memiliki akurasi yang

lebih tinggi secara keseluruhan. Gambar 2.10 merupakan arsitektur umum dari YOLOv7 :

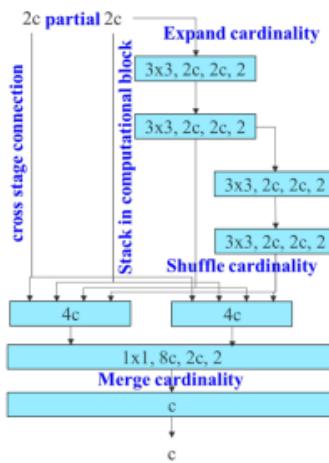


Gambar 2.10 Arsitektur YOLOv7 (roboflow.com)

2.7.1 Backbone

Pada YOLOv7, *backbone* sebagai ekstraksi fitur digunakan adalah E-ELAN (*Extended-Efficient Layer Aggregation Network*) dimana jaringan ini memiliki jalur gradien yang panjang dan penumpukan blok komputasi yang besar, ELAN mencapai keadaan stabil. E-ELAN menggunakan perluasan, pengacakan, dan penggabungan kardinalitas untuk

Pada meningkatkan kemampuan pembelajaran jaringan tanpa merusak jalur gradien asli (Wang et al., 2023). Gambar 2.11 merupakan arsitektur dari *backbone*.



Gambar 2.11 Arsitektur E-ELAN (Wang et al., 2023).

Dalam hal arsitektur, E-ELAN hanya mengubah struktur dalam blok komputasi, sementara arsitektur lapisan transisi tetap tidak berubah. Pendekatan yang digunakan adalah menggunakan konvolusi grup untuk memperluas saluran dan kardinalitas dalam

blok komputasi. Pendekatan yang sama diterapkan pada parameter grup dan pengalih saluran di semua blok komputasi dalam lapisan komputasi. Fitur peta yang dihasilkan oleh setiap blok komputasi diacak ke dalam grup yang sesuai dengan parameter grup yang ditentukan, dan kemudian digabungkan. Saat ini, jumlah saluran dalam setiap grup peta fitur tetap sama dengan jumlah saluran dalam arsitektur aslinya. Akhirnya, grup peta fitur ditambahkan untuk mencapai penggabungan kardinalitas. Dengan mempertahankan desain asli arsitektur ELAN, E-ELAN memungkinkan berbagai kelompok blok komputasi untuk mempelajari fitur yang lebih beragam (Wang et al., 2023).

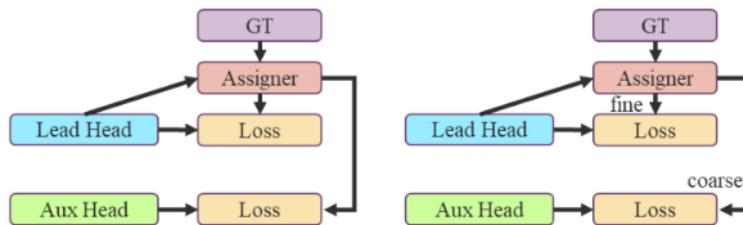
2.7.2 Neck

Pada bagian *neck*, keluaran peta fitur dari lapisan SPPCSPC (*Spatial Pyramid Pooling Cross Stage Partial Convolutional*) terdiri dari dua bagian. Di sini, peta fitur dibagi menjadi dua grup berdasarkan saluran. Grup pertama melakukan tiga operasi konvolusi dan kemudian tiga operasi maksimum *pooling* berturut-turut untuk memisahkan tiga set peta fitur dengan ukuran yang berbeda untuk mendapatkan beberapa fitur. Setelah menggabungkan beberapa fitur, dilakukan operasi konvolusi kedua untuk mendapatkan bagian pertama. Grup kedua melaksanakan operasi konvolusi untuk mendapatkan bagian kedua. Lapisan CUC (*Conv Upsampling Concat*) adalah unit dasar kombinasi peta fitur, termasuk konvolusi, *up-sampling*, dan penggabungan peta fitur. Lapisan REP (*Re-Parameterization*) adalah konsep baru yang menggunakan keterampilan reparameterisasi struktural untuk menyesuaikan struktur dalam inferensi untuk meningkatkan kinerja model. Dengan tiga bagian, lapisan REP dapat memperoleh keluaran peta fitur selama proses pelatihan. Bagian pertama dan kedua melaksanakan operasi konvolusi dan normalisasi *batch*. Bagian ketiga hanya melaksanakan normalisasi *batch*. Inferensi dari REP hanya mempertahankan bagian kedua dari struktur dengan menggunakan reparameterisasi struktural, mengurangi sumber daya komputasi, dan meningkatkan kinerja model.

2.7.3 Head

Pada arsitektur v7, terdapat beberapa *head* yang bertanggung jawab untuk melaksanakan tugas-tugas berbeda. Meskipun kepala utama dalam jaringan YOLO membuat prediksi akhir, menambahkan kepala tambahan pada lokasi tengah jaringan dapat membawa manfaat tertentu. Selama proses pelatihan, kedua kepala detektor ini dan kepala yang

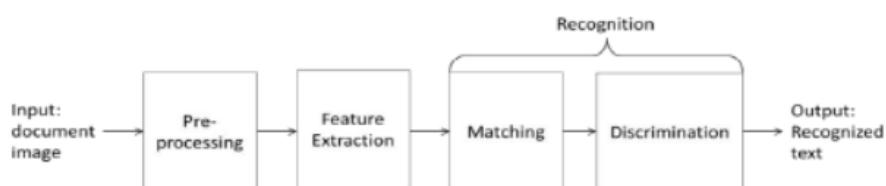
menghasilkan prediksi dipantau. Kepala tambahan ini tidak melalui pelatihan dengan tingkat ketat seperti kepala terakhir karena terdapat lebih sedikit lapisan jaringan yang memisahkan mereka dari tahap prediksi. Oleh karena itu, dalam YOLOv7, berbagai eksperimen dilakukan dengan tingkat pengawasan yang bervariasi untuk kepala tambahan ini. Tingkat pengawasan diatur dari yang kasar hingga halus, di mana informasi pengawasan dikirimkan kembali dari *head* yang memimpin dengan perbedaan tingkat garnulitas.



Gambar 2.12 Head YOLOv7 (Wang et al., 2023).

2.8 Optical Character Recognition (OCR)

OCR ialah teknologi yang berfokus pada proses membaca teks dari dokumen fisik dan mentranslasikan gambar teks menjadi bentuk yang dapat diolah oleh komputer, seperti kode ASCII sebagai contohnya. Sistem OCR memungkinkan konversi teks ke dalam format file komputer yang dapat diedit menggunakan perangkat lunak pengolah kata. Penerapan OCR juga mencakup analisis elektronik terhadap citra untuk mengidentifikasi area yang mengandung informasi tekstual, serta mengekstrak atau mengenali teks dari citra yang diberikan (Jain et al., 2013). Gambar 2.13 merupakan tahapan OCR secara umum.



Gambar 2.13 Tahapan Sistem OCR (Jain et al., 2013).

Tahapan ini dari *preprocessing* dan ekstraksi fitur yang sebelumnya telah dijelaskan, lalu masuk ke dalam tahap pengenalan (*Matching and Discrimination*), citra karakter diubah menjadi kode karakter yang sesuai dengan mencocokkan fitur-fitur karakter yang telah diekstraksi dengan fitur-fitur dari template karakter. Output dari algoritma pengenalan dapat beragam untuk satu citra dan memberikan analisis probabilitas yang setara. Contohnya, pengenalan citra karakter “I” dapat menghasilkan

“1”, “|”, “/”, “”, dan mencatat probabilitas untuk tiap-tiap kemungkinan (Jain et al., 2013).

2.9 Parallel Distributed Deep Learning (Paddle)

Paddle OCR atau disebut juga PP-OCR adalah platform *open-source* yang dikembangkan oleh Baidu *Research* yang dirancang agar mudah digunakan dan efisien untuk tugas pembelajaran mesin skala besar, mencakup serangkaian model dan alat yang telah dilatih sebelumnya untuk mengenali teks dalam gambar dan dokumen serta melatih model OCR khusus. *Paddle OCR* dibagi menjadi dua bagian utama:

Model pengenalan dari PP-OCR berdasarkan algoritma pengenalan teks *Single Visual Text Recognition* (SVTR) (Du et al., 2020). SVTR tidak lagi melibatkan RNN (*Recurrent Neural Network*) dengan memperkenalkan struktur *transformer*, yang dapat mengekstraksi informasi konteks dari gambar garis teks lebih efektif.

Setelah menerapkan proses deteksi teks dan pengenalan teks, kemudian menggabungkannya untuk mencapai alur berikut:

- Deteksi wilayah yang diperlukan dari gambar
- Berikan wilayah yang terdeteksi itu ke library OCR
- Simpan hasil dari OCR dalam format yang diinginkan, contohnya CSV



Gambar 2.14 Kombinasi YOLO dan OCR (Purohit, 2019).

Dari Gambar 2.14, dapat dipahami bahwa kartu identitas dapat melewati YOLO. Kemudian YOLO mendeteksi bidang teks yang diperlukan dan memotongnya dari gambar yang melewati daerah-daerah itu satu persatu dengan *library* OCR yang dapat membaca dan menyimpan informasi dari citra kartu identitas tersebut (Purohit, 2019).

2.10 Open Neural Network Exchange (ONNX)

ONNX merupakan standar terbuka untuk mewakili model pembelajaran mesin.

Dikembangkan oleh Microsoft bekerja sama dengan pemimpin industri lainnya, ONNX bertujuan untuk mengatasi masalah interoperabilitas model. Hal ini memungkinkan pengguna untuk mengonversi model yang dilatih dalam satu kerangka kerja ke dalam format yang dapat digunakan dengan kerangka kerja lain, sehingga lebih mudah untuk beralih di antara berbagai pustaka dan alat pembelajaran mendalam. ONNX berfungsi sebagai jembatan antara berbagai kerangka pembelajaran mendalam, mengurangi risiko vendor *lock-in* dan mendorong inovasi dengan memungkinkan berbagi model secara lancer di seluruh komunitas (Gomedede, 2023).

Peranan ONNX *Runtime* dalam ekosistem kecerdasan buatan (AI) sangat signifikan seperti:

- Portabilitas Model: ONNX dan ONNX *Runtime* bekerja sama untuk menyederhanakan proses pembuatan, pelatihan, dan implementasi model *machine learning* tanpa harus terikat pada kerangka kerja khusus. Ini mendorong kolaborasi dan inovasi dalam komunitas AI dengan mengurangi hambatan untuk berbagi model dan pengetahuan.
- Penerapan Lintas Platform: ONNX *Runtime* memungkinkan pengembang untuk menargetkan berbagai platform dengan satu model, menghasilkan efisiensi biaya dan kemudahan dalam menerapkan AI di berbagai perangkat dan lingkungan.
- Performa dan Efisiensi: Dengan optimalisasi dan kemampuan akselerasi perangkat keras, ONNX *Runtime* memastikan bahwa model AI dapat diimplementasikan dengan performa dan efisiensi tinggi. Aplikasi *real-time* sangat cocok dimanfaatkan dengan teknologi ini seperti kendaraan otonom, perangkat *Internet of Things* (IoT), dan komputasi edge.
- Dukungan Komunitas: ONNX *Runtime* mendapatkan dukungan yang besar dari komunitas yang aktif, yang berkontribusi pada pengembangan dan peningkatan berkelanjutan. Ini sejalan dengan kebutuhan AI yang terus berkembang, memastikan adaptasi terhadap perkembangan teknologi.

2.11 React Native

Framework ini yang dikembangkan oleh *Meta/Facebook* yang bersifat *open-source*. *JavaScript* merupakan bahasa pemrograman *framework* ini dalam membangun *mobile application*. Dengan menggunakan *framework* ini, pengembang web dapat juga mengembangkan aplikasi *mobile* yang menyerupai aplikasi *native*. Keunggulannya terletak pada kemampuannya untuk menggunakan sebagian besar kode yang sama pada berbagai platform, menjadikannya multiplatform. *Framework* ini mendukung pengembangan aplikasi secara bersamaan lintas *platform* (Eisenman, 2017).

Kombinasi *JavaScript* dan *markup* yang mirip *XML* yang disebut *JSX* digunakan dalam *framework* ini. Melalui “*bridge*” *React Native*, API asli yang terdapat dalam iOS (*Objective-C*) atau untuk Android (*Java*) dapat diakses. Dengan demikian, aplikasi yang dikembangkan akan menggunakan komponen UI asli, bukan sebagai tampilan web, memberikan pengalaman pengguna serupa dengan aplikasi mobile lainnya. *React Native* juga menyediakan antarmuka *JavaScript* untuk mengakses fitur platform seperti lokasi pengguna atau kamera. *Framework* mendukung pengembangan aplikasi *mobile* untuk Android dan iOS (Eisenman, 2017).

2.12 REST API

REST (*Representational State Transfer*) merupakan arsitektur perangkat lunak yang digunakan dalam sistem layanan web yang diperantara oleh REST API. REST API bertindak sebagai perantara untuk komunikasi antara aplikasi dan layanan web menggunakan *protocol* HTTP, seperti perintah *GET*, *PUT*, *POST*, dan *DELETE* untuk mengakses dan memanipulasi data. Layanan web adalah server web yang diciptakan khusus untuk melayani kebutuhan situs web atau aplikasi lain dengan cara menerima permintaan dan memberikan respons melalui API, memungkinkan pertukaran informasi antara program komputer.

Teknologi REST populer karena mengonsumsi lebih sedikit *bandwidth*, menjadikannya pilihan yang efisien untuk komunikasi di lingkungan internet. *Application Program Interface* (API) adalah kode yang menghubungkan dan memungkinkan komunikasi antara dua perangkat lunak. REST adalah pilihan yang sangat sesuai untuk digunakan dalam API untuk berkomunikasi dengan layanan *cloud*.

Banyak situs besar seperti Amazon, Google, LinkedIn, dan Twitter telah mengadopsi penggunaan API RESTful untuk mendukung layanan mereka.

2.13 Firebase

Firebase merupakan *framework* yang dikembangkan Andrew Lee dan James Tamplin dengan Perusahaan Envolve pada tahun 2011. *Framework* ini menyediakan berbagai produk dan layanan, termasuk produk awal seperti *Firebase Realtime Database*.(Guntoro, 2019).

Firebase menawarkan keunggulan fiturnya melibatkan otentikasi (*Authentication*), basis data, penyimpanan (*Storage*), hosting, dan analitika. *Firebase Cloud Storage* adalah layanan *Firebase* yang dirancang untuk penyimpanan digital dan konten yang dihasilkan pengguna, seperti gambar, video, dokumen, dan lainnya. Layanan ini menonjol karena keandalannya, kesederhanaan, efisiensinya, dan keamanan yang tinggi. *Firebase Cloud Storage* memungkinkan unggahan dan pengunduhan file secara langsung melalui SDK *Firebase*. Jika terjadi gangguan koneksi internet, layanan ini memungkinkan pengguna untuk melanjutkan proses dari titik terakhir di mana koneksi terputus, sehingga menghemat bandwith dan waktu *users*.

2.14 Confusion Matrix

Confusion matrix merupakan teknik untuk mengukur atau menghitung tingkat performa atau akurasi suatu metode klasifikasi pembelajaran mesin. *Confusion matrix* dapat melakukan analisis kerja pengklasifikasi saat mengidentifikasi setiap kelas yang dilatih. Gambar 2.15 menunjukkan *Confusion matrix* memiliki dua kelas objek, *true class* dan *predicted class*.

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Gambar 2.15. *Confusion Matrix Dua Class* (towardsdatascience.com)

Confusion matrix memiliki 4 komponen antara lain :

- a) *True Positive (TP)*

True Positive merupakan hasil pengenalan benar karena model pengenalan memprediksi hasil positif dan nilai aktualnya juga positif.

- b) *False Positive (FP)*

False Positive merupakan hasil pengenalan pola positif, namun memiliki nilai sebenarnya.

- c) *False Negative (FN)*

False Negative merupakan hasil pengenalan kelas model dengan nilai negative tetapi memiliki *actual value* positif sehingga tidak dapat mendeteksi adanya objek

- d) *True Negave (TN)*

True Negative merupakan semua kemungkinan *bounding box* objek yang benar tidak terdeteksi. *Matrix* ini tidak digunakan untuk pendekripsi objek karena terdapat banyak kemungkinan objek yang seharusnya tidak terdeteksi pada citra.

Ada beberapa *performance matrix* umum yang dapat diukur diantaranya :

2.14.1 Akurasi

Akurasi merupakan perbandingan jumlah data benar dengan jumlah keseluruhan dataset penelitian. Perhitungan akurasi dapat dilihat pada persamaan 2.8

$$\text{Akurasi} = \frac{\text{Jumlah data yang benar}}{\text{jumlah keseluruhan data}} \times 100\% \quad (2.8)$$

2.14.2. Precision

Precision adalah perbandingan antara hasil prediksi yang benar positif dan jumlah keseluruhan hasil prediksi positif, termasuk yang benar positif dan yang salah positif, sehingga mengenali objek yang relevan. Persamaan untuk *precision* dapat ditemukan dalam persamaan 2.9.

$$\text{Precision} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalsePositive}} \quad (2.9)$$

2.14.3. Recall

Recall adalah kemampuan model untuk mengukur seberapa baik model dalam menemukan semua objek positif sehingga menemukan semua kasus yang relevan. Secara khusus, *recall* adalah perbandingan antara jumlah hasil prediksi benar positif

dan total objek positif yang sebenarnya, termasuk yang benar positif dan yang salah negatif. Persamaan untuk *recall* dapat ditemukan dalam persamaan 2.10.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (2.10)$$

2.14.4. F1-Score

F1-Score adalah rasio nilai rata – rata antara *recall* dan *precision*. Oleh karena itu matrik ini digunakan sebagai cara standar untuk mengkombinasikan *precision* dan *recall* menjadi satu ukuran performa. Persamaan F1-Score dapat dilihat pada persamaan 2.11 .

$$F1 - score = 2x \frac{precision \times recall}{precision + recall} \quad (2.11)$$

2.15 Penelitian Terdahulu

Adapun penelitian terdahulu diperlukan dengan tujuan membantu, memperkuat serta mendukung proses penelitian ini. Pertama yaitu penelitian oleh Nguyen dkk (2018). Menggunakan KTP di negara Vietnam sebagai objek untuk diekstraksi data informasinya. Data sebanyak 22 citra kartu yang kualitas rendah dan 73 kualitas medium dengan metode *Convolutional Recurrent Neural Network* (CRNN) mendapatkan hasil akurasi sebesar 95,28% (Nguyen et al., 2018).

Penelitian oleh Syahputra pada tahun 2019 juga melakukan ekstraksi data dari e-KTP yang kemudian data tersebut disimpan menjadi pencatatan daftar rekrutmen anggota partai politik. Kombinasi antara metode OCR dan LSTM dinilai sangat baik untuk melakukan pemindaian data secara akurat pada tiap-tiap variable data sehingga menghasilkan akurasi sebesar 91,42 (Syahputra, 2019).

Ada juga penelitian yang dilakukan oleh Aprilino & Amin pada tahun 2022. Penelitian ini memanfaatkan metode YOLO, menghasilkan sistem dengan deteksi objek yang sangat cepat, serta menggunakan library Tesseract menunjukkan akurasi 100%, dan *threshold* sebesar 0,5. Sementara itu, penggunaan *library* Tesseract mencapai akurasi sebesar 92,32% (Aprilino, 2022).

Kemudian ada penelitian dengan judul oleh Kim dkk yang Mendeteksi tanggal kadaluarsa pada produk yang kemudian memberikan notifikasi kepada para penyandang tunanetra sehingga mengetahui kapan produk mereka kadaluarsa. Dengan

metode OCR dan YOLO, sistem mendapatkan akurasi sebesar 86% dengan waktu yaitu 22,20 detik (Kim et al., 2021).

Adapun penelitian kelima oleh Do dkk pada tahun 2023 yang melakukan penelitian terhadap pendekripsi atribut data dari kartu tanda penduduk Vietnam dengan model YOLOv7 serta pengekstraksi teks dengan VietOCR. Adapun akurasi yang diperoleh sistem secara keseluruhan adalah sebesar 97,98% (Do et al., 2023).

Penelitian terakhir dari Zhang dan Li pada tahun 2023 menerapkan teknologi *Paddle* OCR yang menjadi referensi penulis dalam menyusun penelitian ini. Adapun penelitian mereka ini mengusulkan metode pengenalan kode kesehatan yakni untuk pengenalan teks dalam gambar menggunakan algoritma lokasi target untuk menghasilkan teks yang dibutuhkan dan menerapkan hasil penelitian kedalam sistem pencegahan dan pengendalian epidemi di lingkungan kampus di Tiongkok. Hasil akurasi dari *Paddle* OCR dalam merekognisi kode adalah sebesar 97,34% (Li & Zhang, 2023).

Adapun tabel 2.1 merupakan penelitian terdahulu yang menjadi referensi penguat penulis dalam mengangkat judul ini.

Tabel 2.1. Penelitian Terdahulu

No.	Penulis	Judul	Keterangan
1.	Thanh Cong Nguyen, Dinh Tuan Nguyen & Quoc Long Tran (2018)	<i>Information Extraction From ID Card Via Computer Vision Techniques</i>	Membangun sistem ekstraksi informasi dari kartu identitas di Vietnam dengan data sebanyak 22 citra kartu yang low quality dan 73 medium quality dimana mendapatkan akurasi sebesar 95,28%.
2.	Andika Syahputra (2019)	Pendeteksian Data E-KTP Untuk Pencatatan Rekrutmen Anggota Partai Politik Menggunakan Algoritma <i>Long Short Term Memory</i> (LSTM) Berbasis Android	Sistem pemindaian citra e-KTP secara <i>real-time</i> untuk pencatatan peserta rekrutmen anggota partai politik, dengan data sebanyak 100 buah citra , mengkombinasikan LSTM dengan <i>library</i> TesseractOCR yang diimplementasikan ke perangkat <i>mobile</i> menghasilkan akurasi sebesar 91,42%

3.	Awan Aprilino & Imam Husni Al Amin (2022)	Implementasi Algoritma YOLO dan Tesseract OCR Pada Sistem Deteksi Plat Nomor Otomatis	Deteksi otomatis plat kendaraan, menggunakan 700 data dengan YOLO V3, mencapai hasil pengujian dengan akurasi sempurna 100%, terutama dalam kondisi pencahayaan yang memadai dan threshold settingi 0,5. Penggunaan library Tesseract juga memberikan hasil yang baik, dengan sistem mampu mengenali seluruh karakter pada plat kendaraan, mencapai akurasi sebesar 92,32%.
4.	Min Soo Kim, Mi Kyung Moon & Chang Hee Han (2021)	<i>Expiration Date Notification System Based on YOLO and OCR algorithms for Visually Impaired Person</i>	Mendeteksi tanggal kadaluarsa pada produk yang kemudian memberikan notifikasi kepada para penyandang tunanetra sehingga mengetahui kapan produk mereka kadaluarsa. Dengan metode OCR dan YOLO, sistem mendapatkan akurasi sebesar 86% dengan waktu yaitu 22,20 sekon
5.	Cung Duy Do, Duan Quang Vo & Hoang Tai Toan Vu (2023)	<i>Extraction Information from Vietnamese ID Card Images</i>	Mengekstraksi data dari kartu tanda penduduk Vietnam dengan menggunakan YOLO v7 & VietOCR sebagai text recognition. Dengan 644 data, dimana tiap citra memiliki 16 atribut data. menghasilkan akurasi YOLO v7 97,60% serta akurasi ekstraksi teks sebesar 98,19%
6.	Dan Zhang & Yunjie Li (2023)	<i>Research and Application of Health Code Recognition Based on Paddle OCR under the Background Epidemic Prevention and Control</i>	Penelitian mengusulkan metode pengenalan kode kesehatan, dengan menggunakan teknologi <i>Paddle OCR</i> untuk pengenalan teks dalam gambar, menggunakan algoritma lokasi teks target untuk menghasilkan teks yang dibutuhkan, dan menerapkan hasil penelitian dalam sistem pencegahan dan pengendalian epidemi di lingkungan kampus, untuk membantu komisioner pencegahan epidemi dalam meninjau secara massal kode kesehatan/kartu perjalanan. Adapun hasil akurasi metode <i>Paddle OCR</i> adalah sebesar 97,34%

2.15.1 Perbedaan Penelitian

Perbedaan antara penelitian ini dengan penelitian terdahulu seperti dengan penelitian yang dilakukan oleh Nguyen dkk (2018) yaitu dalam pendeksiannya mereka menggunakan algoritma CRNN (*Convolutional Recurrent Neural Network*) serta

untuk objeknya yaitu citra KTP dari negara Vietnam. Adapun sistem implementasinya berbasis web sedangkan penulis berbasis mobile yaitu android (Nguyen et al., 2018).

Perbedaan antara penelitian penulis dan penelitian Syahputra pada tahun 2019 terletak pada metode pendektsian kartu e-KTP. Syahputra menggunakan algoritma LSTM dengan dataset sebanyak 100 citra e-KTP, mencapai akurasi sebesar 91,42 (Syahputra, 2019).

Terdapat pula perbedaan penelitian penulis dengan penilitian oleh Aprilino dan Amin pada tahun 2022. Pada penelitian tersebut mereka menggunakan metode YOLO v3 sebagai pendektsian. Adapun objek yang dideteksi dan di ekstraksi yaitu citra plat kendaraan mobil (Aprilino, 2022)

Perbedaan objek penelitian juga terdapat pada penelitian Kim dkk tahun 2021, yaitu mereka pendektsian terhadap karakter teks tanggal kadaluarsa pada suatu bungkus produk untuk menjadi notifikasi di *smartphone* sebagai pengingat untuk para pengguna penyandang tuna netra. (Kim et al., 2021)

Penelitian yang dilakukan Do dkk memiliki kesamaan metode yaitu YOLOv7 sebagai referensi saya serta sama-sama menggunakan objek kartu tanda penduduk, namun dalam penelitian mereka, objeknya yaitu kartu tanda penduduk negara Vietnam serta menggunakan library VietOCR sebagai ekstraksi teks nya. (Do et al., 2023).

Adapun perbedaan dengan penelitian terdahulu terakhir oleh Zhang dan Lee pada tahun 2022 yaitu objek dari penelitiannya yang berbeda. Adapun objek penelitian mereka yaitu kode kesehatan di lingkungan rumah sakit sebagai pengendalian epidemi di Tiongkok (Li & Zhang, 2023).

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 Data

Penelitian ini menggunakan data sebanyak 800 citra kartu e-KTP. Citra-citra tersebut diperoleh melalui pengumpulan langsung sebanyak 200 citra, pengumpulan tidak langsung sebanyak 200 citra, dan dataset pada *Roboflow* sebanyak 400 citra. Setiap e-KTP mencakup 19 atribut, antara lain:

- | | |
|------------------------|-------------------------|
| - NIK | - Kecamatan |
| - Nama | - Agama |
| - Tempat Tanggal Lahir | - Status Perkawinan |
| - Jenis Kelamin | - Pekerjaan |
| - Golongan Darah | - Kewarganegaraan |
| - Alamat | - Berlaku Hingga |
| - Provinsi | - Foto |
| - Kabupaten/Kota | - Tanggal Pembuatan KTP |
| - RT/RW | - Tanda Tangan |
| - Kelurahan/Desa | |

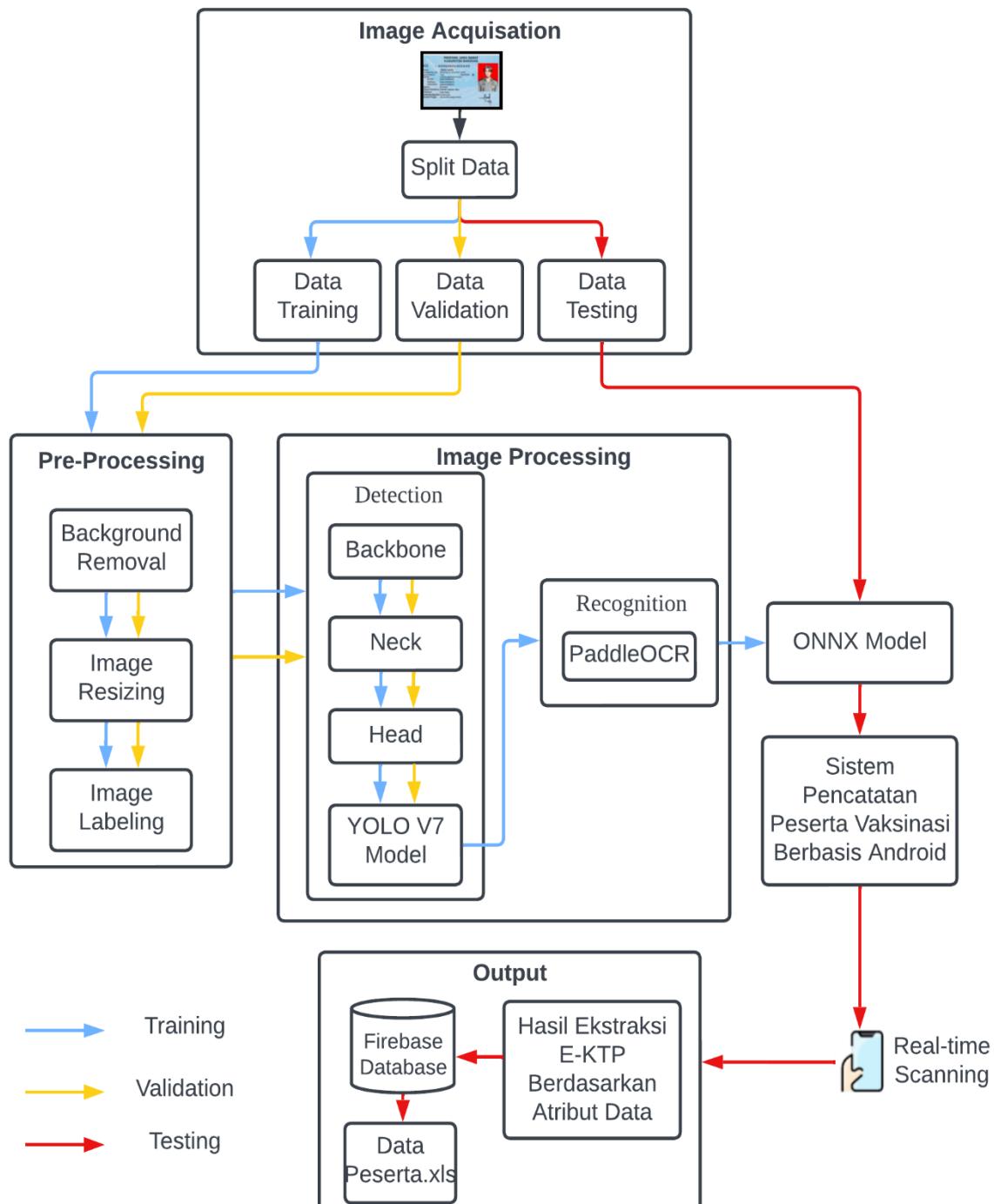
Tabel 3.1 merupakan jumlah dataset yang digunakan berdasarkan jumlah data yang dikumpulkan dan atribut yang digunakan.

Tabel 3.1 Jumlah Dataset Penelitian

Jenis Kartu	Jumlah	Atribut	Jumlah Data
e-KTP	800	19	15200

3.2 Arsitektur Umum

Beberapa langkah dalam membagi sistem pada penelitian ini dapat dilihat pada arsitektur umum pada gambar 3.1.



Gambar 3.1. Arsitektur Umum

3.3 Image Acquisition

Data citra e-KTP dikumpulkan pada tahapan ini yang selanjutnya menjadi dataset penelitian. Proses pengambilan data diambil langsung oleh peneliti maupun secara tidak langsung dengan menggunakan kamera *smartphone* beresolusi minimal 12MP oleh pemilik e-KTP serta dari dataset *Roboflow*. Gambar 3.2. merupakan data Citra yang terkumpul dengan berekstensi .jpg.



Gambar 3.2. E-KTP yang Terkumpul

Pembagian Dataset menjadi tiga bagian yaitu data *training*, *validation* dan *testing*. Tabel 3.2 menjelaskan pembagian dataset tersebut.

Tabel 3.2 Pembagian Dataset

Dataset	Training	Validation	Testing
Persentase (%)	60%	20%	20%
Jumlah citra	480	160	160
Jumlah atribut data	9120	3040	3040

Data *training* digunakan sebagai dataset pelatihan model, lalu *validation* digunakan untuk memvalidasi model hasil latih dari data latih. Adapun penggunaan data *testing* sebagai pengujian model yang telah dihasilkan.

3.4 Image Pre-Processing

Tahap ini bertujuan untuk meningkatkan kualitas citra e-KTP agar dapat diolah lebih baik pada tahap berikutnya. Berikut adalah penjelasan mengenai langkah-langkah dalam proses ini.

3.4.1 Background Removal

Setelah citra dalam format *grayscale*, maka dilakukan penghapusan terhadap latar belakang citra yang tidak diperlukan dalam proses *processing*. Ini adalah langkah penting karena dapat membantu meningkatkan kualitas data yang akan digunakan dalam deteksi teks. Hal ini dapat menggunakan segmentasi untuk menghilangkan latar belakang dan hanya menyisakan kartu KTP serta isinya. Adapun berikut ini merupakan *pseudocode* untuk proses *background removal* citra e-KTP:

```

import library
Load path ktp img
Convert img to hsv color space
Define lower and upper bounds for white color
Create mask for the bg color
Invert mask to obtain the object
Combine mask with ori ktp img to remove bg
Save the result

```



Gambar 3.3 Background Removal e-KTP

3.4.3 Image Resizing

Mengubah ukuran citra merupakan hal yang penting dilakukan pada tahap *preprocessing*. Dengan melakukan *resizing* maka dapat membuat model melakukan proses *training* lebih cepat. Keadaan dimana kita tidak bisa mengatur ukuran data yang kita gunakan oleh karena itu *resizing* diperlukan untuk menyamakan ukuran dari citra yang kita gunakan sebagai *input*. Berikut merupakan *pseudocode* dalam tahapan *resizing*:

```

Set image file path
Read image file and return array with pixel value
Set desired image width
Set desired image height
Get dimension of original image
Set empty array resized (new_h, new_w, c) with all value 0
If desired image width not equal 0:

```

```

Calculate scale factor width = old width/new width
Else:
  Set scale factor width ← 0
  If desired image height not equal 0:
    Calculate scale factor height = old height/new height
  Else:
    Set scale factor height ← 0
  For i in new height:
    For j in new width:
      Set x coordinate ← i * height scale factor
      Set y coordinate ← j * width scale factor
      Calculate coordinate values for 4 surrounding pixels
      Calculate 4 neighboring pixels values using coordinate
      values
      Estimate the pixel value q using neighbouring pixel values
      Set new image array resized at coordinates (i,j) with pixel
      value q

```

Adapun gambar 3.4 merupakan citra Sebelum dan sesudah *resizing*.



Gambar 3.4 Resizing 323x204 piksel

3.4.4 Image Labeling

Selanjutnya dilakukan proses *labeling* pada masing-masing posisi data secara manual melalui aplikasi LabelIMG. Terdapat 19 kelas yang di label sesuai keseluruhan data yang termuat pada citra e-KTP. Dikarenakan proses *processing* nantinya menggunakan YOLOv7, harus diperhatikan bahwa *labelling* harus diatur untuk *support training* YOLO seperti pada gambar 3.5 yang nantinya menghasilkan format labeling berkekstensi txt.



Gambar 3.5 Proses di Labelimg

Output dari tahap pelabelan citra akan membentuk file anotasi citra dengan ekstensi txt. Seperti pada gambar 3.6. File ini akan mendukung proses pelatihan YOLOv7, berisis koordinat dari masing-masing atribut kelas data .

```

0 0.388994 0.140379 0.667932 0.072555
1 0.285579 0.222397 0.476281 0.053628
2 0.323529 0.255521 0.571157 0.063091
3 0.240038 0.299685 0.396584 0.050473
4 0.601518 0.299685 0.174573 0.044164
5 0.299810 0.345426 0.535104 0.053628
6 0.282732 0.422713 0.398482 0.037855
7 0.348197 0.473186 0.552182 0.050473
8 0.219165 0.515773 0.430740 0.047319
9 0.314991 0.616719 0.626186 0.059937
10 0.544592 0.034700 0.447818 0.063091
11 0.535104 0.085174 0.337761 0.056782
12 0.252372 0.388013 0.314991 0.031546
13 0.252372 0.566246 0.500949 0.053628
14 0.192600 0.670347 0.381404 0.041009
15 0.266603 0.722397 0.529412 0.056782
16 0.919355 0.668770 0.161290 0.075710
17 0.872865 0.358044 0.254269 0.463722
18 0.920304 0.793375 0.151803 0.217666

```

Gambar 3.6 Hasil labeling

3.5 YOLO V7

Setelah dataset melalui proses *labelling* maka penelitian ini dilanjutkan ke proses pembuatan model untuk mendeteksi posisi data dari e-KTP dengan *Colaboratory*. Menggunakan *Free GPU colab* sehingga menghasilkan model YOLOv7 dengan hasil yang baik.

3.5.1 Clone Repository

Pada tahap ini, dilakukan kloning dari github *repository* YOLOv7. Selain itu juga diperlukan instalasi terhadap *requirement* yang diperlukan untuk melakukan proses *training*. Gambar 3.7 menampilkan cara *clone repo* dari *github* dan *install requirement*-nya



```
In [1]: # Download YOLOv7 repository and install requirements
!git clone https://github.com/WongKinYiu/yolov7
%cd yolov7
!pip install -r requirements.txt
```

Gambar 3.7 *Clone Repository* YOLOv7

Penelitian ini menggunakan YOLOv7 sebagai metode pendekripsi terhadap posisi data pada citra e-KTP. Sebelum melakukan *training* data, terlebih dahulu menyesuaikan parameter pelatihan yaitu tiap-tiap kategori data pada suatu file yaitu obj.names.

3.5.2 Konfigurasi Model

Membuat file custom_data.yaml pada folder data untuk menentukan letak data *training* dan validasi, menentukan jumlah kelas serta masing-masing nama kelasnya. Konfigurasi ini diperlukan untuk YOLOv7 agar model dapat melakukan *training* dan *validation* berdasarkan data *custom* kita.

```
# number of classes
nc: 19

# class names
names: [ 'nik', 'nama', 'ttl', 'jenis kelamin', 'goldar', 'alamat', 'kel/desa',
|       'kecamatan', 'agama', 'pekerjaan', 'provinsi', 'kab/kota', 'rt/rw',
|       'status', 'kewarganegaraan', 'berlaku', 'tglktp', 'foto', 'ttd' ]
```

Gambar 3.8 konfigurasi YOLOv7 untuk atribut e-KTP

Adapun membuat file baru dengan nama yolov7-ktp.yaml pada folder cfg/training sebagai konfigurasi. File tersebut berisi penentuan jumlah kelas serta proses *training* yang dimulai dari anchors hingga head.

3.5.3 Proses Training

Dari semua data yang telah terkumpul, sebanyak 480 data citra, akan digunakan dalam proses pelatihan. Data ini akan membantu untuk menentukan parameter seperti jumlah *epoch*, *batch size*, dan ukuran gambar yang digunakan dalam pelatihan model. Metode

YOLOv7 memiliki tiga komponen utama dalam arsitekturnya: *backbone*, *neck*, dan *head*.

Backbone dalam YOLOv7 berperan penting dalam mengekstraksi fitur-fitur dari citra yang dimasukkan. Ini melibatkan proses pencarian informasi penting dalam citra. Citra kemudian dibagi menjadi grid dengan ukuran tertentu ($s \times s$). Hasil dari proses *backbone* ini akan menghasilkan vektor yang berisi informasi seperti kelas objek, koordinat *bounding box*, tinggi, dan lebar dari *bounding box*.

YOLOv7 melakukan pencarian informasi secara berulang, yaitu setiap 3 kotak dari *grid*, untuk memahami citra dengan lebih baik. Setelah informasi dari *backbone* terkumpul, selanjutnya adalah proses ekstraksi informasi dalam tahap *neck*. Di tahap ini, berbagai kotak dengan tingkat keyakinan (*confidence level*) yang berbeda akan terbentuk. Biasanya, yang dipilih adalah kotak dengan tingkat keyakinan tertinggi.

Setelah itu, informasi ini akan diteruskan ke tahap *head*. Di dalam tahap *head*, komponen ini akan membentuk *bounding box*, menampilkan nilai *confidence level*, dan menentukan kelas apa yang telah terdeteksi dalam citra tersebut. Gambar 3.19 menunjukkan cara *training* model.

```
!!python train.py --device 0 --batch-size 8 --epochs 200 --img 640 640 --data data/custom_data.yaml --hyp data/hyp.scratch.custom.yaml --cfg cfg/training/yolov7-ktp.yaml
...
2023-10-03 04:01:03.316137: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optimized to use available CPU instructions in performance.
To enable the following instructions: AVX2 AVX512F FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
2023-10-03 04:01:04.531005: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not find TensorRT
YOLOR 🚀 v0.1-126-gb4932d7 torch 2.0.1+cu118 CUDA:0 (Tesla T4, 15101.8125MB)
```

Gambar 3.19 Training YOLOv7

3.5.4 YOLOv7 Model

Maka setelahnya proses *training* dilakukan didapatkan hasil berupa *learned model* dalam bentuk file *pytorch* karena proses *training* sebelumnya. Dalam prosesnya, YOLOv7 membentuk beberapa model dimana terbagi menjadi *last* dan *best model*. *Learned model* ini merupakan output hasil dari *training* deteksi posisi data oleh YOLOv7 yang nantinya akan di load pada tahap selanjutnya yaitu *testing*. Hasil training dapat dilihat pada folder *./runs/train/yolov7-ktp*. Dalam folder tersebut berisi beberapa hasil informasi *training* mulai dari grafik, data, hingga folder *weights* nya.

Setelah model YOLOv7 diperoleh maka model perlu diuji kembali dengan validasi data untuk melihat seberapa akurat model dalam mendeteksi posisi dan jenis informasi yang terdapat pada e-KTP berdasarkan parameter pelatihan sebelumnya.

Gambar 3.10 merupakan contoh hasil deteksi model YOLOv7 dengan *bounding boxes* nya.



Gambar 3.10 Bounding Boxes YOLOv7

3.5.5. Penghitungan *Confidence Score*

IoU (*Interest over Union*) adalah metrik yang digunakan dalam mengevaluasi kinerja YOLO. Adapun rumusnya tampak seperti persamaan 2.5. Sebelum itu harus diketahui terlebih dahulu nilai *intersection* dan *union* nya. Untuk mendapatkan nilai keduanya terlebih dahulu mencari nilai area *ground truth* dan *predicted box*. Rumus area adalah mengikuti persamaan 2.7 . Disini akan dicontohkan penghitungan untuk mencari IoU dan *confidence score* untuk deteksi pasfoto pada KTP(103).jpg seperti pada gambar 3.10.

Diketahui :

$$\text{Ground truth} : - \text{x_min} = 338 \quad - \text{x_max} = 461$$

$$- \text{y_min} = 57 \quad - \text{y_max} = 205$$

$$\text{Predicted box} : - \text{x_min} = 337 \quad - \text{x_max} = 460$$

$$- \text{y_min} = 54 \quad - \text{y_max} = 206$$

Nilai probabilitas : 0,93

$$\text{Area (Ground truth)} = (\text{x_max} - \text{x_min}) * (\text{y_max} - \text{y_min})$$

$$= (461 - 338) * (205 - 57)$$

$$= 123 * 148$$

$$= 18.204$$

$$\text{Area (Predicted box)} = (460 - 337) * (206 - 54)$$

$$= 123 * 152$$

$$= 18.696$$

$$\text{Intersection} = (460 - 338) * (206 - 57)$$

$$= 18.178$$

$$\text{Union} = \text{Area ground truth} + \text{Area predicted} - \text{intersection}$$

$$= 18.204 + 18.696 - 18.178$$

$$= 18.722$$

$$\text{IoU} = \text{Intersection} / \text{Union}$$

$$= 18.178 / 18.722$$

$$= 0,97$$

Maka nilai *confidence score* seperti pada persamaan 2.4

$$\text{Confidence score} = \text{Probabilitas kelas} * \text{IoU}$$

$$= 0,93 * 0,97$$

$$= 0,90$$

3.6 Paddle OCR

Melalui *library Paddle OCR*, data yang telah dideteksi sebelumnya akan dilakukan proses rekognisi dan ekstraksi menjadi *string* pada sistem melalui REST API. Berikut merupakan *pseudocode* dari proses ekstraksi hasil pengenalan karakter teks dari *Paddle OCR*:

```
Import modul Paddle OCR
Inisialisasi objek OCR
Load image e-KTP
Do text recognition with paddleocr
Show extracted text
```

Hasil ekstraksi *Paddle OCR* dapat dilihat pada gambar 3.11.



Gambar 3.11 Ekstraksi Paddle OCR

3.7 Open Neural Network Exchange (ONNX Model)

Hasil dari proses *training* merupakan *learned model* dari YOLOv7 yang berbentuk format *pytorch* kemudian dilakukan konversi menjadi ONNX (*Open Neural Network Exchange*) model. Hal ini bertujuan agar model dapat dijalankan bersamaan dengan library *Paddle OCR*. ONNX dirancang untuk memfasilitasi interoperabilitas antara beberapa kerangka kerja *machine learning*. Setelah di konversi, maka dilakukan integrasi dengan *Paddle OCR*.

3.7.1 Proses Testing

Selanjutnya akan dilakukan proses *testing*. Model diuji coba di perangkat *mobile smartphone*. Hal ini dilakukan sehingga dapat diketahui apakah sistem yang telah dilatih sebelumnya dapat melakukan pendekripsi terhadap data pada e-KTP, serta dapat melakukan rekognisi data dan dilakukan pula penghitungan akurasi yang didapatkan terhadap metode YOLOv7 dan *library Paddle OCR* sehingga dapat dilihat kinerja yang dihasilkan berdasarkan model ONNX yang dihasilkan. Adapun tahapan dari proses pengujian model YOLOv7 dapat dilihat pada *pseudocode* berikut ini :

```

From paddleocr import PaddleOCR
Load class from folder data customdata.yaml
Set device to cuda
Load model best.onnx
If cam equals enable
    Set cudnn.benchmark as true
Read class from folder data customdata.yaml
Set time
Set prediction with parameter model,
confidence_threshold, iou_threshold and classes
For detection per image from prediction
    If camera enabled: Set im0 ← path(path[i])
    Else: Set im0 ← path(path)

```

```

Make save path
Make txt path
Normalization gain whwh
If image length value not 0:
Rescale bounding boxes from img_size to im0 size
For item in det:
    Detect per class
    Convert result detect to string
    Localize result to its class

```

3.8 Output

Data yang sudah dilakukan pencatatan oleh sistem akan masuk dan disimpan oleh *Firebase Firestore Database* seperti pada gambar 3.12.

Field	Type	Value
agama	String	"ISLAM"
alamat	String	"GUNUNG LONCENG DESA LOBU HUALA"
createdAt	Date	"2023-11-07"
fileKtp	String	"https://firebasestorage.googleapis.com/v0/b/catavksin-70be4.appspot.com/o/fotoKTP%2Fktp1699375979421?alt=media&token=5d4125d6-bdef-44b1-9fcc-d38e909cf3e9"
goldar	String	"-
jenkel	String	"LAKI-LAKI"
kecamatan	String	"KUALUH SELATAN"
kelurahan	String	"LOBU HUALA"
nama	String	"ABDUL KARIM LUBIS"
nama_file	String	"fotoKTP/ktp1699375979421"
nik	String	"1122308081000003"
pekerjaan	String	"PELAJAR/MAHASISWA"
tempat_lahir	String	"GUNUNG LONCENG"

Gambar 3.12 Database *Firebase*

Adapun pada sistem ketika *user* hendak melakukan eksport data, maka akan dapat membagikan keseluruhan data yang telah dikelompokkan menjadi berdasarkan masing-masing atribut data peserta dalam bentuk format xls (Microsoft Excel) seperti pada gambar 3.13.

	B	C	D	E	F	G	H	I	J	K	L
1	nama	tempat_lahir	tgl_lahir	goldar_pekerjaan	jenkel	alamat	provinsi	kabupaten	kecamatan	kelurahan	
2	DICKY SOERYONO	SURABAYA	13-12-1984	1984/12/13	Gol.Dx WIRASWASTA	LAKI-LAKI	JL. P HDAYATULLAH CKALIMANTAN KOTA SAMARIN SAMARINDAROTA	PABELUHAN			
3	SYIFA SALSABILAH	JAKARTA	28-02-2001	2001/02/68	-	PELAJAR/MAHASI PEREMPU JL. RAWA SELATAN NC DKI JAKARTA	JAKARTA PUSA JOHAR BARU	KAMPUNG RA			
4	DEWI HANDAYANI	PRINGKUMPUL	1984/8/8	-	MENGURUS RUM LAKI-LAKI	PRINGSEWU SELATAN LAMPUNG	KABUPATEN PRINGSEWU	PRINGSEWU S			
5	MUHAMMAD SAMUDRA	KEBUMEN	07-02-2004	2004/02/07	O	PELAJAR/MAHASI:LAKI-LAKI- GG PLATUK 3	JAWA TENGAH-KABUPATEN KE KEBUMEN	KEBUMEN			
6	DEBBY ANGGRAINI	JAKARTA	20/01/00		KARYAWAN SWAS PEREMPU JL. KECAPI V	DKI JAKARTA	JAKARTA SELAT JAGAKARA	JAGAKARA			
7	DHARMA PUTRA	TALAGO	10-08-1970	1970/08/10	A	PEGAWAI NEGERI Jenem LAKI JL H MISKIN NO.90	SUMATERA BA KOTA BUKITTING MANDIANGIN KOT CAMPAGO IPU				
8	VIA MEI ANJANI	CILACAP	2005/05/0		BELUM/TIDAK BEP PEREMPU:DUSUN JETAK	JAWA TENGAH-KABUPATEN CI KARANGPUCUNG	SINDANGBARA				
9	SITI JUBADEAH	KUNINGAN	1997/03/2		B KARYAWAN SWAS PEREMPU IDUSUNWAGE	JAWA BARAT KABUPATEN KL MANDIRANCAN	MANDIRANCA				
10	AGUWIONO	OKU TIMUR	03-08-1984	1984/08/03	A GURU	LAKI-LAKI JL R. ENDANG SOEWA JAWA BARAT	KABUPATEN B/CIMENYAN	PADASUKA			
11	EKSAN DWI PRASETYO	MAGETAN	03-11-1993	1993/11/03	-	WIRASWASTA LAKI-LAKI-MELATI II	JAWA TIMUR KABUPATEN M BARAT	MANJUNG			
12	ROSDIANA	PALEMBANG	11-09-1987	1987/8/11	AB	MENGURUS RUM PEREMPU JL. SELAMET RIADY LR SUMATERA SE KOTA PALEMB/LUR TIMUR TIGA	SEPLULUH LUR				
13	ADI GUNO	hirPADANG CERMIN.	15-12-1986	1986/12/15	B	WIRASWASTA LAKI-LAKI SIMPANGBANTE DU SUMATERA UTAKUPATEN L'SELESAI		KeVDesPa	A		
14	JELLITA MAKAUDIS	MANADO	22-07-1997	1997/7/22	-	BELUM/TIDAK BELAKI-LAKI LINGKUNGAN IV	SULAWESI UTI KOTA MANADCINGKIL	KOMBOS TIMU			
15	SALSABILA ISNANDYA	BEKASI	07/11-2001	2001/11/7		PELAJAR/MAHASI PEREMPU PERUMOKTARAOEKA SJAWA BARAT	KABUPATEN BEK CIBARUH	SINDANGMUL			
16	ALTHAFIA DEFIYANDREA LASKAN BAEVES	05092004	2004/09/5		AB	PELAJAR/MAHASI PEREMPU PAGUYANGAN	JAWATENGAH KABUPATEN BF PAGUYANGAN	PAGUYANGAN			
17	ABDUL KARIM LUBIS	GUNUNG LONCENG	2000/9/8		-	PELAJAR/MAHASI LAKI-LAKI GUNUNG LONCENG D SUMATERA UTAKUPATEN LAKUALUH SELATAN	LOBU HUALA				
18	HENNY SOPIANI	GUNUNG PAMELA	23-08-1981	1971/08/63	O	PEGAWAI NEGERI PEREMPU JL. KOMP ASBARI NO. SUMATERA UT KOTA BINJAI	BINJAI SELATAN	PUJDADI			
19	GEDE BAGUS PERDANA PUTRA	DENPASAR	14-04-1991	1991/04/14	O	KARYAWAN SWAS LAKI-LAKIL ALASULTRIGGAXKOD BALI	KOTA DENPASAR	DENPASAR TIMUR PENATIH			
20	AIJUB WILLEW DARAWIA ST.MT	PALU	23-11-1953	1953/11/23	AB	ANGGGOTA DPRD kJens kelam BOLAPAPU	SULAWESI TEN KABUPATEN SI KULAWI	BOLAPAPU			
21	PAHRUL ARSANDI	LEMBUAK	31-12-1995	1995/12/31	-	WIRASWASTA Jenis Kelar LEMBUAK BARAT		NARMADA	LEMBUAK		
22	DIKKI SUHANDI	MULIOREJO	2001/02/12		B	PELAJAR/MAHASI LAKI-LAKI DUSUNXI JL. AMPERA	SUMATERA U KABUPATEN DI SUNGAL	MULIOREJO			
23	RINI INDRIVATI	KEBUMEN		2001/08/2	O	PELAJAR/MAHASI PEREMPU BLOK SUKAASHI NO.2	JAWA BARAT KOTA CIMahi CIMAHİ SELATAN	MELONG			

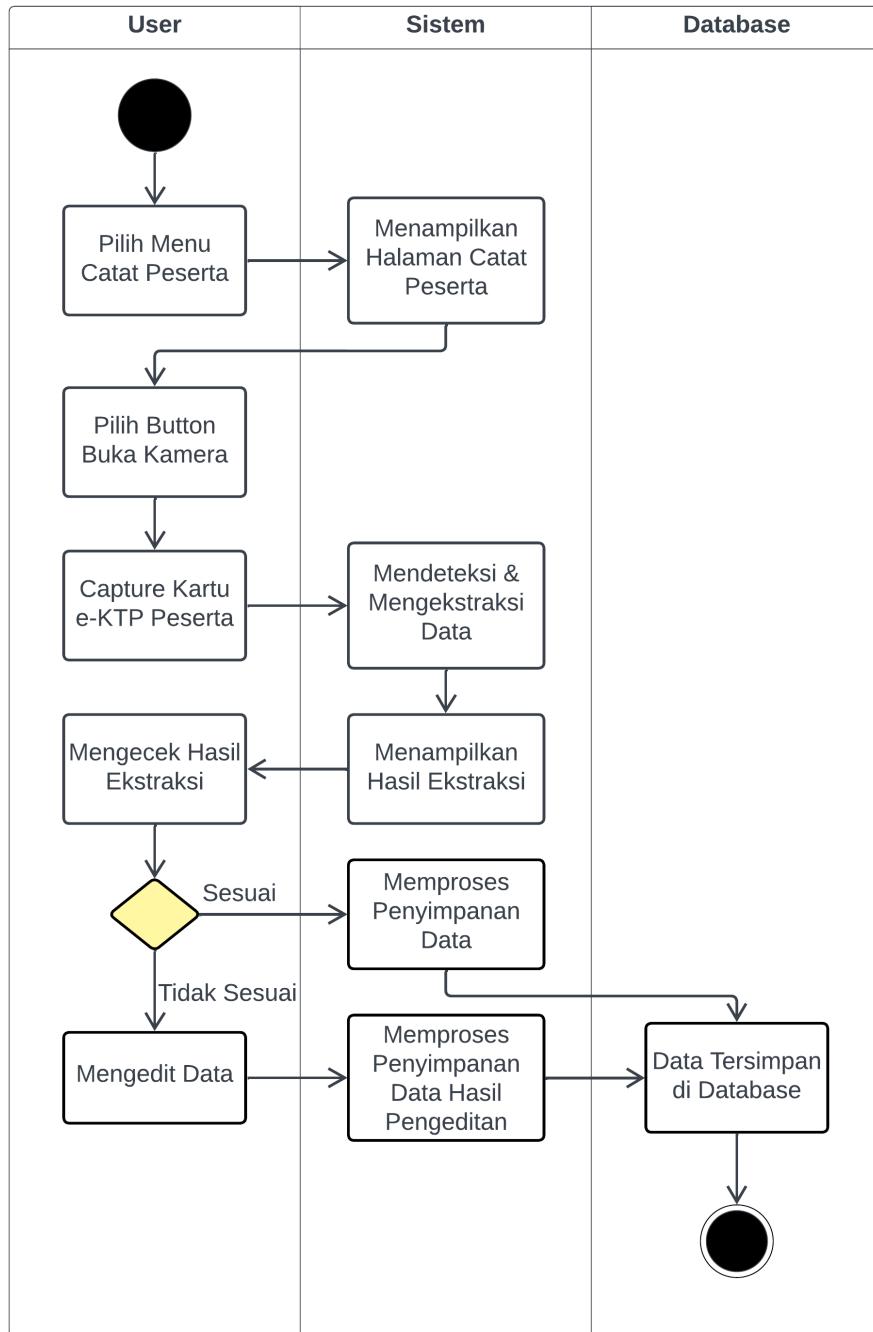
Gambar 3.13 Hasil Ekstraksi Data Peserta di Excel

3.9 Activity Diagram

Activity diagram merupakan visual yang menggambarkan aktivitas atau tindakan yang terjadi pada suatu aplikasi sehingga dapat diketahui serangkaian alur kerja dari satu proses ke proses lain. Dengan ini juga diharapkan *user* memahami dengan mudah cara penggunaan, fungsi yang terdapat pada aplikasi serta proses yang terjadi di aplikasi.

3.9.1 Proses Catat Peserta

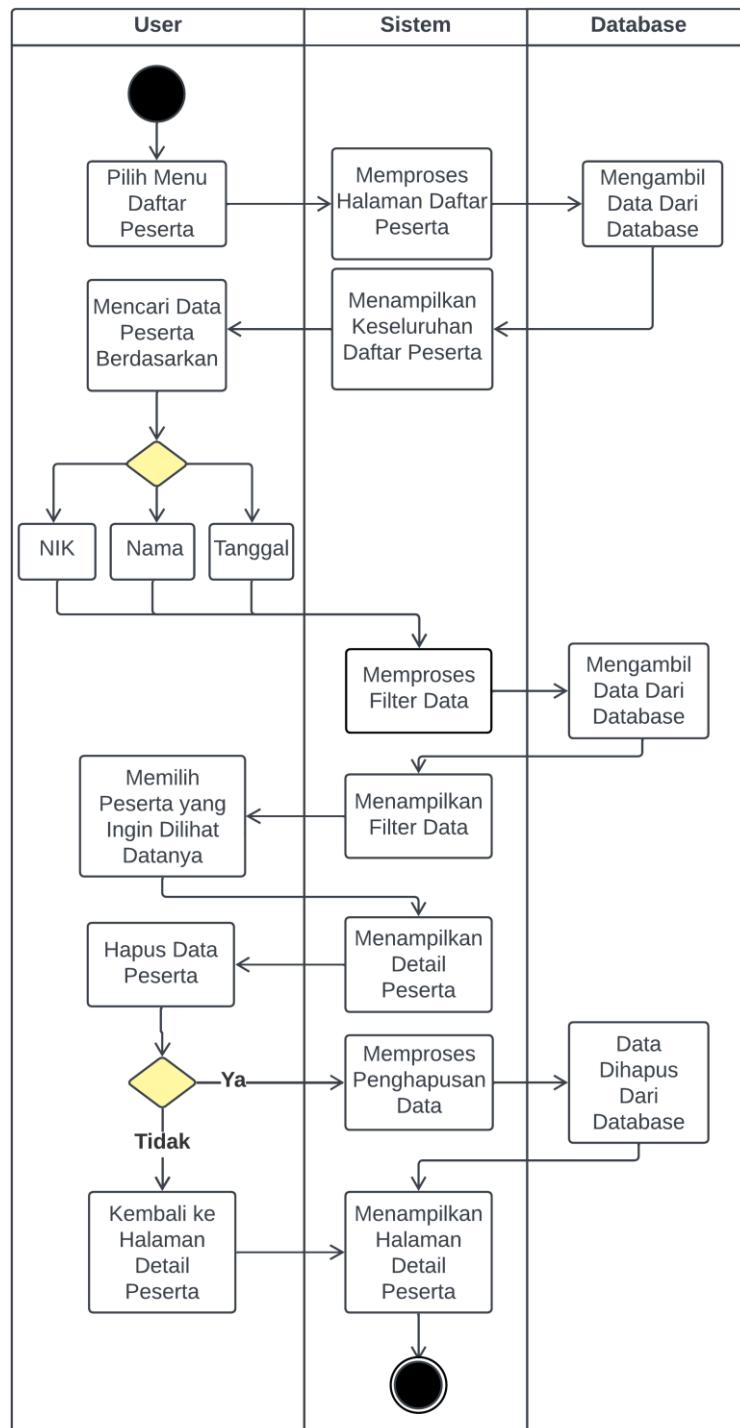
Terdapat beberapa rangkaian proses pada menu catat peserta. Pertama *user* membuka aplikasi android lalu memilih menu catat peserta. Setelah itu maka sistem akan menampilkan halaman catat peserta yang dimana akan tampak halaman *form* data yang masih kosong. Maka untuk melakukan proses catat peserta *user* akan klik buka kamera lalu melakukan pemotretan/meng-*capture* kartu e-KTP yang akan diproses. Setelah itu sistem akan melakukan proses pendekripsi data dan mengekstraksi data tersebut menjadi data teks yang akan dimuat pada sistem. Setelah proses selesai maka sistem akan menampilkan hasil dari ekstraksi data. Apabila data yang terekstraksi sudah sesuai maka dapat memilih tombol simpan untuk menyimpan data ke sistem yang kemudian di simpan kedalam *database* sistem. Namun apabila data yang terekstraksi masih belum sesuai, maka *user* dapat mengedit data yang salah terekstraksi. Setelah mengecek seluruh data dan sesuai maka sistem akan melakukan proses penyimpanan data ke *database* sistem. Tampilan *activity diagram* proses catat peserta dapat dilihat pada gambar 3.14 .



Gambar 3.14 *Activity Diagram* Catat Peserta

3.9.2 Proses Menu Daftar Peserta

Pada menu ini, terdapat rangkaian proses yang dilalui. Adapun *activity diagram* proses ini dapat dilihat pada gambar 3.15.



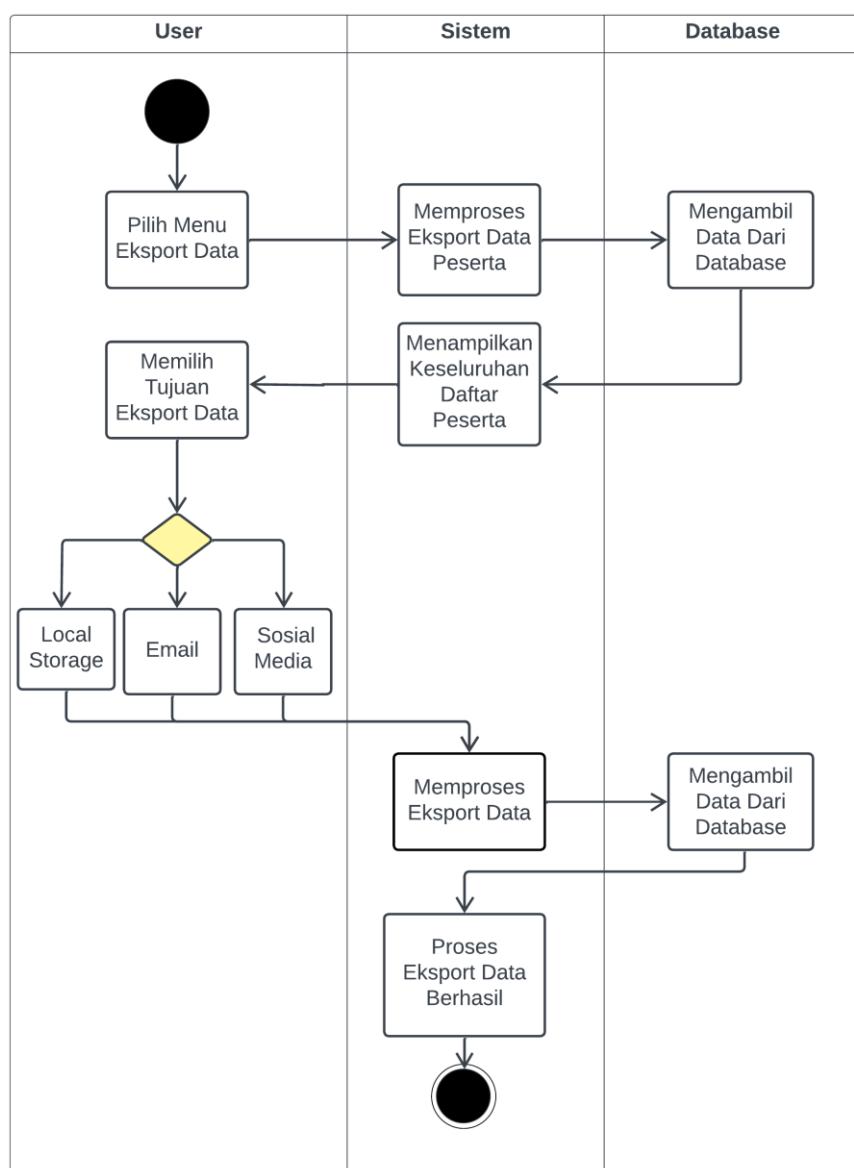
Gambar 3.15 Activity Diagram Daftar Peserta

Proses Dimulai dari *user* memilih menu daftar peserta. Lalu sistem memproses halaman tersebut lalu mengambil data peserta dari *database* lalu keseluruhan data ditampilkan pada halaman daftar peserta. *User* memilih 3 opsi pencarian data, yakni berdasarkan NIK, Nama ataupun Tanggal data masuk. Setelah itu maka sistem akan memproses data, lalu *database* akan memberikan data berdasarkan

pilihan filter *user*. Barulah sistem menampilkan data tersebut. Selanjutnya *user* dapat melihat detail data dengan klik data hasil filter datanya. *User* dapat melakukan hapus atau tidak data tersebut. Apabila hendak menghapus data, maka sistem akan memprosesnya lalu *database* akan menghapus data tersebut. Apabila tidak ingin dihapus maka *user* dapat kembali ke halaman daftar peserta. Dan proses daftar peserta ini selesai.

3.9.3 Proses Eksport Data

Pada proses ini terdapat aktivitas yang dilalui. *Activity diagram* pada proses eksport data ini dapat dilihat pada gambar 3.16.

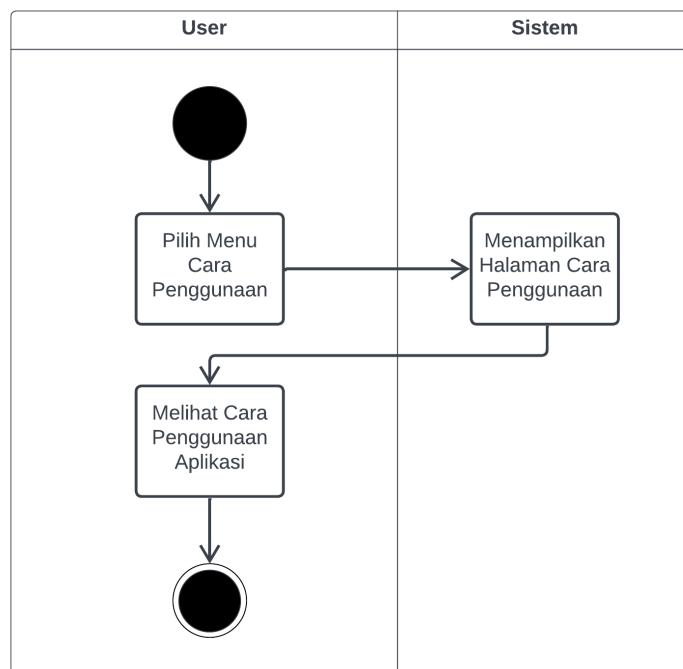


Gambar 3.16 Activity Diagram Eksport Data

Proses dimulai dari *user* memilih menu eksport data, lalu sistem akan melakukan pemrosesan data. *User* diberi pilihan eksport data yakni ke penyimpanan lokal, eksport via *email* dan via sosial media. Setelah memilih maka sistem akan memproses eksport data, lalu database memberi data ke sistem yang kemudian proses eksport data tersebut selesai dilaksanakan.

3.9.4 Proses Cara Penggunaan

Activity diagram menu cara penggunaan dapat dilihat pada gambar 3.17.

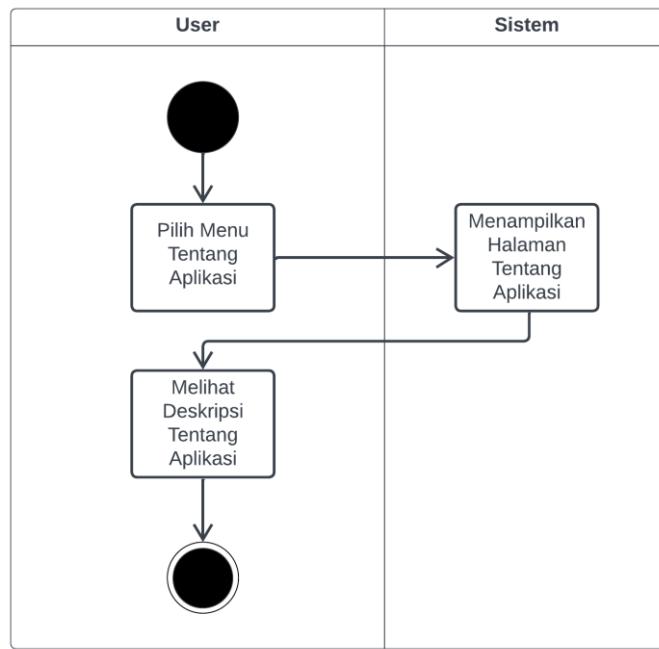


Gambar 3.17 *Activity Diagram* Cara Penggunaan

Pada proses ini, aktivitas yang dilalui yakni *user* memilih menu cara penggunaan. Maka sistem akan menampilkan halaman tersebut secara rinci keseluruhan proses dan instruksi cara penggunaan aplikasi pencatatan peserta vaksinasi COVID-19 ini. *User* dapat Melakukan scroll pada halaman ini.

3.9.5 Proses Tentang Aplikasi

Proses tentang aplikasi memiliki proses antara lain, *user* memilih menu tentang aplikasi, maka sistem akan menampilkan halaman tentang aplikasi. Lalu *user* dapat melihat halaman ini yang terdapat informasi deskripsi tentang aplikasi. Adapun *activity diagram* proses ini dapat dilihat pada gambar 3.18.



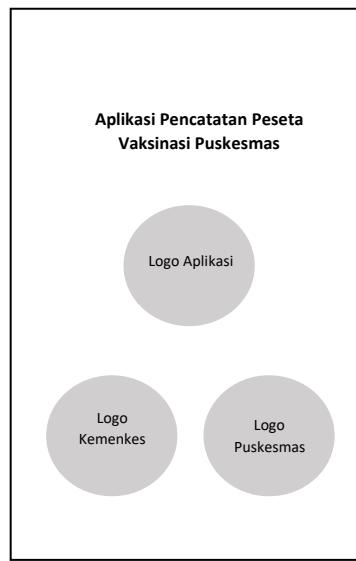
Gambar 3.18 *Activity Diagram* Tentang Aplikasi

3.10 Perancangan Antarmuka Sistem

Bagian ini membuat rancangan awal untuk antarmuka android pencatatan peserta vaksinasi puskesmas beserta dengan berbagai fitur yang tersedia yang nantinya akan digunakan oleh pengguna. Terdapat tampilan halaman pada sistem aplikasi ini seperti:

3.10.1 Rancangan Tampilan Splashscreen

Splashscreen adalah halaman pembuka yang menampilkan judul dan logo aplikasi terdapat pula logo puskesmas dan kemenkes karena dua instansi tersebut merupakan latar belakang dan penunjang dari proses digitalisasi sistem untuk dibidang kesehatan. Tampilan perancangan *splashscreen* aplikasi dapat dilihat pada gambar 3.19.



Gambar 3.19 Rancangan Splashscreen Aplikasi

3.10.2 Rancangan Tampilan Home

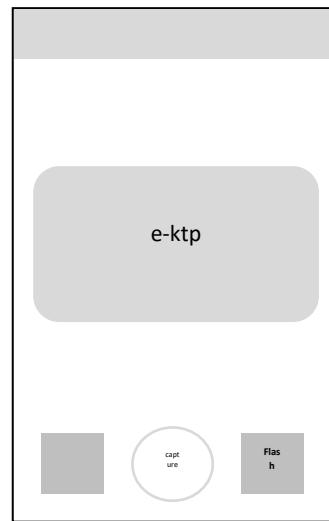
Halaman utama dari aplikasi, memuat informasi utama serta fitur yang terdapat pada aplikasi. Terdapat judul aplikasi dibagian atas, lalu terdapat menu-menu utama aplikasi seperti pada gambar 3.20.



Gambar 3.20 Rancangan Tampilan Home

3.10.3 Rancangan Tampilan Scan E-KTP

Halaman ini merupakan tampilan untuk melakukan pemindaian terhadap kartu e-KTP yang kemudian citra yang ditangkap akan diproses untuk dilakukan deteksi dan rekognisi terhadap data yang termuat pada citra e-KTP tersebut. Layar tampilan ini sama seperti tampilan memotret pada masing-masing *smartphone*. Tampilan tampak pada gambar 3.21.



Gambar 3.21 Rancangan Tampilan *Scan e-KTP*

3.10.4 Rancangan Tampilan Daftar Peserta

Menampilkan seluruh daftar peserta yang dapat difilter berdasarkan nik,nama dan tanggal masuk.. Gambar 3.22 merupakan rancangan dari halaman tersebut.



Gambar 3.22 Rancangan Tampilan Daftar Peserta

3.10.5 Rancangan Tampilan Detail Peserta

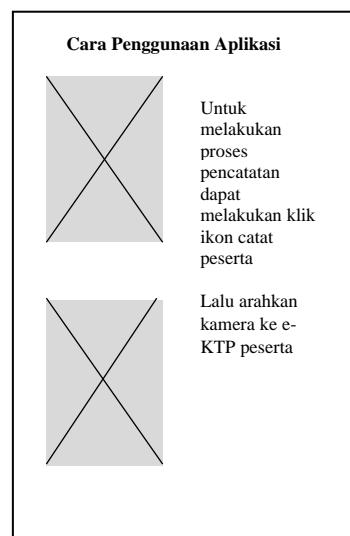
Menampilkan detail data keseluruhan berdasarkan ekstraksi kartu e-KTP, mulai dari foto nik nama hingga tanda tangan. Gambar 3.23 ialah rancangan dari halaman ini.

Daftar Peserta	
	Foto KTP
nik	
nama	
Tempat lahir	
Tanggal lahir	
Jenis Kelamin	

Gambar 3.23 Rancangan Detail Peserta

3.10.6 Rancangan Tampilan Petunjuk Penggunaan Aplikasi

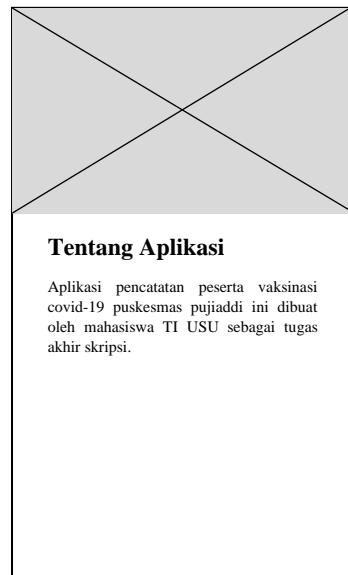
Halaman ini ditujukan kepada *user* yang belum memahami cara penggunaan aplikasi. Menampilkan detail cara penggunaan mulai dari *scan* e-KTP peserta hingga output. Gambar 3.24 merupakan rancangan tampilan halaman petunjuk penggunaan aplikasi.



Gambar 3.24 Rancangan Tampilan Petunjuk Penggunaan

3.10.7 Rancangan Tampilan Tentang

Menampilkan detil aplikasi, latar belakang, tujuan dan juga penjelasan mengenai aplikasi. Terdapat juga gambar puskesmas sebagai tempat pengaplikasian aplikasi ini seperti tampak pada gambar 3.25.



Gambar 3.25 Rancangan Tampilan Tentang

BAB 4

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Sebelum eksekusi rencana yang telah disusun dengan membuat sistem berbasis *mobile* Android, terlebih dahulu menyiapkan beberapa pendukung. Implementasi melibatkan *device* komputer sebagai proses pelatihan dan *smartphone* sebagai pengujian serta penggunaan perangkat keras dan lunak pendukung yang mampu membangun sistem hingga selesai dengan baik. Bahasa pemrograman yang diadopsi untuk implementasi adalah python dan javascript.

4.1.1 Lingkungan Implementasi

Bagian ini merujuk pada spesifikasi *device* yang digunakan selama proses implementasi sistem. Perangkat keras komputer memiliki spesifikasi yakni:

- CPU Intel Core i5-11400H 2.70GHz
- Kapasitas RAM 8GB
- Storage SSD 512GB

Adapun perangkat lunak komputer yang digunakan adalah:

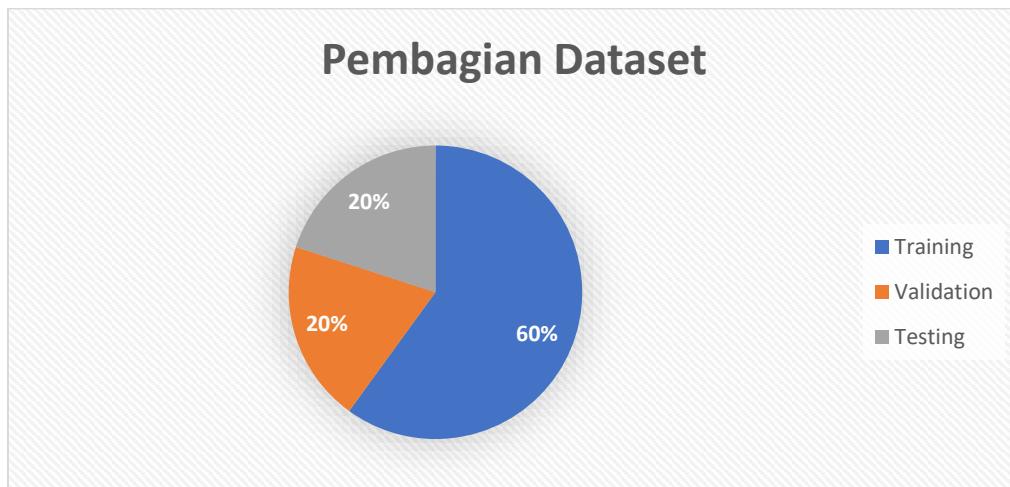
- Google Colaboratory
- OpenCV 4.1.1
- ONNX 1.9.0
- Nodejs 18.17.1
- Postman 10.17.2

Juga terdapat spesifikasi untuk perangkat *smartphone* sebagai pengujian implementasi sistem, seperti berikut:

- Sistem Android versi 12
- CPU Mediatek G95 Octa-core Max 2.05GHz
- RAM 6 GB
- Storage 128GB
- Kamera resolusi 64MP

4.1.2 *Implementasi Data*

Data pada penelitian ini dikumpulkan penulis baik langsung maupun tidak langsung oleh pemilik masing-masing e-KTP. Untuk pengambilan citra menggunakan kamera smartphone dengan resolusi minimal 12 MP serta data lain dikumpulkan dari dataset *Roboflow*. Adapun grafik pembagian data seperti pada gambar 4.1.



Gambar 4.1 Grafik Pembagian Dataset Penelitian

Data yang telah dikumpulkan kemudian dilakukan proses *preprocessing* yang melewati proses *resizing* dan *labelling*. Proses *labelling* dilakukan dengan aplikasi LabelIMG karena aplikasi tersebut mendukung proses *annotation* yang menghasilkan dataset berupa dataset txt *annotation* YOLO yang mendukung untuk proses *processing* dengan YOLOv7. Gambar 4.2 merupakan dataset citra beserta file yang telah dilabel untuk proses pelatihan.



Gambar 4.2 Dataset YOLO Untuk Proses *Training*

4.1.3 Pelatihan Sistem

Proses ini dilakukan untuk dapat mempelajari ciri khas tiap objek yang dideteksi menggunakan metode YOLOv7 dengan memanfaatkan GPU yang terdapat pada Google Colaboratory . Gambar 4.3 merupakan proses dari model yang akan dibuat.

```

+ Kode + Teks
all 160 1600 0.996 0.996 0.995 0.52
Q Epoch gpu_mem box obj cls total labels img_size
127/249 7.38G 0.02366 0.01981 0.0002495 0.04371 68 640: 100% 62/62 [00:51<00:00, 1.21it/s]
Class Images Labels P R mAP@.5 mAP@.5:.95: 100% 10/10 [00:05<00:00, 1.95it/s]
{x} all 160 1600 0.996 0.997 0.993 0.58

Epoch gpu_mem box obj cls total labels img_size
128/249 7.38G 0.02354 0.01963 0.0002896 0.04346 21 640: 100% 62/62 [00:48<00:00, 1.27it/s]
Class Images Labels P R mAP@.5 mAP@.5:.95: 100% 10/10 [00:07<00:00, 1.41it/s]
{y} all 160 1600 0.996 0.997 0.991 0.574

Epoch gpu_mem box obj cls total labels img_size
129/249 7.38G 0.02248 0.02023 0.0002393 0.04301 55 640: 100% 62/62 [00:48<00:00, 1.28it/s]
Class Images Labels P R mAP@.5 mAP@.5:.95: 100% 10/10 [00:05<00:00, 1.80it/s]
{z} all 160 1600 0.994 0.996 0.991 0.567

Epoch gpu_mem box obj cls total labels img_size
130/249 7.38G 0.02242 0.01961 0.0002644 0.04229 47 640: 100% 62/62 [00:49<00:00, 1.27it/s]
Class Images Labels P R mAP@.5 mAP@.5:.95: 100% 10/10 [00:04<00:00, 2.07it/s]
{<>} all 160 1600 0.994 0.997 0.992 0.575

Epoch gpu_mem box obj cls total labels img_size
131/249 7.38G 0.02319 0.02036 0.0002975 0.04384 30 640: 100% 62/62 [00:50<00:00, 1.22it/s]
Class Images Labels P R mAP@.5 mAP@.5:.95: 100% 10/10 [00:05<00:00, 1.89it/s]
{<>} all 160 1600 0.994 0.997 0.993 0.574

```

Gambar 4.3 Proses *Training* YOLOv7 di Colab

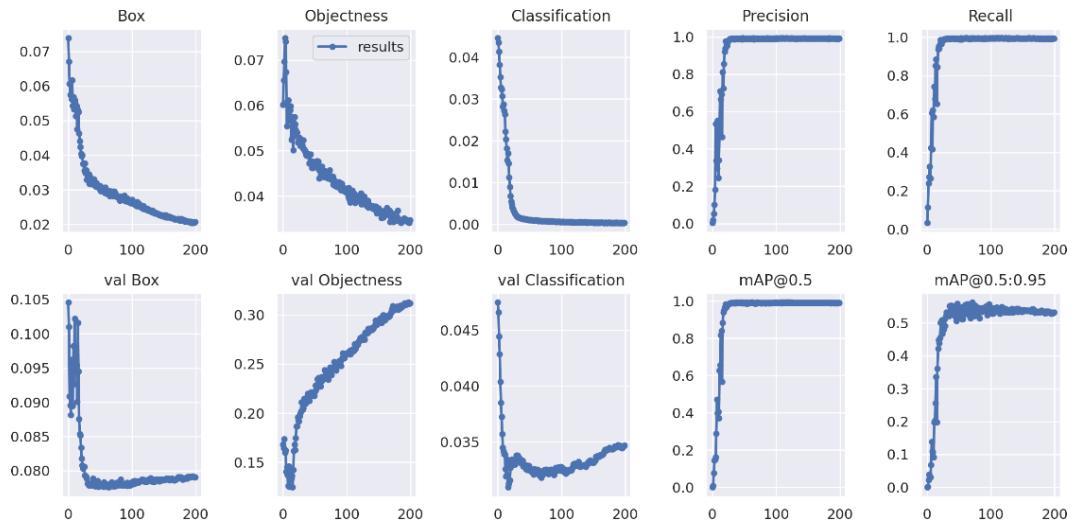
Pada *training* , penulis melakukan percobaan *training* pada tiap parameter yang berbeda seperti *batch-size* mulai dari 4, 8 dan 16 adapun masing-masing *batch-size* dengan *epochs* 200, 250 dan 300. Untuk *batch* 8 dan *epoch* 200 dilakukan selama kurang lebih 3 jam 48 menit dan menghasilkan *precision* sebesar 0.994, *Recall* bernilai 0.996, nilai mAP50 sebesar 0.995 dan juga nilai mAP50-95 sebesar 0.57. Untuk proses *training* *batch* 8 dan 16 dan *epoch* 250 menghabiskan waktu kurang lebih 4 jam 30 menit dan

menghasilkan nilai yang tidak berbeda secara signifikan. Disini penulis memilih menggunakan model yang sudah dilatih dengan *batch* 8 dan *epoch* 200 karena menghasilkan mAP yang lebih baik yang disimpan dengan nama *best.pt* dalam folder weights setelah proses *training* selesai. Adapun tabel 4.1 merupakan perbedaan hasil parameter training model YOLOv7.

Tabel 4.1 Training Model

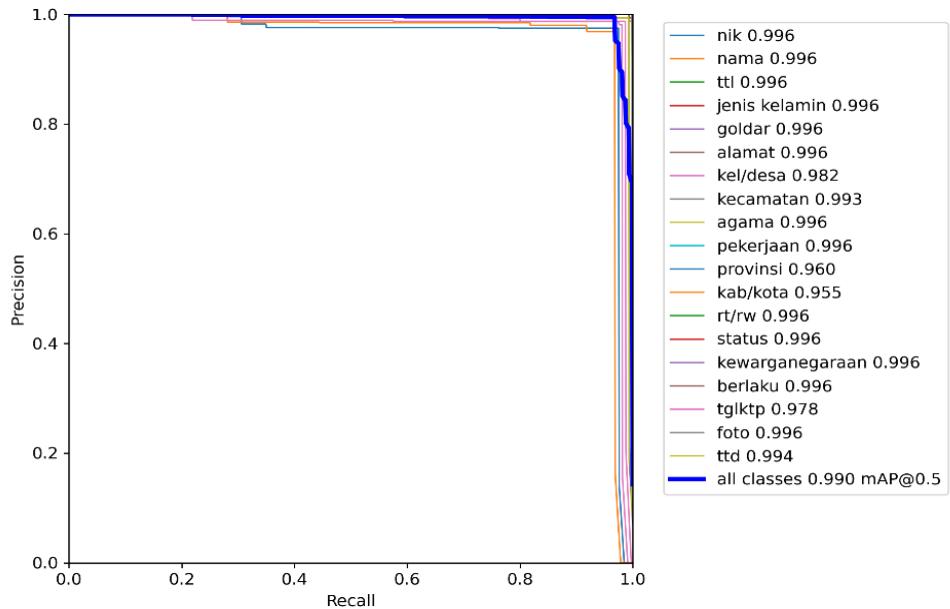
Batch	Epochs	mAP
4	200	0,993
4	250	0,993
4	300	0,992
8	200	0,994
8	250	0,992
8	300	0,990
16	200	0,993
16	250	0,992
16	300	0,992

Kemudian terdapat tiga *loss* yang juga ditunjukkan pada grafik di mana *object loss* menunjukkan seberapa baik algoritma dapat mengalokasikan titik tengah dari objek dan seberapa baik *bounding box* yang diprediksi mencakup objek. *Objectness* merupakan ukuran probabilitas suatu objek ada pada *region of interest* yang diberikan. *Classification loss* menunjukkan seberapa baik algoritma dapat memprediksi *class* yang benar dari objek yang diberikan. Di mana ketiga *loss* tersebut menunjukkan penurunan yang signifikan. Gambar 4.4 merupakan grafik dari hasil *training* dengan *batch* 8 dan *epoch* 200.



Gambar 4.4 Hasil Training Batch 8 Epoch 200

Untuk kurva *precision recall* seperti pada gambar 4.5 yang menampilkan *trade-off* antara *precision* dan *recall* pada *threshold* yang berbeda. Adapun nilai *recall* dapat dilihat pada sumbu x. Adapun sumbu y merupakan nilai *precision*.



Gambar 4.5 Kurva Precision-Recall

4.2 Implementasi Perancangan Antarmuka

Membahas desain tampilan antarmuka berdasarkan perancangan, yaitu:

4.2.1 Tampilan Halaman Splashscreen

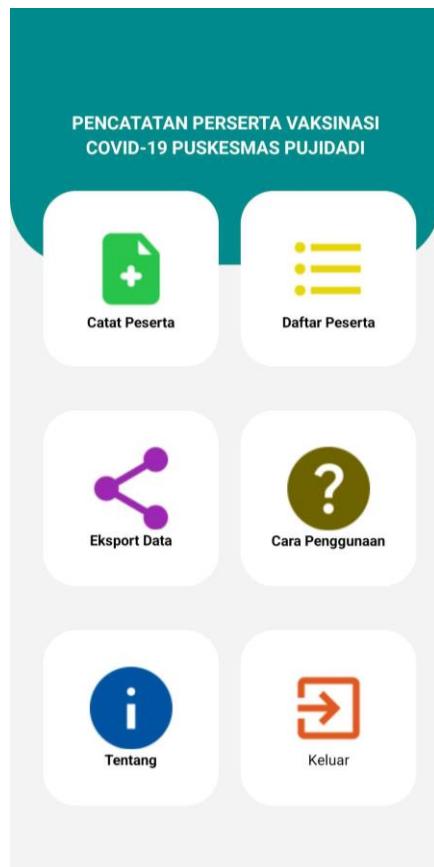
Seperti yang dijelaskan pada bagian rancangan sebelumnya, halaman ini merupakan halaman awal pembuka aplikasi yang menampilkan judul aplikasi, logo aplikasi serta dua logo instansi yang memiliki hubungan dengan implementasi sistem seperti logo puskesmas dan logo kemenkes. Adapun warna dasar merupakan warna *dark cyan* dengan kode hex #008B8B. Warna ini merupakan warna utama dari aplikasi ini dikarenakan diambil dari warna pertama dari logo kemenkes. Gambar 4.6. adalah tampilan antarmuka *splashscreen*.



Gambar 4.6 Halaman *Splashscreen*

4.2.2 Tampilan Halaman Home

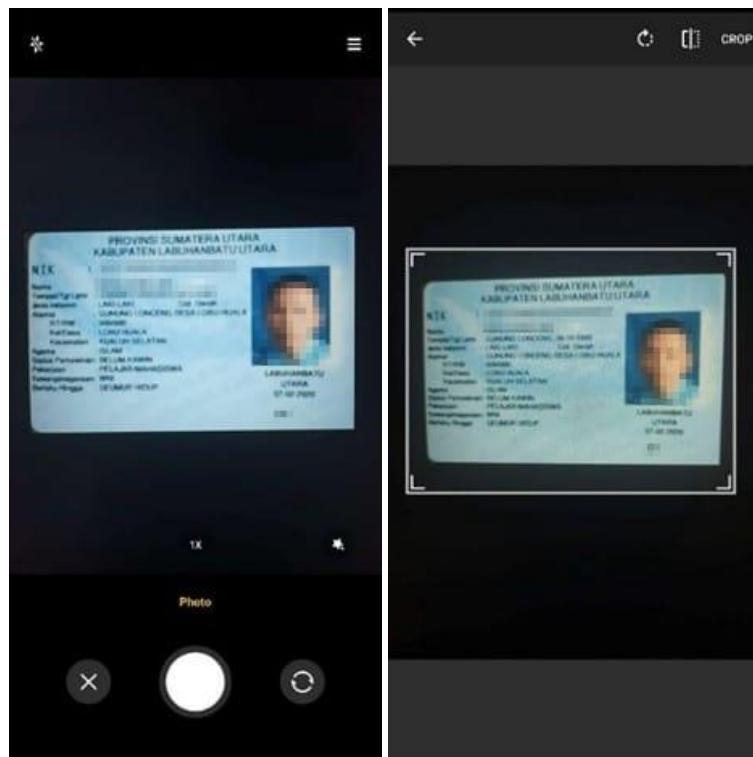
Tampilan awal aplikasi. Masing-masing ikon memiliki warna yang berbeda sehingga memudahkan petugas dalam membedakan tiap-tiap menu. Adapun menu antara lain Catat Peserta, Daftar Peserta, Eksport Data, Cara Penggunaan, Tentang dan Keluar. Adapun tampilan *home screen* seperti tampak pada gambar 4.7.



Gambar 4.7 Halaman *Homescreen*

4.2.3 Tampilan Halaman Scan

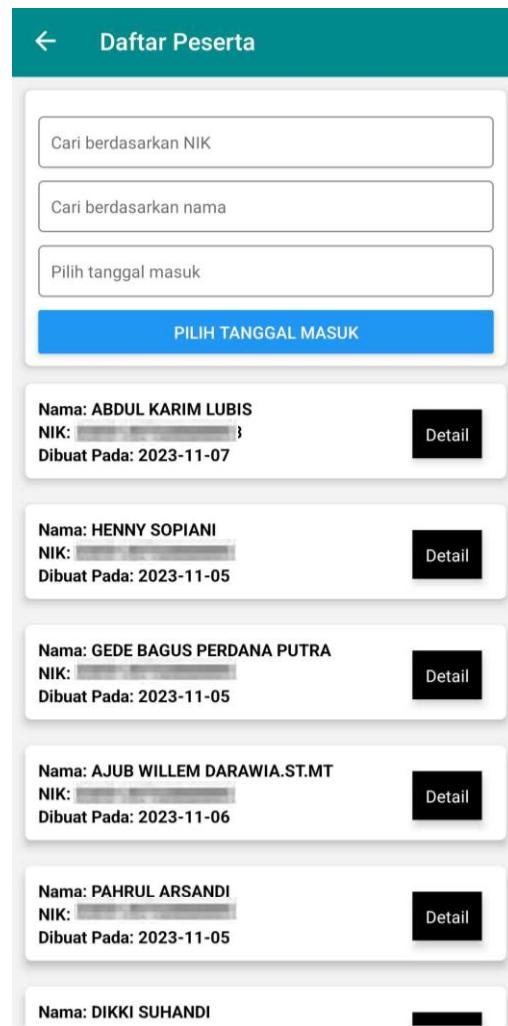
Halaman ini berisi tampilan aplikasi yang akan membuka *hardware* kamera pada *smartphone*. Lalu *user* Melakukan pemotretan terhadap kartu e-KTP yang hendak di ekstraksi datanya, lalu meng-*crop* citra untuk proses *scanning* e-KTP. Untuk tampilan halaman seperti tampak pada gambar 4.8.



Gambar 4.8 Halaman Scan

4.2.4 Tampilan Halaman Daftar Peserta

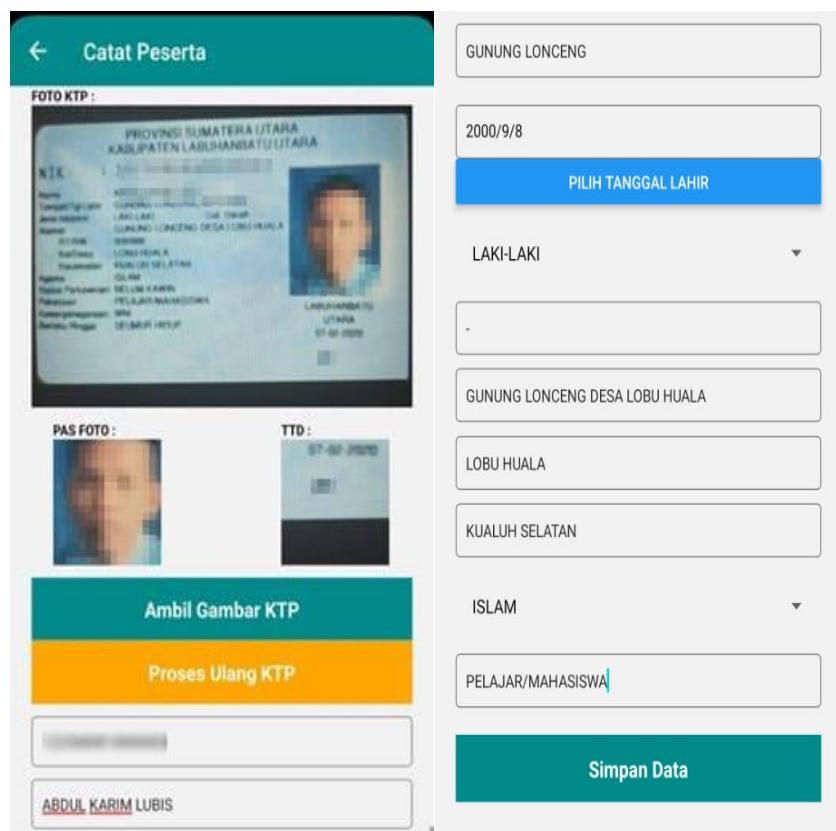
Seperti yang dijelaskan di bagian perancangan, halaman ini memuat seluruh daftar peserta yang telah diinput ke sistem melalui proses pemindaian e-KTP. Pada halaman ini selain menampilkan daftar peserta yang sudah diinput ke sistem, juga terdapat fitur *search* yang berguna untuk mencari peserta berdasarkan nama peserta yang dimasukkan sehingga mempermudah petugas dalam mencari peserta yang dimaksud. Lalu ada pula fitur filter tanggal yang berfungsi untuk melihat data berdasarkan tanggal masuknya data ke android.



Gambar 4.9 Halaman Daftar Peserta

4.2.5 Tampilan Detail Peserta

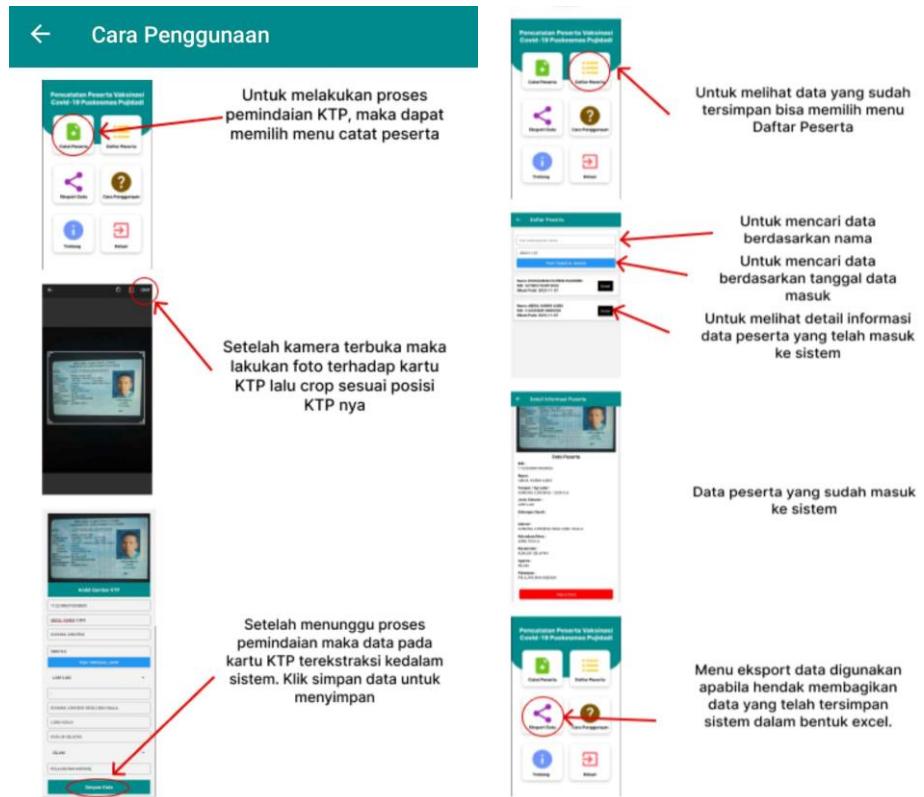
Berisi hasil dari pemindaian e-KTP ditampilkan secara berurutan yang kemudian masing-masing data yang berhasil di ekstrak masih bisa diedit apabila terjadi kesalahan dalam ekstraksi data. Seperti pada Gambar 4.10. Data-data yang berhasil diekstraksi dapat disimpan ataupun dihapus apabila terdapat kesalahan sistem.



Gambar 4.10 Halaman Detail Peserta

4.2.6 Tampilan Halaman Cara Penggunaan

Berisi instruksi dari penggunaan aplikasi untuk mencatat e-KTP peserta dengan pemindaian kartu. Langkah demi Langkah dijelaskan untuk mempermudah petugas baru dalam menggunakan aplikasi pencatatan peserta vaksinasi COVID-19 puskesmas ini. Untuk tampilannya tampak seperti pada gambar 4.11.



Gambar 4.11 Halaman Cara Penggunaan

4.2.7 Tampilan Halaman Tentang

Seperti yang dijelaskan pada perancangan sebelumnya, halaman tentang aplikasi berisi keterangan pembuatan aplikasi yang ditampilkan juga gambar puskesmas Pujidadi sebagai tempat implementasi dari penelitian ini seperti pada gambar 4.12, lalu dijelaskan fitur dari sistem.



Aplikasi Pencatatan Peserta Vaksinasi COVID-19
Puskesmas Pujidadi dibuat oleh mahasiswa Teknologi
Informasi USU untuk membantu dan sebagai digitalisasi
proses pendataan para peserta vaksinasi di lingkungan
Pusat Kesehatan Masyarakat (Puskesmas) Kelurahan
Pujidadi, Kecamatan Binjai Selatan, Kotamadya Binjai,
Sumatera Utara.

Hal ini dilakukan dikarenakan proses input data pasien
masih dilakukan secara manual yang memerlukan waktu
ditengah antusias para peserta vaksinasi COVID-19 yang
sangat banyak saat itu yang mengakibatkan tempat
puskesmas tidak mampu menampung jumlah peserta.

Adapun beberapa fitur yang dimiliki antara lain :

- Ekstraksi Data KTP Peserta
- Edit dan Perbaharui Data Peserta
- Export Data Peserta Berupa csv/excel
- Cara Penggunaan Aplikasi

Dengan adanya aplikasi ini diharapkan dapat memberikan
cara terbaik dan memudahkan petugas kesehatan
puskesmas dalam mengelola data pasien.

Gambar 4.12 Halaman Tentang

4.3 Pengujian Sistem

Tahapan menguji model sistem yang telah dibangun dari proses *training* untuk dapat diketahui kemampuan dari model untuk mendekripsi dan mengekstraksi data dari e-KTP ke sistem android. Pengujian dilakukan dengan *smartphone* yang sudah terpasang aplikasi model yang di *training* dengan *batch* 8 dan *epoch* 200 serta *library PaddleOCR* nya. Data yang digunakan pada pengujian sistem yaitu data uji sebanyak 160 citra e-KTP yang jumlah keseluruhannya memiliki 3040 atribut data. Proses deteksi dan ekstraksi sistem memakan waktu 5-10 detik, bergantung pada kualitas e-KTP dan kecepatan *upload* data jaringan. Terdapat dua pengujian pada pengujian untuk sistem seperti pada gambar 4.2. Pengujian deteksi data yaitu pendekripsi terhadap seluruh data oleh YOLOv7 yang termuat dalam citra e-KTP. Adapun pengujian ekstraksi data yaitu untuk melihat kebenaran hasil ekstraksi dari sistem terhadap karakter teks/data pada citra e-KTP berdasarkan *library Paddle OCR*.

Tabel 4.2 Hasil Pengujian

No.	Nama File	Citra	Pengujian Deteksi Data		Pengujian Ekstraksi Data	
			Data yang Terdeteksi	Data yang Tidak Terdeteksi	Data yang Terekstraksi Benar	Data yang Terekstraksi Salah
1	641.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-Alamat	<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	-
2	642.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-	<ul style="list-style-type: none"> - Nama - Jenis Kelamin - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	<ul style="list-style-type: none"> - NIK - Tempat Tanggal Lahir - Golongan Darah - Alamat

3	643.jpg		<ul style="list-style-type: none"> - NIK - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	<ul style="list-style-type: none"> - NIK - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	-	
4	644.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	- Alamat	
5	645.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kab/Kota - Provinsi 	<ul style="list-style-type: none"> - Kecamatan 	<ul style="list-style-type: none"> - NIK - Nama - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kab/Kota - Provinsi - Agama - Status 	- Tempat Tanggal Lahir

			<ul style="list-style-type: none"> - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 		<ul style="list-style-type: none"> - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	
6	646.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	- Alamat	<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-
7	647.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	-

8	648.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat - Tanggal Lahir - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat - Tanggal Lahir - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	-
9	649.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat - Tanggal Lahir - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat - Tanggal Lahir - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	-
10	650.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat - Tanggal Lahir - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat - Tanggal Lahir - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan 	-

			<ul style="list-style-type: none"> - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 		<ul style="list-style-type: none"> - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	
...
...
...
151	791.jpg		<ul style="list-style-type: none"> - NIK - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	- Nama	<ul style="list-style-type: none"> - NIK - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	-
152	792.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	- Alamat

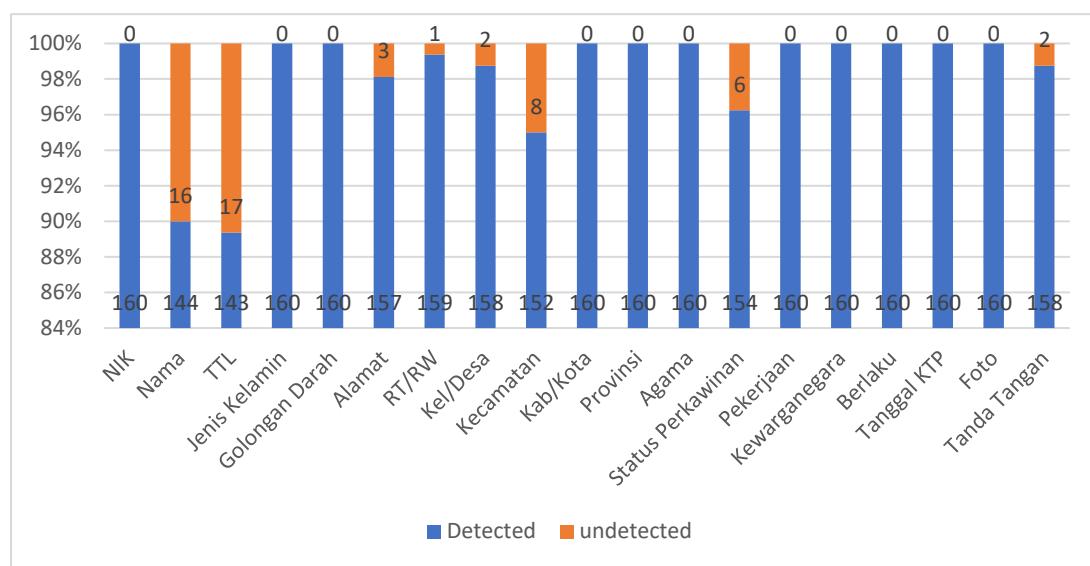
			<ul style="list-style-type: none"> - Foto - Tanda Tangan 			
153	793.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	- Alamat
154	794.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Kelurahan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	- Alamat - Kecamatan
155	795.jpg		<ul style="list-style-type: none"> - NIK - Nama - Jenis Kelamin - Golongan Darah - Alamat - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama 	<ul style="list-style-type: none"> - Tempat Tanggal Lahir 	<ul style="list-style-type: none"> - NIK - Nama - Jenis Kelamin - Golongan Darah - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status 	- Alamat - Tanggal KTP

			<ul style="list-style-type: none"> - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 		<ul style="list-style-type: none"> - Pekerjaan - Kewarganegaraan - Masa Berlaku 	
156	796.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	<ul style="list-style-type: none"> - Kecamatan 	<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	-
157	797.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat Tanggal Lahir - Jenis Kelamin - Golongan Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	...

158	798.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	...
159	799.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	- NIK
160	800.jpg		<ul style="list-style-type: none"> - NIK - Nama - Tempat - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan 	-	<ul style="list-style-type: none"> - NIK - Nama - Tempat - Jenis Kelamin - Golongan - Darah - Alamat - RT/RW - Kelurahan 	-

		<ul style="list-style-type: none"> - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP - Foto - Tanda Tangan 	<ul style="list-style-type: none"> - Kecamatan - Kab/Kota - Provinsi - Agama - Status - Pekerjaan - Kewarganegaraan - Masa Berlaku - Tanggal KTP 	
--	--	---	---	--

Dari tabel 4.2 , maka untuk hasil deteksi dapat dirangkum pada grafik yang terdapat pada gambar 4.13.



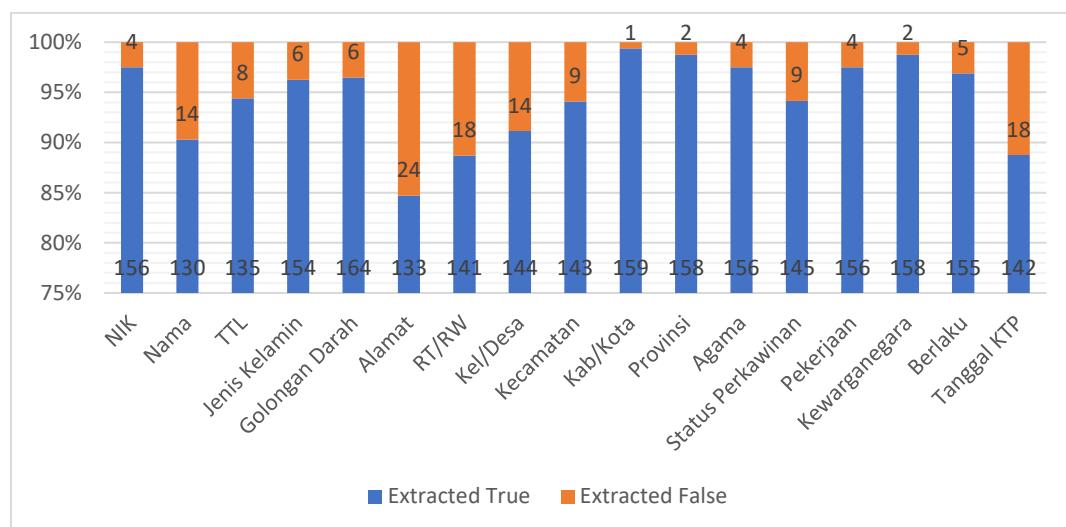
Gambar 4.13 Grafik Persentase Hasil Deteksi Data E-KTP

Dapat dilihat bahwa atribut data NIK, jenis kelamin, golongan darah, kabupaten/kota, provinsi, agama, pekerjaan, kewarganegaraan, berlaku, tanggal KTP dan Foto berhasil terdeteksi secara sempurna. Untuk sisanya seperti nama, TTL, alamat, rt/rw, kel/desa, kecamatan, status perkawinan dan tanda tangan masih terdapat data yang tidak terdeteksi. Untuk data seperti nama dan TTL terdapat beberapa data yang kurang sempurna dideteksi dikarenakan data tersebut memiliki dua baris di masing-masing datanya. Sebagai contoh adalah nama seseorang yang panjang hingga memerlukan dua baris posisi. Hal demikian terjadi juga dengan TTL dan alamat. Pada TTL daerah tempat lahir dan pada alamat, lokasi alamat yang panjang mengakibatkan data tersebut susah untuk dideteksi sehingga sistem tidak mampu mendeteksi data

tersebut. Sehingga menjadi kelemahan sistem ini yaitu data yang panjang hingga 2 baris menjadi susah untuk dideteksi.

Adapun data seperti RT/RW, kel/desa, kecamatan merupakan data yang sangat berdekatan yang berurutan. Hal ini menjadi tantangan dalam melakukan pendekripsi terhadap data tersebut. Untuk data status perkawinan ditemukan bahwa terdapat data yang sudah kabur dan terkena silau sumber cahaya sehingga tidak terdeteksi.

Selain itu, untuk grafik hasil ekstraksi pada penelitian ini dapat dilihat pada gambar 4.14.



Gambar 4.14 Grafik Persentase Hasil Ekstraksi Data E-KTP

Grafik menunjukkan bahwa data alamat paling banyak menjadi data yang terekstraksi kurang tepat. Hal ini terjadi karena data alamat menjadi data yang memiliki teks terbanyak Dimana datanya mencakup dua baris, dikarenakan panjangnya data alamat. Ketika data memiliki dua baris, pada sistem ini bagian baris kedua akan naik menyisip ke bagian baris pertama sehingga terjadi perubahan data yang mengakibatkan data tidak terekstraksi secara sempurna menjadi data yang sesungguhnya. Hal demikian juga terjadi pada data nama dan TTL.

Selanjutnya data seperti NIK, RT/RW, masa berlaku dan tanggal KTP kerap terjadi kesalahan ekstraksi karena data-data tersebut berbentuk angka. Masalah yang terjadi adalah angka nol (0) oleh sistem diekstraksi menjadi huruf o sehingga mengakibatkan kesalahan ekstraksi. Demikian pula pada data-data lain seperti jenis kelamin, golongan darah, kel/desa, kecamatan, kab/kota, provinsi, agama, status

perkawinan, pekerjaan dan kewarganegaraan. Dikarenakan pada e-KTP tiap-tiap hurufnya sangat berdekatan serta melihat kualitas kartu e-KTP tiap warga dari tiap daerah juga terdapat e-KTP dengan kualitas yang kurang baik sehingga dalam hal ini, sistem melakukan ekstraksi huruf menjadi huruf yang berbeda yang memiliki bentuk yang mirip. Adapun teks yang berdekatan menjadi bentuk karakter baru yang berbeda namun memiliki identik dengan gabungan dua karakter tersebut.

4.4 Penghitungan Hasil Uji Deteksi Data E-KTP

Uji deteksi data e-KTP yaitu untuk mengetahui akurasi sistem dalam pendekripsi data atribut pada e-KTP dengan mempertimbangkan jumlah atribut yang sudah ditentukan sebelumnya. Hasil pengujian tersebut terdokumentasi dalam tabel 4.1 sebagian dan terlampir secara keseluruhan pada lampiran 1 dan selanjutnya akan dilakukan perhitungan *confusion matrix* untuk menghitung nilai akurasi. Proses pengujian berdasarkan batasan masalah berlaku jarak pengambilan dengan *range* 9-12cm, dengan *angle* 90 derajat dan cahaya ruangan 400-800 lumen. Tabel 4.3 merupakan rincian lebih lanjut mengenai hasil perhitungan *True Positive*, *False Positive*, dan *False Negative* dari pendekripsi atribut data pada e-KTP.

Tabel 4.3. Hasil TP,FP dan FN Deteksi Data E-KTP

Atribut Data	TP	FP	FN
NIK	160	0	0
Nama	144	0	16
Tempat Tanggal Lahir	143	0	17
Jenis Kelamin	160	0	0
Golongan Darah	160	0	0
Alamat	157	0	3
RT/RW	159	0	1
Kel/Desa	158	0	2
Kecamatan	152	0	8
Kab/Kota	160	0	0
Provinsi	160	0	0
Agama	160	0	0

Status	154	0	6
Pekerjaan	160	0	0
Kewarganegaraan	160	0	0
Berlaku	160	0	0
Tanggal KTP	160	0	0
Foto	160	0	0
Tanda Tangan	158	0	2
Total	2985	0	55

Berdasarkan tabel 4.3, total 2985 data dari citra uji terlihat sebagai *True Positive* (TP). Dengan demikian memiliki hasil positif dan sistem mendeteksi infomasi data dengan benar sesuai dengan posisi datanya secara sempurna. Adapun 55 data sebagai *False Negative* (FN) karena sistem tidak mampu mendeteksi posisi data tersebut.

Hasil perhitungan *Performance Matrix* (akurasi, *precision*, *recall* dan *f1-score*) deteksi masing-masing atribut data e-KTP pada penelitian ini dapat dilihat pada tabel 4.4.

Tabel 4.4. Hasil Penghitungan *Performance Matrix* Deteksi E-KTP

Atribut Data	Akurasi	Precision	Recall	F1-Score
NIK	100%	1	1	1
Nama	90%	1	0,9	0,947
Tempat Tanggal Lahir	89,37%	1	0,893	0,943
Jenis Kelamin	100%	1	1	1
Golongan Darah	100%	1	1	1
Alamat	98,12%	1	0,981	0,99
RT/RW	99,37%	1	0,993	0,996
Kel/Desa	98,75%	1	0,987	0,993
Kecamatan	95%	1	0,95	0,974
Kab/Kota	100%	1	1	1
Provinsi	100%	1	1	1
Agama	100%	1	1	1
Status	96,25	1	0,962	0,98
Pekerjaan	100%	1	1	1

Kewarganegaraan	100%	1	1	1
Berlaku	100%	1	1	1
Tanggal KTP	100%	1	1	1
Foto	100%	1	1	1
Tanda Tangan	98,75%	1	0,987	0,993

Selanjutnya, maka akan dicari nilai akurasi, *precision*, *recall* dan f1-score deteksi sistem secara keseluruhan. Dengan menggunakan persamaan 2.8 maka nilai akurasi untuk uji deteksi dapat dihitung sebagai berikut:

$$\textbf{Akurasi Deteksi Data} = \frac{\text{Jumlah data terdeteksi}}{\text{Jumlah data keseluruhan}} \times 100\%$$

$$\textbf{Akurasi Deteksi Data} = \frac{2985}{3040} \times 100\%$$

$$\textbf{Akurasi Deteksi Data} = 98,19\%$$

Adapun nilai *precision* hasil deteksi data e-KTP ini mengikuti persamaan 2.9 adalah :

$$\textbf{Precision} = \frac{2985}{2985 + 0} = 1$$

Recall diperoleh melalui perbandingan prediksi positif dengan keseluruhan data yang benar positif, mengikuti rumus dari persamaan 2.10.

$$\textbf{Recall} = \frac{2985}{2985 + 55} = 0,981$$

Selanjutnya, F1-score dihitung sebagai rata-rata dari *precision* dan *recall* dengan menggunakan rumus dari persamaan 2.11. yaitu

$$\textbf{F1 - score} = 2 * \frac{1 * 0,981}{1 + 0,981} = 0,99$$

4.5 Penghitungan Hasil Uji Ekstraksi

Hasil uji ekstraksi yaitu fokus pada akurasi sistem dalam mengekstraksi kata/teks dari data e-KTP dengan membandingkan hasil sistem dengan data asli pada citra e-KTP. Dengan merujuk pada tabel 4.1 serta hasil pengujian yang terlampir pada lampiran 2, Langkah selanjutnya melibatkan perhitungan *confusion matrix* untuk memperoleh nilai

akurasi ekstraksi data teks pada citra e-KTP. Informasi lebih lanjut mengenai hasil perhitungan TP, FP dan FN dari ekstraksi data e-KTP tampak pada tabel 4.5.

Tabel 4.5. Hasil TP,FP,FN Ekstraksi Data E-KTP

Atribut Data	TP	FP	FN
NIK	156	4	0
Nama	130	14	0
Tempat Tanggal Lahir	135	8	0
Jenis Kelamin	154	6	0
Golongan Darah	154	6	0
Alamat	133	24	0
RT/RW	141	18	0
Kelurahan/Desa	144	14	0
Kecamatan	143	9	0
Kab/Kota	159	1	0
Provinsi	158	2	0
Agama	156	4	0
Status	145	9	0
Pekerjaan	156	4	0
Kewarganegaraan	158	2	0
Berlaku	155	5	0
Tanggal KTP	142	18	0
Total	2519	148	0

Berdasarkan tabel 4.4 untuk uji ekstraksi data e-KTP, sekitar 2519 atribut data terdaftar sebagai *True Positive* (TP). Adapun sebanyak 148 data termasuk *False Positive*(FP). Lalu terdapat 0 data yang dikategorikan sebagai *False Negative* (FN). Hal ini terlihat bahwa *library Paddle OCR* mampu mengekstraksi data dengan tepat meskipun masih terdapat kesalahan dalam pembacaan karakter teks dikarenakan objek citra yang sudah tidak terlalu jelas.

Hasil perhitungan *performance matrix* (akurasi, *precision*, *recall* dan *f1-score*) ekstraksi masing-masing atribut data pada penelitian ini dapat dilihat pada tabel 4.6.

Tabel 4.6 Hasil Penghitungan *Performance Matrix* Ekstraksi E-KTP

Atribut Data	Akurasi	Precision	Recall	F1-Score
NIK	97,5%	0,975	1	0,987
Nama	90,27 %	0,902	1	0,948
Tempat Tanggal Lahir	94,4 %	0,944	1	0,971
Jenis Kelamin	96,25 %	0,962	1	0,98
Golongan Darah	96,25 %	0,962	1	0,98
Alamat	84,71 %	0,847	1	0,917
RT/RW	89,24 %	0,892	1	0,942
Kelurahan/Desa	91,13 %	0,911	1	0,953
Kecamatan	94,07 %	0,94	1	0,969
Kab/Kota	99,37 %	0,993	1	0,996
Provinsi	98,75%	0,987	1	0,993
Agama	97,5%	0,975	1	0,987
Status	94,15%	0,941	1	0,969
Pekerjaan	97,5 %	0,975	1	0,987
Kewarganegaraan	98,75 %	0,987	1	0,993
Berlaku	96,87%	0,968	1	0,963
Tanggal KTP	88,75%	0,887	1	0,94

Selanjutnya, maka akan dicari nilai akurasi, *precision*, *recall* dan *f1-score* ekstraksi sistem secara keseluruhan. Dengan menggunakan persamaan 2.8 maka nilai akurasi untuk uji deteksi dapat dihitung sebagai berikut:

$$\text{Akurasi Ekstraksi Data} = \frac{2519}{2667} \times 100\% \quad 2519 \\ 2667$$

$$\text{Akurasi Ekstraksi Data} = 94,45\%$$

Adapun nilai *precision* dari ekstraksi data e-KTP ini berdasarkan persamaan 2.9 sebagai berikut:

$$\text{Precision} = \frac{2519}{2519 + 148} = 0,944$$

Nilai *Recall* dari ekstraksi data e-KTP ini berdasarkan persamaan 2.10 adalah :

$$\text{Recall} = \frac{2525}{2525 + 0} = 1$$

Dan nilai f-1 score dari ekstraksi data e-KTP berdasarkan persamaan 2.11 yaitu:

$$F1 - score = 2 * \frac{1 * 0,944}{1 + 0,944} = 0,97$$

Tabel 4.5 merupakan hasil akurasi, *recall*, *precision* dan f1-score keseluruhan pengujian deteksi dan ekstraksi pada penelitian ini.

Tabel 4.5 Hasil *Performance Matrix* Pengujian

Hasil Uji	Akurasi	Recall	Precision	F1-Score
Deteksi Data E-KTP	98,19%	0,981	1	0,99
Ekstraksi Data E-KTP	94,45%	1	0,944	0,97

Penelitian dengan judul Ekstraksi Data E-KTP Untuk Pencatatan Peserta Vaksinasi COVID-19 dengan Metode YOLOv7 dan *Paddle* OCR Berbasis Android ini menghasilkan suatu hasil mampu mendeteksi posisi dari data e-KTP dengan akurasi 98,19% dan mengekstraksi data pada citra e-KTP Indonesia dengan besaran akurasi sebesar 94,45%. Sistem dapat mendeteksi dan mengekstraksi data yang termuat pada citra e-KTP secara *mobile* pada sistem Android secara baik. Adapun kesalahan sistem pada pengekstraksian data dikarenakan objek pada citra yang sudah tidak tampak sehingga sistem tidak mampu memberikan deteksi dan rekognisi terhadap karakter teks tersebut. Selain itu posisi kartu E-KTP dengan sumber cahaya juga perlu diperhitungkan juga yang dapat berpengaruh dalam pengekstraksian teks. Adapun kesalahan lain adalah data teks yang terlalu kecil serta rapat mengakibatkan kesalahan sistem dalam merekognisi data menjadi teks yang berbeda serta data yang memiliki dua baris mengakibatkan teks tersisip sehingga merubah informasi data.

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Hasil kesimpulan penelitian didapatkan berdasarkan hasil pengujian sebelumnya yang antara lainnya adalah :

1. YOLOv7 dapat mendeteksi objek data pada e-KTP dengan sangat baik yakni dengan akurasi mencapai 98,19%, dengan nilai *recall* 0,981, *precision* 1 dan f1-score 0,99.
2. *Library Paddle* OCR dalam pengekstraksian karakter teks dari citra e-KTP menjadi *string* pada sistem *mobile* mendapatkan akurasi yang sangat baik yakni sebesar 94,45%.
3. Proses deteksi dan ekstraksi data pada sistem memerlukan waktu 5-10 detik, bergantung pada kualitas e-KTP.
4. Adapun yang menyebabkan akurasi ekstraksi kurang sempurna karena objek pada e-KTP yang berdekatan dan terlalu rapat menyebabkan sistem tidak mampu melakukan rekognisi ataupun terekognisi menjadi huruf berbeda, serta atribut data yang memuat dua baris tersisip sehingga merubah informasi data.
5. Penerapan metode YOLOv7 dapat dikombinasikan bersamaan dengan *Paddle* OCR memiliki hasil yang efektif dalam ekstraksi data karakter teks ke dalam sistem android secara bersamaan sehingga membantu pekerjaan petugas kesehatan.

5.2 Saran

Berikut ini juga terdapat pula saran agar penelitian bisa lebih baik yakni :

1. Diharapkan penggunaan *preprocessing* yang bervariasi sehingga menghasilkan hasil OCR yang lebih baik.
2. Pengambilan data diharapkan memperhatikan antara posisi e-KTP dengan posisi dimana sumber cahaya berasal agar hasil ekstraksi lebih optimal.
3. Pengembangan sistem selanjutnya diharapkan mampu memberikan rekomendasi ekstraksi karakter teks yang gagal terekstraksi oleh sistem dan mampu mengekstraksi teks yang memuat data hingga dua baris lebih dengan lebih sempurna serta dapat melakukan *scanning* kartu e-KTP dari semua sisi *angle*.
4. Diharapkan menggunakan metode pendekripsi berbeda agar dapat dibandingkan dengan hasil deteksi yang didapatkan oleh metode YOLOv7 serta menggunakan *library* OCR yang lainnya.
5. Apabila memungkinkan, sistem digarapkan dapat terhubung ke data kependudukan sehingga sistem mampu memvalidasi kebenaran data yang diinput secara otomatis berdasarkan nomor induk kependudukan (NIK).

DAFTAR PUSTAKA

- Aprilino, A. 2022. Implementasi Algoritma Yolo Dan Tesseract Ocr Pada Sistem Deteksi Plat Nomor Otomatis. *Jurnal Teknoinfo*, 16(1), 54–59.
- Bui, V., & Chang, L.-C. 2016. Deep learning architectures for hard character classification. *Proceedings on the International Conference on Artificial Intelligence (ICAI)*, 108.
- Chablani, M. 2017. YOLO—You only look once, real time object detection explained. *Towards Data Science*, 21.
- Chen, F., He, Y., & Shi, Y. 2022. Parents' and guardians' willingness to vaccinate their children against COVID-19: a systematic review and meta-analysis. *Vaccines*, 10(2), 179.
- Do, C.D., Vu, D.Q., Vu, H.T.T. 2023. Extraction Information from Vietnamese ID Card Images. Department of Computer Science FPT University Hanoi.
- Du, Y., Li, C., Guo, R., Yin, X., Liu, W., Zhou, J., Bai, Y., Yu, Z., Yang, Y., & Dang, Q. 2020. Pp-ocr: A practical ultra lightweight ocr system. *ArXiv Preprint ArXiv:2009.09941*.
- Eisenman, B. 2017. *Learning React Native: Building Native Mobile Apps with JavaScript*. O'Reilly Media. <https://books.google.co.id/books?id=wgg7DwAAQBAJ>
- Forsyth, D. A., & Ponce, J. 2002. *Computer vision: a modern approach*. prentice hall professional technical reference.
- Gomedé, E. 2023. *ONNX Run-time:Enabling Cross-Platform AI Model Inference*. Medium.Com. <https://medium.com/@evertongomedé/onnx-runtime-enabling-cross-platform-ai-model-inference-80f136ecbb2d>

- Hartley, R., & Zisserman, A. 2003. *Multiple view geometry in computer vision*. Cambridge university press.
- Hu, F., Xia, G.-S., Hu, J., & Zhang, L. 2015. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, 7(11), 14680–14707.
- Ivanov, M. 2022. *The Evolution of the YOLO Neural Networks Family From V1 to V7. Deelvin Machine Learning*. Medium. <https://medium.com/deelvin-machine-learning/the-evolution-of-the-yolo-neural-networks-family-from-v1-to-v7-48dd98702a3d>.
- Jain, A., Dubey, A., Gupta, R., Jain, N., & Tripathi, P. 2013. Fundamental challenges to mobile based ocr. *Vol*, 2(5), 86–101.
- Jonnalagadda, V. K. . 2019. *Object Detection YOLO v1 , v2, v3*. Medium. <https://medium.com/@venkatakrishna.jonnalagadda/object-detection-yolo-v1-v2-v3-c3d5eca2312a>
- Kim, M.-S., Moon, M.-K., & Han, C.-H. 2021. Expiration Date Notification System Based on YOLO and OCR algorithms for Visually Impaired Person. *The Journal of the Korea Institute of Electronic Communication Sciences*, 16(6), 1329–1338.
- Kumar, V. 2022. *Background Removal (general (Pixel-wise) vs Deep Learning Approach*. Medium.Com. https://medium.com/@VK_Venkatkumar/background-removal-general-pixel-wise-vs-deep-learning-approach-4f919fe3dd3a
- Li, Y., & Zhang, D. 2023. Research and Application of Health Code Recognition Based on Paddle OCR under the Background of Epidemic Prevention and Control. *Journal of Artificial Intelligence Practice*, 6(1), 9–16.
- Nguyen, T. C., Nguyen, D. T., & Tran, Q. L. 2018. *Information extraction from id card via computer vision techniques*. VNU University of Engineering and Technology.
- Purohit, K. 2019. *Tutorial:Building A Custom OCR Using YOLO And Tesseract*. Medium.Com. <http://medium.com/saarthi-ai/how-to-build-your-own-ocr-a5bb91b622ba>.

- Rahyagara, A. 2018. *Tutorial Deteksi Objek Menggunakan YOLO (You Only Look Once)*. Medium. <https://medium.com/@andikirahyagara/tutorial-yolo-you-only-look-oncefor-absolutely-noob-c4d5f3751e1f>
- Reddy, B. S. 2021. *Step by Step YOLO Algorithm*. [Www.Madrasresearch.Org.](https://www.madrasresearch.org/post/step-by-step-yolo-algorithm)
<https://www.madrasresearch.org/post/step-by-step-yolo-algorithm>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. 2016. You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 779–788.
- Syahputra, A. 2019. *Pendeteksian Data E-KTP untuk Pencatatan Rekrutmen Anggota Partai Politik Menggunakan Algoritma Long Short Term Memory (LSTM) Berbasis Android* [Faculty of Computer Science and Information Technology].
<http://repositori.usu.ac.id/handle/123456789/24593>
- Szeliski, R.(2022. *Computer vision: algorithms and applications*. Springer Nature.
- Wang, C.Y., Bochkovskiy, A., & Liao, H.Y.M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7464–7475.



**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 : 1930/UN5.2.14.D/SK/SPB/2024**

**DEKAN FAKULTAS ILMU KOMPUTER
DAN TEKNOLOGI INFORMASI UNIVERSITAS SUMATERA UTARA**

- Membaca : Surat Permohonan Mahasiswa Fasilkom-TI USU tanggal 13 Juni 2024 perihal permohonan ujian skripsi:
- Nama : MUHAMMAD ELDWIN PASARIBU
 NIM : 181402086
 Program Studi : Sarjana (S-1) Teknologi Informasi
 Judul Skripsi : Ekstraksi Data E-KTP Untuk Pencatatan Peserta Vaksinasi Covid-19 Dengan Metode Yolo V7 dan Paddle Ocr Berbasis Android
- Memperhatikan : Bawa 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 : Bawa 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 : Membentuk dan mengangkat Tim Penguji Skripsi mahasiswa sebagai berikut:
- Ketua : Dedy Arisandi ST., M.Kom.
 NIP: 197908312009121002
- Sekretaris : Ade Sarah Huzaifah S.Kom., M.Kom
 NIP: 198506302018032001
- Anggota Penguji : Ulfy Andayani S.Kom., M.Kom
 NIP: 198604192015042004
- Anggota Penguji : Annisa Fadhillah Pulungan S.Kom, M.Kom
 NIP: 199308092020012001
- 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, 14 Juni 2024

Ditandatangani secara elektronik oleh:
Dekan



Maya Silvi Lydia
NIP 197401272002122001

