

**DETEKSI GERAKAN DALAM OLAHRAGA YOGA MENGGUNAKAN  
METODE *YOU ONLY LOOK ONCE* (YOLO)**

**SKRIPSI**

**FARIS AL FALAQ**

**191402129**



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

**MEDAN**

**2023**

DETEKSI GERAKAN DALAM OLAHRAGA YOGA MENGGUNAKAN  
METODE *YOU ONLY LOOK ONCE* (YOLO)

SKRIPSI

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

FARIS AL FALAQ

191402129



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

MEDAN

2023

## PERSETUJUAN

Judul : DETEKSI GERAKAN DALAM OLAHRGA YOGA  
MENGGUNAKAN METODE YOU ONLY LOOK  
ONCE (YOLO)

Kategori : SKRIPSI

Nama : FARIS AL FALAQ

Nomor Induk Mahasiswa : 191402129

Program Studi : SARJANA (S1) TEKNOLOGI INFORMASI

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI INFORMASI  
UNIVERSITAS SUMATERA UTARA

Medan, 10 Januari 2024

Komisi Pembimbing:

Pembimbing 2

Niskarto Zendrato S.Kom., M.Kom  
NIP. 198909192018051001

Pembimbing 1

Ainul Hizriadi S.Kom., M.Sc  
NIP. 198510272017061001

Diketahui/disetujui oleh

Program Studi S1 Teknologi Informasi

Ketua

Dedy Arisandi S.T., M.Kom.  
NIP. 197908312009121002

## **PERNYATAAN**

### **DETEKSI GERAKAN DALAM OLAHRAGA YOGA MENGGUNAKAN METODE YOU ONLY LOOK ONCE (YOLO)**

## **SKRIPSI**

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

Medan, 10 Januari 2024

Faris Al Falaq

191402129

## UCAPAN TERIMA KASIH

Puji syukur kepada Tuhan Yang Maha Esa karena atas berkat dan karunia-Nya penulis dapat mengerjakan dan menyelesaikan skripsi yang berjudul ini sebagai syarat dalam memperoleh gelar Sarjana Komputer di Program Studi S1 Teknologi Informasi Universitas Sumatera Utara. Selama pengerjaan skripsi ini juga penulis mendapat berbagai bantuan, bimbingan dan dorongan semangat dari berbagai pihak secara langsung maupun tidak langsung. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan terima kasih kepada:

1. Keluarga penulis, terutama kedua orangtua penulis ibu Sariyanti dan Bapak Rahmat Gustiar yang telah membesarkan saya serta memberikan kasih sayang, doa, masukan dan bimbingan dimulai dari pendidikan hingga menyelesaikan tugas akhir ini, dan juga buat adik adik penulis yaitu M. Fakhri Habibi dan Ikhsan Prayoga yang juga memberikan doa dan dukungan.
2. Bapak Dr. Muryanto Amin, S.Sos, M.Si. yang merupakan Rektor Universitas Sumatera Utara.
3. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
4. Bapak Dedy Arisandi S.T., M.Kom. selaku Ketua Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.
5. Bapak Ainul Hizriadi S.Kom., M.Sc selaku Dosen Pembimbing 1 dan Niskarto Zandrato S.Kom., M.Kom selaku Dosen Pembimbing 2 yang telah memberikan ilmu, saran dan kritik yang membangun, motivasi, dan waktunya selama proses pengerjaan skripsi ini.
6. Seluruh Dosen Program Studi S1 Teknologi Informasi yang telah memberikan ilmu yang bermanfaat bagi penulis selama masa perkuliahan
7. Pegawai dan staff Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara yang mendukung segala urusan administrasi dan akademik selama perkuliahan.
8. Teman teman yang sama mengambil metode YOLOv5, Brilliant Jonathan, Faisal

- Lazuardy, dan Huzaifah Muhammad Lais Lubis yang telah banyak membantu penulis dalam pengerjaan tugas akhir yang telah sama-sama berjuang untuk menyelesaikan tugas akhir.
9. Teman teman grup fore yaitu Irsyad, Nanda, Lais, Nathan, Faisal, Denaya, dan Fadhil yang sudah memberikan ilmu dan pengetahuan pada saat menyelesaikan tugas akhir ini dan menjadi tempat sharing selama proses pengerjaan tugas akhir.
  10. Bang Rhama Permadi Ahmad yang sudah banyak membantu saya dalam urusan administrasi dan membimbing saya selama masa perkuliahan saya.
  11. kepada teman-teman Kom C, teman-teman satu angkatan dan senior yang telah membantu penulis selama masa perkuliahan.

Penulis menyadari bahwa Skripsi ini masih belum sepenuhnya sempurna dan masih banyak kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang dapat membantu menyempurnakan untuk perbaikan di masa mendatang. Penulis juga berharap Skripsi ini dapat bermanfaat bagi semua pihak yang membacanya.

Medan, Desember 2023

Penulis

## ABSTRAK

Yoga adalah salah satu olahraga yang cukup populer di Indonesia, serta yoga juga merupakan salah satu olahraga yang memiliki banyak gerakan dan dapat dilakukan oleh semua rentang usia. Yoga memiliki gerakan yang unik. Jenis gerakan dalam olahraga yoga dianggap dapat memberi keseimbangan dan kesehatan bagi tubuh dan juga mental. Namun dalam prakteknya kesalahan dalam gerakan dapat menimbulkan cedera dengan intensitas ringan ataupun berat. Maka dari itu, dibutuhkan sebuah sistem yang dapat melakukan deteksi pada setiap gerakan yoga yang dilakukan selain memberi informasi nama gerakan juga dapat membuat pengguna lebih percaya diri dengan gerakan yang dilakukan karena akan mendapat informasi apakah gerakan yang dilakukan sudah sesuai. Dalam penelitian ini menggunakan metode *You Only Look Once* versi 5 yang digunakan untuk melakukan deteksi gerakan yoga secara *Real-Time* yang terdiri dari 5 kelas, yaitu Pose Bridge, Pose DownwardDog, Pose Shoulderstand, Pose Tree, dan Plank. Dataset yang digunakan terdiri dari 1.000 data yang dibagi menjadi 80% untuk data training dan 20% data validation. Data testing yang digunakan sebanyak 75 data. Penggunaan metode *YOLOv5* untuk mendeteksi gerakan dalam olahraga yoga mendapatkan akurasi sebesar 92%. Dengan hasil tersebut menunjukkan sistem yang dibuat dengan metode tersebut sudah baik dalam mendeteksi gerakan dalam yoga.

Kata Kunci: Yoga, *YOLOv5*, *Real-Time*

## ***DETECTION OF MOVEMENTS IN YOGA SPORTS USING THE YOU ONLY LOOK ONCE (YOLO) METHOD***

### ***ABSTRACT***

*Yoga is one of the popular sports in Indonesia, and it is also one of the sports that involves various movements suitable for all age groups. Yoga encompasses unique movements. The movements in yoga are believed to promote balance and health for the body as well as the mind. However, in practice, mistakes in these movements can result in injuries of varying intensity. Therefore, a system is needed to detect each yoga movement, providing information about the movement's name and ensuring users' confidence by verifying the accuracy of their poses. This research utilizes the You Only Look Once version 5 (YOLOv5) method for real-time detection of yoga movements, consisting of five classes: Bridge Pose, Downward Dog Pose, Shoulder Stand Pose, Tree Pose, and Plank. The dataset comprises 1,000 data points divided into 80% for training data and 20% for validation data. Additionally, 75 data points were used for testing purposes. The utilization of the YOLOv5 method in detecting yoga movements yielded an accuracy of 92%. This outcome indicates that the system developed with this method performs well in detecting yoga movements.*

*Keywords: Yoga, YOLOv5, Real-Time*



## DAFTAR ISI

<b>PERSETUJUAN</b>	Error! Bookmark not defined.
<b>PERNYATAAN</b>	<b>ii</b>
<b>UCAPAN TERIMA KASIH</b>	<b>iii</b>
<b>ABSTRAK</b>	<b>v</b>
<b><i>ABSTRACT</i></b>	<b>vi</b>
<b>DAFTAR ISI</b>	<b>vii</b>
<b>DAFTAR GAMBAR</b>	<b>ix</b>
<b>DAFTAR TABEL</b>	<b>xi</b>
<b>BAB 1 PENDAHULUAN</b>	<b>1</b>
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 Metodologi Penelitian	5
1.7 Sistematika Penulisan	6
<b>BAB 2 LANDASAN TEORI</b>	<b>8</b>
2.1 Yoga	8
2.2 Cedera Umum Yoga	12
2.3 Computer Vision	13
2.4 Human Pose Estimation	14
2.5 Object Detection	14
2.6 You Only Look Once (YOLO)	15
2.7 YOLO v4	16
2.8 YOLO v5	18
2.9 Matrix	19
2.10 Penelitian Terdahulu	22
2.11 Perbedaan Penelitian	25
<b>BAB 3 ANALISIS DAN PERANCANGAN SISTEM</b>	<b>26</b>
3.1 Data yang digunakan	26
3.2 Analisis Sistem	28

3.2.1 <i>Image Acquisition</i>	30
3.2.2 Image Pre-processing	30
3.2.3 Image Clasification	35
3.3 YOLOv5 Model	37
3.4 Clone Repository	37
3.5 Install Requirement	37
3.6 Menambah data.names dan data.yaml	37
3.7 Konfigurasi YOLOv5	38
3.8 Training Process	38
3.9 Testing Process	39
3.10 Perancangan Interface	40
3.10.1 Activity Diagram	40
3.10.2 Tampilan Home Screen	41
3.10.3 Tampilan Halaman Deteksi	41
3.10.4 Tampilan Halaman Petunjuk	42
3.10.5 Tampilan Halaman Tentang	42
<b>BAB 4 IMPLEMENTASI DAN PENGUJIAN</b>	<b>43</b>
4.1 Implementasi Sistem	43
4.1.1 Perangkat keras dan perangkat lunak	43
4.2 Implementasi Data	43
4.3 Pelatihan Sistem	45
4.4 Implementasi Perancangan Interface	49
4.4.1 Tampilan Halaman Home Screen	49
4.4.2 Tampilan Halaman Deteksi	50
4.4.3 Tampilan Halaman Petunjuk	50
4.4.5 Tampilan Halaman Tentang	51
4.5 Pengujian Sistem	51
<b>BAB 5 KESIMPULAN DAN SARAN</b>	<b>59</b>
5.1 Kesimpulan	59
5.2 Saran	59
<b>DAFTAR PUSTAKA</b>	<b>60</b>

## DAFTAR GAMBAR

<b>Gambar 2.1</b> Pose Bridge (Sumber: Kaggle.com)	9
<b>Gambar 2.2</b> Pose Downward Dog (Sumber: Kaggle.com)	9
<b>Gambar 2.3</b> Plank (Sumber: Pinterest.com)	10
<b>Gambar 2.4</b> Tree (Sumber: Pinterest.com)	11
<b>Gambar 2.5</b> Shoulderstand (Sumber: Pinterest.com)	11
<b>Gambar 2.6</b> 33 Pose landmarks (Kim et al., 2023)	14
<b>Gambar 2.7</b> Object Detection (Sumber: medium.com)	15
<b>Gambar 2.8</b> Desain Konsep YOLO (Redmon et al., 2016)	16
<b>Gambar 2.9</b> Perbandingan penggunaan Backbone (Mahasin & Dewi, 2022)	17
<b>Gambar 2.10</b> Ilustrasi PANet (Liu, n.d., 2018)	17
<b>Gambar 2.11</b> Arsitektur Umum YOLOv4 (Bochkovskiy et al., 2020)	18
<b>Gambar 2.12</b> Arsitektur YOLOv5 (XU et al,2021)	19
<b>Gambar 2.13</b> Persamaan IoU (Sumber: medium.com)	20
<b>Gambar 3.1</b> Arsitektur Umum	29
<b>Gambar 3.2</b> Hasil Labelling	31
<b>Gambar 3.3</b> Hasil Label Berformat .txt	31
<b>Gambar 3.4</b> Pembuatan Bounding Box Menggunakan Roboflow	32
<b>Gambar 3.5</b> Pembuatan Kelas Menggunakan Roboflow	32
<b>Gambar 3.6</b> Proses Augmentasi	34
<b>Gambar 3.7</b> Sebelum NMS	36
<b>Gambar 3.8</b> Sesudah NMS	37
<b>Gambar 3.9</b> Clone Repository	37
<b>Gambar 3.10</b> Data Names dan Data Yaml	37
<b>Gambar 3.11</b> Konfigurasi YOLOv5	38
<b>Gambar 3.12</b> Activity Diagram	40
<b>Gambar 3.13</b> Wireframe Home Screen	41
<b>Gambar 3.14</b> Wireframe Halaman Deteksi	41
<b>Gambar 3.15</b> Wireframe Halaman Petunjuk	42
<b>Gambar 3.16</b> Wireframe Halaman Tentang	42
<b>Gambar 4.1</b> Pose Bridge	44
<b>Gambar 4.2</b> Pose Downward Dog	44

<b>Gambar 4.3</b> Pose Shoulderstand	44
<b>Gambar 4.4</b> Pose Tree	45
<b>Gambar 4.5</b> Plank	45
<b>Gambar 4.6</b> Google Colab	46
<b>Gambar 4.7</b> Grafik loss	48
<b>Gambar 4.8</b> <i>Precision Recall Curve</i>	48
<b>Gambar 4.9</b> Home Screen	49
<b>Gambar 4.10</b> Halaman Deteksi	50
<b>Gambar 4.11</b> Halaman Petunjuk	50
<b>Gambar 4.12</b> Halaman Tentang	51

**DAFTAR TABEL**

<b>Tabel 2.1</b> Penelitian Terdahulu	23
<b>Tabel 3.1</b> Data yang digunakan	26
<b>Tabel 3.2</b> Jumlah Pembagian Kelas Data	27
<b>Tabel 4.1</b> Percobaan Training	46
<b>Tabel 4.2</b> Pengujian Sistem	51
<b>Tabel 4.3</b> Confusion Matrix	55
<b>Tabel 4.4</b> Hasil Pengujian	55
<b>Tabel 4.5</b> Hasil Perhitungan	57

## **BAB 1**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Pada dasarnya setiap orang ingin mempunyai hidup yang lebih berkualitas secara fisik, psikologis, dan spiritual. Manusia dasarnya memiliki motivasi utama untuk meningkatkan kualitas hidupnya, dimana hal tersebut sesuai dengan pendapat dari Maslow yang berpendapat bahwa setiap individu sejak lahir mempunyai kemauan yang aktif ke arah pertumbuhan atau ke arah aktualisasi potensi-potensi yang dimiliki oleh manusia. Pola hidup atau gaya hidup yang tidak sehat dapat disebabkan dari adanya kebiasaan yang dilakukan setiap harinya. Gaya hidup seseorang menentukan kualitas hidup dan kesehatan tubuhnya (Khairunisa, 2015). Dimana untuk meningkatkan kualitas hidup dan kesehatan tubuh setiap orang tentunya harus memperbaiki pola gaya hidupnya dengan melakukan olahraga efektif. Salah satu olahraga efektif yang dapat dilakukan untuk mencapai kualitas hidup yaitu dengan berlatih yoga (Somvir, 2008) sebagai salah satu olahraga yang tidak memerlukan biaya besar dan mudah dilakukan untuk memperoleh kebahagiaan serta kesehatan tubuh. Salah satu instruktur yoga bernama SK mengungkapkan bahwa masyarakat modern membutuhkan latihan yoga yang mana pada saat ini berada dalam kehidupan yang kompetitif dan penuh dengan tantangan. Dengan berada pada kehidupan yang kompetitif dan penuh tantangan membuat manusia fokus pada kehidupan dengan melakukan banyak hal tanpa tujuan atau biasa disebut *human doing* yang menyebabkan masyarakat modern sulit untuk menikmati hidup karena pikiran fokus untuk memenuhi target dan menjadi mudah stres.

Yoga dikenal sebagai sistem filosofi kehidupan masyarakat India kuno (Sindhu, 2007; Stiles, 2002). Pada saat ini yoga telah berkembang menjadi sistem kesehatan yang menyeluruh dan komprehensif. Istilah yoga sendiri berasal dari kata *Yuj dan Yoking* (Bahasa Sanskerta) yang bermakna penyatuan harmonis dari sesuatu yang terpisah (Sindhu, 2007; Stiles, 2002). Penyatuan secara harmonis mempunyai arti yaitu menyatukan antara tubuh, pikiran-perasaan dan aspek spiritual yang ada dalam diri manusia (Stiles, 2002). Yoga merupakan olahraga yang dapat dilakukan oleh semua kalangan dari anak-anak hingga lanjut usia. Yoga sangat mudah untuk dilakukan dan bukan merupakan latihan yang berkaitan dengan agama tertentu. Patanjali (Stiles, 2003) menyebutkan bahwa yoga bukan merupakan agama dan setiap orang sesungguhnya

melakukan yoga. Setiap orang melakukan gerakan yoga karena sesuai dengan tujuan yoga itu sendiri, yaitu untuk mendapat kebahagiaan yang sebenarnya didapatkan melalui proses kesadaran terhadap diri. Menyadari diri sendiri merupakan cara yang dapat dilakukan untuk memperoleh kebahagiaan. Dalam sutra patanjali disebutkan bahwa penyakit yang terdapat dalam tubuh dapat menjadi penghalang serta menjadi kebingungan pikiran apabila tidak melakukan upaya pencegahan yang maksimal. Sehingga, diharapkan melalui gerakan yoga diharapkan penyakit-penyakit yang ada dalam tubuh dapat di tangani.

Yoga merupakan salah satu olahraga yang memiliki banyak gerakan dan dapat dilakukan oleh semua rentang usia. Yoga memiliki gerakan yang unik (Yulinda, Purwaningsih and Sudarta, 2017). Jenis gerakan dalam olahraga yoga dianggap dapat memberi keseimbangan dan kesehatan bagi tubuh dan juga mental (Arisman et al., 2021). Yoga terus berkembang selama berabad-abad dan mendapat pengaruh dari beragam budaya, hingga kini terdapat berbagai jenis yoga. Setiap jenis yoga memiliki gerakan, rutinitas, serta manfaat yang berbeda-beda. Banyak orang memilih yoga untuk berbagai alasan seperti kebugaran, menghilangkan stres atau fisioterapi, karena yoga memberi manfaat dalam segi fisik dan mental.

Sementara itu yoga mungkin tampak seperti aktivitas olahraga yang berdampak rendah, namun para ahli memperingatkan bahwa tanpa adanya pelaksanaan pose atau gerakan yang tepat, yoga juga rentan akan cedera dan seringkali cedera akan berkembang dari waktu ke waktu, sebagai akibat dari praktik gerakan yang salah secara konsisten. Berdasarkan 2620 peserta yoga, dari para peserta yang mengalami cedera diperoleh persentase 28% mengalami cedera ringan dan 68% mengalami cedera sedang kemudian sebanyak 4% dari total sampel mengalami cedera parah, masa pemulihan cedera rata-rata memerlukan waktu selama 6 bulan, peserta lebih sering mengalami cedera di kelas yoga yang dibimbing atau bersama instruktur dengan persentase sebesar 61% daripada kelas yoga dengan praktik mandiri sebesar 39% (Wiese et al., 2019). Karena gerakan yoga dianggap lumayan sulit untuk pemula (Firdausi, 2018), oleh karena itu dibutuhkan teknologi yang dapat membantu mengenalkan gerakan tersebut agar gerakan yoga yang dilakukan benar sehingga meminimalisir risiko cedera. Untuk mendeteksi gerakan yoga secara real-time dapat menggunakan metode You Only Look Once (YOLO) yang dapat memberi informasi langsung kepada pengguna gerakan yoga yang dilakukan benar atau salah.

Beberapa penelitian mengenai deteksi gerakan dalam olahraga sebelumnya sudah pernah dilakukan, diantaranya adalah penelitian dengan menggunakan metode *Long Short-Term Memory* (LSTM) dan Mediapipe. Pengujian dilakukan dalam lokasi dan waktu yang sama sebanyak 5 kali. Frame diproses oleh LSTM untuk klasifikasi dan memprediksi gerakan yang sesuai yaitu: T-Pose, Warrior II Pose, Tree Pose. Berdasarkan pengujian menunjukkan bahwa pengenalan gerakan mendapatkan nilai akurasi sebesar 85% untuk Warrior II Pose, 100% untuk T-Pose, dan 80% untuk Tree Pose (Daniel Tanugraha et al., 2022).

Selanjutnya terdapat penelitian terkait deteksi dan koreksi gerakan dalam olahraga yoga yang menggunakan metode *K-Nearest Neighbors* dan *Posenet*. Penelitian tersebut mampu mendeteksi beberapa gerakan yoga yang berbeda dalam penelitian ini menggunakan lima gerakan yoga berbeda dan dataset yang digunakan berasal dari Kaggle dan sumber lainnya, yang terdiri dari total 1578 gambar. Penelitian ini menghasilkan nilai akurasi yang baik yaitu 98.51% dan penelitian ini bertujuan untuk membantu praktisi yoga terhindar dari cedera dan meningkatkan efektivitas dari gerakan yang dilakukan.

Berdasarkan latar belakang dan hasil penelitian-penelitian terdahulu tersebut, maka penulis mengajukan sebuah penelitian yang akan menghasilkan sebuah sistem dengan menerapkan metode *You Only Look Once* (YOLO) versi 5 yang akan membantu masyarakat khususnya bagi yang ingin melakukan olahraga yoga. Penelitian ini diberi judul **“Deteksi Gerakan Dalam Olahraga Yoga Menggunakan Metode You Only Look Once (YOLO)”**.

## 1.2 Rumusan Masalah

Resiko cedera dan perlunya biaya lebih serta butuh menyesuaikan waktu apabila menggunakan instruktur merupakan permasalahan dalam yoga, oleh karena itu perlunya sistem yang dapat mendeteksi gerakan yoga yang dilakukan benar atau salah sehingga tidak menimbulkan cedera dan bisa mendapatkan manfaat dari setiap gerakan yoga yang dilakukan, dimana sistem dapat mendeteksi setiap gerakan yoga secara real-time sehingga dapat membantu peran instruktur dan keunggulan lainnya yaitu praktisi dapat melakukan gerakan yoga kapan saja.



### 1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah membantu masyarakat untuk mengetahui gerakan yoga yang dilakukan benar atau salah sehingga dapat mengurangi risiko cedera. Gerakan akan di deteksi secara *realtime* dengan menerapkan metode *You Only Look Once* (YOLO) versi 5.

### 1.4 Batasan Masalah

Untuk mendapatkan maksud dan tujuan yang jelas serta mencegah meluasnya ruang lingkup permasalahan dari penelitian, maka diperlukan adanya batasan masalah , yaitu:

1. Dalam penelitian ini hanya untuk mengidentifikasi gerakan atau pose yang dilakukan dalam yoga merupakan gerakan yang benar atau salah. Gerakan yang akan digunakan yaitu Pose Bridge, Pose Downward Dog, Pose Shoulderstand, Pose Tree, dan Plank.
2. Menggunakan citra berekstensi .jpg atau .jpeg
3. Data citra gerakan atau pose dalam yoga menggunakan dataset yang berasal dari web kaggle dan kamera smartphone.
4. Jarak yang dapat dideteksi oleh sistem maksimal sebesar 200 cm dan minimal sebesar 150 cm.

### 1.5 Manfaat Penelitian

Manfaat yang didapatkan dari penelitian ini adalah:

1. Untuk membantu masyarakat khususnya yang ingin melakukan olahraga yoga agar gerakan atau pose yang dilakukan benar
2. Mengurangi resiko cedera yang didapatkan karena melakukan gerakan yang kurang tepat
3. Dapat digunakan sebagai bahan referensi atau sebagai bahan pembelajaran untuk penelitian-penelitian berikutnya agar dapat menggunakan *metode You Only Look Once* (YOLO) dalam mendeteksi objek yang diinginkan.

## 1.6 Metodologi Penelitian

Berikut tahapan-tahapan yang akan dilakukan dalam penelitian ini:

### 1) Studi Literatur

Pada tahapan ini merupakan tahap pembelajaran, pencarian serta pengumpulan informasi yang dibutuhkan dalam penelitian. Referensi terkait yang digunakan adalah tentang pengenalan gerakan olahraga, implementasi deteksi real time, dan penggunaan metode *You Only Look Once* yang didapat dari berbagai sumber seperti artikel, jurnal, skripsi, informasi di internet dan lainnya.

### 2) Analisis Permasalahan

Beberapa bahan referensi yang telah dikumpulkan pada tahapan sebelumnya dianalisis dalam tahap ini untuk bisa memperoleh pemahaman yang baik dan jelas tentang metode yang digunakan untuk penelitian yaitu *You Only Look Once* dalam menyelesaikan permasalahan deteksi gerakan dalam olahraga yoga menggunakan YOLO

### 3) Perancangan Sistem

Langkah-langkah yang dapat dilakukan dalam tahapan ini yaitu meliputi seluruh perancangan, seperti rancangan arsitektur umum, penentuan pengujian dan pelatihan dataset serta pembuatan rancangan antarmuka hingga sampai pada hasil akhir yang diperoleh berdasarkan tahapan hasil dari studi literatur dan analisis permasalahan sebelumnya

### 4) Implementasi

Pada tahap ini merupakan proses implementasi dari sistem yang telah dirancang dengan melakukan pengujian terhadap model *You Only Look Once* dimana setiap rancangan proses dibuat dalam susunan kode program agar menjadi sebuah sistem yang outputnya sesuai dengan tujuan penelitian

### 5) Pengujian Sistem

Pada bagian ini, merupakan hasil dari rancangan sistem yang telah diimplementasikan, setelah itu dilakukan pengujian agar mengetahui sistem yang

telah dibuat dapat mendeteksi gerakan dalam olahraga yoga dan dapat menentukan gerakan yang dilakukan benar atau salah dengan baik serta sesuai dengan tujuan.

#### 6) Dokumentasi dan Penyusunan Laporan

Dalam tahap ini merupakan tahapan peneliti melakukan dokumentasi dan menyusun laporan dari hasil keseluruhan yang telah dikerjakan diantaranya seperti analisis, perancangan sistem, implementasi, dan pengujian sistem untuk pemaparan hasil akhir dari penelitian.

### 1.7 Sistematika Penulisan

Skripsi ini dirangkai dengan lima bagian utama dalam tahapan penulisan, yaitu:

#### **BAB 1: Pendahuluan**

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

#### **BAB 2: Landasan Teori**

Bab dua dalam penelitian ini berisikan tentang teori yang berkaitan dengan yoga, cedera umum yoga, *computer vision*, *human pose estimation*, metode yolo versi terdahulu, dan penjelasan tentang metode yang diterapkan dalam penelitian ini yaitu You Only Look Once (YOLO) versi 5.

#### **Bab 3: Analisis dan Perancangan Sistem**

Bab tiga terdapat alur dari perancangan sistem yang dilakukan dalam penelitian ini dimana terdapat bagian arsitektur umum, dataset yang digunakan, image processing, proses testing dan desain antar muka dalam penelitian ini.

**Bab 4: Implementasi dan Pengujian**

Bab empat terdiri dari hasil implementasi dari model dan sistem yang telah didapatkan dari tahap analisis dan perancangan sistem pada sistem yang telah dibangun.

**Bab 5: Kesimpulan dan Saran**

Dalam bab ini terdapat kesimpulan serta saran yang didapatkan setelah melakukan penelitian dimana ini dapat berguna untuk pengembangan penelitian yang akan dilakukan selanjutnya agar menjadi lebih baik.

## BAB 2 LANDASAN TEORI

### 2.1 Yoga

Yoga diperkirakan sudah ada sejak dari 7000 tahun yang lalu, yang digunakan sebagai cara untuk meningkatkan kekuatan fisik dan psikis. Yoga sendiri tumbuh pertama kali di India dan menyebar ke setiap negara. Yoga berasal dari kata “yuj” yang mempunyai arti menghubungkan serta yoga itu sendiri merupakan pengendalian aktivitas pikiran dan juga sebagai sarana penyatuan antara roh pribadi dengan roh tertinggi (Gunarta, 2017). Selain itu Vivekananda seorang pemimpin spiritual yang memiliki pengaruh besar dari ajaran Vedanta dan Yoga menyatakan bahwa dalam latihan olahraga yoga harus memenuhi tiga syarat, yaitu: 1) melepaskan semua kesenangan duniawi, 2) memiliki keinginan yang kuat dan benar, 3) menjaga pikiran bergerak, menjaga indera, mengalihkan pikiran ke dalam, menderita tanpa mengeluh, menguatkan pikiran akan ide tertentu dan berfikir selalu tentang sifat. Gerakan dalam olahraga yoga dapat dilakukan dalam berbagai posisi seperti tidur, duduk, maupun dengan berdiri. Dalam yoga terdapat delapan tahapan yang ditempuh dalam melaksanakan yoga, salah satunya yaitu *Asana* (sikap tubuh) yang digunakan dalam penelitian ini dimana setiap gerakan harus stabil dan dalam kondisi nyaman (Madja, 2018). Beberapa gerakan yoga berikut mudah untuk dilakukan serta memiliki manfaat yang baik untuk fisik maupun mental. Dibalik banyaknya manfaat yang dapat diperoleh dalam melaksanakan yoga semua itu bisa didapatkan apabila mempraktikannya dengan benar, terdapat beberapa hal yang harus diperhatikan dalam setiap melakukan gerakan karena apabila melakukan gerakan yang salah akan berpotensi cedera.. Berikut beberapa gerakan atau pose yang rentan membuat cedera, yaitu:

#### 2.1.1 Pose Bridge

Pose Bridge (*Setu Bandha Sarvangasana*) dalam yoga. Seperti namanya, pose bridge membuat tubuh Anda terlihat seperti jembatan. Gerakan ini memungkinkan tulang belakang memanjang ke arah pinggul dan paha belakang Anda (Yogajournal, 2023). Seperti pose yoga lainnya, *Setu Bandha Sarvangasana* dapat memberikan berbagai manfaat kesehatan fisik dan mental. Salah satunya pose bridge yang dikenal luas sebagai gerakan yang bisa mengatasi sakit tulang belakang. Contoh gerakan bridge terdapat pada gambar 2.1.



**Gambar 2.1** Pose Bridge (Sumber: Kaggle.com)

#### 2.1.2 Pose Downward Dog

Salah satu pose yoga yang paling dikenal yaitu pose downward dog (Adho Mukha Svanasana) yaitu gerakan yang berguna untuk penguatan dan peregangan seluruh bagian tubuh (Debbimurphy, 2016). Meskipun terlihat mudah namun gerakan ini sering salah dilakukan sehingga menyebabkan cedera. Pose ini biasanya dilakukan menyerupai anjing yang menunduk. Gerakan ini membentuk tubuh seperti huruf A. Pose ini mempunyai beberapa manfaat diantaranya yaitu untuk yang sering mempunyai keluhan nyeri di bagian leher atau punggung, meningkatkan sirkulasi darah, memperbaiki postur tubuh, dan melatih otot kaki. Contoh gerakan dapat dilihat pada gambar 2.2.



**Gambar 2.2** Pose Downward Dog (Sumber: Kaggle.com)

### 2.1.3 Plank

Plank adalah salah satu nama gerakan yoga untuk pemula yang cukup mendasar dan populer. Awali dengan posisi badan yang menghadap ke bawah. Selanjutnya, angkat badan anda dari lantai dengan menopang berat badan pada tangan dan ujung kaki di atas matras. Walau tampak mudah, mempertahankan posisi plank selama 1 menit sebenarnya sudah cukup bagus bagi seorang pemula. Oleh karena itu, selain melatih fisik, plank juga membantu Anda dalam membangun ketahanan mental dan fokus (Rathikarani et al., 2022). Plank juga mampu menambah stamina tubuh sekaligus menguatkan otot-otot perut dan bahu. Gerakan plank dapat dilihat pada gambar 2.3.



**Gambar 2.3** Plank (Sumber: Pinterest.com)

### 2.1.4 Tree

Pose pohon/tree pose (atau vrksasana dalam bahasa Sansekerta) adalah postur berdiri dan keseimbangan klasik yang ditemukan di banyak gaya yoga yang berbeda. Gerakan ini dapat diberikan untuk anak usia dini dan ramah bagi pemula, pose pohon benar-benar menawarkan kombinasi dari semua tentang yoga: keseimbangan, fleksibilitas, kekuatan, dan konsentrasi (Harliana, 2019), tetapi juga memungkinkan untuk memodifikasi gerakan ini untuk tingkat keahlian apa pun.

Tree pose juga membantu meningkatkan fokus, kesadaran, dan konsentrasi. Selain bagus untuk pikiran, juga bagus untuk tubuh, menguatkan tulang belakang, kaki, dan pergelangan kaki. Ini juga merupakan latihan kekuatan panggul, cocok untuk siapa saja yang berurusan dengan pinggul yang kaku atau kencang. Gerakan Tree terdapat pada gambar 2.4 dibawah ini.



**Gambar 2.4** Tree (Sumber: Pinterest.com)

#### 2.1.5 Shoulderstand

Sarvangasana atau Shoulderstand merupakan pose dalam yoga dimana posisi seluruh tubuh seimbang diatas bahu. Gerakan ini juga merupakan bagian dari rangkaian yoga Padha Sadhana. Kata ‘Sarv’ berarti semua, ‘anga’ berarti bagian dari tubuh, dan ‘asana’ berarti sikap tubuh. Berdasarkan namanya, Sarvangasana memengaruhi semua fungsi bagian tubuh. Asana ini sangat bermanfaat dalam menjaga kesehatan mental serta fisik dan juga sering disebut sebagai ‘Ratu Asana’ (Himani Bisht, 2023).

Gerakan ini mempunyai banyak manfaat selain baik untuk kesehatan mental juga mempunyai beberapa manfaat untuk kesehatan fisik diantaranya memperkuat lengan dan bahu serta menjaga tulang belakang tetap fleksibel, meregangkan otot jantung dengan mengembalikan lebih banyak darah vena ke jantung, dapat membantu penyembuhan dari beberapa penyakit seperti sembelit, gangguan pencernaan dan varises. Contoh gerakan shoulderstand terdapat pada gambar 2.5



**Gambar 2.5** Shoulderstand (Sumber: Pinterest.com)



## 2.2 Cedera Umum Yoga

Yoga tidak hanya bermanfaat untuk fisik saja melainkan dapat mengatasi masalah mental juga, sementara itu banyak orang menganggap yoga merupakan aktivitas yang berdampak rendah. Namun para ahli memperingatkan bahwa tanpa pelaksanaan pose atau gerakan yang sesuai, para praktisi yoga juga akan rentan mengalami cedera. Seringkali, cedera yang didapatkan berkembang seiring waktu dikarenakan melakukan praktik postur tubuh yang salah secara konsisten. Peregangan berlebihan pada persendian, ketegangan pada otot tendon merupakan cedera umum yang ditemui dalam olahraga yoga, menurut spesialis ortopedi pada Raffles Hospital Singapore. Seperti halnya melakukan aktivitas fisik apapun, pendekatan dalam yoga yang paling aman yaitu dengan cara mempeleajari pose atau gerakan dengan benar dan tidak melakukan gerakan secara berlebihan. Saat berlatih yoga ada beberapa area khusus yang rentan mengalami cedera sehingga harus lebih diperhatikan (Dr.Lim Yeow Wai, 2022), diantaranya sebagai berikut:

### 1) Leher

Menempatkan terlalu banyak berat badan pada leher saat melakukan gerakan headstand, plow dan shoulder stand. Gerakan tersebut juga bisa melukai leher saat menekuknya terlalu dalam tanpa adanya dukungan dalam pose yoga tertentu. Agar terhindar dari cedera, disarankan untuk tidak memaksakan diri dengan melakukan rentang gerakan ekstrem di luar kemampuan individual, seperti yang dijelaskan dalam sumber Yoga London. Penting juga untuk menjaga agar leher tetap sejajar dengan tulang belakang. Jika sudah terjadi rasa nyeri, disarankan untuk menggunakan kompres es pada area yang terkena dan mengonsumsi kunyit sebagai cara untuk meredakan rasa sakit.

### 2) Bahu

Sebanyak 19 persen dari total cedera dalam praktik yoga adalah cedera pada bahu, seperti yang dilaporkan dalam data yang dipublikasikan oleh Yoga Anatomy. Tekanan pada bahu dapat mengakibatkan kehilangan stabilitas dan berpotensi merobek korset bahu atau manset rotator. Untuk memperbaiki kondisi ini, latihan korektif dapat dilakukan guna menyeimbangkan otot-otot di sekitar sendi tersebut.

### 3) Pergelangan Tangan

Pergelangan tangan rentan terhadap cedera, terutama saat melakukan gerakan yoga tertentu seperti *plank*, *side plank*, *downward facing dog*. Untuk mencegah cedera pada pergelangan tangan, tetapkan telapak tangan datar di atas matras. Ini karena banyak saraf pergelangan

tangan tidak tahan terhadap beban berlebihan. Selain itu, memberikan kompres air hangat pada pergelangan tangan sebelum melakukan yoga juga dapat membantu mencegah cedera.

#### 4) Punggung

Cedera pada punggung sering terjadi pada praktik yoga, terutama saat melakukan gerakan *forward fold (uttanasana)* dan *downward-facing dog*. Beberapa instruktur yoga berpendapat bahwa cedera ini disebabkan oleh lenturan tulang belakang yang tidak sesuai selama gerakan tersebut. Oleh karena itu, disarankan untuk menghindari potensi cedera dengan bangkit dari gerakan forward fold dengan sedikit menekuk lutut dan menggunakan dukungan penopang saat mengangkat tubuh.

#### 5) Lutut

Lutut juga merupakan bagian yang sering mengalami cedera. Beberapa pose yang dapat menyebabkan cedera pada lutut termasuk pose camel, hero, dan lotus. Data statistik dari Yoga Anatomy menunjukkan bahwa hampir 21 persen dari cedera yoga terjadi pada bagian lutut. Lutut yang lemah dapat menjadi perhatian serius, karena dapat menyebabkan kerusakan pada meniskus, yang selanjutnya dapat memicu ketidakstabilan, pembengkakan, nyeri, dan penguncian sendi pada lutut dalam posisi tertentu. Oleh karena itu, praktisi yoga disarankan untuk berhati-hati dan waspada terhadap potensi cedera pada lutut, terutama saat melakukan pose yang memerlukan fleksibilitas ekstra.

#### 6) Hamstring

Individu yang melakukan gerakan yoga memiliki risiko terkena tendonitis hamstring, suatu kondisi di mana otot hamstring meregang secara berlebihan sehingga dapat merusak tendon yang menghubungkannya. Cedera ini mungkin timbul ketika melipat kaki ke depan tanpa menggunakan perut dan paha depan sebagai dukungan yang stabil. Cedera hamstring cenderung terjadi secara tiba-tiba, dimulai dari robekan kecil pada tendon, dan dapat berkembang seiring waktu. Gejala cedera ini mungkin berulang dan dapat menyebabkan rasa sakit yang persisten jika tidak diatasi dengan baik.

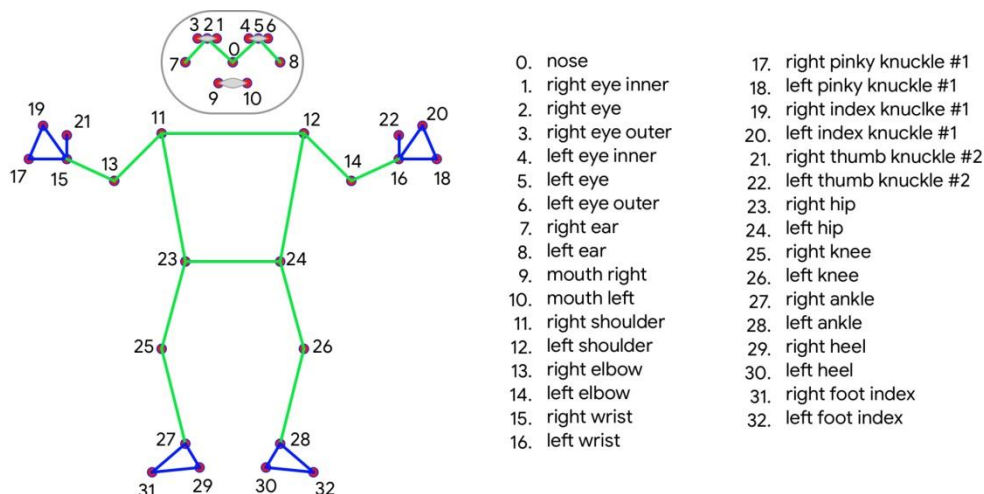
### 2.3 Computer Vision

*Computer Vision* sering disingkat dengan CV adalah bidang studi yang dibuat untuk membantu komputer melihat dan memahami makna dalam bentuk gambar digital seperti foto atau video secara otomatis (Dawson-Howe, 2014). Dalam beberapa dekade terakhir, metode *Deep Learning* menunjukkan hasil untuk mengatasi beberapa tantangan dalam computer vision seperti klasifikasi gambar, deteksi objek, dan pengenalan wajah. Tujuan

dari CV yaitu untuk memahami konten gambar digital dan juga membuat komputer memiliki kemampuan layaknya penglihatan manusia yang baik dalam hal melakukan penangkapan atau perekaman gambar, analisa gambar, ataupun dalam pemahaman gambar.

## 2.4 Human Pose Estimation

*Human pose estimation* merupakan topik penelitian yang mempunyai tujuan untuk melakukan estimasi letak masing-masing sendi yang akan dihubungkan sehingga dari proses tersebut dapat membentuk kerangka manusia yang dapat diambil dari gambar dan video (Steven & Purbowo, 2022). *Human pose estimation* dapat diaplikasikan untuk mendeteksi aktivitas yang dilakukan manusia, pelacakann manusia, dan dunia hiburan seperti *motion capture* animasi dan perfilman. Namun penggunaan Human pose estimation memiliki beberapa tantangan yang paling sering dijumpai seperti fleksibilitas tubuh yang berbeda sehingga bentuk tubuh dapat bergerak menjadi suatu pose yang kompleks atau jarang, bentuk tubuh yang beragam termasuk dengan pakaian, dan lingkungan yang kompleks yang membuat susah membedakan antara foreground dan background. Bagian pose landmarks dapat dilihat pada gambar 2.6.



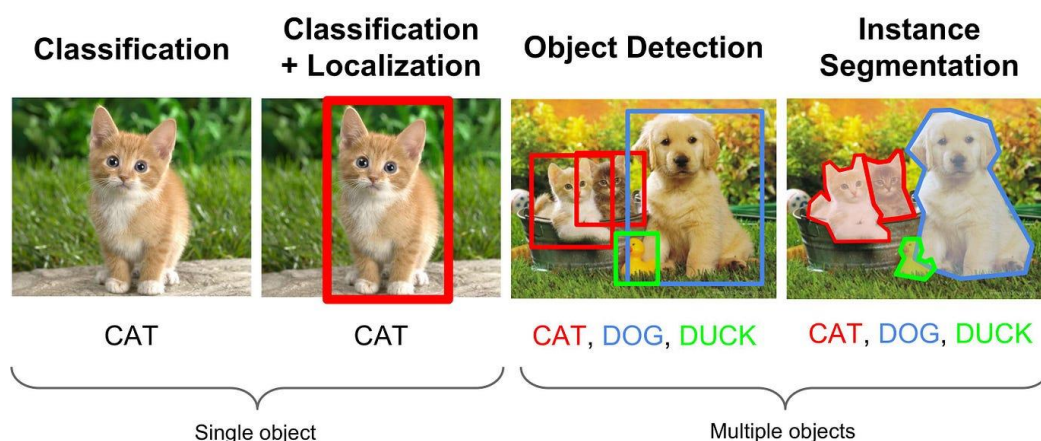
**Gambar 2.6** 33 Pose landmarks (Kim et al., 2023)

## 2.5 Object Detection

*Object detection* merupakan salah satu hal mendasar dalam dunia *computer vision*. Object detection adalah cara sistem untuk menentukan apakah itu merupakan sebuah objek dan menghasilkan lokasi dengan menggunakan bounding box pada area yang telah terdeteksi

sebagai objek. Single object detection merupakan proses mendeteksi hanya satu objek saja, sedangkan multiple object detection merupakan tahapan melakukan deteksi pada banyak objek. Dalam melakukan pendeteksian terhadap suatu objek juga harus memperhatikan pencahayaan, pose, skala (Pathak et al., 2018).

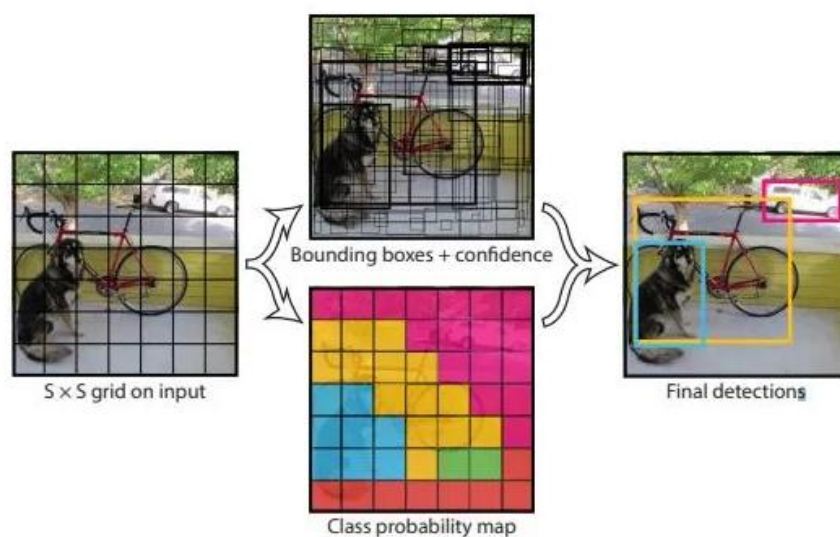
*Real Time Object Detection* adalah deteksi objek yang diambil secara langsung dengan menggunakan alat seperti web camera atau smartphone untuk dijadikan input. Kemudian data yang telah didapatkan secara langsung tadi akan diproses pada tahapan klasifikasi. *Frame per second* sangat diperhatikan dalam hal ini dikarenakan objek yang akan dilakukan deteksi dilakukan secara realtime maka realtime object detection mementingkan kecepatan. Pada gambar 2.7 terdapat penjelasan mengenai object detection.



**Gambar 2.7** Object Detection (Sumber: medium.com)

## 2.6 You Only Look Once (YOLO)

*YOLO* merupakan algoritma yang dirancang untuk mendeteksi objek secara real time. Sistem pendeteksian dilakukan dengan menggunakan repurpose classifier atau localizer untuk melakukan deteksi. Model diterapkan pada beberapa lokasi dan skala pada citra. Daerah dengan citra yang mempunyai nilai atau score tertinggi akan dianggap sebagai pendeteksian. *YOLO* menggunakan pendekatan *Convolutional Neural Network (CNN)* untuk melakukan prediksi dan mendeteksi objek yang ada pada sebuah citra. Dengan jaringan ini akan membagi menjadi beberapa wilayah dan akan melakukan prediksi setiap kotak pembatas dan probabilitas untuk setiap wilayah. Selanjutnya kotak-kotak pembatas atau *grid* akan dibandingkan dengan setiap probabilitas yang akan diprediksi (Redmon et al., 2016). Konsep YOLO terdapat pada gambar 2.8 dibawah ini.



**Gambar 2.8** Desain Konsep YOLO (Redmon et al., 2016)

## 2.7 YOLO v4

Algoritma YOLOv4 diterbitkan 2 tahun setelah versi terakhir dimana kali ini dirancang oleh Alexey Bochkovskiy. Tujuan utama membuat YOLOv4 yaitu merancang operasi yang cepat untuk melakukan deteksi objek dalam sistem produksi serta melakukan optimalisasi untuk komputasi paralel dimana perancang mengharapkan siapa saja yang menggunakan YOLOv4 dapat melatih dan melakukan pengujian sehingga dapat mencapai hasil objek deteksi yang real time, berkualitas tinggi, dan juga dapat menyakinkan pengguna (Bochkovskiy et al., 2020).

YOLOv4 ini merupakan *one-stage detector* yang terdiri atas *backbone*, *neck*, dan *head*, dimana terdapat 3 opsi *backbone* yang dapat digunakan yaitu *CSPDarknet53*, *CSPResNet50*, dan *EfficientNet-B3* dimana berdasarkan beberapa penelitian bahwa CSPDarknet53 memiliki kemampuan yang lebih optimal dibandingkan dengan ops yang lainnya. Selanjutnya pada tahap *Neck* yang digunakan yaitu *PANet (Path Aggregation Network)* yang telah dilakukan modifikasi dengan tujuan mendapatkan nilai akurasi yang lebih baik (Bochkovskiy et al., 2020). Berikutnya yaitu *head* yang digunakan pada versi ini sama dengan versi yang sebelumnya. Berikut penjelasan Arsitektur Umum YOLOv4:

### 1. *Backbone* : CSPDarknet53

Backbone network merupakan tahapan untuk melakukan feature extraction. Dalam tahap ini menggunakan backbone CSPDarknet53 dimana backbone ini merupakan

pengembangan dari backbone pada YOLOv3 yaitu Darknet53 yang telah menambahkan fitur yang bernama Cross Stage Partial (CSP) yang mengintegrasikan feature maps dari mulai tahap awal hingga tahap akhir dalam jaringan. Implementasi CSP mengurangi komputasi sebesar 20% sehingga hal tersebut yang membuat mengungguli arsitektur backbone yang lainnya (Mahasin & Dewi, 2022). Berikut perbandingan antara CSPResNeXt-50 dan EfficientNet-B0, CSPDarkNet53 menghasilkan nilai mAP (Mean Average Precision) yang lebih baik. Adapun hasil perbandingan terdapat pada gambar 2.9

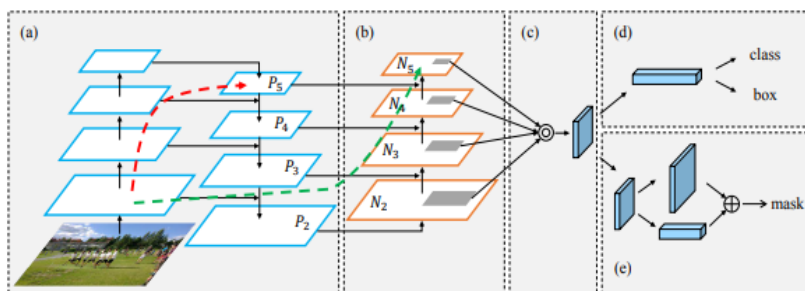
Overall Comparison

Backbone	Precisi-on (%)	Recall (%)	F1-Score (%)	mAP (%)
CSPDarkNet53	63,33%	79,50%	70,40%	<b>83,41%</b>
CSPResNetXt-50	<b>75,60%</b>	<b>81,10%</b>	<b>78%</b>	81,00%
EfficientNet-B0	63,80%	80,40%	71,50%	81,76%

**Gambar 2.9** Perbandingan penggunaan Backbone (Mahasin & Dewi, 2022)

## 2. Neck : PANet

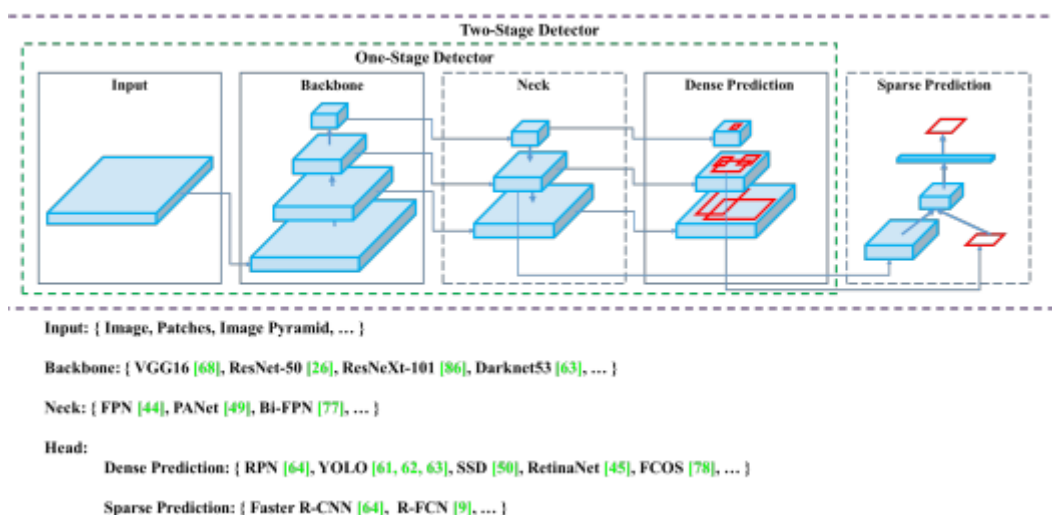
PANet merupakan sebuah pendekatan baru yang hadir dalam YOLOv4, dimana merupakan perpanjangan dari Mask-RCNN yang membuat beberapa perbaikan dari para pendahulunya. PANet meningkatkan kemampuan dengan menambahkan jalur *bottom-up* untuk menambah jalur *top-down* pada *Feature Pyramid Network* (FPN), serta juga menggunakan Adaptive Feature Pooling untuk menerima informasi dari semua level yang ada, menggunakan PANet dapat meningkatkan penyebaran informasi melalui pipelines karena PANet menggabungkan fitur dari semua level dan bahkan dapat memperpendek jarak antara lapisan paling bawah dengan lapisan paling atas (Liu, n.d., 2018). Berikut merupakan ilustrasi PANet pada gambar 2.10.



**Gambar 2.10** Ilustrasi PANet (Liu, n.d., 2018)

### 3. Head

Tahap Head yang dilakukan sama dengan proses yang digunakan pada YOLOv3 dan YOLOv4. Head sendiri berfungsi sebagai jaringan yang memiliki tanggung jawab sebagai detektor untuk bounding box. Selain itu head sendiri merupakan bagian dari jaringan yang digunakan untuk mengklasifikasikan dan mengembalikan anchor box, pemilihan posisi frame dalam algoritma deteksi target secara langsung memengaruhi efek deteksi yang dilakukan (B. Yan et al., 2021). Dalam tahap ini terdapat regresi loss function diantaranya fungsi IOU dimana berfungsi untuk menjadi acuan rasio kesalahan antara kotak yang diprediksi dengan kotak yang sebenarnya. Arsitektur umum YOLOv4 terdapat pada gambar 2.11.



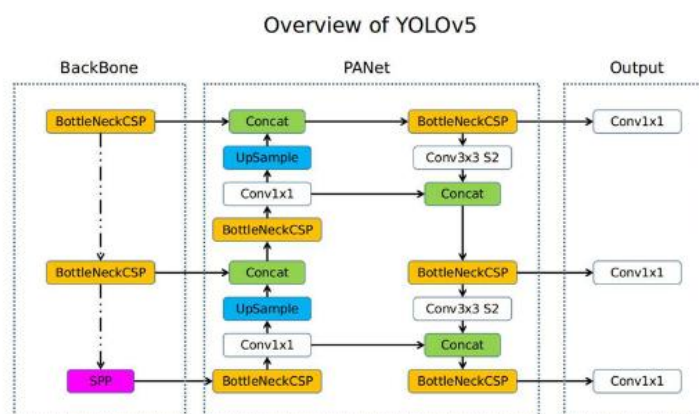
**Gambar 2.11** Arsitektur Umum YOLOv4 (Bochkovski et al., 2020)

## 2.8 YOLO v5

Pada 29 Mei 2020 Glenn Jocher membuat repository di github dengan nama *YOLOv5* yang tidak berisi kode model apapun, setelah itu pada 9 Juni 2020 dia menambahkan pesan commit ke implementasi *YOLOv3* miliknya dengan membuat judul “YOLOv5 greetings”. Implementasi *YOLOv5* milik Jocher ini berbeda dengan beberapa dari rilis yang dilakukan sebelumnya dimana kali ini Jocher tidak menerbitkan makalah mengenai pembahasan *YOLOv5* tetapi hanya merilis implementasi *open source* pada github. Ultralytics yang merupakan perusahaan milik Jocher merubah versi YOLO menjadi sebuah framework *pytorch* yang ditulis dalam bahasa *python*. *YOLOv5* menyediakan 4 versi model yang dapat digunakan yaitu *Yolov5-small*, *Yolov5-medium*, *Yolov5-large*, *Yolov5-extraLarge* (Herwanto, 2023).



Dalam Arsitektur *YOLOv5* mempunyai bentuk yang sama dengan arsitektur yang ada dalam versi sebelumnya. Versi *YOLOv5* juga mempunyai backbone, neck, dan head yang sama seperti *YOLOv4*. Perbedaan *YOLOv4* dan *YOLOv5* terletak hanya pada bahasa yang digunakan yaitu memakai bahasa *Python* dan framework *pytorch* (Taşyürek & Öztürk, 2023). *Pytorch* merupakan pustaka *deep learning* yang dioptimalkan berdasarkan Python dan Torch. Library ini biasa digunakan untuk aplikasi yang menggunakan GPU dan CPU. Dengan menggunakan library ini ilmuwan, pengembang, neural network debugger bisa menjalankan serta menguji bagian dari kode secara real time. Hal ini membuat *YOLOv5* lebih leluasa untuk digunakan baik untuk model kecil hingga model yang besar. Perbedaan juga terdapat pada saat konfigurasi dimana versi sebelumnya menggunakan *.cfg* sedangkan pada *YOLOv5* menggunakan file *.yaml* untuk melakukan konfigurasi. Berikut Arsitektur *YOLOv5* pada gambar 2.12.



**Gambar 2.12** Arsitektur YOLOv5 (XU et al,2021)

## 2.9 Matrix

Pada saat melakukan pengujian pada sebuah metode dalam machine learning memerlukan beberapa acuan untuk melakukan pengujian model machine learning. Acuan yang dapat digunakan yaitu Iou Threshold dan confusion matrix.

### *IoU Threshold*

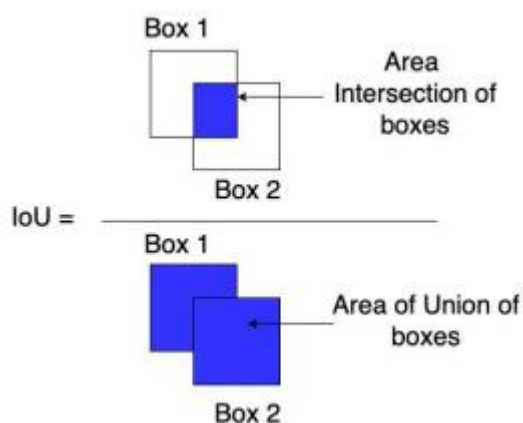
*Intersection Over Union* (IoU) adalah nilai yang didapat berdasarkan statistik kesamaan dan keragaman set sampel yang bertujuan untuk melakukan evaluasi area tumpang tindih (area yang beririsan) antara dua bounding box, yaitu antara bounding box hasil prediksi model dan bounding box data asli. Iou mempunyai nilai minimum yang digunakan agar



hasil yang diprediksi dianggap bernilai True Positive yang disebut dengan *IoU threshold*. Untuk mendapatkan nilai dari IoU threshold, dapat dilakukan dengan cara menggunakan persamaan berikut.

$$\text{IoU} = \frac{\text{Area of Intersection of two boxes}}{\text{Area of Union of two boxes}}$$

Nilai IoU Threshold mempengaruhi nilai mAP, dimana nilai IoU antara 0 sampai 1. Jika menggunakan nilai IoU threshold yang terlalu rendah maka model memiliki jumlah false positive yang besar. Sebaliknya, jika menggunakan nilai IoU Threshold yang lebih tinggi akan menyebabkan pengurangan nilai True Positive yang sangat signifikan ketika proses training (J. Yan et al., 2019). Berikut persamaan IoU pada gambar 2.13.



**Gambar 2.13** Persamaan IoU (Sumber: medium.com)

### *Confusion Matrix*

*Confusion matrix* merupakan metode dalam machine learning yang dapat digunakan untuk mengukur kinerja model yang telah dibuat. *Confusion matrix* pada dasarnya memberi informasi berupa perbandingan antara hasil klasifikasi yang dilakukan oleh model dengan hasil klasifikasi yang sebenarnya (Normawati & Prayogi, 2021). Terdapat empat kombinasi yang berbeda dalam *confusion matrix* yang terdiri dari *true positive*, *true negative*, *false positive*, *false negative*.

1. *True Positive* adalah nilai dimana data yang sebenarnya bernilai positif dan data yang diprediksi bernilai positif.
2. *True Negative* adalah nilai dimana data yang sebenarnya bernilai negatif dan data yang diprediksi bernilai negatif.

3. *False Positive* adalah nilai dimana data yang sebenarnya bernilai negatif dan data yang diprediksi bernilai positif.
4. *False Negative* adalah nilai dimana data yang sebenarnya bernilai positif dan data yang diprediksi bernilai negatif.

Setelah didapatkannya nilai keempat kombinasi data tersebut maka dapat dilakukan pengukuran performa dengan beberapa unsur penting diantaranya yaitu:

#### 1. *Accuracy*

*Accuracy* merupakan nilai yang menunjukkan kedekatan antara nilai prediksi dengan nilai aktual. Untuk mendapatkan nilai *accuracy*, lihat persamaan (1) .

$$accuracy = \frac{TP+TN}{Total\ Data}$$

#### 2. *Recall*

*Recall* menunjukkan keberhasilan model dalam menemukan kembali informasi. Maka *recall* merupakan perbandingan antara prediksi benar positif dengan keseluruhan data bernilai benar positif. Untuk mendapatkan nilai *recall*, lihat persamaan (2) .

$$recall = \frac{TP}{TP+FN}$$

#### 3. *Precision*

*Precision* menunjukkan tingkat keakuratan data yang digunakan dengan hasil prediksi yang didapatkan dari model. Berdasarkan semua kelas positif yang diprediksi dengan benar, berapa data yang menghasilkan nilai yang benar-benar positif. Untuk mendapatkan nilai *precision*, lihat persamaan (3) .

$$precision = \frac{TP}{FP+TP}$$

#### 4. *F1 Score*

F1 Score menggambarkan rasio antara rata-rata *precision* dan *recall* yang dibobotkan. Nilai terbaik dari F1 Score adalah 1 dan terburuknya yaitu 0. Untuk mendapatkan nilai F1 Score, lihat persamaan (4) .

$$F1\ score = \frac{2 * Recall * Precision}{Recall + Precision}$$

## 2.10 Penelitian Terdahulu

Terdapat beberapa penelitian yang berkaitan dengan deteksi gerakan dalam olahraga, diantaranya adalah penelitian yang dilakukan oleh (Daniel Tanugraha et al., 2022) menggunakan metode *Long Short Term Memory (LSTM)* dan *Mediapipe* yang mana dalam penelitian tersebut membahas mengenai pengenalan gerakan dalam olahraga yoga menggunakan tiga gerakan yoga seperti *T-Pose*, *Warrior II Pose*, dan *Tree Pose*. Dalam penelitian tersebut memperoleh nilai akurasi dari training sebesar 91% dan loss sebesar 29%. Setelah itu hasil yang didapatkan dari pengenalan gerakan yang dilakukan diantaranya gerakan *T-Pose*, *Warrior II Pose*, dan *Tree Pose* berturut-turut mendapatkan hasil 100%, 85%, dan 80%.

Penelitian berikutnya dilakukan (Wang, 2022) berjudul *A Deep Learning Algorithm for Special Action Recognition of Football* yang membahas mengenai sistem yang dapat digunakan untuk pengenalan tindakan, identifikasi objek, dan komponen pengenalan wajah dalam olahraga sepakbola dengan menggunakan metode *Support Vector Machine (SVM)* dan *Artificial Neural Network (ANN)*. Peneliti mengungkapkan bahwa setiap komponen yang digunakan dalam olahraga sepakbola tersebut berfungsi dengan baik. Akurasi yang didapatkan dari deteksi wajah pemain sepakbola mendapat akurasi sebesar 98%.

Selanjutnya ada penelitian dari (Raptis et al., 2011) tentang melakukan klasifikasi secara real time gerakan tari menggunakan animasi skeleton. Dalam penelitian ini terdiri dari 28 kelas gerakan, ratusan contoh gerakan direkam memakai alat *XBOX Kinect* dan dilakukan dengan puluhan subjek untuk setiap kelas gerakan yang ada. Dari proses klasifikasi tersebut mendapat nilai akurasi sebesar 96,9% untuk rekaman gerakan yang mempunyai durasi selama 4 detik.

Penelitian selanjutnya berkaitan dengan penggunaan *YOLOv5* dalam mendeteksi bahasa isyarat dalam pengenalan huruf hijaiyah, penelitian ini dilakukan oleh (Iskandar Mulyana et al., 2022) dengan menggunakan dataset yang berasal dari *kaggle* yang diunggah oleh Ammar Sayyed dengan jumlah citra sebanyak 4.247 citra. Berdasarkan penelitian yang dilakukan dengan melakukan epoch sebanyak 150 kali mendapatkan nilai akurasi sebesar 0.95% dimana dengan akurasi tersebut objek akan terdeteksi dengan baik.

Terdapat juga penelitian oleh (Iskandar Mulyana & Rofik, 2022) dengan menggunakan metode *YOLOv5* dan menggunakan dataset private yang dikumpulkan secara pribadi. Dataset yang digunakan untuk proses training oleh peneliti memakai data yang didownload

dari internet dan dari video yang diambil pribadi. Dataset yang didapatkan oleh peneliti sebanyak 1332 citra yang terdiri dari data train sebanyak 666 citra dan data val 666. Akurasi dari penelitian ini berada pada nilai 0.900 atau 90%.

Penelitian berikutnya berkaitan dengan deteksi pose dalam yoga yang dilakukan oleh (Gajbhiye et al., 2022) yang berjudul AI Human Pose Estimation: Yoga Pose Detection and Correction. Penelitian dilakukan dengan menggunakan dua metode yaitu Support Vector Machine (SVM) dan Convolutional Neural Network (CNN) yang mendapatkan nilai test akurasi yang baik yaitu 0.93 dan 0.98 dengan menggunakan dataset yang terdiri dari lima gerakan yoga yang berbeda diantaranya Padmasana (Lotus Pose), Shavasana (Corpse Pose), Tadasana (Mountain Pose), Trikonasana (Triangle pose), and Vrikshasana (Tree pose). Walaupun dengan angka nilai test akurasi yang tinggi sistem masih terdapat kesalahan dalam melakukan klasifikasi gerakan contohnya pada gerakan tadasana dari 17.695 frame untuk gerakan tadasana, 6982 salah diklasifikasikan sebagai trikonasana.

Penelitian dengan judul Yoga Pose Detection and Correction using Posenet and KNN yang dilakukan oleh (Shah et al., 2021) menggunakan metode K-Nearest Neighbors (KNN) dan Posenet yang mendapatkan nilai akurasi sebesar 98.51% yang menggunakan dataset yang berasal dari kaggle dengan data yang terdiri dari lima gerakan yoga diantaranya Utkata Konasana (goddess), Tadasana (mountain), Phalakasana (plank), Vrksasana (tree), Virabhadrasana III (warrior 3) dengan total gambar sebanyak 1578.

**Tabel 2.1** Penelitian Terdahulu

NO	Penulis	Judul	Tahun	Keterangan
1.	Filbert Daniel Tanugraha, Heri Pratikno, Musayyanah, dan Weny Indah Kusumawati	Pengenalan Gerakan Olahraga Berbasis (Long Short- Term Memory) Menggunakan Mediapipe	2022	Penelitian ini menggunakan metode Long Short Term Memory (LSTM) dan mediapipe. Berdasarkan pengujian, menunjukkan bahwa pengenalan gerakan mendapatkan nilai akurasi sebesar 85% untuk Warrior II Pose, 100% untuk T-Pose, dan 80% untuk Tree Pose
2.	Sheng Wang	A Deep Learning Algorithm for Special Action		Penelitian yang dilakukan menggunakan pendekatan Support Vector Machine (SVM) dan

		Recognition of Football	2022	Artificial Neural Network (ANN). Sistem ini menggunakan deep learning untuk pengenalan tindakan, identifikasi objek, dan komponen pengenalan wajah. Penelitian mengungkapkan bahwa komponen yang berbeda dari sistem ini berfungsi dengan baik
3.	Michalis Raptis, Darko Kirovski, Hugues Hoppe	Real-Time Classification of Dance Gestures from Skeleton Animation	2011	Dalam penelitian ini terdiri dari 28 kelas gerakan, ratusan contoh gerakan direkam memakai alat <i>XBOX Kinect</i> dan dilakukan dengan puluhan subjek untuk setiap kelas gerakan yang ada. Kinerja klasifikasi untuk seluruh (N = 120 sampel) rata-rata 96,9% untuk rekaman gerakan yang mempunyai durasi selama 4 detik. Kinerja terendah yang diamati adalah 81,1% .
4.	Dadang Iskandar Mulyana, Muhammad Faizal Lazuardi, Mesra Betty Yel	Deteksi Bahasa Isyarat Dalam Pengenalan Huruf Hijaiyah Dengan Metode YOLOV5	2022	Berdasarkan penelitian dan hasil data train, diperoleh hasil yang cukup tinggi. Penelitian ini dilakukan dengan menggunakan dataset yang berasal dari <i>kaggle</i> yang diunggah oleh Ammar Sayyed dengan jumlah citra sebanyak 4.247 citra. Dimana pada penelitian ini di train sebanyak 150 epoch untuk mendapatkan tingkat akurasi yang tinggi. Hasilnya nilai akurasi menunjukan angka 0.95 yang artinya mendekati angka 1 yaitu sekitar 95%, dimana di akurasi tersebut objek dapat terdeteksi dengan baik.
5.	Dadang Iskandar Mulyana, M Ainur Rofi	Implementasi Deteksi Real Time Klasifikasi Jenis Kendaraan Di Indonesia Menggunakan Metode YOLOV5	2022	Dataset yang digunakan untuk proses training oleh peneliti memakai data yang didownload dari internet dan dari video yang diambil pribadi. Dataset yang didapatkan oleh peneliti sebanyak 1332 citra yang terdiri dari data train sebanyak 666 citra dan data val 666. Penelitian pendeteksian jenis kendaraan ini mendapatkan nilai akurasi yang cukup tinggi

				yaitu nilai akurasinya adalah 0,900 atau 90%. Pendeteksian jenis kendaraan menggunakan metode YOLOV5 berjalan dengan lancar dan nilai akurasinya pun cukup tinggi.
6	Rutuja Gajbhiye, Snehal Jarag, Pooja Gaikwad, Shweta Koparde	AI Human Pose Estimation: Yoga Pose Detection and Correction	2022	Dataset yang digunakan dalam penelitian ini terdiri dari beberapa gerakan yoga, yaitu Padmasana (Lotus Pose), Shavasana (Corpse Pose), Tadasana (Mountain Pose), Trikonasana (Triangle pose), and Vrikshasana (Tree pose) diambil dari jarak 4 meter dan frame rate berada pada 30 fps. Penelitian menggunakan 2 metode yaitu Support Vector Machine (SVM) dan Convolutional Neural Network (CNN), kedua metode tersebut mendapat nilai test akurasi yang tinggi yaitu berada pada nilai 0.93 dan 0.98
7.	Varsha Bhosale, Pranjal Nandeshwar, Abhishek Bale, Janmesh Sankhe	Yoga Pose Detection and Correction using Posenet and KNN	2022	Penelitian ini menggunakan metode <i>K-Nearest Neighbors</i> dan <i>posenet</i> mendapatkan nilai akurasi yang mengesankan yaitu 98.51% dengan menggunakan dataset yang berasal dari kaggle dengan data lima gerakan yoga yang berbeda

### 2.11 Perbedaan Penelitian

Terdapat beberapa perbedaan dengan penelitian yang sudah dilakukan sebelumnya yaitu berkaitan dengan tempat penelitian, waktu penelitian, dan batasan yang digunakan. Penulis menggunakan metode *You Only Look Once Versi 5 (YOLOv5)* sebagai metode yang digunakan dalam mendeteksi objek yang diteliti berupa gerakan atau pose dalam olahraga yoga, dimana hal ini berbeda dengan penelitian sebelumnya yang dilakukan oleh (Daniel Tanugraha et al., 2022) dan (Shah et al., 2021) yang menggunakan metode LSTM dan KNN dan mendeteksi pose gerakan yang berbeda dalam penelitian mereka. Penulis membuat sistem berupa aplikasi berbasis website dan sistem dapat digunakan secara realtime sehingga dapat memudahkan pengguna dalam penggunaan sistem dengan hasil yang dapat diketahui saat itu juga.

### BAB 3



#### ANALISIS DAN PERANCANGAN SISTEM

Dalam bab ini meliputi pembahasan mengenai analisis dan perancangan sistem yang dibuat untuk melakukan deteksi gerakan dalam olahraga yoga. Terdapat beberapa tahapan yang dilakukan yaitu diantaranya pengumpulan data, preprocessing, proses training dan testing dengan metode YOLOv5, pendeteksian menerapkan metode You Only Look Once (YOLO) versi 5.

#### 3.1 Data yang digunakan

Data yang digunakan dalam studi ini adalah data gerakan olahraga yoga yang berasal dari dataset kaggle dan data yang diambil menggunakan kamera smartphone. Citra yang digunakan terdiri dari lima gerakan dan data yang digunakan merupakan gambar berformat .jpg yang berjumlah 1000 gambar. Tabel menunjukkan beberapa contoh gambar yang digunakan dalam penelitian ini.

**Tabel 3.1** Data yang digunakan

No.	Gambar	Jenis Gerakan
1.		Pose Bridge
2.		Pose Downward Dog

3.		Pose Shoulderstand
4.		Pose Tree
5.		Plank

Data gerakan yoga yang digunakan terdiri dari 5 gerakan yaitu Pose Bridge (Setu Bandha Sarvangasana), Pose DownwardDog (Adho mukha svanasana), Pose Shoulderstand (Salamba Sarvangasana), Pose Tree (Vrikshasana), Pose Plank (Phalakasana). Data tersebut dibagi menjadi 2 bagian yaitu data training dan data validasi yang akan digunakan pada proses pelatihan dataset. Data testing yang digunakan dalam penelitian ini akan menggunakan citra yang berasal langsung dari peneliti untuk menguji sistem apakah mampu mendeteksi objek. Pada tabel 3.2 menampilkan jumlah data yang digunakan pada masing-masing gerakan yoga.

**Tabel 3.2** Jumlah Pembagian Kelas Data

Jenis Gerakan	Pose Bridge	Pose DownwardDog	Pose Shoulderstand	Pose Tree	Pose Plank
Jumlah	200	200	200	200	200



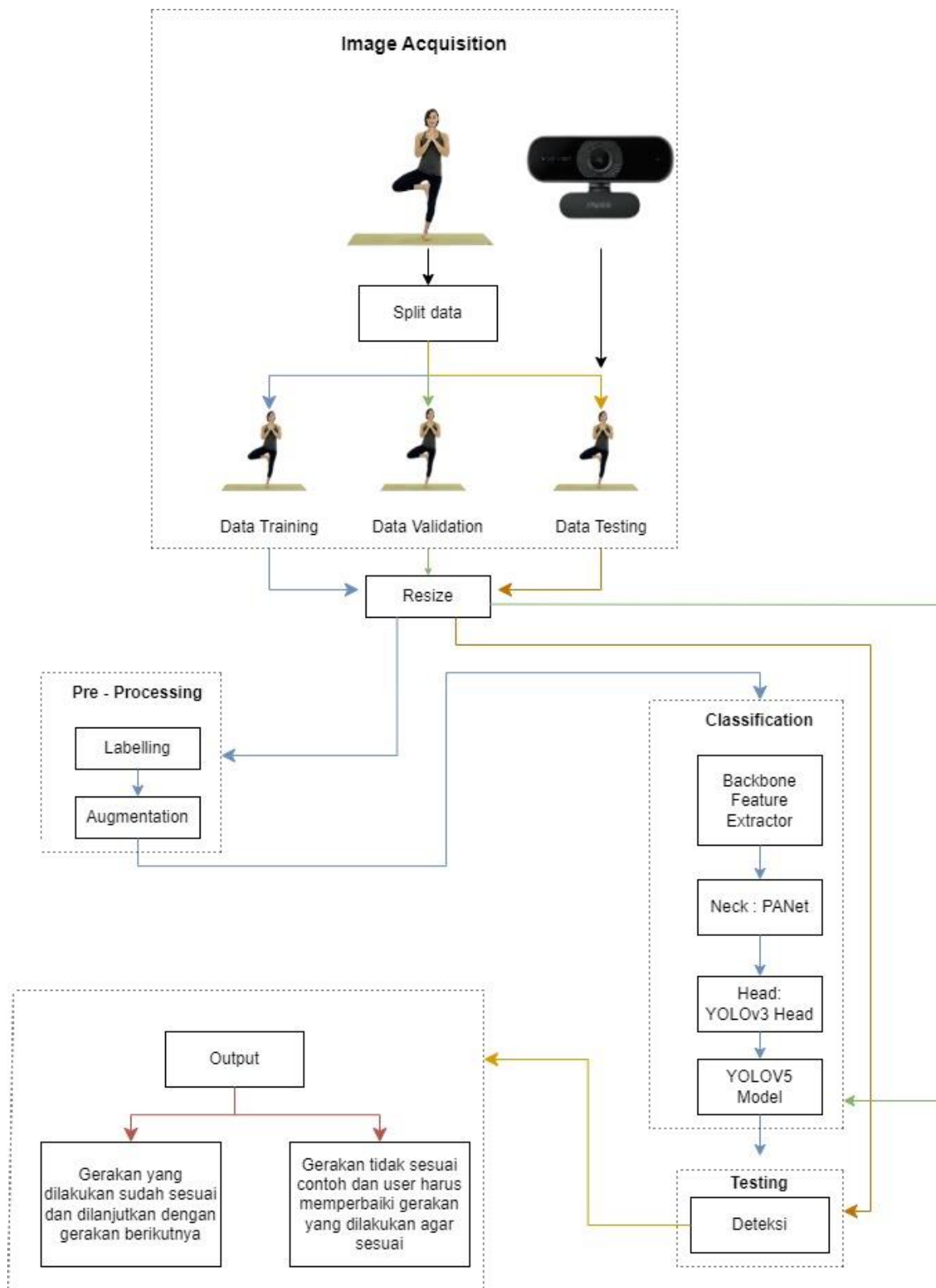
Total data sebanyak 1000 data citra yang dilatih menjadi data train dan data validation dengan masing data dibagi sebanyak 80% data training dan 20% data validation. Pembagian ini dilakukan agar melatih algoritma untuk dapat mengenali ciri-ciri citra dan disebut dengan data latih. Data testing yang akan digunakan sebanyak 100 data *real*.

### 3.2 Analisis Sistem

Dari Gambar 3.1 menunjukkan sistem yang digunakan dalam penelitian ini, penelitian ini dimulai dengan melakukan pengumpulan data dari gabungan antara dataset kaggle dan data yang diambil dari kamera smartphone yang kemudian dibagi menjadi 5 bagian kelas gerakan yoga.

Data yang digunakan kemudian lanjut ketahap *pre-processing* dimana tahap pertama data melalui tahap labelling. Selanjutnya data yang sudah dilabel akan melalui tahapan augmentasi dimana proses yang dilakukan untuk menambah kuantitas pada data.

Tahapan terakhir yang dilakukan adalah melakukan klasifikasi yang dilakukan menggunakan *You Only Look Once* versi 5. Kemudian penyelesaian dari tahapan proses ini, menghasilkan output untuk dapat mengidentifikasi lima jenis gerakan yoga yaitu Pose Bridge, Pose DownwardDog, Pose Shoulderstand, Pose Tree, dan Plank. Gambar 3.1 merupakan arsitektur pada penelitian ini.



**Gambar 3.1** Arsitektur Umum

### 3.2.1 Image Acquisition

Dalam tahapan ini, melakukan pengumpulan data citra gambar yang akan digunakan yaitu citra gambar pose gerakan dalam olahraga yoga yang akan digunakan dalam penelitian ini. Data gambar yang digunakan terdiri dari lima kelas yaitu Pose Bridge, Pose DownwardDog, Pose Shoulderstand, Pose Tree, dan Plank. Proses pengambilan data citra ini dilakukan mengambil dari dataset kaggle dan pengambilan secara langsung menggunakan kamera smartphone dengan format jpg.

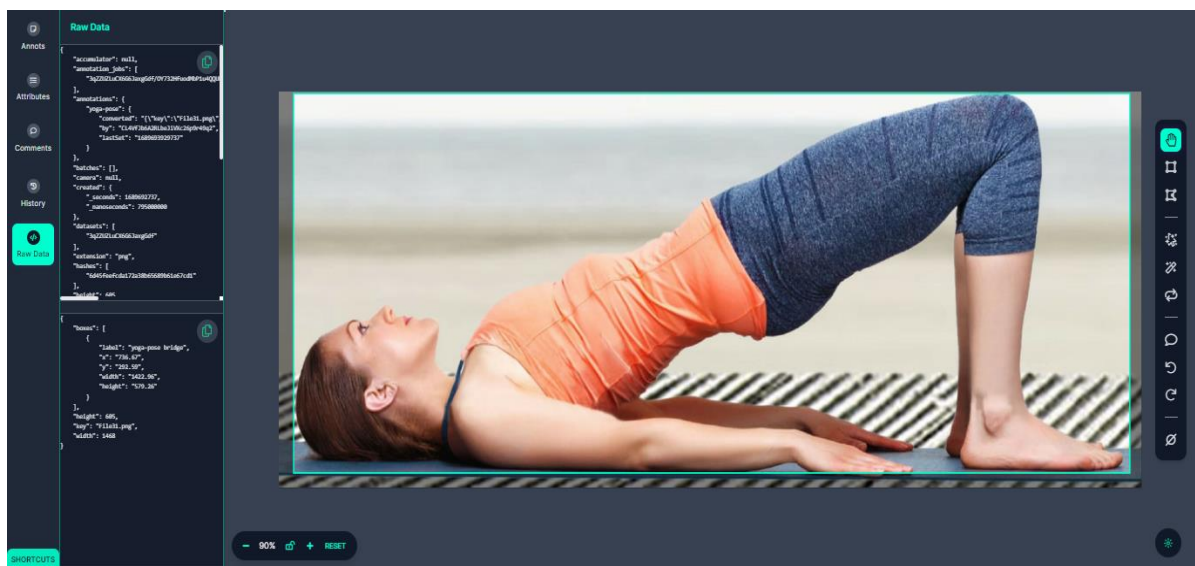
### 3.2.2 Image Pre-processing

Setelah data citra dikumpulkan dan dibagi, data akan di lanjutkan pada tahap pre-processing. Tahapan yang digunakan dalam penelitian ini adalah image labelling, resizing, dan augmentation. Berikut proses yang dilakukan pada tahapan pre-processing, yaitu:

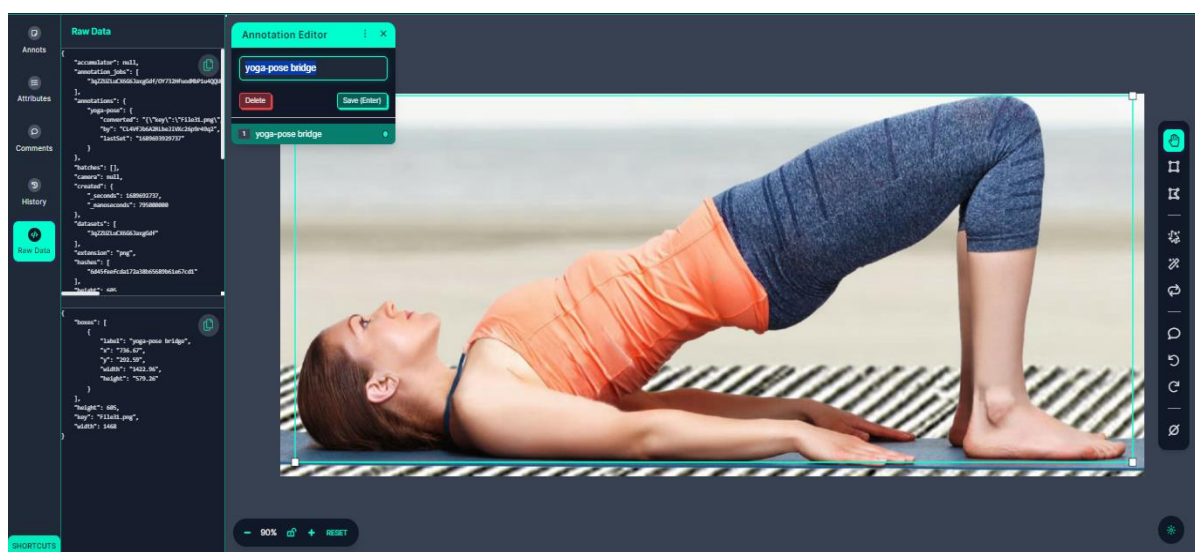
#### 3.2.2.1 Image Labelling

Langkah pertama data citra akan diberikan label yaitu pemberian kelas atau anotasi pada citra yang terdiri sebanyak lima kelas yang terdiri dari Pose Bridge, Pose DownwardDog, Pose Shoulderstand, Pose Tree, dan Plank. Tahapan selanjutnya melakukan pembuatan bounding box untuk mengenali objek yang dideteksi. Proses labelling ini menggunakan tools dari website *roboflow*. Hasil dari proses labelling ini kemudian akan disimpan kedalam bentuk folder dan akan digenerate menjadi bentuk *.txt* yang akan digunakan dalam proses training data. Hasil proses yang dilakukan menggunakan roboflow ditunjukkan pada Gambar 3.2 dan hasil label dalam bentuk *.txt* yang ditunjukkan pada Gambar 3.3, Gambar 3.4 merupakan proses pembuatan bounding box menggunakan roboflow dan Gambar 3.5 merupakan pembuatan kelas menggunakan roboflow.

### Gambar 3.3 Hasil Label Berformat .txt



**Gambar 3.4** Pembuatan Bounding Box Menggunakan Roboflow



**Gambar 3.5** Pembuatan Kelas Menggunakan Roboflow

### 3.2.2.2 Resizing

Tahap ini berfungsi untuk mengubah ukuran citra, mengubah ukuran data merupakan salah satu hal yang penting untuk dilakukan dalam tahap *preprocessing*. Dengan melakukan resizing maka akan membuat model melakukan training lebih cepat. Dengan melakukan resizing maka akan

menyamakan ukuran citra yang kita gunakan sebagai input. Resizing mengubah ukuran gambar menjadi 640x640 pixels yang menggunakan website roboflow.

### 3.2.2.3 Augmentation

Tahapan berikutnya dilanjutkan dengan melakukan augmentasi dimana data yang telah selesai dilabel selanjutnya dilakukan tahap augmentasi yaitu proses penambahan data citra dengan cara melakukan perubahan posisi gambar diantaranya yaitu flip dan horizontal. Hal ini dilakukan agar menambah kuantitas pada data sehingga data yang digunakan lebih banyak, dimana sebelum dilakukan augmentasi jumlah data sebanyak 1000 gambar dan 1601 gambar setelah augmentasi. Dengan melakukan augmentasi data akan menghasilkan performa atau akurasi yang lebih baik (Shorten et al., 2021). Augmentasi dilakukan dengan memanfaatkan web roboflow. Adapun pseudocode augmentasi yang dilakukan sebagai berikut:

```
def augment_data(image):

    image <---
    random_flip_horizontal(image)
    image <---
    normalize_image(image)

    return image

def
    random_flip_horizon
    tal(image):return
    flipped_image
```

```
def
    normalize_im

    age (image) :

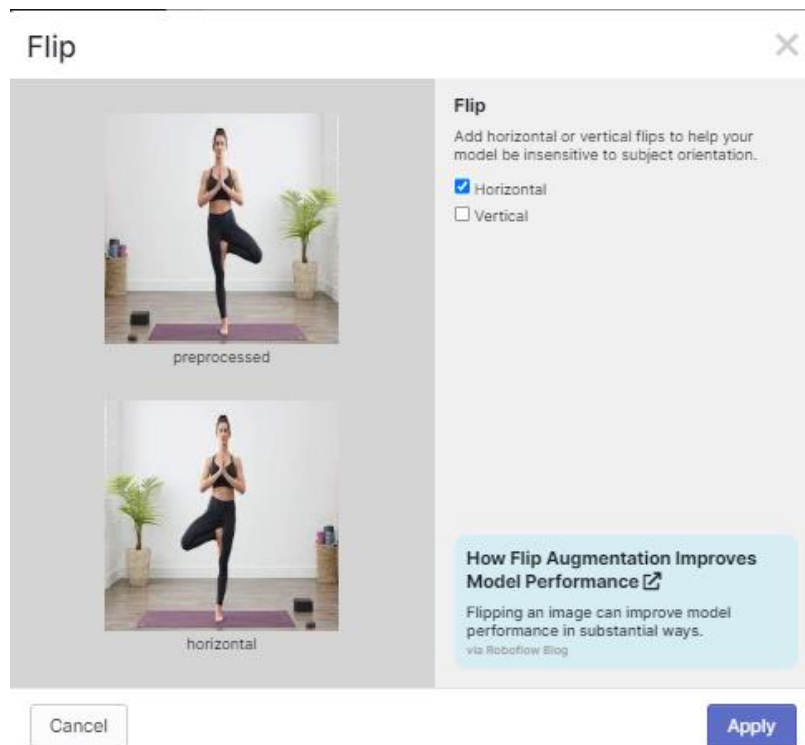
    return

    normalized_i

    mage

    image <---
    load_image (image_path)
    augmented_image <---
    augment_data (image)
```

Adapun proses dan hasil augmentasi pada gambar 3.6



**Gambar 3.6** Proses Augmentasi

### 3.2.3 Image Classification

Tahapan berikutnya yaitu untuk klasifikasi data citra menggunakan algoritma YOLOv5. Algoritma ini merupakan tergolong one stage object detector yang dimana mampu melakukan identifikasi objek dengan berdasarkan bounding box sekaligus juga melakukan klasifikasi objek berdasarkan kemungkinan kelas yang telah dilakukan. Berikut merupakan gambaran umum mengenai cara kerja YOLOv5 pada pendeteksian objek sebagai berikut:

- **Input Gambar**  
*YOLOv5* akan mengambil input gambar dengan bentuk serta ukuran yang tetap untuk digunakan sebagai input
- **Backbone**  
 Gambar input akan melalui *Convolutional Neural Network (CNN)* yang mendalam, kemudian akan dilakukan ekstraksi sekumpulan peta fitur dari gambar
- **Neck**  
 Peta fitur jaringan backbone akan diproses oleh jaringan di neck yang menggabungkan dan melengkapinya untuk menangkap informasi yang lebih kontekstual dan semantic
- **Head**  
 Peta fitur yang disempurnakan kemudian akan melewati jaringan head yang akan mendeteksi objek dengan prediksi kotak pembatas dan probabilitas kelas yang ada pada serangkaian kotak *anchor* yang mencakup skala serta rasio aspek yang berbeda
- **Deteksi Objek**  
 Merupakan hasil akhir dari algoritma yang digunakan dimana terdapat serangkain kotak pembatas dan probabilitas kelas untuk objek yang telah terdeteksi dalam gambar yang diinput
- **Loss Function**  
 Penggunaan GIOU Loss yaitu untuk menghasilkan kotak deteksi dengan cara meningkatkan ukuran bounding box dan secara perlahan bergerak menuju ground truth. Dengan melakukan beberapa iterasi untuk menyelesaikan masalah overlap antara bounding box dengan ground truth agar mendapatkan hasil kotak deteksi yang akurat.



- Non Maximum Suppression

Tahapan akhir dari melakukan proses pendeteksian citra ini yaitu tahap Non Maximum Suppression dimana tahap ini algoritma akan menghapus bounding box yang berlebihan dan hanya menampilkan satu bounding box yang terbaik. Pada gambar 3.7 dan gambar 3.8 menunjukkan perbedaan antara sebelum dilakukan NMS dan setelah NMS.



**Gambar 3.7** Sebelum NMS



**Gambar 3.8** Sesudah NMS

- Learned Model

Model merupakan hasil dari proses training yang dilakukan, model yang telah belajar akan berformat .pt yang menggunakan library PyTorch. Dalam proses training *YOLOv5* akan menghasilkan beberapa model yang terbentuk yang terdiri dari model terakhir (last) dan model terbaik (best). Model (best) merupakan model yang akan digunakan. Learned model berdasarkan hasil output dari pelatihan data, yang merupakan hasil pembelajaran oleh mesin dari proses training yang telah dilakukan, kemudian model akan digunakan pada tahap pengujian (testing).

### 3.3 YOLOv5 Model

Terdapat beberapa langkah yang dilakukan dan ditentukan untuk membuat model YOLOv5 dan kemudian dilanjutkan proses training yaitu dengan menentukan weight yang digunakan, serta melakukan konfigurasi model YOLOv5 serta beberapa parameter lainnya.

### 3.4 Clone Repository

Tahapan ini dilakukan untuk melakukan kloning *YOLO repository* github. Untuk melakukan clone *repository YOLO* dapat dilakukan dengan cara yang ada pada Gambar 3.9

```
# clone YOLOv5 repository
!git clone https://github.com/ultralytics/yolov5 # clone repo
%cd /content/drive/MyDrive/yoga-yolov5-baru/yolov5
!git reset --hard 064365d8683fd002e9ad789c1e91fa3d021b44f0
```

**Gambar 3.9** Clone Repository

### 3.5 Install Requirement

Tahapan ini untuk melakukan instalasi packages yang digunakan untuk penggunaan YOLOv5. Pytorch dan Cuda diinstal untuk memulai proses deteksi memanfaatkan YOLOv5.

### 3.6 Menambah data.names dan data.yaml

Daftar nama kelas yang akan terdeteksi dalam sistem berada pada file data.names, sementara data.yaml meliputi direktori folder yang digunakan untuk pelatihan, validasi, nama kelas yang tersedia, serta jumlah kelas yang akan dipakai. Hal ini terdapat dalam Gambar 3.10

```
# this is the YAML file Roboflow wrote for us that we're loading into this notebook with our data
%cat {dataset.location}/data.yaml

names:
  - yoga-pose bridge
  - yoga-pose downward
  - yoga-pose plank
  - yoga-pose shoulderstand
  - yoga-pose tree
nc: 5
roboflow:
  license: Public Domain
  project: yolov5-yoga
  url: https://universe.roboflow.com/faris-Sylyv/yolov5-yoga/dataset/6
  version: 6
  workspace: faris-Sylyv
```

**Gambar 3.10** Data Names dan Data Yaml

### 3.7 Konfigurasi YOLOv5

Untuk melakukan konfigurasi model *YOLOv5* terdapat dua hal yang perlu diperhatikan yaitu jumlah kelas yang digunakan (classes) dan pengaturan filter untuk melakukan konfigurasi *YOLOv5*. Jumlah kelas tersebut menunjukkan berapa banyak kelas yang digunakan, sedangkan penggunaan pengaturan filter harus disesuaikan dengan nilai kelas yang digunakan. Berikut merupakan langkah untuk menentukan nilai filter dalam *YOLOv5*. Konfigurasi *YOLOv5* terdapat pada gambar 3.11.

```
# define number of classes based on YAML
import yaml
with open(dataset.location + "/data.yaml", 'r') as stream:
    num_classes = str(yaml.safe_load(stream)['nc'])

[ ] #this is the model configuration we will use for our tutorial
%cat /content/drive/MyDrive/yoga-yolov5-baru/yolov5/models/yolov5s.yaml
```

**Gambar 3.11** Konfigurasi YOLOv5

### 3.8 Training Process

Terdapat beberapa nilai yang digunakan saat melakukan *training* yaitu antara lain nilai *batchsize*, *epoch*, dan ukuran atau *size* yang digunakan ketika training dan akan menghasilkan *learned model*. Untuk melakukan proses training data yang digunakan harus lebih banyak dibandingkan dengan data uji agar meningkatkan kompleksitas algoritma pembelajaran yang digunakan untuk menggambarkan ciri gambar yang beragam. Berikut pseudocode dari hasil proses training:

```
DEFINE dataset_location
DEFINE model_config_path
DEFINE weights_path

FUNCTION
    train_model():
        SET img_size <---
        416

        SET batch_size <---
        32
```

```

SET num_epochs <---
200

SET data_path <--- dataset_location +
"/data.yaml" SET config_path <---
"./models/custom_yolov5s.yaml" SET weights <---
''

SET model_name <---
"yolov5s_results" SET cache_enabled
<--- True

EXECUTE train.py script
  with:image <---
  img_size

  batch_size <---
  batch_size epochs <---
  num_epochs

  data <---
  path_data cfg <-
  -- config_path
  weights <---
  weights

  name <---
  model_name cache <-
  -- cache_enabled

CALL train_model() function

```

### 3.9 Testing Process

Setelah menyelesaikan proses training, model yang telah belajar ( learned model) yang telah didapatkan dari proses training akan dilakukan uji coba dengan melakukan pose atau

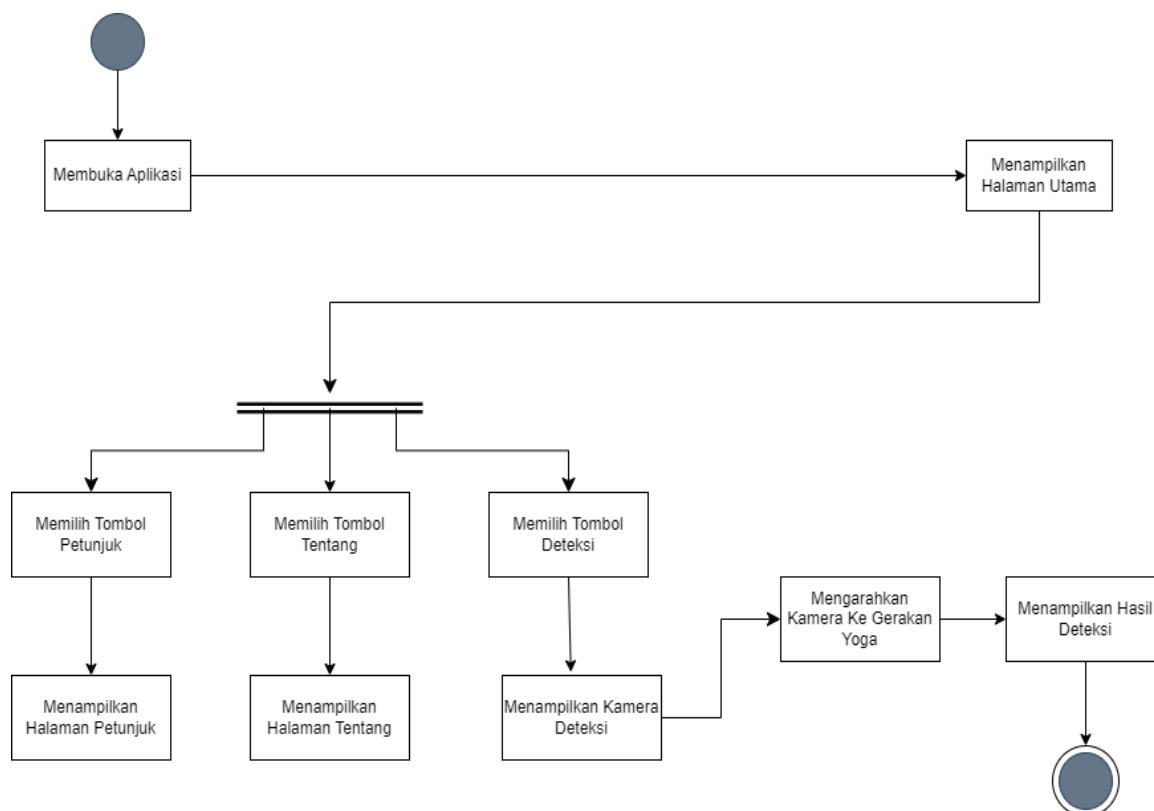
gerakan yoga untuk mengetahui apakah model yang sudah dilatih dapat melakukan deteksi gerakan yoga serta dari proses deteksi tersebut akan dilakukan perhitungan nilai akurasi yang diperoleh dari hasil pengujian yang menggunakan metode YOLOv5 ini sehingga dapat mengetahui hasil kinerja dari model yang didapat.

### 3.10 Perancangan Interface

Rancangan antarmuka (interface) pada aplikasi memiliki tujuan agar bisa ditampilkan dan mendeskripsikan bagaimana gambaran sistem yang telah direncanakan untuk dibuat. Interface yang direncanakan terdiri dari beberapa halaman yaitu diantaranya Home Screen, Deteksi, Petunjuk, dan Tentang.

#### 3.10.1 Activity Diagram

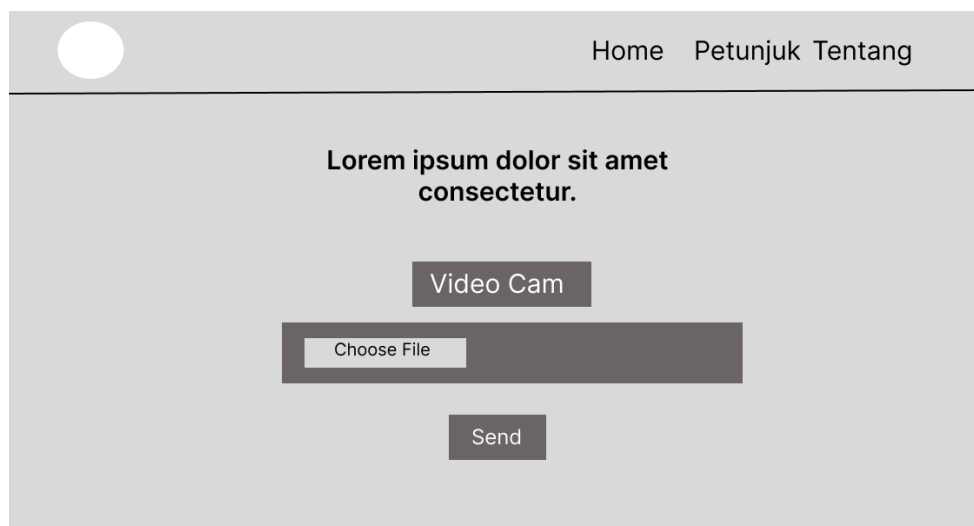
Pembuatan Diagram Aktivitas memiliki tujuan untuk memperlihatkan urutan aktivitas atau tindakan yang dapat dilakukan pada aplikasi. Diagram ini menunjukkan alur kerja dari satu fungsi ke fungsi lain melalui beberapa langkah-langkah yang terjadi pada aplikasi sehingga memudahkan pengguna dalam memahami alur kerja dari sistem tersebut. Diagram aktivitas dapat dilihat dari Gambar 3.12.



**Gambar 3.12** Activity Diagram

### 3.10.2 Tampilan Home Screen

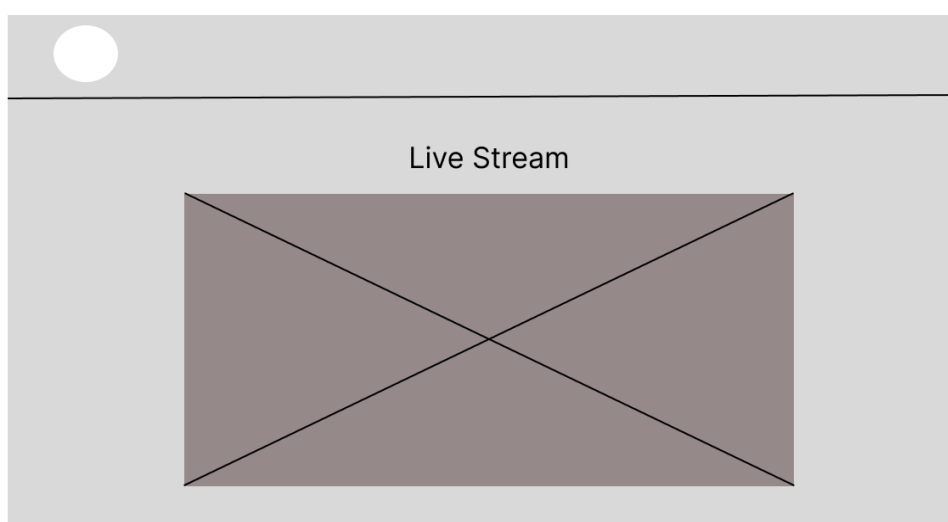
Tampilan berikut merupakan sebuah preview pada aplikasi pada saat aplikasi dijalankan dimana terdapat logo aplikasi serta 3 menu berupa button yang dapat diarahkan kepada halaman setiap menu. Berikut merupakan tampilan wireframe Home Screen aplikasi pada gambar 3.13.



**Gambar 3.13** Wireframe Home Screen

### 3.10.3 Tampilan Halaman Deteksi

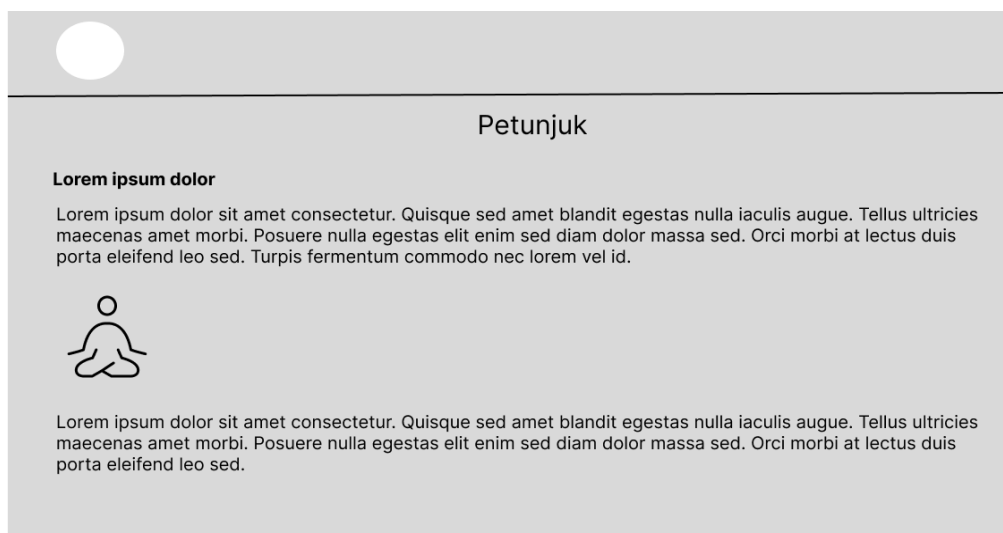
Halaman deteksi merupakan halaman utama dari aplikasi berikut yang akan menampilkan halaman untuk melakukan deteksi gerakan yoga yang akan menggunakan webcam. Pada gambar 3.14 merupakan wireframe halaman deteksi.



**Gambar 3.14** Wireframe Halaman Deteksi

### 3.10.4 Tampilan Halaman Petunjuk

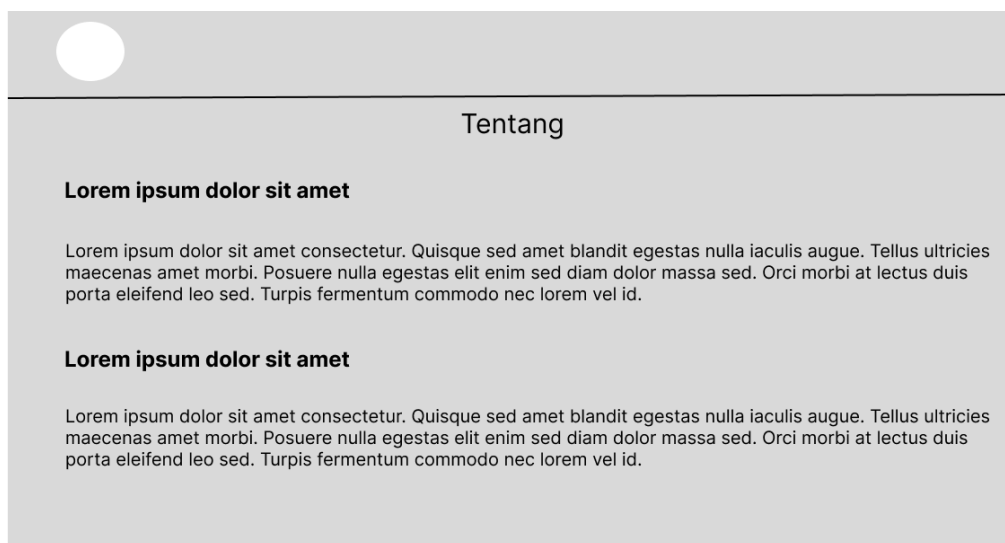
Halaman ini merupakan halaman yang akan memberi informasi mengenai petunjuk penggunaan aplikasi serta juga terdapat informasi mengenai gerakan yang dapat dideteksi serta manfaat dari gerakan tersebut. Pada gambar 3.15 merupakan wireframe halaman petunjuk.



**Gambar 3.15** Wireframe Halaman Petunjuk

### 3.10.5 Tampilan Halaman Tentang

Halaman yang akan menampilkan penjelasan dan informasi mengenai aplikasi ini serta tujuan dalam pembuatan aplikasi kepada pengguna agar lebih mengetahui tentang aplikasi. Berikut menunjukkan wireframe halaman tentang pada gambar 3.16.



**Gambar 3.16** Wireframe Halaman Tentang

## **BAB 4**

### **IMPLEMENTASI DAN PENGUJIAN**

Bab ini terdapat pembahasan mengenai bagaimana proses implementasi deteksi gerakan dalam olahraga yoga menggunakan metode *YOLOv5* dengan rancangan yang telah dibuat sebelumnya dan terdapat analisis serta menunjukkan hasil dari pengujian sistem.

#### **4.1 Implementasi Sistem**

Penggunaan *You Only Look Once (YOLO)* akan diimplementasikan untuk melakukan deteksi berbagai gerakan dalam yoga pada tahap ini. Oleh karena itu, diperlukannya perangkat keras dan perangkat lunak pendukung sebagai berikut:

##### **4.1.1 Perangkat keras dan perangkat lunak**

Untuk membuat sistem pendeteksian ini membutuhkan perangkat keras dan perangkat lunak dengan detail sebagai berikut:

1. Processor AMD A4-9125 Radeon R3 CPU 2.1 GHZ
2. RAM dengan kapasitas 8 GB
3. Kapasitas Hardisk 1TB, SSD 256GB
4. OS Windows 10 Home 64-bit
5. Google Collab GPU Nvidia Cuda V.11.2
6. Flask

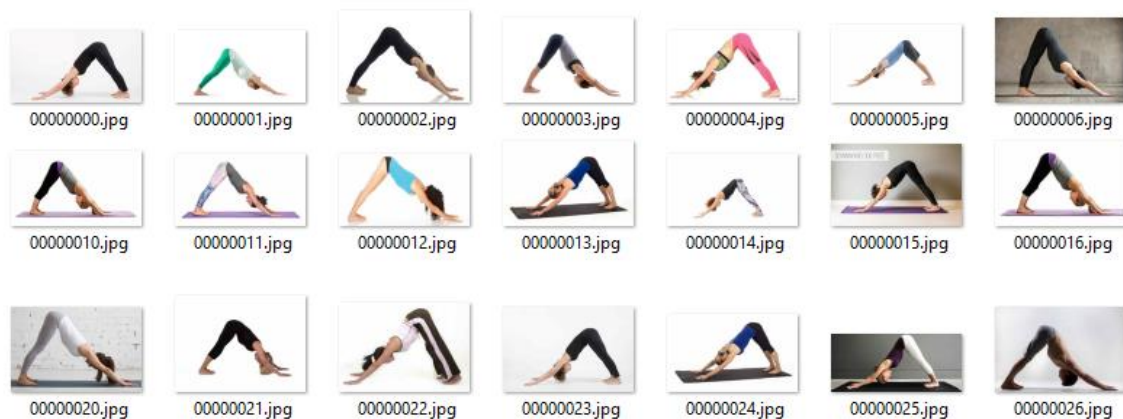
#### **4.2 Implementasi Data**

Data yang digunakan merupakan gambar gerakan atau pose dalam olahraga yoga, yang terdiri dari 5 jenis gerakan yaitu Pose Bridge, Pose DownwardDog, Pose Shoulderstand, Pose Tree, dan Plank yang diperoleh dengan menggunakan kamera smartphone dan dataset dari website kaggle dengan total gambar berjumlah 1000. Berikut merupakan contoh citra gambar yang digunakan seperti yang ditunjukkan dalam gambar 4.1, 4.2, 4.3, 4.4, dan 4.5

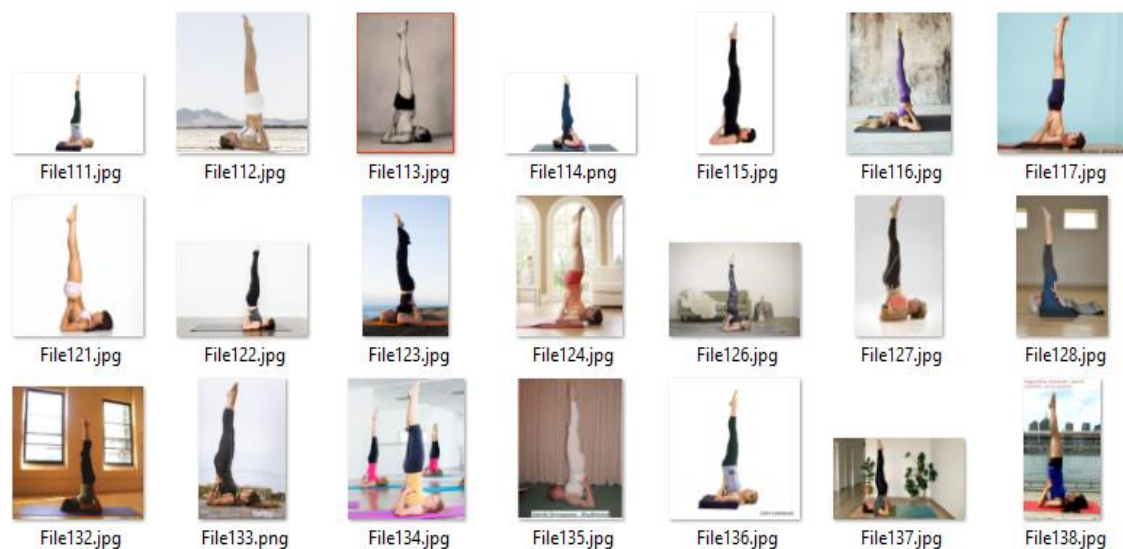




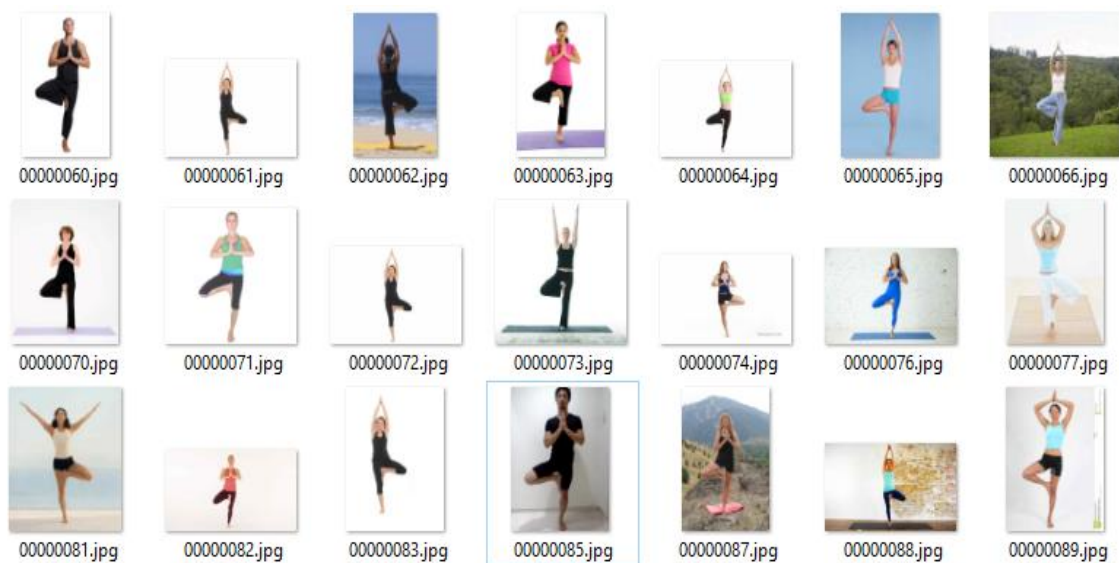
**Gambar 4.1** Pose Bridge



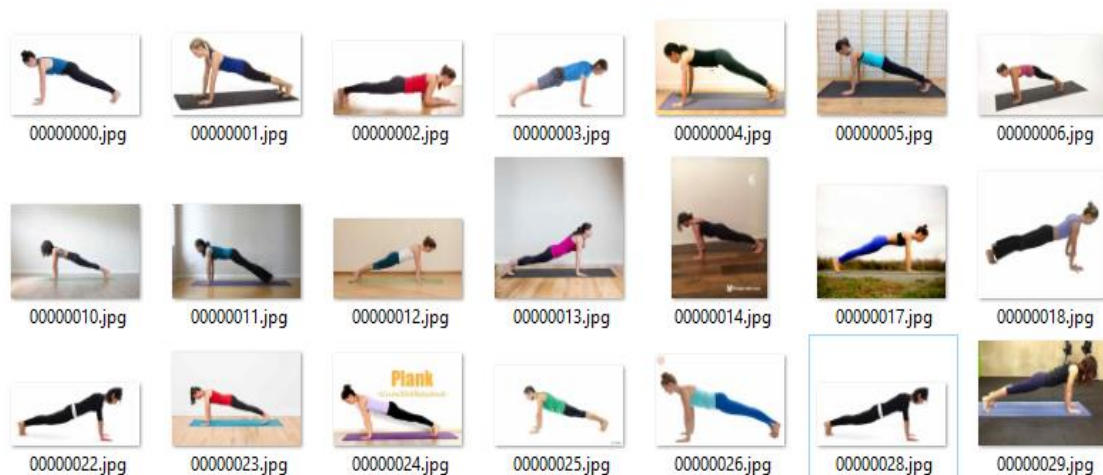
**Gambar 4.2** Pose Downward Dog



**Gambar 4.3** Pose Shoulderstand



**Gambar 4.4** Pose Tree

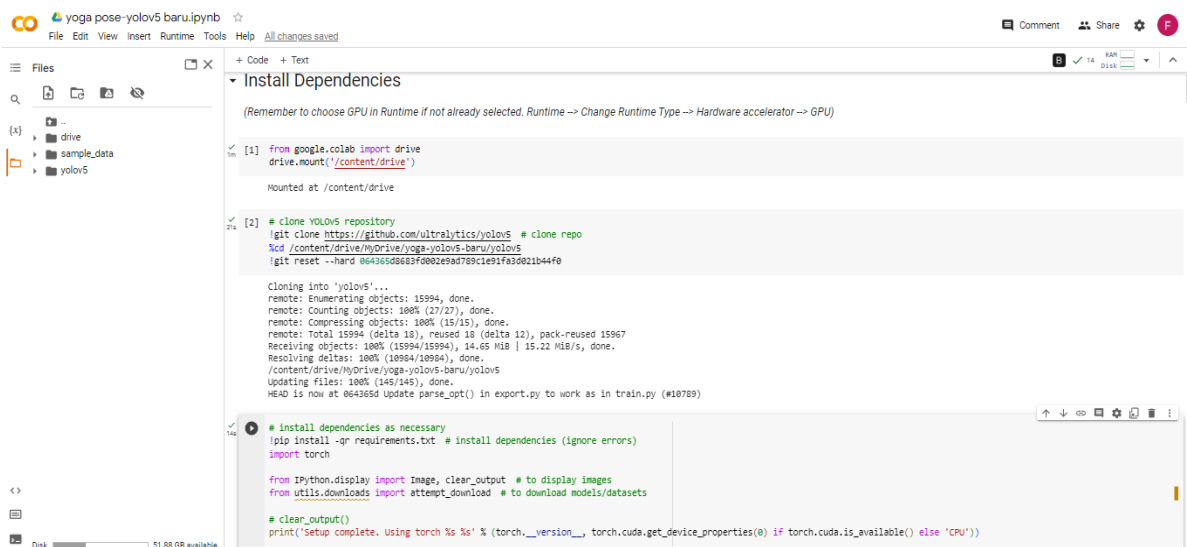


**Gambar 4.5** Plank

### 4.3 Pelatihan Sistem

Setelah melewati tahap preprocessing, proses training data dilakukan supaya model dapat ciri pada setiap objek menggunakan metode *YOLOv5*. Tahapan ini dilakukan menggunakan Google Colab seperti yang dapat dilihat pada Gambar 4.6. Dengan menggunakan Google Colab tersedia pytorch dan runtime GPU NVIDIA yang sudah support CUDA yang dapat digunakan secara gratis dengan batasan runtime 12 jam per hari. Output model yang didapatkan berupa file berekstensi .py. Dalam Proses ini menggunakan repository *YOLOv5* yang telah disediakan dan dapat di clone melalui Github ultralytics. Ukuran gambar yang digunakan yaitu 416 dengan menggunakan weight

pretrained *YOLOv5s* yaitu pretrained weight yang berukuran kecil. Proses training dilakukan menggunakan Google Colab seperti pada gambar 4.6.



**Gambar 4.6** Google Colab

Training data dilakukan dengan melakukan beberapa percobaan dengan menggunakan beberapa kombinasi hyperparameter yaitu batch size dan epoch. Beberapa percobaan dilakukan untuk mendapatkan model yang terbaik dapat dilihat pada Tabel 4.1. Model terbaik yang diperoleh dengan kombinasi *batch size* 32 dan *epoch* 200 dengan nilai *precision* **0.98754**, dimana artinya model yang telah dilatih mampu mendeteksi dengan cukup akurat disebabkan presisi mengukur seberapa banyak prediksi yang bernilai positif yang benar-benar relevan. Serta terdapat nilai *recall* sebesar **0.9896**, yang menunjukkan bahwa model mampu mengingat atau mengidentifikasi semua hasil yang bernilai positif yang sebenarnya dan *mAP-50* sebesar **0.99111**, yang mana menunjukkan jumlah dimana model bisa mengenali objek dengan mempunyai ketepatan yang ditentukan oleh ambang keputusan. Proses training model ini menghabiskan waktu kurang lebih 58 menit 38 detik.

**Tabel 4.1** Percobaan Training

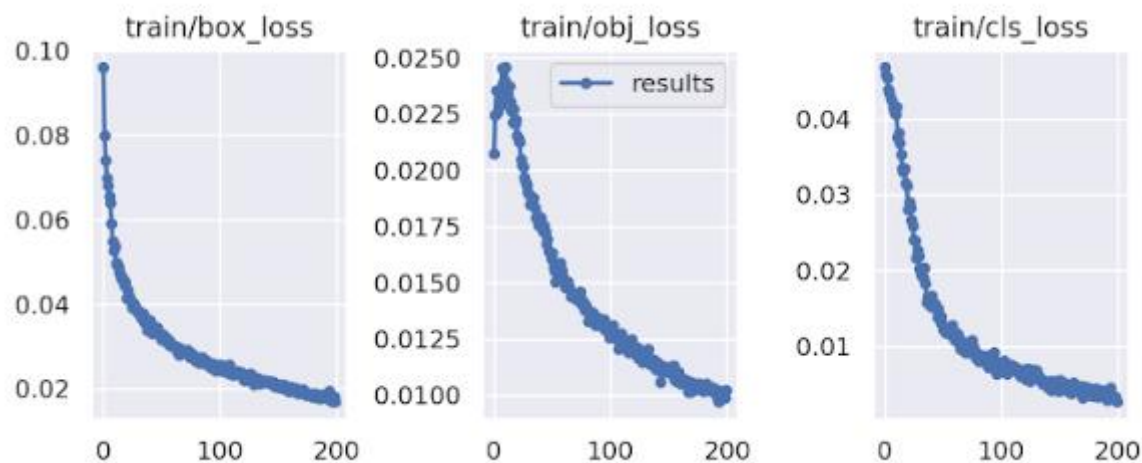
Epoch	Batch Size	mAP	Precision	Recall	Box_loss
50	8	0.97692	0.96259	0.92987	0.032541
100	8	0.98636	0.97023	0.96446	0.024242
150	8	0.99393	0.98378	0.97806	0.021014
200	8	0.99172	0.98505	0.98208	0.018849
50	16	0.94326	0.9389	0.83932	0.035906
100	16	0.98155	0.97203	0.94463	0.026892
150	16	0.98552	0.97051	0.97199	0.022329

200	16	0.99102	0.99008	0.99124	0.018191
50	32	0.94961	0.89804	0.90883	0.033682
100	32	0.99041	0.97467	0.97467	0.026283
150	32	0.99285	0.98083	0.97873	0.022148
<b>200</b>	<b>32</b>	<b>0.99111</b>	<b>0.98754</b>	<b>0.9896</b>	<b>0.01855</b>
50	64	0.94021	0.92306	0.88852	0.035515
100	64	0.98929	0.97072	0.96901	0.02713
150	64	0.99249	0.97211	0.9885	0.022195
200	64	0.991	0.98743	0.98447	0.017644
50	128	0.79715	0.84705	0.74129	0.036082
100	128	0.95681	0.92312	0.91061	0.027895
150	128	0.98092	0.97445	0.94238	0.022708
200	128	0.98827	0.97099	0.97206	0.018892

Setelah melakukan beberapa percobaan training terakhir dilakukan pada *epoch* 200 dengan *batch size* 128 disebabkan tidak adanya perubahan yang signifikan yang terjadi pada lanjutan hasil training. Hasil yang diperoleh dapat dilihat pada Tabel 4.1 menunjukkan bahwa kombinasi antara *batch size* 32 dan *epoch* 200 menghasilkan model dengan performa terbaik. Hal berikut yang dilakukan berhubungan dengan penelitian oleh (Nugroho et al., 2020) menyatakan bahwa proses training dengan menggunakan *batch size* dan *epoch* yang berbeda-beda berperan untuk mendapatkan model terbaik. Berdasarkan beberapa percobaan training yang dilakukan *batch size* yang semakin besar menyebabkan nilai loss yang semakin besar juga. Biasanya *batch size* besar yang digunakan agar mempercepat komputasi, sedangkan menggunakan *batch size* kecil membutuhkan waktu yang lebih lama dalam proses komputasi. Namun training pada penelitian ini menghasilkan hasil yang optimal ketika menggunakan *batch size* yang kecil. Berdasarkan penelitian yang pernah dilakukan sebelumnya oleh (Agusta et al., 2021) pengujian *batch size* pada *epoch* menunjukkan bahwa semakin kecil *batch size* maka semakin baik hasil yang didapatkan walaupun proses komputasi semakin lama.

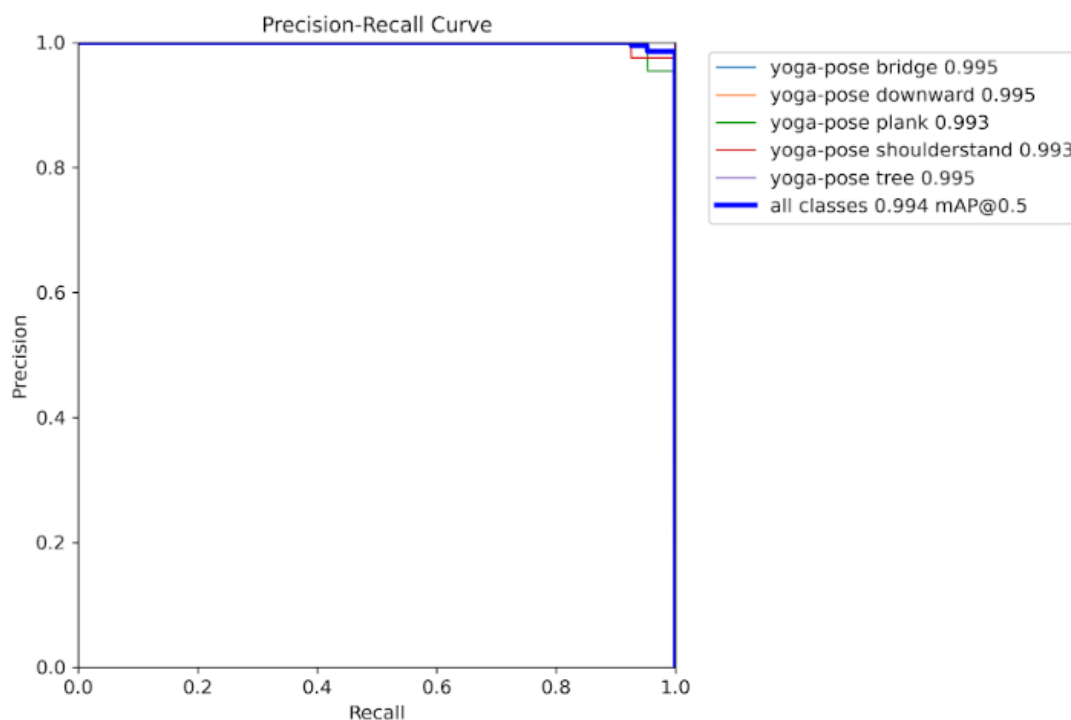
Berdasarkan hasil training tersebut, performa model yang didapatkan dapat dilihat dari grafik loss yang ditunjukkan pada Gambar 4.7, yaitu grafik object loss, objectness loss, dan class loss. Dimana dari kurva object loss menunjukkan seberapa baik algoritma dapat mengalokasikan titik tengah dari sebuah objek dan juga seberapa baik bounding box yang dapat diprediksi. Objectness merupakan ukuran probabilitas suatu objek yang ada dalam region of interest yang diberikan. Classification loss memperlihatkan seberapa baik algoritma yang digunakan dapat melakukan prediksi class yang benar dari objek

yang diberikan. Pada gambar 4.7 menunjukkan grafik loss dari proses training yang dilakukan.



**Gambar 4.7** Grafik loss

*Precision Recall Curve (PR curve)* pada Gambar 4.8, memperlihatkan trade off antara precision dan recall pada threshold yaitu nilai recall yang ditunjukkan pada x-axis dan nilai precision pada y-axis. Berdasarkan kurva diatas menunjukkan bahwa ambang keputusan dari hasil data latih sudah cukup seimbang diantara presisi dan recall. Hasil model yang didapatkan akan tersimpan dengan dua output yaitu best.pt dan last.pt setelah proses training selesai dilakukan.



**Gambar 4.8** Precision Recall Curve

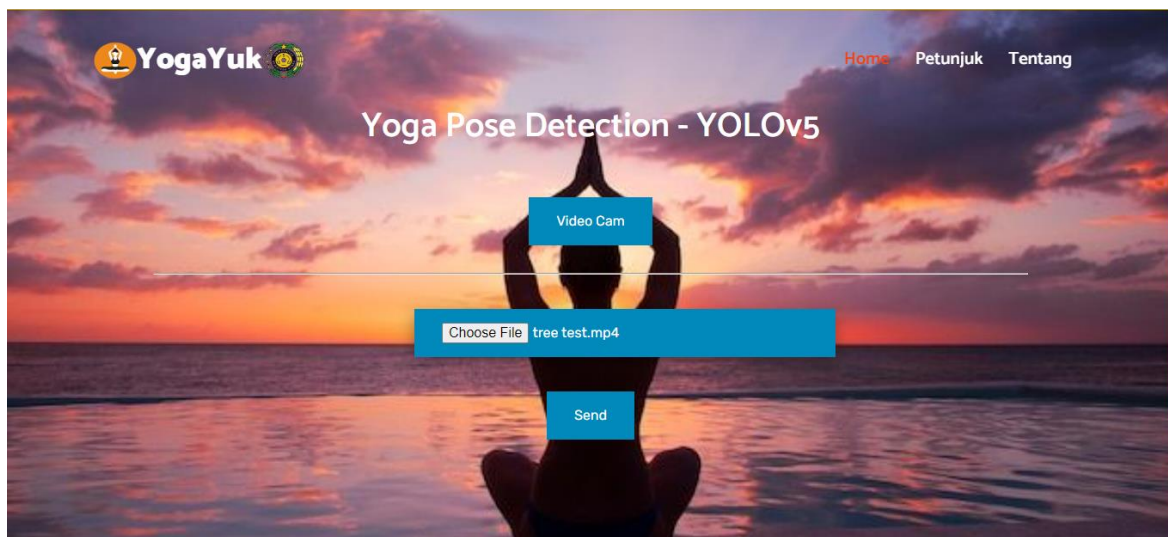


## 4.4 Implementasi Perancangan Interface

Berdasarkan rancangan yang dibuat pada bab tiga maka berikut implementasi dari rancangan interface tersebut:

### 4.4.1 Tampilan Halaman Home Screen

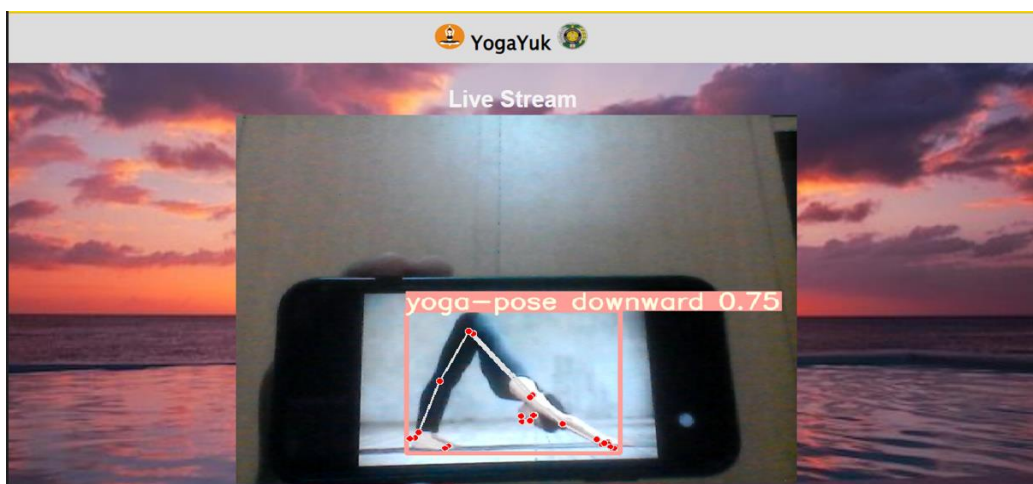
Pada tampilan awal aplikasi terdapat Logo dari aplikasi dan terdapat judul “Yoga Pose Detection”, pertama kali membuka website akan berada pada halaman home yang terdapat 3 tombol, salah satunya tombol Video Cam untuk mulai melakukan deteksi, kemudian dapat mengupload gambar atau video pada bagian Choose File kemudian dapat dilanjutkan dengan klik tombol Send agar memulai deteksi dari file yang dikirim lalu user dapat download file hasil deteksi pada tombol download dibawah Send. Terdapat juga 2 Tombol lainnya yaitu petunjuk dan tentang. Halaman petunjuk berisi informasi mengenai manfaat dari setiap gerakan yoga dan sebelum memulai melakukan deteksi gerakan yoga user juga dapat melihat video langkah-langkah melakukan gerakan yoga dalam halaman petunjuk. Kemudian, pada tombol tentang akan menampilkan halaman yang akan memberikan informasi seputar aplikasi. Pada gambar 4.9 menunjukkan halaman Homescreen.



**Gambar 4.9** Home Screen

#### 4.4.2 Tampilan Halaman Deteksi

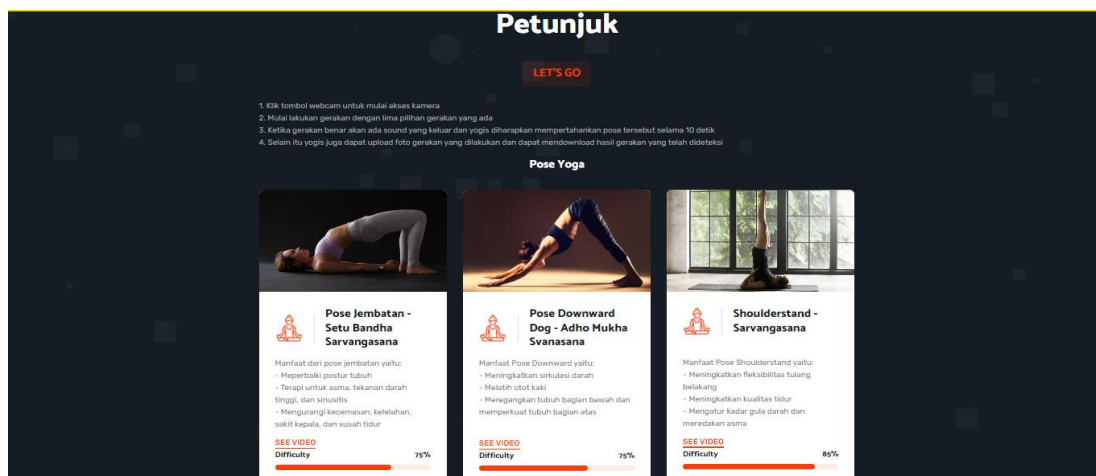
Setelah klik button webcam maka akan memasuki tampilan livestream yang akan membuka kamera webcam. Kemudian pengguna dapat melakukan gerakan yoga dengan menyesuaikan dengan ukuran frame. System akan memunculkan bounding box yang berisi jenis gerakan yang dilakukan. Pada gambar 4.10 memperlihatkan halaman deteksi gerakan.



**Gambar 4.10** Halaman Deteksi

#### 4.4.3 Tampilan Halaman Petunjuk

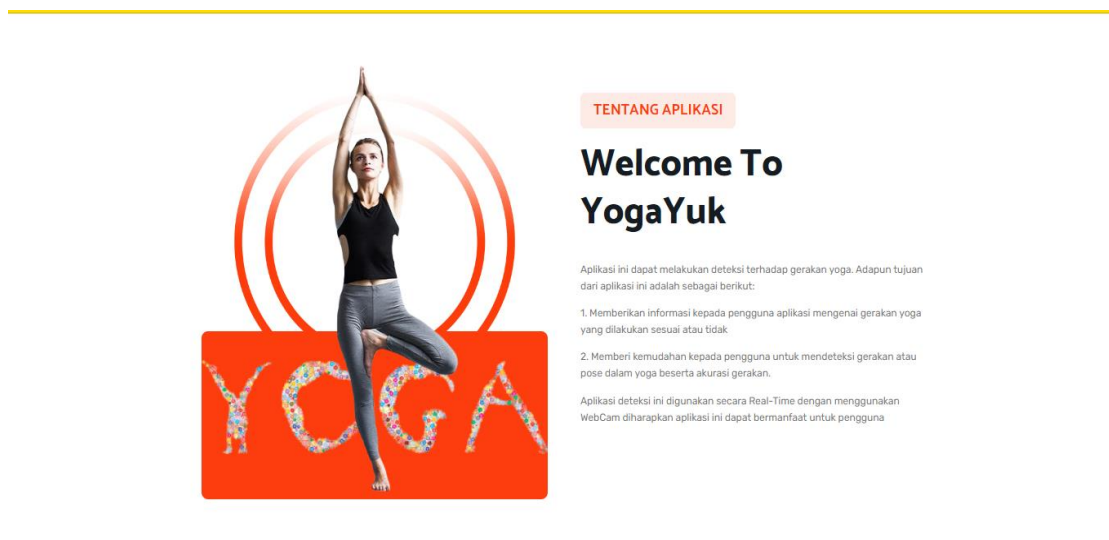
Pada tampilan ini pengguna dapat melihat tampilan mengenai beberapa manfaat dari setiap gerakan yoga yaitu Pose Bridge, Pose DownwardDog, Pose Shoulderstand, Pose Tree, dan Plank dan terdapat video tutorial melakukan setiap gerakan agar pengguna lebih memahami gerakan yoga yang benar. Pada gambar 4.11 dibawah ini terdapat halaman petunjuk.



**Gambar 4.11** Halaman Petunjuk

#### 4.4.5 Tampilan Halaman Tentang

Halaman ini terdapat informasi seputar aplikasi dan tujuan dari pembuatan aplikasi ini agar pengguna juga dapat mengetahui lebih jelas mengenai aplikasi ini. Pada gambar 4.12 menunjukkan halaman tentang dari aplikasi.




**Gambar 4.12** Halaman Tentang




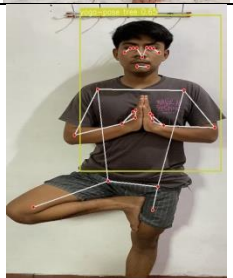


#### 4.5 Pengujian Sistem

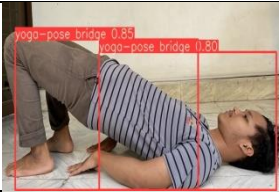




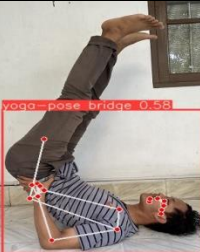

Tahapan ini merupakan tahap untuk melakukan pengujian pada sistem dari model hasil pelatihan yang telah dilakukan pada tahapan sebelumnya untuk mengetahui kemampuan model dalam melakukan deteksi pada gerakan dalam olahraga yoga. Pengujian dilakukan pada halaman website yang telah dibuat dengan hasil model dari pelatihan.


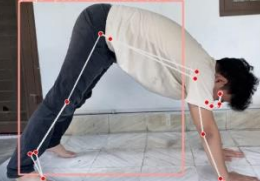



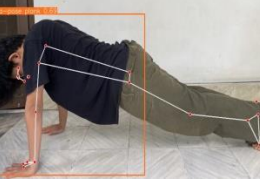


**Tabel 4.2** Pengujian Sistem

No.	Citra	Aktual	Prediksi	Status
1.		Yoga Pose Tree	Yoga Pose Tree	Benar



2.		Yoga Pose Tree	Yoga Pose Tree	Benar
3.		Yoga Pose Tree	Yoga Pose Tree	Benar
4.		Yoga Pose Downward	Yoga Pose Bridge	Salah
5.		Yoga Pose Tree	Yoga Pose Tree	Benar
6.		Yoga Pose Bridge	Yoga Pose Bridge	Benar
7.		Yoga Pose Bridge	Yoga Pose Bridge	Benar
8.		Yoga Pose Bridge	Yoga Pose Bridge	Benar

9.		Yoga Pose Bridge	Yoga Pose Bridge	Benar
10.		Yoga Pose Bridge	Yoga Pose Bridge	Benar
11.		Yoga Pose Shoulderstand	Yoga Pose Bridge	Salah
12.		Yoga Pose Shoulderstand	Yoga Pose Shoulderstand	Benar
13.		Yoga Pose Shoulderstand	Yoga Pose Shoulderstand	Benar
14.		Yoga Pose Shoulderstand	Yoga Pose Bridge	Salah
15.		Yoga Pose Shoulderstand	Yoga Pose Shouderstand	Benar

16.		Yoga Pose Downward	Yoga Pose Downward	Benar
17.		Yoga Pose Downward	Yoga Pose Downward	Benar
18.		Yoga Pose Downward	Yoga Pose Downward	Benar
19.		Yoga Pose Downward	Yoga Pose Downward	Benar
20.		Yoga Pose Downward	Yoga Pose Downward	Benar
21.		Yoga Pose Plank	Yoga Pose Plank	Benar
22.		Yoga Pose Plank	Yoga Pose Plank	Benar
23.		Yoga Pose Plank	Yoga Pose Plank	Benar
24.		Yoha Pose Plank	Yoga Pose Plank	Benar
25.		Yoga Pose Plank	Yoga Pose Plank	Benar

Pengujian yang telah dilakukan diatas dilakukan dengan menggunakan 25 citra data testing dan 75 data testing pada pengujian sebenarnya yang terdapat 15 data uji untuk setiap kelas. Pengujian yang dilakukan menggunakan *inference time* sekitar 1-2 s. Berikut *Confusion Matrix* yang dapat dilihat pada Tabel 4.3.

**Tabel 4.3** Confusion Matrix

		Aktual					Total
		Yoga Pose Tree	Yoga Pose Bridge	Yoga Pose Shoulderstand	Yoga Pose Downward	Yoga Pose Plank	
Prediksi	Yoga Pose Tree	15	0	0	0	0	15
	Yoga Pose Bridge	0	14	2	0	0	16
	Yoga Pose Shoulderstand	1	0	12	0	0	13
	Yoga pose Downward	0	0	0	13	0	13
	Yoga Pose Plank	0	0	0	0	15	15
	Tidak Terdeteksi	0	0	1	1	1	3
Total		16	14	15	14	16	75

Dalam pengujian ini terdapat beberapa kesalahan pada hasil deteksi yang didapatkan. Beberapa kesalahan tersebut dapat terjadi dikarenakan kemiripan beberapa kelas dan angle serta jarak pada saat pendeteksian. Hasil pengujian disajikan pada Tabel 4.4.

**Tabel 4.4** Hasil Pengujian

	TP	FP	FN
Yoga Pose Tree	15	1	0
Yoga Pose Bridge	14	0	2
Yoga Pose Shoulderstand	12	2	1
Yoga Pose Downward	13	0	0
Yoga Pose Plank	15	0	0
Total	69	3	3

Berikut merupakan perhitungan yang dapat dilakukan untuk menghitung nilai dari Akurasi, Presisi, Recall, dan F-1 Score sebagai berikut:

a. *Precision*

Rumus perhitungan Precision dapat dilihat pada persamaan (2)

$$\text{Yoga Pose Tree} = \frac{15}{15+1} \times 100\% = 93,75\%$$

$$\text{Yoga Pose Bridge} = \frac{14}{14+0} \times 100\% = 100\%$$

$$\text{Yoga Pose Shouderstand} = \frac{12}{12+2} \times 100\% = 86\%$$

$$\text{Yoga Pose Downward} = \frac{13}{13+0} \times 100\% = 100\%$$

$$\text{Yoga Pose Plank} = \frac{15}{15+0} \times 100\% = 100\%$$

b. *Recall*

Rumus perhitungan Recall terdapat pada persamaan (3)

$$\text{Yoga Pose Tree} = \frac{15}{15+0} \times 100\% = 100\%$$

$$\text{Yoga Pose Bridge} = \frac{14}{14+2} \times 100\% = 88\%$$

$$\text{Yoga Pose Shouderstand} = \frac{12}{12+1} \times 100\% = 92,3\%$$

$$\text{Yoga Pose Downward} = \frac{13}{13+0} \times 100\% = 100\%$$

$$\text{Yoga Pose Plank} = \frac{15}{15+0} \times 100\% = 100\%$$

c. *F1-Score*

Rumus perhitungan F1-Score terdapat pada persamaan (4)

$$\text{Yoga Pose Tree} = 2 \times \frac{(100\% \times 93,75\%)}{(100\% + 93,75\%)} = 96,77\%$$

$$\text{Yoga Pose Bridge} = 2 \times \frac{(88\% \times 100\%)}{(88\% + 100\%)} = 93,60\%$$

$$\text{Yoga Pose Shouderstand} = 2 \times \frac{(92,3\% \times 86\%)}{(92,3\% + 86\%)} = 90\%$$

$$\text{Yoga Pose Downward} = 2 \times \frac{(100\% \times 100\%)}{(100\% + 100\%)} = 100\%$$

$$\text{Yoga Pose Plank} = 2 \times \frac{(100\% \times 100\%)}{(100\% + 100\%)} = 100\%$$

d. *Accuracy*

Rumus perhitungan Accuracy terdapat pada Persamaan 1.

$$\text{Akurasi} = \frac{69}{75} \times 100\% = 92\%$$

Maka hasil yang diperoleh dari masing-masing kelas dapat dilihat pada Tabel 4.5

**Tabel 4.5** Hasil Perhitungan

	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Yoga Pose Tree	93,75%	100%	96,77%
Yoga Pose Bridge	100%	88%	93,60%
Yoga Pose Shoulderstand	86%	92.3%	90%
Yoga Pose Downward	100%	100%	100%
Yoga Pose Plank	100%	100%	100%

Berdasarkan perhitungan yang telah dilakukan dan hasil diatas, penelitian Deteksi Gerakan Yoga sudah dapat melakukan deteksi gerakan secara Real-time dengan nilai akurasi sebesar 92%. Sistem ini dapat digunakan dengan menggunakan website secara langsung dalam melakukan pengujian. Walaupun dalam proses deteksi masih terdapat beberapa kesalahan, sistem yang digunakan masih dapat digunakan dengan baik dalam melakukan deteksi pada gerakan yoga.

Berikut pengujian sistem dengan mencoba scenario percobaan dengan melakukan percobaan pada posisi yogis dengan jarak yang berbeda. Adapun hasil dari scenario tersebut dapat dilihat pada Tabel 4.6.

**Tabel 4.6** Hasil Percobaan Jarak

No .	Jarak(cm)	Yoga Pose Tree	Yoga Pose Bridge	Yoga Pose Shoulderstand	Yoga Pose Downward	Yoga Pose Plank
1.	50	Tidak terdeteksi	Tidak terdeteksi	Tidak terdeteksi	Tidak terdeteksi	Tidak terdeteksi
2.	100	Yoga Pose Tree	Yoga Pose Plank	Yoga Pose Tree	Tidak Terdeteksi	Yoga Pose Plank
3.	150	Yoga Pose Tree	Yoga Pose Bridge	Yoga Pose Shoulderstand	Yoga Pose Downward	Yoga Pose Plank
4.	200	Yoga Pose Tree	Yoga Pose Bridge	Yoga Pose Shoulderstand	Yoga Pose Downward	Yoga Pose Plank
5	250	Tidak terdeteksi	Tidak terdeteksi	Tidak terdeteksi	Tidak terdeteksi	Tidak terdeteksi

Pada Tabel 4.6 diatas merupakan hasil yang diperoleh dari percobaan sistem yang dilakukan dengan kondisi berbagai jarak. Pada jarak 50 cm sistem tidak dapat melakukan

deteksi dikarenakan jarak yang terlalu dekat. Sementara pada jarak 100 cm beberapa gerakan dapat dideteksi dengan benar oleh sistem sementara gerakan yang lainnya masih belum terdeteksi dan hasil deteksi yang salah. Pada jarak 150 hingga 200 cm sistem mampu melakukan deteksi dengan tepat sesuai dengan kelas yang ada, namun pada jarak 250 cm sistem tidak lagi mampu mendeteksi gerakan yang ada.

## BAB 5

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Setelah dari beberapa proses pengujian dalam penelitian deteksi gerakan yoga yang menggunakan metode *You Only Look Once v5* dapat diperoleh kesimpulan sebagai berikut:

1. Metode *You Only Look Once* dapat mendeteksi gerakan yoga dengan akurasi sebesar 92% dari total 75 data uji.
2. Pengaturan jumlah *epoch* dan *batch size* yang mempunyai pengaruh untuk mendapatkan model dengan hasil terbaik
3. Sistem dapat melakukan deteksi dengan menggunakan webcam secara *real time*
4. Aplikasi website dapat berjalan dengan inference time sebesar 1-2 s.
5. Hasil terbaik dari proses training yang dilakukan diperoleh model pada epoch 200 dengan menggunakan batch size 32

#### 5.2 Saran

Adapun saran berikut dapat digunakan sebagai referensi untuk penelitian lanjutan:

1. Diharapkan untuk penelitian yang dilakukan selanjutnya dapat meningkatkan proses training dan mengumpulkan lebih banyak jumlah data dari setiap jenis gerakan serta dapat menambahkan jenis gerakan yang lebih bervariasi.
2. Penelitian lebih lanjut dapat dilakukan dengan mencoba metode *You Only Look Once* versi terbaru.
3. Penelitian selanjutnya juga diharapkan untuk mencoba menerapkan metode algoritma lain agar bisa dibandingkan dengan hasil yang diperoleh saat menggunakan metode *You Only Look Once Versi 5*.
4. Pengembangan berikutnya dapat dilakukan dalam bentuk aplikasi mobile.



## DAFTAR PUSTAKA

- Agusta, A., Ernawati, I., & Muliawati, A. (2021). Prediksi Pergerakan Harga Saham Pada Sektor Farmasi Menggunakan Algoritma Long Short-Term Memory. *Informatik : Jurnal Ilmu Komputer*, 17(2), 164. <https://doi.org/10.52958/iftk.v17i2.3651>
- Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*. <http://arxiv.org/abs/2004.10934>
- Daniel Tanugraha, F., Pratikno, H., Musayanah, M., & Indah Kusumawati, W. (2022). Pengenalan Gerakan Olahraga Berbasis (Long Short- Term Memory) Menggunakan Mediapipe. *Journal of Advances in Information and Industrial Technology*, 4(1), 37–45. <https://doi.org/10.52435/jaiit.v4i1.182>
- Gajbhiye, R., Jarag, S., Gaikwad, P., & Koparde, S. (2022). AI Human Pose Estimation: Yoga Pose Detection and Correction. *International Journal of Innovative Science and Research Technology*, 7(5). [www.ijisrt.com](http://www.ijisrt.com)
- Gunarta, I. K. (2017). Implementasi Pembelajaran Yoga Dalam Meningkatkan Konsentrasi Belajar Di Sekolah Dasar Negeri 1 Sumerta. *Jurnal Penjaminan Mutu*, 3(2), 180. <https://doi.org/10.25078/jpm.v3i2.198>
- Harliana. (2019). Penerapan Yoga Dalam Menstimulasi Kemampuan Disiplin Anak Usia Dini. *Tumbuh Kembang : Jurnal PG PAUF FKIP Universitas Sriwijaya*, 6(2), 130–135.
- Herwanto, A., & Utaminingrum, F. (2023). Deteksi Hambatan Arsitektural Jalan pada Kursi Roda Pintar menggunakan Tiga Model YOLOv5 berbasis TX2. 7(5), 2255–2259. <http://j-ptiik.ub.ac.id>
- Iskandar Mulyana, D., Faizal Lazuardi, M., & Betty Yel, M. (2022). Deteksi Bahasa Isyarat Dalam Pengenalan Huruf Hijaiyah Dengan Metode YOLOV5. *Jurnal Teknik Elektro Dan Komputasi (ELKOM)*, 4, 145–151.
- Iskandar Mulyana, D., & Rofik, M. A. (2022). Implementasi Deteksi Real Time Klasifikasi Jenis Kendaraan Di Indonesia Menggunakan Metode YOLOV5. *Jurnal Pendidikan Tambusai*, 6(3), 13971–13982. <https://doi.org/10.31004/jptam.v6i3.4825>
- Kim, J. W., Choi, J. Y., Ha, E. J., & Choi, J. H. (2023). Human Pose Estimation Using MediaPipe Pose and Optimization Method Based on a Humanoid Model. *Applied Sciences (Switzerland)*, 13(4). <https://doi.org/10.3390/app13042700>
- Liu, S. (n.d.). *Path Aggregation Network for Instance Segmentation*.
- Madja, I. K. (2018). FUNGSI ASTANGGA YOGA PATANJALI (Perspektif Pendidikan Agama Hindu). *Guna Widya: Jurnal Pendidikan Hindu*, 4(1), 21–32. <https://doi.org/10.25078/gw.v4i1.385>
- Mahasin, M., & Dewi, I. A. (2022). Comparison of CSPDarkNet53, CSPResNeXt-50, and EfficientNet-B0 Backbones on YOLO V4 as Object Detector. *International Journal of Engineering, Science and Information Technology*, 2(3), 64–72. <https://doi.org/10.52088/ijesty.v2i3.291>

- Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, 5(2), 697–711.  
<https://ejurnal.tunasbangsa.ac.id/index.php/jsakti/article/view/369/348>
- Nugroho, P. A., Fenriana, I., & Arijanto, R. (2020). Implementasi Deep Learning Menggunakan Convolutional Neural Network ( Cnn ) Pada Ekspresi Manusia. *Algor*, 2(1), 12–21.
- Pathak, A. R., Pandey, M., & Rautaray, S. (2018). Application of Deep Learning for Object Detection. *Procedia Computer Science*, 132(Iccids), 1706–1717.  
<https://doi.org/10.1016/j.procs.2018.05.144>
- Raptis, M., Kirovski, D., & Hoppe, H. (2011). Real-time classification of dance gestures from skeleton animation. *Proceedings - SCA 2011: ACM SIGGRAPH / Eurographics Symposium on Computer Animation, August 2011*, 147–156.  
<https://doi.org/10.1145/2019406.2019426>
- Rathikarani, V., Abarna, S., & Vijayakumar, K. (2022). Classification Of Yoga Pose Using Pretrained Convolutional Neural Networks. *Journal of Pharmaceutical Negative Results*, 13(8), 3798–3805. <https://doi.org/10.47750/pnr.2022.13.S08.474>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). Efficient convnet-based object detection for unmanned aerial vehicles by selective tile processing. *ACM International Conference Proceeding Series*.  
<https://doi.org/10.1145/3243394.3243692>
- Shah, D., Rautela, V., Sharma, C., & Florence, A. A. (2021). Yoga Pose Detection Using Posenet and k-NN. *2021 International Conference on Computing, Communication and Green Engineering, CCGE 2021, May*.  
<https://doi.org/10.1109/CCGE50943.2021.9776451>
- Shorten, C., Khoshgoftaar, T. M., & Furht, B. (2021). Text Data Augmentation for Deep Learning. In *Journal of Big Data* (Vol. 8, Issue 1). Springer International Publishing.  
<https://doi.org/10.1186/s40537-021-00492-0>
- Steven, G., & Purbowo, A. N. (2022). Penerapan 3D Human Pose Estimation Indoor Area untuk Motion Capture dengan Menggunakan YOLOv4-Tiny, EfficientNet Simple Baseline, dan VideoPose3D. *Jurnal Infra*, 10(2), 1–7.
- Taşyürek, M., & Öztürk, C. (2023). A fine-tuned YOLOv5 deep learning approach for real-time house number detection. *PeerJ Computer Science*, 9, 1–26.  
<https://doi.org/10.7717/PEERJ-CS.1453>
- Wang, S. (2022). A Deep Learning Algorithm for Special Action Recognition of Football. *Mobile Information Systems*, 2022. <https://doi.org/10.1155/2022/6315648>
- Wiese, C., Keil, D., Rasmussen, A. S., & Olesen, R. (2019). Injury in yoga asana practice: Assessment of the risks. *Journal of Bodywork and Movement Therapies*, 23(3), 479–488. <https://doi.org/10.1016/j.jbmt.2018.09.151>
- Yan, B., Fan, P., Lei, X., Liu, Z., & Yang, F. (2021). A real-time apple targets detection method for picking robot based on improved YOLOv5. *Remote Sensing*, 13(9), 1–23.

<https://doi.org/10.3390/rs13091619>

Yan, J., Wang, H., Yan, M., Diao, W., Sun, X., & Li, H. (2019). IoU-adaptive deformable R-CNN: Make full use of IoU for multi-class object detection in remote sensing imagery. *Remote Sensing*, *11*(3), 1–22. <https://doi.org/10.3390/rs11030286>



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://Fasikomti.usu.ac.id>

KEPUTUSAN  
DEKAN FAKULTAS ILMU KOMPUTER  
DAN TEKNOLOGI INFORMASI  
NOMOR : 23/UN5.2.1.14/SK/SPB/2024  
DEKAN FAKULTAS ILMU KOMPUTER  
DAN TEKNOLOGI INFORMASI UNIVERSITAS SUMATERA UTARA

- Membaca** : Surat Permohonan Mahasiswa Fasikom-TI USU tanggal 8 Januari 2024 perihal permohonan ujian skripsi  
Nama : FARIS AL FALAQ  
NIM : 191402129  
Program Studi : Sarjana (S-1) Teknologi Informasi  
Judul Skripsi : Deteksi Gerakan Dalam Olahraga Yoga Menggunakan Metode You Only Look Once (YOLO)
- Memperhatikan** : Bahwa Mahasiswa tersebut telah memenuhi kewajiban untuk ikut dalam pelaksanaan Meja Hijau Skripsi Mahasiswa pada Program Studi Sarjana (S-1) Teknologi Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara TA 2023/2024.
- Menimbang** : Bahwa permohonan tersebut diatas dapat disetujui dan perlu ditetapkan dengan surat keputusan
- Mengingat** : 1. Undang-undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional  
2. Peraturan Pemerintah Nomor 17 tahun 2010 tentang pengelolaan dan penyelenggara pendidikan.  
3. Keputusan Rektor USU Nomor 03/UN5.1.R/SK/SPB/2021 tentang Peraturan Akademik Program Sarjana Universitas Sumatera Utara.  
4. Surat Keputusan Rektor USU Nomor 1876/UN5.1.R/SK/SDM/2021 tentang pengangkatan Dekan Fasikom-TI USU Periode 2021-2026

MEMUTUSKAN

- Menetapkan** :  
**Pertama** : Membentuk dan mengangkat Tim Penguji Skripsi mahasiswa sebagai berikut:  
Ketua : Prof. Dr. Drs. Opim Salam Sitompul M.Sc  
NIP: 196108171987011001  
Sekretaris : Fahrurrozi Lubis B.IT., M.Sc.IT  
NIP: 198610122018052001  
Anggota Penguji : Ainul Hizriadi S.Kom, M.Sc  
NIP: 198510272017061001  
Anggota Penguji : Niskarto Zendrato S.Kom., M.Kom  
NIP: 198909192018051001  
Moderator : -  
Panitera : -
- Kedua** : Segala biaya yang diperlukan untuk pelaksanaan kegiatan ini dibebankan pada Dana Penerimaan Bukan Pajak (PNPB) Fasikom-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.

Ditetapkan di : Medan  
Pada Tanggal : 11 Januari 2024  
Dekan,

MAYA SILVIA LYDIA  
NIP 197401272002122001

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