



## **LAPORAN PENELITIAN KoPSI**

**SocialCam : Sistem Detektor Penerapan *Social Distancing Real-Time*  
Menggunakan Kamera Berbasis Algoritma *Deep Learning YOLOv4*  
Sebagai Upaya Pencegahan Penyebaran COVID-19 di Fasilitas Umum**

**Oleh:**

**Raihan Adhipratama Arvi**

NISN : 0037054239

**Fisika Terapan Rekayasa (FTR)**

**SMA Negeri 1 Sumatera Barat  
Kota Padang Panjang, Provinsi Sumatera Barat  
Tahun 2021**

## PERNYATAAN ORISINALITAS PENELITI

### PERNYATAAN PENELITI

Yang bertanda tangan di bawah ini:

Nama : Raihan Adhipratama Arvi  
Tempat/Tanggal Lahir : Padang / 16 Juli 2003  
NISN : 0037054239  
Asal Sekolah : SMAN 1 Sumatera Barat

dengan ini menyatakan sejujurnya bahwa laporan penelitian saya dengan judul

SocialCam : Sistem Detektor Penerapan *Social Distancing Real-Time* Menggunakan Kamera Berbasis Algoritma *Deep Learning* YOLOv4 Sebagai Upaya Pencegahan Penyebaran COVID-19 di Fasilitas Umum.

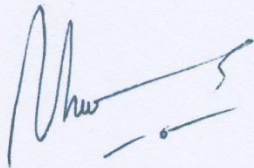
**bersifat orisinal/bukan plagiasi/belum pernah dikompetisikan dan/atau pernah dilombakan tetapi belum mendapat juara/penghargaan di tingkat Nasional/Internasional**

Bilamana di kemudian hari ditemukan ketidaksesuaian dengan pernyataan ini, saya bersedia menerima konsekuensi sesuai aturan KoPSI.

Demikian pernyataan ini dibuat dengan sesungguhnya dan dengan sebenar-benarnya.

Dibuat di: Kota Padang Panjang  
Pada Tanggal: 02 September 2021

Mengetahui,



Guru Pembimbing,  
Nilma Herrita W. S., S.Pd, M.Si  
NIP: 19730222 200501 2 005

Yang membuat pernyataan,



Raihan Adhipratama Arvi  
NISN: 0037054239

# **SocialCam : Sistem Detektor Penerapan *Social Distancing Real-Time* Menggunakan Kamera Berbasis Algoritma *Deep Learning YOLOv4* Sebagai Upaya Pencegahan Penyebaran COVID-19 di Fasilitas Umum**

Oleh:

**Raihan Adhipratama Arvi**

SMA Negeri 1 Sumatera Barat, Jalan Syekh Ibrahim Musa, Sigando,  
Padang Panjang Tim., Kota Padang Panjang, Sumatera Barat 27126

e-mail: [raihanadhiarvi@gmail.com](mailto:raihanadhiarvi@gmail.com)

tel: +62 812 1000 2845

## **ABSTRAK**

*Social distancing* merupakan salah satu cara meminimalisir penyebaran COVID-19. Pemerintah Indonesia sendiri menetapkan *social distancing* sebagai suatu kewajiban di tempat-tempat umum dengan jarak minimal 1 meter antarindividu. Penelitian ini berfokus dalam pengembangan *software* dan *model* detektor *social distancing real-time* berbasis YOLOv4 menggunakan kamera CCTV. *Training model* YOLOv4 dilakukan terhadap 400 gambar *dataset* detektor *pedestrian* dengan 5000 iterasi. Metode analisis data yang digunakan adalah metode deskriptif, dalam kasus ini mendeskripsikan hasil pengembangan program, metode komparatif yang digunakan untuk membandingkan data pengujian terhadap variasi kondisi kontrol, dan perhitungan nilai AP (*average precision*). Program memiliki beberapa fitur unggulan dalam deteksi *social distancing* seperti *input* berupa *file* video dan kamera, kalibrasi otomatis, *video output*, *log*, dan *argument parser*. Program menggunakan algoritma *perspective wrap* sebagai solusi dari distorsi perspektif atau *foreshortening* dari perspektif kamera, sehingga pengukuran jarak yang dilakukan akurat. Program juga akan memberikan peringatan berupa suara apabila minimal pelanggaran telah terdeteksi selama beberapa satuan waktu yang dapat dikonfigurasi. Hasil *training* untuk 5000 iterasi memakan waktu sekitar 16 jam 25 menit dengan *training loss* sebesar 0,8421. Berdasarkan pengujian yang dilakukan, program telah mencapai predikat *real-time* dengan kecepatan deteksi hingga 42,84 FPS. Pengujian akurasi *model* YOLOv4 yang dilakukan terhadap 113 *dataset* validasi menghasilkan *average precision* (AP) untuk AP50 sebesar 82,82% dan AP75 sebesar 59,27%. Program ini berpotensi menjadi solusi tepat bagi instansi pemerintah ataupun swasta dalam automasi deteksi *social distancing* di tempat-tempat umum sebagai salah satu langkah pencegahan penyebaran COVID-19 di Indonesia.

**KATA KUNCI:** *social distancing*, YOLOv4, CCTV

## KATA PENGANTAR

Puji dan syukur penulis hantarkan ke hadirat Allah Swt. Karena atas berkat rahmat dan karunia-Nya penulis telah dapat menyelesaikan laporan hasil penelitian yang berjudul “SocialCam : Sistem Detektor Penerapan *Social Distancing Real-Time* Menggunakan Kamera Berbasis Algoritma *Deep Learning* YOLOv4 Sebagai Upaya Pencegahan Penyebaran COVID-19 di Fasilitas Umum”.

Penulis telah melaksanakan penelitian dan menulis laporan penelitian ini dengan maksimal. Namun, keterbatasan pelaksanaan penelitian membuat laporan penelitian ini tidak luput dari kekurangan. Oleh karena itu, dengan rendah hati penulis memohon kritik dan saran yang membangun dari berbagai pihak demi perbaikan di masa depan.

Penulis memohon maaf apabila ada hal yang kurang berkenaan pada tempatnya. Semoga penelitian ini membawa segenap manfaat bagi para pembaca dan nantinya bisa menjadi acuan dalam pengembangan ilmu pengetahuan Indonesia.

Kota Padang Panjang, 25 September 2021

Penulis,  
Raihan Adhipratama Arvi  
NISN: 0037054239

## DAFTAR ISI

PERNYATAAN ORISINALITAS PENELITI .....	i
ABSTRAK .....	ii
KATA PENGANTAR .....	iii
DAFTAR ISI .....	iv
DAFTAR GAMBAR .....	vi
DAFTAR TABEL .....	vi
DAFTAR DIAGRAM .....	vi
BAB 1. PENDAHULUAN .....	1
1.1    LATAR BELAKANG .....	1
1.2    RUMUSAN MASALAH .....	2
1.3    TUJUAN PENELITIAN .....	2
1.4    MANFAAT PENELITIAN .....	2
BAB 2. TINJAUAN PUSTAKA .....	3
2.1    PENELITIAN TERDAHULU .....	3
2.2    SOCIAL DISTANCING .....	3
2.3    ALGORITMA OBJECT RECOGNITION .....	3
2.4    YOLOv4 .....	4
BAB 3. METODE PENELITIAN .....	5
3.1    WAKTU DAN TEMPAT .....	5
3.2    METODE PENGUMPULAN DATA .....	5
3.3    ALUR PENELITIAN .....	5
3.4    KONSEP KERJA ALAT .....	5
3.5    ALAT DAN BAHAN .....	6
3.6    SOFTWARE DAN LIBRARY .....	6
3.7    DATASET .....	6
3.8    METODE PENGUMPULAN DATASET .....	6
3.9    LABELLING DATASET .....	7
3.10   PROSES TRAINING .....	7
3.11   TEKNIK ANALISIS DATA .....	8
BAB 4. HASIL & PEMBAHASAN .....	9
4.1    ALGORITMA DETEKSI SOCIAL DISTANCING SOCIALCAM .....	9
4.2    HASIL <i>TRAINING MODEL</i> SOCIALCAM .....	13
4.3    PENGUJIAN PERFORMA SOCIALCAM .....	14
BAB 5. KESIMPULAN & SARAN .....	17

5.1	KESIMPULAN.....	17
5.2	SARAN.....	17
	UCAPAN TERIMA KASIH.....	18
	REFERENSI.....	19
	LAMPIRAN .....	20
	LAMPIRAN 1. SURAT PERNYATAAN ORISINALITAS .....	21
	LAMPIRAN 2. SURAT TUGAS PEMBIMBING DARI KEPALA SEKOLAH.....	22
	LAMPIRAN 3. SPESIFIKASI ALAT YANG DIGUNAKAN.....	23
	LAMPIRAN 4. ALAT & BAHAN .....	25
	LAMPIRAN 5. LINK DATASET .....	27
	LAMPIRAN 6. DOKUMENTASI PENELITIAN.....	29
	LAMPIRAN 7. LINK GOOGLE DRIVE & GITHUB FILE PROGRAM SOCIALCAM .....	31
	LAMPIRAN 8. LINK VIDEO SOCIALCAM.....	32
	LAMPIRAN 9. PENGUJIAN RESOLUSI VIDEO.....	33
	LAMPIRAN 10. LINK SOFTWARE YANG DIGUNAKAN.....	34
	LAMPIRAN 11. TIME TABLE PENELITIAN .....	35
	LAMPIRAN 12. DOKUMENTASI PROGRAM SOCIALCAM .....	36
	LAMPIRAN 13. GRAFIK PENGUJIAN PERFORMA FPS.....	37
	LAMPIRAN 14. HYPERPARAMETER PROSES TRAINING .....	38

## DAFTAR GAMBAR

Gambar 1 MS COCO Benchmark (Alexey Bochkovskiy, 2020). .....	4
Gambar 2 Tampilan SocialCam .....	9
Gambar 3 Algoritma Deteksi Objek YOLO.....	10
Gambar 4 Penentuan Centroid .....	10
Gambar 5 Perspective Wrap SocialCam .....	11
Gambar 6 Penentuan Jarak Subjek.....	11
Gambar 7 Nilai Training Loss Model SocialCam .....	13
Gambar 8 Nilai Average Precision (AP) SocialCam. AP50 (kiri) dan AP75 (kanan). .....	14
Gambar 9 Grafik Pengujian Pengukuran Jarak SocialCam .....	16

## DAFTAR TABEL

Tabel 1 Software dan Library .....	6
Tabel 2 Hasil Pengujian Performa FPS SocialCam.....	15

## DAFTAR DIAGRAM

Diagram 1 Alur Penelitian .....	5
Diagram 2 Algoritma Pemberian Peringatan SocialCam .....	12

## BAB 1. PENDAHULUAN

### 1.1 LATAR BELAKANG

Awal tahun 2020 merupakan peristiwa yang bersejarah bagi seluruh dunia. Hal ini disebabkan oleh munculnya wabah penyakit *coronavirus disease* atau COVID-19. Wabah ini menyebar dengan cepat ke seluruh belahan dunia. Negara Indonesia sendiri mengonfirmasikan dua kasus COVID-19 pertama pada tanggal 2 Maret 2020. Satuan Tugas Penanganan COVID-19 terhitung tanggal 26 Mei 2021 menyatakan kasus COVID-19 di Indonesia telah mencapai 1.791.221 terinfeksi, 1.645.263 telah sembuh, dan 49.771 meninggal dunia.

Wabah COVID-19 yang semakin memburuk membuat pemerintah Indonesia menutup tempat-tempat umum seperti pusat perbelanjaan dan sekolah serta memberlakukan pembatasan sosial berskala besar (PSBB) sebagai upaya pemutusan rantai penyebaran COVID-19 di Indonesia. Namun, dikarenakan dampak negatif PSBB pada bidang perekonomian nasional, PSBB kembali dihapuskan dan tempat-tempat umum kembali dibuka secara bertahap. Sebagai upaya pemerintah dalam menekan angka penyebaran COVID-19 di Indonesia, pemerintah menerapkan peraturan adaptasi kebiasaan baru (*new normal*) dengan segenap protokol kesehatan yang harus dipatuhi. Peraturan ini dimuat dalam Keputusan Menteri Kesehatan Republik Indonesia Nomor HK.01.07/MENKES/382/2020.

Salah satu poin penting dalam Keputusan Menteri Kesehatan Republik Indonesia tersebut adalah mewajibkan bagi para individu yang berada di tempat-tempat umum untuk melakukan *social distancing* atau menjaga jarak dengan jarak minimal 1 meter antar individu. Selain itu, pengelola fasilitas umum juga diwajibkan untuk menjaga penerapan aturan *social distancing* pada fasilitas yang dikelola. Misalnya dengan mengatur jarak antri, mengatur jarak kursi, dan menegur pelanggar peraturan *social distancing*. Namun, nyatanya masih banyak ditemukan pelanggaran aturan *social distancing* dalam kehidupan sehari-hari. Hal ini disebabkan oleh berbagai faktor, mulai dari kelalaian pengelola fasilitas umum, tingkat kepedulian masyarakat yang rendah, luasnya area fasilitas umum yang menyulitkan pemantauan penerapan *social distancing*, dan banyaknya jumlah pengunjung yang berada di fasilitas umum tersebut sehingga menyulitkan penerapan peraturan *social distancing*. Hal ini juga membuat angka penyebaran COVID-19 semakin naik.

Salah satu upaya dalam mengatasi masalah tersebut adalah dengan mengembangkan alat yang menggunakan teknologi *deep learning* untuk mendeteksi penerapan *social distancing* pada fasilitas umum melalui kamera secara *real-time*, serta memberikan peringatan secara otomatis apabila ditemukan pelanggar *social distancing*. Alat pertama-tama akan mendeteksi keberadaan setiap manusia pada gambar yang dihasilkan dari kamera, lalu alat akan melakukan perhitungan jarak antar setiap manusia tersebut. Alat ini kemudian dapat dipasang pada fasilitas-fasilitas umum seperti sekolah, pusat perbelanjaan, dan bandar udara sebagai pemantau penerapan *social distancing*. Alat ini berpotensi efektif dan efisien dalam mendeteksi penerapan *social distancing* karena wujudnya yang seperti kamera CCTV, sehingga dapat dipasang di segala sudut ruangan dan dapat menjangkau seluruh area fasilitas umum. Selain itu, alat ini dapat dikontrol oleh orang yang memantau melalui monitor yang menampilkan berbagai tampilan deteksi dari berbagai ruangan secara jarak jauh. Pemantau juga terbantu dalam menjaga penerapan aturan *social distancing* karena proses deteksi berjalan otomatis. Sehingga penerapan *social distancing* pada tempat-tempat umum yang luas dan ramai akan pengunjung dapat dilakukan dengan mudah dan aman bagi orang yang memantau.



Algoritma *deep learning* yang akan digunakan pada alat ini adalah algoritma You Only Look Once V4 (YOLOv4). Algoritma ini digunakan karena didesain untuk proses deteksi objek pada gambar digital, memiliki akurasi yang relatif tinggi dibandingkan algoritma lainnya, serta memiliki kecepatan deteksi yang relatif cepat. Sehingga algoritma YOLOv4 ini sesuai untuk digunakan untuk melakukan deteksi penerapan *social distancing* melalui kamera secara *real-time*. Berdasarkan uraian diatas, penulis mengangkat judul penelitian “SocialCam : Sistem Detektor Penerapan *Social Distancing Real-Time* Menggunakan Kamera Berbasis Algoritma *Deep Learning* YOLOv4 Sebagai Upaya Pencegahan Penyebaran COVID-19 di Fasilitas Umum”.

## 1.2 RUMUSAN MASALAH

Berdasarkan permasalahan diatas, penulis merumuskan rumusan masalah sebagai berikut.

1. Bagaimana algoritma deteksi *social distancing* SocialCam?
2. Bagaimana hasil *training model* deteksi *social distancing* SocialCam?
3. Bagaimana hasil pengujian performa program deteksi *social distancing* SocialCam?

## 1.3 TUJUAN PENELITIAN

Tujuan dari dilakukannya penelitian ini adalah sebagai berikut.

1. Menjelaskan algoritma deteksi *social distancing* SocialCam.
2. Mengetahui hasil *training model* deteksi *social distancing* SocialCam.
3. Mengetahui hasil pengujian performa program deteksi *social distancing* SocialCam.

## 1.4 MANFAAT PENELITIAN

Berikut merupakan manfaat penelitian.

1. Secara Teoritis
  - a) Memberikan sumbangan ilmiah dalam penerapan *deep learning* melalui pengembangan alat deteksi *social distancing*.
  - b) Sebagai bahan referensi bagi penelitian-penelitian selanjutnya yang berhubungan dengan algoritma *deep learning* YOLOv4.
2. Secara Praktis
  - a) Mengembangkan prototipe alat yang berguna untuk meningkatkan kedisiplinan dalam penerapan *social distancing*.
  - b) Menciptakan inovasi baru dalam perkembangan ilmu pengetahuan dalam bidang *deep learning* dan penerapannya.

## BAB 2. TINJAUAN PUSTAKA

### 2.1 PENELITIAN TERDAHULU

Firnanda Al Islama A.P., dkk. (2021) melakukan penelitian dengan judul “*Pengaruh Resolusi Video Terhadap Akurasi Menggunakan Algoritma YOLOv4 dalam Deteksi Citra Objek pada CCTV*”. Penelitian ini bertujuan untuk mengetahui pengaruh resolusi video terhadap akurasi YOLO dalam melakukan klasifikasi dan deteksi objek. Algoritma deteksi objek YOLO diuji dengan resolusi gambar yang berbeda-beda. Resolusi yang digunakan pada penelitian ini adalah 4K, 2K, 1080p, 720p, 480p, 320p, 240p, dan 144p. Akurasi *bounding box* deteksi objek YOLO yang terbaik didapatkan pada video dengan resolusi 4K dengan akurasi mencapai 92% dan akurasi paling buruk didapatkan pada video dengan resolusi 144p karena YOLO tidak dapat melakukan deteksi dalam gambar. Akurasi klasifikasi objek YOLO paling akurat didapatkan pada video dengan resolusi 4K dengan akurasi 100% dan akurasi klasifikasi objek YOLO paling rendah pada resolusi 240p dengan akurasi 71%. Kesimpulannya adalah akurasi objek yang berhasil dikenali oleh YOLO berbanding lurus dengan kualitas resolusi kamera.

Jimin Yu dan Wei Zhang (2021) melakukan penelitian dengan judul “*Face Mask Wearing Detection Algorithm Based on Improved YOLO-v4*”. Penelitian ini berfokus untuk mengembangkan *model* deteksi masker berbasis algoritma YOLOv4 yang disempurnakan. Setelah melalui proses *training* yang memakan waktu 2.834 jam, *model* yang dihasilkan dapat melakukan deteksi pada satu gambar dengan waktu 0.022s dan berjalan pada 54.57 FPS pada deteksi berbasis video.

Penelitian terdahulu digunakan sebagai acuan dalam melakukan penelitian, khususnya dalam mengembangkan *model* deteksi manusia YOLOv4 sebagai dasar program deteksi *social distancing*. Sehingga, *model* yang dihasilkan dapat bekerja dengan optimal dan memiliki akurasi yang baik.

### 2.2 SOCIAL DISTANCING

Salah satu upaya Pemerintah Indonesia dalam mencegah penularan COVID-19 di Indonesia adalah dengan menerapkan peraturan mengenai penerapan protokol kesehatan dalam masa adaptasi kebiasaan baru (*new normal*). Peraturan mengenai protokol kesehatan ini dimuat secara jelas dalam Keputusan Menteri Kesehatan Republik Indonesia Nomor HK.01.07/MENKES/382/2020 yang ditetapkan pada 19 Juni 2020.

Berdasarkan peraturan tersebut, pengelola tempat-tempat umum diwajibkan untuk menerapkan *social distancing*. Hal ini berlaku bagi karyawan maupun bagi pengunjung yang ada. Pengelola juga diwajibkan untuk mengatur jarak antri dan jarak kursi sesuai protokol. Jarak minimal yang direkomendasikan oleh Kementerian Kesehatan RI adalah 1 meter setiap individu.

Alat yang dikembangkan dalam penelitian ini dapat digunakan untuk menunjang penerapan protokol kesehatan pada fasilitas umum dengan algoritma *deep learning* dalam mendeteksi penerapan *social distancing* pada tempat-tempat umum.

### 2.3 ALGORITMA OBJECT RECOGNITION

*Object recognition* atau pengenalan objek, adalah istilah yang merujuk kepada proses komputer yang meliputi identifikasi objek dalam gambar digital. Russakovsky, dkk. (2015) menjelaskan ada tiga jenis *object recognition*.

#### 1. *Image Classification*

*Image classification* merupakan proses yang untuk menentukan kelas dari objek dalam sebuah gambar digital.

## 2. *Object Localization*

*Object localization* merupakan algoritma untuk menentukan lokasi objek dalam sebuah gambar dengan membuat *bounding box* disekeliling objek tersebut.

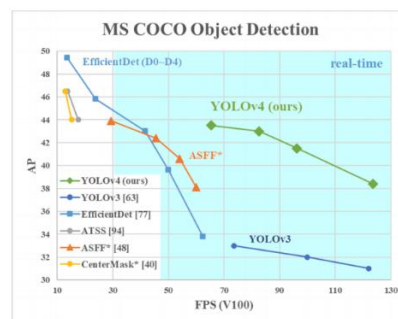
## 3. *Object Detection*

Menentukan lokasi dari sebuah objek dalam sebuah gambar dengan *bounding box* yang digambar disekeliling objek yang terdeteksi disertai dengan nama kelas dari objek yang dideteksi dalam sebuah gambar tersebut.

Berdasarkan uraian diatas, penulis akan menggunakan algoritma *object detection* dalam mendeteksi manusia dalam citra digital dalam melakukan deteksi *social distancing*.

## 2.4 YOLOv4

YOLOv4 adalah sebuah algoritma deteksi objek yang dikembangkan oleh Alexey Bochkovskiy, Chien-Yao Wang, dan Hong-Yuan Mark Liao pada tahun 2020. YOLOv4 merupakan pengembangan lebih lanjut dari algoritma YOLOv3. YOLOv4 mendapatkan predikat *real-time* saat diuji menggunakan dataset MS COCO dengan 43.5 % AP (*average precision*) yang berjalan pada 65 FPS saat dijalankan di kartu grafis Tesla V100. (Bochkovskiy, dkk., 2020) seperti pada Gambar 1.



Gambar 1 MS COCO Benchmark (Alexey Bochkovskiy, 2020).

YOLO (You Only Look Once) adalah salah satu metode deteksi objek yang menggunakan *single convolutional neural network* (CNN) layer yang memprediksikan *bounding box* serta *probability class* secara langsung dalam sekali pengujian (Redmon, dkk., 2016). Algoritma YOLOv4 memakai arsitektur CSPDarknet-53 atau biasa disebut juga dengan CSPNet yang merupakan *backbone* baru yang dapat meningkatkan *learning capability* dari CNN (Tsabita Al Asshifa H.K., 2021). CSPDarknet53 memiliki implementasi seperti CSP, FPN (*Feature Pyramid Network*), SPP (*Spatial Pyramid Pooling*), dan lainnya untuk meningkatkan performa deteksi algoritma (Renjira N.D., 2020).

Salah satu indikator performa deteksi objek YOLOv4 adalah nilai mAP. Nilai mAP merupakan matriks yang mengevaluasi performa dari *model* deteksi objek. Secara sederhana, mAP merupakan rata-rata dari nilai *average precision* (AP) setiap kelas *model* yang berguna untuk mengukur seberapa bagus performa dari *weights* atau *model* hasil *training* data (Kusuma, dkk., 2021).

Penulis akan menggunakan algoritma YOLOv4 dalam mengembangkan *model* deteksi manusia sebagai bagian dari program deteksi *social distancing* pada tempat-tempat umum.

## BAB 3. METODE PENELITIAN

### 3.1 WAKTU DAN TEMPAT

Penelitian ini dilakukan di kediaman penulis, yaitu di Kabupaten Tanah Datar, Provinsi Sumatera Barat dan di Kampus SMAN 1 Sumatera Barat. Hal ini bertujuan untuk mendapatkan data-data yang diperlukan dalam penelitian. Penelitian dilakukan pada rentang waktu bulan Juli – Oktober 2021. Rincian *time table* dari penelitian dapat dilihat pada Lampiran 11.

### 3.2 METODE PENGUMPULAN DATA

Dalam melakukan penelitian ini, penulis menggunakan tiga metode pengumpulan data. Metode-metode tersebut adalah studi literatur, penelitian eksperimen, dan pengujian alat. Studi literatur digunakan sebagai dasar dalam proses pengembangan alat sehingga alat yang dikembangkan menjadi optimal. Penelitian eksperimen merupakan metode yang digunakan untuk mendapatkan pengaruh perlakuan tertentu dengan perlakuan yang lain pada kondisi yang terkendali. Pengujian alat merupakan metode yang dilakukan dengan menguji akurasi *model* deteksi *deep learning* melalui kamera dengan berbagai kondisi berbeda dan menguji algoritma deteksi *social distancing*.

### 3.3 ALUR PENELITIAN

Alur pelaksanaan penelitian secara garis besar dapat dilihat pada Diagram 1.

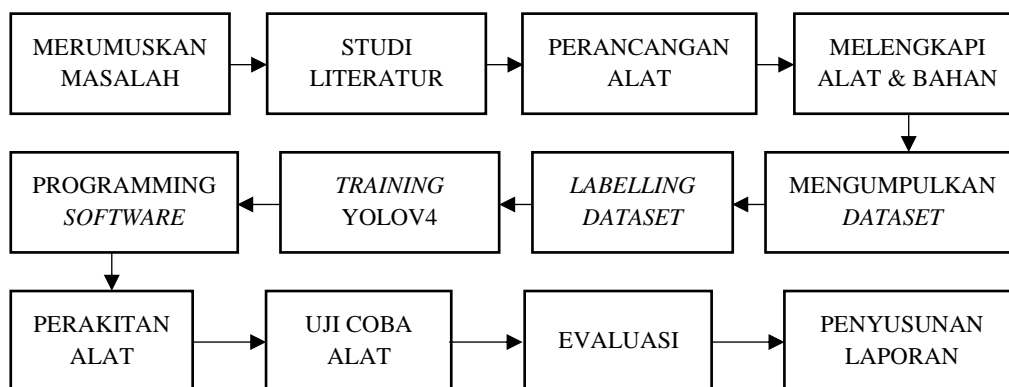


Diagram 1 Alur Penelitian

### 3.4 KONSEP KERJA ALAT

SocialCam merupakan alat yang digunakan untuk mendeteksi penerapan *social distancing* pada tempat-tempat umum melalui kamera secara *real-time*. SocialCam terdiri atas dua komponen utama, yaitu kamera dan komputer yang berfungsi sebagai *server* dalam melakukan deteksi *social distancing*. Penerapan sistem alat dapat berbentuk beberapa *array* atau jaringan kamera yang terhubung dengan satu *server*.

SocialCam pertama-tama akan mendeteksi keberadaan manusia dalam video menggunakan YOLOv4. Keberadaan manusia yang terdeteksi kemudian ditandai dengan kotak (*bounding box*). Program kemudian akan melakukan proses *perspective wrap* terhadap *input*. Program kemudian menentukan jarak dari setiap manusia yang terdeteksi dan melakukan translasi koordinat terhadap *perspective wrap*. Manusia yang berada pada jarak aman ditandai dengan kotak berwarna hijau. Apabila terjadi pelanggaran *social distancing* dan kondisi-kondisi berbunyiya alarm telah terpenuhi, maka kotak akan berwarna merah disertai dengan peringatan suara.

### 3.5 ALAT DAN BAHAN

Alat utama yang akan digunakan dalam penelitian ini adalah komputer, kamera DSLR, kabel HDMI, dan HDMI *capture card*. Komputer akan digunakan dalam pengembangan program, *training model* deteksi manusia berbasis YOLOv4, dan sebagai komputer untuk menjalankan program deteksi *social distancing* saat pengujian alat. Kamera DSLR akan digunakan sebagai kamera pengganti CCTV sementara dalam melakukan pemantauan penerapan *social distancing*. HDMI *capture card* akan digunakan untuk menghubungkan kamera DSLR dengan komputer sehingga hasil kamera dapat dilihat pada komputer. Spesifikasi alat-alat yang akan digunakan disediakan pada Lampiran 3.

### 3.6 SOFTWARE DAN LIBRARY

*Software* dan *library* yang akan digunakan dalam melakukan penelitian ini dapat dilihat pada Tabel 1. *Link* pengunduhan dan detail lengkap *software* dan *library* yang digunakan dapat dilihat pada Lampiran 10.

Tabel 1 Software dan Library

NO	SOFTWARE	LIBRARY
1	Python 3.9	OpenCV
2	PyCharm	Albumentations
3	Darknet	Numpy
4	Bulk Rename Utility	Subprocess
5	VisualStudio 2019	argparse

### 3.7 DATASET

*Dataset* merupakan kumpulan gambar-gambar yang nantinya digunakan sebagai bahan dalam proses *training model* deteksi manusia berbasis YOLOv4. Pada penelitian ini, *dataset* memiliki kelas tunggal (*single-class*), yaitu kelas “*pedestrian*”. Total keseluruhan *dataset* adalah 500 gambar manusia *full-body*.

Kemudian, *dataset* tersebut akan dibagi menjadi dua bagian saat proses *training*. Yaitu *dataset* untuk *training* dan *dataset* validasi. *Dataset* validasi digunakan untuk menguji keakuratan *model* setelah proses *training*. Pada penelitian ini, penulis akan mengalokasikan 113 gambar dari total *dataset* untuk digunakan sebagai validasi *model deep learning*.

*Link* dataset yang digunakan pada penelitian ini terdapat pada Lampiran 5.

### 3.8 METODE PENGUMPULAN DATASET

Dalam proses pengumpulan *dataset*, penulis akan menggunakan beberapa metode. Metode-metode pengumpulan *dataset* adalah sebagai berikut.

#### 1) SMAN 1 Sumatera Barat *Pedestrian Dataset* (SPD)

SMAN 1 Sumatera Barat *Pedestrian Dataset* (SPD) merupakan *dataset* yang berisi gambar-gambar *pedestrian* (pejalan kaki). SPD ini merupakan *dataset* yang penulis kumpulkan sendiri melalui beberapa sumber. Sumber pertama merupakan foto-foto dokumentasi kegiatan sekolah penulis yang diambil dari arsip OSIS SMAN 1 Sumatera Barat. Sumber kedua adalah pengambilan foto langsung pada saat kegiatan PTM terbatas di sekolah penulis, SMAN 1 Sumatera Barat. Dataset ini memiliki total 343 gambar dilengkapi anotasi berjenis YOLO.

## 2) *Penn-Fudan Database for Pedestrian Detection and Segmentation Public Dataset*

Penn-Fudan Database merupakan *dataset* yang digunakan untuk deteksi *pedestrian* atau pejalan kaki. Gambar-gambar dalam *dataset* diambil dari lingkungan kampus University of Pennsylvania, Fudan University, dan jalanan kota. Terdapat 170 gambar dengan 345 *pedestrian*, diantaranya 96 gambar diambil di University of Pennsylvania, dan 74 lainnya diambil di Fudan University. *Dataset* ini memiliki *size* total 51 MB (Computer and Information Science University of Pennsylvania, 2007).

Seluruh gambar dari berbagai sumber *dataset* itu kemudian dijadikan satu. Total keseluruhan adalah 513 gambar.

### 3.9 LABELLING DATASET

*Dataset* gambar yang telah dikumpulkan kemudian akan melalui proses *labelling*. Proses ini bertujuan untuk memberi anotasi koordinat objek yang akan dideteksi kepada setiap gambar *dataset* dan nama kelas objek tersebut. Sehingga pada saat proses *training*, program akan “mempelajari” bagian mana dalam suatu gambar dan termasuk kelas apa bagian dalam gambar tersebut. Proses *labelling* juga digunakan untuk penentuan *ground-truth dataset* validasi untuk pengujian akurasi *model*. Format yang digunakan adalah format anotasi YOLO.

Pada penelitian ini adalah dengan memberi kotak pada setiap bagian pada gambar yang menunjukkan keberadaan manusia. Proses ini akan menghasilkan *file* baru pada setiap gambar *dataset* yang berisi informasi berupa teks. Informasi tersebut adalah informasi koordinat *bounding box* objek yang akan dideteksi serta kelas objek dalam kotak tersebut.

### 3.10 PROSES TRAINING

Proses *training* YOLOv4 akan menggunakan *framework* Darknet yang membutuhkan beberapa langkah pengerjaan. Berikut merupakan langkah-langkah *training model* YOLOv4 yang digunakan.

#### 1. *Proses Install Prerequisite*

*Prerequisite* adalah program-program yang diperlukan sebagai prasyarat dalam melakukan *training*. Beberapa program-program ini adalah Visual Studio, OpenCV, dan *source code* Darknet.

#### 2. *Building DarkNet*

*Building* merupakan proses konversi *source code* Darknet yang telah diunduh sebelumnya menjadi program utuh yang dapat dieksekusi oleh komputer. Proses *building* akan dilakukan menggunakan program Visual Studio 2019.

#### 3. *Konfigurasi Parameter DarkNet*

Konfigurasi parameter DarkNet dilakukan agar *model* yang dihasilkan setelah proses *training* sesuai dengan keinginan. Beberapa parameter yang diatur adalah jumlah kelas, jenis kelas, direktori *dataset training*, direktori *label dataset*, direktori *dataset* validasi, ukuran gambar, dan *pre-trained weight* yang digunakan. *Pre-trained weight* yang digunakan disini adalah YOLOv4.

#### 4. *Training*

Proses *training* akan menghasilkan *model* YOLOv4 *custom* yang akan digunakan untuk mendeteksi keberadaan manusia. Proses *training* juga akan menghasilkan informasi *accuracy* dan *loss* yang berfungsi untuk mengetahui akurasi *training* dari *model*.

### 3.11 TEKNIK ANALISIS DATA

Teknik analisis data yang digunakan adalah teknik deskriptif, teknik komparatif, perhitungan *average precision* (AP), dan perhitungan simpangan baku (*standard deviation*) serta variasi.

#### 1. Teknik deskriptif

Metode penelitian deskriptif dilakukan untuk mengetahui keberadaan variabel independen, baik hanya pada satu variabel atau lebih tanpa membuat perbandingan variabel itu sendiri dan mencari hubungan dengan variabel lain (Sugiyono, 2017). Teknik deskriptif dalam penelitian ini digunakan dalam penjelasan algoritma alat, penjabaran grafik *training loss* dan mekanisme dari proses *training* YOLOv4, dan hasil pengujian langsung alat.

#### 2. Teknik komparatif

Teknik penelitian komparatif merupakan teknik analisis yang membandingkan kondisi satu variabel atau lebih pada dua sampel yang berbeda, atau pada waktu yang berbeda (Sugiyono, 2017). Pada penelitian ini, teknik komparatif digunakan pada pengujian performa SocialCam terhadap resolusi video yang berbeda-beda.

#### 3. Perhitungan *average precision* (AP)

Nilai *average precision* merupakan salah satu metode untuk mengevaluasi performa suatu model deteksi objek. Sedangkan *mean average precision* (mAP), merupakan rata-rata dari nilai *average precision* (AP) dari semua kelas dalam *model* deteksi objek. Sebelum perhitungan mAP, perlu dilakukan pengaturan nilai *threshold* pada IoU untuk menentukan validasi dari objek yang dideteksi. Beberapa nilai mAP yang umum adalah mAP50 dan mAP75 (Kusuma, dkk., 2021).

Penelitian ini menggunakan perhitungan nilai AP karena *model* SocialCam hanya memiliki satu kelas deteksi. Perhitungan nilai AP digunakan untuk menguji akurasi *model pedestrian detector* YOLOv4 SocialCam. Perhitungan nilai AP menggunakan *dataset* validasi yang telah diberi anotasi *ground-truth*. Perhitungan AP dalam penelitian ini menggunakan *intersection over union* (IoU) 0.50 dan IoU 0.75 atau dengan kata lain AP50 dan AP75.

#### 4. Perhitungan variasi data dan simpangan baku (*standard deviation*) data

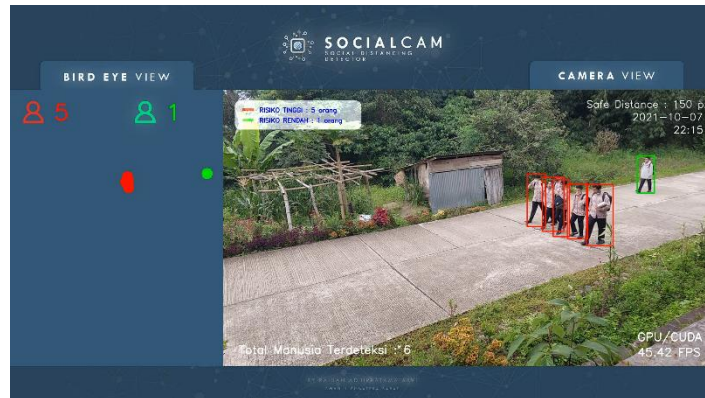
Terdapat beberapa ukuran penyebaran data dalam statistik. Beberapa diantaranya adalah simpangan baku (*standard deviation*) dan variasi data. Variasi data adalah perbandingan antara jumlah dari kuadrat selisih setiap data dan rata-ratanya dengan jumlahnya data. Variasi merupakan ukuran variasi atau ragam dari data-data yang ada. Simpangan baku merupakan kuadrat dari variasi. Simpangan baku mengukur bagaimana penyimpangan dari rata-rata data (Spiegel & Stephens, 2007).

Perhitungan simpangan baku dan variasi data digunakan dalam perhitungan persebaran data jarak maya subjek dalam citra terdistorsi perspektif sebelum translasi dan sesudah translasi *perspective wrap* untuk jarak subjek asli yang sama dari SocialCam.

## BAB 4. HASIL & PEMBAHASAN

### 4.1 ALGORITMA DETEKSI SOCIAL DISTANCING SOCIALCAM

#### 1. Parameter dan Fitur SocialCam



Gambar 2 Tampilan SocialCam

Program SocialCam dijalankan melalui Windows Command Line (CMD) dan memiliki argumen yang akan menentukan bagaimana program berjalan. Tautan video Youtube penjelasan program SocialCam dapat diakses pada Lampiran 8. Link Github SocialCam dapat diakses pada Lampiran 7.

*Command* untuk menjalankan program SocialCam adalah:

```
py SocialCam.py <options> <arguments>
```

Argumen-argumen dan opsi-opsi yang tersedia dalam program SocialCam terdapat dalam Lampiran 12.

#### 2. Kalibrasi dan Konfigurasi Program SocialCam

Ada dua proses kalibrasi yang harus dilakukan agar program SocialCam berjalan semestinya. Kedua proses kalibrasi tersebut adalah kalibrasi jarak dan kalibrasi *perspective point*.

Kalibrasi jarak dilakukan untuk menyetel jarak minimal *social distancing* antar subjek terdeteksi. Hal ini dilakukan dengan kalibrasi manual jarak per piksel (misalnya jarak aman = 80 piksel). Kalibrasi dilakukan dengan mengubah langsung parameter MIN\_DISTANCE pada *file* konfigurasi “socialcam\_config.py”.

Kalibrasi *perspective point* dilakukan untuk menetapkan titik-titik koordinat *warp perspective* untuk proses translasi *warp perspective* atau *bird's eye view*. Kalibrasi dapat dilakukan dengan mengubah langsung parameter SOURCE\_POINTS pada *file* konfigurasi socialcam\_config.py. Cara kedua adalah dengan menggunakan fitur kalibrasi langsung program yang dapat diakses melalui argumen “-getpoints 1”. Argumen ini akan mengaktifkan fitur kalibrasi internal yang memungkinkan pengguna untuk memilih langsung titik perspektif pada gambar.

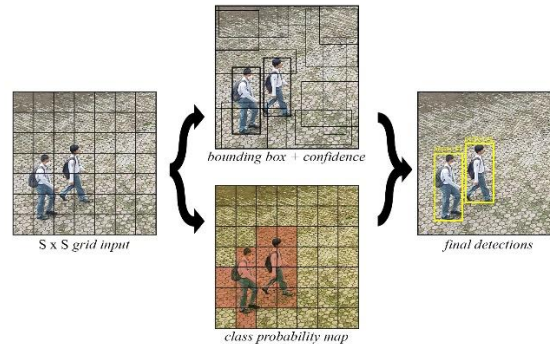
Konfigurasi program SocialCam dapat dilakukan dengan mengubah parameter-parameter yang ada dalam *file config* “socialcam\_config.py”. Terdapat beberapa parameter yang bisa dikonfigurasi



seperti *threshold NMS*, *input default*, *threshold confidence*, dan lain-lain. Penjelasan lengkap mengenai parameter konfigurasi dapat dilihat pada video penjelasan SocialCam di Lampiran 8.

### 3. Deteksi *Pedestrian* Menggunakan YOLO

Algoritma deteksi objek YOLO memiliki beberapa tahapan. Tahapan-tahapan tersebut dapat dilihat pada Gambar 3.



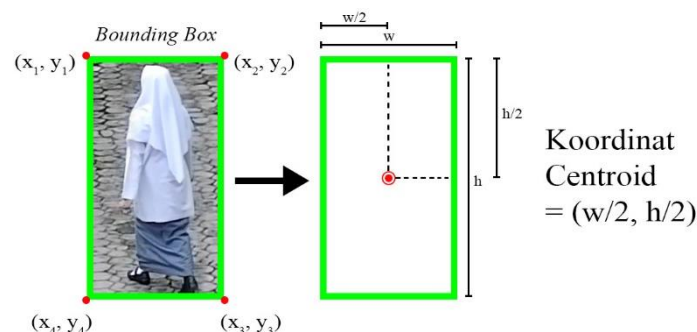
Gambar 3 Algoritma Deteksi Objek YOLO

Pertama, algoritma YOLO membagi input citra menjadi beberapa persegi sejumlah  $S \times S$  *grid*. Masing-masing persegi kemudian akan melakukan proses deteksi objek pada areanya sendiri. Kemudian, setiap persegi *grid* akan melakukan prediksi *bounding box*. Tiap *bounding box* memiliki nilai *confidence rate* yang merupakan persentase seberapa yakin algoritma akan prediksi *bounding box*. Nilai *threshold confidence rate* yang kemudian digunakan sebagai batas *confidence rate* minimal dari *bounding box* sehingga *false prediction* dapat diminimalisir.

YOLO kemudian juga melakukan prediksi kelas dari gambar sesuai kelas yang ada dalam *model*. Pada kasus ini, model YOLO SocialCam merupakan model *single-class* dengan kelas tunggal '*pedestrian*'. Prediksi kelas dilakukan pada persegi-persegi yang diprediksi mengandung objek.

### 4. Penentuan Koordinat Titik Tengah Subjek (*Centroid*)

Penentuan koordinat titik tengah subjek digunakan sebagai langkah awal proses perhitungan jarak subjek. Ilustrasi penentuan titik tengah subjek dapat dilihat pada Gambar 4.

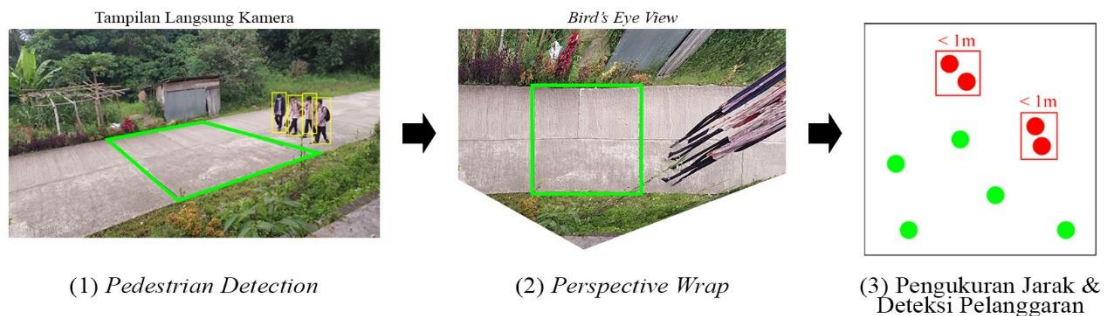


Gambar 4 Penentuan Centroid

Empat titik koordinat *bounding box* hasil deteksi manusia digunakan untuk mencari titik koordinat tengah kotak *bounding box* (*centroid*). Koordinat centroid ini kemudian ditranslasikan melalui gambar perspektif atas (*bird's eye view*) dan akan digunakan untuk menghitung jarak antar subjek terdeteksi.

## 5. *Perspective Wrap (Bird's Eye View)*

Gambar yang diambil melalui kamera memiliki distorsi perspektif atau *foreshortening*. *Foreshortening* adalah ilusi yang membuat suatu jarak di dunia asli yang tampak pada suatu titik pada gambar akan terlihat lebih pendek dibandingkan jarak yang sama di dunia asli pada titik yang berbeda pada gambar. Ilusi *foreshortening* ini akan menjadi masalah dalam perhitungan jarak antarsubjek *social distancing*. SocialCam menggunakan kalkulasi *perspective wrap* untuk mengatasi masalah ini sehingga perhitungan jarak di semua titik pada gambar akan akurat. Proses transformasi *perspective wrap* SocialCam dapat dilihat pada Gambar 5.



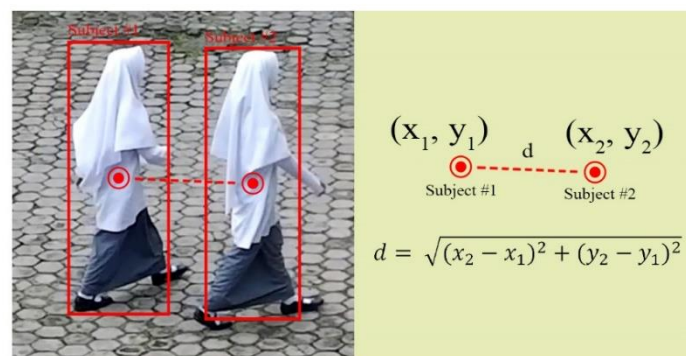
Gambar 5 *Perspective Wrap SocialCam*

*Perspective wrap* akan menjadikan kamera tampak samping menjadi tampak atas virtual (*virtual top-view*) sehingga ukuran objek dalam piksel di segala titik pada gambar adalah sama. Empat titik koordinat pada gambar akan diambil sebagai titik kalibrasi *perspective wrap*. Kemudian, empat titik tersebut akan diubah menjadi persegi sempurna dengan empat titik sudut siku-siku menghasilkan *bird's eye view*.

Koordinat-koordinat *centroid* yang telah didapatkan kemudian akan ditranslasikan sesuai dengan perhitungan *perspective wrap* yang telah dilakukan untuk selanjutnya dilakukan pengukuran jarak dan deteksi pelanggaran. Sehingga, pengukuran jarak dalam gambar akan sama di semua titik pada gambar dan tidak terpengaruh oleh *perspective distortion* atau *foreshortening*.

## 6. Penentuan Jarak antar Subjek

Penentuan jarak antar subjek akan menjadi dasar deteksi pelanggaran *social distancing* dan pemberian peringatan. Jarak antarsubjek adalah jarak *centroids* subjek dalam piksel yang telah ditranslasikan pada gambar perspektif atas (*bird's eye view*). Penentuan jarak antar subjek adalah seperti pada Gambar 6.



Gambar 6 *Penentuan Jarak Subjek*

Jarak antar subjek ditentukan menggunakan rumus *euclidean distance*.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Pelanggaran ditentukan dengan menghitung jarak subjek-subjek yang melebihi batas aman dalam piksel yang telah diatur pada tahapan kalibrasi di awal.

## 7. Pemberian Peringatan

Peringatan program SocialCam berupa suara bel yang dilanjutkan dengan suara peringatan dalam Bahasa Indonesia yang menginstruksikan untuk menjaga jarak. Peringatan suara akan dikeluarkan secara otomatis apabila kondisi peringatan terpenuhi. Algoritma yang penulis kembangkan untuk sistem pemberian alarm SocialCam dapat dilihat pada Diagram 2.

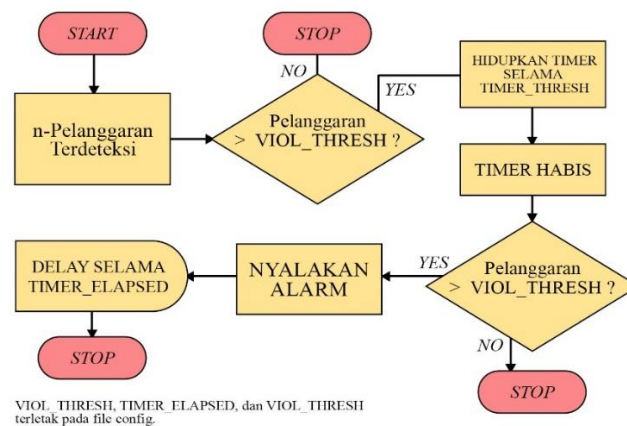


Diagram 2 Algoritma Pemberian Peringatan SocialCam

Ada dua kondisi berbunyiya alarm, yaitu minimal pelanggaran terdeteksi (VIOL\_THRESH) dan lama interval pelanggaran terdeteksi (TIMER\_THRESH). Apabila minimal pelanggaran terlampaui maka *timer* akan hidup. Apabila setelah *timer* telah habis dan minimal pelanggaran masih terlampaui, maka alarm akan hidup. Apabila saat *timer* masih hidup dan minimal pelanggaran kurang dari minimal, maka *timer* akan direset kembali.

Setiap alarm hidup, maka akan ada *timer* (TIMER\_ELAPSED) pasca alarm. Selama TIMER\_ELAPSED ini hidup, alarm tidak dapat hidup kembali. Alarm dapat hidup kembali apabila TIMER\_ELAPSED telah habis. Algoritma ini dirancang untuk meminimalisir *false alarm* dan alarm beruntun.

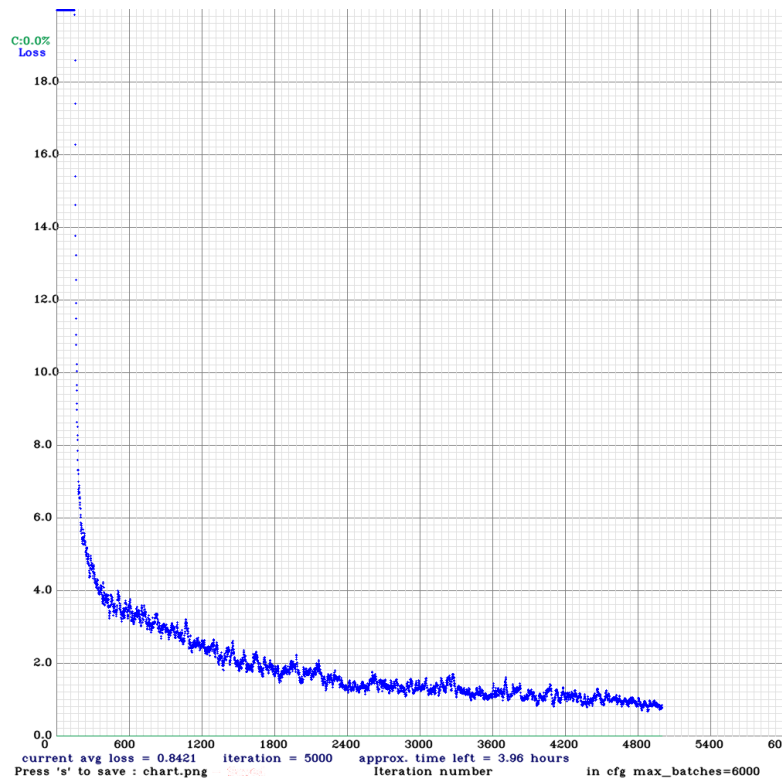
VIOL\_THRESH, TIMER\_THRESH, dan TIMER\_ELAPSED dapat diatur dalam *file* konfigurasi SocialCam “socialcam\_config.py”.

Terdapat 5 buah *file* suara peringatan yang berbeda dalam program ini. Perbedaan terletak pada suara kalimat instruksi untuk menjaga jarak. Apabila kondisi peringatan suara terpenuhi, program SocialCam akan memilih secara acak satu dari lima suara yang tersedia untuk dibunyikan. Hal ini dilakukan agar suara peringatan tidak monoton dan alami.

## 4.2 HASIL *TRAINING MODEL SOCIALCAM*

Proses *training* dilakukan untuk jumlah 5000 iterasi menggunakan kartu grafis NVIDIA GTX1650Ti. Proses *training model* YOLOv4 dimulai pukul 01.20 sampai dengan pukul 17.45 pada hari Sabtu, 2 Oktober 2021. Proses *training* memakan waktu sekitar 16 jam 25 menit. *Hyperparameter* yang digunakan dalam proses *training* ini dapat dilihat pada Lampiran 14.

Gambar 7 berikut merupakan gambar grafik visualisasi *training loss* dari proses *training model pedestrian detection* SocialCam menggunakan Darknet dengan jumlah 5000 iterasi.



Gambar 7 Nilai Training Loss Model SocialCam

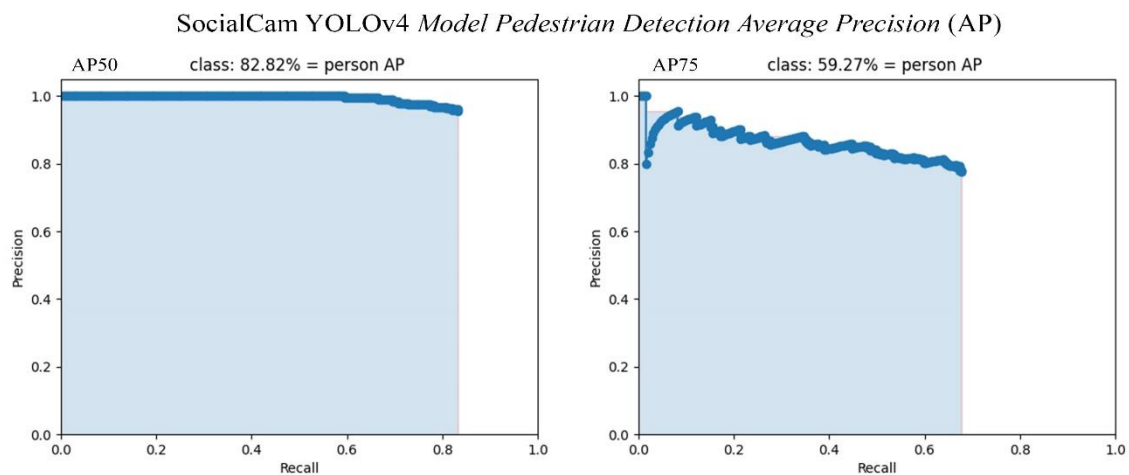
Berdasarkan grafik diatas, proses *training model* memiliki *training loss*  $< 2$  pada iterasi ke-1800 dan terus melandai. Titik *training loss* terendah didapatkan pada iterasi ke-5000 dengan nilai *training loss* sebesar 0.8421.

### 4.3 PENGUJIAN PERFORMA SOCIALCAM

Pengujian performa dibagi tiga, pengujian akurasi deteksi *pedestrian*, pengujian FPS, dan pengujian pengukuran jarak. Pengujian dilakukan dengan menggunakan *file dataset* yang telah disiapkan sebelumnya. Pengujian akurasi menggunakan *file dataset* validasi yang berjumlah 113 gambar. Pengujian performa FPS menggunakan *file dataset* video yang berjumlah 7 video. Sedangkan pengujian pengukuran jarak menggunakan *file dataset* pengukuran jarak.

#### 1. Pengujian Akurasi Deteksi *Pedestrian* SocialCam.

Pengujian dilakukan terhadap *dataset* validasi yang berjumlah 113 gambar. Gambar *dataset* telah melalui tahap anotasi atau *labelling* untuk penentuan *ground-truth* dari seluruh gambar *dataset*. Kemudian, dilakukan tahap deteksi dan ekstraksi *bounding box* menggunakan model deteksi *pedestrian* SocialCam. Menggunakan hasil deteksi ditentukan grafik nilai *precision* dan *recall* dari deteksi *single-class model*. Melalui grafik kemudian dihitung *average precision* (AP) dari *model* dengan *intersection over union* (IoU) = 0.50 dan IoU = 0.75. Tabel *precision* dan *recall* hasil pengujian AP dapat dilihat pada Gambar 8.



Gambar 8 Nilai Average Precision (AP) SocialCam. AP50 (kiri) dan AP75 (kanan).

Hasil pengujian *model pedestrian detection* SocialCam untuk *single-class* '*pedestrian*' dengan IoU = 0.50 menghasilkan nilai AP50 sebesar 82,82%. Sedangkan pengujian yang dilakukan menggunakan IoU = 0.75 menghasilkan nilai AP75 sebesar 59,27%.

Tautan pengunduhan *file weight* SocialCam dapat diakses pada tautan di Lampiran 7. Anotasi koordinat *bounding box* hasil deteksi, dan *dataset* validasi beserta anotasi *ground-truth* yang digunakan berturut-turut dapat diakses pada Lampiran 5.

## 2. Pengujian Performa Program SocialCam terhadap Variasi Resolusi Video *Input*.

Pengujian dilakukan menggunakan cuplikan video sampel SMANSSU\_ASRAMA\_2.mp4 yang di-cut pada *frame* 00.20 - 01.20 sehingga memiliki panjang 1 menit dengan jumlah total 1799 *frame*. Semua video sampel ini berasal dari satu *file* video utama dengan resolusi 4k (3840 x 2160 piksel). *File* video utama ini kemudian melalui proses *resizing* sehingga menghasilkan 7 video dengan resolusi yang lebih rendah. Penjelasan mengenai *file* video yang digunakan dapat dilihat pada Lampiran 9.

Tabel 2 berikut menunjukkan hasil pengujian performa SocialCam terhadap beberapa jenis resolusi video.

Tabel 2 Hasil Pengujian Performa FPS SocialCam

NO	Resolusi	GPU / CUDA*		CPU*	
		Ave. FPS	Ave. FPS PROC	Ave. FPS	Ave. FPS PROC
1	2160p	19,66	11,78	11,69	6,27
2	1440p	29,04	19,51	14,28	7,23
3	1080p	37,42	26,67	15,23	7,66
4	720p	44,47	32,95	16,45	8,03
5	480p	48,77	37,06	16,70	8,19
6	360p	51,91	39,87	17,08	8,28
7	240p	54,93	42,84	17,22	8,25

\*CPU AMD Ryzen 7 4800H 8GB RAM; GPU NVIDIA GTX1650Ti 4GB

Grafik visualisasi dari hasil pengujian dapat dilihat pada Lampiran 13.

Perhitungan FPS dibagi menjadi dua jenis. FPS merupakan perhitungan *frame per second* murni deteksi *pedestrian* atau manusia dari *model* YOLOv4. Sedangkan FPS *processing* (FPS PROC.) merupakan perhitungan *frame per second* total program SocialCam (waktu deteksi + waktu *processing* fitur-fitur program).

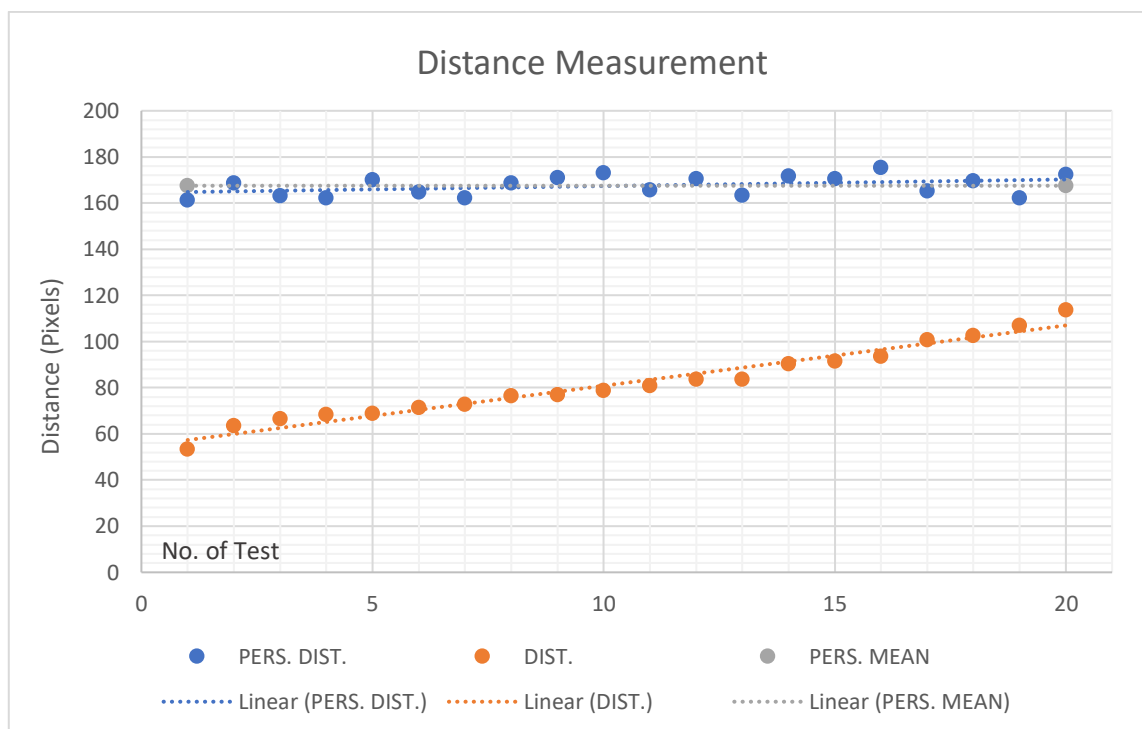
Berdasarkan hasil pengujian, semakin tinggi resolusi maka semakin rendah FPS program. Waktu pemrosesan diluar waktu deteksi murni juga memiliki dampak yang signifikan terhadap FPS program. FPS total tertinggi terdapat pada resolusi 240p GPU / CUDA dengan 42,84 FPS dan FPS total terendah terdapat pada resolusi 2160p CPU yaitu 6,27 FPS.

### 3. Pengujian Algoritma Pengukuran Jarak dan *Perspective Wrap* SocialCam

Pengujian deteksi jarak dilakukan untuk menguji akurasi translasi *wrap perspective* atau *bird's eye view* program terhadap distorsi perspektif kamera. Pengujian dilakukan menggunakan *dataset* pengujian yang diambil dari lantai atas sekolah penulis diarahkan ke arah lapangan sehingga terdapat distorsi perspektif kamera yang cukup signifikan. Dataset yang dipakai dapat diakses pada Lampiran 5.

Pengujian dilakukan terhadap 20 gambar yang setiap gambarnya berisi dua subjek yang memiliki jarak yang sama (1 meter) di dunia asli pada setiap foto. Namun, pada setiap gambar subjek berdiri pada area gambar yang berbeda-beda dengan efek *foreshortening* yang berbeda-beda. Sehingga jarak maya subjek pada gambar digital pun berbeda-beda walau jarak di dunia asli adalah sama. Hasil yang diharapkan adalah jarak maya subjek setelah translasi memiliki variasi data yang serendah-rendahnya.

Hasil pengujian pengukuran jarak langsung dari gambar dan pengukuran jarak setelah ditranslasikan menggunakan *warp perspective* dapat dilihat pada Gambar 9.



Gambar 9 Grafik Pengujian Pengukuran Jarak SocialCam

Pada pengujian dengan 20 sampel gambar, jarak antara dua subjek terdekat memiliki jarak 53,32 px dan jarak terjauh 113,58 px. *Range* data jarak langsung pengujian adalah 60,26. Varian data adalah senilai 235,89 dengan *standard deviation* sebesar 15,36.

Setelah melalui tahap translasi *perspective wrap*, jarak antara dua subjek terdekat adalah 161,23 px dan jarak terjauh 172,32 px dengan *range* data 11,09. Varian data sebesar 17,75 dengan *standard deviation* sebesar 4,21.

Berdasarkan data tersebut, varian data untuk pengukuran jarak telah menurun signifikan dari 17,75 ke 4,21 untuk pengukuran dua subjek dengan jarak 1 meter pada 20 gambar dengan 20 titik pengujian yang berbeda setelah ditranslasikan menggunakan *perspective wrap*.

Hal ini menunjukkan bahwa tahapan *perspective wrap* berpotensi dalam mengatasi masalah *foreshortening* dalam pengukuran jarak dalam gambar yang memiliki distorsi perspektif.



## BAB 5. KESIMPULAN & SARAN

### 5.1 KESIMPULAN

Setelah dilaksanakannya pengembangan *software* SocialCam, pengumpulan *dataset*, proses *training* menggunakan Darknet, evaluasi performa SocialCam terhadap berbagai resolusi video, pengujian nilai *average precision model*, dan pengujian translasi jarak *perspective wrap* atau *bird's eye view* program SocialCam, maka didapatkanlah kesimpulan sebagai berikut ini.

1. Algoritma SocialCam memiliki tahapan-tahapan penting dalam mendeteksi pelanggaran *social distancing*. Berikut ini adalah kesimpulan pada masing-masing indikator.
  - a. Eksekusi program SocialCam memiliki opsi dan argumen pada *argument parser* sebagai penentu bagaimana jalannya program.
  - b. SocialCam harus dikalibrasi terlebih dahulu sebelum dijalankan agar berjalan dengan baik. Adapun kalibrasi yang harus dilakukan adalah kalibrasi titik perspektif dan kalibrasi jarak minimum.
  - c. Deteksi *pedestrian* SocialCam menggunakan YOLOv4 *single-class* dengan kelas '*pedestrian*'.
  - d. Algoritma *perspective wrap* digunakan untuk mengatasi masalah *foreshortening* dalam pengukuran jarak subjek dalam gambar.
  - e. SocialCam memiliki alarm peringatan berupa suara. Kondisi berbunyinya alarm adalah minimal pelanggaran dan lama minimal pelanggaran terdeteksi.
2. Proses *training* SocialCam menghasilkan nilai *training loss model* pada titik iterasi ke-5000 sebesar 0.8421. Proses *training* menggunakan Darknet dengan jumlah 5000 iterasi memakan waktu 16 jam 25 menit.
3. Pengujian performa SocialCam dibagi tiga jenis. Berikut ini adalah kesimpulan pada masing-masing indikator.
  - a. Hasil pengujian akurasi menghasilkan nilai *average precision (AP) model* SocialCam sebesar 82,82% untuk AP50 dan sebesar 59,27% untuk AP75.
  - b. Resolusi video sangat berpengaruh pada performa program. FPS tertinggi didapatkan saat program dijalankan menggunakan GPU dan berada pada resolusi video terendah.
  - c. Algoritma *perspective wrap* dapat mengatasi masalah distorsi perspektif. Hal ini dibuktikan dengan menurunnya *standard deviation* pengukuran jarak sebelum *perspective wrap* yaitu 15,36 menjadi 4,21 setelah ditranslasikan menggunakan *perspective wrap*.

### 5.2 SARAN

Adapun saran-saran untuk pelaksanaan penelitian lebih lanjut adalah sebagai berikut.

1. Perlunya pengembangan alarm peringatan yang bekerja spesifik terhadap kelompok atau individu. Misalnya peringatan yang berupa teguran kepada individu yang menggunakan baju berwarna tertentu atau kelompok yang berada di area tertentu. Sehingga peringatan tidak bersifat umum.
2. Pengembangan algoritma pengukuran jarak yang menggunakan gabungan konfigurasi kamera *stereo* dan *perspective wrap* dalam peningkatan akurasi pengukuran jarak.



## UCAPAN TERIMA KASIH

Penulisan laporan hasil penelitian ini tidak terlepas dari bantuan dan dukungan dari berbagai pihak. Oleh karena itu, penulis menyampaikan ucapan terima kasih dan penghargaan kepada pihak-pihak yang telah membantu:

1. Umi Nilma Herrita Wisda Syam, S.Pd., M.Si. selaku guru pembimbing dalam melakukan penelitian ini yang senantiasa memberikan bimbingan kepada penulis.
2. Bapak Budi Hermawan, S.Pd., M.Si. selaku kepala sekolah yang berperan sebagai penanggung jawab dalam melakukan penelitian ini.
3. Bapak Hasdi Putra, S.T., M.T. selaku Dosen Fakultas Teknologi Informasi Universitas Andalas yang telah memberikan saran-saran atas pelaksanaan penelitian.
4. Pihak Kampus Fakultas Teknologi Informasi Universitas Andalas Kota Padang.
5. Bapak Yan Restu Freski, S.T., M.Eng. selaku Agen di Sekolah Ilmuan Minangkabau (SIM) yang senantiasa memberikan waktu luangnya untuk membimbing dan membina penulis dalam melakukan penelitian ini.
6. Maharaka Fadhilah, Fadil Hidayat dan Farhan Multazam Faras, teman-teman penulis yang telah membantu penulis dalam pelaksanaan penelitian.
7. Semua pihak-pihak lainnya yang penulis tidak dapat sebutkan satu persatu yang telah membantu dalam penyelesaian penelitian ini.

Penulis memohon maaf jika ada hal yang kurang pada penelitian ini. Penulis telah melakukan penelitian dan penulisan laporan ini dengan maksimal. Tetapi masih terdapat beberapa keterbatasan dalam pelaksanaan penelitian. Penulis memohon maaf apabila ada hal yang kurang berkenaan. Semoga penelitian ini membawa segenap manfaat bagi para pembaca dan nantinya bisa menjadi acuan dalam pengembangan ilmu pengetahuan Indonesia.

## REFERENSI

- Bochkovskiy, Alexey, Chien-Yao W., Hong-Yuan M.L. 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv:2004.10934 [cs.CV].
- Computer and Informatics Science University of Pennsylvania. 2007. Penn-Fudan Database for Pedestrian Detection and Segmentation. [https://www.cis.upenn.edu/~jshi/ped\\_html/](https://www.cis.upenn.edu/~jshi/ped_html/) [20 September 2021 pukul 16.30].
- Dhiaegana, Renjira Naufhal, dan Dr. Ir. Rinaldi, M.T. 2020. Penerapan Convolutional Neural Network Untuk Deteksi Pedestrian Pada Sistem Autonomous Vehicle [Skripsi]. Bandung: Sekolah Teknik Elektro dan Informatika (STEI), Institut Teknologi Bandung.
- Kementerian Kesehatan Republik Indonesia. 2020. *Keputusan Menteri Kesehatan Republik Indonesia Nomor HK.01.07/MENKES/382/2020 tentang Protokol Kesehatan bagi Masyarakat di Tempat dan Fasilitas Umum dalam Rangka Pencegahan dan Pengendalian Corona Virus Disease 2019 (COVID-19)*. Jakarta: Sekretariat Negara.
- Kusuma, T. A. A. H., Usman, K., & Saidah, S. (2021). People Counting for Public Transportations Using You Only Look Once Method. *Jurnal Teknik Informatika*, 2(1): 57-66. <https://doi.org/10.20884/1.jutif.2021.2.2.77>
- Putra, Firnanda Al Islama Achyunda, Jeffry A.F., Nadim A., Fitra A.B., Novanto Y. 2020. Pengaruh Resolusi Video Terhadap Akurasi Menggunakan Algoritma YOLOv4 dalam Deteksi Citra Objek pada CCTV.
- Redmon, J., Divvala, S., Girshick, R., dan Farhadi, A. 2016. You Only Look Once: Unified, Real-Time Object Detection. *IEEE Conference on Computer Vision and Pattern Recognition*: 779-788. doi: 10.1109/CVPR.2016.91
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. dan Berg, A.C. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3): 211-252. [arXiv:1409.0575v3](https://arxiv.org/abs/1409.0575v3) [cs.CV].
- Satuan Tugas Penanganan COVID-19. 2021. Peta Sebaran COVID-19. <https://covid19.go.id/peta-sebaran-covid19> [26 Mei 2021 pukul 09.47].
- Spiegel, R.M. dan Stephens, L.J. 2007. *Schaum's Outline of Statistics 4th Edition*. New York: McGraw-Hill.
- Sugiyono. 2017. *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Bandung : Alfabeta.
- Yu, J., & Zhang, W. 2021. Face mask wearing detection algorithm based on improved YOLO-v4. *Sensors*, 21(9): 3263. <https://doi.org/10.3390/s21093263>

## **LAMPIRAN**

- **DAFTAR LAMPIRAN**

**LAMPIRAN 1. SURAT PERNYATAAN ORISINALITAS**

**LAMPIRAN 2. SURAT TUGAS PEMBIMBING DARI KEPALA SEKOLAH**

**LAMPIRAN 3. SPESIFIKASI ALAT YANG DIGUNAKAN**

**LAMPIRAN 4. ALAT & BAHAN**

**LAMPIRAN 5. LINK DATASET**

**LAMPIRAN 6. DOKUMENTASI PENELITIAN**

**LAMPIRAN 7. LINK GOOGLE DRIVE & GITHUB FILE PROGRAM SOCIALCAM**

**LAMPIRAN 8. LINK VIDEO SOCIALCAM**

**LAMPIRAN 9. PENGUJIAN RESOLUSI VIDEO**

**LAMPIRAN 10. LINK SOFTWARE YANG DIGUNAKAN**

**LAMPIRAN 11. TIME TABLE PENELITIAN**

**LAMPIRAN 12. DOKUMENTASI PROGRAM SOCIALCAM**

**LAMPIRAN 13. GRAFIK PENGUJIAN PERFORMA FPS**

**LAMPIRAN 14. HYPERPARAMETER PROSES TRAINING**

## LAMPIRAN 1. SURAT PERNYATAAN ORISINALITAS

### PERNYATAAN PENELITIAN

Yang bertanda tangan di bawah ini:

Nama : Raihan Adhipratama Arvi  
Tempat/Tanggal Lahir : Padang / 16 Juli 2003  
NISN : 0037054239  
Asal Sekolah : SMAN 1 Sumatera Barat

dengan ini menyatakan sejujurnya bahwa laporan penelitian saya dengan judul

SocialCam : Sistem Detektor Penerapan *Social Distancing Real-Time* Menggunakan Kamera Berbasis Algoritma *Deep Learning* YOLOv4 Sebagai Upaya Pencegahan Penyebaran COVID-19 di Fasilitas Umum.

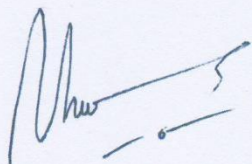
**bersifat orisinal/bukan plagiasi/belum pernah dikompetisikan dan/atau pernah dilombakan tetapi belum mendapat juara/penghargaan di tingkat Nasional/Internasional**

Bilamana di kemudian hari ditemukan ketidaksesuaian dengan pernyataan ini, saya bersedia menerima konsekuensi sesuai aturan KoPSI.

Demikian pernyataan ini dibuat dengan sesungguhnya dan dengan sebenar-benarnya.

Dibuat di: Kota Padang Panjang  
Pada Tanggal: 02 September 2021

Mengetahui,



Guru Pembimbing,  
Nilma Herrita W. S., S.Pd, M.Si  
NIP: 19730222 200501 2 005

Yang membuat pernyataan,



Raihan Adhipratama Arvi  
NISN: 0037054239

## LAMPIRAN 2. SURAT TUGAS PEMBIMBING DARI KEPALA SEKOLAH



PEMERINTAH PROVINSI SUMATERA BARAT  
DINAS PENDIDIKAN  
CABANG DINAS WILAYAH I  
**SMA NEGERI 1 SUMATERA BARAT**

Jl. S Ibrahim Musa RT.009, Sigando, Padang Panjang Timur  
Kota Padang Panjang – Sumatera Barat. 27126  
Telp: (0752) 485544 Website : [www.sman1sumbar.sch.id](http://www.sman1sumbar.sch.id)



### SURAT TUGAS

Nomor : 800 ./0464.a/SMA.01/SB/2021

Dasar : Perintah Kepala Sekolah

Pertimbangan : Demi Kelancaran Kegiatan.

MEMERINTAHKAN KEPADA :

No	Nama/NIP	Pangkat/gol	Jabatan	Ket
1.	NILMA HERRITA WS, S.Pd,M.Si NIP 19730222 200501 2 005	Penata TK I,III/d	Guru	

Untuk : Menjadi Pembimbing Siswa ( Raihan Adhipratama Arvi) dalam Kegiatan  
Lomba KOPSI 2021

Demikian Surat Tugas ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Ditetapkan di : Padang Panjang  
Pada tanggal : 02 Juni 2021



**BUDI HERMAWAN, S.Pd., M.Si**  
Pembina Tk.I  
NIP 19700108 199201 1 002

### LAMPIRAN 3. SPESIFIKASI ALAT YANG DIGUNAKAN

#### 1. KOMPUTER LAPTOP

SPESIFIKASI	NILAI
Merk	ASUS TUF Gaming A15 Laptop
<i>Operating System</i>	Windows 10 Home 64-bit
Kartu Grafis	NVIDIA GeForce® GTX 1650 Ti 128 bits
VRAM VGA	4 GB GDDR6
CPU	AMD Ryzen 7 4800H 2.90 GHz
RAM	8 GB DDR4 3200MHz

#### 2. KAMERA DSLR

SPESIFIKASI	NILAI
Merek	Canon EOS 600D
<i>Focal Length Lensa</i>	18 – 55 mm
Ukuran Sensor	22.3 x 14.9 mm
Ukuran Gambar Maksimal (tanpa <i>crop</i> )	5184 x 3456 <i>pixel</i>

#### 3. HDMI CAPTURE CARD


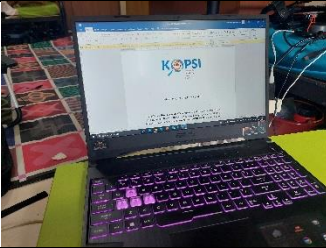




SPESIFIKASI	NILAI
<i>Max Input Resolution</i>	3840×2160@30Hz
<i>Support Video Format</i>	8/10/12bit <i>Deep Color</i>
<i>Max Output Resolution</i>	1920×1080@30Hz
<i>Video Output Format</i>	YUV / JPEG
<i>Input Cable Distance</i>	≤15m AWG26 HDMI <i>standard cable</i>

#### 4. KAMERA SMARTPHONE


SPEKIFIKASI	NILAI
Merek	Samsung Galaxy A52 LTE
<i>Focal Length</i> Lensa	26 mm
Ukuran Sensor	1/1.7X"
Bukaan Lensa	f/1.8
Ukuran Pixel	0.8µm
Resolusi Gambar Maksimal	64 MP (Quad-Bayer)
Resolusi Video Maksimal	4K 30 FPS
Fitur Kamera Utama	OIS, EIS








#### LAMPIRAN 4. ALAT & BAHAN



NO	ALAT	FUNGSI	FOTO
1	Samsung Galaxy A52 LTE	<ul style="list-style-type: none"> <li>Mengambil video sampel pengujian <i>social distancing</i>.</li> </ul>	
2	Laptop Asus TUF Gaming A15	<ul style="list-style-type: none"> <li>Memprogram program SocialCam.</li> <li>Membuat makalah.</li> </ul>	
3	DSLR Canon EOS 600D	<ul style="list-style-type: none"> <li>Dokumentasi penelitian.</li> <li>Pembuatan <i>dataset</i>.</li> </ul>	
4	Tripod	<ul style="list-style-type: none"> <li>Sebagai penyangga <i>smartphone</i> dalam pengambilan video sampel untuk penelitian.</li> </ul>	
5	Holder Smartphone	<ul style="list-style-type: none"> <li>Penyangga <i>smartphone</i> pada <i>tripod</i>.</li> </ul>	
6	Acasis HDMI Capture Card	<ul style="list-style-type: none"> <li>Menyambungkan kamera pada laptop sebagai <i>webcam</i>.</li> </ul>	



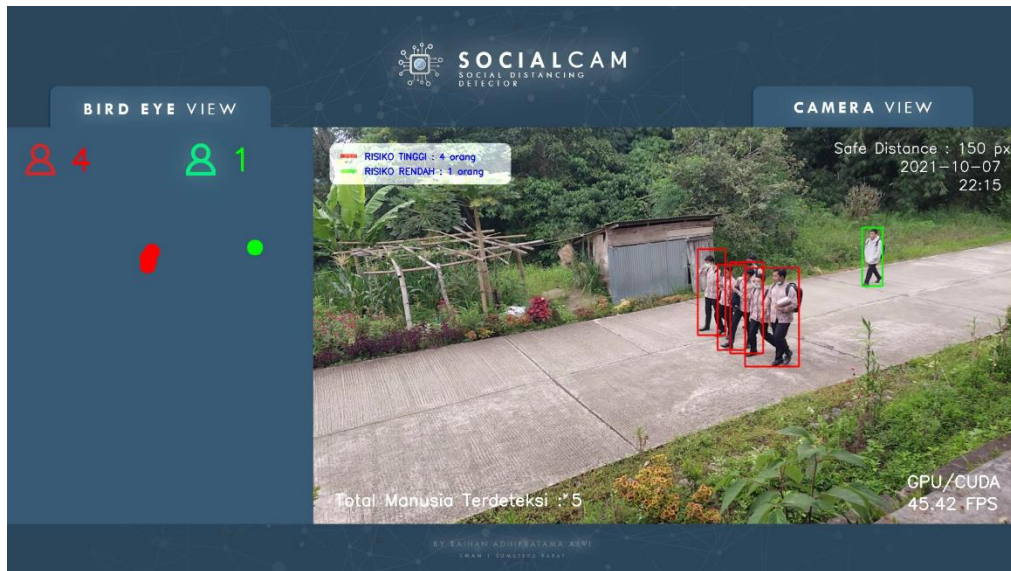
7	Kabel HDMI	<ul style="list-style-type: none"> <li>Menyambungkan kamera DSLR pada <i>capture card</i> sebagai <i>webcam</i>.</li> </ul>	
---	------------	---	---

## LAMPIRAN 5. LINK DATASET

NO	DATASET	LINK	
		LINK LANGSUNG	QR CODE
1	SMA Negeri 1 Sumatera Barat (SMANSSU) Pedestrian Dataset	<a href="https://drive.google.com/file/d/1nYEi4ssqQJKIBVYZlBaB_TlZMdupmcyh/view?usp=sharing">https://drive.google.com/file/d/1nYEi4ssqQJKIBVYZlBaB_TlZMdupmcyh/view?usp=sharing</a>	
2	Penn-Fudan Database for Pedestrian Detection and Segmentation	<a href="https://www.cis.upenn.edu/~jshi/ped_html/">https://www.cis.upenn.edu/~jshi/ped_html/</a>	
3	Dataset Pengujian Performa SocialCam terhadap Variasi Resolusi Video	<a href="https://drive.google.com/drive/folders/1dLADc-Kj8qgErqi1HtJGn3mdPb1BGaQn?usp=sharing">https://drive.google.com/drive/folders/1dLADc-Kj8qgErqi1HtJGn3mdPb1BGaQn?usp=sharing</a>	
4	SMAN 1 Sumatera Barat Pedestrian Video Dataset [Part 1]	<a href="https://drive.google.com/drive/folders/10aps1bqWaQkJzHWxZH2gzuC47Bv_9Ca1?usp=sharing">https://drive.google.com/drive/folders/10aps1bqWaQkJzHWxZH2gzuC47Bv_9Ca1?usp=sharing</a>	
5	SMAN 1 Sumatera Barat Pedestrian Video Dataset [Part 2]	<a href="https://drive.google.com/drive/folders/1Slu5ZJ0Ds0kAHzaRySUG5smqzEF0VmFC?usp=sharing">https://drive.google.com/drive/folders/1Slu5ZJ0Ds0kAHzaRySUG5smqzEF0VmFC?usp=sharing</a>	

6	SMAN 1 Sumatera Barat Pedestrian Video Dataset [Part 3]	<a href="https://drive.google.com/drive/folders/17Z-Hftyv5MZwNL98iyIepM_QsFQfCyDV?usp=sharing">https://drive.google.com/drive/folders/17Z-Hftyv5MZwNL98iyIepM_QsFQfCyDV?usp=sharing</a>	
7	Dataset Pengujian Jarak	<a href="https://drive.google.com/drive/folders/1VgaHluoCFG-Vd0vg9ZJuUgFxfjguUMegV?usp=sharing">https://drive.google.com/drive/folders/1VgaHluoCFG-Vd0vg9ZJuUgFxfjguUMegV?usp=sharing</a>	




## LAMPIRAN 6. DOKUMENTASI PENELITIAN



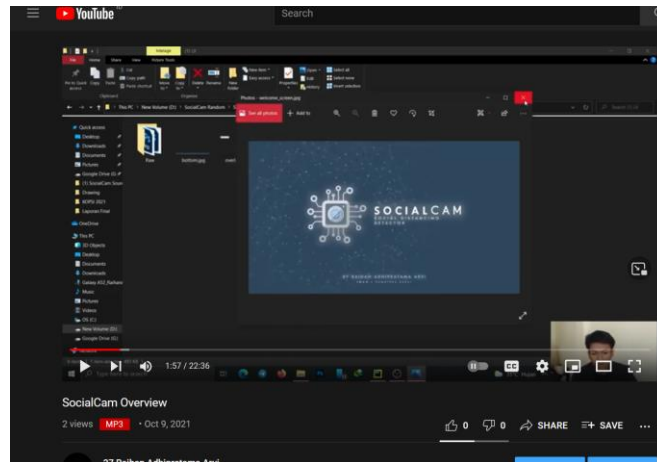
```
File Edit View Navigate Code Refactor Run Tools VCS Window Help SocialCamPy - SocialCamPy
Project SocialCamPy
SocialCamPy
External Libraries
Scratches and Consoles


1 #####
2 ### SocialCam Social Distancing Detection Program
3 ### By: Erlina Edirretana Arvi
4 ### SHAM 3 Sumatera Barat
5 ### 8 October 2022
6 #####
7
8 #####
9
10 # Packages
11 import cv2
12
13 #####
14
15 # ## FUNCTIONS
16 #####
17
18 def draw_circle(event, x, y, flags, param):
19     global ix, iy
20     if event == cv2.EVENT_LBUTTONDOWN:
21         cv2.circle(img, (x, y), 2, (255, 0, 0), 1)
22         ix, iy = x, y
23
24 def get_points(image, numPoints, image_size=(800, 800)):
25     global img
26     img = image.copy()
27     img = cv2.resize(img, image_size)
28     width, height = img.shape[:2]
29     cv2.namedWindow("Image")
30     cv2.setMouseCallback("Image", draw_circle)
31     points = []
32
33 #####
34
35 #####
36
37 #####
38
39 #####
40
41 #####
42
43 #####
44
45 #####
46
47 #####
48
49 #####
50
51 #####
52
53 #####
54
55 #####
56
57 #####
58
59 #####
60
61 #####
62
63 #####
64
65 #####
66
67 #####
68
69 #####
70
71 #####
72
73 #####
74
75 #####
76
77 #####
78
79 #####
80
81 #####
82
83 #####
84
85 #####
86
87 #####
88
89 #####
90
91 #####
92
93 #####
94
95 #####
96
97 #####
98
99 #####
100
101 #####
102
103 #####
104
105 #####
106
107 #####
108
109 #####
110
111 #####
112
113 #####
114
115 #####
116
117 #####
118
119 #####
120
121 #####
122
123 #####
124
125 #####
126
127 #####
128
129 #####
130
131 #####
132
133 #####
134
135 #####
136
137 #####
138
139 #####
140
141 #####
142
143 #####
144
145 #####
146
147 #####
148
149 #####
150
151 #####
152
153 #####
154
155 #####
156
157 #####
158
159 #####
160
161 #####
162
163 #####
164
165 #####
166
167 #####
168
169 #####
170
171 #####
172
173 #####
174
175 #####
176
177 #####
178
179 #####
180
181 #####
182
183 #####
184
185 #####
186
187 #####
188
189 #####
190
191 #####
192
193 #####
194
195 #####
196
197 #####
198
199 #####
200
201 #####
202
203 #####
204
205 #####
206
207 #####
208
209 #####
210
211 #####
212
213 #####
214
215 #####
216
217 #####
218
219 #####
220
221 #####
222
223 #####
224
225 #####
226
227 #####
228
229 #####
230
231 #####
232
233 #####
234
235 #####
236
237 #####
238
239 #####
240
241 #####
242
243 #####
244
245 #####
246
247 #####
248
249 #####
250
251 #####
252
253 #####
254
255 #####
256
257 #####
258
259 #####
260
261 #####
262
263 #####
264
265 #####
266
267 #####
268
269 #####
270
271 #####
272
273 #####
274
275 #####
276
277 #####
278
279 #####
280
281 #####
282
283 #####
284
285 #####
286
287 #####
288
289 #####
290
291 #####
292
293 #####
294
295 #####
296
297 #####
298
299 #####
300
301 #####
302
303 #####
304
305 #####
306
307 #####
308
309 #####
310
311 #####
312
313 #####
314
315 #####
316
317 #####
318
319 #####
320
321 #####
322
323 #####
324
325 #####
326
327 #####
328
329 #####
330
331 #####
332
333 #####
334
335 #####
336
337 #####
338
339 #####
340
341 #####
342
343 #####
344
345 #####
346
347 #####
348
349 #####
350
351 #####
352
353 #####
354
355 #####
356
357 #####
358
359 #####
360
361 #####
362
363 #####
364
365 #####
366
367 #####
368
369 #####
370
371 #####
372
373 #####
374
375 #####
376
377 #####
378
379 #####
380
381 #####
382
383 #####
384
385 #####
386
387 #####
388
389 #####
390
391 #####
392
393 #####
394
395 #####
396
397 #####
398
399 #####
400
401 #####
402
403 #####
404
405 #####
406
407 #####
408
409 #####
410
411 #####
412
413 #####
414
415 #####
416
417 #####
418
419 #####
420
421 #####
422
423 #####
424
425 #####
426
427 #####
428
429 #####
430
431 #####
432
433 #####
434
435 #####
436
437 #####
438
439 #####
440
441 #####
442
443 #####
444
445 #####
446
447 #####
448
449 #####
450
451 #####
452
453 #####
454
455 #####
456
457 #####
458
459 #####
460
461 #####
462
463 #####
464
465 #####
466
467 #####
468
469 #####
470
471 #####
472
473 #####
474
475 #####
476
477 #####
478
479 #####
480
481 #####
482
483 #####
484
485 #####
486
487 #####
488
489 #####
490
491 #####
492
493 #####
494
495 #####
496
497 #####
498
499 #####
500
501 #####
502
503 #####
504
505 #####
506
507 #####
508
509 #####
510
511 #####
512
513 #####
514
515 #####
516
517 #####
518
519 #####
520
521 #####
522
523 #####
524
525 #####
526
527 #####
528
529 #####
530
531 #####
532
533 #####
534
535 #####
536
537 #####
538
539 #####
540
541 #####
542
543 #####
544
545 #####
546
547 #####
548
549 #####
550
551 #####
552
553 #####
554
555 #####
556
557 #####
558
559 #####
560
561 #####
562
563 #####
564
565 #####
566
567 #####
568
569 #####
570
571 #####
572
573 #####
574
575 #####
576
577 #####
578
579 #####
580
581 #####
582
583 #####
584
585 #####
586
587 #####
588
589 #####
590
591 #####
592
593 #####
594
595 #####
596
597 #####
598
599 #####
600
601 #####
602
603 #####
604
605 #####
606
607 #####
608
609 #####
610
611 #####
612
613 #####
614
615 #####
616
617 #####
618
619 #####
620
621 #####
622
623 #####
624
625 #####
626
627 #####
628
629 #####
630
631 #####
632
633 #####
634
635 #####
636
637 #####
638
639 #####
640
641 #####
642
643 #####
644
645 #####
646
647 #####
648
649 #####
650
651 #####
652
653 #####
654
655 #####
656
657 #####
658
659 #####
660
661 #####
662
663 #####
664
665 #####
666
667 #####
668
669 #####
670
671 #####
672
673 #####
674
675 #####
676
677 #####
678
679 #####
680
681 #####
682
683 #####
684
685 #####
686
687 #####
688
689 #####
690
691 #####
692
693 #####
694
695 #####
696
697 #####
698
699 #####
700
701 #####
702
703 #####
704
705 #####
706
707 #####
708
709 #####
710
711 #####
712
713 #####
714
715 #####
716
717 #####
718
719 #####
720
721 #####
722
723 #####
724
725 #####
726
727 #####
728
729 #####
730
731 #####
732
733 #####
734
735 #####
736
737 #####
738
739 #####
740
741 #####
742
743 #####
744
745 #####
746
747 #####
748
749 #####
750
751 #####
752
753 #####
754
755 #####
756
757 #####
758
759 #####
760
761 #####
762
763 #####
764
765 #####
766
767 #####
768
769 #####
770
771 #####
772
773 #####
774
775 #####
776
777 #####
778
779 #####
780
781 #####
782
783 #####
784
785 #####
786
787 #####
788
789 #####
790
791 #####
792
793 #####
794
795 #####
796
797 #####
798
799 #####
800
801 #####
802
803 #####
804
805 #####
806
807 #####
808
809 #####
810
811 #####
812
813 #####
814
815 #####
816
817 #####
818
819 #####
820
821 #####
822
823 #####
824
825 #####
826
827 #####
828
829 #####
830
831 #####
832
833 #####
834
835 #####
836
837 #####
838
839 #####
840
841 #####
842
843 #####
844
845 #####
846
847 #####
848
849 #####
850
851 #####
852
853 #####
854
855 #####
856
857 #####
858
859 #####
860
861 #####
862
863 #####
864
865 #####
866
867 #####
868
869 #####
870
871 #####
872
873 #####
874
875 #####
876
877 #####
878
879 #####
880
881 #####
882
883 #####
884
885 #####
886
887 #####
888
889 #####
890
891 #####
892
893 #####
894
895 #####
896
897 #####
898
899 #####
900
901 #####
902
903 #####
904
905 #####
906
907 #####
908
909 #####
910
911 #####
912
913 #####
914
915 #####
916
917 #####
918
919 #####
920
921 #####
922
923 #####
924
925 #####
926
927 #####
928
929 #####
930
931 #####
932
933 #####
934
935 #####
936
937 #####
938
939 #####
940
941 #####
942
943 #####
944
945 #####
946
947 #####
948
949 #####
950
951 #####
952
953 #####
954
955 #####
956
957 #####
958
959 #####
960
961 #####
962
963 #####
964
965 #####
966
967 #####
968
969 #####
970
971 #####
972
973 #####
974
975 #####
976
977 #####
978
979 #####
980
981 #####
982
983 #####
984
985 #####
986
987 #####
988
989 #####
990
991 #####
992
993 #####
994
995 #####
996
997 #####
998
999 #####
1000
1001 #####
1002
1003 #####
1004
1005 #####
1006
1007 #####
1008
1009 #####
1010
1011 #####
1012
1013 #####
1014
1015 #####
1016
1017 #####
1018
1019 #####
1020
1021 #####
1022
1023 #####
1024
1025 #####
1026
1027 #####
1028
1029 #####
1030
1031 #####
1032
1033 #####
1034
1035 #####
1036
1037 #####
1038
1039 #####
1040
1041 #####
1042
1043 #####
1044
1045 #####
1046
1047 #####
1048
1049 #####
1050
1051 #####
1052
1053 #####
1054
1055 #####
1056
1057 #####
1058
1059 #####
1060
1061 #####
1062
1063 #####
1064
1065 #####
1066
1067 #####
1068
1069 #####
1070
1071 #####
1072
1073 #####
1074
1075 #####
1076
1077 #####
1078
1079 #####
1080
1081 #####
1082
1083 #####
1084
1085 #####
1086
1087 #####
1088
1089 #####
1090
1091 #####
1092
1093 #####
1094
1095 #####
1096
1097 #####
1098
1099 #####
1100
1101 #####
1102
1103 #####
1104
1105 #####
1106
1107 #####
1108
1109 #####
1110
1111 #####
1112
1113 #####
1114
1115 #####
1116
1117 #####
1118
1119 #####
1120
1121 #####
1122
1123 #####
1124
1125 #####
1126
1127 #####
1128
1129 #####
1130
1131 #####
1132
1133 #####
1134
1135 #####
1136
1137 #####
1138
1139 #####
1140
1141 #####
1142
1143 #####
1144
1145 #####
1146
1147 #####
1148
1149 #####
1150
1151 #####
1152
1153 #####
1154
1155 #####
1156
1157 #####
1158
1159 #####
1160
1161 #####
1162
1163 #####
1164
1165 #####
1166
1167 #####
1168
1169 #####
1170
1171 #####
1172
1173 #####
1174
1175 #####
1176
1177 #####
1178
1179 #####
1180
1181 #####
1182
1183 #####
1184
1185 #####
1186
1187 #####
1188
1189 #####
1190
1191 #####
1192
1193 #####
1194
1195 #####
1196
1197 #####
1198
1199 #####
1200
1201 #####
1202
1203 #####
1204
1205 #####
1206
1207 #####
1208
1209 #####
1210
1211 #####
1212
1213 #####
1214
1215 #####
1216
1217 #####
1218
1219 #####
1220
1221 #####
1222
1223 #####
1224
1225 #####
1226
1227 #####
1228
1229 #####
1230
1231 #####
1232
1233 #####
1234
1235 #####
1236
1237 #####
1238
1239 #####
1240
1241 #####
1242
1243 #####
1244
1245 #####
1246
1247 #####
1248
1249 #####
1250
1251 #####
1252
1253 #####
1254
1255 #####
1256
1257 #####
1258
1259 #####
1260
1261 #####
1262
1263 #####
1264
1265 #####
1266
1267 #####
1268
1269 #####
1270
1271 #####
1272
1273 #####
1274
1275 #####
1276
1277 #####
1278
1279 #####
1280
1281 #####
1282
1283 #####
1284
1285 #####
1286
1287 #####
1288
1289 #####
1290
1291 #####
1292
1293 #####
1294
1295 #####
1296
1297 #####
1298
1299 #####
1300
1301 #####
1302
1303 #####
1304
1305 #####
1306
1307 #####
1308
1309 #####
1310
1311 #####
1312
1313 #####
1314
1315 #####
1316
1317 #####
1318
1319 #####
1320
1321 #####
1322
1323 #####
1324
1325 #####
1326
1327 #####
1328
1329 #####
1330
1331 #####
1332
1333 #####
1334
1335 #####
1336
1337 #####
1338
1339 #####
1340
1341 #####
1342
1343 #####
1344
1345 #####
1346
1347 #####
1348
1349 #####
1350
1351 #####
1352
1353 #####
1354
1355 #####
1356
1357 #####
1358
1359 #####
1360
1361 #####
1362
1363 #####
1364
1365 #####
1366
1367 #####
1368
1369 #####
1370
1371 #####
1372
1373 #####
1374
1375 #####
1376
1377 #####
1378
1379 #####
1380
1381 #####
1382
1383 #####
1384
1385 #####
1386
1387 #####
1388
1389 #####
1390
1391 #####
1392
1393 #####
1394
1395 #####
1396
1397 #####
1398
1399 #####
1400
1401 #####
1402
1403 #####
1404
1405 #####
1406
1407 #####
1408
1409 #####
1410
1411 #####
1412
1413 #####
1414
1415 #####
1416
1417 #####
1418
1419 #####
1420
1421 #####
1422
1423 #####
1424
1425 #####
1426
1427 #####
1428
1429 #####
1430
1431 #####
1432
1433 #####
1434
1435 #####
1436
1437 #####
1438
1439 #####
1440
1441 #####
1442
1443 #####
1444
1445 #####
1446
1447 #####
1448
1449 #####
1450
1451 #####
1452
1453 #####
1454
1455 #####
1456
1457 #####
1458
1459 #####
1460
1461 #####
1462
1463 #####
1464
1465 #####
1466
1467 #####
1468
1469 #####
1470
1471 #####
1472
1473 #####
1474
1475 #####
1476
1477 #####
1478
1479 #####
1480
1481 #####
1482
1483 #####
1484
1485 #####
1486
1487 #####
1488
1489 #####
1490
1491 #####
1492
1493 #####
1494
1495 #####
1496
1497 #####
1498
1499 #####
1500
1501 #####
1502
1503 #####
1504
1505 #####
1506
1507 #####
1508
1509 #####
1510
1511 #####
1512
1513 #####
1514
1515 #####
1516
1517 #####
1518
1519 #####
1520
1521 #####
1522
1523 #####
1524
1525 #####
1526
1527 #####
1528
1529 #####
1530
1531 #####
1532
1533 #####
1534
1535 #####
1536
1537 #####
1538
1539 #####
1540
1541 #####
1542
1543 #####
1544
1545 #####
1546
1547 #####
1548
1549 #####
1550
1551 #####
1552
1553 #####
1554
1555 #####
1556
1557 #####
1558
1559 #####
1560
1561 #####
1562
1563 #####
1564
1565 #####
1566
1567 #####
1568
1569 #####
1570
1571 #####
1572
1573 #####
1574
1575 #####
1576
1577 #####
1578
1579 #####
1580
1581 #####
1582
1583 #####
1584
1585 #####
1586
1587 #####
1588
1589 #####
1590
1591 #####
1592
1593 #####
1594
1595 #####
1596
1597 #####
1598
1599 #####
1600
1601 #####
1602
1603 #####
1604
1605 #####
1606
1607 #####
1608
1609 #####
1610
1611 #####
1612
1613 #####
1614
1615 #####
1616
1617 #####
1618
1619 #####
1620
1621 #####
1622
1623 #####
1624
1625 #####
1626
1627 #####
1628
1629 #####
1630
1631 #####
1632
1633 #####
1634
1635 #####
1636
1637 #####
1638
1639 #####
1640
1641 #####
1642
1643 #####
1644
1645 #####
1646
1647 #####
1648
1649 #####
1650
1651 #####
1652
1653 #####
1654
1655 #####
1656
1657 #####
1658
1659 #####
1660
1661 #####
1662
1663 #####
1664
1665 #####
1666
1667 #####
1668
1669 #####
1670
1671 #####
1672
1673 #####
1674
1675 #####
1676
1677 #####
1678
1679 #####
1680
1681 #####
1682
1683 #####
1684
1685 #####
1686
1687 #####
1688
1689 #####
1690
1691 #####
1692
1693 #####
1694
1695 #####
1696
1697 #####
1698
1699 #####
1700
1701 #####
1702
1703 #####
1704
1705 #####
1706
1707 #####
1708
1709 #####
1710
1711 #####
1712
1713 #####
1714
1715 #####
1716
1717 #####
1718
1719 #####
1720
1721 #####
1722
1723 #####
1724
1725 #####
1726
1727 #####
1728
1729 #####
1730
1731 #####
1732
1733 #####
1734
1735 #####
1736
1737 #####
1738
1739 #####
1740
1741 #####
1742
1743 #####
1744
1745 #####
1746
1747 #####
1748
1749 #####
1750
1751 #####
1752
1753 #####
1754
1755 #####
1756
1757 #####
1758
1759 #####
1760
1761 #####
1762
1763 #####
1764
1765 #####
1766
1767 #####
1768
1769 #####
1770
1771 #####
1772
1773 #####
1774
1775 #####
1776
1777 #####
1778
1779 #####
1780
1781 #####
1782
1783 #####
1784
1785 #####
1786
1787 #####
1788
1789 #####
1790
1791 #####
1792
1793 #####
1794
1795 #####
1796
1797 #####
1798
1799 #####
1800
1801 #####
1802
1803 #####
1804
1805 #####
1806
1807 #####
1808
1809 #####
1810
1811 #####
1812
1813 #####
1814
1815 #####
1816
1817 #####
1818
1819 #####
1820
1821 #####
1822
1823 #####
1824
1825 #####
1826
1827 #####
1828
1829 #####
1830
1831 #####
1832
1833 #####
1834
1835 #####
1836
1837 #####
1838
1839 #####
1840
1841 #####
1842
1843 #####
1844
1845 #####
1846
1847 #####
1848
1849 #####
1850
1851 #####
1852
1853 #####
1854
1855 #####
1856
1857 #####
1858
1859 #####
1860
1861 #####
1862
1863 #####
1864
1865 #####
1866
1867 #####
1868
1869 #####
1870
1871 #####
1872
1873 #####
1874
1875 #####
1876
1877 #####
1878
1879 #####
1880
1881 #####
1882
1883 #####
1884
1885 #####
1886
1887 #####
1888
1889 #####
1890
1891 #####
1892
1893 #####
1894
1895 #####
1896
1897 #####
1898
1899 #####
1900
1901 #####
1902
1903 #####
1904
1905 #####
1906
1907 #####
1908
1909 #####
1910
1911 #####
1912
1913 #####
1914
1915 #####
1916
1917 #####
1918
1919 #####
1920
1921 #####
1922
1923 #####
1924
1925 #####
1926
1927 #####
1928
1929 #####
1930
1931 #####
1932
1933 #####
1934
1935 #####
1936
1937 #####
1938
1939 #####
1940
1941 #####
1942
1943 #####
1944
1945 #####
1946
1947 #####
1948
1949 #####
1950
1951 #####
1952
1953 #####
1954
1955 #####
1956
1957 #####
1958
1959 #####
1960
1961 #####
1962
1963 #####
1964
1965 #####
1966
1967 #####
1968
1969 #####
1970
1971 #####
1972
1973 #####
1974
1975 #####
1976
1977 #####
1978
1979 #####
1980
1981 #####
1982
1983 #####
1984
1985 #####
1986
1987 #####
1988
1989 #####
1990
1991 #####
1992
1993 #####
1994
1995 #####
1996
1997 #####
1998
1999 #####
2000
2001 #####
2002
2003 #####
2004
2005 #####
2006
2007 #####
2008
2009 #####
2010
2011 #####
2012
2013 #####
2014
2015 #####
2016
2017 #####
2018
2019 #####
2020
2021 #####
2022
2023 #####
2024
2025 #####
2026
2027 #####
2028
2029 #####
2030
2031 #####
2032
2033 #####
2034
2035 #####
2036
2037 #####
2038
2039 #####
2040
2041 #####
2042
2043 #####
2044
2045 #####
2046
2047 #####
2048
2049 #####
2050
2051 #####
2052
2053 #####
2054
2055 #####
2056
2057 #####
2058
2059 #####
2060
2061 #####
2062
2063 #####
2064
2065 #####
2066
2067 #####
2068
2069 #####
2070
2071 #####
2072
2073 #####
2074
2075 #####
2076
2077 #####
2078
2079 #####
2080
2081 #####
2082
2083 #####
2084
2085 #####
2086
2087 #####
2088
2089 #####
2090
2091 #####
2092
2093 #####
2094
2095 #####
2096
2097 #####
2098
2099 #####
2100
2101 #####
2102
2103 #####
2104
2105 #####
2106
2107 #####
2108
2109 #####
2110
2111 #####
2112
2113 #####
2114
2115 #####
2116
2117 #####
2118
2119 #####
2120
2121 #####
2122
2123 #####
2124
2125 #####
2126
2127 #####
2128
2129 #####
2130
2131 #####
2132
2133 #####
2134
2135 #####
2136
2137 #####
2138
2139 #####
2140
2141 #####
2142
2143 #####
2144
2145 #####
2146
2147 #####
2148
2149 #####
2150
2151 #####
2152
2153 #####
2154
2155 #####
2156
2157 #####
2158
2159 #####
2160
2161 #####
2162
2163 #####
2164
2165 #####
2166
2167 #####
2168
2169 #####
2170
2171 #####
2172
2173 #####
2174
2175 #####
2176
2177 #####
2178
2179 #####
2180
2181 #####
2182
2183 #####
2184
2185 #####
2186
2187 #####
2188
2189 #####
2190
2191 #####
2192
2193 #####
2194
2195 #####
2196
2197 #####
2198
2199 #####
2200
2201 #####
2202
2203 #####
2204
2205 #####
2206
2207 #####
2208
2209 #####
2210
2211 #####
2212
2213 #####
2214
2215 #####
2216
2217 #####
2218
2219 #####
2220
2221 #####
2222
2223 #####
2224
2225 #####
2226
2227 #####
2228
2229 #####
2230
2231 #####
2232
2233 #####
2234
2235 #####
2236
2237 #####
2238
2239 #####
2240
2241 #####
2242
2243 #####
2244
2245 #####
2246
2247 #####
2248
2249 #####
2250
2251 #####
2252
2253 #####
2254
2255 #####
2256
2257 #####
2258
2259 #####
2260
2261 #####
2262
2263 #####
2264
2265 #####
2266
2267 #####
2268
2269 #####
2270
2271 #####
2272
2273 #####
2274
2275 #####
2276
2277 #####
2278
2279 #####
2280
2281 #####
2282
2283 #####
2284
2285 #####
2286
2287 #####
2288
2289 #####
2290
2291 #####
2292
2293 #####
2294
2295 #####
2296
2297 #####
2298
2299 #####
2300
2301 #####
2302
2303 #####
2304
2305 #####
2306
2307 #####
2308
2309 #####
2310
2311 #####
2312
2313 #####
2314
2315 #####
2316
2317 #####
2318
2319 #####
2320
2321 #####
2322
2323 #####
2324
2325 #####
2326
2327 #####
2328
2329 #####
2330
2331 #####
2332
2333 #####
2334
2335 #####
2336
2337 #####
2338
2339 #####
2340
2341 #####
2342
2343 #####
2344
2345 #####
2346
2347 #####
2348
2349 #####
2350
2351 #####
2352
2353 #####
2354
2355 #####
2356
2357 #####
2358
2359 #####
2360
2361 #####
2362
2363 #####
2364
2365 #####
2366
2367 #####
2368
2369 #####
2370
2371 #####
2372
2373 #####
2374
2375 #####
2376
2377 #####
2378
2379 #####
2380
2381 #####
2382
2383 #####
2384
2385 #####
2386
2387 #####
2388
2389 #####
2390
2391 #####
2392
2393 #####
2394
2395 #####
2396
2397 #####
2398
2399 #####
2400
2401 #####
2402
2403 #####
2404
2405 #####
2406
2407 #####
2408
2409 #####
24
```

## LAMPIRAN 7. LINK GOOGLE DRIVE & GITHUB FILE PROGRAM SOCIALCAM

NO	LINK	LINK	
		LINK LANGSUNG	QR CODE
1	SocialCam Official Github Page	<a href="https://github.com/RaihanArvi/SocialCam">https://github.com/RaihanArvi/SocialCam</a>	
2	SocialCam Custom YOLOv4 Model	<a href="https://drive.google.com/drive/folders/1xsnl8BEHXpfZAIgPwPQcYQV8FGAQMkr2?usp=sharing">https://drive.google.com/drive/folders/1xsnl8BEHXpfZAIgPwPQcYQV8FGAQMkr2?usp=sharing</a>	
3	Data Hasil Pengujian SocialCam	<a href="https://drive.google.com/drive/folders/1zMbSyQjX-xBEpiyCcTJyFbIeoPqrGVCY?usp=sharing">https://drive.google.com/drive/folders/1zMbSyQjX-xBEpiyCcTJyFbIeoPqrGVCY?usp=sharing</a>	

## LAMPIRAN 8. LINK VIDEO SOCIALCAM



NO	JUDUL VIDEO	LINK	
		LINK LANGSUNG	LINK QR CODE
1	SocialCam Overview	<a href="https://youtu.be/5KbmtXGbf54">https://youtu.be/5KbmtXGbf54</a>	
Video berisi penjelasan-penjelasan detail mengenai fitur-fitur SocialCam beserta cara penggunaan SocialCam.			

## LAMPIRAN 9. PENGUJIAN RESOLUSI VIDEO

NO	RESOLUSI	PANJANG	LEBAR
1	<b>2160p</b>	3840 px	2160 px
2	<b>1440p</b>	2560 px	1440 px
3	<b>1080p</b>	1920 px	1080 px
4	<b>720p</b>	1280 px	720 px
5	<b>480p</b>	854 px	480 px
6	<b>360p</b>	640 px	320 px
7	<b>240p</b>	426 px	240 px



## LAMPIRAN 10. LINK SOFTWARE YANG DIGUNAKAN

NO	SOFTWARE	LINK
1	PyCharm Community Edition 2021.1.1	<a href="https://download.jetbrains.com/python/pycharm-community-2021.1.exe">https://download.jetbrains.com/python/pycharm-community-2021.1.exe</a>
2	Visual Studio 2019	<a href="https://visualstudio.microsoft.com/vs/">https://visualstudio.microsoft.com/vs/</a>
3	Bulk Rename Utility	<a href="https://www.bulkrenameutility.co.uk">https://www.bulkrenameutility.co.uk</a>
4	Python 3.9.5	<a href="https://www.python.org/downloads/release/python-395/">https://www.python.org/downloads/release/python-395/</a>
5	OpenCV 4.5.2	<a href="https://github.com/opencv/opencv/tree/4.5.2">https://github.com/opencv/opencv/tree/4.5.2</a>
6	OpenCV Contrib 4.5.2	<a href="https://github.com/opencv/opencv_contrib/tree/4.5.2">https://github.com/opencv/opencv_contrib/tree/4.5.2</a>
7	cmdmp3.exe	<a href="https://github.com/jimlawless/cmdmp3">https://github.com/jimlawless/cmdmp3</a>
8	Adobe Photoshop CC 2021	<a href="https://www.adobe.com/sea/products/photoshop">https://www.adobe.com/sea/products/photoshop</a>
9	cuDNN 11.3 Windows x64 v8.2.1.32	<a href="https://developer.nvidia.com/compute/machine-learning/cudnn/secure/8.2.1.32/11.3_06072021/cudnn-11.3-windows-x64-v8.2.1.32.zip">https://developer.nvidia.com/compute/machine-learning/cudnn/secure/8.2.1.32/11.3_06072021/cudnn-11.3-windows-x64-v8.2.1.32.zip</a>
10	CUDA Toolkit 11.0 Update 1	<a href="https://developer.nvidia.com/cuda-11.0-update1-download-archive">https://developer.nvidia.com/cuda-11.0-update1-download-archive</a>
11	CMake	<a href="https://cmake.org/">https://cmake.org/</a>
12	DarkNet	<a href="https://github.com/AlexeyAB/darknet/">https://github.com/AlexeyAB/darknet/</a>
13	LabelImg	<a href="https://github.com/tzutalin/labelImg">https://github.com/tzutalin/labelImg</a>
14	Iriun Webcam (Android Client + Windows Server)	<a href="https://iriun.com/">https://iriun.com/</a>

### LAMPIRAN 11. TIME TABLE PENELITIAN

NO	KETERANGAN	TEMPAT	Waktu Penelitian (Tahun 2021)											
			Bulan 1				Bulan 2				Bulan 3			
			1	2	3	4	1	2	3	4	1	2	3	4
1	Kajian literatur, pengumpulan <i>dataset</i> , pencarian <i>software</i> , dan pengumpulan alat dan bahan.	Kediaman penulis, Kab. Tanah Datar, Prov. Sumatra Barat												
2	Pemrograman program SocialCam serta desain UI.	Kediaman penulis dan Kampus SMAN 1 Sumatera Barat												
3	<i>Training</i> YOLOv4 dan pengujian <i>model</i> YOLOv4.	Asrama dan Kampus SMAN 1 Sumatera Barat												
4	Pengujian performa dan pengujian langsung SocialCam.	Asrama dan Kampus SMAN 1 Sumatera Barat												
5	Analisis data penelitian.	Asrama dan Kampus SMAN 1 Sumatera Barat												
6	Pembuatan laporan hasil penelitian.	Asrama dan Kampus SMAN 1 Sumatera Barat												

## LAMPIRAN 12. DOKUMENTASI PROGRAM SOCIALCAM

### SOCIAL CAM DOCUMENTATION

#### A. SocialCam

SocialCam merupakan program deteksi social distancing berbasis YOLOv4.

#### B. Fitur

- Alarm
- Real-time
- Warp perspective

#### C. Penggunaan

```
py SocialCam.py -i -c -o -g -l -p -a -d  
py SocialCam.py --input --inputcam --output --usegpu --log --getpoints --alarm --display
```

- Digunakan Salah Satu (--input atau -inputcam).

--input (-i) : Lokasi file input. Default ada di config.DEFAULT\_INPUT.

--inputcam (-c) : Nomor webcam (optional). Default kosong.

Tanpa argument -input adalah default dari program.

- Optional

--output (-o) : Nama file yang akan menjadi output. Apabila kosong, tidak ada output. Output dalam format .avi. Default = Kosong

--usegpu (-g) : Pakai backend CUDA atau tidak. 0 = Tidak. 1 = Ya. Default ada di config.USE\_GPU.

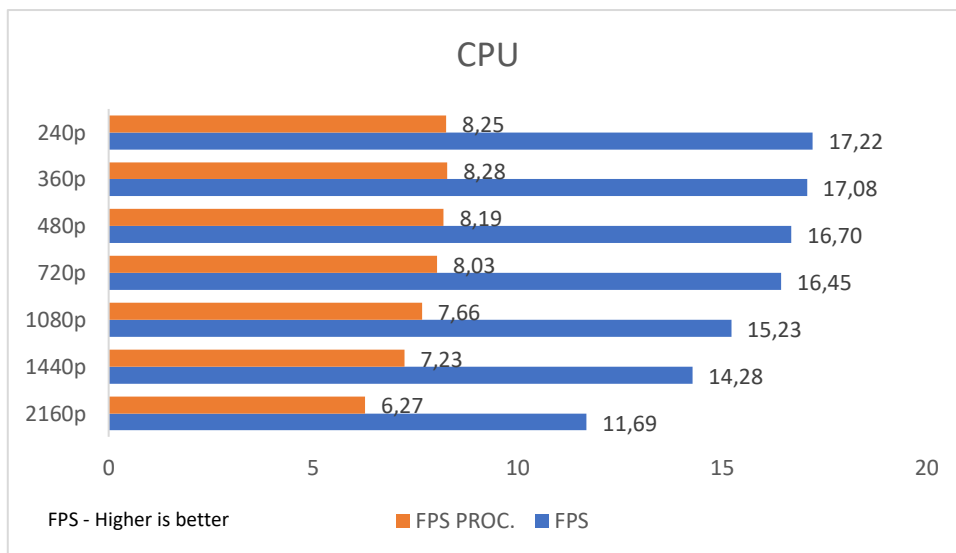
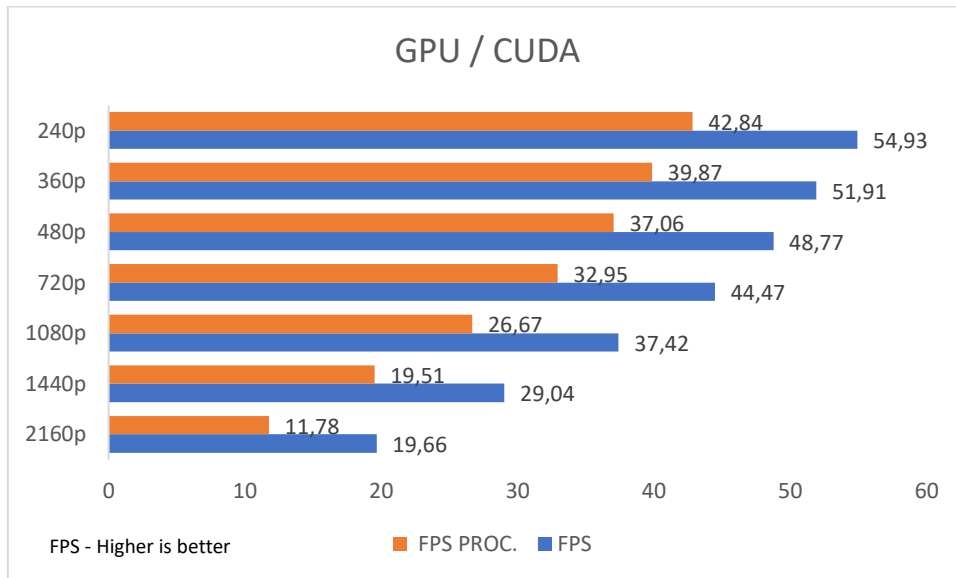
--log (-l) : Nama output log (.txt). Biarkan kosong apabila tidak menginginkan log. Default = Kosong.

Link pengunduhan file dokumentasi:



<https://drive.google.com/file/d/1NUsEc64Z-GmTe4i7lmHQZ9ijH2sL3po4/view?usp=sharing>

### LAMPIRAN 13. GRAFIK PENGUJIAN PERFORMA FPS



#### LAMPIRAN 14. HYPERPARAMETER PROSES TRAINING

NO	PARAMETER	NILAI
1	no. of classes	1
2	batch	64
3	subdivisions	64
4	width	416
5	height	416
6	learning rate	0.001
7	max_batches	6000
8	steps	4800, 5400
9	convolutional filters	18