# **Laporan Proyek Akhir Deep Learning**

**Social Distancing Violation Detection**

## 

**Disusun Oleh :**

Andrew Kurniawan - C14180004

Vincent Ie Putra - C14180020

Marcel Slamet Sugianto - C14180033

Cliffton Sebastian - C14180041

Gerry Steven - C14180044

Michael Alexander Rustan - C14180175

**Semester Genap 2020/2021**

**Fakultas Teknologi Industri**

**Universitas Kristen Petra**

**Surabaya**

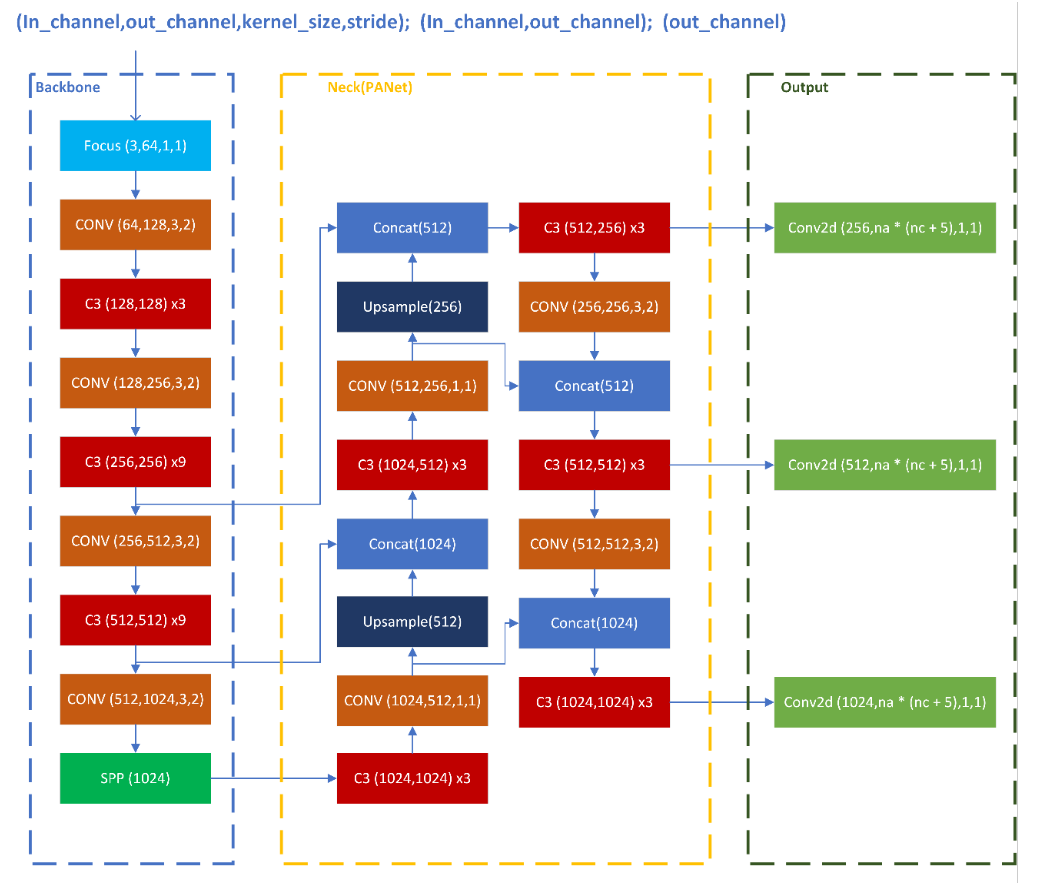
**2020**

## Latar Belakang

Wabah COVID-19 yang telah terjadi hingga tanggal 23 Juni 2021 menurut data dari Kementerian Kesehatan Republik Indonesia telah menginfeksi 2.033.421 jiwa dan terdapat 55.594 jiwa yang telah meninggal dunia (Covid19.go.id, 23 Juni 2021). Salah satu cara yang dapat dilakukan untuk mencegah penyebaran COVID-19 adalah dengan melakukan *social distancing* dengan orang disekitar kita minimal sejauh 1 meter (WHO, 7 Juni 2021). Beberapa negara dan kota di dunia mulai memberlakukan protokol tersebut dan melakukan pengawasan ketat tetapi, masalah yang ada adalah sulitnya oknum-oknum yang terkait untuk melakukan pengawasan tersebut secara langsung sehingga memerlukan bantuan pengawasan melalui CCTV yang dapat mengawasi daerah yang lebih luas.

Dalam proyek ini bertujuan untuk menggunakan metode Deep Learning dengan menggunakan model YOLO v5 dan *bird eye-view* untuk mendeteksi manusia yang terdapat pada video dan menggunakan metode *Euclidean Distance* untuk mendeteksi pelanggaran *social distancing* yang terjadi pada video.

## Arsitektur

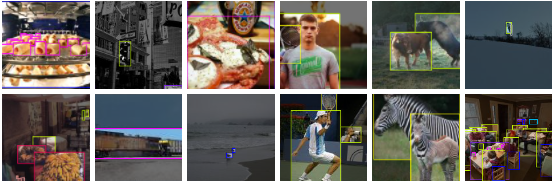


Gambar 1. Arsitektur YOLOv5

## Dataset

Dataset untuk deteksi manusia yang digunakan pada proyek ini adalah *Common Object in Context* 2017 (COCO 2017) yang diambil dari *Roboflow*. Dataset terdiri dari 121.408 gambar yang sudah diberi label *bounding-box ground truth*, dan di preprocessing menjadi 63.101 gambar yang mempunyai label “person” dikarenakan dalam proyek ini menggunakan *person detection* saja sehingga dapat mempercepat proses training.

(<https://public.roboflow.com/object-detection/microsoft-coco-subset/>)



Gambar 2. Contoh preview dari dataset COCO

## Metode

* **Overview**

Pada proyek ini, pertama-tama akan menggunakan metode You Only Look Once (YOLO) v5 untuk deteksi manusia tiap frame pada video dan menghasilkan bounding box pada tiap manusia. Kemudian dari manusia yang sudah dideteksi, setiap bounding box perspektifnya diubah menjadi bird-eye view untuk mendapatkan lokasi setiap manusia. Setiap manusia yang terdeteksi diukur jaraknya ke lokasi manusia lainnya menggunakan Eucledian Distance. Jarak yang dekat akan dianggap pelanggaran social distancing.

* **YOLOv5**

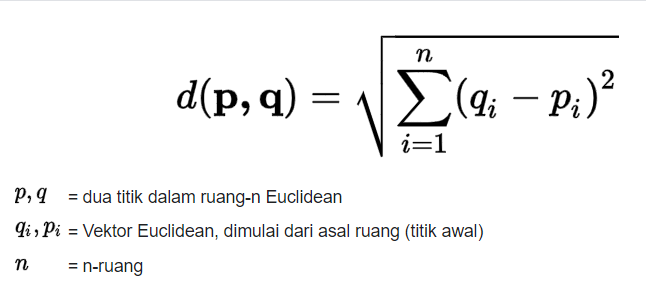
YOLOv5 adalah salah satu metode *object detection* yang membagi gambar menggunakan grid system. Tiap cell di grid ini digunakan untuk mendeteksi sebuah object di dalamnya. YOLOv5 diperkenalkan oleh Glenn Jocher yang menggunakan Pytorch framework pada 18 Mei 2020.

* **Bird-Eye View**

Bird-eye view merupakan istilah dalam fotografi dimana sudut pandang yang di ambil kamera mirip dengan yang dilihat oleh seekor burung yang sedang terbang jadi hasil dari bird-eye view memperlihatkan foto dari ketinggian.

* **Eucledian Distance**

Euclidean Distance antara dua titik dalam ruang Euclidean adalah panjang segmen garis antara dua titik. Ini dapat dihitung dari koordinat Cartesian dari titik - titik menggunakan teorema Pythagoras. Euclidean Distance biasa digunakan untuk mencari jarak antara 2 titik.



Gambar 3. Rumus Euclidean Distance

## Pengujian

Berikut merupakan hal-hal yang diuji pada projek ini :

1. Melakukan training pada YOLOv5 dengan 10 epoch dan 16 batch
2. Melakukan training pada YOLOv5 dengan 10 epoch dan 32 batch
3. Melakukan training pada YOLOv5 dengan 3 epoch dan 16 batch
4. Mengubah kedalaman dan lebar layer saat training 3 epoch dan 16 batch
5. Mengubah learning rate dari 0.1 - 0.00001 menjadi 0.1 - 0.01

## Hasil

1. **Hasil Metrics Training dari Pengujian**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Epoch | Batch | Learning rate | Depth | Width | Precision | Recall | mAP@0.5 | mAP @.5:.95 | Training Time |
| 3 | 16 | 1e-1 s/d 1e-5 | 0.33 (default) | 0.5 (default) | 0.52 | 0.35 | 0.37 | 0.15 | ± 1.5 Jam |
| 3 | 16 | 1e-1 s/d 1e-5 | 0.5 | 0.33 | 0.5 | 0.35 | 0.35 | 0.14 | ± 1.5 Jam |
| 10 | 16 | 1e-1 s/d 1e-5 | 0.33 (default) | 0.5 (default) | 0.74 | 0.52 | 0.60 | 0.32 | ± 5 Jam |
| 10 | 32 | 1e-1 s/d 1e-5 | 0.33 (default) | 0.5 (default) | 0.73 | 0.52 | 0.60 | 0.32 | ± 5 Jam |
| 10 | 16 | 1e-1 s/d 1e-2 | 0.33 (default) | 0.5 (default) | 0.73 | 0.53 | 0.61 | 0.33 | ± 4 Jam |

Tabel 1. Tabel perbandingan pengujian

1. **Hasil Output dari Proyek**

Output dari proyek yang kami hasilkan adalah video yang sudah diberi lingkaran pada masing-masing orang yang terdeteksi. Lingkaran ini bertugas untuk jarak aman yang ditentukan pada setiap orang, jika lingkaran tersebut overlap, warna lingkaran akan berubah warna menjadi warna merah yang menandakan pelanggaran *social distancing*. Hasil dari proyek ini dapat dilihat pada gambar 3.

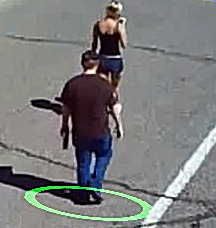
****

Gambar 4. Hasil video dari program.

1. **Kekurangan**

Dalam proyek ini kami mendeteksi beberapa kelemahan khususnya perihal deteksi manusia dan konsistensi deteksi:

* Manusia tidak terdeteksi pada suatu frame karena sebagian tubuhnya terhalang oleh objek lain atau overlap (gambar 5)
* Beberapa bounding box tidak konsisten dalam mendeteksi manusia sehingga jika pada suatu frame terdeteksi objeknya, frame berikutnya tidak terdeteksi meskipun tidak ada halangan (gambar 6)
* Pengelolahan masih tidak bersifat real-time dikarenakan komputasi yang cukup berat sehingga hanya sampai 1-10 Frame per Second



Gambar 5. Orang tidak terdeteksi karena sebagian tubuh terhalang / overlap



Gambar 6. Beberapa bounding box pada suatu frame tidak konsisten

## Kesimpulan dan Saran

Dari hasil dari proyek ini secara dasar metode ini bisa digunakan untuk *social distancing violation detection.* Akan tetapi metode ini masih bisa diperbaiki mulai dari tambahan epoch untuk memperbagus hasil dari *person detector,* mengatur *hyperparameter*, menggunakan input kamera top-down view atau mencari metode merubah perspektif menjadi *bird-eye view* yang lebih bagus untuk memperbagus *distance violation*, hingga mencari metode lain untuk bisa diimplementasikan secara real-time.

Cara lainnya untuk menyempurnakan proyek ini adalah dapat dengan melakukan perbandingan dengan objek yang nyata seperti mobil agar bisa mendapatkan jarak yang nyata antara kedua orang. Faktor lainnya untuk mendapatkan jarak antar adalah dengan cara mendapat informasi mengenai posisi dan angle dari kamera untuk membantu perhitunganya.

Dalam mendeteksi orang, terkadang *confidence* oleh YOLOv5 cukup rendah dikarenakan sudut kamera yang tinggi. Hal ini dikarenakan *data training* yang kurang dalam mendeteksi orang dari sudut tinggi. Oleh karena itu untuk *training* dapat ditambahkan *data training* deteksi manusia dari atas. Sama halnya dalam mengatasi gagalnya mendeteksi manusia akibat overlap, dapat diatasi dengan cara menambahkan variasi posisi pose manusianya disesuaikan dengan kondisi yang dipantau.

## Referensi

Definisi Bird Eye View | Photography Video Creation | WDI Wiki. (n.d.). Retrieved June 24, 2021, from webdev-id.com website: https://webdev-id.com/wdi-wiki/photography-video-creation/bird-eye-view

Gugus Tugas Percepatan Penanganan COVID-19. (2021, June 21). Retrieved June 24, 2021, from covid19.go.id website: https://covid19.go.id/

Rosebrock, A. (2014, August 25). 4 Point OpenCV getPerspective Transform Example. Retrieved June 20, 2021, from pyimagesearch website: https://www.pyimagesearch.com/2014/08/25/4-point-opencv-getperspective-transform-example/

Ultralytics. (n.d.). YOLOv5 Documentation. Retrieved June 24, 2021, from docs.ultralytics.com website: https://docs.ultralytics.com/

World Health Organization. (2021, June 23). Coronavirus disease (COVID-19) advice for the public. Retrieved June 24, 2021, from Who.int website: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public

YOLOV5网络结构\_Q1u1NG的博客-CSDN博客\_yolov5网络. (2020, July 22). Retrieved from blog.csdn.net website: https://blog.csdn.net/Q1u1NG/article/details/107511465