# **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**

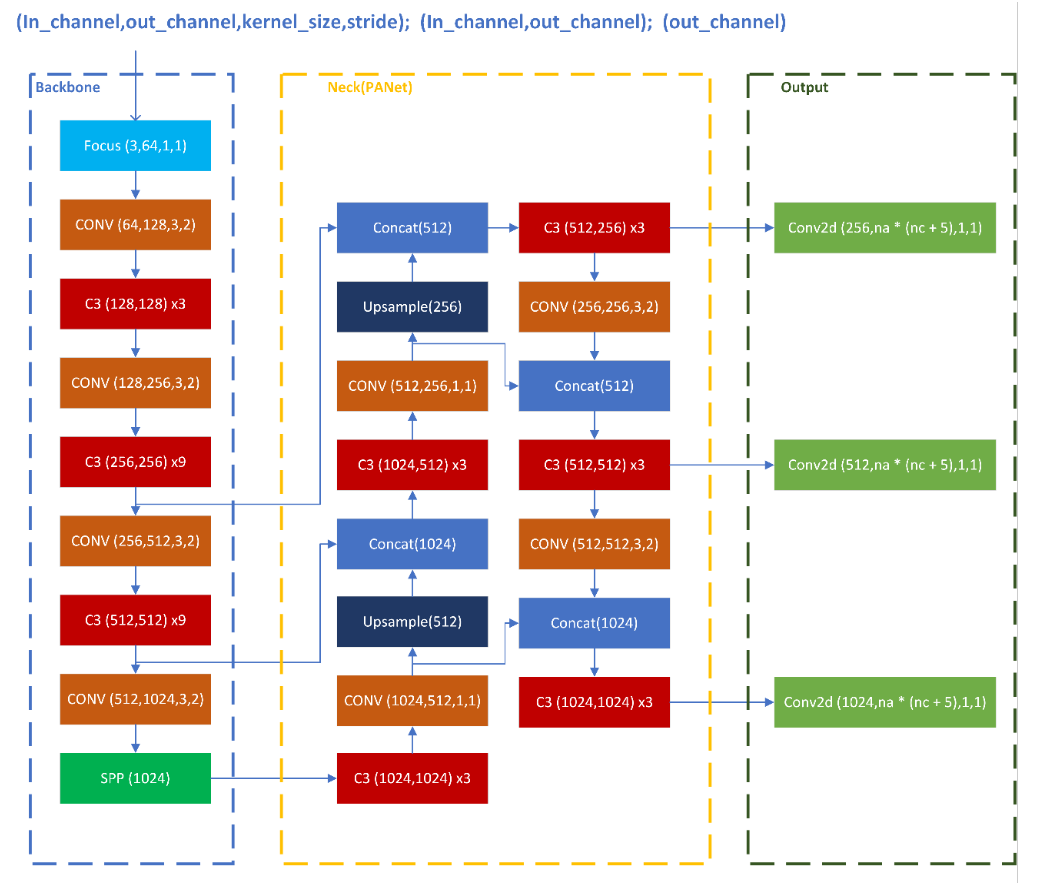
**2020**

## 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

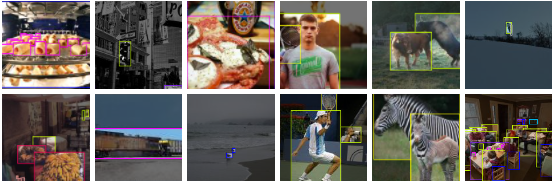


Gambar 1. Arsitektur YOLOv5

## 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**

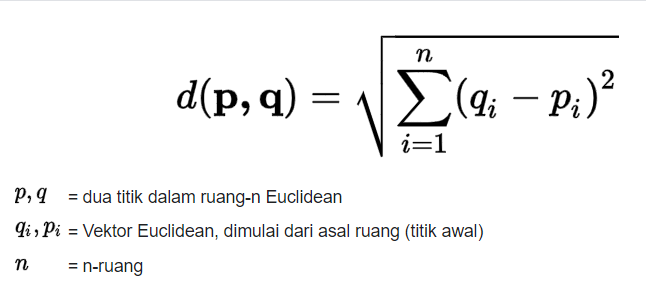
YOLOv5 adalah salah satu metode *object detection* yang membagi gambar menggunakan grid system. Tiap cell di grid ini digunakan untuk mendeteksi sebuah object di dalamnya. YOLOv5 diperkenalkan oleh Glenn Jocher yang menggunakan Pytorch framework pada 18 Mei 2020.

* **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.

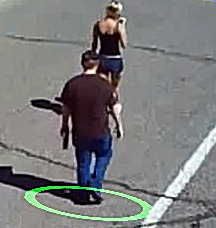
****

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.

## Refleksi Akademis Individu

## Andrew Kurniawan - C14180004

Dari proyek akhir ini, saya belajar banyak hal mengenai Deep Learning. Terutama di bagian hal-hal yang terkait dari mentraining model Deep Learning. Seperti bagaimana cara mengatur beberapa parameter dari modelnya, dataset yang digunakan ternyata harus lebih disesuaikan dengan kondisi dari permasalahanya, dan saya juga belajar dalam mengerjakan proyek Deep learning ini, saya seharusnya lebih berusaha lagi dalam melakukan percobaan-percobaan sendiri seperti merekam video data test atau gambar dataset sendiri agar bisa memiliki data pembanding yang nyata. Seperti bisa lebih paham posisi dari kameranya dan jarak asli antar manusianya, dll. Untuk kedepannya saya akan berusaha lebih berusaha dalam melakukan penelitian atau mengerjakan proyek tentang Deep Learning agar bisa mendapatkan hasil yang lebih maksimal.

## Vincent Ie Putra - C14180020

Dari proyek Deep Learning ini, saya belajar untuk mengatur atau memodifikasi parameter training dari YOLOv5 dan sekaligus melakukan training dengan *device* saya sendiri tanpa mengambil model atau weight yang sudah ada. Selain itu saya juga menyadari bahwa begitu pentingnya tahap pre-processing pada dataset yang akan digunakan untuk training karena ini akan mempengaruhi hasil dari model. Kemudian saya merasa masih perlu belajar banyak hal lagi terutama pada tahap pengujian. Pada proyek ini, pengujian yang dilakukan masih tergolong kurang baik karena tidak memperhatikan variabel-variabel yang diperlukan seperti jarak kamera terhadap objek, tinggi dan kemiringan kamera yang diletakkan, serta ukuran jalan pada tempat yang diuji. Saya menyadari betapa pentingnya mengetahui variabel-variabel tersebut untuk membuat model menjadi lebih baik dan juga dapat melakukan pengujian yang lebih berkualitas. Untuk kedepannya, saya berharap untuk tidak hanya memakai code yang saya temukan secara mentah-mentah, namun dapat melakukan pengujian yang lebih bervariasi dan berkualitas.

**Marcel Slamet Sugianto - C14180033**

Dari proyek dari mata kuliah *deep learning* ini, saya telah banyak mempelajari hal-hal yang berhubungan dengan mata kuliah ini seperti memahami penerapan model YOLO v5 untuk melakukan deteksi pejalan kaki yang ada pada video dan dengan itu dapat mengubah bounding box yang ada menjadi dalam bentuk *bird eye view* untuk melakukan pengecekan jarak antar pejalan kaki apakah melanggar aturan *social distancing* yang ada atau tidak. Tetapi karena jarak yang masih tidak asli dikarenakan menggunakan rekaman dari kamera, kita tidak bisa mengetahui jarak sesungguhnya sehingga cara yang terbaik untuk mengetahui jarak asli antara kedua orang adalah dengan menggunakan lebar suatu objek yang memiliki lebar cukup besar seperti mobil untuk mengetahui jarak antara kedua orang tersebut dan dengan memperhatikan posisi kamera dan angle dari kamera untuk mengukur jarak aslinya. Selain itu ada kekurangan pada proyek ini dimana program terkadang tidak mendeteksi pejalan kaki disaat menghadap pada posisi tertentu seperti membelakangi kamera sehingga terkadang terdeteksi dan tidak terdeteksi, selain itu juga ada posisi dimana pejalan kaki saat hanya terlihat setengah dari badannya saja saat muncul di kamera itu tidak terbaca oleh kamera sehingga menyebabkan deteksi pelanggaran *social distancing* yang ada tidak bisa menjadi maksimal sehingga cara untuk memaksimalkan deteksi pejalan kaki adalah dengan menambah varian pada dataset dengan kondisi-kondisi tertentu dan menambah *epoch* pada proses *training* agar bisa menghasilkan hasil yang lebih maksimal. Harapan saya kedepannya dari hal-hal diatas, saya dapat lebih belajar untuk memperhatikan kondisi dan variabel tertentu yang dapat mengoptimalkan hasil dari program.

**Cliffton Sebastian - C14180041**

Dari proyek akhir *Deep Learning* ini, dari topik Social Distancing Detection dengan YOLO ini, saya belajar cara training custom model dengan arsitektur network YOLO terutama YOLOv5 dan juga belajar bereksperimen dengan mengubah-ngubah parameter untuk training. Saya juga belajar tentang masalah / challange yang ada dalam topik ini. Mulai dari model Human Detection yang harus bagus dan cepat hingga berbagai cara untuk mengevaluasi jarak per orang untuk deteksi pelanggarannya (seperti Posisi Camera, Bird Eye View, Region yang berbeda-beda). Untuk proyek ini, kita menggunakan YOLOv5 untuk deteksi manusia, Bird Eye View Transformation, dan Eucledian Distance. Di YOLOv5, beberapa kekurangan dari model kita seperti bounding box yang jitter/flickering, beberapa orang yang ketutupan objek lain tidak terdeteksi. Beberapa solusi untuk hal ini adalah membuat dataset yang lebih bervariasi dengan sample orang yang tertutup objek lain / menggunakan dataset yang sesuai dengan kebutuhan. Juga melakukan training dengan epoch lebih tinggi serta parameter tuning untuk mengatasi presisi, recall yang kurang bagus. Lalu Masalah dari Bird Eye View Transformation, setelah beberapa percobaan yang kita lakukan, ditemukan masalah dimana menggunakan video orang lain yang berbeda-beda, maka region yang harus di warp juga berbeda. Tanpa memiliki data posisi kamera dari video-video tersebut (Angle, Tinggi, Arah Kamera, Jarak, dll) sangat sulit untuk mendapatkan Transformation Warping yang bagus. Oleh karena itu, ketika melakukan proyek ini, kita harus mengetahui data posisi kamera tersebut agar bisa memaksimalkan hasil warping dan bisa mendapatkan perkiraan jarak yang akurat, sehingga deteksi pelanggaran bisa maksimal. Untuk kedepannya, saya dapat lebih sering untuk mencoba dan bereksperimen pada dataset, model, dan juga hal praktikal lainnya. Contohnya pada proyek ini seperti menggunakan posisi kamera yang terkontrol sehingga bisa memaksimalkan hasil dari program yang dibuat.

**Gerry Steven - C14180044**

Dari proyek ini, saya belajar cukup banyak dalam *deep learning* khususnya menggunakan YOLOv5 dalam mendeteksi manusia dan pelanggaran jaga jarak. Dari proyek ini saya belajar bagaimana cara menggunakan hingga cara modifikasi YOLOv5 sesuai yang diinginkan. Saya juga belajar dataset apa yang sesuai untuk di *training* sesuai dengan kebutuhan program. Selain itu dalam mengestimasi jarak, saya juga belajar bahwa kita memerlukan sudut, jarak, ketinggian kamera suatu video agar dapat mengestimasikan dengan tepat. Saya juga belajar lebih dalam mengenai computer vision dari proyek ini, dimana faktor apa saja dalam pengolahan video dan bagaimana cara mengatasi permasalahan computer vision. Untuk kedepannya, diharapkan dalam proyek ini melakukan training gambar yang sesuai (seperti dari atas), menggunakan alat pemantau untuk mendapatkan hasil yang akurat (seperti drone), serta menggunakan data video aslinya.

## Michael Alexander Rustan - C14180175

Dari proyek akhir Deep Learning ini, Saya banyak belajar mengenai Training suatu model, perspektif, dan juga ketentuan-ketentuan yang perlu diperhatikan saat kita mengerjakan suatu project. Untuk training model, saya baru belajar saat saya ikut di kelas deep learning ini dan mulai memahaminya saat mengerjakan proyek akhir ini. Dalam training, mungkin bisa menambahkan epoch, dan lebih coba mengatur-ngatur hyperparameter. Pada Proyek ini saya menggunakan bird eye view untuk menghitung jaraknya. Dan dari hasilnya pun menurut saya cukup baik tapi mengingat tidak bisa langsung menghitung jaraknya dalam bentuk cm sehingga saya memerlukan informasi tambahan mengenai posisi dari kameranya. Untuk hasil bird eye view ada beberapa kekurangan yang saya dapat yaitu apabila width dari regionnya lebih lebar daripada heightnya akan membuat hasil bounding boxnya menjadi stretch dan perhitungan jaraknya akan terganggu. Oleh karena itu untuk kedepannya saya berharap dapat menemukan beberapa metode bird eye view untuk mendapatkan hasil yang maksimal.

## 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