

Sistem Pengawasan Physical Distancing di Tempat Umum Menggunakan Kamera Berbasis Deep Learning

Rizqy Arya Dinata*¹, Ika Candradewi², Bambang Nurcahyo Prastowo³

¹Program Studi Elektronika dan Instrumentasi, FMIPA UGM, Yogyakarta, Indonesia

^{2,3}Departemen Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta, Indonesia

e-mail: *¹rizqyarya99@mail.ugm.ac.id, ²ika.candradewi@ugm.ac.id, ³prastowo@ugm.ac.id

Abstrak

Pembatasan jarak fisik merupakan salah satu cara yang diterapkan untuk mencegah penyebaran virus pada tempat umum. Pelaksanaan pembatasan jarak fisik tersebut memerlukan pengawasan agar berhasil sesuai harapan. Pengawasan yang dilakukan secara manual terutama pada tempat dengan tingkat keramaian tinggi kurang efektif karena memerlukan banyak petugas di lokasi yang justru akan menambah keramaian.

Pada penelitian ini dikembangkan purwarupa sistem pengawasan pembatasan jarak fisik dengan memanfaatkan kamera CCTV dengan pemrosesan citra digital berbasis computer vision dan deep learning. Metode yang digunakan adalah kombinasi pendeteksian dan pelacakan pedestrian dengan YOLOv4 dan DeepSORT. Metode trigonometri digunakan dalam proses estimasi jarak untuk mendeteksi pelanggaran physical distancing oleh pedestrian. Pada penelitian ini didapatkan hasil pengujian dengan nilai terbaik recall 0,86; precision 0,69 dan mean average precision (mAP) sebesar 0,83 dengan metode pelatihan transfer learning model YOLOv4 dengan maksimum batch 6000 menggunakan 473 data latih dan 119 data validasi. Keseluruhan sistem mencapai kecepatan rata-rata proses real-time yakni pada 24 sampai 26 FPS.

Kata kunci—Deteksi Obyek, Pelacakan Obyek, Deep Learning

Abstract

Physical distancing is one way to prevent the spread of the virus in public places. The implementation of physical distance restrictions requires supervision to be successful as expected. Supervision that is carried out manually, especially in places with high levels of crowds, is less effective because it requires many officers at the location which will actually increase the crowd.

In this study, a prototype monitoring system for physical distance restrictions was developed by utilizing CCTV cameras with digital image processing based on computer vision and deep learning. The method used is a combination of detection and tracking of pedestrians with YOLOv4 and DeepSORT. Trigonometric methods are used in the distance estimation process to detect violations of distance restrictions by pedestrians. In this study, the test results obtained with the best recall value 0.86; precision 0.69 and mean average precision (mAP) of 0.83 with the YOLOv4 transfer learning model training method with a maximum batch of 6000 using 473 training data and 119 validation data. The entire system achieves an average real-time processing speed of 24 to 26 FPS.

Keywords—Object detection, Object tracking, Deep Learning

1. PENDAHULUAN

Pada Desember tahun 2019 telah ditemukan suatu penyakit yang disebabkan oleh virus korona. Virus tersebut dapat menyebar melalui partikel cairan di udara ataupun yang menempel di permukaan benda yang kemudian terhirup atau masuk ke tubuh seseorang melalui hidung atau mulut [1]. Tempat umum seperti pusat perbelanjaan, taman kota, stasiun, kantor, sekolah dan semacamnya yang biasa dikunjungi atau dilalui banyak orang memiliki risiko sebagai tempat penyebaran wabah yang disebabkan virus tersebut [2].

Hal tersebut membuat hampir seluruh pemerintah di dunia menerapkan larangan keluar rumah untuk menghambat dan bahkan menghentikan penyebaran wabah tersebut. Namun kebijakan tersebut dinilai tidak hanya menghambat penyebaran virus melainkan juga menghambat perekonomian karena mempersulit kegiatan masyarakat untuk melaksanakan kegiatan seperti biasanya [3]. Ketidakpastian kapan berakhirnya wabah yang dinamakan Covid-19 ini mengharuskan orang untuk melakukan adaptasi yang disebut dengan new normal atau normal baru. Di mana orang diharuskan menerapkan beberapa kebiasaan baru tempat umum di antaranya mengenakan masker dan menjaga jarak fisik dengan orang lain yang dikenal sebagai physical distancing [4]. Berdasarkan penelitian yang telah dilaksanakan, physical distancing dapat membantu meningkatkan efektivitas pengendalian penyebaran Covid-19 [5].

Pelaksanaan physical distancing tersebut tentunya selain membutuhkan sosialisasi juga memerlukan pengawasan agar dapat terlaksana dengan baik. Pengawasan physical distancing ini sulit dilakukan secara manual terlebih di tempat keramaian. Dengan demikian dibutuhkanlah suatu sistem yang dapat mempermudah pengawasan, seperti sistem pemantau dengan memanfaatkan kamera pengawas yang terdapat di tempat umum. Kamera pengawas dapat memberikan video yang dapat diproses untuk mendapatkan informasi jarak antara orang.

Pada beberapa penelitian yang dilakukan sebelumnya terdapat berbagai macam pendekatan di antaranya, metode deteksi objek tanpa pelacakan menggunakan YOLOv3 sehingga tidak dapat dilakukan pelacakan pelaku pelanggaran physical distancing [6] [7]; Metode deteksi objek YOLOv3, pelacakan dengan DeepSORT namun dengan menggunakan penghitung jarak relasi inversi ukuran bounding box yang membuat penerapan di tempat berbeda membutuhkan modifikasi persamaan [8]; Metode deteksi objek YOLOv4 dan pelacak SORT yang artinya pelacak tidak memiliki pengenalan fitur yang dapat mengurangi terjadinya pertukaran ID [9].

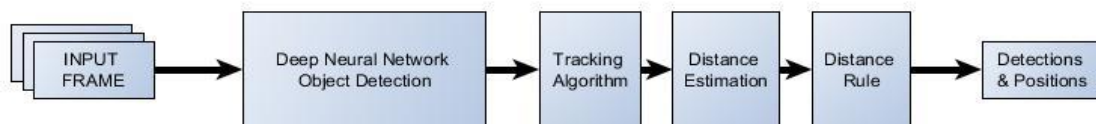
Berangkat dari penjabaran tersebut, penelitian ini mengembangkan sistem pemrosesan citra digital menggunakan metode Deep Learning untuk membuat suatu sistem pengawasan yang dapat mendeteksi terjadinya pelanggaran pelaksanaan physical distancing. Metode Deep Learning yang akan diteliti adalah metode pendeteksian dan pelacakan lokasi orang menggunakan pendeteksi YOLOv4 dikombinasikan dengan pelacakan DeepSORT dan penghitung jarak berbasis perspektif. Metode tersebut dipilih karena memiliki performa yang sesuai dengan kebutuhan sistem yakni kemampuan mendeteksi dan melacak banyak objek secara akurat dalam waktu bersamaan dan waktu pemrosesan real time [10].

2. METODE PENELITIAN

Pada penelitian ini sistem diharapkan untuk dapat memantau pelaksanaan *physical distancing* di tempat umum. Hal tersebut membuat sistem memerlukan kemampuan untuk dapat melakukan pendeteksian pelanggaran *physical distancing*. Pelanggaran yang dimaksud adalah ketika orang memiliki jarak dengan orang lain lebih dekat dari nilai batas jarak yang ditentukan oleh pihak yang berwenang. Pihak yang berwenang yang dimaksud misalnya pengelola tempat atau pemerintah. Kemampuan untuk mendeteksi pelanggaran jarak orang dapat diperoleh dengan berbagai metode. Pada penelitian ini metode yang digunakan adalah berbasis pemrosesan video dengan mengimplementasikan Deep Learning YOLOv4 dan DeepSORT. Metode ini dipilih karena dapat memanfaatkan infrastruktur yang umum ada pada tempat umum yakni kamera CCTV. Kamera CCTV yang terpasang secara tetap di tempat umum biasanya digunakan sebagai alat pemantau keadaan sehingga sesuai bila dapat ditingkatkan kemampuannya dengan

penambahan sistem pengawasan otomatis. Agar dapat mendeteksi pelanggaran jarak orang sistem memerlukan kemampuan untuk mengetahui jarak antara orang. Sistem perlu mengetahui lokasi setiap orang untuk mengetahui jarak antara orang. Orang yang terekam pada CCTV tentunya akan berpindah tempat. Hal ini membuat sistem harus mampu mengetahui tempat orang atau pejalan kaki tersebut sepanjang terlihat oleh kamera. Sehingga dapat disimpulkan bahwa sistem memerlukan kemampuan untuk dapat mendeteksi pejalan kaki atau pedestrian dan melacak perpindahan posisinya untuk kemudian dapat mengetahui jaraknya dengan pejalan kaki lain sehingga sistem dapat mengetahui pedestrian yang melakukan pelanggaran jarak fisik. Proses sistem secara umum dapat dilihat pada

Gambar 1.



Gambar 1 Blok Diagram Sistem

2.1 Bahan dan Peralatan

Penelitian ini menggunakan data citra digital berupa video yang didapatkan dari kamera CCTV yang diekstrak menjadi frame. Frame tersebut kemudian dilabeli dengan perangkat lunak labelimg untuk menjadi dataset. Proses pelatihan model dilakukan menggunakan Darknet yang dijalankan pada laptop Windows 10 dengan spesifikasi prosesor Intel Core i7 9th gen, kartu grafis NVIDIA RTX 2070 mobile, dan RAM 16 GB.

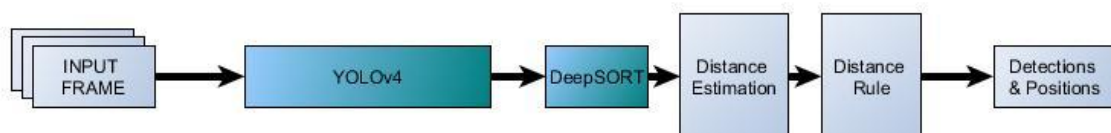
2.2 Pengumpulan Data

Data video yang digunakan pada penelitian ini di dapatkan dari situs <https://cctv.jogjakota.go.id/>. Pada situs tersebut dipilih live stream kamera CCTV untuk direkam kemudian disimpan. Proses perekaman menggunakan aplikasi browser Google chrome dengan ekstensi stream recorder.

Data video tersebut kemudian diekstrak menjadi data citra atau frame. Data citra tersebut kemudian diberikan label untuk menjadi ground truth pada proses pelatihan dan pengujian model. Data yang memiliki ground truth tersebut kemudian dibagi menjadi tiga yakni data uji, latih dan validasi. Data latih dan validasi dibagi dengan perbandingan 20 % data validasi dan 80% data latih. Data uji tidak digunakan pada proses pelatihan model dan hanya digunakan untuk menguji performa model.

2.3 Metode Pendeteksian dan Pelacakan

Tahapan pertama dari sistem adalah pendeteksian dan pelacakan objek. Metode yang diteliti adalah kombinasi metode pendeteksian menggunakan YOLOv4 dengan metode pelacakan menggunakan DeepSORT. Diagram alur dari sistem ini ditunjukkan pada Gambar 2.



Gambar 2 Diagram alur YOLOv4 dengan DeepSORT

Metode pendeteksian menggunakan YOLOv4 memerlukan proses pelatihan menggunakan data yang sesuai dengan data uji. Proses pelatihan model pendeteksi YOLOv4

dilakukan dengan menggunakan darknet. Metode pelacakan DeepSORT juga memerlukan model yang telah dilatih dengan data yang sesuai agar dapat melakukan pelacakan. Namun pada penelitian ini model DeepSORT yang digunakan adalah pre-trained model mars-small128. Model mars-small128 adalah model DeepSORT yang telah dilatih menggunakan dataset MARS. Dataset MARS (Motion Analysis and Re-identification Set) adalah dataset berbasis video untuk proses identifikasi ulang orang. Penggunaan model ini membuat proses pelatihan ulang tidak diperlukan.

2.3.1 Proses Pelatihan Metode Pendeteksian

Proses pelatihan pendeteksi YOLOv4 dilakukan dengan menggunakan datasets yang dibagi menjadi dua yakni data latih dan validasi. YOLOv4 dalam penelitian ini dilatih untuk mendeteksi sebuah objek saja yakni manusia khususnya pedestrian. Hasil dari pendeteksian menggunakan YOLOv4 adalah bounding box sesuai dengan lokasi setiap pedestrian. Data bounding box tersebut kemudian diteruskan ke tahap selanjutnya yakni pelacakan dengan metode DeepSORT. Proses pelatihan model pendeteksi dilakukan dengan bantuan implementasi darknet pada sistem operasi windows oleh Alexey Bochkovskiy [11].

2.3.2 Kombinasi Metode Pendeteksian dan Pelacakan

Proses pelacakan menggunakan DeepSORT akan menerima keluaran dari proses pendeteksian yakni sejumlah informasi *bounding box* pada frame hasil deteksi pejalan kaki YOLOv4. Informasi yang menjadi fokus utama pada proses ini adalah informasi lokasi *bounding box* dan penampakan pejalan kaki. DeepSORT akan mengasosiasikan hasil deteksi pejalan kaki dengan hasil pendeteksian pada beberapa frame sebelumnya. Dengan demikian sistem dapat mengetahui pergerakan setiap pejalan kaki pada frame. Data *bounding box* pejalan kaki yang telah terlacak ini kemudian diteruskan ke proses selanjutnya yakni estimasi jarak antar pejalan kaki.

2.4 Metode Estimasi Jarak dan Pendeteksian Pelanggaran

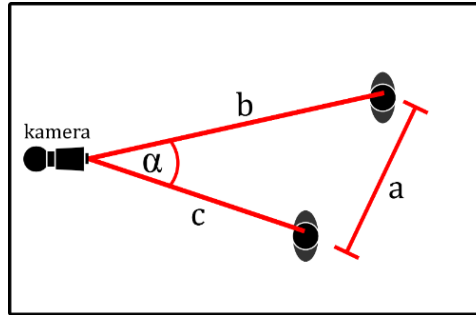
Tahap penghitungan jarak antara pejalan kaki dilakukan setelah didapatkan lokasi bounding box hasil pelacakan setiap pejalan kaki. Jarak antar pejalan kaki dihitung dengan menggunakan informasi yang diketahui yakni spesifikasi, posisi dan orientasi kamera.

2.4.1 Metode Estimasi Jarak

Distance Estimation atau perkiraan jarak dilakukan dengan menggunakan informasi yang diketahui yakni tinggi posisi kamera terhadap lantai, dan posisi penampakan objek pada kamera yang diterjemahkan menjadi sudut. Jarak antara objek dihitung dengan menggunakan rumus trigonometri yang dituliskan pada persamaan **Error! Reference source not found.**

$$a = \sqrt{b^2 + c^2 - 2bc \cos\alpha} \quad (1)$$

Di mana a adalah jarak antara objek, b adalah jarak objek pertama dari kamera, c adalah jarak objek kedua dari kamera dan α adalah sudut antara kedua objek yang didapatkan pada kamera sesuai dengan ilustrasi pada Gambar 3. Sudut α didapatkan dari sudut objek yang di deteksi dengan menerjemahkan posisi horizontal piksel pada citra digital menjadi sudut berdasarkan spesifikasi sudut horizontal field of view kamera.



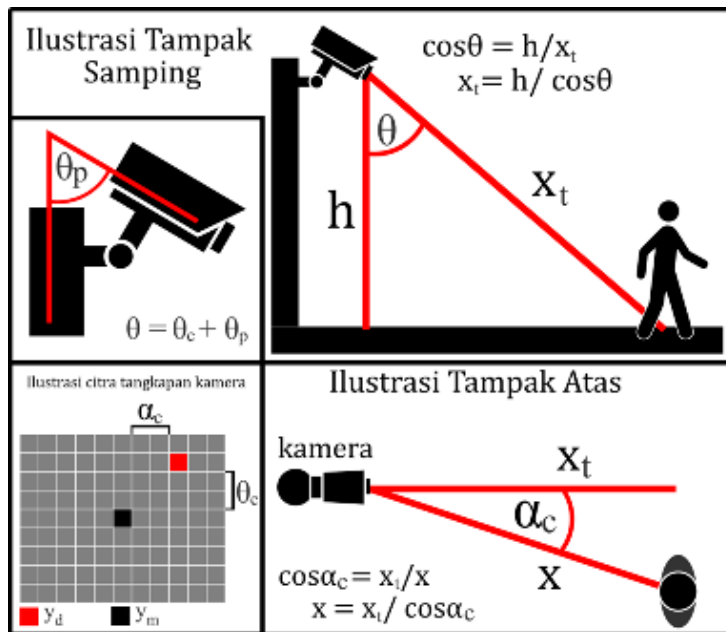
Gambar 3 Ilustrasi jarak antara dua orang

Untuk mendapatkan jarak objek dari kamera dapat menggunakan teknik trigonometri yang dituliskan pada persamaan **Error! Reference source not found.** dan (3).

$$x = \frac{h}{\cos\theta} \tag{2}$$

$$\text{Jarak objek ke kamera} = x = \frac{x_t}{\cos\alpha_c} \tag{3}$$

Di mana x adalah jarak objek ke kamera, h adalah tinggi kamera dari permukaan tanah, x_t adalah jarak dari kamera ke titik di kolom tengah citra yang sejajar dengan objek (nilai θ yang sama), θ adalah sudut objek pada kamera antara garis vertikal kamera dengan permukaan tanah, dan α_c adalah sudut horizontal objek pada citra seperti ditunjukkan pada Gambar 4.



Gambar 4 Ilustrasi jarak antara lokasi kamera dengan pedestrian

Sudut θ didapatkan dengan persamaan $\theta = \theta_c + \theta_p$, di mana θ_p adalah sudut pitch kamera, θ_c adalah sudut pitch objek terdeteksi berdasarkan perspektif. θ_c dihitung menggunakan persamaan (4).

$$\theta_c = \frac{(y_m - y_d) * \theta}{y} \tag{4}$$

di mana, y_m adalah baris tengah citra. y_d adalah posisi koordinat vertikal objek terdeteksi pada citra. θ adalah *field of view* vertikal kamera. y adalah dimensi vertikal citra.

Menggunakan informasi tersebut dapat dihitung jarak pejalan kaki dengan posisi kamera. Kemudian dari informasi jarak antara pejalan kaki dengan kamera tersebut dapat dihitung jarak setiap pejalan kaki dengan pejalan kaki lainnya. Dengan demikian sistem dapat melanjutkan ke tahap berikutnya yakni mendeteksi pelanggaran

2.4.2 Metode Pendeteksian Pelanggaran

Pendeteksian pelanggaran dilakukan dengan menggunakan sebuah nilai ambang batas jarak antara pejalan kaki. Nilai ambang batas tersebut dapat ditentukan berdasarkan ketentuan pihak berwenang. Pada penelitian ini digunakan nilai ambang batas 2 m. Pelanggaran didefinisikan sebagai sepasang pejalan kaki yang jarak antar keduanya kurang dari nilai ambang batas. Sistem akan menandai kedua pejalan kaki tersebut dengan warna *bounding box* berbeda dari orang yang tidak melanggar.

3. HASIL DAN PEMBAHASAN

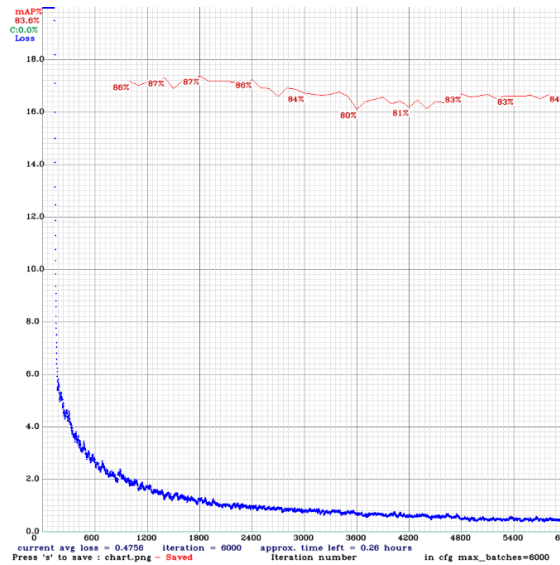
3.1 Dataset

Data video yang didapatkan dipisahkan menjadi dua yakni video untuk proses pelatihan model dan video sebagai data uji. Video untuk pelatihan model diekstrak menjadi 592 frame citra digital dalam format jpg dan diberi label berformat txt dibagi menjadi dua yakni data latih dan data validasi. Data latih berjumlah 473 dan data validasi berjumlah 119. Pembagian data tersebut disimpan pada file *train.txt* dan *validation.txt*. Sedangkan data yang digunakan untuk proses pengujian terdiri dari dua video berbeda. Kedua video berasal dari CCTV Mutiara Lama mengarah ke kanan dan CCTV Depan Dispar mengarah ke kiri. Video dari CCTV Mutiara Lama diekstrak menjadi 1230 frame. Video dari CCTV depan Dispar diekstrak menjadi 926 frame. Keduanya kemudian diberikan anotasi sebagaimana data latih. Gabungan kedua data uji ini kemudian dinamakan data test dengan total berjumlah 2156 frame.

3.2 Hasil Pendeteksian dan Pelacakan

3.2.1 Hasil Pelatihan Pendeteksi

Pada proses pelatihan model pendeteksi digunakan data latih sebanyak 473 dan data validasi sebanyak 119. Proses pelatihan model pendeteksi YOLOv4 menggunakan teknik transfer learning. Bobot pretrained yang digunakan dapat diakses pada <https://github.com/AlexeyAB/darknet>. Proses pelatihan memakan waktu sekitar 20 jam tanpa henti. Proses pelatihan menghasilkan bobot baru model yang bisa digunakan untuk melakukan deteksi pedestrian. Selama proses pelatihan setiap 1000 batch bobot model disimpan. Program juga menyimpan secara otomatis bobot terbaik, terakhir, dan final. Bobot terakhir digunakan untuk melanjutkan progres pelatihan apabila proses pelatihan terputus sebelum tuntas. Bobot akhir adalah bobot model setelah proses pelatihan selesai. Bobot terbaik adalah bobot dengan nilai average loss terendah. Sehingga proses pelatihan menghasilkan delapan file bobot model. Selama proses pelatihan program juga menghasilkan grafik performa model. Grafik tersebut menampilkan nilai average loss setiap batch dan nilai mAP setiap batch yang mulai dihitung dari batch ke 1000.



Gambar 5 Hasil Pelatihan YOLOv4

Proses pelatihan model YOLOv4 berlangsung selama 16 jam tanpa henti. Berdasarkan data hasil pelatihan model pendeteksi didapatkan bahwa model mampu mendeteksi pedestrian dengan baik. Data hasil proses pelatihan yang ditampilkan pada

Gambar 5 menunjukkan bahwa model berhasil mengurangi nilai average loss secara signifikan pada batch awal pelatihan yakni dari batch pertama sampai batch ke-600. Penurunan nilai average loss juga berlanjut secara konsisten setelah batch ke-600 sampai batch terakhir. Berdasarkan hasil pelatihan yang ditampilkan pada grafik juga terlihat progres nilai mAP model yang fluktuatif pada kisaran nilai 80% sampai 87%. Pada batch terakhir pelatihan model memiliki nilai average loss 0,4756 dan dengan nilai mAP sebesar 83,6%. Proses pelatihan menghasilkan delapan bobot dengan nama file berawalan Y4DB5xx dan diikuti akhiran pembeda yakni nilai batch atau kategori bobot. Bobot tersebut antara lain “Y4DB5xx_1000.weights”, “Y4DB5xx_2000.weights”, “Y4DB5xx_3000.weights”, “Y4DB5xx_4000.weights”, “Y4DB5xx_5000.weights”, “Y4DB5xx_6000.weights”, “Y4DB5xx_best.weights”, “Y4DB5xx_final.weights”, dan “Y4DB5xx_last.weights”.

3.2.2 Hasil Pengujian Pendeteksi

Proses pengujian model pendeteksi dilakukan menggunakan data yang tidak digunakan pada proses pelatihan. Data uji yang digunakan berasal dari dua CCTV yakni CCTV depan Dispar yang mengarah ke kiri dan CCTV Mutiara Lama yang mengarah ke kanan. Model pendeteksi diuji terhadap masing-masing data uji tersebut dan juga kombinasi keduanya. Hasil pengujian dari CCTV depan Dispar diberikan akhiran dsprL. Hasil pengujian dari CCTV Mutiara Lama diberikan akhiran mtrIR. Sedangkan pengujian dengan kombinasi kedua data uji tersebut diberikan akhiran test. Hal ini dilakukan untuk memperpendek penamaan. Proses pengujian dilakukan terhadap dua bobot model yang dihasilkan proses pelatihan yakni bobot Y4DB5xx_best dan Y4DB5xx_final. Sampel hasil pengujian pendeteksi terhadap data uji dapat dilihat pada Gambar 6 dan Gambar 7.



Gambar 6 Pengujian bobot Y4DB5xx_final



Gambar 7 Pengujian bobot Y4DB5xx_best

Tabel 1 Hasil Pengujian Pendeteksi

No.	Hasil Uji Model	Precision	Recall	F1 score	TP	FP	FN	average IoU	mAP
1	Y4DB5xx_best mtrlR	0,75	0,91	0,82	3859	1287	364	61,91	0,917836
2	Y4DB5xx_best dsprL	0,63	0,79	0,7	2880	1702	762	47,59	0,72941
3	Y4DB5xx_best test	0,69	0,86	0,77	6739	2989	1126	55,17	0,837324
4	Y4DB5xx_final mtrlR	0,63	0,87	0,73	3691	2197	532	53,07	0,856828
5	Y4DB5xx_final dsprL	0,67	0,66	0,67	2415	1206	1227	51,1	0,636585
6	Y4DB5xx_final test	0,64	0,78	0,7	6106	3403	1759	52,32	0,754444

Berdasarkan hasil pengujian pendeteksi seperti yang terlihat pada Tabel dapat dianalisis bahwa nilai Precision setiap bobot model berada di atas 0,6. Kedua bobot yang diuji memiliki nilai Precision terbaik yakni 0,75 untuk bobot Y4DB5xx_best pada pengujian dengan data dari CCTV Mutiara Lama dan 0,67 untuk bobot Y4DB5xx_final pada pengujian dengan data dari CCTV Depan Dispar. Dapat terlihat juga bahwa bobot final memiliki nilai Precision lebih tinggi dari pada bobot Y4DB5xx_best pada pengujian dengan data dari CCTV Depan Dispar. Hasil pengujian dengan data test menunjukkan bahwa bobot Y4DB5xx_best memiliki performa terbaik yakni dengan nilai recall 0,86 dibandingkan dengan 0,78 pada bobot Y4DB5xx_final. Precision dengan nilai 0,69 dibandingkan dengan 0,64. F1-score dengan nilai 0,77 dibandingkan dengan 0,7. Dan nilai mAP 0,83 dibandingkan dengan 0,75 pada bobot Y4DB5xx_final. Kedua bobot model memiliki nilai mAP terbaik pada pengujian dengan data dari CCTV mutiara lama. Nilai mAP bobot best lebih baik dibandingkan bobot final dalam ketiga pengujian. Melalui hasil ini dapat diketahui bahwa bobot Y4DB5xx_best memiliki performa yang lebih baik dibandingkan bobot Y4DB5xx_final. Sehingga bobot Y4DB5xx_best yang digunakan sebagai bobot model pendeteksi.

3.2.3 Hasil Pengujian Kombinasi Pendeteksi dan Pelacak

Pengujian pendeteksi dan pelacak dilakukan untuk mengetahui performa metode pendeteksi dan pelacak yang menghasilkan masukan untuk estimasi jarak. Proses pengujian pelacak dilakukan setelah dikombinasikan dengan pendeteksi karena metode pelacak tidak dapat bekerja sendiri tanpa masukan dari metode pendeteksi.



Gambar 8 Hasil Pengujian Kombinasi Pendeteksi dan Pelacak

Berdasarkan hasil pengujian kombinasi pendeteksi dan pelacak yang dapat dilihat sampelnya pada Gambar 8 didapatkan bahwa pelacak dapat melakukan pelacakan id pedestrian dengan cukup baik. Hal ini ditandai dengan warna penanda yang ditentukan oleh nilai id pedestrian konsisten dan menghasilkan jejak garis pergerakan setiap pedestrian. Namun pelacak masih memiliki kelemahan ketika pedestrian berdekatan dengan waktu yang cukup lama dengan arah pergerakan yang seragam. Kelemahan ini dapat terlihat pada Gambar 8 di mana terjadi pertukaran id pedestrian ketika sepasang pedestrian berjalan searah secara berdekatan. Setelah dianalisis lebih lanjut pertukaran identitas pedestrian juga terjadi pada pedestrian yang berjarak cukup jauh dari kamera. Jumlah perubahan identitas obyek terlacak pada pengujian dengan data CCTV Depan Dispar dapat dilihat pada **Error! Reference source not found..** Berdasarkan **Error! Reference source not found.** dapat diketahui bahwa kombinasi nilai variabel $\max_cosine_distance$ dan nn_budget terbaik adalah 0,4 dengan 40.

3.3 Hasil Pengujian Estimasi jarak dan Deteksi Pelanggaran

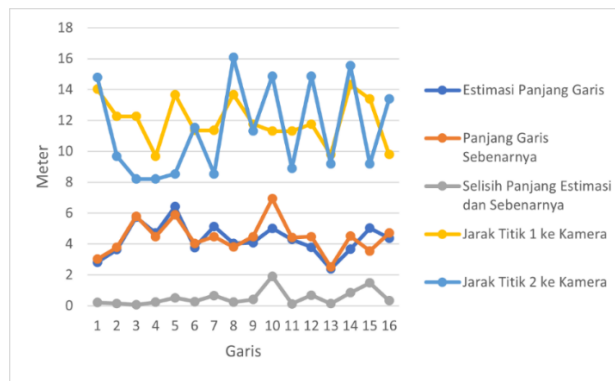
Pengujian komponen estimasi jarak sistem yang dilakukan dengan membandingkan hasil estimasi panjang 16 garis pada 4 lokasi CCTV berbeda menunjukkan bahwa sistem mampu memberikan hasil estimasi mendekati nilai sebenarnya. Hasil estimasi panjang garis dapat dilihat pada Tabel 2.

Tabel 2 Hasil Pengujian Estimasi Jarak

No.	LOKASI	GARIS	Jarak Titik ke Kamera		Panjang Garis (jarak kedua titik)		Selisih Panjang Garis
			Titik 1	Titik 2	Estimasi	Sebenarnya	
1	Depan Dispar -Kiri	A	14,03	14,8	2,81	3,02	0,21
2		B	12,26	9,69	3,63	3,78	0,15
3		C	12,26	8,20	5,74	5,80	0,06
4		D	9,69	8,20	4,71	4,48	0,23
5	Depan Toko Subur -Kiri	A	13,67	8,53	6,42	5,90	0,52
6		B	11,36	11,54	3,77	4,04	0,27
7		C	11,36	8,53	5,14	4,48	0,66
8		D	13,67	16,08	4,03	3,80	0,23
9		A	11,76	11,31	4,08	4,48	0,40

10	Mutiara Lama -Kanan	B	11,31	14,87	5,02	6,93	1,91
11		C	11,31	8,90	4,29	4,41	0,12
12		D	11,76	14,87	3,79	4,48	0,69
13	Perwakilan -Kanan	A	9,79	9,18	2,38	2,51	0,13
14		B	14,32	15,56	3,66	4,51	0,85
15		C	13,39	9,18	5,03	3,55	1,48
16		D	9,79	13,39	4,38	4,71	0,33

Berdasarkan hasil pengujian estimasi jarak didapatkan bahwa sistem dapat mengestimasi jarak sebenarnya secara baik. Dari 16 garis yang diestimasi panjangnya hanya tiga yang memiliki selisih lebih besar dari 1 meter dengan nilai sebenarnya. Perbedaan yang cukup besar ini dapat disebabkan oleh kesalahan pada saat pengukuran *ground truth* atau dapat juga disebabkan oleh kurang tepatnya parameter sistem. Perbedaan hasil estimasi dengan nilai sebenarnya dapat dilihat dengan jelas pada Gambar 9.

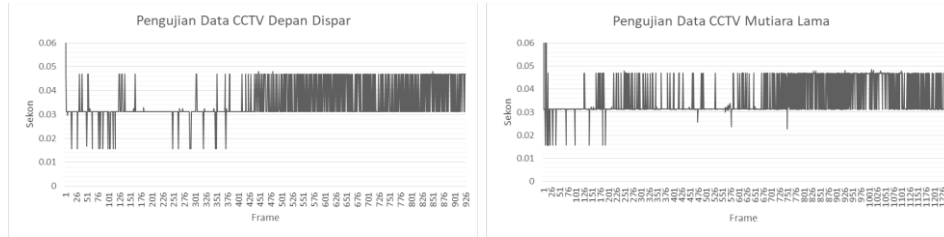


Gambar 9 Perbedaan Hasil Estimasi dengan Nilai Sebenarnya

Berdasarkan hasil pengujian estimasi jarak dapat dianalisis bahwa hasil estimasi jarak sistem dipengaruhi oleh posisi titik yang diukur. Hal ini karena sistem menggunakan koordinat piksel dan mengubahnya menjadi sudut untuk kemudian menghitung jarak dengan teknik trigonometri. Semakin jauh posisi aslinya maka hasil estimasi jarak sistem akan berkurang ketelitiannya.

3.4 Hasil Pengujian Sistem

Pengujian sistem dilakukan untuk mengetahui performa seluruh komponen sistem setelah diintegrasikan menjadi satu. Sistem diuji untuk mengetahui kecepatan proses secara keseluruhan sistem. Setelah dilakukan proses pengujian dapat diketahui bahwa sistem mampu memproses satu frame dalam waktu kurang lebih 0,3 sekon. Pada pengujian dengan data CCTV Depan Dispar didapatkan waktu pemrosesan tercepat 0,015587 sekon dan waktu terlama 4,62506 sekon dengan rata-rata waktu 0,040035 sekon. Pada pengujian dengan data CCTV Mutiara Lama didapatkan waktu pemrosesan tercepat 0,015601 sekon dan waktu terlama 2,434757 sekon dengan rata-rata waktu 0,037882 sekon. Waktu pemrosesan terlama terjadi pada frame awal video. Hal ini bisa disebabkan karena sistem masih melakukan inisialisasi. Hasil pengujian ini dapat dilihat pada Gambar 10. Berdasarkan hasil pengujian ini juga dapat diketahui bahwa sistem dapat memproses dengan kecepatan rata-rata 24 sampai 26 frame per sekon.



Gambar 10 Hasil Pengujian Kecepatan Pemrosesan Sistem



Gambar 11 Hasil Pengujian Sistem

Sistem juga menghasilkan luaran berupa video dengan anotasi. Hasil pengujian sistem menunjukkan bahwa sistem berhasil memberikan penanda pada setiap pelanggar dengan baik. Sebagaimana terlihat pada

Gambar 11, pelanggar jarak diberikan bounding box berwarna merah sedangkan pedestrian lain berwarna hijau.

4. KESIMPULAN

Berdasarkan hasil dari pengujian sistem pengawasan *physical distancing* di tempat umum menggunakan kamera berbasis deep learning yang telah dilakukan pada penelitian ini, dapat ditarik kesimpulan bahwa sistem mampu melakukan proses deteksi dengan algoritma YOLOv4 dengan nilai *mAP* (*mean average precision*) terbaik sebesar 0,83 pada data uji dengan maksimum batch pelatihan sebanyak 6000 menggunakan 473 data latih dan 119 data validasi. Sistem dapat melakukan proses deteksi, pelacakan, estimasi jarak, dan deteksi pelanggaran *physical distancing* dengan kecepatan rata-rata real-time yakni pada 24 sampai 26 FPS.

5. SARAN

Saran untuk lanjutan atau pengembangan dari penelitian ini antara lain mengombinasikan sistem dengan algoritma identifikasi ulang dengan mengintegrasikan banyak kamera agar sistem dapat melakukan pengawasan secara menyeluruh pada area yang luas.

DAFTAR PUSTAKA

- [1] T. Singhal, "A Review of Coronavirus Disease-2019 (COVID-19)," *Indian J Pediatr*, vol. 87, no. 4, pp. 281–286, Apr. 2020, doi: 10.1007/s12098-020-03263-6.
- [2] L. Morawska *et al.*, "How can airborne transmission of COVID-19 indoors be minimised?," *Environment International*, vol. 142, p. 105832, 2020, doi: <https://doi.org/10.1016/j.envint.2020.105832>.
- [3] L. Laeven, "COVID-19 and the effects of social distancing on the economy," Aug. 31, 2020. <https://voxeu.org/article/covid-19-and-effects-social-distancing-economy> (accessed Apr. 26, 2022).
- [4] World Health Organization, "Coronavirus disease (COVID-19) advice for the public," Dec. 29, 2020. <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public> (accessed Jan. 04, 2021).
- [5] A. Kucharski *et al.*, "Effectiveness of isolation, testing, contact tracing, and physical distancing on reducing transmission of SARS-CoV-2 in different settings: a mathematical modelling study," *The Lancet Infectious Diseases*, vol. 20, Jun. 2020, doi: 10.1016/S1473-3099(20)30457-6.
- [6] V. Vinitha and V. Velantina, "Social Distancing Detection System With Artificial Intelligence Using Computer Vision And Deep Learning," *International Research Journal of Engineering and Technology*, vol. 7, no. 8, pp. 4049–4053, Aug. 2020, Accessed: Apr. 26, 2022. [Online]. Available: <https://www.irjet.net/archives/V7/i8/IRJET-V7I8698.pdf>
- [7] Y. C. Hou, M. Z. Baharuddin, S. Yussof, and S. Dzulkifly, "Social Distancing Detection with Deep Learning Model," in *2020 8th International Conference on Information Technology and Multimedia (ICIMU)*, 2020, pp. 334–338. doi: 10.1109/ICIMU49871.2020.9243478.
- [8] N. S. Punn, S. K. Sonbhadra, S. Agarwal, and G. Rai, "Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques," May 2020.
- [9] M. Rezaei and M. Azarmi, "DeepSOCIAL: Social Distancing Monitoring and Infection Risk Assessment in COVID-19 Pandemic," Aug. 2020, doi: 10.3390/app10217514.
- [10] M. Aryal, "Object Detection, Classification, and Tracking for Autonomous Vehicle," Grand Valley State University, 2018. Accessed: Apr. 26, 2022. [Online]. Available: <https://scholarworks.gvsu.edu/theses/912>
- [11] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 2020.