

**JOURNAL OF INFORMATION
SYSTEM AND TECHNOLOGY
MANAGEMENT (JISTM)**www.jistm.com**SISTEM PEMANTAUAN TRAFIK MELALUI PENGESANAN DAN
PENGIRAAN KENDERAAN DENGAN PEMBELAJARAN
MENDALAM**Oh Shi Chew^{1*}, Mohammad Faizul Nasrudin¹¹ Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor Darul Ehsan, Malaysia

Email: ohshichew@hotmail.com

* Corresponding Author

Article Info:**Article history:**

Received date: 10.11.2023

Revised date:

Accepted date:

Published date:

To cite this document:

.

DOI:This work is licensed under [CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)**Abstract:**

Jumlah kenderaan di Malaysia telah meningkat sejak kebelakangan ini disebabkan oleh peningkatan populasi di bandar. Perkara ini telah menyebabkan kesesakan lalu lintas terutamanya di kawasan bandar. Untuk menyelesaikan masalah ini, sistem pemantauan trafik berasaskan vision yang dapat mengesan dan mengira bilangan kenderaan akan dibangunkan melalui YOLO sebagai model pembelajaran mendalam. Melalui pengiraan bilangan kenderaan, maklumat tersebut dapat membantu dalam menambah baik sistem perancangan dan pengurusan trafik. Projek ini bertujuan untuk meningkatkan ketepatan pengesanan dan pengiraan kenderaan untuk jalan dua hala. Kajian ini dimulakan dengan mengumpul pelbagai imej trafik di Malaysia sebagai set data. Seterusnya, model YOLOv5, YOLOv6, YOLOv7 dan YOLOv8 akan dilatih dengan set data latihan praproses. Seterusnya, model yang dilatih akan dinilai dengan set data ujian untuk menentukan model optimum berdasarkan ketepatan dan kelajuan pengesanan. Akhir sekali, model ini diintegrasikan ke dalam sistem berasaskan web dengan Flask API agar dapat diakses oleh pengguna. Dalam API, algoritma DeepSORT telah dimasukkan sebagai penjejak objek berbilang untuk mengekalkan pengesanan pada setiap bingkai agar mendapatkan pengiraan yang tepat. Sistem ini akan menerima video trafik untuk jalan dua hala sebagai input dan memberikan bilangan setiap kelas sebagai output. Hasilnya, model YOLOv8 yang mencapai mAP0.5 sebanyak 0.885, mAP0.5-0.95 sebanyak 0.656 dan FPS sebanyak 92 telah dipilih sebagai model optimum. Model ini juga diuji dengan 5 video trafik dan mencapai ketepatan pengiraan dari julat 86.11% hingga 100%.

Keywords:

Sistem pemantauan trafik, YOLO, pengesanan dan pengiraan kenderaan.

Pengenalan

Proses urbanisasi di Malaysia telah menyebabkan peningkatan populasi di bandar. Mengikut statistik dari Jabatan Perangkaan Malaysia (2022), kadar urbanisasi di Malaysia meningkat kepada 75.1 peratus (24.4 juta penduduk) pada 2020 berbanding 70.9 peratus (19.5 juta penduduk) pada 2010. Peningkatan populasi di kawasan bandar telah menyebabkan pemilik kenderaan sendiri meningkat. Untuk mengurangkan kepadatan kenderaan di kawasan bandar. Abdullah, N. et al. (2007) mencadangkan bahawa pengangkutan awam yang efisien haruslah diwujudkan di kawasan bandar supaya dapat menarik perhatian pemilik kenderaan. Namun, keutamaan pejalan kaki, pengguna basikal dan bus masih kurang dipentingkan. Perkara ini telah mendorong penduduk di kawasan bandar lebih cenderung menggunakan kenderaan sendiri berbanding pengangkutan awam untuk ulang alik. Kesannya, masalah kesesakan lalu lintas kerap berlaku di kawasan bandar. Hal ini kerana jalan raya di kawasan bandar tidak memenuhi keperluan pengguna jalan raya. Menurut Morikawa, T. et al. (2001), pembinaan jalan raya adalah kerja yang bukan senang terutamanya di Kuala Lumpur. Oleh itu, alternatif lain diperlukan untuk menangani masalah kesesakan lalu lintas di kawasan bandar.

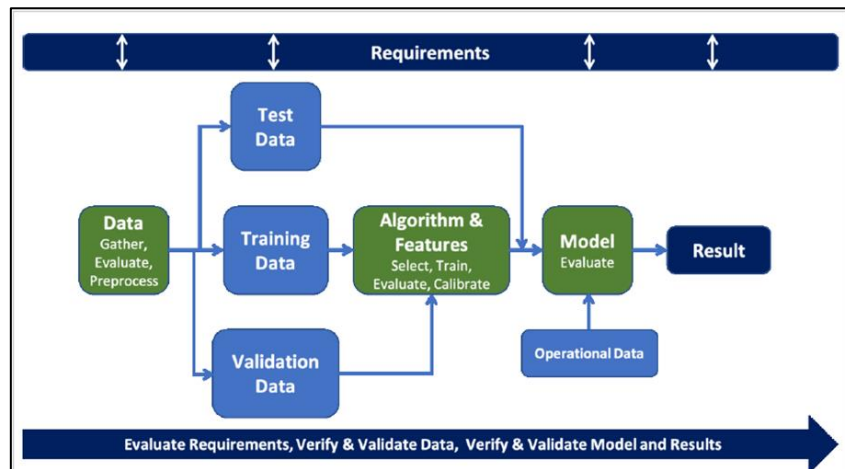
Sistem pemantauan trafik merupakan salah satu penyelesaian untuk masalah tersebut. Sistem ini boleh digunakan untuk mengumpul maklumat aliran trafik dari semasa ke semasa dengan menganalisis video trafik untuk mendapatkan maklumat seperti jenis kenderaan dan bilangannya. Maklumat tersebut dapat digunakan oleh perancang trafik untuk merancang strategi pengurusan dan penyuraian trafik.

Pengesanan objek merupakan fungsi yang sangat asas untuk membangunkan satu sistem yang berteraskan visi komputer. Pada masa ini, pembelajaran mendalam digunakan secara luas untuk fungsi pengesanan objek dan ia boleh dibahagikan kepada dua rangka kerja, iaitu algoritma pengesanan dua fasa dan algoritma pengesanan satu fasa. Algoritma pengesanan dua fasa seperti R-CNN terdiri daripada dua fasa, iaitu mencadangkan kawasan yang mengandungi objek dalam imej dan seterusnya menghitung kawasan tersebut dan mengklasifikasikan objek melalui Convolutional Neural Network (CNN). Menurut Xie, H. et al. (2023), algoritma ini menghasilkan pengesanan yang tepat tetapi kecekapan komputasional lemah dan tidak sesuai digunakan untuk data masa nyata. Algoritma pengesanan satu fasa seperti YOLO dan SSD terus meramal lokasi objek dan jenisnya, ia lebih efisien berbanding dengan algoritma pengesanan satu fasa dalam keadaan masa nyata (Xie, H. et al. 2023).

Dalam projek ini, kaedah mengira yang berdasarkan pengesanan kenderaan sangat bergantung pada ketepatan pengesanan kenderaan agar pertambahan pengiraan berlaku dalam setiap pengesanan. Pengiraan berdasarkan penjejakan memberi anggaran bilangan kenderaan di jalan raya dengan menganalisis pergerakan objek dari satu bingkai ke bingkai seterusnya dan mengenal pasti kenderaan tersebut dengan lorong atau lokasi tertentu. Untuk mendapatkan bilangan kenderaan yang tepat, algorithm seperti SORT atau DeepSORT boleh digunakan untuk penjejakan kenderaan.

Untuk menyelesaikan masalah tersebut, tiga objektif kajian telah ditetapkan. Antaranya ialah membangunkan pelbagai model pembelajaran mendalam seperti YOLO dan mengenal pasti model yang paling efektif terhadap set data latihan khusus untuk trafik Malaysia. Seterusnya, meningkatkan ketepatan pengesanan (mAP) dan pengiraan kenderaan untuk jalan raya dua hala. Akhir sekali, seni bina yang sesuai perlu digunakan untuk mengintegrasikan model pembelajaran mendalam dengan sistem berasaskan web yang berlainan persekitaran.

Metodologi



Rajah 1. Kitar hidup umum AI

Sistem pemantauan ini akan dibangunkan mengikut metodologi kitaran hidup umum AI berdasarkan kerja De Silva & Alahakoon (2022). Metodologi ini diimplementasi dalam projek AI yang bercabaran untuk mengurangkan risiko kegagalan sistem dengan mengenal pasti dan menyelesaikan masalah yang mungkin berlaku secepatnya. Terdapat 4 topik utama yang akan dibentangkan dalam fasa metodologi; iaitu penyediaan set data, latih model, penilaian model dan penggunaan model.

Penyediaan Set Data

Set data telah dikumpul dengan aplikasi web Roboflow. Aplikasi ini mengandungi pelbagai set data imej untuk tujuan pengesanan objek secara sumber terbuka. Aplikasi ini membolehkan pengguna terus mengimport imej sebarang format anotasi dan mengeksport kepada format yang diinginkan. Hal ini memudahkan proses pengumpulan data kerana setiap model YOLO menggunakan format anotasi yang berlainan. Terdapat 6 kelas dalam imej yang dikumpul, iaitu basikal, bas, kereta, motorsikal, pejalan kaki dan lori. Set data ini mengandungi 1125 imej beranotasi yang berkaitan dengan trafik di Malaysia. Pada awalnya, set data telah dibahagikan dengan set data latihan sebanyak 645 imej, set data pengesanan sebanyak 240 imej dan set data ujian sebanyak 240 imej.

Roboflow juga boleh melakukan pemprosesan imej dan penambahan data dengan mudah. Beberapa teknik pemprosesan imej untuk mengurangkan kos komputasional dan meningkatkan prestasi model. Saya telah mengubah saiz setiap imej kepada 640*640 kerana saiz imej dalam set data adalah tidak rata. Selain itu, mengubah imej kepada saiz yang lebih kecil boleh mengurangkan kos komputasional. Saya juga menggunakan kaedah contrast stretching untuk mempertingkatkan kontur dalam imej agar ciri pinggir dapat diekstrak oleh lapisan konvolusi. Bagi penambahan data pula, proses ini akan menambah bilangan set data latihan dengan menggunakan pelbagai teknik pemprosesan imej terhadap set data latihan. Penambahan data secara fotometrik iaitu mengubah saluran warna dalam ruang warna HSL telah digunakan. Komponen dalam HSV seperti warna primer (*Hue*), ketepuan imej (*Saturation*) yang mewakili ketulenan warna dalam imej telah dimanipulasi semasa penambahan data. Dengan memperkenalkan lebih banyak variasi warna melalui manipulasi HSV, model YOLO dapat melakukan generalisasi yang lebih baik pada data umum agar dapat mengelakkan masalah overfitting (Shorten & Khoshgoftaar, 2019). Namun, penambahan data imej fotometrik

mungkin boleh menjejaskan prestasi model apabila ciri warna adalah penting untuk membezakan antara kelas yang berbeza dalam set data (Khalifa et al., 2021). Oleh itu, julat perubahan warna tidak boleh terlalu ketara dan menyebabkan penghapusan ciri penting. Saya telah mengubah warna primer imej dari -15° hingga $+15^{\circ}$, ketepuan imej dalam julat -10% hingga $+10\%$. Selain itu, dedahan imej (exposure) juga diubah dari -10% dan $+10\%$ untuk menahan pencahayaan kamera. Di samping itu, pengaburan dengan Gaussian Blur akan melemahkan hingar dalam imej dengan penapis dari 0 px hingga 1.0 px. Jika penapis terlalu kuat, ia akan menyebabkan kehilangan informasi penting dalam imej dan informasi ini akan gagal diekstrak oleh lapisan konvolusi. Selepas pemprosesan imej dan penambahan data, set data latihan telah tiga kali ganda menjadi sebanyak 1935 imej. Nisbah bagi set data latihan, set data pengesahan dan set data ujian telah menjadi 8:1:1 (<https://app.roboflow.com/project-ch1aj/vehicle-detection-swchc/8>).

Latih Model

Seterusnya, set data latihan akan digunakan untuk melatih model melalui CLI (*Command Line Interface*) di Google Colab. Melalui perintah CLI, *train.py* akan dipanggil untuk melakukan proses latihan. Model *yolov8s.pt* telah digunakan sebagai berat pralatih (*pretrained weight*) untuk melatih model YOLOv8. Hiperparameter yang ditetapkan ialah imej saiz dalam 640×640 , epoch untuk melatih model ialah 50 kali dan *batch_size* ialah 16. Model lain seperti YOLOv5, YOLOv6 dan YOLOv7 akan dilatih dengan hiperparameter yang sama tetapi dengan berat pralatih yang berbeza.

Jadual 1

Berat pralatih dan bilangan parameter model YOLO

Model	Berat Pralatih	Bilangan parameter
YOLOv5	YOLOv5s.pt	7.03M
YOLOv6	YOLOv6s.pt	17.2M
YOLOv7	YOLOv7_training.pt	37.2M
YOLOv8	YOLOv8s.pt	11.1M

Jika saiz berat pralatih besar, bermakna berat itu mempunyai bilangan parameter yang tinggi dan mAP model akan dipertingkatkan. Namun, model ini akan lebih lambat semasa menjalankan inferens terhadap imej. Setelah model tamat latihan pada epoch tersebut, epoch itu akan diuji dengan set data pengesahan. Setelah proses latih model tamat, epoch yang mendapat prestasi yang terbaik akan disimpan dalam *best.pt*.

Penilaian Model

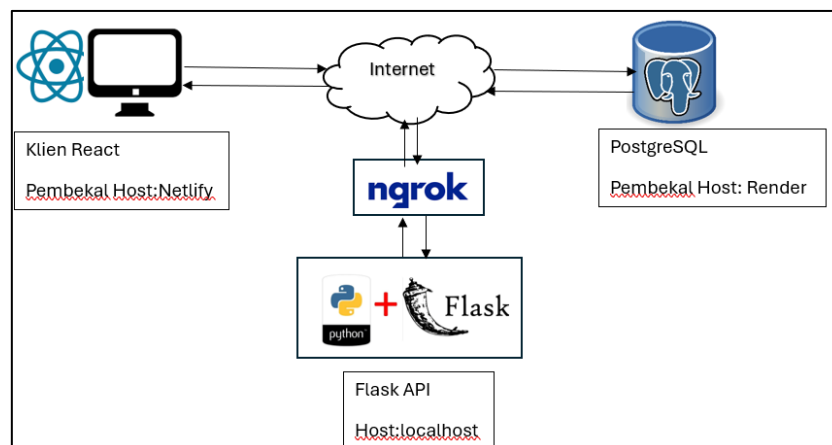
Dalam projek ini, terdapat dua jenis metrik akan digunakan iaitu metrik prestasi dan metrik kelajuan. Metrik prestasi digunakan untuk menilai kekuatan model dalam memastikan ketepatan ramalan. Metrik prestasi yang digunakan untuk menilai model ialah matriks kekeliruan, ketepatan (*precision*), ingatan (*recall*), PR-AUC, skor-F1, mAP50 dan mAP50-95. Tiga element penting dalam matriks kekeliruan iaitu positif benar (TP), positif palsu (FP) dan negatif palsu (FN) akan digunakan untuk mengira metrik prestasi. Negatif benar (TN) tidak

digunakan dalam penilaian prestasi model pengesanan objek kerana TN akan sentiasa infiniti. Hal ini kerana TN bermaksud model membuat ramalan betul bahawa tiada objek dikesan dalam kawasan tiada kotak ground truth. Oleh itu, metrik ketepatan (*accuracy*) tidak digunakan kerana formulanya mengandungi TN. Nilai *Intersection Over Union* (IoU) juga akan dijadikan nilai ambang untuk menentukan ramalan oleh model tersebut adalah TP, FP atau FN. Metrik kelajuan pula menilai kelajuan model dalam melakukan inferens pengesanan objek, terutamanya dalam aplikasi masa nyata. Metrik kelajuan yang digunakan ialah masa inferens dan FPS. Kedua-dua metrik ini adalah sebagai indikator penting dalam proses pemilihan model optimum.

Untuk memilih model optimum, model tersebut harus mencapai keseimbangan (*trade-off*) antara FPS dengan mAP. Semasa proses pemilihan model, model yang tidak mencapai kriteria akan dikenal pasti dan tidak akan menjadi calon pemilihan model. Seterusnya, pemilihan model akan dilakukan dengan menilai aspek seperti mAP50, mAP50-95 dan FPS. Pemilihan akan mementingkan mAP50-95 kerana ia memberikan prestasi model secara menyeluruh terhadap tahap kesukaran pengesanan yang berbeza. Dua graf iaitu graf mAP50 melawan FPS dan graf mAP50-95 melawan FPS akan diplotkan untuk mendapatkan keseimbangan antara prestasi model dengan kelajuan inferens model. Selain itu, kadar peningkatan mAP terhadap kadar penurunan FPS juga akan dinilai. Apabila model optimum telah ditentukan, model tersebut akan diuji dengan video trafik. Setelah ketepatan pengiraan diperoleh daripada semua video, proses pengujian akan ditamatkan.

Penggunaan model

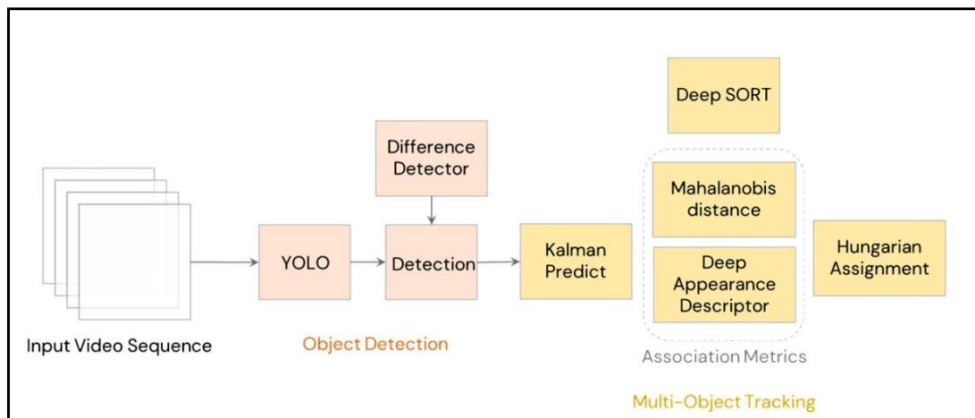
Proses penggunaan model (*model deployment*) membolehkan pengguna melaksanakan permintaan inferens pada model pembelajaran mesin melalui rangkaian Internet. Proses ini mempunyai dua fasa, iaitu fasa pembangunan dan fasa pengeluaran. Pada fasa pembangunan, segala kod akan dijalankan melalui sekitar lokal, iaitu dalam *localhost*. Pada fasa pengeluaran pula, pengguna dapat mengakses sistem ini melalui peranti masing-masing.



Rajah 2. Seni bina sistem pada fasa pengeluaran

Bahagian depan iaitu bahagian klien seperti antara muka yang boleh diakses terus oleh pengguna. Rangka kerja Javascript iaitu React digunakan untuk membina bahagian depan sistem ini. Bahasa pengaturcaraan TypeScript telah digunakan untuk membina bahagian depan React. Bootstrap 5 digunakan sebagai rangka kerja CSS. Pembekal servis awan Netlify telah

digunakan sebagai host bagi aplikasi React. Bahagian belakang pula mengandungi bahagian pelayan dan pangkalan data. Saya telah menggunakan PostgreSQL yang disediakan oleh pembekal servis awan Render sebagai pangkalan data. Flask telah digunakan membangunkan RESTful API yang menerima permintaan HTTP dari klien dan memulangkan respons dalam pelbagai format kepada klien. Ngrok akan mencipta terowong selamat ke mesin localhost untuk menyambung API Flask kepada Internet. Dalam RESTful API tersebut, terdapat algoritma DeepSORT yang penting untuk pengiraan kenderaan.



Rajah 3. Seni bina algoritma DeepSORT

DeepSORT ialah algoritma yang digunakan untuk penjejakan pelbagai objek yang menggunakan penapis Kalman. Setelah memperoleh keputusan pengesanan oleh algoritma YOLO seperti kelas, kotak sempadan dan skor keyakinan, penapis Kalman akan meramalkan keadaan objek seperti kedudukan, halaju dan pecutannya. Penapis Kalman meramalkan keadaan setiap objek dalam bingkai semasa berdasarkan keadaan terakhir yang diketahui dengan mengambil kira dinamik gerakan objek. Untuk menyelesaikan perkaitan antara keadaan Kalman yang telah diramalkan dan ukuran yang baru tiba, masalah peruntukan yang boleh diselesaikan oleh algoritma Hungary harus dibina (Wojke, N et al., 2017). Pada tahap ini, metrik gerakan dan ciri rupa objek sebagai metrik perkaitan mendalam telah digabungkan dengan matriks kos yang menggabungkan jarak kosinus vektor ciri rupa dan jarak Mahalanobis (metrik gerakan) dengan matriks gerbang digunakan untuk mengehadkan ambang matriks kos.

$$\text{Metrik Gerakan: } d^{(1)}(i,j) = (\mathbf{d}_j - \mathbf{y}_i)^T \mathbf{S}_i^{-1} (\mathbf{d}_j - \mathbf{y}_i) \text{ ----(1)}$$

$$\text{Metrik ciri rupa: } d^{(2)}(i,j) = \min \{ 1 - \mathbf{r}_j^T \mathbf{r}_k^{(i)} \mid \mathbf{r}_k^{(i)} \in R_i \} \text{ ----(2)}$$

$$\text{Gabungkan kedua-dua metrik: } c_{i,j} = \lambda d^{(1)}(i,j) + (1 - \lambda) d^{(2)}(i,j) \text{ ----(3)}$$

Penjejakan Objek

```
features = encoder(frame, bboxes)
# convert the detections to deep sort format
dets = []
for bbox, conf, class_name, feature in zip(bboxes, confidences, names, features):
    dets.append(Detection(bbox, conf, class_name, feature))

# run the tracker on the detections
tracker.predict()
tracker.update(dets)

# loop over the tracked objects
for track in tracker.tracks:
    if not track.is_confirmed() or track.time_since_update > 1:
        continue

    # get the bounding box of the object, the name
    # of the object, and the track id
    bbox = track.to_tlbr()
    track_id = track.track_id
    class_name = track.get_class()
    # convert the bounding box to integers
    x1, y1, x2, y2 = int(bbox[0]), int(bbox[1]), int(bbox[2]), int(bbox[3])
```

Rajah 4. Kod segmen berkaitan penjejakan objek melalui DeepSORT

Setelah mendapatkan maklumat penting iaitu kotak sempadan, skor keyakinan dan kelas daripada model YOLO, *encoder* akan mendapatkan vektor ciri untuk setiap objek dalam kotak sempadan. Senarai *dets* akan menyimpan kotak sempadan, skor keyakinan, kelas dan vektor ciri dalam format *Detection* untuk penjejak DeepSORT. Seterusnya, penjejak akan membuat ramalan terhadap kedudukan baru bagi objek yang dijejaki berdasarkan keputusan lama. Penjejak akan menggunakan maklumat dalam *dets* untuk kemas kini penjejakan baharu untuk objek yang baru dikesan dan menamatkan penjejakan untuk objek yang tidak lagi dikesan. Algoritma akan semak sama ada penjejakan ini adalah disahkan dan telah dikemas kini dalam bingkai terakhir. Syarat ini ditetapkan sebagai penapis untuk memproses penjejakan yang sedang dijejaki secara aktif dan dikemaskini secara terkini agar membantu dalam mengabaikan positif palsu atau kehilangan penjejakan. Maklumat objek yang dijejaki iaitu kotak sempadan dalam format (x1,y1, x2,y2), *track_id* sebagai ID unik dan kelas akan didapatkan semula.

Pengiraan Objek

```
for i in range(6):
    if center_y > start_line_A[1] and last_point_y < start_line_A[1] and i == class_id:
        counter_in[i] += 1
        points[track_id].clear()
    elif center_y < start_line_A[1] and last_point_y > start_line_A[1] and i == class_id:
        counter_out[i] += 1
        points[track_id].clear()
```

Rajah 5. Kod segmen berkaitan pengiraan bilangan objek

Bilangan objek kelas tersebut akan ditambah satu apabila objek itu melepasi garisan. Untuk mengukur sama ada objek itu melepasi garisan, satu titik untuk mengenal pasti arah dan pusat koordinat objek akan diguna dalam penghitungan.



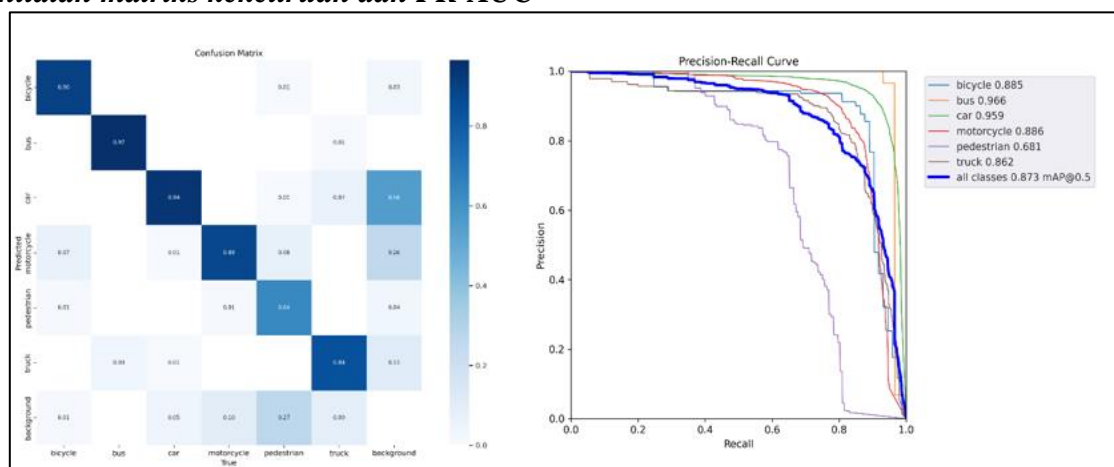
Rajah 6. Kaedah digunakan untuk pengiraan bilangan objek

Titik hijau ialah pusat koordinat bagi objek dan titik merah ialah titik yang mengenal pasti arah objek. Jika pusat koordinat y objek lebih daripada koordinat y garisan dan titik merah berada di atas garisan, bermakna objek tersebut menuju ke bawah, maka *counter_in[i]* akan bertambah satu berdasarkan kelas objek. Manakala pusat koordinat y objek kurang daripada koordinat y garisan dan titik merah berada di bawah garisan, bermaksud objek itu bergerak ke arah bertentangan iaitu ke atas. Maka, *counter_out[i]* akan bertambah satu mengikut kelas objek. Setelah objek itu dikira, objek itu akan dikeluarkan daripada senarai points. Bilangan setiap objek dari dua arah akan disimpan dalam dua senarai, iaitu *counter_in* dan *counter_out*. Akhirnya, dua senarai tersebut akan dipulangkan kepada bahagian klien.

Keputusan dan Perbincangan

Penilaian yang digunakan bagi kajian ini adalah seperti hasil ketepatan dan kehilangan model semasa fasa latihan, pengesahan dan ujian. Hasil dapatan latihan model boleh dijangkakan melalui sejauh mana keberkesanan model capai. Dalam kajian ini, ketepatan model adalah metrik penilaian kritikal. Keberkesanan model boleh dijangkakan melalui keputusan ketepatan dan kehilangan untuk set data latihan dan data pengesahan. Ketepatan merupakan metric yang mengukur prestasi model pada set data tertentu. Ketepatan latihan yang tinggi menunjukkan bahawa model CNN membuat ramalan yang betul dan tepat dengan menggunakan data latihan.

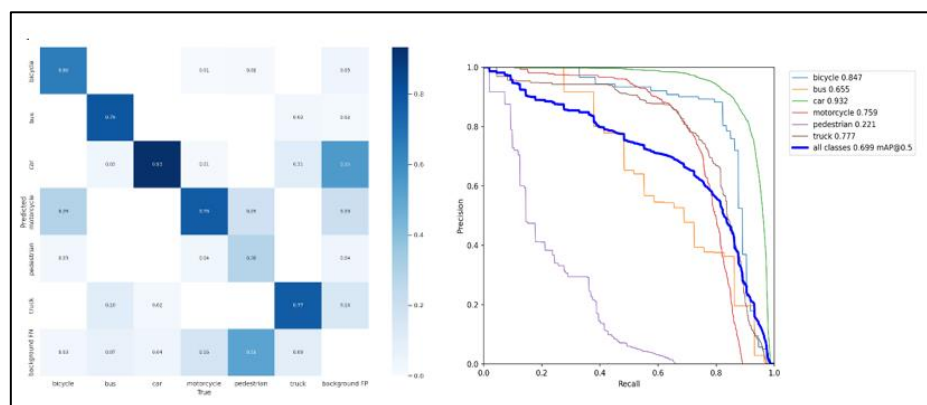
Penilaian matriks kekeliruan dan PR-AUC



Rajah 7. Matriks kekeliruan dan lengkung PR YOLOv5

Jika lihat pada pepenjuru matriks, kelas basikal, bas, kereta, motorsikal dan lori mencapai kebarangkalian lebih daripada 0.8 di mana kelas tersebut akan diramal dan ramalan tersebut adalah betul. Hanya ketepatan ramalan kelas pejalan kaki mencapai kebarangkalian yang kurang baik iaitu sebanyak 0.64. Pada aspek kekeliruan yang signifikan pula, kelas pejalan kaki mempunyai kebarangkalian 0.08 akan diramal sebagai motorsikal. Kelas basikal juga mencatat 0.07 akan diramal sebagai motorsikal. Kelas lori pula mencatat 0.07 akan diramal sebagai kereta manakala kelas kereta mencatat 0.01 akan diramal sebagai lori. Pada segi latar belakang FN pula, kelas pejalan kaki mencapai kebarangkalian paling tinggi iaitu 0.27 akan menjadi ramalan FN. Bagi latar belakang FP pula, kelas kereta mencapai 0.56, kelas motorsikal mencapai 0.26 dan kelas lori mencapai 0.13.

Pada lengkung PR bagi setiap kelas pula, kelas bas mencapai nilai AP paling tinggi iaitu sebanyak 0.966 dan kelas pejalan kaki mencapai nilai AP paling rendah iaitu sebanyak 0.681. Namun, lengkung bagi kelas kereta adalah licin berbanding dengan kelas lain kerana terdapat banyak anotasi dalam set data ujian dan model ini mencapai prestasi baik dalam kelas ini iaitu nilai AP sebanyak 0.959. Nilai mAP yang didapatkan daripada purata AP 6 kelas adalah sebanyak 0.873 pada nilai ambang IOU 0.5.

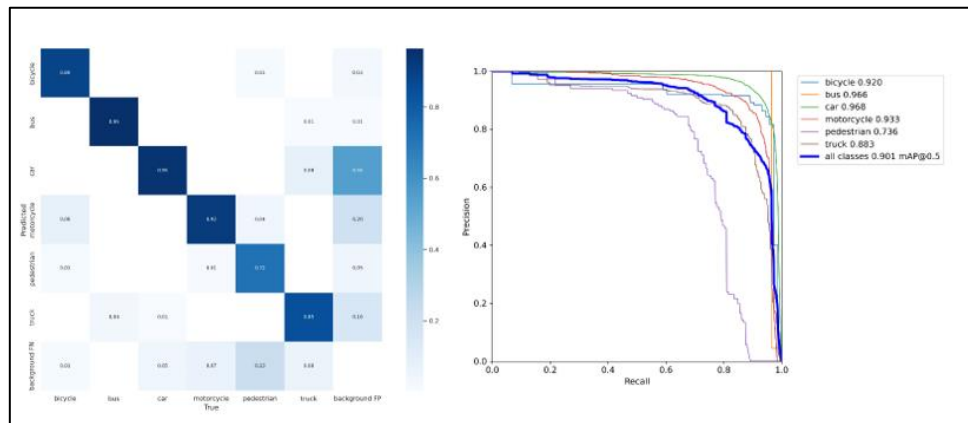


Rajah 8. Matriks kekeliruan dan lengkung PR YOLOv6

Seterusnya, bagi model YOLOv6 pula hanya kelas kereta mencapai kebarangkalian lebih daripada 0.8 di mana kelas tersebut akan diramal dan ramalan tersebut adalah betul. Kelas basikal, bas, mmotorsikal dan lori pula mencapai kebarangkalian antara 0.6 hingga 0.8, iaitu pada tahap sederhana. Namun, kelas pejalan kaki menunjukkan kebarangkalian sebanyak 0.28 menunjukkan bahawa model ini tidak dapat membuat ramalan betul terhadap kelas pejalan kaki. Pada aspek kekeliruan yang signifikan pula, kelas basikal mencatat 0.29 akan diramal sebagai motorsikal manakala kelas motorsikal hanya mencatat 0.01 akan diramal sebagai motorsikal. Kelas lori juga mencapai 0.11 akan diramal sebagai kereta manakala kelas kereta hanya mencatat 0.02 akan diramal sebagai lori. Kekeliruan juga berlaku pada kelas bas dan lori. Kelas bas mencatat 0.10 akan diramal sebagai lori manakala kelas lori mencatat 0.03 akan diramal sebagai bas. Pada segi latar belakang FN pula, kelas pejalan kaki mencatat kebarangkalian paling tinggi iaitu mempunyai 0.51 akan menjadi ramalan FN. Bagi latar belakang FP yang signifikan pula, kelas kereta mencapai 0.55, kelas motorsikal mencapai 0.2 dan kelas lori mencapai 0.14.

Pada lengkung PR bagi setiap kelas pula, kelas kereta mencapai nilai AP paling tinggi iaitu sebanyak 0.932 dan kelas pejalan kaki mencapai nilai AP paling rendah iaitu sebanyak 0.221.

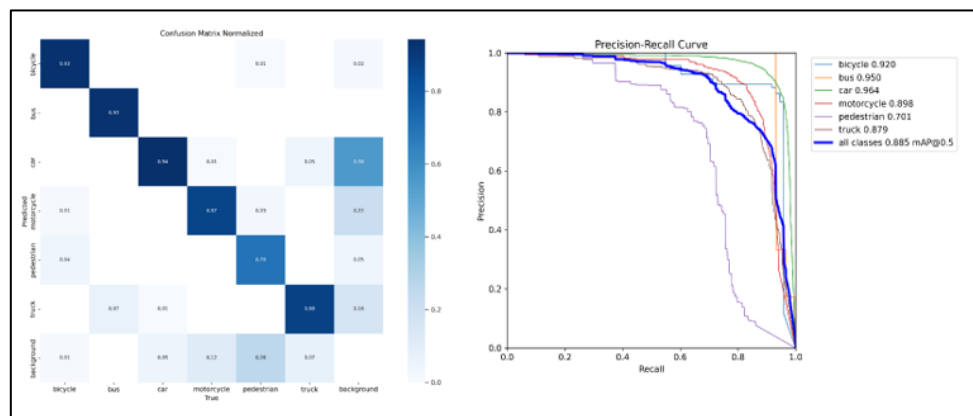
Nilai AP bagi kelas pejalan kaki sangat rendah telah menunjukkan bahawa model tidak dapat membuat ramalan ketepatan tinggi terhadap kelas tersebut dalam menyebabkan ingatan menjadi rendah. Nilai mAP yang didapatkan daripada purata AP 6 kelas adalah sebanyak 0.699 pada nilai ambang IOU 0.5.



Rajah 9. Matrik kekeliruan dan lengkung PR Model YOLOv7

Bagi model YOLOv7 pula, hanya kelas pejalan kaki mencapai kebarangkalian kurang daripada 0.8 pada pepenjur matriks. Pada aspek kekeliruan yang signifikan pula, kelas basikal mencatat 0.08 akan diramal sebagai motorsikal. Kelas lori juga mencapai 0.08 akan diramal sebagai kereta manakala kelas kereta hanya mencatat 0.01 akan diramal sebagai lori. Kekeliruan juga berlaku pada kelas bas dan lori. Kelas bas mencatat 0.04 akan diramal sebagai lori manakala kelas lori mencatat 0.01 akan diramal sebagai bas. Pada segi latar belakang FN pula, kelas pejalan kaki mencatat kebarangkalian paling tinggi iaitu mempunyai 0.23 akan menjadi ramalan FN. Bagi latar belakang FP yang signifikan pula, kelas kereta mencapai 0.55, kelas motorsikal mencapai 0.2 dan kelas lori mencapai 0.16.

Pada lengkung PR bagi setiap kelas pula, kelas kereta mencapai nilai AP paling tinggi iaitu sebanyak 0.968 dan kelas pejalan kaki mencapai nilai AP paling rendah iaitu sebanyak 0.736. Kelas kereta, basikal, bas, motorsikal dan lori mencapai nilai AP yang lebih daripada 0.8, menunjukkan kebolehan model ini dalam mencapai nilai P dan R yang tinggi. Nilai mAP yang didapatkan daripada purata AP 6 kelas adalah sebanyak 0.901 pada nilai ambang IOU 0.5.



Rajah 10. Matrik kekeliruan dan lengkung PR Model YOLOv8

Bagi model YOLOv8 pula, hanya kelas pejalan kaki mencapai kebarangkalian kurang daripada 0.8 pada pepenjur matriks. Pada aspek kekeliruan yang signifikan pula, kelas bas mencatat 0.07 akan diramal sebagai lori. Kekeliruan juga berlaku pada kelas kereta dan lori. Kelas lori mencatat 0.05 akan diramal sebagai kereta manakala kelas kereta mencapai 0.01 akan diramal sebagai lori. Pada segi latar belakang FN pula, kelas pejalan kaki mencatat kebarangkalian paling tinggi iaitu mempunyai 0.26 akan menjadi ramalan FN. Bagi latar belakang FP yang signifikan pula, kelas kereta mencapai 0.56, kelas motorsikal mencapai 0.22 dan kelas lori mencapai 0.16.

Pada lengkung PR bagi setiap kelas pula, kelas kereta mencapai nilai AP paling tinggi iaitu sebanyak 0.964 dan kelas pejalan kaki mencapai nilai AP paling rendah iaitu sebanyak 0.701. Kelas kereta, basikal, bas, motorsikal dan lori mencapai nilai AP yang lebih daripada 0.8, menunjukkan kebolehan model ini dalam mencapai nilai P dan R yang tinggi. Nilai mAP yang didapatkan daripada purata AP 6 kelas adalah sebanyak 0.885 pada nilai ambang IOU 0.5.

Pelbagai informasi boleh didapatkan melalui analisis terhadap matriks kekeliruan dan lengkung PR bagi 4 model. Secara keseluruhannya, empat matriks kekeliruan menunjukkan trend yang sama. Pertama sekali, kekeliruan antara pejalan kaki, motorsikal dan basikal berlaku kerana manusia adalah salah satu ciri yang boleh diekstrak dalam kelas tersebut. Wujudnya juga kekeliruan antara kelas basikal dan motorsikal kerana kedua-duanya mempunyai ciri yang sama iaitu kenderaan dua roda dan saiz yang berdekatan. Model juga akan menghadapi kekeliruan antara kereta dengan lori terutamanya lori tersebut adalah berjenis truck kecil atau van. Kelas lori dan bas juga menyebabkan kekeliruan kerana kedua-duanya ialah kenderaan berat yang bersaiz besar. Ramalan yang melibatkan latar belakang pula, latar belakang mempunyai kebarangkalian diramal sebagai kelas kereta, motor dan lori kerana terdapat imej latihan yang tidak dilabel kerana terlalu jauh daripada kawasan pusat imej tersebut. Jadi, FP akan berlaku pada set data pengesahan dan ujian kerana memang wujudnya objek tersebut dan tidak dilabel. Kelas pejalan kaki pula mempunyai kebarangkalian diramal sebagai latar belakang dan menyebabkan FN meningkat.

Bagi Lengkung PR pula, kelas kereta menunjukkan lengkung yang licin pada empat model. Hal ini kerana kelas kereta mempunyai bilangan anotasi yang paling tinggi. Kelas kereta juga mencapai nilai AP paling tinggi dalam model YOLOv6, YOLOv7 dan YOLOv8. Kelas pejalan kaki pula mencapai AP paling rendah pada empat model. Berdasarkan keputusan diperolehi daripada matriks kekeliruan dan lengkung PR, kelas pejalan kaki mendapatkan prestasi yang rendah berbanding dengan kelas lain.

Penilaian metrik

Jadual 2

Keputusan metrik prestasi untuk 4 model YOLO

Model	Ketepatan	Ingatan	Skor-F1	mAP50	mAP50-95
YOLOv8	0.873	0.844	0.858	0.885	0.658
YOLOv7	0.85	0.875	0.862	0.901	0.656
YOLOv6	0.763	0.632	0.691	0.699	0.435
YOLOv5	0.855	0.83	0.842	0.873	0.622

Berdasarkan jadual 2, YOLOv6 mencapai nilai paling rendah bagi semua metrik iaitu ketepatan, ingatan, skor-F1, mAP50 dan mAP50-95. Model YOLOv8 pula mencapai ketepatan dan mAP50-95 yang paling tinggi. Model YOLOv7 mencapai ingatan, skor-F1 dan mAP50 yang paling tinggi. YOLOv7 mencapai mAP50 lebih tinggi daripada YOLOv8 tetapi mAP50-95 model YOLOv7 lebih rendah daripada YOLOv8. Jelasnya, YOLOv7 mencapai prestasi paling tinggi bagi kebanyakan metrik prestasi seterusnya diikuti dengan YOLOv8, YOLOv5 dan YOLOv6.

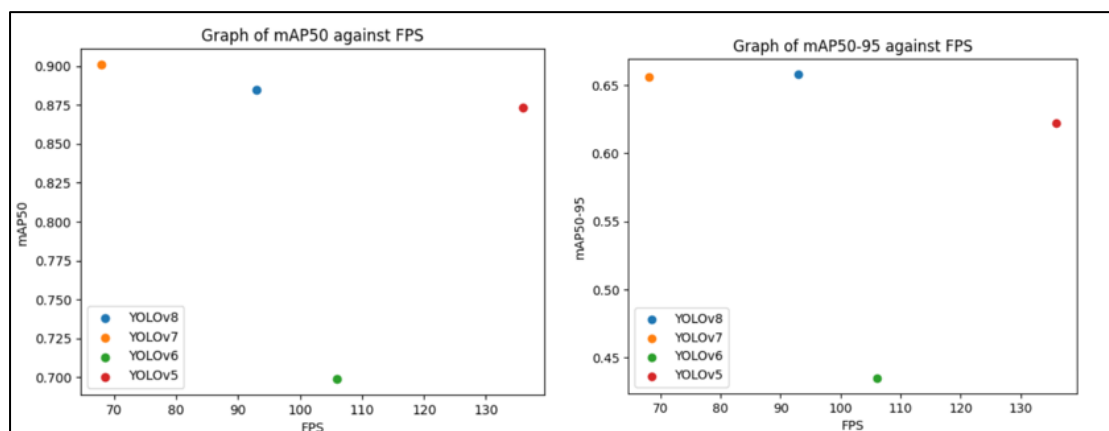
Jadual 3

Keputusan metrik kelajuan dan faktor kelajuan untuk 4 model YOLO

Model	Masa Inferens	FPS	Parameter	GFLOPs
YOLOv8	10.7ms	93	11.1M	28.4
YOLOv7	14.6ms	68	36.5M	103.2
YOLOv6	9.38ms	106	18.5M	45.18
YOLOv5	7.3ms	136	7.02M	15.8

Berdasarkan jadual 3, YOLOv5 mencapai masa inferens yang paling cepat, iaitu sebanyak 7.3ms. YOLOv7 mencapai masa inferens paling lambat, iaitu sebanyak 14.6ms dan angka ini adalah dua kali ganda berbanding dengan YOLOv5. Semakin rendah masa inferens, semakin tinggi nilai FPS. Oleh itu, YOLOv5 mencapai FPS paling tinggi kerana parameter dan GFLOPs bagi model ini adalah paling rendah. Manakala YOLOv7 mencapai FPS paling rendah kerana parameter dan GFLOPs adalah paling tinggi. Bagi YOLOv6 pula, model ini mencapai masa inferens lebih cepat daripada YOLOv8 tetapi parameter dan GFLOPs bagi YOLOv6 adalah lebih banyak berbanding dengan YOLOv8. Keputusan ini menunjukkan kekuatan YOLOv6 dalam metrik kelajuan tetapi model ini tidak menunjukkan kelebihan dalam metrik prestasi dengan set data projek. Jelasnya, YOLOv5 mencapai prestasi paling baik dalam FPS dan masa inferens seterusnya diikuti dengan YOLOv6, YOLOv8 dan YOLOv7.

Keputusan Pemilihan Model

**Rajah 11.** Graf mAP50 melawan FPS dan mAP50-95 melawan FPS

Pada graf pertama, YOLOv7 mencapai mAP50 paling tinggi dan diikuti oleh YOLOv8, YOLOv5 dan YOLOv6. Pada graf kedua pula, YOLOv8 mencapai mAP50-95 yang paling tinggi dan diikuti oleh YOLOv7, YOLOv5 dan YOLOv6. Dalam kedua-dua graf tersebut telah jelas menunjukkan bahawa model YOLOv6 mencapai nilai mAP50 dan mAP50-95 yang sangat rendah jika berbanding dengan model lain. Oleh itu, model YOLOv6 tidak akan dijadikan model optimum. Seterusnya, kadar perubahan mAP50, mAP50-95 dan FPS antara YOLOv5, YOLOv6 dan YOLOv7 akan dikira.

Jadual 4

Kadar perubahan mAP50, mAP50-95 dan FPS antara tiga model

Model	Kadar Perubahan mAP50	Kadar Perubahan mAP50-95	Kadar Perubahan FPS
YOLOv7 dan YOLOv8	-1.78%	0.30%	36.76%
YOLOv8 dan YOLOv5	-1.36%	-5.47%	46.24%
YOLOv7 dan YOLOv5	-3.11%	-5.18%	100%

Dari segi perubahan mAP50 dan FPS, mAP50 bagi YOLOv8 menurun sebanyak 1.78% dari YOLOv7. Namun, mAP50-95 bagi YOLOv8 meningkat sebanyak 0.3% dari YOLOv7. Keputusan ini telah menunjukkan kekuatan YOLOv8 dalam tugas pengesanan objek dalam pelbagai kesukaran berbanding dengan model lain. Selain itu, keputusan ini menunjukkan bahawa YOLOv5 sesuai digunakan dalam projek yang memerlukan model ringan dan masa nyata seperti pesawat udara tanpa pemandu (UAV). Hal ini kerana penurunan mAP50 bagi YOLOv5 hanya sebanyak 3.11% dari YOLOv7 manakala FPS bagi YOLOv5 telah meningkat sebanyak dua kali ganda, iaitu 100%. Namun, untuk projek ini YOLOv8 yang berada di antara YOLOv7 dan YOLOv5 telah menunjukkan keseimbangan antara mAP dengan FPS. Oleh itu, faktor utama model YOLOv8 akan dipilih kerana mAP50-95 meningkat apabila FPS meningkat. Jelasnya, model YOLOv8 menunjukkan nilai paling tinggi dalam mAP50-95 dan membuktikan kekuatan model dalam pengesanan objek secara keseluruhan dengan nilai ambang IoU dari 0.5 hingga 0.95.

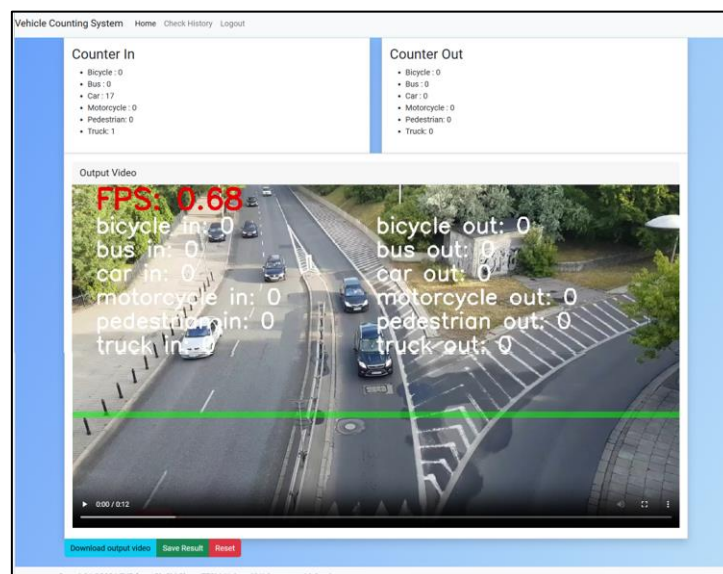
Keputusan Pengujian Terhadap Video Trafik

Jadual 5

Keputusan pengujian terhadap lima video trafik dengan YOLOv8

Video Ujian	Bilangan Masuk	Bilangan Keluar	Ketepatan Pengiraan
Video A	[0,0,17,0,0,1]	[0,0,0,0,0,0]	100%
Video B	[0,0,4,0,0,0]	[0,0,8,1,0,0]	100%
Video C	[0,0,11,1,0,0]	[0,0,18,3,0,0]	86.11%
Video D	[0,0,11,3,0,0]	[0,0,19,2,0,0]	86.11%
Video E	[0,0,2,0,0,0]	[0,0,1,1,0,0]	100%

Berdasarkan keputusan di atas, ramalan terhadap Video A, Video B dan Video E tidak berlaku sebarang kesalahan. Kenderaan pada Video A hanya bergerak ke bawah iaitu masuk dan kenderaan pada Video B bergerak pada dua hala. Walaupun ketepatan pengiraan adalah sama bagi ramalan terhadap Video C dan Video D, keputusan ramalannya adalah berbeza. Bagi video C, pengiraan terlepas berlaku pada kelas basikal dan kelas motorsikal dalam arah masuk. Namun, pengiraan terlepas berlaku pada kelas basikal sahaja bagi Video D. Selain itu, terdapat satu kesalahan klasifikasi terhadap Video C dan dua kesalahan klasifikasi terhadap Video D pada arah keluar. Video E pula mencapai 100% tetapi bilangan sampel adalah sedikit.

**Rajah 12.** Antara muka sistem selepas pengujian pada video A

Kesimpulan

Projek ini membangunkan sistem pemantauan trafik dengan menggunakan model YOLO sebagai penyelesaian terhadap masalah kesesakan lalu lintas. Sistem ini mengumpul bilangan kenderaan dalam trafik dan maklumat ini akan digunakan oleh perancang trafik untuk merancang strategi pengurusan dan penyuraian trafik. Sistem ini menggunakan model YOLO untuk mengesan lokasi dan jenis kenderaan dalam trafik secara automatik. Dalam projek ini, empat model YOLO iaitu YOLOv5, YOLOv6, YOLOv7 dan YOLOv8 telah dilatih dengan set

data latihan. Berdasarkan keputusan metrik pada proses pengujian model, YOLOv8 dikenali sebagai model optimum kerana telah menunjukkan keseimbangan antara mAP dengan FPS dan juga mAP50-95 yang tertinggi antara 4 model. Flask (Python) digunakan untuk membangunkan bahagian belakang iaitu Restful API yang boleh memulangkan keputusan ramalan daripada model kepada bahagian depan iaitu React. Penjejakan kenderaan juga diimplementasi dalam Restful API melalui algoritma DeepSORT untuk mengekalkan pengesanan kenderaan pada setiap bingkai untuk mendapatkan pengiraan setiap kenderaan. Pangkalan data PostgreSQL digunakan untuk menyimpan data keputusan dan data pengguna. Seterusnya, pelbagai servis awan digunakan untuk hosting bahagian belakang, bahagian depan dan pangkalan data agar sistem boleh diakses oleh pengguna melalui Internet.

Projek ini meningkatkan mAP dan pengiraan kenderaan untuk jalan raya dua hala. Pemprosesan imej dan penambahan data yang berkaitan dengan pengubahan maklumat visual imej telah digunakan mengurangkan kesan overfitting. Teknik yang akan mengubah orientasi imej seperti putaran tidak digunakan kerana akan menurunkan mAP. Selain itu, sistem ini dapat mencapai ketepatan pengiraan yang baik terhadap video ujian yang terdiri daripada pelbagai situasi.

Sistem ini mempunyai kekangan dalam aspek kebolehskalaan. Sistem ini tidak dapat menanggung input yang bersaiz besar atau video yang mempunyai tempoh masa yang panjang. Hal ini kerana kerumitan komputasional oleh algoritma dalam sistem ini. Selain itu, masa diambil bagi proses inferens oleh model adalah panjang kerana model melaksanakan ramalan terhadap setiap bingkai dalam video. Pengguna mungkin tidak dapat menyaksikan video output dengan lancar jika lebar jalur rangkaian Internet agak rendah. Hal ini kerana kandungan video keluaran dipulangkan daripada persekitaran lokal melalui HTTP.

Penghargaan

Penulis kajian ini ingin ucapkan setinggi-tinggi penghargaan dan jutaan terima kasih kepada Assoc. Prof. Dr. Mohammad Faizul Nasrudin selaku penyelia penulis kajian ini yang telah memberi tunjuk ajar serta bimbingan untuk menyiapkan projek ini dengan jayanya. Pengertian, dorongan dan sokongan yang berterusan sepanjang tempoh memenuhi tugas ini amat dihargai.

Penulis kajian ini juga ingin mengucapkan ribuan terima kasih kepada fakulti iaitu Fakulti Teknologi dan Sains Maklumat (FTSM) kerana telah memberi peluang untuk mengaplikasikan segala ilmu yang telah diterima sepanjang berada di FTSM. Selain itu, FTSM juga mengadakan beberapa modul atau bengkel untuk membantu dan membimbing pelajar dalam menyempurnakan projek.

Akhir sekali, penulis kajian ini ingin mengucapkan terima kasih kepada semua pihak yang membantu secara langsung mahupun tidak langsung dalam menyempurnakan projek ini. Segala bantuan yang telah dihulurkan amatlah dihargai kerana tanpa bantuan mereka, projek ini tidak dapat dilaksanakan dengan baik. Semoga tuhan merahmati dan memberikan balasan yang terbaik.

References

- Abdullah, N. , Riza Atiq O.K.R., & Amiruddin, I. (2007). Effect of Transportation on Modal Shift from Private Car to Public Transport in Malaysia. *Journal of Applied Sciences*, 7(7), 1013–1018. <https://doi.org/10.3923/jas.2007.1013.1018>
- De Silva, D., & Alahakoon, D. (2022). An artificial intelligence life cycle: From conception to production. *Patterns*, 3(6), 100489. <https://doi.org/10.1016/j.patter.2022.100489>.
- Khalifa, N. E., Loey, M., & Mirjalili, S. (2021). A comprehensive survey of recent trends in deep learning for digital images augmentation. *Artificial Intelligence Review*, 55(3), 2351–2377. <https://doi.org/10.1007/s10462-021-10066-4>
- Jabatan Perangkaan Malaysia. (2022). Penemuan Utama Banci Penduduk dan Perumahan Malaysia 2020: Bandar dan Luar Bandar. https://v1.dosm.gov.my/v1/index.php?r=column/cthem&menu_id=L0pheU43NWJwRWVSZklWdzQ4TlhUUT09&bul_id=ZFRzTG9ubTkveFR4YUY2OXdNNk1GZz09 [23 Disember 2022].
- Morikawa, T., Yamamoto, T., Dissanayake, D., Sanko, N. & Kurauchi, S. et al. (2001). Travel Behavior Analysis and its Implication to Urban Transport Planning for Asian Cities: Case Studies of Bangkok, Kuala Lumpur, Manila, and Nagoya ICRA Project Report. The 5th International Conference of Eastern Asia Society for Transportation Studies, Nagoya University, Japan.
- Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1). <https://doi.org/10.1186/s40537-019-0197-0>
- Wojke, N., Bewley, A., & Paulus, D. (2017). Simple Online and Realtime Tracking with a Deep Association Metric. *arXiv (Cornell University)*. <https://doi.org/10.48550/arxiv.1703.07402>
- Xie, H., Xiao, Z., Liu, W. & Ye, Z. (2023). PVNet: A Used Vehicle Pedestrian Detection Tracking and Counting Method. *Sustainability*, 15(19): 14326. <https://doi.org/10.3390/su151914326>.