

Sistem Pintu Gerbang Otomatis Menggunakan Deep Learning Object Detection

Automatic Gate System Using Deep Learning Object Detection

1st Irham Khalifah Putra
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

irhamkhalifah@student.telkomuniversity.ac.id

2nd Favian Dewanta
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

favian@telkomuniversity.ac.id

3rd Sri Astuti
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

sriastuti@telkomuniversity.ac.id

Abstrak—*Internet of Things (IoT) merupakan salah satu bentuk perkembangan teknologi yang dapat diterapkan di hampir berbagai bidang pekerjaan atau industri. Dengan adanya IoT, aktivitas manusia dapat dimudahkan terutama dalam hal pengawasan dan pengendalian, selama ada koneksi antara (minimal) dua perangkat. IoT dapat dipadukan dengan berbagai perangkat dan teknologi, salah satunya adalah kamera untuk pendeteksi objek (computer vision). Dalam proyek ini, konsep IoT digunakan untuk kontrol gerbang dengan menggunakan kamera untuk mendeteksi kode QR dan plat nomor kendaraan terdaftar. Proyek ini menggunakan beberapa alat; di antaranya laptop, motor servo, sensor ultrasonik, Raspberry Pi, dan webcam. Dari hasil training dataset, diperoleh confusion matrix untuk mengukur kualitas dataset. Dari confusion matrix, akurasi 90%, presisi 86%, recall 100%, dan F1- score 92%. Quality of Service (QoS) juga diperoleh dari pengiriman data dari Raspberry Pi ke database Firebase; throughput 29,66 kbps, packet loss rata-rata 0%, delay rata-rata 24,244 ms, dan jitter rata-rata 32,3351 ms.*

Kata kunci—*QoS, deteksi objek, dataset, Raspberry Pi, Internet of Things, database*

Abstract—*Internet of Things (IoT) is a form of technology development that can be implemented in almost various fields of work or industry. With the existence of IoT, human activities can be facilitated, especially in terms of supervision and control -as long as there is a connection between (at least) two devices. IoT can be combined with various devices and technologies, one of which is a camera for object detection (computer vision). In this project, the IoT concept is used for gate control by using a camera to detect QR codes and registered vehicle number plates. This project utilizes several tools; including laptops, servo motors, ultrasonic sensors, Raspberry Pi, and webcams. From the results of the training dataset, a confusion matrix was obtained to measure the quality of the dataset. From the confusion matrix, the accuracy is 90%, precision is 86%, recall is*

100%, and F1- score is 92%. Quality of Service (QoS) is also obtained from sending data from the Raspberry Pi to the Firebase database; the throughput is 29,66 kbps, the average packet loss is 0%, the average delay is 24,244ms, and the average jitter is 32,3351 ms.

Keywords—*QoS, object detection, dataset, Raspberry Pi, Internet of Things, database*

I. PENDAHULUAN

Internet of Things (IoT) adalah suatu metode/teknologi yang dirancang dan dikembangkan untuk memfasilitasi berbagai aktivitas manusia yang saling terhubung dengan konektivitas jaringan. Saat ini, Internet of Things telah banyak digunakan di berbagai sektor, yang banyak digunakan untuk pemantauan dan pengendalian. IoT dapat bekerja setidaknya ada dua perangkat yang terintegrasi satu sama lain; satu perangkat untuk mengontrol atau memantau dan perangkat lainnya adalah perangkat yang akan dikendalikan atau dipantau. IoT dapat dipadukan dengan berbagai teknologi lainnya, salah satunya adalah deep learning, khususnya untuk pendeteksian objek (computer vision).

Diperkirakan pada tahun 2021, perkembangan dan penggunaan IoT di Indonesia akan meningkat. Seperti yang kita ketahui, berbagai perangkat IoT telah dikembangkan untuk berbagai bidang, mulai dari pendidikan, kesehatan, transportasi, militer, dan masih banyak lagi[1]. Selain meluasnya penggunaan IoT, perkembangan dan penggunaan teknologi kecerdasan buatan (AI) juga semakin meningkat. Hal itu terlihat dari pengembangan software Google Assistant yang dapat mengenali perintah suara dengan baik, sistem keamanan biometrik pengenalan wajah pada smartphone, deteksi penggunaan masker di fasilitas umum, hingga otentikasi pengenalan wajah untuk keamanan aplikasi layanan keuangan. Implementasi AI di Indonesia banyak menggunakan teknologi ALPR (*Automatic License Plate Recognition*) untuk penerapan sistem ETLE (*Electronic Traffic Law Enforcement*) atau electronic ticketing. Untuk saat ini, sistem ALPR atau ANPR (*Automatic Number Plate Recognition*) mengadopsi

metode *Optical Character Recognition* (OCR)[2]. OCR bekerja dengan memproses gambar: membuat gambar fisik menjadi gambar, menghilangkan noise, membuat dan mengevaluasi gambar sebagai hitam putih, dan membuat gambar menjadi segmen-segmen terpisah; kemudian lanjutkan ke tahap pengenalan karakter AI: pengenalan fitur dan pola; dan yang terakhir adalah proses koreksi post processing, dengan mengajarkan dan membatasi AI tentang karakter yang akan dikenali nantinya[3][4]. Sejauh ini, tidak ada yang salah dengan teknologi OCR, sejauh digunakan dalam menerjemahkan teks menggunakan kamera atau untuk membaca dokumen. Namun, untuk masalah keamanan pada kendaraan (identitas), akan menjadi masalah jika pelat nomor dipalsukan. Oleh karena itu, inovasi baru dibuat dalam riset ini untuk meminimalisir pelanggaran pemalsuan/peniruan plat nomor kendaraan.

Proyek penelitian ini, merancang sebuah sistem gerbang otomatis menggunakan teknologi deep learning object detection dan QR code. Pembuatan tugas akhir ini memanfaatkan salah satu algoritma pendeteksi objek deep learning yaitu YOLOv4. Algoritma ini digunakan untuk mendeteksi apakah kendaraan memiliki kode QR pada plat nomornya. Dalam proyek ini, kode QR berisi data kendaraan seperti plat nomor, warna, merek, dan jenis/varian kendaraan. Basis data Firebase juga digunakan sebagai media penyimpanan data hasil computer vision: plat nomor dengan kode QR atau tidak, data kode QR dibaca oleh sistem, dan waktu saat objek terdeteksi.

II. KAJIAN TEORI

A. Internet of Things (IoT)

Internet of Things adalah teknologi yang memungkinkan untuk menghubungkan mesin, peralatan, dan objek fisik lainnya dengan sensor dan aktuator jaringan untuk memperoleh data dan mengelola kinerjanya sendiri. Adapun kemampuan seperti berbagi data, remote control, dan sebagainya, termasuk objek di dunia nyata. Misalnya makanan, elektronik, barang koleksi, peralatan apa saja, termasuk makhluk hidup, yang semuanya terhubung ke jaringan lokal dan global melalui sensor yang disematkan dan selalu aktif. Sejauh ini, IoT paling erat kaitannya dengan komunikasi *machine-to-machine* (M2M) di bidang manufaktur dan elektronik, minyak, dan gas[5][6]. Di era digital ini, Internet of Things telah diterapkan di berbagai bidang, antara lain:

1. Energi
2. Infrastruktur
3. Transportasi
4. Keamanan
5. Pendidikan
6. Medis/Kesehatan
7. Teknologi dan jaringan

B. Deep Learning

Deep learning adalah bagian dari pembelajaran mesin (*machine learning*), yang pada dasarnya adalah jaringan saraf dengan tiga atau lebih lapisan. Jaringan saraf ini mencoba untuk mensimulasikan perilaku otak manusia—walaupun jauh dari kemampuan yang sesuai—memungkinkannya untuk “belajar” dari sejumlah besar data. Sementara jaringan saraf dengan satu lapisan masih

dapat membuat perkiraan perkiraan, lapisan tersembunyi tambahan dapat membantu mengoptimalkan dan menyempurnakan akurasi[7].

Deep learning mendorong banyak aplikasi dan layanan kecerdasan buatan (AI) yang meningkatkan otomatisasi, melakukan tugas analitis dan fisik tanpa campur tangan manusia. Teknologi pembelajaran mendalam terletak di balik produk dan layanan sehari-hari (seperti asisten digital, remote TV yang diaktifkan dengan suara, dan deteksi penipuan kartu kredit) serta teknologi baru (seperti mobil yang dapat menyetir sendiri).

C. Object Detection

Deteksi objek (*object detection*) biasanya kerap tertukar dengan pengenalan gambar (*image/object recognition*), penting bagi kita untuk mengklarifikasi perbedaan di antara mereka. Pengenalan gambar memberikan label pada gambar. Gambar seekor anjing menerima label “mobil”. Gambar dua anjing, masih diberi label “mobil”. Deteksi objek, di sisi lain, menggambar kotak di sekitar setiap anjing dan memberi label “mobil”. Model memprediksi di mana setiap objek dan label apa yang harus diterapkan. Dengan cara itu, deteksi objek memberikan lebih banyak informasi tentang gambar daripada pengenalan[8].

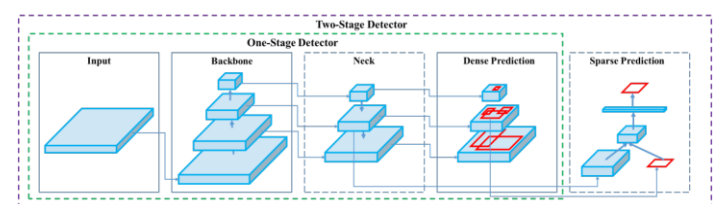
D. QR Code

Quick Response Code atau QR code adalah kode batang dua dimensi yang dikembangkan oleh Denso Wave Corporation pada tahun 1994. Ciri utama QR code dibandingkan dengan *barcode* tradisional adalah posisi vertikal dan horizontal dapat digunakan untuk menyimpan data sedangkan kode batang biasa hanya dapat digunakan untuk menyimpan data menyimpan informasi pada satu posisi saja[9]. Karena alasan inilah QR code dapat menyimpan lebih banyak informasi.

E. YOLOv4

YOLOv4 adalah model deteksi objek satu tahap yang ditingkatkan pada YOLOv3 dengan beberapa kumpulan trik dan modul yang diperkenalkan dalam literatur. Bagian komponen di bawah ini merinci trik dan modul yang digunakan. Algoritma YOLO dapat dijalankan menggunakan CPU atau dijalankan dengan akselerasi GPU. Untuk melakukan akselerasi GPU, mungkin akan diperlukan kartu grafis berbasis Nvidia yang bagus dengan inti CUDA[10].

Pada grafik dataset MS COCO [11] YOLOv4 memiliki *mean average precision* (mAP) sebesar 64,9% dan FPS lebih dari 25 dan kurang dari 50. Berdasarkan data grafik tersebut, YOLOv4 memiliki presisi yang paling baik di antara model lainnya.



Gambar 1.

Two-Stage Detector dari algoritma detektor[10]

Gambar 1 menunjukkan bagaimana *detector* bekerja dengan melalui *One-Stage Detector* dan *Two-Stage Detector*. Dimulai dari *One-Stage Detector*, di mana dengan input berupa gambar, *patches*, *image pyramid*, dan lainnya; *backbone* dapat menggunakan VGG16, ResNet-50, ResNeXt-101, Darknet53, dan lainnya; *neck* yang banyak digunakan adalah FPN, dapat digunakan pula PANet, Bi-FPN, dan lainnya; pada bagian atas (*head*), tahap *dense prediction* yang dapat menggunakan RPN, YOLO, SSD, RetinaNet, FCOS, dan lainnya. Masuk ke tahap *Two-Stage Detector*, masih pada bagian *head*, *Sparse Prediction* yang dapat terdiri dari algoritma Faster R-CNN, R-FCN, atau yang lainnya[10].

F. Raspberry Pi

Raspberry Pi adalah komputer seukuran kartu kredit berbiaya rendah yang dihubungkan ke monitor komputer atau TV, dan menggunakan *keyboard* dan *mouse* standar. Ini adalah perangkat kecil yang memungkinkan orang-orang dari segala usia untuk menjelajahi komputasi, dan belajar bagaimana memprogram dalam bahasa seperti Scratch dan Python. Ini mampu melakukan semua yang diharapkan dari sebuah komputer desktop, mulai dari menjelajah internet dan memutar video definisi tinggi, hingga membuat *spreadsheet*, pengolahan kata, dan bermain game[12][13].



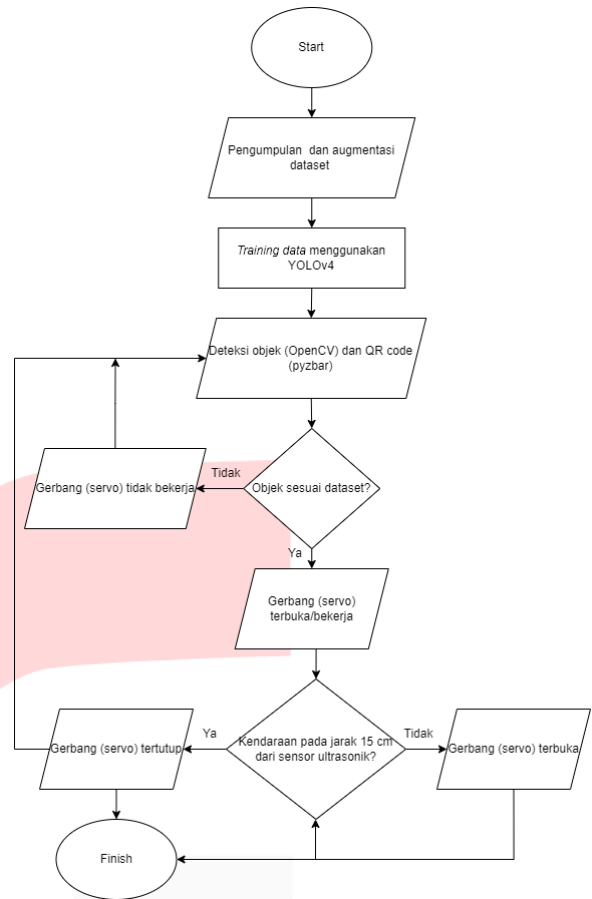
Gambar 2. Raspberry Pi 4

Gambar 2 merupakan gambar Raspberry Pi 4. Riset ini menggunakan Raspberry Pi 4 dengan spesifikasi RAM 4 GB, *processor* Broadcom BCM2711, Quad core 64-bit SoC 1.5GHz, serta GPU *memory* sebesar 800MB.

III. MODEL SISTEM DAN PERANCANGAN

A. Perancangan Sistem

Penelitian tugas akhir ini adalah pengembangan dan integrasi teknologi *Internet of Things*, *deep learning*, dan *computer vision*. Proyek ini membuat prototipe pintu gerbang rumah yang dapat buka dan tutup dengan menggunakan 2 buah motor servo. Kedua motor servo dikendalikan melalui Raspberry Pi secara otomatis. Untuk membuka gerbang digunakan webcam untuk mendeteksi ada tidaknya plat nomor (*deep learning* + *computer vision*) dan membaca kode QR pada plat nomor (*steganografi* + *computer vision*). Proyek ini juga menggunakan sensor ultrasonik HC-SR04 sebagai 'pemicu' untuk menutup pintu gerbang secara otomatis ketika suatu objek terdeteksi pada jarak 15 cm dan akan terbuka kembali ketika objek di depan sensor ultrasonik telah menjauh (lebih dari 15cm).

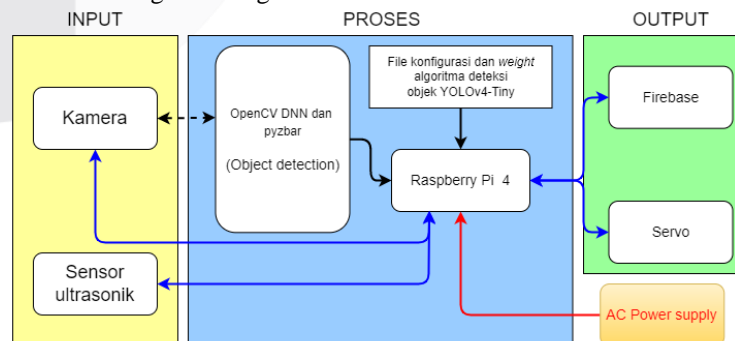


Gambar 3.

Flowchart langkah kerja sistem

Gambar 3 menunjukkan alur kerja dari sistem pada *prototype*. Ketika plat nomor terdeteksi dan kode QR dibaca oleh kamera, kedua informasi tersebut akan dikirim dari Raspberry Pi ke database Firebase. Kemudian dilakukan perhitungan kualitas jaringan untuk pengiriman informasi data dari Raspberry Pi ke Firebase real-time database menggunakan parameter Quality of Service (QoS) berupa throughput, packet loss, delay, dan jitter. Parameter yang digunakan didasarkan pada standar ITU - *Telecommunication Standardization Sector* (ITU-T)[12][14][15].

B. Perancangan Perangkat Keras

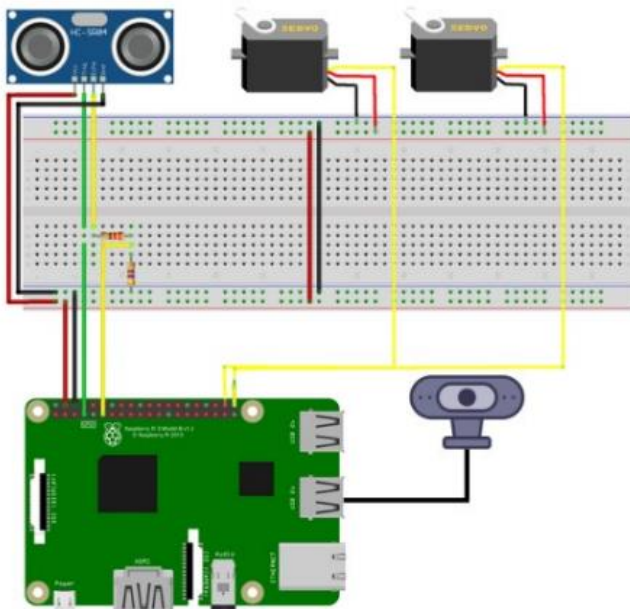


Gambar 4.

Blok sistem hardware dan integrasinya ke software

Gambar 4 menunjukkan hubungan kerja sistem yang terdiri dari *hardware* dan *software*. Blok sistem pada gambar 4 menunjukkan bagaimana hubungan kamera yang mengirim data ke Firebase melalui mikrokomputer Raspberry Pi.

Pada tugas akhir ini Raspberry Pi dihubungkan dengan beberapa komponen elektronika seperti sensor ultrasonik HC-SR04, 2 buah motor servo DC, dan webcam yang ditempatkan di bagian gerbang.



Gambar 5.
Desain perangkat keras

Gambar 5 merupakan hasil perancangan rangkaian elektronika menggunakan *software* Fritzing. Pada perancangan ini beberapa komponen elektronika dihubungkan ke GPIO dan port USB pada Raspberry Pi, berikut penjelasan koneksi antar perangkatnya:

1. Pin *trigger* sensor ultrasonik ke GPIO 4,
2. Pin *echo* sensor ultrasonic ke GPIO 17 (juga dihubungkan dengan resistor 330 Ω dan 470 Ω),
3. Pin VCC dan GND sensor ultrasonic masing-masing ke pin 5V dan GND Raspberry Pi,
4. Servo 1 dengan pin pulsanya ke GPIO 20,
5. Servo 2 dengan pin pulsanya ke GPIO 21,
6. Pin VCC dan GND servo masing-masing ke pin 5V dan GND,
7. Webcam USB dihubungkan ke *port* USB Raspberry Pi.

IV. HASIL PERCOBAAN DAN ANALISIS

A. Deep Learning Object Detection

Pada analisis pendeteksian objek menggunakan algoritma model *deep learning* YOLOv4, sistem berhasil mendeteksi plat nomor dengan cukup baik. Pada percobaan ini pelat nomor dideteksi dari jarak 50 cm, 60 cm, 70 cm, 80 cm, 90 cm, dan 100 cm, serta percobaan kedua dengan jarak uji setiap kelipatan 50 cm dimulai dari 150 cm hingga 500 cm. Setiap jarak dilakukan 5 kali percobaan. Hasilnya, sistem dapat mendeteksi plat nomor dari jarak 50 cm hingga jarak maksimum terbatas 100 cm.

Pada percobaan kedua, sistem dapat mendeteksi plat nomor dengan baik pada rentang jarak 150 cm hingga 400 cm. Model *deep learning* pada sistem bekerja tidak terlalu optimal (tidak bisa mendeteksi objek) mulai pada jarak 450 cm hingga 500 cm.



Gambar 6.

Hasil deteksi objek menggunakan webcam

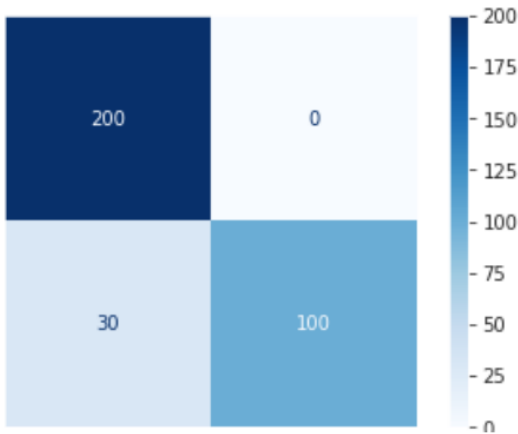
Dari hasil percobaan deteksi objek pada gambar 6, dapat dilihat akurasi deteksi pada *box terminal* (pada bagian *scores*) adalah sebesar 0,9196 atau sebesar 91%. Ini berarti bahwa objek yang dideteksi memiliki nilai kemiripan sebesar 91% dengan *dataset* yang ada pada program[10][16].



Gambar 7.

Grafik akurasi *training* dan validasi

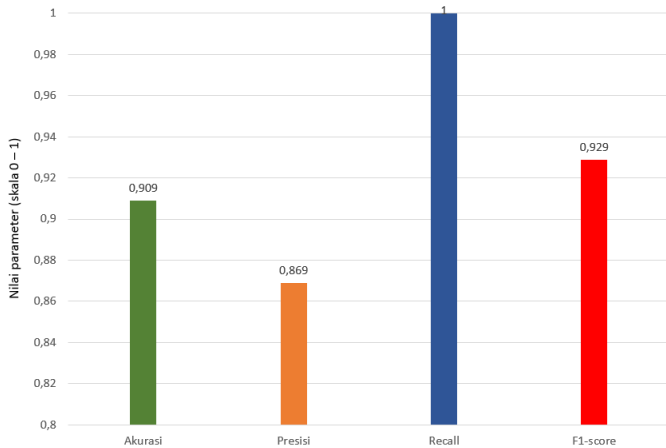
Gambar 7 menunjukkan hasil *training* dan validasi dataset. Dalam pelatihan ini digunakan 2.000 dataset. Pada proses ini diperoleh rata-rata akurasi pelatihan sebesar 0,75 atau 75%. Serta diperoleh juga confusion matrix dari hasil *training* dataset.



Gambar 8. Confusion matrix

Gambar 8 adalah matriks konfusi yang telah didapatkan dari hasil training dataset. Dalam matriks konfusi ini diperoleh beberapa nilai sebagai berikut:

1. True Positive = 200
2. False Negative = 0
3. False Positive = 30
4. True Negative = 100

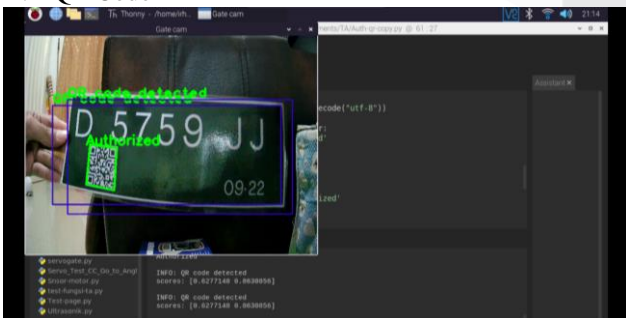


Gambar 9.

Hasil perhitungan parameter dari confusion matrix

Gambar 9 menunjukkan nilai-nilai parameter (akurasi, presisi, *recall*, dan *F1-score*) yang diperoleh dari confusion matrix. Semua nilai parameter yang telah didapatkan adalah baik karena nilai atau skornya lebih dari 80% dengan batasan hingga 1 atau 100% [8][10].

B. QR Code



Gambar 10.

Hasil pembacaan QR code

Gambar 10 merupakan hasil percobaan pembacaan QR code. Percobaan ini melakukan uji pembacaan kode QR oleh sistem melalui kamera dengan beberapa

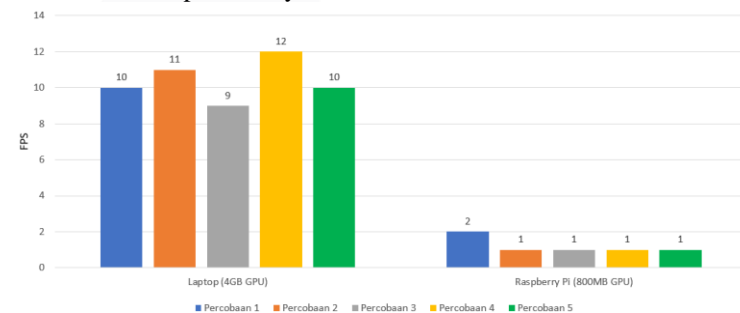
pengaturan jarak; yaitu 20 cm, 25 cm, 30 cm, 35 cm, 40 cm, 45 cm, dan 50 cm. Pada setiap jarak, setiap percobaan dilakukan sebanyak 7 kali. Berdasarkan hasil percobaan, sistem dapat membaca kode QR dengan baik pada jarak 20 sampai 40 cm. Namun pembacaan tidak terlalu optimal (terkadang tidak terbaca pada beberapa percobaan) pada jarak 45 cm dan 50 cm.

C. Performa *Computer Vision* dengan 2 GPU Berbeda

Pada pengujian ini, kinerja pendeteksian objek diuji menggunakan *computer vision real-time* menggunakan webcam dan 2 perangkat yang berbeda. Pengujian ini dilakukan untuk mengetahui pengaruh GPU perangkat terhadap pendeteksian objek. Ada 2 perangkat yang akan digunakan dengan spesifikasi berbeda:

1. Laptop Lenovo ideapad 110
 - a. RAM: 8GB
 - b. *Processor*: Intel Core i3, 4 CPU @ 2GHz, 64-bit
 - c. GPU: 4GB
2. Minikomputer Raspberry Pi 4 Model B
 - a. RAM: 4GB
 - b. *Processor*: Broadcom BCM2711, Quad core, 64-bit, SoC @ 1.5GHz
 - c. GPU: 800MB

Dalam pengujian pengaruh GPU terhadap kinerja *computer vision*, dilakukan 5 percobaan untuk setiap perangkat dengan ukuran GPU yang berbeda. Berikut adalah hasil eksperimennya:



Gambar 11.

Hasil pengujian pengaruh GPU terhadap FPS

Berdasarkan hasil percobaan seperti pada gambar 11, dapat disimpulkan bahwa ukuran memori GPU sangat berpengaruh terhadap kinerja *computer vision*. Semakin besar dan bagus GPU pada komputer maka semakin baik pula *frame per second* yang akan dihasilkan [17].

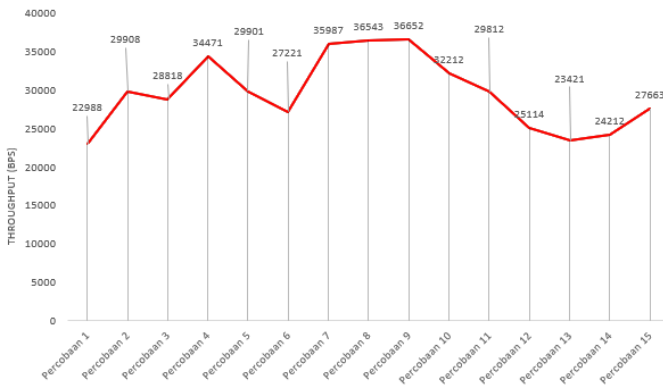
D. *Quality of Service*

Perhitungan *Quality of Service* dilakukan untuk mengukur kualitas jaringan khususnya pada saat percobaan. Sebelum melakukan analisis QoS, telah dirancang pengalamanan IP dari Raspberry Pi dan Firebase seperti yang ditunjukkan oleh tabel 1 di bawah. Tujuan rancangan pengalamanan IP adalah untuk mempermudah analisis (filter trafik jaringan) saat pengukuran QoS di Wireshark.

Pengukuran QoS dilakukan dari pengiriman data deteksi objek dan pembacaan kode QR dari Raspberry Pi ke database *real-time* Firebase dan dilakukan 15 kali percobaan untuk mendapatkan hasil pengukuran QoS.

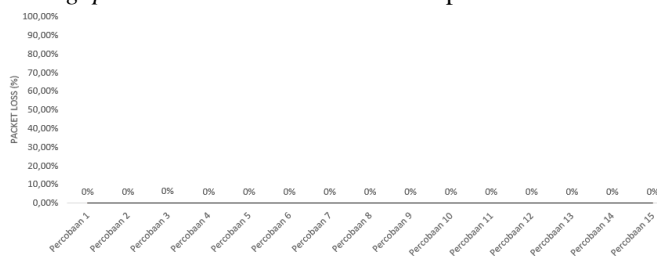
Tabel 1.
Rancangan pengalaman IP untuk uji QoS

Media	Alamat IP
Raspberry Pi	192.168.43.85
Firebase	74.125.24.139



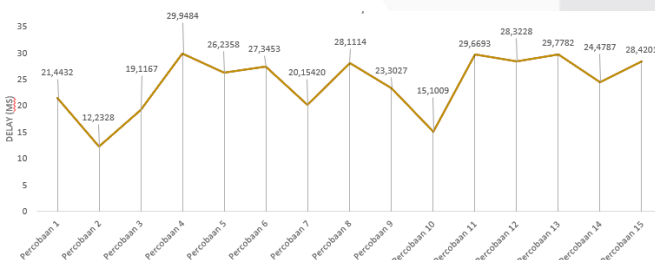
Gambar 12.
Grafik pengujian throughput

Throughput adalah jumlah total kedatangan paket yang berhasil diamati di tujuan selama interval waktu tertentu dibagi dengan durasi interval waktu tersebut. Berdasarkan hasil perhitungan seperti gambar 12 di atas, throughput rata-rata adalah 29.661.533 bps.



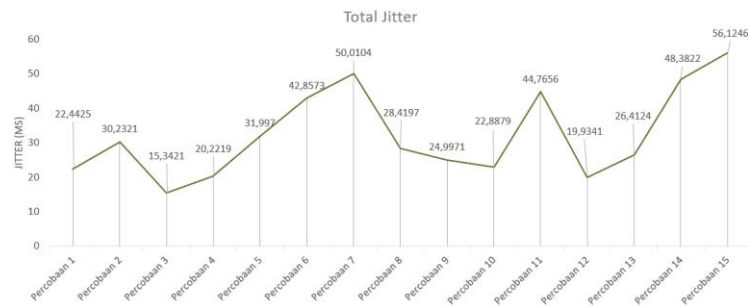
Gambar 13.
Grafik pengujian packet loss

Pengujian performansi jaringan selanjutnya adalah pengambilan data packet loss. Packet loss adalah jumlah paket yang hilang selama pengiriman. Dari hasil 15 kali uji coba seperti ditunjukkan gambar 13 di atas, rata-rata packet loss adalah 0% sehingga packet loss dikategorikan sangat baik karena sesuai dengan standar ketentuan yang digunakan[15].



Gambar 14.
Grafik pengujian delay

Pada eksperimen observasi delay, diambil 15 data observasi. Dari hasil perhitungan seperti gambar 14 diatas, didapatkan delay rata-rata sebesar 24,244 ms, sehingga delay dikategorikan sangat baik berdasarkan standar yang digunakan[15].



Gambar 15.
Grafik pengujian jitter

Pada percobaan observasi jitter diambil 15 data jitter dari observasi. Dari hasil perhitungan yang ditunjukkan gambar 15, rata-rata jitter adalah 32,3351 ms sehingga delay dikategorikan baik berdasarkan standar yang digunakan[15].

V. KESIMPULAN

Kesimpulan dari analisis eksperimen yang diperoleh, antara lain sebagai berikut:

- Deteksi objek (pelat nomor) dapat ditangkap kamera secara real-time dengan baik hingga jarak uji maksimal 100 cm (1 meter). Dan nilai yang diperoleh dari parameter confusion matrix lebih dari 80%.
- Kamera yang terintegrasi dengan program OpenCV menggunakan bahasa pemrograman Python dapat mendeteksi dan membaca kode QR menggunakan library pyzbar Python. Pada sistem pembacaan kode QR pada plat nomor, sistem dapat membaca kode QR dan menerjemahkannya (decode) dengan baik hingga jarak 40 cm dari jarak uji maksimal hingga 50 cm.
- Sistem dapat mengirimkan informasi hasil deteksi objek dan pembacaan kode QR dari Raspberry Pi ke database Firebase dengan baik; dengan throughput rata-rata 26431 bps, packet loss rata-rata 0%, delay rata-rata 9,1318 ms, dan jitter rata-rata 16,5996 ms.

REFERENSI

- Mathilda Gian Ayu, "Perkembangan dan Penggunaan IoT di Indonesia Tahun 2021 Diprediksi Meningkat," *Cloud Computing Indonesia*, 2020. <https://www.cloudcomputing.id/berita/perkembangan-dan-penggunaan-iot-di-indonesia>.
- Y. Zhao, J. Gu, C. Liu, S. Han, Y. Gao, and Q. Hu, "License Plate Location Based on Haar-like Cascade Classifiers and Edges," *Second WRI Glob. Congr. Intell. Syst. 2010 Second WRI Glob. Congr. Intell. Syst. Licens.*, pp. 2–5, 2010, doi: 10.1109/GCIS.2010.55.
- B. Y. Amirgaliyev, K. K. Kuvatov, Z. Y. Baibatyr, M. Z. Kairanbay, C. A. Kenshimov, and A. K. Jantassov, "License Plate Verification Method For Automatic License Plate Recognition Systems," *2015 Twelve Int. Conf. Electron. Comput. Comput.*, vol. 1, no. 1, pp. 3–5, 2015, doi: 10.1109/ICECCO.2015.7416892.
- A. Menon, "Detection And Recognition of Multiple License Plate From Still Images," *2018 Int. Conf. Circuits Syst. Digit. Enterp. Technol.*, pp. 1–5, 2018.

- [5] Y. Yudhanto, "Apa itu IOT (Internet Of Things) ?," 2007. [Online]. Available: <https://ilmukomputer.org/wp-content/uploads/2015/05/apa-itu-iot-internet-of-things.pdf>.
- [6] M. Asad, S. Qaisar, and A. Basit, "Client Based Access Layer QoS Provisioning in Beyond 5G IoT Networks," *2020 3rd Int. Conf. Adv. Commun. Technol. Netw.*, vol. 1, no. 2, pp. 1–8, 2022, doi: 10.1109/CommNet49926.2020.9199612.
- [7] I. O. P. C. Series and M. Science, "Face Mask Detection for Covid-19 Pandemic using Pytorch in Deep Learning," *IOP Conf. Ser. Mater. Sci. Eng.*, 2021, doi: 10.1088/1757-899X/1070/1/012061.
- [8] F. Rofii, G. Priyandoko, M. I. Fanani, and A. Suraji, "Vehicle Counting Accuracy Improvement By Identity Sequences Detection Based on Yolov4 Deep Neural Networks," *Teknik*, vol. 42, no. 2, pp. 169–177, 2021, doi: 10.14710/teknik.v42i2.37019.
- [9] T. D. Indriasari, M. Sc, F. S. R. S. T, and M. Kom, "Analisis dan Perancangan Layanan Perpustakaan UAJY Berbasis Mobile dengan Memanfaatkan QR Code," *http://e-journal.uajy.ac.id/*, vol. 1, no. 1, pp. 1–47, 2012.
- [10] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," 2020, [Online]. Available: <http://arxiv.org/abs/2004.10934>.
- [11] G. T. U. A. Colleges *et al.*, "Microsoft COCO: Common Objects in Context," *Eccv*, vol. 1, no. 1, pp. 740–755, 2014, doi: 10.48550/arXiv.1405.0312.
- [12] Y. Orland Kristiawan, J. Triyono, and P. Haryani, "Membangun Server Portable Berbasis Raspberry Pi Sebagai Media Streaming," *J. JARKOM*, vol. 7, no. 1, pp. 35–43, 2019.
- [13] A. Kumar and G. Rahul, "IOT based Electrical Device Surveillance and Control System," *2019 4th Int. Conf. Internet Things Smart Innov. Usages*, pp. 1–5, 2019.
- [14] N. Seitz, "ITU-T QoS Standards for IP-Based Networks," no. June, pp. 82–89, 2003.
- [15] ITU-T, "G.1010: End-user multimedia QoS Categories," *Int. Telecommun. Union*, vol. 1010, 2001, [Online]. Available: http://scholar.google.com.au/scholar?hl=en&q=ITU-T+Recommendation+G.1010&btnG=&as_sdt=1,5&as_sdtp=#7.
- [16] J. Page, M. Bain, and F. Mukhlis, "The Risks of Low Level Narrow Artificial Intelligence," *2018 IEEE Int. Conf. Intell. Saf. Robot.*, pp. 1–6, 2018.
- [17] A. Mackin, F. Zhang, and D. R. Bull, "A Study of High Frame Rate Video Formats," *IEEE Trans. Multimed.*, vol. PP, pp. 1–14, 2018, doi: 10.1109/TMM.2018.2880603.