

Validasi Kendaraan Pada Sistem Parkir Cerdas Berbasis Pengenalan Citra Menggunakan Raspberry Pi 3

Agit Amrullah¹, Uyock Anggoro Saputro^{*2}

^{1,2}Informatika, Universitas Amikom Yogyakarta

E-mail: ¹agit@amikom.ac.id, ^{*2}uyock@amikom.ac.id

Abstrak

Solusi parkir cerdas dalam mengatasi permasalahan lahan parkir telah dikembangkan baik dari bentuk sistem informasi hingga terintegrasi. Pada ketersediaan zona parkir, beberapa solusi parkir cerdas, menggunakan bentuk implementasi sensor ultrasonik, infrared, dan pengenalan citra. Permasalahan yang terjadi pada kendaraan yang akan parkir pada zona parkir adalah validasi objek kendaraan parkir, sehingga akan sangat dimungkinkan objek bukan kendaraan dianggap sebagai kendaraan yang akan parkir. Pada penelitian ini akan dikembangkan mekanisme pendeteksian dan validasi dini pada kendaraan yang akan parkir pada suatu zona parkir cerdas berbasis pengenalan citra kendaraan menggunakan peralatan Microcontroller dengan tipe ESP32-Cam dan Raspberry Pi 3 sebagai server pada sistem parkir cerdas. Pada hasil pengujian yang didapatkan, kendaraan ataupun objek dapat terdeteksi dengan baik melalui akuisisi citra pada sisi belakang kendaraan yang akan terparkir dengan bentuk output validasi pada lokasi parkir menggunakan buzzer sebagai output hasil validasi dan pengarah jarak tepi zona parkir dengan kendaraan yang terdeteksi dan tervalidasi dengan tingkat akurasi sebesar 87% dan waktu validasi rata-rata 22 detik dengan menggunakan metode CNN pada framework YOLO v3.

Kata Kunci—*parkir cerdas, cnn, esp32cam, raspberry pi 3*

1. PENDAHULUAN

Solusi parkir cerdas terus dikembangkan dalam menangani permasalahan dan mengembangkan solusi parkir di kemudian hari. Bahkan dalam menunjang pertumbuhan ekonomi di masyarakat dengan seiring pertumbuhan transportasi[1], permasalahan lahan parkir akan terus meningkat. Berdasarkan Bura et.al, beberapa solusi parkir cerdas yang menggunakan sensor pada area parkir untuk mengidentifikasi apakah terdapat lahan yang kosong ataupun tidak, telah dirasa cukup untuk memberikan informasi terkait lahan parkir[2]. Namun pada kenyataannya kelemahan penggunaan sensor, baik sensor menggunakan bentuk sensor ultrasonik ataupun lainnya dalam mendeteksi ketersediaan lahan [3], belum memiliki sistem validasi kendaraan yang terparkir. Hal ini timbul karena cara kerja sensor ultrasonik yang dipergunakan adalah penggunaan gelombang suara yang dikirimkan (transmit) dan penerimaan gelombang suara (receive) yang mana akan mendeteksi objek apabila mengalami propagasi dari gelombang suara yang diterima [4].

Metode lain dalam melakukan pendeteksian dikembangkan dalam bentuk citra visual, semisal pada Semi-Autonomous Parking Measurement System (SAPOMS) [5] yang melakukan akuisisi lahan parkir menggunakan citra visual beserta *splicing image* mobil yang terparkir pada zona parkir yang dipergunakan. Bentuk akuisisi citra memberikan kemudahan dalam akuisisi dan rekognisi secara visual, serta pada sistem tersebut penggunaan Raspberry Pi sebagai pemroses utama menghasilkan *cost* yang lebih efisien dalam jangka panjang baik pada perawatan dan pengembangan.

Review sematis pada sistem parkir cerdas yang diimplementasikan pada suatu kota [6], menghasilkan kesimpulan bahwa parkir cerdas tidak menyelesaikan permasalahan secara menyeluruh pada konsep *smart city*, namun lebih kepada mengurangi dampak seperti polusi, kemacetan, kecelakaan lalu lintas, waktu yang terbuang karena pencarian tempat parkir setidaknya membutuhkan 3,5 hingga 14 menit [7], dan dampak psikologis pengendara yang dilakukan pada konsep parkir cerdas dengan memaksimalkan manajemen pada zona parkir. Pada penelitian yang dilakukan oleh Lookmuang et.al [8], desain parkir cerdas yang diusulkan adalah dapat mengakomodasi informasi pengenalan citra pelat nomor kendaraan, manajemen zona parkir, dan fungsionalitas pencarian kendaraan pada zona parkir berdasarkan data nomor pelat.

Pada penelitian ini, akan diusulkan bagaimana menangani permasalahan akuisisi objek kendaraan awal yang akan parkir pada zona parkir cerdas dengan melakukan validasi jenis kendaraan yang akan terparkir. Penelitian menggunakan akuisisi citra objek mobil sebagai kendaraan yang akan terparkir pada zona parkir mobil dengan citra bagian belakang mobil sesuai mekanisme parkir paralel pada zona parkir secara umum. Tujuan dari penelitian ini adalah melakukan degradasi distrupsi informasi melalui rekognisi citra kendaraan sehingga kendaraan yang terparkir sesuai pada zona parkir yang diberikan dan jenis kendaraan yang sesuai pada zona parkir yang dipergunakan, dengan berbasis mikrokontroler dalam melakukan akuisisi citra dan rekognisi citra menggunakan *single board computer* sebagai edge computing sebelum dilakukan pemrosesan lanjut pada service parkir cerdas.

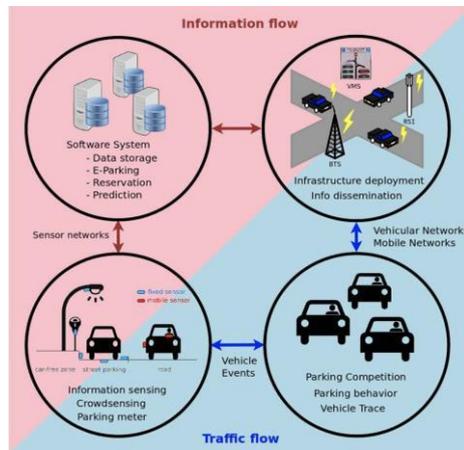
2. METODE PENELITIAN

2.1. Kajian Penelitian

Pada penelitian Salsabillah, penggunaan ESP32-Cam pada deteksi kendaraan dengan realtime database menggunakan Firebase, menghasilkan pendeteksian kendaraan berhasil dilakukan dengan delay tertinggi 65,95 detik namun bentuk komputasi yang dilakukan dalam pemrosesan deteksi kendaraan tidak didefinisikan secara spesifik[9]. Pada penelitian Hasan, permodelan parkir cerdas dalam memberikan alokasi parkir yang kosong menggunakan Google Cloud berbasis IoT, menghasilkan reduksi waktu 50% dengan menggunakan infrastruktur edge computing dengan raspberry pi [10]. Penelitian Nguyen dalam sistem monitoring ketersediaan zona parkir menggunakan raspberry pi 4 menghasilkan tingkat akurasi 88-97% dengan pemrosesan rata-rata 35 detik dalam setiap pemberian informasi zona parkir yang kosong [11].

2.2. Parkir Cerdas

Secara umum, parkir dibedakan menjadi empat kategori, diantaranya adalah parkir pada ruang jalan (on-street), parkir umum diluar ruang jalan (public off-street), parkir swasta non-residensial diluar ruang jalan (private non-residential off-street), dan parkir pribadi dalam pemukiman (private residential parking [12]. Semakin berkembangnya permasalahan dalam suatu kota, terutama masalah manajemen parkir, diperlukan integrasi pengetahuan dari multidisiplin ilmu untuk memberikan solusi yang dapat menyelesaikan permasalahan parkir dalam suatu kota [13]. Secara mendasar dalam parkir cerdas, setidaknya terdapat 2 alur informasi yang dikelola yaitu informasi dan kemacetan. Bentuk esensi dari ekosistem informasi parkir cerdas dapat dilihat pada gambar di bawah ini :



Gambar 1. Ekosistem parkir cerdas [13]

Pada gambar 1, dapat dilihat alur informasi dari pengendara mencari tempat parkir yang kosong akan membuat suatu pencarian lahan bersifat kompetitif. Solusi sistem parkir cerdas yang ditawarkan akan memberikan informasi terkait ketersediaan zona (Variable Message Signs/VMS) parkir sehingga pengendara akan lebih cepat untuk mendapatkan informasi mengenai ketersediaan lahan parkir.

2.3. ESP32Cam

ESP32 adalah chip generasi ke-dua yang dikeluarkan oleh perusahaan Espressif dengan berisi Wifi dan Bluetooth. Modul kamera kemudian ditambahkan dalam beberapa vendor yang mengembangkan menggunakan mikrokontroler ini. Resolusi kamera pada seri OV-2640 dapat menggunakan berbagai macam resolusi, dengan resolusi tertingginya adalah 1600x1200 [14]. Berikut ini adalah waktu buffer time untuk ESP32Cam dalam menangkap gambar :

Tabel 1. Waktu Pemrosesan Gambar pada ESP32Cam [14]

Resolusi	Pixel	Durasi Akuisisi Gambar (ms)	Durasi Proses Gambar (ms)	Total Durasi (ms)
QVGA	7600	70	9	79
VGA	307200	145	54	199
XGA	786432	463	130	593
SXGA	1310720	540	254	792

2.4. Sensor Ultrasonik HC-SR04

HC-SR04 merupakan rangkaian sensor jarak dengan gelombang ultrasonik, dimana pada sensor tersebut terdapat dua bagian yaitu bagian penerima dan pemancar yang berfungsi sebagai pembangkit gelombang dan penerima gelombang. Sensor ultrasonik HC-SR04 yang memiliki 4 pin. satu pin VCC adalah pin input tegangan dan pin GND diimbangi untuk grounding, sedangkan dua pin lainnya adalah pin trigger dan echo yang akan mempengaruhi gelombang ultrasonik itu sendiri. (Wijayanti, M. 2022).

2.5. Raspberry Pi 3

Sebuah *single-chip* komputer, Raspberry Pi, yang merupakan golongan micro CPU pada keluarga ARM dapat menjalankan operasi dengan basis perintah Linux. Penggunaan pendeteksian wajah dan gerakan dalam menggunakan Raspberry Pi menggunakan model B [15].

2.6. OpenCV

Open Source Computer Vision Library (OpenCV) adalah salah satu open library yang dapat dipergunakan dalam bahasa pemrograman Python dan dipergunakan dalam pemrosesan citra secara *real-time* [16]. Bentuk OpenCV sebagai sebuah library, menjadikannya dapat diintegrasikan kedalam sebuah web service.

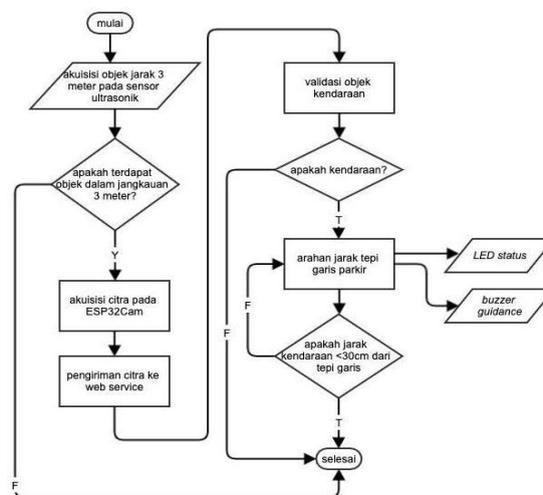
2.7. You Only Look Once (YOLO) v3

Algoritma YOLO pada dasarnya adalah penggunaan metode convolutional neural network (cnn) yang mana dapat memprediksi multi *box* lokasi pada sebuah objek. Pengembangan versi 3 pada YOLO, akurasi dan kecepatan ditingkatkan dengan menggunakan data set dari COCO dan ImageNet dapat melakukan training objek sejumlah 9.000 item [17].

3. HASIL DAN PEMBAHASAN

3.1. Sistem yang Diusulkan

Bentuk sistem yang dibangun pada penelitian ini adalah berbentuk pengembangan dari model sistem parkir cerdas sebelumnya dengan fokus pada pendeteksian dan validasi objek pada zona parkir kendaraan mobil yang akan melakukan parkir pada zona parkir yang tersedia. Bentuk alur sistem yang akan dibangun ditampilkan pada gambar 2.

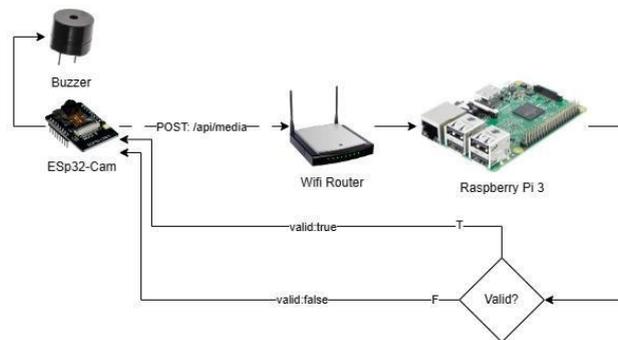


Gambar 2. Alur Sistem Validasi Kendaraan pada Parkir Cerdas

3.2. Desain Service pada Server

Proses akuisisi citra dilakukan melalui ESP32-Cam kemudian dilakukan pengiriman data ke server Raspberry Pi 3 melalui wifi router sebagai gateway jaringan nirkabel ke bentuk kabel.

Pemrosesan citra atas validasi kendaraan pada Raspberry Pi 3, akan menghasilkan output data validasi dengan nilai *true* atau *false*, yang mana akan menjadi input pada proses berikutnya pada mikrokontroler untuk menentukan buzzer akan menjadi alarm dengan bentuk beep delay 0,5 detik selama 2 menit apabila data akuisisi bernilai tidak valid atau akan menjadi pendamping dalam parkir kendaraan secara tepat pada zona yang tersedia dengan bentuk beep delay 3 detik pada jarak kendaraan ke titik belakang parkir sepanjang 3 meter, beep delay 2 detik untuk jarak kendaraan ke titik belakang parkir sepanjang 2 meter dan beep dengan delay 1 detik untuk jarak kendaraan ke titik belakang parkir sepanjang 1 meter, serta beep dengan nada panjang untuk jarak kurang dari 1 meter kendaraan ke titik belakang parkir. Bentuk proses arsitektural validasi pada sistem parkir cerdas yang diusulkan dapat dilihat pada gambar 3 di bawah ini.



Gambar 3. Proses Validasi pada Sistem

Pada sisi server, Raspberry Pi 3 menjalankan service Http dengan bahasa pemrograman Python dalam melakukan validasi kendaraan. Penelitian ini menggunakan mobil sebagai bentuk kendaraan yang dianggap valid, sehingga apabila selain mobil (sepeda motor, manusia, dan sepeda) pada jarak 3 meter dari titik zona akuisisi selain mobil mendekat maka akan menghasilkan output validasi *salah/failed*. Bentuk *Application Programming Interface* (API) service yang dibangun pada sistem ini, dapat dilihat pada tabel 2 dibawah ini.

Tabel 2. API Service pada Sistem yang Dibangun

URL API	Http Verbs	Input	Output
/api/media	POST	Body : { name: "imageFile", content-Type: image/jpeg }	<p><i>Success</i> :</p> <pre>{'success':True, 'recognized': True, 'data': { 'filename': newname+ext, 'dfilename': newname+"_detected"+ext,'estimatedtime':end_t ime-start_time }, 'code': status_code, 'message': res[200] }</pre> <p><i>Failed</i>:</p> <pre>{ 'success': 'false', 'code': status_code, 'message': res[400], 'error': str(_error), }</pre>
/api/media/ slug	GET	Slug-image	Raw file result

3.3. Akuisisi Citra

Proses akuisisi citra dilakukan melalui ESP32-Cam kemudian dilakukan pengiriman data ke server Raspberry Pi 3 melalui wifi router sebagai gateway jaringan nirkabel ke bentuk kabel. Pemrosesan citra atas validasi kendaraan pada Raspberry Pi 3, akan menghasilkan output data

Dalam melakukan akuisisi citra, mobil akan mendekati jarak parkir dengan jangkauan 3 meter dari titik zona parkir kendaraan. Apabila jarak telah memenuhi, maka citra belakang kendaraan akan diakuisisi oleh mikrokontroler dengan resolusi gambar 800x600 pixels (SVGA). Mekanisme akuisisi citra dapat dilihat pada definisi gambar 4 dibawah ini.



Gambar 4. Proses Akuisisi Citra pada Kendaraan Berdasarkan Jarak dari Sensor

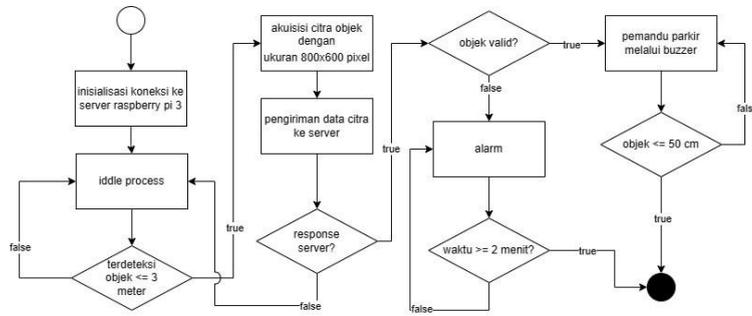
Pengujian yang didapatkan dari objek yang mendekati sensor jarak didapatkan hasil yang dapat dilihat pada tabel 3 sebagai berikut.

Tabel 3. Hasil Akuisisi Objek Mobil Pada Single Sensor Jarak HC-SR04

Jarak Objek (meter)	Angle	Status
1	0°	Terdeteksi
1	22,5°	Terdeteksi
1	45°	Terdeteksi
2	0°	Terdeteksi
2	22,5°	Terdeteksi
2	45°	Terdeteksi
3	0°	Terdeteksi
3	22,5°	Terdeteksi
3	45°	Terdeteksi

3.4. Validasi Citra

Citra yang telah diakuisisi kemudian dilakukan validasi, mekanisme validasi apabila kendaraan terverifikasi sebagai sebuah kendaraan berjenis mobil maka akan diberikan notasi suara dari buzzer sebagai arahan untuk parkir sesuai dengan garis zona parkir yang tersedia, namun apabila objek yang dideteksi tidak sesuai akan memberikan alarm dari buzzer selama 2 menit dengan tone pada nada tinggi (sinyal digital dengan output nilai 200). Mekanisme validasi citra dijabarkan pada diagram flowchart pada gambar 5 di bawah ini.



Gambar 5. Mekanisme Validasi Citra

Kendaraan yang telah diakuisisi citra oleh mikrokontroller, kemudian akan mengirimkan hasil gambar ke server Raspberry melalui protokol servis Http. Gambar yang dikirimkan memiliki besaran file 46 KB dengan resolusi 800x600 pixel yang diproses menjadi informasi dengan hasil terdeteksi atau tidak terdeteksi kendaraan dengan jenis mobil. Pada gambar 6 dibawah ini sebagai contoh validasi kendaraan dengan status terdeteksi, dan akan memberikan informasi ke mikrokontroller untuk dilakukan arahan sinyal ke zona parkir dengan jarak <30 cm dari sensor jarak



Gambar 6. Validasi Citra pada Kendaraan yang Terdeteksi Sebagai Kendaraan Berjenis Mobil

Pengujian validasi objek dilakukan pada objek selain kendaraan yang mendekat pada zona parkir. Akuisisi citra akan dikirimkan apabila objek yang mendekat pada zona parkir dengan jarak <= 3 meter dari sensor jarak. Pada gambar 7 dapat dilihat akuisisi citra yang menangkap objek pada jarak 3 meter namun tidak terdeteksi sebagai objek kendaraan mobil.



Gambar 7. Akuisisi Citra dengan Hasil Tidak Terdeteksi Kendaraan Berjenis Mobil

3.5. Evaluasi

Berikut ini adalah hasil pengujian dengan iterasi 35 kali berdasarkan pengambilan citra dan hasil validasi objek kendaraan berjenis mobil yang dapat dilihat pada tabel 4 di bawah ini.

Tabel 4. Hasil Estimasi dan Akurasi Validasi Citra Kendaraan

Percobaan Ke-	Estimasi (detik)	Akurasi (%)
1	24,394713	83,50
2	21,050581	83,32
3	20,94165	94,60
4	21,51963	84,91
5	21,134072	0
6	21,198803	83,21
7	21,19819	85,94
8	21,49397	92,16
9	21,301567	80,50
10	21,151005	89,66
11	20,946236	36,22
12	21,102701	0
13	21,483287	90,72
14	21,226746	92,88
15	21,488546	0
16	21,073561	0
17	31,334863	99,09
18	21,500889	91,97
19	21,534014	88,77
20	21,919332	82,59
21	21,403743	90,77
22	21,088165	83,88
23	21,254539	91,98
24	21,125736	93,51
25	20,971801	83,56
26	21,124318	86,60
27	20,955681	87,31
28	32,375843	93,25
29	21,04423	0
30	21,374896	98,27
31	31,905471	78,89
32	21,144249	88,28
33	21,212203	93,69
34	28,063533	94,46
35	21,182675	85,63

Pada tabel 4 diatas dapat pada total 35 iterasi yang dilakukan terhadap deteksi kendaraan mobil di server Raspberry, rata-rata estimasi deteksi dilakukan pada kendaraan adalah sebesar 22,43489826 detik dengan tingkat akurasi rata-rata sebesar 74,58%. Beberapa hasil pengujian

memiliki hasil 0%, yang berarti tidak terdeteksi sebagai citra kendaraan berjenis mobil. Hal ini dapat terjadi dikarenakan hasil akuisisi citra kendaraan tidak sempurna pada pencahayaan yang didapatkan atau ketika parkir kendaraan memarkirkan kendaraan dengan tanpa melewati zona parkir yang seharusnya, sehingga ketika citra diakuisisi, objek terlalu dekat pada peralatan mikrokontroler. Pada gambar 8 dapat dilihat objek yang terlalu dekat pada mikrokontroler yaitu pada jarak ≤ 1 meter untuk kendaraan mobil berjenis *city car* tidak terdeteksi sebagai kendaraan berjenis mobil.



Gambar 8. Validasi Citra pada Kendaraan yang Tidak Terdeteksi Sebagai Kendaraan Berjenis Mobil

4. KESIMPULAN

Solusi parkir cerdas terus dikembangkan dalam menangani permasalahan dan mengembangkan solusi parkir di kemudian hari. Berdasarkan penelitian yang dilakukan proses validasi citra berhasil dilakukan dan mampu mengidentifikasi objek pada jarak kurang dari 3 meter dan sudut 0° sampai 45°, dengan tingkat akurasi tertinggi pada angka 99.09 persen dan rata-rata akurasi dari 35 percobaan yang dilakukan sebesar 74,58% dengan respond time rata-rata 22,43489826 detik.

DAFTAR PUSTAKA

- [1] M. Tsani, A. Rupaka, L. Asmoro, and B. Pradana, "ANALISIS SENTIMEN REVIEW TRANSPORTASI MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE BERBASIS CHI SQUARE," *smartcomp*, vol. 9, no. 1, pp. 35–39, Jan. 2020, doi: 10.30591/smartcomp.v9i1.1817.
- [2] H. Bura *et al.*, "An Edge Based Smart Parking Solution Using Camera Networks and Deep Learning," *IEEE international conference on cognitive computing (ICCC)*, p. 8, Jul. 2018.
- [3] I. Jung, "An IoT-based smart parking management system," *Int. J. Computational Vision and Robotics*, vol. 10, no. 2, p. 11, 2020.
- [4] V. A. Zhmud, N. O. Kondratiev, K. A. Kuznetsov, V. G. Trubin, and L. V. Dimitrov, "Application of ultrasonic sensor for measuring distances in robotics," *J. Phys.: Conf. Ser.*, vol. 1015, p. 032189, May 2018, doi: 10.1088/1742-6596/1015/3/032189.
- [5] N. Pazos *et al.*, "Dynamic Street-Parking Optimisation," in *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, Crans-Montana, Switzerland: IEEE, Mar. 2016, pp. 1020–1026. doi: 10.1109/AINA.2016.171.
- [6] M. Z. Abidin and R. Pulungan, "A Systematic Review of Machine-vision-based Smart Parking Systems," *Scientific Journal of Informatics*, vol. 7, no. 2, p. 16, 2020.

- [7] V. Paidi, H. Fleyeh, J. Håkansson, and R. G. Nyberg, "Smart Parking Tools Suitability for Open Parking Lots: A Review.," in *Proceedings of the 4th International Conference on Vehicle Technology and Intelligent Transport Systems*, Funchal, Madeira, Portugal: SCITEPRESS - Science and Technology Publications, 2018, pp. 600–609. doi: 10.5220/0006812006000609.
- [8] R. Lookmuang, K. Nambut, and S. Usanavasin, "Smart parking using IoT technology," in *2018 5th International Conference on Business and Industrial Research (ICBIR)*, Bangkok: IEEE, May 2018, pp. 1–6. doi: 10.1109/ICBIR.2018.8391155.
- [9] A. A. Salsabillah, I. H. Santoso, and A. I. Irawan, "Implementation On Smart Parking Using ESP32Cam In Telkom University Premises," vol. 8, no. 6, p. 3138, Dec. 2022.
- [10] M. O. Hasan, Md. M. Islam, and Y. Alsaawy, "Smart Parking Model based on Internet of Things (IoT) and TensorFlow," in *2019 7th International Conference on Smart Computing & Communications (ICSCC)*, Sarawak, Malaysia, Malaysia: IEEE, Jun. 2019, pp. 1–5. doi: 10.1109/ICSCC.2019.8843651.
- [11] T. Nguyen *et al.*, "An Adaptive Vision-based Outdoor Car Parking Lot Monitoring System," in *2020 IEEE Eighth International Conference on Communications and Electronics (ICCE)*, Phu Quoc Island, Vietnam: IEEE, Jan. 2021, pp. 445–450. doi: 10.1109/ICCE48956.2021.9352090.
- [12] A. D. Limantara, Y. C. S. Purnomo, and S. W. Mudjanarko, "PEMODELAN SISTEM PELACAKAN LOT PARKIR KOSONG BERBASIS SENSOR ULTRASONIC DAN INTERNET OF THINGS (IOT) PADA LAHAN PARKIR DILUAR JALAN," p. 10.
- [13] T. Lin, H. Rivano, and F. Le Mouel, "A Survey of Smart Parking Solutions," *IEEE Trans. Intell. Transport. Syst.*, vol. 18, no. 12, pp. 3229–3253, Dec. 2017, doi: 10.1109/TITS.2017.2685143.
- [14] K. Dokic, "Microcontrollers on the Edge – Is ESP32 with Camera Ready for Machine Learning?," in *Image and Signal Processing*, A. El Moataz, D. Mammass, A. Mansouri, and F. Nouboud, Eds., in *Lecture Notes in Computer Science*, vol. 12119. Cham: Springer International Publishing, 2020, pp. 213–220. doi: 10.1007/978-3-030-51935-3_23.
- [15] Z. Balogh, M. Magdinv, and G. György, "Motion Detection and Face Recognition using Raspberry Pi, as a Part of, the Internet of Things," *APH*, vol. 16, no. 3, May 2019, doi: 10.12700/APH.16.3.2019.3.9.
- [16] I. B. Mustaffa and S. F. B. M. Khairul, "Identification of fruit size and maturity through fruit images using OpenCV-Python and Rasberry Pi," in *2017 International Conference on Robotics, Automation and Sciences (ICORAS)*, Melaka: IEEE, Nov. 2017, pp. 1–3. doi: 10.1109/ICORAS.2017.8308068.
- [17] L. Ouyang and H. Wang, "Vehicle target detection in complex scenes based on YOLOv3 algorithm," *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 569, p. 052018, Aug. 2019, doi: 10.1088/1757-899X/569/5/052018.