

Penghitung Jumlah Pengunjung Objek Wisata dengan Metode *Deep Learning MobileNet-SSD*

Mohammad Heri Saputra¹, Danang Erwanto², Royb Fatkhur Rizal³

^{1,2,3}Program Studi Teknik Elektro,

Fakultas Teknik,

Universitas Islam Kediri, Kediri

¹herisaputra33@yahoo.co.id, ²danangerwanto@uniska-kediri.ac.id,

³fatkhurizal15@gmail.com

Abstrak

Pada kondisi pandemi Covid-19, pengelola objek-objek wisata melakukan pembatasan jumlah pengunjung dengan penghitungan jumlah pengunjung yang masuk secara manual, yaitu dilakukan oleh pegawai/karyawan dengan menggunakan alat *counter* manual atau melalui penghitungan tiket sehingga sangat memungkinkan terjadi kesalahan dalam penghitungan jumlah pengunjung. Penelitian ini membuat sistem penghitungan jumlah pengunjung dengan memasang sebuah *webcam*. *Webcam* tersebut dapat dipasang pada pintu masuk maupun pintu keluar objek wisata. Metode yang digunakan pada penelitian ini adalah pemanfaatan model pra-terlatih *MobileNet-SSD* sebagai model *deep learning*. Dari penelitian yang dilakukan, *MobileNet-SSD* mampu mendeteksi jumlah pengunjung yang keluar dan masuk sehingga dapat menghitung jumlah pengunjung dan melakukan pembatasan jumlah pengunjung pada objek wisata.

Kata kunci: *MobileNet-SSD*, *deep learning*, pengolahan citra digital

Abstract

In the midst of Covid-19 pandemic, the managers of tourist destinations restrict the number of visitors by manually calculating the number of visitors, which is carried out by employees using manual counters or through ticket counting, making it prone to counting errors in the number of visitors calculation. This study designs a system for calculating the number of visitors by installing a webcam. The webcam can be installed at the entrance or exit of a tourist attraction. The method used in this research employs the pre-trained *MobileNet-SSD* model as a deep learning model. From the research conducted, *MobileNet-SSD* is able to detect the number of visitors coming in and out so that it is able to count the number of visitors and limit the number of visitors of tourist objects.

Keywords: *MobileNet-SSD*, *deep learning*, digital image processing

1. Pendahuluan

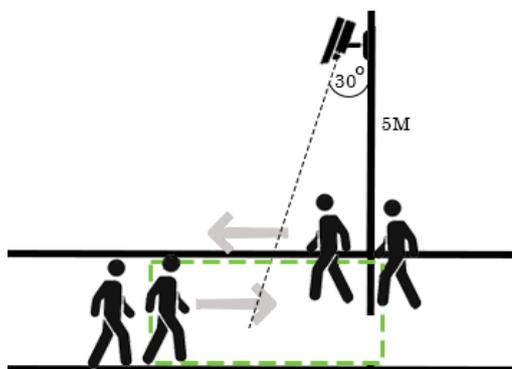
Sebagian besar pengelola objek wisata masih menggunakan cara manual untuk menghitung jumlah pengunjung, baik melalui alat *counter* maupun tiket. Penghitungan secara manual ini dirasa masih kurang efektif dan efisien karena masih memungkinkan terjadinya kesalahan penghitungan sehingga data hasil penghitungan jumlah pengunjung tidak akurat. Oleh karena itu dibutuhkan suatu sistem yang dapat digunakan untuk mendeteksi dan menghitung pengunjung objek wisata.

Pengolahan citra digital merupakan model pengolahan dan analisis citra yang banyak melibatkan persepsi visual agar dapat menjadi citra lain yang sesuai [1]. Saat ini, pemanfaatan pengolah citra digital pada instrumentasi telah cukup luas, mulai dari penghitungan objek mikroskopis, pengukuran kecepatan objek bergerak, dan lain sebagainya [2]. Selain itu kamera video maupun *webcam* juga telah diterapkan untuk kepentingan pendeteksian pergerakan manusia [3] dan keamanan, misalnya di perpustakaan, pertokoan, perkantoran, tempat wisata, dan lain sebagainya, telah menggunakan *webcam* untuk mengamati keluar masuknya pengunjung di tempat tersebut [4]. Penelitian untuk mendeteksi orang secara *realtime* dengan *MobileNet-SSD* telah dilakukan menggunakan *Movidius Neural Compute Stick pada Raspberry Pi* [5]. Selain itu, *MobileNet-SSD* diterapkan pada sistem penghitung jumlah pengunjung restoran [6].

Penelitian ini menggunakan kamera sebagai *input* video gambar dan diproses dengan menggunakan metode *deep learning MobileNet-SSD* menggunakan *OpenCV Python*. Metode tersebut dipilih karena dapat digunakan untuk mendeteksi objek secara *realtime* [7] dan mampu mendeteksi banyak objek dengan cepat [8], [9]. Hasil deteksi ini selanjutnya digunakan untuk penghitungan jumlah pengunjung objek wisata secara otomatis. Selain itu, hasil deteksi ini juga dapat dimanfaatkan untuk pembatasan jumlah pengunjung serta jarak antar pengunjung yang akan masuk maupun keluar objek wisata dalam rangka menyesuaikan dengan kondisi pandemi Covid-19. Sistem ini diharapkan mampu memberikan kontribusi dalam penerapan protokol kesehatan untuk mencegah penyebaran virus COVID-19 pada objek wisata.

2. Metode Penelitian

Penelitian ini menggabungkan pendeteksian objek dan pelacakan objek yang telah terdeteksi. Proses pendeteksian dan pelacakan dilakukan secara *realtime* dengan *sampling* gambar sebesar 30 *frame per second* (fps). Proses pendeteksian dan pelacakan menggunakan *MobileNet-SSD*, di mana jaringan *MobileNet* digunakan sebagai ekstraksi ciri dan jaringan SSD sebagai *object detector* [10]. *MobileNet* adalah kelas *deep convolutional neural networks* yang ringan dan kinerjanya lebih cepat daripada model lainnya [11]. Dalam penelitian ini, kamera diletakkan setinggi 5 meter dari permukaan tanah dengan sudut 30°. Gambar 1 memperlihatkan ilustrasi penempatan kamera pada sistem penghitung jumlah pengunjung.



Gambar 1. Ilustrasi penempatan kamera

Setelah citra diperoleh melalui *webcam*, selanjutnya dilakukan pemrosesan awal terhadap citra tersebut (*image pre-processing*) yang meliputi *resize* citra *input* kamera yang

diperkecil menjadi ukuran dengan lebar 400 pixel dengan memanfaatkan fungsi *resize()* pada modul *imutils*. *Resize* citra *input* dilakukan untuk memperkecil resolusi asli yang dihasilkan oleh *webcam* sehingga dapat mempercepat proses *deep learning*, karena pada dasarnya ukuran citra sangat berpengaruh terhadap kecepatan proses *deep learning*. Selain dilakukan *resizing* terhadap citra *input*, pada tahap *pre-processing* ini juga dilakukan kalkulasi dari nilai *mean subtraction* serta *scaling*.

Proses *image recognition* untuk mengenali manusia dimulai dari *sampling* gambar, selanjutnya gambar *sampling* dijadikan *binary large object (blob)*. *File blob* tersebut dapat diproses dengan *feed-forward* ke *neural network* dan diproses dengan *file* model *MobileNet-SSD* yang telah di-load sebelumnya. Kedua fase ini dilakukan menggunakan *MobileNet-SSD* yang telah di-load oleh modul *DNN OpenCV* tanpa harus membangun jaringan *neural network* sendiri. *MobileNet-SSD* mendeteksi objek dalam citra dengan membuat kotak pembatas (*bounding box*) dan pelabelan (ID) [12], yang selanjutnya diproses pada sistem pelacakan.

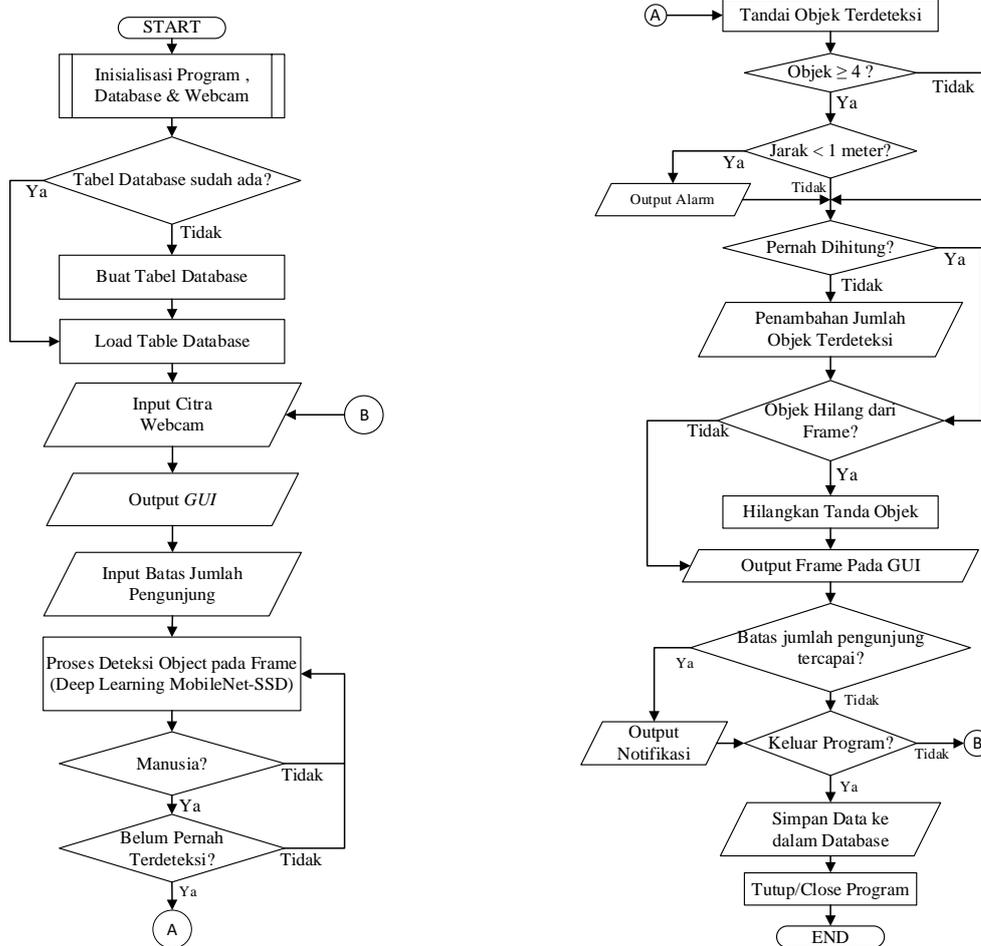
Sistem pelacakan atau *tracking* pada penelitian ini menggunakan modul *dlib*. Koordinat yang telah didapat pada proses deteksi kemudian akan dimasukkan ke dalam fungsi modul *dlib* agar dapat dilakukan pelacakan atau *tracking* pada objek yang telah terdeteksi. Proses ini akan dilakukan pada setiap objek yang terdeteksi pada *frame* secara simultan (*multi-tracking*). Setiap pelacak pada objek akan dimasukkan ke dalam daftar pelacakan untuk kemudian di-*update* posisinya pada setiap *frame*, untuk menentukan apakah objek yang terdeteksi adalah objek baru yang kemudian harus dimasukkan ke dalam daftar pelacakan ataukah objek lama yang sudah ada di dalam *frame* sebelumnya untuk selanjutnya dilakukan penghitungan jarak antar titik deteksi. Jika dekat maka akan dianggap objek yang sama dengan sebelumnya, sebaliknya jika jaraknya lebih dari yang telah ditentukan, maka akan dianggap sebagai objek baru serta akan dimasukkan ke dalam daftar pelacakan dan diberikan sebuah tanda (ID) pada objek tersebut.



Gambar 2. Titik (*dot*) hasil pendeteksian

Pada penelitian ini, sebuah kotak pembatas (*bounding box*) tidak ditampilkan sebagai *output* pada gambar, melainkan menggantinya dengan sebuah titik (*dot*) berwarna hijau tepat di tengah daripada kotak pembatas itu sendiri, seperti yang diperlihatkan pada Gambar 2. Setiap titik (*dot*) yang muncul di *frame* menandakan bahwa objek (manusia) telah terdeteksi di dalam *frame* pada saat *sampling*, kemudian setiap titik deteksi tersebut

akan bergerak mengikuti objek dan diberi sebuah tanda, yaitu berupa ID objek yang digunakan sebagai pengenal bahwa objek tersebut sudah terdeteksi sebelumnya. Saat proses pelacakan dimulai, maka titik (*dot*) *output* dari deteksi akan mengikuti pergerakan objek hingga objek menghilang atau keluar dari *frame* kamera. Jika objek yang dilacak menghilang setelah melewati beberapa *frame* yang telah ditentukan, maka objek dihapus dari daftar pelacakan serta *output* titik (*dot*) pada *frame* akan dihilangkan atau dihapus.



Gambar 3. Rancangan alur sistem

Proses penambahan jumlah pengunjung yang masuk maupun keluar dilakukan dengan melakukan penghitungan nilai koordinat pelacakan menggunakan sumbu Y pada *frame* sebagai dasar penghitungan (semakin ke atas semakin besar dan semakin ke bawah semakin kecil). Penghitungan dilakukan setelah objek melewati garis tengah pada *frame*, hal ini dilakukan untuk menghindari kesalahan penghitungan jika terdapat objek yang bergerak kembali ke arah sebaliknya sebelum melewati garis tengah (pengunjung tidak jadi masuk atau keluar). Jika sudah melewati garis tengah pada *frame* maka penghitungan akan dimulai dengan mengurangi koordinat terbaru dengan koordinat lama sehingga diperoleh nilai baru di mana jika nilai tersebut negatif (kurang dari nilai yang telah ditentukan) maka akan diasumsikan objek bergerak ke atas dan akan dihitung sebagai penambahan pada objek masuk. Sebaliknya, jika nilai baru yang didapat adalah nilai positif (lebih dari nilai yang telah ditetapkan) maka akan diasumsikan objek tersebut bergerak ke bawah dan akan dihitung sebagai penambahan pada objek keluar.

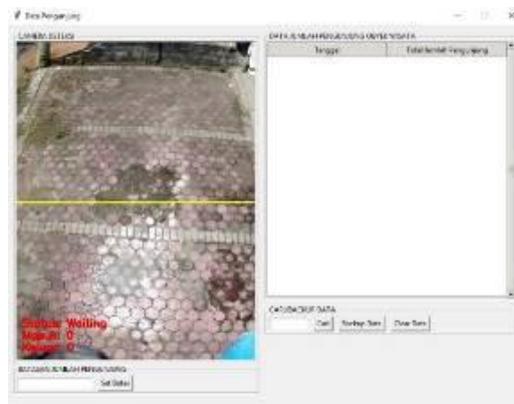
Dengan menggunakan metode ini maka secara otomatis sistem hanya akan mendeteksi objek yang diinginkan. Dalam penelitian ini objek tersebut adalah manusia sebagai pengunjung objek wisata. Jika ada objek lain selain manusia yang tertangkap pada kamera atau *webcam*, maka akan diabaikan dan tidak akan dihitung sebagai pengunjung objek wisata. Gambar 3 memperlihatkan rancangan diagram alir sistem pendeteksi pengunjung objek wisata menggunakan metode *MobileNet-SSD*.

3. Hasil dan Pembahasan

Pada dasarnya pengujian yang dilakukan pada penelitian ini sesuai dengan rancangan alur sistem yang telah dijelaskan pada bab sebelumnya, yaitu penghitungan jumlah pengunjung yang masuk dan keluar pada objek wisata, kemudian pengujian terhadap ketepatan deteksi objek (pengunjung) serta pengujian pada penyimpanan data jumlah pengunjung dan pembatasan jumlah pengunjung sesuai dengan nilai yang ditentukan sebelumnya.

3.1. Tampilan Antarmuka

Gambar 4 merupakan tampilan antarmuka sistem pendeteksi dan penghitung pengunjung objek wisata berbasis *MobileNet-SSD* dengan menggunakan pemrograman *Python*. Garis tengah berwarna kuning yang ada pada *frame* digunakan sebagai tanda bahwa objek manusia yang terdeteksi telah melewati pintu keluar atau masuk.



Gambar 4. Tampilan antarmuka sistem pendeteksian dan penghitungan pengunjung objek wisata

3.2. Pendeteksian dan Penghitungan

Untuk melakukan proses pendeteksian dan pelacakan, pada sistem ditambahkan *object tracker* sebagai bagian dari proses deteksi yang berguna untuk meringankan beban komputasi sistem pendeteksian objek. Penambahan jumlah pengunjung pada sistem ditentukan dengan mengikuti pergerakan dari pengunjung (objek) yang terdeteksi. Jika pengunjung bergerak ke arah atas dan telah melewati garis batas tengah pada *frame* maka akan dihitung sebagai pengunjung yang keluar, sedangkan jika pengunjung bergerak ke arah bawah dan telah melewati garis batas tengah pada *frame* maka akan dihitung sebagai pengunjung masuk. Hasil pendeteksian dan pelacakan objek berupa manusia diperlihatkan pada Gambar 4 berikut.

Gambar 4 memperlihatkan bahwa pendeteksian dan pelacakan objek dapat dilakukan oleh sistem. Hal ini ditunjukkan dengan adanya pemberian *ID* pada *frame* oleh angka berwarna hijau.



Gambar 4. Hasil pendeteksian dan pelacakan objek

3.3. Pengujian Sistem

Pada pengujian ini *webcam* diletakkan pada ketinggian 5 meter dengan sudut ke bawah 30° . Untuk menguji ketepatan deteksi objek pada sistem, peneliti telah melakukan percobaan atau uji coba sistem dengan objek pengunjung secara individu dan kelompok. Tabel 1 menunjukkan hasil pengujian sistem di dalam ruangan dengan kondisi pengunjung individu.

Tabel 1. Pengujian dalam ruangan dengan pengunjung individu

No	Citra	Keterangan
1		Satu orang pengunjung masuk dan terdeteksi satu objek dengan ID 0. Tetapi belum terhitung masuk karena belum melewati garis kuning di tengah <i>frame</i>
2		Satu orang pengunjung masuk dan terdeteksi satu objek dengan ID 0. Pada sistem sudah terhitung masuk karena sudah melewati garis kuning di tengah <i>frame</i>
3		Satu orang pengunjung keluar dan terdeteksi satu objek dengan ID 1. Tetapi belum terhitung keluar karena belum melewati garis kuning di tengah <i>frame</i>
4		Satu orang pengunjung keluar dan terdeteksi satu objek dengan ID 1. Pada sistem sudah terhitung keluar karena sudah melewati garis kuning di tengah <i>frame</i>

Dari data yang ditunjukkan pada Tabel 1 terlihat bahwa sistem dapat mendeteksi dan menghitung objek pengunjung yang keluar dan masuk pada kondisi ruangan. Hasil pendeteksian objek pengunjung dengan memberikan pelabelan berupa *ID*. Penghitungan objek pengunjung dilakukan setelah pengunjung tersebut melewati garis kuning.

Tabel 2. Pengujian dengan pengunjung berkelompok

No	Citra	Keterangan
1		Tiga orang pengunjung masuk dan terdeteksi tiga objek dengan <i>ID</i> 4, 5 dan 6. Tetapi belum terhitung masuk karena belum melewati garis kuning di tengah <i>frame</i>
2		Tiga orang pengunjung masuk dan terdeteksi tiga objek dengan <i>ID</i> 4, 5 dan 6. Pada sistem sudah terhitung masuk karena sudah melewati garis kuning di tengah <i>frame</i>
3		Dua orang pengunjung masuk dan terdeteksi satu objek dengan <i>ID</i> 10 karena posisi kedua pengunjung tersebut menempel.
4		Empat orang pengunjung masuk dan terdeteksi empat objek dengan <i>ID</i> 7, 8, 9, dan 10. Pada sistem belum terhitung keluar karena belum melewati garis kuning di tengah <i>frame</i>
5		Empat orang pengunjung masuk dan terdeteksi empat objek dengan <i>ID</i> 7, 8, 9 dan 10. Pada sistem sudah terhitung keluar karena sudah melewati garis kuning di tengah <i>frame</i>

Untuk pengujian yang dilakukan dengan pengunjung secara berkelompok disajikan pada Tabel 2. Dari data yang ditunjukkan pada Tabel 2 terlihat bahwa sistem dapat mendeteksi dan menghitung objek pengunjung yang keluar dan masuk pada kondisi ruangan. Namun, terdapat *error* saat melakukan pendeteksian seperti yang ditunjukkan pada pengujian nomor 3. Hal tersebut terjadi karena posisi pengunjung 1 menempel dengan pengunjung lainnya sehingga terdeteksi sebagai satu objek.

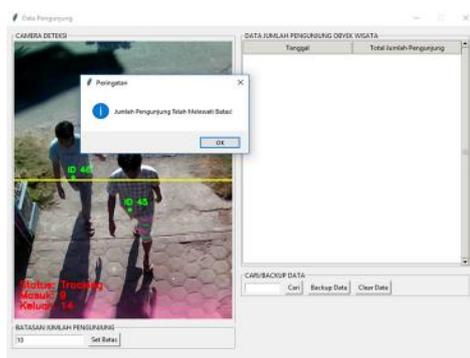
3.3. Pengujian Pembatasan Jumlah Pengunjung

Pembatasan jumlah pengunjung pada sistem ini dilakukan dengan memasukkan angka batasan pengunjung yang ditetapkan pada kotak isian (*text box*) yang terdapat pada *graphical user interface* (GUI) sistem di bagian "BATASAN JUMLAH PENGUNJUNG" seperti yang diperlihatkan pada Gambar 5.



Gambar 5. Text Box batasan jumlah pengunjung

Dari hasil pengujian, jika pengunjung yang telah memasuki objek wisata berjumlah sama dengan angka yang telah ditetapkan sebagai batasan maka akan muncul sebuah *output* berupa notifikasi peringatan bahwa jumlah pengunjung telah melewati batas seperti yang diperlihatkan pada Gambar 6.



Gambar 5. Hasil pengujian pembatasan jumlah pengunjung

4. Kesimpulan

Berdasarkan hasil pengujian terhadap implementasi penghitung jumlah pengunjung objek wisata dengan metode *deep learning MobileNet-SSD* menggunakan *OpenCV*, maka dapat disimpulkan bahwa *MobileNet-SSD* dapat diterapkan sebagai pendeteksi dan penghitung jumlah pengunjung objek wisata. Salah satu kelemahan pada sistem pendeteksian menggunakan metode *deep learning MobileNet-SSD* adalah jika terdapat dua objek yang menempel, maka salah satu objek tidak terdeteksi atau kedua objek tersebut dianggap sebagai satu objek, sehingga hanya menghasilkan satu objek terdeteksi.

Daftar Pustaka

- [1] D. Erwanto, Y. Bismo Utomo, F. Alif Fiolana, M. Yahya, "Pengolahan citra digital untuk menentukan kadar asam askorbat pada buah dengan metode titrasi

- iodimetri," *Multitek Indones. J. Ilm.*, vol. 12, no. 2, pp. 73–84, 2018, [Online]. Available: <http://journal.umpo.ac.id/index.php/multitek>.
- [2] S. Lande, R. Lim, K. Gunadi, *et al.*, "Program penghitung jumlah orang lewat menggunakan webcam," *J. Inform.*, vol. 5, no. 2, pp. 121–126, 2004.
- [3] L. B. Setyawan, F. D. Setiaji, *et al.*, "Aplikasi webcam untuk menjejak pergerakan manusia di dalam ruangan," *Techné J. Ilm. Elektrotek.*, vol. 12, no. 01, pp. 51–60, 2013.
- [4] E. Nilamsari, "Penghitung jumlah orang lewat dengan metode normalized sum-squared differences (NSSD)," 2008.
- [5] F. Sindy, "Pendeteksian objek manusia secara real time dengan metode Mobilenet-SSD menggunakan movidius neural compute stick pada raspberry PI," 2019.
- [6] W. I. Kusumawati, H. Pratikno, Y. P. Admaja, "Sistem penghitung jumlah pengunjung restoran menggunakan kamera berbasis single shot detector (SSD)," *J. Technol. Informatics*, vol. 3, no. 1, pp. 19–26, 2021.
- [7] A. Younis, L. Shixin, S. Jn, Z. Hai, "Real-time object detection using pre-trained deep learning models MobileNet-SSD," in *Proceedings of 2020 the 6th International Conference on Computing and Data Engineering*, 2020, pp. 44–48.
- [8] W. Liu *et al.*, "Ssd: Single shot multibox detector," in *European conference on computer vision*, 2016, pp. 21–37.
- [9] A. G. Howard *et al.*, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv Prepr. arXiv1704.04861*, 2017.
- [10] I. B. Pakpahan, I. C. Dewi, "Pendeteksian lubang pada jalanan menggunakan metode SSD-Mobilenet," *Indonesian J. Electron. Instrum. Syst.*, vol. 11, no. 2.
- [11] S. Ghoury, C. Sungur, A. Durdu, "Real-time diseases detection of grape and grape leaves using faster R-CNN and SSD mobilenet architectures," in *International conference on advanced technologies, computer engineering and science (ICATCES 2019)*, 2019, pp. 39–44.
- [12] F. Arifin, H. Artanto, T. S. G. Nurhasanah, *et al.*, "Fast COVID-19 detection of chest X-ray images using single shot detection mobilenet convolutional neural networks," *J. Southwest Jiaotong Univ.*, vol. 56, no. 2, 2021.

