JITE (Journal of Informatics and Telecommunication Engineering)

Available online <http://ojs.uma.ac.id/index.php/jite> DOI : 10.31289/jite.v5i2.6118

Received: 15 November 2021

Accepted: 21 December 2021

Published: 26 January 2022

OpenCV Using on a Single Board Computer for Incorrect Facemask-Wearing Detection and Capturing

Julham1)*, Meryatul Husna1), Arif Ridho Lubis2)

1) Program Studi Teknologi Rekayasa Multimedia Grafis, Politeknik Negeri Medan, Indonesia

2) Program Studi Manajemen Informatika, Politeknik Negeri Medan, Indonesia

*Corresponding Email: julham@polmed.ac.id

Abstrak

OpenCV (Open Source Computer Vision Library) adalah sebuah pustaka perangkat lunak yang ditujukan untuk pengolahan citra dinamis secara real-time, yang dibuat oleh Intel. Dalam penelitian ini, pustaka tersebut akan digunakan untuk mendeteksi wajah, hidung dan mulut. Selanjutnya sistem dibekali pengetahuan bahwa jika mulut dan hidung atau salah satunya terdeteksi, maka wajah tersebut belum menggunakan masker dengan benar dan sistem merekam wajah tersebut. Sistem didukung oleh perangkat penangkap gambar berupa camcorder, perangkat pengolah berupa single board computer, seperti Raspberry Pi dan perangkat penampil berupa monitor. Dan hasilnya adalah sistem mampu memberi keputusan apakah wajah tersebut menggunakan masker dengan benar atau tidak. Dengan cara memberi label disekitar wajah berupa garis bersudut warna merah, jika belum benar menggunakan maskernya. Tingkat keberhasilannya adalah 88,4% dengan menggunakan parameter detektor yakni : faktor skala = 1,1 untuk semua pustaka objek wajah, hidung dan mulut.

Kata Kunci: skala, faktor, detektor, camcorder, objek.

Abstract

OpenCV (Open Source Computer Vision Library) is a software library intended for real-time dynamic image processing, created by Intel. In this study, the library will be used to detect the face, nose and mouth. Furthermore, the system is equipped with the knowledge that if the mouth and nose or one of them is detected, then the face has not used the mask correctly and the system records the face. The system is supported by an image capture device in the form of a camcorder, a processing device in the form of a single board computer, such as a Raspberry Pi and a display device in the form of a monitor. And the result is that the system is able to make a decision whether the face is wearing a mask correctly or not. By means of labeling around the face in the form of red angular lines, if not properly use the mask. The success rate is 88.4% using detector parameters, namely: scale factor = 1.1 for all face, nose and mouth object libraries.

Keywords: scale, factor, detector, camcorder, object.

How to Cite : Julham, Husna, M., & Lubis, A. R. (2021). OpenCV Using on a Single Board Computer for Incorrect Facemask-Wearing Detection and Capturing. *JITE (Journal of Informatics and Telecommunication Engineering)*, 5(2), 315–323.

I. PENDAHULUAN

Penggunaan masker penutup hidung dan mulut sekarang ini merupakan hal yang harus dilakukan dalam berkegiatan di luar rumah. Hal ini menjadi budaya baru yang harus dipatuhi seluruh masyarakat dan bahkan warga dunia untuk menekan penularan COVID-19 (Riana, 2021). Peran pemerintah dalam menjaga budaya ini cukup signifikan. Salah satu contohnya adalah pemerintah melalui media-media menggencarkan kegiatan 3M (memakai masker, mencuci tangan dan menjaga jarak) (Wiryawan, 2020). Pemakaian masker yang benar dapat membantu menekan penularannya. Untuk itu masker harus dapat menutupi hidung dan mulut. Tentunya pengawasan penggunaan masker yang benar dibutuhkan sekali. Salah satu alternatifnya adalah menggunakan sistem kamera pintar yang dapat membedakan mana yang sesuai dan tidak sesuai.

Hal tersebut mendorong munculnya berbagai macam pendekatan mengenai hal tersebut, misalnya memanfaatkan deep learning untuk mendeteksi apakah menggunakan masker atau tidak (Susanto, 2020) (Lambacing, 2020) (Harfi, 2020), dengan memanfaatkan neural network (A. R. Lubis, 2021) dalam deteksi masker (Budiman, 2021) (Putri, 2020) (Singh, 2021). Tulisan-tulisan tersebut bertujuan sama yakni untuk mengoptimalkan sistem dengan pelatihan (train) terhadap gambar wajah yang memakai masker dan tidak memakai masker dengan metode tertentu, sehingga menghasilkan dataset yang diharapkan. Sedangkan pada penelitian ini juga memiliki tujuan yang sama tetapi menggunakan estimasi keberadaan mulut dan hidung berdasarkan dataset yang tersedia di OpenCV, sehingga membantu sistem dalam memutuskan apakah wajah tersebut menggunakan masker atau tidak.

II. METODE PENELITIAN

Dataset wajah, hidung dan mulut dalam tulisan ini menggunakan dataset yang sudah tersedia di pustaka OpenCV (Chau, 2019) (Heryana, 2020) (Julham., 2021). Ketiga dataset ini akan dijalankan secara sekaligus dalam bahasa pemrograman Processing dengan menggunakan metode research and development (Sugiyono, 2017). Berikut gambar 1 flowchart sistemnya

Gambar 1. Desain penelitian

Adapun penjelasan dari gambar 1 adalah sebagai berikut

1. Diawali dengan memuat dataset-dataset yang diperlukan yakni: dataset wajah yang dikenal dengan OpenCV.CASCADE_FRONTALFACE pada pustaka OpenCV, dataset hidung yang dikenal dengan OpenCV.CASCADE_NOSE pada pustaka OpenCV, dataset mulut yang dikenal dengan OpenCV.CASCADE_MOUTH pustaka OpenCV.
2. Selanjutnya dataset wajah diolah dan menghasilkan dua pilihan kondisi. Kondisi pertama, objek terdeteksi yang ditandai dengan bentuk segiempat dan kondisi kedua, objek tidak terdeteksi.
3. Jika objek wajah terdeteksi, maka dilanjutkan dengan mengolah dataset hidung dan menghasilkan dua pilihan kondisi. Kondisi pertama, objek terdeteksi yang ditandai dengan garis berwarna biru dan kondisi kedua, objek tidak terdeteksi.
4. Jika objek hidung tidak terdeteksi, maka dilanjutkan dengan mengolah dataset mulut dan menghasilkan dua pilihan kondisi. Kondisi pertama, objek terdeteksi yang ditandai dengan garis berwarna hijau dan kondisi kedua, objek tidak terdeteksi.
5. Jika objek mulut tidak terdeteksi, maka sistem tidak menampilkan bentuk garis apapun, yang berarti bahwa asumsi wajah tersebut menggunakan masker dan sistem tidak menyimpan wajah tersebut.

- 6 . Sedangkan jika hanya objek mulut yang terdeteksi, atau hanya objek hidung terdeteksi atau kedua-duanya terdeteksi, maka menampilkan bentuk segiempat dengan garis berwarna merah disekeliling wajah yang diamati. Hal ini akan memicu sistem untuk menyimpan wajah tersebut ke komputer.

III. HASIL DAN PEMBAHASAN

Sistem deteksi objek wajah, mulut dan hidung dalam penelitian ini berdasarkan pada dataset yang sudah tersedia di pustaka OpenCV . Ketiga dataset tersebut dimuat sekaligus dengan menggunakan variabel yang berbeda-beda dalam bahasa pemrograman Processing. Selain itu, perangkat camcorder seperti pada gambar 2 digunakan untuk menangkap objek wajah dan ini merupakan masukan sistemnya.

Gambar 2. Perangkat penangkap objek

Input atau masukan sistem ini memiliki kemampuan menangkap citra objek yang dapat diatur optical zoom-nya, sehingga data objek dapat diproses. Jenis camcorder ini diatur dengan mode rasio capture 4:3 dan terminal keluaran RCA, serta tambahan perangkat converter AV to USB agar dapat tersambung dengan modul prosesnya. Diagram blok sistem perangkat kerasnya dapat dilihat pada gambar 3

Gambar 3. Diagram blok proses

Modul Proses adalah perangkat keras berupa single board computer, dalam hal ini menggunakan Raspberry Pi 3 Model B. Modul penampil sebagai output atau keluaran sistem adalah perangkat keras sistem yang digunakan untuk melihat hasil respon dari perangkat proses yang berupa layar penampil, seperti monitor.

Dalam hal mendukung kerja perangkat keras tersebut, maka perangkat lunak yang digunakan adalah bahasa Processing dengan mode Java. Bahasa ini dapat berjalan di sistem operasi Windows, sistem Operasi Linux dan sistem operasi MacOS. Selanjutnya dibutuhkan plugin tambahan yakni OpenCV. Plugin tersebut digunakan untuk memudahkan pemrograman deteksi objek dengan mengakses pustaka yang disediakan. Seperti untuk mendeteksi wajah, maka menggunakan dataset wajah yang dikenal dengan OpenCV.CASCADE_FRONTALFACE, dengan menggunakan perintah dasar: `opencv.cascade(OpenCV.CASCADE_FRONTALFACE );`. Pada pustaka tersebut akan muncul garis persegi panjang jika dianggap olehnya sebagai objek wajah, seperti pada gambar 4.

Gambar 4. Ilustrasi objek wajah yang ditandai

Munculnya garis seperti pada gambar 4, nantinya akan digunakan untuk membantu memastikan ada atau tidak objek hidung dan mulut. Selanjutnya dengan bantuan dataset hidung dan mulut yang tersedia di pustaka OpenCV, maka ditelusurilah objek hidung dan mulut. Ruang lingkup yang ditelusuri adalah ruang lingkup yang dibatasi oleh garis persegi panjang hasil deteksi objek wajah tersebut. Berikut gambar 5, ilustrasi posisi objek dari ruang lingkup yang ditelusuri.

Gambar 5. Ilustrasi pembagian posisi objek

Gambar 5. merupakan ilustrasi yang ditempelkan ke tampilan monitor komputer dengan bahasa pemrograman Processing, dimana garis x (horisontal) menyatakan lebar layar dan garis y (vertikal) menyatakan tinggi layar. Jika dimisalkan Y1 adalah batas wajah atas, Y2 adalah batas wajah bawah, X1 adalah batas wajah kiri dan X2 adalah batas wajah kanan, maka pernyataan bentuk kalimat matematika pertidaksamaan linear dua variabel tersebut adalah sebagai berikut:

$$Y \geq Y1 \quad (1)$$

$$Y \leq Y2 \quad (2)$$

$$X \geq X1 \quad (3)$$

$$X \leq X2 \quad (4)$$

Setelah didapat daerah atau wilayah wajah, maka dilanjutkan dengan menentukan daerah atau wilayah hidung. Untuk menentukannya, maka diasumsikan sebagai berikut:

1. Diawali dengan membagi menjadi 6 bagian daerah atau wilayah tersebut (garis-garis putus Ya, Yb, Yc, Yd, Ye, seperti pada gambar 5).

2. Lalu ditandai dengan menggunakan garis horisontal yang transparan.

3. Daerah atau wilayah hidung yang akan dideteksi oleh sistem dinyatakan dalam kalimat matematika pertidaksamaan linear dua variabel sebagai berikut:

$$X \geq X1 \quad (5)$$

$$X \leq X2 \quad (6)$$

$$Y \geq Yb \quad (7)$$

$$Y \leq Yd \quad (8)$$

$$\text{dimana, } Yb = Y1 + 2 \cdot (Y2 - Y1) / 6 \text{ dan } Yd = Y1 + 4 \cdot (Y2 - Y1) / 6$$

Dengan cara yang sama, daerah atau wilayah mulut yang akan dideteksi oleh sistem dinyatakan dalam kalimat matematika pertidaksamaan linear dua variabel sebagai berikut:

$$X \geq X1 \quad (9)$$

$$X \leq X2 \quad (10)$$

$$Y \geq Yd \quad (11)$$

$$Y \leq Ye \quad (12)$$

$$\text{dimana, } Yd = Y1 + 4 \cdot (Y2 - Y1) / 6 \text{ dan } Ye = Y1 + 5 \cdot (Y2 - Y1) / 6$$

A. Pengaturan Parameter Detektor

Penggunaan pustaka OpenCV untuk deteksi objek dalam tulisan ini memerlukan pengaturan-pengaturan parameternya. Hal ini dikarenakan ketiga pustaka OpenCV tersebut dimuat dan berproses secara bersamaan. Untuk itu diawali dengan mengetahui parameter yang digunakan dalam tulisan ini, yakni

1. Haar Cascade XML, yakni kumpulan data training (dataset) yang dihadirkan untuk fungsi Haar Classifier. Classifier memakai data yang disimpan pada file XML untuk memutuskan bagaimana mengklasifikasi tiap lokasi gambar/objek. Ada banyak file XML yang disediakan di OpenCV, sehingga penggunaanya tergantung dari kebutuhan.
2. Faktor skala, yakni peningkatan skala rata-rata kelulusan pendeteksian objek yang diinginkan pada lebih dari sebuah objek tersebut. Pengaturan yang lebih tinggi membuat detektor berjalan lebih cepat, tapi jika terlalu tinggi, maka dapat melompat terlalu cepat antara skala dan banyak objek tersebut yang terlewat.
3. Flag, yaitu mode operasi yang digunakan. Untuk OpenCV versi 1.1 dan versi sebelumnya, mode operasi yang ada hanya satu, yaitu CV_HAAR_DO_CANNY_PRUNING. Sedangkan untuk versi yang lebih baru, terdapat tiga mode tambahan, yaitu CV_HAAR_SCALE_IMAGE, CV_HAAR_FIND_BIGGEST_OBJECT, dan CV_HAAR_DO_ROUGH_SEARCH.

Tujuan dari parameter-parameter tersebut adalah agar sistem dapat bekerja secara optimal. Optimal maksudnya adalah bahwa sistem mampu memberi respon cepat dan sistem mampu menghasilkan kesalahan yang kecil dalam mendeteksi objek yang dimaksud.

B. Implementasi Sistem

Sistem yang diimplementasikan berupa perangkat keras dan perangkat lunak. Secara blok diagram perangkat kerasnya seperti pada gambar 3. Dan implementasinya seperti pada gambar 6.

Gambar 6. Implementasi dengan resolusi hasil olahan 800x600 pixel

Adapun penjelasan dari gambar 6 adalah sebagai berikut:

1. Sistem diawali dengan menangkap gambar objek yang dalam tulisan ini menggunakan perangkat camcorder seperti yang terlihat pada huruf "A" pada gambar 6. Rasio layar tangkapnya menggunakan 4:3 dengan pengaturan pencahayaan secara otomatis. Keluaran dari camcorder ditujukan ke perangkat proses menggunakan model RCA atau dikenal dengan AV (kabel gambar dan suara terpisah).
2. Hasil tangkapan oleh perangkat camcorder diteruskan ke perangkat proses yakni berupa single board computer, dalam hal ini menggunakan Raspberry Pi 3 Model B (seperti yang terlihat di huruf "B" pada gambar 6). Lalu perangkat tersebut dipasangkan aplikasi bahasa pemrograman Processing dengan mode Java dan pustaka OpenCV yang cocok dengan sistem operasi Linux. Di sisi lain, diperlukan juga keyboard dan monitor untuk mendukung kerja dan pengambilan datanya.

3. Untuk melihat hasil respon dari perangkat proses (Raspberry Pi 3 Model B), maka dalam tulisan ini menggunakan monitor LCD ukuran 7 inchi seperti yang terlihat di huruf "C" pada gambar 6.

Selain dari ketiga perangkat tersebut, ada juga perangkat pendukung lainnya yakni tripod. Perangkat ini digunakan untuk meletakkan semua perangkat penelitian (point A, B dan C). Dengan bantuan tripod, pengarahannya untuk menangkap gambar objek sebagai bahan pengujian penelitian ini dapat dengan mudah dilakukan.

Selanjutnya sistem akan didukung oleh perangkat lunak pemrograman Processing dengan mengacu pada flowchart seperti gambar 1. Berikut urutan ringkasan dari program yang diimplementasikan:

1. Diawali dengan mengidentifikasi parameter resolusi dari layar tampil yang digunakan dan memuat pustaka untuk objek wajah, hidung dan mulut, seperti pada gambar 7.

```
size(640,480);
video = new Capture(this, 640, 480);
opencv1 = new OpenCV(this, video.width/skala, video.height/skala);
opencv2 = new OpenCV(this, video.width/skala, video.height/skala);
opencv3 = new OpenCV(this, video.width/skala, video.height/skala);
opencv1.loadCascade(OpenCV.CASCADE_FRONTALFACE);
opencv2.loadCascade(OpenCV.CASCADE_MOUTH);
opencv3.loadCascade(OpenCV.CASCADE_NOSE);
```

Gambar 7. Listing program memuat pustaka objek yang diperlukan

Pada penelitian ini nilai resolusi layar akan diubah-ubah dengan cara mengubah nilai dari size dan video = new Capture yang terdapat pada gambar 7.

2. Selanjutnya listing program untuk mengatur detektor parameternya seperti pada gambar 8

```
Rectangle[] faces1 = opencv1.detect(1.1, 3, 1, 0, 0);
Rectangle[] faces2 = opencv2.detect(1.1, 3, 1, 0, 0);
Rectangle[] faces3 = opencv3.detect(1.1, 3, 1, 0, 0);
```

Gambar 8. Listing program pengatur parameter detektor

Adapun penjelasannya adalah bahwa objek yang diwakili oleh variabel faces1 (untuk wajah), faces2 (untuk mulut) dan faces3 (untuk hidung) akan membentuk garis persegi panjang (diistilahkan dengan Rectangle[]) jika mendeteksi objek yang sesuai dengan pustaka masing-masing. Untuk dapat mendeteksi objek tersebut diperlukan pengaturan detektor parameter dengan bentuk umum berikut: Opencv.detect (double scaleFactor, int minNeighbors, int flags, int minSize, int maxSize), dimana nilai dari scaleFactor dalam penelitian ini akan diubah-ubah, sedangkan nilai yang lainnya tetap, yakni untuk minNeighbors = 3, flags = 1, minSize = 0, maxSize = 0.

3. Untuk objek wajah yang terdeteksi tidak benar memakai masker, maka ditangani oleh listing program seperti gambar 9.

```
PImage cropped = createImage(faces1[i].width*skala,
faces1[i].height*skala, RGB);|
cropped.copy(video, faces1[i].x*skala, faces1[i].y*skala,
faces1[i].width*skala, faces1[i].height*skala, 0, 0,
faces1[i].width*skala, faces1[i].height*skala);
cropped.updatePixels();
cropped.save("data/face-"+hour()+minute()+second()+".jpg");
```

Gambar 9. Listing program untuk menyimpan objek wajah

Listing program seperti pada gambar 9 akan bekerja ketika objek wajah yang terdeteksi, terlihat hidungnya atau mulutnya berdasarkan dataset-dataset yang ada.

C. Pengambilan dan Pembahasan Data

Untuk pengambilan datanya, diperlukan parameter faktor skala (variabel scaleFactor) dan parameter resolusi tampilan hasil olahan sistem (seperti yang terlihat pada sebelah kanan dari gambar 6). Adapun nilai faktor skala yang digunakan adalah 1,1; 1,3; 1,5; 1,7; 1,9 dan 2,1. Sedangkan nilai resolusi yang digunakan bergantung dari camcordernya, yakni mode rasio capture 4:3 yang menurut standar layar atau standar tampilan (https://id.wikipedia.org/wiki/Resolusi_layar) adalah sebagai berikut:

1. Standar VGA dengan resolusi width = 640 pixel dan height = 480 pixel (640x480)
2. Standar SVGA dengan resolusi width = 800 pixel dan height = 600 pixel (800x600)
3. Standar XGA dengan resolusi width = 1024 pixel dan height = 768 pixel (1024x768)

Selanjutnya objek pengamatan berupa wajah dengan masker kain yang tidak benar penggunaannya (seperti pada gambar 10) dan untuk objek wajah dengan masker bedah.

Gambar 10. Contoh hasil olahan dengan resolusi 1024x 768 pixel

Data pengamatan yang disajikan dalam tabel berupa bobot nilai yang memiliki arti sebagai berikut:

1. Pernyataan tidak ada, diberikan nilai 0
2. Pernyataan terkadang ada, diberikan nilai 0,5
3. Pernyataan ada, diberikan nilai 1

Tabel 1. Bobot nilai pengamatan objek wajah dengan masker kain menggunakan resolusi 1024x768

Faktor skala	Kemunculan warna pada objek wajah sebelah kiri			Kemunculan warna pada objek wajah sebelah kanan			Total bobot nilai
	merah	biru	hijau	merah	biru	hijau	
1,1	1	1	0	1	1	1	5
1,3	0	0	0	0,5	0,5	0,5	1,5
1,5	0	0	0	0,5	0	0,5	1
1,7	0	0	0	0	0	0	0
1,9	0	0	0	0	0	0	0
2,1	0	0	0	0	0	0	0

Tabel 2. Pengamatan objek wajah dengan masker kain menggunakan resolusi 800x600

Faktor skala	Kemunculan warna pada objek wajah sebelah kiri			Kemunculan warna pada objek wajah sebelah kanan			Total bobot nilai
	merah	biru	hijau	merah	biru	hijau	
1,1	1	1	0	1	1	1	5
1,3	0	0	0	1	0	1	2
1,5	0	0	0	0,5	0	0,5	1
1,7	0	0	0	0	0	0	0
1,9	0	0	0	0	0	0	0
2,1	0	0	0	0	0	0	0

Tabel 3. Pengamatan objek wajah dengan masker kain menggunakan resolusi 600x480

Faktor skala	Kemunculan warna pada objek wajah sebelah kiri			Kemunculan warna pada objek wajah sebelah kanan			Total bobot nilai
	merah	biru	hijau	merah	biru	hijau	
1,1	0	0	0	1	0	1	2
1,3	0	0	0	1	0	1	2
1,5	0	0	0	0	0	0	0
1,7	0	0	0	0	0	0	0

1,9	0	0	0	0	0	0	0
2,1	0	0	0	0	0	0	0

Tabel 4. Pengamatan objek wajah dengan masker bedah menggunakan resolusi 1024x768

Faktor skala	Kemunculan warna pada objek wajah sebelah kiri			Kemunculan warna pada objek wajah sebelah kanan			Total bobot nilai
	merah	biru	hijau	merah	biru	hijau	
1,1	1	1	1	1	1	0	5
1,3	1	1	1	0,5	0,5	0	4
1,5	1	0,5	1	0,5	0,5	0	3,5
1,7	1	1	0,5	0,5	0,5	0	3,5
1,9	0,5	0,5	0,5	0,5	0,5	0	2,5
2,1	0,5	0,5	0,5	0,5	0,5	0	2,5

Tabel 5. Pengamatan objek wajah dengan masker bedah menggunakan resolusi 800x600

Faktor skala	Kemunculan warna pada objek wajah sebelah kiri			Kemunculan warna pada objek wajah sebelah kanan			Total bobot nilai
	merah	biru	hijau	merah	biru	hijau	
1,1	1	1	1	1	1	0	5
1,3	1	1	1	0,5	0,5	0	4
1,5	1	0,5	1	0,5	0,5	0	3,5
1,7	1	1	0,5	0,5	0,5	0	3,5
1,9	0,5	0,5	0,5	0,5	0,5	0	2,5
2,1	1	0,5	1	0	0	0	2,5

Tabel 6. Pengamatan objek wajah dengan masker bedah menggunakan resolusi 600x480

Faktor skala	Kemunculan warna pada objek wajah sebelah kiri			Kemunculan warna pada objek wajah sebelah kanan			Total bobot nilai
	merah	biru	hijau	merah	biru	hijau	
1,1	1	1	0,5	1	1	0	4,5
1,3	1	1	1	0,5	0,5	0	4
1,5	1	0,5	1	0,5	0,5	0	3,5
1,7	1	1	0,5	0,5	0,5	0	3,5
1,9	0,5	0,5	0,5	0,5	0,5	0	2,5
2,1	0	0	0	0	0	0	0

Selanjutnya didapat keseluruhan total bobot nilai dari masing-masing faktor skala adalah sebagai berikut:

1. Faktor skala 1,1 dengan keseluruhan total bobot nilai adalah $5 + 5 + 2 + 5 + 5 + 4,5 = 26,5$. Sehingga rata-rata bobot nilai adalah $26,5/6 = 4,42$.
2. Faktor skala 1,3 dengan nilai keseluruhan total bobot nilai adalah $1,5 + 2 + 2 + 4 + 4 + 4 = 17,5$. Sehingga rata-rata bobot nilai adalah $17,5/6 = 2,92$.
3. Faktor skala 1,5 dengan keseluruhan total bobot nilai adalah $1 + 1 + 0 + 3,5 + 3,5 + 3,5 = 12,5$. Sehingga rata-rata bobot nilai adalah $12,5/6 = 2,08$.
4. Faktor skala 1,7 dengan keseluruhan total bobot nilai adalah $0 + 0 + 0 + 3,5 + 3,5 + 3,5 = 10,5$. Sehingga rata-rata bobot nilai adalah $10,5/6 = 1,75$.
5. Faktor skala 1,9 dengan keseluruhan total bobot nilai adalah $0 + 0 + 0 + 2,5 + 2,5 + 2,5 = 7,5$. Sehingga rata-rata bobot nilai adalah $7,5/6 = 1,25$.
6. Faktor skala 2,1 dengan keseluruhan total bobot nilai adalah $0 + 0 + 0 + 2,5 + 2,5 + 0 = 5$. Sehingga rata-rata bobot nilai adalah $5/6 = 0,83$.

Total bobot nilai ideal dari gambar 10 adalah 5. Dikarenakan sistem seharusnya dapat mendeteksi objek hidung sebelah kiri, objek wajah sebelah kiri, objek hidung sebelah kanan, objek mulut sebelah kanan, dan objek wajah sebelah kanan. Sehingga faktor skala yang mendekati bobot nilai ideal adalah 1,1 dengan tingkat keberhasilan $(4,42/5) \times 100\% = 88,4\%$.

IV. SIMPULAN

Dari hasil penelitian didapat bahwa nilai faktor skala yang mendekati bobot nilai ideal adalah 1,1 dengan tingkat keberhasilan $(4,42/5) \times 100\% = 88,4\%$. Setiap resolusi memberikan pengaruh terhadap kemampuan sistem dalam mendeteksi objek. Begitu juga dengan penggunaan nilai faktor skalanya, yang mana semakin tinggi nilai faktor skala, maka semakin rendah kemampuan sistem dalam mendeteksi objek.

V. UCAPAN TERIMA KASIH

Para penulis dengan penuh rasa syukur menyampaikan penghargaan dan mengucapkan terima kasih atas dukungan finansial yang diberikan melalui dana DIPA Politeknik Negeri Medan tahun 2021, dengan nomor kontrak: B/273/PL.5/PT.01.05/2021 tertanggal 29 Juli 2021.

DAFTAR PUSTAKA

- A. R. Lubis, S. P., Y. Fatmi, Al-Khowarizmi, Julham and Y. Y. Lase. (2021). Detection of HOG Features on Tuberculosis X-Ray Results Using SVM and KNN. Paper presented at the 2021 2nd International Conference on Innovative and Creative Information Technology (ICITech).
- Budiman, B., Lubis, C., & Perdana, N. (2021). PENDETEKSIAN PENGGUNAAN MASKER WAJAH DENGAN METODE CONVOLUTIONAL NEURAL NETWORK. Jurnal Ilmu Komputer dan Sistem Informasi, 9(1), 40-47.
- Chau, S., Banjarnahor, J., Irfansyah, D., Kumala, S., & Banjarnahor, J. (2019). Analisis Pendeteksian Pola Wajah Menggunakan Metode Haar-Like Feature. JITE (Journal of Informations and Telecommunication Engineering, 2(2), 69-76.
- Harfi, M. A. R. I. P., Dedi Ari. (2020). Prototipe Pendeteksi Masker pada Ruangan Wajib Masker untuk Kendali Pintu Otomatis berbasis Deep Learning sebagai Pencegahan Penularan COVID-19. Paper presented at the Simposium Nasional Ke-19 RAPI 2020.
- Heryana, N., Mayasari R., & Baihaqi, Kiki Ahmad. (2020). Penerapan Haar Cascade Classification Model untuk Deteksi Wajah, Hidung, Mulut, dan Mata Menggunakan Algoritma Viola-Jones. Techno Xplore Jurnal Ilmu Komputer dan Teknologi Informasi, 5(1), 21-25. doi:<https://doi.org/10.36805/technoxplore.v5i1.1064>
- Julham., A., H.A. (2021). Qr Code Reader System Assisted with Face Detection Using Multi Webcam. JITE (Journal Of Informatics And Telecommunication Engineering), 4(2), 364-372.
- Lambacing, M., & Ferdiansyah, F. (2020). RANCANG BANGUN NEW NORMAL COVID-19 MASKER DETEKTOR DENGAN NOTIFIKASI TELEGRAM BERBASIS INTERNET OF THINGS. Dinamik, 25(2), 77-84. doi:10.35315/dinamik.v25i2.8070
- Putri, T. S. N. P., Al Fikih, Mohamad & Setyawan, Novendra. (2020). FACE MASK DETECTION COVID-19 USING CONVOLUTIONAL NEURAL NETWORK (CNN). Paper presented at the Seminar Nasional Teknologi dan Rekayasa (SENTRA) 2020.
- Riana, F. (2021, 2 Maret 2021). Setahun Pandemi Covid-19, Ini Aneka Kebijakan Pemerintah dan Kritiknya tempo.co. Retrieved from <https://nasional.tempo.co/read/1462167/8-juta-penduduk-indonesia-sudah-mengikuti-vaksinasi-covid-19>
- Singh, S., Ahuja, U., Kumar, M., Kumar, K. & Sachdeva, M. (2021). Face mask detection using YOLOv3 and faster R-CNN models: COVID-19 environment. Multimedia Tools and Applications. doi:10.1007/s11042-021-10711-8
- Sugiyono, M. (2017). penelitian & pengembangan (Research and Development/R&D): Bandung: Penerbit Alfabeta.
- Susanto, S., Putra, Febri Alwan, Analia, Riska and Suciningtyas, Ika Karlina Laila Nur. (2020). The Face Mask Detection For Preventing the Spread of COVID-19 at Politeknik Negeri Batam. Paper presented at the 2020 3rd International Conference on Applied Engineering (ICAE).
- Wiryanawan, I. W. (2020). KEBIJAKAN PEMERINTAH DALAM PENANGANAN PANDEMI VIRUS CORONA DISEASE 2019 (COVID-19) DI INDONESIA Paper presented at the Prosiding Seminar Nasional Webinar Nasional Universitas Mahasaraswati Denpasar, Denpasar.