

Pendeteksi Masker Pada Wajah Dengan Menggunakan Algoritma Haar Cascade Classifier

¹ Renaldy William Lijaya Therry, ² Zacky Yaser Malik Gumiwang,
^{*3} Wahyu S.J.Saputra

^{1,2,3}Informatika, Ilmu Komputer, UPN 'Veteran' Jawa Timur
Jl. Rungkut Madya No.1, Gn. Anyar, Kec. Gn. Anyar, Kota SBY, Jawa Timur 60294
e-mail: wahyu.s.j.saputra.if@upnjatim.ac.id

Received: May 26, 2022, **Revised:** June 26, 2022, **Accepted:** July 20, 2022

Abstrak

Hanya dua hal yang dapat melawan pandemi saat ini adalah masker wajah dan pembersih. Tapi ada orang yang tidak mematuhi aturan atau peraturan apa pun. Untuk membatasi orang-orang dari datang ke tempat-tempat umum dan menjaga diri serta orang lain dalam bahaya, perlu ada sebagai mekanisme untuk cepat, efisien dan sederhana Metode Deteksi Masker Wajah di tempat umum. Karya ini berfokus pada pengembangan model Untuk Masker Wajah deteksi yang dapat dengan mudah dipasang di tempat-tempat umum seperti stasiun kereta api, halte bus, rumah sakit, belanja mal, kantor, stadion dll. Model yang diusulkan akan menerima gambar dari kamera CCTV yang sudah dipasang di sebagian besar tempat umum lainnya saat ini dan akan mencoba mendeteksi jika ada orang di sana tanpa topeng. Model yang diusulkan akan menggunakan model CNN dalam dan algoritma Haar Cascade untuk tujuan ini.

Kata kunci: Komputer, Pendeteksi Wajah, Algoritma Haar Cascade, Keras, Open CV

I. Pendahuluan

Untuk mencegah penyebaran berbagai virus, setiap orang harus memakai masker wajah setidaknya di tempat umum. Tapi banyak ada orang-orang yang tidak mematuhi aturan COVID dan mereka menciptakan masalah tidak hanya untuk diri mereka sendiri tetapi untuk orang lain juga. Dengan demikian ada kebutuhan untuk mengidentifikasi orang-orang dan menghukum mereka sehingga jenis kelalaian dapat dihentikan dan mereka yang mengikuti norma harus diselamatkan dari penyakit berbahaya ini. Deteksi masker di antara orang banyak dapat memecahkan masalah ini. Untuk mengatasi masalah ini kami telah mencoba mengembangkan model yang dapat mengambil CCTV feed di tempat umum dan memberitahu dalam kasus siapa pun tanpa topeng terdeteksi. Ini solusi tidak hanya akan membantu dalam memeriksa orang memakai masker wajah atau tidak untuk melindungi dari virus, juga dapat digunakan dalam berbagai aplikasi seperti pusat ATM, Bank dll. Misalnya jika seseorang masuk ke dalam dengan memakai topeng maka secara otomatis memberitahukan kepada otorisasi utama. Deteksi Masker Wajah adalah metode mendeteksi masker wajah. Dengan perkembangan deteksi wajah kita dapat melihat bahwa seseorang mengenakan topeng yang setuju dan memungkinkan masuknya mereka bisa sangat membantu masyarakat. Dalam model Deteksi Masker Wajah kami memiliki dua fase mereka adalah sebagai berikut:

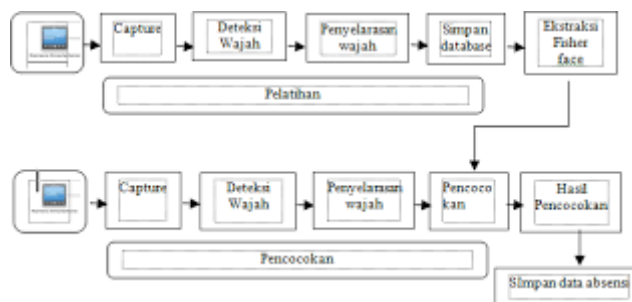
1. Melatih detektor masker wajah
2. Menerapkan detektor masker wajah



This work is licensed under a [Creative Commons Attribution 4.0 International License](http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta).
<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.831>

yang berarti diklasifikasikan ke dalam fase pelatihan dan fase penyebaran.



Gambar 1: Dua Fase Pendeteksi Wajah.

1 Metodologi

Dalam beberapa dekade terakhir, deteksi wajah dan pengenalan wajah dipelajari secara komprehensif. Seperti Viola dan Jones diciptakan face detector yang merupakan cascade Adaboost menggunakan algoritma Haar Cascade. Kemudian bahwa berbagai karya bekerja pada fitur yang berbeda. Seperti berbagai pengklasifikasi yang berarti jenis topeng, mengidentifikasi untuk bergerak objek, gambar hidup dan sebagainya. Dan juga di ruang medis deteksi masker wajah melalui atribut wajah. Dalam hal itu proses pertama mendeteksi wajah dan kemudian masker untuk memberikan keamanan. Dalam metode yang diusulkan penulis mengusulkan berbagai algoritma seperti OHMM, algoritma kaskade HAAR untuk meningkatkan deteksi kinerja objek.

PAKET IMPOR

1. Keras
2. Tensorflow
3. OpenCV

1. Keras

Keras adalah yang menyediakan antarmuka python untuk artificial neural networks dan itu adalah perpustakaan open source. Ini menyediakan API tingkat tinggi dan juga mengurangi jumlah tindakan pengguna. lapisan mana yang digunakan dalam model CNN adalah dengan menggunakan Keras.

2. Tensorflow

TensorFlow, antarmuka untuk mendemonstrasikan kemampuan pembelajaran mesin, digunakan untuk menggunakan sistem ML di konstruksi berbagai aplikasi ilmu komputer seperti analisis sensorik, mengenali suara, ekstraksi lokasi, metode pemrosesan gambar canggih seperti visi komputer, conclusion teks, pengambilan data dll. Dalam model yang diusulkan, seluruh struktur CNN berurutan (terdiri dari beberapa lapisan) menggunakan TensorFlow kembali. Ini juga digunakan untuk mengubah ukuran data (gambar) dalam pemrosesan data. OpenCV OpenCV berguna untuk mengidentifikasi gerakan objek, warna, mengenali objek, gerakan mata, tindakan track kamera. Ini adalah platform open source yang open source visi komputer dan perpustakaan perangkat lunak pembelajaran mesin.



This work is licensed under a [Creative Commons Attribution 4.0 International License](http://creativecommons.org/licenses/by/4.0/).
<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

II. Analisis dan Pembahasan

Dalam pendekatan yang diusulkan ini, kami mengusulkan cara mendeteksi masker wajah menggunakan haar cascade di Open CV dan Keras di detail:

Model Pengujian

Dalam proses ini pertama-tama kita perlu melatih detektor masker wajah, kita dapat memecah proses ini menjadi dua sub-Bagian. Masing-masing sub-Bagian akan memiliki langkah-langkahnya sendiri seperti yang disebutkan pada gambar di bawah ini:

Tahap pertama adalah pelatihan:

Pada tahap pelatihan pertama-tama kita akan memuat kumpulan data Deteksi Masker wajah kita, akan melatih model Menggunakan Keras dan Tensor Flow di Google Colab.

Penyebaran Tahap Kedua:

Setelah kumpulan data dilatih dalam fase penyebaran, kami akan pindah ke Memuat model Deteksi Masker Wajah, akan lakukan deteksi, dan selanjutnya akan mengklasifikasikan setiap wajah seperti halnya dengan topeng atau tanpa topeng. biarkan pertama memiliki deskripsi tentang kumpulan data yang akan kita gunakan untuk melatih model kita. Dalam mudah deskriptif terutama kita mendapatkan gambar dengan wajah dan menjalankannya melalui algoritma klasifikasi kaskade. Classifier adalah memberikan wilayah kepentingan dari wajah dengan tinggi dan lebar. Selanjutnya kita akan mengubah ukuran wilayah yang diinginkan menjadi dimensi 100x100 dan meneruskannya untuk pra dilatih CNN, itu akan memberi kita probabilitas sebagai output. kami mengambil gambar sebagai kumpulan data yang diklasifikasikan menjadi tanpa topeng dan dengan topeng. Sekarang kita akan menggunakan gambar-gambar ini untuk membangun model CNN (convolutional neural network) yang dalam dengan menggunakan HAAR algoritma untuk mengidentifikasi apakah orang memakai topeng atau tidak memakai topeng. Fase pelatihan dan penyebaran

diklasifikasikan ke dalam fase berikut.

1. Mengumpulkan Data
2. Preprocessing Data
3. Memisahkan data
4. Membangun model CNN
5. Pra-pelatihan model CNN
6. Trainig model CNN
7. Memberi label informasi
8. Mengimpor proses deteksi wajah
9. Mengidentifikasi wajah dengan topeng dan tanpa topeng

1. Pengumpulan Data

Dalam Deteksi Masker Wajah Langkah pertama dan terpenting adalah mengumpulkan data. Ini kita sebut sebagai dataset, yang satu ini berguna untuk mendeteksi siapa yang memakai dan siapa yang tidak memakai topeng. Proses ini juga disebut sebagai visualisasi data. Dalam data visualisasi kami mempartisi dataset menjadi dua kelas yaitu dengan mask dan tanpa mask. Pengenalan Data adalah langkah yang sangat penting dalam setiap pekerjaan analisis ilmu data, setelah data yang diperlukan dikumpulkan, diproses sesuai yang dibutuhkan dan dimodelkan, itu harus divisualisasikan untuk menarik kesimpulan. Deteksi Data juga merupakan fitur disiplin pengiriman data yang luas, yang bertujuan untuk mengidentifikasi, mendeteksi, memanipulasi, memformat dan mengirimkan data



This work is licensed under a [Creative Commons Attribution 4.0 International License](http://creativecommons.org/licenses/by/4.0/).
<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.831>

dalam proses yang sangat efisien. kami memutuskan hari menjadi dua kelompok dan diberi label dalam proses ini yang pertama dan terutama langkah adalah untuk memiliki pengetahuan tentang jumlah total gambar dalam dataset kami di kedua bertopeng dan non-bertopeng.



Gambar 2: Tanpa Masker



Gambar 3: Dengan Masker

2. Preprocessing Data

Preprocessing Data juga disebut sebagai augmentasi data. Ini adalah langkah sebelum melatih data dan menguji data. Dalam hal ini kami memiliki empat langkah. Mereka adalah:

1. Mengubah ukuran gambar
2. mengubah gambar menjadi array
3. Menggunakan mobilenetv2 menambah data.
4. Sekarang melakukan label untuk mengkodekan.

Dalam sebagian besar kasus praktis, kami memiliki kumpulan data terbatas karena biaya pengumpulan data tinggi dan juga waktu memakan, tetapi sebagian besar model CNN membutuhkan kumpulan data yang sangat besar. Untuk menggunakan dataset kami secara efisien dan untuk teknik augmentasi data yang sepenuhnya potensial digunakan. Penambahan Data atau augmentasi data mengacu pada suatu proses yang dapat membantu dalam meningkatkan jumlah data yang kita miliki dengan menambahkan salinan modifikasi dari data yang ada atau dengan membuat data baru dengan berbagai metode dasar seperti rotasi, inversi, ekspansi dll. dari data yang ada. Ia bekerja seperti biasa dan membantu mengurangi keseimbangan saat melatih model pembelajaran mesin. Proses ini hampir sama seperti sampling dalam analisis data .. Dalam proses ini, kami akan menyiapkan kumpulan data dua tahap untuk menambahkan beberapa gambar ke model pelatihan kami. Dalam proses pengolahan data ini kami akan memutar dan memindai setiap gambar dalam kumpulan data kami. Setelah Data proses augmentasi, kita melihat ini, misalnya kita memiliki jumlah total 144 gambar kabur dan gambar diputar juga termasuk ketika gambar akan meningkat karena di sini kita mengambil beberapa foto dengan memasukkan mereka secara terpisah.

3. Memisahkan Data



This work is licensed under a [Creative Commons Attribution 4.0 International License](http://creativecommons.org/licenses/by/4.0/).
<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.831>

Setelah fase augmentasi atau preprocessing, kami membagi data menjadi train set dan test set. Di sini kita membelah delapan puluh persen untuk set pelatihan dan tersisa ke set tes. Kami akan menggunakan data set kereta untuk melatih Deep kami Model CNN. Karena pelatihan membutuhkan sejumlah besar data, kami telah menyimpan 80% dari dataset kami untuk pelatihan saja. Setelah pelatihan model kita perlu menguji model pada data yang tak terlihat. Untuk itu kita akan menggunakan test dataset. Jadi, kita akan membagi dataset kami dalam dua bagian dengan rasio split 0,8, yang setara dengan 80 persen dari dataset, yang kita akan gunakan untuk tujuan pelatihan dan sisanya 20% gambar akan digunakan untuk menguji model CNN kami. Sebagai contoh, total dari satu set pelatihan dengan label yang mengatakan ya berarti Topeng adalah 1024 sepintar itu, dalam satu set tes dengan topeng itu berarti Anda memiliki label 144. dan dengan pengecualian topeng berlabel untuk pelatihan dan pengujian pengaturan data dari kumpulan data adalah 144 dan 96.

4. Membangun Model

Setelah memisahkan dataset, langkah selanjutnya adalah membangun model. kami akan membangun jaringan Konvolusional berurutan Model dengan lapisan yang berbeda. Urutan proses pada setiap lapisan adalah Conv2Dlayer, kemudian MaxPooling2D layer, kemudian meratakan lapisan, lapisan untuk putus sekolah dan kemudian lapisan padat di akhir. Dalam fase ini kami memiliki enam Langkah mereka adalah:

1. Preprocessing gambar pelatihan.
2. Mobile Net V2 model CNN dasar
3. Menginisialisasi berbagai parameter model CNN
4. Menyusun dan menyetel model CNN
5. Pelatihan model CNN
6. Untuk proses prediksi lebih lanjut menyimpan model.

Dalam dropout dan lapisan padat, kami telah menggunakan fungsi softmax untuk mengekstrak satu set vektor yang menyediakan peluang untuk masing-masing dari dua kategori.



Gambar 4: Masker untuk Data Training

5. Pre-Training model CNN

Setelah membangun model kami, langkah selanjutnya adalah pra pelatihan model cnn yang berarti membiarkan kita membangun ' kereta generator ' dan generator konfirmasi berarti generator validasi sesuai model kami pada langkah berikutnya untuk Pelatihan Jaringan Saraf Konvolusional (CNN).

6. Training model

Langkah selanjutnya yang akan mengikuti pelatihan pra-model adalah, untuk melatih model yang kami usulkan, ini adalah yang paling penting proses. Kami akan melatih model kami pada Dataset



This work is licensed under a [Creative Commons Attribution 4.0 International License](http://creativecommons.org/licenses/by/4.0/).
<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.831>

pelatihan kami dengan kedua jenis data yaitu dengan mask dan tanpa masker menggunakan paket keras di Google Colab. Kami dapat melatih model kami sebanyak mungkin tetapi kami menemukan 30 sudah cukup. Di mana pengulangan membantu meningkatkan akurasi, jumlah pengulangan mengarah ke peningkatan akurasi, tetapi mengambil banyak waktu. Kita dapat melatih model kami beberapa tambahan jumlah untuk mencapai akurasi yang lebih tinggi tetapi itu tidak menghasilkan banyak perubahan hasil yang signifikan, selain itu juga dapat menyebabkan overfitting model kami karena yang kita dapat memiliki akurasi yang tinggi dalam waktu pelatihan tetapi saat pengujian waktu akurasi kami akan rendah. Kami mengamati bahwa setelah 30 pengulangan model kami adalah 98,86 persen akurat dengan set pelatihan dan 96,19 persen dengan dataset tes. Hasil di atas menunjukkan bahwa model kami sangat baik dilatih tanpa kesempatan untuk menjadi terlalu seimbang.

7. Memberi label informasi

Di sebelah pelatihan dan pembuatan model, kami memiliki dua kemungkinan untuk output kami. mereka adalah 'nol' sesuai dengan tanpa topeng dan 'satu' sesuai dengan topeng. untuk menunjukkan secara efektif bahwa dengan batas hijau dan merah untuk kedua kategori kami juga menetapkan warna persegi panjang batas menggunakan RGB (Merah Hijau Biru) nilai-nilai. label dict = 0 berarti 'tanpa topeng', 1 berarti 'Dengan topeng' mendikte atau perintah warna = 0: (merah) (0,0,255), 1: (Hijau) (0,255,0).

8. Model deteksi wajah berjenjang

Kami mencoba menguji model kami secara praktis, untuk mengetahui apakah kami mengenakan masker wajah atau tidak dari feed dari webcam PC kami. Untuk ini, pertama-tama, kita perlu menggunakan deteksi wajah. Untuk mencapai tugas ini kami telah menulis kode terpisah untuk mengambil input dari webcam PC, maka kita akan menerapkan model detektor wajah untuk mendeteksi wajah di gambar dari webcam PC. Untuk ini, kami menggunakan pengklasifikasi kaskade fitur berbasis Haar yang terkenal untuk menemukan fitur wajah. Algoritma diklasifikasikan cascade ini dirancang untuk OpenCV untuk mendapatkan wajah depan dengan melatih ribuan gambar. Dan kemudian akan menerapkan model kami di atasnya untuk mengidentifikasi apakah orang tersebut memakai topeng atau tidak.

9. Mengidentifikasi orang dengan atau tanpa topeng

Kami akan menggunakan webcam atau CCTV atau kamera lain untuk umpan video langsung, dan dapat mendeteksi secara real time jika semua orang memakai topeng atau tidak. Video akan dibaca dalam bingkai, algoritma deteksi wajah berbasis HAAR akan digunakan pada frame tersebut. Setelah wajah terdeteksi, kami akan melanjutkan dengan prosedur berikut. Dari ditemukan frame yang berisi wajah, preprocessing akan dilakukan termasuk menyesuaikan ukuran gambar, mengubahnya menjadi cocok untuk, menginstal paket pra-pemrosesan menggunakan Mobile Net V2. Setelah itu, kita akan memprediksi bahwa input gambar terdiri dari orang dengan topeng atau tidak.

III. Hasil dan Pembahasan

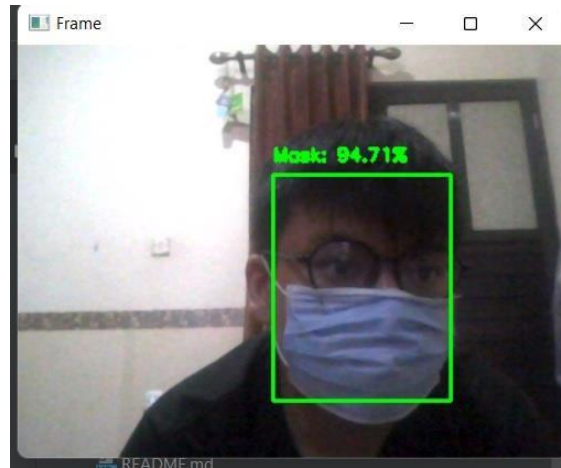
Dalam makalah ini kami mengklasifikasikan dataset menjadi dua kelas: dengan topeng dan tanpa topeng. Jika data dengan gambar maka akan memberikan kotak persegi panjang warna hijau dengan akurasi seperti gambar di bawah ini. Jika input tanpa topeng maka memberikan kotak persegi panjang warna merah di sekitar gambar dan di sini kita dapat mengidentifikasi apakah orang tersebut memakai



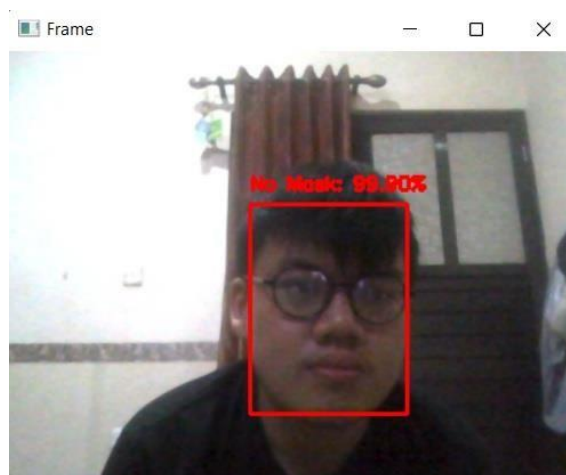
This work is licensed under a [Creative Commons Attribution 4.0 International License](http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta).
<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.831>

masker dengan benar atau tidak. Di sini benar berarti menutupi tanda tanah atau tidak (Mulut dan hidung) dan juga kita dapat mengidentifikasi malpraktek, jika beberapa orang menutupi landmark mereka dengan tangan itu bisa identifikasi dengan mudah dengan algoritma Haar Cascade. Dalam makalah ini, kami mengklasifikasikan gambar menjadi tiga kategori dengan topeng dan tanpa topeng serta halfly menutupi landmark (mulut dan hidung).



Gambar 5: Dengan Masker



Gambar 6: Tanpa Masker

IV. Kesimpulan

Dalam makalah ini, kami telah meningkatkan deteksi wajah menggunakan Python, Kamera, OpenCV dan aliran tensor. Kami membuat model deteksi wajah untuk melihat apakah ada yang memakai topeng. Kami melatih model menggunakan kamera untuk membangun jaringan. Kami mengklasifikasikan dataset ke dalam tiga kategori. Mereka adalah dengan topeng tanpa topeng dan halfly tertutup topeng yang berarti tidak tercakup landmark benar. Berikut ini kami implementasikan dengan menggunakan algoritma haar cascade di buka CV dan Keras. Pada saat itu, kami mewakili tugas pembelajaran dan eksekusi model. Memanfaatkan instrumen ML penting dan peningkatan strategi strategi ini telah mencapai ketepatan yang cukup tinggi. Ini cenderung dimanfaatkan untuk berbagai macam kegunaan. Mengenakan penutup mungkin wajib segera, mengingat Darurat Covid-19. Banyak organisasi spesialis publik akan meminta agar klien mengenakan kerudung akurat untuk mendapatkan keuntungan dari administrasi mereka. Model yang disampaikan akan memberikan



This work is licensed under a [Creative Commons Attribution 4.0 International License](http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta).
<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.831>

kontribusi yang luar biasa bagi Jenderal kerangka perawatan medis. Selanjutnya, kami memperluas kumpulan data dan mencoba mengklasifikasikan berbagai kelas seperti detect jenis masker berarti n95, kapas, bedah atau tidak.

V. Referensi

- [1] Mingjie Jiang, Xinqi Fan and Hong Yan RETINAFACEMASK: A FACE MASK DETECTOR, June, 2020.
- [3] Z. Wang, G. Wang, B. Huang, Z. Xiong, Q. Hong, H. Wu, P. Yi, K. Jiang, N. Wang, Y. Pei et al., "Masked face recognition dataset and application," arXiv preprint arXiv:2003.09093, 2020
- [4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol. 1. IEEE, 2005, pp. 886–893.
- [5] S. Ge, J. Li, Q. Ye, and Z. Luo, "Detecting masked faces in the wild with l1e-cnns," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2682–2690
- [6] Y. Li, B. Sun, T. Wu, and Y. Wang. Face detection with end to-end integration of a convnet and a 3d model. In ECCV, pages 122–138, 2016.
- [7] S. S. Farfade, M. J. Saberian, and L. Li. Multi-view fac detection using deep convolutional neural networks. In ACM ICMR, pages 643–650, 2015.
- [9] Nicolo Bonettini, Edoardo Daniele Cannas and Sara Mandelli Video Face Manipulation Detection Through Ensemble of CNNs



This work is licensed under a [Creative Commons Attribution 4.0 International License](http://creativecommons.org/licenses/by/4.0/).
<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>