

# Aplikasi Pendeteksi Masker Secara Real-Time Menggunakan Metode Software Development Life Cycle

Nathasya G.M. Lonteng \*<sup>1</sup>  
Joelian E. Runturambi <sup>2</sup>  
Kristofel Santa <sup>3</sup>

<sup>1,2,3</sup>Program Studi Teknik Informatika, Fakultas Teknik, Universitas Negeri Manado, Indonesia  
\*e-mail: [nathasyalonteng@gmail.com](mailto:nathasyalonteng@gmail.com)<sup>1</sup>, [joelianrunturambi@gmail.com](mailto:joelianrunturambi@gmail.com)<sup>2</sup>, [kristofelsanta@unima.ac.id](mailto:kristofelsanta@unima.ac.id)<sup>3</sup>

## Abstrak

Dengan munculnya pandemi covid-19 pada beberapa tahun lalu mendorong berbagai macam perubahan seperti penggunaan masker yang pada saat ini sudah semakin sering ditemui, kesadaran terhadap pentingnya penggunaan masker pada saat ini menyebabkan munculnya kebutuhan akan pemeriksaan penggunaan masker pada beberapa tempat publik yang biasanya ramai. Pemeriksaan penggunaan masker ini biasanya dilakukan secara manual sehingga membutuhkan tenaga manusia untuk dapat melakukannya, hal inilah yang menyebabkan keterbatasan dalam pemeriksaan masker karena akan dibutuhkannya ketersediaan petugas agar dapat melakukan pemeriksaan masker setiap saat. Aplikasi pendeteksi penggunaan masker merupakan salah satu solusi agar pemeriksaan masker ini dapat semakin efektif, dalam penggunaannya aplikasi akan mendeteksi wajah yang menggunakan masker dan tidak menggunakan masker dengan memanfaatkan library OpenCV serta python dalam pembuatannya yang akan membuatnya menampilkan hasil seperti konfirmasi penggunaan masker dan tidak menggunakan masker secara langsung. Dengan pengujian yang telah dilakukan, aplikasi ini telah berhasil dijalankan dan mampu untuk melakukan pendeteksian terhadap wajah yang menggunakan masker maupun tidak, sehingga aplikasi ini dapat siap digunakan.

**Kata kunci:** Aplikasi, Deteksi Masker, Python, OpenCV

## Abstract

In response to the Covid-19 pandemic, many changes have come about, including the widespread use of masks. It is now widely recognized that masks play an essential role in public safety, leading to increased scrutiny of mask usage in busy places. However, manual inspection of masks is limited by the availability of human resources. This is where a mask detection application comes in - it offers a more effective solution to mask inspection. By leveraging the OpenCV library and Python, the application is able to accurately detect whether a person is wearing a mask or not, displaying the results immediately. Extensive testing has been conducted, with successful results, making this application ready for practical use.

**Keywords:** Application, Mask Detection, Python, OpenCV

## PENDAHULUAN

Peristiwa pandemi Covid-19 yang pernah melanda berbagai negara di dunia telah menyebabkan banyaknya korban meninggal. Dalam Covid-19 terkandung virus SARS-Cov-2, virus ini dapat menyebar melalui percikan yang dihasilkan dari batuk atau bersin, yang kemudian percikan ini masuk ke dalam mata, mulut maupun hidung (Diya Atiqa, 2022)

Agar dapat menekan angka penularan pada saat itu, maka dilakukan berbagai upaya yaitu dengan disiplin melakukan physical distancing, hidup bersih dan sehat, seperti melakukan cuci tangan dengan sabun atau hand sanitizer dan secara disiplin menggunakan protokol kesehatan, seperti menggunakan masker (Illahika dkk., 2023)

Dengan dinyatakan berakhirnya masa pandemi covid-19, maka kehidupan masyarakat telah kembali berjalan seperti sebelumnya. Namun meskipun telah berakhir namun ada hal yang semakin menjadi perhatian masyarakat, yaitu terkait kesehatan.

Di masa pasca Covid-19 saat ini, meskipun kebiasaan hidup bersih penggunaan protokol kesehatan berupa masker sudah tidak seketat dulu saat sedang terjadi pandemi, namun nyatanya kebiasaan untuk menggunakan masker ini masih melekat erat dalam kehidupan sehari-hari

masyarakat (Holifatuz Zahro dkk., 2022). Beberapa tempat tertentu bahkan masih mewajibkan untuk penggunaan masker. Pemeriksaan dilakukan ketika akan masuk ke tempat tersebut. Pemeriksaan ini sendiri masih dilakukan secara manual.

Pemeriksaan secara manual ini tentunya memiliki keterbatasan, hal ini dikarenakan dibutuhkannya tenaga manusia (petugas) yang akan melakukan pemeriksaan sepanjang waktu (Pratama dkk., 2023)

Dengan munculnya permasalahan ini maka penulis melakukan penelitian dengan memanfaatkan machine learning yang dapat membantu dalam pembuatan aplikasi sistem deteksi secara otomatis (Rachman & Maurits, 2021)

Sistem akan melakukan pendeteksian serta klasifikasi terhadap wajah yang sedang berada di depan kamera apakah orang tersebut menggunakan atau tidak (Baay dkk., 2021)

Dengan dilakukannya penelitian ini, penulis berharap agar pemeriksaan penggunaan masker di area-area wajib masker dapat dilakukan lebih efektif, dan efisien, serta dapat membantu pekerjaan dari petugas pemeriksaan masker.

## **METODE**

Pada penelitian untuk membuat aplikasi pendeteksi masker secara real-time ini digunakan metode Software Development Life Cycle (SDLC). Metode SDLC merupakan sebuah tahapan yang digunakan dalam pembangunan software, agar akan dapat menghasilkan hasil software yang dapat memenuhi keinginan serta kebutuhan pengguna (Romli dkk., 2023).

Setiap tahapan yang ada dalam SDLC dapat dipaparkan melalui tujuan yang ingin dicapai serta hasil yang diperoleh dari kegiatan tersebut (Widharma, 2017). Terdapat 5 tahapan dalam SDLC

1. Perencanaan

Pada tahap ini dilakukan pengumpulan data informasi terkait dengan pengembangan aplikasi. Teknik pengumpulan informasi dilakukan dengan melakukan study pustaka yaitu mengumpulkan informasi dengan cara membaca buku-buku, serta jurnal-jurnal yang memiliki keterkaitan dengan topik penelitian yang dipilih (Terok dkk., 2020), selain itu dilakukan perencanaan aplikasi dengan mencari informasi dari video-video yang relevan dengan penelitian

2. Analisis

Menganalisis cara kerja sistem terhadap objek yang akan dideteksi dengan cermat untuk memastikan bahwa hasilnya memenuhi harapan.

3. Perancangan

Pada fase perancangan, akan dibuat rancangan mengenai cara kerja dari program yang nantinya akan dibangun pada tahapan berikutnya. Digunakan flowchart, struktur navigasi, diagram use case, diagram aktivitas, dan tampilan beberapa dataset yang digunakan dalam konstruksi sistem deteksi ini.

4. Implementasi

Pada tahap ini dilakukan implementasi yaitu dengan pembuatan kode program dengan menggunakan bahasa pemrograman Python, framework TensorFlow, Library OpenCV, Keras, dan MobileNET

5. Pengujian

Pengujian merupakan tahapan terakhir dalam metode ini. Dilakukan beberapa kali pengujian dengan menjalankan sistem pada beberapa versi desktop serta perangkat yang berbeda. Hal ini bertujuan untuk memastikan bahwa tidak ada kesalahan pada sistem sebelum digunakan

## **HASIL DAN PEMBAHASAN**

Dilakukan penelitian dengan menggunakan metode yang sudah dijelaskan sebelumnya, berikut ini merupakan hasil serta pembahasan dalam pengembangan aplikasi pendeteksi masker

Bagian hasil penelitian memuat hasil analisis uji hipotesis yang dapat menyertakan tabel, grafik, dan sebagainya.

Pembahasan memuat interpretasi dan evaluasi terhadap hasil penelitian, serta ulasan berbagai permasalahan terkait yang dipandang dapat memengaruhi hasil penelitian. Deskripsi pada bagian ini menitikberatkan pada analisis secara kritis secara substansial terhadap hasil penelitian, selain itu ditambahkan juga kelemahan dalam penelitian.

### **Analisis Kebutuhan**

Dalam pengembangan suatu program, diperlukan sebuah analisis yang mengilustrasikan cara program tersebut dapat digunakan. Analisis kebutuhan merupakan suatu pendekatan untuk menggambarkan persyaratan yang diperlukan dalam penelitian, terutama dalam pembuatan program deteksi pemakaian masker secara real-time.

a. Analisis Kebutuhan Perangkat Lunak

Dalam melakukan pembangunan aplikasi ini dilakukan analisis kebutuhan perangkat lunak sebagai berikut :

- Windows 11 64 bit
- Jupyter Notebook
- Python 3.9.7
- Draw.io
- Github
- Tensorflow 2.3.4
- OpenCV 4.2.0
- Keras 2.3.1

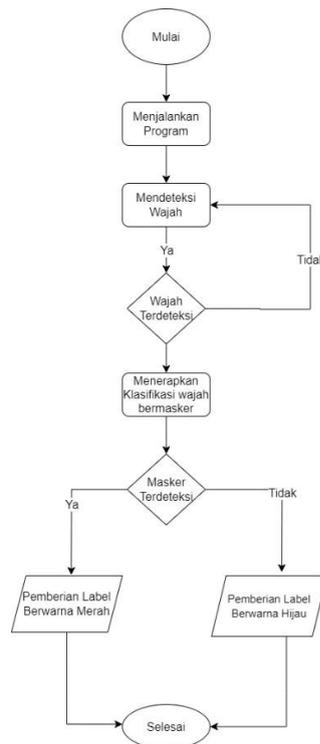
b. Analisis Kebutuhan Perangkat Keras

Dilakukan juga analisis kebutuhan dari perangkat keras dalam pembuatan aplikasi pendeteksi masker ini, yaitu sebagai berikut :

- 16GB RAM
- 512 SSD
- AMD RX560X
- 1080 Front Camera

### **Flowchart**

Flowchart digunakan untuk mengilustrasikan rangkaian program yang dibuat, khususnya dalam konteks aplikasi sistem deteksi pemakaian masker secara real-time dengan menggunakan framework TensorFlow dan Library OpenCV pada wajah



Gambar 1. Flowchart Aplikasi Pendeteksi Masker

### Struktur Navigasi

Fungsi dari struktur navigasi adalah mengilustrasikan alur proses aplikasi yang dikembangkan. Dalam pembuatan aplikasi sistem deteksi pemakaian maskser secara real-time dengan menggunakan framework TensorFlow dan library OpenCV, Struktur navigasi yang diterapkan adalah tipe struktur navigasi linier. Berikut adalah jenis struktur navigasi yang digunakan pada aplikasi tersebut

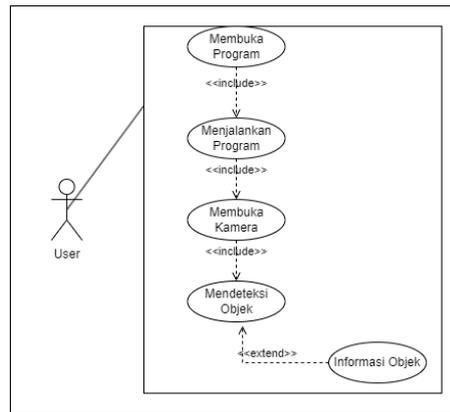


Gambar 2. Struktur Navigasi Aplikasi Pendeteksi Masker

### Perancangan UML Diagram

Dalam pembangunan aplikasi pendeteksi masker ini, untuk itu dilakukan pemodelan sistem dengan menggunakan UML diagram, sebagai berikut

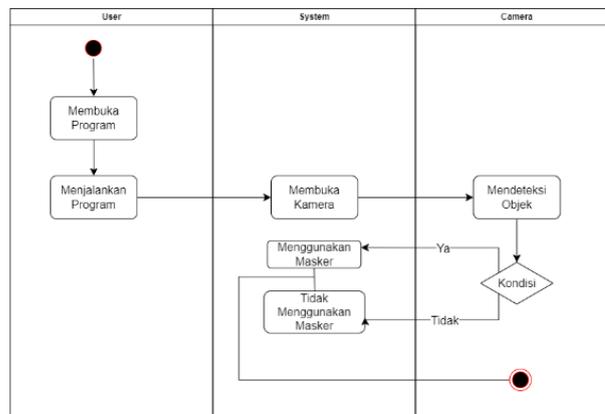
- a. Use Case Diagram



Gambar 3. Use Case Diagram Program

Dalam penggambaran dengan use case diagram ini menjelaskan bahwa user yang akan menggunakan aplikasi ini harus membuka program ini terlebih dahulu, dengan membuka program maka akan otomatis menjalankan program dan akan membuka kamera yang nantinya secara otomatis akan mendeteksi penggunaan masker berdasarkan informasi dari dataset yang telah diberikan ke dalam program.

b. Activity Diagram



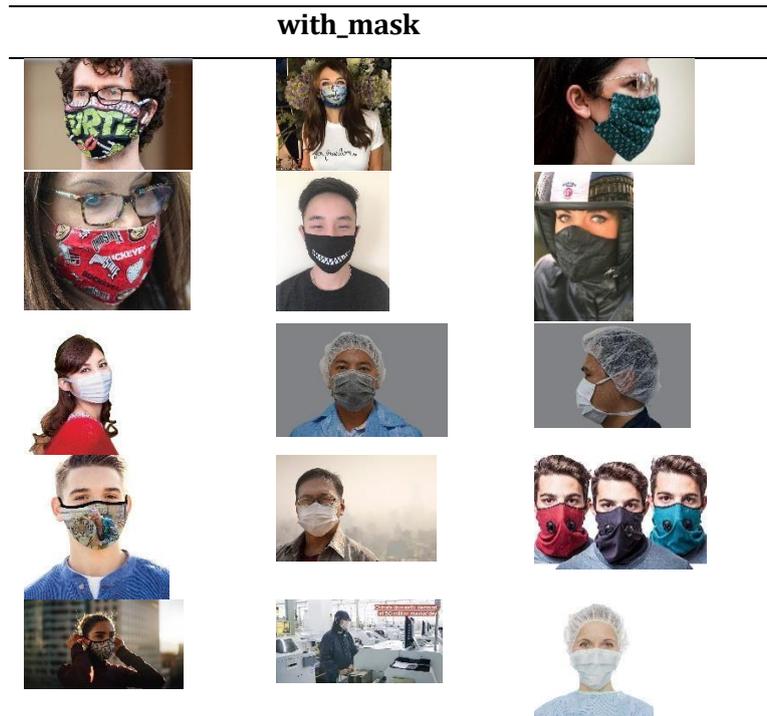
Gambar 4. Activity Diagram Program

Dalam penggambaran dengan activity diagram ini menjelaskan bagaimana cara kerja dari sistem. Saat user membuka aplikasi maka program akan berjalan, dengan berjalannya program sistem akan beraksi dengan membuka kamera, kamera yang telah terbuka akan mendeteksi objek yang berada di depannya, dari hasil deteksi objek ini akan diberikan 2 kondisi. Bila objek terdeteksi menggunakan masker maka akan diberikan keterangan pada tampilan kamera jika objek memang terdeteksi menggunakan masker, namun apabila sistem tidak mendeteksi penggunaan masker terhadap objek maka pada tampilan layar untuk kamera akan menampilkan keterangan bahwa objek tidak mengenakan masker.

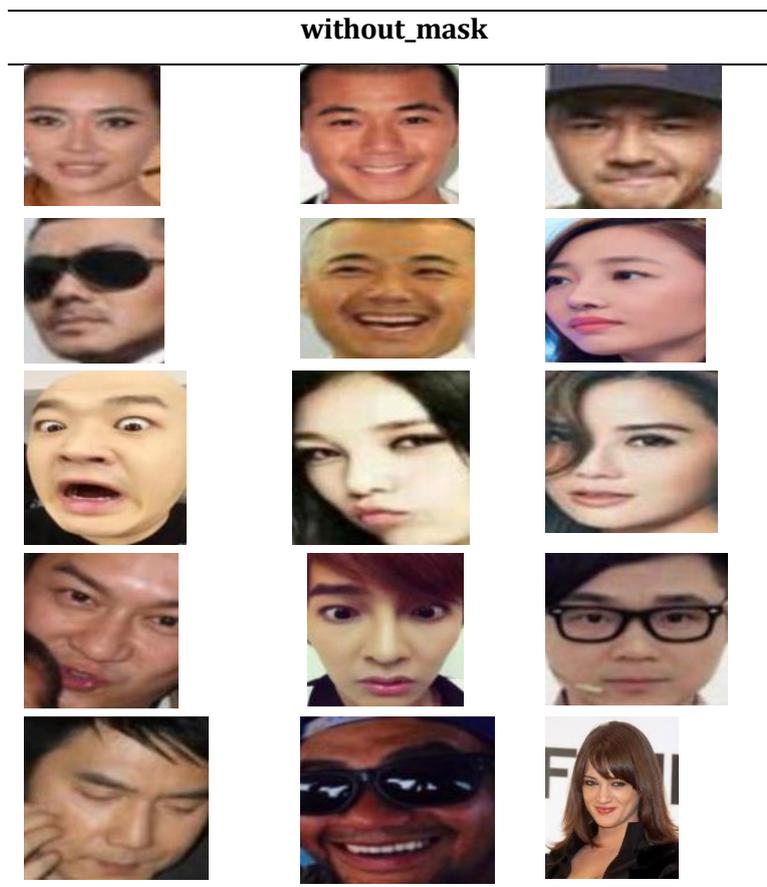
**Pengumpulan Dataset Gambar**

Dataset ini terbagi ke dalam dua bagian, yaitu with\_mask yang berisi 1.915 gambar individu atau objek dengan penggunaan masker, dan without\_mask yang berisi 1.918 gambar individu atau objek tanpa penggunaan masker. Setiap gambar yang terdapat dalam dataset ini terdiri dari berbagai variasi keadaan dan kondisi yang berbeda, termasuk perbedaan latar belakangnya.

Tabel 1. Dataset gambar with\_mask



Tabel 2. Dataset gambar without\_mask



### Uji Coba Program

Uji coba program bertujuan untuk mengevaluasi performa pada berbagai perangkat laptop, dengan tujuan mengobservasi bagaimana program berfungsi pada lokasi yang berbeda serta bagaimana jalannya program pada perangkat yang berbeda. Berikut adalah hasil pengujian yang dilakukan pada dua perangkat laptop yang berbeda.

Tabel 3. Uji Coba Program

Device	Spesifikasi	Hasil
Asus TUF FX505DY	AMD Ryzen 5 16GB RAM 512GB SSD RX560X 1080 Camera	Pada perangkat Laptop Asus TUF FX505DY, program berjalan dengan baik dan mendeteksi dengan baik.
HP 14S DK0073AU	AMD A4-9125 4GB RAM 1TB HDD SATA HP TrueVision HD Camera	Pada perangkat HP 14S DK0073AU, program dapat berjalan dengan baik dan mendeteksi dengan baik



Gambar 5. Hasil Deteksi Tanpa Masker



Gambar 6. Hasil Deteksi Tanpa Masker 2

Pada Gambar 5 dan 6 dilakukan uji coba program, dengan kondisi tidak menggunakan masker dan dilakukan uji menggunakan 1 wajah serta 2 wajah. Hasil yang didapatkan yaitu program berhasil untuk melakukan pendeteksian jumlah wajah serta melakukan pendeteksian masker, dimana pada percobaan kali ini tidak digunakan masker, sehingga muncul notifikasi pada tampilan bahwa "Mask Off" tidak terdeteksinya penggunaan masker.



Gambar 7. Hasil Deteksi Menggunakan Masker



Gambar 8. Hasil Deteksi Menggunakan Masker 2

Pada Gambar 7 dan 8 dilakukan uji coba program, dengan kondisi menggunakan masker. Uji coba masih dilakukan 2 kali. Pada percobaan yang pertama program berhasil melakukan pendeteksian terhadap 1 wajah dan berhasil melakukan pendeteksian penggunaan masker. Untuk percobaan kedua program juga berhasil untuk melakukan pendeteksian 2 wajah dan berhasil melakukan pendeteksian penggunaan masker.

## KESIMPULAN

Berdasarkan permasalahan yang ditemukan oleh penulis terkait pemeriksaan dalam penggunaan masker yang masih dilakukan secara manual atau membutuhkan tenaga manusia (petugas), maka dilakukan penelitian mengenai pembuatan program pendeteksi masker secara real-time, penelitian ini menggunakan metode SDLC. Berdasarkan hasil uji coba yang telah dilakukan sebelumnya maka hasil yang didapatkan yaitu aplikasi yang dibuat dapat berjalan serta beroperasi dengan baik, dimana aplikasi berhasil dalam melakukan pendeteksian terhadap jumlah wajah serta pendeteksian penggunaan masker maupun tidak. Dengan dibuatnya aplikasi ini maka masyarakat dapat memanfaatkan aplikasi ini untuk mempermudah pekerjaan dalam

pemeriksaan penggunaan masker, agar nantinya pekerjaan ini dapat menjadi lebih efektif serta efisien.

## DAFTAR PUSTAKA

- Baay, M. N., Irfansyah, A. N., & Attamimi, M. (2021). Sistem Otomatis Pendeteksi Wajah Bermasker Menggunakan Deep Learning. *Jurnal Teknik ITS*, 10(1), 64–70.
- Diya Atiqa, U. (2022). Perilaku Masyarakat dalam Penggunaan Masker sebagai Upaya Pencegahan Penularan Covid-19 di Pasar Kebayoran Jakarta Selatan Community Behavior Mask of Mask Usage in Prevention Efforts of Covid-19 Transmission in Kebayoran Market of South Jakarta. *Jurnal Ilmiah Kesehatan Masyarakat*, 14, 118–124.
- Holifatuz Zahro, N., Ade Setiawan, G., Amroellah, A., & Abdurachman Saleh Situbondo, U. (2022). POLA HIDUP SEHAT PASCA PANDEMI COVID-19 POST COVID-19 PANDEMIC HEALTHY LIFESTYLE. *JURNAL PENGABDIAN*, 1.
- Illahika, A. P., Pravitasari, D. N., & Tunjungsari, F. (2023). Hubungan Pengetahuan dan Perilaku Penggunaan Masker Sebagai Usaha Pencegahan Penularan COVID-19. *INFOKES: Jurnal Ilmiah Rekam Media dan Informatika Kesehatan*, 13(2), 117–121. <https://doi.org/10.22435/hsr.v2i3.469>
- Pratama, N. A., Sari, A. C., & Batubara, P. K. N. (2023). Aplikasi Deteksi Wajah Menggunakan Pemrograman Python. *JITKO: Jurnal Inovasi Teknologi dan Komputer*, 1–4. <https://ejournal.harapanutama.org/index.php/JITKO>
- Rachman, A. A., & Maurits, I. (2021). SISTEM DETEKSI PEMAKAIAN MASKER PADA WAJAH SECARA REAL TIME MENGGUNAKAN FRAMEWORK TENSORFLOW DAN LIBRARY OPENCV. *JUIT Jurnal Ilmiah Teknik*, 2(1), 49–59.
- Romli, R., Wiyanto, W., & Butsianto, S. (2023). PENGEMBANGAN APLIKASI PERSEDIAAN BARANG BERBASIS WEB MENGGUNAKAN METODE SDLC PADA CV PADU NUSANTARA JAKARTA. *JINTEKS (Jurnal Informatika Teknologi Sains)*, 5(3), 468–476.
- Terok, F. R., Sangkop, I. F., & Santa, K. (2020). Sistem Pendeteksi Gerakan Berbasis Internet of Things (Iot). *JOINTER*, 01(01), 25–29.
- Widharma, I. G. S. (2017). PERANCANGAN SIMULASI SISTEM PENDAFTARAN KURSUS BERBASIS WEB DENGAN METODE SDLC. *Matrix : Jurnal Manajemen Teknologi dan Informatika*, 7(2), 38. <https://doi.org/10.31940/matrix.v7i2.527>