

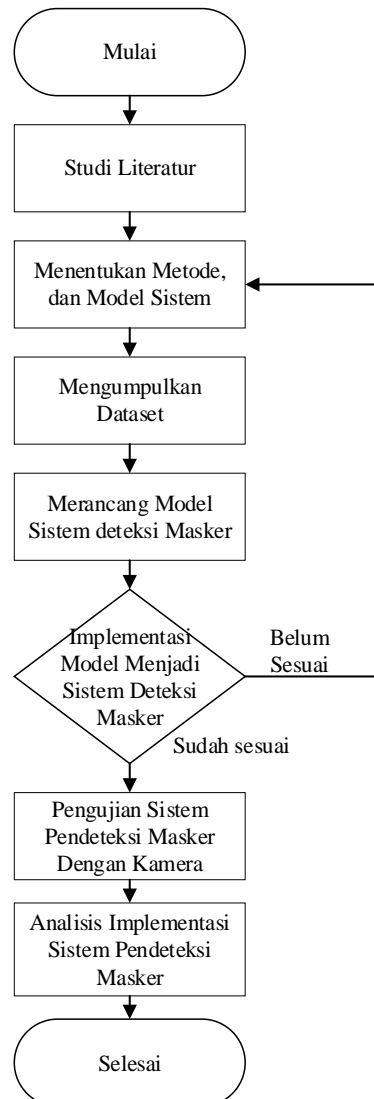
BAB III

METODE PENELITIAN

Pada bab 3 ini akan membahas terkait dengan alat yang dipakai untuk menunjang penelitian dan pembuatan sistem mulai dari alur penelitian, pengumpulan dataset, pembuatan dan pengembangan desain sistem deteksi masker, pengembangan model arsitektur, implementasi sistem desain pada pendeteksi masker, dan pengujian sistem.

3.1. ALUR PENELITIAN

Pada penelitian ini dimulai dari studi literatur, perancangan dari sistem, sampai dengan kesimpulan penelitian yang ditunjukkan pada blok diagram dibawah ini.



Gambar 3.1 Alur Penelitian

Tahapan pertama untuk proses perancangan adalah studi literatur, dimana tahapan ini berkaitan untuk mengkaji berbagai materi-materi pendukung yang berkaitan dengan *convolutional neural network*, *MobileNetV2* dan materi pendukung lainnya untuk membangun sistem deteksi masker. Kemudian pada tahapan berikutnya adalah melakukan penentuan metode apa yang akan digunakan dan bagaimana pembentukan model yang akan digunakan dalam sebuah sistem deteksi masker. Selain itu pada tahapan ini dikaji juga bagaimana keunggulan metode yang akan digunakan dibandingkan dengan menggunakan metode lainnya.

Proses berikutnya adalah melakukan pengumpulan *dataset*, dimana *dataset* ini digunakan sebagai data yang diolah dan dilakukan pembelajaran oleh mesin. Pengumpulan dataset ini dilakukan dengan cara mengambil beberapa gambar secara langsung dengan menggunakan kamera dan juga menggunakan beberapa gambar dari relawan yang bersedia fotonya digunakan. Selain itu memperbanyak jumlah gambar maka ada beberapa gambar yang diambil dari website *kaggle*.

Setelah pengumpulan *dataset* selesai maka sistem mulai dirancang menggunakan metode yang telah ditentukan. Perancangan ini akan dilakukan pada *google collab* dengan bentuk format *google colab* yang sama seperti pada *jupyter notebook*. Kemudian jika model sudah dirancang maka akan dilakukan proses *training* untuk mengetahui pembentukan model sudah dapat diimplementasikan dalam sistem deteksi masker atau tidak. Untuk implementasi model pada sistem deteksi masker yang perlu diperhatikan juga yakni bagaimana keakuratan sistem dalam klasifikasi gambar, maka ketika model sudah sesuai akan lanjut kepada proses pengujian dengan kamera dan jika belum sesuai maka akan kembali kepada proses pembaharuan metode.

Dari hasil pengujian dengan menggunakan kamera akan diujikan dalam beberapa parameter pengujian seperti halnya pengujian dengan parameter jarak, intensitas cahaya dan pengujian dengan menggunakan beberapa jenis masker. Hasil pengujian kemudian akan dianalisa untuk mengetahui keakuratan sistem deteksi dalam klasifikasi gambar

3.2. ALAT YANG DIGUNAKAN

Pada subbab ini membahas tentang perangkat keras dan perangkat lunak yang digunakan pada penulisan penelitian ini.

3.2.1. Perangkat Keras (*Hardware*)

Perangkat keras yang dipakai dalam penelitian ini diantaranya laptop yang terintegrasi dengan webcam dan spesifikasi sebagai berikut:

1. *Processor Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz 1.99 GHz.*
2. *Operating System Windows 10*
3. *RAM 4,00 GB.*
4. *Intel® HD Graphics 520.*
5. *Webcam Xiaovv HD 720p.*

3.2.2. Perangkat Lunak (*Software*)

Selain perangkat keras penggunaan perangkat lunak juga dipakai dalam penelitian ini yakni

1. *Google Colaboratory.*

Penulisan *source code* menjadi pengerjaan utama dalam penelitian ini dimana dalam membuat *source code* tentu akan membutuhkan *software* yang dapat mengeksekusi program yang akan digunakan. Pemilihan *google colab* sebagai *tools* yang digunakan dalam menulis *source code* mempunyai alasan tersendiri. *Google colab* merupakan salah satu produk yang diciptakan oleh Google yang berbasis *cloud* yang dapat digunakan secara gratis. Google colab dapat diaplikasikan sebagai *coding environment* bahasa pemrograman *python* dengan bentuk format *notebook* yang sama seperti *Jupyter Notebook*. Di dalam *google colab* ada tiga *processor* handal yang disediakan yakni untuk keperluan mesin pembelajaran (*machine learning*) yaitu : *Central Processing Unit* (CPU), *Graphic Processing Unit* (GPU) dan *Tensor Processing Unit* (TPU).

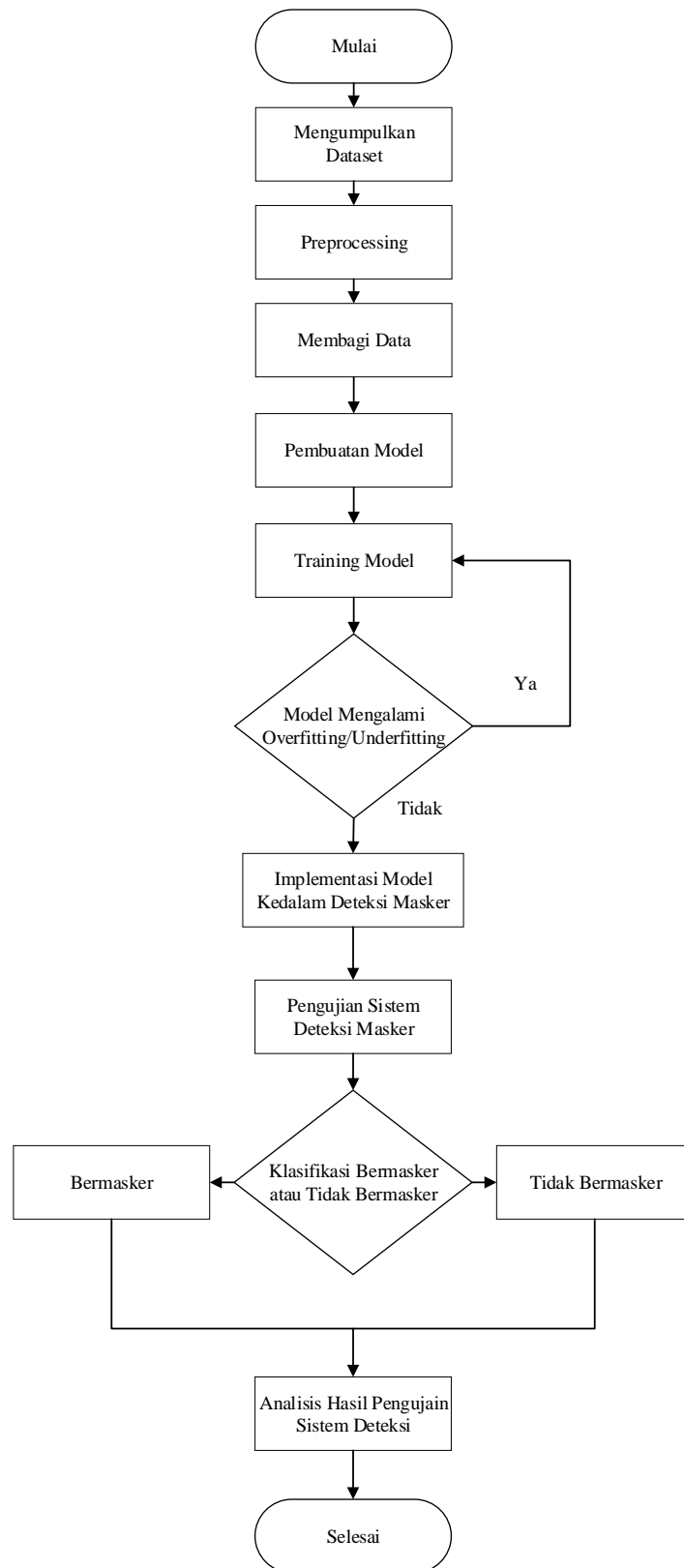
2. Peramban Internet *Google Chrome* versi 97.0.4692.99 (64-bit)

Untuk menjalankan *google colab* tentu dilakukan pada peramban internet *google chrome* yang juga akan terintegrasi pada *google drive*. *Google drive* akan saling terintegrasi dengan *google colab* dimana untuk penyimpanan *source code* dan juga pemanggilan dataset yang disimpan pada *google drive*.

3.3. RANCANGAN SISTEM

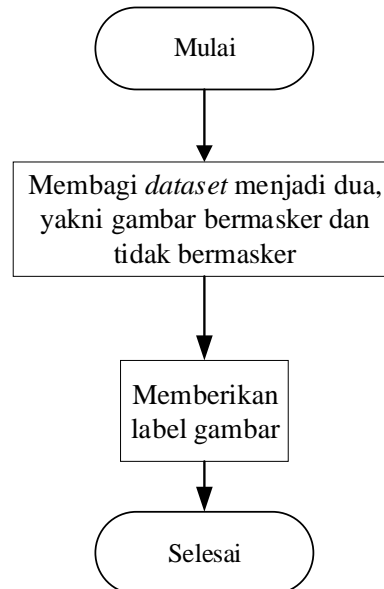
Bentuk dari rancangan sistem yang dibuat menggunakan *flowchart* yang dapat dilihat pada gambar 3.2. Pada diagram flowchart tersebut menjelaskan rancangan

sistem dimulai dari pengumpulan dataset sampai dengan perancangan sistem selesai.



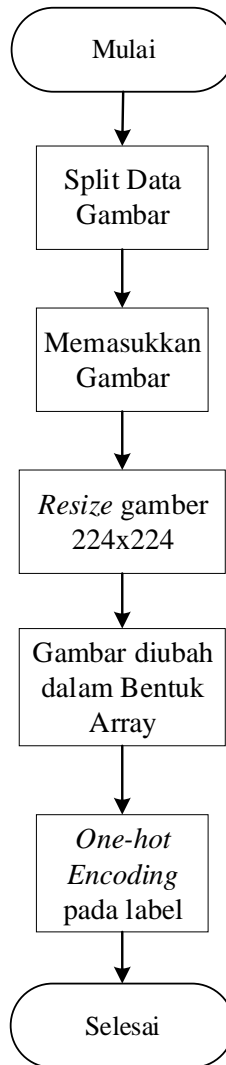
Gambar 3.2 Rancangan Sistem

Berdasarkan gambar 3.2 untuk perancangan sistem deteksi masker dilakukan beberapa tahapan mulai dari tahapan mengumpulkan data, *preprocessing*, hingga kepada analisis hasil pengujian sistem deteksi. Dari setiap tahapan dalam perancangan sistem akan dilakukan beberapa proses tahapan lagi didalamnya dan akan diuraikan setiap prosesnya sebagai berikut



Gambar 3.3 Tahapan Mengumpulkan *Dataset*

Pada perancangan sistem tahapan pertama dimulai dari tahapan mengumpulkan dataset, yaitu proses ini akan mencari dan mengumpulkan *dataset* untuk diolah sebagai pembelajaran mesin. Untuk *dataset* tersebut diperoleh dari basis data terbuka dan juga beberapa penambahan foto yang diambil sendiri, kemudian memasukkan data dalam dua kategori yaitu *dataset* wajah bermasker dan juga *dataset* wajah tidak bermasker. Setelah proses pengumpulan *dataset* selesai yang dilakukan adalah mengupload *dataset* kepada *google drive* untuk nanti dipanggil dan di olah di *google colab*. Dalam tahapan ini juga dilakukan pemberian label pada gambar.

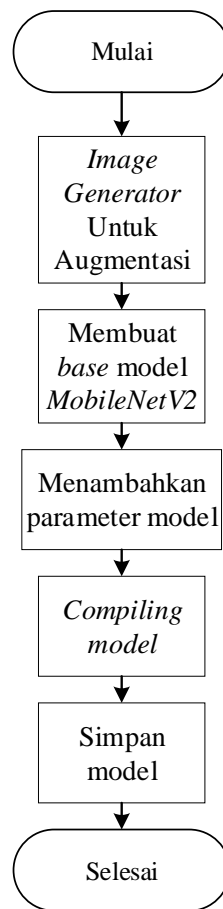


Gambar 3.4 Tahapan *Preprocessing*

Tahapan kedua adalah *preprocessing data*, dimana di dalam tahapan ini dilakukan beberapa pengerjaan seperti halnya mempersiapkan gambar untuk diolah, mengupload gambar pada sistem, mengubah ukuran gambar dan hal lainnya. Perubahan ukuran gambar dilakukan guna menyamaratakan semua ukuran gambar yang ada pada *dataset*. Dalam proses ini gambar akan diubah kedalam bentuk *array*, dimana konversi data gambar ke *array* ini dilakukan untuk menyimpan data yang bertipe sama. *One-hot encoding* juga dilakukan pada tahapan ini dimana nilainya akan diubah dalam vektor biner 0 dan 1.

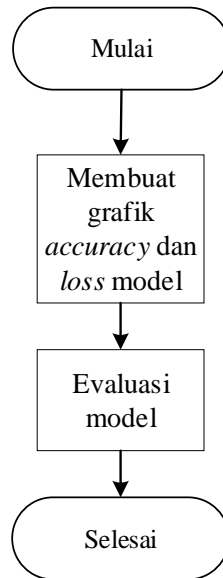
Tahapan ketiga adalah membagi *dataset* menjadi data tes dan data latih. Untuk data latih diberikan data 80% dan data tes 20%. Komposisi nilai tersebut diberikan

bertujuan agar sistem dapat melakukan pembelajaran lebih banyak dan mendapatkan fitur ciri dari tiap citra yang ada pada sebuah gambar.



Gambar 3.5 Tahapan Pembuatan Model

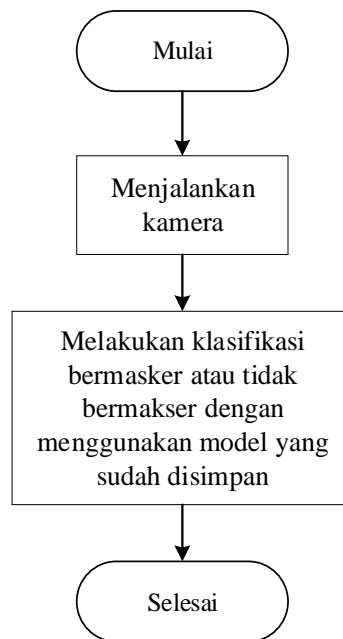
Tahapan keempat yang dilakukan adalah pembuatan model dengan menggunakan arsitektur *MobileNetV2*. Selain pembentukan model di dalam proses ini juga dilakukan data augmentasi seperti halnya untuk merotasi gambar, *blurring* gambar, *flip horizontal* pada gambar dan lainnya. Hal ini dilakukan guna memperkaya gambar dan mendapatkan ciri dari tiap sisi yang ada pada gambar. Selain melakukan augmentasi pada gambar pada tahapan ini dilakukan penambahan beberapa parameter *layer* didalamnya seperti proses *pooling* yang akan menggunakan *average pooling*, aktivasi fungsi menggunakan *Relu* dan klasifikasi yang dilakukan dengan menggunakan *softmax*.



Gambar 3.6 Tahapan Training Model

Proses tahapan kelima yakni melakukan training pada model. Dari proses *training* ini juga dilakukan pencarian nilai *hyperparameter training* yang menghasilkan nilai akurasi terbaik dan nilai *loss* terendah. Penentuan parameter *training* yang akan dicari adalah nilai *learning rate*, *batch size*, dan *epoch* yang paling baik. Pengujian ini dilakukan untuk mengetahui apakah model yang sudah dibangun mengalami kondisi *overfitting* atau *underfitting*. Kondisi sistem yang mengalami *overfitting* akan mengakibatkan pembacaan sistem yang terlalu akurat sehingga nilai akurasinya sangat tinggi tetapi tidak stabil antara data tes dan data latih. Kemudian jika sistem mengalami *underfitting* maka membuat sistem mendapatkan nilai akurasi yang rendah sehingga untuk proses klasifikasi menghasilkan pembacaan yang salah. Penentuan nilai parameter *training* menjadi salah satu kunci untuk mendapatkan nilai akurasi tertinggi dan *loss* paling rendah, sehingga untuk proses *training* akan dilakukan percobaan beberapa kali hingga mendapatkan nilai paling bagus untuk digunakan dalam proses *training* model. Hasil *training* model akan digambarkan pada sebuah grafik dimana akan ditampilkan dua buah grafik, yakni grafik pertama akan menggambarkan nilai akurasi model dan grafik kedua akan menggambarkan nilai *loss* model. Pada setiap grafik juga nantinya akan ditampilkan *validation accuracy*, *training accuracy*, *validation loss* dan *training loss*. Penggambaran ini akan memperlihatkan

bagaimana hasil pembelajaran yang sudah dilakukan oleh sistem dengan pengujian secara aktual.



Gambar 3.7 Tahapan Implementasi Model

Tahapan terakhir adalah implementasi model menjadi sistem deteksi masker. Pada tahapan ini model yang sudah dibentuk akan menjadi algoritma dalam klasifikasi sebuah citra. Pengujian akan langsung dilakukan secara *realtime* dengan menggunakan kamera. Citra wajah yang diinputkan dengan menggunakan kamera akan dianalisa oleh sistem dan akan diklasifikasi apakah citra *inputan* termasuk dalam kelas wajah bermasker atau kelas tidak bermasker. Tahapan dari proses keenam dapat dilihat pada bagan berikut :

3.4. METODE PENGUJIAN

Metode pengujian akan dilakukan dengan beberapa metode yakni sebagai berikut :

3.4.1. Metode Pengujian Sistem

Pada pengujian sistem yang akan dilakukan adalah mengecek seluruh sistem yang telah dirancang guna mengetahui secara keseluruhan bagaimana sistemnya bekerja, apakah bekerja dengan akurasi yang baik atau tidak. Pengujian akan mengacu pada hasil *training* model yakni dengan melihat tingkat akurasi serta tingkat *error* yang dihasilkan. Sehingga, proses *training* akan berlangsung dengan mengujikan beberapa nilai parameter *training* yang kemudian akan dibandingkan

nilai antara tiap parameter *training*. Selain dari hasil *training*, model yang dibangun juga akan diuji menggunakan *confusion matrix*. Penambahan *confusion matrix* akan mengukur bagaimana performansi dari model yang dibangun dengan *MobileNetV2*. Pembacaan nilai yang dilakukan dengan menggunakan *confusion matrix* juga dapat menguji model dengan parameter uji *accuracy*, *recall*, *preposition*, serta *f1-score* guna menambah keputusan sistem dalam identifikasi *input* wajah bermasker dan tidak bermasker

3.4.2. Metode Pengujian Sistem Berdasarkan Jarak

Untuk pengujian berdasarkan jarak metode yang digunakan adalah dengan pengujian di beberapa jarak antara webcam dengan wajah yang akan dideteksi. Jarak yang akan digunakan dalam pengujian ini yaitu 50 cm, 100 cm, 150 cm, 200 cm dan 250 cm. Untuk proses pengujian dari setiap jarak akan dilakukan beberapa kali pengujian dengan berkisar dua atau tiga kali pengujian guna mendapatkan hasil data yang akurat di tiap jarak. Dari jarak yang telah ditentukan tersebut akan dianalisa seberapa akurat sistem dapat bekerja di tiap jarak yang telah ditentukan, dan apakah semakin jauh jarak sistem masih dapat mendeteksi atau tidak.

3.4.3. Metode Pengujian Sistem Berdasarkan Cahaya

Pengujian sistem berdasarkan cahaya yang akan dilakukan adalah melakukan pengujian dalam ruangan gelap dan ruangan terang sehingga dapat mengetahui perbedaan di dua kondisi tersebut. Pengujian pada kondisi terang dan gelap akan diberi ukuran intensitas cahaya yakni dengan satuan lux, dimana untuk kondisi terang bernilai kurang lebih 450 lux dan kondisi gelap dibawah 100 lux. Dalam pengujian dengan kondisi gelap dan terang pemberian jarak juga tetap akan diberikan sehingga menambah analisa seberapa jauh sistem deteksi dapat bekerja di kondisi terang dan kondisi gelap.

3.4.4 Metode Pengujian Sistem Berdasarkan Jenis Masker

Pada pengujian ini sistem deteksi yang telah dirancang nantinya akan diujikan pada beberapa jenis masker. Jenis masker yang akan diuji merupakan jenis masker yang umum beredar di masyarakat seperti masker KF94, KN95 dan masker kain. Pengujian ini bertujuan untuk mengetahui kemampuan sistem deteksi dalam mendeteksi beberapa jenis masker yang bentuknya berbeda-beda.