

**SKRIPSI**

***REAL-TIME FACEMASK DETECTION* DALAM KERUMUNAN  
MENGUNAKAN *DRONE* SEBAGAI IMPLEMENTASI PROTOKOL  
KESEHATAN**

**Disusun dan diajukan oleh**

**FAUZAN ALIF ANWAR**

**D121171514**



**DEPARTEMEN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS HASANUDDIN**

**MAKASSAR**

**2022**

**LEMBAR PENGESAHAN SKRIPSI**  
**REAL-TIME FACEMASK DETECTION DALAM KERUMUNAN**  
**MENGGUNAKAN DRONE SEBAGAI IMPLEMENTASI PROTOKOL**  
**KESEHATAN**

**Disusun dan diajukan oleh**

**FAUZAN ALIF ANWAR**

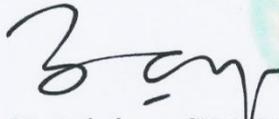
**D121171514**

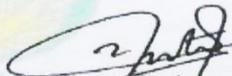
Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika Fakultas Teknik Universitas Hasanuddin pada tanggal 23 September 2022 dan dinyatakan telah memenuhi syarat kelulusan.

Menyetujui,

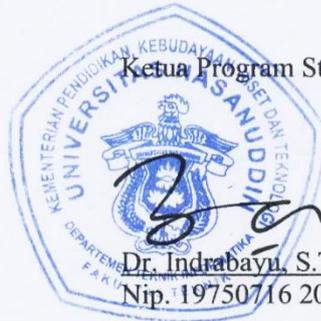
Pembimbing Utama,

Pembimbing Pendamping,

  
Dr. Indrabayu, ST., MT.  
Nip. 197507162002121004

  
Dr. Eng. Intan Sari Areni, ST., MT.  
Nip. 197502032000122002

Ketua Program Studi,

  
  
Dr. Indrabayu, S.T., M.T.  
Nip. 19750716 200212 1 004

## PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Fauzan Alif Anwar

NIM : D1211711514

Departemen : Teknik Informatika

Jenjang : S1

Menyatakan dengan ini karya tulisan saya berjudul:

### REAL-TIME FACEMASK DETECTION DALAM KERUMUNAN MENGUNAKAN DRONE SEBAGAI IMPLEMENTASI PROTOKOL KESEHATAN

Adalah karya tulisan saya sendiri dan bukan merupakan pengambilalihan tulisan orang lain bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Gowa, 25 September 2022

Yang menyatakan,  
  
Fauzan Alif Anwar

## KATA PENGANTAR

Assalaamu'alaykum Warahmatullahi Wabarokaatuh.

Segala puji dan syukur penulis panjatkan kepada Allah subhanahu wa ta'ala, Tuhan Yang Maha Esa atas berkat, izin, petunjuk, dan karunia-Nya sehingga tugas akhir yang berjudul "*Real-Time Facemask Detection Dalam Kerumunan Menggunakan Drone Sebagai Implementasi Protokol Kesehatan*" ini dapat diselesaikan sebagai salah satu syarat dalam menyelesaikan jenjang Starta-1 pada Departemen Teknik Informatika Fakultas Teknik, Universitas Hasanuddin.

Dalam penyusunan penelitian ini disajikan hasil penelitian terkait judul yang telah diangkat dan telah melalui proses pencarian dari berbagai sumber baik jurnal penelitian, prosiding pada seminar-seminar nasional/internasional, buku maupun situs-situs di internet.

Penulis menyadari bahwa dalam penyusunan dan penulisan laporan skripsi ini tidak lepas dari banyak bantuan, bimbingan serta dukungan dari berbagai pihak, dari masa perkuliahan sampai dengan masa penyusunan tugas akhir. Oleh karena itu, penulis dengan senang hati menyampaikan terima kasih kepada:

1. Kedua orang tua penulis, yakni ayahanda penulis Bapak dr. Anwar Umar.,M.Kes. dan ibunda Prof. Dr. Gusnawaty, M.Hum. yang telah mendukung sepenuhnya untuk kegiatan-kegiatan baik akademik maupun non akademik.
2. Bapak Dr. Indrabayu S.T., M.T., M.Bus.Sys. selaku Pembimbing I yang telah banyak memberi bimbingan, inspirasi, motivasi, masukan bermanfaat,

dan kesempatan agar penulis banyak melakukan inovasi yang dapat memberikan sumbangsih.

3. Ibu Dr. Eng. Intan Sari Areni, ST, MT. yang juga banyak memberikan bimbingan penulisan skripsi, memberi banyak keyakinan, perhatian, motivasi, dan masukan yang bermanfaat kepada penulis.
4. Nur Hazimah Alimuddin, SE yang selalu mendampingi, membantu, dan mementori penulisan skripsi.
5. Muhammad Zul Fahmi Sadrah, ST sebagai teman sejawat prodi yang tidak pernah bosan untuk dijadikan tempat konsultasi mengenai *source code* yang digunakan dalam penyelesaian tugas akhir ini serta yang selalu menjadi teman produktif semasa perkuliahan.
6. Irfan Ripat, Alfarabi Alif Putra, Irzam Kasyfillah, Muh Aries Sila, Amiruddin, Adhi Nugraha, Muhammad Fadhil, dan teman-teman sejawat produktif ruko, terima kasih karena telah memberikan banyak dorongan semangat.
7. Dosen-dosen yang saya hormati dan telah mengajarkan ilmu informatika selama masa perkuliahan tatap muka
8. Teman-teman RECOGN17ER yang selalu menjadi supportive system bayangan dalam pengerjaan skripsi, tanpa kalian, rasanya jadi hambar-hambar saja untuk menyelesaikan skripsi
9. Dan teman-teman lainnya, yang tidak dapat saya sebutkan satu persatu,  
Akhir kata, penulis berharap semoga segala bantuan mendapatkan berkah dari yang Maha Kuasa sehingga apa yang dilakukan dapat menjadi batu loncatan

untuk mencapai yang lebih baik pada inovasi yang akan datang, juga dapat memberikan manfaat bagi pengembangan ilmu selanjutnya. Aamiin.

Wassalaamu'alaykum Warahmatullahi Wabarokaatuh

Gowa, 11 Agustus 2022

Penulis

## ABSTRAK

Tahun 2020, penyebaran virus menjadi sorotan public karena penyebaran virus jenis baru bernama Covid-19. Penyebaran virus terbagi menjadi 2 macam, yaitu penyebaran langsung dan tak langsung. Dikarenakan penyebaran virus yang sulit dihindari dan penyebarannya tidak dapat dilihat dengan mata biasa, maka pemerintah Indonesia membuat kebijakan “Protokol Kesehatan”. Protokol Kesehatan adalah protocol untuk menaati 3M (Memakai masker, Menjaga jarak, dan Mencuci tangan) sehingga menjadi alasan penelitian ini untuk dilakukan menggunakan *drone* agar dapat membantu SDM prokes (polisi, kepala RT/RW, relawan *Covid Hunter*) dalam mengevaluasi daerah mana saja yang patuh menaati prokes. Penelitian ini memberikan alternatif penerapan Covid-19 selain CCTV karena sifat *drone* yang fleksibel (dapat diterbangkan) yang menggunakan sistem RTMP (*Real Time Messaging Protocol*) sehingga dapat memproses video yang di-*streaming* ke laptop SDM *Covid Hunter* secara *real-time*. Selain itu, sistem ini menggunakan metode *deep learning* MobileNet V2 agar model yang berjalan dapat lebih ringan dan menggunakan *fully connected layer* dari model yang telah di-*fine tuning*. Dengan adanya sistem ini, SDM *Covid Hunter* dapat mengambil informasi dan mengetahui secara lebih cepat daerah mana saja yang patuh menjalankan protokol kesehatan secara *real-time*. Data latih pada penelitian ini menggunakan data primer dan data sekunder. Untuk pengujian sistem, digunakan 4 skenario kondisi hadap wajah, 18 skenario kondisi jarak dengan sudut  $x$  (jarak horizontal *drone* ke objek wajah) mulai jarak 5 meter hingga 10 meter dan sudut  $y$  (jarak vertikal *drone* ketinggian dari atas tanah), serta mengambil 12 *frame* dari setiap skenario sehingga total data hasil yang digunakan berjumlah 216 *frame*. Hasil pengujian sistem menggunakan *Confusion Matrix* menunjukkan nilai rata-rata akurasi pada Skenario 1 (wajah hadap kiri) hingga 91%, Skenario 2 sebesar 88,38% (wajah hadap depan), Skenario 3 sebesar 88,22% (wajah hadap kanan) dan pada Skenario 4 hingga 76,56% (wajah hadap secara acak).

**Kata Kunci :** Covid 19, *drone*, *real-time*, klasifikasi masker, MobileNetV2, *fine tuning*, *confusion matrix*.

## DAFTAR ISI

HALAMAN SAMPUL .....	i
LEMBAR PENGESAHAN SKRIPSI .....	ii
KATA PENGANTAR .....	iv
ABSTRAK .....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR .....	xi
DAFTAR TABEL.....	xv
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	3
1.3. Tujuan Penelitian.....	3
1.4 Batasan Penelitian .....	3
1.5. Manfaat Penelitian.....	4
1.6. Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA.....	6
2.1 Pengertian Wajah .....	6
2.2 Pengertian Masker.....	8
2.3 Visi Komputer .....	10
2.4 <i>Machine Learning</i> .....	11
2.5 <i>Convolutional Neural Network (CNN)</i> .....	12
2.6 <i>Pooling Layer</i> .....	14
2.7 <i>Dataset</i> .....	15
2.8 Data Training.....	16
2.9 Citra Digital.....	18
2.10 Operasi Morfologi Citra .....	22
2.11 Citra <i>Gray-Scale</i> (Keabuan).....	24
2.12 Analisis Blob .....	26
2.13 <i>Video Streaming Digital</i> .....	27
2.14 Tensorflow.....	32

2.15	<i>Confusion Matrix</i> .....	32
2.16	<i>Unmanned Aerial Vehicle (UAV)</i> .....	35
2.17	Arsitektur MobileNet V2.....	36
2.18	<i>Real Time Messaging Protocol (RTMP)</i> .....	38
2.19	<i>Transfer Learning via Fine Tuning</i> .....	39
BAB III METODOLOGI PENELITIAN .....		40
3.1	Tempat dan Waktu Penelitian .....	40
3.2	Instrumen Penelitian .....	40
3.3	Tahapan Penelitian .....	41
3.3.1	Studi Literatur Terkait.....	42
3.3.2	Identifikasi Kebutuhan Penelitian .....	43
3.3.3	Perancangan Sistem Algoritma .....	43
3.3.4	Pembuatan dan Pengumpulan <i>Dataset facemask</i> .....	44
3.3.5	Meng- <i>import</i> RetinaFace sebagai <i>face detector</i> .....	49
3.3.6	<i>Preprocessing &amp; Export model facemask detector</i> .....	51
3.3.7	Validasi Sistem .....	63
3.3.8	Analisis Akurasi Sistem Deteksi <i>face-mask</i> .....	64
3.3.9	Penulisan Skripsi.....	64
3.4	Pengujian Algoritma ke <i>Drone</i> .....	65
3.4.1	<i>Drone</i> yang digunakan .....	65
3.4.2	Tahapan menggunakan sistem ke <i>drone</i> .....	66
3.4.3	Skenario Pengambilan Data Hasil.....	69
3.5	Perancangan Implementasi Sistem .....	72
3.6	Analisis Kerja Sistem .....	75
BAB IV HASIL DAN PEMBAHASAN .....		76
4.1	Hasil Penelitian.....	76
4.1.1	Hasil Operasi Data Citra .....	76
4.1.2	Hasil Pengujian Sistem Aplikasi .....	81
4.1.3	Hasil nilai akurasi validasi data menggunakan <i>Confusion Matrix</i> ..	82
4.1.4	Hasil nilai akurasi validasi data menggunakan <i>Confusion Matrix</i> dengan jarak objek wajah lebih dekat.....	101
4.1.5	Performa <i>Real-Time Streaming Drone</i> . .....	105

4.2 Pembahasan .....	106
BAB V.....	111
5.1 Kesimpulan.....	111
5.2 Saran .....	111
DAFTAR PUSTAKA .....	113
LAMPIRAN.....	118

## DAFTAR GAMBAR

<b>Gambar 2. 1</b> Ilustrasi landmark fitur deteksi wajah. ....	8
<b>Gambar 2. 2</b> Contoh masker dan jenisnya. ....	9
<b>Gambar 2. 3</b> Contoh Arsitektur MLP. ....	13
<b>Gambar 2. 4</b> Proses konvolusi pada CNN ....	14
<b>Gambar 2. 5</b> Perbedaan Max Pooling dan Average Pooling ....	15
<b>Gambar 2. 6</b> Contoh Dataset. ....	16
<b>Gambar 2. 7</b> Rumus matriks citra digital. ....	19
<b>Gambar 2. 8</b> Contoh Sistem Koordinat Citra Digital. ....	20
<b>Gambar 2. 9</b> Representasi Warna RGB pada Citra Digital. ....	21
<b>Gambar 2. 10</b> Representasi citra RGB dan kanal warna penyusunnya. ....	21
<b>Gambar 2. 11</b> Contoh konversi image RGB menjadi image gray-scale. ....	25
<b>Gambar 2. 12</b> Alur diagram Video Streaming. ....	29
<b>Gambar 2. 13</b> Contoh Drone ....	35
<b>Gambar 2. 14</b> Cara kerja deteksi MobileNet ....	37
<b>Gambar 2. 15</b> Konvolusi Depthwise (2-5) dan Pointwise (6-7) ....	37
<b>Gambar 3. 1</b> Tahapan penelitian. ....	42
<b>Gambar 3. 2</b> Contoh foto dari dataset primer. ....	45
<b>Gambar 3. 3</b> Skenario Pengambilan Dataset. ....	45
<b>Gambar 3. 4.</b> Skenario sudut hadap dataset objek wajah. (Sudut (A) bernilai 0°, Sudut (B) bernilai 45°, Sudut (C) bernilai 90°, Sudut (D) bernilai 135°, dan Sudut (E) bernilai 180°). ....	46
<b>Gambar 3. 5.</b> Hasil foto Pengambilan Data Primer. ....	47
<b>Gambar 3. 6.</b> Dataset sekunder tidak menggunakan masker. ....	47
<b>Gambar 3. 7.</b> Dataset sekunder yang telah disunting menggunakan masker. ....	48
<b>Gambar 3. 8.</b> 68 point landmark wajah (Sagonas, 2016). ....	48
<b>Gambar 3. 9.</b> Contoh gambar masker yang digunakan untuk augmentasi dataset sekunder yang digunakan. ....	49

<b>Gambar 3. 10.</b> Benchmark dataset WIDER FACE untuk berbagai oklusi deteksi wajah. ....	50
<b>Gambar 3. 11.</b> Blok Diagram Tahapan Preprocessing & Training dataset. ....	51
<b>Gambar 3. 12.</b> a. Gambar latih sebelum di-resizing (berukuran 6000x4000 px). b. Gambar latih setelah di-resizing.....	53
<b>Gambar 3. 13.</b> Contoh Proses Data Augmentation: a. citra original, b. Rotation, c. zoom-range, d. width-shift range, e. height-shift range, f. shear-range, g. horizontal flip, dan h. fill mode.....	54
<b>Gambar 3. 14.</b> Arsitektur MobileNetV2.....	55
<b>Gambar 3. 15.</b> Stride (perpindahan posisi) konvolusi ketika melakukan perkalian dengan filter / kernel. ....	56
<b>Gambar 3. 16.</b> Muatan blok Bottleneck dari model MobileNet V2 (Holleman, 2018) .....	57
<b>Gambar 3. 17.</b> Diagram Alur Feature Extraction oleh model MobileNet V2. ....	59
<b>Gambar 3. 18.</b> Alur Diagram Fully Connected Layer menggunakan metode Fine-tuning sebagai Localization dan Classification. ....	60
<b>Gambar 3. 19.</b> Hasil training dataset untuk model klasifikasi masker. ....	62
<b>Gambar 3. 20.</b> a. Menguji model pada citra dataset latih, b. Menguji model pada citra dataset validasi. ....	63
<b>Gambar 3. 21.</b> Contoh tabel Confusion Matrix. ....	64
<b>Gambar 3. 22.</b> Drone DJI Mavic Air yang digunakan dalam mendeteksi objek penelitian .....	66
<b>Gambar 3. 23.</b> Alur diagram menggunakan sistem di drone.....	66
<b>Gambar 3. 24.</b> Alur cara menggunakan RTMP ke drone .....	68
<b>Gambar 3. 25.</b> Ilustrasi Jarak Skenario Pengambilan Data Hasil (Titik biru adalah lokasi terbang drone).....	70
<b>Gambar 3. 26.</b> Contoh Point of View drone pada jarak (x) 5 meter dan tinggi drone dari atas tanah (y) 3 meter.....	71
<b>Gambar 3. 27.</b> (Skenario 1) Seluruh objek wajah menghadap kiri 90° terhadap drone.....	71

<b>Gambar 3. 28.</b> (Skenario 2) Seluruh objek wajah menghadap depan $0^{\circ}$ terhadap drone.....	72
<b>Gambar 3. 29.</b> (Skenario 3) Seluruh objek wajah menghadap kanan $-90^{\circ}$ terhadap drone.....	72
<b>Gambar 3. 30.</b> (Skenario 4) Objek wajah menghadap secara acak terhadap drone .....	72
<b>Gambar 3. 31.</b> (Tahap 1) Flowchart perancangan sistem latih dan (Tahap 2) Flowchart perancangan sistem uji. ....	73
<b>Gambar 4. 1.</b> Citra yang didapatkan secara real-time dari drone ke laptop. ....	76
<b>Gambar 4. 2.</b> Melakukan localization deteksi dengan mendeteksi wajah menggunakan RetinaFace. ....	77
<b>Gambar 4. 3.</b> Cropping Citra: a. Citra awal secara utuh. b. Citra frame yang telah di-crop. ....	78
<b>Gambar 4. 4.</b> Resize Citra Wajah: a. Citra wajah ukuran bervariasi. b. Citra ukuran $224 \times 224$ pixel.....	79
<b>Gambar 4. 5.</b> Objek Wajah yang terdeteksi bermasker di dalam frame secara real-time.....	80
<b>Gambar 4. 6.</b> Informasi Video streaming realtime yang disimpan untuk kemudian mengevaluasi Hasil Analisis Model.....	81
<b>Gambar 4. 7.</b> Grafik visualisasi level akurasi pada skenario 1 (objek deteksi menghadap kiri). ....	86
<b>Gambar 4. 8.</b> Citra / Gambar input klasifikasi masker. ....	87
<b>Gambar 4. 9.</b> Grafik visualisasi level akurasi pada skenario 2 (objek deteksi menghadap depan). ....	90
<b>Gambar 4. 10.</b> Citra / Gambar input klasifikasi masker, terlihat model mendeteksi tidak menggunakan masker, namun objek menggunakan masker (False Negative). ....	91
<b>Gambar 4. 11.</b> Grafik visualisasi level akurasi pada skenario 3 (objek deteksi menghadap kanan). ....	94

<b>Gambar 4. 12.</b> Citra / Gambar input klasifikasi masker, terlihat model mendeteksi tidak menggunakan masker, namun objek menggunakan masker (False Negative). .....	94
<b>Gambar 4. 13.</b> Grafik visualisasi level akurasi pada skenario 4 (objek deteksi menghadap secara acak).....	98
<b>Gambar 4. 14.</b> Contoh False Negative pada Skenario 4 di jarak (10,5) .....	99
<b>Gambar 4. 15.</b> Grafik visualisasi level akurasi pada skenario jarak (7,5). .....	102
<b>Gambar 4. 16.</b> Hasil deteksi dan klasifikasi wajah bermasker / tidak bermasker .....	103
<b>Gambar 4. 17.</b> Model deteksi wajah tidak mendeteksi parameter wajah pada 4 object wajah, sehingga tidak ada hasil yang ditampilkan dari model klasifikasi masker .....	109

## DAFTAR TABEL

<b>Tabel 2. 1</b> Contoh Hasil Model Training CNN .....	17
<b>Tabel 2. 2</b> Ukuran Parameter dan Komputasi pada Depthwise Separable Convolution.....	38
<b>Tabel 3. 1.</b> Parameter pipeline training. ....	51
<b>Tabel 4. 1.</b> Jarak aktual kamera drone ke objek wajah.....	83
<b>Tabel 4. 2.</b> Nilai Akurasi Sistem pada Skenario 1 (wajah menghadap kiri). ....	85
<b>Tabel 4. 3.</b> Performa confusion matrix skenario 1 (hadap kiri).....	87
<b>Tabel 4. 4.</b> Nilai Akurasi Sistem pada Skenario 2 (wajah menghadap depan). ...	89
<b>Tabel 4. 5.</b> Performa confusion matrix skenario 2 (hadap depan).....	91
<b>Tabel 4. 6.</b> Nilai Akurasi Sistem pada Skenario 3 (wajah menghadap kanan). ...	93
<b>Tabel 4. 7.</b> Performa confusion matrix skenario 3 (hadap kanan).....	95
<b>Tabel 4. 8.</b> Nilai Akurasi Sistem pada Skenario 4 (wajah menghadap secara acak). .....	97
<b>Tabel 4. 9.</b> Performa confusion matrix skenario 4 (hadap secara acak).....	99
<b>Tabel 4. 10.</b> Nilai akurasi sistem pada skenario (7,5) dengan wajah saling didekatkan. ....	102
<b>Tabel 4. 11.</b> Nilai precision sistem pada skenario (7,5) dengan wajah saling didekatkan. ....	104
<b>Tabel 4. 12.</b> Nilai recall sistem pada skenario (7,5) dengan wajah saling didekatkan. ....	104
<b>Tabel 4. 13.</b> Nilai F1-Score sistem pada skenario (7,5) dengan wajah saling didekatkan. ....	104
<b>Tabel 4. 14.</b> Perbandingan performa aplikasi ketika menggunakan / tidak menggunakan threading. ....	105
<b>Tabel 4. 15.</b> Rangkuman nilai akurasi berdasarkan skenario jarak dan skenario hadap wajah.....	107

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Virus merupakan mikroorganisme patogen yang hidup hanya pada sel makhluk hidup lainnya dengan cara bereplikasi dikarenakan virus tidak dapat hidup sendiri, dan virus juga tidak memiliki perlengkapan seluler untuk bereproduksi sendiri (Suprobowati & Kurniati, 2018). Media penyebaran virus pun beragam, dapat melalui pertukaran udara, pertukaran cairan, ataupun media-media lainnya seperti sentuhan satu sama lain, yang dapat mengakibatkan perpindahan virus dari sel terinfeksi ke sel lainnya yang masih sehat. Karena ukurannya yang sangat kecil, penyebaran virus antar makhluk ke makhluk lainnya dapat dikatakan sukar untuk dihindari karena gaya hidup yang mengharuskan masyarakat mesti berinteraksi satu sama lainnya agar kebutuhan bisa terpenuhi (Suprobowati & Kurniati, 2018).

Tahun 2020, Penyebaran virus kembali menjadi sorotan publik karena penyebaran virus jenis baru bernama virus Covid-19. Penyebaran virus terbagi menjadi 2 macam, yaitu penyebaran secara langsung dan penyebaran secara tak langsung. Dikarenakan penyebaran virus yang sulit untuk dihindari dan penyebarannya tidak dapat dilihat dengan mata biasa, maka pemerintah Indonesia membuat kebijakan baru yaitu “Protokol Kesehatan”.

Protokol Kesehatan adalah protokol yang mewajibkan warga negara dalam menaati 3M (Memakai Masker, Menjaga Jarak, dan Mencuci Tangan). Protokol Kesehatan (Prokes) wajib ditaati dan dilakukan oleh setiap warga negara agar

penyebaran virusnya dapat diminimalisir. Pada penerapannya, prokes ini diharapkan dapat berjalan secara efektif untuk memutuskan rantai penyebaran virus. Namun pada penerapannya di beberapa daerah / wilayah, Prokes ini belum berjalan secara efektif seperti yang diharapkan karena masih kurangnya pengawasan dalam penerapan prokes tersebut dan masih kurangnya wawasan masyarakat dalam menerapkan prokes secara baik dan benar (seperti bagaimana menggunakan masker yang tepat). Ini berdasarkan dari salah satu kutipan Epidemiolog UI yang mengatakan “Pengetatan (Prokes) yang diinstruksikan oleh Menteri Luhut tidak akan efektif. Sebab, saat ini yang paling penting adalah bagaimana pengawasannya” (Yunis, 2020) dan kutipan dari salah satu website yaitu “Protokol Kesehatan Covid-19 yang utama adalah menjaga jarak dengan orang lain, mencuci tangan menggunakan sabun dengan air mengalir, serta menggunakan masker. Nah, nyatanya masih ada banyak orang yang salah menggunakan masker” (Biananda, 2020) selanjutnya data yang menunjukkan peringkat Indonesia pada Q3 tahun 2020, protocol Kesehatan dinilai kurang efektif karena Indonesia menduduki peringkat 97 dari 100 negara dalam *Covid-19 Regional Safety Assessment* (Ahdiat, 2020) yang menunjukkan penetapan prokes adalah penetapan / protokol yang sudah benar dan dibutuhkan, namun masih kurang maksimal dalam penerapannya, terkhususnya dalam menggunakan masker.

Berdasarkan alasan diatas, dibutuhkan penelitian suatu bantuan teknologi *up-to-date* yang dapat memaksimalkan salah satu peraturan penerapan protokol Kesehatan (penggunaan masker), dengan memanfaatkan teknologi *Artificial Intelligence* (AI) teknologi terkini yang diimplementasikan ke dalam *drone* yang

dapat mendeteksi *real-time* wajah-wajah yang menggunakan masker. dan pendeteksian dapat dilaksanakan lebih fleksibel. Dengan demikian teknologi ini dapat membantu pemerintah ataupun SDM pengawas daerah tertentu dalam menentukan Kawasan atau daerah mana yang membutuhkan pengawasan lebih untuk menerapkan protokol kesehatan.

### **1.2. Rumusan Masalah**

Berdasarkan latar belakang yang telah diuraikan di atas, rumusan masalah pada penelitian ini adalah:

1. Bagaimana alur sistem deteksi wajah bermasker atau tidak meskipun berada di kerumunan secara *real-time* menggunakan *drone*.
2. Bagaimana performa *streaming* deteksi wajah bermasker atau tidak secara *real-time* menggunakan *drone*.

### **1.3. Tujuan Penelitian**

Dalam melakukan penelitian ini, tujuan yang ingin dicapai adalah:

1. Mengetahui alur sistem deteksi wajah bermasker atau tidak meskipun berada di kerumunan secara *real-time* menggunakan *drone*.
2. Mengetahui performa *streaming* deteksi wajah bermasker atau tidak secara *real-time* menggunakan *drone*.

### **1.4 Batasan Penelitian**

Adapun batasan-batasan pada penelitian ini adalah:

1. Wajah deteksi dibatasi untuk menghadap ke kiri, depan, kanan, dan menghadap secara acak dianalisis menggunakan masker atau tidak
2. Parameter kerumunan yang digunakan berjumlah maksimal 5 orang.

3. Wajah tidak saling tertutup oleh objek yang lainnya, sehingga wajah harus terlihat secara utuh (tidak *overlapped*).
4. Oklusi warna dan variasi masker tidak digunakan sebagai acuan pengambilan data hasil.
5. *Angle* kamera *drone* dan *Angle* wajah objek tidak menjadi variable dalam penelitian.
6. Evaluasi akhir hanya berdasarkan pada objek wajah yang berhasil terdeteksi oleh model.
7. Intensitas cahaya juga tidak menjadi variabel, karena pengambilan data hasil dikondisikan sekitar siang dan sore hari.

#### **1.5. Manfaat Penelitian**

Hasil dari penelitian ini akan menjadi pengetahuan (*knowledge*) mengenai implementasi AI (*Artificial Intelligence*) pada *drone* yang dapat mendeteksi & mengklasifikasi wajah bermasker. Serta hasil penelitian dapat menjadi referensi bagi instansi yang terkait dengan prokes, dalam membuat keputusan daerah mana saja yang memerlukan penetapan prokes lebih ketat pada pengawasan penerapan.

#### **1.6. Sistematika Penulisan**

Untuk mendapatkan gambaran umum mengenai narasi penelitian secara keseluruhan, maka bab pada penelitian yang dilakukan diuraikan menjadi beberapa tahapan dari penulisan secara sistematis, yaitu:

### **BAB 1 PENDAHULUAN**

Bab pendahuluan menguraikan secara umum mengenai hal yang menyangkut latar belakang, perumusan masalah, tujuan, batasan penelitian, manfaat, dan sistematika penulisan.

## **BAB II TINJAUAN PUSTAKA**

Bab tinjauan pustaka berisi teori-teori dasar dan penelitian-penelitian terkait dari berbagai literatur tentang hal-hal yang berhubungan dengan *facemask detection*, serta penguraian metode, algoritma, dan arsitektur (*MobileNet V2*) yang digunakan.

## **BAB III METODOLOGI PENELITIAN**

Bab metodologi penelitian berisikan tentang perencanaan alur diagram sistem, metode yang digunakan, dan penerapan algoritma serta teknik *facemask detection*.

## **BAB IV HASIL DAN PEMBAHASAN**

Bab hasil dan pembahasan berisikan hasil pendeteksian *facemask* menggunakan *drone* secara *real-time*, hasil validasi deteksi menggunakan *Confusion Matrix*, serta pembahasan terkait hasil penelitian

## **BAB V PENUTUP**

Bab penutup berisi tentang kesimpulan yang diperoleh berdasarkan hasil penelitian yang telah dilakukan dan diselesaikan serta saran untuk pengembangan sistem lebih lanjut.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Pengertian Wajah**

Menurut pengertian Kamus Besar Bahasa Indonesia (KBBI), wajah adalah bagian depan dari kepala (KBBI, 2022). Lebih lanjutnya, Wajah adalah organ pusat untuk ekspresi, pengenalan, dan komunikasi manusia. Wajah terdiri dari empat organ perasa yang sangat penting, yaitu hidung, mata, telinga, dan lidah. Pada tubuh manusia, wajah berada di bagian anterior (depan) kepala dan memanjang dari dahi hingga ke dagu. Bentuk dan rupa wajah dinilai berdasarkan struktur tulang dan otot wajah. Bagian-bagian wajah meliputi beberapa bagian, diantaranya:

- Dahi, kulit yang berada tepat dibawah batas tumbuh rambut dan berbatasan dengan alis mata.
- Mata, berada di dalam rongga mata dan dilindungi oleh bulu dan kelopak mata.
- Hidung, termasuk lubang hidung dan septum.
- Pipi, bagian yang melindungi rahang dan tulang maxilla
- Mulut, bagian dimana dapat ditemukan lidah dan gigi.

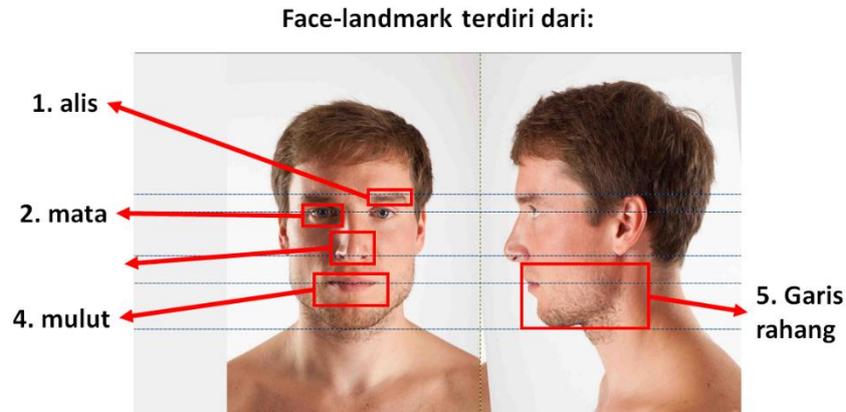
Wajah juga memiliki simpul kulit unik yang akan bereaksi pada berbagai rangsangan melalui jaringan ujung saraf yang luas. Wajah juga merupakan bagian tubuh utama yang menjalankan proses melihat, makan, dan berbicara (Docdoc, n.d.)

Dalam ilmu Kesehatan wajah dijelaskan berdasarkan karakteristik anatominya merupakan bagian *anterior* (struktur bagian depan) dari kepala, dengan batas kedua telinga di *lateral* (struktur terjauh dari garis pertengahan tubuh), dagu

di *inferior* (bagian terendah) dan garis batas tumbuhnya rambut di *superior* (bagian teratas). Wajah merupakan bagian dari tubuh manusia yang bisa dikenali dengan teknologi biometric untuk mengenal suatu individu tertentu. Pada teknologi biometric ciri-ciri biologis dari manusia seperti wajah bisa memberikan informasi yang unik. Pada wajah sendiri terdapat 66 fitur titik yang dapat menjadi ciri khusus pada suatu wajah (Kalansuriya, T. R., & Dharmaratne, 2013).

Terdapat beberapa perbedaan bentuk fitur wajah antara laki-laki dan perempuan. Di antaranya seperti mata laki-laki memiliki ukuran mata yang kecil dan jarak mata dengan alis lebih dekat, sedangkan mata perempuan memiliki ukuran mata yang lebih besar dan jarak mata dengan alis lebih besar. Sedangkan pada fitur mulut / bibir, mulut laki-laki memiliki jarak dasar hidung ke bibir atas lebih jauh dan ukuran yang lebih tipis. Sedangkan pada mulut perempuan jarak dasar hidung ke bibir atas lebih sedikit dan ukurannya lebih tebal (Wulansari et al., 2017). Bentuk hidung laki-laki umumnya lebih besar dari bentuk hidung perempuan, bentuk wajah laki-laki lebih kotak, sedangkan perempuan membentuk wajah berbentuk hati serta bentuk dagu laki-laki lebih melebar, sedangkan dagu perempuan lebih membulat (Kalansuriya, T. R., & Dharmaratne, 2015)

Diperlukan titik *landmark* wajah seperti mata, hidung, mulut, dan fitur wajah lainnya karena mengenali *landmark* wajah merupakan proses penting untuk mendeteksi wajah. Secara praktek subsistem ini sangat menentukan kualitas dari sistem pengenalan wajah (Lim, 2003). Dalam praktikal pendeteksian wajah, algoritma *Machine Learning* juga akan mengunci fitur wajah tersebut, diantaranya mata, alis, hidung, mulut, dan rahang seperti yang ditunjukkan pada Gambar 2.1.



**Gambar 2. 1** Ilustrasi landmark fitur deteksi wajah.

## 2.2 Pengertian Masker

Menurut pengertian Kamus Besar Bahasa Indonesia (KBBI), masker adalah alat untuk menutup muka / kain penutup mulut dan hidung (seperti yang dipakai oleh dokter, perawat di rumah sakit). Masker adalah perlindungan pernafasan yang digunakan sebagai metode untuk melindungi individu dari menghirup zat-zat bahaya atau kontaminan yang berada di udara, perlindungan pernafasan atau masker tidak dimaksudkan untuk menggantikan metode pilihan yang dapat menghilangkan penyakit, tetapi digunakan untuk melindungi secara memadai pemakainya (Cohen & Jacob, 2012). Masker secara luas digunakan untuk memberikan perlindungan terhadap partikel dan *aerosol* yang dapat menyebabkan bahaya bagi system pernafasan yang dihadapi oleh orang yang tidak memakai alat pelindung diri, bahaya partikel dan *aerosol* dari berbagai ukuran dan sifat kimia yang berbeda dapat membahayakan manusia, maka NIOSH merekomendasikan masker yang menggunakan filter (Eshbaugh et al., 2009).

Masker memiliki banyak tipe, salah satunya adalah *Air Purifying Respirators* (API), masker ini menggunakan filter atau *catridge* yang dapat

mencegah zat-zat berbahaya yang berada di udara (Mayes & Bender, 2012). Masker mempunyai jenis yang dapat melindungi tergantung dengan tingkat bahaya dari paparan *aerosol* atau partikel bahaya yang berada di udara. Menurut (Cohen & Jacob, 2012) jenis masker mempunyai jenis sebagai berikut yang juga ditunjukkan pada Gambar 2.2.

- *Quarter mask* adalah sebuah respirator yang meliputi hidung dan mulut dengan penutup wajah memanjang dari atas hidung sampai bawah mulut, masker ini biasanya digunakan untuk perlindungan terhadap bahaya partikel yang rendah.
- *Half mask* adalah sebuah respirator setengah topeng yang menutupi hidung dan mulut dengan penutup wajah yang memanjang dari atas (Wahyuni, 2018) hidung hingga ke bawah dagu, masker ini digunakan untuk semua jenis bahaya, termasuk partikel, uap, dan gas yang dapat membahayakan pemakainya.
- *Full facepiece* adalah sebuah respirator dengan penuh penutup wajah yang mencakup seluruh kepala, masker ini biasanya digunakan pada partikel, *aerosol* dan gas yang dapat mengiritasi mata.



**Gambar 2. 2** Contoh masker dan jenisnya.

Sumber: (Cohen & Jacob, 2012)

### 2.3 Visi Komputer

Visi Komputer merupakan sub disiplin ilmu dari kecerdasan buatan yang mempelajari bagaimana mesin dapat mengenali objek yang diamati atau diobservasi. Dapat disebut juga sebagai *Machine Vision*, ilmu ini mengembangkan teori-teori dan algoritma dimana informasi yang berguna diekstraksi dan dianalisis dari sebuah citra penelitian, sekumpulan citra, atau citra yang berurutan dari sebuah komputasi yang dibuat oleh komputer.

Lebih lanjut, definisi visi komputer adalah ilmu dan teknologi mesin yang dapat melihat, sehingga mesin mampu mengekstrak informasi dari gambar yang diperlukan untuk menyelesaikan tugas tertentu. Sebagai suatu disiplin ilmu, visi computer berkaitan dengan teori di balik sistem buatan bahwa ekstrak informasi dari gambar. Data gambar dapat mengambil banyak bentuk, seperti urutan video pandangan dari beberapa kamera, atau data multi-dimensi dari *scanner* medis. Sebagai disiplin teknologi, visi komputer berusaha untuk menerapkan teori dan model untuk pembangunan system (Mario, 2013).

Visi komputer memiliki banyak manfaat dan implementasi yang dapat dilakukan, 5 diantaranya yakni sebagai berikut:

- *Cognitive Vision*
- *Computational Intelligence*
- *Robot Vision*
- *Smart Cameras*
- *Optics*

Boyle dan Thomas mengatakan bahwa *Computer Vision* lebih daripada pengenalan, *Computer Vision* melakukan operasi “*Low level processing*” sebagai algoritma *Image Processing* yang murni. Mereka juga yang menggolongkan *image processing* ke dalam *Computer Vision* (Boyle & Thomas, 1988).

## **2.4 *Machine Learning***

*Machine Learning* adalah salah satu bidang ilmu komputer yang memberikan kemampuan pembelajaran kepada komputer untuk mengetahui sesuatu tanpa pemrogram yang jelas. Istilah *Machine Learning* pada dasarnya adalah proses komputer untuk belajar dari data. Tanpa adanya data, komputer tidak akan bisa belajar apa-apa. Semua pengetahuan *Machine Learning* pasti akan melibatkan data. Data bisa saja sama, akan tetapi algoritma dan pendekatannya berbeda-beda untuk mendapatkan hasil yang optimal (Sindy, 2019)

*Machine Learning* merupakan salah satu cabang dari disiplin ilmu kecerdasan buatan (*Artificial Intelligence*) yang membahas mengenai pembangunan sistem yang berdasarkan pada data. Dalam pembelajaran *Machine Learning*, terdapat beberapa skenario-skenario seperti berikut.

### *1. Supervised Learning.*

Penggunaan skenario *Supervised Learning* adalah pembelajaran menggunakan masukan data pembelajaran yang telah diberi label. Setelah itu membuat prediksi dari data yang telah diberi label.

### *2. Unsupervised Learning.*

Penggunaan skenario *Unsupervised Learning* yakni pembelajaran menggunakan masukan data pembelajaran yang tidak diberi label. Setelah itu

mencoba untuk mengelompokkan data berdasarkan karakteristik-karakteristik yang ditemukan.

### 3. *Reinforcement Learning*.

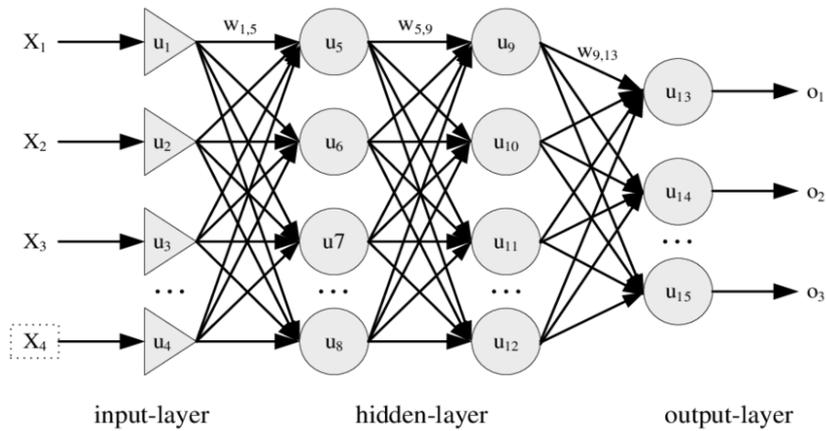
Pada penggunaan skenario *Reinforcement Learning* adalah fase pembelajaran dan tes saling dicampur. Untuk mengumpulkan informasi pembelajar secara aktif dengan berinteraksi ke lingkungan sehingga untuk mendapatkan balasan untuk setiap aksi dari pembelajar (Sindy, 2019)

Penelitian ini berdasarkan dari *Machine Learning* yang menggunakan *Supervised Learning* sehingga penelitian ini memiliki *dataset* yang akan memiliki label.

## 2.5 *Convolutional Neural Network (CNN)*

*Convolutional Neural Network* adalah pengembangan dari *Multilayer Perceptron* (MLP) yang didesain untuk mengolah data 2 dimensi. CNN termasuk dalam jenis *Deep Neural Network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra. Pada kasus klasifikasi citra, MLP kurang sesuai untuk digunakan karena tidak menyimpan informasi spasial dari data citra dan menganggap setiap piksel adalah fitur yang independen sehingga menghasilkan hasil yang kurang baik (Suartika E. P et al., 2016).

CNN memiliki kesamaan konsep pada MLP, namun dalam CNN setiap neuron dipresentasikan dalam bentuk 2 dimensi, tidak seperti MLP yang setiap neuron hanya berukuran 1 dimensi.



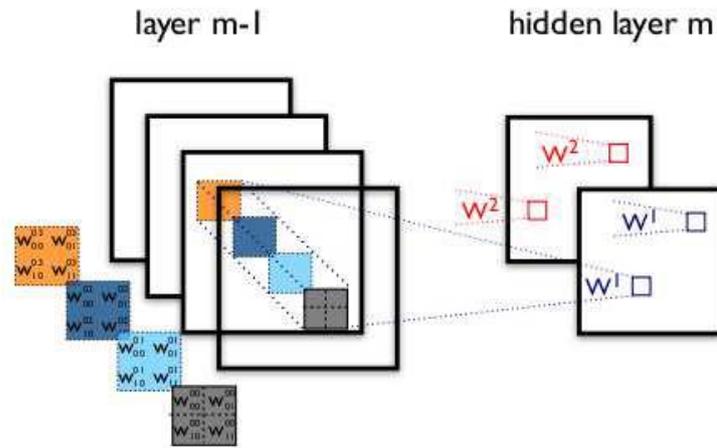
**Gambar 2. 3** Contoh Arsitektur MLP.

Sumber : (Wahyunggoro et al., 2013)

Sebuah MLP seperti pada Gambar 2.4 memiliki  $i$  layer (kotak tidak terlihat yang memblok input dan hidden layer) dengan masing-masing layer berisi  $j$  neuron (lingkaran putih). MLP menerima input data satu dimensi dan mempropagasikan data tersebut pada jaringan hingga menghasilkan output. Pada CNN, data yang dipropagasikan pada jaringan adalah data 2 dimensi, sehingga operasi *linear* dan parameter bobot pada CNN berbeda. Pada CNN operasi *linear* menggunakan operasi konvolusi, sedangkan bobot tidak lagi 1 dimensi saja, namun berbentuk empat dimensi yang merupakan kumpulan kernel konvolusi seperti pada Gambar 2.4 diatas, dimensi bobot pada CNN adalah:

$$neuron\ input \times neuron\ output \times tinggi \times lebar \quad 2.1$$

Mengingat sifat proses konvolusi, maka CNN hanya dapat digunakan pada data yang memiliki struktur 2 dimensi seperti citra dan suara (Suartika E. P et al., 2016). Berikut ilustrasi proses konvolusi CNN pada Gambar 2.5.



**Gambar 2. 4** Proses konvolusi pada CNN

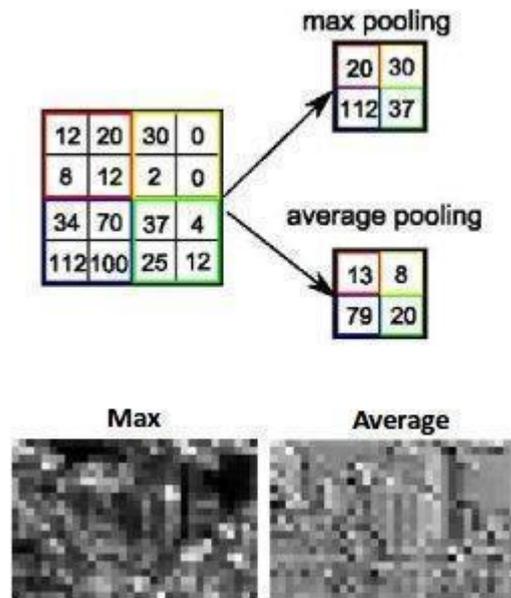
Sumber: (Suartika E. P et al., 2016)

## 2.6 Pooling Layer

*Pooling layer* dilakukan untuk mengurangi variasi-variasi, mengurangi kerumitan perhitungan dan ekstrak fitur tingkat rendah dari lingkungan sekitar. *Pooling Layer* terdapat 2 jenis yaitu: *max pooling* dan *average pooling*. *Max pooling* mengekstrak fitur yang paling penting seperti tepi sedangkan *average pooling* lebih halus dibandingkan dengan *max pooling* (Rahman, 2017)

Meskipun keduanya digunakan untuk alasan yang sama, *max pooling* lebih baik untuk mengekstrak fitur ekstrem. *Average pooling* terkadang tidak dapat mengekstrak dan menghasilkan nilai rata-rata yang mungkin atau tidak penting untuk deteksi objek. *Average pooling* membawa semua ke dalam hitungan dan mengalirkannya kelapisan berikutnya yang berarti semua nilai benar-benar digunakan untuk pemetaan fitur dan pembuatan keluaran yang merupakan perhitungan yang sangat umum. Jika tidak diperlukan semua masukan dari lapisan konvolusi maka akan mendapatkan akurasi yang buruk untuk *average pooling*.

Pada klasifikasi objek *pooling layer* yang banyak digunakan adalah *average pooling*. Pemilihan *pooling layer* tergantung pada jenis *dataset* yang digunakan (Rahman, 2017). Berikut ilustrasi perbedaan *max pooling* dan *average pooling* pada Gambar 2.6.



**Gambar 2. 5** Perbedaan Max Pooling dan Average Pooling

Sumber: (Rahman, 2017)

## 2.7 Dataset

*Dataset* adalah kumpulan data atau kumpulan parameter mengenai objek tertentu. Lebih jelasnya, dalam proses pelatihan komputer dalam mengenali suatu *objek*, dibutuhkan kumpulan parameter yang dapat dijadikan patokan dalam mengenali suatu objek tersebut. Kemudian, suatu kumpulan parameter yang bisa digunakan sebagai bahan pengenalan komputer terhadap objek tertentu tadi, membentuk parameter baru. Contohnya jika menginginkan untuk mengenali bentuk sapi Zebu (berpunuk sendiri), maka dibutuhkan setidaknya beberapa foto sapi yang

dikumpulkan menjadi 1 folder, dan diperlihatkan bahwa inilah bentuk dari sapi Zebu.

	A	B	C	D	E	F	G
1	ID	Age	Gender	District	SATV	SATM	GPA
2	54419	18	M	38	368	253	3.52
3	62516	22	M	5	670	496	1.11
4	55509	21	F	54	639	439	2.68
5	36489	19	M	49	368	465	3.11
6	36387	21	F	36	620	306	2.16
7	95507	20	F	13	512	593	2.83
8	16360	20	M	52	621	377	2.79
9	12838	18	F	44	571	544	2.13
10	73450	20	F	59	647	746	2.08
11	26869	18	F	28	337	371	2.28
12	48552	22	M	63	260	498	3.24
13	23416	19	M	51	476	294	2.31
14	42635	19	F	35	677	241	3.19
15	67448	19	F	55	335	533	1.81
16	34689	21	F	42	585	708	1.80
17	32763	22	F	20	556	787	1.18

**Gambar 2. 6** Contoh *Dataset*.

Sumber: (David, n.d.)

*Dataset* juga dapat disebut sebagai *Database*, sehingga ini juga merupakan sekumpulan data yang tersusun dan tersimpan rapi dalam komputer, dan dapat diolah maupun dimanipulasi dengan menggunakan *software* (perangkat lunak) sekehendak penggunaannya untuk dijadikan informasi tertentu. *Database* adalah kumpulan tabel, hubungan (relasi-relasi), dan lain-lain yang berkaitan dengan penyimpanan data.

## 2.8 Data Training

*Data training* adalah data yang digunakan untuk perhitungan probabilitas dari data berdasarkan data pembelajaran yang dilakukan (Pratiwi et al., 2016). Data ini dikenal sebagai data pelatihan / pembelajaran yang memungkinkan algoritma untuk mengenali pola-pola yang membedakan satu kelas dengan yang lain, guna mencari model yang paling tepat. Apabila training data tidak mampu

merepresentasikan populasi, maka model yang dihasilkan pembelajaran (training) tidak bagus (Putra, 2020).

Dalam implementasinya, *data training* dipakai pada beberapa *framework* yang melakukan pelatihan data, salah satunya yaitu Tensorflow. Tensorflow adalah pustaka sumber terbuka yang digunakan untuk membuat, mengembangkan, dan melakukan data training pada model *machine learning*. *Machine learning* adalah ilmu komputer untuk membantu dan memecahkan masalah secara otomatis (Sumit Das et al., 2015)

Selain itu, *deep learning* juga memanfaatkan data training dalam implementasinya. Salah satu metode deep learning yang memanfaatkan data training yaitu CNN (*convolutional neural network*). Algoritma ini biasa digunakan untuk keperluan pendeteksian objek. Secara teknis, *convolutional network* merupakan arsitektur yang bisa *training* dari beberapa tahap. Masukan dan keluaran dari masing-masing tahapan adalah *array* yang disebut *feature map*, hasil pengolahan dari semua lokasi pada input. Perhitungan *input* kedalam konvolusional menggunakan rumus berikut.

$$input\_size + 2 * padding - (filter\_size - 1) \quad 2.2$$

Sumber : (Anggraini, 2020)

Setelah melakukan tahapan pemodelan CNN lainnya seperti *convolutional*, *pooling*, *flatten* hingga mendapatkan model jaringan, dilakukan pengujian nilai akurasi untuk mendapatkan hasil terbaik.

**Tabel 2. 1** Contoh Hasil Model Training CNN

Class	Loss	Val-Loss	Accuracy	Val-Accuracy
Class_1	0,3976	0,4143	0,8204	0,8538
Class_2	0,3845	0,3855	0,8588	0,8626
Class_3	0,3513	0,3741	0,8705	0,8607
Class_4	0,3287	0,3596	0,8793	0,8719
Class_6	0,3153	0,3361	0,8825	0,8782

Sumber : (Anggraini, 2020)

Baik *machine learning* maupun *deep learning*, keduanya mengacu pada pengembangan sistem yang mampu belajar dengan menyediakan data sebagai bahan belajar (*training*) sebelum mengeluarkan *output* dan akurasi data. Lebih banyak jumlah data pelatihan daripada data pengujian dan nilai ambang batas yang lebih tinggi, memberikan hasil yang lebih baik dalam akurasi (Sumit Das et al., 2015)

## 2.9 Citra Digital

Citra digital adalah gambar 2 dimensi yang bisa ditampilkan pada layar komputer sebagai himpunan / diskrit nilai digital yang disebut *pixel/picture elements*. Dalam tinjauan secara matematis, citra merupakan fungsi kontinu dan intensitas cahaya dalam bidang 2 dimensi. lebih lanjut, citra digital adalah sebuah matriks indeks baris maupun kolomnya menyatakan sebuah titik pada citra tersebut dan elemen matriksnya (yang dikenal sebagai elemen gambar / *picture element* / *pixel*) menyatakan tingkat keabuan di titik tersebut.

Citra digital merupakan citra  $f(x,y)$  dimana dilakukan diskritisasi koordinat spasial / sampling dan diskritisasi tingkat kwantisasi (Kecemerlangan/keabuannya). Citra digital adalah suatu fungsi intensitas cahaya  $f(x,y)$  yang nilai  $x$  dan nilai  $y$  adalah koordinat spasial. Nilai fungsi tersebut di setiap titik  $(x,y)$  adalah tingkat kecermerlangan citra di titik tersebut.

Jadi, informasi yang terkandung di dalamnya bersifat diskrit. Citra digital tidaklah selalu merupakan hasil langsung dari data rekaman sebuah sistem. Tetapi terkadang merupakan hasil rekaman data yang sifatnya kontinyu seperti gambar pada monitor tv, foto pada sinar-x, dan lain-lain. Dengan begitu untuk memperoleh suatu citra digital dibutuhkan sebuah proses konversi, sehingga selanjutnya citra tersebut bisa diproses menggunakan komputer.

Matriks yang dinyatakan untuk citra digital adalah matriks berukuran  $N$  (baris/tinggi)  $\times$   $M$  (kolom/lebar). Lebih lanjutnya dapat dilihat pada Gambar 2.8

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & f(1,1) & \dots & f(1,M-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

**Gambar 2. 7** Rumus matriks citra digital.

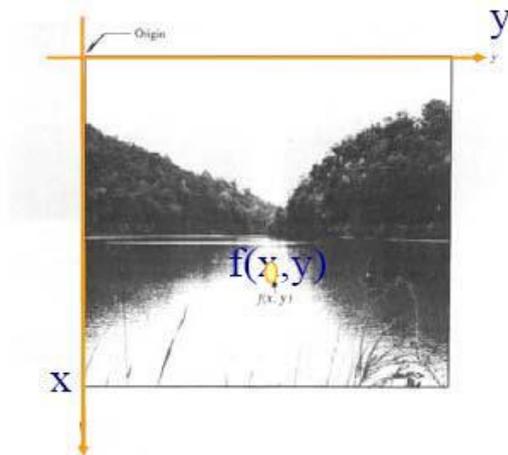
Keterangan :

$$N = \text{jumlah baris} \quad 0 = y = N-1 \quad 2.3$$

$$M = \text{jumlah kolom} \quad 0 = x = M-1$$

$L =$  maksimal warna intensitas  $O = f(x,y) = L-1$  (gray level/derajat keabuan). 2.4

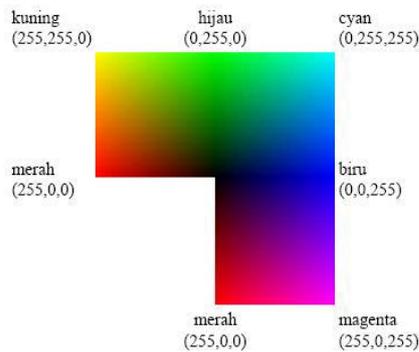
Teknologi dasar untuk menciptakan dan menampilkan warna pada citra digital berdasarkan pada penelitian bahwa sebuah warna merupakan kombinasi dari tiga warna dasar, yaitu merah, hijau, dan biru (*Red, Green, Blue - RGB*). Sistem koordinat pada sebuah citra digital dapat dilihat pada Gambar 2.9.



**Gambar 2. 8** Contoh Sistem Koordinat Citra Digital.

Citra digital ini digunakan untuk mengolah dan memproses sebuah gambar / video penjabaran tiap warna pada vector  $(x,y,z)$  dengan  $(r,g,b)$ , sehingga contoh dari koordinat sebuah jenis warna dapat dituliskan sebagai berikut: warna = RGB (30, 75, 255). Putih = RGB (255,255,255), sedangkan untuk hitam = RGB (0,0,0).

Bentuk Representasi warna dari sebuah citra digital dapat dilihat pada Gambar 2.10.



**Gambar 2. 9** Representasi Warna RGB pada Citra Digital.

Misalnya terdapat gambar berukuran  $100px \times 100px$  dengan *color encoding* 24 bit dengan  $R = 8$  bit,  $G = 8$  bit,  $B = 8$  bit, maka *color encoding* akan mampu mewakili  $0 \dots 16.777.215$  warna (mewakili hingga 16 juta warna), dan ruang *disk* yang dibutuhkan  $= 100 \times 100 \times 3$  bit (karena RGB)  $= 30.000$  bit  $= 30$  KB atau  $100 \times 100 \times 24$  bit  $= 240.000$  bit  $= 240$ KB (Yarso, 2013). Ilustrasi representasi citra RGB dapat dilihat pada Gambar 2.11.



**Gambar 2. 10** Representasi citra RGB dan kanal warna penyusunnya.

Citra digital dapat dibagi menjadi beberapa kategori, diantaranya sebagai berikut.

- Citra Biner

Citra Biner (*binary image*) adalah citra yang hanya mempunyai nilai derajat keabuan yaitu hitam dan putih. Piksel-piksel objek bernilai 1 dan piksel-piksel latar belakang bernilai 0. Piksel yang berwarna putih berisi nilai 0 dan piksel yang berwarna hitam berisi nilai 1 pada saat menampilkan citra.

- Citra Keabuan

Citra Keabuan (*greyscale*) adalah citra yang disetiap pikselnya mengandung satu *layer* dimana nilai intensitasnya berada pada nilai 0 (hitam) – 255 (putih).

- Citra Berwarna

Citra warna (RGB) adalah citra digital yang memiliki informasi warna pada setiap pikselnya. Sistem pewarnaan citra warna ada beberapa macam seperti RGB, HSV, dan CMYK. RGB adalah model warna yang terdiri dari merah, hijau, dan biru yang digabungkan dalam membentuk suatu susunan warna yang luas. Untuk monitor komputer, nilai rentangnya paling kecil = 0 dan paling besar = 255 (Sindy, 2019)

## **2.10 Operasi Morfologi Citra**

Morfologi adalah teknik pengolahan citra digital dengan menggunakan bentuk (*shape*) sebagai pedoman dalam pengolahan. Nilai dari setiap *pixel* dalam citra digital hasil diperoleh melalui proses perbandingan antara *pixel* yang bersesuaian pada citra digital masukan dengan *pixel* tetangganya. Operasi Morfologi bergantung pada urutan kemunculan dari *pixel*, tidak memperhatikan nilai *numeric* dari *pixel* sehingga teknik Morfologi sesuai apabila digunakan untuk melakukan pengolahan *binary image* dan *grayscale image* (Distira et al., 2012)

Dengan mengatur atau memilih ukuran dan bentuk dari matrik kernel (*structuring element*) yang digunakan maka dapat diatur sensitivitas operasi morfologi terhadap bentuk tertentu (*spesifik*) pada citra digital masukan. Secara matematis, citra merupakan fungsi kontinu dengan intensitas cahaya pada bidang 2 dimensi. Agar bisa diolah dengan kemampuan digital, maka suatu citra harus dipresentasikan secara numerik dengan nilai-nilai diskrit. Representasi dari fungsi kontinu menjadi nilai-nilai diskrit disebut digitalisasi citra. Tujuan dari pengolahan citra ini adalah memperbaiki kualitas suatu citra sehingga dapat diinterpretasi dengan mudah oleh manusia atau sebuah komputer.

Dalam konteks yang lebih luas, pengolahan citra digital lebih mengacu pada pemrosesan setiap 2 data dimensi. Pengolahan citra digital adalah sebuah disiplin ilmu yang mempelajari tentang bagaimana teknik pengolahan sebuah citra. Citra yang dimaksud di sini adalah sebuah gambar diam (foto) maupun gambar bergerak (video / video *streaming*) (Sindy, 2019). Berikut beberapa teknik pengolahan citra digital yang dapat digunakan dalam penelitian ini.

- *Resizing*

*Resizing* adalah Teknik proses pengolahan citra digital yang mengubah ukuran panjang dan lebar citra digital pada arah vertikal dan/atau horizontal.

- *Mean Substraction*

*Mean Substraction* adalah proses pengolahan citra digital untuk mengatasi permasalahan perubahan iluminasi pada gambar input yang ada pada *dataset*. Untuk menangani variasi intensitas dan normalisasi, dilakukan dengan cara menghitung nilai piksel rata-rata pada *dataset* pelatihan dan mengurangnya dari setiap gambar

selama pelatihan. Tahap ini dilakukan dengan inisialisasi 3 buah variable yaitu:  $\mu_R$ ,  $\mu_G$ , dan  $\mu_B$  (Sindy, 2019). Kemudian disaat gambar siap melalui jaringan yang ada, dilakukan proses mengurangi nilai rata-rata  $\mu$  untuk setiap *input channel* pada citra input:

$$\begin{aligned} R (Red) &= R - \mu_R \\ G (Green) &= G - \mu_G \\ B (Blue) &= B - \mu_B \end{aligned} \tag{2.5}$$

- *Scaling*

*Scaling* terhadap standar deviasi oleh *training set* dibutuhkan pada gambar agar dapat diproses oleh jaringan *Deep Learning* berdasarkan citra masukan serta dapat mengkoordinasikan skala *bounding box* yang diproses pada ukuran citra input, *display*, maupun citra yang akan disimpan (Sindy, 2019). *Scaling factor*,  $\sigma$ , yang ditambahkan dalam sebuah normalisasi:

$$\begin{aligned} R (Red) &= (R - \mu_R) / \sigma \\ G (Green) &= (G - \mu_G) / \sigma \\ B (Blue) &= (B - \mu_B) / \sigma \end{aligned} \tag{2.6}$$

Lebih lanjutnya pada penelitian ini, teknik Operasi Morphologi Citra Digital sangat membantu dalam proses pengolahan foto sebagai *dataset* dari program yang dibuat.

## 2.11 Citra *Gray-Scale* (Keabuan)

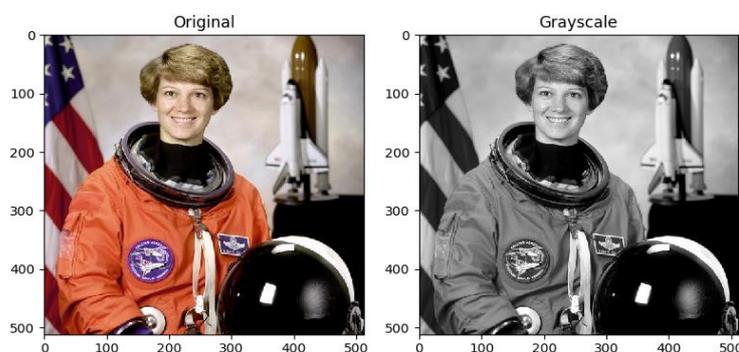
Proses awal yang banyak dilakukan dalam *image processing* adalah mengubah citra berwarna menjadi citra *gray-scale*. Proses mengubah menjadi

komposisi *gray-scale* digunakan untuk menyederhanakan model citra. Seperti telah dijelaskan sebelumnya, citra berwarna terdiri dari 3 *layer* matriks yaitu *R-layer* (*red*), *G-layer* (*green*), dan *B-layer* (*blue*). Sehingga untuk melakukan proses selanjutnya memerlukan 3 *layer* tersebut. Bila setiap proses perhitungan dilakukan menggunakan 3 *layer*, maka dilakukan 3 perhitungan yang sama. Sehingga konsep ini diubah dengan mengubah 3 *layer* RGB menjadi 1 *layer* matriks *gray-scale* dan hasilnya adalah citra *gray-scale*. Dalam citra ini tidak terdapat warna apapun selain warna hitam, putih, dan warna antara hitam dan putih (abu-abu) (Distira et al., 2012)

Untuk mengubah citra berwarna yang mempunyai nilai matriks, masing-masing *layer* R, G, B menjadi citra *gray-scale* dengan nilai *s*, maka konversi dapat dilakukan dengan mengambil rata-rata dari nilai R, G, dan B sehingga dapat dituliskan menjadi berikut.

$$s = \frac{r + g + b}{3} \quad 2.7$$

Berikut ilustrasi *image* yang telah dikonversi RGB ke *gray-scale* pada Gambar 2.12.



**Gambar 2. 11** Contoh konversi image RGB menjadi image gray-scale.

## 2.12 Analisis Blob

Analisis Blob adalah suatu proses menganalisa atau mengukur properti-properti geometri dari suatu *blob* (*Binary Large Objek*) yang terdapat pada citra biner. Pada pengaplikasiannya, Analisis Blob digunakan untuk mengeliminasi blob yang tidak memiliki kriteria yang sedang dicari dan mempertahankan hanya sejumlah blob yang relevan berdasarkan karakteristik geometri yang telah ditentukan. Selanjutnya, blob yang tidak dieliminasi tersebut digunakan untuk analisa selanjutnya. Hal ini digunakan untuk menghemat waktu komputasi pada pengolahan citra (Wahyuni, 2018)

Blob merupakan area pada citra digital yang memiliki sifat yang konstan atau bervariasi dalam kisaran tertentu. Untuk melakukan komputasi *blob detection*, digunakan determinan dari matriks Hessian (DoH) dari citra. Jika diberikan titik  $x=(x,y)$  pada citra I, matrik Hessian  $H(x,\sigma)$  pada  $x$  dengan skala  $\sigma$  didefinisikan sebagai berikut.

$$H(x, \sigma) = \{L_{xx}(x, \sigma) \ L_{xy}(x, \sigma) \ L_{xy}(x, \sigma) \ L_{yy}(x, \sigma)\} \quad 2.10$$

Dimana adalah  $L_{xx}$  solusi dari turunan kedua fungsi Gaussian  $\frac{\partial^2}{\partial x^2} g(\sigma)$  dengan citra I pada titik  $x$ . Definisi ini berlaku juga untuk  $L_{xy}(x, \sigma)$  dan  $L_{yy}(x, \sigma)$ . Fungsi Gaussian didefinisikan sebagai  $g(x)$  berikut.

$$g(\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad 2.11$$

Determinan dari matriks Hessian sering digunakan sebagai dasar algoritma karena sifat invariant terhadap skala, kestabilan, dan berulang dengan mudah (Nuryanto, 2017)

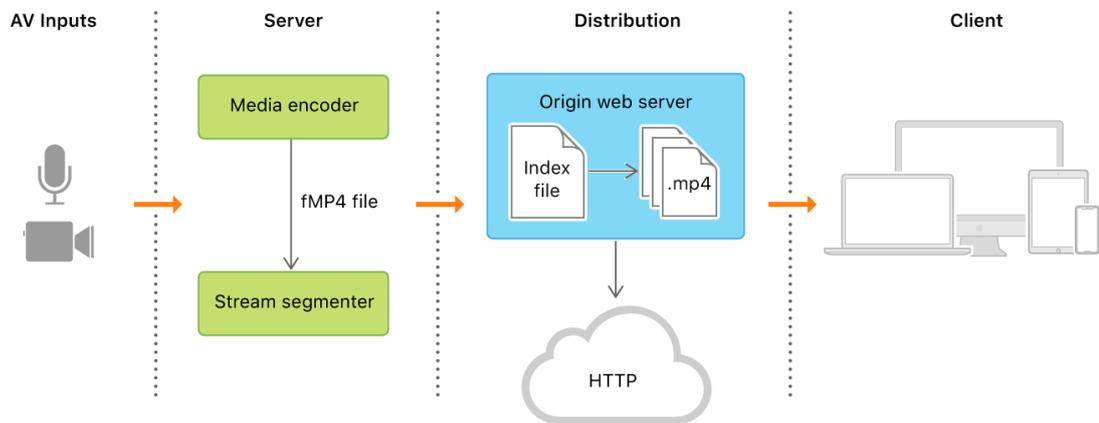
### **2.13 Video Streaming Digital**

Video merupakan suatu media yang sangat penting untuk komunikasi dan hiburan selama puluhan tahun ini. Pada pertama kalinya, video diolah dan ditransmisikan dalam bentuk *analog*. Perkembangan di bidang komputer telah membantu terbentuknya video dalam bentuk *digital*. Video digital merupakan representasi elektronik dari gambar visual bergerak dalam bentuk data digital yang dikodekan. Berbeda dengan video analog yang merepresentasikan gambar visual bergerak dengan sinyal analog. Video digital terdiri dari serangkaian gambar digital yang ditampilkan secara beruntun dan berurutan. Video digital juga dapat diartikan sebagai teknologi pengiriman sinyal elektronik dari suatu gambar yang bergerak. Aplikasi umum dari sinyal video digital yaitu seperti televisi, namun ia bisa juga digunakan dalam aplikasi lain di dalam bidang teknik, saintifik, produksi dan juga keamanan. Berdasarkan bahasa, kata video ini berasal dari kata latin. Salah satu penerapan video digital yang digunakan dalam transmisi data adalah video *streaming*.

Video *Streaming* adalah teknologi pengiriman data, video atau audio dalam bentuk yang telah dikompresi melalui jaringan internet yang ditampilkan oleh suatu *player* secara realtime. Pengguna memerlukan *player* yang merupakan aplikasi khusus untuk melakukan dekompresi dan mengirimkan data berupa video ke tampilan layar monitor dan data berupa suara ke *speaker*. Sebuah *player* dapat

berupa suatu bagian dari *browser* atau sebuah perangkat lunak. Inti dari *streaming* adalah membagi data dan *encoding*, kemudian mengirimkannya melalui jaringan dan pada saat data sampai pada pengguna maka akan dilakukan *decoding* serta pembacaan data. Ciri-ciri aplikasi *streaming* yaitu distribusi audio, video, dan multimedia pada jaringan secara *realtime* atau *on demand*. Setelah ter-*streaming*, *transfer* media data *digital* dari *server* dan diterima oleh pengguna sebagai *realtime stream* simultan sehingga pengguna tidak perlu menunggu keseluruhan data di-*download* karena *server* mengirimkan data yang diperlukan setiap selang waktu tertentu. Hal ini memungkinkan pengguna untuk menjalankan *file content* seketika dengan periode *buffer* pendek.

Ada beberapa video *streaming* antara lain *webcast*, dimana tayangan yang ditampilkan merupakan siaran langsung (*live*) dan *Video on Demang* (VoD), dimana tayangan yang ditampilkan sudah terlebih dahulu disimpan dalam *server*. Faktor-faktor yang mempengaruhi distribusi video *streaming* melalui jaringan antara lain: besarnya *bandwidth*, waktu tunda (*delay*), *lost packet*, dan juga teknik mendistribusikan video tersebut ke beberapa tujuan secara merata dan efisien (Apostolopoulos et al., 2002). Berikut ilustrasi alur diagram *video streaming* pada Gambar 2.13.



**Gambar 2. 12** Alur diagram Video Streaming.

Sumber: (Apple, n.d.)

Ada tiga cara umum yang biasa digunakan dalam menerima *stream* data sebagai berikut.

### 1. *Download*

Pada penerimaan *stream* data dengan cara *download*, akses video dilakukan dengan cara melakukan *download* terlebih dahulu suatu *file* multimedia dari *server*. Penggunaan cara ini mengharuskan keseluruhan suatu *file* multimedia harus diterima secara lengkap pada pengguna. *File* multimedia yang sudah diterima kemudian disimpan pada tempat penyimpanan yang ada di komputer. Pengguna baru dapat mengakses video tersebut setelah berhasil menerima *file* multimedia tersebut secara lengkap. Keuntungan dari penggunaan cara *download* ini adalah akses yang lebih cepat ke salah satu bagian dari *file* tersebut. Sedangkan kekurangannya adalah pengguna yang ingin mengakses video tersebut harus menunggu terlebih dahulu sampai keseluruhan *file* multimedia tersebut diterima secara lengkap.

### 2. *Streaming*

Pada penerimaan video secara *streaming*, pengguna dapat melihat suatu *file* multimedia hampir bersamaan ketika *file* tersebut mulai diterima. Penggunaan cara ini mengharuskan pengiriman suatu *file* multimedia ke pengguna secara konstan (*realtime*). Hal ini bertujuan agar pengguna dapat menyaksikan video yang diterima secara langsung tanpa ada bagian yang hilang. Keuntungan dari cara ini adalah pengguna tidak perlu menunggu hingga suatu *file* multimedia dikirimkan secara lengkap. Dengan demikian, penggunaan cara ini memungkinkan sebuah *server* untuk melakukan pengiriman siaran secara langsung kepada pengguna.

### 3. *Progressive Downloading*

*Progressive Downloading* adalah suatu metode *hybrid* yang merupakan hasil penggabungan antara metode *download* dan metode *streaming*, dimana video yang sedang diakses dapat diterima dengan cara *download* sehingga *player* yang ada pada pengguna sudah dapat mulai menampilkan video tersebut sejak sebagian dari *file* tersebut diterima walaupun *file* tersebut belum diterima secara lengkap.

Secara umum, terdapat 4 komponen dari *streaming* yaitu:

- *Input*

Sumber dari video yang akan di-*streaming*. Sumber tersebut dapat berupa *file* video, DVD, MPEG, dan seterusnya.

- *Encoder*

Bagian dari aplikasi *server* yang bertugas untuk mengubah video sumber menjadi sebuah format yang sesuai dengan transmisi *streaming*, dimana format ini umumnya memiliki tingkat kompresi tinggi sehingga dapat ditransmisikan dengan baik pada suatu media jaringan.

- *Server*

*File* hasil *encoding* kemudian didistribusikan oleh *server* kepada pengguna. Pada aplikasi yang digunakan, *encoder* dan *server* berada pada satu aplikasi yang sama yang terintegrasi satu sama lain.

- *Player / output*

*Player* berfungsi untuk melakukan *decoding* terhadap *file* hasil *streaming* dan menampilkannya pada pengguna.

Penerapan teknologi *video streaming* mengharuskan dilakukannya perancangan sistem dan jaringan secara matang untuk memungkinkan pengiriman *video streaming* yang berkualitas baik. Faktor yang memengaruhi proses *video streaming* pada jaringan adalah *Bandwidth*.

*Bandwidth* didefinisikan sebagai jumlah *bit* informasi yang melalui suatu jaringan dalam periode waktu tertentu. *Bandwidth* yang tersedia di internet pada umumnya tidak dapat diketahui secara pasti dan sangat bervariasi terhadap waktu. Besarnya *bandwidth* yang tersedia pada jaringan sangat memengaruhi proses kerja suatu *video streaming*. Jika *server* melakukan pengiriman sebuah video dengan *bit rate* tinggi yang melebihi kapasitas *bandwidth* yang tersedia, maka akan terjadi kemacetan sehingga paket-paket tersebut akan di-*drop*. Hal ini akan menyebabkan terjadinya penurunan kualitas video yang diterima. Hal ini juga berlaku apabila *server* melakukan pengiriman dengan *bit rate* yang lebih rendah. Oleh karena itu, seorang perancang jaringan harus mampu memperkirakan besar kapasitas *bandwidth* yang tersedia dan menyesuaikannya dengan *bit rate* video yang dikirimkan.

## 2.14 Tensorflow

Tensorflow adalah *open source software library* untuk komputasi berperforma tinggi. Tensorflow memiliki arsitektur yang sangat fleksibel sehingga dapat digunakan oleh banyak platform (GPU, CPU, TPU) dan dari *desktop* ke *server* hingga perangkat di ujung sistem. Pada awalnya dikembangkan oleh tim Google Brain dalam organisasi Google AI. Tensorflow berfokus pada *deep learning* dan *machine learning* dan juga inti komputasi numerik yang fleksible yang dapat digunakan pada domain saintifik lainnya. Tensorflow adalah *deep learning library* terpopuler saat ini. Perusahaan besar seperti Google, Nvidia, Kakao, Snapchat, dan Twitter menggunakan Tensorflow (Fahrudin, 2019).

## 2.15 Confusion Matrix

Matriks Konfusi merupakan tabel pencatat hasil kerja klasifikasi. Matriks Konfusi berisi informasi tentang klasifikasi yang dapat diprediksi dengan sistem klasifikasi. Kinerja sistem umumnya dievaluasi menggunakan data dalam bentuk matriks (Santra & Christy, 2012). Melalui *Confusion Matrix* keakuratan, tingkat kesalahan, ketepatan dan nilai penarikan dapat diketahui. Kuantitas Matriks Konfusi dapat diringkus menjadi 2 nilai, yaitu akurasi dan laju *error*. Dengan mengetahui jumlah data yang diklasifikasikan dengan benar, dapat diketahui akurasi hasil prediksi dan dengan mengetahui jumlah data yang diklasifikasikan secara salah, dapat diketahui laju *error* dari prediksi yang dilakukan.

Pada pengukuran kinerja menggunakan *confusion matrix*, terdapat 4 istilah sebagai representasi hasil proses klasifikasi. Keempat istilah tersebut adalah *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN).

Nilai *True Negative* (TN) merupakan jumlah data negatif yang terdeteksi dengan benar, sedangkan *False Positive* (FP) merupakan data negatif namun terdeteksi sebagai data positif. Sementara itu, *True Positive* (TP) merupakan data positif yang terdeteksi benar. *False Negative* (FN) merupakan kebalikan dari *True Positive*, sehingga data positif, namun terdeteksi sebagai data negatif (Saifullah, 2019).

Berdasarkan nilai *True Negative* (TN), *False Positive* (FP), *False Negative* (FN), dan *True Positive* (TP) dapat diperoleh nilai akurasi, presisi, *error* dan *recall*. Namun untuk penelitian ini hanya akan membahas tentang akurasi dan *error* pada sistem. Akurasi menggambarkan seberapa akurat sistem dapat mengklasifikasi data secara benar. Dengan kata lain, nilai akurasi merupakan perbandingan antara data yang terklasifikasi benar dengan keseluruhan data (Saifullah, 2019). Nilai akurasi dapat diperoleh dengan Persamaan 2.5.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} * 100\% \quad 2.5$$

Dimana:

- *TP* adalah *True Positive*, yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem untuk kelas ke-i.
- *TN* adalah *True Negative*, yaitu jumlah data negatif yang terklasifikasi dengan benar oleh sistem untuk kelas ke-i.
- *FN* adalah *False Negative*, yaitu jumlah data negatif namun terklasifikasi salah oleh sistem untuk kelas ke-i.
- *FP* adalah *False Positive*, yaitu jumlah data positif namun terklasifikasi salah oleh sistem (Saifullah, 2019)

Selanjutnya, jika ingin mengetahui perbandingan nilai antara *True Positive* (TP) dengan banyaknya data yang **diprediksi positif**, dituliskan dengan persamaan 2.7.

$$Precision = \frac{TP}{TP + FP} * 100\% \quad 2.7$$

Jika ingin mengetahui perbandingan nilai antara *True Positive* (TP) dengan banyaknya data yang **sebenarnya positif**, dituliskan dengan persamaan 2.8

$$Recall = \frac{TP}{TP + FN} * 100\% \quad 2.8$$

Diketahui, jika nilai salah satu dari *Precision* atau *Recall* tinggi, maka nilai salah satu yang lain cenderung lebih rendah. Sebagai contoh jika nilai *Precision* sangat tinggi atau mendekati skor 1.0, maka nilai dari persamaan *Recall* akan cenderung lebih rendah. Jika hanya menggunakan 2 persamaan ini, dapat terjadi bias pada kondisi dunia nyata. Oleh sebab itu tidak boleh hanya menggunakan skor ini untuk melakukan pemilihan model. Untuk mengatasi hal ini, diperlukan persamaan *F1-Score* yang dituliskan sebagai persamaan 2.9.

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} * 100\% \quad 2.9$$

Nilai yang didapatkan kemudian menggunakan rumus *F1-Score* berfungsi sebagai *harmonic mean* dari *precision* dan *recall*. Nilai terbaik dari *F1-Score* adalah 1.0 dan nilai terburuk adalah 0. Secara representasi, jika *F1-Score* punya skor yang baik mengindikasikan bahwa model klasifikasi memiliki *precision* dan *recall* yang baik pula (Saifullah, 2019).

## 2.16 *Unmanned Aerial Vehicle (UAV)*

*Unmanned Aerial Vehicle* atau yang biasa disebut sebagai *Drone* adalah pesawat tanpa awak yang dapat berfungsi dengan pengendali jarak jauh oleh pilot. *Drone* dapat digunakan untuk membawa muatan baik berupa senjata atau muatan yang lainnya. Saat ini *Drone* selain banyak digunakan di bidang militer juga banyak digunakan untuk misi pencarian dan penyelamatan. Sebelumnya *drone* masih dikendalikan dengan secara manual atau menggunakan remote kontrol. Namun sekarang *drone* bisa dikendalikan secara semi otomatis dan juga *drone* dapat diprogram pada komputer yang terpasang pada *drone* tersebut. Berikut salah satu foto bentuk *drone* yang banyak digunakan oleh banyak orang pada Gambar 2.14.



**Gambar 2. 13** Contoh Drone

Sumber: (EASA, 2018)

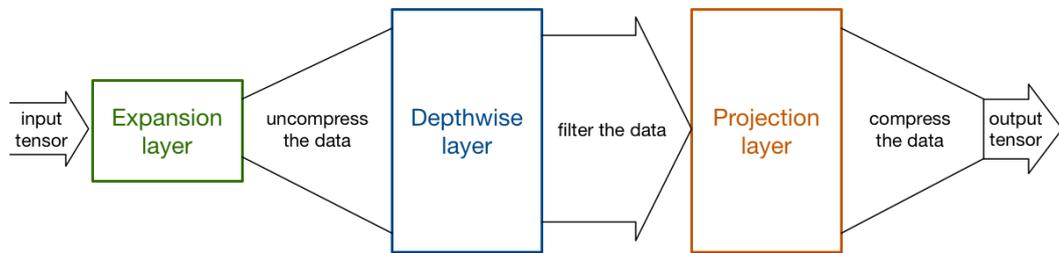
Ada banyak kegunaan *drone*, contohnya untuk mengambil gambar kondisi bencana alam, melacak korban, mengumpulkan dan mengirimkan sampel medis ke daerah terpencil yang sulit dijangkau oleh manusia, *drone* juga bisa menggunakan sensor inframerah untuk mendeteksi manusia dan membantu dalam misi pencarian dan penyelamatan (Radiansyah, 2017)

Akhir-akhir ini ada banyak *drone* model baru dengan bahan yang murah dan dapat diperintah dengan mudah seperti *drone DJI Tello*. *Drone* ini mengandalkan

konfigurasi quadcopter yang biasa digunakan oleh banyak profesional atau industri pesawat tak berawak. *Drone DJI Tello* dilengkapi dengan perangkat keras, yang dapat digunakan untuk computer vision, AI, dan aplikasi terkait robotika kooperatif (Kinaneva et al., 2019)

## **2.17 Arsitektur MobileNet V2**

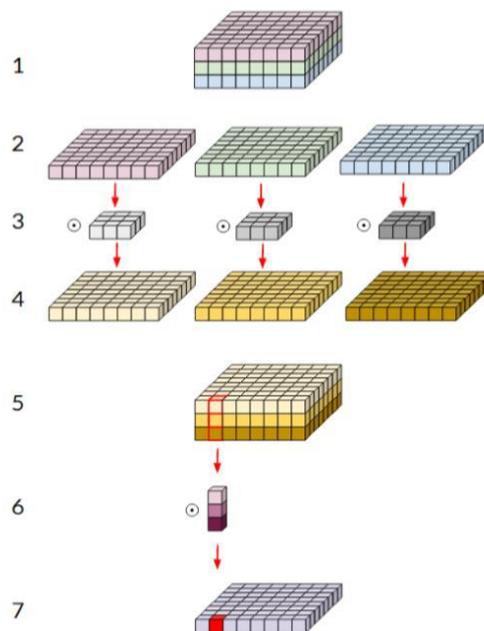
*MobileNet V2* adalah salah satu metode yang digunakan dalam melakukan pendeteksian objek. *MobileNet V2* merupakan salah satu arsitektur *Convolutional Neural Network* (CNN) yang dapat digunakan untuk mengatasi kebutuhan akan sumber komputasi berlebih. *MobileNet* dibangun dan dikembangkan oleh peneliti dari Google agar system CNN dapat digunakan pada ponsel maupun sistem tertanam, kemudian selanjutnya dikembangkan lebih lanjut hingga *MobileNet* telah mencapai versi ke-2 (V2). Perbedaan yang mendasar dari arsitektur *MobileNet V2* dan arsitektur CNN secara general terletak pada ketebalan filter yang digunakan untuk membaca input gambar. *MobileNet* membagi konvolusi menjadi *depthwise convolution* dan *pointwise convolution*. Arsitektur *MobileNet* memanfaatkan *Batch Normalization* (BN) dan *Rectified-Linear Unit* (ReLU) untuk *depthwise convolution* dan *pointwise convolution*. Filter gambar pada *MobileNet V2* dibangun di atas arsitektur jaringan yang efisien dengan menggunakan konvolusi dapat dipisahkan secara mendalam, sehingga metode ini menjadi lebih ringan dan dapat berjalan pada perangkat keras yang telah terintegrasi system di dalamnya (Sindy, 2019) Berikut ilustrasi cara kerja arsitektur *MobileNet V2* pada Gambar 2.15.



**Gambar 2. 14** Cara kerja deteksi MobileNet

Sumber: (Hollemaans, 2018)

*Depthwise Separable Convolution* (DSC) menggantikan konvolusi standard dengan 2 tahap operasi: 1. *Depthwise Convolution* dimana setiap filter  $D_F \times D_F$  hanya melakukan proses filter terhadap sebuah fitur map input secara mendalam; 2. *Pointwise Convolution* yang merupakan  $1 \times 1$  convolution layer yang digunakan untuk menggabungkan jalur informasi dari *depthwise layer* (Sindy, 2019) Ilustrasi DSC dapat dilihat pada Gambar 2.16.



**Gambar 2. 15** Konvolusi *Depthwise* (2-5) dan *Pointwise* (6-7)

Sumber: (Sindy, 2019)

DSC menyebabkan jalur konvolusi menjadi jauh lebih efisien dengan menggunakan parameter yang jauh lebih sedikit. Penggunaan parameter dan daya komputasi dapat dilihat pada Tabel 2.2 berikut.

**Tabel 2. 2** Ukuran Parameter dan Komputasi pada Depthwise Separable

Convolution

<i>Layer</i>	Ukuran Parameter	Ukuran Komputasi
Standard Conv	$F \times F \times C_1 \times C_2$	$F \times F \times D_M \times D_M \times C_1 \times C_2$
Depthwise Separable	$F \times F \times C_1 + 1 \times 1 \times C_1 \times C_2$	$F \times F \times D_M \times D_M \times C_1 + 1 \times 1 \times C_1 \times C_2$

### 2.18 Real Time Messaging Protocol (RTMP)

*Real Time Messaging Protocol* (RTMP) adalah aplikasi lapisan *proprietary* protokol yang dikembangkan oleh *Adobe System*. *Real Time Messaging Protocol* (RTMP) juga menyediakan dua arah yang pesan layanannya multiplex seperti *TCP [RFC0793]* yang dimaksudkan untuk membawa aliran parallel video, audio, dan pesan data, dengan informasi waktu yang terkait. Protokol *RTMP* memiliki beberapa variasi seperti protokol “*plain*” yang bekerja di atas dan menggunakan *TCP* nomor *port* 1935 secara *default*. *RTMPS* yang *RTMP* melalui koneksi *TLS / SSL*, *RTMPE* yang *RTMP* dienkripsi menggunakan mekanisme keamanan *Adobe* sendiri (Reressy et al., 2016). Pada penelitian ini, RTMP digunakan sebagai jalur komunikasi laptop dan *smartphone*.

## **2.19 *Transfer Learning via Fine Tuning***

*Transfer Learning* adalah teknik mengambil parameter model latih dari model yang telah teroptimasi dengan baik ke model yang ingin dibuat. *Transfer learning* terbagi menjadi 2, yaitu *Transfer learning via Feature Extraction* dan *Transfer Learning via Fine Tuning* (Rosebrock, 2019)

*Fine Tuning* adalah *training* model dengan *task* yang spesifik menggunakan model yang telah dilatih dengan *dataset* yang besar. *Fine tuning* dibutuhkan sebagai efisiensi latih data dengan secara lebih cepat dan efisien dibandingkan jika harus *training data* dari awal karena teknik *fine tuning* menggunakan *pre-trained model* (Fatyanosa, 2020)