

***HYBRID MODEL TRANSFER LEARNING RESNET50
DENGAN SUPPORT VEKTOR MACHINE UNTUK
DETEKSI MASKER PADA IMAGE WAJAH***

THESIS

**Oleh:
EKO AGUS MOH. IQBAL
NIM. 210605220004**



**PROGRAM STUDI MAGISTER INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2023**

***HYBRID MODEL TRANSFER LEARNING RESNET50
DENGAN SUPPORT VEKTOR MACHINE UNTUK
DETEKSI MASKER PADA IMAGE WAJAH***

THESIS

**Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan dalam
Meperoleh Gelar Magister Komputer (M.Kom)**

**Oleh:
EKO AGUS MOH. IQBAL
NIM. 210605220004**

**PROGRAM STUDI MAGISTER INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2023**

***HYBRID MODEL TRANSFER LEARNING RESNET50
DENGAN SUPPORT VEKTOR MACHINE UNTUK
DETEKSI MASKER PADA IMAGE WAJAH***

THESIS

**Diajukan kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan dalam
Meperoleh Gelar Magister Komputer (M.Kom)**

**Oleh:
EKO AGUS MOH. IQBAL
NIM. 210605220004**

**PROGRAM STUDI MAGISTER INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2023**

**HYBRID MODEL TRANSFER LEARNING RESNET50
DENGAN SUPPORT VEKTOR MACHINE UNTUK
DETEKSI MASKER PADA IMAGE WAJAH**

THESIS

**Oleh:
EKO AGUS MOH. IQBAL
NIM. 210605220004**

Telah diperiksa dan disetujui untuk diuji:
Tanggal 13 Juni 2023

Pembimbing I,



Dr. Ririen Kusumawati, S.Si., M.Kom
NIP. 19720309 200501 2 002

Pembimbing II,



Dr. Irwan Budi Santoso, M.Kom
NIP. 19770103 201101 1 004

Mengetahui,

Kepala Program Studi Magister Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Sahyo Crysdian
NIP. 19740424 200901 1 008

**HYBRID MODEL TRANSFER LEARNING RESNET50
DENGAN SUPPORT VEKTOR MACHINE UNTUK
DETEKSI MASKER PADA IMAGE WAJAH**

THESIS

**Oleh:
EKO AGUS MOH. IQBAL
NIM. 210605220004**

**Telah Dipertahankan di Depan Dewan Penguji Thesis
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Magister Komputer (M.Kom)
Tanggal 13 Juni 2023**

Susunan Dewan Penguji

Penguji Utama : Dr. M. Ainul Yaqin, M.Kom
NIP. 19761013 200604 1 004

Ketua Penguji : Dr. Yunifa Miftachul Arif, M.T
NIP. 19830616 201101 1 004

Sekretaris Penguji : Dr. Ririen Kusumawati, S.Si., M.Kom
NIP. 19720309 200501 2 002

Anggota Penguji : Dr. Irwan Budi Santoso, M.Kom
NIP. 19770103 201101 1 004

Tanda Tangan

()
()
()
()

Mengetahui dan Mengesahkan,
Kepala Program Studi Magister Informatika
Fakultas Sains dan Teknologi
Universitas Islam Maulana Malik Ibrahim Malang



Wahyo Crysdiyan
NIP. 19740424 200901 1 008

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Eko Agus Moh. Iqbal

NIM : 210605220004

Program Studi : Magister Informatika

Fakultas : Sains dan Teknologi

Menyatakan dengan sebenarnya bahwa Thesis yang saya tulis in benar-benar merupakan hasil karya saya sendiri, buka merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai tulisan dan pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Thesis ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 13 Juni 2023

Saya membuat pernyataan,



Eko Agus Moh. Iqbal
NIM. 210605220004

MOTTO

*"Berikan dan Lakukan yang terbaik,
sebelum menjadi yang terbaik"*

PERSEMBAHAN

Dengan mengucapkan syukur Alhamdulillah robbil alamin, Thesis ini saya persembahkan untuk:

1. Kedua orang tua tercinta yang kasih dan sayangnya selalu tulus
2. Istri tercinta yang hanya satu-satunya yang selalu memberikan doa dan dukungan
3. Seluruh keluarga yang selalu memberikan doa dan semangat
4. Seluruh Civitas Akademika Universitas Negeri Maulana Malik Ibrahim Malang yang telah memberikan kesempatan untuk menambah ilmu teknologi dan agama.
5. Seluruh rekan-rakan PT Datamax Teknologi Indonesia yang memberikan dukungan dan kesempatan belajar.
6. Seluruh rekan-rekan mahasiswa Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Maulana Malik Ibrahim Malang semua angkatan atas kerjasamanya selama ini.
7. Bapak, Ibu, saudara dan rekan-rekan sekalian yang tidak bisa saya sebutkan satu persatu dalam mendukung Thesis ini hingga bisa diselesaikan

KATA PENGANTAR

Assalamu'alaikum Wr. Wb

Syukur *Alhamdulillah* penulis haturkan kehadiran Allah SWT yang telah melimpahkan Rahmat dan Hidayah-Nya, sehingga penulis dapat menyelesaikan studi di Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang sekaligus menyelesaikan Thesis ini dengan baik.

Selanjutnya penulis haturkan ucapan terima kasih seiring do'a dan harapan *jazakumullah ahsanal jaza'* kepada semua pihak yang telah membantu terselesaikannya Thesis ini. Ucapan terima kasih ini penulis sampaikan kepada:

1. Ibu Dr. Ririen Kusumawati, M.Kom dan Bapak Dr. Irwan Budi Santoso, M.Kom, selaku dosen pembimbing Thesis, yang telah banyak memberikan pengarahan dan pengalaman yang berharga.
2. Segenap civitas akademika Program Studi Magister Informatika, terutama seluruh Bapak / Ibu dosen, terima kasih atas segenap ilmu dan bimbingannya.
3. Keluarga tercinta yang senantiasa memberikan do'a dan semangat
4. Semua rekan-rekan seperjuangan yang ikut mendukung dan membantu.

Penulis menyadari bahwa dalam penyusunan Thesis ini masih terdapat kekurangan dan penulis berharap semoga Thesis ini bisa memberikan manfaat kepada para pembaca khususnya bagi penulis secara pribadi. *Amiinn Yaa Rabbal Alamin.*

Wasalamu'alaikum Wr. Wb

Malang, 13 Juni 2023

Penulis

DAFTAR ISI

HALAMAN JUDL	i
HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN.....	iii
HALAMAN PENGESAHAN.....	iv
HALAMAN PERNYATAAN.....	v
MOTTO	vi
HALAMAN PERSEMBAHAN.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL	xii
ABSTRAK.....	xiii
ABSTRACT	xiv
مستخلص البحث	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Pernyataan Masalah.....	5
1.3 Tujuan Penelitian.....	5
1.4 Manfaat Penelitian.....	5
1.5 Ruang Lingkup Masalah.....	6
BAB II STUDI PUSTAKA	7
1.1 Deteksi Masker Pada Image Wajah	7
1.2 Kerangka Teori.....	10
BAB III DESAIN PENELITIAN.....	15
3.1 Prosedur Penelitian	15
3.1.1 Pengumpulan Data	15
3.1.2 Desain Sistem.....	16
3.1.3 Eksperimen	26
3.1.4 Metode Evaluasi.....	27
3.2 Instrument Penelitian	28
BAB IV MODEL TRANSFER LEARNING RESNET50.....	29
4.1 Desain	29
4.2 Uji Coba.....	31
4.2.1 Uji Coba RGB – ResNet50 (Model <i>Freeze</i>).....	32

4.2.2 Uji Coba RGB – ResNet50 (Model <i>Unfreeze</i>)	35
4.2.3 Uji Coba HSV – ResNet50 (Model <i>Freeze</i>)	36
4.2.4 Uji Coba HSV – ResNet50 (Model <i>Unfreeze</i>)	38
4.3 Komparasi Performa pada Hasil Uji Coba	39
BAB V HYBRID MODEL RESNET50 DAN SUPPORT VECTOR MACHINE	41
5.1 Desain	41
5.2 Uji Coba.....	42
5.2.1 Uji Coba RGB – ResNet50 (Model <i>Freeze</i>) & SVM	42
5.2.2 Uji Coba RGB – ResNet50 (Model <i>Unfreeze</i>) & SVM.....	46
5.2.3 Uji Coba HSV – ResNet50 (Model <i>Freeze</i>) & SVM.....	49
5.2.4 Uji Coba HSV – ResNet50 (Model <i>Unfreeze</i>) & SVM	53
5.2 Komparasi Performa pada Hasil Uji Coba	56
BAB VI PEMBAHASAN.....	58
6.1 Komparasi dan Performa Metode.....	58
6.1.1 Performa ResNet50	59
6.1.2 Performa Hybrid ResNet50 dan SVM.....	60
6.1.3 Komparasi Kedua Model	62
6.1.4 Komparasi Dengan Penelitian Sebelumnya	63
6.2 Implementasi Hasil	64
BAB VII KESIMPULAN	69
8.1 Kesimpulan	69
8.2 Saran	69
Daftar Pustaka	70

DAFTAR GAMBAR

Gambar 3.1	Prosedur penelitian.....	15
Gambar 3.2	Desain Sistem.....	17
Gambar 3.3	<i>Colorspace</i> HSV	18
Gambar 3.4	Arsitektur CNN Secara Umum	19
Gambar 3.5	Residual Network (ResNet)	22
Gambar 3.6	Arsitektur ResNet50.....	23
Gambar 3.7	Perhitungan ResNet50.....	24
Gambar 3.8	Support Vector Machine	26
Gambar 4.1	Desain Proses Klasifikasi ResNet50	29
Gambar 3.2	Konversi RGB ke HSV	30
Gambar 4.3	Akurasi dan <i>Validation</i> Akurasi RGB - ResNet50 (<i>Freeze</i>).....	34
Gambar 4.4	<i>Loss</i> dan <i>Validation Loss</i> RGB - ResNet50 (<i>Freeze</i>)	34
Gambar 4.5	Akurasi dan <i>Validation</i> Akurasi RGB - ResNet50 (<i>Unfreeze</i>)	35
Gambar 4.6	<i>Loss</i> dan <i>Validation Loss</i> RGB - ResNet50 (<i>Unfreeze</i>).....	36
Gambar 4.7	Akurasi dan <i>Validation</i> Akurasi HSV – ResNet50 (<i>Freeze</i>)	37
Gambar 4.8	<i>Loss</i> dan <i>Validation Loss</i> HSV - ResNet50 (<i>Freeze</i>).....	37
Gambar 4.9	Akurasi dan <i>Validation</i> Akurasi HSV - ResNet50 (<i>Unfreeze</i>).....	38
Gambar 4.10	<i>Loss</i> dan <i>Validation Loss</i> HSV - ResNet50 (<i>Unfreeze</i>)	37
Gambar 5.1	Model <i>Hybrid</i> ResNet50 & SVM	41
Gambar 5.2	Uji Coba <i>Split Validation</i> RGB – ResNet50 SVM (<i>Freeze</i>).....	44
Gambar 5.3	<i>Confusion Matrix</i> RGB – ResNet50 (Model <i>Freeze</i>) & SVM.....	45
Gambar 5.4	Uji Coba <i>Split Validation</i> RGB – ResNet50 SVM (<i>Unfreeze</i>)	47
Gambar 5.5	<i>Confusion Matrix</i> RGB – ResNet50 (Model <i>Unfreeze</i>) & SVM.....	48
Gambar 5.6	Uji Coba <i>Split Validation</i> HSV – ResNet50 SVM (<i>Freeze</i>).....	51
Gambar 5.7	<i>Confusion Matrix</i> HSV – ResNet50 (Model <i>Freeze</i>) & SVM.....	52
Gambar 5.8	Uji Coba <i>Split Validation</i> HSV – ResNet50 SVM (<i>Unfreeze</i>)	54
Gambar 5.9	<i>Confusion Matrix</i> HSV – ResNet50 (Model <i>Unfreeze</i>) & SVM.....	55
Gambar 6.1	Contoh <i>image</i> wajah tidak menggunakan masker.....	59
Gambar 6.2	Contoh <i>image</i> wajah menggunakan masker.....	59
Gambar 6.3	Video frame terdeteksi tidak memakai masker.....	65
Gambar 6.4	Video <i>frame</i> terdeteksi memakai masker.....	65
Gambar 6.5	Video <i>frame</i> terdeteksi tidak memakai masker.....	66

DAFTAR TABEL

Tabel 2.1 Kerangka Teori dengan Data Sama	11
Tabel 2.2 Kerangka Teori dengan Data Berbeda	11
Tabel 2.3 Ranking akurasi dari yang terbaik	12
Tabel 3.1 Kasus Uji.....	27
Tabel 4.1 <i>Split Validation</i> ResNet50 (RGB).....	27
Tabel 4.2 <i>Split Validation</i> ResNet50 (HSV).....	33
Tabel 4.3 Komparasi Performa Akurasi Uji Coba ResNet50	40
Tabel 5.1 <i>Split Validation</i> ResNet50 (<i>freeze</i>) - SVM (RGB) 0.001	43
Tabel 5.2 <i>Split Validation</i> ResNet50 (<i>freeze</i>) - SVM (RGB) 0.01	43
Tabel 5.3 <i>Split Validation</i> ResNet50 (<i>freeze</i>) - SVM (RGB) 0.0001	44
Tabel 5.4 <i>K-Fold Croos Validation</i> Model ResNet50 (<i>Freeze</i>) – SVM (RGB)....	44
Tabel 5.5 <i>Split Validation</i> ResNet50 (<i>Unfreeze</i>) - SVM (RGB) 0.001	46
Tabel 5.6 <i>Split Validation</i> ResNet50 (<i>Unfreeze</i>) - SVM (RGB) 0.01	47
Tabel 5.7 <i>Split Validation</i> ResNet50 (<i>Unfreeze</i>) - SVM (RGB) 0.0001	47
Tabel 5.8 <i>K-Fold Croos Validation</i> ResNet50 (<i>Unfreeze</i>) – SVM (RGB).....	48
Tabel 5.9 <i>Split Validation</i> ResNet50 (<i>freeze</i>) - SVM (HSV) 0.001.....	50
Tabel 5.10 <i>Split Validation</i> ResNet50 (<i>freeze</i>) - SVM (HSV) 0.01.....	50
Tabel 5.11 <i>Split Validation</i> ResNet50 (<i>freeze</i>) - SVM (HSV) 0.0001.....	50
Tabel 5.12 <i>K-Fold Croos Validation</i> ResNet50 (<i>freeze</i>) – SVM (HSV).....	51
Tabel 5.13 <i>Split Validation</i> ResNet50 (<i>Unfreeze</i>) - SVM (HSV).....	53
Tabel 5.14 <i>Split Validation</i> ResNet50 (<i>Unfreeze</i>) - SVM (HSV).....	54
Tabel 5.15 <i>Split Validation</i> ResNet50 (<i>Unfreeze</i>) - SVM (HSV).....	54
Tabel 5.16 <i>K-Fold Croos Validation</i> ResNet50 (<i>Unfreeze</i>) – SVM (HSV)	55
Tabel 5.17 Komparasi Performa Akurasi Uji Coba ResNet50 & SVM	56
Tabel 6.1 Hasil <i>experiment</i> kedua model.....	62
Tabel 6.2 Hasil Komparasi dengan penelitian sebelumnya	63

ABSTRAK

Iqbal, Eko Agus Moh, 2023, Hybrid Model Transfer Learning ResNet50 dengan Support Vector Machine untuk Deteksi Masker Pada Image Wajah, Program Magister Informatika Universitas Islam Negeri Maulana Malik Ibrahim, Pembimbing: (1) Dr. Ririen Kusumawati, M.Kom (2) Dr. Irwan Budi Santoso, M.Kom

Kata Kunci: Deteksi Masker, Transfer Learning, HSV, ResNet50, SVM

Virus Covid-19 menyebabkan krisis kesehatan di Indonesia. Virus ini sangat mematikan sehingga banyak menimbulkan korban jiwa yang menyebabkan seluruh dunia termasuk pemerintah setempat menaruh fokus utama terhadap pandemi Covid-19. Pemerintah Indonesia telah mengeluarkan beberapa kebijakan untuk mencegah penyebaran wabah ini, salah satunya memakai masker ditempat umum. Salah satu pendekatan yang banyak digunakan dalam bidang *computer vision* yaitu *Convolutional Neural Network* (CNN) transfer learning. Dalam penelitian ini model hybrid ResNet50 dengan SVM dengan *preprocessing* RGB ke HSV disajikan untuk mendeteksi masker pada image wajah. Model ini terdiri dari tiga komponen proses. Pertama *preprocessing* image RGB ke HSV, Komponen kedua untuk Ekstrasi Fitur dengan ResNet50 dan ketiga klasifikasi masker pada image wajah dengan Support Vector Machine (SVM). Dari dataset sebanyak 7328 data image dilakukan pelatihan dan pengujian. Model pertama, tidak dilakukan *preprocessing* terhadap data image dengan ResNet50 menghasilkan akurasi 86.52%. Model kedua, model dengan *preprocessing* mengubah data image dari RGB ke HSV dengan ResNet50 menghasilkan akurasi 99.18%. Model ketiga, tidak dilakukan *preprocessing* pada image data dengan ResNet50 dan SVM memiliki akurasi 90.55%. Model keempat, model dengan *preprocessing* mengubah data image dari RGB ke HSV dengan ResNet50 dan SVM memiliki akurasi 98.36%.

ABSTRACT

Iqbal, Eko Agus Moh, 2023, Hybrid Model Transfer Learning ResNet50 with Support Vector Machine for Face Mask Detection, Master Study in Computer Science, State Islamic University Maulana Malik Ibrahim, Advisors: (1) Dr. Ririen Kusumawati, M.Kom (2) Dr. Irwan Budi Santoso, M.Kom

Keywords: Mask Detection, Transfer Learning, HSV, ResNet50, SVM

The Covid-19 virus caused a health crisis in Indonesia. This virus is so deadly that it has caused many fatalities which have caused the whole world including the government to pay major attention to the Covid-19 pandemic. The Indonesian government has issued several policies to prevent the spread of this epidemic, one of which is wearing a mask in public places. One approach that is widely used in the field of computer vision is the Convolutional Neural Network (CNN) transfer learning. In this study, Hybrid Model Transfer Learning ResNet50 and SVM with RGB to HSV preprocessing is presented to detect masks in facial images. This model consists of three process components. The first is preprocessing RGB images to HSV, the second component is for Feature Extraction with ResNet50 and the third is mask classification on face images with Support Vector Machine (SVM). From dataset of 7328 training and testing data were carried out. The first model, without preprocessing the image data with ResNet50, produces an accuracy of 86.52%. The second model, the model with preprocessing converts image data from RGB to HSV with ResNet50 resulting in an accuracy of 99.18%. in the third model, without preprocessing with ResNet50 and SVM which has an accuracy of 90.55%. The fourth model, the model with preprocessing converts image data from RGB to HSV with ResNet50 and SVM resulting in an accuracy of 98.36%.

مستخلص البحث

إقبال, أيكو أغوس محمد. 2023. نموذج نقل التعلم الهجين ريسنيت50 آلة المتجهات الداعمة لاكتشاف الكمامة على صورة الوجه. برنامج دراسة ماجستير المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف: (1) الدكتور ريرين كوسوماواتي الماجستير. (2) الدكتور اروان بودي سانتوسو الماجستير. **الكلمات المفتاحية:** اكتشاف الكمامة, نقل التعلم, *HSV, Resnet50, SVM*.

تسبب فيروس كوفيد-19 في أزمة صحية في إندونيسيا. هذا الفيروس قاتل لدرجة أنه تسبب من الوفيات حول العالم، لذلك ركزت الحكومة على جائحة كوفيد-19. أصدرت الحكومة الإندونيسية عدة سياسات لمنع انتشار هذا الوباء، من بينها اشتراط ارتداء الكمامة في الأماكن العامة. نهج واحد يستخدم على نطاق واسع في مجال رؤية الكمبيوتر وهو الشبكة العصبية مع SVM مع ResNet50) نقل التعلم. في هذا البحث، إن نموذج هجين CNN التلافيفية (قدّم إلى اكتشاف الكمامة على صورة الوجه. يتكون هذا HSV إلى RGB معالجة مسبقة إلى RGB النموذج من ثلاثة مكونات عملية. المكون الأول هو المعالجة المسبقة لصورة والثالث هو ResNet50. المكون الثاني مخصص لاستخراج الميزات باستخدام HSV (من مجموعة SVM تصنيف الكمامة على صور الوجه باستخدام آلة المتجهات الداعمة) بيانات من 7328 بيانات صورة، تم إجراء التدريب والتجريب. النموذج الأول، بدون المعالجة، ينتج دقة تبلغ 86.52%. النموذج الثاني، ResNet50 المسبقة لبيانات الصورة باستخدام ResNet50 مع HSV إلى RGB النموذج ذو المعالجة المسبقة يحول بيانات الصورة من مما ينتج عنه دقة تصل إلى 99.18%. النموذج الثالث، بلا معالجة مسبقة لبيانات الصورة بدقة 90.55%. النموذج الرابع، النموذج ذو المعالجة المسبقة SVM و ResNet50 باستخدام ResNet50 مع HSV إلى RGB يحول بيانات الصورة من

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam bidang kesehatan transformasi digital telah banyak diterapkan baik secara klinis ataupun non-klinis. Penyebaran virus secara kontak langsung dapat menyebar dengan cepat, sehingga teknologi merupakan bagian dari solusi untuk menghindari penyebaran virus. Wabah yang menginfeksi saluran pernapasan ini disebabkan virus yang disebut *Severe Acute Respiratory Syndrome Corona Virus 2* (SARS-CoV-2) (Boldog *et al.*, 2020). Virus ini sangat mematikan hingga menimbulkan korban jiwa yang membuat seluruh dunia termasuk pemerintah setempat menaruh fokus pada pandemi Covid-19. Segala sesuatu yang terjadi di dunia, termasuk pandemi Covid-19 adalah atas kehendak Allah, dalam Al-Qur'an surah At-Taghabun: 11

مَا أَصَابَ مِنْ مُصِيبَةٍ إِلَّا بِإِذْنِ اللَّهِ وَمَنْ يُؤْمِنُ بِاللَّهِ يَهْدِ اللَّهُ قَلْبَهُ وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ

“Tidak ada sesuatu musibah pun yang menimpa seseorang kecuali dengan izin Allah; Dan barang siapa yang beriman kepada Allah, niscaya Dia akan memberi petunjuk kepada hatinya. Dan Allah Maha Mengetahui segala sesuatu.”

Sebagai umat islam yang beriman kepada *Qadha* dan *Kadar* Allah, harus selalu sabar dan tawakal kepada Allah sehingga kita dapat menerima dan meyakini bahwa adanya Covid-19 saat ini adalah bentuk musibah dan ujian dari Allah. Meski sudah takdir yang diberikan Allah, manusia tetap berikhtiar untuk mengupayakan tidak tertular virus Covid19, sebagaimana yang dijelaskan pada HR Bukhari dan Muslim dari Usamah bin Zaid.

قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ الطَّاعُونَ آيَةُ الرَّجْزِ ابْتَلَى اللَّهُ عَزَّ وَجَلَّ بِهِ نَاسًا مِنْ عِبَادِهِ فَإِذَا سَمِعْتُمْ بِهِ فَلَا تَدْخُلُوا عَلَيْهِ وَإِذَا وَقَعَ بِأَرْضِ وَأَنْتُمْ بِهَا فَلَا تَفِرُّوا مِنْهُ

“Rasulullah shallallahu ‘alaihi wasallam bersabda: “*Tha’un* (wabah penyakit menular) adalah suatu peringatan dari Allah Subhanahu WaTa’ala untuk menguji hamba-hambanya dari kalangan manusia. Maka apabila kamu mendengar penyakit itu berjangkit di suatu negeri, janganlah kamu masuk ke negeri itu. Dan apabila wabah itu berjangkit di negeri tempat kamu berada, jangan pula kamu lari daripadanya.” (HR Bukhari dan Muslim dari Usamah bin Zaid).

Hadits di atas menjelaskan bagaimana menghindari daerah-daerah yang terjangkit wabah penyakit. Selain itu, penggunaan masker dapat digunakan untuk melindungi diri saat berkontak dengan orang yang terinfeksi atau untuk pengendalian terhadap orang yang telah terinfeksi. Dalam hal ini orang yang terinfeksi menggunakan masker untuk mencegah penularan lebih lanjut (*World Health Organization*, 2020).

Indonesia, merupakan negara yang terkena dampak dari penyebaran virus covid19, telah melakukan beberapa langkah penanggulangan dalam mencegah penyebaran virus SARS-Cov-2. Adapun beberapa langkah kebijakan telah diterapkan pemerintah Indonesia dalam mencegah penyebaran virus ini, diantaranya, yaitu *physical quarantine* mencegah terjadinya kontak fisik untuk meredam penyebaran virus dan *lockdown* di daerah-daerah yang termasuk dalam zona merah penyebaran virus, (Nurkholis, 2020). Selain itu, mengeluarkan surat edaran Nomor HK.02.02/I/385/2020 Tahun 2020 tentang memakai masker dan menyediakan Sarana Cuci Tangan Pakai Sabun (CTPS) untuk mencegah penularan Covid-19. Salah satu protokolnya mengatur bahwa setiap individu wajib menggunakan masker pada saat melakukan aktivitas di luar rumah. Dalam beberapa kasus, seseorang tidak mengetahui bahwa dirinya terinfeksi dan dapat menularkan

penyakit tersebut kepada orang lain. Dengan bantuan sistem deteksi masker, orang yang tidak mematuhi penggunaan masker dapat dideteksi, yang memungkinkan identifikasi dini dapat dilakukan terhadap kasus potensial dan pengambilan langkah-langkah yang diperlukan untuk mencegah penyebaran lebih lanjut.

Pendekatan yang banyak digunakan dalam bidang *computer vision* saat ini adalah *Convolutional Neural Network* (CNN) sebagai algoritma dalam *Deep Learning* yang populer untuk klasifikasi citra. Beberapa arsitektur yang dimiliki algoritma CNN diantaranya seperti Alexnet, ResNet, VGG, EfficientNet, dll. Dalam kompetisi *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC) tahun 2015 Arsitektur ResNet memiliki *error rates paling rendah jika dibanding dengan arsitektur lainnya*. Arsitektur ini pada kompetisi ILSVRC mencapai performa yang baik sehingga mampu menjuari kompetisi tersebut (Russakovsky *et al.*, 2015). ILSVRC adalah kompetisi tahunan yang mempertandingkan berbagai arsitektur CNN untuk klasifikasi citra. Metode SVM merupakan metode klasifikasi karena dapat mengklasifikasi dengan data yang berdimensi tinggi dan memberikan hasil percobaan yang menunjukkan keefektifan. Selain itu, SVM dapat melakukan pengenalan pola data yang *nonlinier* terpisah (Cortes, Vapnik and Saitta, 1995).

Beberapa penelitian terkait telah dilakukan untuk deteksi masker pada wajah. Sebagai contoh, penelitian yang menggabungkan *transfer learning* dengan *machine learning*, yaitu VGG19 & KNN dengan akurasi 96.65%, MobileNetV2 & KNN dengan akurasi 94.92%, Xception & SVM dengan akurasi 94.57% dan MobileNetV2 & SVM 97.11% pada penelitian *Control The Covid-19: Face Mask Detection Using Transfer Learning* (Oumina, El Makhfi and Hamdi, 2020) Penggabungan metode ini juga dilakukan dengan metode ResNet50 dan SVM

dengan akurasi mencapai 99.49% (Loey *et al.*, 2021) dan MobileNetV2 dengan Global Polling Blob dengan akurasi 99% (Venkateswarlu, Kakarla and Prakash, 2020). Selain dengan *Machine Learning* ada juga dilakukan penggabungan antara *transfer learning* dengan *transfer learning* seperti penggabungan ResNet50 dengan MobileNet dengan akurasi 98.20% (Sethi, Kathuria and Kaushik, 2021). Penelitian berikutnya tidak melakukan penggabungan atau pendeteksian dilakukan dengan satu metode saja yaitu MobileNetV2 dengan tingkat akurasi 82.76% (Sanjaya and Rakhmawan, 2020) dan CNN Standar dengan akurasi 96.90% (Negi *et al.*, 2020).

Pada penelitian-penelitian sebelumnya, tidak dilakukannya *preprocessing* pada *input image* padahal perubahan *image* dari RGB ke HSV dapat menangani skenario yang kompleks dengan lebih baik, yang dapat berdampak signifikan pada kinerja metode yang digunakan (Moreira *et al.*, 2022). Selain itu, menambahkan *preprocessing* pada *image* yaitu pengubahan *colorspace image* dari RGB ke HSV mampu menghemat waktu pelatihan dan meningkatkan akurasi model *transfer learning* VGG19 (Nguyen, Nguyen and Ngo, 2022). Karenanya pada penelitian ini penulis menggunakan *Hybrid model Transfer Learning* ResNet50 dengan *Support Vector Machine* (SVM) untuk deteksi masker pada *image* wajah, dimana terdapat *preprocessing* pada data *image* dari RGB ke HSV. Model *Transfer Learning* ResNet50 akan ditugaskan untuk proses ekstraksi fitur dan SVM pada proses klasifikasi wajah memakai masker atau tidak.

1.2 Pernyataan Masalah

Berdasarkan uraian latar belakang di atas, maka pernyataan masalah pada penelitian ini adalah

1. Apa model terbaik dari *transfer learning* ResNet50 dengan SVM untuk membedakan masker pada *image* wajah.
2. Bagaimana meningkatkan akurasi pendeteksian masker pada *image* wajah pada model *transfer learning* ResNet50 dengan SVM.

1.3 Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah

1. Mendapatkan model terbaik dalam deteksi masker pada *image* wajah.
2. Dapat mengetahui proses yang mempengaruhi tingkat akurasi pada model *transfer learning* ResNet50 dengan SVM dalam melakukan deteksi masker pada *image* wajah.

1.4 Manfaat Penelitian

Manfaat yang dapat diberikan dari penelitian ini ialah sebagai berikut:

1. Dapat digunakan dalam keamanan publik atau area yang memerlukan identifikasi wajah individu untuk memastikan bahwa orang-orang mematuhi peraturan penggunaan masker.
2. Memudahkan pengawasan dan penegakan kebijakan terhadap penggunaan masker di tempat-tempat umum.
3. Meningkatkan kesadaran tentang pentingnya penggunaan masker untuk melindungi diri dari berbagai penyakit menular khususnya Covid-19.

1.5 Ruang Lingkup Masalah

Untuk memberikan batasan yang jelas dalam penelitian ini maka ruang lingkup penelitian ini hanya meliputi:

1. Dimensi untuk data gambar yang akan digunakan baik itu data latih dan data uji adalah 128 x128 *pixel*.
2. *Image* data latih maupun data uji ada yang berupa data segmentasi atau *image* yang diedit (image wajah yang ditempelkan masker)
3. Keluaran yang dihasilkan adalah mendeteksi apakah wajah tersebut menggunakan masker atau tidak.

BAB II

STUDI PUSTAKA

1.1 Deteksi Masker Pada Image Wajah

(Kosasih, 2020) melakukan penelitian untuk pengenalan wajah yang dimana metode yang digunakan ialah metode Isomap. Metode ini digunakan untuk mentransformasi *vector* tersebut menjadi *vector* fitur wajah. Setelah fitur wajah di peroleh, berikutnya dilakukan pengujian terhadap data uji menggunakan algoritma KNN untuk mengklasifikasikannya. Dari hasil klasifikasi didapatkan tingkat akurasi sebesar 83.33%.

(Oumina, El Makhfi and Hamdi, 2020) mengusulkan pendeteksian wajah menggunakan masker atau tidak berdasarkan pemilihan fitur menggunakan CNN antara lain VGG19, Xception, MobileNetV2 yang kemudian diklasifikasikan menggunakan KNN dan SVM. Pada penelitian ini dilakukan *eksperiment* antara metode ekstraksi ciri dengan metode klasifikasi yang saling berpasangan seperti VGG19 dengan akurasi KNN mencapai 96,65%, Kemudian MobileNetV2 dengan akurasi KNN sebesar 94,92%, Kemudian Xception dengan akurasi SVM mencapai 94,57% dan MobileNetV2 dengan SVM menghasilkan akurasi tertinggi diantara pasangan lainnya yaitu mencapai 97,11%.

(Yaday, 2020) mengusulkan sebuah sistem yang membantu keselamatan orang-orang di tempat umum dengan memantau secara otomatis apakah mereka menjaga jarak atau tidak dan menggunakan masker atau tidak. Sistem yang diusulkan menggunakan pendekatan *transfer learning*. Pada pendekatan *transfer learning*, penulis menggunakan arsitektur MobileNetV2 dan SSD untuk

mendeteksi jumlah orang untuk jarak jauh dan mendeteksi masker pada wajah secara *real time*. Akurasi yang dihasilkan mencapai 91,7%.

(Negi *et al.*, 2020) mengusulkan untuk pendekteksian masker pada wajah dilakukan dengan menggunakan metode klasifikasi yaitu CNN. Sebelum diproses dengan CNN, data citra diproses terlebih dahulu dan ditambah yang diubah ukurannya menjadi 224 x 224 x 3. Penambahan data yang digunakan dalam penelitian ini memungkinkan untuk meningkatkan data secara dramatis untuk model pelatihan tanpa benar-benar mengumpulkan data baru. Setiap citra masukan yang telah di *augmentasi* akan melewati *convolutional layer* dan dengan menggunakan *softmax* mendefinisikan citra dengan nilai antara 0 dan 1. Dari hasil pengujian pada penelitian ini mencapai akurasi sebesar 96,90% untuk mengklasifikasikan masker pada wajah atau bukan.

(Venkateswarlu, Kakarla and Prakash, 2020) mengusulkan metode MobileNet dan Global Pooling Block untuk mendeteksi wajah memakai masker. MobileNet terlatih menghasilkan peta fitur. Kemudian Global Pooling Block peta fitur *multi* dimensi ini diubah menjadi *vektor* satu dimensi yang memiliki 64 fitur. Selanjutnya *softmax layer* dengan dua neuron mengambil 64 fitur dan melakukan klasifikasi biner yaitu masker atau tidak memakai masker. Keakuratan metode ini menghasilkan 99% dari data sebenarnya, dan 100% untuk data simulasi.

(Sanjaya and Rakhmawan, 2020) pendekteksian masker pada wajah pada penelitian ini dikembangkan dengan algoritma MobileNetV2 yang digunakan untuk klasifikasi citra. Dengan MobileNetV2, dibangun model yang akan digunakan untuk proses prediksi/klasifikasi. Dari model yang dihasilkan diuji di 25 kota di

Indonesia. untuk akurasi tertinggi mencapai 82,76% untuk kota jambi dan terendah mencapai 64,14% di kota surabaya.

(Loey *et al.*, 2021) mengusulkan model *hybrid* yaitu pada *preprocessing* menggunakan Resnet50 yang merupakan model ekstraksi ciri citra dan setelah itu menggunakan algoritma Decision Tree, SVM dan Ensemble untuk proses klasifikasi masker pada wajah. Penulis menyajikan perbandingan antara Algoritma Decision Tree, SVM dan Ensemble menggunakan tiga set data dimana klasifikasi menggunakan SVM lebih unggul dari Decesion Tree dan algoritma Ensemble. Akurasi dari masing-masing dataset tersebut adalah *Real-world Masked Face Dataset* (RMFD) mencapai 99,64%, *Simulated Masked Face Dataset* (SMDF) mencapai 99,49% dan *Labeled Faces in the Wild* (LFW) mencapai 100%.

(Sethi, Kathuria and Kaushik, 2021) melakukan penelitian deteksi masker pada wajah di area publik untuk mengurangi penyebaran Covid-19. Teknik yang diusulkan menangani oklusi secara efisien dalam situasi padat dengan memanfaatkan satu dan dua *detektor* tahap dalam *preprocessing*. Model yang digunakan dalam *preprocessing* ini adalah ResNet50, Alexnet dan MobileNet. Resnet50 digunakan untuk mendapatkan objek berupa leher dan kepala. Sedangkan AlexNet dan MobileNet digunakan untuk mendapatkan hasil terbaik untuk pendeteksian masker pada wajah. Teknik yang diusulkan mencapai akurasi 98,2%

(Suresh, Palangappa *and* Bhuvan, 2021) dalam penelitiannya mengusulkan untuk membangun model deteksi masker pada wajah secara *real time* menggunakan CNN. Dimana pendeteksian dilakukan secara otomatis di ruang publik seperti sekolah, pasar, perkantoran dll. Penulis menggunakan algoritma MobileNet untuk mendeteksi seseorang menggunakan masker atau tidak. Sedangkan untuk

klasifikasi menggunakan algoritma atau arsitektur MobileNet dari OpenCV. Model ini berfungsi sebagai tulang punggung yang kemudian melatih model menggunakan *TensorFlow*. Akurasi pada penelitian ini mencapai 97,75%.

(Dewi and Ismawan, 2021) Pada Penelitian ini, metode CNN memiliki *pre-trained neural network* sebagai metode ekstraksi fitur yang kemudian disimpan dan digunakan sebagai pengenalan atau klasifikasi. Proses klasifikasi dilakukan dengan membandingkan nilai citra yang ada di database dengan data citra inputan yang diuji. Klasifikasi ini menggunakan *Linear kernel SVM*. Akurasi pada penelitian dengan menggunakan metode ini ialah sebesar 99.84%.

(Huda *et al.*, 2022) penelitian ini dilakukan dengan menggunakan metode *preprocessing* citra sel darah merah dengan cara *resize* gambar, *augmentasi* gambar, dan *reduksi* fitur gambar menggunakan metode *Principal Component Analysis (PCA)*, yang dimana hasil dari fitur ini *reduksi* ini tersebut selanjutnya diklasifikasikan menggunakan metode CNN yaitu Arsitektur DenseNet dengan akurasi terbaik yaitu 98.30%.

1.2 Kerangka Teori

Pada sub bab ini disajikan kerangka teori dari hasil studi literatur dengan cara mengkategorikan parameter atau variabel sebagai input data kerangka teori dan mencantumkan metode apa yang digunakan dan apa keluaran dari hasil kerangka teori tersebut.

Kerangka teori pada penelitian sebelumnya dengan data yang sama pada penelitian ini diberikan pada Tabel 2.1.

Tabel 2.1 Kerangka Teori dengan Data Sama

No	Nama & Tahun	Judul	Metode		Akurasi
			Ekstraksi Fitur	Klasifikasi	
1	(Oumina, El Makhfi and Hamdi, 2020)	<i>Control The COVID-19 Pandemic: Face Mask Detection Using Transfer Learning</i>	VGG19	KNN	96.65%
			MobileNetV2	KNN	94.92%
			Exception	SVM	94.57%
			MobileNetV2	SVM	97.11%
2	(Venkateswarlu, Kakarla and Prakash, 2020)	<i>Face mask detection using MobileNet and Global Pooling Block</i>	MobileNet	Global Polling Block	99%

Sedangkan kerangka teori pada penelitian sebelumnya dengan data yang berbeda dengan data penelitian ini diberikan pada tabel 2.2.

Tabel 2.2 Kerangka Teori dengan Data Berbeda

No	Nama & Tahun	Judul	Metode		Akurasi
			Ekstraksi Fitur	Klasifikasi	
1	(Kosasih, 2020)	Kombinasi Metode Isomap dan KNN pada <i>Image Processing</i> untuk Pengenalan Wajah	Isomap	KNN	83.33%
2	(Negi <i>et al.</i> , 2020)	<i>Face Mask Detection Classifier and Model Pruning with Keras-Surgeon</i>	CNN		96.90%
3	(Sanjaya and Rakhmawan, 2020)	<i>Face Mask Detection Using MobileNetV2 in The Era of COVID-19 Pandemic</i>	MobileNetV2		82.76%
4	(Dewi and Ismawan, 2021)	<i>Implementasi Deep Learning Menggunakan CNN Untuk Sistem Pengenalan Wajah</i>	Pre-trained Neural Network	SVM	99.84%
5	(Yadav, 2020)	<i>Deep Learning based Safe Social Distancing and Face Mask Detection in Public Areas for COVID-19 Safety Guidelines Adherence</i>	MobileNetV2	SSD	91.70%

No	Nama & Tahun	Judul	Metode		Akurasi
			Ekstraksi Fitur	Klasifikasi	
6	(Huda <i>et al.</i> , 2022)	Klasifikasi Malaria Menggunakan Citra Sel Darah Merah dengan Algoritma CNN	PCA	DenseNet	98.30%
7	(Loey <i>et al.</i> , 2021)	<i>A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic</i>	ResNet50	SVM	99.49%
8	(Sethi, Kathuria and Kaushik, 2021)	<i>Face mask detection using deep learning: An approach to reduce risk of Coronavirus spread</i>	ResNet50	MobileNet, AlexNet	98.20%
9	(Suresh, Palangappa and Bhuvan, 2021)	<i>Face Mask Detection by using Optimistic Convolutional Neural Network</i>	CNN	MobileNet	97.75%

Dari kerangka teori pada tabel 2.1 dan tabel 2.2 untuk deteksi wajah menggunakan masker atau tidak, data masukan diambil dari dataset citra orang yang memakai masker dan citra orang yang tidak menggunakan masker. Sebagian besar pengolahan data dilakukan dengan menggabungkan dua algoritma. Pada tahap *preprocessing* menggunakan CNN dengan arsitektur yang berbeda. Kemudian tahap selanjutnya adalah proses klasifikasi untuk mendeteksi wajah menggunakan masker atau tidak. Adapun urutan dengan akurasi tertinggi adalah sebagai berikut:

Tabel 2.3 Ranking akurasi dari yang terbaik

No	Nama	Tahun	Metode	Akurasi
1	Dewi & Ismawan	2021	PreTrained - SVM	99.84%
2	Loey dkk	2021	ResNet50 - SVM	99.49%

No	Nama	Tahun	Metode	Akurasi
3	Venkateswarlu dkk	2020	MobileNet - GPB	99%
4	Sethi dkk	2021	ResNet50 – AlexNet – MobileNet	98.20%
5	Huda dkk	2022	PCA – DenseNet	98.30%
6	Shures dkk	2021	MobileNet	97.75%
7	Oumina dkk	2020	MobileNetV2 - SVM	97.11%
8	Negi dkk	2020	Standart CNN	96.90%
9	Oumina dkk	2020	VGG19 - KNN	96.65%
10	Oumina dkk	2020	MobileNetV2 - KNN	94.92%
11	Oumina dkk	2020	Exception - SVM	94.57%
12	Yadav	2022	MobileNetV2 - SSD	91.7%
13	Kosasih	2020	Isomap - KNN	83.33%
14	Sanjaya & Rachmawan	2020	MobileNetV2	82.96%

Pada tabel 2.3, *hybrid* model lebih banyak digunakan untuk mendapatkan hasil akurasi yang terbaik seperti pada pasangan CNN *Pre-Trained Neural Network* dan SVM dengan akurasi mencapai 99.84% namun objek yang di teliti adalah pengenalan wajah manusia. Pada urutan kedua ada *hybrid* CNN ResNet50 dan SVM dengan akurasi mencapai 99.49% yang di mana objek citra yang di teliti adalah masker pada wajah. Sedangkan untuk data yang sama dengan peneliti akurasi terbaik mencapai 99% dari *hybrid* MobileNet dengan Global Polling Block.

Oleh karena itu, penelitian ini dengan penambahan data dari data penelitian sebelumnya akan ditindaklanjuti dengan menggunakan *hybrid* model *transfer learning* Resnet50 dan *Support Vector Machine* (SVM) untuk deteksi masker pada wajah, dengan *preprocessing* segmentasi *image* ke HSV(Hue Saturation Value).

Adapun perbaruan dari penelitian ini ada penambahan *preprocessing* pada *input image* yaitu perubahan *colorsapace image* dari RGB ke HSV yang mampu menghemat waktu untuk pelatihan dan meningkatkan akurasi model *transfer learning* VGG19 (Nguyen, Nguyen and Ngo, 2022). *Colorspace image* HSV menangani skenario yang kompleks jauh lebih baik, yang dapat memiliki pengaruh

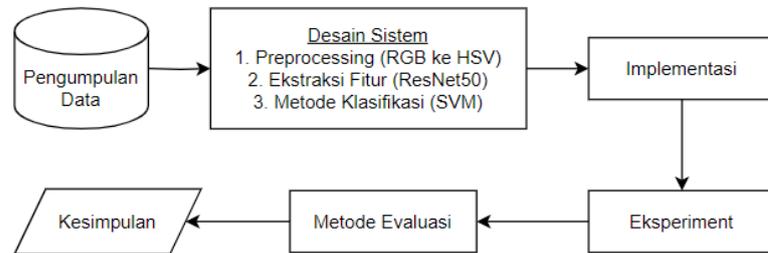
besar pada performa metode yang digunakan (Moreira *et al.*, 2022). Sedangkan pemilihan SVM sebagai metode klasifikasi karena mampu melakukan klasifikasi dengan data yang berdimensi tinggi serta memberikan hasil uji coba yang menunjukkan keefektifan. Selain itu SVM mampu melakukan pengenalan pola data yang *non-linear* terpisah (Cortes, Vapnik and Saitta, 1995).

BAB III

DESAIN PENELITIAN

3.1 Prosedur Penelitian

Prosedur penelitian yang akan dilakukan untuk deteksi masker pada wajah terdiri dari pengumpulan data, perancangan sistem, dan implementasi, *eksperiment*, evaluasi, dan kesimpulan. Berikut ini adalah tahapan yang dilakukan pada penelitian seperti pada gambar 3.1.



Gambar 3.1 Prosedur penelitian

3.1.1 Pengumpulan Data

Menurut metode yang digunakan dalam pengumpulan data pada penelitian ini, data dapat diklasifikasikan sebagai data sekunder, dimana data tidak diperoleh peneliti secara langsung. Namun data ini dalam bentuk dokumen atau arsip yang dimiliki oleh lembaga atau seseorang yang menjadi subjek penelitian.

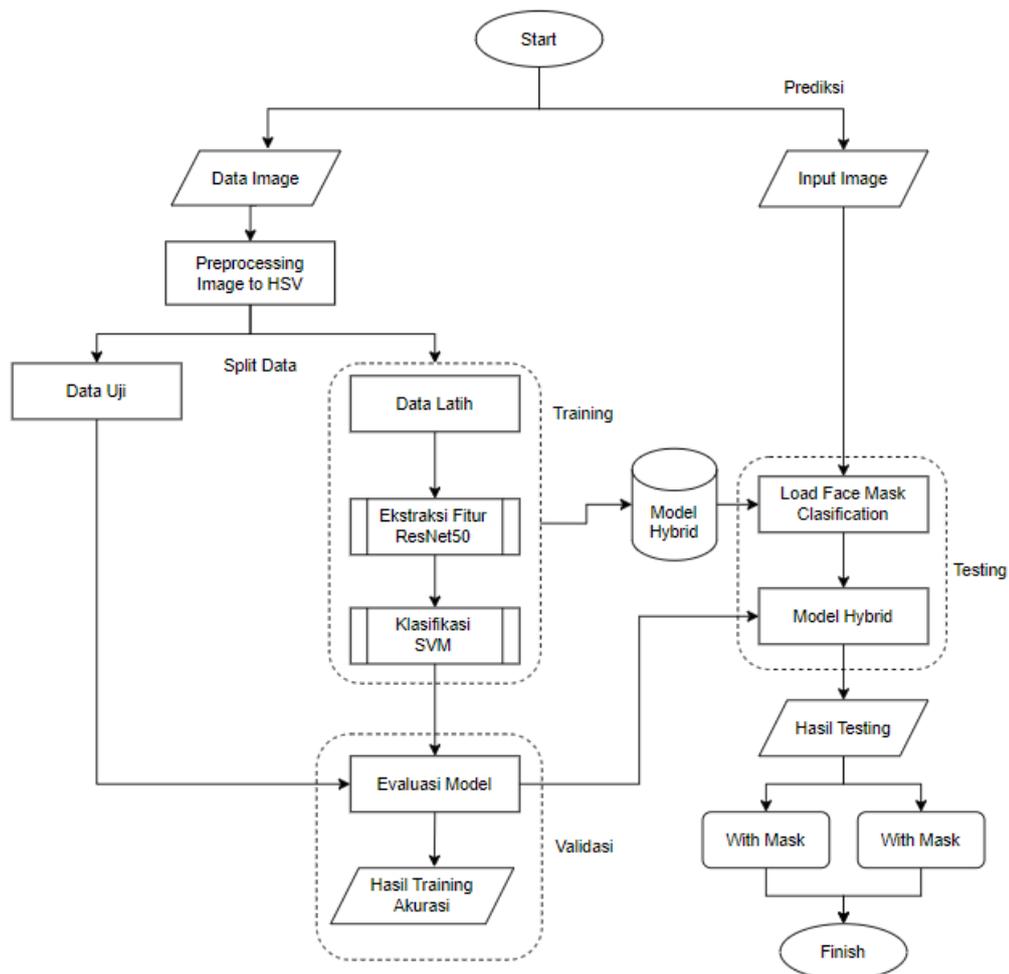
Data yang akan digunakan bersumber dari data penelitian sebelumnya yaitu dari <https://github.com/prajnasb> dengan total 1376 data baik yang memakai masker atau tidak memakai masker (Prajna Bhandary, 2020). Kemudian data ini ditambahkan dengan data berupa gambar berdimensi 128x128 *pixel*. Adapun sumber dari data tambahan pada penelitian ini merupakan data publik yaitu data

deteksi masker wajah yang berasal dari *website* <https://kaggle.com> dalam bentuk siap pakai berkaitan dengan penelitian yang akan dilakukan. Jumlah data *image* yang menggunakan masker sebanyak 2965 file citra dan jumlah data yang tidak menggunakan masker sebanyak 2987 file citra (Vijay Kumar, 2021). Jadi total data yang di gunakan pada penelitian ini adalah 7328 data.

Pelatihan terhadap metode dilakukan dengan teknik *Split Validation* dimana teknik ini dengan secara acak membagi data menjadi dua bagian, yaitu data *training* dan data *testing*. Kemudian akan dilakukan *training* dan *testing* berdasarkan berdasarkan *split ratio* dengan 90:10, 75:25, 60:40, 50:50 dan 40:60 dengan total data 7328.

3.1.2 Desain Sistem

Model yang disajikan mencakup tiga komponen utama, yang pertama *preprocessing* data *image*, yaitu pengubahan *colorspace* dari RGB ke HSV. Komponen kedua adalah *Transfer Learning* model ResNet50 sebagai ekstraktor fitur dan komponen ketiga adalah pembelajaran mesin yaitu SVM. Pada proses klasifikasi, lapisan terakhir di ResNet-50 akan dihapus dan diganti dengan SVM untuk klasifikasi untuk meningkatkan kinerja model pada penelitian ini.



Gambar 3.2 Desain Sistem

3.1.2.1 Hue Sturation Value (HSV)

Penambahan *preprocessing* data *image* yaitu segmentasi *image* dari RGB ke HSV, dapat menghemat waktu pelatihan dan meningkatkan akurasi model *transfer learning* VGG19 (Nguyen, Nguyen and Ngo, 2022). Untuk mengubah data *image* RGB ke HSV, masukan *image* dengan ukuran 128x128 *pixel* yang kemudian diproses menggunakan rumus sebagai berikut:

$$r = \frac{R}{(R+G+B)}, \quad g = \frac{G}{(R+G+B)}, \quad b = \frac{B}{(R+G+B)} \quad (3.1)$$

$$V = \max(r, g, b) \quad (3.2)$$

$$S = \begin{cases} 0, & V = 0 \\ 1 - \frac{\min(r,g,b)}{V}, & V > 0 \end{cases} \quad (3.3)$$

$$H = \begin{cases} 0, & \text{jika } S = 0 \\ \frac{60*(g-b)}{S*V}, & \text{jika } V = r \\ 60 * \left[2 + \frac{b-r}{S*V} \right], & \text{jika } V = g \\ 60 * \left[4 + \frac{r-g}{S*V} \right], & \text{jika } V = b \end{cases} \quad (3.4)$$

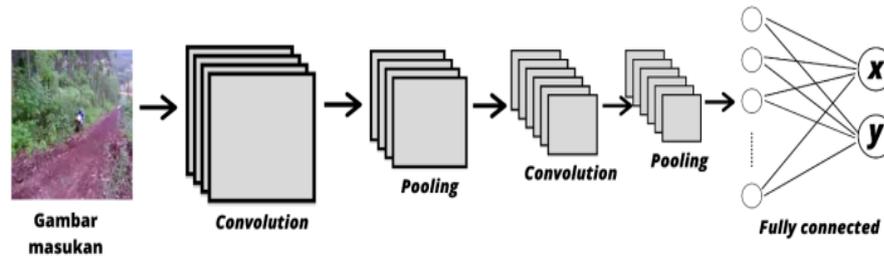
Berikut ini adalah contoh *colorspace image* yang telah di ubah ke HSV terlihat pada gambar 3.3



Gambar 3.3 *colorspace* HSV

3.1.2.2 Convolutional Neural Network (CNN)

Karena kedalaman jaringan, menjadikan bagian dari jenis *deep learning*. CNN adalah penggabungan dari beberapa lapisan proses pada operasi konvolusi yang diadaptasi dari sistem saraf biologis manusia yang menggunakan beberapa elemen secara paralel. Dengan CNN *neuron* nampak dalam bentuk dua dimensi, sehingga pada pemrosesan input berupa gambar CNN sangat cocok digunakan, efisien dan banyak digunakan dalam aplikasi *computer vision*. Berikut arsitektur CNN diilustrasikan pada Gambar 3.4



Gambar 3.4 Arsitektur CNN Secara Umum

Sumber: (Faiz Nashrullah, Suryo Adhi Wibowo and Gelar Budiman, 2020)

1. Lapisan Input

Lapisan input ini menerima gambar mentah dan di teruskan ke lapisan berikutnya untuk ekstraksi fitur.

2. Convolution

Pada layer ini, berbagai filter diterapkan pada gambar untuk mendapatkan fitur gambar. Fitur-fitur ini digunakan untuk menghitung kesamaan pada setiap proses pengujian. Metode untuk menghitung proses konvolusi adalah mengalikan *pixel* gambar dengan filter untuk mendapatkan *output feature maps*.

Pada operasi konvolusi terdiri dari *padding*, *stride*, dan filter sebagai parameter.

Perhitungan ukuran *feature maps* dilakukan dengan cara sebagai berikut:

$$Output = \frac{n+2p-f}{s} \quad (3.5)$$

Keterangan:

n = Panjang / Tinggi Input

f = Panjang / Tinggi Kernel Filter

p = Padding

s = stride

- a. *Padding same*, juga dikenal sebagai *zero padding*. Cara kerjanya menambahkan *pixel* bernilai 0 disekeliling *matriks* untuk menghindari kehilangan *pixel* citra saat pergeseran kernel saat proses konvolusi.

- b. *Padding valid* mengambil titik tengah pada operasi konvolusi saat menentukan nilai *matriks* pada *feature maps*. Perhitungannya menggunakan persamaan diatas, dimana nilai $p = 0$.
- c. *Strides* adalah parameter yang menentukan jumlah pergeseran filter. Jika nilai *strides* 2, maka filter akan bergeser sebanyak 2 pixel secara horisontal dan vertikal. Semakin kecil pergeseran *stride*, maka fitur yang akan didapat akan semakin detail.

3. Batch Normalization

Batchnormalization diperuntukan mengurangi pergeseran kovarian atau memperlancar distribusi pada nilai input yang terus berubah selama proses pelatihan akibat perubahan pada *layer* sebelumnya. Fungsi ini berjalan sebelum fungsi aktivasi pada lapisan input. Proses *batchnormalization* memiliki proses *zero-center* input yang menghitung *mini mean* dan *minibatch* varian kemudian dinormalisasi dan terakhir dihitung *scale & shift* nya. Adapun penjelasan tiap proses yang dilakukan yaitu sebagai berikut:

- a. *Zero-center input*, Proses ini menghitung rata-rata input dan standar deviasi (input *minibatch*), oleh karena itu disebut *batchnormalization* karna menormalkan data *minibatch* untuk mempercepat proses pelatihan dan meningkatkan *learning rates* pada saat pembuatan model. Terdapat empat proses yaitu menghitung *minibatch mean* dan *minibatch* varian, normalisasi input, dan *scale and shift*.

Persamaan *Minibatch Mean*

$$\mu_B \leftarrow \frac{1}{m} \sum_i^m x_i \quad (3.6)$$

Persamaan *Minibatch Varian*

$$\sigma_B \leftarrow \frac{1}{m} \sum_i^m (x_i - \mu_B) \quad (3.7)$$

Keterangan :

m = jumlah dataset pelatihan

μ_B = nilai rata-rata dari *batch*

σ_B = standar deviasi dari *minibatch*

b. Persamaan Normalisasi Input

$$\hat{x}_i \leftarrow \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \quad (3.8)$$

c. Persamaan *Scale dan Shift*

$$y_i \leftarrow \gamma \hat{x}_i + \beta = BN\gamma, (x_i) \quad (3.9)$$

Keterangan :

y_i = *output scaled dan shifted* dari operasi *batchnormalization*

γ = *scale* dengan nilai = 1

β = *shift* dengan nilai = 0

4. **Rectified Linear Unit (ReLU)**

Lapisan setelah konvolusi adalah operasi *Rectified Linear Unit*. Lapisan ini menggantikan angka negatif dari lapisan konvolusi dengan 0 (nol), yang membantu untuk *training* yang lebih cepat dan efektif.

$$f(x_i) = \max(0, x_i) \begin{cases} x_i, & \text{if } x_i \geq 0 \\ 0, & \text{if } x_i < 0 \end{cases} \quad (3.10)$$

Keterangan:

(x_i) = nilai dari ReLU *activation*

X_i = nilai *matriks* dari citra

5. **Pooling**

Lapisan *pooling* menerima fitur hasil ekstraksi untuk mengurangi dimensi citra dan mengurangi parameter dalam penyimpanan informasi penting. Tujuannya adalah menjaga nilai maksimum dari setiap lapisan. *Pooling* dibagi menjadi *Maxpooling* dan *Global Average Pooling*. *Maxpooling* mengambil

nilai maksimum pada hasil konvolusi sedangkan *Global Average Pooling*, mengambil nilai rata rata pada hasil konvolusi.

$$Output_{maxpool} = \frac{n-f}{s} + 1 \quad (3.11)$$

Keterangan:

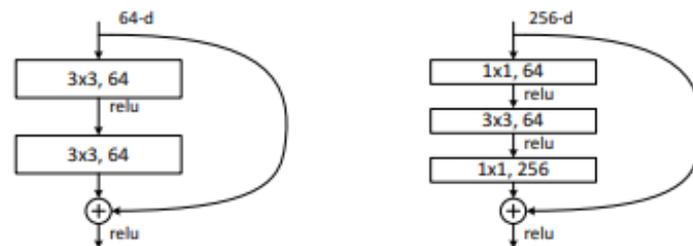
n = Panjang / Tinggi Input

f = Panjang / Tinggi Kernel Filter

s = stride

3.1.2.3 ResNet50

ResNet adalah jenis *deep network* berbasis pembelajaran residual. Pembelajaran semacam ini dapat memfasilitasi pelatihan jaringan dengan mempertimbangkan *input layer* sebagai referensi (He *et al.*, 2015).

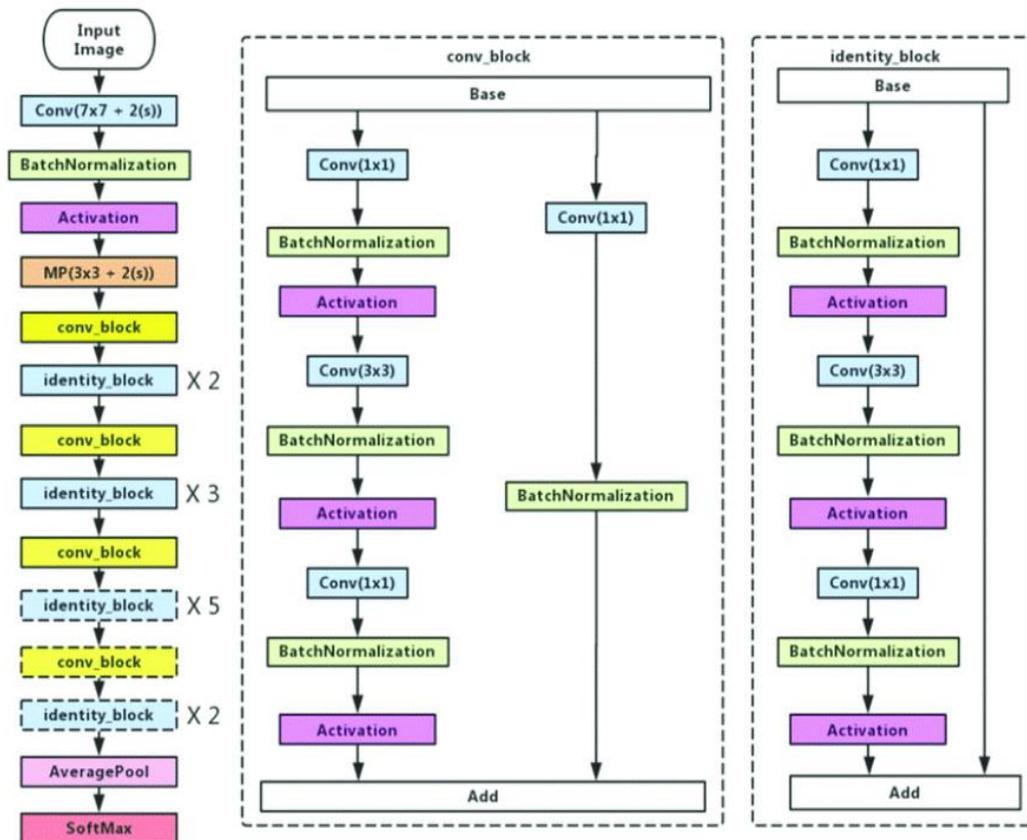


Gambar 3.5 Residual Network (ResNet)
(Sumber He, *et al* 2015)

ResNet50 merupakan bagian dari varian ResNet yang memiliki 50 layer. ResNet versi sebelumnya *skip connection* dilakukan sebanyak 2 *layer*, sedangkan untuk ResNet50 dengan 3 *layer* dan terdapat 1x1 *convolution layer*. ResNet50 memperkenalkan konsep baru yaitu *shortcut connections*. Konsep *shortcut connections* ResNet50 ini terkait dengan masalah *vanishing gradient* yang terjadi saat mencoba memperdalam struktur jaringan.

Dalam konsep *shortcut connections*, fitur *input* dari *layer* sebelumnya dijadikan input pada *output* dari *layer* berikutnya. Metode ini dilakukan agar fitur-

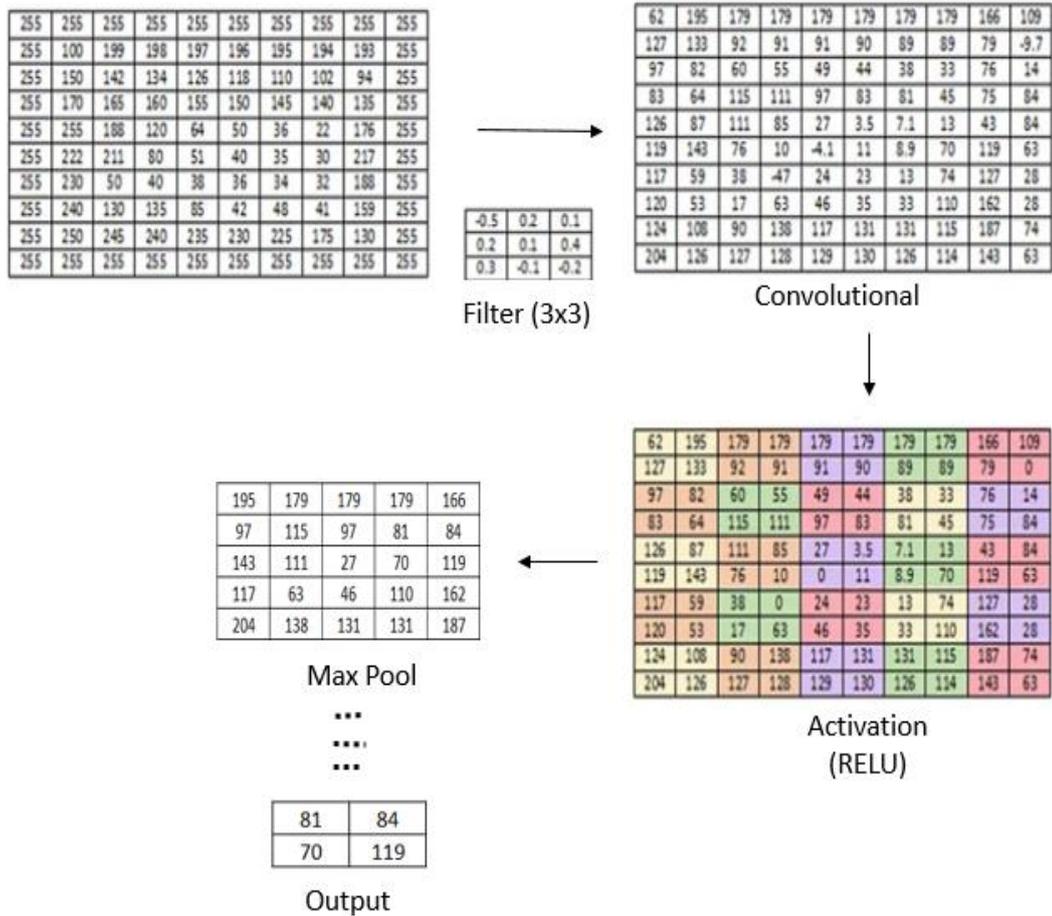
fitur penting pada proses konvolusi tidak hilang. ResNet50 terdiri dari lima proses konvolusi yang dilanjutkan dengan *average pooling* dan terakhir dengan *fully connected layer*.



Gambar 3.6 Arsitektur ResNet50

Sumber: (Faiz Nashrullah, Suryo Adhi Wibowo and Gelar Budiman, 2020)

Sebagai contoh adapun detail proses untuk mendapat output dari ResNet50 adalah yang terlihat pada gambar 3.7.



Gambar 3.7 Perhitungan ResNet50

3.1.2.4 Support Vector Machine (SVM)

SVM menggunakan pemetaan *nonlinier* untuk mengubah data pelatihan asli menjadi dimensi yang lebih tinggi. Dalam dimensi baru ini, akan mencari *hyperplane* untuk memisahkan secara linier dan dengan pemetaan *nonlinier* yang tepat ke dimensi lebih tinggi, data dari dua kelas selalu dapat dipisahkan dengan *hyperplane* tersebut (Cortes, Vapnik and Saitta, 1995).

Dalam teknik ini, kita berusaha untuk mencari fungsi pemisah (*klasifier*) yang optimal yang dapat memisahkan dua kelas yang berbeda. Teknik ini berusaha

menemukan fungsi pemisah (*hyperplane*) terbaik diantara fungsi yang tidak terbatas jumlahnya untuk memisahkan dua macam obyek.

Dataset diberikan variabel x_i , sedangkan untuk kelas pada dataset diberikan variabel y_i . Metode SVM membagi dataset menjadi dua kelas. Kelas pertama yang dipisah oleh *hyperplane* bernilai 1, sedangkan kelas lainnya bernilai -1.

$$X_i \cdot W + b \geq 1 \text{ untuk } Y_i = 1 \quad (3.12)$$

$$X_i \cdot W + b \leq -1 \text{ untuk } Y_i = -1 \quad (3.13)$$

Keterangan:

X_i = data ke - i

W = nilai bobot *support vector* yang tegak lurus dengan *hyperplane*

b = nilai bias

Y_i = kelas data ke - i

Bobot *vector* (w) adalah garis vektor yang tegak lurus antara titik pusat kordinat dengan garis *hyperplane*. Bias (b) merupakan kordinat garis *relative* terhadap titik kordinat.

$$b = -\frac{1}{2}(w \cdot x^+ + w \cdot w^-) \quad (3.14)$$

$$w = \sum_{i=1}^n a_i y_i x_i \quad (3.15)$$

Keterangan:

b = nilai bias

$w \cdot x^+$ = nilai bobot untuk kelas data positif

$w \cdot x^-$ = nilai bobot untuk kelas data negatif

w = bobot vektor

a_i = nilai bobot data ke- i

y_i = kelas data ke- i

x_i = data ke- i

H_1 adalah *hyperplane* pendukung dari kelas +1 yang memiliki fungsi $w \cdot x + b = +1$.

$$\text{Margin} = |dH_1 dH_2| \frac{2}{\|w\|} \quad (3.16)$$

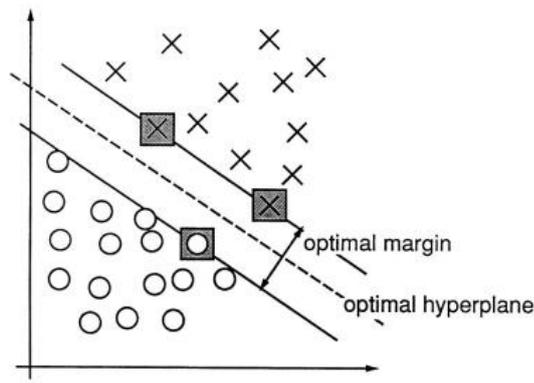
Keterangan :

$dH1$ = jarak *hyperplane* pendukung kelas +1
 $dH2$ = jarak *hyperplane* pendukung kelas -1

Kemudian untuk menentukan optimal *hyperplane* kedua kelas menggunakan persamaan berikut :

$$\text{Minimize } [w] = \frac{1}{2} ||w||^2 \quad (3.17)$$

$$\text{Dengan } (x_i \cdot w + b) - 1 \geq 0, i = 1, \dots, n \quad (3.18)$$



Gambar 3.8 *Support Vector Machine*
 Sumber: (Cortes, Vapnik and Saitta, 1995)

3.1.3 Eksperiment

Pada tahap *eksperiment* ini digunakan dua kerangka kerja, yang pertama adalah pembuatan dan pengujian model yang dilakukan di *Google Colab Processor GPU*, yang kedua implementasi model pada aplikasi berbasis *web* dengan bahasa *pyhon* yaitu *Flask*. Model akan di pasang di sebuah PC dengan spesifikasi CPU Intel Core i5 dengan RAM 8 GB. Adapun kasus uji yang akan dilakukan adalah sebagai berikut:

1. Pengujian akan dilakukan dengan data *image standart* yaitu RGB
2. Sebelum data dilakukan *train* dengan Model ResNet50 dan SVM data terlebih dahulu diubah menjadi *image HSV* pada *preprocessing*.

3. Model ResNet50 tidak dilakukan *Freeze* atau dengan kata lain membawa hasil model yang telah dilatih.
4. Model ResNet50 akan dilakukan *Freeze* dengan data dari penelitian ini.

Kasus uji ini dapat dilihat pada table 3.1.

Tabel 3.1 Kasus Uji

No	Data	ResNet50	ResNet50 & SVM
1	RGB	<i>Unfreeze</i>	<i>Unfreeze</i>
2	RGB	<i>Freeze</i>	<i>Freeze</i>
3	HSV	<i>Unfreeze</i>	<i>Unfreeze</i>
4	HSV	<i>Freeze</i>	<i>Freeze</i>

3.1.4 Metode Evaluasi

Hasil yang akan ditampilkan ialah model *transfer learning* ResNet50 dengan SVM yang telah dilatih. Pada tahap metode evaluasi akan dilakukan rekapitulasi hasil uji coba dan dilakukan analisis dengan pembuatan grafik dan tabel. Analisis ini digunakan untuk menunjukkan keberhasilan uji coba yang dilakukan oleh *hybrid* model ResNet50 dengan SVM, sehingga dapat digunakan untuk penelitian selanjutnya.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3.19)$$

$$Precision = \frac{TP}{TP+FP} \quad (3.20)$$

$$Recall = \frac{TP}{TP+FN} \quad (3.21)$$

$$F1\ Score = \frac{2 \cdot Precision \cdot Recall}{Precision+Recall} \quad (3.22)$$

Keterangan:

TP = *True Positif*

TN = *True Negatif*

FP = *False Positif*

FN = *False Negatif*

3.2 *Instrument* Penelitian

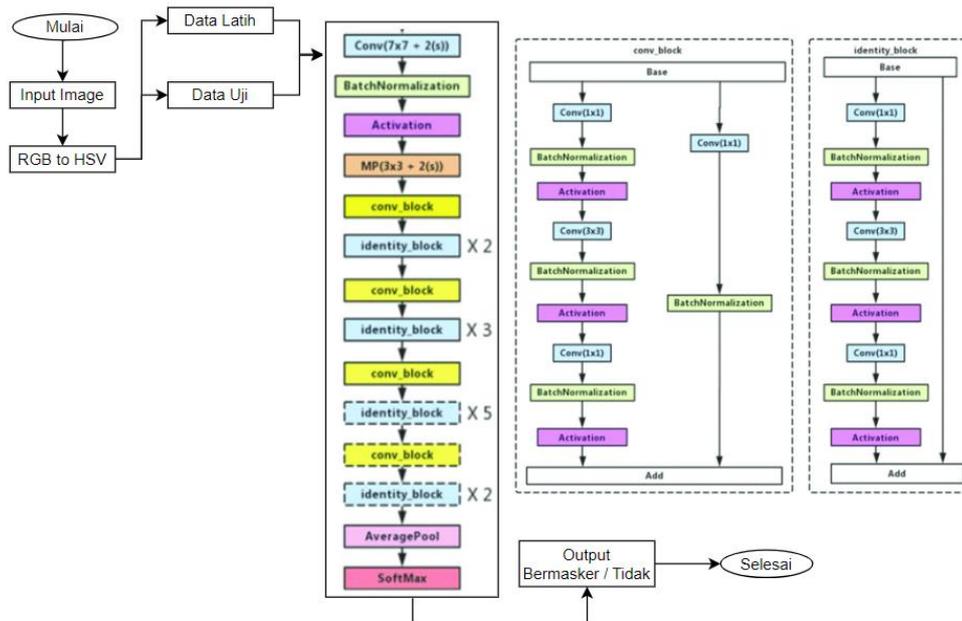
Bagian ini merupakan bagian penting dari sebuah penelitian. Baik tidaknya hasil uji coba yang dilakukan tergantung dari *instrument* penelitian. Dalam penelitian ini variabel bebasnya adalah data citra dengan dimensi 128x128 *pixel* yang masing-masing diberi label apakah memakai masker atau tidak. Data yang digunakan bersifat publik, dari data peneliti sebelumnya ditambah dengan data yang diperoleh dari Kaggle. Untuk variabel terikatnya adalah kinerja hasil penelitian yaitu tingkat kesalahan. Sedangkan variabel antara dalam penelitian ini adalah hasil deteksi masker wajah.

BAB IV

MODEL TRANSFER LEARNING RESNET50

4.1 Desain

Transfer learning umumnya mengacu pada proses dimana model yang dilatih pada suatu problem dapat digunakan dalam kasus lain karena adanya keterkaitan masalah. ResNet50 merupakan model *transfer learning* dari keluarga ResNet yang memiliki 50 *layer*. Untuk dapat melakukan klasifikasi *image* masker ResNet50 perlu melakukan *training* data, sehingga perlu disiapkan dataset *image* wajah yang memakai masker dan *image* wajah yang tidak memakai masker. Setelah itu data diolah dengan model ResNet50 yang dimana 1000 kelas pada ResNet50 akan ganti dengan menjadi 2 kelas. Adapun desain diagram alur dalam proses klasifikasi *image* wajah dengan ResNet50 yang terlihat pada gambar 4.1



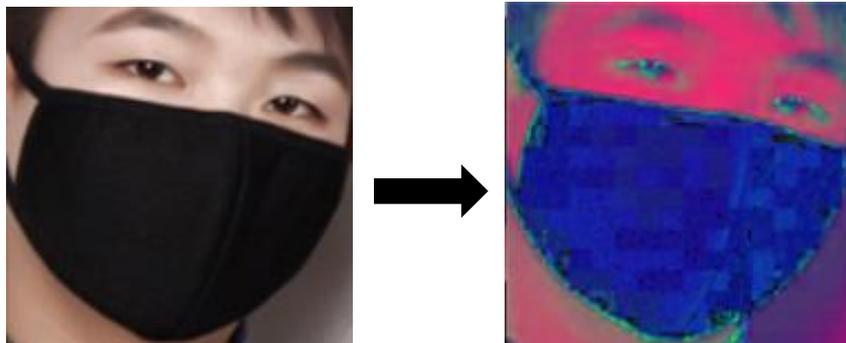
Gambar 4.1 Desain Proses Klasifikasi ResNet50

Sumber: diolah dari (Faiz Nashrullah, Suryo Adhi Wibowo and Gelar Budiman, 2020)

Pada gambar 4.1 terdapat proses *preprocessing* terhadap dataset image wajah yaitu RGB to HSV. Hal ini merupakan bagian dari upaya penulis menaikkan akurasi dari model ResNet50.

```
without_mask = os.listdir(os.getcwd() +
"/drive/MyDrive/dataset/without-mask")
for x in without_mask:
    imag = cv2.imread(os.getcwd() + "/drive/MyDrive/dataset/without-
mask/" + x)
    hsvImg = cv2.cvtColor(imag, cv2.COLOR_RGB2HSV)
    img_from_ar = Image.fromarray(hsvImg, 'HSV')
    resized_image = img_from_ar.resize((128, 128))
    data.append(np.array(resized_image))
    labels.append(0)
```

Dari baris *code* di atas akan menghasilkan image HSV yang akan di proses oleh model ResNet50. Konversi dari RGB ke HSV terlihat pada gambar 4.2.



4.2 Konversi RGB ke HSV

Setelah data image ini diubah dari RGB ke HSV kemudian dijadikan *array* untuk disimpan sesuai kelasnya masing-masing yaitu memakai masker atau tidak memakai masker. Selanjutnya data akan diproses dengan model ResNet50 dan dilakukan klasifikasi dengan *Activation Softmax*. Adapun *code* program untuk *load* data adalah sebagai berikut

```
s=np.arange(datamask.shape[0])
np.random.shuffle(s)
datamask=datamask[s]
labels=labels[s]
num_classes=len(np.unique(labels))
data_length=len(datamask)
(x_train, x_test) = datamask[(int)(0.25*data_length):],
datamask[::(int)(0.25*data_length)]
# HSV
x_train[0, :, :] = x_train.astype('float32')[0, :, :]/180
x_train[1:3, :, :] = x_train.astype('float32')[1:3, :, :]/255
x_test[0, :, :] = x_test.astype('float32')[0, :, :]/180
x_test[1:3, :, :] = x_test.astype('float32')[1:3, :, :]/255
(y_train, y_test)=labels[(int)(0.25*data_length):],
labels[::(int)(0.25*data_length)]
```

4.2 Uji Coba

Pada tahap ini sistem akan melakukan 4 pendekatan uji coba diantaranya yang pertama data image RGB – Model *Freeze*, RGB – Model *Unfreeze*, HSV – Model *Freeze*, HSV – Model *Unfreeze*. Code untuk menjalankan ResNet50 terlihat dibawah ini

```
pretrained_model =
tf.keras.applications.ResNet50(include_top=False,
input_shape=(128, 128, 3), classes=2, weights='imagenet',
pooling='avg')
```

Code untuk freeze model ResNet50 sebagai berikut

```
for layer in pretrained_model.layers:
    layer.trainable = False
```

Code untuk unfreeze sebagian model ResNet50 sebagai berikut

```
for layer in pretrained_model.layers:
    if layer.name == 'conv5_block1_1_conv':
        break
    layer.trainable = False
```

Code untuk mengganti *top* dari 1000 kelas menjadi 2 kelas

```
x = pretrained_model.output
x = tf.keras.layers.Flatten()(x)
x = tf.keras.layers.Dense(512, activation='relu')(x)
x = tf.keras.layers.Dense(128, activation='relu')(x)
predictions = tf.keras.layers.Dense(2,
activation='softmax')(x)
resnet_model =
tf.keras.models.Model(inputs=pretrained_model.input,
outputs=predictions)
```

Uji coba dilakukan dengan *Split Validation* dimana teknik uji coba ini membagi data menjadi bagian secara acak, sebagai data *training* dan sebagian lainnya sebagai data *testing*. Dengan menggunakan *split validation* akan dilakukan percobaan *training* berdasarkan berdasarkan *split ratio* yang telah ditentukan 90:10, 75:25, 60:40, 50:50 dan 40:60 dengan jumlah data sebanyak 7328.

Berikut adalah percobaan dengan *split validation* untuk model ResNet50 dengan data input *image* RGB seperti yang terlihat pada tabel 4.1

Tabel 4.1 *Split Validation* Model ResNet50 (RGB)

Percobaan	Data Training		Data Testing		Model Freeze		Model Unfreeze	
	Ratio	Total	Ratio	Total	Akurasi	Loss	Akurasi	Loss
Percobaan 1	90%	6596	10%	732	87.30%	0.32	93.85%	0.23
Percobaan 2	75%	5496	25%	1832	86.52%	0.34	93.03%	0.52
Percobaan 3	60%	4397	40%	2931	87.07%	0.33	90.41%	0.25
Percobaan 4	50%	3664	50%	3664	86.14%	0.36	93.04%	0.25
Percobaan 5	40%	2932	60%	4396	85.85%	0.39	90.67%	0.28

Pada table 4.1 dari percobaan 1 hingga percobaan 5 akurasi model terbaik yang dilakukan *freeze* ataupun *unfreeze* pada percobaan 1 dengan data *training* 90% dan data *testing* 10% dengan masing-masing akurasi 87.30% untuk model *freeze* dan 93.85% untuk model *unfreeze*.

Adapun *split validation* untuk model ResNet50 dengan data input HSV seperti yang terlihat pada tabel 4.2.

Tabel 4.2 *Split Validation* Model ResNet50 (HSV)

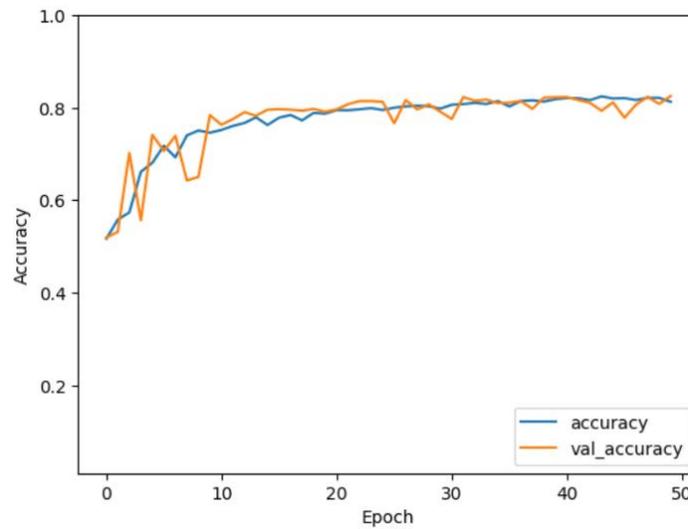
Percobaan	Data Training		Data Testing		Model Freeze		Model Unfreeze	
	Ratio	Total	Ratio	Total	Akurasi	Loss	Akurasi	Loss
Percobaan 1	90%	6596	10%	732	97.27%	0.025	99.18%	0.034
Percobaan 2	75%	5496	25%	1832	97.71%	0.105	99.13%	0.045
Percobaan 3	60%	4397	40%	2931	97.58%	0.095	98.70%	0.069
Percobaan 4	50%	3664	50%	3664	97.16%	0.113	98.83%	0.067
Percobaan 5	40%	2932	60%	4396	97.13%	0.113	98.64%	0.073

Untuk model *freeze* akurasi terbaik pada percobaan 2 dengan akurasi 97.71% dengan *split ratio* 75:25. Sedangkan untuk model *unfreeze* akurasi terbaik pada percobaan 1 dengan akurasi 99.18% dengan *split ratio* 90:10.

4.2.1 Uji Coba RGB – ResNet50 (Model Freeze)

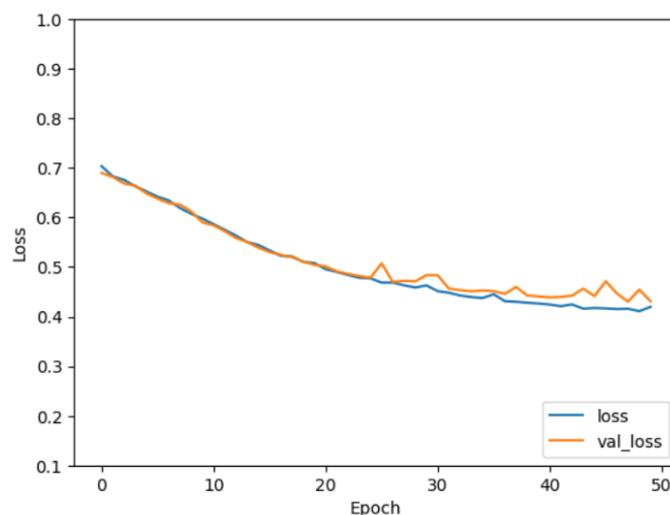
Pada uji coba ini dilakukan dengan image aslinya yaitu RGB tanpa dilakukan *preprocessing* terhadap dataset image wajah. Sebagai contoh data dibagi menjadi 2 bagian yaitu data latih 75% dan data uji 25%. Sedangkan pada model ResNet50 dilakukan *freeze* terhadap bobotnya dimana tidak dilakukan perubahan bobot saat melakukan pelatihan kembali dengan data baru yang akan *ditraining*.

Saat dilakukan *training* 50 *epoch* dengan *learning rate* 0.0001, akurasi untuk data *training* mencapai 92.36%. Sedangkan *validation* akurasi atau akurasi terhadap data uji mencapai 86.52%. Adapun pergerakan akurasi (line berwarna biru) dan *validation* akurasi (line berwarna kuning) dari *epoch* 1 hingga 50 terlihat pada gambar 4.3.



Gambar 4.3 Akurasi dan *Validation* Akurasi RGB - ResNet50 (*Freeze*)

Kemudian selain akurasi dihitung juga *loss function*. Pada kasus ini menggunakan *Categorical Cross Entropy* untuk menyesuaikan bobot model selama *training*. Tujuannya adalah untuk meminimalkan *loss*, yaitu semakin kecil *loss* semakin baik modelnya. Model yang sempurna memiliki *cross-entropy loss* 0. Adapun *loss* data *training* (*line* berwarna biru) mencapai 0.173 dan *validation loss* / *loss* untuk data *testing* (*line* berwarna kuning) mencapai 0.372. Adapun pergerakan *loss* dari *epoch* 1 hingga 50 dapat terlihat pada gambar 4.4

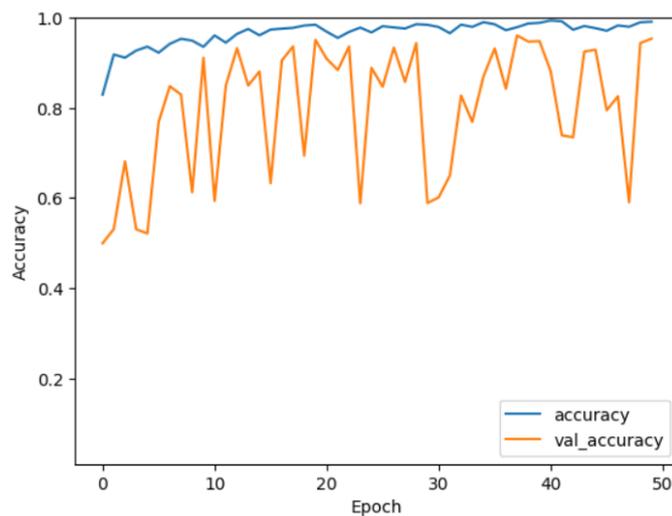


Gambar 4.4 Loss dan *Validation Loss* RGB - ResNet50 (*Freeze*)

4.2.2 Uji Coba RGB – ResNet50 (Model *Unfreeze*)

Pada uji coba kedua dilakukan dengan *image* aslinya yaitu RGB tanpa dilakukan *preprocessing* terhadap dataset image wajah. Sebagai contoh data dibagi menjadi 2 bagian yaitu data latih 75% dan data uji 25%. Sedangkan pada model ResNet50 dilakukan *Unfreeze* sebagian yaitu *Convolutional* yang ke 5 dimana dilakukan perubahan bobot saat melakukan pelatihan kembali dengan data baru yang akan *training*.

Saat dilakukan *training* 50 *epoch* dengan *learning rate* 0.0001, akurasi untuk data *training* mencapai 98.73%. Sedangkan *validation* akurasi atau akurasi terhadap data uji mencapai 93.01%. Adapun pergerakan akurasi (*line* berwarna biru) dan *validation* akurasi (*line* berwarna kuning) dari *epoch* 1 hingga 50 terlihat pada gambar 4.5.

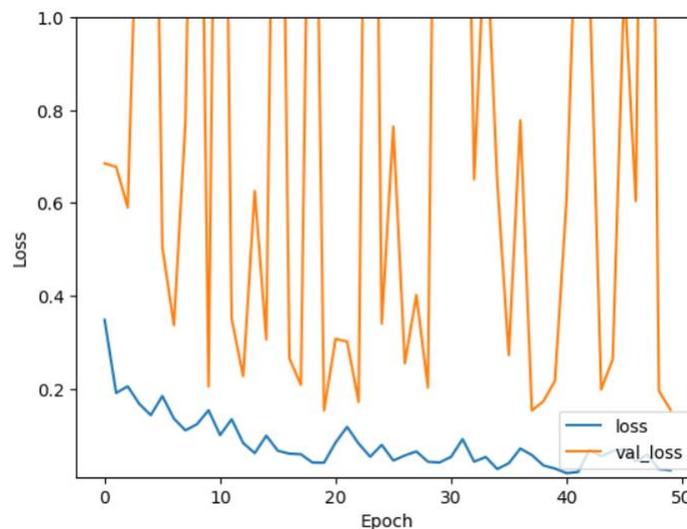


Gambar 4.5 Akurasi dan *Validation* Akurasi RGB - ResNet50 (*Unfreeze*)

Pada gambar 4.5 untuk *epoch* ke 50 *validation* akurasi mencapai 93.01 namun pada *line* berwarna kuning akurasinya sangat tidak stabil, misalnya saja pada *epoch*

ke 48 jauh turun ke bawah hanya 60%. Namun dengan melakukan *unfreeze* ini dapat mencapai akurasi yang tinggi dibandingkan dengan melakukan *freeze*.

Kemudian selain akurasi dihitung juga *loss function*. Adapun *loss* data *training* (*line* berwarna biru) mencapai 0.03 dan *validation loss* / *loss* untuk data *testing* (*line* berwarna kuning) mencapai 0.52. Adapun pergerakan *loss* dari *epoch* 1 hingga 50 dapat terlihat pada gambar 4.6



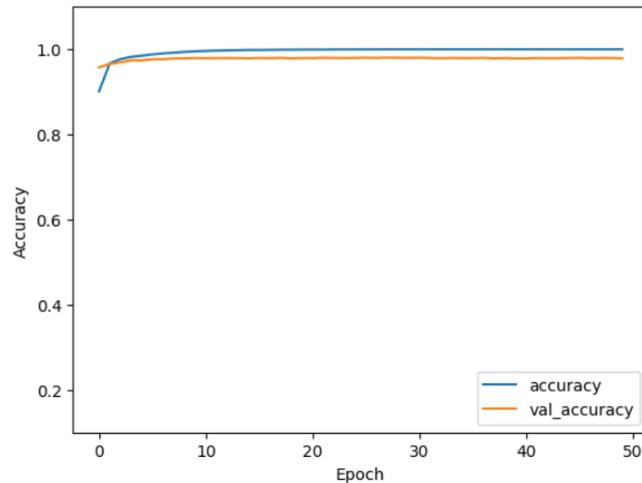
Gambar 4.6 *Loss* dan *Validation Loss* RGB - ResNet50 (*Unfreeze*)

4.2.3 Uji Coba HSV – ResNet50 (Model *Freeze*)

Pada uji coba ketiga dilakukan dengan image RGB yang dilakukan *preprocessing* yaitu dataset image wajah diubah menjadi HSV. Sebagai contoh data dibagi menjadi 2 bagian yaitu data latih 75% dan data uji 25%. Sedangkan pada model ResNet50 dilakukan *freeze* terhadap bobotnya dimana tidak dilakukan perubahan bobot saat melakukan pelatihan kembali dengan data baru yang akan *ditraining*.

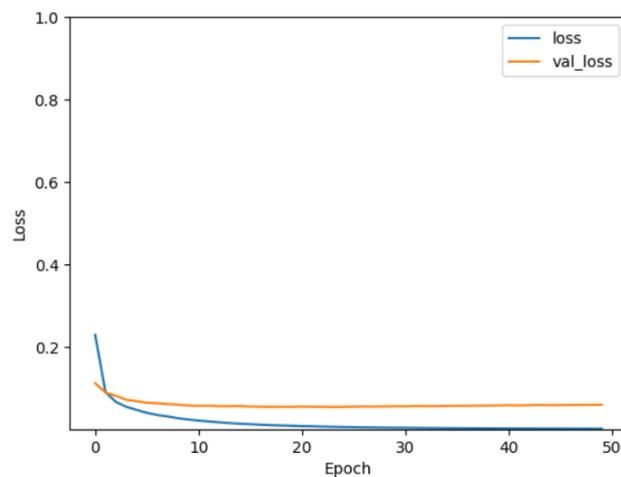
Saat dilakukan *training* 50 *epoch* dengan learning rate 0.0001, akurasi untuk data *training* mencapai 99.98%. Sedangkan *validation* akurasi atau akurasi

terhadap data uji mencapai 97.71%. Adapun pergerakan akurasi (*line* berwarna biru) dan *validation* akurasi (*line* berwarna kuning) dari *epoch* 1 hingga 50 terlihat pada gambar 4.7.



Gambar 4.7 Akurasi dan *Validation* Akurasi HSV - ResNet50 (*Freeze*)

Kemudian selain akurasi dihitung juga *loss function*. Adapun *loss* data *training* (*line* berwarna biru) mencapai 0.00065 dan *validation loss* / *loss* untuk data *testing* (*line* berwarna kuning) mencapai 0.105. Adapun pergerakan *loss* dari *epoch* 1 hingga 50 dapat terlihat pada gambar 4.8

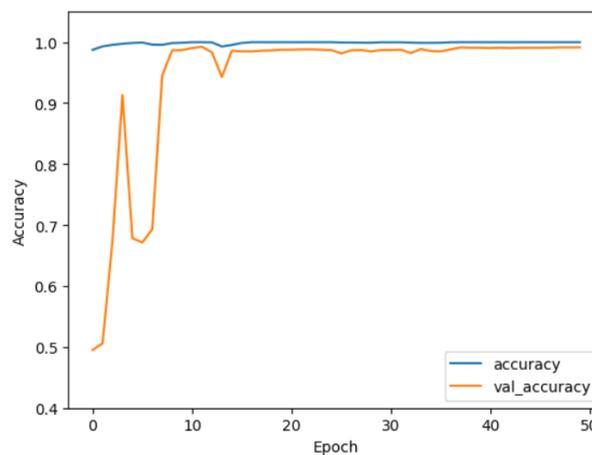


Gambar 4.8 *Loss* dan *Validation Loss* HSV - ResNet50 (*freeze*)

4.2.4 Uji Coba HSV – ResNet50 (Model *Unfreeze*)

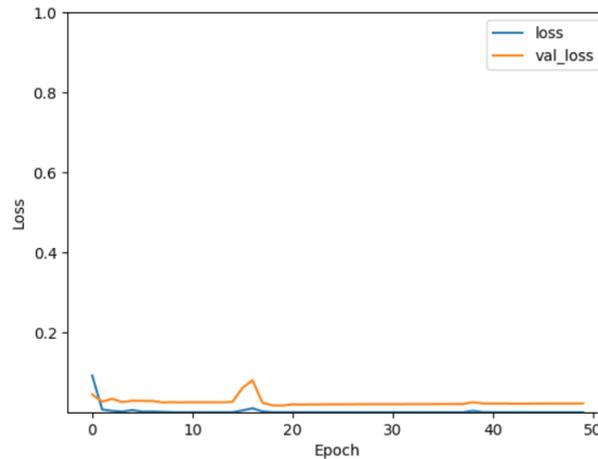
Pada test uji keempat dilakukan dengan *image* RGB yang dilakukan *preprocessing* yaitu dataset *image* wajah diubah menjadi HSV. Sebagai contoh data dibagi menjadi 2 bagian yaitu data latih 75% dan data uji 25%. Sedangkan pada model ResNet50 dilakukan *Unfreeze* sebagian yaitu *Convolutional* yang ke 5 dimana dilakukan perubahan bobot saat melakukan pelatihan kembali dengan data baru yang akan ditraining.

Saat dilakukan *training* 50 *epoch* dengan *learning rate* 0.0001, akurasi untuk data *training* mencapai 99.98%. Sedangkan *validation* akurasi atau akurasi terhadap data uji mencapai 99.13%. Adapun pergerakan akurasi (*line* berwarna biru) dan *validation* akurasi (*line* berwarna kuning) dari *epoch* 1 hingga 50 terlihat pada gambar 4.9.



Gambar 4.9 Akurasi dan *Validation* Akurasi HSV - ResNet50 (*Unfreeze*)

Adapun *loss* data *training* (*line* berwarna biru) mencapai 0.00003 dan *validation loss* / *loss* untuk data *testing* (*line* berwarna kuning) mencapai 0.045. Adapun pergerakan *loss* dari *epoch* 1 hingga 50 dapat terlihat pada gambar 4.10



Gambar 4.10 *Loss dan Validation Loss HSV - ResNet50 (Unfreeze)*

4.3 Komparasi Performa pada Hasil Uji Coba

Dari uji coba yang telah dilakukan dapat dilihat bahwa terjadi peningkatan akurasi saat dilakukan *unfreeze* pada model ResNet50 jika dibanding dengan dilakukannya *freeze*. Selain itu perubahan image RGB ke HSV pada saat proses *preprocessing* memiliki andil perubahan peningkatan akurasi yang sangat signifikan baik dari akurasi data latih atau akurasi pada data uji (*validation* akurasi). Komparasi ini dapat terlihat dari tabel 4.3

Tabel 4.3 Komparasi Performa Akurasi Uji Coba ResNet50

No	Data Image	ResNet50	Akurasi
1	RGB	<i>Freeze</i>	86.52%
2	RGB	<i>Unfreeze</i>	93.85%
3	HSV	<i>Freeze</i>	97.71%
4	HSV	<i>Unfreeze</i>	99.18%

Pada table 4.3 model terbaik dengan akurasi 99.18% adalah model ResNet50 yang dimana *image* input nya dilakukan *preprocessing* RGB ke HSV dan dilakukan *Unfreeze* sebagian pada model *convolutional 5 block* ke 1. *Unfreeze* dilakukan pada *convolutional 5 block 1* disebabkan karena proses pelatihan yang cepat dan dapat menghindari *overfitting* pada model yang dilatih.

Sebagai sebuah contoh dari percobaan 1 dimana input image RGB dengan model *freeze* hasil akurasi mencapai 86.52%. Namun setelah dilakukan perubahan dari RGB ke HSV dengan model *freeze* akurasi meningkat menjadi 97.71%. Untuk contoh kedua dari percobaan 2 dimana input image RGB yang dilakukan *unfreeze* hasil akurasi mencapai 93.85%. Namun setelah dilakukan perubahan dari RGB ke HSV dengan dilakukan *unfreeze* akurasi meningkat menjadi 99.18%. Sehingga dapat disimpulkan bahwa akurasi model ResNet50 dapat meningkat jika dilakukan *preprocessing* pada input *image* dari RGB ke HSV.

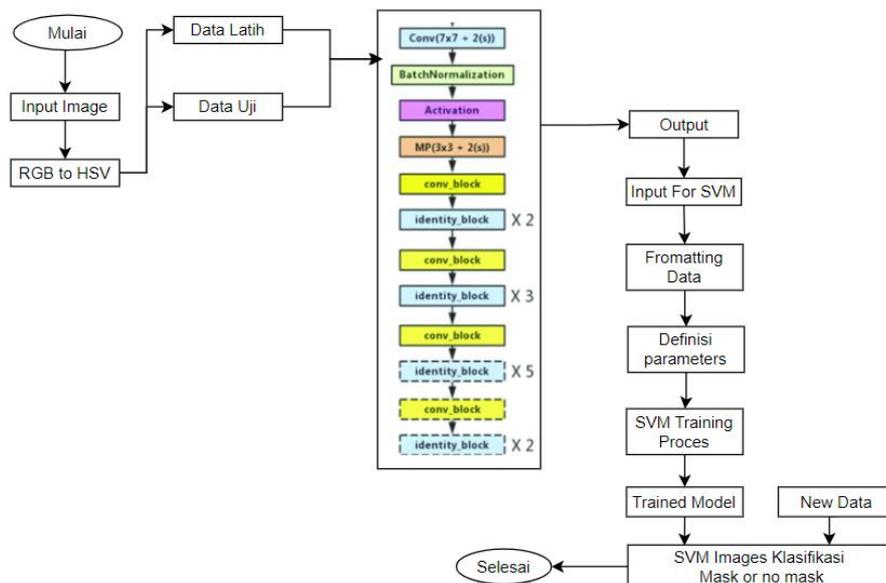
BAB V

HYBRID MODEL RESNET50 DAN SUPPORT VECTOR MACHINE

5.1 Desain

Hybrid model pada penelitian ini adalah penggabungan secara khusus dari dua metode yaitu *transfer learning* dan *machine learning*. Hybrid ini mengacu pada penggabungan elemen-elemen dari kedua metode tersebut. Pendekatan ini dapat membantu mengatasi keterbatasan data *training*. Dalam penelitian ini dilakukan *hybrid* dua metode yaitu pertama sebagai ekstraksi fitur dan metode kedua sebagai klasifikasi. Adapun metode yang digunakan untuk ekstraksi fitur *image* adalah Model ResNet50 yang telah dijelaskan pada bab sebelumnya. Kemudian metode yang digunakan untuk klasifikasi adalah Support Vector Machine (SVM).

Pada proses klasifikasi, lapisan terakhir di ResNet-50 akan dihapus dan diganti dengan SVM. Sehingga *output* dari Model ResNet50 bisa dikatakan sebagai *input* oleh SVM.



Gambar 5.1 Model *Hybrid* ResNet50 & SVM

Pada Gambar 5.1 dijelaskan *hybrid* model mulai dari *input* data *image* wajah, kemudian dilakukan *preprocessing* yaitu mengubah *image* RGB menjadi HSV. Setelah proses ekstraksi fitur dilakukan dengan model ResNet50 yang kemudian ditruskan untuk dilakukan proses pelatihan dengan metode SVM. Adapun *code* untuk melakukan klasifikasi dengan SVM adalah sebagai berikut.

```
feature_extractor = resnet_model.predict(x_train)
features = feature_extractor.reshape(feature_extractor.shape[0], -1)
x_for_SVM = features #This is our X input to SVM
param = [{
    "kernel": ["poly"],
    "C": [1e+3],
    "gamma": [1e-3]
}]
svm = SVC(probability=True, random_state=42)
svm_model = GridSearchCV(svm, param,
    cv=3, n_jobs=-1, verbose=1)
svm_model.fit(x_for_SVM, y_train)
```

5.2 Uji Coba

Pada tahap ini sistem akan melakukan 4 penerapan uji coba untuk model *Hybrid* ResNet50 dan SVM, diantaranya adalah data *image* RGB – Model ResNet50 di *Freeze*, RGB – Model ResNet50 di *Unfreeze*, HSV – Model ResNet50 di *Freeze*, HSV – Model ResNet59 di *Unfreeze*. Sedangkan pada proses pelatihan metode SVM, kernel yang digunakan adalah *polynomial*, $C = 1000$ dan $\gamma = 0.001, 0.01, 0.0001$. Dataset dilakukan dengan *split validation* dengan *split ratio* yang telah ditentukan yaitu 90:10, 75:25, 60:40, 50:50 dan 40:60.

5.2.1 Uji Coba RGB – ResNet50 (Model *Freeze*) & SVM

Pada uji coba ini dilakukan dengan *image* aslinya yaitu RGB tanpa dilakukan *preprocessing* terhadap dataset *image* wajah. Sedangkan pada model ResNet50

dilakukan *freeze* terhadap bobotnya dimana tidak lakukan perubahan bobot saat melakukan pelatihan kembali dengan data baru yang akan dilatih.

Uji coba dilakukan dengan *split validation* dengan *split ratio* yang telah ditentukan yaitu 90:10, 75:25, 60:40, 50:50 dan 40:60 dengan *learning rate* 0.001. Adapun percobaan untuk *hybrid* RGB – ResNet50 (Model *freeze*) dengan SVM seperti terlihat pada tabel 5.1 dengan akurasi terbaik pada percobaan pertama dengan *split ratio* pada data 90:10 dengan akurasi mencapai 90.98%.

Tabel 5.1 Split Validation Model ResNet50 (*Freeze*) - SVM (RGB) – Learning rate 0.001

Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	90.98%	91.03%	90.96%	90.98%
Percobaan 2	75%	5496	25%	1832	90.86%	90.79%	90.99%	90.88%
Percobaan 3	60%	4397	40%	2931	89.66%	89.67%	89.67%	89.66%
Percobaan 4	50%	3664	50%	3664	90.42%	90.46%	90.41%	90.42%
Percobaan 5	40%	2932	60%	4396	87.51%	87.57%	87.52%	87.51%

Uji coba yang kedua dilakukan *split validation* dengan *split ratio* yang sama namun dengan *learning rate* 0.01. Pada percobaan ini akurasi terbaik terjadi pada percobaan 2 yaitu 90.55% seperti yang terlihat pada tabel 5.2.

Tabel 5.2 Split Validation Model ResNet50 (*Freeze*) - SVM (RGB) – Learning rate 0.01

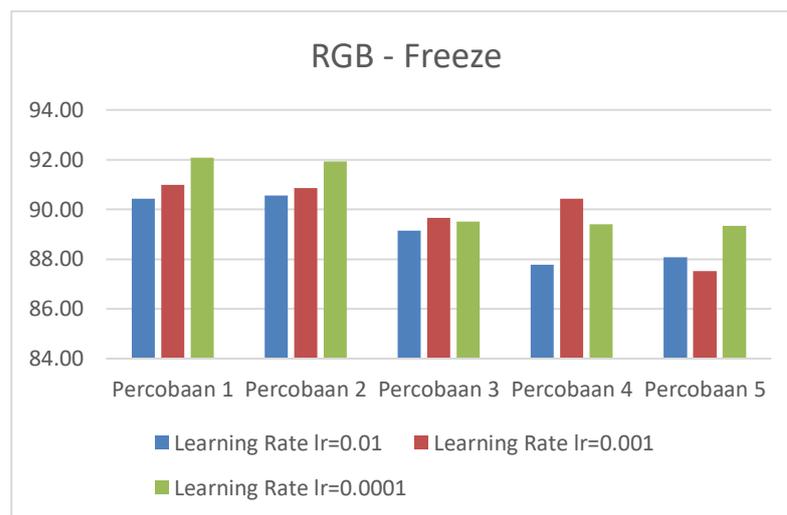
Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	90.43%	90.41%	90.45%	90.43%
Percobaan 2	75%	5496	25%	1832	90.55%	90.55%	90.55%	90.55%
Percobaan 3	60%	4397	40%	2931	89.15%	89.16%	89.15%	89.15%
Percobaan 4	50%	3664	50%	3664	87.77%	87.77%	87.77%	87.77%
Percobaan 5	40%	2932	60%	4396	88.08%	88.10%	89.10%	88.27%

Uji coba yang ketiga dilakukan *split validation* dengan *split ratio* yang sama namun dengan *learning rate* 0.0001. Pada percobaan ini akurasi terbaik terjadi pada percobaan 1 yaitu 92.07% seperti yang terlihat pada tabel 5.3.

Tabel 5.3 Split Validation Model ResNet50 (*Freeze*) - SVM (RGB) – Learning rate 0.0001

Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	92.07%	92.03%	92.06%	92.05%
Percobaan 2	75%	5496	25%	1832	91.92%	91.94%	91.91%	91.92%
Percobaan 3	60%	4397	40%	2931	89.52%	89.53%	89.53%	89.53%
Percobaan 4	50%	3664	50%	3664	89.41%	89.42%	89.42%	89.41%
Percobaan 5	40%	2932	60%	4396	89.33%	89.34%	89.32%	89.33%

Berikut ini disajikan grafik uji coba split validation model ResNet50 (*Freeze*) dan SVM dengan *preprocessing* dari RGB to HSV



Gambar 5.2 Uji Coba *split validation* RGB - ResNet50 – SVM (*Freeze*)

Selanjutnya adalah percobaan dengan menggunakan *k-fold cross validation* dengan $k=5$, dimana dataset dibagi menjadi 5 bagian dan salah satunya sebagai data test yang berbeda dari *fold* 1 hingga *fold* 5, terlihat pada tabel 5.4.

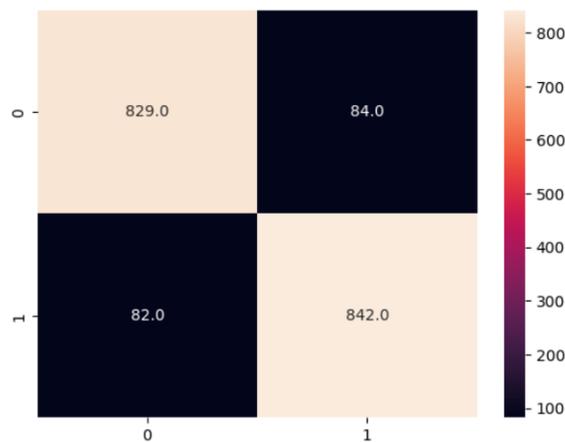
Tabel 5.4 *K-Fold Cross Validation* Model ResNet50 (*Freeze*) - SVM (RGB)

Fold	Data Set					Akurasi	Presisi	Recall	F1 Score
	Test	Train	Train	Train	Train				
Fold-1	Test	Train	Train	Train	Train	89.36%	89.26%	89.09%	89.18%
Fold-2	Train	Test	Train	Train	Train	90.40%	90.22%	90.39%	90.30%
Fold-3	Train	Train	Test	Train	Train	89.17%	90.27%	87.43%	88.83%
Fold-4	Train	Train	Train	Test	Train	87.99%	88.22%	87.25%	87.73%
Fold-5	Train	Train	Train	Train	Test	89/54%	90.19%	88.35%	89.26%

Pada percobaan diatas dapat diambil sebuah nilai rata-rata dari *fold* ke 1 hingga *fold* 5 dengan menjumlah semua akurasi dan dibagi jumlah *fold*. Adapun rata-rata akurasi pada model ini adalah 89.30%. Sedangkan untuk standar deviasi

dapat dihitung dengan $S = \sqrt{\frac{\sum_i^n (x_i - \bar{x})^2}{n-1}} = 0.0098$.

Salah satu contoh untuk mendapatkan hasil uji dilakukan dengan *confusion matrix* dimana *training* model untuk data uji memiliki *confusion matrix* dengan TP=829, FP=84, FN=82, dan TN=842 seperti yang terlihat pada gambar 5.3



Gambar 5.3 *Confusion Matrix* RGB – ResNet50 (Model Freeze) & SVM

Dari *confusion matrix* diatas dapat ditentukan Akurasi, Presisi, *Recall* dan F1 *Score* seperti pada perhitungan di bawah ini

a. Akurasi

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{829 + 842}{829 + 842 + 84 + 82} = 0.9086$$

$$= 90.86\%$$

b. Presisi

$$Precision = \frac{TP}{TP + FP} = \frac{829}{829 + 84} = 0.9079 = 90.79\%$$

c. *Recall*

$$Recall = \frac{TP}{TP + FN} = \frac{829}{829 + 82} = 0.9099 = 90.99\%$$

d. *F1 Score*

$$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * 0.9079 * 0.9099}{0.9079 + 0.9099} \\ = 0.9088 = 90.88\%$$

5.2.2 Uji Coba RGB – ResNet50 (Model *Unfreeze*) & SVM

Pada uji coba ini dilakukan dengan *image* aslinya yaitu RGB tanpa dilakukan *preprocessing* terhadap dataset image wajah. Sedangkan pada model ResNet50 dilakukan *unfreeze* sebagian mulai *convolutional* 5 block 1.

Uji coba dilakukan dengan *split validation* dengan *split ratio* yang telah ditentukan yaitu 90:10, 75:25, 60:40, 50:50 dan 40:60. Adapun percobaan untuk hybrid RGB – ResNet50 (Model *freeze*) dengan SVM seperti terlihat pada tabel 5.5 dengan akurasi terbaik pada percobaan pertama dengan *split ratio* pada data 90:10 dengan akurasi mencapai 91.12%.

Tabel 5.5 *Split Validation* Model ResNet50 (*Unfreeze*) - SVM (RGB) – *learning rate* 0.001

Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	91.12%	91.15%	91.13%	91.12%
Percobaan 2	75%	5496	25%	1832	89.98%	90.95%	89.62%	90.28%
Percobaan 3	60%	4397	40%	2931	90.45%	90.42%	90.46%	90.44%
Percobaan 4	50%	3664	50%	3664	89.08%	89.10%	89.10%	89.08%
Percobaan 5	40%	2932	60%	4396	88.63%	88.63%	88.63%	88.63%

Uji coba yang kedua dilakukan *split validation* dengan *split ratio* yang sama namun dengan *learning rate* 0.01. Pada percobaan ini akurasi terbaik terjadi pada percobaan 1 yaitu 90.84% seperti yang terlihat pada tabel 5.6.

Tabel 5.6 *Split Validation Model ResNet50 (Unfreeze) - SVM (RGB) – learning rate 0.01*

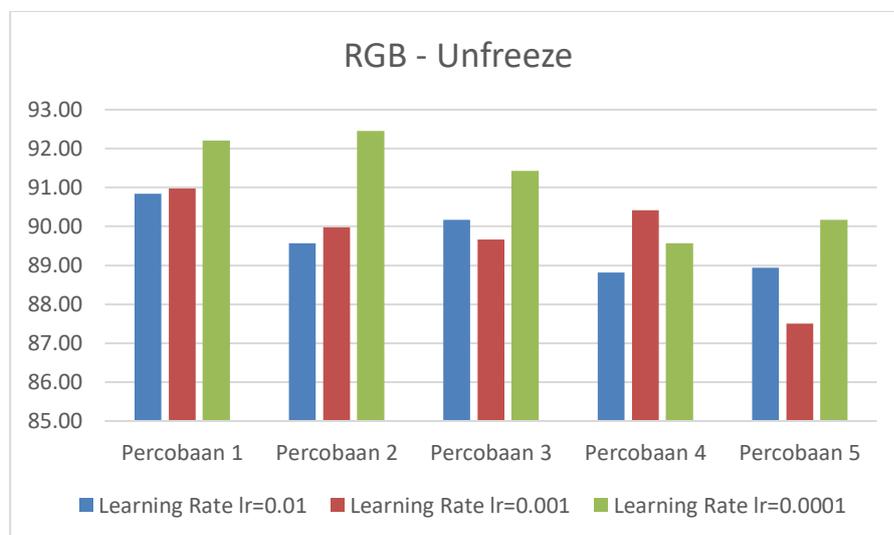
Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	90.84%	90.83%	90.86%	90.84%
Percobaan 2	75%	5496	25%	1832	89.57%	89.57%	89.57%	89.57%
Percobaan 3	60%	4397	40%	2931	90.17%	90.19%	90.15%	90.16%
Percobaan 4	50%	3664	50%	3664	88.81%	88.81%	88.81%	88.81%
Percobaan 5	40%	2932	60%	4396	88.94%	88.94%	88.94%	88.94%

Uji coba yang ketiga dilakukan *split validation* dengan *split ratio* yang sama namun dengan *learning rate* 0.0001. Pada percobaan ini akurasi terbaik terjadi pada percobaan 2 yaitu 92.46% seperti yang terlihat pada tabel 5.7.

Tabel 5.7 *Split Validation Model ResNet50 (Freeze) - SVM (HSV) – learning rate 0.0001*

Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	92.21%	92.21%	92.21%	92.21%
Percobaan 2	75%	5496	25%	1832	92.46%	92.48%	92.47%	92.47%
Percobaan 3	60%	4397	40%	2931	91.43%	91.44%	91.43%	91.43%
Percobaan 4	50%	3664	50%	3664	89.57%	89.57%	89.57%	89.57%
Percobaan 5	40%	2932	60%	4396	90.17%	90.19%	90.15%	90.16%

Berikut ini disajikan grafik uji coba *split validation* model ResNet50 (*Freeze*) dan SVM dengan preprocessing dari RGB to HSV

Gambar 5.4 Uji Coba *split validation* RGB - ResNet50 – SVM (*Unfreeze*)

Selanjutnya adalah percobaan dengan menggunakan *k-fold cross validation* dengan $k=5$, dimana dataset dibagi menjadi 5 bagian dan salah satunya sebagai data test yang berbeda dari *fold 1* hingga *fold 5*, terlihat pada tabel 5.8.

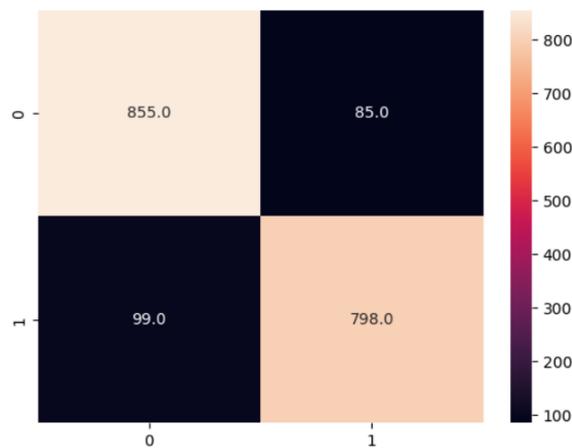
Tabel 5.8 *K-Fold Croos Validation Model ResNet50 (Unfreeze) - SVM (RGB)*

Fold	Data Set					Akurasi	Presisi	Recall	F1 Score
Fold-1	Test	Train	Train	Train	Train	90.73%	91.28%	89.95%	90.61%
Fold-2	Train	Test	Train	Train	Train	89.90%	91.43%	87.91%	89.64%
Fold-3	Train	Train	Test	Train	Train	87.72%	87.71%	87.55%	87.63%
Fold-4	Train	Train	Train	Test	Train	90.45%	91.06%	89.56%	90.30%
Fold-5	Train	Train	Train	Train	Test	88.17%	87.30%	89.21%	88.25%

Pada percobaan diatas dapat diambil sebuah nilai rata-rata dari *fold* ke 1 hingga *fold 5* dengan menjumlah semua akurasi dan dibagi jumlah *fold*. Adapun rata-rata akurasi pada model ini adalah 89.39%. Sedangkan untuk standar deviasi

dapat dihitung dengan $S = \sqrt{\frac{\sum_i^n (x_i - \bar{x})^2}{n-1}} = 0.0136$.

Untuk mendapatkan hasil uji dilakukan dengan *confusion matrix* dimana *training* model untuk data uji memiliki *confusion matrix* dengan TP=855, FP=85, FN=99, dan TN=798 seperti yang terlihat pada gambar 5.5



Gambar 5.5 *Confusion Matrix RGB – ResNet50 (Model Unfreeze) & SVM*

Dari *confusion matrix* diatas dapat ditentukan Akurasi, Presisi, *Recall* dan F1 *Score* seperti pada perhitungan di bawah ini

a. Akurasi

$$\begin{aligned} Accuracy &= \frac{TP + TN}{TP + TN + FP + FN} = \frac{855 + 798}{855 + 798 + 85 + 99} = 0.8998 \\ &= 89.98\% \end{aligned}$$

e. Presisi

$$Precision = \frac{TP}{TP + FP} = \frac{855}{855 + 85} = 0.9095 = 90.95\%$$

f. Recall

$$Recall = \frac{TP}{TP + FN} = \frac{855}{855 + 99} = 0.8962 = 89.62\%$$

g. F1 Score

$$\begin{aligned} F1\ Score &= \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * 0.9095 * 0.8962}{0.9095 + 0.8962} \\ &= 0.9028 = 90.28\% \end{aligned}$$

5.2.3 Uji Coba HSV – ResNet50 (Model Freeze) & SVM

Pada *test* uji ini dilakukan dengan image RGB yang di *preprocessing* menjadi *image* HSV pada dataset *image* wajah. Sedangkan pada model ResNet50 dilakukan *freeze* terhadap bobotnya dimana tidak lakukan perubahan bobot saat melakukan pelatihan kembali dengan data baru yang akan dilatih.

Uji coba dilakukan dengan *split validation* dengan *split ratio* yang telah ditentukan yaitu 90:10, 75:25, 60:40, 50:50 dan 40:60. Adapun percobaan untuk *hybrid* HSV – ResNet50 (Model *freeze*) dengan SVM seperti terlihat pada tabel 5.9 dengan akurasi terbaik pada percobaan pertama dengan *split ratio* pada data 90:10 dengan akurasi mencapai 98.09%.

Tabel 5.9 *Split Validation Model ResNet50 (Freeze) - SVM (HSV) – learning rate 0.001*

Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	98.09%	98.10%	98.09%	98.09%
Percobaan 2	75%	5496	25%	1832	96.18%	96.11%	96.32%	96.21%
Percobaan 3	60%	4397	40%	2931	97.27%	97.26%	97.29%	97.27%
Percobaan 4	50%	3664	50%	3664	96.53%	96.54%	96.54%	96.53%
Percobaan 5	40%	2932	60%	4396	93.53%	93.54%	93.54%	93.54%

Uji coba yang kedua dilakukan *split validation* dengan *split ratio* yang sama namun dengan *learning rate* 0.01. Pada percobaan ini akurasi terbaik terjadi pada percobaan 3 yaitu 96.51% seperti yang terlihat pada tabel 5.10.

Tabel 5.10 *Split Validation Model ResNet50 (Freeze) - SVM (HSV) – learning rate 0.01*

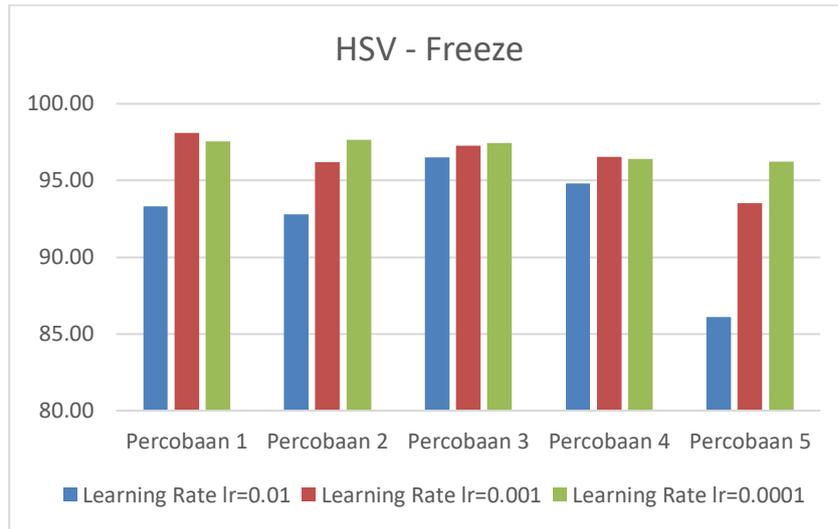
Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	93.30%	93.31%	93.30%	93.31%
Percobaan 2	75%	5496	25%	1832	92.79%	92.84%	92.83%	92.79%
Percobaan 3	60%	4397	40%	2931	96.51%	96.52%	96.52%	96.52%
Percobaan 4	50%	3664	50%	3664	94.81%	94.82%	94.82%	94.81%
Percobaan 5	40%	2932	60%	4396	86.10%	86.15%	86.09%	86.09%

Uji coba yang ketiga dilakukan *split validation* dengan *split ratio* yang sama namun dengan *learning rate* 0.0001. Pada percobaan ini akurasi terbaik terjadi pada percobaan 2 yaitu 97.65% seperti yang terlihat pada tabel 5.11.

Tabel 5.11 *Split Validation Model ResNet50 (Freeze) - SVM (HSV) – learning rate 0.0001*

Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	97.54%	97.54%	97.55%	97.54%
Percobaan 2	75%	5496	25%	1832	97.65%	97.65%	97.65%	97.65%
Percobaan 3	60%	4397	40%	2931	97.44%	97.44%	97.44%	97.44%
Percobaan 4	50%	3664	50%	3664	96.40%	96.42%	96.39%	96.40%
Percobaan 5	40%	2932	60%	4396	96.24%	96.25%	96.24%	96.25%

Berikut ini disajikan grafik uji coba *split validation* model ResNet50 (*Freeze*) dan SVM dengan *preprocessing* dari RGB to HSV



Gambar 5.6 Uji Coba *split validation* HSV - ResNet50 – SVM (*Freeze*)

Selanjutnya adalah percobaan dengan menggunakan *k-fold cross validation* dengan $k=5$, dimana dataset dibagi menjadi 5 bagian dan salah satunya sebagai data test yang berbeda dari *fold 1* hingga *fold 5*, terlihat pada tabel 5.12.

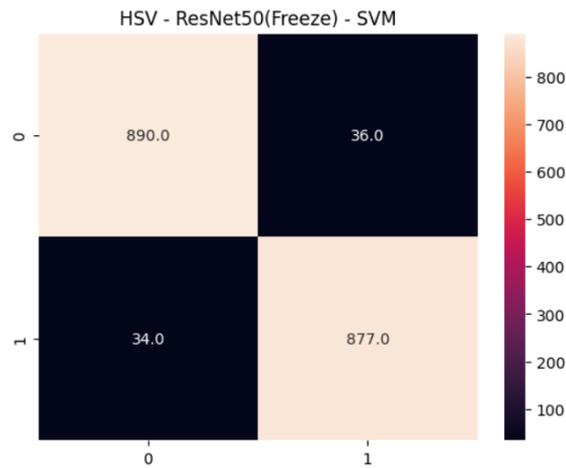
Tabel 5.12 *K-Fold Croos Validation* Model ResNet50 (*Freeze*) - SVM (HSV)

Fold	Data Set					Akurasi	Presisi	Recall	F1 Score
Fold-1	Test	Train	Train	Train	Train	97.82%	97.96%	97.60%	97.78%
Fold-2	Train	Test	Train	Train	Train	97.82%	97.25%	98.33%	97.79%
Fold-3	Train	Train	Test	Train	Train	97.18%	96.87%	97.41%	97.14%
Fold-4	Train	Train	Train	Test	Train	96.82%	96.85%	96.67%	96.76%
Fold-5	Train	Train	Train	Train	Test	97.18%	96.53%	97.78%	97.15%

Pada percobaan diatas dapat diambil sebuah nilai rata-rata dari *fold* ke 1 hingga *fold 5* dengan menjumlah semua akurasi dan dibagi jumlah *fold*. Adapun rata-rata akurasi pada model ini adalah 97.36%. Sedangkan untuk standar deviasi

dapat dihitung dengan $S = \sqrt{\frac{\sum_i^n (x_i - \bar{x})^2}{n-1}} = 0.0041$.

Untuk mendapatkan hasil uji dilakukan dengan *confusion matrix* dimana training model untuk *split ratio* 75:25 memiliki *confusion matrix* dengan TP=890, FP=36, FN=34, dan TN=877 seperti yang terlihat pada gambar 5.7



Gambar 5.7 *Confusion Matrix* HSV – ResNet50 (Model *Freeze*) & SVM

Dari *confusion matrix* diatas dapat ditentukan Akurasi, Presisi, *Recall* dan F1 *Score* seperti pada perhitungan di bawah ini

a. Akurasi

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{890 + 877}{890 + 877 + 36 + 34} = 0.9618$$

$$= 96.18\%$$

b. Presisi

$$Precision = \frac{TP}{TP + FP} = \frac{890}{890 + 36} = 0.9611 = 96.11\%$$

c. *Recall*

$$Recall = \frac{TP}{TP + FN} = \frac{890}{890 + 34} = 0.9632 = 96.32\%$$

d. F1 *Score*

$$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * 0.9611 * 0.9632}{0.9611 + 0.9632}$$

$$= 0.9028 = 96.21\%$$

5.2.4 Uji Coba HSV – ResNet50 (Model *Unfreeze*) & SVM

Pada uji coba ini dilakukan dengan *image* RGB yang di *preprocessing* menjadi *image* HSV pada dataset *image* wajah. Sedangkan pada model ResNet50 dilakukan *unfreeze* sebagian mulai convolutional 5 block 1.

Uji coba dilakukan dengan split validation dengan *split ratio* yang telah ditentukan yaitu 90:10, 75:25, 60:40, 50:50 dan 40:60. Adapun percobaan untuk *hybrid* HSV – ResNet50 (Model *freeze*) dengan SVM seperti terlihat pada tabel 5.13 dengan akurasi terbaik pada percobaan pertama dengan *split ratio* pada data 90:10 dengan akurasi mencapai 98.36%.

Tabel 5.13 *Split Validation* Model ResNet50 (*Unfreeze*) - SVM (HSV) – *learning rate* 0.001

Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	98.36%	98.37%	98.38%	98.36%
Percobaan 2	75%	5496	25%	1832	96.18%	96.11%	96.32%	96.21%
Percobaan 3	60%	4397	40%	2931	97.27%	97.26%	97.29%	97.27%
Percobaan 4	50%	3664	50%	3664	97.29%	97.29%	97.31%	97.30%
Percobaan 5	40%	2932	60%	4396	96.06%	96.10%	96.06%	96.06%

Uji coba yang kedua dilakukan *split validation* dengan *split ratio* yang sama namun dengan *learning rate* 0.01. Pada percobaan ini akurasi terbaik terjadi pada percobaan 1 yaitu 97.81% seperti yang terlihat pada tabel 5.14.

Tabel 5.14 *Split Validation* Model ResNet50 (*Unfreeze*) - SVM (HSV) – *learning rate* 0.01

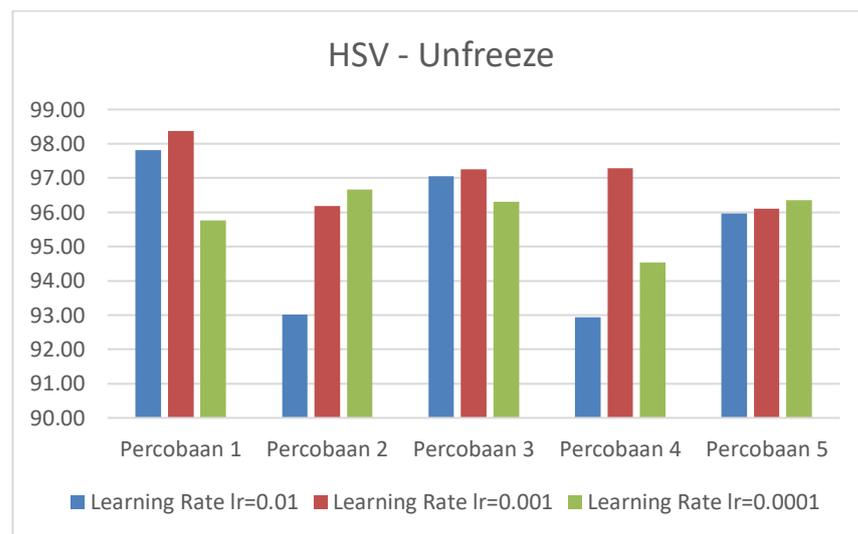
Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	97.81%	97.65%	98.16%	97.91%
Percobaan 2	75%	5496	25%	1832	93.01%	93.03%	93.01%	93.03%
Percobaan 3	60%	4397	40%	2931	97.06%	97.07%	97.06%	97.07%
Percobaan 4	50%	3664	50%	3664	92.93%	92.94%	92.97%	92.93%
Percobaan 5	40%	2932	60%	4396	95.97%	95.98%	95.98%	95.97%

Uji coba yang kedua dilakukan *split validation* dengan *split ratio* yang sama namun dengan *learning rate* 0.0001. Pada percobaan ini akurasi terbaik terjadi pada percobaan 2 yaitu 96.67% seperti yang terlihat pada tabel 5.15.

Tabel 5.15 *Split Validation Model ResNet50 (Unfreeze) - SVM (HSV) – learning rate 0.0001*

Percobaan	Data Training		Data Testing		Confusion Matrix			
	Ratio	Total	Ratio	Total	Akurasi	Presisi	Recall	F1 Score
Percobaan 1	90%	6596	10%	732	95.76%	95.77%	95.76%	95.76%
Percobaan 2	75%	5496	25%	1832	96.67%	96.67%	96.67%	96.67%
Percobaan 3	60%	4397	40%	2931	96.31%	96.31%	96.32%	96.31%
Percobaan 4	50%	3664	50%	3664	94.54%	94.54%	94.54%	94.54%
Percobaan 5	40%	2932	60%	4396	96.36%	96.39%	96.38%	96.36%

Berikut ini disajikan grafik uji coba *split validation* model ResNet50 (*Unfreeze*) dan SVM dengan *preprocessing* dari RGB to HSV

Gambar 5.8 Uji Coba *split validation* HSV - ResNet50 – SVM (*Unfreeze*)

Selanjutnya adalah percobaan dengan menggunakan *k-fold cross validation* dengan $k=5$, dimana dataset dibagi menjadi 5 bagian dan salah satunya sebagai data test yang berbeda dari *fold 1* hingga *fold 5*, terlihat pada tabel 5.16.

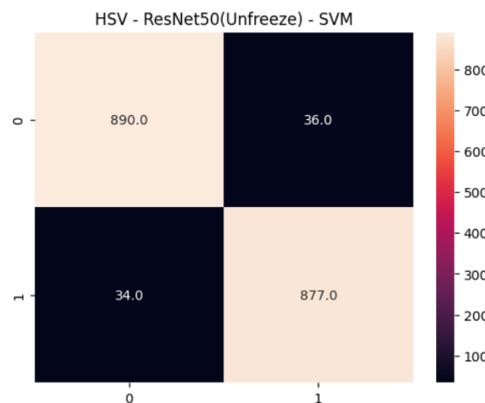
Tabel 5.16 *K-Fold Croos Validation Model ResNet50 (Unfreeze) - SVM (HSV)*

Fold	Data Set					Akurasi	Presisi	Recall	F1 Score
	Test	Train	Train	Train	Train				
Fold-1	Test	Train	Train	Train	Train	97.64%	97.13%	98.19%	97.66%
Fold-2	Train	Test	Train	Train	Train	97.54%	96.95%	98.19%	97.57%
Fold-3	Train	Train	Test	Train	Train	96.72%	95.90%	97.64%	96.76%
Fold-4	Train	Train	Train	Test	Train	96.91%	96.09%	97.83%	96.95%
Fold-5	Train	Train	Train	Train	Test	97.36%	97.63%	97.10%	97.37%

Pada percobaan diatas dapat diambil sebuah nilai rata-rata dari *fold* ke 1 hingga *fold* 5 dengan menjumlah semua akurasi dan dibagi jumlah *fold*. Adapun rata-rata akurasi pada model ini adalah 97.23%. Sedangkan untuk standar deviasi

dapat dihitung dengan $S = \sqrt{\frac{\sum_i^n (x_i - \bar{x})^2}{n-1}} = 0.00401$.

Untuk mendapatkan hasil uji dilakukan dengan *confusion matrix* dimana *training* model untuk data uji memiliki *confusion matrix* dengan TP=890, FP=36, FN=34, dan TN=877 seperti yang terlihat pada gambar 5.9



Gambar 5.9 *Confusion Matrix* HSV – ResNet50 (Model *Unfreeze*) & SVM

Dari *confusion matrix* diatas dapat ditentukan Akurasi, Presisi, *Recall* dan F1 *Score* seperti pada perhitungan di bawah ini

a. Akurasi

$$\begin{aligned} Accuracy &= \frac{TP + TN}{TP + TN + FP + FN} = \frac{890 + 877}{890 + 877 + 36 + 34} = 0.9618 \\ &= 96.18\% \end{aligned}$$

b. Presisi

$$Precision = \frac{TP}{TP + FP} = \frac{890}{890 + 36} = 0.9611 = 96.11\%$$

c. *Recall*

$$Recall = \frac{TP}{TP + FN} = \frac{890}{890 + 34} = 0.9632 = 96.32\%$$

d. F1 Score

$$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * 0.9611 * 0.9632}{0.9611 + 0.9632}$$

$$= 0.9028 = 96.21\%$$

5.2 Komparasi Performa pada Hasil Uji Coba

Dari uji coba yang telah dilakukan dapat dilihat bahwa terjadi sedikit peningkatan akurasi bahkan cenderung sama saat dilakukan *freeze* ataupun *unfreeze* pada model ResNet50 & SVM. Namun perubahan *image* RGB ke HSV pada saat proses *preprocessing* memiliki perubahan peningkatan akurasi yang signifikan pada akurasi data uji. Sedangkan uji coba dengan *learning rate* 0.001 memiliki akurasi yang lebih tinggi jika dibandingkan dengan *learning rate* 0.01 dan 0.0001. Pada tabel 5.13 ini disajikan data yang memiliki akurasi tertinggi dari masing-masing uji coba dengan *validation split* diatas sehingga dari tabel ini dapat dilakukan analisa dan mencari model terbaik yang akan digunakan menjadi model untuk deteksi masker pada image wajah.

Tabel 5.17 Komparasi Performa Akurasi Uji Coba ResNet50 & SVM

Uji	Data Image	ResNet50	Gamma	Akurasi	Presisi	Recall	F1 Score
1.1	RGB	<i>Freeze</i>	0.001	90.98%	91.03%	90.96%	90.98%
1.2	RGB	<i>Freeze</i>	0.01	90.55%	90.55%	90.55%	90.55%
1.3	RGB	<i>Freeze</i>	0.0001	92.07%	92.03%	92.06%	92.05%
2.1	RGB	<i>Unfreeze</i>	0.001	91.12%	91.15%	91.13%	91.12%
2.2	RGB	<i>Unfreeze</i>	0.01	90.84%	90.83%	90.86%	90.84%
2.3	RGB	<i>Unfreeze</i>	0.0001	92.46%	92.48%	92.47%	92.47%
3.1	HSV	<i>Freeze</i>	0.001	96.18%	96.11%	96.32%	96.21%
3.2	HSV	<i>Freeze</i>	0.01	96.51%	96.52%	96.52%	96.52%
3.3	HSV	<i>Freeze</i>	0.0001	97.65%	97.65%	97.65%	97.65%
4.1	HSV	<i>Unfreeze</i>	0.001	98.36%	98.37%	98.38%	98.36%
4.2	HSV	<i>Unfreeze</i>	0.01	97.81%	97.65%	98.16%	97.91%
4.2	HSV	<i>Unfreeze</i>	0.0001	96.67%	96.67%	96.67%	96.67%

Pada table 5.17 dapat dianalisa bahwa model terbaik dengan akurasi 98.36% adalah model ResNet50 & SVM yang dilakukan *unfreeze* dimana *image* inputnya dilakukan *preprocessing* RGB ke HSV pada *learning rate* 0.001.

Sebagai sebuah contoh dari percobaan 1.2 dimana input *imgae* RGB dengan model *freeze* dan *learning rate* 0.01 hasil akurasi mencapai 90.55%. Namun setelah dilakukan perubahan dari RGB ke HSV dengan model *freeze* dan *learning rate* 0.01 akurasi meningkat menjadi 96.51%. Untuk contoh kedua dari percobaan 2.1 dimana input *image* RGB yang dilakukan *unfreeze* dengan *learning rate* 0.001 hasil akurasi mencapai 91.12%. Namun setelah dilakukan perubahan dari RGB ke HSV dengan dilakukan *unfreeze* dan *learning rate* 0.001 akurasi meningkat menjadi 98.36%. Untuk contoh ketiga dari percobaan 1.3 dimana input *image* RGB yang dilakukan *freeze* dengan *learning rate* 0.0001 hasil akurasi mencapai 92.07%. Namun setelah dilakukan perubahan dari RGB ke HSV dengan dilakukan *freeze* dan *learning rate* 0.0001 akurasi meningkat menjadi 97.65%.

Sehingga dapat disimpulkan dari 5 percobaan *split validation* dan percobaan pada *learning rate* 0.01, 0.001 dan 0.0001, bahwa akurasi suatu model ResNet50 dan SVM dapat meningkat jika dilakukan *preprocessing* pada input *image* dari RGB ke HSV.

BAB VI

PEMBAHASAN

6.1 Komparasi dan Performa Metode

Strategi pengujian dilakukan dengan *split validation* yang dilakukan percobaan *training* berdasarkan *split ratio* yang telah ditentukan 90:10, 75:25, 60:40, 50:50 dan 40:60 dengan jumlah data sebanyak 7328. Pengujian ini dilakukan dengan melibatkan 4 pendekatan model *transfer learning* ResNet50 dan *Hybrid* ResNet50 dengan SVM.

- a. Pendekatan pertama yaitu *image* input tidak dilakukan *preprocessing* sehingga *image* inputan adalah *image* dengan *colorspace* RGB dan pada model ResNet50 dilakukan *freeze*.
- b. Pendekatan kedua yaitu *image input* tidak dilakukan *preprocessing* sehingga *image* inputan adalah *image* dengan *colorspace* RGB dan pada model ResNet50 dilakukan *unfreeze*.
- c. Pendekatan ketiga yaitu *image* inputan dilakukan *preprocessing* mengubah *image* RGB ke HSV dan model ResNet50 dilakukan *freeze*.
- d. Pendekatan keempat yaitu *image* inputan dilakukan *preprocessing* mengubah *image* RGB ke HSV dan model ResNet50 dilakukan *unfreeze*.

Untuk melakukan uji coba ini diperlukan data. Adapun contoh atau sampel data yang tidak memakai masker disajikan pada gambar 6.1



Gambar 6.1 Contoh image wajah tidak menggunakan masker
(Sumber: diolah dari <https://kaggle.com>)

Sedangkan contoh atau sampel data yang memakai masker dengan motif dan jenis masker yang bermacam-macam disajikan pada gambar 6.2



Gambar 6.2 Contoh *image* wajah menggunakan masker
(Sumber: diolah dari <https://kaggle.com>)

6.1.1 Performa ResNet50

Dari empat pendekatan yang telah dilakukan dapat dilihat bahwa terjadi peningkatan akurasi saat dilakukan *unfreeze* pada model ResNet50 jika dibanding

dengan dilakukannya *freeze*. Selain itu perubahan image RGB ke HSV pada saat proses *preprocessing* membuat peningkatan akurasi yang sangat signifikan baik dari akurasi data latih atau akurasi pada data uji (*validation* akurasi).

Hasil performa pada pendekatan pertama yaitu image input tidak dilakukan *preprocessing* sehingga *image* inputan adalah *image* dengan *colorspace* RGB dan pada model ResNet50 dilakukan *freeze* menghasilkan akurasi sebesar 86.52% dengan *loss* sebesar 0.41. Pendekatan kedua yaitu *image* input tidak dilakukan *preprocessing* sehingga *image* inputan adalah *image* dengan *colorspace* RGB dan pada model ResNet50 dilakukan *unfreeze* mendapat akurasi sebesar 93.85%.

Sedangkan untuk pendekatan ketiga yaitu *image* inputan dilakukan *preprocessing* mengubah *image* RGB ke HSV dan model ResNet50 dilakukan *freeze* menghasilkan akurasi yang meningkat dari *image* RGB yang mencapai 97.71%. Pendekatan yang keempat yaitu *image* inputan dilakukan *preprocessing* mengubah *image* RGB ke HSV dan model ResNet50 dilakukan *unfreeze* menghasilkan akurasi yang tinggi yaitu mencapai 99.18%.

Sehingga pada experiment hasil uji coba untuk model *transfer learning* ResNet50 mendapat akurasi terbaik sebesar **99.18%**. Model ini mengubah inputan *image colorspace* dari RGB menjadi HSV pada *preprocessing*. Selain itu juga model ini melakukan *unfreeze* sebagian pada model ResNet50 pada *convolutional 5 block 1*.

6.1.2 Performa Hybrid ResNet50 dan SVM

Dari uji coba yang telah dilakukan dengan empat pendekatan dapat dilihat bahwa sedikit terjadi perubahan akurasi bahkan cenderung memiliki akurasi yang sama saat dilakukan *freeze* ataupun *unfreeze* pada model ResNet50 & SVM. Namun

perubahan *image* RGB ke HSV pada saat proses *preprocessing* memiliki perubahan peningkatan akurasi yang signifikan pada akurasi data uji. Begitupun dengan uji coba yang dilakukan dengan *learning rate* yang berbeda dari 0.01, 0.001 dan 0.0001, dimana akurasi tertinggi terjadi pada uji coba dengan *learning rate* 0.001 model yang dilakukan *preprocessing* dengan mengubah RGB ke HSV dan model yang *unfreeze*.

Hasil performa pada pendekatan pertama yaitu *image* input tidak dilakukan *preprocessing* sehingga *image* inputan adalah *image* dengan *colorsapce* RGB dan pada model ResNet50 dilakukan *freeze* dengan *learning rate* 0.001 menghasilkan akurasi sebesar 90.98% dengan presisi = 91.03% , *recall* = 90.96% dan F1 = 90.98%. Pendekatan kedua yaitu *image* input tidak dilakukan *preprocessing* sehingga *image* inputan adalah *image* dengan *colorsapce* RGB dan pada model ResNet50 dilakukan *unfreeze* dengan *learning rate* 0.001 mendapat akurasi sebesar 91.12% dengan presisi = 91.15% , *recall* = 91.13% dan F1 = 91.12%..

Adapun pendekatan ketiga yaitu *image* inputan dilakukan *preprocessing* mengubah *image* RGB ke HSV dan model ResNet50 dilakukan *freeze* dengan *learning rate* 0.001 menghasil akurasi yang meningkat dari *image* RGB yang mencapai 96.18 % dengan presisi = 96.11% , *recall* = 96.32% dan F1 = 96.21%. Pendekatan yang keempat yaitu *image* inputan dilakukan *preprocessing* mengubah *image* RGB ke HSV dan model ResNet50 dilakukan *unfreeze* dengan *learning rate* 0.001 menghasil akurasi yang sama dengan pendekatan ke 3 yaitu mencapai akurasi 98.36% dengan presisi = 98.37% , *recall* = 98.38% dan F1 = 98.36%.

Sehingga dapat disimpulkan dari percobaan *split validation* dan percobaan pada *learning rate* 0.01, 0.001 dan 0.0001, bahwa akurasi suatu model ResNet50 dan

SVM dapat meningkat jika dilakukan *preprocessing* pada input *image* dari RGB ke HSV.

6.1.3 Komparasi Kedua Model

Terdapat 8 hasil *eksperiment* dengan 2 model atau metode antara ResNet50 dan *Hybrid* ResNet50 dengan SVM dalam 4 pendekatan yang telah disebutkan di atas. Paparan akurasi terlihat pada table 6.1

Tabel 6.1 Hasil experiment kedua model

No	Data Image	Medel/Metode	Freeze/Unfreeze	Akurasi
1	RGB	ResNet50	<i>Freeze</i>	85.26%
2	RGB	ResNet50	<i>Unfreeze</i>	93.03%
3	HSV	ResNet50	<i>Freeze</i>	97.71%
4	HSV	ResNet50	<i>Unfreeze</i>	99.18%
5	RGB	Hybrid ResNet50 & SVM	<i>Freeze</i>	90.98%
6	RGB	Hybrid ResNet50 & SVM	<i>Unfreeze</i>	91.12%
7	HSV	Hybrid ResNet50 & SVM	<i>Freeze</i>	96.18%
8	HSV	Hybrid ResNet50 & SVM	<i>Unfreeze</i>	98.38%

Dari table 6.1 untuk akurasi terbaik didapatkan 99.18% untuk ResNet50 yang dimana modelnya dilakukan *unfreeze* pada *convolutional* ke 5 *block* 1. Dan untuk input data dilakukan *preprocessing* yaitu mengubah RGB ke HSV. Adapun metode *Hybrid* model *transfer learning* ResNet50 dengan SVM memiliki akurasi 96.38%. Model *hybrid* ini juga untuk hasil terbaiknya perlu dilakukan *preprocessing* dari RGB ke HSV. Sedang untuk *freeze* dan *unfreeze* pada ResNet50-SVM tidak berpengaruh terhadap peningkatan akurasi.

Model *Hybrid* ResNet50 & SVM yang merupakan model yang diusulkan pada penelitian ini memiliki akurasi yang kurang baik jika dibandingkan dengan model ResNet50 tanpa *hybrid*. Hal ini dikarenakan faktor dari *preprocessing* yaitu pengubahan *image* dari RGB ke HSV. Melihat dari tabel 6.1 dimana ResNet50

memiliki akurasi 85.26% jika *image* masih RGB atau belum diproses ke HSV. Sedangkan untuk *hybrid* ResNet50 & SVM tanpa adanya *preprocessing* input *image* RGB mendapatkan akurasi 90.86%.

6.1.4 Komparasi Dengan Penelitian Sebelumnya

Metode yang telah penulis lakukan eksprement yaitu RGB to HSV – ResNet50 (*unfreeze*) dan RGB to HSV – ResNet50 & SVM, akan dilakukan komparasi dengan penelitian sebelumnya. Dimana hasil komparasi terlihat pada tabel 6.2.

Tabel 6.2 Hasil Komparasi dengan penelitian sebelumnya

No	Nama	Kesamaan Data	Metode	Akurasi
1	(Loey <i>et al.</i> , 2021)	Berbeda	ResNet50 - SVM	99.49%
2	(Venkateswarlu, et al, 2020)	Sama	MobileNet – GPB	99%
3	(Suresh et al, 2021)	Berbeda	MobileNet	97.75%
4	(Oumina et al, 2020)	Sama	MobileNetV2 - SVM	97.11%
5	(Negi <i>et al.</i> , 2020)	Berbeda	Standart CNN	96.90%
6	(Oumina et al, 2020)	Sama	VGG19 - KNN	96.65%
7	(Oumina et al, 2020)	Sama	MobileNetV2 - KNN	94.92%
8	(Oumina et al, 2020)	Sama	Exception - SVM	94.57%
9	(Yadav, 2020)	Berbeda	MobileNetV2 - SSD	91.70%
10	(Sanjaya and Rakhmawan, 2020)	Berbeda	MobileNetV2	82.96%
11	Penelitian Model 1	Sama +	RGB to HSV – ResNet50	99.18%
12	Penelitian Model 2	Sama +	RGB to HSV - ResNet50 - SVM	98.36%

Dari tabel 6.2 RGB to HSV – ResNet50 (*unfreeze*) menduduki peringkat akurasi pada urutan kedua. Namun saat kita lakukan perbandingan dengan penelitian sebelumnya yang menggunakan data yang sama dengan peneliti dan dilakukan penambahan data maka RGB to HSV – ResNet50 memiliki akurasi yang paling tinggi dengan akurasi 99.18% mengungguli MobileNet – GPB dengan akurasi 99%.

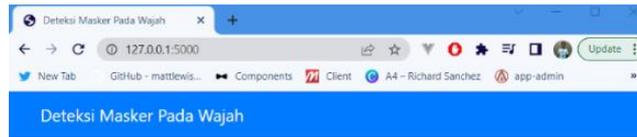
Sedangkan untuk RGB to HSV – *Hybrid ResNet50 & SVM* pada urutan ke 4. Dan urutan yang ke 3 jika menggunakan data yang sama dengan peneliti yaitu dengan akurasi 98.36%.

Dari hasil experiment ini maka dapat kita nyatakan bahwa penambahan data untuk pelatihan mempengaruhi hasil akurasi. *Preprocessing* data *image* dari RGB ke HSV sebelum dilakukan pelatihan berpengaruh sangat signifikan dalam peningkatan akurasi. Selain itu juga melakukan *unfreeze* pada model ResNet50 juga memiliki pengaruh pada model RGB to HSV – ResNet50 ataupun Model Hybrid ResNet50 – SVM.

6.2 Implementasi Hasil

Hasil komparasi dan analisa dari beberapa metode dan telah ditentukan metode dengan performa terbaik yaitu model RGB to HSV – ResNet50. Maka selanjutnya dilakukan implementasi menggunakan *Flask Python* untuk dapat melakukan deteksi menggunakan masker atau tidak secara *real time* atau menggunakan video *frame* berbasis *web*.

Adapun implementasi hasil yang dilakukan seperti yang terlihat pada gambar 6.3 terdeteksi tidak memakai masker.

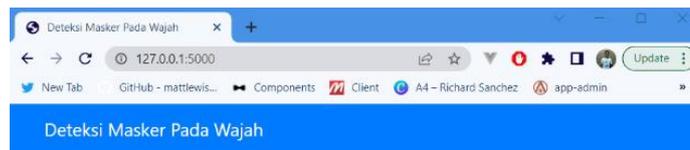


Deteksi Masker Pada Image Wajah



Gambar 6.3 Video *frame* terdeteksi tidak memakai masker

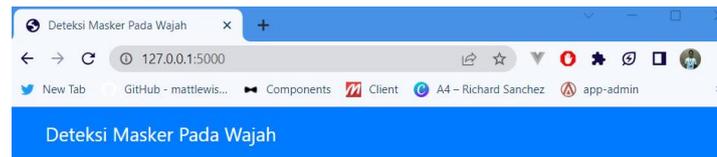
Sedangkan untuk implementasi hasil yang menggunakan masker seperti yang terlihat pada gambar 6.4



Deteksi Masker Pada Image Wajah



Gambar 6.4 video *frame* terdeteksi memakai masker



Deteksi Masker Pada Image Wajah



Gambar 6.5 video *frame* terdeteksi tidak memakai masker

Adapun saat melakukan pendeteksian masker dengan menutup mulut dan hidung seolah-olah menggunakan masker, akan tetapi tetap terdeteksi tidak menggunakan masker. Hal ini dikarenakan bentuk dan warna masker yang telah ditraining dan mampu dikenali. Selain itu hal ini terdeteksi karena warna dari kulit tangan sama dengan kulit wajah sehingga mampu dikenali bahwa itu bukan sebuah masker.

Dari tiga hasil diatas dapat disimpulkan bahwa metode yang diterapkan dapat digunakan untuk melakukan pendeteksian masker baik ditempat tertutup ataupun tempat-tempat umum.

6.3 Integrasi Islam

Segala sesuatu yang terjadi di dunia, termasuk pandemi Covid-19 adalah atas kehendak Allah, dalam Al-Qur'an surah At-Taghabun: 11

مَا أَصَابَ مِنْ مُصِيبَةٍ إِلَّا بِإِذْنِ اللَّهِ وَمَنْ يُؤْمِنُ بِاللَّهِ يَهْدِ اللَّهُ قَلْبَهُ وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ

“Tidak ada sesuatu musibah pun yang menimpa seseorang kecuali dengan izin Allah; Dan barang siapa yang beriman kepada Allah, niscaya Dia akan memberi petunjuk kepada hatinya. Dan Allah Maha Mengetahui segala sesuatu.”

Sebagai umat islam yang beriman kepada *Qadha* dan *Kadar* Allah, harus selalu sabar dan tawakal kepada Allah sehingga kita dapat menerima dan meyakini bahwa adanya Covid-19 saat ini adalah bentuk musibah dan ujian dari Allah. Meski sudah takdir yang diberikan Allah, manusia tetap berikhtiar untuk mengupayakan tidak tertular virus Covid19

Dalam pandangan Islam menjaga kesehatan dan mencegah penyebaran penyakit termasuk COVID-19 sangat dianjurkan. Islam mengajarkan pentingnya menjaga kesehatan dan kebersihan diri serta melindungi kehidupan manusia. Oleh karena itu penggunaan masker sebagai langkah pencegahan penyebaran penyakit adalah sesuatu didukung.

Meskipun islam tidak secara khusus membahas penggunaan masker dalam konteks Covid-19, namun prinsip-prinsip islam yang lebih umum dapat diterapkan. Prinsip-prinsip tersebut mencakup menjaga kehidupan diri sendiri dan orang lain, serta menghindari tindakan yang dapat membahayakan kesehatan. Seperti yang dijelaskan dalam surat Al-Baqarah ayat 195

وَأَنْفِقُوا فِي سَبِيلِ اللَّهِ وَلَا تُلْقُوا بِأَيْدِيكُمْ إِلَى التَّهْلُكَةِ . وَأَحْسِنُوا . إِنَّ اللَّهَ يُحِبُّ الْمُحْسِنِينَ

“Dan infakkanlah (hartamu) di jalan Allah, dan janganlah kamu jatuhkan (diri sendiri) ke dalam kebinasaan dengan tangan sendiri, dan berbuat baiklah. Sungguh, Allah menyukai orang-orang yang berbuat baik” (Q.S Al-Baqarah: 195).

Ayat ini menunjukkan pentingnya menjaga diri dari bahaya dan melindungi diri sendiri. Dalam konteks Covid-19, menggunakan masker dapat menjadi salah satu langkah untuk melindungi diri dari penularan wabah penyakit. Sehingga pada penelitian ini dibuat suatu sistem yang dapat mendeteksi masker pada wajah yang bertujuan untuk mendeteksi orang-orang yang memakai masker dalam mematuhi protokol kesehatan guna untuk menjaga kesehatan diri sendiri dan kesehatan bersama.

BAB VII KESIMPULAN

8.1 Kesimpulan

Dari hasil *test* uji dan pembahasan metode untuk melakukan deteksi masker pada image wajah dapat diambil kesimpulan sebagai berikut:

- a. Model terbaik dengan tingkat akurasi 99.18% adalah Model *Transfer Learning* ResNet50 yang dilakukan *unfreeze* dan dilakukan *preprocessing* pada *image* yaitu mengubah RGB ke HSV.
- b. Bahwa *preprocessing* pada data *image* yaitu mengubah *colorspace* dari RGB ke HSV dan dilakukan *unfreeze* pada model *transfer learning* ResNet50 ataupun *Hybrid* ResNet50 dengan SVM dapat meningkatkan akurasi dari model yang dilatih.

8.2 Saran

Dalam Penelitian ini tentunya dapat dijadikan acuan untuk pengembangan pada penelitian berikutnya, oleh karena itu terdapat beberapa saran dari penulis.

- a. Penelitian dapat dilakukan untuk melakukan pengenalan wajah seseorang walaupun menggunakan masker.
- b. Metode *hybrid* dengan transfer learning dapat dilakukan dengan banyak kombinasi untuk mendapatkan hasil yang lebih baik.

Daftar Pustaka

- Boldog, P. *et al.* (2020) 'Risk assessment of novel coronavirus COVID-19 outbreaks outside China', *Journal of Clinical Medicine*, 9(2). Available at: <https://doi.org/10.3390/jcm9020571>.
- Cortes, C., Vapnik, V. and Saitta, L. (1995) '*Support-Vector Networks Editor, Machine Learning*'. Kluwer Academic Publishers.
- Dewi, N. and Ismawan, F. (2021) 'Implementasi Deep Learning Menggunakan CNN Untuk Sistem Pengenalan Wajah', *Faktor Exacta*, 14(1), p. 34. Available at: <https://doi.org/10.30998/faktorexacta.v14i1.8989>.
- Faiz Nashrullah, Suryo Adhi Wibowo and Gelar Budiman (2020) 'The Investigation of Epoch Parameters in ResNet-50 Architecture for Pornographic Classification', *Journal of Computer, Electronic, and Telecommunication*, 1(1). Available at: <https://doi.org/10.52435/complete.v1i1.51>.
- He, K. *et al.* (2015) 'Deep Residual Learning for Image Recognition', *IEEE Conference on Computer Vision and Pattern Recognition (pp. 770-778)*. *IEEE* [Preprint]. Available at: <http://arxiv.org/abs/1512.03385>.
- Huda, N. *et al.* (2022) 'Klasifikasi Malaria Menggunakan Metode Image Processing Dari Sel Darah Merah Dengan Algoritma Convolutional Neural Network', *JOINS (Journal of Information System)*, 7(2), pp. 166–177. Available at: <https://doi.org/10.33633/joins.v7i2.7068>.
- Kosasih, R. (2020) 'Kombinasi Metode Isomap dan KNN pada Image Processing Untuk Pengenalan Wajah.
- Loey, M. *et al.* (2021) 'A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic', *Measurement: Journal of the International Measurement Confederation*, 167. Available at: <https://doi.org/10.1016/j.measurement.2020.108288>.
- Moreira, G. *et al.* (2022) 'Benchmark of Deep Learning and a Proposed HSV Colour Space Models for the Detection and Classification of Greenhouse Tomato', *Agronomy*, 12(2). Available at: <https://doi.org/10.3390/agronomy12020356>.

- Negi, A. *et al.* (2020) 'Face Mask Detection Classifier and Model Pruning with Keras-Surgeon', in *2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering, ICRAIE 2020 - Proceeding*. Institute of Electrical and Electronics Engineers Inc. Available at: <https://doi.org/10.1109/ICRAIE51050.2020.9358337>.
- Nguyen, T.H., Nguyen, T.N. and Ngo, B.V. (2022) 'A VGG-19 Model with Transfer Learning and Image Segmentation for Classification of Tomato Leaf Disease', *AgriEngineering*, 4(4), pp. 871–887. Available at: <https://doi.org/10.3390/agriengineering4040056>.
- Nurkholis (2020) 'Dampak Pandemi Novel-Corona Virus Disiase (Covid-19) Terhadap Psikologi Dan Pendidikan Serta Kebijakan Pemerintah', *Jurnal PGSD*, Volume 6 (1) 2020.
- Oumina, A., El Makhfi, N. and Hamdi, M. (2020) 'Control the COVID-19 Pandemic: Face Mask Detection Using Transfer Learning', in *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science, ICECOCS 2020*. Institute of Electrical and Electronics Engineers Inc. Available at: <https://doi.org/10.1109/ICECOCS50124.2020.9314511>.
- Prajna Bhandary (2020) 'Mask Classifier', <https://github.com/prajnasb/observations>.
- Russakovsky, O. *et al.* (2015) *Detecting avocados to zucchinis: what have we done, and where are we going?* Available at: www.image-net.org/challenges/LSVRC/2012/analysis/.
- Sanjaya, S.A. and Rakhmawan, S.A. (2020) 'Face Mask Detection Using MobileNetV2 in the Era of COVID-19 Pandemic', in *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy, ICDABI 2020*. Institute of Electrical and Electronics Engineers Inc. Available at: <https://doi.org/10.1109/ICDABI51230.2020.9325631>.
- Sethi, S., Kathuria, M. and Kaushik, T. (2021) 'Face mask detection using deep learning: An approach to reduce risk of Coronavirus spread', *Journal of Biomedical Informatics*, 120. Available at: <https://doi.org/10.1016/j.jbi.2021.103848>.

- Suresh, K., Palangappa, M.B. and Bhuvan, S. (2021) 'Face Mask Detection by using Optimistic Convolutional Neural Network', in *Proceedings of the 6th International Conference on Inventive Computation Technologies, ICICT 2021*. Institute of Electrical and Electronics Engineers Inc., pp. 1084–1089. Available at: <https://doi.org/10.1109/ICICT50816.2021.9358653>.
- Venkateswarlu, I.B., Kakarla, J. and Prakash, S. (2020) 'Face mask detection using MobileNet and global pooling block', in *4th IEEE Conference on Information and Communication Technology, CICT 2020*. Institute of Electrical and Electronics Engineers Inc. Available at: <https://doi.org/10.1109/CICT51604.2020.9312083>.
- Vijay Kumar (2021) 'Face Mask Detection', <https://www.kaggle.com/datasets/vijaykumar1799/face-mask-detection>.
- World Health Organization (2020) *Coronavirus disease 2019 (COVID-19): situation report*.
- Yadav, S. (2020) 'Deep Learning based Safe Social Distancing and Face Mask Detection in Public Areas for COVID-19 Safety Guidelines Adherence', *International Journal for Research in Applied Science and Engineering Technology*, 8(7), pp. 1368–1375. Available at: <https://doi.org/10.22214/ijraset.2020.30560>.