

TESIS

**Analisis Pengaruh Augmentasi Data pada Pengenalan Ekspresi Wajah
Menggunakan Metode YOLOv3**



Disusun oleh:

Nama : Biva Candra Luffi Adlatma
NIM : 19.51.1279
Konsentrasi : Business Intelligence

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA**

2022

TESIS

**Analisis Pengaruh Augmentasi Data pada Pengenalan Ekspresi Wajah
Menggunakan Metode YOLOv3**

**Analysis of the Effect of Data Augmentation on Facial Expression
Recognition using the YOLOv3 Method**

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister



Disusun oleh:

Nama : Biva Candra Lutfl Adiatma
NIM : 19.51.1279
Konsentrasi : Business Intelligence

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA**

2022

HALAMAN PENGESAHAN

**Analisis Pengaruh Augmentasi Data pada Pengenalan Ekspresi Wajah
Menggunakan Metode YOLOv3**

**Analysis of the Effect of Data Augmentation on Facial Expression Recognition
using the YOLOv3 Method**

Dipersiapkan dan Disusun oleh

Biva Candra Lutfi Adiatma

19.51.1279

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Senin, 3 Januari 2022

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 3 Januari 2022

Rektor

Prof. Dr. M. Suyanto, M.M.

NIK. 190302001

HALAMAN PERSETUJUAN

**Analisis Pengaruh Augmentasi Data pada Pengenalan Ekspresi Wajah
Menggunakan Metode YOLOv3**

**Analysis of the Effect of Data Augmentation on Facial Expression Recognition
using the YOLOv3 Method**

Dipersiapkan dan Disusun oleh

Biva Candra Lutfi Adiatma

19.51.1279

Telah Ditujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Senin, 3 Januari 2022

Pembimbing Utama

Anggota Tim Penguji

Prof. Dr. Ema Utami, S. Si., M. Kom

NIK. 190302037

Dr. Andi Sunvoto, M. Kom.

NIK. 190302052

Pembimbing Pendamping

Alva Hendi Muhammad, S.T., M.Eng., Ph.D.

NIK. 190302493

Anggit Dwi Hartanto, M. Kom

NIK. 190302163

Prof. Dr. Ema Utami, S. Si., M. Kom

NIK. 190302037

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer.

Yogyakarta, 3 Januari 2022

Direktur Program Pascasarjana

Prof. Dr. Kusriani, M.Kom.

NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Biva Candra Lutfi Adiatma
NIM : 19.51.1279
Konsentrasi : Business Intelligence

Menyatakan bahwa Tesis dengan judul berikut:
**Analisis Pengaruh Augmentasi Data pada Pengenalan Ekspresi Wajah
Menggunakan Metode YOLOv3**

Dosen Pembimbing Utama : Prof. Dr. Ema Utami, S. Si., M. Kom
Dosen Pembimbing Pendamping : Anggit Dwi Hartanto, M. Kom

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, 3 Januari 2022

Yang Menyatakan,



METERAI
TEMPEL
1000
SABALIX67409666

Biva Candra Lutfi Adiatma

HALAMAN PERSEMBAHAN

Pertama dan paling utama, saya ucapkan puji syukur kepada Allah SWT yang telah memberikan kemudahan dan kelancaran dalam proses pembuatan tesis ini. Tesis ini saya persembahkan untuk:

1. Orang tua saya khususnya Ibu Sri Lestari yang senantiasa memberikan semangat, motivasi, dukungan moral maupun *financial*, dan doa, semoga selalu dalam lindungan-Nya.
2. Ibu Prof. Dr. Ema Utami, S. Si., M. Kom dan Bapak Anggit Dwi Hartanto, M. Kom yang telah memberikan bimbingan aktif selama pelaksanaan penelitian, semoga mendapatkan banyak keberkahan dan dilancarkan segala urusannya.
3. Kakak Lia Andiani Pamularsih yang selalu mendukung dan menyemangati adiknya, semoga selalu diberi kemudahan serta kelancaran.
4. Keluarga besar kelas MTI20-A yang selalu membantu dan memberi saran demi kelancaran pengerjaan tesis, semoga selalu semangat dan sukses.
5. Mas Umam Maulana yang telah bersedia untuk meminjamkan laptop untuk mempermudah dalam penelitian.

HALAMAN MOTTO

“Allah tidak membebani seseorang melainkan sesuai kesanggupannya”

QS Al Baqarah 286

“Kamu tidak akan tahu kesalahan apa yang membuatmu gagal sebelum kamu memulai, *so just do it*”



KATA PENGANTAR

Alhamdulillah, segala puji dan syukur penulis panjatkan sehadirat Allah SWT karena atas segala karunia dan ridho-Nya, sehingga tesis yang berjudul “Pengenalannya Ekspresi Wajah Berbasis Emosi Menggunakan Metode *YOLO v3*” dapat diselesaikan dengan tepat waktu.

Tesis ini disusun untuk memenuhi salah satu syarat memperoleh gelar Magister Komputer pada program studi Magister Teknik Informatika Universitas Amikom Yogyakarta.

Penyelesaian tesis yang sangat berharga ini tidak lepas dari bantuan dan dukungan dari berbagai pihak. Pada kesempatan ini, penulis mengucapkan rasa syukur dan terima kasih kepada:

1. Orang tua Ibu Sri Lestari dan Kakak Lia Andiani Pamularsih yang senantiasa memberikan do'a, semangat, motivasi, dukungan moral maupun *financial* kepada penulis agar senantiasa semangat dalam menuntut ilmu dan menyelesaikan tanggung jawab.
2. Ibu Prof. Dr. Ema Utami, S. Si., M. Kom selaku pembimbing utama yang telah membimbing, membantu, dan memotivasi dalam penulisan tesis ini sehingga dapat terselesaikan dengan baik.
3. Bapak Anggit Dwi Hartanto, M. Kom selaku pembimbing pendamping yang telah membimbing, membantu, dan memotivasi dalam penulisan tesis ini sehingga dapat terselesaikan dengan baik.

4. Dosen Penguji yang telah memberikan saran yang baik demi kemajuan tesis ini.
5. Direktur Program Pascasarjana, jajarannya, staf dan rekan-rekan. Magister Teknik Informatika Universitas Amikom Yogyakarta.

Yogyakarta, 28 Januari 2021

Penulis



DAFTAR ISI

HALAMAN JUDUL.....	ii
HALAMAN PENGESAHAN.....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PERNYATAAN KEASLIAN TESIS.....	v
HALAMAN PERSEMBAHAN.....	vi
HALAMAN MOTTO.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR TABEL.....	xv
DAFTAR GAMBAR.....	xvi
INTISARI.....	xx
<i>ABSTRACT</i>	xxi
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Masalah.....	1
1.2. Rumusan Masalah.....	6
1.3. Batasan Masalah.....	6
1.4. Tujuan Penelitian.....	7
1.5. Manfaat Penelitian.....	8
BAB II TINJAUAN PUSTAKA.....	9
2.1. Tinjauan Pustaka.....	9
2.2. Keaslian Penelitian.....	12

2.3. Landasan Teori.....	29
2.3.1. Ekspresi Wajah.....	29
2.3.2. <i>MixUp</i>	30
2.3.3. Citra Digital.....	31
2.3.4. Computer Vision.....	32
2.3.5. <i>Convolutional Neural Network</i>	33
2.3.6. Lapisan <i>Pooling</i>	39
2.3.7. <i>You Only Look Once (YOLO)</i>	42
BAB III METODE PENELITIAN.....	45
3.1. Jenis, Sifat, dan Pendekatan Penelitian.....	45
3.1.1. Jenis Penelitian.....	45
3.1.2. Sifat Penelitian.....	45
3.1.3. Pendekatan Penelitian.....	45
3.2. Metode Pengumpulan Data.....	46
3.3. <i>Japanese Female Facial Expression (JAFFE)</i>	46
3.4. <i>Affectnet</i>	47
3.5. <i>Real-world Affective Faces (RAF) Databases</i>	48
3.6. Metode Analisis Data.....	49
3.7. Alur Penelitian.....	49
BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....	57
4.1. Pengumpulan Data.....	57
4.1.1. <i>Japanese Female Facial Expression (JAFFE)</i>	57

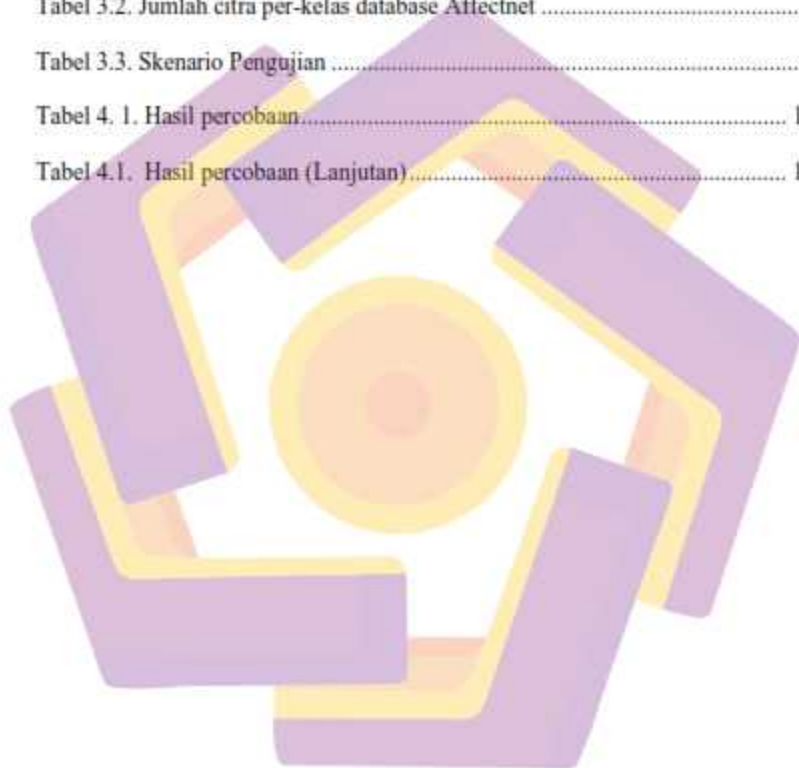
4.1.2. Affectnet	57
4.1.3. Real-world Affective Faces Database	58
4.2. Augmentasi Data	59
4.3. MixUp	60
4.4. Anotasi Citra	61
4.5. Membagi data <i>train</i> dan data <i>test</i>	63
4.6. Arsitektur YOLOv3	64
4.7. Persiapan skenario	66
4.7.1. Skenario dataset JAFFE	68
4.7.1.1. Skenario JAFFE dengan rasio 80/20	68
4.7.1.2. Skenario JAFFE dengan rasio 70/30	68
4.7.1.3. Skenario JAFFE + Augmentasi dengan rasio 80/20	68
4.7.1.4. Skenario JAFFE + Augmentasi dengan rasio 70/30	69
4.7.1.5. Skenario JAFFE + MixUp dengan rasio 80/20	69
4.7.1.6. Skenario JAFFE + MixUp dengan rasio 70/30	70
4.7.1.7. Skenario JAFFE + Augmentasi + MixUp dengan rasio 80/20	70
4.7.2. Skenario dataset Affectnet	71
4.7.2.1. Skenario Affectnet dengan rasio 80/20	71
4.7.2.2. Skenario Affectnet dengan rasio 70/30	71
4.7.2.3. Skenario Affectnet + Augmentasi dengan rasio 80/20	72
4.7.2.4. Skenario Affectnet + Augmentasi dengan rasio 70/30	73
4.7.2.5. Skenario Affectnet + MixUp dengan rasio 80/20	73
4.7.2.6. Skenario Affectnet + MixUp dengan rasio 70/30	73
4.7.2.7. Skenario Affectnet + Augmentasi + MixUp dengan rasio 80/20	74
4.7.3. Skenario dataset RAF-DB	74
4.7.3.1. Skenario RAF-DB dengan rasio 80/20	74

4.7.3.2.	Skenario RAF-DB dengan rasio 70/30	75
4.7.3.3.	Skenario RAF-DB + Augmentasi dengan rasio 80/20	75
4.7.3.4.	Skenario RAF-DB + Augmentasi dengan rasio 70/30	76
4.7.3.5.	Skenario RAF-DB + MixUp dengan rasio 80/20	76
4.7.3.6.	Skenario RAF-DB + MixUp dengan rasio 70/30	77
4.7.3.7.	Skenario RAF-DB + Augmentasi + MixUp dengan rasio 80/20	77
4.8.	Proses Pelatihan Data	78
4.9.	Proses Deteksi	78
4.9.1.	Resize Citra	78
4.9.2.	Konvolusi dan ReLU(Reactified Linier Unit)	79
4.9.3.	Max Pooling	81
4.9.4.	Output 13*13, 26*26, 32*32	81
4.9.5.	Bounding Box Confidence dan Prediksi Kelas Objek	82
4.9.6.	Non-Max Suppression	82
4.10.	Hasil Klasifikasi	84
4.11.	Perhitungan Kinerja Model	84
4.12.	Hasil <i>Training</i> dan <i>Testing</i> data	85
4.12.1.	JAFFE	86
4.12.1.1.	Skenario 1	86
4.12.1.2.	Skenario 2	87
4.12.1.3.	Skenario 3	89
4.12.1.4.	Skenario 4	90
4.12.1.5.	Skenario 5	92
4.12.1.6.	Skenario 6	93
4.12.1.7.	Skenario 7	95

4.12.2. Affectnet	96
4.12.2.1. Skenario 1	96
4.12.2.2. Skenario 2	98
4.12.2.3. Skenario 3	99
4.12.2.4. Skenario 4	101
4.12.2.5. Skenario 5	102
4.12.2.6. Skenario 6	104
4.12.2.7. Skenario 7	105
4.12.3. RAF-DB	107
4.12.3.1. Skenario 1	107
4.12.3.2. Skenario 2	108
4.12.3.3. Skenario 3	110
4.12.3.4. Skenario 4	111
4.12.3.5. Skenario 5	113
4.12.3.6. Skenario 6	114
4.12.3.7. Skenario 7	116
4.13. Analisis Hasil Penelitian	117
4.14. Perbandingan Hasil <i>Training</i>	120
4.15. Perbandingan Waktu Deteksi	121
4.16. Perbandingan Hasil Testing	123
4.17. Pengaruh augmentasi data	126
BAB V PENUTUP	128
5.1. Kesimpulan	128
5.2. Saran	129
DAFTAR PUSTAKA	130

DAFTAR TABEL

Tabel 2.1. Matriks literatur review dan posisi penelitian.....	12
Tabel 3.1. Jumlah citra perkelas dataset <i>JAFFE</i>	46
Tabel 3.2. Jumlah citra per-kelas database Affectnet	47
Tabel 3.3. Skenario Pengujian	55
Tabel 4. 1. Hasil percobaan.....	118
Tabel 4.1. Hasil percobaan (Lanjutan).....	119



DAFTAR GAMBAR

Gambar 2.1. Ekpresi Wajah Manusia	30
Gambar 2.2. Arsitektur Convolutional Neural Network	34
Gambar 2.3. <i>Convolutional Layer Filter</i>	35
Gambar 2.4. Convolutional Layer Pertama.....	36
Gambar 2.5. <i>Convolutional Filter</i> dengan <i>Stride</i>	37
Gambar 2.6. <i>Convolutional Zero Padding</i>	37
Gambar 2.7. Fungsi Aktivasi ReLU.....	38
Gambar 2.8. <i>Maxpool Layer</i>	39
Gambar 2.9. Perubahan matriks 2 dimensi menjadi vektor	40
Gambar 2.10. Arsitektur ResNet50.....	42
Gambar 2.11. Arsitektur YOLO	43
Gambar 2.12. Ilustrasi Metode <i>YOLO</i>	44
Gambar 3.1. Contoh data pada database <i>JAFFE</i>	47
Gambar 3.2. Alur Penelitian.....	50
Gambar 4.1. Citra pada dataset <i>JAFFE</i>	57
Gambar 4.2. Citra pada dataset <i>Affectnet</i>	58
Gambar 4.3. Citra pada dataset <i>Affectnet</i>	58
Gambar 4.4. Contoh proses augmentasi data.	59
Gambar 4.5. Contoh gambar dengan <i>MixUp</i>	61
Gambar 4.6. Proses Anotasi Citra	62
Gambar 4.7. Anotasi citra pada ekspresi wajah marah	62

Gambar 4.8. Citra yang telah dianotasi	63
Gambar 4.9. File <i>train.txt</i> dan <i>test.txt</i> yang telah terbentuk	64
Gambar 4.10. Arsitektur <i>YOLOv3</i>	65
Gambar 4.11. Hasil augmentasi dataset <i>JAFFE</i> skenario 3	69
Gambar 4.12. Mixup pada dataset <i>JAFFE</i> skenario 5	70
Gambar 4.13. Hasil augmentasi dataset <i>Affectnet</i> skenario 3	72
Gambar 4.14. Contoh <i>mixup</i> pada dataset <i>Affectnet</i> skenario 5.....	73
Gambar 4.15. Hasil augmentasi dataset <i>RAF-DB</i> skenario 3	76
Gambar 4.16. Contoh <i>mixup</i> pada dataset <i>RAF-DB</i> skenario 5	77
Gambar 4.17. <i>Resize</i> Citra.....	79
Gambar 4.18. Proses Konvolusi.....	80
Gambar 4.19. Proses aktivasi <i>ReLU</i>	80
Gambar 4.20. Proses <i>Max Pooling</i>	81
Gambar 4.22. <i>Intersection Over Union (IoU)</i>	83
Gambar 4.23. Contoh hasil <i>Confusion Matrix</i>	84
Gambar 4. 24. Grafik training dan validation loss skenario 1	86
Gambar 4. 25. <i>Confusion Matrix</i> skenario 1	87
Gambar 4. 26. Grafik training dan validation loss skenario 1	88
Gambar 4. 27. <i>Confusion Matrix</i> skenario 2	88
Gambar 4. 28. Grafik training dan validation loss skenario 3	89
Gambar 4. 29. <i>Confusion Matrix</i> skenario 3	90
Gambar 4. 30. Grafik training dan validation loss skenario 4	91
Gambar 4. 31. <i>Confusion Matrix</i> skenario 4	91

Gambar 4. 32. Grafik training dan validation loss skenario 5	92
Gambar 4. 33. <i>Confusion Matrix</i> skenario 5	93
Gambar 4. 34. Grafik training dan validation loss skenario 6	94
Gambar 4. 35. <i>Confusion Matrix</i> skenario 6	94
Gambar 4. 36. Grafik training dan validation loss skenario 7	95
Gambar 4. 37. <i>Confusion Matrix</i> skenario 7	96
Gambar 4. 38. Grafik training dan validation loss skenario 1	97
Gambar 4. 39. <i>Confusion Matrix</i> skenario 1	97
Gambar 4. 40. Grafik <i>training</i> dan <i>validation loss</i> skenario 2	98
Gambar 4. 41. <i>Confusion Matrix</i> skenario 2	99
Gambar 4. 42. Grafik training dan validation loss skenario 3	100
Gambar 4. 43. <i>Confusion Matrix</i> skenario 3	100
Gambar 4. 44. Grafik training dan validation loss skenario 4	101
Gambar 4. 45. <i>Confusion Matrix</i> skenario 4	102
Gambar 4. 46. Grafik training dan validation loss skenario 5	103
Gambar 4. 47. <i>Confusion Matrix</i> skenario 5	103
Gambar 4. 48. Grafik training dan validation loss skenario 6	104
Gambar 4. 49. <i>Confusion Matrix</i> skenario 6	105
Gambar 4. 50. Grafik training dan validation loss skenario 7	106
Gambar 4. 51. <i>Confusion Matrix</i> skenario 7	106
Gambar 4. 52. Grafik training dan validation loss skenario 1	107
Gambar 4. 53. <i>Confusion Matrix</i> skenario 1	108
Gambar 4. 54. Grafik training dan validation loss skenario 2	109

Gambar 4. 55. <i>Confusion Matrix</i> skenario 2	109
Gambar 4. 56. Grafik training dan validation loss skenario 3	110
Gambar 4. 57. <i>Confusion Matrix</i> skenario 3	111
Gambar 4. 58. Grafik training dan validation loss skenario 4	112
Gambar 4. 59. <i>Confusion Matrix</i> skenario 4	112
Gambar 4. 60. Grafik training dan validation loss skenario 5	113
Gambar 4. 61. <i>Confusion Matrix</i> skenario 5	114
Gambar 4. 62. Grafik training dan validation loss skenario 6	115
Gambar 4. 63. <i>Confusion Matrix</i> skenario 6	115
Gambar 4. 64. Grafik training dan validation loss skenario 7	116
Gambar 4. 65. <i>Confusion Matrix</i> skenario 7	117
Gambar 4. 66. Hasil Training	120
Gambar 4. 67. Contoh <i>log testing</i>	121
Gambar 4. 69. Perbandingan grafik hasil testing	124
Gambar 4. 70. Perbandingan dataset <i>JAFPE</i>	125

INTISARI

Ekspresi wajah adalah salah satu cara manusia untuk berkomunikasi serta berinteraksi. Ekpresi yang muncul secara tidak sengaja adalah perasaan yang sebenarnya dirasakan. Kesalahan dalam menilai perasaan orang lain dikarenakan hanya melihat secara sekilas. Perlu adanya inovasi sistem untuk dapat melakukan pengawasan terhadap ekspresi seseorang. *Machine learning* telah diterapkan untuk pembuatan pengenalan ekspresi wajah walaupun mengalami kesuksesan dibidang tertentu masih terdapat kelemahan untuk mendapatkan hasil yang optimal ketika dihadapkan dengan dataset yang besar. Solusi untuk mengatasi hal ini adalah dengan melakukan pendekatan dengan metode *deep learning*. Permasalahan yang sering terjadi pada *deep learning* adalah kurangnya dataset yang digunakan. Salah satu solusi untuk memecahkan kekurangan ini adalah dengan melakukan augmentasi data. Ada banyak metode augmentasi data akan tetapi pada penelitian ini menggunakan pendekatan *Crop, Shear, Rotate, Flip* dan *MixUp*. Dalam penelitian ini akan dilakukan perbandingan kinerja dan waktu deteksi untuk mencari pengaruh augmentasi data terhadap 3 dataset yang berbeda yaitu *Affectnet, RAF-DB*, dan *JAFFE* menggunakan metode *YOLOv3*. Untuk mendapatkan hasil yang terbaik maka setiap dataset akan dilakukan pengujian dengan 7 scenario yang berbeda. Dari hasil yang didapatkan penelitian ini menyimpulkan *Affectnet* 66,71%, *RAF-DB* 80,42%, dan *JAFFE* 92,85%.

Kata kunci: Pembelajaran Mendalam, Augmentasi Data, *YOLOv3*, *Affectnet*, *RAF-DB*, *JAFFE*, *MixUp*

ABSTRACT

Facial expressions are a way for humans to communicate and interact. Expressions that appear accidentally are feelings that are actually felt. Mistakes in judging other people's feelings are caused by only seeing at a glance. There needs to be a system innovation to be able to monitor someone's expression. Machine learning has been applied to make facial expression recognition even though it has been successful in certain fields there are still weaknesses to get optimal results when faced with large datasets. The solution to overcome this is to approach the deep learning method. The problem that often occurs in deep learning is the lack of datasets used. One solution to solve this deficiency is to perform data augmentation. There are many methods of data augmentation but in this study using the Crop, Shear, Rotate, Flip and MixUp approaches. In this study, a comparison of performance and detection time will be carried out to find the effect of data augmentation on 3 different datasets, namely Affectnet, RAF-DB, and JAFFE using the YOLOv3 method. To get the best results, each dataset will be tested with 7 different scenarios. From the results obtained, this study concludes Affectnet 66.71%, RAF-DB 80.42%, and JAFFE 92.85%.

Keyword: Deep Learning, Data Augmentation, YOLOv3, Affectnet, RAF-DB, JAFFE, MixUp

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Salah satu cara bagi manusia untuk berkomunikasi dan berinteraksi secara non-verbal adalah melalui ekspresi wajah. Ekspresi wajah dapat menunjukkan emosi yang dialami seseorang seperti sedih, bahagia, kaget, takut, marah, dan *bad mood*. (Reyhan Achmad Rizal, Suardin Gulo, Octaviana Della C. Sihombing, Ardi Bernandustahi Miduk Napitupulu, Amsal Yusuf Gultom, 2019). Seseorang dapat memiliki ekspresi wajah yang dikendalikan sendiri dengan sengaja, namun secara umum ekspresi wajah dapat muncul tanpa disadari sebagai manifestasi perasaan atau emosi pada saat itu. Ekspresi wajah sangat menarik untuk dipelajari karena merupakan komunikasi non-verbal yang dapat digunakan untuk menyampaikan pesan sosial dalam kehidupan manusia dan menggambarkan keadaan emosional seseorang. Oleh karena itu, guna mengetahui kondisi emosional seseorang secara lebih akurat, dibutuhkan suatu metode untuk mengenali ekspresi wajah.

Terkadang seseorang ingin menyembunyikan perasaan atau emosi yang dialaminya, namun hal ini biasanya sulit untuk dilakukan karena secara alami ekspresi wajah akan menunjukkan perasaan yang sebenarnya. Seperti halnya seseorang yang ingin menyembunyikan perasaan kebenciannya terhadap orang lain, tetapi pada saat tertentu secara tidak sengaja akan menunjukkan perasaannya di wajahnya, meskipun dia sangat pandai menyembunyikan perasaan. Akibatnya, banyak orang salah dalam membaca emosi seseorang karena hanya melihat secara

sekilas dari ekspresi wajah. Guna membaca emosi dari ekspresi wajah secara akurat, diperlukan pengamatan secara terus menerus terhadap perubahan ekspresi seseroang. Namun, hal ini tidak dapat dilakukan oleh manusia secara langsung karena pada waktu-waktu tertentu akan mengalami kejenuhan yang mengakibatkan ketidakakuratan (Susanti et al., 2019). Menentukan emosi pada sebuah video adalah salah satu cara untuk mengatasi masalah ini.

Langkah awal yang harus dilakukan dalam melakukan analisis ekspresi adalah deteksi wajah. Deteksi wajah memiliki tujuan untuk menentukan apakah terdapat wajah atau tidak dalam gambar, dimana letak wajah berada, dan ukuran setiap wajah dalam gambar. Dalam deteksi wajah Ada beberapa tantangan seperti posisi wajah, skala wajah, ekspresi wajah, wajah yang tertutup oleh benda lain, dan kondisi pencahayaan (S. Yang et al., 2016). Metode yang biasa digunakan dalam deteksi wajah termasuk metode *Knowledge-Based*, pendekatan *Feature Invariant*, metode *Template Matching*, dan metode *Appearance-Based* (M. H. Yang et al., 2002). Metode *Knowledge-Based* adalah metode deteksi wajah yang bergantung pada seperangkat aturan buatan manusia, misalnya wajah terdiri dari dua mata, satu hidung dan satu mulut dalam jarak tertentu dan posisinya relatif terhadap bagian lain (Chauhan, 2014). Pendekatan *Feature Invariant* adalah teknik untuk deteksi wajah dengan mengekstraksi fitur struktural wajah, seperti kulit wajah (Mahmoodi, 2017) dan tekstur wajah (Saito et al., 2017). Pendekatan *Template Matching* menggunakan templat wajah yang telah ditentukan atau parameter untuk menemukan dan mendeteksi wajah, dengan menghitung nilai korelasi antara template dan gambar yang dimasukkan (Chauhan, 2014). Metode *Appearance-*

Based tergantung pada serangkaian gambar wajah yang dilakukan dengan pelatihan untuk mengetahui model wajah (Chauhan, 2014).

Seiring dengan pengembangan metode pengenalan wajah dan peningkatan kemampuan perangkat keras, banyak penelitian tidak hanya melakukan deteksi wajah tetapi juga dilakukan pengenalan wajah termasuk pengenalan ekspresi wajah. Secara umum, sistem pengenalan wajah biasanya terdiri dari 3 bagian, yaitu *preprocessing*, *feature extraction*, dan *classification* (Y. Wang et al., 2019). Bagian *preprocessing* terdiri dari 2 tahapan. Tahapan pertama, data yang didapatkan biasanya terdapat *noise* dan *kontras* sehingga tidak dapat langsung digunakan. Tahapan kedua, informasi gambar yang didapatkan tidak sesuai dengan ukuran dan sudutnya. Oleh karena itu, *preprocessing* gambar perlu dilakukan. *Preprocessing* merupakan transisi sebelum gambar dilakukan ekstraksi fitur. Klasifikasi adalah bagian terakhir dalam hal pengenalan ekspresi wajah. Maka dari itu kesesuaian ekstraksi fitur dengan metode klasifikasi sangat perlu dipertimbangkan. Beberapa metode yang digunakan untuk ekstraksi fitur antara lain seperti *Local Binary Pattern* (LBP) (Husdi, 2016), Gabor (Jupiyandi et al., 2020), *Local Gabor Binary Pattern* (LGBP) (Canedo & Neves, 2019), *Scale Invariant Feature Transform* (SIFT) (H. Wang & Hou, 2020), dan *Histogram of Oriented Gradient* (HOG) (Esau et al., 2019). Pada pendekatan tradisional, ekstraksi fitur LBP, HOG, dan SIFT telah banyak digunakan karena kinerjanya yang terbukti baik dan juga memiliki biaya komputasi yang murah. Setelah ekstraksi fitur, metode klasifikasi harus diterapkan untuk melakukan pengenalan ekspresi wajah, seperti *Support Vector Machine* (Mustakim et al., 2017), *Random Forest* (Fitri et al., 2019), *Neural Network*

(Jupiyandi et al., 2020), dll. Meskipun metode tersebut telah mencapai kesuksesan besar di bidang tertentu, terdapat beberapa kelemahan yang melekat pada ekstraksi fitur dengan metode tradisional. Kelemahan tersebut antara lain fitur yang tidak berpengaruh pada klasifikasi atau fitur penting yang memiliki pengaruh besar pada klasifikasi dapat dimungkinkan akan hilang ketika menggunakan ekstraksi fitur tradisional. Hal tersebut dikarenakan fitur-fiturnya dibuat secara mandiri, walaupun sebenarnya telah dipertimbangkan dan diantisipasi sebelumnya. Hal ini yang menyebabkan sulitnya mendapatkan hasil pengklasifikasian yang baik pada jumlah *dataset* yang banyak untuk pengenalan ekspresi wajah.

Penelitian yang dilakukan (Hung, Lin, & Lai, 2019) terhadap pengenalan ekspresi wajah Untuk mengetahui tingkat akurasi dari dataset KDEF, JAFFE, FER2013 menggunakan arsitektur Dense_FaceLiveNet terhadap 7 kelas ekspresi wajah. Perbandingan yang digunakan yaitu 70 training, 10 validasi dan 20 testing. Dari eksperimen yang dilakukan dengan menggunakan arsitektur Dense_FaceLiveNet terhadap 7 kelas ekspresi wajah serta diterapkan augmentasi data acak seperti *scaling* dan *translation* yang menghasilkan akurasi untuk dataset JAFFE 90,27% dengan perbandingan subset 70 training, 10 validasi dan 20 testing. Karena dataset yang dipergunakan memiliki jumlah yang sedikit guna untuk meningkatkan akurasi perlu dilakukan proses pelatihan dengan teknik augmentasi data agar lebih memperkaya data training.

Penelitian yang dilakukan (Ding et al., 2020) terhadap pengenalan ekspresi wajah menggunakan dataset RAF-DB terhadap 7 kelas ekspresi wajah menggunakan arsitektur VGG-FACE serta augmentasi data *random brightness* dan *saturation*.

Dari eksperimen yang telah dilakukan didapatkan akurasi masing masing kelas yaitu anger 0.784, disgust 0.644, fear 0.622, happy 0.911, sad 0.812, surprise 0.845, neutral 0.806 dengan rata rata akurasi 0.775. Rendahnya tingkat akurasi pada kelas disgust dan fear perlu dilakukan augmentasi data guna meningkatkan akurasi pada kelas tersebut.

Penelitian yang dilakukan (Y. Li et al., 2020) terhadap pengenalan ekspresi wajah Untuk mendapatkan tingkat akurasi yang lebih baik dari state-of-the art menggunakan arsitektur XceptionNet dan DenseNet terhadap dataset Affectnet. Data yang digunakan 2000 per kelas serta diterapkan augmentasi secara acak di semua kasus termasuk perubahan *brighness* dan *contrast*. Dari eksperimen yang dilakukan dengan menggunakan arsitektur XceptionNet dataset affectnet mendapatkan tingkat akurasi 59%. Perlu dilakukan penelitian lebih lanjut menggunakan arsitektur deep learning lainnya. Dan dapat dilakukan penambahan sampel data per kelas.

Alternatif cara untuk mengatasi permasalahan diatas adalah dengan menggunakan pendekatan *deep learning*. Salah satu metode *deep learning* yang populer saat ini adalah *You Only Look Once (YOLO)*. YOLO merupakan salah satu metode *deep learning* yang menggunakan arsitektur *Convolutional Neural Network (CNN)*. YOLO memiliki akurasi dan kecepatan yang tinggi.

Berdasarkan permasalahan dan beberapa penelitian di atas, dalam kaitannya dengan penelitian ini guna menyelesaikan tantangan, penulis mencoba meneliti dengan menggunakan metode *YOLOv3* untuk pengenalan ekspresi wajah berbasis emosi. Optimalisasi model *YOLOv3* akan dilakukan pada saat proses *pre-*

processing data dengan melakukan pemerataan distribusi data ke setiap kelas sehingga model tidak hanya mengenali kelas kelas tertentu saja serta mencari pengaruh augmentasi data terhadap kinerja metode *YOLOv3*. Kehadiran metode *YOLOv3* sangat diperlukan karena proses pengenalan ekspresi wajah memerlukan pemrosesan data yang cepat dan tingkat akurasi yang tinggi.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan maka dapat dirumuskan rumusan masalah sebagai berikut:

- a. Berapa tingkat akurasi yang dihasilkan metode *YOLOv3* sebelum dan sesudah augmentasi data untuk deteksi ekspresi wajah?
- b. Berapa waktu deteksi yang dihasilkan oleh metode *YOLOv3* sebelum dan sesudah augmentasi data untuk deteksi ekspresi wajah?
- c. Apakah perlakuan augmentasi data pada dataset citra wajah untuk pengenalan ekspresi wajah dengan metode *YOLOv3* berpengaruh terhadap tingkat akurasi yang dihasilkan?

1.3. Batasan Masalah

Guna membatasi ruang lingkup yang terlalu luas atau melebar serta agar penelitian ini bisa lebih fokus untuk dilakukan maka perlu pembatasan masalah. Berikut adalah batasan masalah yang ada dalam penelitian:

- a. Penelitian ini hanya membahas performa metode dalam pengenalan ekspresi wajah yaitu *YOLOv3*.

- b. Data yang akan diuji dalam bentuk gambar.
- c. Dataset yang digunakan berasal dari *Japanese Female Facial Expression (JAFFE)* dengan jumlah 213 data dengan 7 ekspresi wajah.
- d. Dataset yang digunakan berasal dari *Affectnet* dengan jumlah 21000 dengan 7 ekspresi wajah.
- e. Dataset yang digunakan berasal dari *RAF-DB* dengan jumlah 15339 dengan 7 ekspresi wajah.
- f. Ekspresi wajah yang digunakan sebagai objek penelitian adalah *neutral, happiness, sadness, surprise, anger, disgust, dan fear*.
- g. Penelitian ini menggunakan bahasa pemrograman *Python* untuk membantu proses *pre-processing, testing* data uji.
- h. Proses *training* akan dilakukan menggunakan framework *darknet*.
- i. Pelabelan dataset menggunakan aplikasi *labelImg*.
- j. Penelitian ini hanya sebatas kerangka implementasi metode *YOLOv3*.
- k. Penelitian ini tidak membahas pengaruh, arah dan intensitas cahaya terhadap citra ekspresi terhadap citra ekspresi wajah.
- l. Hasil pengenalan wajah akan dievaluasi akurasi dengan menggunakan metode *confusion matrix*.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah:

- a. Mengetahui tingkat akurasi metode *YOLOv3* menggunakan maupun tanpa proses augmentasi data.

- b. Mengetahui waktu deteksi yang dihasilkan oleh metode *YOLOv3* menggunakan maupun tanpa proses augmentasi data.
- c. Mengetahui pengaruh augmentasi data terhadap tingkat akurasi yang dihasilkan oleh metode *YOLOv3*.

1.5. Manfaat Penelitian

Penelitian ini diharapkan dapat bermanfaat bagi:

- a. Pembaca, menjadi salah satu bahan bacaan untuk menambah ilmu dan wawasan terkait penelitian tentang pengenalan ekspresi wajah.
- b. Penulis, mendapatkan banyak ilmu, wawasan, pengalaman baru, dan dapat mengimplementasikan ilmu yang sudah dipelajari selama kuliah pada sebuah penelitian yang bermanfaat.
- c. Peneliti lain, menjadi salah satu referensi untuk menambah ilmu dan wawasan untuk mencoba penelitian tentang pengenalan ekspresi wajah, atau sebagai referensi untuk mengembangkan penelitian ini menjadi lebih baik.
- d. Bidang lain yang terkait, penelitian ini dapat menjadi salah satu metode sistem pengenalan ekspresi wajah yang dapat diimplementasikan pada sistem baru maupun yang sudah dibangun sebagai verifikasi keamanan atau dll.

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

Beberapa penelitian terdahulu yang dijadikan acuan dan tinjauan pustaka dalam penelitian diantaranya adalah:

Penelitian yang dilakukan oleh (Hung, Lin, Lai, et al., 2019) tentang pengenalan ekspresi wajah Untuk mengetahui tingkat akurasi dari ketiga dataset yaitu affectnet, jaffe, dan CK+ menggunakan arsitektur ResNet-50 terhadap 3 kelas ekspresi wajah perbandingan yang digunakan yaitu 80 training 10 testing dan 10 validasi. Dari eksperimen yang dilakukan dengan menggunakan arsitektur ResNet-50 terhadap beberapa dataset dan 3 kelas ekspresi wajah didapatkan tingkat akurasi. Akurasi yang didapatkan untuk dataset JAFFE last-5 70.18, last-10 72.28, last-50 82.89, dan last-75 85.34. Perlu dilakukan penelitian lebih lanjut dengan lebih banyak ekspresi wajah kaitannya masing masing dataset memiliki 7 ekspresi wajah yang telah disediakan. Rendahnya tingkat akurasi pada kelas surprise. Perlu dilakukan pengujian terhadap dataset jaffe dengan arsitektur yang berbeda dengan penambahan augmentasi data serta perbandingan subset *training* dan *testing* yang lebih menantang.

Penelitian yang dilakukan oleh (Hung, Lin, & Lai, 2019) tentang pengenalan ekspresi wajah Untuk mengetahui tingkat akurasi dari dataset KDEF, JAFFE, FER2013 menggunakan arsitektur Dense_FaceLiveNet terhadap 7 kelas ekspresi wajah. Perbandingan yang digunakan yaitu 70 training, 10 validasi dan 20

testing. Dari eksperimen yang dilakukan dengan menggunakan arsitektur Dense_FaceLiveNet terhadap 7 kelas ekspresi wajah menghasilkan akurasi untuk dataset JAFFE 90.27% dengan perbandingan subset 70 training, 10 validasi dan 20 testing dikarenakan dataset yang dipergunakan memiliki jumlah yang sedikit guna untuk meningkatkan akurasi perlu dilakukan proses pelatihan dengan teknik augmentasi data agar lebih memperkaya data training.

Penelitian yang dilakukan oleh (Y. Li et al., 2020) tentang pengenalan ekspresi wajah untuk mendapatkan tingkat akurasi yang lebih baik dari *state-of-the-art* menggunakan arsitektur *XceptionNet* dan *DenseNet* terhadap dataset Affectnet. Data yang digunakan 2000 perkelas. Dari eksperimen yang dilakukan dengan menggunakan arsitektur *XceptionNet* dataset affectnet mendapatkan tingkat akurasi 59%. Perlu dilakukan penelitian lebih lanjut menggunakan arsitektur deep learning lainnya. Dan dapat dilakukan penambahan sampel data per kelas.

Penelitian yang dilakukan oleh (Siqueira et al., 2020) tentang pengenalan ekspresi wajah Untuk mendapatkan tingkat akurasi yang tinggi dengan menggunakan arsitektur buatan yaitu ESR yang berbasis Convolution Neural Network dilakukan pengujian dengan masing masing data 5000 per kelas yang dilakukan pada 8 kelas. Dari eksperimen yang dilakukan arsitektur ESR mendapatkan tingkat akurasi 59.3%. Kecilnya tingkat akurasi pada kelas Disgust dan Surprise. Maka perlu dilakukan penambahan augmentasi data pada setiap kelas untuk meningkatkan keragaman dan varian gambar training.

Penelitian yang dilakukan oleh (Y. Li et al., 2019) tentang pengenalan ekspresi wajah untuk mendapatkan tingkat akurasi yang lebih baik dari penelitian

sebelumnya menggunakan arsitektur ACNN dilakukan pengujian terhadap dataset RAFDB, Affectnet, dan FER. Dari eksperimen yang dilakukan dengan menggunakan arsitektur ACNN mendapatkan tingkat akurasi untuk pACNN 83.27/78.05 dan gACNN 85.07/80.54. Rendahnya tingkat akurasi pada kelas disgust dan fear perlu dilakukan augmentasi data guna meningkatkan akurasi pada kelas tersebut. Perlu dilakukan penelitian dengan menggunakan arsitektur yang berbeda dan akan ditambahkan metode augmentasi data.

Penelitian yang dilakukan oleh (Ding et al., 2020) tentang pengenalan ekspresi wajah dengan melakukan eksperimen menggunakan arsitektur VGG-FACE terhadap 7 kelas pada dataset RAFDB. Dari eksperimen yang telah dilakukan didapatkan akurasi masing masing kelas yaitu anger 0.784, disgust 0.644, fear 0.622, happy 0.911, sad 0.812, surprise 0.845, neutral 0.806 dengan rata rata akurasi 0.775. Rendahnya tingkat akurasi pada kelas disgust dan fear perlu dilakukan augmentasi data guna meningkatkan akurasi pada kelas tersebut dan dilakukan penelitian dengan menggunakan arsitektur yang berbeda.

2.2. Keaslian Penelitian

Tabel 2.1. Matriks literatur review dan posisi penelitian Analisis Pengaruh Augmentasi Data pada Pengenalan Ekspresi Wajah Menggunakan Metode YOLOv3

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Suran atau Kelemahan	Perbandingan
1	Penerapan Metode Single-Layer Feed-Forward Neural Network Menggunakan Kernel Gabor untuk Pengenalan Ekspresi Wajah	Johanes Cristanto, Ken Ratri Retno Wardani, Jurnal Telematika, 2017	Tujuan dari penelitian ini adalah mengukur akurasi penggunaan metode ekstraksi fitur Gabor, adaboost untuk seleksi fitur, dan Feed Forward	Hasil yang didapatkan adalah banyaknya jumlah epoch yang dilakukan maka akan meningkatkan hasil akurasi dari model AdaBoost, akan tetapi terdapat titik jenuh sehingga harus dicari nilai epoch yang optimal. Dalam hal akurasi pada	Tidak dilakukan <i>preprocessing</i> data pemerataan dan penyeleksian ekspresi wajah sehingga pada proses pengujian terdapat ekspresi yang akurasinya kurang baik seperti ekspresi Sedih dan Netral. Pemilihan metode	Pada penelitian ini akan dilakukan <i>preprocessing</i> data yaitu pemerataan data pada 7 ekspresi wajah yang akan diklasifikasikan dan akan dilakukan seleksi wajah yang memiliki ekspresi sesuai dengan kelasnya. Akan

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
			Neural Network untuk mengenali 7 ekspresi wajah manusia yang ada di dalam citra.	pengenalan beberapa ekspresi wajah tidak mendapatkan hasil yang maksimal karena data training yang digunakan kurang kuat dan tidak memiliki ekspresi. Pada penelitian ini pemilihan metode ekstraksi fitur sangat berpengaruh yaitu dengan menggunakan metode ekstraksi fitur Gabor.	untuk klasifikasi ekspresi wajah yaitu <i>Single - Layer Feed Forward Neural Network</i> tidak memberikan peningkatan yang signifikan karena hasil keluaran dari AdaBoost hanya akan dipilih 1.	digunakan metode yang berbeda yaitu deep learning sehingga tidak membutuhkan ekstraksi fitur karena proses ekstraksi dan training akan menjadi satu kesatuan.

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
2	Sistem Pakar Pengenalan Ekspresi Wajah Manusia Menggunakan Metode Kohonen Self Organizing Dan Principal Componen Analysis	Hardiansyah, Bagus Primandati, Puteri Noraisya, Journal of Information Technology, 2018	Tujuan dari penelitian ini adalah pengimplementasi an metode K-SOM dengan ekstraksi fitur PCA untuk mengenali ekspresi wajah	Pada penelitian ini didapatkan kesimpulan bahwa metode yang diusulkan dapat digunakan untuk pengenalan ekspresi wajah dengan kinerja yang baik yaitu 80%. Terdapat kesalahan dalam pengenalan ekspresi wajah karena data yang digunakan untuk data latih terdapat noise.	Data yang digunakan terlihat gelap dikarenakan pengambilan gambar dilakukan pada malam hari dan tanpa menggunakan flash hanya menggunakan bantuan penerangan lampu sehingga terdapat noise. Kinerja dapat ditingkatkan lagi dengan penambahan jumlah data	Pada penelitian ini kelas ekspresi wajah yang digunakan hanya 6 kelas, pada penelitian selanjutnya akan ditambahkan 1 kelas agar data lebih variatif dan untuk jumlah datasetnya akan lebih diperbanyak lagi. Objek data yang digunakan sebagai data uji berupa gambar nantinya pada

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
					yang diuji dan dilatih karena pada penelitian ini hanya menggunakan total 90 data untuk data uji dan latih.	penelitian selanjutnya akan dilakukan pengujian untuk objek data video.
3	PENGENALAN EKSPRESI WAJAH MANUSIA MENGGUNAKAN TAPIS GABOR 2-D DAN SUPPORT	Mustakim, Arif Santoso, Imam Zahra, Ajub Ajulian, Jurnal Ilmiah Teknik Elektro, 2017	Tujuan dari penelitian ini adalah mengklasifikasikan ekspresi wajah dengan menggunakan	Pengenalan ekspresi wajah pada dataset Japanese Female Facial Expression (JAFFE), dihasilkan kinerja sistem terbaik pada variasi 5 panjang gelombang	Pada metode Support Vector Machine untuk fungsi aktivasinya menggunakan limer, akan lebih baik jika dapat diujikan dengan kernel lain dan melihat	Pada penelitian berikutnya akan menggunakan metode yang berbeda yaitu YOLOv3 yang akan menggunakan kernel atau fungsi aktivasi

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
	VECTOR MACHINE (SVM)		fitur ekstraksi tapis gabor 2-D dan metode klasifikasi SVM serta membandingkan data uji dari JAFFE dan data wajah orang indonesia	dengan variasi 8 sudut orientasi dan klasifikasi SVM multikelas One-Against-All dengan nilai akurasi, sensitivitas dan spesifisitas masing-masing sebesar 85,92%, 86,86% dan 97,66%.	hasilnya. Seharusnya dataset JAFFE dan Ekspresi wajah orang indonesia dibuat seimbang sehingga dapat berpengaruh terhadap hasil yang dikeluarkan.	<i>ReLU</i> . Selain perbedaan metode yang digunakan perbedaan juga terdapat pada data yang akan diuji yang berupa video.
4	PENGENALAN CITRA EKSPRESI	Andy Rizky Wiyono, Elly Matul Imah,	Membuat perangkat lunak	Pengimplementasian algoritma PCA dan ELM	Metode yang digunakan dapat mengklasifikasikan	Menggunakan metode YOLOv3 karena dikenal

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
	WAJAH MENGGUNAKAN ALGORITMA PRINCIPAL COMPONENT ANALYSIS (PCA) DAN EXTREME LEARNING MACHINE (ELM)	Jurnal Ilmiah Matematika, 2018	pengenalan ekspresi wajah dengan menggunakan metode Principal Component Analysis (PCA) dan Extreme Learning Machine (ELM)	untuk pengenalan citra ekspresi wajah, setelah dilakukan pengujian didapatkan hasil yang maksimum ketika menggunakan 90 fitur, dengan fungsi aktivasi sigmoid, dengan rasio 4:1, epoch 65 dengan hasil testing akurasi 71,5% dan training akurasi 93,1%.	akan tetapi hasil yang didapatkan kurang maksimal serta memiliki tingkat pemrosesan yang lambat	dengan kecepatan dan akurasi.

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
5	EKSTRAKSI FITUR BERBASIS AVERAGE FACE UNTUK PENGENALAN EKSPRESI WAJAH	Jarot Dwi Prasetyo, Zachol Fatah, Taufik Saleh, Jurnal Ilmiah Informatika, 2017	Mengimplementasikan metode ekstraksi fitur average face pada metode klasifikasi SVM-RBF untuk pengenalan ekspresi wajah	Metode ekstraksi fitur berpengaruh terhadap hasil akhir yaitu akurasi sebesar 90% hal ini dikarenakan terdapat penambahan data pada dataset akan tetapi berpengaruh terhadap menambah durasi waktu proses pengujian.	Pemilihan metode untuk klasifikasi dan banyaknya dataset yang digunakan sangat berpengaruh terhadap performa dan lamanya proses pengujian.	Dataset yang digunakan akan dilakukan pemerataan serta akan digunakan metode deep learning sehingga proses ekstraksi fitur dan klasifikasi akan dilakukan secara sekaligus
6	Pengenalan Ekspresi Wajah	Erwin Yulizar Fardani, Anditya	Menggabungkan Local Gabor	Penggunaan filter gabor hanya menghasilkan	Tidak dilakukan <i>preprocessing</i> data, data	Akan dilakukan <i>preprocessing</i> fitur

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
	Menggunakan LGBP dan SVM	Arifianto, Kurniawan Nur Ramadhani, e- Proceeding of Engineering, 2018	Binary Pattern(LGBP) dan Support Vektor Machine(SVM) untuk pengenalan ekspresi wajah	akurasi 56%, penggunaan klasifikasi SVM one vs all hanya menghasilkan akurasi 48.78% dan 42.77%, dan penggunaan kernel polynomial dengan metode ekstraksi gabor dan svm mendapatkan akurasi 69%.	yang didapatkan langsung diekstrak sehingga berpengaruh terhadap hasil ekstraksi fitur kemudian menurunkan hasil dari klasifikasinya.	dengan pemerataan data setiap kelasnya.

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
7	Real-time facial expression recognition for affect identification using multi-dimensional SVM	Meshach, W. Thambu Hemajothi, S. Anita, E. A.Mary, Journal of Ambient Intelligence and Humanized Computing, 2020	Mengidentifikasi pengaruh metode Multi Dimensi Support Vektor Machine (MDSVM) terhadap klasifikasi ekspresi wajah	Hasil membuktikan bahwa metode yang diusulkan lebih efisien dibandingkan dengan jenis metode klasifikasi Multiclass lainnya.	Batasan utama metodologi ini adalah kemungkinan manusia untuk menyembunyikannya emosi nyata dengan ekspresi wajah eksternal yang berbeda, yang menyebabkan kesalahan klasifikasi dari keadaan sempurna.	Akan digunakan metode yang dapat mengenali ekspresi wajah secara berkelanjutan yaitu dengan metode YOLOv3 pada video

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
8	Facial Expression Recognition using Support Vector Machine (SVM) and Convolutional Neural Network (CNN)	Muhammed, Afeefa Mol, Ramsi Vijay, L. Revathy Ajith, S S Shamna, A R. International Journal of Research in Engineering, Science and Management Volume-3, Issue-8, August-2020.	Penelitian ini mengusulkan metode pengenalan ekspresi wajah berbasis Support Vector Machine (SVM) dengan ekstraksi fitur menggunakan CNN	SVM bekerja lebih baik daripada teknik lain yang sudah ada, dan dengan meningkatkan kinerja keseluruhan pengenalan ekspresi wajah	Kedepannya bisa mendapatkan lebih banyak ekspresi wajah dengan memberikan lebih banyak pelatihan pada gambar menggunakan CNN.	Menambahkan dataset yang akan digunakan. Menggunakan metode yang berbeda.

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
9	Deep Residual Learning for Analyzing Customer Satisfaction using Video Surveillance	Nahemia Sugianto, Dian Tjondronegoro, Beau Tydd, 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2018	Untuk mengetahui tingkat akurasi dari ketiga dataset yaitu affectnet, jaffe, dan CK+ menggunakan arsitektur ResNet-50 terhadap 3 kelas ekspresi wajah perbandingan yang digunakan yaitu 80 training	Dari eksperimen yang dilakukan dengan menggunakan arsitektur ResNet-50 terhadap beberapa dataset dan 3 kelas ekspresi wajah didapatkan tingkat akurasi. Akurasi yang didapatkan untuk dataset JAFFE last-5 70.18, last-perbandingan 10-72.28, last-50 82.89, dan last-75 85.34	Perlu dilakukan penelitian lebih lanjut dengan lebih banyak ekspresi wajah kaitannya masing masing dataset memiliki 7 ekspresi wajah yang telah disediakan. Rendahnya tingkat akurasi pada kelas surprise.	Akan dilakukan pengujian terhadap dataset jaffe dengan arsitektur yang berbeda dengan penambahan augmentasi data serta perbandingan subset training dan testing yang lebih menantang

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
			10 testing dan 10 validasi			
10	Recognizing learning emotion based on convolutional neural networks and transfer learning	Jason C. Hung, Kuan-Cheng Lin, Nian-Xiang Lai, Applied Soft Computing Journal, 2019	Untuk mengetahui tingkat akurasi dari dataset KDEF, JAFFE, FER2013 menggunakan arsitektur Dense_FaceLiveNet terhadap 7 kelas ekspresi	Dari eksperimen yang dilakukan dengan menggunakan arsitektur Dense_FaceLiveNet terhadap 7 kelas ekspresi wajah menghasilkan akurasi untuk dataset JAFFE 90,27% dengan perbandingan subset 70	Karena dataset yang dipergunakan memiliki jumlah yang sedikit guna untuk meningkatkan akurasi perlu dilakukan proses pelatihan dengan teknik augmentasi data agar lebih memperkaya data training.	Akan dilakukan penambahan augmentasi data terhadap dataset JAFFE.

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
			wajah. Perbandingan yang digunakan yaitu 70 training, 10 validasi dan 20 testing	training, 10 validasi dan 20 testing		
11	Facial expression recognition in the wild, by fusion of deep learnt and hand-crafted features	G. Viswanatha Reddy, C.V.R. Dharma Savarni, Snchasis Mukherjee, Cognitive Systems Research, 2020	Untuk mendapatkan tingkat akurasi yang lebih baik dari state-of-the-art menggunakan	Dari eksperimen yang dilakukan dengan menggunakan arsitektur XceptionNet dataset affectnet mendapatkan tingkat akurasi 59%	Perlu dilakukan penelitian lebih lanjut menggunakan arsitektur deep learning lainnya. Dan dapat dilakukan	Penambahan jumlah data per kelas dan akan dilakukan pengujian menggunakan arsitektur deep learning yang berbeda

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
			arsitektur XceptionNet dan DenseNet terhadap dataset Affectnet. Data yang digunakan 2000 per class		penambahan sampel data per kelas	
12	Efficient Facial Feature Learning with Wide Ensemble-Based	Henrique Siqueira, Sven Magg, Stefan Wermter, The Thirty-Fourth AAAI Conference on	Untuk mendapatkan tingkat akurasi yang tinggi dengan	Dari eksperimen yang dilakukan arsitektur ESR mendapatkan tingkat akurasi 59.3%	Kecilnya tingkat akurasi pada kelas Disgust dan Surprise	Perlu dilakukan penambahan augmentasi data pada setiap kelas untuk meningkatkan

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
	Convolutional Neural Networks	Artificial Intelligence, 2020	menggunakan arsitektur buatan yaitu ESR yang berbasis Convolution Neural Network dilakukan pengujian dengan masing masing data 5000 per kelas yang dilakukan pada 8 kelas.			keragaman dan varian gambar training

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
13	Occlusion aware facial expression recognition using CNN with attention mechanism	Hui Ding, Peng Zhou, Rama Chellappa, [arXiv:2005.06040v1 [cs.CV], 2020]	Untuk mendapatkan tingkat akurasi yang lebih baik dari penelitian sebelumnya menggunakan arsitektur ACNN dilakukan pengujian terhadap dataset RAFDB,	Dari eksperimen yang dilakukan dengan menggunakan arsitektur ACNN mendapatkan tingkat akurasi untuk pACNN 83.27/78.05 dan gACNN 85.07/80.54	Rendahnya tingkat akurasi pada kelas disgust dan fear perlu dilakukan augmentasi data guna meningkatkan akurasi pada kelas tersebut	Akan dilakukan penelitian dengan menggunakan arsitektur yang berbeda dan akan ditambahkan metode augmentasi data

Tabel 2.1. Matriks literatur review dan posisi penelitian (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
			Affectnet, dan FER			
14	Deep Neural Network Augmentation: Generating Faces for Affect Analysis	Dimitrios Kollias, Shiyang Cheng, Evangelos Ververas, Irene Kotsia, Stefanos Zafeiropoulos, arXiv:1811.05027v2 [cs.CV], 2019	Dilakukan eksperimen menggunakan arsitektur VGG-FACE terhadap 7 kelas pada dataset RAFDB	Dari eksperimen yang telah dilakukan didapatkan akurasi masing masing kelas yaitu anger 0.784, disgust 0.644, fear 0.622, happy 0.911, sad 0.812, surprise 0.845, neutral 0.806 dengan rata rata akurasi 0.775	Rendahnya tingkat akurasi pada kelas disgust dan fear perlu dilakukan augmentasi data guna meningkatkan akurasi pada kelas tersebut	Akan dilakukan penelitian dengan menggunakan arsitektur yang berbeda dan akan ditambahkan metode augmentasi data

2.3. Landasan Teori

2.3.1. Ekspresi Wajah

Ekspresi wajah ataupun mimik merupakan hasil dari satu ataupun lebih gerakan ataupun posisi otot pada wajah. Ekspresi wajah ialah salah satu wujud komunikasi non-verbal, serta bisa mengantarkan kondisi emosi dari seorang kepada orang yang mengamatinya. Ekspresi wajah ialah salah satu metode berarti dalam mengantarkan pesan sosial dalam kehidupan manusia.

Manusia bisa hadapi ekspresi wajah tertentu secara terencana, namun biasanya ekspresi wajah dirasakan secara tidak terencana akibat perasaan ataupun emosi manusia tersebut. Umumnya amat susah buat menyembunyikan perasaan ataupun emosi tertentu dari wajah, meski banyak orang yang merasa amat mau melaksanakannya. Misalnya, orang yang berupaya menyembunyikan perasaan bencinya terhadap seorang, pada dikala tertentu tanpa terencana hendak menampilkan perasaannya tersebut di mukanya, meski dia berupaya menampilkan ekspresi netral. Ikatan perasaan serta ekspresi wajah pula bisa berjalan kebalikannya, pengamatan menampilkan kalau melaksanakan ekspresi wajah tertentu dengan terencana, bisa pengaruhi ataupun menimbulkan perasaan terpaat betul- betul terjalin.



Gambar 2.1. Ekspresi Wajah Manusia

Sebagian ekspresi wajah bisa dikenal artinya dengan gampang, apalagi oleh anggota spesies yang berbeda, misalnya kemarahan serta kepuasan. Tetapi, sebagian ekspresi yang lain susah dimaksud, misalnya ketakutan serta kejjikan kadangkala susah dibedakan semacam yang nampak pada Gambar 2.1. Tidak hanya itu, kadang-kadang sesuatu wajah bisa disalah artikan hadapi emosi tertentu, sebab lapisan otot-otot wajah orang tersebut secara natural menyamai wajah seorang yang hadapi ekspresi tertentu, misalnya wajah seorang yang nampak senantiasa tersenyum.

2.3.2. *MixUp*

Mixup adalah teknik augmentasi data yang menghasilkan kombinasi berbobot dari pasangan gambar acak dari data pelatihan. Diberikan dua gambar dan label kebenaran dasarnya: $(x_i, y_i), (x_j, y_j)$ contoh pelatihan sintetis (\hat{x}, \hat{y}) dihasilkan sebagai:

$$\hat{x} = \lambda x_i + (1 - \lambda)x_j \quad (2.1)$$

$$\hat{y} = \lambda y_i + (1 - \lambda)y_j \quad (2.2)$$

di mana $\lambda \sim \text{Beta}(\alpha = 0.2)$ sampel secara independen untuk setiap contoh yang diperbesar (Zhang et al., 2018).

2.3.3. Citra Digital

Citra ialah sesuatu representasi(cerminan), kemiripan, ataupun imitasi dari sesuatu objek. Citra dipecah jadi 2, ialah Citra Analog, citra analog ialah citra yang bertabiat kontinyu. Misalnya foto pada monitor tv, dll (T d Sutoyo, Edy Mulyanto, Vincent Suhartono, 2009). Citra Digital, citra digital ialah citra yang bisa diolah oleh *personal computer*.

Suatu citra digital bisa mewakili oleh suatu matriks yang terdiri dari M kolom N baris, dimana perpotongan antara kolom serta baris diucap piksel (piksel= *picture element*), ialah elemen terkecil dari suatu citra. Piksel memiliki 2 parameter, ialah koordinat serta keseriusan ataupun warna. Nilai yang ada pada koordinat(x, y) merupakan $f(x, y)$, ialah besar keseriusan ataupun warna dari piksel di titik itu.

Oleh karena itu, sebuah citra digital dapat ditulis ke dalam bentuk persamaan 1 berikut:

$$f(x, y) = f(0, 0) \dots f(0, M - 1) \quad f(1, 0) \dots f(1, M - 1) \quad \dots \quad f(N - 1, 0) \dots f(N - 1, M - 1) \quad (2.3)$$

Bersumber pada cerminan tersebut, secara matematis citra digital bisa dituliskan selaku fungsi intensitas $f(x, y)$, dimana harga x(baris) serta y(kolom) ialah koordinat posisi serta $f(x, y)$ merupakan nilai fungsi pada tiap titik(x, y) yang melaporkan besar intensitas citra ataupun tingkatan keabuan ataupun warna dari piksel di titik tersebut. Pada proses digitalisasi(sampling serta kuantitas) diperoleh

besar baris M serta kolom N sampai citra membentuk matriks $M \times N$ serta jumlah tingkatan keabuan piksel G (T d Sutoyo, Edy Mulyanto, Vincent Suhartono, 2009).

2.3.4. Computer Vision

Visi komputer(*Computer Vision*) merupakan salah satu teknologi yang sangat banyak dipakai pada era ini. Teknologi *Computer Vision* ini ialah salah satu bidang dari teknologi *Artificial Intelligence*. *Computer Vision* pula ialah serta kumpulan dari metode- metode buat memperoleh, memproses, menganalisis sesuatu foto ataupun dalam makna lain *Computer Vision*, ialah kumpulan metode- metode yang digunakan buat menciptakan angka- angka ataupun simbol- simbol yang didapat dari foto yang diambil dari dunia nyata supaya komputer bisa paham apa arti dari foto tersebut. Inti dari teknologi *Computer Vision* merupakan buat menduplikasi keahlian penglihatan manusia kedalam barang elektronik sehingga barang elektronik bisa menguasai serta paham makna dari foto yang dimasukkan (Sonka et al., 1993).

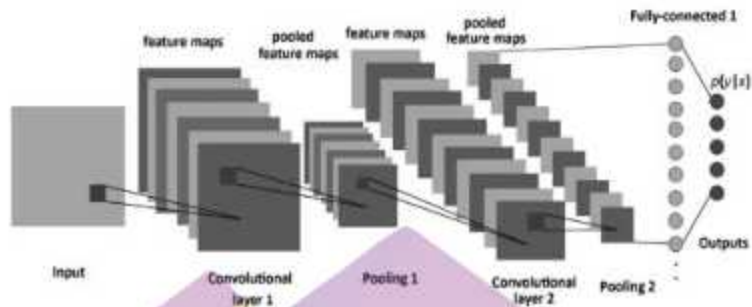
Uraian foto pada komputer ini dicoba dengan menguraikan data simbolis dari informasi foto dengan memakai model yang dibentuk dengan dorongan geometri, fisika, statistika, serta teori dan metode- metode yang lain (Ponce & Forsyth, 2012). *Computer Vision* pula dideskripsikan selaku sesuatu teknologi buat mengotomatisasi serta mengintegrasikan bermacam proses serta representasi buat menciptakan anggapan penglihatan (Dana Harry Ballard & Brown, 1982)

Dalam disiplin ilmu, *Computer Vision* berkaitan dengan teori- teori di balik sistem buatan yang mengekstrak data dari foto. Data yang diekstrak dari foto bisa

berbentuk data- data yang berbeda- beda, semacam urutan jalanya video, intensitas sinar, ataupun perspektif dari sudut foto yang berbeda- beda. Dalam disiplin teknologi, *computer vision* mengusahakan metode supaya teori- teori serta model bisa diterapkan buat pembangunan sistem pada sistem komputer. Contoh sebagian aplikasi yang memakai *computer vision* semacam, perlengkapan navigasi serta kontrol. *Computer Vision* saat ini ini kerap digunakan buat mengetahui wajah pada foto(*face detection*), mengidentifikasi ekspresi wajah(*facial expression recognition*).

2.3.5. Convolutional Neural Network

Convolutional Neural Network merupakan salah satu jenis metode *Deep Neural Network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra. CNN digunakan untuk menganalisis gambar visual, mendeteksi dan mengenali objek pada citra, yang merupakan vektor berdimensi tinggi yang akan melibatkan banyak parameter untuk mencirikan jaringan. Pada CNN memiliki beberapa bagian yaitu lapis masukan (*input layer*), beberapa lapis tersembunyi (*hidden layer*), dan lapis keluaran (*output layer*). Pada lapis tersembunyi terdapat beberapa lapisan yang terdiri dari lapisan ekstraksi fitur dan lapisan klasifikasi yang mana dapat dilihat pada Gambar 2.



Gambar 2.2. Arsitektur Convolutional Neural Network

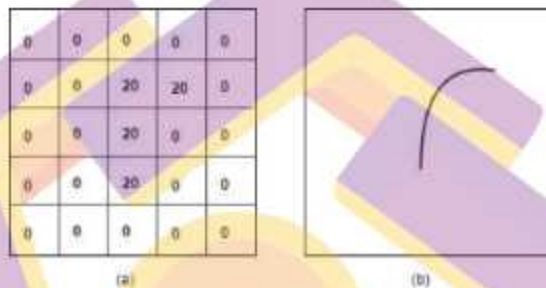
2.3.5.1. Lapisan Ekstraksi Fitur

Lapisan ekstraksi fitur citra terletak pada awal arsitektur yang tersusun atas beberapa lapisan dan setiap lapisan tersusun atas neuron yang terkoneksi pada daerah lokal (*local region*) lapisan sebelumnya. Lapisan pertama adalah lapisan konvolusi (*convolution layer*) dan lapisan kedua adalah lapisan *pooling* (*Pooling Layer*). Pada keluaran setiap lapisan digunakan suatu fungsi aktivasi.

2.3.5.2. Convolutional Layer

Convolution layer adalah blok utama didalam CNN yang terdiri dari beragam filter yang diinisialisasi secara acak untuk melakukan operasi konvolusi yang berfungsi sebagai ekstraksi fitur untuk mempelajari representasi fitur dari suatu input gambar. Pada *convolution layer*, neuron tersusun menjadi *feature maps*. Setiap *neuron* pada *feature map* sebagai *receptive field*, terhubung pada *neuron-neuron* dari *convolution layer* sebelumnya melalui serangkaian bobot yang dilatih, biasa juga disebut dengan filter bank (Lecun et al., 2015). *Convolution layer*

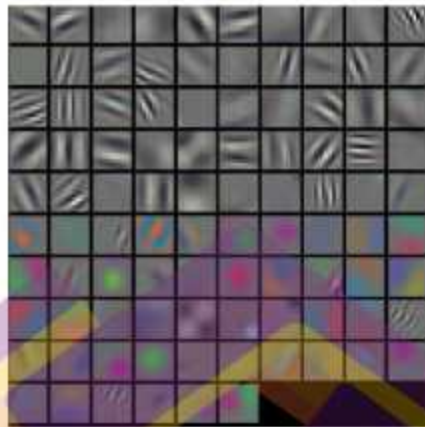
pertama merupakan *low-level filtering* yang digunakan untuk mendeteksi *edge*, *curve*. Sebagai contoh *filter* yang digunakan adalah $5 \times 5 \times 3$ dan filter ini akan melakukan deteksi *curve*. Sebagai deteksi *curve*, *filter* akan memiliki struktur piksel dimana didalamnya akan terdapat nilai yang lebih tinggi sepanjang area yang terdeteksi sebagai *curve* seperti yang digambarkan pada gambar 3.



(a) Representasi piksel pada *filter*, (b) Visualisasi *curve* yang terdeteksi

Gambar 2.3. *Convolutional Layer Filter*

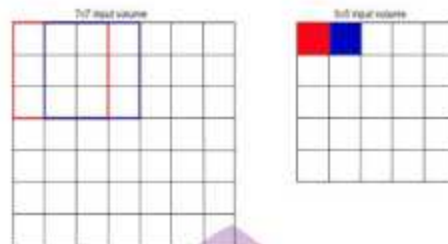
Deteksi *edge* dan *curve* hanya bentuk sederhana dari banyak *filter* lainnya yang digunakan pada *convolution layer* pertama, berikut dibawah ini adalah gambar 4 visualisasi dari *weight convolution layer* pertama. Convolution layer berikutnya merupakan *higher level feature* untuk mendeteksi fitur yang lebih kompleks.



Gambar 2.4. Convolutional Layer Pertama

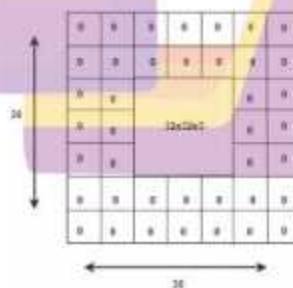
a. *Stride* dan *Padding*

Stride mengontrol seberapa besar pergeseran dari suatu *filter* pada *convolution layer* saat melewati serangkaian data. *Stride* biasanya diatur sedemikian rupa sehingga volume dari output yang dihasilkan adalah bilangan bulat dan bukan pecahan. Jumlah *stride* yang digunakan mengurangi ukuran volume pada layer berikutnya, dan dapat diatur seberapa banyak *overlap* yang dibutuhkan antara 2 nilai output di dalam layer.



Gambar 2.5. Convolutional Filter dengan Stride

Pada gambar 2.5 merupakan ilustrasi dari convolution layer filter dengan *stride* atau pergeseran *receptive field* sebanyak 1 unit, sehingga output volume menyusut dari input volume 7x7 menjadi 5x5. semakin besar *receptive field* ditingkatkan maka akan output volume akan terus berangsur menyusut lebih kecil. Untuk mendapatkan output volume yang tidak terlalu kecil atau sama besarnya dengan input volume, maka *zero padding* dapat diterapkan pada *layer* tersebut.



Gambar 2.6. Convolutional Zero Padding

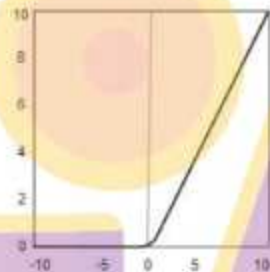
Pada Gambar 2.6 *zero padding* ditambahkan sebanyak 2 unit disekitar *input volume*, sehingga menghasilkan $36 \times 36 \times 3$. Dengan *stride* sebesar 1 unit maka *zero-padding* dapat dihitung dengan persamaan 2:

$$Padding = \frac{(K - 1)}{2} \quad (2.4)$$

Dimana K adalah ukuran dari *filter*. *Stride* sebesar 1 unit maka *output volume* akan sama besarnya dengan *input volume*-nya.

2.3.5.3. ReLU (Rectified Linear Unit)

$$f(x) = \max(0, x) \quad (2.5)$$



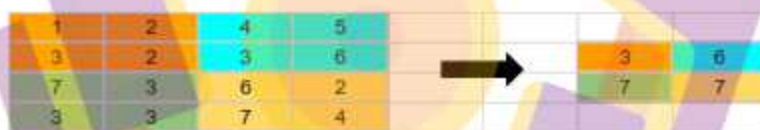
Gambar 2.7. Fungsi Aktivasi ReLU

ReLU merupakan fungsi aktivasi yang digunakan pada *neuron* seperti fungsi aktivasi lainnya, setiap node menggunakan fungsi aktivasi *rectifier* yang dikenal dengan *node* ReLU. Alasan utama ReLU digunakan karena lebih efisien dalam melakukan komputasi dibandingkan dengan fungsi aktivasi konvensional seperti *sigmoid* dan *hyperbolic tangent*, tanpa membuat perbedaan yang signifikan terhadap generalisasi pada akurasi. Fungsi aktivasi *rectifier* digunakan sebagai

pengganti fungsi aktivasi *linear* untuk menambahkan *non-linearity* kedalam *network*, jika *network* hanya dapat menghitung fungsi linier.

2.3.6. Lapisan *Pooling*

Setelah ReLU, biasanya pada lapisan berikutnya diterapkan *Max Pooling*. Tujuan dari penggunaan *max pooling* adalah untuk melakukan *downsampling* dari representasi, mengurangi dimensi yang memungkinkan hal ini dilakukan untuk membantu penyesuaian secara berlebihan dengan menyediakan bentuk representasi dengan nilai maksimum dari input. Selain itu, *max pooling* mengurangi biaya komputasi dengan mengurangi jumlah parameter untuk dipelajari dan memberikan inventarisasi translasi dasar ke representasi internal.

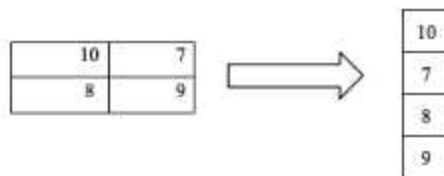


Gambar 2.8. *Maxpool Layer*

2.3.6.1. Lapisan Klasifikasi

a. *Flatten*

Flatten mengubah nilai dari sebuah matriks dua dimensi menjadi matriks satu dimensi atau biasa disebut vektor. *Array* ini disusun dari atas kiri matriks ke kanan setelah itu pindah ke baris selanjutnya matriks dimulai dari kiri dan seterusnya. Vektor ini akan digunakan sebagai *input* dari *fully-connected layer*.



Gambar 2.9. Perubahan matriks 2 dimensi menjadi vektor

b. *Fully Connected Layer*

Fully Connected layer adalah merupakan *multilayer perceptron* (MLP) tradisional yang menggunakan fungsi aktivasi *softmax* pada lapisan output. Istilah "*Fully Connected*" menyatakan bahwa setiap *neuron* pada layer sebelumnya adalah terhubung kedalam setiap *neuron-neuron* pada lapisan berikutnya.

Output dari *convolution* dan *pooling* layer merepresentasikan *high level features* hasil ekstraksi dari *input* gambar. Layer ini pada dasarnya mendapatkan *volume input* (*convolution, ReLU* atau *pooling Layer*) dan menghasilkan vektor dimensi N dimana N adalah jumlah kelas. Tujuannya adalah menggunakan fitur-fitur tersebut untuk mengklasifikasikan gambar kedalam berbagai kelas berdasarkan yang sudah diinisialisasi pada data *training*.

Perbedaan utama dari MLP adalah dari input layer yang bukan vektor, melainkan nilai dari volume aktivasi yang diambil sebagai input.

Fully connected layer dapat didefinisikan seperti pada persamaan 3:

$$z_i^{(l)} = \sum_{j=1}^{m_1^{(l-1)}} m_2^{(l-1)} \sum_{s=1}^{m_3^{(l-1)}} w_{i,j,s}^{(l)} (y_i^{(l-1)})_{r,s} \quad (2.6)$$

Tujuan dari *fully connected* secara keseluruhan adalah mengatur parameter weight $w_{i,j} (l)$ atau $w_{i,j,r,s} (l)$ untuk membuat representasi *stochastic likelihood* dari setiap kelas berdasarkan dari *activation map* yang dihasilkan oleh serangkaian *convolutional, non-linearity, rectification* dan *pooling layer*.

c. Softmax

Fungsi *softmax* menghitung distribusi probabilitas dari kejadian di atas 'n' dari berbagai kejadian. Secara umum, fungsi ini akan menghitung probabilitas masing-masing kelas tujuan melalui semua kelas yang ada. Kemudian probabilitas melakukan kalkulasi yang sangat membantu untuk menentukan kelas tujuan dari input yang diberikan.

Keuntungan utama menggunakan *Softmax* adalah output dengan rentang probabilitas. Rentang akan 0 sampai 1, dan jumlah semua probabilitas akan sama dengan satu. Jika fungsi *softmax* digunakan untuk model *multi-classification*, *softmax* dapat mengembalikan probabilitas masing-masing kelas dan kelas target akan memiliki probabilitas tinggi.

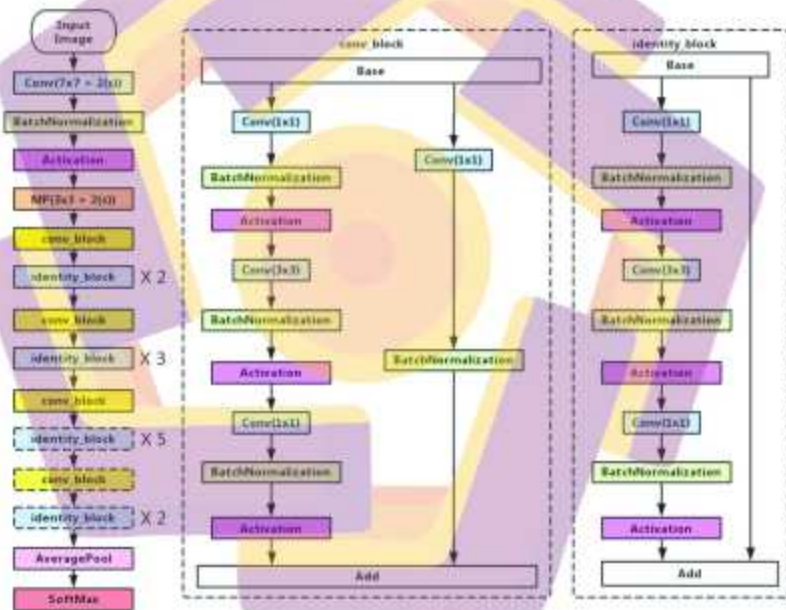
Fungsi softmax didefinisikan sebagai persamaan 4 :

$$F(x_i) = \frac{\text{Exp}(x_i)}{\sum_{j=0}^k \text{Exp}(x_j)} \quad \text{dimana } i = 0,1,2,\dots,k \quad (2.7)$$

2.3.6.2. Residual Neural Network

ResNet merupakan singkatan dari *residual neural network* yang memiliki kelebihan yang disebut *residual block* yang memiliki fungsi untuk mencegah

vanishing gradient problem yang mengakibatkan nilai *gradient* menjadi kecil sehingga menjadi tidak akurat. ResNet pertama kali diperkenalkan pada tahun 2015 oleh Kaiming He di ajang *ImageNet Large Scale Visual Recognition Competition (ILSVRC)* (Russakovsky et al., 2015). ResNet memiliki beberapa macam arsitektur seperti Resnet-18 dan Resnet-50. Perbedaan angka pada nama ResNet adalah penerapan jumlah banyaknya layer yang digunakan pada arsitektur tersebut. Arsitektur ResNet-50 dapat dilihat pada Gambar 2.10.



Gambar 2.10. Arsitektur ResNet50

2.3.7. You Only Look Once (YOLO)

Metode *YOLO* adalah metode yang didasarkan dengan *regression* yang memprediksi menggunakan *bounding box* dan *class object* untuk menentukan

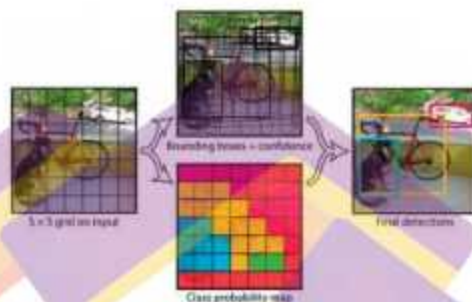
location dari sebuah objek pada citra dalam satu kali algoritma. Algoritma YOLO bekerja dengan cara membagi citra menjadi beberapa sel, setiap sel digunakan untuk memprediksi 5 *bounding box* jika objek pada citra ada lebih dari satu. Prediksi akan menghasilkan nilai prediksi. Hasil prediksi akan dikumpulkan dan *bounding box* dengan probabilitas terkecil akan dihapus. Sedangkan probabilitas yang memiliki nilai terbesar merupakan hasil akhir prediksi. Proses pada algoritma YOLO dapat dilihat pada Gambar 2.11.



Gambar 2.11. Arsitektur YOLO

Prediksi pada YOLO menggunakan arsitektur yang mirip seperti *Convolutional Neural Network* (CNN). YOLO hanya menggunakan *convolutional layer* dan *pooling layer*. *Convolutional layer* terakhir pada arsitektur YOLO disesuaikan dengan jumlah kelas dan jumlah kotak prediksi yang ditentukan.

Rumus untuk menghitung ukuran keluaran pada convolutional layer terakhir dapat dilihat pada persamaan (5).



Gambar 2.12. Ilustasi Metode YOLO

$$Y = S, S, B \times (5 + C) \quad (2.8)$$

Keterangan:

Y = Ukuran keluaran

S = Jumlah baris atau kolom *grid*

B = Jumlah prediksi yang diinginkan pada setiap *grid*

C = Jumlah kelas yang digunakan untuk prediksi

Sebelum pendeteksian, algoritma YOLO membutuhkan proses anotasi terlebih dahulu untuk data. Setiap data memiliki nama kelas, titik koordinat X objek, titik koordinat Y objek, panjang *bounding box*, dan lebar *bounding box*

BAB III

METODE PENELITIAN

3.1. Jenis, Sifat, dan Pendekatan Penelitian

3.1.1. Jenis Penelitian

Jenis penelitian adalah penelitian eksperimental yaitu dengan melakukan pengujian terhadap data kepada suatu metode dengan tujuan untuk mendapatkan kesimpulan yang berupa fakta data statistik. Eksperimen yang dilakukan yaitu dengan menggunakan metode *YOLOv3* yang diperbaiki pada dataset yang akan digunakan untuk membuat model *YOLOv3* untuk mencari akurasi terbaik dan juga pengaruh augmentasi data pada metode *YOLOv3*.

3.1.2. Sifat Penelitian

Dari segi sifatnya penelitian ini bersifat deskriptif dengan pemahaman masalah yang baik dan dayang yang terstruktur.

3.1.3. Pendekatan Penelitian

Penelitian ini menggunakan pendekatan kuantitatif, yaitu penjelasan bagaimana sesuatu dibangun dan bekerja yang berupa fakta dan bersifat objektif dengan skala numerik.

3.2. Metode Pengumpulan Data

Pada penelitian ini akan menggunakan tiga dataset yang berbeda yaitu dataset *Affectnet*, *RAF-DB* dan *JAFFE*.

3.3. Japanese Female Facial Expression (JAFFE)

Japanese Female Facial Expression (JAFFE) dataset. *JAFFE* memiliki total 213 gambar wajah dari 10 orang yang berbeda yang menampilkan tujuh jenis ekspresi: *neutral*, *happiness*, *sadness*, *surprise*, *angry*, *disgust*, dan *fear* (Lyons et al., 2017). Pembagian data pada setiap ekspresi dapat dilihat pada Tabel 3.1.

Tabel 3.1. Jumlah citra perkelas dataset *JAFFE*

No	Kelas	Jumlah
1	<i>anger</i>	30
2	<i>disgust</i>	29
3	<i>fear</i>	32
4	<i>happiness</i>	31
5	<i>neutral</i>	30
6	<i>sadness</i>	31
7	<i>surprise</i>	30
Total		213

Pada masing-masing orang terdapat empat gambar, dengan ukuran piksel 256 x 256. Gambar tampak depan dengan ukuran yang hampir sama. Rambut diikat untuk memperlihatkan semua zona ekspresi. Contoh gambar pada database *JAFFE* dapat dilihat pada Gambar 3.1.



Gambar 3.1. Contoh data pada database *JAFFE*

3.4. *Affectnet*

Affectnet merupakan dataset ekspresi wajah di alam liar. *Affectnet* berisi lebih dari 1M gambar wajah yang bersumber dari internet dengan menanyakan tiga mesin pencari utama menggunakan 1250 kata kunci terkait emosi dalam enam Bahasa berbeda. Sekitar setengah dari gambar yang diambil (~440K) telah dianotasi secara manual untuk kehadiran tujuh ekspresi wajah terpisah (model kategori) dan intensitas valensi dan gairah (model dimensi). *Affectnet* saat ini merupakan dataset ekspresi wajah di alam liar terbesar (Mollahosseini et al., 2019). Berikut adalah pembagian data perkelas pada database *Affectnet* dapat dilihat pada Tabel 3.2:

Tabel 3.2. Jumlah citra per-kelas database *Affectnet*

No	Kelas	Jumlah
1	<i>happy</i>	75374
2	<i>Sad</i>	134915
3	<i>Surprise</i>	25959
4	<i>Fear</i>	14590
5	<i>Disgust</i>	6878
6	<i>Anger</i>	4303
7	<i>Contempt</i>	25382
8	<i>None</i>	33588
9	<i>Uncertain</i>	12145
10	<i>Non-face</i>	82915
Total		420299

3.5. Real-world Affective Faces (RAF) Databases

RAF-DB adalah database ekspresi wajah berskala besar dengan sekitar 30 ribu gambar wajah yang sangat beragam yang diunduh dari internet. Berdasarkan anotasi crowdsourcing, setiap gambar telah diberi label secara independent oleh sekitar 40 annotator. Gambar dalam database ini sangat bervariasi dalam usia subjek, jenis kelamin, dan etnis, pose kepala, kondisi pencahayaan, oklusi, (misalnya kacamata, rambut wajah, atau oklusi diri), operasi pasca-pemrosesan (misalnya berbagai filter dan efek khusus), dll. RAF-DB memiliki keragaman besar, jumlah besar, dan anotasi yang kaya, termasuk:

- a. 29672 jumlah gambar dunia nyata.
- b. Sebuah distribusi ekspresi 7-dimensi vector untuk setiap gambar.
- c. Dua subset yang berbeda: subset label tunggal, termasuk 7 kelas emosi dasar; subset dua tab, termasuk 12 kelas emosi majemuk,
- d. 5 lokasi landmark akurat, 37 *automatic landmark location, bounding box*, ras, rentang usia, dan anotasi atribut gender per gambar,
- e. Dasar output classifier untuk emosi dasar dan emosi senyawa.

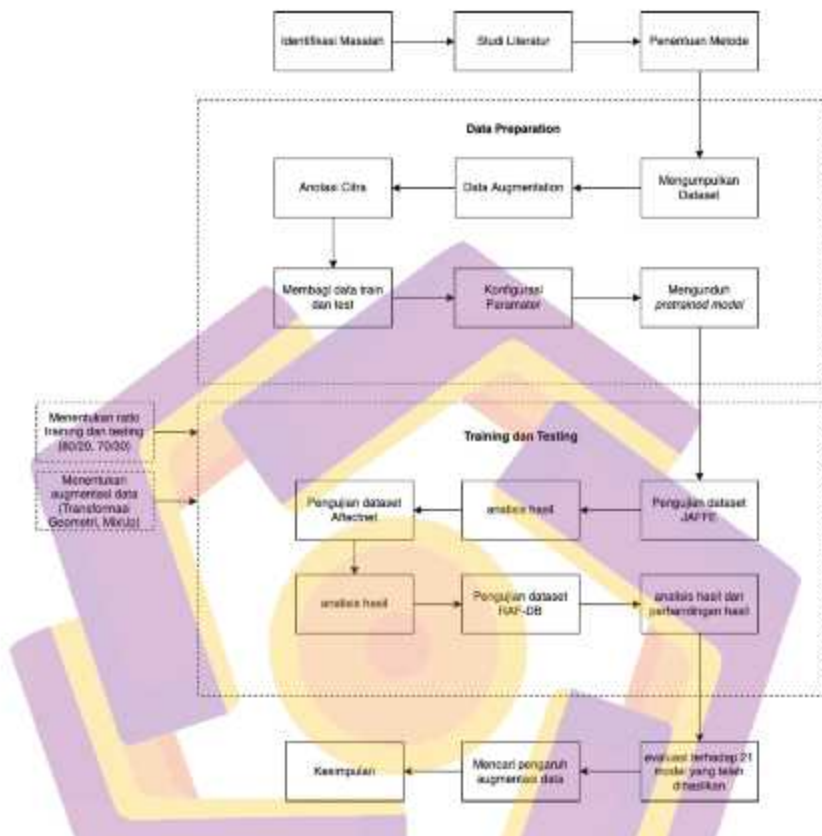
Untuk dapat mengukur kinerja entri pengikut secara objektif, database telah dipecah menjadi set pelatihan dan set tes dimana ukuran set pelatihan lima kali lebih besar dari set tes, dan ekspresi di kedua set memiliki hamper distribusi identik (S. Li et al., 2017).

3.6. Metode Analisis Data

Dalam menganalisa data pada penelitian ini menggunakan metode *YOLOv3*. Tahapannya yaitu menyiapkan dataset yang akan digunakan. Citra yang didapatkan akan dibagi menjadi 2 buah *subset* yaitu untuk data latih dan data uji dengan persentase perbandingan 80:20 dan untuk lebih menantang akan dilakukan juga pengujian menggunakan rasio 70:30. Kemudian akan dilakukan pendistribusian citra terhadap data latih sama rata untuk setiap kelas dan variasi yang ada. Tahap selanjutnya adalah melakukan anotasi citra. Anotasi citra merupakan proses pelabelan dengan cara memberikan kotak batas (*bounding box*) beserta nama kelas pada objek di setiap citra. Tahap selanjutnya melakukan proses *training* data, *training* data pada *YOLOv3* membutuhkan 2 *file* yaitu **.cfg* dan **.weight (pre-trained model)*. *File *.cfg* merupakan konfigurasi *hyper-parameter* untuk model sedangkan **.weights* merupakan bobot hasil pelatihan yang dapat digunakan untuk melakukan *transfer learning*. *File* ini bisa didapatkan dari website resmi *YOLOv3* yaitu (<https://pjreddie.com/darknet/yolo/>). Selanjutnya membuat *file* dengan ekstensi *classes.names* yang berisi 7 kelas ekspresi wajah. Kemudian proses klasifikasi akan dilakukan dengan menggunakan framework *darknet* yang berjalan pada bahasa pemrograman *python*.

3.7. Alur Penelitian

Alur penelitian yang akan dilakukan pada penelitian ini dapat dilihat pada bagan pada Gambar 3.3:



Gambar 3.2. Alur Penelitian

Alur penelitian secara sistematis dapat dilihat pada Gambar 3.2 dan akan dijelaskan sebagai berikut:

a. Identifikasi masalah

Proses ini merupakan tahap dimana peneliti mencari tahu permasalahan yang ada dengan mencari sumber informasi berupa artikel terkait dan berita-berita publik dengan sumber terpercaya.

b. Studi Literatur

Pada proses ini, peneliti mencari tahu informasi dengan membaca jurnal penelitian dan buku yang dianggap relevan dengan permasalahan yang akan diangkat untuk memilih metode atau algoritma yang dianggap sesuai dengan permasalahan.

c. Penentuan Metode

Setelah membaca berbagai jurnal yang berkaitan dengan permasalahan pada proses ini penulis akan menentukan metode yang akan digunakan untuk menyelesaikan permasalahan yang ada. Metode yang akan digunakan pada penelitian ini adalah *You Only Look Once* versi 3 (*YOLOv3*) untuk membantu proses deteksi dan klasifikasi.

d. Data *preprocessing*

1) Mengumpulkan Dataset

Setelah menentukan metode yang digunakan, tahap berikutnya yang akan dilakukan adalah mengumpulkan data yang diperlukan dan relevan dengan metode yang akan digunakan. Data pada penelitian ini merupakan data sekunder yang mana sumber didapat dari orang ketiga atau tidak secara langsung. Data yang digunakan adalah data citra wajah *JAFFE* yang mana untuk mendapatkannya harus *request access* terlebih dahulu kepada pihak *zenodo*. Setelah mendapatkan akses kemudian akan dikirimkan link untuk mengunduh dataset *JAFFE*. *JAFFE* memiliki total 213 gambar wajah dari 10 orang yang berbeda yang menampilkan tujuh jenis ekspresi: netral, bahagia, sedih, terkejut,

marah, jijik, dan ketakutan. Pada masing-masing orang terdapat empat gambar, dengan ukuran piksel 256 x 256.

2) Merubah format dari *.tiff ke *.jpg

Data citra wajah yang telah didapatkan dari tahap pengumpulan dataset masih berupa citra dengan format *tiff*. Untuk dapat diproses menggunakan *darknet* maka data citra perlu dikonversi menjadi *jpg*.

3) Augmentasi Data

Setelah data citra menjadi format *JPEG* atau yang berekstensi *jpg*, maka selanjutnya adalah melakukan augmentasi data. Ada dua augmentasi data yang dilakukan yaitu kombinasi *Flip*, *Rotate*, *Shear*, *Crop* dan *MixUp*. Kelima augmentasi dipilih sebagai metode augmentasi data karena augmentasi data sendiri adalah proses penduplikasian citra tanpa merubah makna dari citra tersebut. Augmentasi *Flip* yang akan dilakukan adalah horizontal, *Rotate* yang dipilih antara -15 derajat dan +15 derajat, *Crop* 10% dan *Shear* +15% Horizontal dan +15% Vertical. Sedangkan untuk *MixUp* adalah kombinasi antara 2 gambar menjadi 1.

4) Anotasi Citra

Anotasi citra adalah proses proses pelabelan dengan memberikan kotak pembatas (*bounding box*) pada sebuah citra yang akan menghasilkan file *.txt* yang berisi informasi tentang lokasi object dalam suatu gambar.

5) Membagi data latih dan data uji

Tahap selanjutnya setelah citra dianotasi adalah dengan membagi keseluruhan dataset baik data sebelum augmentasi ataupun sesudah augmentasi menjadi dua bagian yaitu 80 % sebagai *training set* dan 20% sebagai *testing set*. Data juga dibagi kedalam *subset* 70% untuk *training data* dan 30% untuk *testing data*. Pembagian data dipisahkan ke dalam sebuah file berekstensi *.txt* yang mana akan menghasilkan *train.txt* dan *test.txt* kedua file tersebut berisi lokasi letak citra pada lokasi penyimpanan.

6) Konfigurasi *hyper-parameter*

Tahap selanjutnya setelah membagi data *train* dan *test* adalah dengan menyiapkan file yang berekstensi **.cfg* yang mana *file* ini berisi konfigurasi *hyper-parameter* yang dapat disesuaikan dengan kebutuhan untuk keperluan mendapatkan kinerja terbaik dari model *YOLOv3*.

7) *Transfer-learning*

Tahap selanjutnya setelah menyiapkan file **.cfg* adalah mendapatkan model pra-pelatihan (*pre-trained model*) dari (<https://pjreddie.com/darknet/yolo/>) yang sebelumnya telah dilatih oleh *ImageNet*. Fungsi dari model pra-pelatihan adalah sebagai *starting point* sebelum melakukan proses *training* untuk mendapatkan model yang baru.

e. Skenario *YOLOv3*

Karena jenis penelitian ini bersifat eksperimen maka alur penelitian ini disediakan langkah skenario. Sebelum peneliti memulai tahap skenario ada beberapa tahap sebelumnya yang dilakukan. Salah satu tahapnya adalah menyiapkan data citra dengan *preprocessing* sebagai *training set* dan *testing set*. Pada penelitian tahap skenario *YOLOv3* ini peneliti ingin mencari akurasi tertinggi. Maka dilakukan pengujian terhadap 3 database yang berbeda 2 dari ketiga database merupakan database terbesar dan terbaru pada pengenalan ekspresi wajah. Pada ketiga database tersebut akan dilakukan 7 skenario antara lain. Skenario pertama dan kedua adalah pengujian yang dilakukan pada database asli tanpa proses augmentasi data menggunakan arsitektur *YOLOv3* yang menggunakan *backbone* Darknet-53. Skenario ketiga dan keempat dilakukan augmentasi data yaitu Flip Horizontal, Crop 10%, Rotation antara -15% dan +15%, dan Shear +-Horizontal, +-15% Vertical kemudian akan dibedakan berdasarkan ratio pembagian data latih dan data uji. Skenario kelima dan keenam akan digunakan metode augmentasi MixUp dan akan dibedakan pada ratio untuk data latih dan data uji. Skenario ketujuh dengan menggunakan ratio data latih dan data uji 80/20 kemudian akan menggabungkan metode augmentasi Flip Horizontal, Crop 10%, Rotation antara -15% dan +15%, dan Shear +-Horizontal, +-15% Vertical dengan MixUp. Maka masing masing dataset akan dilakukan pengujian dengan 7 skenario sehingga akan didapatkan keseluruhan 21 skenario yang mana dapat dilihat pada Tabel 3.3.

Tabel 3.3. Skenario Pengujian

No	Dataset	Skenario	Keterangan
1	<i>JAFFE, Affectnet, RAF-DB</i>	Skenario 1	Skenario ini bertujuan untuk menguji penggunaan perbandingan subset pelatihan dan pengujian dengan rasio 80/20 tanpa menggunakan metode augmentasi data
2		Skenario 2	Skenario ini bertujuan untuk menguji penggunaan perbandingan subset pelatihan dan pengujian dengan rasio 70/30 tanpa menggunakan metode augmentasi data, rasio ini dipilih karena penelitian sebelumnya hanya menggunakan rasio 80/20
3		Skenario 3	Skenario ini bertujuan untuk menguji pengaruh augmentasi data transformasi geometri dengan kombinasi rasio 80/20
4		Skenario 4	Skenario ini bertujuan untuk menguji pengaruh augmentasi data transformasi geometri dengan kombinasi rasio 70/30
5		Skenario 5	Skenario ini bertujuan untuk menguji pengaruh augmentasi data yang lain yaitu MixUp dengan kombinasi rasio 80/20
6		Skenario 6	Skenario ini bertujuan untuk menguji pengaruh augmentasi data yang lain yaitu MixUp dengan kombinasi rasio 70/30
7		Skenario 7	Skenario ini bertujuan untuk menguji pengaruh kombinasi dari dua metode augmentasi data yaitu Transformasi Geometri dan Mixup dengan menggunakan rasio 80/20

f. *Training Model*

Setelah menentukan skenario yang ada untuk masing masing dataset yaitu *JAFFE*, *Affectnet*, *RAF-DB* peneliti mulai melakukan percobaan menggunakan dataset citra ekspresi wajah yang telah dipersiapkan sebelumnya. *Training* data bertujuan untuk melatih ketiga skenario yang ada agar saat tahap evaluasi dapat menghasilkan nilai akurasi yang tinggi.

g. *Evaluasi Hasil/Pengujian Data*

Pada tahap ini, model yang telah melalui tahap *preprocessing* dan tahap latih akan dilakukan pengujian model evaluasi. Dalam tahapan evaluasi akan mendapatkan nilai - nilai *confusion matrix* dari setiap skenario untuk menentukan nilai *precision*, dan *recall*.

h. *Kesimpulan*

Setelah mendapatkan hasil penelitian berupa data fakta yang dihasilkan oleh 21 skenario dari 3 dataset yang berbeda, tahap selanjutnya adalah membuat kesimpulan dengan menyajikan hasil dari percobaan yang telah dilakukan dengan beberapa fakta terkait metode *YOLOv3* dan pengaruh augmentasi data terhadap tingkat akurasi pada metode *YOLOv3*.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1. Pengumpulan Data

4.1.1.1. *Japanese Female Facial Expression (JAFFE)*

Dataset yang digunakan pada penelitian ini berasal dari database *Japanese Female Facial Expression (JAFFE)* dataset. *JAFFE* memiliki total 213 gambar wajah dari 10 orang yang berbeda yang menampilkan tujuh jenis ekspresi: *neutral*, *happiness*, *sadness*, *surprise*, *anger*, *disgust*, dan *fear* (Lyons et al., 2017). Pada masing-masing orang terdapat empat gambar, dengan ukuran piksel 256 x 256. Gambar tampak depan dengan ukuran yang hampir sama. Rambut diikat untuk memperlihatkan semua zona ekspresi. Citra pada dataset *JAFFE* dapat dilihat pada Gambar 4.1.



Gambar 4.1. Citra pada dataset *JAFFE*

4.1.2. *Affectnet*

Affectnet merupakan dataset ekspresi wajah di alam liar terbesar yang memiliki total lebih dari 1M citra wajah. Pada penelitian ini akan dilakukan pengujian terhadap dataset *Affectnet* permasalahan yang terjadi pada dataset *affectnet* adalah tidak seimbangny jumlah data yang telah dianotasi secara manual

(Y. Li et al., 2020). Oleh karena itu akan dilakukan pengujian dan hanya diambil 3000 citra untuk 7 kelas ekspresi wajah. Sehingga total data citra yang akan digunakan pada penelitian ini adalah 21000. Citra pada dataset *Affectnet* dapat dilihat pada Gambar 4.2.



Gambar 4.2. Citra pada dataset *Affectnet*

4.1.3. *Real-world Affective Faces Database*

Real-world Affective Faces Database atau *RAF-DB* merupakan salah satu dataset dengan jumlah data terbesar saat ini untuk ekspresi wajah. *RAF-DB* memiliki total data 30K citra ekspresi wajah. *RAF-DB* juga telah menyediakan data untuk *training* dan juga *testing* dengan ratio pembagian 80:20 yaitu 12.271 untuk *training* dan 3068 untuk *testing*. Citra pada dataset *RAF-DB* dapat dilihat pada Gambar 4.3.



Gambar 4.3. Citra pada dataset *Affectnet*

4.2. Augmentasi Data (Transformasi Geometri)

Setelah selesai pada tahapan pengumpulan dataset yang berupa dataset citra ekspresi wajah dengan kelas *neutral*, *happiness*, *sadness*, *surprise*, *anger*, *disgust*, dan *fear*, selanjutnya akan dilakukan tahapan pengolahan data yaitu augmentasi data dengan tujuan untuk memaksimalkan citra dan proses deteksi dan klasifikasi. Augmentasi data *cropping*, *translation*, dan *horizontal flip* biasa digunakan pada data pelatihan jaringan saraf, transformasi augmentasi meningkatkan keragaman gambar dan juga dapat mengurangi atau bahkan menghindari *overfitting* (Ho et al., 2019). Pada tahapan ini akan menjelaskan proses augmentasi data yang akan digunakan untuk beberapa skenario kecuali skenario yang menggunakan citra asli. Augmentasi data yang digunakan adalah *Flip Horizontal*, *Crop 10%*, *Shear +15% Horizontal* dan *+15% Vertical* dan *Rotate* antara -15 dan $+15$. Berikut adalah augmentasi data yang akan dilakukan dapat dilihat pada Gambar 4.4:



Gambar 4.4. Contoh proses augmentasi data.

4.3. MixUp

Setelah data asli ataupun data baru selesai dibuat maka akan digabungkan dengan menggunakan metode augmentasi *MixUp*. Metode ini sangat efisien dalam hal regularisasi terhadap model dalam *computer vision* (Liang et al., 2018).

Pada jurnal artikel *mixup* dijelaskan (Liang et al., 2018), dalam jurnal tersebut dijelaskan sebagai contoh pada dataset CIFAR-10 daripada menggunakan dataset mentah lebih baik menggabungkan 2 data gambar menjadi satu secara linear sehingga akan mendapatkan *tensor* data citra baru. Dengan persamaan 2.1 pada bab 2. Variabel λ adalah nilai antara 0 dan 1 yang mewakili skala perbandingan antara ukuran gambar 1 dengan gambar 2. Target label yang ditetapkan untuk gambar baru tersebut adalah kombinasi yang sama dari target asli, dengan rumus sesuai pada persamaan 2.2 bab 2. Variabel λ adalah nilai antara 0 dan 1 yang mewakili bobot pada label gambar, misal terdapat dua citra ekspresi wajah pada gambar dan gambar ekspresi wajah kedua memiliki bobot $\lambda = 0.2$ maka label klasifikasi adalah 70% label gambar 1 dan 30% label gambar 2. Hasil akhir dari proses ini berupa gabungan linear antara 2 gambar sehingga model dapat belajar lebih efisien (Sanjaya & Ayub, 2020). Contoh gabungan dari *mixup* dapat dilihat pada Gambar 4.5:



Gambar 4.5. Contoh gambar dengan *MixUp*

```

if (mixup == 1) {
  // mix images
  for (j = 0; j < d2.X.cols; ++j) {
    d.X.vals[i][j] = (d.X.vals[i][j] + d2.X.vals[i][j]) / 2.0f;
  }
  // mix labels
  for (j = 0; j < d2.y.cols; ++j) {
    d.y.vals[i][j] = (d.y.vals[i][j] + d2.y.vals[i][j]) / 2.0f;
  }
}

```

4.4. Anotasi Citra

Anotasi citra adalah pemberian kotak pembatas (*bounding box*) pada citra yang berisi informasi dari citra tersebut. Pemberian kotak pembatas dilakukan secara manual dari setiap citra yang telah didapatkan. Peng-anotasian citra menggunakan aplikasi tambahan yaitu *labellmg*. Proses anotasi citra dapat dilihat pada Gambar 4.6.



Gambar 4.6. Proses Anotasi Citra

Setelah proses anotasi citra maka akan menghasilkan file *.txt. File ini sesuai dengan nama citra yang dianotasi. Berikut adalah contoh anotasi citra yang telah dibuat dapat dilihat pada Gambar 4.7.

```
0 0.498047 0.597891 0.488281 0.660156
```

Gambar 4.7. Anotasi citra pada ekspresi wajah marah

Kolom pertama merupakan angka yang mewakili kelas objek. Ini berkisar dari 0 sampai dengan n akan tetapi pada penelitian ini terdapat 7 kelas yang berarti 0 sampai dengan 6. Kelas 0 merepresentasikan bahwa citra tersebut merupakan citra

dengan kelas marah (*anger*). Kolom kedua dan ketiga, merupakan koordinat x dan y dari pusat kotak pembatas, dinormalisasi dengan lebar dan tinggi gambar masing-masing. Kolom keempat dan kelima, lebar dan tinggi masing-masing adalah lebar dan tinggi kotak pembatas, dinormalisasi dengan lebar dan tinggi gambar masing-masing. Dari anotasi file pada Gambar 4.7. maka akan didapatkan kotak pembatas seperti yang terlihat pada Gambar 4.8.



Gambar 4.8. Citra yang telah dianotasi

4.5. Membagi data *train* dan data *test*

Tahap selanjutnya adalah pembagian data untuk *training* dan data untuk *testing*. Pembagian data dilakukan untuk proses training untuk membentuk model dan juga untuk evaluasi dari model yang telah terbentuk. Pada penelitian ini akan digunakan ratio perbandingan data training dan data latihan 80:20 dan 70:30. Ratio 80:20 dipilih karena beberapa penelitian terdahulu (Hung, Lin, & Lai, 2019) menggunakan ratio tersebut. Dan agar menjadi tantangan baru untuk penelitian ini maka akan dilakukan pengujian juga menggunakan ratio 70:30.

```
#split 80/20
p_test = p[:int(len(p) * 0.20)]
p = p[int(len(p) * 0.20):]
```

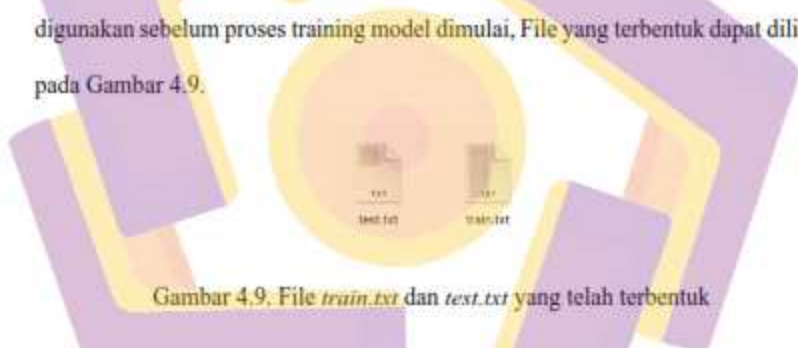
```

with open('train.txt', 'w') as train_txt:
    for e in p:
        train_txt.write(e)
with open('test.txt', 'w') as test_txt:
    for e in p_test:
        test_txt.write(e)

#split 70/30
p_test = p[:int(len(p) * 0.20)]
p = p[int(len(p) * 0.30):]
with open('train.txt', 'w') as train_txt:
    for e in p:
        train_txt.write(e)
with open('test.txt', 'w') as test_txt:
    for e in p_test:
        test_txt.write(e)

```

Berikut adalah hasil dari pembagian data training dan data testing yang akan digunakan sebelum proses training model dimulai, File yang terbentuk dapat dilihat pada Gambar 4.9.



Gambar 4.9. File *train.txt* dan *test.txt* yang telah terbentuk

4.6. Arsitektur YOLOv3

YOLOv3 menggunakan arsitektur dari Darknet53 yang artinya memiliki 53 *convolutional layers*. YOLOv3 memiliki 53 lapis konvolusional yang disebut darknet53 yang terdiri dari struktur convolutional dan residual (Fandisyah et al., 2021). Berikut adalah arsitektur YOLOv3 yang digunakan pada penelitian ini dapat dilihat pada Gambar 4.10.

4.7. Persiapan skenario

Langkah berikutnya setelah mempersiapkan arsitektur YOLOv3 adalah pembuatan skenario, pembuatan skenario dilakukan sesuai dengan jenis penelitian yang bersifat eksperimental dimana peneliti akan mencoba bereksperimen menggunakan metode yang telah ditetapkan sebelumnya.

Metode augmentasi data yang akan digunakan pada penelitian ini adalah Transformasi geometri (Flip, Rotate, Shear, Crop) dan MixUp. Alasan penggunaan transformasi geometri adalah penulis ingin mengetahui pengaruh dari metode augmentasi data dengan transformasi geometri, (Yasmina et al., 2019) Pendistribusian data gambar yang tidak seimbang pada setiap kelas pada dataset memberikan hasil yang kurang optimal pada beberapa kelas pada saat proses pelatihan karena jumlah gambar yang dimiliki lebih sedikit dibandingkan dengan kelas yang lainnya. Untuk memastikan pada saat pelatihan didapatkan hasil yang baik maka augmentasi data dilakukan salah satunya dengan metode transformasi geometri. Alasan lain penggunaan transformasi geometri ini adalah pada penelitian (Hung, Lin, & Lai, 2019) penggunaan transformasi geometri dapat mengurangi overfitting pada saat proses pelatihan dan memberikan lebih banyak varian data sehingga model belajar lebih robust. Kemudian juga akan dilakukan kombinasi serta pengujian secara tunggal dengan menggunakan metode augmentasi data MixUp. MixUp diperkenalkan pada tahun 2018 yang merupakan bentuk metode augmentasi data baru. Konsep MixUp yaitu menggabungkan 2 gambar menjadi satu sehingga akan menghasilkan citra data baru. Pada jurnal [] telah dijelaskan daripada menggunakan dataset mentah lebih baik menggabungkan 2 data gambar menjadi

atau secara linear sehingga akan mendapatkan tensor data citra baru. Pada penelitian tersebut telah diimplementasikan *MixUp* pada dataset CIFAR-10, hasil yang didapatkan yaitu model yang dihasilkan selalu mengalami peningkatan dari segi akurasi. Selain metode augmentasi data yang dibedakan maka pada penelitian ini juga akan dibedakan dari segi ratio subset data pelatihan yang akan dibagi menjadi dua yaitu 80/20 dan 70/30. Alasan menggunakan subset ratio 80/20 karena mengacu pada penelitian sebelumnya (Hung, Lin, & Lai, 2019), (Y. Li et al., 2020), (Ding et al., 2020). Ratio ini tetap akan dilakukan karena peneliti ingin mendapatkan hasil dari metode yang berbeda, sedangkan alasan pemilihan ratio 70/30 yaitu agar pada penelitian ini lebih menantang serta mengetahui hasil dari augmentasi data ketika menggunakan ratio dengan data pelatihan yang lebih kecil.

Sehingga terdapat 7 skenario pada masing masing dataset yang dibuat, skenario 1,2 adalah implementasi metode YOLOv3 pada dataset asli yang nantinya akan digunakan sebagai pembandingan dengan skenario yang menggunakan metode augmentasi data. kemudian skenario 2,3 akan dilakukan proses augmentasi data transformasi geometri, kemudian 5,6 akan dilakukan proses MixUp, dan skenario 7 akan digabungkan antara transformasi geometri dengan MixUp. Pada skenario 1-6 akan dibedakan juga penggunaan rasio data uji dan data latih yang mana pada penelitian ini akan menggunakan rasio 70/30 dan 80/20. Untuk mengetahui pengaruh augmentasi data terhadap tingkat akurasi yang dihasilkan oleh metode YOLOv3, sesuai dengan rumusan masalah pada penelitian ini. Keseluruhan skenario pada masing masing dataset akan menggunakan arsitektur yang sama

perubahan akan dilakukan terhadap perlakuan ketiga dataset. Berikut adalah susunan skenario yang ada.

4.7.1. Skenario dataset JAFFE

4.7.1.1. Skenario JAFFE dengan rasio 80/20

Pada skenario 1 ini, dataset *JAFFE* yang digunakan sebagai data latih yaitu sebanyak 171 citra kemudian untuk data uji yang digunakan sebanyak 42 citra. Pada skenario ini tidak dilakukan proses augmentasi data ataupun proses mixup. Proses pelatihan hanya menggunakan data asli yang dibagi berdasarkan rasio 80:20.

4.7.1.2. Skenario JAFFE dengan rasio 70/30

Pada skenario 2 ini, dataset *JAFFE* yang digunakan sebagai data latih yaitu sebanyak 150 citra kemudian untuk data uji yang digunakan sebanyak 63 citra. Pada skenario ini sama seperti skenario 1, data tidak dilakukan proses augmentasi data ataupun proses mixup. Berbeda dengan skenario 1, proses pelatihan hanya menggunakan data asli yang dibagi berdasarkan rasio 70:30.

4.7.1.3. Skenario JAFFE + Augmentasi dengan rasio 80/20

Pada skenario 3 ini, dataset *JAFFE* yang digunakan sebagai data latih dan data uji sama seperti skenario 1. Akan tetapi pada skenario 3 akan dilakukan proses augmentasi data yaitu dengan menambahkan augmentasi seperti *Flip*, *Crop*, *Rotate*, dan *Shear*. Yang mana dari hasil augmentasi pada data training akan ada penambahan 3x lipat data latih sehingga data yang digunakan sebagai data pelatihan menjadi sekitar 513 sedangkan untuk data uji tidak ada penambahan data

karena pada data pengujian diperlukan data asli bukan data hasil modifikasi. Gambar juga akan di *resize* ke ukuran 416x416. Proses augmentasi data menggunakan *framework Roboflow*. Dataset diupload ke website roboflow untuk augmentasi dan *pre-processing* data. Dataset hasil augmentasi kemudian didownload dalam bentuk file *.zip* kemudian di *extract* dalam sebuah *folder*.

```
!curl -L "https://app.roboflow.com/ds/r7gxbFXmxO?key =P4RWGf2c2R"
>roboflow.zip; unzip roboflow.zip; rm roboflow.zip
```

Berikut adalah contoh hasil dari augmentasi dapat dilihat pada Gambar 4.11:



Gambar 4.11. Hasil augmentasi dataset *JAFFE* skenario 3

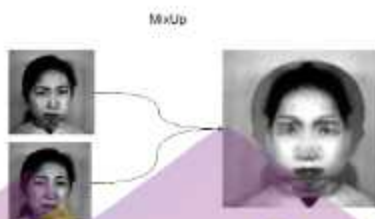
4.7.1.4. Skenario *JAFFE* + Augmentasi dengan rasio 70/30

Pada skenario 4 ini, dataset *JAFFE* yang digunakan untuk data latih dan data uji sama dengan skenario 2 dan dilakukan proses augmentasi data seperti pada skenario ke 3. Sehingga data latih menjadi 450 dan untuk data uji tetap sama tidak ada penambahan data.

4.7.1.5. Skenario *JAFFE* + *MixUp* dengan rasio 80/20

Pada skenario 5 ini, dataset *JAFFE* yang digunakan untuk data latih dan data uji sama dengan skenario 1 kemudian dilakukan proses *mixup*. *Mixup* dilakukan

pada proses pelatihan data. Untuk memberi gambaran bagaimana proses *mixup* dilakukan dan data bagaimana data yang dihasilkan maka berikut adalah contoh proses *mixup* yang dilakukan pada proses pelatihan dapat dilihat pada Gambar 4.12:



Gambar 4.12. Mixup pada dataset *JAFFE* skenario 5

4.7.1.6. Skenario *JAFFE* + *MixUp* dengan rasio 70/30

Pada skenario 6 ini, dataset *JAFFE* yang digunakan untuk data latih dan data uji sama dengan skenario 2 kemudian dilakukan proses *mixup* sama halnya dengan skenario 5. Hal yang berbeda adalah pada skenario ini akan menggunakan rasio 70/30.

4.7.1.7. Skenario *JAFFE* + *Augmentasi* + *MixUp* dengan rasio 80/20

Pada skenario 7 ini, dataset *JAFFE* yang digunakan untuk data latih dan data uji sama dengan skenario 1 kemudian data latih akan dilakukan proses *augmentasi* data *Flip*, *Crop*, *Rotate*, dan *Shear*. Dari data yang dihasil kemudian akan ditambahkan proses *mixup*. Dalam kata lain pada skenario ini mengkombinasikan proses *augmentasi* data dan proses *mixup*. Dari hasil data *augmentasi* akan menggabungkan 2 data menjadi 1 data pada setiap pelatihan dengan menggunakan proses *mixup*.

4.7.2. Skenario dataset *Affectnet*

4.7.2.1. Skenario *Affectnet* dengan rasio 80/20

Pada skenario 1 ini, dataset *Affectnet* yang digunakan sebagai data latih yaitu sebanyak 2.400 citra untuk masing masing kelas sehingga untuk keseluruhan menggunakan 7 kelas ekspresi wajah menjadi 16.800 kemudian untuk data uji yang digunakan sebanyak 600 citra perkelas sehingga total apabila menggunakan 7 kelas yaitu 4200. Pada skenario ini tidak dilakukan proses augmentasi data ataupun proses mixup. Proses pelatihan hanya menggunakan data asli yang dibagi berdasarkan rasio 80:20.

4.7.2.2. Skenario *Affectnet* dengan rasio 70/30

Pada skenario 2 ini, dataset *Affectnet* yang digunakan sebagai data latih yaitu sebanyak 2100 citra perkelas sehingga untuk total keseluruhan apabila digunakan 7 kelas yaitu 14.700, lebih sedikit jika dibandingkan skenario 1 pada dataset *affectnet*. Kemudian untuk data uji yang digunakan sebanyak 900 citra perkelas sehingga total keseluruhan untuk 7 kelas yaitu 6300 citra lebih banyak jika dibandingkan dengan skenario 1 pada dataset *affectnet*. Pada skenario ini sama seperti skenario 1, data tidak dilakukan proses augmentasi data ataupun proses mixup. Berbeda dengan skenario 1, proses pelatihan hanya menggunakan data asli yang dibagi berdasarkan rasio 70:30.

4.7.2.3. Skenario *Affectnet* + Augmentasi dengan rasio 80/20

Pada skenario 3 ini, dataset *Affectnet* yang digunakan sebagai data latih dan data uji sama seperti skenario 1 dataset *affectnet*. Akan tetapi pada skenario 3 akan dilakukan proses augmentasi data yaitu dengan menambahkan augmentasi seperti *Flip*, *Crop*, *Rotate*, dan *Shear*. Yang mana dari hasil augmentasi pada data training akan ada penambahan 3x lipat data latih sehingga data yang digunakan sebagai data pelatihan menjadi sekitar 50.400 sedangkan untuk data uji tidak ada penambahan data karena pada data pengujian diperlukan data asli bukan data hasil modifikasi. Gambar juga akan di *resize* ke ukuran 416x416. Proses augmentasi data menggunakan *framework Roboflow*. Dataset diupload ke website *roboflow* untuk augmentasi dan *pre-processing* data. Dataset hasil augmentasi kemudian didownload dalam bentuk file *.zip* kemudian di *extract* dalam sebuah *folder* pada *google drive*.

```
!curl -L "https://app.roboflow.com/ds/r7gxbFXmxO?key=P4RWGf2c2R"
>roboflow.zip; unzip roboflow.zip; rm roboflow.zip
```

Berikut adalah contoh hasil dari augmentasi dapat dilihat pada Gambar 4.13:



Gambar 4.13. Hasil augmentasi dataset *Affectnet* skenario 3

4.7.2.4. Skenario *Affectnet* + *Augmentasi* dengan rasio 70/30

Pada skenario 4 ini, dataset *Affectnet* yang digunakan untuk data latih dan data uji sama dengan skenario 2 dataset *affectnet* dan dilakukan proses augmentasi data seperti pada skenario ke 3 dataset *affectnet*. Sehingga data latih menjadi 44.100 dan untuk data uji tetap sama tidak ada penambahan data.

4.7.2.5. Skenario *Affectnet* + *MixUp* dengan rasio 80/20

Pada skenario 5 ini, dataset *Affectnet* yang digunakan untuk data latih dan data uji sama dengan skenario 1 pada dataset *affectnet* kemudian dilakukan proses *mixup*. *Mixup* dilakukan pada proses pelatihan data. Untuk memberi gambaran bagaimana proses *mixup* dilakukan dan bagaimana data yang dihasilkan maka berikut adalah contoh proses *mixup* yang dilakukan pada proses pelatihan.



Gambar 4.14. Contoh *mixup* pada dataset *Affectnet* skenario 5

4.7.2.6. Skenario *Affectnet* + *MixUp* dengan rasio 70/30

Pada skenario 6 ini, dataset *Affectnet* yang digunakan untuk data latih dan data uji sama dengan skenario 2 dataset *Affectnet* kemudian dilakukan proses *mixup*

sama halnya dengan skenario 5 dataset *Affectnet*. Hal yang berbeda adalah pada skenario ini akan menggunakan rasio 70/30.

4.7.2.7.Skenario *Affectnet* + *Augmentasi* + *MixUp* dengan rasio 80/20

Pada skenario 7 ini, dataset *Affectnet* yang digunakan untuk data latih dan data uji sama dengan skenario 1 dataset *Affectnet* kemudian data latih akan dilakukan proses augmentasi data *Flip*, *Crop*, *Rotate*, dan *Shear*. Dari data yang dihasil kemudian akan ditambahkan proses *mixup*. Dalam kata lain pada skenario ini mengkombinasikan proses augmentasi data dan proses *mixup*. Dari hasil data augmentasi akan menggabungkan 2 citra menjadi 1 citra pada setiap pelatihan dengan menggunakan proses *mixup*.

4.7.3. Skenario dataset *RAF-DB*

4.7.3.1.Skenario *RAF-DB* dengan rasio 80/20

Pada skenario 1 ini, dataset *RAF-DB* yang digunakan sebagai data latih yaitu sebanyak 12.271 citra kemudian untuk data uji yang digunakan sebanyak 3068 citra. Pada skenario ini tidak dilakukan proses augmentasi data ataupun proses *mixup*. Proses pelatihan hanya menggunakan data asli yang dibagi berdasarkan rasio 80:20. Pembagian ini sesuai dengan penelitian sebelumnya dan dari sumber dataset telah menyediakan pembagian subset untuk data latih dan data uji.

4.7.3.2.Skenario RAF-DB dengan rasio 70/30

Pada skenario 2 ini, dataset *RAF-DB* yang digunakan sebagai data latih yaitu sebanyak 10.738 citra kemudian untuk data uji yang digunakan sebanyak 4.601 citra. Pada skenario ini sama seperti skenario 1 dataset *RAF-DB*, data tidak dilakukan proses augmentasi data ataupun proses mixup. Berbeda dengan skenario 1, proses pelatihan hanya menggunakan data asli yang dibagi berdasarkan rasio 70:30. Penggunaan rasio ini agar penelitian ini lebih memiliki tantangan untuk mendapatkan tingkat akurasi yang baik dengan rasio perbandingan yang berbeda.

4.7.3.3.Skenario RAF-DB + Augmentasi dengan rasio 80/20

Pada skenario 3 ini, dataset *RAF-DB* yang digunakan sebagai data latih dan data uji sama seperti skenario 1 dataset *RAF-DB*. Akan tetapi pada skenario 3 dilakukan proses augmentasi data yaitu dengan menambahkan augmentasi seperti *Flip*, *Crop*, *Rotate*, dan *Shear*. Yang mana dari hasil augmentasi pada data training ada penambahan 3x lipat data latih sehingga data yang digunakan sebagai data pelatihan menjadi sekitar 36.813 sedangkan untuk data uji tidak ada penambahan data karena pada data pengujian diperlukan data asli bukan data hasil modifikasi. Gambar juga akan di *resize* ke ukuran 416x416. Proses augmentasi data menggunakan *framework Roboflow*. Dataset diupload ke website *roboflow* untuk augmentasi dan *pre-processing* data. Dataset hasil augmentasi kemudian didownload dalam bentuk file *.zip* kemudian di *extract* dalam sebuah *folder* pada *google drive*.


```
!curl -L "https://app.roboflow.com/ds/r7gxbFXmxO?key =P4RWGf2c2R"
>roboflow.zip; unzip roboflow.zip; rm roboflow.zip
```

Berikut adalah contoh hasil dari augmentasi dapat dilihat pada Gambar 4.15:



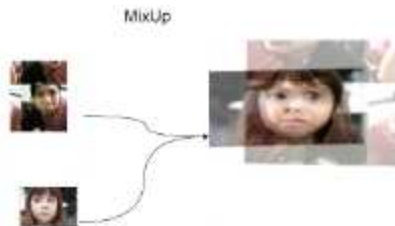
Gambar 4.15. Hasil augmentasi dataset *RAF-DB* skenario 3

4.7.3.4. Skenario *RAF-DB* + Augmentasi dengan rasio 70/30

Pada skenario 4 ini, dataset *RAF-DB* yang digunakan untuk data latih dan data uji sama dengan skenario 2 dataset *RAF-DB* dan dilakukan proses augmentasi data seperti pada skenario ke 3 dataset *RAF-DB*. Sehingga data latih menjadi 32.314 dan untuk data uji tetap sama tidak ada penambahan data.

4.7.3.5. Skenario *RAF-DB* + MixUp dengan rasio 80/20

Pada skenario 5 ini, dataset *RAF-DB* yang digunakan untuk data latih dan data uji sama dengan skenario 1 kemudian dilakukan proses *mixup*. *Mixup* dilakukan pada proses pelatihan data. Untuk memberi gambaran bagaimana proses *mixup* dilakukan dan bagaimana data yang dihasilkan maka berikut adalah contoh serta proses *mixup* yang dilakukan pada proses pelatihan dapat dilihat pada Gambar 4.16.



Gambar 4.16: Contoh *mixup* pada dataset *RAF-DB* skenario 5

4.7.3.6. Skenario *RAF-DB* + *MixUp* dengan rasio 70/30

Pada skenario 6 ini, dataset *RAF-DB* yang digunakan untuk data latih dan data uji sama dengan skenario 2 dataset *RAF-DB* kemudian dilakukan proses *mixup* sama halnya dengan skenario 5 dataset *RAF-DB*. Hal yang berbeda adalah pada skenario ini akan menggunakan rasio 70/30.

4.7.3.7. Skenario *RAF-DB* + *Augmentasi* + *MixUp* dengan rasio 80/20

Pada skenario 7 ini, dataset *RAF-DB* yang digunakan untuk data latih dan data uji sama dengan skenario 1 dataset *RAF-DB* kemudian data latih akan dilakukan proses augmentasi data *Flip*, *Crop*, *Rotate*, dan *Shear*. Dari data yang dihasil kemudian akan ditambahkan proses *mixup*. Dalam kata lain pada skenario ini mengkombinasikan proses augmentasi data dan proses *mixup*. Dari hasil data augmentasi akan menggabungkan 2 citra menjadi 1 citra pada setiap pelatihan dengan menggunakan proses *mixup*.

4.8. Proses Pelatihan Data

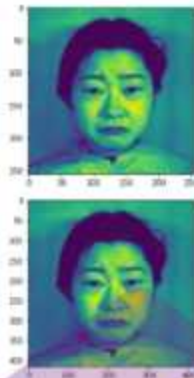
Proses pelatihan dan eksperimen dilakukan menggunakan *Jupyter Notebook* dengan menggunakan *framework darknet* yang bersifat *open source* pada *Google Colab* dengan spesifikasi CPU Intel(R) Xeon(R) CPU @ 2.00GHz, RAM 13 GB, GPU Tesla P100-PCIE-16GB, HARDDISK 125 GB.

Setelah melakukan tahap persiapan data atau *pre-processing* pada dataset, pembagian data latih dan data uji, kemudian akan dilakukan konfigurasi *hyperparameter* pada metode *YOLOv3*. Parameter yang digunakan yaitu *learning_rate 0.001, momentum 0.9, decay 0.0005, network 416x416, max_batches 14000, steps 11200 dan 12600*. Setelah konfigurasi parameter selesai maka sudah siap untuk dilakukan tahap selanjutnya yaitu proses pelatihan atau pembelajaran terhadap data. Pada proses pelatihan data sendiri terdapat beberapa tahapan yaitu klasifikasi dan deteksi. Selanjutnya akan dijelaskan beberapa tahapan deteksi pada proses pelatihan data.

4.9. Proses Deteksi

4.9.1. *Resize Citra*

Proses pertama yang dilakukan oleh metode *YOLOv3* adalah merubah ukuran citra sesuai dengan konfigurasi file **.cfg* yaitu *416*416*. Ilustrasi pada proses ini dapat dilihat pada Gambar 4.17.

Gambar 4.17. *Resize* Citra

4.9.2. *Konvolusi dan ReLU(Reactified Linier Unit)*

Kemudian setelah citra telah di-*resize* dengan ukuran yang sudah ditentukan, citra tersebut dikonversikan ke dalam bentuk nilai array untuk 3 *channel* citra yaitu *Red*, *Green*, dan *Blue*. Langkah selanjutnya yaitu melakukan konvolusi terhadap citra tersebut dengan mengalihkan dan menjumlahkannya pada sebuah kernel atau filter dengan matriks 3*3. Konvolusi dilakukan terhadap keseluruhan *channel* untuk mendapatkan suatu dalam mendeteksi atau mengklasifikasi objek. Untuk lebih jelasnya ilustrasi dari proses konvolusi akan ditunjukkan pada Gambar 4.18.

$$\begin{aligned}
 y &= (X1*k1) + (X2*k2) + (X3*k3) + (X7*k4) + (X8*k5) + (X9*k6) + (X13*k7) + (X14*k8) \\
 &+ (X15*k9) \\
 y &= (168*1) + (103*0) + (113*(-1)) + (166*1) + (97*0) + (87*(-1)) + (161*1) + (97*0) \\
 &+ (114*(-1)) \\
 y &= 168 + 0 + (-113) + 166 + 0 + (-87) + 161 + 0 + (-114) \\
 y &= 181
 \end{aligned}$$

Selanjutnya akan dilakukan pergeseran satu kotak ke kanan dan akan dilakukan perhitungan seperti yang di atas sampai keseluruhan piksel berhasil dikonvolusi.



Gambar 4.18. Proses Konvolusi

Setelah berhasil mendapatkan nilai dari proses konvolusi, selanjutnya dilakukan proses *ReLU*. Proses ini bekerja disaat ada nilai hasil perhitungan konvolusi yang bernilai minus dan akan dijakikan nilai (0). Berikut adalah proses *ReLU* pada Gambar 4.19.

Red							
181	-142	-214	-79	181	0	0	0
152	-102	-150	-80	152	0	0	0
94	-77	-102	-73	94	0	0	0
49	-32	-58	-7	49	0	0	0
Green							
186	-138	-204	-64	186	0	0	0
167	-94	-151	-86	167	0	0	0
116	-59	-121	-97	116	0	0	0
72	-5	-83	-43	72	0	0	0
Blue							
201	-149	-199	-75	201	0	0	0
169	-87	-123	-87	169	0	0	0
112	-39	-65	-81	112	0	0	0
71	17	-29	-29	71	17	0	0

Gambar 4.19. Proses aktivasi *ReLU*

4.9.3. Max Pooling

Proses ini dilakukan setelah mendapatkan nilai dari aktivasi *ReLU*. *Max Pooling* bertujuan untuk melakukan *down-sampling* terhadap citra agar mengurangi jumlah parameternya. *Max pooling* dilakukan dengan *filter* 2×2 dan *stride* 2, artinya setiap matriks akan selalu terbagi menjadi setengahnya. Berikut ini adalah tabel dari proses *max pooling* yang dapat dilihat pada Gambar 4.20.

Red							
180	10	15	10	10	180	10	10
150	10	15	10	10	150	10	10
80	10	15	10	10	80	10	10
10	10	15	10	10	10	10	10
Green							
160	10	15	10	10	160	10	10
117	10	15	10	10	117	10	10
114	10	15	10	10	114	10	10
72	10	15	10	10	72	10	10
Blue							
100	10	15	10	10	100	10	10
100	10	15	10	10	100	10	10
112	10	15	10	10	112	10	10
71	10	15	10	10	71	10	10

Gambar 4.20. Proses *Max Pooling*

4.9.4. Output 13×13 , 26×26 , 32×32

Proses konvolusi – *ReLU* – *Max Pooling* dilakukan hingga mendapatkan *output* jaringan dengan matriks 13×13 , 26×26 , dan 52×52 seperti pada Gambar 4.13.



Gambar 4.21. *Grid Cell* 13×13

4.9.5. *Bounding Box Confidence dan Prediksi Kelas Objek*

Setelah menghasilkan output *grid cell* dengan matriks 13×13 , *YOLO* akan membagi citra inputan ke dalam *grid cells* tersebut untuk dilakukan deteksi objek yang telah berisi 36 data untuk *bounding boxes* dan prediksi kelas. Untuk setiap *bounding box*, *grid cells* juga akan memprediksi kelas yang berfungsi sebagai klasifikasi untuk memberikan probabilitas pada semua kelas yang mungkin terdeteksi. Skor kepercayaan untuk *bounding box* dan prediksi kelas digabungkan menjadi satu hasil akhir yang memberi tahu kemungkinan bahwa *bounding box* ini berisi jenis objek tertentu.

4.9.6. *Non-Max Suppression*

Diakhir proses konvolusi akan dihasilkan *grid cells* untuk mendeteksi objek, dimana setiap *grid cell* akan memprediksi 5 *bounding box*. Jadi total *bounding box* yang diprediksi adalah *grid cells* yang akan digunakan $\times 5 =$ total *bounding box* yang ada yang artinya ada sebagian *bounding box* yang mempunyai skor kepercayaan rendah. Sehingga metode *YOLOv3* hanya menyimpan dan menampilkan kotak yang mempunyai skor prediksi di atas ambang batas (*threshold*) *IoU* dengan menggunakan persamaan 4.22. Berikut ini adalah ilustrasi *IoU*.



Gambar 4.22. *Intersection Over Union (IoU)*

Dari titik koordinat objek pada Gambar 4.22 dan *anchor box* berwarna hijau, merah, dan biru yang ditunjukkan pada Gambar 4.22 dapat dihitung *IoU* seperti berikut:

$$A_{n \text{ Gtruth}} (\text{Hijau}) = 48 (\text{height}) \times 42 (\text{weight})$$

$$A_{n \text{ Gtruth}} = 2016$$

$$A_{u \text{ truthP1}} (\text{merah}) = 49 \times 44$$

$$A_{u \text{ truthP1}} = 2516$$

$$\text{IOU1} = 2016/2516$$

$$\text{IOU1} = 0.94$$

$$A_{u \text{ truthP2}} (\text{biru}) = 32 \times 71$$

$$A_{u \text{ truthP2}} = 2272$$

$$\text{IOU2} = 2016/2272$$

$$\text{IOU2} = 0.88$$

Jadi dari hasil perhitungan *IoU* maka *IoU* yang akan disimpan dan ditampilkan dengan nilai *IoU* sebesar 0.94 atau 94%.

4.10. Hasil Klasifikasi

Setelah keseluruhan proses selesai, maka akan mendapatkan hasil klasifikasi sesuai dengan objek yang diinginkan, beserta dengan *bounding box* dan skor kepercayaannya.

4.11. Perhitungan Kinerja Model

Perhitungan kinerja model ini sebagai ukuran seberapa baik sebuah model dalam mendeteksi dan mengklasifikasi sebuah citra. Pada penelitian ini perhitungan kinerja model akan menggunakan metode *confusion matrix*. Metode ini akan mencari nilai persentase dari *precision* dan *recall* dari sebuah model arsitektur. Agar bisa lebih mengerti dengan perhitungan ini, peneliti akan memberikan sebuah contoh perhitungan *confusion matrix*. Pada contoh ini, penulis akan menggunakan jumlah kelas yang sesuai dengan kelas pada penelitian yaitu 7 kelas. Untuk lebih lengkapnya bisa dilihat pada Gambar 4.23.

Confusion Matrix

threshold confidence: 0.5
threshold: 0.5

	anger	disgust	neutral	happiness	surprise	fear	total
anger	0	0	0	0	0	0	0
disgust	0	0	0	0	0	0	0
neutral	0	0	0	0	0	0	0
happiness	0	0	0	32	1	0	33
surprise	0	0	0	0	0	0	0
fear	0	0	0	0	0	0	0
total	0	0	0	32	1	0	33

	precision	recall	avg 100
anger	0.00 %	0.00 %	0.00 %
disgust	0.00 %	0.00 %	0.00 %
neutral	100.00 %	0.00 %	0.00 %
happiness	100.00 %	100.00 %	100.00 %
surprise	0.00 %	0.00 %	0.00 %
fear	0.00 %	0.00 %	0.00 %
total	0.00 %	0.00 %	0.00 %

Gambar 4.23. Contoh hasil *Confusion Matrix*

Dalam penentuan nilai pada *confusion matrix* terdapat *True Positive (TP)* merupakan data yang diprediksi dengan tepat sebagai keluaran *positive* atau benar.

True Negative (TN) merupakan data yang diprediksi tepat apabila keluaran berupa negatif atau salah. *False Positive (FP)* merupakan data prediksi dengan kurang tepat apabila keluaran berupa positif atau benar. Yang terakhir adalah *False Negative (FN)* merupakan data yang diprediksi kurang tepat atau data tidak dapat terdeteksi. Berikut perhitungan dari *confusion matrix*:

1. *Precision*

Pada proses pencarian nilai presisi, rumus yang digunakan adalah

$$TP/(TP+FP)$$

$$(0.72 + 0.88 + 1 + 1 + 0.85 + 0.66 + 1) / 7 = 0.87 \text{ atau } 87\%$$

2. *Recall*

Pada proses pencarian nilai *recall*, rumus yang digunakan adalah

$$TP/(TP+FN)$$

$$(0.88 + 0.88 + 0.7 + 0.91 + 0.67 + 0.89 + 1) / 7 = 0.84 \text{ atau } 84\%$$

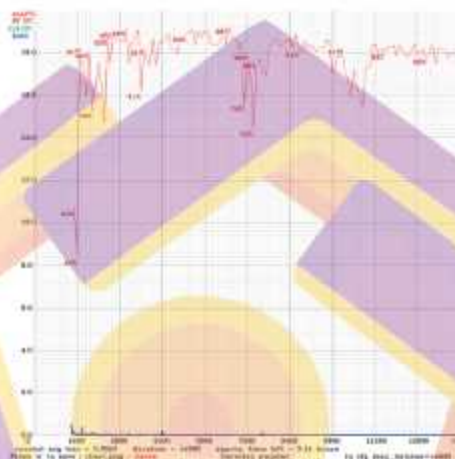
4.12. Hasil *Training* dan *Testing* data

Setelah selesai melakukan pelatihan dan didapatkan model yang baru. Selanjutnya, model yang telah dibangun akan diuji menggunakan data uji sesuai dengan skenario masing masing dataset.

4.12.1. JAFFE

4.12.1.1. Skenario 1

Skenario 1 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan dataset asli. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.24.



Gambar 4. 24. Grafik training dan validation loss skenario 1

Gambar 4.24 memperlihatkan *avg. loss* sebesar 0.0043. mAP yang didapatkan pada batch 1000 pertama adalah 42% kemudian mengalami kenaikan signifikan pada batch 1500 dengan nilai mAP 91%. Puncak mAP tertinggi didapatkan pada batch 3000. Hasil akhir rata rata mAP yang didapatkan adalah 87.3%. Model yang telah terbentuk dari proses *training* akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.25.

Confusion Matrix

Overall accuracy: 8.5

Overall F1: 0.5

		anger	disgust	fear	happiness	neutral	sadness	surprise	total
Groundtruth	anger	5	0	0	0	0	0	0	5
	disgust	0	7	0	0	0	0	0	7
	fear	0	0	9	0	0	0	0	9
	happiness	0	0	0	7	0	0	0	7
	neutral	0	0	0	0	0	0	0	0
	sadness	0	0	0	0	0	0	0	0
	surprise	0	0	0	0	0	0	2	2
	total	5	7	9	7	0	0	2	29
anger	precision	100.00 %	recall	65.25 %	avg F1	76.36 %			
disgust	precision	75.76 %	recall	100.00 %	avg F1	72.19 %			
fear	precision	75.00 %	recall	85.71 %	avg F1	76.36 %			
happiness	precision	100.00 %	recall	100.00 %	avg F1	75.26 %			
neutral	precision	100.00 %	recall	50.00 %	avg F1	77.29 %			
sadness	precision	00.00 %	recall	00.00 %	avg F1	01.02 %			
surprise	precision	33.33 %	recall	100.00 %	avg F1	69.09 %			

Gambar 4.25. Confusion Matrix skenario 1

Pengujian dilakukan menggunakan 42 data uji terhadap model skenario 1. Pada Gambar 4.25 didapatkan tingkat akurasi presisi tertinggi yaitu 100% pada tiga kelas yaitu *anger*, *happiness*, dan *neutral*. Dan tingkat presisi terendah yaitu pada kelas *surprise* yaitu 33.33%.

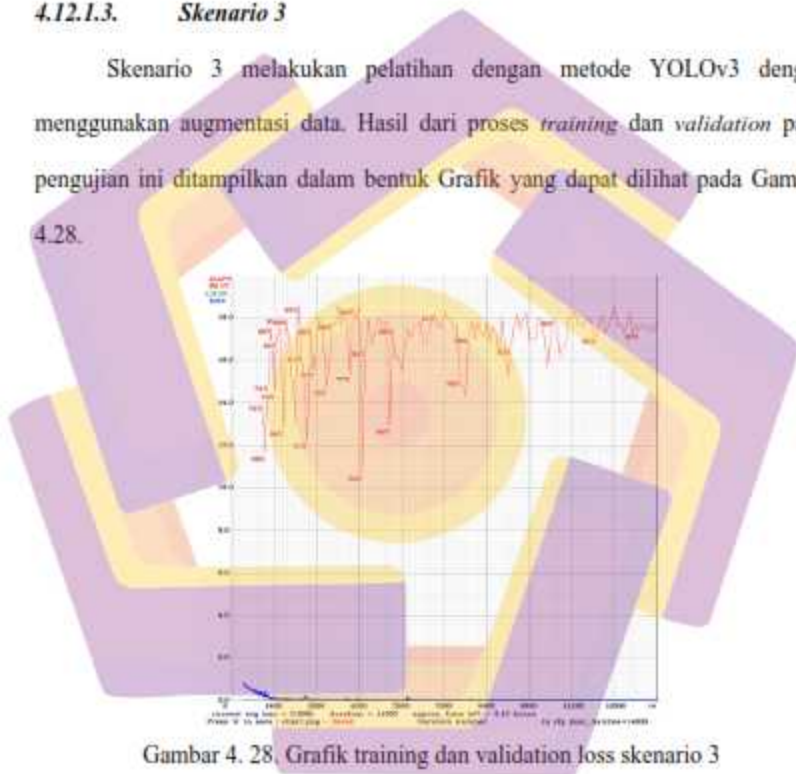
4.12.1.2. Skenario 2

Skenario 2 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.26.

Pengujian dilakukan menggunakan 63 data uji terhadap model skenario 2. Pada Gambar 4.27 didapatkan tingkat akurasi presisi tertinggi yaitu 100% pada tiga kelas yaitu *fear*, *happiness*, dan *surprise*. Dan tingkat akurasi presisi terendah yaitu pada kelas *sadness* yaitu 66.67%.

4.12.1.3. Skenario 3

Skenario 3 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.28.



Gambar 4. 28. Grafik training dan validation loss skenario 3

Gambar 4.28 memperlihatkan *avg. loss* sebesar 0.0084. mAP yang didapatkan pada batch 1000 pertama adalah 58% kemudian mengalami kenaikan bertahap pada batch 1400 dengan nilai mAP 74%. Puncak mAP tertinggi didapatkan pada sekitar batch 2000 yaitu 93%. Hasil akhir rata rata mAP yang

didapatkan adalah 89.1%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.29.

		anger	disgust	fear	happiness	neutral	sadness	surprise	total
Actual Class	anger	2	2	0	0	0	0	0	4
	disgust	0	2	0	0	0	0	0	2
	fear	0	0	2	0	0	2	0	4
	happiness	0	0	0	2	0	0	0	2
	neutral	0	0	0	0	2	0	0	2
	sadness	0	0	0	0	0	2	0	2
	surprise	0	0	0	0	0	0	2	2
total	2	2	2	2	2	2	2	12	
Precision	anger	100.00 %	100.00 %	0.00 %	0.00 %	0.00 %	0.00 %	0.00 %	100.00 %
	disgust	58.33 %	100.00 %	0.00 %	0.00 %	0.00 %	0.00 %	0.00 %	58.33 %
	fear	100.00 %	0.00 %	100.00 %	0.00 %	0.00 %	0.00 %	0.00 %	100.00 %
	happiness	100.00 %	0.00 %	0.00 %	100.00 %	0.00 %	0.00 %	0.00 %	100.00 %
	neutral	0.00 %	0.00 %	0.00 %	0.00 %	100.00 %	0.00 %	0.00 %	100.00 %
	sadness	0.00 %	0.00 %	0.00 %	0.00 %	0.00 %	100.00 %	0.00 %	100.00 %
	surprise	0.00 %	0.00 %	0.00 %	0.00 %	0.00 %	0.00 %	100.00 %	100.00 %

Gambar 4. 29. *Confusion Matrix* skenario 3

Pengujian dilakukan menggunakan 42 data uji terhadap model skenario 3. Pada Gambar 4.29 didapatkan tingkat akurasi presisi tertinggi yaitu 100% pada tiga kelas yaitu *anger*, *fear*, dan *happiness*. Dan tingkat akurasi presisi terendah yaitu pada kelas *disgust* yaitu 58.33%.

4.12.1.4. Skenario 4

Skenario 4 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.30.



Gambar 4.30. Grafik training dan validation loss skenario 4

Gambar 4.30 memperlihatkan *avg. loss* sebesar 0.0075. mAP yang didapatkan pada batch 1000 pertama adalah 49% kemudian mengalami kenaikan bertahap pada batch 1400 dengan nilai mAP 76%. Puncak mAP tertinggi didapatkan pada sekitar batch 7700 yaitu 91%. Hasil akhir rata rata mAP yang didapatkan adalah 82.0%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.31.

```

Confusion Matrix
-----
Strategi partisi: 0.5
Metode: 0.5
-----

```

		actual target	total
		0 1	0 1
Strategi partisi	anger	7 2	9
	blangkon	0 0	0
	karang	0 0	0
	kepalak	0 0	0
	kepulauan	0 0	0
Metode	anger	7 2	9
	blangkon	0 0	0
	karang	0 0	0
	kepalak	0 0	0
	kepulauan	0 0	0
		total	9 2
		total	9 2

```

-----

```

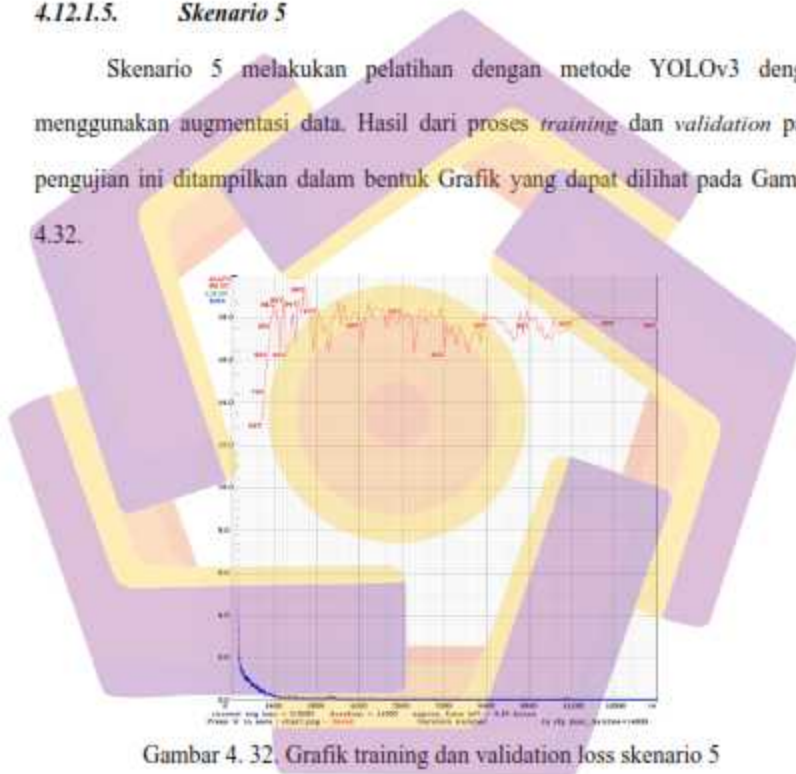
		avg acc	avg f1
anger	avg acc: 87.78 %	avg f1: 77.76 %	avg f0.5: 86.72 %
blangkon	avg acc: 76.46 %	avg f1: 77.78 %	avg f0.5: 64.52 %
karang	avg acc: 100.00 %	avg f1: 79.00 %	avg f0.5: 68.67 %
kepalak	avg acc: 100.00 %	avg f1: 88.91 %	avg f0.5: 73.34 %
kepulauan	avg acc: 62.22 %	avg f1: 62.50 %	avg f0.5: 58.36 %
total	avg acc: 86.67 %	avg f1: 80.85 %	avg f0.5: 62.58 %
total	avg acc: 86.67 %	avg f1: 80.85 %	avg f0.5: 62.58 %

Gambar 4.31. Confusion Matrix skenario 4

Pengujian dilakukan menggunakan 63 data uji terhadap model skenario 4. Pada Gambar 4.31 didapatkan tingkat akurasi presisi tertinggi yaitu 100% pada dua kelas yaitu *fear*, dan *happiness*. Dan tingkat akurasi presisi terendah yaitu pada kelas *sadness* yaitu 47.06%.

4.12.1.5. Skenario 5

Skenario 5 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.32.



Gambar 4. 32. Grafik training dan validation loss skenario 5

Gambar 4.32 memperlihatkan *avg. loss* sebesar 0.0283. mAP yang didapatkan pada batch 1000 pertama adalah 66% kemudian mengalami kenaikan bertahap pada batch 1400 dengan nilai mAP 95%. Puncak mAP tertinggi didapatkan pada sekitar batch 2500 yaitu 98%. Hasil akhir rata rata mAP yang

didapatkan adalah 89.3%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.33.

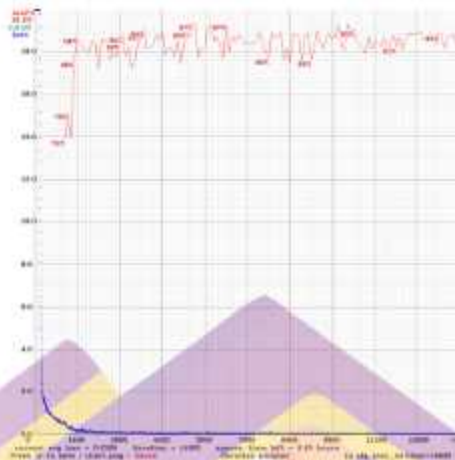
		anger	disgust	fear	happiness	neutral	sadness	surprise	total
class actual	anger	0	0	0	0	0	0	0	0
	disgust	0	2	0	0	0	0	0	2
	fear	0	0	0	0	0	0	1	1
	happiness	0	0	0	2	0	0	0	2
	neutral	0	0	0	0	7	0	1	8
	sadness	0	0	0	1	0	1	0	2
	surprise	0	0	0	0	0	0	0	0
	total	0	2	0	2	7	1	0	12
precision									
anger	precision	100.00 %	recall	0.00 %	avg. f1	0.00 %			
disgust	precision	75.00 %	recall	100.00 %	avg. f1	87.50 %			
fear	precision	100.00 %	recall	0.00 %	avg. f1	0.00 %			
happiness	precision	0.00 %	recall	100.00 %	avg. f1	0.00 %			
neutral	precision	0.00 %	recall	0.00 %	avg. f1	0.00 %			
sadness	precision	100.00 %	recall	50.00 %	avg. f1	75.00 %			
surprise	precision	0.00 %	recall	100.00 %	avg. f1	0.00 %			

Gambar 4. 33. *Confusion Matrix* skenario 5

Pengujian dilakukan menggunakan 42 data uji terhadap model skenario 5. Pada Gambar 4.33 didapatkan tingkat akurasi presisi tertinggi yaitu 100% pada empat kelas yaitu *anger*, *neutral*, *fear*, dan *sadness*. Dan tingkat akurasi presisi terendah yaitu pada kelas *surprise* yaitu 33.33%.

4.12.1.6. Skenario 6

Skenario 6 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.34.



Gambar 4.34. Grafik training dan validation loss skenario 6

Gambar 4.34 memperlihatkan *avg. loss* sebesar 0.0326, mAP yang didapatkan pada batch 1000 pertama adalah 70% kemudian mengalami kenaikan bertahap pada batch 1400 dengan nilai mAP 94%. Puncak mAP tertinggi didapatkan pada sekitar batch 5000 yaitu 97%. Hasil akhir rata rata mAP yang didapatkan adalah 91.0%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.35.

```

Confusion Matrix
-----
Matrix confusion: 6 x 6
Matrix size: 6 x 6
-----

```

	anger	disgust	neutral (background)	surprise	happiness	sadness	surprise	miss	total
anger	4	0	0	0	0	0	0	2	4
disgust	0	0	0	0	0	0	0	0	0
neutral	0	0	0	0	0	0	0	2	2
surprise	0	0	0	11	0	0	0	0	11
happiness	0	0	0	0	0	0	0	0	0
sadness	0	0	0	0	0	0	0	0	0
surprise	0	0	0	0	0	0	0	0	0
miss	0	0	0	0	0	0	0	2	2
total	4	0	0	11	0	0	0	4	19

```

-----

```

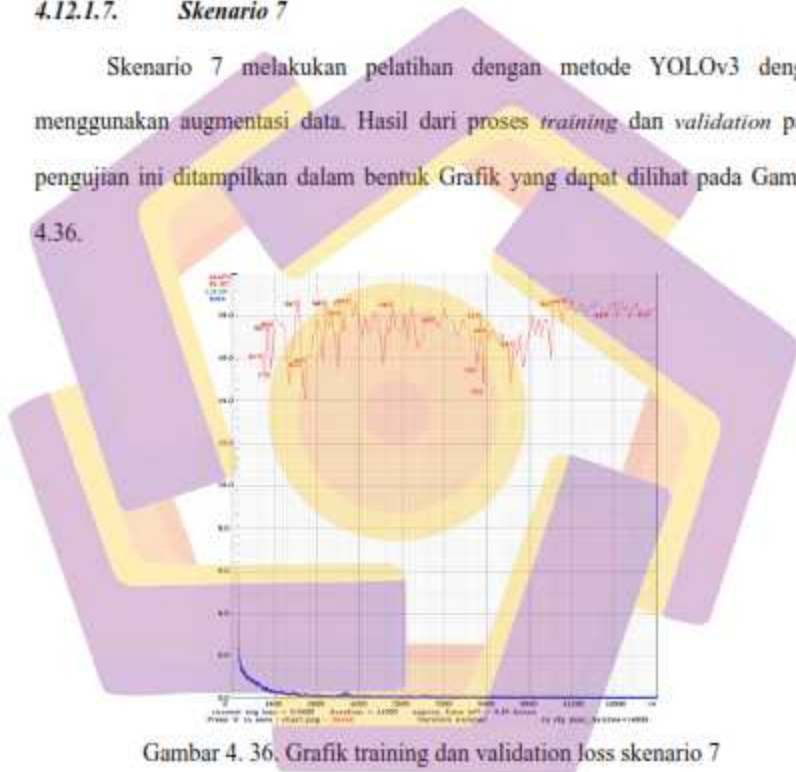
anger	precision	100.00 %	recall	44.44 %	avg. f1	66.66 %
disgust	precision	75.00 %	recall	100.00 %	avg. f1	85.00 %
neutral	precision	100.00 %	recall	100.00 %	avg. f1	95.00 %
surprise	precision	100.00 %	recall	100.00 %	avg. f1	95.00 %
happiness	precision	100.00 %	recall	100.00 %	avg. f1	95.00 %
sadness	precision	100.00 %	recall	100.00 %	avg. f1	95.00 %
surprise	precision	75.00 %	recall	100.00 %	avg. f1	87.50 %

Gambar 4.35. *Confusion Matrix* skenario 6

Pengujian dilakukan menggunakan 63 data uji terhadap model skenario 6. Pada Gambar 4.35 didapatkan tingkat akurasi presisi tertinggi yaitu 100% pada lima kelas yaitu *anger*, *neutral*, *fear*, *happiness*, dan *sadness*. Dan tingkat akurasi presisi terrendah yaitu pada kelas *disgust*, *surprise* yaitu 75%.

4.12.1.7. Skenario 7

Skenario 7 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.36.



Gambar 4. 36. Grafik training dan validation loss skenario 7

Gambar 4.36 memperlihatkan *avg. loss* sebesar 0.0326. mAP yang didapatkan pada batch 1000 pertama adalah 81% kemudian mengalami kenaikan bertahap pada batch 1400 dengan nilai mAP 89%. Puncak mAP tertinggi didapatkan pada sekitar batch 3000 yaitu 94%. Hasil akhir rata rata mAP yang

didapatkan adalah 91.4%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.37.

		anger	disgust	fear	happiness	neutral	sadness	surprise	none	total
Actual	anger	0	2	0	0	0	0	0	0	2
	disgust	0	2	0	0	0	2	0	0	2
	fear	0	0	0	0	0	0	0	0	0
	happiness	0	0	0	7	0	0	0	0	7
	neutral	0	0	0	0	7	0	0	0	7
	sadness	0	0	0	0	0	6	0	0	6
	surprise	0	0	0	0	0	0	6	0	6
	none	0	0	0	0	0	0	1	1	1
	total	0	2	0	7	7	6	6	1	23
Precision		0%	100%	0%	100%	100%	100%	100%	100%	
Recall		0%	100%	0%	100%	100%	100%	100%	100%	
F1 Score		0%	100%	0%	100%	100%	100%	100%	100%	
Accuracy		0%	100%	0%	100%	100%	100%	100%	100%	
Loss		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	

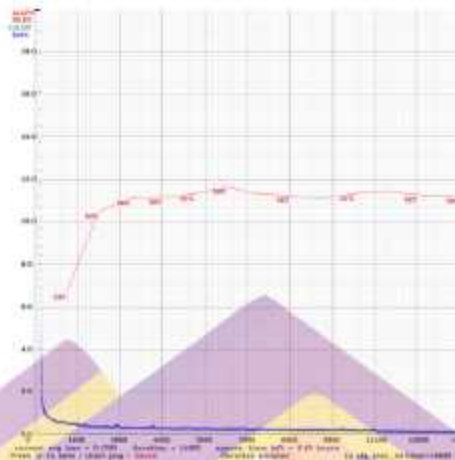
Gambar 4.37. *Confusion Matrix* skenario 7

Pengujian dilakukan menggunakan 42 data uji terhadap model skenario 7. Pada Gambar 4.37 didapatkan tingkat akurasi presisi tertinggi yaitu 100% pada empat kelas yaitu *anger*, *fear*, *happiness*, dan *neutral*. Dan tingkat akurasi presisi terendah yaitu pada kelas *surprise* yaitu 60%.

4.12.2. Affectnet

4.12.2.1. Skenario 1

Skenario 1 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan dataset asli dengan rasio 80:20. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.38.



Gambar 4.38. Grafik training dan validation loss skenario 1

Gambar 4.38 memperlihatkan *avg. loss* sebesar 0.1783. mAP yang didapatkan pada batch 1000 pertama adalah 33% kemudian mengalami kenaikan pada batch 2100 dengan nilai mAP 52%. Puncak mAP tertinggi didapatkan pada batch 6300 dengan mAP 58%. Hasil akhir rata rata mAP yang didapatkan adalah 56.0%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.39.

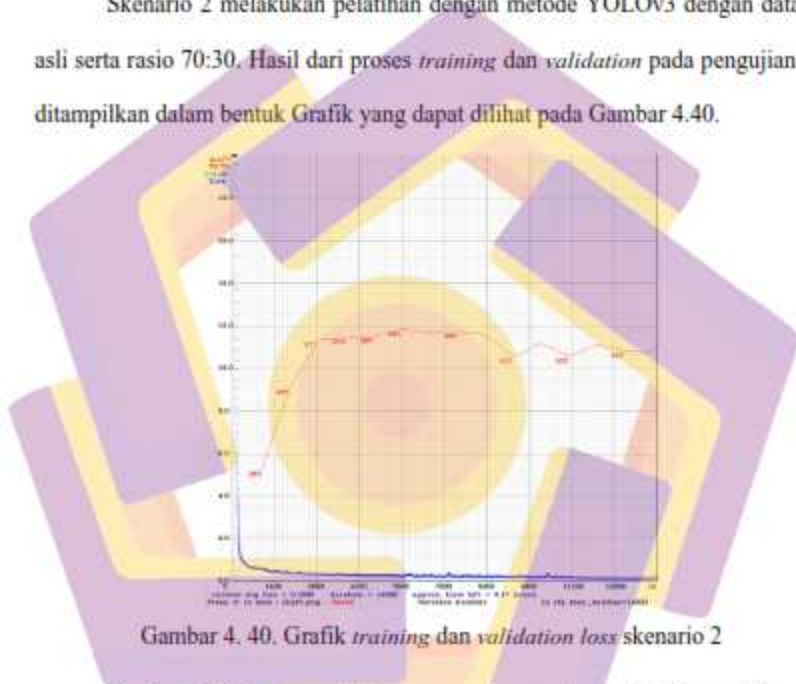
Confusion Matrix		Dititik (actual) : 8 %		Dititik (pred) : 8 %					
		netral	hebat	jelek	disprekasi	baik	disputasi	amargi	remed
groundtruth	netral	256	20	40	20	0	10	19	351
	hebat	54	206	9	22	0	10	2	248
	jelek	30	1	257	14	0	22	11	242
	disprekasi	0	0	0	252	0	4	0	248
	baik	2	0	0	76	0	0	0	183
prediction	netral	2	22	39	11	30	267	21	213
	hebat	29	4	55	7	34	19	167	303
	jelek	0	0	0	0	0	0	0	0
	disprekasi	0	0	0	0	0	0	0	0
	baik	248	403	488	403	206	403	212	
netral	precision:	81.88 %	recall:	25.38 %	avg 100:	81.88 %			
hebat	precision:	81.41 %	recall:	85.38 %	avg 100:	75.87 %			
jelek	precision:	23.24 %	recall:	41.95 %	avg 100:	90.87 %			
disprekasi	precision:	81.77 %	recall:	81.38 %	avg 100:	90.87 %			
baik	precision:	84.14 %	recall:	42.33 %	avg 100:	75.88 %			
disputasi	precision:	85.88 %	recall:	44.52 %	avg 100:	75.28 %			
amargi	precision:	71.98 %	recall:	27.83 %	avg 100:	75.38 %			

Gambar 4.39. Confusion Matrix skenario 1

Pengujian dilakukan menggunakan 4200 data uji terhadap model skenario 1. Pada Gambar 4.39 didapatkan tingkat akurasi presisi tertinggi yaitu 71.98% pada kelas *happy*. Dan tingkat presisi terendah yaitu pada kelas *sad* yaitu 53.54%.

4.12.2.2. Skenario 2

Skenario 2 melakukan pelatihan dengan metode YOLOv3 dengan dataset asli serta rasio 70:30. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.40.



Gambar 4. 40. Grafik *training* dan *validation loss* skenario 2

Gambar 4.40 memperlihatkan *avg. loss* sebesar 0.1666. mAP yang didapatkan pada batch 1000 pertama adalah 26% kemudian mengalami kenaikan pada batch 1700 dengan nilai mAP 45%. Puncak mAP tertinggi didapatkan pada batch 5600 yaitu 59%. Hasil akhir rata rata mAP yang didapatkan adalah 55.7%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan

data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.41.

```

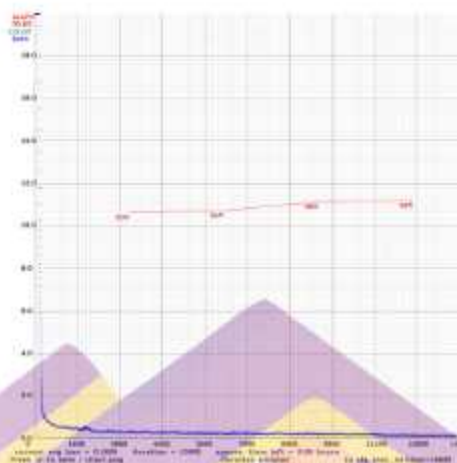
Confusion Matrix
-----
threshold confidence: 0.5
threshold: 0.5
-----
      | netral | happy | sad | surpris | fear | disgust | anger | total |
-----|-----|-----|-----|-----|-----|-----|-----|-----|
netral| 470   | 21   | 49  | 23      | 13   | 14     | 12    | 597
happy | 75    | 639  | 12  | 15      | 8    | 16     | 6     | 761
sad    | 623   | 7    | 940 | 12      | 39   | 49     | 14    | 1471
surpris| 62    | 34   | 18  | 223     | 137  | 7      | 0     | 444
fear   | 17    | 6    | 49  | 189     | 442  | 11     | 23    | 686
disgust| 36    | 31   | 66  | 11      | 36   | 422    | 13    | 585
anger  | 106   | 6    | 19  | 9        | 21   | 103    | 14    | 278
total  | 943   | 711  | 109 | 398     | 688  | 651    | 127   |
-----
netral| precision: 49.84% | recall: 52.46% | avg.200: 51.43%
happy | precision: 85.09% | recall: 78.53% | avg.200: 76.71%
sad    | precision: 63.37% | recall: 38.38% | avg.200: 70.97%
surpris| precision: 63.25% | recall: 35.84% | avg.200: 69.16%
fear   | precision: 69.55% | recall: 49.12% | avg.200: 70.11%
disgust| precision: 63.41% | recall: 49.34% | avg.200: 75.83%
anger  | precision: 63.36% | recall: 17.11% | avg.200: 62.24%
  
```

Gambar 4.41. *Confusion Matrix* skenario 2

Pengujian dilakukan menggunakan 6300 data uji terhadap model skenario 2. Pada Gambar 4.41 didapatkan tingkat akurasi presisi tertinggi yaitu 85.09% pada kelas *happy*. Dan tingkat akurasi presisi terendah yaitu pada kelas *netral* yaitu 49.84%.

4.12.2.3. Skenario 3

Skenario 3 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data dan rasio 80:20. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.42.



Gambar 4.42. Grafik training dan validation loss skenario 3

Gambar 4.42 memperlihatkan *avg. loss* sebesar 0.1626, mAP yang didapatkan pada batch 2800 pertama adalah 53% kemudian mengalami kenaikan pada batch 5800 dengan nilai mAP 54%. Puncak mAP tertinggi didapatkan pada sekitar batch 9100 yaitu 56%. Hasil akhir rata rata mAP yang didapatkan adalah 55.9%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.43.

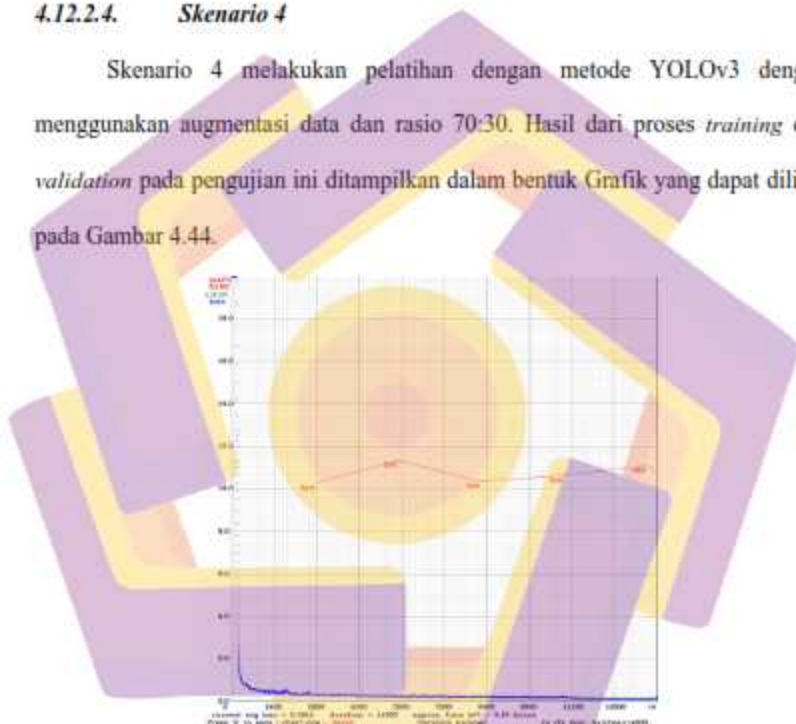
Confusion Matrix									
Class: 0=0, 1=1, 2=2, 3=3, 4=4, 5=5, 6=6, 7=7, 8=8, 9=9									
	netral	hempel	hempel (suspicious)	hempel	disruptif	anger	senam	teror	
netral	276	0	30	16	4	7	25	260	997
hempel	11	402	4	12	4	0	4	4	437
hempel (suspicious)	70	1	230	12	0	21	38	221	551
hempel	17	17	10	252	31	0	7	227	559
disruptif	7	4	4	93	206	0	21	11	231
anger	23	23	27	4	10	249	44	221	598
senam	43	2	28	5	12	52	282	231	608
teror	7	0	14	6	0	0	0	27	54
total	426	444	418	426	145	807	322	522	
netral	precision: 53.08 %	recall: 38.36 %	avg 200: 72.33 %						
hempel	precision: 98.38 %	recall: 44.26 %	avg 200: 76.67 %						
hempel (suspicious)	precision: 56.34 %	recall: 38.69 %	avg 200: 75.28 %						
hempel	precision: 59.29 %	recall: 42.67 %	avg 200: 75.28 %						
disruptif	precision: 61.38 %	recall: 24.33 %	avg 200: 75.22 %						
anger	precision: 66.49 %	recall: 46.73 %	avg 200: 76.58 %						
senam	precision: 42.93 %	recall: 23.67 %	avg 200: 76.34 %						

Gambar 4.43. Confusion Matrix skenario 3

Pengujian dilakukan menggunakan 4200 data uji terhadap model skenario 3. Pada Gambar 4.43 didapatkan tingkat akurasi presisi tertinggi yaitu 90.55% pada kelas *happy*. Dan tingkat akurasi presisi terendah yaitu pada kelas *netral* yaitu 53.88%.

4.12.2.4. Skenario 4

Skenario 4 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data dan rasio 70:30. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.44.



Gambar 4. 44. Grafik training dan validation loss skenario 4

Gambar 4.44 memperlihatkan *avg. loss* sebesar 0.1614. mAP yang didapatkan pada batch 2800 pertama adalah 51% kemudian mengalami kenaikan bertahap pada batch 5500 dengan nilai mAP 57%. Puncak mAP tertinggi didapatkan pada sekitar batch 5500 yaitu 57%. Hasil akhir rata rata mAP yang

didapatkan adalah 53.9%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.45.

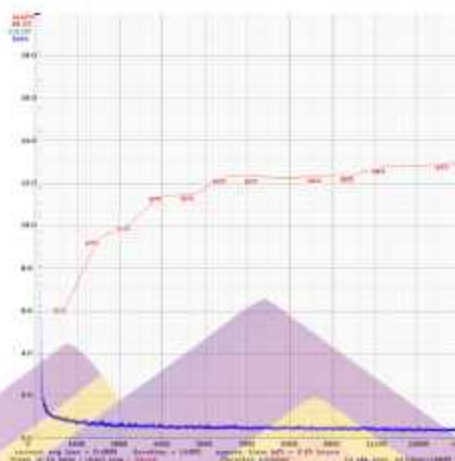
		neutral	happy	sad	surprised	fear	disgust	anger	total
Actual	neutral	280	18	40	22	2	11	40	371
	happy	67	615	13	14	7	10	4	716
	sad	112	7	862	10	10	10	10	951
	surprised	70	20	11	262	14	7	5	489
	fear	21	5	40	120	210	29	17	372
	disgust	10	10	61	10	21	80	40	222
	anger	75	4	12	7	14	62	311	481
total	718	606	632	309	309	305	340		
		Precision							
neutral	precision	80.36 %	neutral	41.82 %	sad	200 %	94.30 %		
happy	precision	89.26 %	happy	91.30 %	surprised	200 %	77.40 %		
sad	precision	32.29 %	sad	80.36 %	fear	200 %	70.36 %		
surprised	precision	20.30 %	surprised	71.29 %	disgust	200 %	80.29 %		
fear	precision	67.61 %	disgust	80.36 %	anger	200 %	74.29 %		
disgust	precision	41.82 %	anger	80.36 %	total	200 %	77.31 %		
anger	precision	61.30 %	total	20.70 %			70.30 %		

Gambar 4. 45. *Confusion Matrix* skenario 4

Pengujian dilakukan menggunakan 6300 data uji terhadap model skenario 4. Pada Gambar 4.45 didapatkan tingkat akurasi presisi tertinggi yaitu 89.65% pada kelas *happy*. Dan tingkat akurasi presisi terendah yaitu pada kelas *neutral* yaitu 50,86%.

4.12.2.5. *Skenario 5*

Skenario 5 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan *mixup* dan rasio 80:20. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.46.



Gambar 4.46. Grafik training dan validation loss skenario 5

Gambar 4.46 memperlihatkan *avg. loss* sebesar 0.4898, mAP yang didapatkan pada batch 1000 pertama adalah 31% kemudian mengalami kenaikan bertahap pada batch 2100 dengan nilai mAP 47%. Puncak mAP tertinggi didapatkan pada sekitar batch 14000 yaitu 65%. Hasil akhir rata rata mAP yang didapatkan adalah 64.2%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.47.

		netral	kecewa	sed	terkejut	marah	kegugupan	senang	total
netral	40	21	0	0	7	11	44	111	
kecewa	24	404	7	14	2	0	3	460	
sed	40	0	107	14	10	14	139	197	
terkejut	20	20	1	100	11	4	121	159	
marah	7	20	42	147	200	12	27	497	
kegugupan	0	16	34	11	21	100	14	180	
senang	40	0	32	13	25	56	111	164	
total	74	12	13	24	34	13	13	143	
total	300	374	491	371	489	620	513		

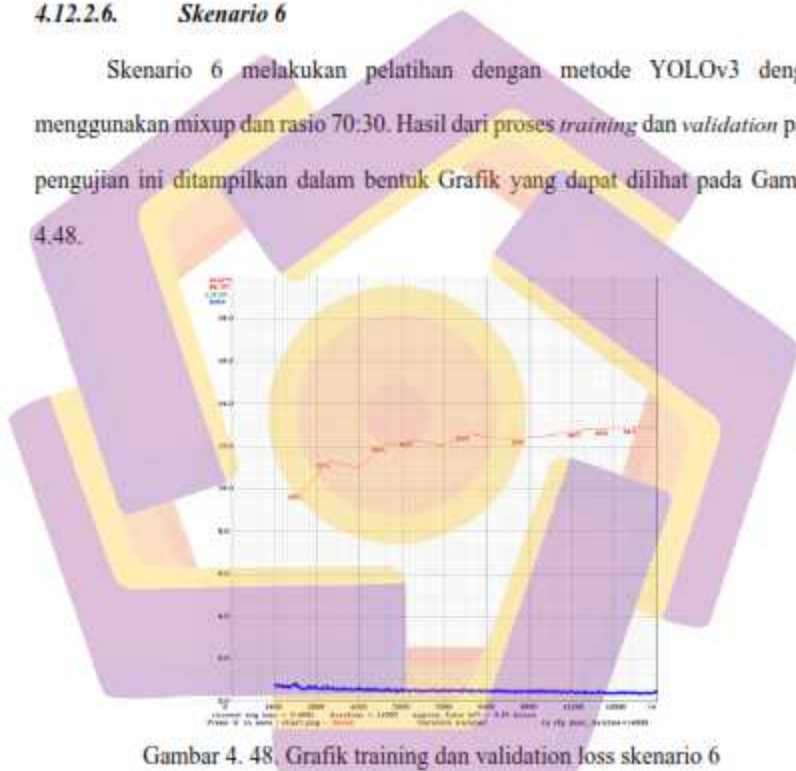
netral	precision: 50.34 %	recall: 51.69 %	avg f1: 52.57 %
kecewa	precision: 21.32 %	recall: 75.85 %	avg f1: 52.26 %
sed	precision: 48.24 %	recall: 43.75 %	avg f1: 46.00 %
terkejut	precision: 42.38 %	recall: 34.43 %	avg f1: 31.36 %
marah	precision: 67.21 %	recall: 49.67 %	avg f1: 58.13 %
kegugupan	precision: 48.38 %	recall: 21.75 %	avg f1: 32.22 %
senang	precision: 48.51 %	recall: 51.83 %	avg f1: 50.18 %

Gambar 4.47. *Confusion Matrix* skenario 5

Pengujian dilakukan menggunakan 4200 data uji terhadap model skenario 5. Pada Gambar 4.47 didapatkan tingkat akurasi presisi tertinggi yaitu 84.32% pada kelas *happy*. Dan tingkat akurasi presisi terendah yaitu pada kelas *surprise* yaitu 56.34%.

4.12.2.6. Skenario 6

Skenario 6 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan mixup dan rasio 70:30. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.48.



Gambar 4. 48. Grafik training dan validation loss skenario 6

Gambar 4.48 memperlihatkan *avg. loss* sebesar 0.0326. mAP yang didapatkan pada batch 2100 pertama adalah 49% kemudian mengalami kenaikan bertahap pada batch 3100 dengan nilai mAP 57%. Puncak mAP tertinggi didapatkan pada sekitar batch 11200 yaitu 64%. Hasil akhir rata rata mAP yang

didapatkan adalah 64.3%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.49.

Confusion Matrix

Overall classification: 0.8
through: 0.8

	neutral	happy	sad	surprised	fear	angry	none	total
Actual	126	25	64	35	8	18	34	207
Predicted	76	249	25	41	4	12	5	44
neutral	119	8	411	19	26	48	164	807
happy	88	29	29	121	202	4	14	756
sad	17	4	55	144	450	29	23	722
surprised	28	21	65	75	42	440	101	672
fear	111	13	44	26	14	79	164	399
angry	23	9	14	23	14	11	14	98
none	373	789	734	304	76	622	327	

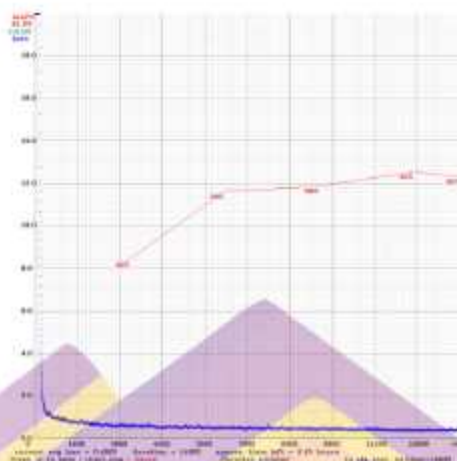
neutral	prec: 92.03%	recall: 57.45%	f1: 73.01%
happy	prec: 94.80%	recall: 76.95%	f1: 85.36%
sad	prec: 81.64%	recall: 39.28%	f1: 54.56%
surprised	prec: 56.25%	recall: 52.95%	f1: 54.59%
fear	prec: 64.27%	recall: 54.09%	f1: 59.13%
angry	prec: 76.56%	recall: 48.54%	f1: 61.42%
none	prec: 62.69%	recall: 38.89%	f1: 50.15%

Gambar 4. 49. Confusion Matrix skenario 6

Pengujian dilakukan menggunakan 6300 data uji terhadap model skenario 6. Pada Gambar 4.49 didapatkan tingkat akurasi presisi tertinggi yaitu 86,23% pada kelas *happy*. Dan tingkat akurasi presisi terendah yaitu pada kelas *neutral* yaitu 52,93%.

4.12.2.7. Skenario 7

Skenario 7 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data ditambah mixup dengan rasio 80:20. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.50.



Gambar 4.50. Grafik training dan validation loss skenario 7

Gambar 4.50 memperlihatkan *avg. loss* sebesar 0.4922. mAP yang didapatkan pada batch 3100 pertama adalah 42% kemudian mengalami kenaikan bertahap pada batch 6200 dengan nilai mAP 58%. Puncak mAP tertinggi didapatkan pada sekitar batch 12000 yaitu 63%. Hasil akhir rata rata mAP yang didapatkan adalah 61.5%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.51.

Confusion Matrix		Actual \ Predicted		Actual		Predicted		Total	
		netrac	netwp	netwp	netrac	netrac	netwp	netrac	netwp
netrac	netrac	323	18	48	17	4	5	29	337
	netwp	40	638	14	66	4	5	7	696
	total	363	656	62	83	8	10	36	733
	accuracy	89	73	14	77	57	16	7	140
netwp	netrac	10	7	62	183	274	19	14	126
	netwp	20	19	53	14	23	262	14	161
	total	30	26	115	197	397	281	28	398
	accuracy	66	7	42	14	16	38	237	183
total	33	5	10	12	25	6	13	19	
total	696	662	496	529	428	245	714	714	

netrac	precision	52.28 %	recall	54.77 %	avg 100%	76.18 %			
netwp	precision	66.74 %	recall	75.54 %	avg 100%	76.23 %			
total	precision	58.25 %	recall	46.73 %	avg 100%	76.20 %			
netrac	precision	61.88 %	recall	51.58 %	avg 100%	54.77 %			
netwp	precision	66.83 %	recall	45.87 %	avg 100%	75.49 %			
total	precision	71.34 %	recall	43.54 %	avg 100%	75.21 %			
netwp	precision	64.37 %	recall	39.58 %	avg 100%	75.14 %			

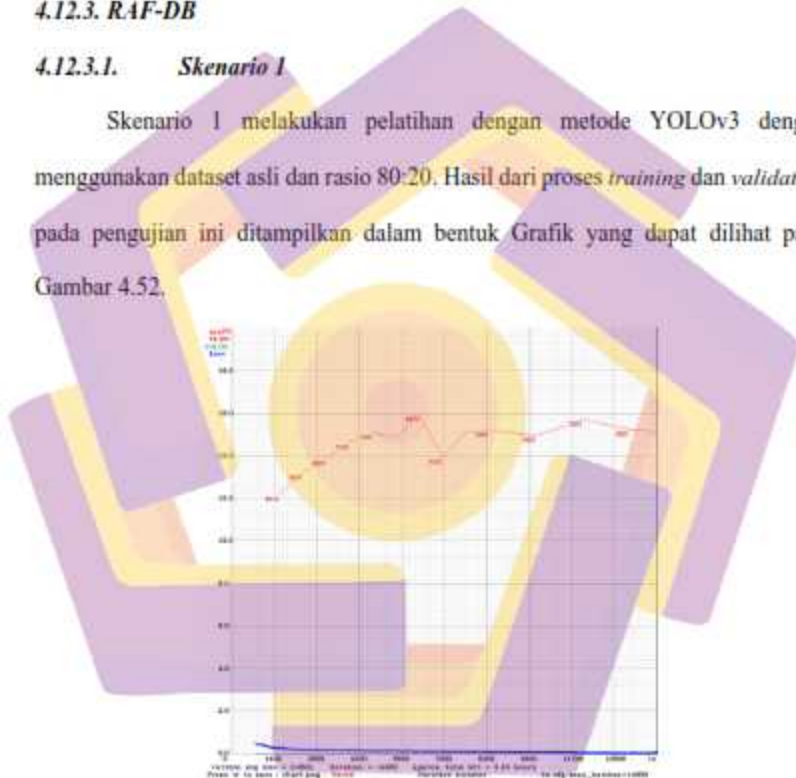
Gambar 4.51. Confusion Matrix skenario 7

Pengujian dilakukan menggunakan 4200 data uji terhadap model skenario 7. Pada Gambar 4.51 didapatkan tingkat akurasi presisi tertinggi yaitu 86.74% pada kelas *happy*. Dan tingkat akurasi presisi terendah yaitu pada kelas *neutral* yaitu 52.24%.

4.12.3. RAF-DB

4.12.3.1. Skenario 1

Skenario 1 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan dataset asli dan rasio 80:20. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.52.



Gambar 4. 52. Grafik training dan validation loss skenario 1

Gambar 4.52 memperlihatkan *avg. loss* sebesar 0.0631. mAP yang didapatkan pada batch 1400 pertama adalah 61% kemudian mengalami kenaikan pada batch 2100 dengan nilai mAP 66%. Puncak mAP tertinggi didapatkan pada

batch 6000 dengan nilai mAP 80%. Hasil akhir rata rata mAP yang didapatkan adalah 74.2%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.53.

		surprise	fear	disgust	happiness	sadness	anger	neutral	total
Groundtruth	surprise	132	2	2	1	3	0	1	140
	fear	0	23	1	1	5	1	2	33
	disgust	0	1	52	4	1	2	7	67
	happiness	0	0	22	124	15	0	0	161
	sadness	0	0	0	0	206	2	0	208
	anger	0	0	0	0	0	85	1	86
	neutral	22	0	0	21	29	10	0	82
	total	154	23	75	164	250	103	10	619
Precision	surprise	90.88%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	90.88%
	fear	26.18%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	26.18%
	disgust	43.33%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	43.33%
	happiness	90.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	90.00%
	sadness	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%
	anger	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%
	neutral	83.33%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%	83.33%
	total	83.33%	26.18%	43.33%	90.00%	100.00%	100.00%	83.33%	83.33%

Gambar 4.53. *Confusion Matrix* skenario 1

Pengujian dilakukan menggunakan 3068 data uji terhadap model skenario 1. Pada Gambar 4.53 didapatkan tingkat akurasi presisi tertinggi yaitu 90.88% pada kelas *happiness*. Dan tingkat presisi terendah yaitu pada kelas *disgust* yaitu 43.33%.

4.12.3.2. Skenario 2

Skenario 2 melakukan pelatihan dengan metode YOLOv3 dengan dataset asli dan rasio 70:30. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.54.



Gambar 4.54. Grafik training dan validation loss skenario 2

Gambar 4.54 memperlihatkan *avg. loss* sebesar 0.0571. mAP yang didapatkan pada batch 1000 pertama adalah 51% kemudian mengalami kenaikan bertahap pada batch 1600 dengan nilai mAP 61%. Puncak mAP tertinggi didapatkan pada batch 11200 yaitu 76%. Hasil akhir rata rata mAP yang didapatkan adalah 74.1%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.55.

```

Confusion Matrix
train confidence: 0.5
thresh: 100 = 0.5

      | [surprise] | [fear] | [disgust] | [happiness] | [sadness] | [anger] | [neutral] | [none] | total |
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
Actual\
| surprise| 522      | 0       | 2       | 1       | 1       | 2       | 2       | 298   | 828
| fear     | 0       | 12      | 2       | 1       | 2       | 0       | 0       | 17    | 32
| disgust  | 0       | 0       | 60      | 7       | 5       | 5       | 10      | 85    | 130
| happiness| 2       | 2       | 13      | 50     | 7       | 2       | 21      | 97    | 129
| sadness  | 4       | 7       | 23      | 5      | 317    | 1       | 31      | 355   | 657
| anger    | 5       | 4       | 18      | 9       | 2       | 268    | 4       | 306   | 311
| neutral  | 5       | 0       | 19      | 6       | 12     | 0       | 254    | 296   | 305
| none     | 7       | 0       | 13      | 4       | 8       | 1       | 5       | 38    | 48
| total    | 548     | 31     | 237    | 83     | 416    | 277    | 328    | 1639  |

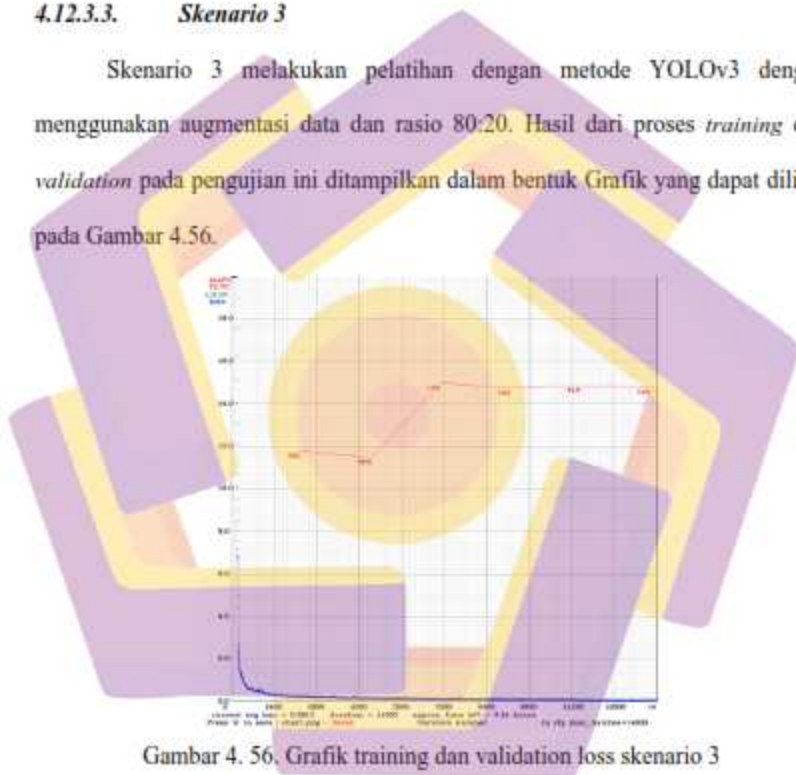
Predicted\
| surprise| precision: 92.77 % | recall: 56.26 % | avg F0.5: 84.74 %
| fear     | precision: 32.45 % | recall: 15.19 % | avg F0.5: 82.85 %
| disgust  | precision: 43.31 % | recall: 36.91 % | avg F0.5: 83.49 %
| happiness| precision: 53.21 % | recall: 25.77 % | avg F0.5: 83.44 %
| sadness  | precision: 91.95 % | recall: 46.72 % | avg F0.5: 82.54 %
| anger    | precision: 94.92 % | recall: 53.16 % | avg F0.5: 82.11 %
| neutral  | precision: 73.12 % | recall: 34.41 % | avg F0.5: 81.52 %
  
```

Gambar 4.55. Confusion Matrix skenario 2

Pengujian dilakukan menggunakan 4601 data uji terhadap model skenario 2. Pada Gambar 4.55 didapatkan tingkat akurasi presisi tertinggi yaitu 94.92% pada kelas *anger*. Dan tingkat akurasi presisi terendah yaitu pada kelas *fear* yaitu 32.43%.

4.12.3.3. Skenario 3

Skenario 3 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data dan rasio 80:20. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.56.



Gambar 4. 56. Grafik training dan validation loss skenario 3

Gambar 4.56 memperlihatkan *avg. loss* sebesar 0.0813. mAP yang didapatkan pada batch 2500 pertama adalah 59% kemudian mengalami penurunan pada batch 4500 dengan nilai mAP 57%. Puncak mAP tertinggi didapatkan pada sekitar batch 7000 yaitu 75%. Hasil akhir rata rata mAP yang didapatkan adalah

72.7%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.57.

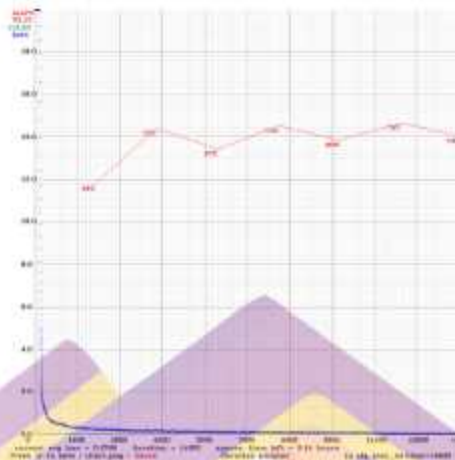
		anger	disgust	fear	happiness	neutral	sadness	surprise	none	total
Actual	anger	164	18	2	3	0	4	0	1	188
	disgust	3	82	0	13	27	18	1	21	166
	fear	0	4	33	0	2	0	0	21	58
	happiness	2	12	1	1065	28	4	3	34	1135
	neutral	2	15	0	22	460	45	19	79	627
	sadness	0	18	1	15	21	300	0	55	419
	surprise	0	0	19	13	11	2	200	0	255
	none	0	1	24	10	20	12	21	0	108
	total	172	236	79	1200	605	467	289	0	3068
anger		precision: 88.89%	recall: 64.00%	acc: 2000: 73.36%						
disgust		precision: 22.50%	recall: 61.25%	acc: 2000: 77.83%						
fear		precision: 47.14%	recall: 88.20%	acc: 2000: 77.34%						
happiness		precision: 81.25%	recall: 89.62%	acc: 1000: 76.86%						
neutral		precision: 89.29%	recall: 71.87%	acc: 1000: 74.13%						
sadness		precision: 78.00%	recall: 76.95%	acc: 1000: 77.36%						
surprise		precision: 82.76%	recall: 72.64%	acc: 1000: 77.62%						

Gambar 4. 57. *Confusion Matrix* skenario 3

Pengujian dilakukan menggunakan 3068 data uji terhadap model skenario 3. Pada Gambar 4.57 didapatkan tingkat akurasi presisi tertinggi yaitu 88.89% pada kelas *anger*. Dan tingkat akurasi presisi terendah yaitu pada kelas *fear* yaitu 47.14%.

4.12.3.4. Skenario 4

Skenario 4 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data dan rasio 70:30. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.58.



Gambar 4.58. Grafik training dan validation loss skenario 4

Gambar 4.58 memperlihatkan *avg. loss* sebesar 0.0766, mAP yang didapatkan pada batch 2100 pertama adalah 59% kemudian mengalami kenaikan bertahap pada batch 4100 dengan nilai mAP 72%. Puncak mAP tertinggi didapatkan pada sekitar batch 8100 yaitu 73%. Hasil akhir rata rata mAP yang didapatkan adalah 70.1%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.59.

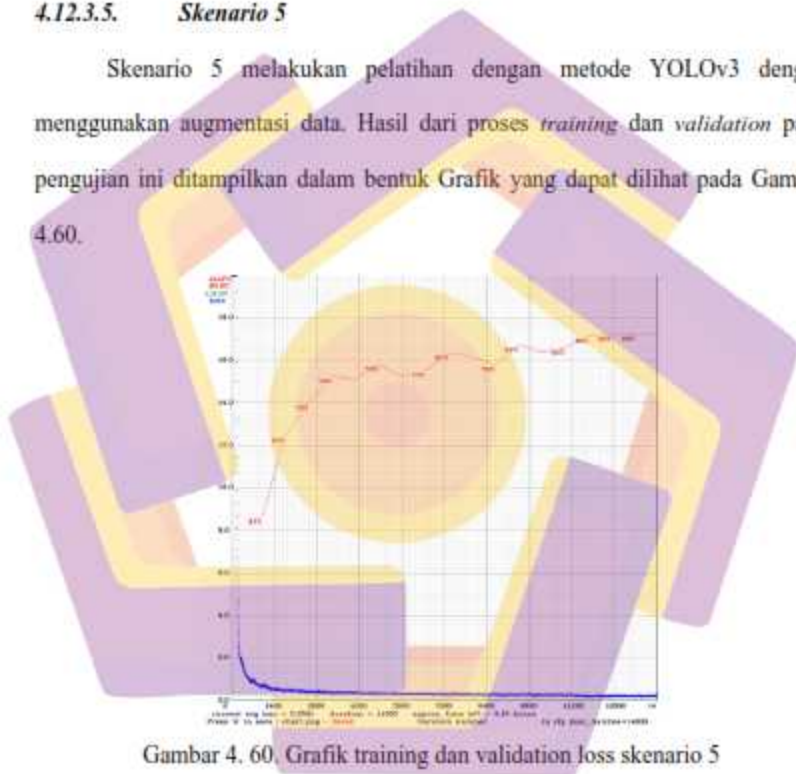
Confusion Matrix										
Matrix confidence: 0.8										
Matrix size: 8 x 8										
		anggr1	alasan1	hari	keamanan	kegiatan1	komisi	survei1	uang	total
Groundtruth	anggr1	222	0	2	2	5	0	0	0	231
	alasan1	32	324	4	0	22	0	2	0	384
	hari	0	0	43	2	7	0	0	0	52
	keamanan	0	10	0	1000	42	0	0	0	1052
	kegiatan1	0	12	0	0	420	0	0	0	432
	komisi	0	0	0	0	0	991	0	0	991
	survei1	0	0	0	0	0	0	991	0	991
	uang	22	32	22	182	12	23	0	0	311
total	254	356	47	1082	400	400	400	0	3031	
anggr1	precision:	73.27 %	recall:	79.48 %	avg f1:	74.83 %				
alasan1	precision:	86.76 %	recall:	63.18 %	avg f1:	75.72 %				
hari	precision:	45.16 %	recall:	81.18 %	avg f1:	63.16 %				
keamanan	precision:	89.84 %	recall:	87.83 %	avg f1:	72.75 %				
kegiatan1	precision:	72.79 %	recall:	72.98 %	avg f1:	72.88 %				
komisi	precision:	84.83 %	recall:	75.22 %	avg f1:	75.88 %				
survei1	precision:	88.25 %	recall:	77.88 %	avg f1:	78.16 %				

Gambar 4.59. Confusion Matrix skenario 4

Pengujian dilakukan menggunakan 4601 data uji terhadap model skenario 4. Pada Gambar 4.59 didapatkan tingkat akurasi presisi tertinggi yaitu 89,64% pada kelas *happiness*. Dan tingkat akurasi presisi terendah yaitu pada kelas *fear* yaitu 45,65%.

4.12.3.5. Skenario 5

Skenario 5 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.60.



Gambar 4. 60. Grafik training dan validation loss skenario 5

Gambar 4.60 memperlihatkan *avg. loss* sebesar 0.2504. mAP yang didapatkan pada batch 1000 pertama adalah 43% kemudian mengalami kenaikan bertahap pada batch 1700 dengan nilai mAP 62%. Puncak mAP tertinggi didapatkan pada sekitar batch 11500 yaitu 86%. Hasil akhir rata rata mAP yang

didapatkan adalah 85.9%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.61.

Confusion Matrix

Group Confusion: 0.5
Group Size: 0.5

		Prediction							
		surprise	fear	disgust	happiness	sadness	anger	neutral	none
Groundtruth	surprise	273	2	2	6	3	2	7	40
	fear	18	20	4	1	4	4	3	23
	disgust	2	8	92	6	7	5	14	100
	happiness	8	2	12	808	2	5	22	860
	sadness	8	4	17	4	363	3	27	50
	anger	3	2	8	3	1	124	3	24
	neutral	13	4	15	16	19	1	60	142
	none	4	2	2	5	2	1	4	19
	total	323	38	130	827	427	130	182	
surprise	precision	85.52 %	recall	85.46 %	avg. f1	85.52 %			
fear	precision	22.30 %	recall	52.78 %	avg. f1	32.35 %			
disgust	precision	27.30 %	recall	57.00 %	avg. f1	40.50 %			
happiness	precision	86.32 %	recall	52.41 %	avg. f1	66.54 %			
sadness	precision	25.24 %	recall	75.94 %	avg. f1	35.12 %			
anger	precision	22.67 %	recall	76.54 %	avg. f1	44.18 %			
neutral	precision	25.34 %	recall	73.28 %	avg. f1	47.53 %			

Gambar 4. 61. *Confusion Matrix* skenario 5

Pengujian dilakukan menggunakan 3068 data uji terhadap model skenario 5. Pada Gambar 4.61 didapatkan tingkat akurasi presisi tertinggi yaitu 96.73% pada kelas *happiness*. Dan tingkat akurasi presisi terendah yaitu pada kelas *disgust* yaitu 57.50%.

4.12.3.6. Skenario 6

Skenario 6 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.62.



Gambar 4.62. Grafik training dan validation loss skenario 6

Gambar 4.62 memperlihatkan *avg. loss* sebesar 0.2195. mAP yang didapatkan pada batch 1000 pertama adalah 41% kemudian mengalami kenaikan bertahap pada batch 1600 dengan nilai mAP 58%. Puncak mAP tertinggi didapatkan pada sekitar batch 10500 yaitu 83%. Hasil akhir rata rata mAP yang didapatkan adalah 82.4%. Model yang telah terbentuk dari proses training akan di uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.63.

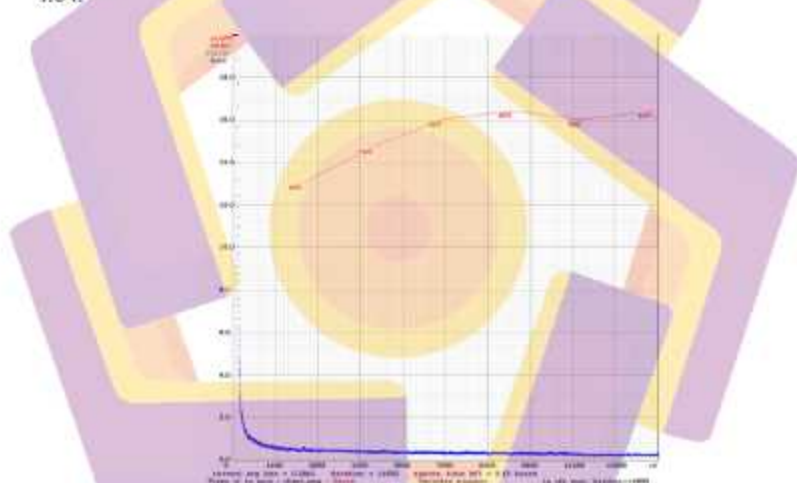
		sempurna	baik	ditunda	buruk	tidak terprediksi	error	total	
Actual Class	sempurna	839	28	5	3	38	7	915	100%
	baik	6	120	0	0	0	0	126	100%
	ditunda	2	2	119	2	0	0	123	100%
	buruk	39	5	29	1271	17	13	1404	100%
	tidak terprediksi	1	2	41	2	363	7	386	100%
Predicted Class	sempurna	0	2	24	2	221	8	237	100%
	baik	36	2	28	2	0	0	68	100%
	ditunda	6	12	11	0	0	0	29	100%
	buruk	402	78	228	1205	678	206	2527	100%
	tidak terprediksi	0	0	0	0	0	0	0	0%
Accuracy	sempurna	91.11%	96.43%	98.10%	99.27%	99.73%	99.93%	99.93%	99.93%
	baik	84.55%	90.81%	94.38%	96.38%	97.52%	97.78%	97.78%	97.78%
	ditunda	88.61%	90.61%	90.18%	91.63%	92.20%	92.63%	92.63%	92.63%
	buruk	98.15%	99.25%	99.25%	99.57%	99.73%	99.83%	99.83%	99.83%
	tidak terprediksi	87.46%	91.43%	92.48%	94.11%	94.11%	94.27%	94.27%	94.27%
Precision	sempurna	92.87%	99.08%	99.08%	99.27%	99.27%	99.27%	99.27%	99.27%
	baik	78.54%	99.61%	97.21%	99.61%	99.61%	99.61%	99.61%	99.61%

Gambar 4.63. Confusion Matrix skenario 6

Pengujian dilakukan menggunakan 4601 data uji terhadap model skenario 6. Pada Gambar 4.63 didapatkan tingkat akurasi presisi tertinggi yaitu 98.15% pada kelas *happiness*. Dan tingkat akurasi presisi terendah yaitu pada kelas *fear* 44.87%.

4.12.3.7. Skenario 7

Skenario 7 melakukan pelatihan dengan metode YOLOv3 dengan menggunakan augmentasi data. Hasil dari proses *training* dan *validation* pada pengujian ini ditampilkan dalam bentuk Grafik yang dapat dilihat pada Gambar 4.64.



Gambar 4. 64. Grafik training dan validation loss skenario 7

Gambar 4.64 memperlihatkan *avg. loss* sebesar 0.2842. mAP yang didapatkan pada batch 2100 pertama adalah 65% kemudian mengalami kenaikan bertahap pada batch 4500 dengan nilai mAP 74%. Puncak mAP tertinggi didapatkan pada sekitar batch 9100 yaitu 82%. Hasil akhir rata rata mAP yang didapatkan adalah 79.8%. Model yang telah terbentuk dari proses training akan di

uji kembali menggunakan data uji dan dievaluasi menggunakan *confusion matrix*. Hasil dari *confusion matrix* dapat dilihat pada Gambar 4.65.

Confusion Matrix

Strat. confidence: 0.5
 thresh_001: 0.5

	anger	disgust	fear	happiness	neutral	sadness	surprise	wow	total
anger	127	1	0	0	5	1	4	13	148
disgust	17	57	0	0	17	21	17	0	109
fear	1	0	46	2	1	4	12	4	70
happiness	0	1	2	180	22	0	5	0	210
neutral	0	0	0	29	483	0	19	0	531
sadness	0	1	0	14	33	791	2	0	840
surprise	4	0	0	5	5	1	289	14	318
wow	4	1	1	16	19	15	0	0	65
total	169	69	69	129	534	961	384	0	3065

anger	precision: 77.91 %	recall: 76.66 %	avg F0.5: 75.42 %
disgust	precision: 86.14 %	recall: 51.09 %	avg F0.5: 75.94 %
fear	precision: 87.02 %	recall: 67.14 %	avg F0.5: 76.58 %
happiness	precision: 88.78 %	recall: 92.15 %	avg F0.5: 74.58 %
neutral	precision: 86.12 %	recall: 71.12 %	avg F0.5: 78.73 %
sadness	precision: 76.12 %	recall: 48.72 %	avg F0.5: 73.82 %
surprise	precision: 76.32 %	recall: 87.66 %	avg F0.5: 73.78 %

Gambar 4. 65. *Confusion Matrix* skenario 7

Pengujian dilakukan menggunakan 3068 data uji terhadap model skenario 7. Pada Gambar 4.65 didapatkan tingkat akurasi presisi tertinggi yaitu 88.78% pada kelas happiness. Dan tingkat akurasi presisi terendah yaitu pada kelas fear 67.65%.

4.13. Analisis Hasil Penelitian

Setelah dilakukan pengolahan data dan proses pembelajaran model terhadap masing-masing skenario percobaan dengan menggunakan data latih, diperoleh beberapa model klasifikasi sebanyak skenario yang digunakan. Model tersebut kemudian dilakukan pengujian menggunakan data uji untuk mengetahui tingkat akurasi dan kinerja dari masing-masing model. Berikut adalah hasil pelatihan dan pengujian terhadap model klasifikasi dapat dilihat pada Tabel 4.1 berikut ini:

Tabel 4. 1. Hasil percobaan

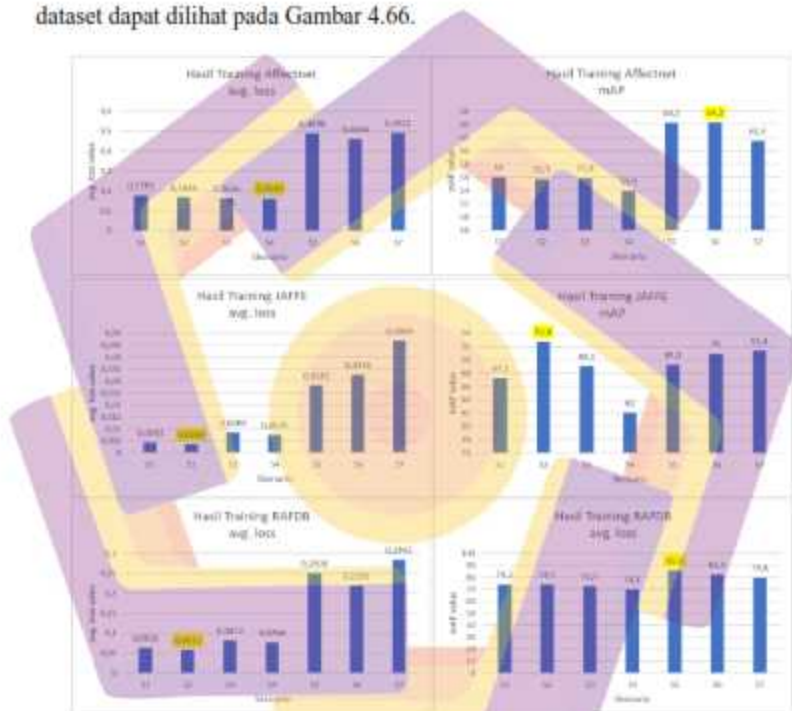
Affectnet												
Methods	Augmentation	Ratio T/T (Training/Test)	Training		Testing							
			avg_loss	mAP	neutral	happy	sad	surprise	fear	disgust	anger	avg_acc
Yolov3	x	80/20	0.1783	56	0.63	0.87	0.54	0.61	0.64	0.66	0.72	0.667143
Yolov3	x	70/30	0.1666	55.7	0.5	0.85	0.62	0.63	0.61	0.65	0.63	0.641429
Yolov3 + Augmentation	v	80/20	0.1626	55.9	0.54	0.91	0.56	0.59	0.63	0.66	0.63	0.645714
Yolov3 + Augmentation	v	70/30	0.1614	53.9	0.51	0.9	0.58	0.6	0.68	0.65	0.61	0.647143
Yolov3 + MixUp	v	80/20	0.4898	64.2	0.56	0.84	0.6	0.61	0.68	0.7	0.61	0.657143
Yolov3 + MixUp	v	70/30	0.4604	64.3	0.53	0.86	0.61	0.6	0.65	0.71	0.63	0.655714
Yolov3 + Augmentation + MixUp	v	80/20	0.4922	61.5	0.52	0.87	0.56	0.62	0.67	0.76	0.63	0.661429
JAFPE												
Methods	Augmentation	Ratio T/T (Training/Test)	Training		Testing							
			avg_loss	mAP	anger	disgust	fear	happiness	neutral	sadness	surprise	avg_acc
Yolov3	x	80/20	0.0043	87.3	1	0.79	0.75	1	1	0.8	0.33	0.808571
Yolov3	x	70/30	0.0034	92.8	0.73	0.89	1	1	0.86	0.67	1	0.878571
Yolov3 + Augmentation	v	80/20	0.0084	89.1	1	0.58	1	1	0.86	0.75	0.75	0.848571
Yolov3 + Augmentation	v	70/30	0.0075	82	0.87	0.7	1	1	0.83	0.47	0.67	0.791429
Yolov3 + MixUp	v	80/20	0.0283	89.3	1	0.7	1	0.87	1	1	0.33	0.842857
Yolov3 + MixUp	v	70/30	0.0326	91	1	0.75	1	1	1	1	0.75	0.928571
Yolov3 + Augmentation + MixUp	v	80/20	0.0469	91.4	1	0.83	1	1	1	0.75	0.6	0.882857

Tabel 4.1. Hasil percobaan (Lanjutan)

Methods	Augmentation	Ratio T/T (Training/Test)	RAFDDB									
			Training		Testing							
			avg_loss	mAP	surprise	fear	disgust	happiness	sadness	anger	neutral	avg_acc
Yolov3	x	80/20	0,0631	74,2	0,86	0,74	0,43	0,91	0,86	0,86	0,87	0,79
Yolov3	x	70/30	0,0571	74,1	0,97	0,32	0,43	0,93	0,92	0,95	0,73	0,744286
Yolov3 + Augmentation	v	80/20	0,0813	72,7	0,83	0,47	0,53	0,87	0,79	0,89	0,81	0,741429
Yolov3 + Augmentation	v	70/30	0,0766	70,1	0,88	0,46	0,61	0,9	0,85	0,73	0,73	0,737143
Yolov3 + MixUp	v	80/20	0,2504	85,9	0,85	0,65	0,57	0,97	0,83	0,83	0,86	0,797143
Yolov3 + MixUp	v	70/30	0,2195	82,4	0,9	0,45	0,5	0,98	0,87	0,83	0,75	0,754286
Yolov3 + Augmentation + MixUp	x	80/20	0,2842	79,8	0,78	0,68	0,88	0,89	0,78	0,78	0,84	0,804286

4.14. Perbandingan Hasil Training

Dalam percobaan ini dilakukan perbandingan nilai avg. loss dan juga mAP pada saat training. Perbandingan akan dilakukan perdataset berdasarkan masing masing scenario setiap dataset. Berikut grafik untuk masing masing scenario tiap dataset dapat dilihat pada Gambar 4.66.



Gambar 4. 66. Hasil Training

Pada Gambar 4.66 diperlihatkan perbandingan presentasi avg. loss dan mAP yang didapatkan selama proses training.

Dari keseluruhan grafik yang dihasilkan pada saat proses pelatihan pada setiap skenario tidak mengalami *overfitting*. Hasil yang didapatkan pada saat proses

pengujian memang memiliki tingkat akurasi yang sedikit berbeda jika dibandingkan dengan saat proses validasi. Karena pada proses pelatihan data lebih dekat dengan set validasi dan data memang disetel dengan baik pada saat menggunakan set validasi, sehingga model akan memiliki performa lebih baik di set validasi. *Overfitting* terjadi saat pelatihan jika akurasi pada proses pelatihan terus meningkat sedangkan validasi menurut antar epoc. Permasalahan ini tidak terjadi pada proses pelatihan yang telah dilakukan. Sehingga dapat dikatakan grafik dari hasil pelatihan normal.

4.15. Perbandingan Waktu Deteksi

Untuk mengetahui waktu deteksi yang dihasilkan oleh *YOLOv3* maka dilakukan perhitungan waktu deteksi pada saat pengujian. Waktu deteksi diperoleh dari *log testing*, dari keseluruhan data testing di jumlahkan kemudian dirata-ratakan. *Log testing* yang didapatkan dapat dilihat pada Gambar 4.67.

```

custom_data/KA_A02_40_100
duration 1.011608 s
custom_data/KA_A03_61_100
duration 0.095803 s
custom_data/KA_011_42_100
duration 0.088254 s
custom_data/KA_012_63_100
duration 1.021608 s
custom_data/KA_013_66_100
duration 1.017931 s
custom_data/KA_721_65_100
duration 0.323521 s
custom_data/KA_722_66_100
duration 0.308720 s
custom_data/KA_723_67_100
duration 0.358020 s
custom_data/KA_724_68_100
duration 0.311188 s

```

Gambar 4. 67. Contoh *log testing*

```

with open(test_duration) as f:
    data = []
    while f:
        line = f.readline()
        if len(line) == 0:
            break

        if line.split()[0] == 'duration':
            data.append(float(line.split()[1]))

```

```

print('Your total time:')
print(sum(data))

print('Your data:')
print(len(data))

print('Your detection time is:')
print(sum(data)/len(data))

```

Detail waktu deteksi dapat dilihat pada Gambar 4.68.

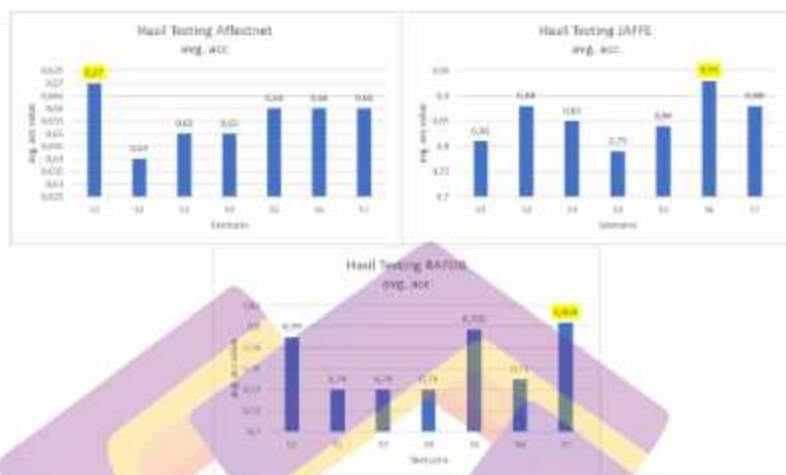




Gambar 4. 68. Perbandingan grafik waktu deteksi

4.16. Perbandingan Hasil Testing

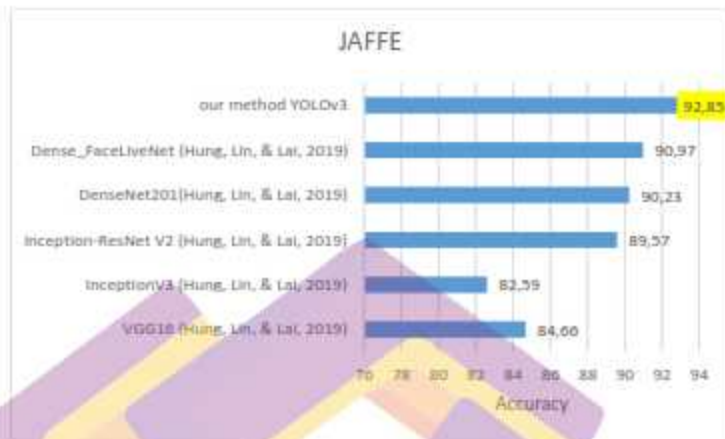
Dalam percobaan ini dilakukan perbandingan hasil training berdasarkan nilai precision dan recall dari 3 dataset dengan masing masing memiliki 7 skenario. Kemudian dari hasil perbandingan akan diambil skenario dengan nilai akurasi tertinggi untuk dibandingkan dengan penelitian sebelumnya. Berikut grafik hasil yang didapatkan pada tiap skenario dataset dapat dilihat pada Gambar 4.69.



Gambar 4. 69. Perbandingan grafik hasil testing

Pada Gambar 4.68, dapat dilihat pada pengujian terhadap dataset affectnet hasil akurasi tertinggi didapatkan oleh scenario 1 yang mana tanpa proses augmentasi data dengan rasio perbandingan 80:20 sebesar 67%. Untuk pengujian terhadap dataset *JAFFE* hasil akurasi tertinggi didapatkan oleh scenario 6 yang mana menggunakan proses augmentasi *mixup* dengan rasio 70:30 sebesar 93%. Kemudian untuk pengujian terhadap dataset *RAF-DB* hasil akurasi tertinggi didapatkan oleh scenario 7 yang mana menggunakan proses augmentasi data *Crop*, *Rotate*, *Shear*, *Flip* dan *MixUp* dengan rasio perbandingan data 80:20 sebesar 80.4%.

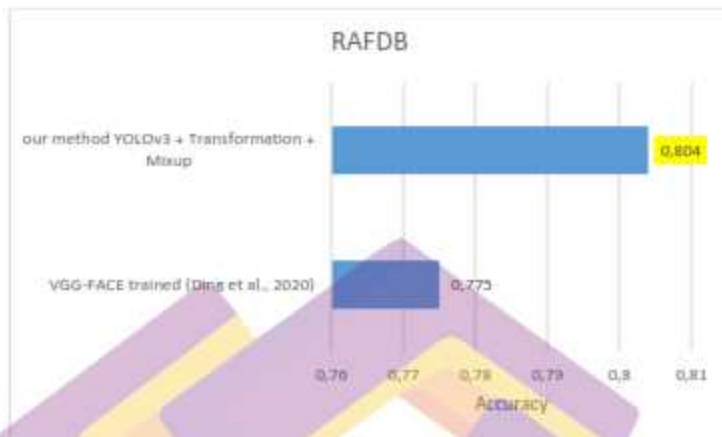
Setelah didapatkan hasil akurasi tertinggi dari scenario berikut perbandingan dengan penelitian sebelumnya dapat dilihat pada Gambar 4.70, Gambar 4.71, dan Gambar 4.72.



Gambar 4. 70. Perbandingan dataset *JAFFE*



Gambar 4. 71. Perbandingan dataset *Affectnet*



Gambar 4. 72. Perbandingan dataset *RAF-DB*

Pada Gambar 4.69, hasil dari percobaan terhadap dataset *JAFFE* menggunakan metode *YOLOv3* mendapatkan penambahan dari segi akurasi $\pm 1.88\%$ dari penelitian sebelumnya. Pada Tabel 4.70 hasil dari percobaan terhadap dataset *Affectnet* menggunakan metode *YOLOv3* mendapatkan penambahan akurasi $\pm 7.71\%$ dari penelitian sebelumnya. Pada Tabel 4.71 hasil dari percobaan terhadap dataset *RAF-DB* menggunakan metode *YOLOv3* mendapatkan penambahan akurasi $\pm 2.9\%$.

4.17. Pengaruh augmentasi data

Dari segala percobaan yang telah dilakukan oleh peneliti mulai dari mendefinikan masalah hingga mendapatkan hasil dari penelitian ada beberapa hal yang bisa didapatkan. Yang pertama adalah mengetahui tingkat akurasi yang dihasilkan oleh metode *YOLOv3* untuk pengenalan ekspresi wajah. Selain itu juga mengetahui waktu deteksi yang dihasilkan oleh metode *YOLOv3*. Dari hasil penelitian yang telah dilakukan jika merujuk pada permasalahan, dapat

disimpulkan bahwa penelitian yang dilakukan sudah berhasil menjawab permasalahan awal, dimana untuk mendeteksi ekspresi wajah dapat menggunakan metode *YOLOv3* dengan proses augmentasi data dengan akurasi untuk dataset *JAFFE* 93%, *Affectnet* 66.71%, dan *RAF-DB* 80.4%. Kemudian untuk waktu deteksi yang dihasilkan adalah pada dataset *JAFFE* 0.94 detik, *Affectnet* 0.98 detik, dan *RAF-DB* 0.98 detik yang mana merupakan waktu yang cukup cepat untuk mendapatkan hasil apabila nantinya model diterapkan untuk mendeteksi ekspresi wajah pada sebuah video yang membutuhkan hasil yang segera sebagai bahan pertimbangan untuk analisis ekspresi wajah.

Dalam penelitian ini dilakukan percobaan dengan menggunakan dua metode augmentasi data yang berbeda yaitu Transformasi Geometri dan MixUp. Transformasi Geometri yang berupa penggabungan dari *Rotate*, *Flip*, *Crop*, *Shear*. Penambahan data dengan menggunakan transformasi geometri pada set pelatihan tidak memberikan efek yang signifikan terhadap akurasi yang didapatkan. Transformasi geometri hanya menambahkan jumlah data akan tetapi model tidak memiliki variasi baru ketika model digunakan untuk mendeteksi citra wajah yang terhalang oleh benda lain seperti hasil pengujian pada dataset *Affectnet* dan *RAF-DB*. Dari hasil penelitian yang didapatkan dari serangkaian percobaan dengan berbagai macam skenario yang telah dilakukan. Maka teknik augmentasi data terbaik yang dapat meningkatkan hasil akurasi ketika menggunakan metode *YOLOv3* dan ketika mendapatkan dataset dengan jumlah yang besar serta tidak merata pada setiap kelasnya yaitu dengan menggunakan augmentasi data *MixUp*.

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil pengujian dan analisis yang telah dilaksanakan terhadap 7 skenario pada 3 dataset yang berbeda untuk pengenalan ekspresi wajah menggunakan metode *YOLOv3* dapat disimpulkan:

1. Tingkat akurasi yang dihasilkan oleh metode *YOLOv3* tanpa proses augmentasi data untuk dataset *Affectnet* 66.71%, *JAFFE* 87.85%, *RAF-DB* 79% sedangkan tingkat akurasi tertinggi yang dihasilkan oleh metode *YOLOv3* dengan proses augmentasi data untuk dataset *Affectnet* 66.14%, *JAFFE* 92.85%, *RAF-DB* 80.42%.
2. Waktu deteksi yang dihasilkan oleh metode *YOLOv3* tanpa proses augmentasi data pada dataset *Affectnet* 0.981 detik, *JAFFE* 1.04 detik, *RAF-DB* 0.96 detik sedangkan waktu deteksi yang dihasilkan oleh metode *YOLOv3* dengan proses augmentasi data pada dataset *Affectnet* 0.984 detik, *JAFFE* 0.949, *RAF-DB* 0.97.
3. Perlakuan terhadap citra untuk pengenalan ekspresi wajah menggunakan augmentasi data berpengaruh terhadap akurasi dan waktu deteksi dari model yang dihasilkan. Pada dataset *Affectnet* augmentasi berpengaruh kurang baik karena menurunkan hasil akurasi dibandingkan dengan tanpa augmentasi sedangkan untuk waktu deteksi tidak berbeda jauh. Pada dataset *JAFFE* augmentasi berpengaruh baik karena meningkatkan hasil

akurasi dan juga menurunkan waktu deteksi. Dan yang terakhir pada dataset *RAF-DB*, augmentasi berpengaruh baik terhadap akurasi tetapi sedikit meningkatkan waktu deteksi.

5.2. Saran

Berikut ini adalah beberapa saran yang dapat dijadikan pedoman untuk melakukan pengembangan penelitian ini, antara lain:

1. Permasalahan pada dataset di alam liar seperti *affectnet* dan *raf-db* adalah ekspresi wajah tertutup oleh benda lain seperti tangan, kepala orang lain, ataupun benda mati. Maka diperlukan metode augmentasi yang dapat memanipulasi hal-hal tersebut.
2. Melakukan kombinasi antara dua dataset untuk mendapatkan hasil yang lebih optimal.
3. Melakukan percobaan dengan augmentasi data tetapi dengan model dan arsitektur yang berbeda seperti DenseNet, MobileNet, VGG16 dan lain sebagainya.
4. Dalam pemberian anotasi pada citra wajah yang didapatkan masih menggunakan cara manual menggunakan bantuan tools. Kedepannya dapat dilakukan menggunakan teknik *Face Detection* agar dapat mempermudah dalam pemberian anotasi pada jumlah dataset yang besar.

DAFTAR PUSTAKA

PUSTAKA BUKU

T d Sutoyo, Edy Mulyanto, Vincent Suhartono, O. D. N. (2009). *T d Sutoyo, Edy Mulyanto, Vincent Suhartono, Oky D Nurhayati*. Yogyakarta: Andi.

Dana Harry Ballard & Brown, C. M. (1982). *Computer vision*. Englewood Cliffs, N.J Prentice-Hall, 523.

Ponce, J., & Forsyth, D. (2012). Computer vision: a modern approach. In *Computer*, xxix, 761 p.

PUSTAKA MAJALAH, JURNAL ILMIAH ATAU PROSIDING

Canedo, D., & Neves, A. J. R. (2019). Facial expression recognition using computer vision: A systematic review. *Applied Sciences (Switzerland)*, 9(21), 1–31. <https://doi.org/10.3390/app9214678>

Chaitanya, Sarath, S., Malavika, Prasanna, & Karthik. (2020). Human Emotions Recognition from Thermal Images using Yolo Algorithm. *Proceedings of the 2020 IEEE International Conference on Communication and Signal Processing, ICCSP, 2020*, 1139–1142. <https://doi.org/10.1109/ICCSP48568.2020.9182148>

Chauhan, M. (2014). *Study & Analysis of Different Face Detection Techniques*. 5(2), 1615–1618.

Chen, W., Huang, H., Peng, S., Zhou, C., & Zhang, C. (2020). YOLO-face: a real-time face detector. *Visual Computer*. <https://doi.org/10.1007/s00371-020-01831-7>

Cristanto, J., Ratri, K., & Wardani, R. (2017). Penerapan Metode Single-Layer Feed-Forward Neural Network Menggunakan Kernel Gabor untuk Pengenalan Ekspresi Wajah. *Jurnal Telematika*, 12(1), 1–5.

Dana Harry Ballard & Brown, C. M. (1982). *Computer vision*. Englewood Cliffs,

N.J Prentice-Hall.

- Ding, H., Zhou, P., Chellappa, R., Kollias, D., Cheng, S., Ververas, E., Kotsia, I., Zafeiriou, S., Farzaneh, A. H., & Qi, X. (2020). Deep Neural Network Augmentation: Generating Faces for Affect Analysis. *Proceedings - 2021 IEEE Winter Conference on Applications of Computer Vision, WACV 2021*, 128(5), 1455–1484. <https://doi.org/10.1109/IJCB48548.2020.9304923>
- Esau, N., Wetzel, E., Kleinjohann, L., Kleinjohann, B., Reyhan Achmad Rizall, Suardin Gulol, Octavriana Della C. Sihombing1, Ardi Bernandustahi Miduk Napitupulu1, Amsal Yusuf Gultom1, T. J. S., Area, U. M., Deteksi, U., Pada, E., Wang, H., Hou, S., Studi, P., Teknik, M., Komputer, F. I., Sriwijaya, U., Primasiwi, C., Tjandrasa, H., A. Navastara, D., Drid, K., Allaoui, M., ... Viviani, P. (2019). Real time object detection using CNN. *ArXiv*, 6(1), 69–76. <https://doi.org/10.3390/app9214678>
- Fandisyah, A., Fandisyah, A. F., Iriawan, N., & Winahju, W. S. (2021). Deteksi Kapal di Laut Indonesia Menggunakan YOLOv3. *Jurnal Sains Dan Seni ITS*, 10(1), D25–D32. http://ejournal.its.ac.id/index.php/sains_seni/article/view/59312%0Ahttps://ejournal.its.ac.id
- Fardani, E. Y., Arifianto, A., & Ramadhani, K. N. (2018). Pengenalan Ekspresi Wajah Menggunakan LGBP dan SVM. 5(3), 1–8.
- Fitri, V. A., Andreswari, R., & Hasibuan, M. A. (2019). Sentiment analysis of social media Twitter with case of Anti-LGBT campaign in Indonesia using Naive Bayes, decision tree, and random forest algorithm. *Procedia Computer Science*, 161, 765–772. <https://doi.org/10.1016/j.procs.2019.11.181>
- Hardiansyah, B., & Primandari, P. N. (2018). Sistem Pakar Pengenalan Ekspresi Wajah Manusia Menggunakan Metode Kohonen Self Organizing Dan Principal Componen Analysis. *INTEGER: Journal of Information Technology*, 3(2), 43–54. <https://doi.org/10.31284/j.integer.2018.v3i2.310>
- Ho, D., Liang, E., Stoica, I., Abbeel, P., & Chen, X. (2019). Population based augmentation: Efficient learning of augmentation policy schedules. *36th International Conference on Machine Learning, ICML 2019, 2019-June*, 4843–4856.
- Hung, J. C., Lin, K. C., & Lai, N. X. (2019). Recognizing learning emotion based on convolutional neural networks and transfer learning. *Applied Soft Computing Journal*, 84, 105724. <https://doi.org/10.1016/j.asoc.2019.105724>

- Husdi, H. (2016). Pengenalan Ekspresi Wajah Pengguna Elearning Menggunakan Artificial Neural Network Dengan Fitur Ekstraksi Local Binary Pattern Dan Gray Level Co-Occurrence Matrix. *ILKOM Jurnal Ilmiah*, 8(3), 212–219. <https://doi.org/10.33096/ilkom.v8i3.58.212-219>
- Jupiyandi, S., Saniputra, F. R., Pratama, Y., Dharmawan, M. R., Cholissodin, I., Pendeteki, P., Dengan, O., You, M., Look, O., Harahap, M., Elfrida, J., Agusman, P., Rafael, M., Abram, R., Andrianto, K., kunci-Visi Komputer, K., Arus Lalu Lintas, P., Kendaraan, D., lia dwi jayanti, ... K.R., N. (2020). No 主観的健康感を中心とした在宅高齢者における健康関連指標に関する共分散構造分析Title. *العراقية - كلية العلوم - جامعة بغداد*, بغداد , الع ارق. 17(1), 6–38. <https://doi.org/10.1109/ICACCS48705.2020.9074315>
- Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. In *Nature*. <https://doi.org/10.1038/nature14539>
- Li, S., Deng, W., & Du, J. P. (2017). Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017-January*. <https://doi.org/10.1109/CVPR.2017.277>
- Li, Y., Zeng, J., Shan, S., Chen, X., Siqueira, H., Magg, S., Wermter, S., Viswanatha Reddy, G., Dharma Savarni, C. V. R., & Mukherjee, S. (2020). Facial expression recognition in the wild, by fusion of deep learnt and hand-crafted features. *IEEE Transactions on Image Processing*, 28(level 1), 5800–5809. <https://doi.org/10.1109/TIP.2018.2886767>
- Liang, D., Yang, F., Zhang, T., & Yang, P. (2018). Understanding mixup training methods. *IEEE Access*, 6, 58774–58783. <https://doi.org/10.1109/ACCESS.2018.2872698>
- Lyons, M., Kamachi, M., & Gyoba, J. (2017). Japanese Female Facial Expression (JAFFE) Database. In *Proceedings of third international conference on automatic face and gesture recognition*.
- Mahmoodi, M. R. (2017). Fast and Efficient Skin Detection for Facial Detection. *ArXiv*.
- Meshach, W. T., Hemajothi, S., & Anita, E. A. M. (2020). Real-time facial expression recognition for affect identification using multi-dimensional SVM. *Journal of Ambient Intelligence and Humanized Computing*, 0123456789. <https://doi.org/10.1007/s12652-020-02221-6>

- Mollahosseini, A., Hasani, B., & Mahoor, M. H. (2019). AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild. *IEEE Transactions on Affective Computing*, 10(1). <https://doi.org/10.1109/TAFFC.2017.2740923>
- Muhammed, A., Mol, R., Vijay, L. R., Ajith, S. S., & Shamna, A. R. (2020). *Facial Expression Recognition using Support Vector Machine (SVM) and Convolutional Neural Network (CNN)*, 8.
- Mustakim, A., Santoso, I., & Zahra, A. A. (2017). Pengenalan Ekspresi Wajah Manusia Menggunakan Tapis Gabor 2-D Dan Support Vector Machine (Svm). *Transient*, 6(3), 232. <https://doi.org/10.14710/transient.6.3.232-238>
- Ponce, J., & Forsyth, D. (2012). Computer vision: a modern approach. In *Computer*.
- Prasetyo, J. D., Fatah, Z., & Saleh, T. (2017). Ekstraksi Fitur Berbasis Average Face Untuk Pengenalan Ekspresi Wajah. *Jurnal Ilmiah Informatika*, 2(2), 130–134. <https://doi.org/10.35316/jimi.v2i2.464>
- Reyhan Achmad Rizall, Suardin Gulol, Octavriana Della C. Sihombing1, Ardi Bernandustahi Miduk Napitupulu1, Amsal Yusuf Gultom1, T. J. S. (2019). Analisis Gray Level Co-Occurrence Matrix (GlcM) Dalam Mengenali Citra Ekspresi Wajah. *Jurnal Mantik*, 3(January), 31–38.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3), 211–252. <https://doi.org/10.1007/s11263-015-0816-y>
- Saito, S., Wei, L., Hu, L., Nagano, K., & Li, H. (2017). Photorealistic facial texture inference using deep neural networks. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017-January*, 2326–2335. <https://doi.org/10.1109/CVPR.2017.250>
- Sanjaya, J., & Ayub, M. (2020). Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup. *Jurnal Teknik Informatika Dan Sistem Informasi*, 6(2), 311–323. <https://doi.org/10.28932/jutisi.v6i2.2688>

- Sonka, M., Hlavac, V., & Boyle, R. (1993). Image Processing, Analysis and Machine Vision. In *Image Processing, Analysis and Machine Vision*. <https://doi.org/10.1007/978-1-4899-3216-7>
- Susanti, R., Fadillah, N., Chakraborty, S., Das, D., Alamsyah, D., Pratama, D., Hao, H., Pei, M., Zhao, M., Rehman, Y. A. U., Po, L. M., Liu, M., Zou, Z., Ou, W. W., Zhao, Y., Sonare, B., Mokadam, S., Hulsurkar, Y., Bhange, M., ... Ramanan, D. (2019). Penentuan Emosi Pada Video Dengan Convolutional Neural Network. *Sensors (Switzerland)*, *3*(2), 1627–1645. <https://doi.org/10.1007/s11042-014-2322-6>
- T d Sutoyo, Edy Mulyanto, Vincent Suhartono, O. D. N. (2009). *T d Sutoyo, Edy Mulyanto, Vincent Suhartono, Oky D Nurhayati*. Yogyakarta: Andi.
- Wang, H., & Hou, S. (2020). Facial Expression Recognition based on the Fusion of CNN and SIFT Features. *ICEIEC 2020 - Proceedings of 2020 IEEE 10th International Conference on Electronics Information and Emergency Communication*, 190–194. <https://doi.org/10.1109/ICEIEC49280.2020.9152361>
- Wang, Y., Li, Y., Song, Y., & Rong, X. (2019). Facial expression recognition based on auxiliary models. *Algorithms*, *12*(11), 1–15. <https://doi.org/10.3390/a12110227>
- Wiyono, A. R., & Imah, E. M. (2018). Pengenalan Citra Ekspresi Wajah Menggunakan Algoritma Principal Component Analysis (Pca) Dan Extreme Learning Machine (Elm). *Jurnal Ilmiah Matematika*, *6*(2), 2–6.
- Yang, M. H., Kriegman, D. J., & Ahuja, N. (2002). Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(1), 34–58. <https://doi.org/10.1109/34.982883>
- Yang, S., Luo, P., Loy, C. C., & Tang, X. (2016). WIDER FACE: A face detection benchmark. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-December*, 5525–5533. <https://doi.org/10.1109/CVPR.2016.596>
- Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. (2018). MixUp: Beyond empirical risk minimization. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*.