

**KLASIFIKASI MOTILITAS PADA SPERMA MANUSIA
BERDASARKAN KECEPATAN GERAK SPERMA BERBASIS MODEL
YOLOv8**

(Skripsi)

Oleh

Cely Febita Ilham

2117051022



**JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
2025**

**KLASIFIKASI MOTILITAS PADA SPERMA MANUSIA
BERDASARKAN KECEPATAN GERAK SPERMA BERBASIS MODEL
YOLOv8**

Oleh

Cely Febita Ilham

Skripsi

**Sebagai Salah Satu Syarat untuk Mendapat Gelar
SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam**



**JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
2025**

ABSTRAK

KLASIFIKASI MOTILITAS PADA SPERMA MANUSIA BERDASARKAN KECEPATAN GERAK SPERMA BERBASIS MODEL YOLOv8

Oleh

CELY FEBITA ILHAM

Motilitas sperma merupakan indikator penting dalam menilai tingkat kesuburan pria. Penilaian manual terhadap motilitas sperma cenderung subjektif dan memakan waktu, sementara teknologi seperti CASA memiliki biaya yang tinggi dan keterbatasan fleksibilitas. Penelitian ini bertujuan untuk mengembangkan metode klasifikasi motilitas sperma berdasarkan kecepatan geraknya menggunakan model YOLOv8 yang diintegrasikan oleh algoritma ByteTrack. Dataset yang digunakan merupakan VISEM dataset dengan format video, yang kemudian diekstrak menjadi 1.330 citra dan dilatih dalam enam skenario dengan kombinasi 3 *epoch* (50, 100, 150) dan 2 *learning rate* (0.002 dan 0.0002). Model YOLOv8 berhasil mendeteksi objek sperma dengan performa terbaik pada 100 *epoch* dan *learning rate* 0.002, menghasilkan presisi 0.824, *recall* 0.844, *F1-Score* 0.838, dan mAP 0.86. YOLOv8 lalu diintegrasikan dengan ByteTrack dan menghasilkan koordinat *bounding box* yang digunakan untuk menghitung kecepatan gerak, yang menjadi dasar klasifikasi ke dalam tiga kategori: progresif, non-progresif, dan *immotile*. Hasil klasifikasi menunjukkan dominasi objek progresif pada ketiga video uji. Evaluasi terhadap akurasi program dibandingkan dengan penghitungan manual menghasilkan tingkat akurasi sebesar 91,81%, 89,05%, dan 86,67% pada masing-masing video. Penelitian ini menunjukkan bahwa pendekatan ini dapat digunakan untuk mengklasifikasikan motilitas sperma berdasarkan kecepatan geraknya secara lebih efisien dan objektif.

Kata Kunci : motilitas sperma, YOLOv8, ByteTrack, deteksi dan pelacakan objek, kecepatan gerak, klasifikasi.

ABSTRACT

CLASSIFICATION OF HUMAN SPERM MOTILITY BASED ON SPERM MOVEMENT SPEED USING A MODEL-YOLOv8 APPROACH

By

CELY FEBITA ILHAM

Sperm motility is a crucial indicator in assessing male fertility. Manual evaluation of sperm motility tends to be subjective and time-consuming, while technologies such as CASA are costly and lack flexibility. This study aims to develop a sperm motility classification method based on movement speed using the YOLOv8 model integrated with the ByteTrack algorithm. The dataset used is the VISEM dataset in video format, which was extracted into 1,330 images and trained under six scenarios combining three epoch values (50, 100, 150) and two learning rates (0.002 and 0.0002). The YOLOv8 model successfully detected sperm objects, with the best performance achieved at 100 epochs and a learning rate of 0.002, resulting in a precision of 0.824, recall of 0.844, F1-score of 0.838, and mAP of 0.86. YOLOv8 was then integrated with ByteTrack to generate bounding box coordinates, which were used to calculate the movement speed of each object. These speed values were then used as the basis for classifying sperm motility into three categories: progressive, non-progressive, and immotile. The classification results showed a dominance of progressive objects in all three test videos. An evaluation comparing the program's accuracy to manual calculations yielded accuracy levels of 91.81%, 89.05%, and 86.67%, respectively. This study demonstrates that an object detection and tracking-based approach can be utilized to classify sperm motility more efficiently and objectively based on movement speed.

Keywords: *sperm motility, YOLOv8, ByteTrack, object detection and tracking, movement speed, classification.*

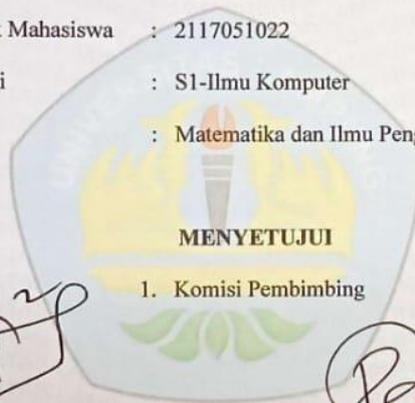
Judul Skripsi : **KLASIFIKASI MOTILITAS PADA SPERMA
MANUSIA BERDASARKAN KECEPATAN
GERAK SPERMA BERBASIS MODEL
YOLOv8**

Nama Mahasiswa : **Cely Febita Ilham**

Nomor Pokok Mahasiswa : 2117051022

Program Studi : S1-Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam



1. Komisi Pembimbing

Dr. Aristoteles, S.Si., M.Si.
NIP. 198105212006041002

Ridho Sholehurrohman, M.Mat.
NIP. 232111970128101

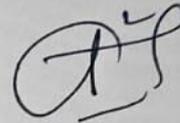
2. Ketua Jurusan Ilmu Komputer

Dwi Sakethi, S.Si., M.Kom.
NIP. 196806111998021001

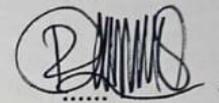
MENGESAHKAN

1. Tim Penguji

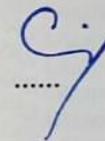
Ketua : Dr. Aristoteles, S.Si., M.Si.



Sekretaris : Ridho Sholehurrohman, M.Mat.



Penguji
Bukan Pembimbing : Rico Andrian, S.Si., M.Kom.



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Heri Satria, S.Si., M.Si.
NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi : 16 Mei 2025

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Cely Febita Ilham

NPM : 2117051022

Dengan ini menyatakan bahwa skripsi saya yang berjudul **“Klasifikasi Motilitas Pada Sperma Manusia Berdasakan Kecepatan Gerak Sperma Berbasis Model YOLOv8”** merupakan hasil karya saya sendiri dan bukan karya orang lain. Seluruh tulisan yang tertuang di skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila ini di kemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 2 Juni 2025



Cely Febita Ilham
NPM. 2117051022

RIWAYAT HIDUP



Penulis dilahirkan di Bandar Jaya, Lampung tengah tanggal 19 Februari 2003, sebagai anak kedua dari dua bersaudara, dari Bapak Azwir dan Ibu Yanti Mulyani. Penulis menyelesaikan pendidikan formal pertama kali di Taman Kanak-Kanak (TK) Nurul Huda Yukum Jaya pada tahun 2008, lalu pendidikan Sekolah Dasar (SD) diselesaikan di SDN 3 Yukum Jaya pada tahun 2015. Kemudian pendidikan menengah pertama di SMP Negeri 1 Terbanggi Besar yang diselesaikan pada tahun 2018, lalu melanjutkan ke pendidikan menengah atas di SMA Negeri 1 Terbanggi Besar yang diselesaikan pada tahun 2021.

Pada tahun 2021, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur SNMPTN. Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan antara lain.

1. Menjadi Anggota Muda Ilmu Komputer (ADAPTER) Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2021/2022.
2. Menjadi anggota Bidang Kaderisasi Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2022/2023 dan periode 2023/2024.
3. Menjadi Sekretaris Pelaksana kegiatan PKKMB Jurusan Ilmu Komputer pada periode 2023/2024.

4. Menjadi Asisten Dosen Jurusan Ilmu Komputer pada mata kuliah Basis Data tahun 2023 serta mata kuliah Analisis Desain dan Sistem Informasi pada tahun 2024.
5. Melaksanakan Kerja Praktik di PT. IPC Terminal Petikemas Panjang pada periode 2023/2024 dengan program kerja *website*.
6. Mengikuti Studi Independen MSIB di RevoU *Tech Academy* pada tahun 2024.
7. Mengikuti Kuliah Kerja Nyata (KKN) pada periode II tahun 2024 di Desa Srimenanti, Kecamatan Sribhawono, Kabupaten Lampung Timur

MOTTO

“Allah tidak akan membebani seseorang melainkan sesuai dengan kesanggupannya”

(QS. Al-Baqarah 2:286)

“Seberat apapun rintangannya, pulanglah sebagai sarjana”

“Rasakanlah setiap proses yang kamu tempuh dalam hidupmu, sehingga kamu tau betapa hebatnya dirimu sudah berjuang sampai detik ini”

“Orang tua dirumah menanti kepulanganmu dengan hasil yang membanggakan, jangan kecewakan mereka. Simpan keluhmu, sebab letihmu tak sebanding dengan perjuangan mereka menghidupimu”

“Tugas kita bukanlah untuk berhasil, tugas kita adalah mencoba karena didalam mencoba itulah kita menemukan kesempatan untuk berhasil”

(Buya Hamka)

PERSEMBAHAN

Alhamdulillahirobbilalamin

Puji syukur kehadiran Allah Subhanahu Wa Ta'ala atas segala rahmat dan karunia-Nya sehingga skripsi ini dapat diselesaikan dengan sebaik-baiknya. Shalawat serta salam selalu tercurahkan kepada junjungan Nabi Agung Muhammad Shallallahu'Alaihi Wasallam.

Kupersembahkan karya ini kepada:

Kedua Orang Tua dan Kakak Tercinta

Yang selalu mendukung, mendo'akan, memberikan cinta dan kasih sayang yang tak terhingga yang selalu menyertai. Kuucapkan terima kasih sebesar-besarnya atas kesabaran, doa, pengorbanan dan perjuangan yang menjadi sumber kekuatanku hingga titik ini.

Seluruh Keluarga Besar Ilmu Komputer 2021

Yang senantiasa memberikan semangat, bantuan dan dukungan selama masa studi berlangsung.

Almamater Tercinta, Universitas Lampung dan Jurusan Ilmu Komputer

Tempat bernaung mengemban semua ilmu untuk menjadi bekal hidup.

SANWACANA

Puji syukur penulis ucapkan kehadiran Tuhan Yang Maha Esa, karena atas rahmat dan hidayah-Nya skripsi ini dapat diselesaikan. Skripsi dengan judul “Klasifikasi Motilitas Pada Sperma Manusia Berdasarkan Kecepatan Gerak Sperma Berbasis Model YOLOv8” adalah salah satu syarat untuk memperoleh gelar sarjana Komputer di Universitas Lampung.

Dalam kesempatan ini penulis mengucapkan terima kasih kepada:

1. Allah Subhanahu Wa Ta'ala yang selalu menjadi rumah untuk penulis mencurahkan segala rasa, harap dan kegelisahan yang tidak dapat dimengerti oleh manusia melalui sujud dan doa. Segala puji dan syukur penulis panjatkan atas segala limpahan rahmat, kekuatan, dan ketenangan yang menyertai setiap langkah perjalanan ini.
2. Mamah, Kakak, dan Papah yang selalu mengajarkan kebaikan, memberikan doa, memberikan motivasi dan memberi dukungan baik moral, kasih sayang maupun material yang luar biasa dengan segala ketulusan tanpa pamrih, hingga penulis berhasil sampai di tahap ini, menyelesaikan penulisan skripsi untuk meraih gelar sarjana. Terima kasih atas segala dukungan dan kasih sayang yang tulus serta tiada henti.
3. Bapak Dr. Eng. Heri Satria, S.Si., M.Si. selaku Dekan FMIPA Universitas Lampung.
4. Bapak Dwi Sakethi, S.Si., M.Kom. selaku Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.
5. Ibu Yunda Heningtyas, M.Kom. selaku Sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
6. Bapak Tristiyanto, Ph.D. selaku Pembimbing Akademik penulis yang selalu mendukung peningkatan akademik penulis.

7. Bapak Dr. Aristoteles, S.Si., M.Si. selaku Pembimbing Utama yang telah memberikan arahan, ide, kritik, serta saran dalam menyelesaikan penelitian ini. Terima kasih penulis ucapkan kepada Bapak karena telah mempermudah setiap proses tanpa membebani, telah percaya kepada penulis dalam penelitian dan penulisan skripsi, serta memberikan peluang kepada penulis untuk mengembangkan potensi dalam penelitian lebih lanjut. Semoga Allah swt senantiasa melimpahkan kesehatan, keberkahan, serta balasan yang terbaik untuk setiap kebaikan yang Bapak berikan.
8. Bapak Ridho Sholehurrohman, M.Mat. selaku Pembimbing Kedua yang selalu dapat meluangkan waktu, memberikan arahan dan bantuan ketika mengalami kesulitan dalam proses penelitian dan penulisan skripsi. Terima kasih atas segala kesabaran Bapak untuk mempermudah setiap proses dan selalu membuka pintu konsultasi dengan lapang, serta memahami setiap kesulitan yang penulis hadapi. Semoga Allah swt senantiasa melimpahkan kesehatan, keberkahan, kemudahan atas segala urusan serta balasan yang terbaik untuk setiap kebaikan yang Bapak berikan.
9. Bapak Rico Andrian, S.Si, M.Kom. selaku Pembahas Skripsi yang telah memberikan masukan serta saran yang bermanfaat untuk perbaikan skripsi ini.
10. Seluruh Staf dan Karyawan Jurusan Ilmu Komputer, Ibu Ade Nora Maela, Bang Zainuddin, Mas Nofal, dan Mas Syam yang telah membantu segala urusan administrasi penulis di Jurusan Ilmu Komputer.
11. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmu, motivasi dan pengalaman hidup selama penulis menempuh pendidikan di Jurusan Ilmu Komputer Universitas Lampung.
12. Kepada seseorang yang tidak kalah penting kehadirannya, Fikri Al Hafizh. Terima kasih telah menjadi bagian dalam setiap proses penulis. Berkontribusi banyak dalam proses penulis menyusun skripsi, baik tenaga, waktu, maupun materi. Telah menjadi rumah, pendamping dalam segala hal yang menemani, mendukung, ataupun menghibur dalam kesedihan, mendengar keluh kesah, dan memberi semangat untuk pantang menyerah dalam setiap proses yang penulis lalui.

13. Sahabat seperbimbingan skripsi Salma Dara Canita dan Nabillah Aisyah yang senantiasa membantu penulis untuk *sharing*, mengurus berkas, segala bantuan kecil hingga besar yang tidak dapat disebutkan satu persatu, saling memotivasi dan menyemangati di saat buntu dan panik. Terima kasih atas kebersamaan dan dukungannya selama proses ini.
14. Sahabat tersayang penulis Nawang Virli Noviawati dan Jenny Adeti, yang selalu menghibur penulis dengan segala cerita dan tingkah yang lucu, selalu ingat penulis dimana pun keberadaannya, selalu peduli dalam setiap situasi. Terima kasih telah menjadi tempat berbagi tawa, cerita, haru dan sedih dari setiap perjalanan yang tidak seindah ekspektasi.
15. Sahabat seperjuangan semasa kuliah Amalia Nurul Rahmawati, Salma Dara Canita, Dwika Pangestuti, Jenita Lusiana Lukmawati, Lathifah Putri Aresti, dan Sarah Rosniani yang selalu mendukung, menemani, dan berbagi cerita indah selama masa perkuliahan. Terima kasih telah menjadi rekan yang selalu ada, selalu peduli, selalu menerima dan mengerti penulis dari awal mahasiswa baru hingga saat ini. Semoga pertemanan ini tetap erat meski masa kuliah telah usai.
16. Himakom yang telah memberikan wadah untuk penulis mengasah *soft skill*, menambah banyak wawasan, pengalaman, pertemanan dan pengetahuan yang sebelumnya belum pernah didapatkan. Himakom juga membantu penulis mengembangkan berbagai keterampilan yang awalnya diragukan mampu dikuasai, namun ternyata justru menjadi kekuatan yang membawa penulis lebih percaya diri dan berkembang secara personal maupun profesional.
17. Teman-teman KKN terkompak dan terlucu, Ade, Andi, Decy, Vanesa, Chika dan Revi yang selalu memberikan dukungan dan semangat kepada penulis. Terima kasih atas kesan, canda, tawa, dan cerita seru yang kalian berikan semasa KKN dan setelahnya. Kebersamaan kita bukan hanya memperluas pengalaman, tetapi meninggalkan kenangan yang akan terus penulis ingat sebagai salah satu momen terbaik dalam perjalanan ini.
18. Keluarga Ilmu Komputer 2021 yang telah memberikan pengalaman berharga. Terima kasih telah menjadi rekan kelompok, rekan diskusi, dan rekan yang

berjuang selama menjalankan studi di Jurusan Ilmu Komputer Universitas Lampung.

19. Seluruh pihak yang telah membantu dan memberikan dukungan, baik secara langsung maupun tidak langsung selama perkuliahan hingga penyelesaian skripsi ini.
20. Dan yang terakhir, kepada diri saya sendiri. Cely Febita Ilham. Terima kasih atas segala semangat dan kerja kerasnya, walau sering kali merasa putus asa atas apa yang diusahakan dan belum berhasil, namun terima kasih tetap menjadi manusia yang selalu mau berusaha dan tidak lelah mencoba. Terima kasih karena telah memutuskan tidak menyerah sesulit apapun proses penyusunan skripsi ini, serta Selamat karena telah menyelesaikannya dengan sebaik dan semaksimal mungkin.

Penulis menyadari bahwa penyusunan skripsi ini masih jauh dari kata sempurna. Namun, penulis sangat mengharapkan skripsi ini dapat bermanfaat bagi para civitas akademik Universitas Lampung, khususnya mahasiswa Ilmu Komputer.

Bandar Lampung, 2 Juni 2025



Cely Febita Ilham
NPM. 2117051022

DAFTAR ISI

	Halaman
DAFTAR ISI	xvi
DAFTAR TABEL	xx
DAFTAR GAMBAR	xxii
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	5
1.3 Batasan Masalah	5
1.4 Tujuan	6
1.5 Manfaat	6
II. TINJAUAN PUSTAKA	9
2.1 Penelitian Terdahulu	9
2.2 Sperma	10
2.3 Motilitas	11
2.4 Klasifikasi	12
2.5 Pengolahan Citra Digital	13
2.6 Pengolahan Video Digital	14
2.7 Citra <i>Resize</i>	15
2.8 Deteksi Objek	15
2.9 <i>Tracking</i> Objek	15
2.10 <i>Computer Vision</i>	16
2.11 <i>Artificial Intelligence</i>	16
2.12 <i>Machine Learning</i>	17
2.13 <i>Deep Learning</i>	18
2.14 <i>Convolutional Neural Network (CNN)</i>	20
2.14.1 <i>Convolution Layer</i>	20

2.14.2 <i>Rectified Linear Unit (ReLU)</i>	22
2.14.3 <i>Pooling Layer</i>	24
2.14.4 <i>Fully Connected Layer</i>	25
2.15 <i>You Only Look Once (YOLO)</i>	27
2.16 <i>You Only Look Once V8 (YOLOv8)</i>	30
2.16.1 <i>Convolution Layer</i>	33
2.16.2 Modul C2f	34
2.16.3 Modul <i>Spatial Pyramid Pooling Fast (SPPF)</i>	35
2.16.4 <i>Anchor Free Detection</i>	37
2.16.4 <i>Non-Maximum Suppression (NMS)</i>	37
2.17 <i>Hyperparameter</i>	38
2.18 <i>Confusion Matrix</i>	39
2.19 <i>Mean Average Precision (MAP)</i>	43
2.20 ByteTrack	44
2.21 Perhitungan Titik Tengah (<i>Centroid</i>).....	45
2.22 Jarak <i>Euclidean</i>	46
2.23 Kecepatan	47
2.24 Analisis Data	48
2.25 Pembersihan Data	48
2.26 Penyaringan Data	49
2.27 Perhitungan <i>Error</i> dan Akurasi	49
III. METODE PENELITIAN	51
3.1 Waktu dan Tempat	51
3.1.1 Tempat Penelitian	51
3.1.2 Waktu Penelitian	51
3.2 Alat Pendukung	53
3.3 Tahapan Penelitian	54
3.3.1 Pengumpulan Dataset	55
3.3.2 <i>Pre-Processing</i>	55
3.3.3 Pembagian Dataset	56
3.3.4 <i>Training Model</i>	56
3.3.5 <i>Input Video</i>	58

3.3.6 <i>Tracking</i> ByteTrack	59
3.3.7 Perhitungan Jarak <i>Euclidean</i>	60
3.3.8 Perhitungan Kecepatan	60
3.3.9 Analisis Hasil Perhitungan Kecepatan	61
3.3.10 Evaluasi Hasil Klasifikasi	62
IV. HASIL DAN PEMBAHASAN	63
4.1 Pengumpulan Data	63
4.2 <i>Pre-Processing</i>	64
4.2.1 Ekstraksi Video ke <i>Image</i>	64
4.2.2 <i>Resize Image</i>	65
4.2.3 Anotasi Data	66
4.3 Pembagian Dataset	67
4.4 <i>Hyperparameter Tuning</i>	69
4.5 <i>Training Model</i>	70
4.6 Hasil <i>Training</i> dan Evaluasi Model YOLOv8	73
4.7 <i>Tracking</i> Objek	84
4.7.1 Parameter ByteTrack	84
4.7.2 Model YOLOv8	85
4.7.3 File Inputan Video Sperma	86
4.7.4 Kode Program <i>Tracking</i>	86
4.8 Hasil <i>Tracking</i> Objek	87
4.8.1 Hasil <i>Tracking</i> dalam Bentuk File	88
4.8.2 Hasil <i>Tracking</i> dalam Bentuk Visual	88
4.9 Implementasi Perhitungan Jarak <i>Euclidean</i>	90
4.10 Perhitungan Kecepatan Gerak Objek	96
4.11 Analisis Hasil Perhitungan Kecepatan dan Klasifikasi Gerak Objek ...	97
4.11.1 Analisis Kecepatan Video Satu Durasi 5 Detik	97
4.11.2 Analisis Kecepatan Video Dua Durasi 10 Detik	108
4.11.3 Analisis Kecepatan Video Tiga Durasi 5 Detik	118
4.12 Perhitungan Manual	132
4.13 Pengujian Hasil Klasifikasi Objek	136
4.13.1 Pengujian Video Satu Durasi 5 Detik	136

4.13.2 Pengujian Video Dua Durasi 10 Detik.....	139
4.13.3 Pengujian Video Tiga Durasi 5 Detik.....	141
4.14 Analisis Hasil Performa Program	145
V. KESIMPULAN DAN SARAN	150
5.1 Kesimpulan	150
5.2 Saran	151
DAFTAR PUSTAKA	152

DAFTAR TABEL

Tabel	Halaman
Tabel 1. Penelitian Terdahulu	8
Tabel 2. <i>Confusion Matrix</i>	40
Tabel 3. Jadwal Kegiatan	52
Tabel 4. Inputan Video Sperma	58
Tabel 5. <i>Hyperparameter Tuning</i>	69
Tabel 6. Hasil Pengujian <i>Learning Rate</i> pada 50 <i>Epoch</i>	73
Tabel 7. Hasil Analisis <i>Tracking</i> ByteTrack pada <i>Video Input</i>	87
Tabel 8. Representasi <i>Output Tracking</i>	88
Tabel 9. Representasi Hasil Perhitungan Jarak	95
Tabel 10. Representasi Hasil Perhitungan Kecepatan Objek	96
Tabel 11. Objek Video Satu Terdeteksi oleh ByteTrack	98
Tabel 12. Objek Video Satu Setelah <i>Post-Processing</i>	100
Tabel 13. Statistik Deskriptif Video Satu	105
Tabel 14. Rata-Rata Kecepatan Tiap Objek	106
Tabel 15. Objek Video Dua Terdeteksi oleh ByteTrack	109
Tabel 16. Objek Video Dua Setelah <i>Post-Processing</i>	110
Tabel 17. Statistik Deskriptif Video Dua	115
Tabel 18. Rata-Rata Kecepatan Tiap Objek	116
Tabel 19. Objek Video Tiga Terdeteksi oleh ByteTrack	119
Tabel 20. Objek Video Tiga Setelah <i>Post-Processing</i>	122
Tabel 21. Statistik Deskriptif Video Tiga	127
Tabel 22. Rata-Rata Kecepatan Tiap Objek	128
Tabel 23. Koordinat <i>Bounding Box</i> Manual	134
Tabel 24. Pengujian Video Satu Durasi Lima Detik	138
Tabel 25. Pengujian Video Dua Durasi Sepuluh Detik	140

Tabel 26. Pengujian Video Tiga Durasi Lima Detik	143
Tabel 27. Perbandingan <i>Post-Processing</i> Ketiga Video	146
Tabel 28. Hasil Klasifikasi Motilitas Sperma pada Ketiga Video	148
Tabel 29. Tabel Perbandingan Akurasi dan <i>Error</i> Ketiga Video	149

DAFTAR GAMBAR

Gambar	Halaman
Gambar 1. Morfologi Sperma	10
Gambar 2. Klasifikasi Multi-Kelas	12
Gambar 3. Jenis Citra Digital	13
Gambar 4. Representasi Video Digital	14
Gambar 5. Keterkaitan AI, ML, dan DL	17
Gambar 6. Ilustrasi <i>Deep Learning</i>	19
Gambar 7. Ilustrasi CNN	20
Gambar 8. Ilustrasi Proses <i>Convolutional Layer</i>	21
Gambar 9. Grafik Fungsi ReLU	23
Gambar 10. Grafik Fungsi SiLU	23
Gambar 11. Contoh <i>Max Pooling Layer</i>	24
Gambar 12. Contoh <i>Average Pooling Layer</i>	25
Gambar 13. Ilustrasi <i>Fully Connected Layer</i>	26
Gambar 14. YOLO Mendeteksi Objek	27
Gambar 15. Proses YOLO Mendeteksi Objek	28
Gambar 16. Model Mendeteksi Objek	29
Gambar 17. Arsitektur Jaringan	30
Gambar 18. Perbandingan YOLOv5 s/d YOLOv8	31
Gambar 19. Arsitektur YOLOv8	32
Gambar 20. <i>Convolution Layer</i>	33
Gambar 21. Modul C2f	34
Gambar 22. Modul SPPF	36
Gambar 23. Anchor <i>Free Detection</i> YOLOv8	37
Gambar 24. Ilustrasi <i>Non-Maximum Suppression (NMS)</i>	38
Gambar 25. Proses Pelacakan Objek dengan ByteTrack.....	44
Gambar 26. Alur Penelitian	54

Gambar 27. Dataset Visem	64
Gambar 28. Hasil Ekstraksi <i>Image</i>	65
Gambar 29. Hasil <i>Resize Image</i>	66
Gambar 30. Anotasi Data menggunakan LabelImg	66
Gambar 31. File Hasil Anotasi	67
Gambar 32. File YAML	71
Gambar 33. Grafik Perbandingan Waktu <i>Training</i>	74
Gambar 34. Perbandingan Presisi, <i>Recall</i> dan <i>F1 Score</i>	75
Gambar 35. Grafik Hasil Proses <i>Training 50 Epoch & Learning Rate 0.002</i>	76
Gambar 36. Grafik Hasil Proses <i>Training 50 Epoch & Learning Rate 0.0002</i> ...	77
Gambar 37. Grafik Hasil Proses <i>Training 100 Epoch & Learning Rate 0.002</i> ...	79
Gambar 38. Grafik Hasil Proses <i>Training 100 Epoch & Learning Rate 0.0002</i> .	80
Gambar 39. Grafik Hasil Proses <i>Training 150 Epoch & Learning Rate 0.002</i> ...	81
Gambar 40. Grafik Hasil Proses <i>Training 150 Epoch & Learning Rate 0.0002</i> .	82
Gambar 41. Hasil Pengujian Model YOLOv8	83
Gambar 42. Parameter ByteTrack	84
Gambar 43. Model YOLOv8	85
Gambar 44. Deteksi tanpa <i>Tracking</i>	89
Gambar 45.(A) <i>Frame</i> Pertama Hasil <i>Tracking</i> (B) <i>Frame</i> Kedua Hasil <i>Tracking</i>	89
Gambar 46. Ilustrasi Perhitungan <i>Centroid</i>	91
Gambar 47. Ilustrasi Perhitungan Jarak	92
Gambar 48. Ilustrasi Menentukan <i>Centroid</i> Terdekat	93
Gambar 49. (A) <i>Frame</i> 22 (B) <i>Frame</i> 23 (C) <i>Frame</i> 24	98
Gambar 50. Representasi Objek Non Sperma Terdeteksi Video Satu	100
Gambar 51. Objek ID 26 Masuk <i>Frame</i>	101
Gambar 52. (A) ID 3 di <i>Frame</i> Terakhir (B) ID 3 Telah Keluar <i>Frame</i>	102
Gambar 53. Grafik dengan Nilai Kecepatan Ekstrem Video Satu	103
Gambar 54. Grafik Kecepatan Sperma Tanpa Nilai Ekstrem Video Satu	104
Gambar 55. Grafik Klasifikasi Berdasarkan Kecepatan	107
Gambar 56. Visualisasi Hasil Klasifikasi Video Satu	107
Gambar 57. Representasi Hasil <i>Tracking</i> Video Dua	110
Gambar 58. (A)Objek ID 11 Masuk <i>Frame</i> (B)Objek ID 32 Masuk <i>Frame</i>	112

Gambar 59. (A) ID 3 di <i>Frame</i> Terakhir (B) ID 3 Keluar <i>Frame</i>	113
Gambar 60. Grafik Video Dua dengan Nilai Kecepatan Ekstrem	113
Gambar 61. Grafik Kecepatan Sperma Tanpa Nilai Ekstrem Video Dua	114
Gambar 62. Grafik Klasifikasi Berdasarkan Kecepatan	117
Gambar 63. Visualisasi Hasil Klasifikasi Video Dua	118
Gambar 64. (A) <i>Frame</i> 11 (B) <i>Frame</i> 12 (C) <i>Frame</i> 14	120
Gambar 65. Representasi Objek Non Sperma Terdeteksi Video Tiga	121
Gambar 66. (A) Objek ID 59 Masuk <i>Frame</i> (B) Objek ID 75 Masuk <i>Frame</i>	123
Gambar 67. (A) ID 41 di <i>Frame</i> Terakhir (B) ID 41 Telah Keluar <i>Frame</i>	124
Gambar 68. Grafik Video Tiga dengan Nilai Kecepatan Ekstrem	125
Gambar 69. Grafik Kecepatan Sperma Tanpa Nilai Ekstrem Video Tiga	126
Gambar 70. Grafik Klasifikasi Berdasarkan Kecepatan	131
Gambar 71. Visualisasi Hasil Klasifikasi Video Tiga	131
Gambar 72. Representasi <i>Frame</i> Citra Deteksi Objek YOLOv8	132
Gambar 73. Representasi Koordinat <i>Bounding Box</i>	132
Gambar 74. Konversi Koordinat dalam 640x640	133
Gambar 75. Perbandingan Metode Manual dengan Program Video Satu	136
Gambar 76. Perbandingan Metode Manual dengan Program Video Dua	139
Gambar 77. Perbandingan Metode Manual dengan Program Video Tiga	142

I. PENDAHULUAN

1.1 Latar Belakang

Reproduksi merupakan proses biologis pada makhluk hidup untuk menghasilkan keturunan dan melestarikan jenisnya agar tidak punah. Reproduksi pada manusia dilakukan secara seksual yang diawali dengan proses fertilisasi, yaitu proses bersatunya sel sperma dengan ovum untuk membentuk sebuah zigot. Keberhasilan proses fertilisasi pada laki-laki tergantung pada kualitas sel sperma. Sel sperma yang memiliki kualitas tidak baik, kemungkinan terburuk akan terjadi infertilitas. Infertilitas merupakan suatu gangguan dari sistem reproduksi, yaitu ketidakmampuan untuk hamil secara alami. Masalah infertilitas yang semakin cepat diketahui maka akan semakin baik, sehingga dapat mencegah dan menghindari ketidakesuburan (Tamsir *et al.*, 2023). Infertilitas pria dapat diatasi dengan mengubah kebiasaan, seperti berhenti merokok, menghindari alkohol, mengurangi konsumsi makanan tidak sehat dan mengurangi stress. Keberhasilan fertilisasi dengan indeks prediksi tertinggi terdapat pada kualitas sperma yang mempunyai daya hidup tinggi, morfologi yang normal dan motilitas yang progresif. Motilitas progresif merupakan kemampuan gerak maju sperma di dalam lingkungan zat cairnya untuk membantu sperma menembus sel pelindung yang mengelilingi ovum, yang akan berpengaruh pada terjadinya kehamilan (Shari, 2022).

Kualitas sperma diukur berdasarkan volume air mani, jumlah sperma, kepadatan sperma, pergerakan sperma (motilitas), kecepatan sperma, bentuk sperma (morfologi) dan viabilitas sperma. Parameter penting dalam menilai kualitas semen manusia yaitu motilitas sperma. Motilitas sperma tidak hanya mencakup kecepatan gerak sperma, tetapi juga persentasi sperma yang bergerak. Sperma

di klasifikasikan dalam tiga kelas berdasarkan kecepatannya yaitu total motil (semua spermatozoa yang bergerak lebih dari 10 $\mu\text{m}/\text{detik}$), progresif motil (semua spermatozoa yang bergerak maju lebih dari 20 $\mu\text{m}/\text{detik}$) dan local motil/non-progresif (semua spermatozoa yang bergerak pada 10–20 $\mu\text{m}/\text{detik}$) (Susilawati, 2011). *World Health Organization*(WHO) menyatakan sperma yang bergerak progresif memiliki persentase bergerak lebih dari 32% dan sperma yang bergerak total (progresif dan non-progresif) memiliki persentase 40% dalam volume ejakulasi yang normal (Cooper *et al.*, 2009).

Motilitas sperma merupakan salah satu faktor penting dalam menentukan tingkat kesuburan pria. Penilaian motilitas sperma umumnya dilakukan secara manual melalui pengamatan mikroskopis, yang memiliki keterbatasan dalam hal kecepatan dan objektivitas. Teknologi seperti *Computer-Aided Sperm Analysis* (CASA) telah digunakan untuk membantu menganalisis pergerakan sperma secara otomatis, tetapi perangkat ini memiliki kekurangan yaitu dari harga yang mahal dan akses yang terbatas. Cara kerja dalam *software* CASA tidak terbuka untuk umum, sehingga banyak penelitian yang mempelajari cara kerja dalam mengukur tingkat kesehatan sperma secara akurat, murah dan cepat. *Deep learning* menawarkan pendekatan yang lebih efisien dalam menentukan kecepatan gerak sperma. Metode ini mampu untuk mendeteksi objek dengan baik tanpa membutuhkan waktu yang lama dan mampu digunakan sebagai basis dalam menghitung kecepatan gerak sperma dengan lebih presisi dan minim kesalahan, yang membuat proses pengambilan keputusan menjadi lebih cepat dan objektif. Program berbasis *deep learning* dapat dijalankan pada perangkat dengan spesifikasi umum (*low-end device*) dan dilakukan di berbagai lokasi, sehingga lebih fleksibel tanpa bergantung pada teknologi tertentu.

Perkembangan teknologi yang semakin maju mendorong algoritma *machine learning* berkembang dengan lebih pesat, salah satunya dalam pemrosesan citra untuk mendeteksi sebuah objek. Deteksi objek membutuhkan struktur jaringan yang baik guna bekerja secara akurat dan cepat. Algoritma *You Only Look Once* (YOLO) diperkenalkan sebagai pendekatan yang inovatif yang mampu

memprediksi fitur map pada objek secara langsung. *You Only Look Once* (YOLO) merupakan detektor objek yang cepat dan akurat serta mampu mendeteksi objek secara sederhana dengan hanya melihat sekali pada gambar untuk mendeteksinya. Metode lain seperti R-CNN berbeda dengan YOLO yang melihat bagian gambar secara terpisah, YOLO mempertimbangkan keseluruhan gambar saat melakukan prediksi, sehingga dapat mengurangi kesalahan deteksi objek pada latar belakang gambar. Deteksi objek pada YOLO menjadi masalah regresi dimana model akan langsung memprediksi koordinat *bounding box* objek dan probabilitas kelasnya menggunakan jaringan konvolusi secara bersamaan. YOLO tidak hanya mendeteksi gambar tetapi juga mampu sebagai pelacakan pada objek bergerak dan berubah bentuk dalam gambar (Redmon *et al.*, 2016). YOLO telah berkembang dari versi pertama (YOLO-v1) hingga berbagai versi lanjutan, termasuk YOLO-v8 yang dirilis oleh Ultralytics pada tahun 2023 yang dikenal dengan peningkatan kemampuannya dalam mendeteksi secara akurat dan cepat serta mampu mendeteksi objek yang lebih kecil (Nusman *et al.*, 2024).

Penelitian terkait model YOLOv8 dalam mendeteksi objek yang lebih kecil telah dilakukan pada dataset video sperma menggunakan model YOLO (*you only look once*) versi 4 dengan mengklasifikasikan sperma manusia dan non sperma berdasarkan morfologinya. Hasil dari penelitian tersebut menunjukkan bahwa YOLOv4 mampu mengidentifikasi objek dengan nilai *average precision* yang tinggi untuk objek sperma maupun non-sperma (Aristoteles *et al.*, 2022). Penelitian ini merupakan pengembangan dari penelitian sebelumnya, dengan menggunakan model YOLO dengan versi 8 (YOLOv8) yang memiliki akurasi dan keandalan deteksi yang lebih tinggi dibanding model sebelumnya (Fahim & Hasan, 2023), serta lebih efisien dari segi waktu pelatihan (Marçal *et al.*, 2024). Model ini dijadikan sebagai dasar untuk perhitungan kecepatan gerak sperma dan mengklasifikasikannya ke dalam kategori progresif, non-progresif dan *immotile*.

Model YOLOv8 dalam penelitian ini berfungsi sebagai detektor objek yang akan diintegrasikan dengan algoritma ByteTrack yaitu algoritma pelacakan objek, karena hasil pelacakan objek yang akan digunakan dalam perhitungan kecepatan

pergerakan sperma pada video. Hasil deteksi tanpa adanya pelacakan objek, menjadikan objek tidak saling mengenali apakah objek di *frame* saat ini merupakan objek yang sama di *frame* berikutnya. *Tracker* yang digunakan dalam penelitian yaitu ByteTrack, metode dalam *Multi Object Tracking* (MOT) yang mampu untuk melacak objek *high-confidence* dan masih mempertimbangkan objek yang *low-confidence*, sehingga pelacakan menjadi lebih akurat dan tidak kehilangan objek (Zhang *et al.*, 2021). Penelitian (Rezky *et al.*, 2023) terkait deteksi kecelakaan lalu lintas pada CCTV menggunakan YOLOv8 dengan melakukan *object tracking* oleh ByteTrack untuk melacak pergerakan objek pada setiap *frame* dengan memisahkan kotak deteksi menjadi dua kelompok, kotak dengan skor tinggi dan kotak dengan skor rendah.

ByteTrack digunakan untuk melacak sperma yang telah terdeteksi oleh YOLOv8, sehingga memungkinkan perhitungan jarak perpindahan objek antar *frame*. Informasi perpindahan objek tersebut akan digunakan untuk menghitung kecepatan gerak sperma dalam satuan mikrometer per detik $\mu\text{m}/\text{detik}$, yang kemudian menjadi dasar untuk klasifikasi motilitas sperma ke dalam tiga kelas, yaitu progresif (kecepatan $>20 \mu\text{m}/\text{detik}$), non-progresif (kecepatan $10\text{-}20 \mu\text{m}/\text{detik}$) dan *immotile* (kecepatan $< 10 \mu\text{m}/\text{detik}$). Penelitian serupa diterapkan oleh (Valiuškaitė *et al.*, 2020) dalam penelitiannya yang menghitung kecepatan gerak sperma berdasarkan deteksi kepala sperma, pelacakan pergerakan antar *frame* dan perhitungan jarak centroid dalam satuan mikrometer (μm). Penelitian tersebut berhasil mencapai akurasi sebesar 91.77% dengan MAE 2.92 dan mendapat prediksi antara kecepatan sperma aktual dengan prediksi program sebesar 0.969 pada dataset VISEM. Hasil ini menunjukkan bahwa pendekatan pelacakan dan perhitungan kecepatan berbasis deteksi objek dapat diterapkan secara efektif pada sel manusia, termasuk dalam konteks motilitas sperma.

Model YOLOv8 yang diintegrasikan dengan algoritma ByteTrack diusulkan dalam penelitian ini karena mampu mendeteksi objek kecil yang bergerak cepat, seperti sperma, serta melakukan pelacakan secara *real-time* dalam data video. Pendekatan ini diharapkan dapat menghasilkan klasifikasi motilitas sperma

manusia berdasarkan kecepatan geraknya dengan lebih akurat dan efisien. Klasifikasi motilitas yang dilakukan berdasarkan tiga kategori tertentu, yaitu progresif (kecepatan $>20 \mu\text{m/detik}$), non-progresif (kecepatan $10\text{-}20 \mu\text{m/detik}$) dan *immotile* (kecepatan $<10 \mu\text{m/detik}$). Motilitas yang semakin progresif mengurangi adanya infertilitas, sehingga penelitian ini memiliki kontribusi dalam memanfaatkan teknologi *deep learning* dalam kesehatan reproduksi. Penelitian terkait klasifikasi motilitas sperma manusia berbasis YOLOv8 yang terintegrasi dengan ByteTrack belum pernah dilakukan sebelumnya, untuk itu penelitian ini dapat dijadikan sebagai referensi di masa depan dengan algoritma atau data yang serupa. Hasil dari penelitian ini berupa evaluasi model YOLOv8 menggunakan mAP, presisi, *recall* dan *f1-Score*, analisis hasil perhitungan program, klasifikasi kecepatan gerak sperma berdasarkan rata-rata kecepatan di seluruh *frame* dalam durasi video, serta evaluasi dari hasil klasifikasi motilitas sperma menggunakan program dengan klasifikasi perhitungan manual.

1.2 Rumusan Masalah

Rumusan masalah yang didapatkan pada penelitian ini yaitu bagaimana klasifikasi motilitas sperma berdasarkan kecepatan gerak sperma menggunakan model YOLOv8 yang terintegrasi dengan algoritma pelacakan ByteTrack?

1.3 Batasan Masalah

Batasan masalah yang digunakan sebagai fokus dalam tujuan penelitian ini sebagai berikut :

1. Penelitian menggunakan Bahasa Pemrograman Python melalui *tools* Google Colab untuk mengimplementasikan algoritma YOLOv8 dan algoritma perhitungan kecepatan gerak sperma.
2. Dataset yang digunakan adalah video sperma manusia berjumlah 85 video dengan durasi 2-7 menit, berekstensi AVI dari sumber dataset VISEM. Data ini digunakan sebagai inputan citra.

3. Penelitian dilakukan dengan menggunakan 1330 gambar dengan nama kelas *sperm* untuk proses pelatihan dengan menggunakan model YOLOv8.
4. Pengukuran motilitas sperma berdasarkan kecepatan geraknya terdapat tiga kelas yaitu progresif (kecepatan $>20 \mu\text{m}/\text{detik}$), non-progresif (kecepatan $10-20 \mu\text{m}/\text{detik}$) dan *immotile* (kecepatan $<10 \mu\text{m}/\text{detik}$).
5. Model YOLOv8 hanya digunakan sebagai detektor objek, sedangkan pelacakan dilakukan menggunakan ByteTrack.
6. Klasifikasi didapatkan dari proses perhitungan kecepatan gerak sperma berdasarkan koordinat *bounding box* dari *tracking* ByteTrack.
7. Video sperma yang digunakan sebagai sampel berdurasi 5 detik dan 10 detik yang akan dianalisis secara menyeluruh untuk mewakili performa program klasifikasi motilitas sperma.

1.4 Tujuan

Tujuan dari penelitian ini adalah:

1. Menerapkan dan mengevaluasi performa model YOLOv8 dalam mendeteksi sperma manusia berdasarkan dataset VISEM.
2. Menghitung kecepatan gerak sperma berdasarkan koordinat *bounding box* hasil *tracking* ByteTrack.
3. Mengklasifikasikan motilitas sperma ke dalam tiga kategori berdasarkan kecepatan geraknya serta menganalisis hasil klasifikasi kecepatan gerak sperma pada video.
4. Mengevaluasi akurasi klasifikasi motilitas sperma dengan membandingkan hasil perhitungan program dan manual.

1.5 Manfaat

Manfaat dari penelitian yang dilakukan sebagai berikut:

1. Menghasilkan implementasi model YOLOv8 dalam mendeteksi sperma.

2. Menentukan klasifikasi motilitas sperma berdasarkan kecepatan gerak sperma yang terdeteksi menggunakan YOLOv8 dan ByteTrack.
3. Mendukung perkembangan teknologi dalam analisis sperma dengan pendekatan *deep learning*.
4. Menjadi referensi bagi penelitian di masa depan dalam pengembangan metode deteksi dan klasifikasi sperma dengan model atau algoritma lain.
5. Menjadi referensi bagi praktisi kesehatan terkait perkembangan teknologi untuk mendeteksi sel manusia.

II. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu yang berkaitan dengan analisis motilitas sperma telah banyak dilakukan dengan berbagai metode yang di antaranya menggunakan algoritma *machine learning* dan *deep learning*. Penelitian yang relevan dan menjadi landasan dalam penelitian ini terdapat pada Tabel 1 berikut.

Tabel 1. Penelitian Terdahulu

No.	Penulis	Judul	Objek	Pokok Pembahasan
1.	(Aristoteles <i>et al.</i> , 2022)	<i>Identification of Human Sperm based on Morphology Using the You Only Look Once Version 4 Algorithm</i>	Sperma dan Non Sperma	Penelitian tersebut mendapati nilai AP (<i>Average Precision</i>) 90.31% untuk objek sperma dan 68.13% untuk objek non sperma, serta di dapatkan hasil pelatihan dengan nilai mAP (<i>Mean Average Precision</i>) sejumlah 79.58%.
2.	(Valiuškaitė <i>et al.</i> , 2020)	<i>Deep Learning Based Evaluation of Spermatozoid Motility For Artificial Insemination</i>	Sperma	Penelitian ini mendeteksi kepala sperma mencapai 91.77%, dengan MAE 2.92 dan prediksi antara kecepatan gerak sperma aktual dengan prediksi program sebesar 0.969.

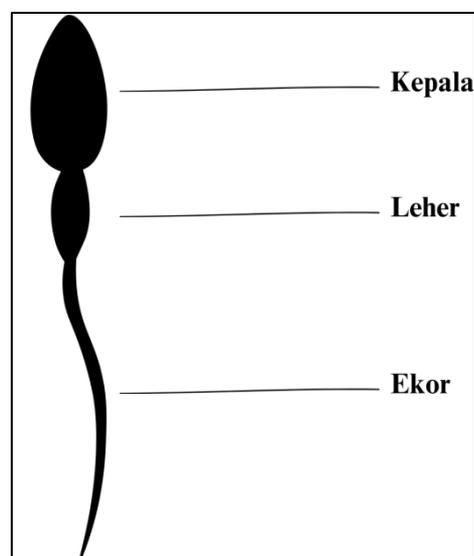
No.	Penulis	Judul	Objek	Pokok Pembahasan
3.	(Rezky <i>et al.</i> , 2023)	Deteksi Kecelakaan Lalu Lintas dan Tidak Otomatis Pada Rekaman CCTV Indonesia Menggunakan <i>Deep Learning</i>	Kendaraan (Kecelakaan dan Tidak Kecelakaan)	Melacak objek terdeteksi menggunakan ByteTrack dan mengklasifikasikan kelas menggunakan model MobileNetV2 dengan akurasi 0.89, dan beban komputasi rata-rata 21 ms/frame.
4.	Penelitian yang akan dilakukan	Klasifikasi Motilitas Pada Sperma Berdasarkan Kecepatan Gerak Sperma Menggunakan Model YOLOv8	Sperma	Klasifikasi motilitas sperma berdasarkan kecepatan gerak sperma yang terdeteksi.

Tabel 1 menyajikan beberapa literatur terdahulu yang memiliki keterkaitan dengan penelitian ini. Penelitian (Aristoteles *et al.*, 2022), terkait pengenalan objek sperma dan non-sperma dilakukan menggunakan algoritma YOLOv4 dengan fokus utama pada morfologi. Penelitian tersebut menghasilkan nilai AP sebesar 90,31% untuk objek sperma dan 68,13% untuk objek non-sperma, serta nilai mAP sebesar 79,58%. Penelitian ini belum membahas motilitas sperma yang berkaitan dengan kecepatan gerak. Penelitian oleh (Valiuškaitė *et al.*, 2020) mengkaji motilitas sperma menggunakan pendekatan *deep learning*, dengan hasil deteksi kepala sperma mencapai 91,77% dan korelasi prediksi kecepatan gerak sebesar 0,969. Model yang digunakan tidak memanfaatkan algoritma deteksi *real-time* seperti YOLO, tidak menggunakan algoritma *tracker* seperti ByteTrack dan tidak mengklasifikasikan kecepatan gerak sperma ke dalam kategori motilitas. Penelitian (Rezky *et al.*, 2023) menerapkan algoritma ByteTrack untuk melacak

hasil deteksi objek kendaraan, namun objek penelitian tidak berfokus pada sperma, serta tidak tertuju pada aspek kecepatan gerak objek secara spesifik. Penelitian ini dikembangkan untuk mengisi kekosongan dalam literatur terkait klasifikasi motilitas sperma berdasarkan kecepatan gerak. Pendekatan menggunakan model YOLOv8 dipilih karena mampu mendeteksi objek kecil yang bergerak cepat, seperti sperma dengan *real-time*. Integrasi dengan algoritma ByteTrack digunakan untuk pelacakan objek dan menjadi dasar dalam perhitungan kecepatan gerak sperma. Penelitian ini diharapkan memberikan kontribusi dalam klasifikasi motilitas sperma berdasarkan parameter kecepatan, yang belum banyak dikaji dalam penelitian sebelumnya.

2.2 Sperma

Sperma adalah sel yang dihasilkan dari testis di sistem reproduksi pria melalui *tubulus seminiferous*. Sperma dibentuk pada suhu di bawah suhu tubuh ($+ 35^{\circ}\text{C}$) dan menghasilkan sekitar 200 hingga 250 juta spermatozoa. (Suhartini, 2012).



Gambar 1. Morfologi Sperma.

Gambar 1 menunjukkan bahwa sel sperma dibedakan berdasarkan kepala, leher dan ekor. Kepala sperma berbentuk lonjong pipih berisi inti (*nucleus*), leher yang menghubungkan kepala dengan bagian tengah dan ekor yang memiliki aksi

bergetar sehingga sperma dapat bergerak dengan cepat. Panjang ekor sperma sepanjang kira-kira 10 kali ukuran kepala (Suhartini, 2012).

2.3 Motilitas

Motilitas adalah kemampuan pergerakan sel, termasuk sel sperma. Penelitian yang dilakukan oleh Mortimer motilitas adalah salah satu faktor penting sebagai penentu kualitas sperma pada analisa sperma dan penentu keberhasilan sperma menembus *mukus serviks* serta menentukan keberhasilan proses fertilisasi (Djaelani *et al.*, 2009).

Motilitas sperma merupakan kemampuan pergerakan sperma yang dapat dinilai baik pergerakannya secara individu maupun *massa*. Pergerakan sperma ini dipengaruhi oleh berbagai macam faktor. Motilitas sperma memberikan pengaruh pada tingkat fertilitas sperma dan harus memiliki kemampuan bergerak yang lincah dan progresif sehingga dapat membuahi ovum betina (Jatra *et al.*, 2022). Motilitas yang baik ditandai dengan kemampuan kepala sperma bergerak cepat dalam satu arah, sementara motilitas buruk ketika kepala sperma bergerak lambat dan tidak teratur (Valiuškaite *et al.*, 2020).

1. Motilitas Progresif

Motilitas progresif adalah pergerakan pada sperma yang bergerak secara cepat dan terarah. Motilitas progresif bergerak dengan kecepatan $>20 \mu\text{m/detik}$ (Susilawati, 2011).

2. Motilitas Non-Progresif

Motilitas sperma non-progresif adalah pergerakan pada sperma yang bergerak secara tidak terarah dan lambat. Motilitas non-progresif bergerak dengan kecepatan antara $10 - 20 \mu\text{m/detik}$ (Susilawati, 2011).

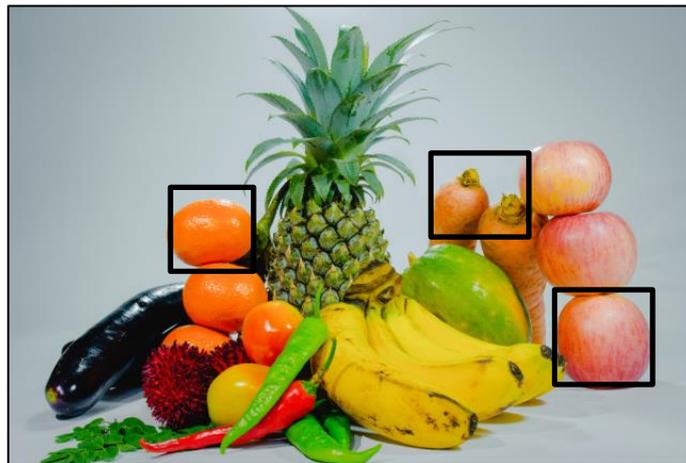
3. *Immotile*

Immotile adalah sperma yang memiliki kecepatan $<10 \mu\text{m/detik}$ atau dianggap sebagai sperma yang tidak bergerak (Susilawati, 2011).

2.4 Klasifikasi

Klasifikasi merupakan cara pengelompokan data sesuai ciri-ciri atau karakteristik pada data tersebut. Klasifikasi dapat dilakukan dengan banyak cara dalam prosesnya, baik secara manual ataupun dengan bantuan teknologi. Klasifikasi dibuat dari sekumpulan data latih yang telah ditentukan untuk kemudian seluruh fitur di kelompokkan ke dalam kelas yang sesuai (Rahayuningsih *et al.*, 2018). Penelitian terkait sperma yang dilakukan, klasifikasi diperuntukan pada citra. Klasifikasi citra merupakan proses pengelompokan piksel pada suatu citra ke dalam sejumlah *class* (kelas) dan setiap kelas dapat menggambarkan suatu entitas dengan ciri-ciri tertentu. Tujuan dari klasifikasi citra adalah menghasilkan peta tematik yang mana pada satu warna mewakili suatu objek tertentu (Ardiansyah *et al.*, 2022).

Klasifikasi multi-kelas merupakan klasifikasi yang digunakan pada penelitian terkait. Klasifikasi multi-kelas adalah klasifikasi yang terdapat lebih dari dua kategori yang diklasifikasikan dalam sebuah citra dan citra tersebut tidak termasuk dalam lebih dari satu kategori. Klasifikasi multi-kelas pada akurasi merupakan kriteria evaluasi yang umum digunakan. (Bariyah *et al.*, 2021).



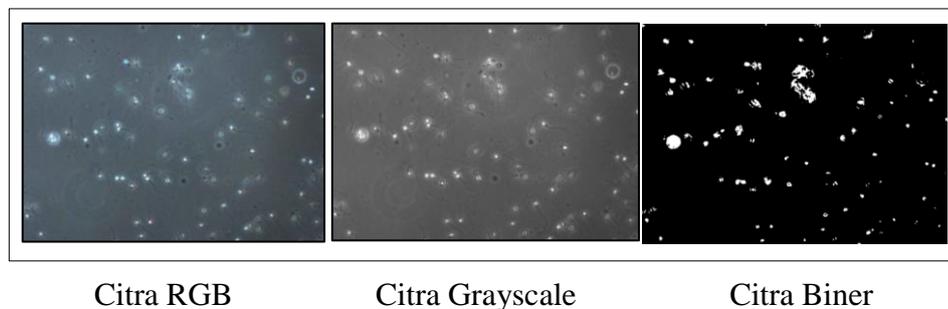
Gambar 2. Klasifikasi Multi-Kelas.

Gambar 2 merupakan ilustrasi dari klasifikasi multi kelas yang terlihat bahwa citra akan diklasifikasikan dalam empat jenis buah yang berbeda, dimana setiap citra

hanya terdiri dari satu objek yang masuk dalam satu jenis buah saja (Bariyah *et al.*, 2021).

2.5 Pengolahan Citra Digital

Pengolahan citra digital merupakan disiplin ilmu yang mempelajari teknik dalam mengolah citra, citra yang dimaksud adalah berupa gambar diam (foto) atau gambar yang bergerak (video). Digital adalah pengolahan citra yang dilakukan menggunakan komputer secara digital (Ratna, 2020). Pengertian lain terkait dengan pemrosesan dan pengolahan citra merupakan metode untuk melakukan beberapa operasi untuk mendapatkan citra yang disempurnakan dan mengekstrak beberapa informasi berguna di dalamnya. Citra digital adalah kumpulan titik yang dinamakan piksel (*pixel* atau *picture element*). Setiap piksel digambarkan sebagai satu kotak kecil yang memiliki koordinat posisi (Arnita *et al.*, 2023). Tiga jenis citra yang umum digunakan dalam pengolahan citra. Ketiga jenis citra tersebut yaitu, citra berwarna, citra berskala keabuan dan citra biner (Arnita *et al.*, 2023).



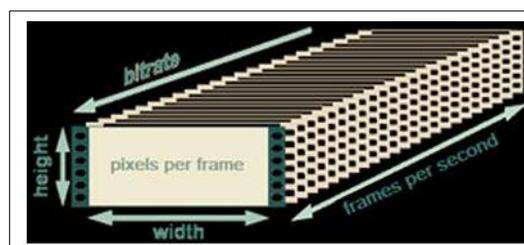
Gambar 3. Jenis Citra Digital.

Gambar 3 merupakan jenis citra digital yang umum digunakan dalam pengolahan citra. Citra berwarna dikenal dengan istilah *truecolor* atau citra RGB, merupakan jenis citra yang menyajikan warna dalam bentuk komponen R (*red/merah*), G (*green/hijau*), dan B (*blue/biru*) yang jika digabungkan akan membentuk suatu susunan warna yang luas. Citra *grayscale* merupakan citra yang memiliki gradasi hitam dan putih hingga menghasilkan warna abu, yang biasa dikenal dengan istilah *gray scale* atau *gray level*. Jenis gambar piksel 16 bergantung pada

intensitasnya. Intensitas berkisar antara 0 sampai dengan 255. Nilai 0 menyatakan hitam dan nilai 255 menyatakan putih, sedangkan citra biner atau yang dikenal dengan istilah dikenal dengan istilah *binary bilevel image* merupakan citra yang hanya dapat mengasumsikan dua kemungkinan nilai, 0 atau 1. Nilai 0 menyatakan warna hitam dan nilai 1 menyatakan warna putih. Gambar biner dinilai paling ‘murah’ karena menggunakan penyimpanan dan juga persyaratan pemrosesan yang paling sedikit. Contoh gambar biner adalah gambar garis, teks tercetak pada halaman putih, atau siluet (Arnita *et al.*, 2023).

2.6 Pengolahan Video Digital

Video merupakan pergerakan objek dalam waktu tertentu yang terdiri dari rangkaian *frame* (citra) yang direkam umumnya antara 25 hingga 30 *frame* per detik, dengan tujuan supaya mata manusia dapat menangkap gerakannya. Video digital adalah representasi informasi melalui serangkaian gambar yang diambil dan ditampilkan sesuai dengan sistem pemindaian standar, *frame rate*, dan ukuran gambar yang digunakan oleh teknologi video. Citra video digital dapat diambil menggunakan kamera video digital atau perangkat yang dilengkapi kamera video digital (Wardhani dan Madenda, 2016).



Gambar 4. Representasi Video Digital (Sholehurohman, 2021).

Teori matematis merepresentasikan video digital sebagai fungsi $f(m, n, t)$, dimana m menyatakan baris ke- m dari citra, n menyatakan pixel ke- n pada baris ke- m dari citra, dan t menyatakan *frame* dari citra pada waktu ke- t . Fungsi f menyatakan nilai intensitas dari warna (*pixel*) citra yang berada diposisi (m,n) pada waktu t (Sholehurrohman, 2021).

2.7 Citra *Resize*

Resize adalah proses mengubah dimensi sebuah citra digital. *Resize* dilakukan untuk menyelaraskan dimensi pada seluruh data citra dengan membuat citra memiliki dimensi yang serupa antara baris dan kolomnya (Djampi *et al.*, 2019). Citra *resize* merupakan proses untuk mengubah ukuran lebar dan tinggi sebuah citra. Proses citra *resize* menjadi penting karena ukuran yang terdapat pada citra terlalu besar dan tidak selalu sama, oleh karena itu agar tidak ada perbedaan ukuran dan mengganggu kinerja sistem untuk proses lebih lanjut (Rosnita, 2021).

2.8 Deteksi Objek

Deteksi objek adalah teknologi komputer yang terkait dengan visi komputer dan pemrosesan gambar yang berhubungan dengan mendeteksi *instance object semantic* dari kelas tertentu (seperti manusia, bangunan, atau mobil) dalam gambar atau video digital. Domain yang dideteksi dengan baik dari objek termasuk deteksi wajah dan deteksi pejalan kaki. Deteksi objek memiliki aplikasi di banyak bidang komputer, termasuk pengambilan gambar dan pengawasan video (Fauzi *et al.*, 2020).

2.9 Tracking Objek

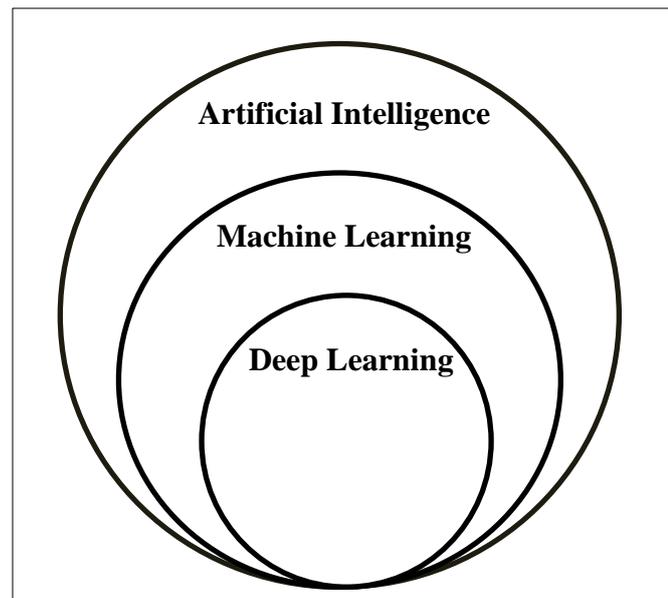
Pelacakan objek dilakukan setelah objek terdeteksi pada *frame* pertama yang dilakukan untuk mengikuti pergerakan objek pada *frame to frame*. Pelacakan objek dilakukan dengan proses perubahan posisi objek yang dihitung dengan membandingkan posisi *bounding box* objek pada *frame* saat ini dengan *frame* sebelumnya. *Tracking* objek dilakukan untuk memantau gerakan objek dari waktu ke waktu (Rezky *et al.*, 2023).

2.10 *Computer Vision*

Computer Vision merupakan sebuah sistem otomatis yang digunakan untuk melakukan analisa citra dan video oleh *computer* untuk memperoleh informasi dan pemahaman dari suatu obyek. *Computer Vision* merupakan kemampuan dari sebuah mesin atau *computer* dalam melihat atau mengenali sebuah citra dengan sama atau bahkan dapat melebihi kemampuan pengelihatannya manusia asli (Mulya *et al.*, 2023). Proses pada *Computer Vision* dilakukan dengan memperoleh, mengambil atau mengakuisisi citra. Melakukan operasi pengolahan citra untuk memprediksi data citra. Menganalisis dan menginterpretasi citra dan mempunyai tujuan tertentu, misalnya mengklasifikasikan pola motilitas sperma dengan pengenalan pola gerak seperti progresif, non-progresif dan *immotile* (Ardiansyah *et al.*, 2022).

2.11 *Artificial Intelligence*

Artificial Intelligence atau kecerdasan buatan merupakan pemanfaatan teknologi yang semakin canggih dengan berbagai inovasi digunakan sebagai solusi untuk membantu pekerjaan manusia. Pembahasan lain terkait *artificial intelligence* merupakan kecerdasan buatan yang berperilaku layaknya manusia seperti kemampuan mengenali pola, kemampuan belajar dan kemampuan dalam mengambil keputusan (Pakpahan, 2021).



Gambar 5. Keterkaitan AI, ML dan DL (Arnita *et al.*, 2022).

Gambar 5 menunjukkan bahwa *artificial intelligence* memiliki fungsi seperti *machine learning* dan *deep learning* yang memungkinkan penggunaannya untuk memproses sejumlah besar pekerjaan dalam waktu yang sangat singkat menggunakan kemampuan pemrosesan komputer yang meningkat secara eksponensial. *Artificial intelligence*, *machine learning* dan *deep learning* memiliki keterkaitan dan saling berhubungan satu sama lain (Athania dan Dewayanto, 2023).

2.12 *Machine Learning*

Penelitian oleh (Arnita *et al.*, 2023) menyatakan *Machine learning* merupakan salah satu cabang dari disiplin dalam kecerdasan buatan (*artificial intelligent*) yang membahas pembangunan sistem berdasarkan data. Program *Machine Learning* mengikuti cara belajar manusia, yakni belajar dari contoh dan masalah sebelumnya. *Machine Learning* berisi sebuah algoritma yang bersifat generik (umum) dimana algoritma tersebut dapat menghasilkan sesuatu yang menarik atau bermanfaat dari sejumlah data tanpa harus menulis kode yang spesifik. Algoritma yang generik tersebut ketika diberikan sejumlah data maka akan dapat

membangun sebuah model atau inferensi dari data tersebut. Pembagian *Machine Learning* menjadi beberapa kelompok sebagai berikut :

1. *Supervised Learning*

Supervised Learning merupakan jenis pembelajaran terawasi dalam *machine learning* yang menggunakan data berlabel untuk melatih algoritma supaya dapat memetakan *input* ke *output* yang diinginkan sehingga model dapat meningkatkan akurasi seiring dengan bertambahnya data.

2. *Unsupervised Learning*

Unsupervised Learning merupakan algoritma *machine learning* yang datanya tidak memiliki label secara eksplisit dan model mampu belajar dari data dengan menemukan pola yang implisit. Jenis pembelajaran ini hanya mempunyai variable input tapi tidak mempunyai variable output yang berhubungan. Tujuannya adalah untuk memodelkan struktur data dan menyimpulkan fungsi yang mendeskripsikan data tersebut.

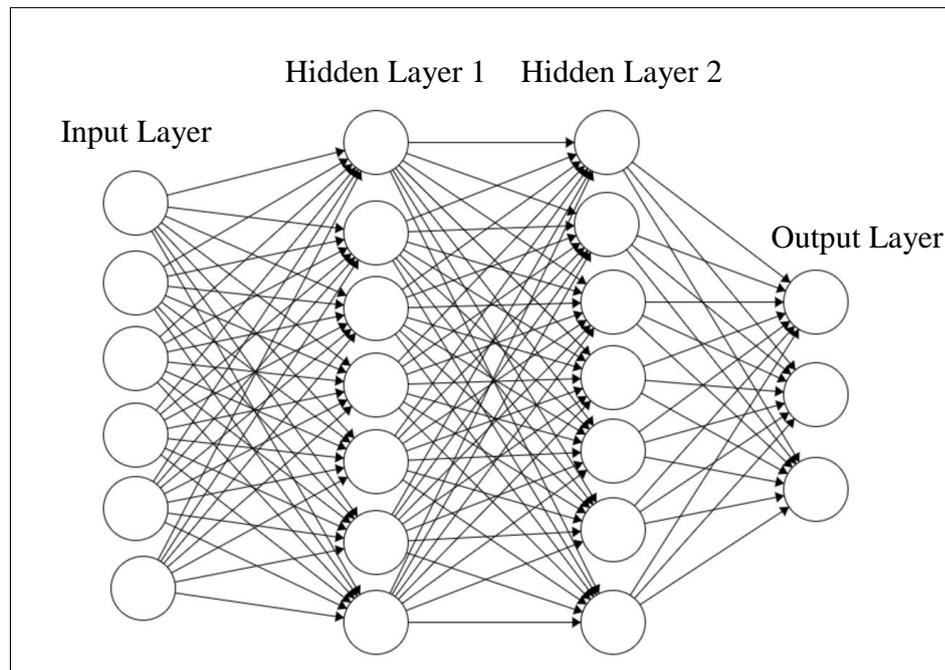
3. *Reinforcement Learning*

Reinforcement Learning merupakan pembelajaran dalam *machine learning* yang bekerja untuk berinteraksi dengan lingkungan untuk mempelajari serangkaian tindakan dengan mengikuti metode *trial-and-error*. Tipe *input* data pada *reinforcement learning* yaitu data sudah didefinisikan.

2.13 Deep Learning

Deep learning merupakan pendekatan *machine learning* yang juga merupakan bagian dari kecerdasan buatan (*Artificial Intelligence*) yang mampu belajar dari data yang tidak terstruktur atau tidak diberi label secara langsung, dengan representasi berdasarkan otak manusia. *Deep learning* berperan untuk dapat mengklasifikasikan atau menemukan pola melalui sebuah *classifier*, data mentah tersebut harus diubah menjadi format yang sesuai (Halbouni *et al.*, 2022).

Deep Learning menggunakan algoritma berlapis dalam upaya memahami data secara lebih baik, serta penggunaan algoritma tidak hanya terbatas pada pembuatan rangkaian hubungan yang bisa dijelaskan sama halnya pada regresi yang lebih mendasar, tetapi untuk membuat representasi terdistribusi yang berinteraksi berdasarkan serangkaian faktor, *deep learning* bergantung pada lapisan algoritma linear (Pakpahan, 2021). Gambar 6 berikut menjelaskan ilustrasi terkait *deep learning*.

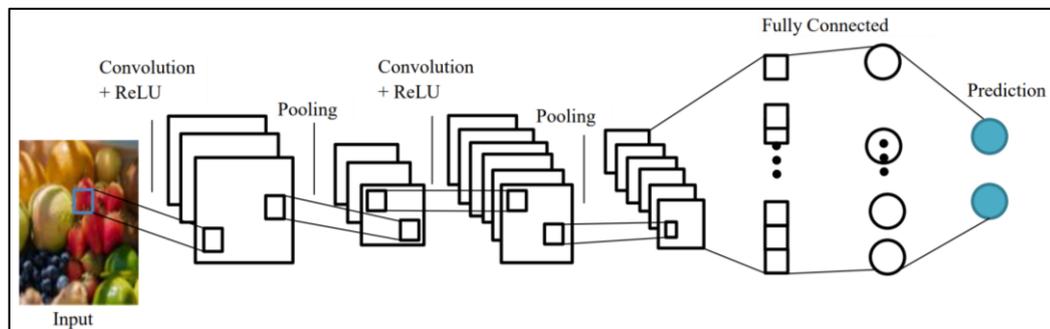


Gambar 6. Ilustrasi *Deep Learning* (Rizki *et al.*, 2020).

Gambar 6 merupakan jaringan pada *deep learning* terdiri dari beberapa layer. Layer-layer tersebut merupakan kumpulan dari node-node. Node hanya sebagai tempat perhitungan terjadi. Node input yang digabung dengan *weight* (bobot), setelah *input* dan *weight* dijumlahkan dan jumlahnya melewati tahapan yang disebut fungsi aktivasi node, untuk menentukan seberapa jauh sinyal yang berlangsung lebih lanjut melalui jaringan, hal tersebut mempengaruhi hasil akhir. *Deep Learning* didefinisikan secara teknis yaitu *machine learning* yang memiliki lebih dari satu *hidden layer*. Ilustrasi terkait *deep learning* dapat dilihat pada Gambar 6. yang terdapat 4 layer dan tiap layer mempunyai jumlah node yang berbeda-beda (Rizki *et al.*, 2020).

2.14 Convolutional Neural Network

Convolutional Neural Network (CNN) merupakan salah satu algoritma penerapan *deep learning* yang dapat digunakan untuk mengolah data citra seperti mengklasifikasikan data gambar atau suara (Dinata *et al.*, 2023). CNN digunakan untuk mengolah data dengan struktur grid yaitu salah satunya berupa citra dua dimensi dan mampu memproses data dengan dimensi tinggi seperti video. CNN menggunakan kernel dua dimensi dan tinggi dimensi di setiap unit pada lapisan CNN yang akan dikonvolusi. Proses menggabungkan fitur dengan fitur spasial serupa media input maka digunakan kernel yang ada di dalam CNN. CNN berfungsi untuk mengurangi jumlah dari variabel agar lebih mudah untuk dilakukan pembelajaran yaitu dengan menggunakan parameter. CNN menunjukkan bahwa metode jaringan tersebut menggunakan operasi dari matematika yang disebut konvolusi. CNN juga dilatih untuk mempelajari fitur dari objek untuk dapat memprediksinya (Jinan *et al.*, 2022). Gambar 7 berikut merupakan ilustrasi arsitektur CNN.



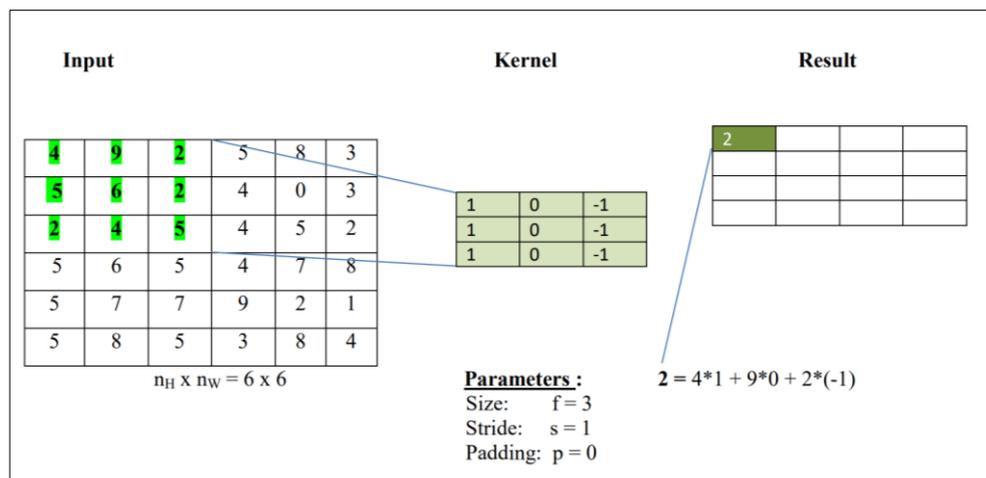
Gambar 7. Ilustrasi CNN.

Gambar 7 merupakan lapisan penyusun sebuah CNN terdiri dari *Convolution*, *ReLU*, *Pooling* dan *Fully Connected Layer*, dengan penjelasan sebagai berikut.

2.14.1 Convolution Layer

Convolution Layer merupakan proses ekstraksi fitur dengan menggunakan filter atau kernel. Proses konvolusi dilakukan dengan menggeser filter hingga mencakup seluruh bagian citra. Jumlah pergeseran filter disebut *stride*. Proses konvolusi yang ingin mengekstraksi semua piksel pada citra, maka dapat dilakukan

dengan cara menambah piksel bernilai 0 pada setiap sisi matriks. Pergeseran setiap filter akan dilakukan perkalian matriks antara matriks input dengan matriks filter, hasil proses tersebut berupa matriks 2 dimensi. Gambar 8 berikut merupakan ilustrasi terkait *Convolution Layer* (Sholawati *et al.*, 2022).



Gambar 8. Ilustrasi Proses *Convolution Layer*.

Parameter konvolusi selain memiliki parameter ukuran *kernel filter* dan jumlah *filter*, juga terdapat parameter lainnya yang mempengaruhi bentuk dari output hasil konvolusi yaitu *stride* dan *padding* (Hutauruk *et al.*, 2020).

a. *Stride*

Stride adalah ukuran yang menyatakan jumlah pergeseran (langkah) yang dilakukan oleh kernel. Ukuran ini biasa digunakan untuk mengurangi ukuran *output* (Hutauruk *et al.*, 2020). Nilai *stride* adalah 1 maka konvolusi akan bergeser secara horizontal atau vertikal (Azis *et al.*, 2021).

b. *Padding*

Padding adalah ukuran yang menyatakan jumlah penambahan border diseluruh tepian *input*, yang berfungsi meminimalisir kehilangan informasi pada tepian citra (*input*), hal ini dikarenakan proses konvolusi itu sendiri, dimana biasanya bagian tepi dari citra akan terlewat oleh *kernel*, kecuali untuk kernel berukuran 1x1. Metode yang digunakan pada untuk mengatasi masalah ini adalah penggunaan

zero-padding yaitu menambahkan nilai 0 pada setiap tepian citra (*input*) (Hutauruk *et al.*, 2020).

Ukuran *output* dari hasil konvolusi dapat dihitung dengan menggunakan persamaan (1) (Hutauruk *et al.*, 2020).

$$O = 1 + \frac{N + 2P - F}{S} \quad (1)$$

Keterangan :

O : Ukuran *output* yang didapat setelah konvolusi.

N : Ukuran *input*.

P : Jumlah *Padding*.

F : Ukuran *kernel filter*.

S : Jumlah *stride*.

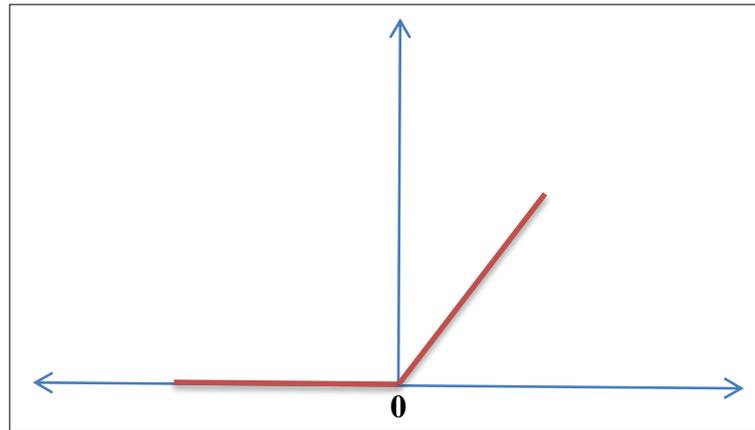
Rumus pada persamaan (1) digunakan untuk menghitung ukuran *output* berdasarkan ukuran *input*, *padding*, ukuran kernel dan *stride* yang digunakan. *Padding* dan *stride* mempengaruhi seberapa besar ukuran *output* setelah filter diterapkan pada gambar (Hutauruk *et al.*, 2020).

2.14.2 *Rectified Linear Unit (ReLU)*

ReLU atau *rectified linear unit* merupakan fungsi aktivasi pada model CNN untuk menentukan *output* pada citra hasil konvolusi (Herdianto dan Nasution, 2022). ReLU berfungsi untuk menghilangkan nilai negatif pada citra. Persamaan (2) berikut ini merupakan persamaan pada fungsi ReLU.

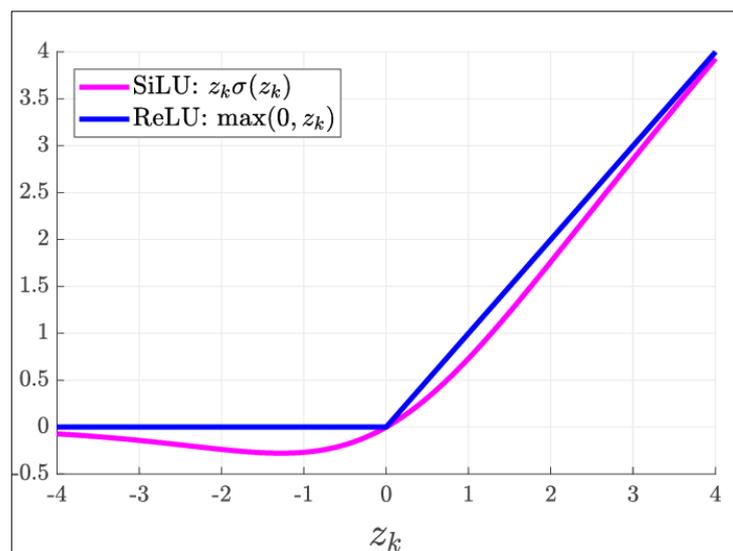
$$f(x) = \begin{cases} x & (x \geq 0) \\ 0 & (x < 0) \end{cases} \quad (2)$$

Persamaan (2) menunjukkan ketika terdapat nilai negatif, ReLU akan mengubahnya menjadi 0 dan ketika nilai positif ReLU akan mempertahankannya (Pangestu *et al.*, 2020). Gambar 9 berikut merupakan fungsi ReLU yang mana x merupakan input neuron (Efrian dan Latifa, 2022) beserta dengan grafiknya.



Gambar 9. Grafik Fungsi ReLU (Herdianto dan Nasution, 2022).

Gambar 9 merupakan fungsi aktivasi ReLU. Fungsi ini kemudian dikembangkan menjadi fungsi aktivasi SiLU atau *Sigmoid Linear Unit*, yang digunakan dalam YOLOv8.



Gambar 10. Grafik Fungsi SiLU (Ardiansyah, 2023).

Gambar 10 merupakan grafik fungsi aktivasi SiLU yang dikenal dengan memiliki gradien halus dan dapat memberikan manfaat dalam kinerja model (Hussain, 2024). Fungsi aktivasi SiLU untuk mengaktifkan neuron yang akan digunakan dalam proses ekstraksi fitur di *backbone* seperti dilihat pada Gambar 9. terkait grafik SiLU yang dekat pada nilai negatif tetapi tidak menyentuh nilai 0, yang artinya SiLU masih dapat menerima *input* negatif. SiLU merupakan salah satu fitur yang terdapat dalam jaringan CBS (*convolutional layer, batch normalization* dan SiLU). Jaringan ini dirancang untuk mengekstrak fitur dan mengatasi gradien.

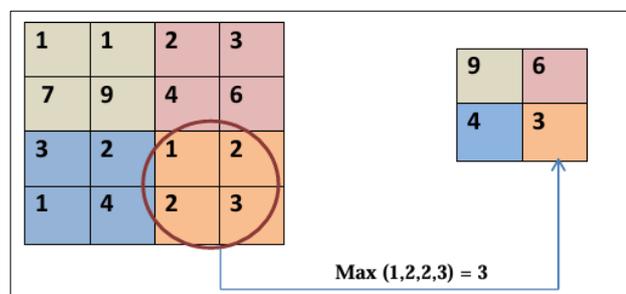
Fungsi aktivasi SiLU berguna untuk memperkuat kapasitas jaringan saraf untuk mempelajari pola yang rumit dan memproses fitur *input* secara efisien yang dapat meningkatkan kinerja pada model (Zhou *et al.*, 2024). Perhitungan pada fungsi SiLU dilakukan dengan mengalikan input dengan fungsi sigmoid, sehingga menyerupai ReLU yang bersifat kontinu dan “undershooting”. Teori matematis menyatakan fungsi SiLU dijelaskan dalam Persamaan (3) berikut (Salma dan Hidayat, 2024).

$$R_5(z) = \begin{cases} z, & \text{for } z > 0, \\ z\sigma(z), & \text{for } z \leq 0, \end{cases} \quad (3)$$

Persamaan (3) menunjukkan variabel z merupakan *input* dan $\sigma(z)$ adalah fungsi *sigmoid*.

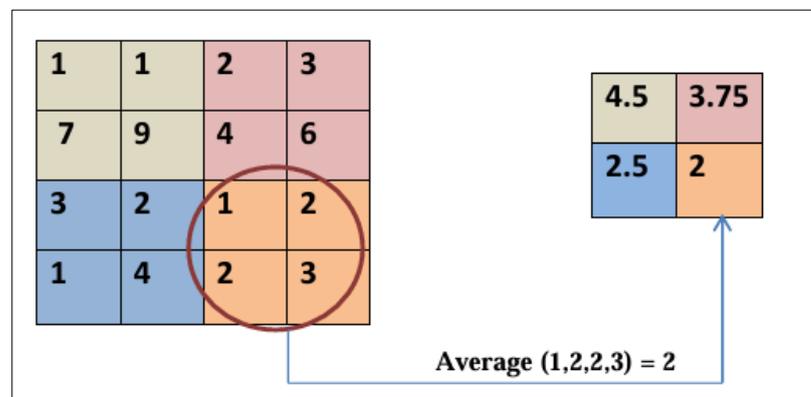
2.14.3 Pooling Layer

Pooling merupakan langkah penting dalam CNN yang berfungsi untuk mengurangi dimensi dari *feature map* dengan menggabungkan sekumpulan nilai menjadi sejumlah nilai yang lebih kecil. Pooling hanya menyimpan informasi penting dan menghilangkan informasi yang tidak relevan, sehingga dapat membantu mengurangi *overfitting* (Gholamalinezhad and Khosravi, 2020). Metode *pooling* yang digunakan dalam CNN meliputi *max pooling* dan *average pooling*. *Max pooling* berfungsi untuk mengurangi variabilitas dengan menyaring nilai terbesar dalam konvolusi seperti ditunjukkan pada Gambar 11.



Gambar 11. Contoh *Max Pooling Layer* (Gholamalinezhad and Khosravi, 2020).

Gambar 11 merupakan contoh dari *max pooling layer*, sebuah filter berukuran 2 x 2 bergerak di atas *input* yang lebih besar, dan untuk setiap area yang ditutupi oleh filter, nilai maksimum dari elemen-elemen tersebut diambil sebagai nilai *output*. Area pertama pada filter 2 x 2 meliputi elemen-elemen [1, 2, 2, 3], dengan nilai maksimum 3 yang menjadi *output* untuk area tersebut. Proses ini diterapkan di seluruh citra untuk menghasilkan *feature map* yang berukuran lebih kecil. Metode *pooling layer* selanjutnya adalah *average pooling*, yang berfungsi untuk melakukan ekstraksi fitur dengan menghitung rata-rata dari nilai-nilai dalam area tertentu, seperti ditunjukkan pada Gambar 12.



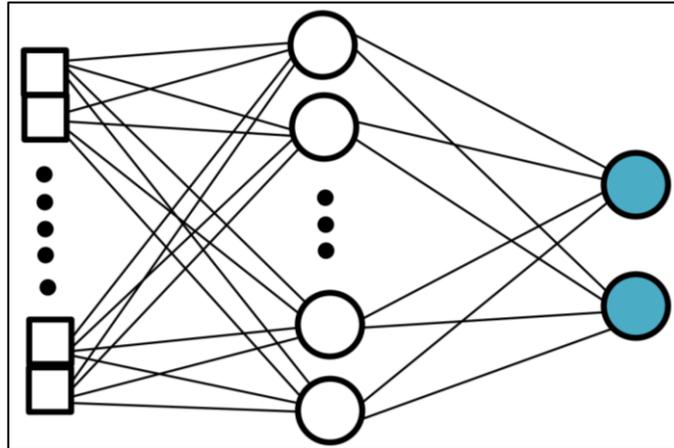
Gambar 12. Contoh *Average Pooling Layer* (Gholamalinezhad and Khosravi, 2020).

Gambar 12 menunjukkan proses *average pooling* dengan menggunakan filter 2 x 2, di mana *layer* ini menghitung rata-rata dari nilai-nilai dalam area yang ditutupi filter, sebagai contoh pada area pertama filter yang mencakup elemen-elemen [1, 2, 2, 3]. Rata-rata dari nilai tersebut adalah 2, yang digunakan sebagai *output* untuk area tersebut.

2.14 .4 *Fully Connected Layer*

Fully connected layer merupakan layer yang digunakan untuk proses fitur yang terintegrasi penuh sebelum masuk ke lapisan *softmax* untuk proses klasifikasi. Lapisan ini terdiri dari sejumlah neuron, sebelum masuk *fully connected layer* masukan akan diubah menjadi vektor dengan cara diratakan supaya terhubung

sepenuhnya. Lapisan ini bertujuan untuk mengolah data sehingga data dapat di klasifikasikan (Gunawan *et al.*, 2024).



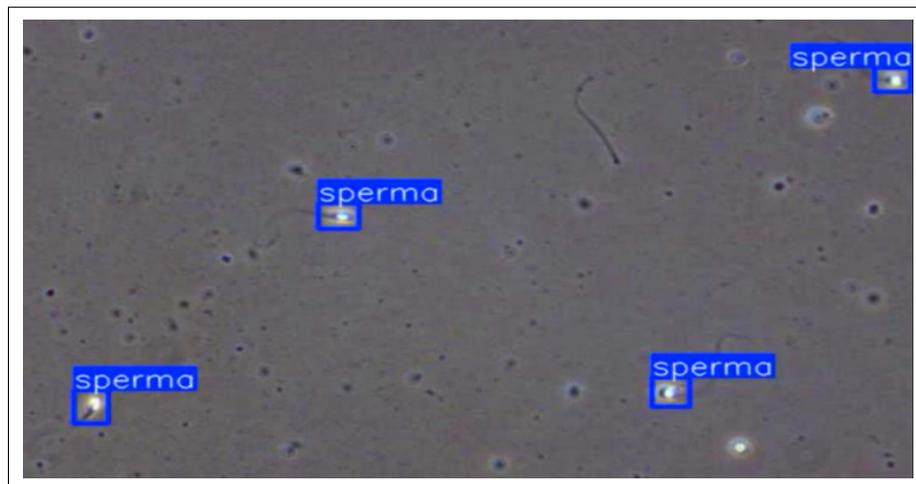
Gambar 13. Ilustrasi *Fully Connected Layer*.

Gambar 13 menunjukkan proses perubahan hasil fitur dari *pooling* menjadi vektor, dengan cara meratakan nilai-nilai matriks dari layer sebelumnya. Jaringan ini umumnya menggunakan lapisan yang terhubung sepenuhnya di mana setiap piksel dianggap sebagai neuron terpisah. Dalam proses ini biasanya diterapkan metode '*dropout*'. Metode ini bertujuan untuk menonaktifkan beberapa sambungan yang terhubung setiap neuron untuk menghindari *overfitting*. Proses terakhir adalah klasifikasi *input* terhadap targetnya yaitu, ke dalam 3 kelas (Nurhikmat, 2018).

Tahap terakhir klasifikasi, melibatkan metode *backpropagation* yang bertujuan untuk meminimalkan kesalahan dengan menyesuaikan bobot berdasarkan perbedaan antara output dan target yang diinginkan (Armelia *et al.*, 2024). Proses *backpropagation* digunakan saat training pada CNN dan setelah proses *feedforward*. Proses *feedforward* merupakan proses yang menghasilkan beberapa lapisan yang digunakan untuk mengklasifikasikan data citra dengan menggunakan bobot dan bias yang diperbarui dari proses *backpropagation*. *Backpropagation* akan melakukan penelusuran kesalahan dari proses *feedforward* mulai dari lapisan output hingga lapisan pertama (Cahya *et al.*, 2021).

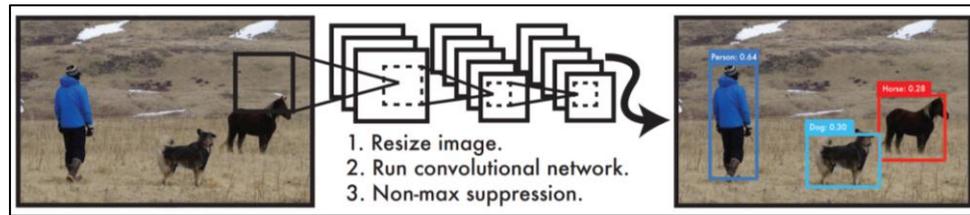
2.15 *You Only Look Once (YOLO)*

You Only Look Once (YOLO) adalah sebuah algoritma yang dikembangkan untuk mendeteksi sebuah objek secara *real-time*. Sistem pendeteksian yang dilakukan adalah dengan menggunakan *repurposed classifier* atau *localizer* untuk melakukan deteksi. Model diterapkan pada sebuah citra di beberapa lokasi dan skala. Wilayah pada citra yang diberi *score* paling tinggi akan dianggap sebagai hasil deteksi. YOLO menggunakan pendekatan Jaringan Saraf Tiruan (JST) untuk mendeteksi sebuah objek pada sebuah citra. Jaringan ini membagi citra menjadi beberapa wilayah dan memprediksi setiap kotak pembatas (*bounding box*) dan probabilitas untuk setiap wilayah seperti pada ilustrasi Gambar 14 berikut (Putro *et al.*, 2020).



Gambar 14. YOLO Mendeteksi Objek.

Gambar 14 merupakan model YOLO dalam mendeteksi objek sperma. Penelitian yang dilakukan oleh Redmon, YOLO memperlakukan deteksi objek sebagai masalah regresi, artinya YOLO langsung memprediksi *bounding box* dan *class probability* dari gambar secara langsung dalam satu proses. Proses keseluruhan pada deteksi objek dilakukan oleh satu jaringan neural, sehingga bisa dioptimalkan secara keseluruhan untuk meningkatkan kinerja deteksi. Dengan adanya sistem ini, hanya perlu “melihat” gambar sekali untuk memprediksi objek apa yang ada dan di mana letaknya seperti yang terlihat pada Gambar 15 (Redmon *et al.*, 2016).



Gambar 15. Proses Yolo Mendeteksi Objek (Redmon *et al.*, 2016).

Gambar 15 menunjukkan YOLO mendeteksi gambar secara sederhana dan langsung, yaitu dengan mengubah ukuran, kemudian proses menjalankan satu jaringan konvolusi pada gambar tersebut, lalu mengatur threshold hasil deteksi berdasarkan model confidence. Komponen keseluruhan deteksi dijadikan satu ke *single neural network*. *Single neural network* menggunakan fitur dari seluruh gambar untuk memprediksi B kotak pembatas (*bounding box*), artinya jaringan YOLO menganalisis dan memproses gambar secara keseluruhan dan semua objek di dalamnya, tidak hanya fokus pada area tertentu. Desain YOLO juga memungkinkan proses pelatihan model secara menyeluruh (*end-to-end*) dan dapat langsung dilatih dari awal hingga akhir dalam satu alur secara *real-time*.

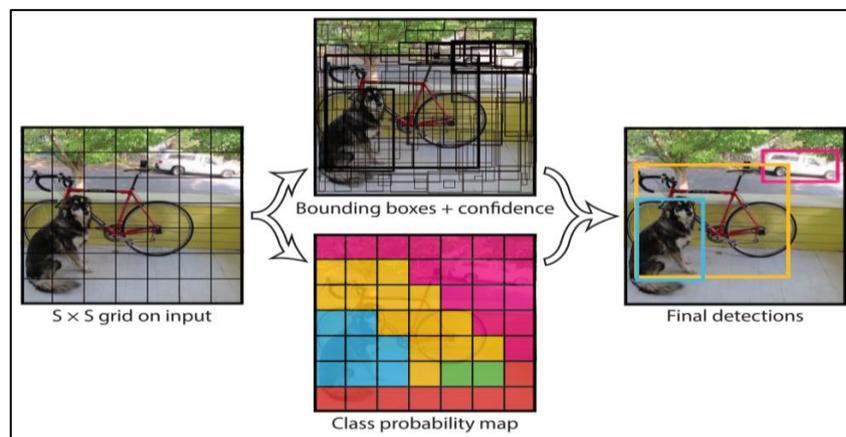
Tahap sebelum melakukan proses input, gambar yang akan dibagi menjadi grid $S \times S$, jika suatu objek jatuh ke dalam sel grid, maka sel akan bertanggung jawab untuk mendeteksi objek tersebut, setiap sel grid akan memprediksi kotak pembatas (*bounding box*) dan skor kepercayaan (*confidence scores*). *Confidence scores* menggambarkan seberapa yakin model bahwa kotak tersebut mengandung objek dan seberapa akurat prediksi yang di buat. *Confidence score* didefinisikan sebagai $Pr(\text{Object}) \times \text{IOU}_{\text{truth pred}}$, jika tidak ada objek dalam sel tersebut, *confidence score* nol; namun jika ada objek, *confidence scores* sama dengan nilai *intersection over union (IOU)* antara kotak yang diprediksi dan kebenaran yang sesungguhnya.

Kotak pembatas (*bounding box*) terdiri dari 5 memprediksi : x, y, w, h dan kepercayaan (*confidence*). Koordinat (x,y) menunjukan posisi kotak pembatas (*bounding box*) yang berisi objek. Koordinat ini dihitung relatif terhadap batas-batas sel grid tempat objek itu berada. Lebar dan tinggi kotak diprediksi relatif

terhadap seluruh gambar. Prediksi kepercayaan (*confidence*) pada YOLO menunjukkan seberapa baik kotak pembatas (*bounding box*) yang diprediksi oleh model sesuai dengan kotak kebenaran (*ground truth*), dengan setiap sel grid yang memprediksi C probabilitas kelas kondisional $\Pr(\text{Class}_i | \text{Object})$ sesuai dengan keberadaan objek dalam sel grid. Objek yang diprediksi hanya satu set kelas probabilitas kelas per sel grid, terlepas dari jumlah kotak B . Proses pengujian dilakukan dengan mengalikan probabilitas kelas kondisional dengan prediksi *individual box confidence* menggunakan persamaan (4) berikut.

$$\Pr(\text{Class}_i | \text{Object}) \times \Pr(\text{Object}) \times \text{IOU}_{\text{truth pred}} = \Pr(\text{Class}_i) \text{IOU}_{\text{truthpred}} \quad (4)$$

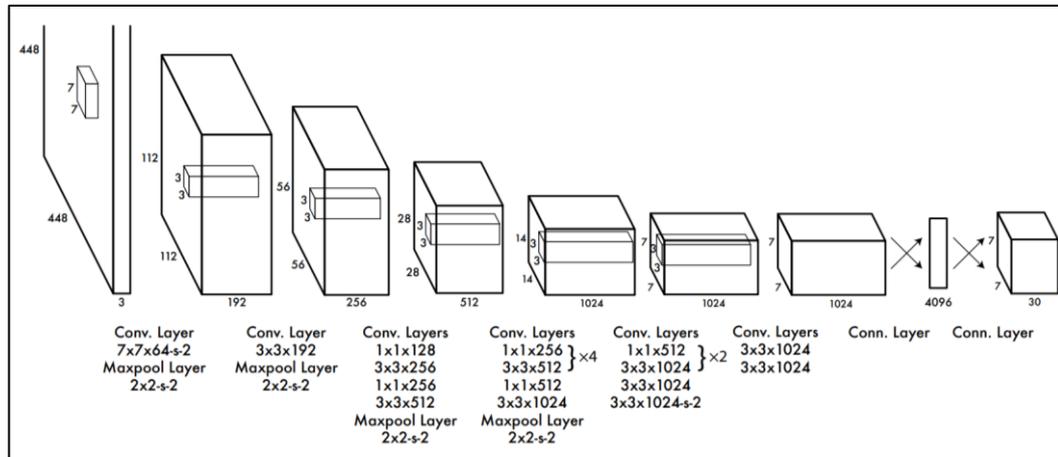
Persamaan (4) digunakan untuk menghitung nilai kepercayaan spesifik kelas dari sebuah kotak pembatas (*bounding box*) sehingga model dapat menentukan apakah suatu objek dalam kotak tersebut termasuk kedalam kelas tertentu dan seberapa kotak dalam mencakup objek. Gambar 16 merupakan ilustrasi dalam prediksi B *bounding box*, *confidence score* dan C *class probabilities*.



Gambar 16. Model Mendeteksi Objek (Redmon *et al.*, 2016).

Gambar 16 merupakan representasi model YOLO menggunakan jaringan *Convolutional Neural Network* (CNN) dengan arsitektur yang terinspirasi dari *GoogLeNet*. YOLO memiliki 24 lapisan konvolusi diikuti oleh 2 lapisan *fully connected*. YOLO menggunakan cara yang sederhana untuk memproses gambar yaitu dengan kombinasi lapisan reduksi atau mengurangi ukuran menjadi 1 x 1 dan lapisan konvolusi dengan lapisan 3 x 3 yang bertujuan untuk mengurangi

kompleksitas jaringan dan mempertahankan efektivitas dalam mengekstrak fitur dari gambar. Output dari jaringan YOLO adalah tensor $7 \times 7 \times 30$ berisi objek yang dideteksi pada gambar. Arsitektur jaringan pada *case* ini seperti pada Gambar 17 berikut.

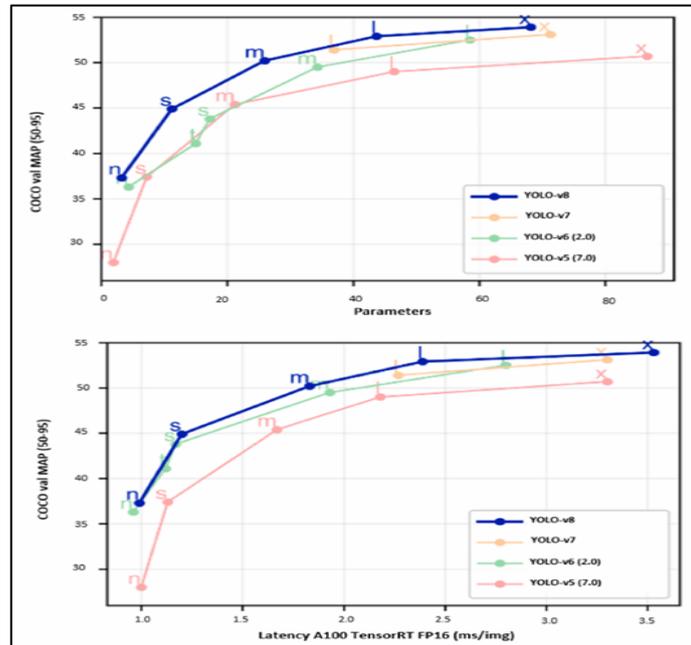


Gambar 17. Aritektur Jaringan (Redmon *et al.*, 2016).

2.16 You Only Look Once V8 (YOLOv8)

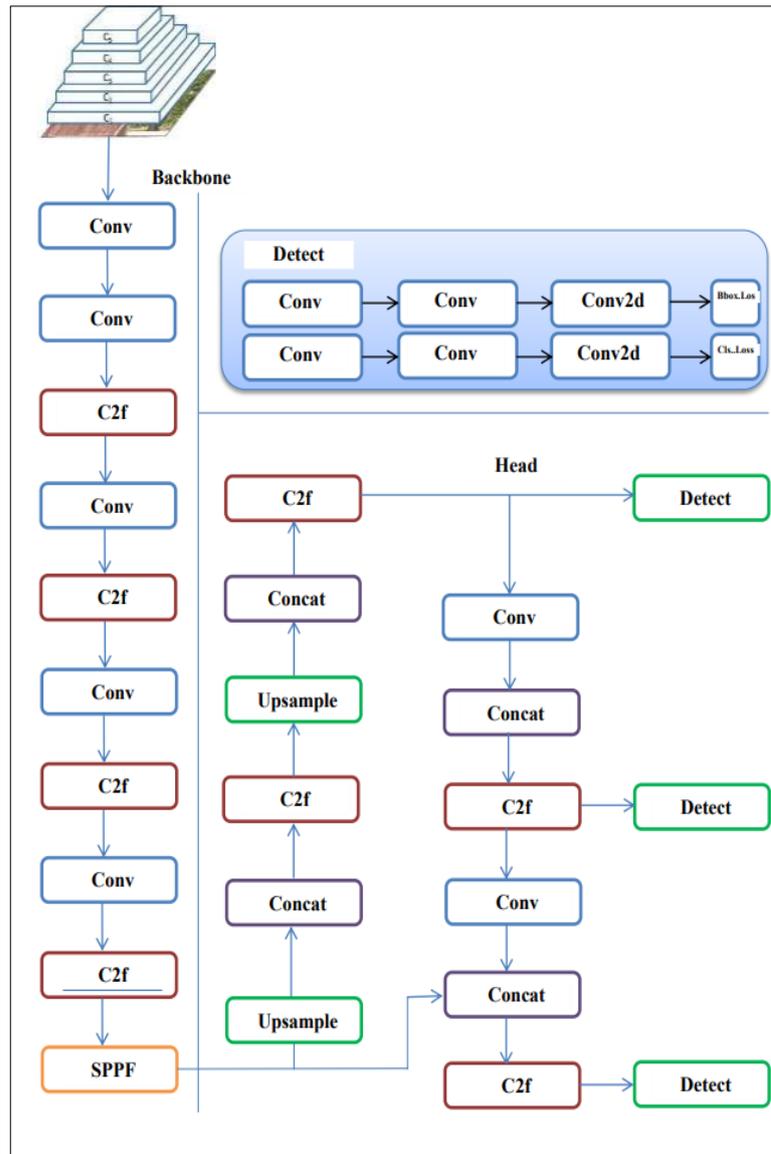
YOLOv8 merupakan algoritma deteksi terbaru yang diperkenalkan dalam seri algoritma YOLO, yang cocok untuk proses deteksi objek, klasifikasi gambar dan segmentasi instan (Yuan *et al.*, 2024). YOLOv8 memiliki kemampuan untuk melihat seluruh gambar sekali kemudian melewati jaringan saraf sekali langsung mendeteksi objek yang ada. YOLOv8 merupakan generasi terbaru dari YOLOv7 (Gibran *et al.*, 2024). *Ultralytics* memperkenalkan YOLOv8 di Januari 2023 yang menandai evolusi besar dalam seri YOLO dengan memberikan berbagai peningkatan dan kemampuan lebih fleksibel kepada pengguna. YOLOv8 merupakan pengembangan model yang canggih dalam YOLO. Gambar 18. menunjukkan bahwa semua varian YOLOv8 menghasilkan *throughput* (kecepatan pemrosesan) yang lebih baik dengan jumlah parameter yang serupa dibandingkan dengan YOLOv5 dan YOLOv6 yang dilatih pada resolusi 640 piksel. YOLOv8 didefinisikan bahwa lebih cepat dalam melakukan deteksi objek, yang menunjukkan adanya perbaikan arsitektur yang efisien untuk digunakan pada perangkat keras. YOLOv8 berfokus untuk digunakan pada perangkat *edge* dan

YOLOv8 akan tetap memberikan kecepatan inferensi (prediksi) yang tinggi pada perangkat tersebut (Tyagi *et al.*, 2024).



Gambar 18. Perbandingan YOLOv5 s/d YOLOv8 (Tyagi *et al.*, 2024).

Gambar 18 menunjukkan bahwa model YOLOv8 memiliki peningkatan performa dan akurasi lebih baik dari sebelumnya. Hal ini dicapai melalui penambahan lapisan pemrosesan gambar dan penggunaan teknik *machine learning* yang lebih canggih. Perbedaan dengan YOLO sebelumnya adalah dalam hal penggunaan *anchor boxes*. YOLOv8 memperkenalkan teknik *anchor-free* yang lebih sederhana dan efisien yang membuatnya lebih mudah diimplementasikan dan lebih cepat dalam melakukan deteksi objek. YOLOv8 menampilkan beberapa peningkatan seperti jaringan tulang punggung (*backbone network*), leher (*neck*) dan kepala (*head*) seperti pada Gambar 19 (Yanto *et al.*, 2023).



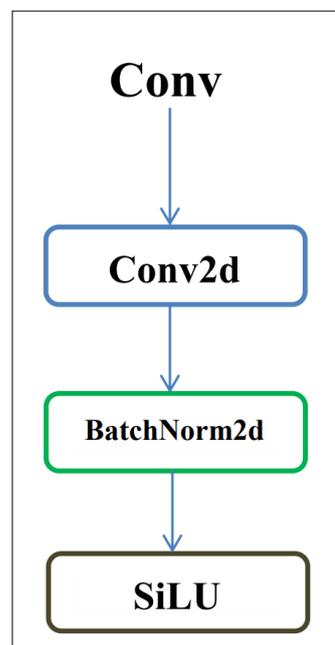
Gambar 19. Arsitektur YOLOv8 (Yanto *et al.*, 2023).

Gambar 19 merupakan arsitektur YOLOv8 terdiri dari jaringan tulang punggung (*backbone network*), leher (*neck*) dan kepala (*head*). *Backbone network* menggunakan *Feature Pyramid Network* (FPN) untuk mengekstraksi fitur dari gambar input, sedangkan *neck* menggunakan serangkaian *Cross Layer Connection* (CLC) untuk menyempurnakan fitur ini. *Head* mengambil fitur yang disempurnakan dan memprediksi *bounding box*, *confidence scores* dan akurasi untuk setiap objek dalam citra (Yanto *et al.*, 2023). *Backbone* dibentuk dengan menumpuk lapisan convolusi (conv), modul C2f yang berisi dua conv dan *n* *Bottleneck* serta modul SPPF yang berisi dua conv yang ditempatkan di awal dan akhir, juga beberapa lapisan *maxpooling* dan *concat* (Lei *et al.*, 2024).

Modul deteksi yang berada pada komponen *head* menggunakan serangkaian lapisan konvolusi dan linear untuk memetakan fitur berdimensi tinggi ke *output bounding box* dan kelas objek. Arsitektur secara keseluruhan dirancang untuk menjadi cepat dan efisien, tetapi tetap mencapai akurasi deteksi yang tinggi. Teori pada legenda diagram, persegi panjang mewakili lapisan, dengan label yang menjelaskan jenis lapisan dan parameter yang relevan. Panah menggambarkan aliran data antara lapisan, dengan arah panah menunjukkan aliran data dari satu lapisan ke lapisan berikutnya (Manurung *et al.*, 2024). Jaringan utama pada arsitektur YOLOv8 meliputi modul *conv*, *C2f* dan *SPPF*.

2.16.1 Convolution Layer

Convolution Layer merupakan lapisan konvolusi yang melakukan operasi konvolusi, *Batch Normalization* (BN) dan fungsi aktivasi SiLU pada gambar *input*. Semua lapisan konvolusi menggunakan parameter yang sama. Ukuran kernel diatur menjadi 3 x 3 dengan *stride* 2 dan *padding* 1, disingkat menjadi 3|2|1 (Lei *et al.*, 2024).



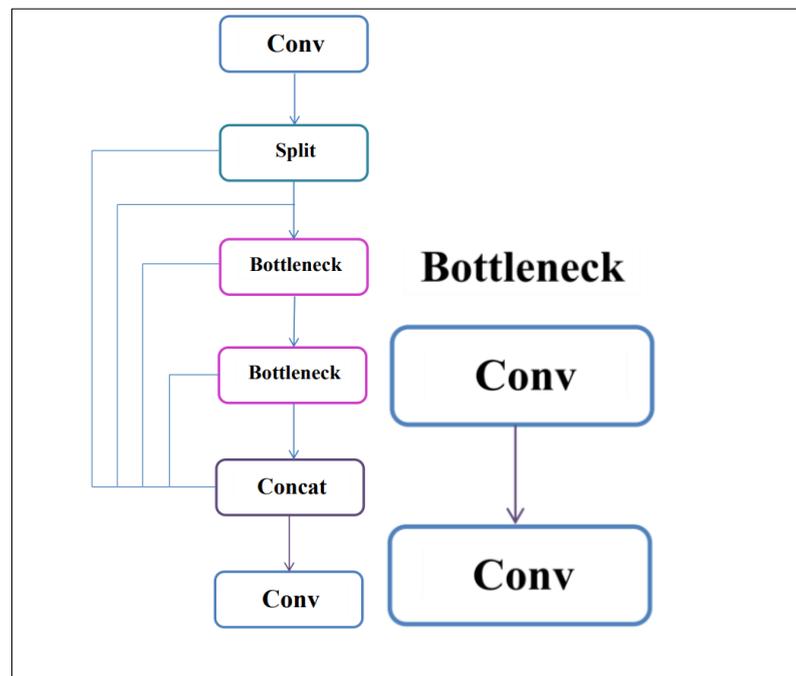
Gambar 20. *Convolution Layer* (Yanto *et al.*, 2023).

Gambar 20 merupakan proses *convolution layer*. Lapisan ini terdiri dari jaringan CBS (*Convolution, Batch Normalization, SiLU Activation*), yang dirancang untuk mengekstrak fitur dan mengekstraksi informasi spasial lokal dari inputan.

Informasi spasial yang dimaksud dalam konteks dimana objek berada, jarak antara objek dan hubungan antar objek. Lapisan BN berfungsi untuk menormalkan fitur yang telah di konvolusi, sehingga meningkatkan kemampuan generalisasi model dan mengurangi ketergantungan pada parameter inisialisasi. *Batch Normalization* yang ada pada lapisan ini membuat fitur yang di ekstrasi tetap memiliki kualitas yang baik. Fungsi aktivasi SiLU berfungsi sebagai elemen *non-linear*, yang memperkuat kapasitas jaringan saraf untuk belajar pola yang rumit dan memproses fitur inputan (Zhou *et al.*, 2024).

2.16.2 Modul C2F

YOLOv8 memperkenalkan modul C2f terbaru yang menggantikan modul c3 pada versi sebelumnya (Yao, 2024). Modul C2f memperkenalkan modifikasi pada *CSPLayer* dengan menggabungkan fitur penting dari objek dengan informasi kontekstual sehingga membuat prediksi lebih akurat (Cheng *et al.*, 2024). Modul C2f merupakan poin penting dalam *Backbone* dan *Neck*. Modul ini memperkuat informasi penting yang dibutuhkan model supaya tidak kehilangan secara detail saat informasi bergerak melalui banyak lapisan.

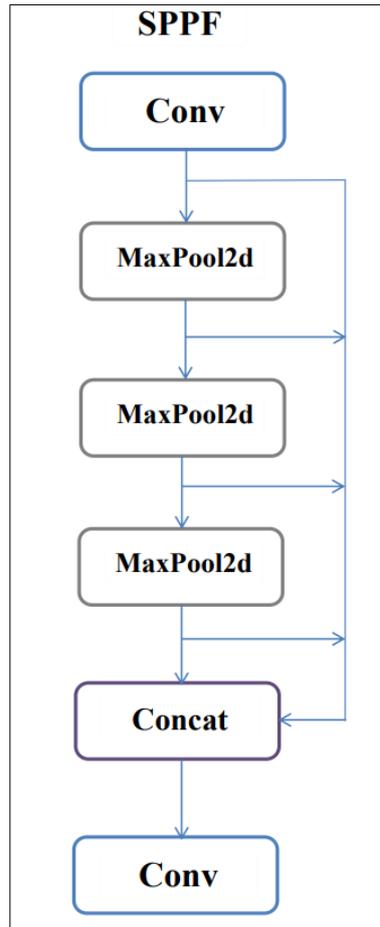


Gambar 21. Modul C2f (Yanto *et al.*, 2023).

Gambar 21 menjelaskan bahwa modul C2f terdiri dari lapisan *split* di bagian awal yang membagi *input feature map* menjadi beberapa bagian, n *Bottleneck* yang merupakan unit inti dalam pemrosesan fitur dan dua lapisan konvolusi (conv) yang terletak diawal dan akhir, yang berfungsi untuk mengekstraksi dan menyatukan kembali seluruh fitur. *Bottleneck* memiliki dua lapisan konvolusi (conv) dengan koneksi residual, yaitu proses yang memungkinkan informasi dari *input* dilewatkan langsung ke output. Proses ini bertujuan untuk mencegah hilangnya informasi penting dan mempercepat proses konvergensi model saat *training*. Modul C2f mampu menyeimbangkan antara efisiensi komputasi dan kedalaman fitur yang diekstrak dari citra input (Lei *et al.*, 2024).

2.16.3 Modul *Spatial Pyramid Pooling Fast* (SPPF)

Modul SPPF yang merupakan lapisan *pooling piramida spasial* terinspirasi dari SPP memungkinkan penggabungan fitur terkecil dari bagian tertentu pada gambar dan fitur secara keseluruhan pada gambar serta mendapatkan banyak informasi fitur (Cheng *et al.*, 2024). Lapisan SPPF dan lapisan konvolusi pada *backbone* akan memproses fitur pada berbagai ukuran dan lebih efisien, sementara lapisan *upsample* meningkatkan resolusi peta fitur. Lapisan SPPF yang terletak di akhir *backbone* melakukan ekstraksi fitur penting lebih lanjut yang akan digunakan untuk gambar input (Manurung *et al.*, 2024).

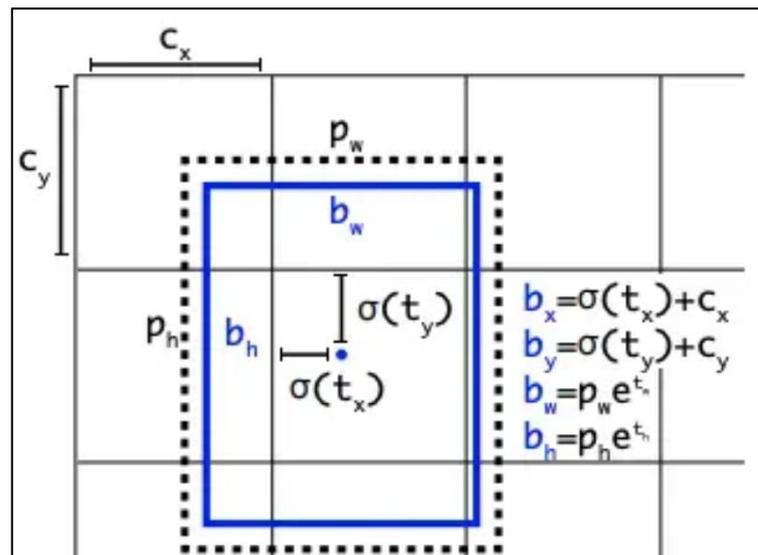


Gambar 22. Modul SPPF (Yanto *et al.*, 2023).

Gambar 22 merupakan struktur modul SPPF yang digunakan dalam arsitektur YOLOv8. Modul ini dirancang untuk mengekstraksi fitur dari berbagai skala secara efisien tanpa menambah beban komputasi. Struktur SPPF terdiri dari tiga *maxpool2d* yang berfungsi untuk melakukan *pooling* dengan ukuran kernel yang sama namun dilakukan secara berurutan, sehingga menciptakan efek piramida spasial. Proses ini membuat model menangkap informasi dari area yang lebih luas tanpa kehilangan detail local, selain itu dalam modul SPPF terdapat dua lapisan konvolusi (*conv*) yang terletak dibagian awal dan akhir. Lapisan konvolusi di bagian awal berfungsi untuk mengubah dimensi fitur sebelum masuk ke proses *pooling*, sedangkan lapisan konvolusi di bagian akhir digunakan untuk menggabungkan hasil *pooling* menjadi representasi fitur yang lebih banyak (Manurung *et al.*, 2024).

2.16.4 Anchor Free Detection

Pembaruan lain dari YOLOv8 yaitu dengan adanya *anchor free detection*, dimana YOLOv8 langsung memprediksi pusat objek tanpa perlu menghitung ukuran atau posisi dari *anchor box* yang sudah ditentukan sebelumnya (Yanto *et al.*, 2023). YOLOv8 memperkenalkan deteksi objek *anchor free* untuk meningkatkan prediksi *bounding box*, hal ini menjadi lebih efisien karena *feature map* yang lebih besar dan jaringan konvolusi yang lebih efisien, sehingga mengurangi waktu pelatihan dan risiko *overfitting* (Manurung *et al.*, 2024). Adanya pendekatan ini, membantu mengurangi jumlah prediksi kotak (*box prediction*) dan mempercepat proses *Non-Maximum Suppression* (NMS), yaitu tahap pemrosesan lanjutan yang menyaring deteksi objek berdasarkan tingkat kepercayaan (*confidence score*) dan menghindari deteksi berlebih (Niu *et al.*, 2022). Gambar 23 merupakan *anchor free detection* pada YOLOv8.

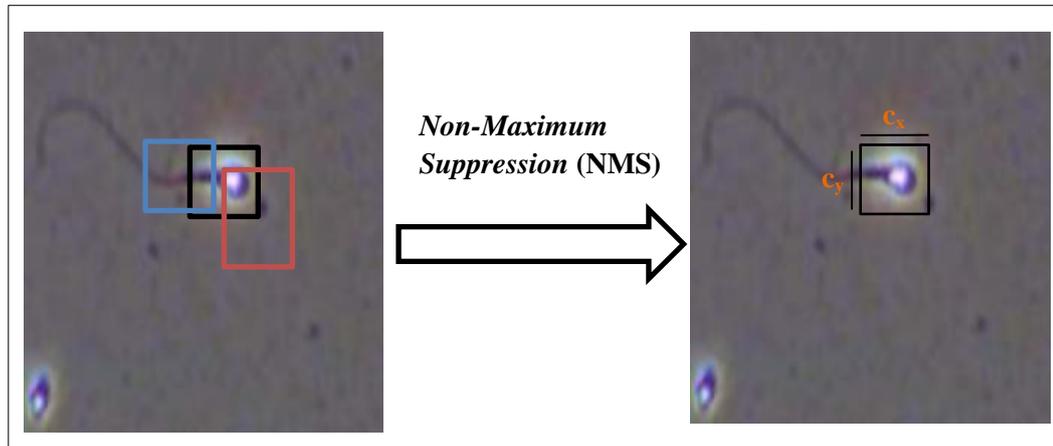


Gambar 23. Anchor Free Detection YOLOv8 (Yanto *et al.*, 2023).

2.16.5 Non-Maximum Suppression (NMS)

Non-Maximum Suppression (NMS) adalah metode yang digunakan dalam pemrosesan gambar (*image processing technique*) untuk menekan nilai atau fitur yang tidak maksimal dalam gambar. NMS dalam pemrosesan gambar digunakan untuk menghilangkan deteksi objek yang tumpang tindih dan hanya menyisakan

kotak deteksi terbaik dengan skor tertinggi (Amat & Luthfi, 2023). NMS memastikan bahwa model YOLO ketika terdapat *double bounding box* akan memilih *bounding box* dengan *high confidence score* untuk dipertahankan, dan *bounding box* lainnya akan dihilangkan.



Gambar 24. Ilustrasi *Non-Maximum Suppression* (NMS).

Gambar 24 merupakan ilustrasi proses *Non-Maximum Suppression* (NMS) yang diterapkan dalam deteksi objek. Gambar sebelah kiri, terlihat beberapa *bounding box* mengalami tumpang tindih, terlihat dengan kotak warna berbeda mengelilingi objek yang sama. Gambar ini menunjukkan bahwa model mendeteksi objek lebih dari satu kali pada lokasi yang hampir sama. Proses NMS kemudian memilih *bounding box* dengan confidence score tertinggi, dan menghapus *bounding box* lain yang memiliki tingkat tumpang tindih (IoU) tinggi terhadapnya. Hasil akhir dari proses NMS dapat dilihat pada gambar di bagian kanan, di mana hanya satu *bounding box* yang tersisa. Proses ini membantu meningkatkan akurasi deteksi dan mencegah duplikasi dalam penghitungan objek.

2.17 *Hyperparameter*

Hyperparameter merupakan suatu parameter yang dapat digunakan untuk mengatur dan mengontrol sebuah metode permodelan sehingga dapat menghasilkan performa yang lebih baik. *Hyperparameter* berbeda dengan parameter lain yang bisa saja berubah ketika atau seiring dengan berjalannya proses pelatihan data, parameter ini tidak dapat diubah-ubah karena nilainya

ditetapkan sebelum proses pelatihan data. *Hyperparameter* penting digunakan dan didefinisikan karena dapat mempengaruhi keandalan dan performa model tetapi tidak berpengaruh pada performa model. Meningkatkan performa model membutuhkan pengaturan beberapa *hyperparameter* seperti *epoch* yang merupakan nilai untuk menentukan berapa kali proses pelatihan model untuk membaca dan mempelajari dataset, *batch size* merupakan jumlah kelompok yang berisi beberapa data yang akan digunakan untuk melatih model dan data akan dibaca serta diingat oleh model dengan lebih baik *learning rate* merupakan waktu untuk mencapai titik optimal pada proses pelatihan model, semakin besar *learning rate* maka proses pelatihan akan berjalan dengan cepat dan *image size* merupakan ukuran gambar sampel. *Hyperparameter* lainnya yang digunakan yaitu momentum yang berfungsi untuk mempercepat konvergensi dengan mempertimbangkan parameter sebelumnya dan mengakumulasi arah pergerakan supaya proses pelatihan menjadi lebih stabil dan cepat, kemudian *scale* yang berfungsi untuk gambar dapat diperkecil atau diperbesar selama *training*. pengaturan untuk *optimizer* yang mempelajari bagaimana model dapat memberikan prediksi yang lebih baik dan sesuai dengan data yang digunakan selama pelatihan. Kofigurasi *hyperparameter* membantu pada proses pelatihan model dengan ukuran yang ditentukan terhadap dataset secara berulang kali sampai ditemukannya kombinasi *hyperparameter* dan model yang sesuai (Putra *et al.*, 2023).

2.18 *Confusion Matrix*

Confusion Matrix adalah tabel yang menyatakan klasifikasi jumlah data uji yang benar dan jumlah data uji yang salah (Putra dan Wibowo, 2020). *Confusion Matrix* dapat digunakan untuk menghitung nilai keakuratan pada perbandingan metode untuk menilai mana metode yang terbaik (Rahayu dan Novia, 2021). Pengukuran dengan *Confusion Matrix* digunakan untuk memberikan informasi perbandingan dari hasil klasifikasi yang dilakukan oleh algoritma yang digunakan dengan hasil klasifikasi sebenarnya (Romadloni *et al.*, 2022). Evaluasi

menggunakan *Confusion Matrix* menghasilkan nilai Akurasi, Presisi, *Recall*, serta *f1-score* (Putra dan Wibowo, 2020).

Tabel 2. *Confusion Matrix*

		Prediksi	
		Positif	Negatif
Aktual	Positif	<i>True Positive (TP)</i>	<i>False Negative (FN)</i>
	Negatif	<i>False Positive (FP)</i>	<i>True Negative (TN)</i>

Keterangan :

True Positive (TP) = Prediksi positif dan data yang sebenarnya positif.

False Negative (FN) = Prediksi negatif dan data yang sebenarnya positif.

False Positive (FP) = Prediksi positif dan data yang sebenarnya negatif.

True Negative (TN) = Prediksi negatif dan data yang sebenarnya negatif.

Evaluasi menggunakan *confusion matrix* menghasilkan nilai akurasi, presisi, *recall*, serta *f1-Score* (Putra dan Wibowo, 2020).

a. Akurasi

Nilai akurasi menggambarkan seberapa akurat model dalam mengklasifikasikan dengan benar (Romadloni *et al.*, 2022). Akurasi merupakan rasio dari objek relevan yang terpilih dari seluruh objek yang dipilih, juga sebagai perbandingan antara objek yang berhasil diidentifikasi dengan benar dengan total seluruh objek yang ada. Tingkat kesalahan dihitung sebagai perbandingan antara objek yang salah diidentifikasi dengan jumlah total objek yang ada (Arini *et al.*, 2020). Proses menghitung tingkat akurasi digunakan rumus pada persamaan (5) berikut.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (5)$$

Rumus akurasi pada persamaan (5) menggambarkan rasio dari hasil prediksi yang benar terhadap keseluruhan jumlah data yang diuji. Prediksi yang benar terdiri dari dua jenis, yaitu *True Positive* (TP) dan *True Negative* (TN). *True Positive* mengacu pada jumlah data yang memang positif dan berhasil diprediksi sebagai positif oleh model, sedangkan *True Negative* adalah data yang seharusnya negatif dan juga berhasil diprediksi sebagai negatif. Prediksi yang salah ditandai dengan *False Positive* (FP) dan *False Negative* (FN). *False Positive* terjadi ketika model memprediksi data sebagai positif sementara data tersebut sebenarnya negatif, dan *False Negative* terjadi ketika model memprediksi data sebagai negatif sementara data tersebut sebenarnya positif. Akurasi untuk performa model diperoleh dari jumlah *True Positive* (TP) ditambah *True Negative* (TN), dibagi dengan total seluruh data yang terdiri dari TP, TN, FP, dan FN, kemudian dikalikan 100 persen untuk menghasilkan nilai dalam bentuk persentase (Kohsasih *et al.*, 2022).

b. Presisi

Presisi merupakan tingkat ketepatan antara informasi diinginkan oleh pengguna dengan hasil prediksi yang diberikan oleh model atau sistem (Hartanti *et al.*, 2018). Presisi menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh model (Romadloni *et al.*, 2022). Presisi digunakan untuk mengukur seberapa besar proporsi dari kelas data positif yang berhasil diprediksi dengan benar dari keseluruhan hasil prediksi kelas positif (Putra dan Wibowo, 2020).

$$Presisi = \frac{TP}{TP + FP} \times 100\% \quad (6)$$

Rumus presisi pada persamaan (6) menggambarkan rasio prediksi positif yang benar dengan hasil keseluruhan dari prediksi kelas positif. Prediksi positif yang benar yaitu *True Positive* (TP) mengacu pada jumlah data yang benar positif dan berhasil diprediksi sebagai positif oleh model, sementara prediksi yang salah ditunjukkan dengan *False Negative* (FN), merupakan data yang sebenarnya negative namun keliru memprediksi sebagai positif oleh model. Presisi untuk

performa model diperoleh dengan nilai *True Positive* (TP) dibagi dengan seluruh data nilai positif yaitu, *True Positive* (TP) dan *False Positive* (FP), kemudian dikalikan dengan 100 persen untuk menghasilkan nilai dalam bentuk persentase (Kohsasih *et al.*, 2022).

c. *Recall*

Recall adalah metrik evaluasi yang digunakan untuk mengukur seberapa banyak data positif yang berhasil ditemukan oleh model dari seluruh data yang memang seharusnya positif, *recall* yang bernilai tinggi menunjukkan bahwa sistem atau model dapat mengklasifikasikan *class object* dengan benar (Kusuma dan Mauizah, 2023). *Recall* digunakan untuk mengevaluasi kemampuan sistem untuk menemukan dokumen yang relevan terhadap *query* (Arini *et al.*, 2020) dan menunjukkan tingkat keberhasilan model dalam mendeteksi sebuah informasi yang diinginkan oleh pengguna (Hartanti *et al.*, 2018).

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (7)$$

Rumus *recall* pada persamaan (7) menggambarkan rasio antara prediksi yang benar-benar positif dengan data global yang sebenarnya positif. Prediksi yang benar bahwa data merupakan positif dan berhasil diprediksi sebagai positif ditandai dengan *True Positive* (TP), sementara data yang sebenarnya positif namun model salah prediksi sebagai negative ditandai dengan *False Negative* (FN). *Recall* fokus kepada kelengkapan dalam menangkap seluruh data positif yang ada, yang diperoleh dari jumlah data positif dari *True Positive* (TP) yang dibagi dengan keseluruhan data yang sebenarnya positif yaitu *True Positive* (TP) dan *False Negative* (FN), dan dihasilkan dalam bentuk persentase (Kohsasih *et al.*, 2022).

d. *F1-Score*

F1-Score adalah parameter ukuran keberhasilan *retrieval* yang menggabungkan presisi dan *recall*. Nilai tersebut didapat dari perhitungan perkalian nilai presisi

dan *recall* yang kemudian dibagi dengan hasil penjumlahan presisi dan recall dan dikalikan dua seperti pada persamaan (8) (Arifin *et al.*, 2021).

$$F1 - Score = 2 \times \frac{Recall \times Presisi}{Recall + Presisi} \quad (8)$$

Rumus *F1-Score* pada persamaan (8) menunjukkan bahwa nilai ini diperoleh dari kombinasi antara *recall* dan presisi. *Recall* merupakan kemampuan model untuk mendeteksi seluruh data positif yang berhasil ditemukan, sementara presisi merupakan seberapa benar hasil prediksi positif yang dihasilkan model. *F1-Score* digunakan untuk menyeimbangkan *recall* dan presisi, dengan menghitung rata-rata antara *recall* dan presisi. Nilai *F1-Score* tergantung daripada nilai *recall* dan presisi, jika nilai keduanya tinggi maka nilai *F1-Score* juga akan tinggi (Kohsasih *et al.*, 2022).

2.19 Mean Average Precision (mAP)

Nilai *mean average precision* (mAP) adalah nilai rata-rata dari *average precision*, yang didapatkan dari setiap nilai *precision* item yang relevan dan menggunakan nilai 0 untuk item tidak relevan, karena tidak terdeteksi oleh sistem. Nilai mAP digunakan untuk mengevaluasi model dengan menunjukkan seberapa baik sistem dalam mendeteksi semua kelas item. Nilai mAP dihitung berdasarkan dari beberapa nilai AP yang telah terdeteksi (Fandisyah *et al.*, 2021). Rumus untuk menghitung mAP terdapat pada persamaan (9) berikut.

$$mAP = \frac{1}{c} \sum_{i=1}^c AP_i \quad (9)$$

Keterangan :

c = Jumlah kelas .

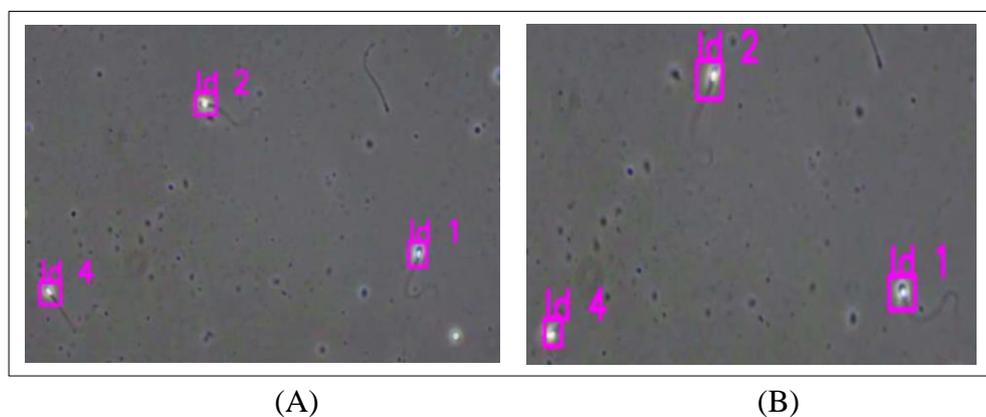
AP_i = Nilai *Average Precision* dari kelas ke- i .

$\sum_{(i=1 \text{ sampai } c)} AP_i =$ Jumlah semua AP dari tiap kelas.

Rumus mAP pada persamaan (9) digunakan untuk menghitung rata-rata dari nilai *Average Precision* (AP) pada setiap kelas objek yang terdeteksi. mAP memberikan gambaran tentang akurasi keseluruhan dari model dalam mendeteksi objek (Saputra *et al.*, 2024).

2.20 ByteTrack

ByteTrack adalah algoritma pelacakan objek untuk mendapatkan keberadaan objek pada *frame to frame* dalam video. Proses algoritma ini bekerja dengan dua langkah utama yaitu deteksi objek menggunakan algoritma deteksi dan asosiasi objek untuk menghubungkan deteksi pada objek di *frame* saat ini dengan objek di *frame* sebelumnya. ByteTrack memiliki keunggulan yang terletak pada proses asosiasi dua tahap yang menggabungkan *high-confidence* dan *low-confidence* untuk memulihkan objek yang terlewat dan tetap menjaga konsistensi pelacakan, terutama dalam lingkungan yang dinamis (Hidayat *et al.*, 2017).



Gambar 25. Proses Pelacakan Objek dengan Bytetrack.

Gambar 25 menunjukkan algoritma ByteTrack melakukan pelacakan pada masing-masing objek di *frame* saat ini dengan *frame* berikutnya secara konsisten. ByteTrack menggunakan kotak deteksi dengan nama kelas ID dan angka unik yang menjadi identitas sperma dari *frame* pertama ke *frame* selanjutnya.

2.21 Perhitungan Titik Tengah (*Centroid*)

Pelacakan objek menghasilkan titik koordinat dari *bounding box* berupa koordinat awal (x_1, y_1) dan koordinat akhir (x_2, y_2) . Titik koordinat ini yang akan digunakan untuk menghitung titik tengah (*centroid*). *Centroid* merupakan titik tengah dari setiap *bounding box* objek sperma pada masing-masing *frame*. Nilai *centroid* dihitung dengan menggunakan rumus pada persamaan (10) (Prasetya *et al.*, 2024).

$$\left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2} \right) \quad (10)$$

Keterangan :

x_1, y_1 = Titik koordinat kiri atas *bounding box*.

x_2, y_2 = Titik koordinat kanan bawah *bounding box*.

Persamaan (10) menunjukkan *centroid* dari *bounding box* yang dihitung dengan cara mengambil rata-rata dari koordinat awal (titik kiri atas) dan koordinat akhir (titik kanan bawah) *bounding box* tersebut (Valiuškaitė *et al.*, 2020).

2.22 Jarak *Euclidean*

Perhitungan titik tengah (*centroid*) *bounding box* dari objek di masing-masing *frame* menghasilkan koordinat (x, y) . Koordinat ini yang akan digunakan untuk menghitung jarak pada masing-masing objek antar *frame* menggunakan rumus jarak *euclidean*. Jarak *euclidean* adalah perhitungan jarak dari dua buah titik *euclidean space* (meliputi bidang *euclidean* dua dimensi atau tiga dimensi). Jarak *Euclidean* merupakan jarak yang umum digunakan untuk data numerik, untuk dua titik data x dan y dalam ruang d -dimensi (Mileanasari *et al.*, 2020). Perhitungan jarak pada koordinat menggunakan rumus *euclidean distance* (d) pada persamaan (11) berikut.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (11)$$

Keterangan :

$d(x, y)$ = Jarak *Euclidean* antara titik x dan y.

x_i dan y_i = Komponen ke-i dari masing-masing titik koordinat.

n = Jumlah Dimensi

Persamaan (11) digunakan untuk menghitung jarak perpindahan objek dalam ruang 2 dimensi. Titik x mewakili koordinat awal dan koordinat y mewakili koordinat akhir dari centroid. Perhitungan jarak menetapkan ambang batas sejauh 80 piksel untuk menentukan kecocokan antar *frame*. *Threshold* ini digunakan dalam konteks video mikroskopis berkecepatan tinggi, dan dianggap cukup untuk membedakan pergerakan objek yang signifikan dari *frame* satu ke *frame* berikutnya (Valiuškaitė *et al.*, 2020).

Hasil dari perhitungan jarak *Euclidean* menghasilkan satuan dalam bentuk piksel, untuk menyesuaikan pengukuran dalam dunia nyata, perlu dikonversi ke dalam satuan mikrometer (μm), dengan 1 piksel = 0.66 μm di kondisi nyata. Angka 0.66 didapatkan dari hasil kalibrasi antara mikroskop dan kamera. Konversi piksel ke mikrometer (μm) menggunakan rumus pada persamaan (12) berikut (Valiuškaitė *et al.*, 2020).

$$\text{Jarak}(\mu\text{m}) = \text{Jarak}(\text{px}) \times 0.66\mu\text{m} \quad (12)$$

Persamaan (12) menunjukkan bahwa nilai jarak (μm) didapatkan dari perhitungan nilai jarak dalam satuan piksel dan nilai konversi piksel ke mikrometer. Konversi pada persamaan (12) diperlukan untuk nilai jarak yang diperoleh dapat digunakan dalam proses perhitungan kecepatan serta relevan terhadap standar pengukuran motilitas sperma.

2.23 Kecepatan

Kecepatan adalah kemampuan suatu objek untuk melakukan gerakan-gerakan dengan cepat dalam satuan waktu tertentu. Kecepatan akan menghasilkan klasifikasi dari hasil perhitungan menggunakan rumus pada persamaan (13) berikut (Giarida & Rosyani, 2024).

$$Kecepatan = \frac{\text{Jarak}}{\text{Waktu Pengukuran}} \quad (13)$$

Persamaan (13) merupakan rumus dasar dari kecepatan, yang menyatakan bahwa kecepatan adalah hasil pembagian antara jarak yang ditempuh dengan waktu yang dibutuhkan untuk menempuh jarak tersebut. Kecepatan terkait motilitas sperma dihitung menggunakan penyederhanaan dari persamaan (13) menjadi :

$$Kecepatan = \frac{\text{Jarak } (\mu\text{m})}{\frac{1}{FPS}} \quad (14)$$

Rumus pada persamaan (14) adalah penyesuaian dari rumus dasar kecepatan ke dalam konteks analisis video, karena data posisi objek yang diperoleh dari hasil *tracking* antar *frame*, maka waktu pengukuran bukan dalam satuan detik, tetapi dihitung berdasarkan *frame per second* (FPS) dari video. Satu *frame* dalam video memiliki selang waktu $1/FPS$, sehingga waktu yang digunakan untuk menghitung kecepatan antar dua *frame* adalah $1/FPS$, kemudian rumus ini disederhanakan dari persamaan (14) menjadi rumus pada persamaan (15).

$$Kecepatan = \text{Jarak } (\mu\text{m}) \times FPS \quad (15)$$

Persamaan (15) merupakan hasil penyederhanaan dari persamaan (14), karena membagi dengan $1/FPS$ sama dengan mengalikan jarak dengan FPS. Rumus ini

digunakan dalam perhitungan kecepatan objek hasil *tracking*, dimana satuan jarak telah dikonversi ke mikrometer (μm) dan waktu yang diperoleh dari informasi FPS video.

2.24 Analisis Data

Analisis data adalah proses mencari dan menyusun secara sistematis data yang diperoleh dari hasil pengumpulan data, pengamatan atau eksperimen untuk kemudian ditafsirkan dan ditarik kesimpulan untuk dapat diinformasikan kepada informan (Fadilla & Wulandari, 2023). Analisis data salah satunya dapat dilakukan pada data kuantitatif. Metode analisis data kuantitatif adalah metode komputasi dan statistik yang berfokus pada analisis statistik, matematik atau numerik dari kumpulan data, maka dari itu; untuk menggunakan metode ini dalam penelitian kuantitatif menggunakan data yang dapat diukur atau dinumerikan. Metode penelitian data kuantitatif dimulai dari fase statistik deskriptif dan ditindaklanjuti dengan analisis yang lebih spesifik untuk mendapatkan lebih banyak wawasan (Afif *et al.*, 2023).

2.25 Pembersihan Data (*Data Cleaning*)

Data cleaning adalah tahapan pembersihan data yang melibatkan identifikasi, perbaikan, serta penghapusan kesalahan dan ketidakakuratan dalam data guna meningkatkan kualitas dan keandalan data. Tujuan utama dari pembersihan data adalah menciptakan data yang bersih, lengkap dan konsisten sehingga data tersebut dapat diandalkan dalam proses analisis dan pengambilan keputusan yang akurat. Proses pembersihan data dapat mencakup penghapusan data duplikat, perbaikan nilai yang hilang, membersihkan data dari *noise* atau *error* untuk menghindari distorsi hasil analisis (Rizky *et al.*, 2024).

2.26 Penyaringan Data (*Filtering*)

Filtering adalah proses untuk menyaring atau mengevaluasi data untuk menghilangkan nilai-nilai yang tidak relevan atau nilai ekstrem yang dapat mempengaruhi ketidakkuratan dalam proses analisis data (Muarif & Winarno, 2022). Proses *filtering* dilakukan dengan membatasi nilai *outlier* dengan melihat hasil analisis dari statistik deskriptif.

Statistik deskriptif adalah sebuah metode untuk mengumpulkan, mengolah, menabelkan, menganalisis dan menampilkan data dalam bentuk grafik. Data yang dipresentasikan kemudian akan dianalisis dan ditafsirkan untuk mendapatkan kesimpulan. Statistika deskriptif membahas tentang menyajikan data agar mudah dipahami, dengan memberikan informasi mengenai pemusatan data (*mean, median, modus*), kecenderungan gugus data (kuartil, desil, persentil), dan penyebaran data (simpangan baku, *range, varius*) (Putra *et al.*, 2023).

2.27 Perhitungan *Error* dan Akurasi

Error dan akurasi digunakan untuk mengevaluasi performa program dalam menghitung kecepatan gerak objek, dengan membandingkannya pada hasil perhitungan manual. *Error* menunjukkan besarnya perbedaan antara nilai hasil program dengan nilai manual, sedangkan akurasi menunjukkan seberapa mendekati hasil program terhadap nilai yang benar. Rumus yang digunakan untuk menghitung persentase *error* menggunakan persamaan (16) (Giarida & Rosyani, 2024).

$$Error = \frac{|Nilai Manual - Nilai Program|}{Nilai Manual} \times 100\% \quad (16)$$

Keterangan :

Nilai manual = Hasil perhitungan kecepatan objek yang dihitung tanpa program.

Nilai program = Hasil perhitungan kecepatan objek yang diperoleh dari *output* program.

$|x|$ (nilai mutlak) = Selisih absolut yang bernilai positif.

Persamaan (16) merupakan rumus *error* yang dinyatakan dalam bentuk persentase kesalahan yang menunjukkan seberapa besar perbedaan antara nilai yang dihitung secara manual dengan yang dihasilkan program. Keakuratan program dalam menghitung kecepatan gerak sperma digunakan rumus akurasi (17) berikut.

$$Akurasi = 100\% - Error \quad (17)$$

Rumus dalam persamaan (17) menunjukkan bahwa semakin kecil nilai error, maka semakin tinggi akurasi program dalam menghitung kecepatan objek. Akurasi dihitung untuk setiap objek yang bertujuan untuk menganalisis performa program dalam variasi gerakan objek.

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat

3.1.1 Tempat Penelitian

Penelitian ini dilaksanakan di Lab. Komputasi Dasar Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung yang berlokasi di Jl. Prof. Dr. Ir Soemantri Brojonegoro No. 1, Gedong Meneng, Kec. Rajabasa, Kota Bandar Lampung.

3.1.2 Waktu Penelitian

Penelitian ini dilakukan pada semester ganjil sampai semester genap dengan perkiraan waktu bulan September 2024 – Mei 2025. Tabel 3 berikut merupakan jadwal kegiatan yang akan dilakukan.

3.2 Alat Pendukung

Alat pendukung yang digunakan dalam penelitian ini, yaitu:

a. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam penelitian ini yaitu dengan spesifikasi sebagai berikut:

1. Laptop : HP Laptop 14s-cf2xxx
2. *Processor* : Intel(R) Pentium(R) CPU 6405U
3. Penyimpanan : SSD 256
4. RAM : 8,00 GB
5. GPU : Intel (R) UHD Graphics Family

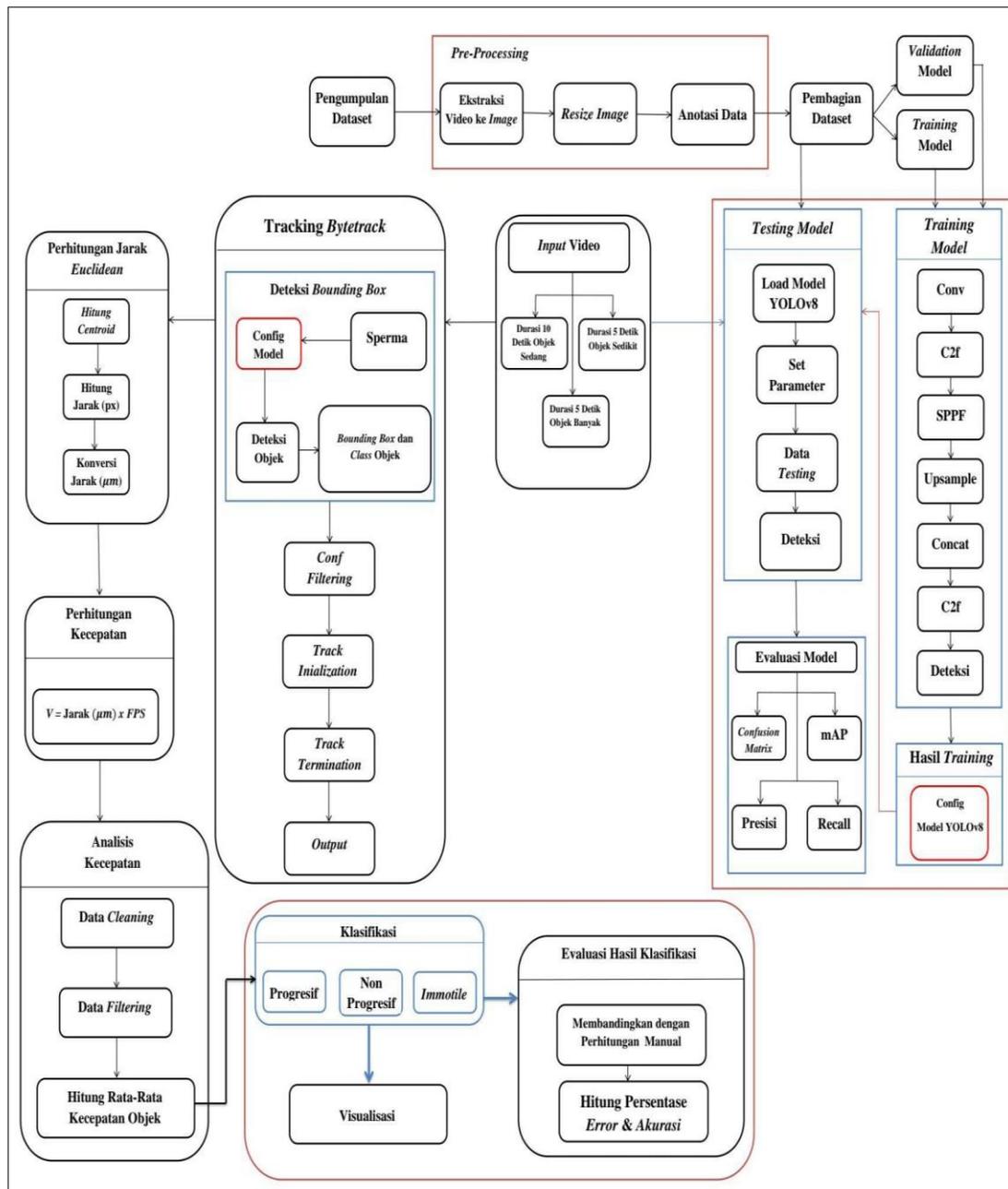
b. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam proses penelitian ini yaitu:

1. Sistem Operasi *Windows 10 Home Single Language* 64-bit.
2. *Visual Studio Code* versi 1.91.1.
3. *Python* versi 3.9.0.
4. *Google Collab*
5. *Google Drive*

3.3 Tahapan Penelitian

Dalam membuat metode terdapat tahapan-tahapan atau kerangka yang dilakukan, sehingga alur dari sebuah penelitian dapat tersampaikan garis besarnya. Gambar 26 berikut merupakan tahapan penelitian yang dilakukan.



Gambar 26. Alur Penelitian.

3.3.1 Pengumpulan Dataset

Dataset yang dikumpulkan berasal dari video sperma yang diperoleh dari *opensource* <https://datasets.simula.no/visem/>. Dataset bersifat multi modal karena mencakup berbagai sumber data seperti video, analisis biologis dan data peserta. Dataset VISEM berisi data dari 85 peserta anonim laki-laki berusia 18 tahun ke atas. Peserta laki-laki di rekrut melalui surat kabar, program penurunan berat badan, serta klinik obesitas dan fertilitas yang telah mendapat persetujuan dari Komite Etik Penelitian Medis dan Kesehatan Regional Norwegia bagian Tenggara. Video sperma terdiri dari 85 video, di mana setiap video merepresentasikan satu peserta, seperti video 1_09.09.02_SSW.avi, berasal dari peserta dengan ID 1. Perekaman video motilitas sperma ditempatkan pada *stage* mikroskop yang dipanaskan pada suhu 37°C dan diamati di bawah perbesaran 400x. Video direkam dengan menggunakan kamera yang dipasang pada mikroskop dan disimpan dalam format AVI. Kumpulan data ini berisi 35,2 *gigabyte* video yang berdurasi antara dua hingga tujuh menit. Resolusi video adalah 640 x 480 piksel dengan *frame rate* 50 FPS.

3.3.2 Pre-Processing

a. Ekstraksi Video ke Citra

Tahap pertama *pre-processing* yang dilakukan yaitu ekstraksi video ke citra, dimana data video yang berasal dari VISEM dengan format .AVI akan di ekstrak menjadi *frame-frame* citra dengan format .JPG untuk diolah lebih lanjut ke tahap *Resize Image*. Ekstensi JPG digunakan karena memiliki ukuran yang kecil tetapi tetap mempertahankan detail pada objek sehingga menghemat ruang penyimpanan. Ekstraksi dilakukan dengan 2 *frame* per 30 detik, yang menghasilkan 1330 citra dari 85 video sperma.

b. *Resize Image*

Setelah melakukan ekstraksi pada dataset VISEM, tahap kedua *pre-processing* yaitu gambar yang telah di ekstrak akan dilakukan pengukuran pada citra 640 x 480 yang

telah di ekstraksi untuk mengubah ukuran citra menjadi 640 x 640 *pixel* supaya ukuran menjadi sama pada setiap data. *Resize Image* berguna untuk memastikan kualitas data dan menghasilkan data yang lebih siap untuk diolah ke proses selanjutnya.

c. Anotasi Data

Anotasi data merupakan proses menandai kelas pada citra, pada proses ini data citra kelas yang telah selesai melakukan tahap *pre-processing* akan menjalankan ke tahap selanjutnya yaitu, anotasi data menggunakan *Lebellmg* yang bersumber dari <https://github.com/HumanSignal/labelImg>. *LabelImg* merupakan *tools* dari anaconda yang berfungsi untuk melabeli data dengan *bounding box* dan memberi nama kelas pada objek *sperm*. Tujuan dari anotasi data yaitu untuk dapat membantu program mengenali objek serta membuat label kelas dengan nama *sperm*. *Bounding box* dilakukan pada objek bagian kepala hingga leher sperma.

3.3.3 Pembagian Dataset

Pembagian dataset yang dilakukan pada penelitian ini dibagi menjadi 3 bagian yaitu data latih, data validasi dan data uji. Dataset akan dibagi menjadi 80:10:10 dengan 80% data latih 10% data validasi dan 10% data uji. Data latih merupakan sekumpulan data yang akan digunakan untuk melatih model. Data validasi merupakan data yang akan digunakan untuk validasi pada proses training model. Selanjutnya, untuk data uji merupakan sekumpulan data yang akan diuji pada model terkait.

3.3.4 Training Model

Proses *training* model YOLOv8 dilakukan menggunakan dataset gambar yang telah diproses sebelumnya. Dataset dibagi menjadi data latih dan validasi, di mana data latih digunakan untuk membangun model, sementara data validasi digunakan untuk mengevaluasi performa model secara berkala selama pelatihan. Model YOLOv8

mengenali gambar melalui serangkaian proses ekstraksi fitur menggunakan lapisan konvolusi (conv). Lapisan conv digunakan untuk menangkap fitur dasar dari gambar sperma, selanjutnya lapisan C2f (*Cross Stage Partial with Fusion*) mempertahankan informasi detail dari fitur yang telah di ekstraksi. SPPF (*Spatial Pyramid Pooling-Fast*) yang akan membantu model untuk mengenali objek dalam berbagai ukuran dengan meningkatkan konteks spasial. Fitur yang telah diekstraksi, kemudian akan diperbesar menggunakan *upsample* supaya detail tetap terjaga. Seluruh informasi akan digabungkan kembali oleh concat sebelum di proses oleh C2f kembali. Model kemudian akan menghasilkan *bounding box* dengan label kelas 'sperm'. Proses *training* menggunakan *hyperparameter* seperti *epoch*, *learning rate*, *batch size*, *optimizer ADAM*, *image size*, *momentum* dan *scale* yang disesuaikan untuk mendapat performa model yang optimal. Log evaluasi yang dihasilkan selama pelatihan mencakup metrik seperti nilai *loss*, presisi, *recall* dan *mean average precision (MAP)*.

Model yang telah selesai dilatih, akan diuji dengan dataset yang belum pernah digunakan pada proses pelatihan dan validasi. Dataset yang digunakan berupa dataset gambar dan video sperma. Tahap pengujian dilakukan untuk mengevaluasi kemampuan model dalam mendeteksi objek pada data uji. Pengujian dimulai dengan menggunakan model YOLOv8 yang telah dilatih, dengan menyesuaikan parameter seperti *confidence threshold* untuk menentukan sensitivitas deteksi. Model diterapkan pada dataset uji sebanyak 133 gambar sperma yang belum pernah digunakan dalam proses *train*, untuk di deteksi dan menampilkan hasil *bounding box* pada gambar sperma. Proses pengujian model juga dilakukan pada video sperma untuk mengamati performa model dalam kondisi dunia nyata. Tahap pengujian model tidak dilakukan evaluasi menggunakan mAP atau *confusion matrix*, melainkan hanya menampilkan hasil deteksi dari model.

Pengujian yang telah dilakukan, model selanjutnya akan di evaluasi untuk mengukur performa yang optimal dalam mendeteksi objek. Metrik yang digunakan untuk mengamati performa model yaitu *confusion matrix*, untuk mengamati kesalahan

prediksi model, serta *mean average precision* (mAP) sebagai ukuran utama performa deteksi. Presisi dan *recall* juga digunakan untuk mengukur keakuratan model dalam mendeteksi objek yang benar dibanding dengan objek yang salah. Evaluasi bertujuan untuk memastikan bahwa model yang telah dilatih memiliki performa yang optimal sebelum diterapkan pada data di dunia nyata. Hasil evaluasi yang menentukan bahwa model perlu atau tidak untuk diperbaiki atau dilatih ulang guna meningkatkan keakuratannya dalam mendeteksi objek.

3.3.5 Input Video

Sampel video sperma yang dijadikan sebagai input video untuk program perhitungan kecepatan gerak sperma memiliki durasi yang bervariasi. Video akan dilakukan pada 2 tahap, tahap pertama yaitu hanya untuk proses deteksi menggunakan model YOLOv8, yang menghasilkan file koordinat *bounding box* dari setiap objek yang terdeteksi. Tahap kedua merupakan proses deteksi yang diintegrasikan dengan algoritma pelacakan ByteTrack, yang berfungsi untuk melacak pergerakan objek dari satu *frame* ke *frame* berikutnya. Pemilihan sampel video dilakukan untuk merepresentasikan performa program klasifikasi motilitas sperma secara menyeluruh, serta analisis dapat dilakukan secara mendalam, sehingga program yang dikembangkan bersifat general dan dapat diterapkan pada video lain dalam dataset.

Tabel 4. Inputan Video Sperma

Video	Asal Video	FPS	Durasi	Objek
Video satu	14_09.08.24_SSW.avi	49	5 detik	Jumlah Objek Sedikit
Video dua	38_09.08.24_SSW.avi	49	10 detik	Jumlah Objek Sedang
Video tiga	30_09.03.25_SSW.avi	49	5 detik	Jumlah Objek Banyak

Tabel 4 merupakan daftar inputan video yang digunakan dalam proses *tracking*. Video uji yang digunakan merupakan sampel video yang berasal dari dataset visem yang digunakan dalam penelitian ini. Video satu dengan durasi 5 detik memiliki jumlah objek yang sedikit. Video dua dengan durasi 10 detik memiliki jumlah objek yang tergolong sedang, atau jumlah objek yang lebih banyak dari video satu dan video tiga memiliki durasi yang sama dengan video satu, yaitu 5 detik dengan jumlah objek yang lebih banyak dari video satu dan dua.

3.3.6 Tracking ByteTrack

Proses *tracking* menggunakan video input pada Tabel 4 dengan interval waktu dan jumlah objek yang berbeda. Video input akan dilakukan *looping* pada tiap *frame*. *Frame* akan di *resize* lalu di deteksi menggunakan model YOLOv8 yang telah dilatih dan menghasilkan *bounding box* beserta kelas objek. YOLOv8 hanya berfungsi sebagai detektor objek tanpa mengetahui apakah objek di *frame* saat ini sama dengan objek di *frame* berikutnya, untuk itu pelacakan objek dilakukan menggunakan ByteTrack untuk melacak pergerakan objek antar *frame*.

ByteTrack melakukan pelacakan dengan menyaring deteksi berdasarkan tingkat *confidence score*. Objek dengan *high-confidence* digunakan dalam *tracking* utama sedangkan objek dengan *low-confidence* masih dipertimbangkan jika tidak ada objek *high-confidence* yang cocok. Proses selanjutnya, ByteTrack akan mencocokkan *bounding box* saat ini dengan *bounding box* sebelumnya menggunakan metode *Intersection Over Union (IOU)*. IOU digunakan untuk mengukur tingkat kesamaan antara *bounding box frame* saat ini dengan sebelumnya, jika IOU lebih besar dari (*match_thresh* = 0.8), maka objek dianggap sama, tetapi jika *bounding box* tidak cocok dengan objek yang sudah ada, ByteTrack akan membuat ID baru untuk objek tersebut, asalkan *confidence score* objek lebih tinggi dari 0.25. Pemberian ID terhadap objek dilakukan pada tahap *track initialization*, yaitu ketika ByteTrack mencocokkan objek yang terdeteksi antar *frame* dan menetapkan ID untuk objek baru

yang masuk frame, jika tidak ditemukan kecocokan dengan objek yang ada, maka ID baru akan diberikan selama nilai *confidence* objek tersebut berada di ambang batas yang ditentukan. Objek yang tidak muncul dalam beberapa *frame*, ByteTrack akan menghapusnya dari *tracking*. Hasil pelacakan objek berupa video, gambar dan file yang informasi terkait masing-masing objek dalam *frame*. Video dan gambar berisikan objek dengan *bounding box* dan kelas ID yang menjadi identitas sperma, sedangkan file berisikan kolom *frame*, ID, x1, y1, x2 dan y2.

3.3.7 Perhitungan Jarak *Euclidean*

Objek sperma yang telah di *tracking* akan dihitung jaraknya menggunakan rumus *Euclidean*. Koordinat awal dan koordinat akhir pada *bounding box* hasil *tracking* akan dihitung untuk mendapatkan koordinat titik tengah (*centroid*). *Centroid* digunakan untuk menghitung jarak dari pergerakan sperma menggunakan rumus *Euclidean*. Hasil dari perhitungan jarak dengan satuan *pixel*, jika jaraknya >80 piksel program akan menghitung jarak minimum dan melakukan asosiasi objek yang sesuai dengan ID objek. Hasil perhitungan jarak yang masih dalam satuan piksel, akan dikonversi menjadi satuan mikrometer (μm), karena ukuran 1 piksel di video input yaitu 0,0002 meter dengan perbesaran mikroskop 400x maka 1 piksel pada video setara dengan 0,66 μm pada keadaan asli.

3.3.8 Perhitungan Kecepatan

Klasifikasi motilitas sperma berdasarkan kelasnya ditentukan oleh kecepatan gerak sperma. Waktu yang dibutuhkan oleh sperma progresif untuk bergerak dalam 1 detik memiliki kecepatan >20 $\mu\text{m}/\text{detik}$. Perhitungan kecepatan melibatkan jarak dalam satuan micrometer (μm) dan jumlah FPS masing masing video input. *Output* dari perhitungan kecepatan berupa file yang berisi informasi terkait kecepatan gerak objek sepanjang video sperma.

3.3.9 Analisis Hasil Perhitungan Kecepatan

Tahapan ini merupakan proses analisis pada program perhitungan kecepatan gerak sperma yang dimulai dari *post-processing* data hasil pelacakan objek. Analisis data bertujuan untuk mengetahui data yang digunakan untuk perhitungan kecepatan merupakan data yang bersih dan akurat. Proses analisis data terdiri dari beberapa langkah, yaitu pembersihan data (*data cleaning*), penyaringan data (*data filtering*), perhitungan kecepatan rata-rata tiap objek yang terdeteksi dan visualisasi dalam bentuk grafik serta video.

Tahap pembersihan data dilakukan untuk menghilangkan data yang tidak valid, seperti objek yang hanya muncul dalam satu *frame*, objek yang bukan merupakan sperma namun tetap terdeteksi, atau ID objek yang berpindah-pindah secara tidak wajar. Data-data ini eksekusi supaya hasil analisis menjadi lebih akurat dan representatif. Data yang di *cleaning*, akan dilanjutkan pada proses penyaringan atau *filtering*, dengan memisahkan data yang memiliki nilai ekstrem untuk tidak diikutsertakan ke dalam proses analisis utama.

Proses *post-processing* yang telah dilakukan, akan dilanjutkan pada proses berikutnya, yaitu data yang telah siap akan dihitung kembali kecepatan masing-masing objeknya kemudian dilakukan perhitungan kecepatan rata-rata dari tiap objek. Hasil perhitungan kecepatan rata-rata pada tiap objek, kemudian akan diklasifikasikan ke dalam kelas berdasarkan kecepatan gerak objek, yaitu progresif (kecepatan $>20 \mu\text{m/detik}$), non-progresif (kecepatan $10\text{-}20 \mu\text{m/detik}$) dan *immotile* (kecepatan $<10 \mu\text{m/detik}$). Hasil kecepatan yang telah dihitung dan objek yang telah diklasifikasikan, kemudian akan divisualisasikan dalam bentuk grafik dan video sperma dengan *bounding box* dan nama kelas objek. Visualisasi bertujuan untuk memudahkan pemahaman terhadap kecepatan sperma yang telah diklasifikasi ke dalam tiga kategori, yaitu progresif, non-progresif dan *immotile*.

3.3.10 Evaluasi Hasil Klasifikasi

Hasil klasifikasi dari kecepatan rata-rata tiap objek akan di evaluasi untuk mengetahui apakah proses klasifikasi yang dilakukan oleh program sesuai dengan proses klasifikasi hasil perhitungan menggunakan manual. Klasifikasi dilakukan pada masing-masing objek dengan ID yang sebagai identitas objek tersebut. Evaluasi dilakukan dengan membandingkan kecepatan masing-masing objek dari hasil perhitungan manual dengan perhitungan program dengan menggunakan persentase *error* dan akurasi, semakin rendah nilai *error* maka akurasi akan semakin tinggi dan performa model akan semakin baik.

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Hasil dan evaluasi terkait penelitian Klasifikasi Motilitas Sperma Berdasarkan Kecepatan Gerak Sperma Berbasis Model YOLOv8, terdapat beberapa kesimpulan yang dapat diambil terkait kinerja model YOLOv8 dengan algoritma ByteTrack untuk menghitung dan mengklasifikasikan motilitas sperma sesuai dengan kelas dan *threshold* yang ditentukan, yaitu progresif (kecepatan $> 20 \mu\text{m/s}$), non-progresif (kecepatan $10\text{-}20 \mu\text{m/s}$) dan *immotile* (kecepatan $< 10 \mu\text{m/s}$) sebagai berikut.

1. Penelitian ini berhasil mengimplementasikan model YOLOv8 untuk deteksi objek sperma dalam video dengan nama kelas *sperm*. Deteksi menggunakan model YOLOv8 pada skenario 100 *epoch* dan 0.002 *learning rate* menunjukkan hasil yang baik pada metrik presisi, *recall*, *F1-Score* dan mAP, dengan nilai presisi 0.824, *recall* 0.844, *F1-Score* 0.838 dan mAP 0.86.
2. Penelitian ini berhasil menghitung kecepatan gerak sperma berdasarkan koordinat hasil *tracking* ByteTrack untuk masing-masing objek pada ketiga video sampel, dengan jumlah objek sebanyak 6 objek pada video satu, 11 objek pada video dua, dan 49 objek pada video tiga.
3. Penelitian ini berhasil mengklasifikasikan motilitas sperma ke dalam tiga kategori, yaitu progresif, non-progresif, dan *immotile* berdasarkan kecepatan rata-rata setiap objek, dengan hasil klasifikasi pada video satu terdiri dari 4 objek progresif, 1 objek non-progresif dan 1 objek *immotile*, pada video dua terdiri dari 7 objek progresif, 3 objek non-progresif dan 1 objek *immotile*, dan pada video tiga terdiri dari 40 objek progresif, 5 objek non-progresif dan 4 objek *immotile*.

4. Penelitian ini berhasil mengevaluasi akurasi program dengan membandingkan hasil perhitungan program terhadap hasil manual, menghasilkan tingkat *error* sebesar 8,19% pada Video Satu, 10,95% pada Video Dua, dan 13,33% pada Video Tiga, dengan akurasi masing-masing sebesar 91,81%, 89,05%, dan 86,67%.

5.2 Saran

Rekomendasi dari hasil penelitian yang telah dilakukan untuk pengembangan lebih lanjut sebagai berikut:

1. Melakukan penambahan variasi pada dataset dengan menyertakan citra objek sperma yang memiliki kualitas visual tinggi, seperti ketajaman yang baik, fokus yang jelas, serta kontras pencahayaan yang optimal yang berguna untuk meningkatkan performa pelatihan model YOLOv8.
2. Mengembangkan algoritma *tracking* yang mampu mengatasi kehilangan deteksi, dengan cara menyesuaikan parameter atau mengintegrasikan pendekatan tambahan pada algoritma *tracking* yang digunakan.
3. Merancang dan mengimplementasikan program berbasis antarmuka pengguna (*user interface*) yang mempermudah proses otomatisasi dalam analisis kecepatan gerak sperma, dengan kemungkinan pengembangan lebih lanjut ke dalam bentuk *website* dan *mobile* apabila diperlukan.

DAFTAR PUSTAKA

- ABADIE, A., ANGRIST, J., & IMBENS, G. (2009). *Motilitas spermatozoa manusia setelah disimpan beku dengan medium TES-Tris yolc citrat (TES-TYC)*. 19(11), 1649–1654.
- Afreyna Fauziah, D., Maududie, A., & Nuritha, I. (2018). Klasifikasi Berita Politik Menggunakan Algoritma K-nearest Neighbor (Classification of Political News Content using K-Nearest Neighbor). *Berkala Sainstek*, 6, 106–114.
- Afif, Z., Azhari, D. S., Kustati, M., & Sepriyanti, N. (2023). Penelitian Ilmiah (Kuantitatif) Beserta Paradigma , Pendekatan , Asumsi Dasar, Karakteristik, Metode Analisis Data Dan Outputnya. *INNOVATIVE: Journal Of Social Science Research*, 3(3), 682–693.
- ARDIANSYAH, M. I. (2023). IMPLEMENTASI YOU ONLY LOOK ONCE v5 (YOLOv5) UNTUK IDENTIFIKASI GENIUS ANGGREK DI INDONESIA BERBASIS MOBILE. In *Nucl. Phys.* (Vol. 13, Issue 1). Universitas Lampung.
- Ardiansyah, M. R., Supit, Y., Said, M. S., Komputer, S., Tinggi, S., & Informatika, M. (2022). *Menggunakan Algoritma Yolo*. 7(1).
- Arifin, N., Enri, U., & Sulistiyowati, N. (2021). Penerapan Algoritma Support Vector Machine (SVM) dengan TF-IDF N-Gram untuk Text Classification. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, 6(2), 129.
- Arini, A.-, Wardhani, L. K., & Octaviano, D.-. (2020). Perbandingan Seleksi Fitur Term Frequency & Tri-Gram Character Menggunakan Algoritma Naïve Bayes Classifier (Nbc) Pada Tweet Hashtag #2019gantipresiden. *Kilat*, 9(1), 103–114.
- Aristoteles, Syarif, A., Sutyarso, Lumbanraja, F. R., & Hidayatullah, A. (2022). Identification of Human Sperm based on Morphology Using the You Only Look Once Version 4 Algorithm. *International Journal of Advanced Computer Science and Applications*, 13(7), 424–431.
- Aristoteles, A., Syarif, A., & Lumbanraja, F. R. (2023). Systematic Review: Perkembangan Machine Learning Pada Sperma Manusia. *Jurnal Teknoinfo*, 17(1), 112.

- Aristyanto, M. Y., & Kurniawan, R. (2021). Pengembangan Metode Neural Machine Translation Berdasarkan Hyperparameter Neural Network. *Seminar Nasional Official Statistics*, 2021(1), 935–946.
- Armelia, R. D., Andrian, R., & Junaidi, A. (2024). Perbandingan Kinerja Backpropagation dan Convolutional Neural Network untuk Klasifikasi Citra Batik Lampung. *Jurnal Komputasi*, 12(1), 11–18.
- Athania Purba, K., & Dewayanto, T. (2023). Penerapan Artificial Intelligence, Machine Learning Dan Deep Learning Pada Kurikulum Akuntansi-a Systematic Literature Review. *Diponegoro Journal of Accounting*, 12(3), 1–15.
- Azis, N., Herwanto, H., & Ramadhani, F. (2021). Implementasi Speech Recognition Pada Aplikasi E-Prescribing Menggunakan Algoritme Convolutional Neural Network. *Jurnal Media Informatika Budidarma*, 5(2).
- Azwar Amat, M., & Luthfi, M. (2023). Teknik Pemrosesan Gambar Digital pada Hasil Pengelasan TIG Aluminum Paduan untuk Aplikasi Pengukuran Lebar Manik Las. *Journal of Applied Mechanical Technology*, 2(1), 10–18.
- Bariyah, T., Rasyidi, M. A., & Ngatini, N. (2021). Convolutional Neural Network untuk Metode Klasifikasi Multi-Label pada Motif Batik. *Techno.Com*, 20(1), 155–165.
- Basith, A., & Zuhriyah, A. (2022). Kesehatan reproduksi remaja di Desa Sukowati Kecamatan Kapas Kabupaten Bojonegoro. *Jurnal Abdimas Dewantara*, 5(2), 233–241.
- Cheng, L., Zhang, D., & Zheng, Y. (2024). Road Object Detection in Foggy Complex Scenes Based on Improved YOLOv8. *IEEE Access*, 12(August), 107420–107430.
- Cooper, T. G., Noonan, E., von Eckardstein, S., Auger, J., Baker, H. W. G., Behre, H. M., Haugen, T. B., Kruger, T., Wang, C., Mbizvo, M. T., & Vogelsong, K. M. (2009). World Health Organization reference values for human semen characteristics. *Human Reproduction Update*, 16(3), 231–245.
- Dinata, M. I., Sulistianingsih, N., & Yusuf, S. A. A. (2023). Implementasi Deep Learning Dalam Klasifikasi Citra Gambar Dengan Menggunakan Metode CNN. *Journal of Information Technology System*, 14–19.
- Djampri, S. R. E., Tena, S., & Maggang, A. A. (2019). *PENGENALAN CITRA GARIS TELAPAK TANGAN DENGAN METODE DISCRETE WAVELET TRANSFORM DAN MAHALANOBIS DISTANCE* *Dasar Teori Citra Digital*. 2015, 440–449.
- Dr.Arnita, Faridawaty Marpaung, S.Si., M. S., & Fitrihuda Aulia, Nita Suryani S.Kom, R. C. N. S. K. (2022). *Computer Vision Dan Pengolahan Citra Digital* (A. B. Surya & Desain (eds.)). digilib.unimed.ac.id.

- Du, J. (2018). Understanding of Object Detection Based on CNN Family and YOLO. *Journal of Physics: Conference Series*, 1004(1).
- Dwi Hartanti, Kusriani, & Taufiq, E. L. (2018). Penerapan Naïve Bayes Dalam Prediksi Ketercapaian Nilai Kriteria Ketuntasan Minimal Siswa. *Jusikom Prima*, 2(1), 15–22.
- Efrian, M. R., & Latifa, U. (2022). Image Recognition Berbasis Convolutional Neural Network (Cnn) Untuk Mendeteksi Penyakit Kulit Pada Manusia. *Power Elektronik : Jurnal Orang Elektro*, 11(2), 276.
- Fahim, F., & Hasan, M. S. (2024). Enhancing the reliability of power grids: A YOLO based approach for insulator defect detection. *E-Prime - Advances in Electrical Engineering, Electronics and Energy*, 9(June), 100663.
- Fandisyah, A. F., Iriawan, N., & Winahju, W. S. (2021). Deteksi Kapal di Laut Indonesia Menggunakan YOLOv3. *Jurnal Sains Dan Seni ITS*, 10(1).
- Fani Nurona Cahya*, Nila Hardi, Dwiza Riana, S. H. (2024). Klasifikasi Penyakit Mata Menggunakan Convolutional Neural Network Model Resnet-50. *Jurnal Rekayasa Sistem Informasi Dan Teknologi*, 1(3), 199–206.
- Fauzi, Y., Andiono, E., & Khamali, M. (2020). Aplikasi Object Detection and Tracking Untuk Penyandang Tunanetra dengan Internet of Things (IoT) (Menggunakan Bahasa Pemrograman Phyton). *Universitas Budiluhur, Jakarta 1 Jln. Raya Cilegon Serang KM.08 Kramatwatu, 12260*, 1–6.
- Febrianti, L., Sulaeman, N., Apriansa, Alda, Y., & Atifah, Y. (2023). Pemanfaatan Biji Pepaya (*Carica papaya*) Untuk Menurunkan Produksi Sperma Pada Mus musculus Jantan. *Prosiding ...*, 752–762.
- Fuad, A. (2020). Prediksi Tingkat Kualitas Kesuburan Pria Dengan Jaringan Saraf Tiruan Backpropagation. *JIKO (Jurnal Informatika Dan Komputer)*, 3(2),
- Gelar Guntara, R. (2023). Pemanfaatan Google Colab Untuk Aplikasi Pendeteksian Masker Wajah Menggunakan Algoritma Deep Learning YOLOv7. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 5(1), 55–60.
- Gholamalinezhad, H., & Khosravi, H. (2020). *Pooling Methods in Deep Neural Networks, a Review*.
- Gibran, H., Purnama, B., & Kosala, G. (2024). Pengoptimasian Pengukuran Kepadatan Jalan Raya Dengan Cctv Menggunakan Metode Yolov8. *Technomedia Journal*, 9(1), 31–45.
- Gunawan, R., Hanafie, D. M. I., & Elanda, A. (2024). Klasifikasi Jenis Ras Kucing Dengan Gambar Menggunakan Convolutional Neural Network (CNN). *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, 18(4), 1–8.

- Halbouni, A., Gunawan, T. S., Habaebi, M. H., Halbouni, M., Kartiwi, M., & Ahmad, R. (2022). Machine Learning and Deep Learning Approaches for CyberSecurity: A Review. *IEEE Access*, *10*(MI), 19572–19585.
- Herdianto, & Nasution, D. (2022). Klasifikasi Objek Menggunakan Metode Convolutional Neural Network (CNN). *Prosiding SNASTIKOM: Seminar Nasional Teknologi Informasi & Komunikasi Paper*, 330–336.
- Hidayat, H. U., Perdana, R. S., Kusyanti, A., Studi, P., Informatika, T., Komputer, F. I., & Brawijaya, U. (2017). Pengembangan Sistem Penghitungan Penumpang Otomatis dalam Kondisi Wajah Penumpang yang Dikaburkan Menggunakan Algoritma YOLOv8 dan ByteTrack. *1*(1), 1–7.
- Hidayatullah, P., Nuriyadi, M., Awaludin, I., & Kusumo, R. D. (2016). Pengukuran Tingkat Motilitas Sperma Berdasarkan Linearitas Sperma Menggunakan Metode Adaptive Local Threshold Dan Ellipse Detection. *Jurnal Informatika*, *13*(2), 44–50.
- Hussain, M. (2024a). YOLOv1 to v8: Unveiling Each Variant-A Comprehensive Review of YOLO. *IEEE Access*, *12*(March), 42816–42833. <https://doi.org/10.1109/ACCESS.2024.3378568>
- Hussain, M. (2024b). *YOLOv5, YOLOv8 and YOLOv10: The Go-To Detectors for Real-time Vision*. 1–12.
- Hutauruk, J. S. W., Matulatan, T., & Hayaty, N. (2020). Deteksi Kendaraan secara Real Time menggunakan Metode YOLO Berbasis Android. *Jurnal Sustainable: Jurnal Hasil Penelitian Dan Industri Terapan*, *9*(1), 8–14.
- Indrawan, G. T., Nilogiri, A., & Faruq, H. A. Al. (2022). Diagnosis COVID-19 Berdasarkan Citra X-ray Paru-Paru Menggunakan Metode Convolutional Neural Network. *Jurnal Smart Teknologi*, *3*(4), 340–349.
- Jinan, A., Hayadi, B. H., & Utama, U. P. (2022). Klasifikasi Penyakit Tanaman Padi Menggunakan Metode Convolutional Neural Network Melalui Citra Daun (Multilayer Perceptron). *Journal of Computer and Engineering Science*, *1*(2), 37–44.
- Kusuma, D. H., & Mauizah, M. (2023). Deteksi Lampu Lalu Lintas Menggunakan YOLO untuk Autonomous Car. *Journal of Computer Engineering, Network, and Intelligent Multimedia*, *1*(1), 21–27. Kusuma, T. A. A. H., Usman, K., & Saidah, S. (2021). *Jurnal Teknik Informatika (Jutif)*, *2*(1), 57–66.
- Kohsasih, K. L., & Situmorang, Z. (2022). Analisis Perbandingan Algoritma C4.5 dan Naïve Bayes Dalam Memprediksi Penyakit Cerebrovascular. *Jurnal Informatika*, *9*(1), 13–17.
- Lei, C., Zeng, J., Xia, Y., & Pang, F. (2024). Aircraft type recognition based on YOLOv8. *Journal of Physics: Conference Series*, *2787*(1), 6–11.

- Lu, Z., Zhu, X., Guo, H., Xie, X., Chen, X., & Quan, X. (2024). FishFocusNet: An improved method based on YOLOv8 for underwater tropical fish identification. *IET Image Processing*, July, 1–16.
- Manurung, D. G., Pinasthika, M. R., Vasya, M. A. O., Putri, R. A. D. S., Tampubolon, A. P., Prayata, R. F., Nisa, S. K., & Yudistira, N. (2024). Deteksi Dan Klasifikasi Hama Potato Beetle Pada Tanaman Kentang Menggunakan YOLOV8. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 11(4), 723–734.
- Marçal, D., Câmara, A., Oliveira, J., & de Almeida, A. (2024). Evaluating R-CNN and YOLO V8 for Megalithic Monument Detection in Satellite Images. *Lecture Notes in Computer Science*, 14834 LNCS, 162–170.
- Marcell, R. P., Fernando, B., Andika, R. P., Patrick, A. N., Josh, B. F., Tria, F. T., & Rahmi, Y. N. (2023). Marcell R.P., Fernando B., Andika R.P., Patrick A.N., Josh B.F., Tria F.T., Rahmi Y.N. Penggunaan Bahasa Pemrograman Python untuk Menganalisis Perubahan Iklim yang Diakibatkan oleh Pemanasan Global. *Jurnal Iklim Dan Cuaca*, 1(2), 160–170.
- Mardiyah, M. I. (2020). Implementasi Deep Learning untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Kebun dan Sawah. *Universitas Islam Indonesia*, June.
- Mileanasari, F., Anisa, F., Abdillah, M. S., & Setyawan, N. (2020). *Monitoring of Physical Distance for Covid-19*. 33–38.
- Muarif, A. S., & Winarno, E. (2022). Sistem Rekomendasi Tempat Parkir di Kota Lama Semarang Menggunakan Collaborative Filtering. *Jurnal Ilmiah Universitas Batanghari Jambi*, 22(2), 906.
- Mulya, M. A., Zaenul Arif, & Syefudin. (2023). Tinjauan Pustaka Sistematis : Penerapan Metode Gabor Wavelet Pada Computer Vision. *Journal Of Computer Science And Technology (JOCSTEC)*, 1(2), 83–88.
- Nasional, S., Elektro, T., Informasi, S., & Informatika, T. (2024). *Penerapan Algoritma YOLO yang Efektif Digunakan Untuk Deteksi Citra Fundus Retina*. 165–171.
- Nasution, A. H., & Utara, M. S. (2020). *Vol. X No. 2, Januari - Juni 2020 ISSN: 2086-4205*. X(2), 72–86.
- NURHIKMAT, T. (2018). IMPLEMENTASI DEEP LEARNING UNTUK IMAGE CLASSIFICATION MENGGUNAKAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK (CNN) PADA CITRA WAYANG GOLEK [Universitas Islam Indonesia]. In *Universitas Islam Indonesia* (Vol. 6, Issue 1).
- Nusman, B., Rahman, A. Y., Putera, R. P., Malang, U. W., Technology, C., & Brawijaya, U. (2024). *LOBSTER AGE DETECTION USING DIGITAL VIDEO-BASED YOLO V8*. 5(4), 1155–1163.

- Pakpahan, R. (2021). Analisa Pengaruh Implementasi Artificial. *Journal of Information System, Informatics and Computing*, 5(2), 506–513.
- Pangaribuan, H. (2019). Optimalisasi Deteksi Tepi Dengan Metode Segmentasi Citra. *Information System Development (ISD)*, 4(1), 30–38.
- Prasetya, H., Setianingsih, C., & Prasasti, A. L. (2024). Sistem Penghitung Manusia Menggunakan Drone. *Jurnal Nasional SAINS Dan TEKNIK*, 2(1), 29–32.
- Prima, Y., Marbun, D., Ginting, G., & Lubis, I. (2017). Aplikasi Pembelajaran Sistem Reproduksi Pada Manusia Dengan Metode Computer Based Instruction. *Majalah Ilmiah INTI*, 12(2), 245–251.
- Prof. Dr. Ir. Trinil Susilawati, M. (2011). *SPERMATOLOGY*. UB Press.
- Putra, D. R. R., & Saputra, R. A. (2023). Implementasi Convolutional Neural Network (Cnn) Untuk Mendeteksi Penggunaan Masker Pada Gambar. *Jurnal Informatika Dan Teknik Elektro Terapan*, 11(3), 710–714.
- Putra, D., & Wibowo, A. (2020). Prediksi Keputusan Minat Penjurusan Siswa SMA Yadika 5 Menggunakan Algoritma Naïve Bayes. *Prosiding Seminar Nasional Riset Dan Information Science (SENARIS)*, 2, 84–92.
- Putra, W. P. N., Pradana, A. I., & Nurchim. (2024). Implementasi Sistem Penghitungan Volume Kendaraan Menggunakan. *Jurnal Fasilkom*, 14(2), 443–450.
- Rahayu, W. I., Prianto, C., & Novia, E. A. (2021). Perbandingan Algoritma K-Means dan Naive Bayes untuk Memprediksi Prioritas Pembayaran Tagihan Rumah Sakit Berdasarkan Tingkat Kepentingan pada PT. Pertamina (Persero). *Jurnal Teknik Informatika*, 13(2), 1–8.
- Rahayuningsih, I., Wibawa, A. D., & Pramunanto, E. (2018). Klasifikasi Bahasa Isyarat Indonesia Berbasis Sinyal EMG Menggunakan Fitur Time Domain (MAV, RMS, VAR, SSI). *Jurnal Teknik ITS*, 7(1).
- Ratna, S. (2020). Pengolahan Citra Digital Dan Histogram Dengan Phyton Dan Text Editor Phycharm. *Technologia: Jurnal Ilmiah*, 11(3), 181.
- Ratnawati, D., Isnaini, N., & Susilawati, T. (2017). Pemanfaatan casa dalam observasi motilitas spermatozoa semen cair Sapi Madura dalam pengencer berbeda. *Jurnal Ilmu-Ilmu Peternakan*, 27(1), 80–95.
- Ratnawati, D., Isnaini, N., & Susilawati, T. (2019). Factors Affecting Spermatozoa Motility Analysis using CASA. *Indonesian Bulletin of Animal and Veterinary Sciences*, 29(3), 145.
- Redmon Joseph, Santosh Divvala, Ross Girshick, A. F. (2016). You Only Look Once: Unified, Real-Time Object Detection Joseph. *ACM International Conference Proceeding Series*, 1–10.

- Rezky, A., Bagir, A., Pamerean, D., & Makhrus, F. (2023). Deteksi Kecelakaan Lalu Lintas Otomatis Pada Rekaman CCTV Indonesia Menggunakan Deep Learning. *Buletin Pagelaran Mahasiswa Nasional Bidang Teknologi Informasi Dan Komunikasi*, 1(1), 1–5.
- Ridho Aji Pangestu, Basuki Rahmat, & Fetty Tri Anggraeny. (2020). Implementasi Algoritma Cnn Untuk Klasifikasi Citra Lahan Dan Perhitungan Luas. *Jurnal Informatika Dan Sistem Informasi (JIFoSI)*, 1(1), 166–174.
- Rizki, M., Basuki, S., & Azhar, Y. (2020). Implementasi Deep Learning Menggunakan Arsitektur Long Short Term Memory(LSTM) Untuk Prediksi Curah Hujan Kota Malang. *Jurnal Repositor*, 2(3), 331–338.
- Rizki, I. N., Prayoga, D., Puspita, M. L., & Huda, M. Q. (2024). Implementasi Exploratory Data Analysis Untuk Analisis Dan Visualisasi Data Penderita Stroke Kalimantan Selatan Menggunakan Platform Tableau. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(1). <https://doi.org/10.23960/jitet.v12i1.3856>
- Rizky Fadilla, A., & Ayu Wulandari, P. (2023). Literature Review Analisis Data Kualitatif: Tahap Pengumpulan Data. *Mitita Jurnal Penelitian*, 1(No 3), 34–46.
- Romadloni, P., Adhi Kusuma, B., & Maulana Baihaqi, W. (2022). Komparasi Metode Pembelajaran Mesin Untuk Implementasi Pengambilan Keputusan Dalam Menentukan Promosi Jabatan Karyawan. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 6(2), 622–628.
- Rosnita, L. (2021). Sistem Pendeteksi Pola Lafadz Muhammad Pada Citra Al-Qur'an Menggunakan Metode Peirce. *Jurnal Teknologi Terapan and Sains 4.0*, 1(3), 275.
- Sahafi, A., Koulaouzidis, A., & Lalinia, M. (2024). Polypoid Lesion Segmentation Using YOLO-V8 Network in Wireless Video Capsule Endoscopy Images. *Diagnostics*, 14(5), 0–16.
- Salma, K., & Hidayat, S. (2024). Deteksi Antusiasme Siswa dengan Algoritma Yolov8 pada Proses Pembelajaran Daring. *Jurnal Indonesia : Manajemen Informatika Dan Komunikasi*, 5(2), 1611–1618.
- Saputra, D. A., & Rahman, A. Y. (2024). *DETEKSI KESEGERAN IKAN LAYUR BERDASARKAN CITRA MATA MENGGUNAKAN YOLOV8*. 8(5), 10263–10270.
- Shari, A. (2022). Seleksi Spermatozoa Pada Fertilisasi In Vitro (IVF). *Indonesian Journal of Health Science*, 2(1), 1–8.
- Sholawati, M., Auliasari, K., & Ariwibisono, F. (2022). Pengembangan Aplikasi Pengenalan Bahasa Isyarat Abjad Sibi Menggunakan Metode Convolutional Neural Network (Cnn). *JATI (Jurnal Mahasiswa Teknik Informatika)*, 6(1), 134–144.

- Sholehurrohman, R. (2021). PENGEMBANGAN YOLOv3 DENGAN FITUR EKSTRAKTOR MobileNetv2 UNTUK DETEKSI DAN KLASIFIKASI KENDARAAN BEGERAK. *TESIS - KM185401*, 137.
- Singh, C. (2023). Machine Learning in Pattern Recognition. *European Journal of Engineering and Technology Research*, 8(2), 63–68.
- Sudjana, A. W., & Supeno, H. (2021). Implementasi Deep Learning untuk Object Detection Menggunakan Algoritma YOLO (You Only Look Once) pada Rambu Lalu Lintas di Indonesia. *Universitas Pasundan*, 1–8.
- Suhartini, S. M. (2012). Sistem Reproduksi Pria. *Histologi Dasar Junqueira*, 362–374.
- Tamsir, N., R, T., H. Umar, S. A., & Rosida, V. (2023). Algoritma Naïve Bayes untuk Prediksi Kualitas Sperma. *DoubleClick: Journal of Computer and Information Technology*, 6(2), 101.
- Tyagi, H., Kumar Saroj, V., & Agarwal, A. (2024). *Evolution of YOLO: Exploring the Advancements in YOLOv8 for Real-Time Wildlife Detection*. 1–9.
- Valiuškaitė, V., Raudonis, V., Maskeliūnas, R., Damaševičius, R., & Krilavičius, T. (2021). Deep learning based evaluation of spermatozoid motility for artificial insemination. *Sensors (Switzerland)*, 21(1), 1–14.
- Wardhani, I. P., & Madenda, S. (2016). Algoritma Ekstraksi Video Frame Berdasarkan Analisis Histogram Warna HCL. *Jurnal Ilmiah KOMPUTASI*, 15(2), 77–84.
- Wardani, K. R. R., Suryalim, H., Engel, V. J. L., & Christian, H. (2023). Analisis Pemilihan Optimizer dalam Arsitektur Convolution Neural Network VGG16 dan Inception untuk Sistem Pengenalan Wajah. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, 9(2), 186.
- Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, 415, 295–316.
- Yanto, Y., Aziz, F., & Irmawati, I. (2023). Yolo-V8 Peningkatan Algoritma Untuk Deteksi Pemakaian Masker Wajah. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(3), 1437–1444.
- Yao, G. (2024). *HP-YOLOv8 : High-Precision Small Object Detection Algorithm*.
- Yasir, A., Satria, W., & Yuanda, P. (2023). Digital Image Processing Metode Median Filtering Dan Morfologi Opening Dalam Reduksi Noise Citra. *Warta Dharmawangsa*, 17(4), 1687–1701.
- Yuliyanto, A., & Kristanto, N. F. (2024). *Mengejar Kinerja Maksimal : Teknik Pengoptimalan Terkini dalam Pembelajaran Mesin SMK Telkom Purwokerto*. 2(3).

- Yuan, Z., Shao, P., Li, J., Wang, Y., Zhu, Z., Qiu, W., Chen, B., Tang, Y., & Han, A. (2024). YOLOv8-ACU: improved YOLOv8-pose for facial acupoint detection. *Frontiers in Neurorobotics*, 18.
- Zikri, G., & Perani, R. (2024). Perancangan Sistem Pengukur Kecepatan Kendaraan Berbasis Kamera Menggunakan Algoritma YOLO. *Jurnal Ilmu Komputer Dan Pendidikan*, 2(3), 1–8.
- Zhou, G., Liu, M., Wang, H., & Zheng, Y. (2024). Power equipment image enhancement processing based on YOLO-v8 target detection model under MSRCR algorithm. *International Journal of Low-Carbon Technologies*, 19(April), 1717–1724.