BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian mengenai deteksi *Diabetic Retinopathy* (DR) berbasis citra fundus telah berkembang pesat dengan memanfaatkan metode *deep learning* khususnya *Convolutional Neural Network* (CNN). Berbagai studi menunjukkan bahwa pendekatan ini mampu meningkatkan akurasi proses klasifikasi dan deteksi DR secara otomatis dibandingkan metode konvensional. Model CNN banyak digunakan dalam identifikasi tingkat keparahan DR melalui ekstraksi fitur citra secara otomatis, dan juga mampu menangani variasi kompleks pada citra fundus[49], [50]. Beberapa penelitian juga telah mengembangkan sistem berbasis *deep learning* untuk deteksi lesi spesifik seperti mikroaneurisma dan eksudat yang merupakan indikator utama dalam diagnosis DR[32].

Ringkasan penelitian yang relevan dengan topik penelitian ini ditunjukkan pada Tabel 2.1.

Tabel 2. 1 Penelitian Relevan

No	Judul	Penulis	Tahun	Hasil	Relevansi
1	A New Approach for Detecting Fundus Lesions Using Image Processing and Deep Neural Network Architecture	Carlos Santos, Marilton Aguiar, Daniel Welfer, Bruno Belloni	2022	Penelitian ini mengusulkan pendekatan baru deteksi lesi fundus DR menggunakan YOLOv5 yang dikombinasikan dengan *image processing*, augmentasi data	Sebagai landasan penggunaan arsitektur YOLO untuk deteksi lesi DR pada citra fundus. Penggunaan *dataset* DDR

No	Judul	Penulis	Tahun	Hasil	Relevansi
	Based on YOLO Model			dan *transfer learning*. Model diuji pada *dataset* DDR dan IDRiD untuk mendeteksi empat jenis lesi yaitu mikroaneurisma (MA), eksudat keras (EX), perdarahan (HE), dan eksudat lunak (SE). Hasil menunjukkan peningkatan pada F1-score dan mAP dibandingkan model *baseline*, meskipun deteksi MA masih menjadi tantangan akibat ukurannya yang sangat kecil.	dan IDRiD serta strategi augmentasi data yang sama juga menjadi acuan bagi *pipeline* YOLO pada penelitian ini. Namun YOLOv5 kesulitan mendeteksi lesi mikroskopis. Luarannya juga sebatas bounding box.
2	An Improved Microaneurysm Detection Model Based on SwinIR and YOLOv8	Bowei Zhang, Jing Li, Yun Bai, Qing Jiang,	2023	Penelitian ini mengembangkan model MA-YOLO untuk mendeteksi mikroaneurisma (MA) pada citra	Bukti keunggulan YOLOv8 yang dimodifikasi untuk deteksi

No	Judul	Penulis	Tahun	Hasil	Relevansi
		Biao Yan, Zhenhua Wang		*fluorescein angiography* (FFA). SwinIR digunakan dalam merekonstruksi citra super-resolusi, dan lapisan deteksi MA tambahan disisipkan di antara neck dan head YOLOv8. *Tranfer learning* diterapkan antara citra resolusi tinggi dan rendah, serta fungsi loss Wise-IoU dioptimalkan untuk mengatasi ketidakseimbangan distribusi sampel. Model MA-YOLO mencapai recall 88,23%, presisi 97,98%, F1-score 92,85%, dan AP 94,62% di mana hasil ini	lesi DR berukuran sangat kecil seperti MA. Meskipun akurasi YOLOv8 tinggi, model hanya berhenti di tahap lokalisasi spasial tanpa ekstraksi bentuk morfologi lesi.

No	Judul	Penulis	Tahun	Hasil	Relevansi
				melampaui SSD, RetinaNet, YOLOv5, YOLOX, dan YOLOv7.	
3	Automated Detection of Diabetic Retinopathy Lesions in Ultra-Widefield Fundus Images Using an *Attention*-Augmented YOLOv8 Framework	Lei-Si Hu, Jie Wang, Heng-Ming Zhang, Hai-Yu Huang	2025	Studi ini mengintegrasikan dua mekanisme *attention*-convEMA dan convSimAM ke dalam *backbone* YOLOv8 untuk deteksi lesi DR pada 3.388 citra fundus *ultra-widefield* (UWF) beresolusi 2.600x2.048 piksel. YOLOv8+convSimAM mencapai hasil terbaik dengan presisi deteksi perdarahan 0,910 dibandingkan YOLOv8 dasar yaitu 0,815 dan	Penggunaan YOLOv8 berbasis *attention mechanism* untuk deteksi lesi DR multi-kelas. Hal ini membuktikan YOLOv8 yang diperkuat *attention* secara signifikan meningkatkan deteksi lesi kecil. Keterbatasan penelitian ini yaitu fokus murni pada peningkatan deteksi kelas

No	Judul	Penulis	Tahun	Hasil	Relevansi
				recall 0,804 untuk eksudat keras. Model terbukti unggul dalam mendeteksi lesi kecil dan kompleks seperti *cotton wool spots* dan membrane epiretinal yang sering terlewat oleh model standar.	lesi, belum menyentuh aspek visualisasi presisi jaringan retina.
4	Diabetic Retinopathy Features Segmentation without Coding Experience with Computer Vision Models YOLOv8 and YOLOv9	Nicola Rizzieri, Luca Dall'Asta, Maris Ozoliņš	2024	Penelitian ini menguji performa YOLOv8 dan YOLOv9 pada segmentasi lesi DR (MA, HE, EX, dan diskus optikus) menggunakan 100 citra dari database MESSIDOR yang dianotasi secara manual. Augmentasi data diterapkan untuk meningkatkan keragaman data	Menunjukkan keterbatasan pendekatan YOLO-only untuk segmentasi lesi DR, yang menjadi justifikasi utama penelitian untuk menggabungkan YOLO dengan U-Net dalam *pipeline*

No	Judul	Penulis	Tahun	Hasil	Relevansi
				training. Hasilnya menjanjikan dengan Map yang kompetitif, tetapi studi ini secara eksplisit menyatakan bahwa hasilnya belum siap untuk implementasi klinis, terutama segmentasi MA sehingga merekomendasikan *preprocessing* dan ekstraksi ciri yang lebih lanjut.	hibrid. Penelitian ini juga secara empiris menunjukkan bahwa memaksa model YOLO untuk melakukan segmentasi level piksel memberikan hasil yang kurang representatif.
5	FFU-Net: Feature Fusion U-Net for Lesion Segmentation of Diabetic Retinopathy	Yifei Xu, Zhuming Zhou, Xiao Li, Nuo Zhang, Meizi Zhang, Pingping Wei	2021	Penelitian ini mengusulkan FFU-Net yaitu varian U-Net yang ditingkatkan dengan mengganti *pooling layer* menggunakan lapisan konvolusional, penambahan blok *Multiscale Feature*	Sebagai referensi kunci arsitektur U-Net untuk segmentasi lesi DR dan sebagai *baseline* perbandingan. Celah riset pada

No	Judul	Penulis	Tahun	Hasil	Relevansi
				Fusion (MSFF) pada *encoder*, dan *Contextual Channel Attention* (CCA) pada *skip connection*. Fungsi loss *Balanced Focal Loss* juga diperkenalkan sebagai solusi *class imbalance*. Model penelitian menggunakan *dataset* IDRiD, di mana FFU-Net meningkatkan performa segmentasi dibandingkan U-Net dasar sebesar 11,97% (SEN), 10,68% (IoU), dan 5,79% (DICE).	penelitian ini yaitu Pemrosesan segmentasi U-Net secara langsung pada keseluruhan citra fundus memakan komputasi tinggi dan memicu positif palsu (*noise*).
6	Segmentation of Diabetic Retinopathy Images Using Deep Feature Fused Residual	Alharbi, M., Guptra, D.	2023	Penelitian ini memperkenalkan DFFR-U-Net, yaitu arsitektur U-Net yang dimodifikasi	Pengembangan U-Net dengan blok residual untuk meningkatkan akurasi

No	Judul	Penulis	Tahun	Hasil	Relevansi
	with U-Net (DFFR-U-Net)			dengan *bottleneck* berbasis blok residual untuk segmentasi lesi DR. Model dilatih dan divalidasi pada *dataset* IDRiD dengan focus pada segmentasi perdarahan (HM), eksudat keras (HE), dan diskus optikus (OD). DFFR-U-Net menghasilkan performa optimal pada ketiga jenis lesi dibandingkan dengan metode segmentasi U-Net konvensional, khususnya lesi yang batasnya tidak jelas seperti perdarahan.	segmentasi lesi DR pada *dataset* IDRiD. Celah riset pada penelitian ini yaitu U-Net konvensional bisa rentan kehilangan fitur spesifik jika citra masukannya tidak difilter terlebih dahulu.
7	Hemorrhage Semantic Segmentation in Fundus	Skouta, A., Elmoufidi, A., Jai-	2022	Penelitian ini mengembangkan model CNN berbasis fusi dua	Referensi arsitektur berbasis U-Net ganda

No	Judul	Penulis	Tahun	Hasil	Relevansi
	Images for the Diagnosis of Diabetic Retinopathy by Using a Convolutional Neural Network	Andalousi, S., Ouchetto, O.		arsitektur U-Net untuk melakukan segmentasi empat jenis lesi retina pada citra fundus menggunakan *dataset* IDRiD dan DIARETDB1. Model *GlobalNet* menunjukkan performa terbaik untuk segmentasi perdarahan dan eksudat lunak, sedangkan model fusi unggul untuk eksudat keras dan MA.	untuk segmentasi multi-lesi DR dan diperlukan penggunaan seperti CLAHE sebagai tahap *preprocessing* yang diterapkan pada *pipeline* penelitian ini agar model dapat bekerja lebih optimal.
8	Identifying the Key Components in ResNet-50 for Diabetic Retinopathy *Grading* from Fundus Images: A Systematic Investigation	Yijin Huang, Lin Li, Pujin Cheng, Junyan Lyu	2023	Penelitian ini menganalisis komponen kunci dalam framework ResNet-50 untuk *grading* DR pada *dataset* EyePACS. Penelitian ini menunjukkan performa model sangat sensitif	Referensi dalam pemilihan fungsi *loss*, strategi augmentasi, dan resolusi input untuk pelatihan model *deep learning* DR

No	Judul	Penulis	Tahun	Hasil	Relevansi
				terhadap resolusi input, fungsi *loss*, dan komposisi augmentasi data. Penggunaan *Mean Square Error* (MSE) sebagai fungsi *loss* meningkatkan metrik Quadratic Weighted Kappa (QWK).	yang menjadi pertimbangan dalam tahap pelatihan YOLO dan U-Net pada penelitian ini.
9	YOLO-U-Net Architecture for Detecting and Segmenting the Localized MRI Brain Tumore Image	Nur Iriawan, Anindya A. Pravitasari, Ulfa S. Nuraini	2024	Penelitian ini menggabungkan YOLOv3/YOLOv4 dengan U-Net berbasis FCN untuk segmentasi dalam satu *pipeline* pada citra MRI otak. Model YOLO mendeteksi dan melokalisasi area tumor dalam *bounding box*, kemudian hasilnya diteruskan ke U-Net untuk segmentasi batas	Relevan sebagai bukti konsep *pipeline* dua tahap YOLO dan U-Net di domain citra medis yang menjadi dasar arsitektur hibrid yang diusulkan pada penelitian ini untuk lesi DR.

No	Judul	Penulis	Tahun	Hasil	Relevansi
				tumor secara presisi.	
10	Enhanced TumorNet: Leveraging YOLOv8s and U-Net for Superior Brain Tumor Detection and Segmentation Utilizing MRI Scans	Waqas Zafar, et al.	2024	Penelitian ini menggunakan pendekatan model hibrid YOLOv8s dan U-Net untuk deteksi serta segmentasi tumor otak pada citra MRI. Model dilatih pada *dataset* TCIA dan CGA menghasilkan presisi 97,8%, akurasi 98,6%, *recall* 95,2%, F1-score 96,3% dan ROC-AUC 98,5%. Pendekatan dua tahap ini lebih unggul dibanding segmentasi *standalone*.	Secara langsung menunjukkan efektivitas strategi YOLO kemudian ROI yang menjadi input ke U-Net dalam citra medis, dengan performa yang tinggi.
11	*Deep Learning*-Based Prediction of	Gouda Alwakid, Walaa Gouda,	2023	Penelitian ini membandingkan dua scenario klasifikasi DR	Penggunaan CLAHE sebagai tahapan wajib

No	Judul	Penulis	Tahun	Hasil	Relevansi
	Diabetic Retinopathy Using CLAHE and ERSGAN for Enhancement	Mohammad Humayun, N.Z. Jhanjhi		dengan *preprocessing* CLAHE + ESRGAN dibandingkan yang tanpa *preprocessing*. Hasil penelitian ini membuktikan bahwa CLAHE secara dramatis meningkatkan visibilitas fitur retina dan kinerja model *deep learning* untuk DR.	*preprocessing* pada *pipeline* penelitian sebelum data dimasukkan ke model YOLO maupun U-Net.

2.1.1 State of the Art dan Gap Penelitian

Berdasarkan penelitian terdahulu, pendekatan YOLO banyak digunakan untuk mendeteksi lokasi lesi DR karena kemampuan deteksi objek secara cepat. Namun, *output* YOLO umumnya masih terbatas pada *bounding box*, sehingga belum memberikan batas area lesi secara detail pada tingkat piksel. Kondisi ini menjadi keterbatasan jika sistem membutuhkan informasi bentuk dan area lesi.

Di sisi lain, U-Net banyak digunakan untuk segmentasi citra medis karena mampu menghasilkan mask segmentasi yang lebih detail. Akan tetapi, segmentasi langsung pada citra fundus penuh memiliki tantangan karena area lesi DR relatif kecil dibandingkan latar belakang retina yang jauh lebih luas[33], [34]. Lesi seperti mikroaneurisma memiliki ukuran sangat kecil dan jumlah piksel yang terbatas,

sehingga model segmentasi dapat mengalami kesulitan dalam membedakan lesi dari struktur retina lain.

Gap penelitian yang diangkat dalam penelitian ini adalah perlunya evaluasi terhadap pendekatan hibrid yang menggabungkan deteksi lokasi menggunakan YOLOv8m dan segmentasi ROI menggunakan U-Net. Berbeda dari penelitian yang hanya berfokus pada YOLO-only atau U-Net-only, penelitian ini membandingkan tiga pendekatan, yaitu YOLO-only, U-Net-only full image, dan pipeline hibrid YOLO+U-Net. Perbandingan tersebut dilakukan untuk melihat apakah penggunaan ROI dari YOLOv8m dapat membantu meningkatkan hasil segmentasi U-Net dibandingkan segmentasi langsung pada citra penuh.

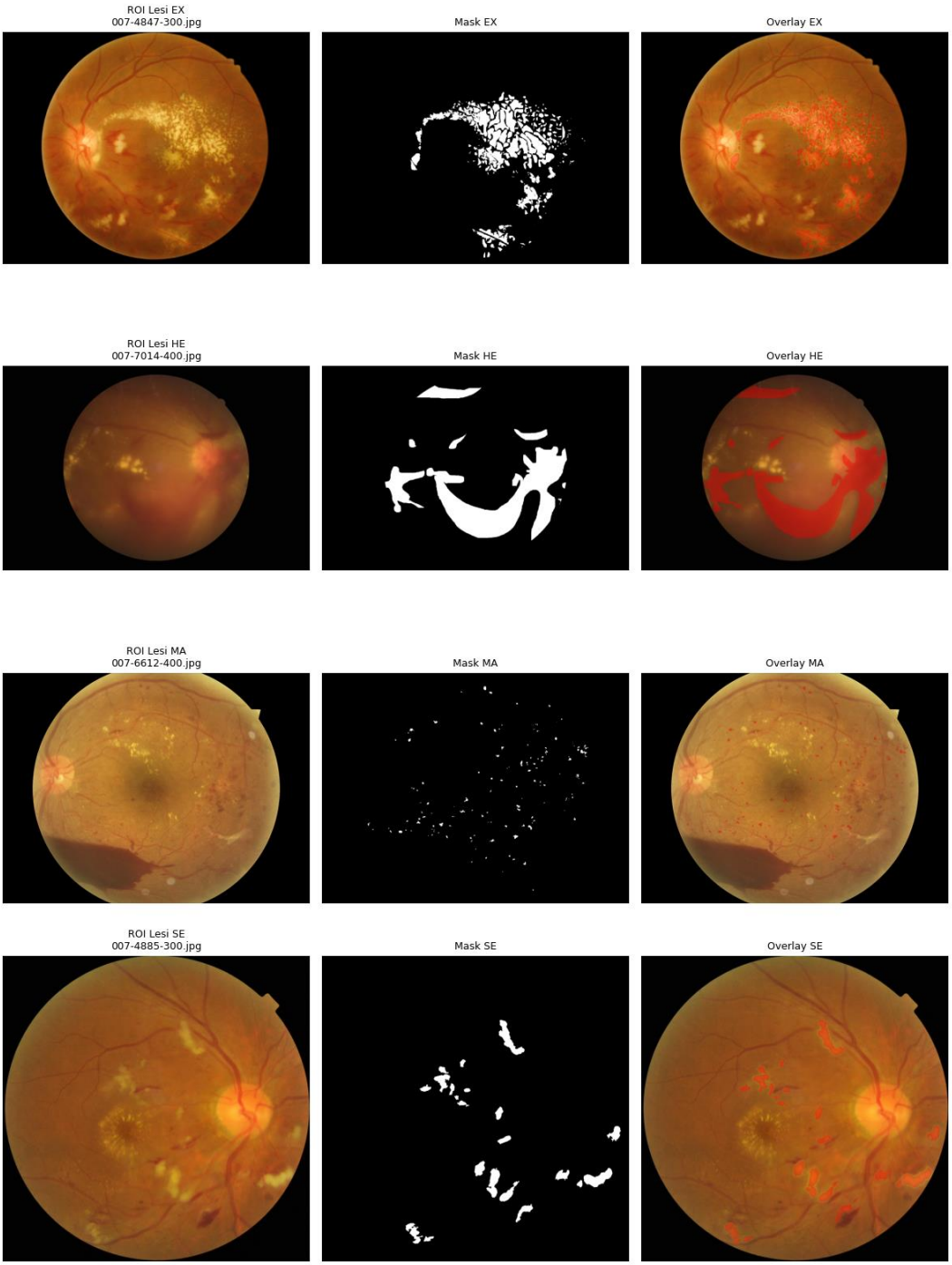
2.2 Teori yang berkaitan

2.2.1 Retinopati Diabetik

Retinopati Diabetik (DR) merupakan komplikasi mikrovaskular kronis dari diabetes melitus yang ditandai oleh kerusakan progresif pada pembuluh darah retina karena kondisi hiperglikemia dalam waktu yang lama. DR adalah salah satu penyebab utama kebutaan yang dapat dicegah pada usia produktif secara global[1]. Secara klinis, DR diklasifikasikan menjadi dua stadium utama berdasarkan ada atau tidaknya neovaskularisasi retina yaitu *Non-Proliferative Diabetic Retinopathy* (NPDR) dan *Proliferative Diabetic Retinopathy* (PDR)[51].

Stadium NPDR adalah fase awal DR yang ditandai dengan abnormalitas vaskular retina tanpa pertumbuhan pembuluh darah baru dan fase ini dibagi menjadi tiga sub-stadium[51]:

1. Mild NPDR: Setidaknya satu mikroaneurisma pada retina mata.
2. Moderate NPDR: Ditemukan beberapa mikroaneurisma, perdarahan *dot-and-blot*, *venous beading*, dan *cotton-wool-spots*.
3. Severe NPDR: Stadium ini didiagnosis berdasarkan aturan “4-2-1 rule”, yaitu perdarahan difus intraretinal di 4 kuadran, *venous beading* di 2 kuadran, atau *Intraretinal Microvascular Abnormalities* (IRMA) di 1 kuadran.



Gambar 2. 1 Lesi pada Citra Fundus Mata DDR

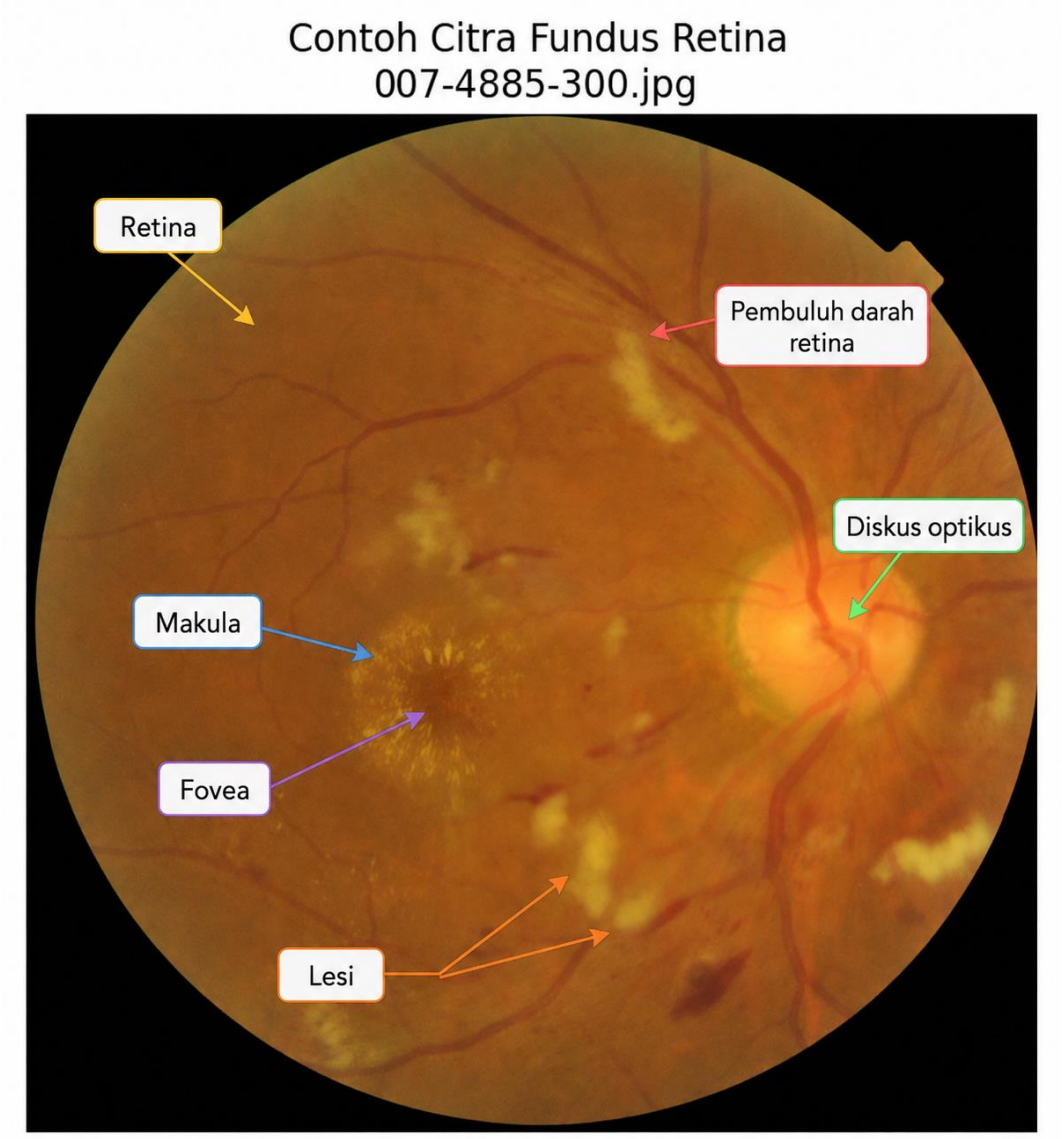
Stadium PDR ditandai oleh neovaskularisasi retina atau diskus optikus akibat iskemia retina yang progresif, di mana pembuluh darah baru yang terbentuk sangat rapuh dan rentan rupture sehingga dapat menyebabkan perdarahan vitreus dan

kebutaan permanen[51]. Terdapat empat lesi utama yang menjadi target deteksi dan segmentasi yang teranotasi pada *dataset* DDR, sebagaimana yang ditunjukkan pada Gambar 2.1:

1. Mikroaneurisma (MA): Tonjolan berbentuk bulat kecil di dinding kapiler retina akibat lemahnya dinding pembuluh darah, hal ini merupakan tanda DR paling awal. Lesi ini memiliki ukuran diameter 10-100 µm pada citra fundus[52].
2. Perdarahan (Hemorrhage/HE): Darah bocor ke jaringan retina akibat ruptur mikroaneurisma, tampak sebagai bercak merah gelap dengan batas tidak teratur[52].
3. Eksudat Keras (Hard Exudate/EX): Endapan lipid dengan warna kuning cerah yang memiliki batas area tegas karena peningkatan permeabilitas kapiler[52].
4. Eksudat lunak (Soft Exudate/SE): Eksudat lunak disebut juga sebagai *cotton-wool spots* yang merupakan infark kecil pada lapisan serat saraf retina, lesi ini tampak sebagai bercak putih keabu-abuan dengan batas area yang tidak tegas[52].

2.2.2 Citra Fundus Retina dan *Dataset* DDR

Citra fundus retina merupakan representasi fotografis dari permukaan bagian dalam mata yang mencakup retina, diskus optikus, makula, fovea, dan pembuluh darah retina yang ditangkap menggunakan kamera fundus dengan sudut tangkapan 30°–50°[24]. Citra ini umumnya diperoleh menggunakan kamera fundus dan digunakan untuk membantu pemeriksaan penyakit retina, termasuk Diabetic Retinopathy (DR). Pada citra fundus, struktur retina dan lesi patologis dapat diamati secara visual, sehingga citra ini sering digunakan sebagai data utama dalam pengembangan sistem deteksi dan segmentasi lesi DR.Tantangan utama dalam analisis citra fundus yaitu variasi iluminasi antar alat pengambilan citra, kontras rendah antar retina, ukuran lesi kecil seperti mikroaneurisma, dan ketidakseimbangan distribusi kelas antara piksel lesi dan non-lesi[53].



Gambar 2. 2 Contoh Citra Fundus Mata

Gambar 2.2 menunjukkan contoh citra fundus retina dari *dataset* DDR. Pada citra ini, area retina terlihat sebagai bidang melingkar berwarna oranye-kemerahan. Di sisi kanan gambar terdapat area terang berbentuk bulat yang merupakan diskus optikus, yaitu tempat keluarnya saraf optik dan pembuluh darah retina. Dari diskus optikus tersebut, tampak pembuluh darah retina yang menyebar ke berbagai arah.

Pada bagian tengah citra, sedikit ke arah kiri dari diskus optikus, terdapat area yang lebih gelap yang dapat diidentifikasi sebagai area makula dan fovea. Area ini penting karena berperan dalam penglihatan pusat. Selain struktur anatomi normal,

citra juga menunjukkan beberapa temuan lesi yang berkaitan dengan retinopati diabetik. Lesi berwarna putih-kekuningan yang tersebar pada citra dapat diidentifikasi sebagai eksudat, sedangkan bercak merah gelap yang terlihat pada beberapa area dapat mengarah pada perdarahan retina.

Dengan demikian, citra fundus tidak hanya menampilkan struktur anatomi retina, tetapi juga dapat digunakan untuk mengamati tanda-tanda retinopati diabetik. Pada penelitian ini, citra fundus digunakan sebagai input utama untuk mendeteksi dan melakukan segmentasi lesi seperti EX, HE, MA, dan SE.

Dataset utama yang digunakan dalam penelitian ini adalah *Dataset for Diabetic Retinopathy* (DDR) yang berasal dari *Chinese Academy of Sciences* dan merupakan salah satu *dataset* DR berskala besar yang banyak digunakan dalam penelitian deteksi dan segmentasi lesi DR[53]. DDR berisi 13.673 citra fundus berwarna yang dikumpulkan dari 9.598 pasien di 147 rumah sakit di 23 provinsi di China, di mana tujuh *grader* ahli mengklasifikasikan citra ke dalam enam kelas berdasarkan kualitas citra dan tingkat keparahan[53]. Sebanyak 757 citra dengan DR dipilih untuk dianotasi pada tingkat piksel-level pada empat jenis lesi utama yaitu Mikroaneurisma (MA), Perdarahan (HE), Eksudat Keras (EX), dan Eksudat Lunak (SE)[53].

DDR menyediakan dua jenis anotasi, yaitu anotasi *image-level* untuk *grading* tingkat keparahan DR sesuai skala *International Clinical Diabetic Retinopathy* (ICDR) dan anotasi piksel-level untuk empat jenis lesi DR[54]. Struktur dari *dataset* DDR ini dapat dilihat pada Tabel 2.2.

Tabel 2. 2 Struktur Dataset DDR

Split	Jumlah Citra	Tipe Anotasi	Resolusi Rata-rata
Train	6.835 (DR *grading*)/383 (segmentasi & deteksi)	*Image-level label* + *binary mask* per lesi (EX, HE, MA, SE) + *bounding box*	2.400x2.400 px

Validation	2.733 (DR *grading*)/149 (segmentasi & deteksi)	*Image-level label* + *binary mask* per lesi (EX, HE, MA, SE) + *bounding box*	1.624x1.232 px
Test	4.105 (DR *grading*)/225 (segmentasi & deteksi)	*Image-level label* + *binary mask* per lesi (EX, HE, MA, SE) + *bounding box*	2.592x1.728 px
Total	13.673 (DR *grading*)/757 (segmentasi & deteksi)		

Label klasifikasi DR pada *dataset* DDR mengikuti skala ICDR dengan lima tingkat keparahan yang dapat dilihat pada Tabel 2.3[54].

Tabel 2. 3 Label Klasifikasi DR pada Dataset DDR

Label	Tingkat Keparahan
0	No Diabetic Retinopathy
1	Mild NPDR
2	Moderate NPDR
3	Severe NPDR
4	Proliferative DR (PDR)

Dataset ini memiliki keunggulan dalam skala dan keragaman data klinis yang telah dikumpulkan dari puluhan rumah sakit sehingga lebih mempresentasikan variasi populasi pasien dunia nyata[53]. DDR juga menyediakan anotasi simultan untuk klasifikasi DR, segmentasi lesi piksel-level, dan deteksi lesi berbasis bounding box, sehingga ideal untuk penelitian dengan pendekatan hibrid[53]. *Dataset* DDR tersedia secara publik dengan lisensi CC BY 4.0 dan dapat diakses melalui https://huggingface.co/datasets/ctmedtech/DDR-dataset.

2.2.3 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah arsitektur *deep learning* yang dirancang untuk memproses data berbentuk grid spasial seperti citra digital dengan menggunakan operasi konvolusi untuk ekstraksi fitur secara hierarkis dari representasi lokal ke global[55]. CNN menjadi fondasi dari hamper seluruh metode *deep learning* modern dalam analisis citra medis karena mampu mengekstrak fitur secara otomatis tanpa bergantung pada rekayasa fitur manual[22].

1. Lapisan Konvolusi

   Operasi konvolusi pada piksel (i, j) untuk *feature map* ke-k pada layer ke-1 didefinisikan sebagai:

$$x_j^l = f\left(\sum_{i \in M_j} x_i^{l-1} * k_{ij}^l + b_j^l\right)$$

Persamaan 2. 1 Operasi Lapisan Konvolusi

Sumber: [56]

   Pada Persamaan 2.1, x_j^lmerupakan keluaran atau *feature map* ke-j pada lapisan ke-l, sedangkan x_i^{l-1}merupakan *feature map* ke-i dari lapisan sebelumnya. Simbol $M_j$menunjukkan *local receptive field*, yaitu area input yang digunakan untuk menghasilkan fitur pada posisi tertentu. Selanjutnya, k_{ij}^lmerupakan parameter kernel konvolusi yang digunakan untuk mengekstraksi fitur, simbol $*$menunjukkan operasi konvolusi, b_j^lmerupakan bias, dan $f(\cdot)$merupakan fungsi aktivasi yang diterapkan setelah proses konvolusi.

2. Fungsi Aktivasi ReLU

   Fungsi *Rectified Linear Unit* (ReLU) diterapkan di setelah tiap lapisan konvolusi untuk memasukkan non-linearitas dan mengatasi masalah *vanishing gradient*[55].

$$f_{ReLU}(x) = \max(0, x)$$

Persamaan 2. 2 Fungsi Aktivasi Rectified Linear Unit

Sumber: [57]

Keterangan:

$f_{ReLU}(x)$ adalah output dari fungsi aktivasi ReLU.

x adalah nilai input yang masuk ke fungsi aktivasi.

$\max(0, x)$ adalah operasi untuk memilih nilai terbesar antara 0 dan x.

Jika x bernilai positif, maka output ReLU adalah x.

Jika x bernilai negatif atau nol, maka output ReLU adalah 0.

3. Lapisan Max Pooling

   Max pooling mengurangi dimensi spasial *feature map* namun tetap mempertahankan fitur paling dominan dan meningkatkan invariant translasi[55]:

$$P(i,j) = \max_{(m,n)\in R_{ij}} A(m,n)$$

Persamaan 2. 3 Operasi Max Pooling pada Convolutional Neural Network

Sumber: [58]

Keterangan:

$P(i,j)$ adalah output hasil max pooling pada posisi (i,j).

i dan j adalah indeks posisi spasial pada output pooling.

max adalah operasi untuk mengambil nilai terbesar.

m, n adalah indeks posisi elemen di dalam region pooling.

R_{ij} adalah region atau area pooling pada posisi (i,j).

$A(m,n)$ adalah nilai aktivasi pada posisi (m,n) di dalam activation map.

2.2.4 *Preprocessing* Citra – CLAHE dan Normalisasi

Preprocessing adalah tahap krusial sebelum citra fundus dimasukkan ke dalam model *deep learning* karena tanpa adanya *preprocessing* yang baik, kontras rendah

dan variasi iluminasi pada citra fundus dapat menghambat kemampuan model dalam deteksi lesi berukuran kecil[37]. Penggunaan CLAHE pada citra fundus dapat membantu untuk meningkatkan akurasi model DR dibandingkan tanpa *preprocessing* tersebut[37]. Beberapa tahapan dalam *preprocessing* ini meliputi:

1. CLAHE (*Contrast Limited Adaptive Histogram Equalization*)

   CLAHE adalah teknik peningkatan kontras lokal yang membagi citra menjadi blok kecil atau *tiles* kemudian melakukan ekualisasi histogram secara adaptif pada setiap blok[59]. CLAHE berbeda dari *Histogram Equalization* (HE) global, CLAHE membatasi amplifikasi kontras menggunakan parameter *clip limit* (C) guna mencegah *noise* berlebih[59]. Proses CLAHE dilakukan melalui beberapa tahap, pertama, citra dibagi menjadi blok kontekstual (*tiles*) berukuran M×N piksel. Kemudian histogram $h(i)$ dihitung di setiap blok, yang selanjutnya dipotong pada nilai *clip limit* C.

$$H_{clip}(k) = \min\{H(k), \mathcal{C}\}$$

Persamaan 2. 4 Operasi Histogram Clipping pada CLAHE

Sumber: [60]

   Keterangan:

   $H_{clip}(k)$ adalah nilai histogram setelah proses clipping pada bin ke-k.

   $H(k)$ adalah nilai histogram awal pada bin ke-k.

   $\mathcal{C}$ adalah nilai clip limit atau batas maksimum histogram.

   k adalah indeks bin histogram.

   min adalah operasi untuk memilih nilai paling kecil antara $H(k)$ dan $\mathcal{C}$.

   Tahap selanjutnya, piksel yang terpotong akan didistribusi ulang secara merata ke seluruh bin histogram. Lalu fungsi transformasi kumulatif di terapkan pada setiap piksel dalam blok[59]:

$$f_{i,j}(n) = \frac{N-1}{M} \sum_{k=0}^{n} h_{i,j}(k)$$

Persamaan 2. 5 Fungsi Transformasi Cumulative Distribution Dunction pada CLAHE

Sumber: [61]

Berdasarkan Persamaan 2.5, fungsi transformasi CDF menghitung akumulasi nilai histogram dari intensitas 0 sampai intensitas n pada setiap region lokal citra. Nilai hasil akumulasi tersebut kemudian dinormalisasi menggunakan faktor (N-1)/M agar hasil transformasi berada pada rentang grayscale yang sesuai. Dalam CLAHE, proses ini digunakan untuk memetakan nilai intensitas piksel lama ke nilai intensitas baru sehingga kontras lokal pada citra dapat meningkat.

2. Normalisasi Piksel

   Normalisasi menyamakan rentang nilai piksel untuk mempercepat konvergensi pelatihan model[37]:

$$x_{norm} = \frac{x - \mu}{\sigma}$$

Persamaan 2. 6 Fungsi Normalisasi Piksel

Sumber: [37]

Di mana μ dan σ adalah rata-rata dan standar deviasi piksel yang pada umumnya menggunakan statistik ImageNet: μ = [0.485, 0.456, 0.406] dan σ = [0.229, 0.224, 0.225] per channel RGB.

3. Resize

   Seluruh citra diubah ke dimensi yang sama menggunakan interpolasi bilinear yaitu 640×640 piksel untuk YOLO secara *default* dan 512×512 piksel untuk U-Net.

2.2.5 You Only Look Once (YOLO) dan YOLOv8m

YOLO (*You Only Look Once*) merupakan arsitektur *deep learning* untuk deteksi objek secara *real-time* yang pertama kali dikenalkan oleh Redmon et al. pada tahun 2016[62]. YOLO memformulasikan deteksi objek sebagai masalah regresi Tunggal yang memetakan piksel citra langsung ke koordinat *bounding box* dan probabilitas kelas secara simulant dalam satu *forward pass* yang menjadikannya jauh lebih cepat dibandingkan metode *two-stage* seperti R-CNN[62].

1. Mekanisme Kerja YOLO

YOLO membagi citra input ke dalam grid S×S sel[62]. Setiap sel grid memprediksi B yaitu *bounding box* dengan output tensor berukuran:

$$S \times S \times (B \times 5 + C)$$

Setiap *bounding box* merepresentasikan lima nilai yaitu x, y, w, h, *confidence*[62]. *Confidence score* setiap *bounding box* didefinisikan sebagai:

$$\text{Confidence} = P(\text{Object}) \times \text{IoU}_{\text{truth}}^{\text{pred}}$$

2. Fungsi Loss YOLO

Fungsi loss YOLO menggabungkan empat komponen yaitu loss lokalisasi, loss *confidence* sel objek, loss *confidence* sel tanpa objek, dan loss klasifikasi[62]:

$$\mathcal{L} = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} [(x_i - \hat{x_i})^2 + (y_i - \hat{y_i})^2]$$

$$+\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left[\left(\sqrt{w_i} - \sqrt{\hat{w_i}}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h_i}}\right)^2 \right]$$

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left(C_i - \hat{C_i}\right)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{noobj} \left(C_i - \hat{C_i}\right)^2$$

$$+ \sum_{i=0}^{S^2} 1_i^{obj} \sum_{c \in classes} \left(p_i(c) - \hat{p_i}(c)\right)^2$$

Persamaan 2. 7 Fungsi Loss YOLO

Sumber: [62]

λ_{coord} = 5 adalah bobot loss lokalisasi dan λ_{noobj} 0.5 adalah bobot untuk sel tanpa objek[62]. Penggunaan akar kuadrat pada *w* dan *h* memberikan penalti yang proporsional terhadap ukuran objek untuk sehingga ketika terjadi kesalahan deteksi lesi kecil seperti MA maka akan diberikan penalti yang lebih besar.

3. Arsitektur YOLOv8

YOLOv8 dirilis Ultralytics pada Januari 2023[63]. YOLOv8 memiliki beberapa peningkatan arsitektur utama dibandingkan versi sebelumnya yaitu Backbone C2f (*Cross-Stage Partial Bottleneck with Two Convolutions*) yang menggabungkan fitur tingkat tinggi dengan informasi kontekstual guna meningkatkan akurasi[63]. *Anchor-free detection head* yang tidak membutuhkan *anchor box* predefinisi sehingga lebih fleksibel terhadap variasi ukuran dan rasio objek[63]. Kemudian peningkatan lainnya adalah *Decoupled head* yang memisahkan prediksi lokalisasi dan klasifikasi independent, kemudian fungsi loss CIoU (*Complete Intersection over Union*) untuk regresi *bounding box* yang lebih presisi[63].

$$L_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

Persamaan 2. 8 Fungsi CIoU

Sumber: [63]

ρ adalah jarak Euclidean antar pusat *bounding box*, c adalah diagonal kotak pembatas terkecil, α adalah parameter keseimbangan, dan v mengukur konsistensi rasio aspek.

4. Arsitektur YOLOv8m

YOLOv8m merupakan varian berukuran *medium* (*medium-scale*) dari lima varian YOLOv8 yang dirilis Ultralytics, yaitu *nano*, *small*, *medium*, *large*, dan *extra-large*[63]. Kelima varian tersebut dibangun di atas arsitektur dasar yang sama, namun dibedakan melalui mekanisme penskalaan gabungan (*compound scaling*) yang mengatur kedalaman (*depth*) dan lebar (*width*) jaringan secara bersamaan[63]. YOLOv8m dikonfigurasikan dengan nilai depth_multiple = 0,67 dan width_multiple = 0,75, menghasilkan jaringan dengan 295 lapisan dan sekitar 25,9 juta parameter dengan kebutuhan komputasi 79,3 GFLOPs[63].

Secara struktural, YOLOv8m mempertahankan arsitektur *backbone–neck–head* yang menjadi fondasi YOLOv8[63]. Pada *backbone*, blok C2f yang lebih dalam dibandingkan varian kecil memungkinkan ekstraksi hierarki

fitur dari tingkat rendah hingga fitur semantik tingkat tinggi secara lebih komprehensif, dilengkapi modul SPPF di ujung backbone untuk menangkap konteks spasial pada berbagai skala[63]. Bagian neck menggabungkan FPN dan PAN untuk menghasilkan fusi fitur multi-skala dua arah yang memperkuat kemampuan deteksi objek berukuran kecil, sementara detection head menggunakan desain *anchor-free* dan *decoupled* yang memisahkan cabang regresi dan klasifikasi secara independen[63].

Posisi YOLOv8m sebagai varian medium menjadikannya pilihan yang seimbang antara kapasitas representasi dan efisiensi komputasi[63]. Model ini lebih kapabel dari YOLOv8s (11,2 juta parameter) dalam menangkap pola fitur yang kompleks, namun tetap lebih efisien dibandingkan YOLOv8l (43,7 juta parameter, 165,7 GFLOPs) sehingga masih dapat dilatih dengan sumber daya komputasi yang wajar[64]. Pada tolok ukur COCO, YOLOv8m mencapai nilai AP (val) sebesar 50,2%, menjadikannya pilihan yang relevan untuk tugas deteksi objek yang membutuhkan ketelitian tinggi namun tetap mempertimbangkan keterbatasan sumber daya[64].

2.2.6 U-Net

U-Net merupakan arsitektur CNN untuk segmentasi citra semantik yang dikenalkan oleh Ronneberger et al. pada tahun 2015 yang dirancang untuk segmentasi citra medis dengan data yang labelnya terbatas[65]. Arsitektur U-Net berbentuk seperti huruf "U" terdiri dari tiga komponen utama yaitu *encoder*, *bottleneck*, dan *decoder*[65]. Ketiga komponen tersebut masing-masing dihubungkan oleh *skip connection*[65]. Setiap level *encoder* terdiri dari dua konvolusi 3×3 + ReLU yang diikuti *max pooling* 2×2 dengan *stride* 2[65]. Jumlah *feature channel* berlipat ganda pada setiap level sehingga resolusi spasial menurun sementara kedalaman fitur semantik meningkat[65]. *Skip connection* adalah inovasi kunci U-Net yang menggabungkan fitur spasial resolusi tinggi *encoder* dengan fitur semantik *decoder* untuk mempertahankan informasi lokasi batas objek yang hilang selama *downsampling*[65]:

$$F_{decoder}^{l} = \text{Conv}\left(\text{Concat}\left(F_{encoder}^{l}, \text{UpConv}\left(F_{decoder}^{l+1}\right)\right)\right)$$

Persamaan 2. 9 Operasi Decoder U-Net dengan Skip Connection

Sumber: [65]

Setiap level *decoder* terdiri dari *transposed convolution* 2×2 (*upsampling*) dilanjutkan dengan *concatenation* dengan *feature map* dari *encoder* level yang setara melalui *skip connection*, dan dua konvolusi 3×3 + ReLU sampai *segmentation mask* yang beresolusi sama dengan citra input diperoleh[65].

Segmentasi lesi DR memiliki ketidakseimbangan kelas yang tinggi, sehingga ada dua fungsi loss yang dikombinasikan yaitu:

1. Binary Cross-Entropy (BCE) Loss

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)]$$

Persamaan 2. 10 Binary Cross Entropy Loss

Sumber: [66]

2. Dice Loss

$$L_{Dice} = 1 - \frac{2\sum_i y_i \hat{y}_i}{\sum_i y_i + \sum_i \hat{y}_i + \varepsilon}$$

Persamaan 2. 11 Fungsi Dice Loss

3. Combined BCE-Dice Loss

$$L_{total} = \alpha \cdot L_{BCE} + \beta \cdot L_{Dice}$$

Persamaan 2. 12 Fungsi Combined BCE-Dice Loss

y_i adalah label *ground truth* piksel ke- i, $\hat{y}_i$ adalah prediksi model, ε mencegah pembagian dengan nol, dan α, β adalah bobot yang disesuaikan dengan karakteristik *dataset*.

U-Net memiliki sejumlah keunggulan yang menjadikannya pilihan arsitektur dasar yang kuat untuk segmentasi citra medis, khususnya dalam konteks pipeline hibrid yang dikembangkan dalam penelitian ini. Pertama, mekanisme *skip connection*

yang menghubungkan fitur dari setiap level *encoder* ke *decoder* yang setara memungkinkan model mempertahankan informasi spasial resolusi tinggi yang hilang saat downsampling, sehingga hasil segmentasi pada level piksel memiliki presisi batas (*boundary delineation*) yang tinggi[67]. Kedua, U-Net dirancang untuk dapat dilatih secara efektif dengan jumlah data berlabel yang terbatas, yang merupakan karakteristik umum dataset segmentasi citra medis termasuk dataset DR[65], [67]. Ketiga, U-Net telah terbukti secara empiris efektif dalam berbagai tugas segmentasi citra medis dan menjadi fondasi standar dalam literatur segmentasi citra medis modern[41], [68].

Dibandingkan dengan varian yang lebih kompleks, terdapat pertimbangan penting. *Attention* U-Net menambahkan attention gates pada jalur skip connection untuk menyaring fitur yang relevan dan menekan informasi yang tidak diperlukan, dengan tujuan utama memfokuskan perhatian model pada area lesi di dalam citra penuh[68]. Namun, dalam pipeline hibrid yang dikembangkan pada penelitian ini, peran fokalisasi area lesi tersebut sudah diemban oleh YOLOv8m melalui mekanisme ekstraksi ROI, sehingga U-Net hanya memproses patch ROI yang sudah terlokalisasi, bukan citra fundus penuh. Kondisi ini secara inheren mengurangi relevansi mekanisme attention gate yang dirancang untuk menangani variasi besar dalam lokasi dan ukuran objek pada citra penuh[68]. U-Net++ di sisi lain menggunakan jalur *skip* dan *dense* yang meningkatkan kapasitas model, namun dengan konsekuensi peningkatan kompleksitas komputasional yang signifikan[41]. Mengingat pelatihan dilakukan secara terpisah per kelas lesi (EX, HE, MA, SE) dengan data ROI yang berjumlah terbatas per kelas, penggunaan arsitektur yang lebih sederhana seperti U-Net standar dengan *encoder pretrained* lebih dapat diandalkan untuk menjaga stabilitas pelatihan dan menghindari *overfitting*.

2.2.7 Augmentasi Data

Augmentasi data adalah teknik memperbesar ukuran dan variasi *dataset* pelatihan secara artifisial dengan menerapkan transformasi geometris dan fotometrik pada citra asli tanpa mengubah label atau anotasinya[69]. Augmentasi penting dilakukan pada *dataset* citra medis karena pada umumnya berukuran kecil guna mencegah

overfitting dan meningkatkan generalisasi model[69]. Teknik augmentasi yang sering diterapkan pada citra fundus retina meliputi rotasi, *flipping*, *scaling*, dan konfigurasi kontras[69]. Transformasi geometris diterapkan secara bersamaan pada koordinat label[63]. Sedangkan untuk segmentasi U-Net, transformasi diterapkan secara identik pada citra dan *mask ground truth* secara bersamaan.

2.2.8 *Region of Interest* (ROI) Extraction

Region of Interest (ROI) *extraction* adalah proses memotong sub-area tertentu dari citra berdasarkan koordinat yang telah diidentifikasi oleh model deteksi[40]. Pada *pipeline* hibrid, ROI diekstrak dari *bounding box* hasil prediksi YOLO sebagai input segmentasi untuk U-Net. U-Net akan memproses area relevan dan tidak terbebani oleh area non-lesi yang dominan[40].

$$x_1 = \left(x_c - \frac{w}{2}\right) \cdot W_{img}, \quad x_2 = \left(x_c + \frac{w}{2}\right) \cdot W_{img}$$

$$y_1 = \left(y_c - \frac{h}{2}\right) \cdot H_{img}, \quad y_2 = \left(y_c + \frac{h}{2}\right) \cdot H_{img}$$

Persamaan 2. 13 Konversi Koordinat Bounding Box YOLO ke Koordinat Piksel Absolut

ROI diekstrak dari citra asli pada *region* $[y_1 : y_2, x_1 : x_2]$ kemudian diubah ke ukuran input U-Net. Koordinat ROI juga dapat diperluas dengan faktor *padding* $p \in [0.1,0.3]$:

$$x_1' = \max(0, x_1 - p \cdot w), \quad x_2' = \min(W_{img}, x_2 + p \cdot w)$$

Persamaan 2. 14 Ekspansi Koordinat Bounding Box dengan Faktor Padding dan Pembatasan Citra

Pendekatan ini terbukti meningkatkan akurasi segmentasi secara signifikan dibandingkan segmentasi langsung pada citra penuh karena model segmentasi dapat berfokus pada area lesi yang sudah di lokalisasi[40].

2.2.9 Metrik Evaluasi

Evaluasi performa model deteksi YOLO dan segmentasi U-Net dilakukan menggunakan beberapa metrik standar yaitu:

1. Confusion Matrix

True Positive (TP): Lesi diprediksi benar sebagai lesi

True Negative (TN): Non-lesi diprediksi benar sebagai non-lesi

False Positive (FP): Non-lesi salah diprediksi sebagai lesi (Type I Error)

False Negative (FN): Lesi tidak terdeteksi oleh model (Type II Error)

2. Precision

$$\text{Precision} = \frac{TP}{TP + FP}$$

Persamaan 2. 15 Metrik Evaluasi Precision

Sumber: [66]

3. Recall (Sensitivity)

$$\text{Recall} = \frac{TP}{TP + FN}$$

Persamaan 2. 16 Metrik Evaluasi Recall

Sumber: [66]

4. F1-Score

$$F_1 = 2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Persamaan 2. 17 Metrik Evaluasi F1-Score

Sumber: [66]

5. Intersection over Union (IoU)

Mengukur tabrakan antara area prediksi (A) dan *ground truth* (B)[24]:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} = \frac{TP}{TP + FP + FN}$$

Persamaan 2. 18 Metrik Evaluasi IoU

Sumber: [66]

6. Dice Coefficient (DSC)

$$\text{DSC} = \frac{2|A \cap B|}{|A| + |B|} = \frac{2TP}{2TP + FP + FN}$$

Persamaan 2. 19 Metrik Evaluasi Dice Coefficient

Sumber: [66]

7. Mean Average Precision (mAP)

   Merupakan metrik agregat untuk evaluasi deteksi objek YOLO[24]:

$$\text{AP} = \int_0^1 p(r)\ dr, \quad \text{mAP} = \frac{1}{C}\sum_{c=1}^{C} \text{AP}_c$$

Persamaan 2. 20 Metrik Evaluasi Map

Sumber: [66]

   Pada penelitian ini digunakan mAP@0.5 (IoU threshold = 0.5) dan mAP@0.5:0.95 sebagai metrik standar evaluasi YOLO[24].

8. Specificity

$$\text{Specificity} = \frac{TN}{TN + FP}$$

Persamaan 2. 21 Specificity

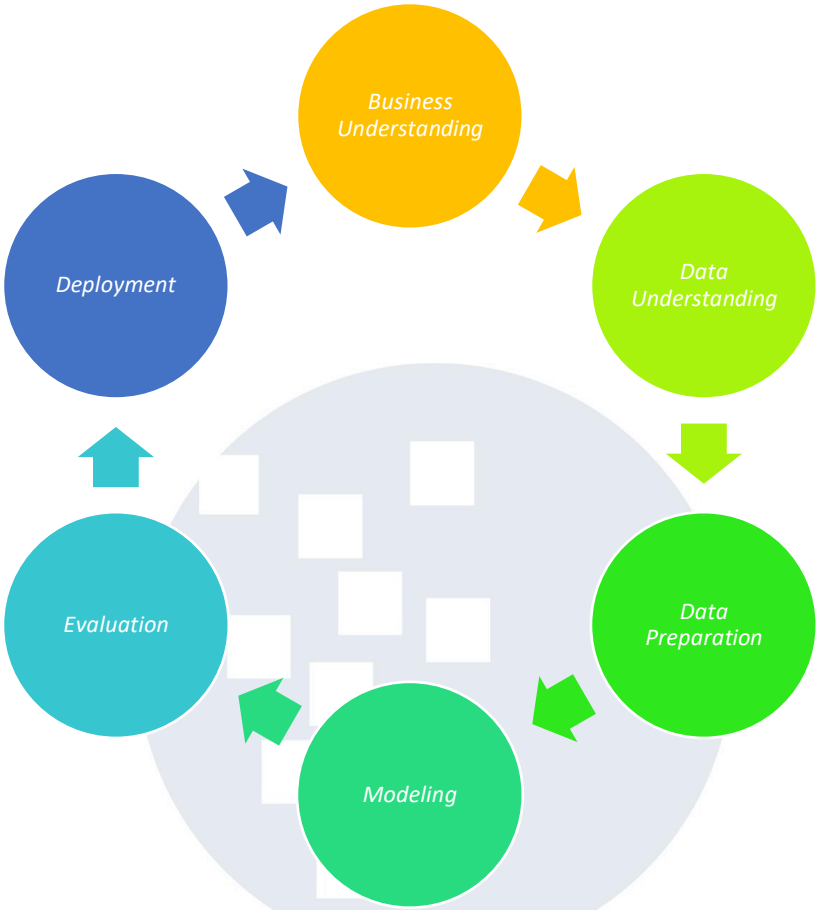
Sumber: [66]

   TN = Jumlah prediksi negatif yang benar (*True Negative*)

   FP = Jumlah prediksi positif yang salah (*False Positive*).

2.2.10 *Cross-Industry Standard Process for Data Mining* (CRISP-DM)

Cross-Industry Standard Process for Data Mining (CRISP-DM) merupakan kerangka kerja terstruktur yang menjadi standar industri dalam pengembangan proyek data mining maupun machine learning. Pendekatan ini bersifat iteratif dan adaptif, sehingga memungkinkan peneliti untuk kembali ke tahapan sebelumnya jika ditemukan ketidaksesuaian data atau anomali pada pemodelan.



Gambar 2. 3 CRISP-DM Diagram

Model CRISP-DM terdiri dari enam tahapan utama, yaitu *Business Understanding* (pemahaman masalah dan tujuan), *Data Understanding* (pengumpulan dan eksplorasi karakteristik data), *Data Preparation* (pembersihan dan penyiapan data), *Modeling* (pembangunan dan pelatihan algoritma), *Evaluation* (pengujian kinerja model), dan *Deployment* (penerapan hasil model ke dalam purwarupa atau sistem nyata). Penggunaan *framework* CRISP DM ideal digunakan karena sifatnya yang iterative dan sistematis sehingga cocok digunakan dalam penelitian berbasis data dan pengembangan model *machine learning*[70].

2.3 Framework/Algoritma yang digunakan

2.3.1 PyTorch

PyTorch merupakan *open-source deep learning framework* berbasis Python yang dikembangkan oleh Meta AI Research dan dirilis ke publik pada tahun 2016.

PyTorch menggunakan paradigma *dynamic computational graph* atau dikenal sebagai *define-by-run*, di mana graf komputasi dibangun secara dinamis saat eksekusi berlangsung. Pendekatan ini membuat PyTorch lebih intuitif dan fleksibel dalam proses pengembangan serta *debugging* model dibandingkan *framework* berbasis *static graph* seperti TensorFlow versi awal[71].

PyTorch dipilih sebagai framework utama karena ekosistem PyTorch mendukung integrasi langsung dengan library Ultralytics YOLOv8m dan *segmentation-models-pytorch* yang digunakan dalam penelitian ini. PyTorch juga menyediakan antarmuka *autograd* yang menghitung gradien secara otomatis sehingga pelatihan model menjadi lebih efisien[71]. Kemudian PyTorch juga telah banyak digunakan dalam penelitian di bidang citra medis dan terbukti menghasilkan performa yang baik[72].

Operasi utama PyTorch berbasis pada struktur data *tensor* yang merupakan generalisasi dari matriks multidimensi dan dapat diproses secara parallel menggunakan GPU melalui antarmuka CUDA[71].

$$T_{out} = f(W * T_{in} + b)$$

Persamaan 2. 22 Operasi PyTorch

Berdasarkan Persamaan 2.20, W adalah bobot (*weight*), T_{in} adalah tensor input, b adalah bias, dan f adalah fungsi aktivasi yang diterapkan tiap elemen.

2.3.2 Ultralytics YOLOv8

Ultralytics YOLOv8 adalah implementasi resmi dari arsitektur YOLOv8 yang dikembangkan oleh Ultralytics dan dirilis pada Januari 2023[63]. Framework ini menyediakan antarmuka Python untuk pelatihan, validasi, dan inferensi model deteksi objek berbasis YOLOv8. Ultralytics YOLOv8 digunakan dalam penelitian ini karena menyediakan dukungan penuh untuk tugas deteksi objek dengan format *dataset* YOLO dan kompatibel dengan *dataset* yang digunakan[63].

Ultralytics YOLOv8 menyediakan lima varian model berdasarkan ukuran dan kompleksitas. Varian-varian tersebut mencakup YOLOv8n (*nano*), YOLOv8s (*small*), YOLOv8m (*medium*), YOLOv8l (*large*), dan YOLOv8x (*extra-large*)[63].

Penelitian ini menggunakan varian YOLOv8m karena posisinya yang strategis di antara lima varian YOLOv8 yang tersedia. Secara spesifik, lima varian YOLOv8 memiliki karakteristik sebagai berikut berdasarkan jumlah parameter dan kapasitas deteksi[63], [73]: YOLOv8n (Nano) memiliki sekitar 3,2 juta parameter dan dirancang untuk deployment pada perangkat dengan sumber daya terbatas seperti mobile atau embedded system, sehingga kapasitas fiturnya tidak memadai untuk deteksi lesi kecil multi-kelas yang kompleks; YOLOv8s (Small) memiliki sekitar 11,2 juta parameter dengan kecepatan yang baik namun kapasitas fitur yang masih terbatas; YOLOv8m (*Medium*) memiliki sekitar 25,9 juta parameter yang memberikan keseimbangan antara kapasitas representasi fitur dan efisiensi komputasi; YOLOv8l (*Large*) memiliki sekitar 43,7 juta parameter dengan akurasi lebih tinggi tetapi kebutuhan komputasi yang jauh lebih besar; dan YOLOv8x (*Extra-Large*) memiliki sekitar 68,2 juta parameter yang meskipun memberikan akurasi tertinggi, membutuhkan sumber daya GPU yang jauh melebihi lingkungan komputasi yang tersedia dalam penelitian ini.

Pemilihan YOLOv8m didasarkan pada tiga pertimbangan utama. Pertama, kapasitas 25,9 juta parameter memberikan kemampuan representasi fitur yang memadai untuk mendeteksi lesi DR yang berukuran kecil dan bervariasi seperti mikroaneurisma, yang membutuhkan kapasitas model yang lebih besar dibandingkan varian ringan[24]. Kedua, varian YOLOv8m masih dapat dilatih secara stabil dalam satu sesi dengan sumber daya GPU yang tersedia (batch size 16, epoch 100), tidak seperti YOLOv8l dan YOLOv8x yang memerlukan memory GPU jauh lebih besar. Ketiga, penggunaan YOLOv8m untuk deteksi lesi DR juga sejalan dengan studi-studi terdahulu yang menunjukkan bahwa arsitektur YOLOv8 berhasil mendeteksi lesi kecil seperti eksudat dan perdarahan dengan performa tinggi pada citra fundus[24], [74].

2.3.3 Segmentation Models Pytorch (SMP)

Segmentation Models PyTorch (SMP) merupakan library Python *open-source* yang menyediakan implementasi berbagai arsitektur segmentasi citra semantik berbasis PyTorch termasuk U-Net, U-Net++, FPN, dan DeepLabV3+[67]. SMP digunakan

dalam penelitian ini karena menyediakan implementasi U-Net yang modular dan fleksibel dengan *encoder backbone* yang dapat diganti sesuai kebutuhan tanpa harus membangun ulang arsitektur.

U-Net diimplementasikan menggunakan SMP dengan konfigurasi berikut, *encoder* menggunakan ResNet-34 yang telah di-*pretrain* pada ImageNet sebagai *feature extractor*, sedangkan *decoder* menggunakan arsitektur U-Net standar dengan *skip connection*[67]. Penggunaan *pretrained encoder* dapat meningkatkan performa segmentasi pada *dataset* citra medis dibandingkan pelatihan dari awal[75]. SMP juga menyediakan fungsi loss yang relevan untuk segmentasi DR, termasuk *Dice Loss* dan *Binary Cross-Entropy Loss* yang dapat dikombinasikan sesuai karakteristik *dataset*[67].

2.3.4 OpenCV

Open Source Computer Vision Library (OpenCV) merupakan library pemrosesan citra dan visi komputer berbasis C++ dengan *binding* Python yang sering digunakan dalam penelitian di bidang pengolahan citra digital[76]. OpenCV digunakan untuk tahap *preprocessing* citra fundus, khususnya penerapan CLAHE pada *green channel* citra RGB, konversi ruang warna, dan *resize* citra sebelum dimasukan ke pelatihan model[76].

Implementasi CLAHE menggunakan OpenCV dilakukan dengan fungsi cv2.createCLAHE() dengan parameter cliplimit dan tileGridSize yang dapat dikonfigurasi sesuai karakteristik citra fundus. Penggunaan CLAHE melalui OpenCV dapat meningkatkan visibilitas lesi berukuran kecil secara signifikan[37].

2.3.5 Albumentations

Albumentations adalah library augmentasi citra berbasis Python yang digunakan untuk keperluan *computer vision* dan *deep learning[69]*. Library ini digunakan karena mendukung augmentasi simultan antara citra dan *annotation* (*bounding box* untuk YOLO atau *segmentation mask* untuk U-Net) dalam satu operasi konsisten sehingga koordinat anotasi tidak perlu dihitung manual setelah transformasi geometris diterapkan[77].

Transformasi augmenasi yang diterapkan di penelitian ini meliputi *horizontal flip*, *vertical flip*, rotasi acak sekitar 30°, perubahan kecerahan dan kontras, serta *random crop*. Albumentations dapat meningkatkan generalisasi model *deep learning* pada *dataset* medis dan mengurangi risiko *overfitting*[77].

2.4 Tools/software yang digunakan

2.4.1 Python

Python merupakan Bahasa pemrograman tingkat tinggi yang bersifat *interpreted*, *dinamis*, dan berorientasi objek yang menjadi Bahasa utama dalam pengembangan sistem berbasis *deep learning*[78]. Python digunakan dalam penelitian ini karena ekosistem library yang memadai untuk keperluan komputasi ilmiah, pengolahan citra, dan pengembangan model *machine learning*. Seluruh tahapan penelitian ini mulai dari *preprocessing*, pelatihan model YOLO, dan U-Net, hingga evaluasi performa model menggunakan Python versi 3.10.

2.4.2 Google Colab

Google Colab atau Google Colaboratory merupakan platform berbasis cloud yang menyediakan IDE berbentuk Jupyter Notebook. Platform ini digunakan untuk menjalankan dan mendokumentasikan kode Python secara langsung melalui browser tanpa perlu instalasi *environment* secara lokal. Google Colab mendukung penggunaan teks, gambar, tabel, HTML, LaTeX, dan visualisasi hasil dalam satu dokumen *notebook* sehingga cocok digunakan untuk pengembangan dan dokumentasi machine learning.

2.4.3 LabelImg

LabelImg merupakan *tool* anotasi citra berbasis grafis yang bersifat *open-source* dan digunakan untuk membuat anotasi *bounding box* pada citra secara manual[73]. Dalam penelitian ini, LabelImg digunakan untuk memverifikasi dan melakukan anotasi tambahan pada subset citra dari *dataset* DDR yang memerlukan penyesuaian anotasi. labelImg mendukung ekspor anotasi dalam format YOLO TXT yang kompatibel dengan format imput yang dibutuhkan Ultralytics YOLOv8[73].

2.4.4 Numpy

NumPy merupakan library Python fundamental untuk komputasi numerik yang menyediakan struktur data *array* multidimensi (*ndarray*) beserta berbagai fungsi matematis yang dioptimalkan untuk operasi berbasis vector dan matriks[79]. Dalam penelitian ini, NumPy digunakan secara luas pada tahap *preprocessing* citra untuk operasi normalisasi piksel, manipulasi dimensi array citra, serta konversi antara representasi citra OpenCV dan tensor PyTorch. NumPy juga digunakan dalam proses komputasi metrik evaluasi seperti IoU dan *Dice Coefficient* pada level array secara efisien[79].

2.4.5 Matplotlib

Matplotlib adalah sebuah library visualisasi data berbasis Python yang banyak digunakan untuk menghasilkan grafik dan plot dalam format yang dapat dikustomisasi[67]. Dalam penelitian ini, Matplotlib digunakan untuk memvisualisasikan kurva *loss* dan metrik evaluasi selama proses pelatihan model, menampilkan hasil deteksi *bounding box* dari YOLOv8m secara visual dan visualisasi *mask* segmentasi yang dihasilkan U-Net terhadap *ground truth* pada citra fundus yang diuji. Visualisasi ini penting untuk analisis kualitatif performa model selain dari metrik kuantitatif[67].