

BAB 2

LANDASAN TEORI

2.1 Leukemia

Leukemia adalah salah satu bentuk kanker yang menyerang jaringan pembentuk darah, khususnya sumsum tulang. Kondisi ini ditandai oleh proliferasi abnormal sel darah putih yang mengganggu proses pembentukan sel darah normal [16, 17]. Dalam penelitian ini, pembahasan leukemia difokuskan pada perubahan morfologi sel darah yang dapat diamati melalui citra mikroskopis sebagai dasar dalam proses klasifikasi menggunakan metode *deep learning*.

2.1.1 Definisi Leukemia

Penyakit ini merupakan keganasan hematologis pada jaringan pembentuk darah yang terjadi akibat proliferasi abnormal sel leukosit imatur di dalam sumsum tulang, sehingga menghasilkan sel-sel abnormal yang tidak mampu berdiferensiasi secara normal [16]. Kondisi ini ditandai oleh terjadinya proliferasi sel abnormal secara klonal yang berkaitan dengan perubahan genetik pada sel hematopoietik, sehingga menghasilkan sel-sel yang belum matang dan tidak berdiferensiasi secara normal serta mendominasi lingkungan sumsum tulang [17].

Sebagai bagian dari kelompok penyakit kanker, leukemia turut berkontribusi terhadap tingginya beban penyakit kanker secara global yang hingga kini masih menjadi persoalan kesehatan prioritas di berbagai wilayah dunia [18]. Proliferasi sel abnormal yang tidak terkendali tersebut mengganggu proses hematopoiesis, sehingga menyebabkan penurunan produksi sel darah normal. Dampak dari kondisi tersebut membuat penderita rentan mengalami sejumlah manifestasi klinis, di antaranya anemia, peningkatan kerentanan terhadap infeksi bakteri maupun virus, serta gangguan pembekuan darah [2]. Perbedaan sifat sel abnormal pada leukemia digunakan sebagai dasar dalam klasifikasi penyakit ini, yaitu berdasarkan kecepatan pertumbuhan sel menjadi akut dan kronik, serta berdasarkan asal sel menjadi mieloid dan limfoid [16].

Salah satu bentuk leukemia akut, seperti *Acute Myeloid Leukemia* (AML), ditandai dengan peningkatan jumlah sel *blast* lebih dari 20% pada sumsum tulang atau darah perifer, yang menunjukkan dominasi sel imatur dan perkembangan penyakit yang berlangsung cepat. Tingginya proporsi sel *blast* pada leukemia

akut berhubungan dengan perkembangan penyakit yang berlangsung cepat akibat dominasi sel imatur dalam sumsum tulang [19]. Kondisi ini menuntut adanya metode diagnosis yang cepat dan akurat, mengingat pemeriksaan mikroskopis konvensional masih memerlukan sumber daya waktu dan tenaga yang tidak sedikit. Oleh sebab itu, pendekatan berbasis citra dengan dukungan teknologi komputasi, khususnya metode *deep learning*, menjadi salah satu solusi yang berkembang dalam meningkatkan efisiensi dan akurasi proses identifikasi penyakit [9].

2.2 Citra Mikroskopis Sel Darah

Perubahan morfologi sel darah yang berkaitan dengan leukemia dapat diamati melalui citra mikroskopis apusan darah tepi atau *peripheral blood smear*. Citra mikroskopis sel darah merupakan representasi visual dari sel-sel darah yang diperoleh melalui proses preparasi sampel darah dan pengamatan menggunakan mikroskop, sehingga menghasilkan informasi morfologi yang dapat dianalisis lebih lanjut [20]. Karakteristik seperti bentuk inti sel, ukuran sel, warna sitoplasma, serta keberadaan granula pada jenis sel tertentu menjadi indikator penting dalam membedakan sel darah normal dan abnormal [21].

2.2.1 Blood Smear Image

Apusan darah tepi atau *peripheral blood smear* (PBS) merupakan metode preparasi sampel darah yang dilakukan dengan meneteskan sejumlah kecil darah pada kaca objek, kemudian diratakan menjadi lapisan tipis, dikeringkan, dan diwarnai sebelum diamati di bawah mikroskop [20]. Preparat apusan darah yang dihasilkan memungkinkan visualisasi morfologi masing-masing sel darah secara individual, sehingga pemeriksaan ini menjadi langkah penting dalam evaluasi awal pasien dengan gangguan hematologi [22].

Dalam proses preparasi tersebut, teknik pewarnaan memegang peranan penting dalam menghasilkan kontras visual yang memadai antar jenis sel darah. Pewarnaan yang paling umum digunakan pada apusan darah adalah kelompok pewarnaan Romanowsky, seperti Wright, Giemsa, dan kombinasi Wright-Giemsa, yang secara luas digunakan dalam pemeriksaan hematologi karena kemampuannya dalam menampilkan detail morfologi sel darah secara jelas dan konsisten [23, 24].

Mekanisme pewarnaan Romanowsky didasarkan pada interaksi antara komponen sel dengan dua jenis zat warna utama, yaitu pewarna basa seperti

methylene blue dan turunannya, yaitu *azure*, yang berikatan dengan struktur sel yang bersifat asam, terutama asam nukleat pada inti sel, sehingga menghasilkan warna biru hingga ungu. Sebaliknya, pewarna asam seperti *eosin* berikatan dengan komponen sitoplasma dan granula yang bersifat basa, sehingga menghasilkan warna merah muda hingga jingga. Kombinasi kedua jenis pewarna ini memungkinkan diferensiasi morfologi sel darah secara visual berdasarkan karakteristik struktur internalnya [23, 24].

2.2.2 Karakteristik Morfologi dan Jenis Sel Darah

Pemeriksaan apusan darah tepi atau *peripheral blood smear* merupakan prosedur diagnostik fundamental dalam hematologi klinis. Melalui pemeriksaan ini, morfologi sel darah dapat dievaluasi secara visual berdasarkan karakteristik nukleus, sitoplasma, granula, dan ukuran sel menggunakan pewarnaan khusus hematologi, seperti *Wright-Giemsa* atau *Pappenheim stain* [25]. Pemeriksaan diferensial leukosit atau *leukocyte differential count* menjadi bagian inti dari evaluasi ini, di mana setiap jenis sel diidentifikasi dan dihitung proporsinya untuk mendeteksi kelainan hematologis [21, 26].

Dalam konteks diagnosis leukemia melalui apusan darah tepi, tidak semua jenis sel diamati dengan bobot yang sama. Penilaian morfologi awal memainkan peran penting dalam deteksi dan diagnosis leukemia akut, serta menjadi titik awal yang tidak tergantikan sebelum pemeriksaan lanjutan, seperti imunofenotip dan genetika molekuler dilakukan [27]. Secara khusus, myeloblast menjadi salah satu sel penting untuk diidentifikasi karena peningkatan jumlah sel blast mieloid pada darah tepi atau sumsum tulang dapat menjadi indikasi kuat yang mengarah pada *Acute Myeloid Leukemia* (AML) [28].

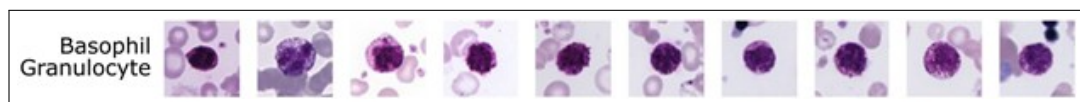
Di sisi lain, sel-sel normal seperti *segmented neutrophil*, *monocyte*, dan *basophil* turut menjadi acuan penting dalam penilaian ini. Proporsi serta morfologi ketiga sel tersebut dapat berubah secara signifikan seiring perkembangan penyakit. Selain itu, penurunan *neutrophil* dan *monocyte* serta perubahan rasio *basophil* merupakan temuan yang sering dijumpai pada pasien leukemia [29]. Sementara itu, *erythroblast* yang dalam keadaan normalnya hanya terdapat di dalam sumsum tulang, apabila ditemukan dalam darah tepi orang dewasa dapat menjadi indikator adanya gangguan hematopoietik serius [30].

Berdasarkan uraian tersebut, *basophil*, *erythroblast*, *monocyte*, *myeloblast*, dan *segmented neutrophil* merupakan beberapa jenis sel yang dapat diamati dalam

analisis morfologi sel darah melalui citra apusan darah tepi. Setiap jenis sel mempunyai ciri morfologi yang berbeda, seperti ukuran sel, bentuk inti, struktur sitoplasma, maupun keberadaan granula. Perbedaan karakteristik tersebut menjadi dasar dalam proses identifikasi jenis sel darah secara visual maupun komputasional [21, 25].

A Basophil

Basophil adalah salah satu tipe leukosit yang memiliki fungsi dalam mekanisme pertahanan tubuh, khususnya terhadap alergen dan proses inflamasi. Sel ini termasuk dalam kelompok granulosit dan memiliki jumlah paling sedikit dalam sirkulasi darah, yaitu kurang dari 1% dari total leukosit. Sitoplasma *basophil* dipenuhi oleh granula dengan kandungan histamin dan heparin yang tinggi, yang berfungsi sebagai mediator utama dalam reaksi hipersensitivitas tipe I. Selain berperan dalam respons alergi, *basophil* juga terlibat dalam regulasi sistem imun, termasuk pada kondisi infeksi, penyakit autoimun, serta gangguan alergi kronis [31]. Contoh morfologi sel *basophil* pada citra mikroskopis apusan darah tepi dapat dilihat pada Gambar 2.1.

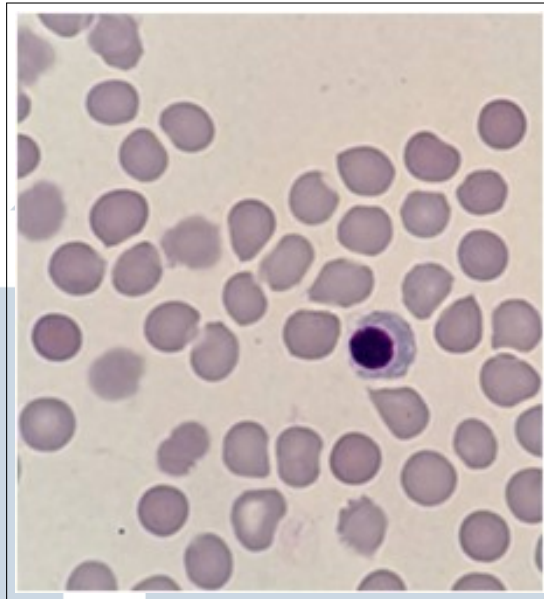


Gambar 2.1. Basophil

Sumber: [25]

B Erythroblast

Erythroblast merupakan sel prekursor eritrosit yang masih memiliki inti dan berperan dalam proses eritropoiesis di sumsum tulang. Sel ini mengalami perkembangan melalui beberapa tahap diferensiasi hingga akhirnya kehilangan inti dan berubah menjadi retikulosit, yang kemudian berkembang menjadi eritrosit matang. Pada keadaan fisiologis, erythroblast secara eksklusif berlokasi di sumsum tulang dan jarang dijumpai dalam darah perifer. Kehadiran *erythroblast* dalam darah tepi dapat menjadi indikator adanya gangguan hematologis, seperti anemia berat, hemolisis, kelainan sumsum tulang, maupun leukemia [32]. Contoh morfologi sel *erythroblast* pada citra mikroskopis apusan darah tepi dapat dilihat pada Gambar 2.2.



Gambar 2.2. Erythroblast

Sumber: [32]

C Monocyte

Monocyte merupakan salah satu jenis sel darah putih dengan ukuran terbesar, yaitu berkisar antara 12–20 μm . Ciri morfologi yang paling khas dari sel ini adalah inti berbentuk ginjal atau *kidney-shaped nucleus*, sehingga relatif mudah dikenali pada pemeriksaan darah perifer. Secara fungsional, *monocyte* berperan sebagai sel fagosit sekaligus penyaji antigen yang mampu mengenali serta mengeliminasi patogen, debris seluler, dan sel-sel yang rusak. Selain itu, *monocyte* dapat berdiferensiasi menjadi makrofag dan sel dendritik yang memiliki peran penting dalam menjaga keseimbangan sistem imun, terutama pada kondisi infeksi dan peradangan [33]. Contoh morfologi sel *monocyte* pada citra mikroskopis apusan darah tepi dapat dilihat pada Gambar 2.3.



Gambar 2.3. Monocyte

Sumber: [25]

D Segmented Neutrophil

Segmented neutrophil merupakan neutrofil yang sudah berada pada fase maturasi sempurna, yang ditandai ditandai oleh inti sel yang terbagi menjadi 2 hingga 5 lobus dan saling terhubung. Hal ini berbeda dengan *band neutrophil* yang masih bersifat imatur dan memiliki inti berbentuk melengkung tanpa segmentasi yang sempurna. Sel ini merupakan jenis leukosit dengan jumlah terbanyak dalam darah, yaitu sekitar 60–70% dari total sel darah putih, serta berperan sebagai lini pertama dalam sistem pertahanan imun bawaan tubuh. Sitoplasma *segmented neutrophil* mengandung granula antimikroba, seperti lisozim, defensin, dan mieloperoxidase atau *myeloperoxidase* (MPO), yang berfungsi dalam menghancurkan patogen melalui mekanisme fagositosis, degranulasi, produksi *reactive oxygen species* (ROS), serta pembentukan *neutrophil extracellular traps* (NETs) [34]. Contoh morfologi sel *segmented neutrophil* pada citra mikroskopis apusan darah tepi dapat dilihat pada Gambar 2.4.

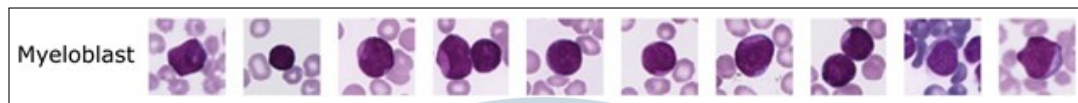


Gambar 2.4. Segmented Neutrophil

Sumber: [25]

E Myeloblast

Myeloblast merupakan sel progenitor mieloid paling awal yang dapat dikenali secara morfologis di sumsum tulang dan menjadi tahap awal diferensiasi menuju sel darah putih matur, seperti neutrofil dan monosit. Pada keadaan fisiologis, *myeloblast* hanya ditemukan dalam jumlah kecil dalam sumsum tulang. Namun, proliferasi *myeloblast* yang tidak terkendali merupakan salah satu karakteristik utama *Acute Myeloid Leukemia* (AML), yaitu keganasan hematologis yang dicirikan oleh penumpukan sel *blast* mieloid imatur di dalam sumsum tulang maupun sirkulasi darah perifer, sehingga mengganggu proses pembentukan normal sel darah merah dan trombosit [35]. Contoh morfologi sel *myeloblast* pada citra mikroskopis apusan darah tepi dapat dilihat pada Gambar 2.5.



Gambar 2.5. Myeloblast

Sumber: [25]

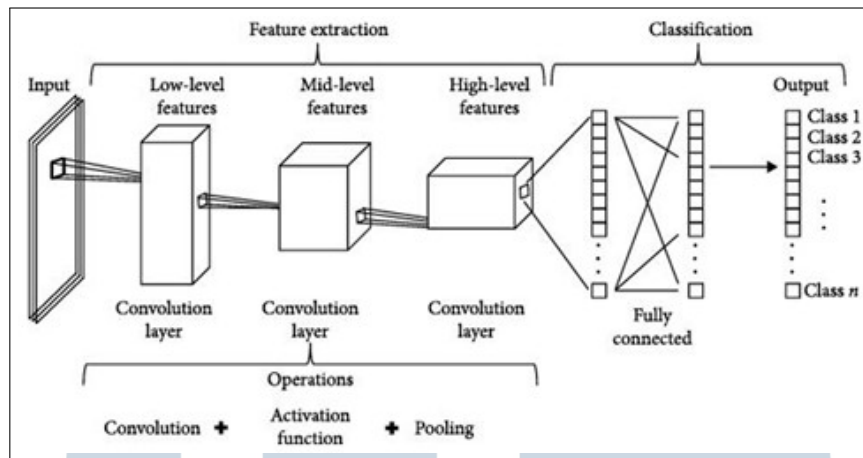
2.3 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu arsitektur deep learning yang dikembangkan guna mengolah data berstruktur *grid*, khususnya citra, melalui proses ekstraksi fitur secara otomatis dan hierarkis menggunakan serangkaian lapisan komputasi. CNN mampu menangkap dependensi spasial dalam citra melalui penerapan *filter* atau *kernel*, sehingga memungkinkan pembelajaran representasi fitur secara adaptif, berawal dari pola dasar seperti tepian dan tekstur hingga fitur-fitur abstrak tingkat tinggi [36, 37].

Secara struktural, CNN tersusun atas sejumlah elemen inti, yaitu *convolutional layer* yang difungsikan sebagai ekstraksi fitur, *activation function* yang memungkinkan model mempelajari pola nonlinier yang kompleks, *pooling layer* yang digunakan guna mereduksi dimensionalitas spasial, serta *fully connected layer* yang berperan dalam proses klasifikasi pada tahap akhir [36, 37].

Dibandingkan dengan metode konvensional, CNN memiliki keunggulan pada mekanisme *parameter sharing* pada lapisan konvolusi, yang mampu mengurangi jumlah parameter yang harus dilatih tanpa mengorbankan kemampuan representasi fitur. Hal ini menjadikan CNN sebagai salah satu pendekatan utama dalam berbagai tugas *computer vision* modern [36, 37].

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 2.6. Arsitektur Convolutional Neural Network (CNN)

Sumber: [38]

Berdasarkan Gambar 2.6, arsitektur CNN tersusun atas dua fase utama, yakni *feature extraction* dan *classification*. Pada fase *feature extraction*, citra masukan diolah melalui sejumlah *convolutional layer* guna menghasilkan representasi fitur secara bertingkat, dimulai dari fitur berkarakter sederhana hingga fitur yang semakin kompleks, melalui operasi *convolution*, *activation function*, dan *pooling*. Selanjutnya, fitur yang diperoleh digunakan pada tahap *classification* melalui *fully connected layer* guna menghasilkan keluaran berupa kelas tertentu. Proses ini memungkinkan CNN melakukan klasifikasi citra secara otomatis berdasarkan pola fitur yang telah dipelajari selama proses pelatihan model.

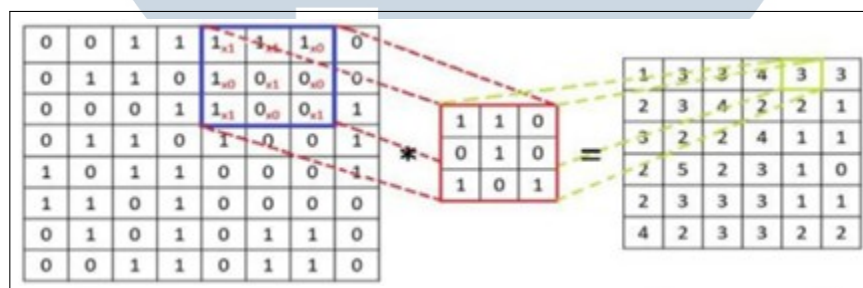
2.3.1 Convolution Layer

Convolution layer adalah elemen inti dalam arsitektur CNN yang berfungsi dalam mengekstraksi fitur signifikan dari data masukan, khususnya citra. Proses pada *layer* tersebut dijalankan dengan menerapkan *filter* atau *kernel* yang digeser berdasarkan nilai *stride* pada setiap bagian *input* untuk menghasilkan *feature map*. Setiap *kernel* melakukan operasi perkalian elemen dan penjumlahan atau *dot product* pada area lokal citra, sehingga mampu menangkap sejumlah pola spesifik, di antaranya tepian, tekstur, maupun bentuk objek [39, 37]. Output dari proses tersebut berupa sekumpulan *feature map* yang merekam karakteristik esensial dari input untuk diproses pada tahap selanjutnya.

Secara teknis, *kernel* pada *convolution layer* merupakan matriks berukuran lebih kecil dibandingkan *input* yang bergerak berdasarkan parameter tertentu,

seperti *stride* dan *padding*. Nilai *stride* memengaruhi pergeseran *kernel*, di mana penerapan *stride* dengan nilai yang lebih besar akan berdampak pada berkurangnya dimensi *output* yang dihasilkan [40, 39, 37]. Sementara itu, *padding* digunakan untuk mengontrol ukuran *output* serta mempertahankan informasi pada bagian tepi citra. Selain itu, penggunaan banyak *kernel* dalam satu *layer* memungkinkan jaringan mengekstraksi berbagai jenis fitur secara simultan. Proses ini juga menerapkan konsep *weight sharing* dan *local connectivity*, yang membuat CNN lebih efisien dalam jumlah parameter serta meningkatkan efektivitas proses pembelajaran [40].

Dengan demikian, *convolution layer* tidak hanya berfungsi sebagai tahap awal dalam ekstraksi fitur, tetapi juga menjadi fondasi utama dalam pembentukan representasi data pada CNN. Kemampuannya dalam menangkap informasi spasial secara efektif menjadikan *layer* ini memiliki peran krusial dalam beragam penerapan pengolahan citra, mencakup klasifikasi maupun deteksi objek [39, 37].



Gambar 2.7. Convolution Layer

Sumber: [41]

Berdasarkan Gambar 2.7, proses *convolution* dijalankan melalui mekanisme menggeser sebuah *kernel* pada setiap posisi *input* citra. Pada tiap pergeseran, nilai pada kernel akan dioperasikan secara perkalian terhadap nilai piksel yang saling berkorespondensi, lalu hasilnya dijumlahkan untuk menghasilkan satu nilai pada *feature map*. Proses ini dilakukan secara berulang hingga seluruh area citra diproses.

Ilustrasi tersebut menunjukkan bagaimana sebuah *kernel* berukuran kecil mampu mengekstraksi informasi lokal dari citra, seperti pola tepi atau tekstur tertentu. Hasil akhir berupa *feature map* merepresentasikan respons dari *kernel* terhadap pola yang terdeteksi pada *input*. Semakin banyak *kernel* yang digunakan, maka semakin beragam fitur yang dapat dipelajari oleh jaringan.

Dengan demikian, proses *convolution* memberi kemampuan kepada model

dalam mengidentifikasi sejumlah pola signifikan yang terkandung dalam citra secara bertahap, untuk kemudian digunakan dalam tahap pemrosesan berikutnya pada CNN.

2.3.2 Pooling Layer

Pooling layer adalah salah satu elemen krusial dalam arsitektur *Convolutional Neural Network* (CNN) yang bertugas melakukan pengurangan dimensi *feature map* dengan tetap mempertahankan informasi utama. Proses ini dilakukan melalui operasi agregasi pada wilayah lokal, sehingga volume data menjadi lebih ringkas dan komputasi menjadi jauh lebih efisien. Selain itu, *pooling layer* juga membantu menghasilkan representasi fitur yang lebih *robust* terhadap variasi kecil pada citra, seperti pergeseran atau perubahan posisi objek [42, 43].

A Average Pooling

Average pooling merupakan teknik *pooling* yang beroperasi dengan mengkalkulasi nilai rata-rata dari setiap wilayah lokal yang terdapat pada *feature map*. Setiap area kecil akan direpresentasikan oleh satu nilai yang diperoleh dari hasil penghitungan rata-rata seluruh elemen yang ada di dalamnya, sehingga informasi yang diperoleh bersifat lebih menyeluruh [42, 43].

B Max Pooling

Max pooling merupakan metode *pooling* yang bekerja dengan menyeleksi nilai tertinggi dari setiap region lokal yang terdapat pada *feature map*. Pendekatan tersebut menjaga fitur yang paling menonjol dalam suatu area, sehingga efektif dalam menonjolkan karakteristik penting dari citra [42, 43].

C Global Average Pooling (GAP)

Global Average Pooling (GAP) adalah evolusi dari *average pooling* yang mengkalkulasi rata terhadap seluruh nilai pada masing-masing *feature map*. Dengan demikian, setiap *feature map* dikompresi hingga satu nilai tunggal yang merangkum keseluruhan informasi fitur secara global [42].

D Global Max Pooling (GMP)

Global Max Pooling (GMP) merupakan variasi dari *max pooling* yang menyeleksi nilai tertinggi dari keseluruhan elemen yang terdapat dalam setiap *feature map*. Berbeda dengan *max pooling* konvensional yang beroperasi pada area lokal, metode ini melakukan agregasi secara global, sehingga setiap *feature map* diringkas menjadi satu nilai yang paling menonjol [44].

2.3.3 Activation Function

Activation function adalah elemen nonlinier dalam jaringan saraf tiruan yang berfungsi mengkonversi *input* menjadi *output* pada setiap neuron, sehingga jaringan mampu mengenali pola-pola kompleks yang terkandung dalam data. Tanpa adanya *activation function*, jaringan hanya akan menghasilkan transformasi linear yang terbatas dan tidak mampu merepresentasikan kompleksitas data nyata, seperti citra medis [45].

Fungsi ini memiliki peran penting dalam proses pelatihan jaringan karena memengaruhi aliran gradien selama proses *backpropagation*, sehingga model dapat belajar secara efektif [46]. Di antara berbagai fungsi aktivasi yang tersedia, salah satu yang paling jamak digunakan adalah ReLU atau *Rectified Linear Unit*, yakni fungsi yang beroperasi dengan mempertahankan nilai positif sembari mengonversi nilai negatif menjadi nol. Karakteristik ini membuat ReLU memiliki komputasi yang sederhana serta mampu meningkatkan efisiensi pelatihan, terutama pada jaringan yang dalam [46, 45].

2.3.4 Fully Connected Layer

Fully Connected Layer (FCL) adalah salah satu lapisan dalam arsitektur *Convolutional Neural Network* (CNN) yang pada umumnya diposisikan di bagian terminal jaringan. Layer ini difungsikan untuk menghubungkan representasi fitur yang telah dihasilkan pada tahap sebelumnya ke dalam output berupa hasil klasifikasi atau prediksi. Dalam layer ini, setiap neuron terkoneksi dengan semua unit pada lapisan yang mendahuluinya atau *fully connected*, sehingga memfasilitasi integrasi informasi secara menyeluruh [36].

Sebelum masuk ke *Fully Connected Layer*, *feature map* yang dihasilkan dari proses konvolusi dan *pooling* terlebih dahulu dikonversi menjadi sebuah vektor berdimensi tunggal melalui proses *flattening*. Vektor tersebut kemudian

diproses menggunakan operasi matematis yang melibatkan parameter bobot dan bias, serta fungsi aktivasi untuk menghasilkan keluaran akhir [47]. Pada kasus klasifikasi, fungsi aktivasi *Softmax* lazim dimanfaatkan guna mentransformasikan output menjadi nilai probabilitas pada setiap kelas. Dengan demikian, lapisan ini berperan sebagai tahap akhir dalam pengambilan keputusan yang didasarkan pada representasi fitur yang telah diserap oleh jaringan.

2.3.5 Dropout Layer

Dropout layer adalah salah satu teknik regularisasi yang diterapkan dalam arsitektur *Convolutional Neural Network* (CNN) guna meminimalkan terjadinya *overfitting* pada model. Metode tersebut beroperasi dengan menonaktifkan sejumlah neuron secara acak dalam proses pelatihan, sehingga jaringan tidak memiliki ketergantungan terhadap neuron-neuron tertentu dan mampu mengembangkan representasi fitur yang lebih umum [36].

2.4 Transfer Learning

Transfer learning merupakan salah satu pendekatan dalam *deep learning* yang memanfaatkan pengetahuan yang telah dipelajari oleh suatu model pada tugas sebelumnya untuk membantu menyelesaikan tugas baru yang memiliki karakteristik serupa. Konsep ini didasarkan pada pemanfaatan representasi fitur yang telah dipelajari dari dataset berskala besar sehingga model tidak perlu mempelajari seluruh karakteristik data dari awal. Dengan demikian, proses pelatihan dapat berlangsung lebih efisien baik dari sisi waktu maupun kebutuhan data dibandingkan dengan pelatihan model dari awal (*training from scratch*) [48].

Pada bidang klasifikasi citra medis, *transfer learning* banyak diterapkan karena dataset yang tersedia umumnya memiliki jumlah data berlabel yang terbatas. Model *pre-trained* telah mempelajari berbagai karakteristik visual dasar, seperti tepi (*edges*), tekstur, bentuk, dan pola objek dari dataset berskala besar, sehingga representasi fitur tersebut masih dapat dimanfaatkan pada berbagai tugas klasifikasi citra medis. Pemanfaatan pengetahuan yang telah dipelajari sebelumnya memungkinkan model mencapai performa yang baik dengan waktu pelatihan yang lebih singkat serta kebutuhan sumber daya komputasi yang lebih rendah dibandingkan melatih model dari awal [48].

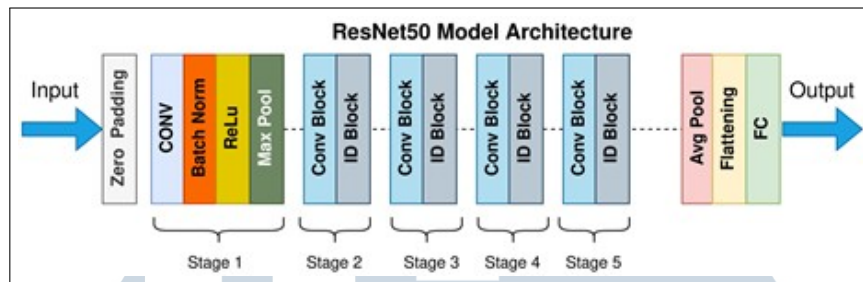
Secara umum, implementasi *transfer learning* dilakukan melalui dua

pendekatan, yaitu *feature extraction* dan *fine-tuning*. Pada pendekatan *feature extraction*, parameter pada model *pre-trained* dipertahankan sehingga model hanya berperan sebagai pengekstraksi fitur, sedangkan proses pelatihan dilakukan pada lapisan klasifikasi yang ditambahkan sesuai dengan kebutuhan tugas baru. Adapun pada pendekatan *fine-tuning*, sebagian atau seluruh parameter model *pre-trained* diperbarui selama proses pelatihan sehingga representasi fitur dapat menyesuaikan karakteristik dataset yang digunakan. Pemilihan pendekatan tersebut bergantung pada jumlah data, tingkat kemiripan antara dataset sumber dan dataset target, serta sumber daya komputasi yang tersedia [48].

2.5 ResNet-50

ResNet-50 adalah salah satu arsitektur *Convolutional Neural Network* dengan kedalaman 50 lapisan yang dikembangkan guna menanggulangi permasalahan *vanishing gradient* pada jaringan yang sangat dalam. Permasalahan ini dapat menyebabkan proses pelatihan menjadi kurang optimal karena nilai gradien cenderung mendekati nol pada lapisan awal. Untuk mengatasi hal tersebut, ResNet-50 memperkenalkan konsep *residual learning* melalui penggunaan *skip connection* maupun *shortcut connection*, yakni suatu mekanisme yang memberi kemampuan bagi input pada suatu lapisan untuk dialirkan secara langsung ke lapisan selanjutnya. Pendekatan tersebut berkontribusi stabil dalam menjaga aliran informasi [49].

Struktur utama ResNet-50 tersusun atas sejumlah elemen penting, meliputi *convolution layer*, *batch normalization*, fungsi aktivasi ReLU, *residual block*, serta *fully connected layer*. Perlu dicatat bahwa *batch normalization* berperan dalam menstabilkan distribusi aktivasi antarlapis, sehingga berkontribusi pada percepatan dan kestabilan proses pelatihan model. Selain itu, arsitektur ini menerapkan desain *bottleneck* yang terdiri dari rangkaian konvolusi berukuran 1×1 , 3×3 , dan 1×1 . Desain ini bertujuan untuk mengoptimalkan efisiensi komputasi tanpa mengorbankan kapabilitas model dalam mengekstraksi fitur secara mendalam. Melalui mekanisme tersebut, ResNet-50 mampu melakukan *hierarchical feature learning*, yaitu mengabstraksi fitur secara bertahap, berawal dari pola dasar hingga representasi yang kian kompleks [49].



Gambar 2.8. Arsitektur ResNet-50

Sumber: [50]

Berdasarkan Gambar 2.8, arsitektur ResNet-50 diawali dengan tahap *initial convolution* yang terdiri dari proses *zero padding*, *convolution*, *batch normalization*, dan aktivasi ReLU, kemudian diikuti oleh *max pooling*. Selanjutnya, jaringan dibagi ke dalam beberapa *stage* yang masing-masing terdiri dari susunan *convolution block* dan *identity block*. *Convolution block* digunakan untuk menyesuaikan dimensi fitur, sedangkan *identity block* mempertahankan dimensi tersebut dengan memanfaatkan *shortcut connection*. Mekanisme ini memungkinkan informasi yang berasal dari lapisan-lapisan sebelumnya tetap tersalurkan serta mengurangi degradasi performa pada jaringan yang dalam.

Pada bagian akhir arsitektur, fitur yang telah diekstraksi akan diproses melalui *average pooling* untuk mereduksi dimensi, kemudian dilanjutkan dengan proses *flattening* dan *fully connected layer* untuk menghasilkan *output*. Secara keseluruhan, arsitektur ini memungkinkan proses ekstraksi fitur yang efektif sekaligus mempertahankan stabilitas selama proses pelatihan, sehingga secara luas dimanfaatkan dalam beragam tugas klasifikasi citra.

2.6 K-Fold Cross Validation

K-Fold Cross Validation merupakan salah satu teknik evaluasi model yang membagi dataset menjadi beberapa bagian (*fold*) dengan ukuran yang relatif sama untuk memperoleh estimasi performa model yang lebih representatif. Berbeda dengan metode *holdout* yang hanya melakukan satu kali pembagian data menjadi data latih dan data uji, *K-Fold Cross Validation* melakukan proses pelatihan dan evaluasi secara berulang sehingga setiap bagian dataset memperoleh kesempatan menjadi data evaluasi. Pendekatan ini banyak digunakan dalam pengembangan model *machine learning* maupun *deep learning* karena mampu memberikan estimasi kemampuan generalisasi model yang lebih stabil terhadap data yang belum

pernah dilihat sebelumnya [51, 52].

Pada metode *K-Fold Cross Validation*, dataset terlebih dahulu dibagi menjadi sebanyak k bagian (*fold*) yang saling terpisah. Selanjutnya, pada setiap iterasi satu *fold* digunakan sebagai data validasi, sedangkan $k - 1$ *fold* lainnya digunakan sebagai data pelatihan. Proses tersebut diulangi hingga seluruh *fold* memperoleh giliran sebagai data validasi sebanyak satu kali. Setiap iterasi menghasilkan sebuah model dengan nilai performa yang berbeda sesuai dengan pembagian data yang digunakan. Setelah seluruh proses selesai, performa model diperoleh dari nilai rata-rata hasil evaluasi pada seluruh *fold*, sehingga mampu memberikan gambaran yang lebih menyeluruh mengenai kemampuan model dibandingkan hanya menggunakan satu kali pembagian dataset [51].

Pemilihan jumlah *fold* memengaruhi keseimbangan antara kualitas evaluasi dan kebutuhan komputasi. Secara umum, nilai $k=5$ atau $k=10$ merupakan konfigurasi yang paling banyak digunakan karena mampu menghasilkan estimasi performa yang baik dengan biaya komputasi yang masih dapat diterima. Semakin besar jumlah *fold*, semakin banyak pula proses pelatihan yang harus dilakukan sehingga waktu komputasi akan meningkat. Oleh karena itu, pemilihan jumlah *fold* perlu mempertimbangkan ukuran dataset, kompleksitas model, serta sumber daya komputasi yang tersedia [51].

Dibandingkan metode *holdout*, *K-Fold Cross Validation* memiliki beberapa keunggulan, yaitu mampu mengurangi pengaruh pemilihan data secara acak (*sampling bias*), memanfaatkan seluruh data sebagai data pelatihan maupun data evaluasi secara bergantian, serta menghasilkan estimasi performa yang lebih konsisten. Pendekatan ini sangat bermanfaat ketika jumlah data relatif terbatas karena setiap sampel tetap berkontribusi dalam proses evaluasi model. Namun demikian, metode ini memerlukan waktu pelatihan yang lebih lama dibandingkan metode *holdout* karena proses pelatihan harus dilakukan sebanyak jumlah *fold* yang digunakan [51].

2.7 Analisis Gap Penelitian

Berdasarkan hasil telaah terhadap beberapa penelitian terdahulu, diketahui bahwa berbagai pendekatan *deep learning* telah berhasil diterapkan untuk klasifikasi citra sel darah pada kasus Acute Myeloid Leukemia (AML). Penelitian-penelitian tersebut menggunakan arsitektur yang berbeda-beda dan menunjukkan performa yang baik sesuai dengan karakteristik dataset yang digunakan. Selain

itu, ResNet-50 juga telah banyak dimanfaatkan pada berbagai penelitian klasifikasi citra medis dan menunjukkan performa yang kompetitif dalam mengekstraksi fitur citra. Oleh karena itu, analisis gap dilakukan untuk menunjukkan posisi penelitian ini dibandingkan penelitian sebelumnya sekaligus mengidentifikasi peluang pengembangan yang masih dapat dieksplorasi.

Tabel 2.1. Perbandingan Penelitian Terdahulu

Peneliti	Objek Penelitian	Metode	Hasil
Shaheen <i>et al.</i> (2021)	Klasifikasi citra sel darah AML	AlexNet	Akurasi 98,58%.
Elhassan <i>et al.</i> (2023)	Klasifikasi <i>atypical white blood cells</i> pada AML	Two-stage DCAE-CNN	Akurasi 97%, Presisi 98%, Sensitivitas 97%, AUC 99,7%.
Cambay <i>et al.</i> (2024)	Klasifikasi citra gastrointestinal	ResNet-50	Baseline 94,3%; Modifikasi 94,78%.
Grace dan Gunawan (2025)	Klasifikasi kanker payudara	CNN, ResNet-50, Vision Transformer	ResNet-50 memperoleh akurasi tertinggi 97%.

Berdasarkan Tabel 2.1, dapat diketahui bahwa penelitian mengenai klasifikasi citra sel darah pada kasus AML telah dilakukan menggunakan berbagai arsitektur *deep learning*, seperti AlexNet maupun model Two-stage DCAE-CNN [11, 12]. Sementara itu, penelitian pada domain citra medis lainnya menunjukkan bahwa ResNet-50 mampu memberikan performa yang kompetitif dalam proses klasifikasi [15, 14]. Meskipun demikian, berdasarkan telaah yang dilakukan, masih terdapat peluang untuk mengeksplorasi penerapan ResNet-50 berbasis transfer learning pada klasifikasi citra sel darah AML. ResNet-50 dipilih dalam penelitian ini karena menerapkan konsep *residual learning* melalui mekanisme *skip connection*, sehingga mampu mempertahankan kualitas pembelajaran pada jaringan yang lebih dalam serta membantu mengatasi permasalahan *vanishing gradient*. Selain itu, penggunaan model *pre-trained* pada ResNet-50 memungkinkan proses ekstraksi fitur dilakukan secara lebih efektif melalui pemanfaatan pengetahuan yang telah dipelajari dari dataset *ImageNet*. Oleh karena itu, penelitian ini mengimplementasikan ResNet-50 berbasis *transfer learning* untuk mengklasifikasikan lima jenis sel darah berdasarkan karakteristik morfologinya [49].

2.8 Evaluasi Model

Evaluasi model merupakan tahap yang bertujuan untuk mengukur performa model setelah proses pelatihan dan pengujian berlangsung. Tahap ini dimaksudkan untuk mengetahui kemampuan model dalam melakukan klasifikasi berdasarkan hasil prediksi yang dihasilkan. Pengukuran performa dilakukan dengan memanfaatkan sejumlah metrik, di antaranya *confusion matrix*, *accuracy*, *precision*, *recall*, dan *F1-score*, agar performa model dapat dianalisis secara komprehensif [49].

A Confusion Matrix

Confusion matrix adalah tabel pengukuran yang difungsikan guna mengevaluasi kesesuaian antara hasil prediksi model dengan nilai aktualnya. Matriks ini tersusun dari empat elemen pokok, yakni True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN). Melalui matriks ini, dapat diperoleh informasi mengenai banyaknya prediksi yang tepat maupun keliru pada masing-masing kelas [49].

B Accuracy

Accuracy adalah metrik yang dimanfaatkan untuk mengukur tingkat kebenaran model secara menyeluruh, yaitu rasio antara jumlah prediksi yang tepat terhadap keseluruhan data yang diuji. Nilai *accuracy* dapat dirumuskan sebagai berikut:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.1)$$

Semakin tinggi nilai *accuracy* yang diperoleh, maka kian optimal pula kemampuan model dalam melakukan klasifikasi [49].

C Precision

Precision dimanfaatkan untuk mengkuantifikasi tingkat keakuratan model dalam memprediksi kelas positif. Metrik ini mencerminkan rasio antara jumlah prediksi positif yang tepat terhadap keseluruhan prediksi positif yang dikeluarkan oleh model. Rumus *precision* adalah sebagai berikut:

$$Precision = \frac{TP}{TP + FP} \quad (2.2)$$

Nilai *precision* yang tinggi mengindikasikan bahwa model memiliki tingkat kesalahan prediksi positif yang minimal [49].

D Recall

Recall merupakan metrik digunakan untuk mengkuantifikasi kapabilitas model dalam mengenali keseluruhan data positif yang sesungguhnya. mencerminkan rasio antara jumlah prediksi positif yang tepat terhadap keseluruhan data positif yang tersedia. Rumus *recall* adalah sebagai berikut:

$$Recall = \frac{TP}{TP + FN} \quad (2.3)$$

Semakin tinggi nilai *recall*, maka kian optimal kapabilitas model dalam mengidentifikasi data positif [49].

E F1-Score

F1-score merupakan metrik yang mengintegrasikan *precision* dan *recall* guna mengukur keselarasan performa model. Metrik ini sangat relevan digunakan ketika terjadi ketimpangan distribusi data pada setiap kelas. Rumus *F1-score* adalah sebagai berikut:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (2.4)$$

Nilai *F1-score* yang tinggi mengindikasikan bahwa model mencapai keselarasan yang optimal antara keakuratan prediksi dan kapabilitas deteksi [49].