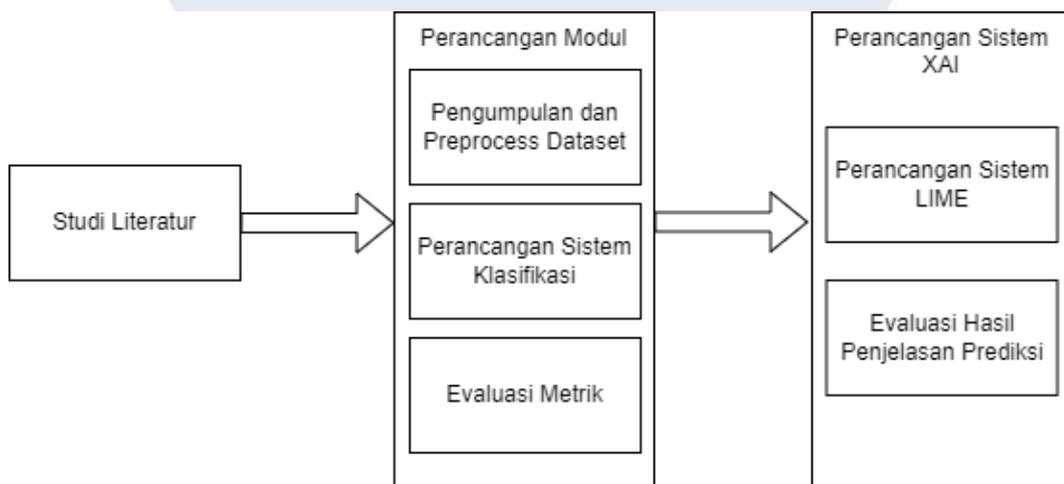


BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Metode Penelitian

Pada penelitian ini, penulis melakukan beberapa tahapan yang penulis lakukan untuk meneliti dan merancang sistem *deep learning* untuk mencapai tujuan dan menjawab masalah yang sudah penulis sebutkan sebelumnya. Metode penelitian tersebut dimulai dari studi literatur, dan dilanjutkan dengan perancangan modul yang terdiri dari pengumpulan dan *preprocess* dataset, perancangan sistem klasifikasi, dan evaluasi metrik. Lalu yang terakhir akan dilanjutkan dengan perancangan sistem XAI dan evaluasi hasil penjelasannya. Alur penelitian tersebut dapat dilihat pada gambar 3.1.



Gambar 3.0.1 Alur penelitian

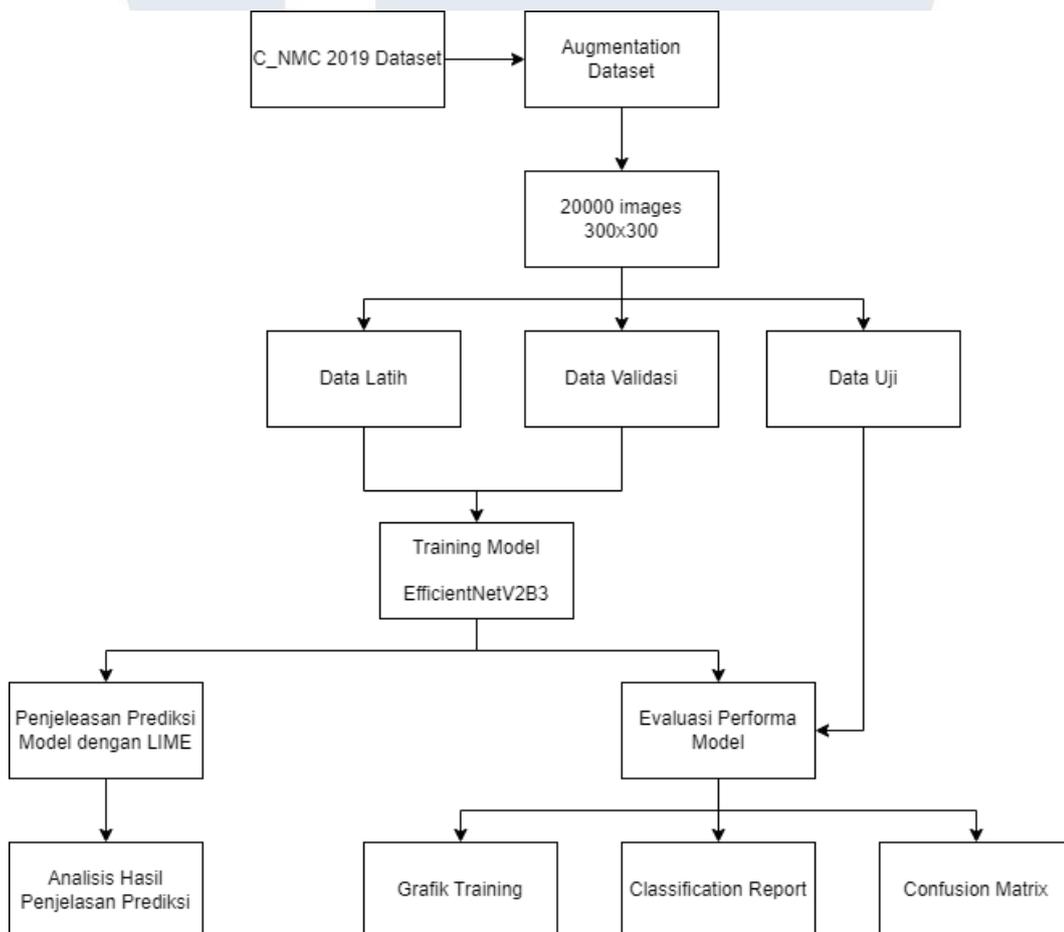
3.2 Studi Literatur

Dalam melakukan studi literatur, penulis sudah membaca dan mempelajari terlebih dahulu tentang *deep learning*, CNN, klasifikasi leukemia terutama ALL, dan *explainable AI*. Dengan mempelajari berbagai penelitian terdahulu, penulis mendapatkan inspirasi untuk metode dan model yang akan digunakan untuk klasifikasi sel leukemia ALL. Penulis juga melakukan diskusi dengan dosen pembimbing yang merupakan salah satu dosen dari program studi Teknik Komputer tentang penelitian yang dilakukan

oleh penulis. Penulis juga membaca berbagai website tentang penggunaan *deep learning* dan beberapa modelnya untuk diaplikasikan ke dataset yang digunakan oleh penulis, dan juga tentang *explainable AI* dan penggunaannya.

3.3 Perancangan Model

Perancangan model memiliki proses dimulai dari pemerolehan dataset, pengolahan dataset, pembuatan sistem klasifikasi, evaluasi metrik dari model tersebut. Model yang dibuat juga akan dihubungkan kepada sistem LIME untuk menjelaskan hasil prediksi dari model tersebut yang dimana setelah itu hasilnya akan dianalisis. Alur perancangan model tersebut dapat dilihat pada gambar 3.2.



Gambar 3.0.2 Alur perancangan model

3.3.1 Pengumpulan dan pengolahan dataset

Pengumpulan data pada penelitian ini menggunakan dataset terbuka yang didapat dari *website* Kaggle dengan judul *Leukemia Classification* dan memiliki dataset dengan nama C-NMC 2019 yang dibuat oleh SBILab. Dataset ini diklaim merupakan representasi dari gambar mikroskopik yang didapat dikarenakan memiliki *staining noise* dan *illumination error*. Dataset ini memiliki total gambar sebanyak 15135 gambar yang didapat dari 118 pasien. Dataset tersebut berisi gambar-gambar mikroskopik dari sel darah setiap pasien yang sudah dilakukan *preprocessing* dan segmentasi sebelumnya dimana setiap gambar hanya berisi 1 sel saja dan setiap gambar sel akan memiliki *background* gelap untuk menambah kontras dari sel tersebut dan *background* gambar. Semua gambar tersebut juga sudah dibagi menjadi 3 *folder* berisikan data *train*, *validation*, dan *test*. Setiap data tersebut juga sudah diklasifikasikan oleh ahli enkologi menjadi 2 kelas, yaitu ALL atau sel leukemia, dan HEM, sel yang sehat dan tidak terkena kanker. Penulis hanya mengambil gambar dari *folder train* dan *validation* saja dikarenakan kedua data tersebut memiliki *truth label* sedangkan data *test* tidak memiliki sehingga tidak bisa digunakan pada penelitian ini. Data *train* memiliki jumlah sebanyak 10661 gambar yang terdiri dari 7272 gambar kelas ALL dan 3389 kelas HEM. Sedangkan untuk data validasi, terdiri dari 1867 gambar yang terdiri dari 1219 gambar kelas ALL dan 648 kelas HEM. Tabel isi dari dataset tersebut dapat dilihat pada tabel 3.1.

Tabel 2 Tabel isi dari dataset

	ALL	HEM
Train	7272	3389
Validasi	1219	648
Test	2586	

Dikarenakan data latih yang tidak seimbang antara 2 kelas tersebut, maka penulis melakukan *oversampling* dengan cara augmentasi menggunakan ImageDataGenerator sehingga kedua kelas memiliki jumlah gambar sebanyak 10000 gambar. Tipe augmentasi yang digunakan adalah rotasi dengan *range* 20

derajat, *vertical flip* dan *horizontal flip* yang berdasarkan penelitian sebelumnya[4][5], dikarenakan augment-augment itu hanya memutar dan membalikan gambar sehingga kecil kemungkinan ada gambar sel yang kepotong karena pergeseran gambar. Tahapan lainnya adalah mengubah ukuran gambar menjadi 300x300, dikarenakan model EfficientNetV2B3 memiliki input shape yang diharapkan yaitu 300x300[20].

3.3.2 Perancangan Sistem Klasifikasi

Perancangan sistem klasifikasi pada penelitian ini dilakukan menggunakan model yang didasari dari penelitian sebelumnya. Model yang akan dipakai yaitu EfficientNetV2B3. Model ini dipilih karena model ini adalah versi yang lebih baru dari keluarga. Model tersebut juga diambil berdasarkan penelitian yang dilakukan oleh Amreen Batool dan Yung-Cheol Byun, dimana penelitian tersebut memakai model EfficientNetB3 dan mendapatkan hasil akurasi dan F1-score yang tinggi[4].

Untuk penelitian ini, layer yang akan dipakai terinspirasi dari penelitian sebelumnya[4], dengan penambahan batch normalization agar model bisa lebih stabil dan meningkatkan akurasi dan parameter lainnya seperti dropout rate penulis ubah menjadi nilai yang paling tinggi yang bisa didapat oleh model tersebut. Susunan layer tersebut dapat dilihat pada gambar 3.3 dengan penjelasan sebagai berikut:

- Batch normalization adalah metode yang digunakan untuk melakukan pelatihan dengan lebih cepat dan lebih stabil melalui normalisasi setiap input layer dengan cara dilakukan re-center atau re-scaling.
- Flatten layer adalah layer yang membuat input hasil ekstraksi fitur yang sebelumnya memiliki layer yang banyak, menjadi satu layer saja.
- Dense layer adalah layer yang akan menjadi inti dari proses klasifikasi. Layer ini akan memiliki 1024 neuron dimana neuron-neuron tersebut menggunakan fungsi relu untuk menyelesaikan masalah non-linear pada hasil dari setiap neuron

- Dropout layer digunakan untuk mencegah model menjadi overfitting dengan cara mengurangi koneksi dari neuron ke neuron selanjutnya. 0.2 adalah rasio pengurangan neuron tersebut dan neuron yang dihilangkan akan dipilih secara acak.
- Dense layer terakhir memiliki fungsi mirip seperti dense layer sebelumnya dan akan mengeluarkan output hasil menjadi klasifikasi sesuai jumlah kelas dataset yang dipakai.

```
demo_resnet_model.add(BatchNormalization())
demo_resnet_model.add(Flatten())
demo_resnet_model.add(Dense(1024, activation='relu'))
demo_resnet_model.add(Dropout(0.2))
demo_resnet_model.add(Dense(2, activation='softmax'))
```

Gambar 3.0.3 Susunan Fully Connecting Layer

Untuk hyper parameter yang digunakan pada penelitian ini, penulis menggunakan optimizer adam, dengan learning rate 0,0001, batch size 32 dan epoch 60. Optimizer adam dipakai dikarenakan adam merupakan campuran antara RMSProp dan AdamGrad dan memiliki performa yang bagus dari penelitian terdahulu[4]. Batch size 32 digunakan karena untuk angka yang lebih besar, maka membutuhkan resource yang lebih besar, terutama dibagian memory dan lebih dari 32 menyebabkan device yang dipakai oleh penulis kehabisan memory. Epoch 60 digunakan, karena dari pengalaman penulis dan penelitian sebelumnya[4], jumlah epoch yang kecil masih rentan terhadap hasil model yang tidak stabil dan dengan epoch yang besar maka memungkinkan model memiliki waktu yang cukup untuk mempelajari fitur-fitur dari data yang dipakai.

3.3.3 Evaluasi Metrik pada Model

Evaluasi metrik pada model sangat penting untuk memeriksa hasil performa dari model tersebut, mengukur kinerjanya, mengoptimalkan model dan

bisa menjadi bahan analisis penulis dalam mengukur performa model. Evaluasi metrik yang penulis gunakan dalam penelitian ini seperti berikut :

- Grafik pembelajaran atau performa untuk melihat kinerja model tersebut dari sisi pelatihan dan validasi. Grafik tersebut memiliki sumbu x sebagai jumlah epoch dan sumbu y sebagai nilai dari akurasi dan loss dari hasil pelatihan dan loss. Dari grafik tersebut bisa kestabilan model tersebut.
- *Classification report* adalah laporan yang akan menunjukkan nilai akurasi, presisi, recall, F-1 score dan support dari hasil pengujian. Dimana akurasi adalah keakuratan model untuk memprediksi secara benar, presisi adalah akurasi antara data yang diminta dengan hasil prediksi yang diberikan model, recall menunjukkan keberhasilan model dalam menemukan kembali informasi dan F1 score adalah hasil rata-rata gabungan dari presisi dan recall. Rumus perhitungan untuk mendapat nilai akurasi, presisi, recall, F1 score adalah sebagai berikut :

$$\text{Akurasi} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{FN} + \text{TN})$$

$$\text{Presisi} = (\text{TP}) / (\text{TP} + \text{FP})$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

$$\text{F-1 Score} = (2 * \text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

- *Confusion matrix* adalah metrik yang dimana akan menunjukkan tabel dengan isi seberapa banyak gambar yang diprediksi oleh model tersebut terhadap 4 kategori, yaitu *True Positive*, *True Negative*, *False Positive*, dan *False Negative*. Hasil dari *confusion matrix* tersebut akan berbentuk *heatmap* dengan warna biru gelap jika makin banyak gambar yang masuk ke salah satu kategori tersebut.

UNIVERSITAS
MULTIMEDIA
NUSANTARA