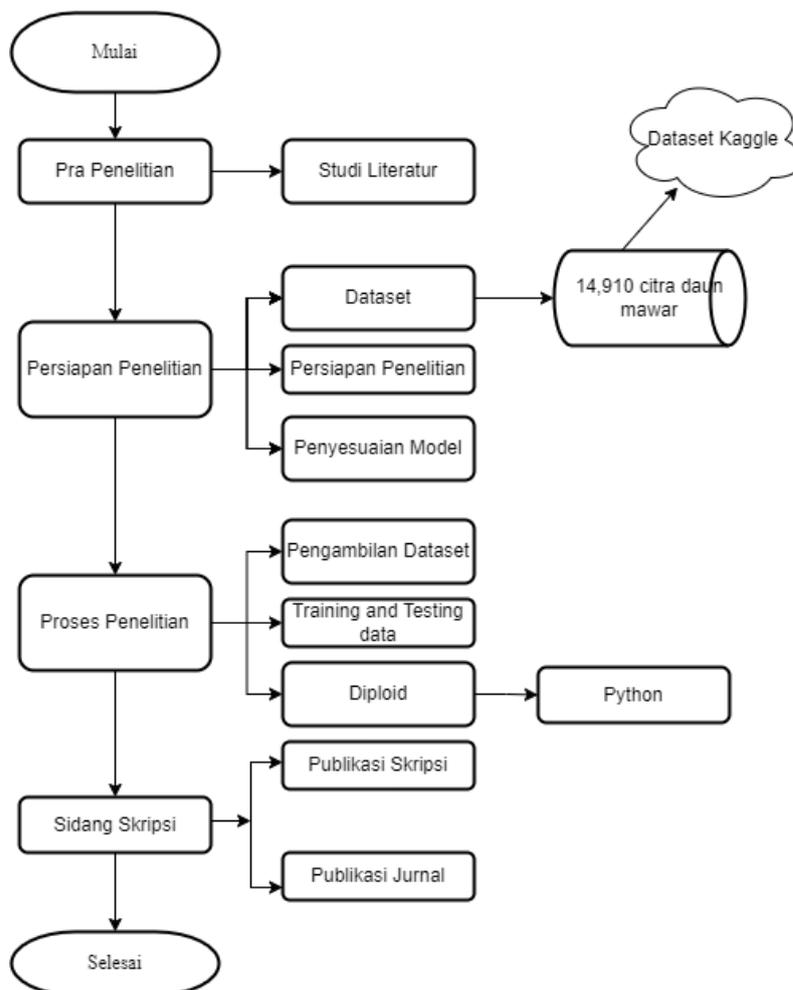


BAB III METODOLOGI PENELITIAN

Dalam penelitian ini, arsitektur squeeze net digunakan untuk mengidentifikasi penyakit pada daun mawar. Penyakit tersebut diklasifikasikan berdasarkan nama penyakit *Rose_sawfly_Rose_slug*, *Healthy_Leaf_Rose*, dan *Rose_Rust*.

3.1 Alur Penelitian

Konsep atau gambaran dari penelitian yang akan dilakukan disebut alur penelitian, dan alur penelitian tersebut digambarkan sebagai berikut dapat dilihat pada gambar 3.1



Gambar 3.1 Diagram Alur Penelitian

3.2 Pra Penelitian

Proses ini dilakukan untuk mendapatkan literatur dan referensi jurnal sebagai bahan acuan penelitian ini, di antaranya:

- a. Mempelajari penelitian sebelumnya tentang subjek yang sedang dipelajari.
- b. Mempelajari jenis metode dan algoritma penelitian untuk membantu memahami jenis penelitian yang sedang dilakukan.
- c. Memahami diagram aliran PRISMA untuk analisis data

3.3 Persiapan Penelitian

Ada beberapa langkah yang diambil untuk mempersiapkan penelitian ini, yang dijelaskan di sini: pengumpulan data, perancangan algoritma, dan proses klasifikasi.

3.3.1 Dataset

Dalam penelitian ini, saya menggunakan dataset yang diperoleh dari *Kaggle* yang berisi gambar daun mawar dengan tiga kelas utama: Daun Mawar Sehat (*Healthy Leaf*), Karat Mawar (*Rose Rust*), dan Daun Mawar yang Tergerogoti oleh Ulat (*Rose Sawfly/Rose Slug*). Dataset ini terbagi menjadi tiga bagian utama: data latih, data uji, dan data validasi. Dapat dilihat pada table 3.1 menggambarkan bagaimana dataset yang digunakan dalam penelitian ini dikelompokkan:

Table 3.1 Table Dataset

No	Nama	Jenis	Jumlah
1	<i>Healthy_Leaf_Rose</i>	<i>Training</i>	100
2	<i>Rose_Rust</i>	<i>Training</i>	360
3	<i>Rose_sawfly_Rose_slug</i>	<i>Training</i>	380
7	<i>Healthy_Leaf_Rose</i>	<i>Testing</i>	78
8	<i>Rose_Rust</i>	<i>Testing</i>	53
9	<i>Rose_sawfly_Rose_slug</i>	<i>Testing</i>	79
10	<i>Healthy_Leaf_Rose</i>	<i>Validation</i>	2450
11	<i>Rose_Rust</i>	<i>Validation</i>	2450
12	<i>Rose_sawfly_Rose_slug</i>	<i>Validation</i>	2450

Dataset ini menunjukkan keberagaman yang baik dalam representasi kelas-kelas penyakit daun mawar. Meskipun demikian, terdapat sedikit

ketidakseimbangan dalam jumlah sampel antar kelas, dengan jumlah sampel untuk kelas Rose Rust dan Rose Sawfly/Rose Slug sedikit lebih banyak dibandingkan dengan kelas Healthy Leaf. Namun, perbedaan ini tidak signifikan dan dianggap dapat diterima untuk keperluan penelitian ini.

3.4 Penyesuaian Model CNN

Dalam pengembangan model deep learning, terkadang perubahan kecil pada berbagai komponen model dapat memiliki dampak besar pada kinerja akhir model (Arfida et al., 2023). Dalam konteks ini, saya telah melakukan beberapa penyesuaian terhadap model deep learning yang sudah ada, termasuk perubahan pada fungsi aktivasi, optimizer, dan jumlah epoch. Berikut adalah penjelasan rinci tentang perubahan-perubahan tersebut.

1. Fungsi Aktivasi

Saya memutuskan untuk mengubah fungsi aktivasi dari peneliti sebelumnya ReLU menjadi *Leaky* ReLU. Alasan di balik ini adalah bahwa *Leaky* ReLU dapat membantu mengatasi masalah *neuron* mati yang dapat terjadi dengan ReLU standar. Dengan memperkenalkan gradien kecil untuk input negatif, *Leaky* ReLU dapat membantu dalam proses pembelajaran model dengan lebih baik.

2. Optimizer

Peneliti sebelumnya menggunakan optimizer adam untuk melatih model. Namun, setelah saya mengevaluasi performa model, saya memutuskan untuk mencoba menggunakan optimizer RMSProp. Alasan di balik ini adalah karena RMSProp menggunakan teknik khusus di mana ia menghitung rata-rata bergerak dari kuadrat gradien dalam waktu sebelumnya. Dengan metode ini, RMSProp dapat secara adaptif (menyesuaikan), mengatur kecepatan belajar selama proses pelatihan berdasarkan seberapa besar atau kecilnya gradien yang telah diamati sebelumnya. Teknik ini membantu dalam menangani perubahan yang cepat dalam data atau arsitektur model yang kompleks, sehingga memungkinkan model untuk belajar dengan lebih baik dari pengalaman-pengalaman sebelumnya.

3. Jumlah Epoch

Epochs merupakan salah satu faktor kunci yang memengaruhi waktu pemrosesan dalam deep learning. Oleh karena itu, dalam eksperimen ini, dipilih untuk menggunakan jumlah epochs yang sama pada setiap percobaan. Hal ini dilakukan agar evaluasi model dapat dilakukan secara adil dan valid, tanpa adanya perbedaan yang signifikan dalam waktu pemrosesan yang dapat memengaruhi hasil akhir. Dengan menjaga jumlah epochs dapat membandingkan hasil dari berbagai konfigurasi model secara lebih konsisten dan objektif (Findley, Enrico and Irianto, 2002).

Saya juga menyesuaikan jumlah epoch, penyesuaian ini dilakukan berdasarkan evaluasi terhadap tren loss dan akurasi pada dataset pelatihan dan validasi. Tujuan penyesuaian ini adalah untuk mengoptimalkan performa model dan mencegah overfitting atau underfitting.