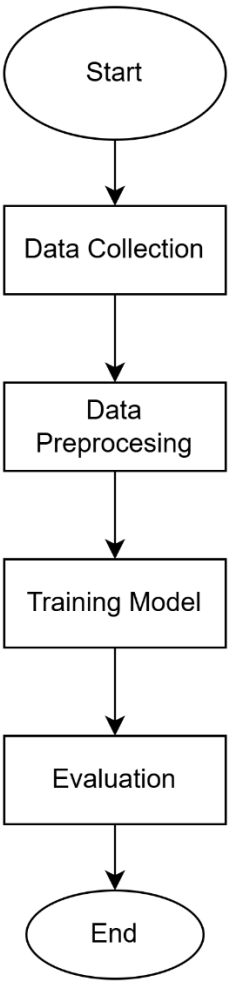
BAB III

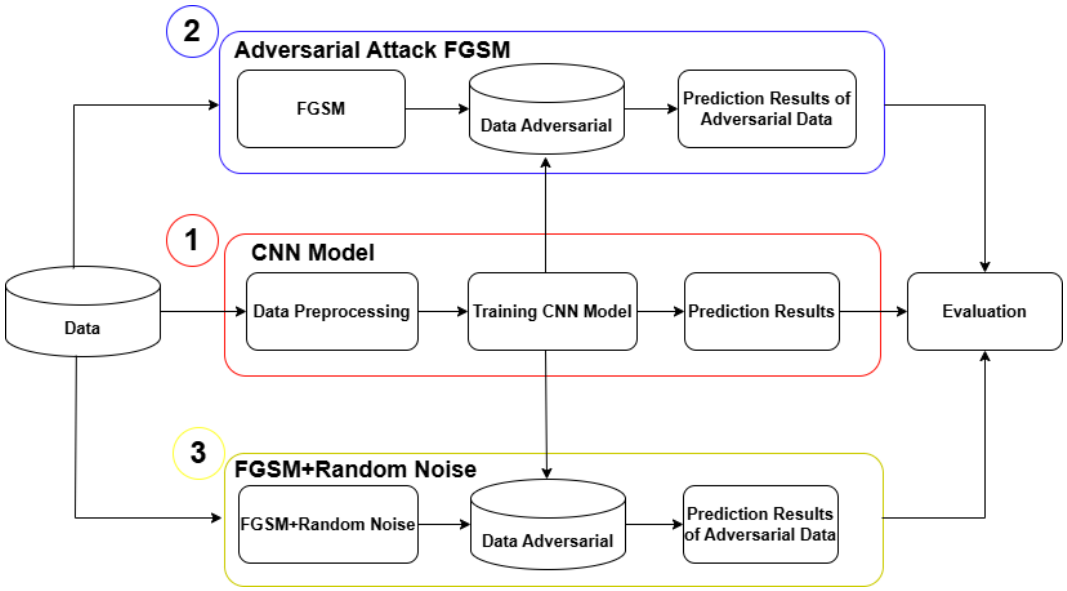
METODOLOGI PENELITIAN

Metode penelitian Metode penelitian ini secara keseluruhan disajikan menggunakan alur penelitian. meningkatkan serangan FGSM dengan menggunakan *random noise* untuk menguji model CNN. Rangkaian tahapan penelitian ini dapat dilihat pada Gambar 3.1.



Gambar 3. 2 Metode Penelitian

Metode penelitian ini secara keseluruhan disajikan menggunakan alur penelitian. meningkatkan serangan FGSM dengan menggunakan *random noise* untuk menguji model CNN melibatkan skenario pengujian seperti yang ditunjukan pada gambar 3.2.



Gambar 3. 2 Skenario Pengujian

Gambar 3.2 merupakan skenario pengujian dari meningkat FGSM menambahkan *random noise* untuk menguji model CNN (Zhao et al., 2022). Penelitian ini menggunakan tiga skenario, pertama modelan CNN tanpa *adversarial attack*, kedua pengujian model CNN menggunakan FGSM, dan ketiga pengujian model CNN

menggunakan FGSM yang digabungkan *random noise*. Ketiga skenario membutuhkan *dataset* citra untuk kebutuhan model CNN dan *adversarial attack*. Pemodelan CNN memerlukan tahapan data *preprocessing* untuk mengolah data dan tahapan *training* model untuk melatih model dengan menggunakan data yang sudah diproses, selanjutnya model akan diuji dengan data *test* (Zhang et al., 2022). Pada tahapan *adversarial attack* dengan FGSM maupun FGSM yang dikombinasikan *random noise*, skenario tersebut masing-masing menggunakan *dataset* citra untuk menguji model CNN. Setelah model diujikan maka akan evaluasi dilihat dari akurasi model yang dihasilkan (Tang & Zhang, 2024). Penjelasan detail setiap tahapannya yaitu sebagai berikut:

3.1. Data

Tahapan ini, untuk membuat sebuah pemodelan CNN maupun *adversarial attack* membutuhkan *dataset* gambar. Model membutuhkan kumpulan data untuk menghasilkan data yang optimal, begitupun *adverserial attack* membutuhkan data gambar untuk dilakukan perubahan. Pada penelitian (Golgooni et al., 2023) dan (Waghela, 2024) menggunakan dataset CIFAR-10 dapat digunakan untuk pemodelan CNN karena memiliki keberagaman kelas dan bervariasi. Maka dari itu, penelitian ini menggunakan dataset CIFAR-10.

3.2. *Convolutional Neural Network*

Tahapan ini merupakan tahapan untuk melakukan pemodelan CNN. Data CIFAR-10 masuk ke dalam tahap *data preprocessing*. Pada tahap *preprocessing, dataset* diaugmentasi supaya lebih bervariasi. Augmentasi yang dilakukan diantaranya adalah rotasi, *width shift, height shift,* dan *horizontal flip* (Musa et al.,

2021). Selanjutnya, *dataset* akan digunakan oleh model CNN untuk *training*. Model CNN akan memahami pola dari *dataset* gambar dilakukan saat *training*. Kemudian, diujikan dengan data uji tanpa dilakukan *adversarial attack.*

3.3. *Adversarial Attack FGSM*

Perubahan data input untuk pengujian model CNN menggunakan FGSM. Data CIFAR-10 diubah menggunakan FGSM yang dijadikan data *test* untuk model CNN. Menguji ketahanan model CNN terhadap serangan adversarial, dilakukan modifikasi pada data input menggunakan metode *Fast Gradient Sign Method* (FGSM). Setiap sampel dalam dataset tersebut dimodifikasi oleh FGSM untuk menghasilkan contoh adversarial yaitu input yang tampak normal bagi manusia, namun dirancang untuk mengecoh model. Hasil modifikasi ini kemudian digunakan sebagai data pengujian untuk model CNN. Tujuan dari pendekatan ini adalah untuk mengevaluasi model dalam menghadapi gangguan kecil yang dapat menyebabkan kesalahan klasifikasi.

3.4. *Adversarial Attack FGSM* dan *Random noise*

Data asli CIFAR-10 akan diproses menggunakan FGSM yang ditambahkan *random noise* untuk menghasilkan data *adversarial*. Setelah itu, data *adversarial* tersebut akan diuji terhadap model CNN. Berikut persamaan (7) *perturbation* merupakan perhitungan dari FGSM (Hassan et al., 2022) yang ditambahkan dengan *random noise* (Barkam et al., 2023).

$$x_{adv} = x + \epsilon \, . sign \, (\nabla_x J(x, y)) + N(\mu, \sigma^2) \quad (7)$$

Persamaan (7) merupakan perhitungan FGSM yang ditambahkan dengan *random noise*. Gambar original CIFAR-10 yang diubah dengan FGSM. Gambar asli diubah berdasarkan gradien fungsi loss $J(x, y)$. Gradien dihitung terhadap input x untuk menunjukan perubahan terkecil yang dapat meningkatkan nilai loss. Nilai gradien diubah menjadi *sign* untuk memberikan arah peningkatan loss. Koefisien ϵ mengatur besarnya perubahan. Kemudian, ditambahkan *random noise* yang mengikuti distribusi *gaussian* dengan rata-rata μ dan variansi σ^2. *Noise* bertujuan memberikan variasi acak pada data *adversarial* yang membuat model lebih sulit mengenali gambar.

3.5. *Evaluation*

Pada tahap ini, performa model dievaluasi menggunakan *confusion matrix*. Tabel ini digunakan untuk membandingkan hasil prediksi model dengan label sebenarnya pada *dataset*. Model dibandingkan hasil akurasi, recall, f1-score dan presisi dari model yang memprediksi dengan data uji asli dan data uji *adversarial attack* (Islam et al., 2022).