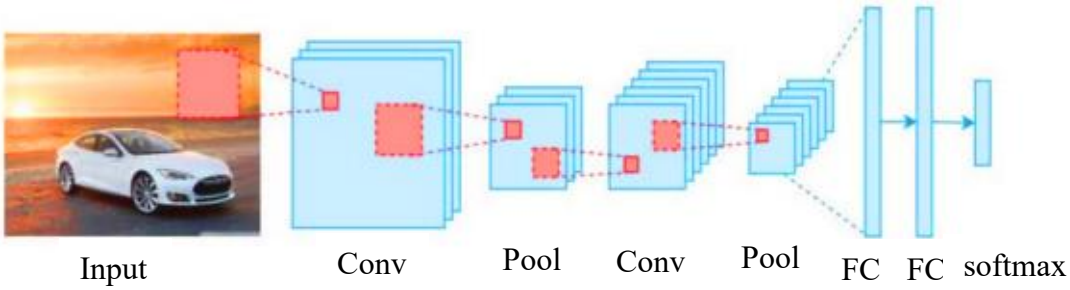
BAB II

LANDASAN TEORI

2.1 *Convolutional Neural Network*

Salah satu arsitektur *neural network* yang dirancang untuk mengolah data spasial, seperti gambar atau video adalah *Convolutional Neural Network* (Shiri et al., 2023). CNN dapat melakukan, pengenalan wajah, objek deteksi, analisis medis dan bahkan *Augmented Reality* (Islam et al., 2022). Kinerja CNN dapat dilihat pada arsitekturnya terdapat pada gambar 2.1.



Gambar 2. 1 CNN Arsitektur (Viswanatha et al., 2022).

Gambar 2.1 menunjukan arsitektur dari CNN. Pada arsitektur tersebut terdapat input gambar yang akan diproses oleh CNN. Kemudian, lapisan *Convolutional* untuk memfilter gambar menghasilkan *feature map* (Sen et al., 2023). Selanjutnya, *Pooling layer* merupakan teknik untuk mengurangi dimensi dari *feature map* dengan mengambil nilai antara maksimum, minimum atau rata-rata (Remerscheid et al., 2022). Setelah beberapa kali diproses konvolusi dan *pooling*, kemudian *Fully Connected Layer* (FC) merupakan hasil menjadi vektor satu dimensi yang bisa digunakan untuk klasifikasi (Antonio et al., 2023). *Softmax*

adalah salah satu fungsi aktivasi yang digunakan dalam CNN untuk masalah klasifikasi multi kelas (Sen et al., 2023).

Peforma model CNN dilihat dari nilasi akurasi yang dihasilkan. Berikut persamaan (1) untuk menghitung akurasi dari model(Abdu-Aguye & Nandakumar, 2023).

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Persamaan (1) digunakan untuk menghitung akurasi, di mana komponennya diperoleh dari *confusion*. *True Positive* (TP) mengacu pada jumlah kasus yang diprediksi sebagai positif dan benar positif. *True Negative* (TN) adalah jumlah kasus yang diprediksi negatif dan benar negatif. *False Positive* (FP) merujuk pada kesalahan model ketika memprediksi kelas positif, padahal sebenarnya negatif. Sedangkan *False Negative* (FN) merupakan kesalahan model ketika memprediksi kelas negatif, padahal sebenarnya positif.

Kinerja model dinilai menggunakan metrik *recall* pada data pengujian. Nilai *recall* tersebut dihitung menggunakan Persamaan (2). (Citra R et al., 2024).

$$recall = \frac{TP}{TP+TN} \quad (2)$$

Persamaan (2) merepresentasikan rasio antara jumlah prediksi benar yang dilakukan oleh model terhadap total sampel aktual pada kelas tersebut. *Recall* menggambarkan kemampuan model dalam mengenali seluruh kasus positif yang ada. Semakin tinggi nilai *recall*, semakin baik model dalam mendeteksi sebagian besar sampel positif.

Kinerja model dievaluasi menggunakan metrik *precision* pada data pengujian. Tingkat *precision* model dihitung menggunakan Persamaan (3). (Guo et al., 2023).

$$Presisi = \frac{TP}{TP+FP} \quad (3)$$

Persamaan (3) menunjukkan perbandingan antara jumlah prediksi positif yang benar dengan seluruh prediksi positif yang dibuat oleh model. Nilai *precision* yang tinggi mengindikasikan bahwa model jarang salah dalam mengidentifikasi sampel sebagai positif.

Kinerja model diukur menggunakan nilai *F1-score* pada data pengujian. Persamaan (4) digunakan untuk menghitung *F1-score* dari model. (Antonio et al., 2023).

$$F1score = 2x\frac{precision\ x\ recall}{precision + recall} \quad (4)$$

Persamaan (4) merupakan rata-rata dari presisi dan recall, yang berfungsi menyeimbangkan kedua metrik tersebut. Metrik ini sangat berguna dalam situasi dengan ketidakseimbangan kelas, karena membantu model mempertimbangkan baik presisi maupun recall secara seimbang.

2.2 *Adversarial*

Adversarial adalah suatu pendekatan di mana input dimodifikasi secara halus sehingga tetap terlihat normal oleh manusia, namun dirancang untuk mengecoh model *machine learning* atau *deep learning* (Sen et al., 2023). Perubahan atau modifikasi ini biasanya sangat kecil sehingga tidak terdeteksi oleh penglihatan manusia, tetapi cukup untuk menyebabkan model menghasilkan prediksi yang salah

(Yang et al., 2022). Metode *adversarial* secara umum dikategorikan menjadi dua bagian, yaitu *adversarial attack* dan *adversarial defense.*

2.2.1 *Adversarial Attack*

Adversarial attack adalah teknik untuk melakukan serangan terhadap kelemahan *machine learning* terutama pada *deep learning*. *Adversarial attack* dimanipulasi secara halus sehingga model dapat menghasilkan prediksi yang salah (Sen et al., 2023). Perubahan dilakukan secara spesifik untuk melakukan serangan model tertentu. Memanfaatkan bahwa pemodelan *machine learning* maupun *deep learning* bergantung pola pola numerik (Waghela, 2024). Terdapat beberapa algoritma *adversarial attack* yaitu, *Fast Gradient Sign Method* (FGSM), *Projected Gradient Descent (*PGD) dan *Carlini & Wagner* (C&W) *attack* (Li, 2021).

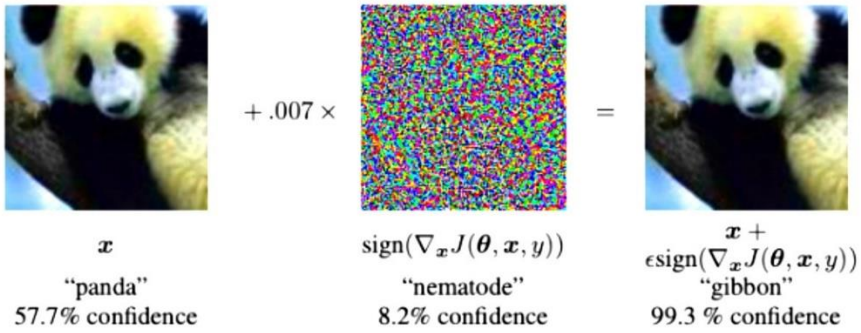
2.3 *Fast Gradient Sign Method*

Fast Gradient Sign Method (FGSM) merupakan salah satu teknik *adversarial attack*. FGSM bekerja dengan memanfaatkan gradien dari fungsi loss pada model, sehingga termasuk dalam kategori *white-box attack* (Sen et al., 2023). Berikut persamaan (5) *perturbation* rumus perhitungan FGSM (Hassan et al., 2022).

$$x_{adv} = x + \epsilon \, . sign \, (\nabla_x J(x, y)) \tag{5}$$

Pada persamaan (5) merupakan perhitugan dari FGSM. Pada persamaan tersebut terdapat x sebagai input asli yang akan diubah dengan menambahkan gangguan kecil. Perhitungan gangguan kecil berdasarkan gradien dari fungsi loss $J(x, y)$. Gradien $(\nabla_x J(x, y))$ menunjukan perubahan pada input x dengan meningkatkan nilai loss (Sen & Dasgupta, 2023). Fungsi *sign* menentukan tanda

positif atau negatif dari gradien untuk menentukan arah perubahan yang minimal tetapi efektif. Kemudian, ϵ merupakan epsilon yang dapat mengatur besar kecilnya perubahan (Hassan et al., 2022). Semakin besar epsilon maka perubahan akan signifikan dan dapat terlihat oleh manusia. Hasil x_{adv} adalah input yang termodifikasi yang tidak terlihat oleh manusia tetapi dapat membuat model salah prediksi. Berikut gambar 2.2 merupakan contoh dari *adversarial attack* menggunakan FGSM.



Gambar 2. 2 Contoh Penerapan FGSM.

Pada gambar 2.2 merupakan penerapan dari FGSM. Terlihat gambar panda sebelah kiri merupakan data input asli yang belum dilakukan perubahan oleh FGSM. Gambar panda disebelah kanan diprediksi *gibbon* oleh model karena gambar tersebut sudah dilakukan perubahan dengan nilai *gradien loss* dengan nilai epsilon 0.007 tapi tidak terlihat perubahan ketika dilihat oleh manusia.

2.4 *Random Noise*

Random noise adalah gangguan yang ditambahkan secara acak ke dalam data. *Random noise* meniru ketidakpastian atau variasi alami yang terjadi secara alami (Pavlitskaya et al., 2022). Serangan *random noise* termasuk black-box

dikarenakan tidak memerlukan informasi model. *Noise* sering diterapkan di dalam banyak bidang, seperti pemrosesan citra, *machine learning*, *deep learning*, dan pelatihan model (Abdu-Aguye & Nandakumar, 2023). Berikut persamaan (6) perhitungan *random noise* menggunakan *Gaussian Noise* (Barkam et al., 2023).

$$N(\mu, \sigma^2) \tag{6}$$

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{(-\frac{(x-\mu)^2}{2\sigma^2})}$$

Persamaan (6) merupakan perhitungan dari *random noise*. Pada perhitungannya terdapat parameter utama yaitu μ nilai rata rata dari distribusi normal (distribusi *gaussian*). Pada umumnya, mean ditetapkan 0, yang berarti *noise* tersebar simetris di sekitar nilai asli data. Standar deviasi (σ) untuk mengontrol besar sebaran *noise*.

2.5 Penelitian Terkait

Penelitian terkait melakukan berbagai pendekatan untuk melakukan serangan terhadap CNN. Beberapa penelitian tersebut menjadi landasan dalam melakukan penelitian ini. Penelitian terkait terdapat pada Tabel 2.1 yang dijadikan melandasi penelitian ini.

Tabel 2. 1 Penelitian Terkait

No	Penulis	*Adversarial Attack*	Pembahasan
1	(Golgooni et al., 2023)	FGD	Model DNN menggunakan data CIFAR-10 dilatih menggunakan FGSM sering terjadi

No	Penulis	*Adversarial Attack*	Pembahasan
			overfitting, sehingga menurunnya akurasi model. Gradien input yang kecil memiliki tanda yang tidak stabil, memperparah *overfitting*. Oleh karena itu, solusi yang dilakukan adalah menggunakan metode *Projected Gradient Descent* (PGD) lebih dapat mengurangi *overfitting* karena model belajar dari data yang diserang secara iteratif, selain itu juga mengefisienkan komputasi.
2	(Yang et al., 2022)	FGSM	Pemodelan DNN menggunakan dataset ImageNet dalam mengenali pola gambar. DNN memiliki kelemahan terhadap perturbasi kecil sehingga rentan dengan serangan. FGSM dapat mengeksplorasi kelemahan DNN dengan perubahan perubahan kecil, kemudian memberikan pelatihan terhadap model supaya dapat mengurangi kerentanan terhadap serangan.
3	(Attias, 2024)	Black-Box (BB)	CNN sensitif terhadap perubahan kecil pada input yang tidak terlihat dan dapat menyebabkan kesalahan prediksi. Regularizer dengan mengidentifikasi pixel

No	Penulis	*Adversarial Attack*	Pembahasan
			similarities dalam data gambar. Pendekatan regularizer lebih praktis untuk mencapai efek serupa. *Black Box* melakukan serangan dengan mengubah input perlahan sampai model melakukan salah prediksi. Peningkatan akurasi lebih tinggi ketika terapkan regularizer ketika mendapatkan serangan.
4	(Sen & Dasgupta, 2023)	FGSM	Serangan *adversarial* FGSM dan *Adversarial* Patch mampu menurunkan kinerja model klasifikasi gambar secara signifikan, bahkan untuk model yang telah teruji keakuratannya seperti ResNet-34, GoogleNet, dan DenseNet-161. Hal ini menegaskan perlunya pengembangan model yang lebih tahan terhadap serangan atau penerapan mekanisme pertahanan yang efektif terhadap serangan *adversarial*.
5	(Tang & Zhang, 2024)	FGSM	Penelitian ini menguji efektivitas Test-Time Pixel-Level *Adversarial* (TPAP) pada DNN menggunakan dataset seperti CIFAR-10 dan ImageNet, termasuk FGSM. Keunggulan

No	Penulis	*Adversarial Attack*	Pembahasan
			TPAP terletak pada kemampuannya meningkatkan ketahanan terhadap serangan yang tidak dikenal tanpa banyak akurasi pada data bersih. Namun, metode ini memiliki kekurangan, yaitu bergantung pada FGSM, sehingga performanya terbatas terhadap serangan dengan pola yang berbeda.
6	(Waghela, 2024)	PGD	Penelitian ini menggabungkan pelatihan *adversarial* dan teknik pra-pemrosesan dengan noise acak untuk meningkatkan ketahanan model ResNet terhadap serangan PGD. Pendekatan ini fleksibel dan efektif, namun membutuhkan komputasi tinggi dan dapat menurunkan performa pada data non-*adversarial*. Pengaturan noise acak harus hati-hati untuk menjaga keseimbangan antara robustitas dan akurasi.
7	(Choi & Tian, 2022)	PGD	Pemodelan YOLO menggunakan dataset KITTI dan COCO untuk menguji efektivitas serangan dan pertahanan PGD. Serangan yang diajukan terbukti lebih efektif (45,17% pada KITTI dan 43,50% pada COCO)

No	Penulis	*Adversarial Attack*	Pembahasan
			dibandingkan serangan tradisional. Sebagai solusi, pendekatan *objectness-aware adversarial training* diperkenalkan, yang berhasil meningkatkan robustitas detektor terhadap serangan sebesar 21% mAP di KITTI dan 12% mAP di COCO.
8	(Hirano et al., 2021)	FGSM	Penelitian ini menguji tujuh arsitektur model DNN yang berbeda untuk melihat sejauh mana kerentanannya terhadap FGSM untuk klasifikasi kanker kulit, retinopati diabetik yang dapat dirujuk, dan pneumonia, baik yang tidak ditargetkan maupun yang ditargetkan, dan menemukan bahwa gangguan ini sangat efektif dalam mengklasifikasikan input secara salah, dengan tingkat keberhasilan lebih dari 80% untuk kedua jenis serangan tersebut.
9	(Wu et al., 2022)	*Random Noise*	Mode *Neural Network* mengurangi serangan yang mencoba mengubah hasil prediksi dan *nilai perturbation*. Model ini diuji menggunakan dataset CIFAR-10 dan serangan menggunakan random noise yang

No	Penulis	*Adversarial Attack*	Pembahasan
			dikombinasikan dengan PGD. Model dapat berhasil mencapai robust accuracy 89.62%. randomized smoothing dengan PGD, yang digunakan untuk membuat model lebih tahan terhadap gangguan. Hasilnya menunjukkan bahwa model yang diusulkan lebih baik daripada teknik randomized smoothing sekitar 5%, terutama ketika ada jari-jari radius yang lebih besar dari 0.5
10	(Lin et al., 2024)	*Random Noise (RN)*	Model VGG memanfaatkan jaringan *adversarial* generatif (GAN) untuk mengatasi tantangan dalam meningkatkan resolusi gambar seismik sekaligus mengurangi *random noise* . Ini sangat penting dalam pencitraan seismik, di mana metode tradisional sering sintetik ini membantu model untuk lebih generalisasi, meskipun hanya dilatih menggunakan data seismik sintetis. Serangan *adversarial* ini berusaha memanipulasi data input (gambar seismik) dengan menambahkan random noise atau gangguan yang tidak terlihat oleh

No	Penulis	*Adversarial Attack*	Pembahasan
			manusia, tetapi dapat mengacaukan model dan menyebabkan kesalahan dalam prediksi atau interpretasi. Model dirancang untuk lebih tahan terhadap serangan tersebut dengan menggunakan noise acak dalam pelatihan, yang membantu model menjadi lebih robust dan dapat menangani variasi dan ketidak sempurnaan data dunia nyata
11	Penelitian ini	FGSM+*Random Noise*	Penelitian ini, menggunakan model CNN yang rentan terhadap serangan terutama pada perubahan. FGSM menggunakan *random noise* untuk *adversarial attack* terhadap CNN untuk menguji kerentanan model. FGSM melakukan serangan tidak hanya berdasarkan pada *gradient loss* tetapi dikombinasikan dengan *random noise* sehingga membuat *noise* secara acak untuk meningkatkan variasi serangan dalam menguji kerentanan CNN. Model dievaluasi untuk mengenali dari serangan FGSM yang dikombinasikan dengan *random noise* untuk mengurangi resiko serangan.

Pada tabel 2.1 menunjukan penelitian terkait sebagai acuan untuk penelitian ini. *Adversarial attack* terhadap model CNN telah diidentifikasi sebagai isu signifikan yang mempengaruhi ketahanan model terhadap perubahan kecil pada input. Beberapa penelitian mengungkapkan bahwa model CNN, seperti yang digunakan pada dataset CIFAR-10, ImageNet, dan dataset lainnya, sangat rentan terhadap perubahan kecil pada data input, yang dapat menyebabkan kesalahan prediksi dan penurunan akurasi (Yang et al., 2022). Salah satu pendekatan yang sering digunakan untuk melakukan serangan ini adalah menggunakan metode FGSM, dapat mengeksploitasi kerentanannya dengan membuat perubahan kecil pada data input (Sen & Dasgupta, 2023).

FGSM bekerja dengan menghitung *gradien* dari fungsi *loss* terhadap input, kemudian mengubah input berdasarkan arah *gradien* untuk menghasilkan data yang diserang, namun serangan ini seringkali menyebabkan *overfitting* pada model (Golgooni et al., 2023).

Penambahkan *random noise* acak dalam *adversarial attack*, FGSM dapat menyebabkan gangguan yang lebih beragam pada model CNN (Wu et al., 2022). Penelitian oleh (Wu et al., 2022) menunjukkan bahwa penggunaan *random noise* bersama dengan PGD menghasilkan akurasi yang lebih baik pada model yang dilatih untuk mengatasi gangguan tersebut dan nilai *perturbation* meningkat. *Random noise* ini membuat serangan lebih sulit diprediksi dan lebih beragam, sehingga menguji ketahanan model CNN dalam menghadapi variasi serangan yang

lebih luas (Lin et al., 2024). Oleh karena itu, kombinasi FGSM dengan *random noise* diharapkan dapat meningkatkan efektivitas serangan dan membantu dalam evaluasi ketahanan model CNN terhadap *adeversarial attack* yang lebih kompleks dan bervariasi. Perbandingan setiap penelitian dapat dianalisis melalui matriks penelitian yang disajikan pada Tabel 2.2.

Tabel 2. 2 Matriks Penelitian

No	**Penulis**	***Adversarial Attack***				***Neural Network***
		FGSM	**PGD**	***Random Noise***	**Black-Box**	
1	(Golgooni et al., 2023)		√			DNN
2	(Yang et al., 2022)	√				DNN
3	(Attias, 2024)				√	CNN.
4	(Sen & Dasgupta, 2023)	√				DenseNet
5	(Tang & Zhang, 2024)	√				DNN
6	(Waghela, 2024)		√			ResNet
7	(Choi & Tian, 2022)		√			YOLO
8	(Hirano et al., 2021)	√				DNN.
9	(Wu et al., 2022)		√	√		*Neural Network*
10	(Lin et al., 2024)			√		VGG
11	Penelitian ini	√		√		CNN

Pada tabel 2.2, matriks penelitian menyoroti perbedaan antara penelitian ini, dengan penelitian lainnya. Peluang penelitian ini, dengan mengembangkan metode FGSM yang menggunakan *gradien loss* untuk menghasilkan serangan pada model. Dalam pendekatan ini, FGSM dimodifikasi dengan menambahkan elemen *random noise*, yang bertujuan untuk meningkatkan variasi dalam serangan terhadap model CNN. Memasukkan komponen *random noise*, serangan yang dilakukan tidak hanya

berdasarkan pada *gradien loss*, tetapi juga menciptakan serangan yang lebih kompleks. Hal ini memberikan tantangan baru bagi model CNN, karena serangan yang dihasilkan lebih beragam, sehingga dapat membantu untuk menguji ketahanan dan kemampuan model dalam menghadapi gangguan yang lebih variatif.