BAB 2
LANDASAN TEORI

Landasan teori pada penelitian ini mencakup konsep AI-Generated Content (AIGC), deteksi gambar sintetis, CNN, ResNet18, *transfer learning*, representasi frekuensi, DCT, FFT, kurasi *dataset*, metrik evaluasi, dan interpretasi keluaran model. Pembahasan penelitian terdahulu digunakan untuk menunjukkan konteks, keterbatasan studi sebelumnya, serta posisi penelitian ini.

2.1 Tinjauan Teori

2.1.1 AI-Generated Content (AIGC) dan *Diffusion Models*

AI-Generated Content (AIGC) merupakan konten digital yang dihasilkan secara otomatis dengan bantuan algoritma *Artificial Intelligence*. Dalam konteks citra, perkembangan AIGC semakin pesat seiring munculnya model generatif modern yang mampu menghasilkan gambar dengan tingkat kemiripan visual yang sangat tinggi terhadap karya buatan manusia. Salah satu pendekatan yang saat ini banyak digunakan dalam sintesis gambar adalah *Denoising Diffusion Probabilistic Models* (DDPM), seperti Stable Diffusion dan Midjourney. Model ini bekerja dengan mempelajari distribusi data melalui dua proses utama, yaitu *forward diffusion* yang menambahkan *noise* secara bertahap pada data, serta *reverse diffusion* yang merekonstruksi kembali data tersebut menjadi citra yang utuh [1].

Dibandingkan dengan generasi model sebelumnya, *diffusion models* mampu menghasilkan gambar dengan kualitas visual yang lebih tinggi dan detail yang lebih halus. Meskipun demikian, gambar yang dihasilkan tetap berpotensi meninggalkan jejak atau *artifacts* tertentu, khususnya pada domain frekuensi. Jejak tersebut umumnya sulit dikenali secara langsung oleh pengamatan manusia, tetapi dapat diidentifikasi melalui analisis komputasi yang lebih mendalam [1]. Oleh karena itu, pemanfaatan pendekatan berbasis frekuensi menjadi penting dalam penelitian deteksi ilustrasi digital buatan AI, karena mampu mengungkap pola-pola tersembunyi yang tidak selalu terlihat pada representasi visual biasa.

2.1.2 Deteksi Gambar Sintetis dan Artefak Frekuensi

Deteksi gambar sintetis bertujuan untuk mengenali apakah suatu citra berasal dari proses pembuatan manusia atau dari model generatif. Pada citra yang dihasilkan oleh model generatif modern, perbedaan visual antara citra sintetis dan citra asli sering kali semakin kecil, sehingga pengamatan manual tidak selalu cukup untuk membedakan keduanya [1]. Kondisi tersebut membuat pendekatan berbasis pembelajaran mesin dan pembelajaran mendalam menjadi relevan karena model dapat mempelajari pola yang tidak mudah diamati secara langsung.

Selain pola pada domain spasial, beberapa penelitian menunjukkan bahwa proses generatif dapat meninggalkan pola tertentu pada domain frekuensi. Frank et al. menunjukkan bahwa operasi *upsampling* pada arsitektur generatif dapat meninggalkan *grid-like artifacts* yang lebih mudah diamati melalui analisis frekuensi [7]. Pada domain anime, permasalahan ini menjadi menarik karena gambar anime memiliki karakteristik visual yang berbeda dari citra natural, seperti garis tepi yang tegas, pewarnaan bergaya, dan tekstur yang tidak selalu mengikuti pola dunia nyata. Oleh karena itu, analisis frekuensi digunakan dalam penelitian ini untuk melihat apakah representasi spektral dapat menjadi informasi tambahan bagi CNN dalam membedakan gambar anime buatan AI dan karya manusia.

2.1.3 Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNN) merupakan salah satu arsitektur *deep learning* yang paling banyak digunakan dalam pemrosesan citra. CNN dirancang untuk memproses data berbentuk grid, seperti gambar, dengan mempertahankan informasi spasial yang terkandung di dalamnya. Menurut Alzubaidi et al. (2021), CNN memiliki keunggulan dibandingkan jaringan saraf tradisional karena kemampuannya dalam mempelajari fitur visual secara otomatis dan hierarkis melalui beberapa lapisan utama [4].

Secara umum, CNN terdiri atas tiga komponen utama. Pertama, *convolutional layer*, yaitu lapisan yang berfungsi untuk mengekstraksi fitur dengan menggunakan *learnable kernels*. Lapisan ini mampu mengenali pola lokal seperti tepi, tekstur, dan bentuk sederhana. Kedua, *pooling layer*, yaitu lapisan yang digunakan untuk mereduksi dimensi spasial atau melakukan *downsampling*, sehingga beban komputasi dapat dikurangi dan risiko *overfitting* menjadi lebih kecil. Ketiga, *fully connected layer*, yaitu lapisan akhir yang berfungsi memetakan

fitur hasil ekstraksi ke dalam kelas keluaran tertentu.

Melalui struktur tersebut, CNN dapat mempelajari representasi citra secara bertahap, mulai dari fitur sederhana pada lapisan awal hingga pola yang lebih kompleks pada lapisan yang lebih dalam. Dalam penelitian ini, CNN digunakan karena kemampuannya dalam menangkap karakteristik visual dari ilustrasi digital, baik yang dibuat oleh manusia maupun yang dihasilkan oleh AI. Dengan demikian, CNN menjadi dasar yang kuat untuk proses klasifikasi citra pada penelitian ini.

2.1.4 ResNet18 dan *Transfer Learning*

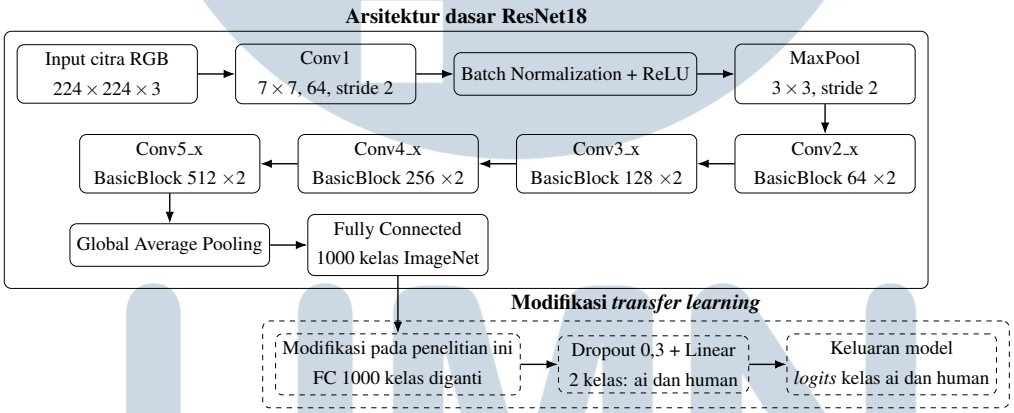
Residual Network (ResNet) merupakan arsitektur CNN yang diperkenalkan untuk mengatasi permasalahan degradasi performa pada jaringan yang semakin dalam. He et al. mengusulkan penggunaan *residual connection*, yaitu koneksi pintas yang memungkinkan informasi dari suatu lapisan diteruskan langsung ke lapisan berikutnya [12]. Secara umum, sebuah blok residual mempelajari selisih atau residual antara masukan dan keluaran yang diharapkan. Jika masukan blok dinyatakan sebagai x dan fungsi konvolusional di dalam blok dinyatakan sebagai $F(x)$, maka keluaran blok dapat ditulis sebagai $y = F(x) + x$ ketika ukuran fitur masukan dan keluaran sama. Penjumlahan ini membuat jaringan tidak harus mempelajari seluruh pemetaan dari awal, tetapi cukup mempelajari koreksi terhadap representasi sebelumnya. Dengan mekanisme tersebut, model dapat mempelajari fungsi residual sehingga proses pelatihan jaringan yang lebih dalam menjadi lebih stabil.

ResNet18 adalah salah satu varian ResNet yang memiliki 18 lapisan berbobot utama. Arsitektur dasar ResNet18 diawali oleh satu lapisan konvolusi awal, *batch normalization*, ReLU, dan *max pooling*, kemudian dilanjutkan oleh empat kelompok *residual layer*. Empat kelompok tersebut tersusun dari delapan *BasicBlock*, dengan setiap *BasicBlock* umumnya terdiri atas dua lapisan konvolusi 3×3 dan satu *residual connection*. Penamaan ResNet18 mengacu pada satu lapisan konvolusi awal, enam belas lapisan konvolusi di dalam delapan *BasicBlock*, dan satu lapisan *fully connected* pada bagian akhir. Setelah proses ekstraksi fitur, ResNet18 menggunakan *global average pooling* dan lapisan klasifikasi untuk menghasilkan skor kelas.

ResNet18 pada penelitian ini digunakan sebagai *backbone* utama karena arsitekturnya relatif ringan, telah banyak digunakan pada tugas klasifikasi citra, dan mendukung penerapan *transfer learning*. Pada skenario RGB, modifikasi

utama dilakukan pada lapisan klasifikasi akhir. Pada skenario gabungan, ResNet18 dipertahankan sebagai cabang spasial dan dipadukan dengan cabang frekuensi yang menerima representasi DCT atau FFT.

Transfer learning merupakan pendekatan yang memanfaatkan bobot awal model yang telah dilatih pada *dataset* besar sebelum disesuaikan dengan *dataset* target. Pendekatan ini dipilih karena model *pretrained* telah mempelajari pola visual umum seperti tepi, tekstur, bentuk, dan komposisi dasar. Secara umum, *transfer learning* dapat dilakukan melalui *feature extraction* atau *fine-tuning*. Pada *feature extraction*, sebagian besar bobot *backbone* dibekukan dan hanya lapisan klasifikasi akhir yang dilatih. Pada *fine-tuning*, bobot *backbone* tetap diperbarui selama proses pelatihan agar representasi yang telah dipelajari dari *dataset* sumber dapat disesuaikan dengan *dataset* target. Pada penelitian ini, bobot awal ResNet18 berasal dari ImageNet, sedangkan lapisan klasifikasi akhir disesuaikan untuk dua kelas penelitian, yaitu *ai* dan *human*.



Gambar 2.1. Alur arsitektur dasar ResNet18 dan modifikasi lapisan klasifikasi melalui *transfer learning*

Gambar 2.1 memperlihatkan arsitektur dasar ResNet18 dan modifikasi lapisan klasifikasi melalui *transfer learning*. Diagram tersebut diadaptasi dari arsitektur ResNet yang dijelaskan oleh He et al. [12]. Arsitektur ResNet18 terdiri atas lapisan konvolusi awal, *batch normalization*, ReLU, *max pooling*, empat kelompok *residual layer*, *global average pooling*, dan lapisan *fully connected*. Pada arsitektur asli, lapisan *fully connected* digunakan untuk menghasilkan 1000 kelas ImageNet. Dalam penelitian ini, bagian tersebut diganti menjadi *dropout* dan lapisan *linear* untuk dua kelas, yaitu *ai* dan *human*. Keluaran dari lapisan *linear* berupa *logits* yang kemudian dapat dikonversi menjadi probabilitas menggunakan fungsi *softmax*.

2.1.5 Representasi Frekuensi pada Citra Digital

Dalam pengolahan citra, domain spasial merepresentasikan gambar berdasarkan nilai piksel pada posisi tertentu, sedangkan domain frekuensi merepresentasikan pola perubahan intensitas pada gambar. Komponen frekuensi rendah umumnya berkaitan dengan struktur global dan perubahan intensitas yang halus, sedangkan komponen frekuensi tinggi berkaitan dengan detail, tepi, tekstur, atau perubahan intensitas yang tajam. Representasi frekuensi digunakan pada penelitian ini karena beberapa artefak hasil proses generatif dapat lebih mudah diamati melalui pola spektral dibandingkan melalui tampilan visual biasa [7].

Pada citra digital, representasi frekuensi tidak menyatakan frekuensi dalam satuan waktu seperti Hz, tetapi frekuensi spasial yang menggambarkan seberapa cepat nilai intensitas piksel berubah pada arah horizontal dan vertikal. Perubahan yang lambat berkaitan dengan area halus atau struktur besar pada gambar, sedangkan perubahan yang cepat berkaitan dengan tepi, detail, dan tekstur. DCT dan FFT digunakan dalam penelitian ini karena keduanya merupakan teknik transformasi yang dapat memetakan citra dari domain spasial ke domain frekuensi, tetapi keduanya menghasilkan bentuk representasi yang berbeda.

2.1.6 Discrete Cosine Transform (DCT)

Discrete Cosine Transform (DCT) merupakan teknik transformasi yang digunakan untuk mengubah representasi sinyal dari domain spasial ke domain frekuensi. Dalam pengolahan citra digital, DCT banyak dimanfaatkan karena mampu memadatkan sebagian besar energi citra ke dalam sejumlah kecil koefisien frekuensi [6]. Karakteristik ini menjadikan DCT efektif untuk menganalisis pola tersembunyi yang sulit diamati secara langsung pada domain spasial.

Secara umum, DCT memisahkan citra menjadi dua kelompok komponen frekuensi. Komponen frekuensi rendah atau *DC component* merepresentasikan informasi utama seperti rata-rata intensitas global citra, sedangkan komponen frekuensi tinggi atau *AC components* memuat detail halus, perubahan tajam, dan tekstur citra. Dalam konteks deteksi citra sintetis, komponen frekuensi tinggi menjadi penting karena berbagai proses generatif sering kali meninggalkan pola anomali pada bagian tersebut.

Persamaan matematis DCT dua dimensi dapat dinyatakan sebagai berikut [6]:

$$F(u,v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cos\left[\frac{(2x+1)u\pi}{2N}\right] \cos\left[\frac{(2y+1)v\pi}{2N}\right] \quad (2.1)$$

Pada persamaan tersebut, $f(x,y)$ menyatakan nilai piksel pada domain spasial, sedangkan $F(u,v)$ menyatakan koefisien DCT pada posisi frekuensi (u,v). Indeks u dan v tidak menunjukkan posisi piksel, tetapi menunjukkan basis kosinus horizontal dan vertikal. Nilai u dan v yang kecil mewakili perubahan intensitas yang lambat atau frekuensi rendah, sedangkan nilai yang besar mewakili perubahan intensitas yang cepat atau frekuensi tinggi.

2.1.7 Fast Fourier Transform (FFT)

Fast Fourier Transform (FFT) merupakan algoritma efisien untuk menghitung *Discrete Fourier Transform* (DFT). DFT digunakan untuk mengubah representasi sinyal atau citra dari domain spasial ke domain frekuensi. Cooley dan Tukey memperkenalkan algoritma FFT yang memungkinkan perhitungan transformasi Fourier dilakukan dengan kompleksitas komputasi yang lebih rendah dibandingkan perhitungan DFT secara langsung [13].

Dalam pengolahan citra digital, FFT dua dimensi dapat digunakan untuk menganalisis distribusi frekuensi pada gambar. Komponen frekuensi rendah umumnya merepresentasikan struktur global citra, sedangkan komponen frekuensi tinggi merepresentasikan perubahan intensitas yang tajam, detail halus, dan tekstur. Representasi FFT menghasilkan bilangan kompleks, sehingga informasi yang dapat dianalisis mencakup *magnitude spectrum* dan *phase spectrum*.

Persamaan DFT dua dimensi dapat dinyatakan sebagai berikut:

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} \quad (2.2)$$

Pada persamaan tersebut, $f(x,y)$ menyatakan nilai piksel pada posisi (x,y), sedangkan $F(u,v)$ menyatakan koefisien frekuensi kompleks pada posisi (u,v). Nilai magnitudo menggambarkan kekuatan komponen frekuensi tertentu, sedangkan nilai fase menyimpan informasi pergeseran atau struktur posisi dari komponen frekuensi tersebut.

Berbeda dari DCT yang menghasilkan koefisien riil berbasis kosinus, FFT

menghasilkan koefisien kompleks. Akibatnya, representasi FFT pada penelitian ini dipisahkan menjadi dua kanal, yaitu *log magnitude spectrum* dan *phase spectrum*. Magnitudo menunjukkan energi atau kekuatan frekuensi, sedangkan fase menyimpan informasi posisi relatif komponen frekuensi. Perbedaan ini menjadi alasan FFT digunakan sebagai pembanding DCT, karena keduanya sama-sama bekerja di domain frekuensi tetapi menyediakan representasi numerik yang berbeda.

2.1.8 Perbedaan DCT dan FFT

DCT dan FFT sama-sama mengubah citra dari domain spasial ke domain frekuensi, tetapi keduanya memiliki fungsi dan karakteristik yang berbeda. DCT menggunakan basis kosinus dan menghasilkan koefisien riil, sehingga umum digunakan untuk pemadatan energi dan analisis komponen frekuensi citra. Pada visualisasi DCT, komponen frekuensi rendah berada di bagian kiri atas peta koefisien. Sebaliknya, FFT menghitung spektrum Fourier kompleks yang memuat magnitudo dan fase. Setelah *FFT shift*, komponen frekuensi rendah ditampilkan di tengah spektrum. Dengan demikian, DCT lebih sederhana dari sisi kanal fitur karena menghasilkan satu peta magnitudo, sedangkan FFT memberikan informasi tambahan berupa fase.

Pada penelitian ini, DCT berfungsi untuk melihat apakah pola koefisien kosinus, terutama komponen frekuensi menengah dan tinggi, dapat membantu membedakan gambar anime buatan AI dari karya manusia. FFT berfungsi sebagai pembanding karena dapat menangkap distribusi energi frekuensi serta fase. Perbedaan hasil DCT dan FFT tidak berarti salah satu transformasi lebih benar, melainkan menunjukkan cara representasi frekuensi yang berbeda. Oleh karena itu, hasil DCT dan FFT dievaluasi melalui performa klasifikasi, bukan melalui kesamaan bentuk visual spektrum.

2.1.9 Kombinasi Fitur Spasial dan Frekuensi dalam Klasifikasi Citra

Dalam klasifikasi citra, informasi yang diperoleh dari domain spasial dan domain frekuensi dapat saling melengkapi. Domain spasial memungkinkan model mengenali pola visual yang tampak langsung pada citra, seperti bentuk, struktur, warna, dan tekstur. Sementara itu, domain frekuensi memungkinkan model mengidentifikasi distribusi energi dan pola spektral yang tidak selalu terlihat secara kasatmata. Dengan menggabungkan kedua pendekatan tersebut, sistem klasifikasi

dapat memperoleh representasi citra yang lebih menyeluruh.

Pada penelitian ini, CNN digunakan untuk mengekstraksi fitur spasial dari ilustrasi digital, sedangkan DCT dan FFT digunakan untuk menangkap karakteristik frekuensi yang berpotensi mengandung *artifacts* hasil generasi AI. Kombinasi ini dipilih karena ilustrasi digital buatan AI tidak hanya dapat dianalisis dari tampilan visualnya, tetapi juga dari pola frekuensi tersembunyi yang dihasilkan selama proses sintesis citra. Dengan demikian, integrasi CNN dengan DCT dan FFT digunakan sebagai skenario pembanding untuk menilai pengaruh fitur frekuensi terhadap performa klasifikasi.

2.1.10 Kurasi *Dataset*, *Data Leakage*, dan *Perceptual Hash*

Pada tugas klasifikasi citra, keberadaan gambar yang sama atau sangat mirip pada subset data yang berbeda dapat menimbulkan risiko *data leakage*. *Data leakage* terjadi ketika informasi dari data validasi atau data uji secara tidak langsung telah muncul pada data latih, sehingga hasil evaluasi dapat terlihat lebih tinggi daripada kemampuan generalisasi model yang sebenarnya. Oleh karena itu, pemeriksaan duplikasi visual menjadi bagian penting dalam proses kurasi *dataset* sebelum pelatihan model dilakukan.

Perceptual hash adalah teknik pembentukan sidik jari citra yang dirancang agar gambar yang secara visual sama atau sangat mirip menghasilkan nilai hash yang sama atau berdekatan. Berbeda dari hash kriptografis yang sangat sensitif terhadap perubahan bit kecil, *perceptual hash* berusaha mempertahankan kemiripan berdasarkan persepsi visual. Monga dan Evans menjelaskan bahwa *perceptual image hashing* dapat dibangun berdasarkan fitur visual yang relatif tahan terhadap distorsi yang tidak signifikan secara perseptual [14]. Kajian lain mengenai *image hashing* juga menunjukkan bahwa teknik ini umum digunakan untuk menemukan gambar duplikat, *near-duplicate*, atau gambar yang memiliki kemiripan visual dalam kumpulan data besar [15].

Dalam penelitian ini, *perceptual hash* digunakan sebagai alat bantu kurasi *dataset*, bukan sebagai fitur masukan model. Pemeriksaan dilakukan untuk menemukan gambar dengan nilai hash yang sama pada *dataset* aktif. Apabila beberapa gambar berada dalam kelompok hash yang sama, hanya satu gambar dipertahankan dan sisanya dipindahkan dari *dataset* aktif. Langkah ini penting untuk mengurangi potensi kebocoran data antarsubset, misalnya ketika gambar yang sama atau sangat mirip muncul sekaligus pada data latih dan data uji. Namun,

metode ini tetap memiliki keterbatasan karena hash yang sama terutama menangkap duplikasi yang sangat dekat, sedangkan variasi seperti *crop*, penyuntingan sebagian, atau perubahan gaya yang lebih besar dapat menghasilkan hash berbeda. Pustaka ImageHash digunakan pada implementasi penelitian karena menyediakan beberapa metode hash citra, seperti *average hashing*, *perceptual hashing*, *difference hashing*, dan *wavelet hashing* [16].

2.1.11 Ketidakseimbangan Kelas, *Class Weight*, dan *Macro F1-score*

Ketidakseimbangan kelas terjadi ketika jumlah sampel pada satu kelas jauh lebih besar dibandingkan kelas lain. Kondisi ini dapat membuat model lebih condong memprediksi kelas mayoritas dan menghasilkan *accuracy* yang tinggi, tetapi performa pada kelas minoritas rendah [17]. Pada penelitian ini, jumlah data kelas *ai* lebih besar dibandingkan kelas *human*, sehingga evaluasi tidak cukup hanya mengandalkan *accuracy*.

Salah satu pendekatan untuk menangani ketidakseimbangan kelas adalah *class weighting*, yaitu memberikan bobot kesalahan yang lebih besar pada kelas dengan jumlah sampel lebih sedikit. Dengan demikian, kesalahan pada kelas minoritas memiliki pengaruh yang lebih besar terhadap proses optimasi dibandingkan jika seluruh kelas diberi bobot yang sama. Selain itu, *macro F1-score* digunakan sebagai metrik utama pemilihan model karena menghitung rata-rata performa setiap kelas dengan bobot yang sama, sehingga performa kelas minoritas tidak tertutupi oleh dominasi kelas mayoritas [18, 11].

2.1.12 Metrik Evaluasi Kinerja Model

Dalam penelitian ini, model klasifikasi digunakan untuk membedakan ilustrasi digital buatan AI dan ilustrasi digital buatan manusia. Oleh karena itu, diperlukan metrik evaluasi yang dapat menggambarkan performa model secara komprehensif. Salah satu metrik yang paling umum digunakan adalah *accuracy*, yaitu persentase prediksi yang benar terhadap seluruh data uji. Namun, *accuracy* saja tidak selalu cukup untuk menilai performa model secara menyeluruh, terutama apabila distribusi kelas tidak seimbang atau ketika konsekuensi kesalahan pada masing-masing kelas memiliki tingkat kepentingan yang berbeda [11].

Metrik evaluasi klasifikasi umumnya diturunkan dari *confusion matrix*, yang terdiri atas empat komponen utama, yaitu *True Positive* (TP), *True Negative* (TN),

False Positive (FP), dan *False Negative* (FN). Berdasarkan komponen tersebut, beberapa metrik yang digunakan dalam penelitian ini adalah sebagai berikut:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}, \tag{2.3}$$

$$\text{Precision} = \frac{TP}{TP+FP}, \tag{2.4}$$

$$\text{Recall} = \frac{TP}{TP+FN}, \tag{2.5}$$

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{2.6}$$

Accuracy memberikan gambaran umum mengenai seberapa sering model menghasilkan prediksi yang benar. *Precision* menunjukkan proporsi prediksi positif yang benar-benar termasuk ke dalam kelas positif. *Recall* menunjukkan kemampuan model dalam mendeteksi seluruh data yang benar-benar termasuk ke dalam kelas positif. Sementara itu, *F1-score* merupakan metrik gabungan yang menyeimbangkan nilai *precision* dan *recall*, sehingga cocok digunakan ketika kedua jenis kesalahan, yaitu *False Positive* dan *False Negative*, sama-sama perlu diperhatikan.

Pada klasifikasi dengan distribusi kelas tidak seimbang, metrik *precision*, *recall*, dan *F1-score* dapat dihitung menggunakan *macro average* dan *weighted average*. *Macro average* menghitung rata-rata metrik setiap kelas dengan bobot yang sama, sehingga cocok untuk melihat performa model terhadap setiap kelas secara seimbang. Sebaliknya, *weighted average* menghitung rata-rata metrik dengan mempertimbangkan jumlah data pada masing-masing kelas, sehingga lebih dipengaruhi oleh kelas dengan jumlah sampel lebih besar. Menurut Thambawita et al., penggunaan beberapa metrik evaluasi secara bersamaan penting untuk memberikan gambaran performa model yang lebih realistis dan mengurangi bias dalam interpretasi hasil klasifikasi [11]. Oleh sebab itu, penelitian ini tidak hanya menggunakan *accuracy*, tetapi juga *precision*, *recall*, *F1-score*, dan *confusion matrix*.

2.1.13 Interpretasi Keluaran Model Berbasis *Softmax*

Model klasifikasi pada penelitian ini menghasilkan dua nilai keluaran mentah atau *logits*, yaitu skor untuk kelas *ai* dan skor untuk kelas *human*. Nilai *logits* belum dapat langsung dibaca sebagai probabilitas karena nilainya tidak dibatasi pada rentang 0 sampai 1 dan jumlahnya tidak harus sama dengan 1. Untuk memperoleh probabilitas kelas, fungsi *softmax* digunakan pada keluaran model.

$$p_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{2.7}$$

Pada persamaan tersebut, z_i adalah *logit* untuk kelas ke-i, C adalah jumlah kelas, dan p_i adalah probabilitas kelas ke-i setelah *softmax*. Prediksi kelas ditentukan dari probabilitas tertinggi. *Confidence level* pada sistem inferensi kemudian didefinisikan sebagai nilai probabilitas *softmax* dari kelas yang diprediksi. Sebagai contoh, apabila model menghasilkan probabilitas 0,92 untuk kelas *ai* dan 0,08 untuk kelas *human*, maka prediksi model adalah *ai* dengan *confidence level* 92%. Nilai ini digunakan untuk membantu interpretasi prediksi tunggal, bukan sebagai metrik evaluasi utama penelitian. Evaluasi utama tetap didasarkan pada *accuracy*, *precision*, *recall*, *F1-score*, *confusion matrix*, dan waktu inferensi. Selain itu, nilai *confidence level* tidak selalu berarti probabilitas tersebut telah terkalibrasi sempurna terhadap kenyataan. Guo et al. menunjukkan bahwa jaringan saraf modern dapat memiliki probabilitas prediksi yang kurang terkalibrasi, sehingga nilai probabilitas *softmax* perlu dipahami sebagai interpretasi keyakinan relatif model, bukan sebagai jaminan peluang kebenaran yang sempurna [19].

2.2 Penelitian Terdahulu

Penelitian mengenai deteksi gambar sintetis telah banyak berkembang seiring meningkatnya kualitas model generatif. Sebagian besar penelitian terdahulu banyak membahas deteksi *deepfake*, manipulasi wajah, atau citra sintetis pada domain citra natural. Frank et al. menunjukkan bahwa gambar yang dihasilkan oleh arsitektur generatif dapat memiliki pola khas pada domain frekuensi, terutama akibat proses *upsampling* yang meninggalkan artefak spektral tertentu [7]. Temuan ini menunjukkan bahwa analisis pada domain frekuensi dapat menjadi pelengkap penting terhadap analisis visual pada domain spasial.

Corvi et al. meneliti deteksi citra sintetis yang dihasilkan oleh model

diffusion dan menunjukkan bahwa citra dari generator modern semakin sulit dibedakan secara visual dari citra asli [1]. Zhu et al. memperkenalkan GenImage sebagai *benchmark* berskala besar untuk mengevaluasi kemampuan model dalam mendeteksi citra yang dihasilkan oleh berbagai model generatif [2]. Kedua penelitian tersebut menegaskan bahwa deteksi citra AI perlu diuji secara sistematis menggunakan model klasifikasi yang mampu menangkap pola visual maupun pola tersembunyi pada gambar.

Pada domain anime, Zhu et al. memperkenalkan AnimeDL-2M sebagai *dataset* berskala besar untuk deteksi dan lokalisasi gambar anime buatan AI pada era model *diffusion* [10]. *Dataset* tersebut menunjukkan bahwa domain anime memiliki karakteristik visual yang berbeda dari citra natural, sehingga membutuhkan perhatian khusus dalam pengembangan metode deteksi. Selain itu, Ju et al. mengusulkan pendekatan *Global and Local Feature Fusion* (GLFF) untuk menggabungkan berbagai representasi fitur dalam deteksi citra buatan AI [5]. Penelitian lain seperti FreqNet juga menunjukkan bahwa informasi frekuensi dapat dimanfaatkan untuk mendukung proses deteksi citra sintetis [9].

Penelitian terkait dari Knowledge Center Universitas Multimedia Nusantara yang dekat dengan topik ini adalah penelitian Kusuma dan Natalia mengenai deteksi gambar anime buatan AI menggunakan *deep learning* [20]. Penelitian tersebut menggunakan pendekatan *transfer learning* dengan MobileNetV2 dan MobileNetV3 untuk membedakan gambar anime buatan NovelAI dan gambar karya manusia dari Danbooru2021. Hasil penelitian tersebut menunjukkan bahwa klasifikasi gambar anime buatan AI dapat dilakukan dengan performa tinggi, dengan akurasi pada rentang 96,8% sampai 97,2%.

Keterkaitan penelitian tersebut dengan tugas akhir ini terletak pada domain dan tujuan klasifikasi, yaitu sama-sama membedakan gambar anime buatan AI dan gambar anime karya manusia. Perbedaannya, penelitian ini menggunakan *dataset* AnimeDL-2M yang lebih besar dan berfokus pada analisis representasi frekuensi berbasis DCT dan FFT pada CNN. Dengan demikian, penelitian ini melanjutkan ruang kajian pada domain anime dengan menitikberatkan perbandingan fitur visual RGB dan fitur frekuensi, termasuk varian fitur frekuensi yang mempertahankan informasi kanal RGB.

2.3 *Research Gap* dan Posisi Penelitian

Berdasarkan penelitian-penelitian terdahulu, deteksi citra buatan AI telah banyak dikaji pada domain citra natural, *deepfake*, manipulasi wajah, dan *benchmark* umum. Namun, penelitian yang secara khusus mengevaluasi klasifikasi gambar anime buatan AI dan karya manusia masih lebih terbatas. Padahal, anime memiliki karakteristik visual yang berbeda dari foto dunia nyata, seperti garis tepi yang tegas, pewarnaan stilisasi, dan tekstur yang tidak selalu mengikuti distribusi citra natural.

Penelitian Kusuma dan Natalia telah menunjukkan bahwa klasifikasi gambar anime buatan AI dan karya manusia dapat dilakukan menggunakan *transfer learning*. Akan tetapi, penelitian tersebut belum membahas representasi frekuensi DCT dan FFT serta belum menggunakan AnimeDL-2M sebagai *dataset*. Di sisi lain, AnimeDL-2M telah menyediakan *dataset* khusus untuk domain anime, tetapi masih terdapat ruang untuk mengevaluasi bagaimana fitur spasial dan fitur frekuensi sederhana dapat dibandingkan secara langsung pada tugas klasifikasi anime buatan AI dan karya manusia. Oleh karena itu, penelitian ini mengisi celah tersebut dengan membandingkan ResNet18 RGB, ResNet18+DCT *Grayscale*, dan ResNet18+FFT *Grayscale* pada subset AnimeDL-2M. Setelah hasil awal dianalisis, penelitian ini juga menambahkan ResNet18+DCT RGB dan ResNet18+FFT RGB sebagai eksperimen lanjutan untuk melihat pengaruh informasi frekuensi berbasis kanal warna.

Ringkasan penelitian terdahulu dan posisi penelitian ini ditunjukkan pada Tabel 2.1.

Tabel 2.1. Ringkasan penelitian terdahulu dan *research gap*

Penelitian	**Fokus dan Metode**	**Keterbatasan atau Gap**	**Posisi terhadap Penelitian Ini**
Frank et al. (2020) [7]	Deteksi gambar sintetis menggunakan analisis frekuensi.	Belum berfokus pada domain anime dan belum membandingkan DCT serta FFT untuk klasifikasi anime AI dan karya manusia.	Menjadi dasar penggunaan artefak frekuensi sebagai fitur pendukung deteksi citra buatan AI.
Zhu et al. (2023) [2]	Benchmark deteksi gambar AI pada berbagai generator.	Berfokus pada citra natural dan *benchmark* umum, bukan karakteristik visual khusus anime.	Menunjukkan pentingnya evaluasi detektor gambar AI secara sistematis.
Ju et al. (2023) [5]	Deteksi citra sintesis AI dengan *Global and Local Feature Fusion* (GLFF).	Membahas fusi fitur, tetapi tidak mengevaluasi DCT dan FFT secara khusus pada gambar anime.	Mendukung gagasan bahwa penggabungan beberapa jenis fitur dapat membantu deteksi citra AI.
Tan et al. (2024) [9]	Deteksi deepfake melalui pembelajaran berbasis frekuensi.	Tidak menjadikan anime sebagai domain utama dan tidak membandingkan baseline RGB dengan DCT dan FFT secara langsung.	Menguatkan alasan penggunaan fitur frekuensi sebagai pelengkap fitur spasial.
Zhu et al. (2025) [10]	Deteksi dan lokalisasi gambar anime buatan AI melalui AnimeDL-2M.	Menyediakan *dataset* khusus anime, tetapi masih membuka ruang untuk perbandingan CNN dengan representasi DCT dan FFT yang sederhana.	Menjadi sumber *dataset* dan dasar pemilihan domain anime pada penelitian ini.
Kusuma dan Natalia (2024) [20]	Klasifikasi gambar anime AI dan karya manusia menggunakan MobileNetV2 dan MobileNetV3.	Belum mengevaluasi representasi frekuensi DCT dan FFT serta belum menggunakan AnimeDL-2M.	Menjadi pembanding terdekat pada domain anime AI dan karya manusia.
Penelitian ini	Klasifikasi gambar anime AI dan karya manusia menggunakan lima skenario ResNet18.	Dibatasi pada domain anime, subset AnimeDL-2M, dan lima skenario ResNet18; belum menguji lintas generator atau ketahanan pasca-pemrosesan.	Mengisi gap dengan mengevaluasi pengaruh fitur frekuensi DCT dan FFT pada klasifikasi anime AI dan karya manusia.