BAB 2
LANDASAN TEORI

2.1 *Uncanny Valley*

Uncanny valley merupakan fenomena di mana respons manusia terhadap entitas non-manusia seperti robot atau *avatar* digital meningkat seiring dengan kemiripannya terhadap manusia, tetapi hanya sampai titik tertentu. ketika kemiripan tersebut sangat tinggi tetapi masih terasa ”tidak sempurna” atau artifisial, justru muncul perasaan tidak nyaman, aneh, atau bahkan menyeramkan. Fenomena inilah yang disebut sebagai *uncanny valley*. Beberapa penjelasan psikologis tentang *uncanny valley effects* (UVE) mencakup ketidaksesuaian antara harapan dan perilaku yang ditunjukkan oleh agen, ketidakmampuan untuk mengkategorikan agen sebagai manusia atau mesin, seperti *avatar chatbot* yang sangat menyerupai manusia, tetapi tidak sepenuhnya realistis, dapat memicu perasaan tidak nyaman tersebut, yang pada akhirnya berdampak negatif terhadap pengguna tersebut [2].

2.2 *ChatGPT*

ChatGPT merupakan salah satu model bahasa berbasis kecerdasan buatan *Artificial Intelligence* (AI) yang dikembangkan oleh *OpenAI*. Model ini termasuk dalam kategori Large Language Models (LLM), yang dilatih menggunakan teknik *deep learning*, khususnya arsitektur transformer. *ChatGPT* dirancang untuk memahami dan menghasilkan teks secara alami dalam berbagai konteks percakapan, sehingga mampu berperan sebagai *virtual avatar* dalam menjawab pertanyaan, memberikan penjelasan, hingga menulis konten dalam berbagai gaya bahasa [6]. *ChatGPT* menggunakan *unsupervised learning* dan *reinforcement learning from human feedback* (RLHF) untuk menyesuaikan jawabannya berdasarkan preferensi dan umpan balik manusia. Kemampuan ini menjadikan *ChatGPT* sebagai alat bantu yang sangat potensial dalam berbagai bidang, termasuk pendidikan, penelitian, dan pengembangan perangkat lunak [10].

2.3 *Text-to-Speech*

Text-to-Speech (TTS) adalah teknologi yang mengubah teks tertulis menjadi suara sintetis menyerupai manusia. Teknologi ini memainkan peran penting dalam

pengembangan *virtual avatar*, pembaca layar bagi penyandang disabilitas, serta aplikasi interaktif berbasis suara lainnya. Sistem TTS modern umumnya terdiri dari *text analysis* yang bertugas mengubah teks menjadi representasi linguistik dan *speech synthesis* menghasilkan gelombang suara dari representasi tersebut [11]. Salah satu platform TTS yang canggih dan banyak digunakan saat ini adalah *Microsoft Azure Speech Service*. Layanan ini memungkinkan pengembang untuk menyintesis ucapan dalam berbagai bahasa dan gaya bicara menggunakan model neural *Text-to-Speech* (Neural TTS). Salah satu keunggulan utama dari *Azure* adalah kemampuannya untuk menghasilkan suara yang sangat natural, menyerupai intonasi dan ritme manusia [7].

2.3.1 *Microsoft Azure Text-to-Speech*

Microsoft Azure Text-to-Speech (*Azure* TTS) adalah layanan berbasis *cloud* yang disediakan oleh *Microsoft* melalui platform *Azure Cognitive Services*. Layanan ini memungkinkan konversi teks menjadi suara secara *real-time* menggunakan teknologi sintetis berbasis kecerdasan buatan (AI). *Azure* TTS mendukung ratusan suara dalam berbagai bahasa dan aksen, dengan pilihan gaya bicara seperti ramah, profesional, atau ekspresif.

Salah satu keunggulan utama *Azure* TTS adalah fitur menghasilkan *speech viseme* dengan menganalisis dan memprediksi suara TTS menjadi *viseme ID, Scalable Vector Graphics* (SVG), atau *blend shapes*. Fitur ini sangat berguna untuk animasi bibir sinkron secara *real-time*, cocok untuk aplikasi seperti *virtual avatar*. *Azure* TTS juga menyediakan kontrol granular terhadap *output* suara melalui fitur *speech synthesis Markup Language* (SSML), yang memungkinkan pengembang mengatur jeda, intonasi, penekanan, dan bahkan pengucapan fonetik untuk kata tertentu [7].

2.4 *MetaHuman*

MetaHuman adalah teknologi generasi terbaru dari *Epic Games* yang memungkinkan pembuatan karakter manusia digital dengan tingkat realisme yang sangat tinggi. Dirancang untuk *Unreal Engine* 5, *MetaHuman* menggabungkan sistem *rigging* canggih, pemrosesan *real-time*, dan pustaka data karakter berkualitas tinggi untuk menghasilkan wajah dan tubuh digital yang tampak hidup. Keunggulan utama dari *MetaHuman* terletak pada kemampuannya dalam menyimulasikan

ekspresi wajah secara mendetail, termasuk perubahan halus pada struktur wajah dan tekstur kulit yang merespons pergerakan otot-otot wajah secara natural.

Salah satu fitur paling menonjol dari *MetaHuman* adalah deformasi wajah dinamis. Saat ekspresi wajah berubah misalnya tersenyum, mengernyit, atau berbicara bentuk wajah tidak hanya mengalami pergeseran posisi secara kasar, tetapi juga terjadi deformasi otot dan kulit yang lebih realistis. Misalnya, saat alis dinaikkan, kulit di dahi akan tampak berkerut, atau saat tersenyum, bagian pipi ikut terangkat dan membentuk lipatan sesuai anatomi wajah manusia. Hal ini dimungkinkan karena *MetaHuman* didukung oleh sistem *Facial Rig* yang kompleks dan *pose-driven animation* yang secara akurat memodelkan pergerakan wajah manusia berdasarkan data pemindaian aktor nyata.

Selain bentuk, tekstur wajah *MetaHuman* juga bersifat responsif terhadap perubahan ekspresi. Tidak hanya gerakan otot yang berubah, tetapi tekstur kulit seperti kerutan, garis halus, dan bayangan akan menyesuaikan secara *real-time* terhadap kondisi wajah yang sedang dianimasikan. Ini menciptakan efek visual yang jauh lebih hidup dan meyakinkan, mengurangi kesan artifisial yang sering muncul pada karakter digital. Misalnya, saat wajah menunjukkan ekspresi marah, kulit sekitar mata akan tampak mengencang, muncul guratan halus di antara alis, dan bayangan di wajah ikut berubah, sehingga ekspresi emosi terlihat lebih nyata.

Karakter *MetaHuman* juga dirancang dengan material *shading* dan *subsurface scattering* yang realistis, yang meniru bagaimana cahaya menembus dan dipantulkan oleh lapisan kulit manusia. Hal ini membuat pencahayaan pada wajah *MetaHuman* terlihat sangat natural, terutama ketika digunakan dalam lingkungan dengan pencahayaan dinamis seperti di *Unreal Engine* 5. Selain itu, simulasi rambut, mata, dan gigi juga dirancang dengan kualitas sinematik, yang menjadikan karakter terlihat hidup dari berbagai sudut pandang dan jarak.

Kemampuan *MetaHuman* dalam menangani animasi ekspresi wajah dengan akurat dan efisien menjadikannya sangat ideal untuk berbagai aplikasi, mulai dari film, gim, hingga simulasi interaktif. Karakter yang dibuat dengan *MetaHuman* dapat langsung digunakan dalam *pipeline* produksi *Unreal Engine*, lengkap dengan *control rig* bawaan yang memungkinkan pengendalian ekspresi wajah, *lip-sync*, dan gerakan tubuh secara presisi [12].

2.5 *Lip-sync*

Lip-sync adalah proses mencocokkan gerakan bibir karakter digital dengan *audio* ucapan, sehingga tampak seolah-olah karakter tersebut benar-benar berbicara. Teknologi *lip-sync* memainkan peran penting dalam bidang animasi atau permainan video. Sinkronisasi yang tepat sangat berpengaruh terhadap persepsi realisme dan kualitas interaksi manusia-komputer [8]. *Lip-sync* memerlukan pemahaman mendalam tentang fonem dan bagaimana masing-masing fonem diwakili oleh bentuk mulut tertentu yang sering disebut sebagai *viseme*. Animator tradisional secara manual menggambar rangkaian bentuk mulut untuk setiap suara dalam dialog, yang kemudian dianimasikan agar bergerak seiring dengan suara. Namun dengan kemajuan teknologi komputer, *lip-sync* kini dianimasikan secara *real time* menggunakan teknologi seperti *deep learning* dan *3D morphable models* [13].

2.5.1 Fonem

Fonem adalah unit bunyi terkecil dalam suatu bahasa yang memiliki fungsi untuk membedakan makna antar kata. Fonem sendiri tidak memiliki makna secara independen, tetapi ketika dikombinasikan dengan fonem lain, dapat membentuk kata-kata yang memiliki arti. Sebagai contoh, kata "*bat*" dan "*pat*" dibedakan hanya oleh fonem awal */b/* dan */p/*, yang menunjukkan bahwa */b/* dan */p/* adalah dua fonem yang berbeda dalam bahasa inggris [13].

2.5.2 Viseme

Visemes adalah representasi visual dari fonem unit terkecil dalam bahasa yang menghasilkan suara. Meskipun fonem mengacu pada suara yang diucapkan, *visemes* menggambarkan gerakan bibir dan bagian wajah lainnya saat fonem tersebut diucapkan. Beberapa fonem mungkin memiliki *viseme* yang serupa karena gerakan bibirnya hampir sama. Kata-kata seperti "*baby*," "*ball*," dan "*man*" melibatkan penutupan bibir atas dan bawah, sehingga digabungkan dalam satu *viseme* yang sama secara visual akan tampak serupa meskipun berbeda dalam suara [14].

2.6 Perhitungan Rata-rata

Rata-rata adalah ukuran statistik fundamental yang diperoleh dengan menjumlahkan seluruh nilai data kemudian membaginya dengan banyaknya data tersebut. Secara matematis, jika terdapat sampel amplitudo $X_1, X_2, ..., X_N$, rata-rata dihitung dengan Rumus 2.1 [15].

$$M = \frac{1}{N} \sum_{i=1}^{N} X_i \tag{2.1}$$

2.7 Konversi ke Desibel

Skala desibel (dB) adalah skala logaritmik yang umum digunakan untuk menyatakan intensitas sinyal audio. Dalam sistem audio, amplitudo atau tekanan suara dikonversi menjadi desibel untuk mencerminkan persepsi manusia yang bersifat logaritmik. Rumus 2.2 merupakan rumus untuk mendapatkan nilai desibel [16].

$$dB = 20 \cdot \log_{10}(\frac{M}{M_{ref}}) \tag{2.2}$$

M adalah nilai rata-rata, dan M_{ref} adalah nilai referensi. Perbedaan rasio amplitudo diubah menjadi perbedaan linier dalam satuan dB. Skala desibel sangat berguna karena dapat mengubah rentang dinamika sinyal audio yang besar menjadi skala yang lebih mudah dikelola secara komputasi dan lebih sesuai dengan persepsi pendengaran manusia [16].

2.8 Normalisasi

Normalisasi adalah proses mengubah rentang nilai data agar berada dalam skala tertentu yang konsisten, biasanya antara 0 dan 1 dengan Rumus 2.3 [17].

$$X_{\text{new}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \tag{2.3}$$

X adalah nilai asli, X_{min} adalah nilai terkecil dalam data, dan X_{max} adalah nilai terbesar. Tujuan normalisasi adalah untuk menyelaraskan skala data yang

bervariasi agar dapat diproses atau dibandingkan secara adil dan efisien. Salah satu metode yang umum digunakan adalah normalisasi *min-max*, yaitu teknik yang mengatur nilai terkecil menjadi 0 dan nilai terbesar menjadi 1, sementara nilai lainnya disesuaikan secara proporsional di antaranya [17]. Hasil normalisasi akan menjadi lebih seragam dan stabil, serta lebih mudah digunakan dalam berbagai aplikasi komputasi seperti pengendalian animasi, pemodelan statistik, atau pembelajaran mesin. Normalisasi juga membantu mencegah dominasi fitur tertentu akibat skala yang lebih besar, sehingga seluruh komponen data dapat berkontribusi secara seimbang dalam proses analisis atau visualisasi.

2.9 *Smoothing*

Langkah memperhalus (*smoothing*) digunakan jika ingin transisi nilai lama dengan nilai baru dengan mulus dan tidak berubah terlalu tajam antar dalam waktu singkat. Rumus 2.4 merupakan rumus *smoothing* [15].

$$N_{\text{smooth}} = (1 - \alpha) \cdot N_{\text{prev}} + \alpha \cdot N_{\text{new}} \tag{2.4}$$

N_{prev} adalah nilai fitur setelah *smoothing* sebelumnya, N_{new} adalah nilai fitur yang baru dihitung dan α adalah faktor pelunakan $(0 < \alpha < 1)$. Rumus ini memberikan bobot $(1 - \alpha)$ pada nilai lama dan α pada nilai baru, sehingga perubahan terjadi secara bertahap. Pendekatan ini sangat efektif untuk mengurangi fluktuasi mendadak pada volume atau amplitudo [15].

2.10 *System Usability Scale* (SUS)

System Usability Scale (SUS) adalah instrumen pengukuran kegunaan yang sederhana. Tujuan utama SUS adalah untuk mengukur persepsi pengguna terhadap kegunaan suatu sistem dengan cepat dan mudah. SUS memberikan nilai tunggal (0–100) yang menggambarkan tingkat kegunaan secara keseluruhan [9].

SUS berbentuk kuesioner dengan 10 pernyataan singkat yang berkaitan dengan pengalaman penggunaan sistem. Setiap pernyataan dijawab dengan skala Likert 5 poin, di mana 1 = Sangat tidak setuju dan 5 = Sangat setuju. Susunan ini dimaksudkan untuk mencegah responden memberi jawaban asal pada semua item. Dengan harus memperhatikan nada pernyataan, responden cenderung memikirkan setiap pertanyaan dengan seksama. Contoh beberapa item SUS [9]:

1. *I think that i would like to use this system.*
2. *I found the system unnecessarily complex.*
3. *I thought the system was easy to use.*
4. *I think that I would need the support of a technical person to be able to use this system.*
5. *I found the various functions in this system were well integrated.*
6. *I thought there was too much inconsistency in this system.*
7. *I would imagine that most people would learn to use this system very quickly.*
8. *I found the system very awkward to use.*
9. *I felt very confident using the system.*
10. *I needed to learn a lot of things before i could get going with this system.*

Skor akhir SUS dihitung dalam beberapa langkah sederhana. Setiap item ganjil (1, 3, 5, 7, 9), hitung kontribusi nilai sebagai (nilai jawaban – 1) nilai 1 menjadi 0 poin, nilai 5 menjadi 4 poin (rentang kontribusi 0–4). Setiap item genap (2, 4, 6, 8, 10), kontribusi nilai dihitung sebagai (5 – nilai jawaban). Contohnya, jawaban 5 pada item genap menghasilkan kontribusi 0, sedangkan jawaban 1 menghasilkan kontribusi 4 (rentang 0–4). Jumlah semua kontribusi nilai dari ke-10 item merupakan hasil total nilai mentah SUS (rentang 0–40 karena 10 item × 0–4). Hasil akhir SUS didapatkan dengan mengalikan total nilai mentah dengan 2,5 seperti pada Rumus 2.5. Operasi ini mengonversi rentang 0–40 menjadi skala 0–100 [18].

$$\text{SUS Score} = 2,5 * \sum(\text{item score}) \tag{2.5}$$

Nilai SUS bersifat relatif dan bersumber dari data empiris besar. Secara umum, nilai minimum SUS adalah 0 dan maksimum 100, nilai lebih tinggi menunjukkan kinerja lebih baik, dan nilai rendah menunjukkan perlunya perbaikan besar. Penggunaannya di berbagai studi telah menghasilkan konvensi praktis bahwa meningkatkan nilai SUS ke kisaran di atas 70–80 berarti mencapai pengalaman pengguna yang memuaskan [18]. Tabel 2.1 merupakan pengukuran model SUS dengan nilai kategori, rangka nilai, dan nilai persentase.

Tabel 2.1. Pengukuran dengan model SUS

Nilai kategori	Nilai SUS	Nilai persentase
A+	84.1 – 100	96 – 100
A	80.8 – 84.0	90 – 95
A-	78.9 – 80.7	85 – 89
B+	77.2 – 78.8	80 – 84
B	74.1 – 77.1	70 – 79
B-	72.6 – 74.0	65 – 69
C+	71.1 – 72.5	60 – 64
C	65.0 – 71.0	41 – 59
C-	62.7 – 64.9	35 – 40
D	51.7 – 62.6	15 – 34
F	0 – 51.6	0 – 14

2.11 Skala Likert

Skala Likert digunakan dalam instrumen angket untuk mengukur sikap, pendapat, atau persepsi responden terhadap suatu pernyataan sikap, opini, atau preferensi. Pada skala Likert, responden diminta menunjukkan tingkat persetujuan atau ketidaksetujuan menggunakan 5 tingkat pilihan jawaban, misalnya ”Sangat Setuju (SS), Setuju (S), Netral, Tidak Setuju (TS), Sangat Tidak Setuju (STS)”. Pemilihan jumlah poin tergantung pada kebutuhan penelitian, namun 5–7 poin merupakan konfigurasi yang paling umum [19]. Hasilnya berupa skor ordinal yang mencerminkan tingkat sikap atau opini individu. Dengan kata lain, skala Likert mengkuantifikasi data subjektif sehingga dapat dianalisis secara statistik [19].