

BAB II

TINJAUAN LITERATUR

2.1. Penelitian Terkait

Berikut adalah penelitian sebelumnya yang relevan sebagai acuan pengembangan:

1. Oyedotun dan Khashman (2017) dalam *Neural Computing and Applications* menunjukkan bahwa CNN mampu mengenali gesture tangan secara efektif melalui ekstraksi fitur spasial hierarkis dari citra tangan. Dalam studi tersebut, CNN dilatih pada dataset *gesture* tangan dan dibandingkan dengan stacked denoising autoencoder, menunjukkan tingkat kesalahan klasifikasi yang lebih rendah. Penelitian ini menjadi landasan utama penerapan CNN dalam sistem pengenalan gesture yang dikembangkan.
2. Zhang et al. (2020) dalam "*MediaPipe Hands*" mendemonstrasikan sistem deteksi landmark tangan *real-time* berbasis CNN yang mampu mendeteksi 21 titik keypoint dengan latensi rendah. Penelitian ini menjadi fondasi model *HandPose TensorFlow.js* yang digunakan secara langsung dalam penelitian ini sebagai implementasi CNN di *browser*.
3. Moryossef et al. (2020) mengusulkan model deteksi bahasa isyarat *real-time* berbasis fitur *optical flow* dari human pose estimation yang diproses dengan model recurrent, mencapai akurasi hingga 91% dengan waktu inferensi di bawah 4 milidetik per frame. Sistem ini mampu mengidentifikasi rangkaian gesture secara berkesinambungan dengan efisiensi komputasi tinggi, relevan sebagai acuan pipeline deteksi *real-time* berbasis landmark pada web.

4. Paul et al. (2024) mengembangkan sistem pengenalan bahasa isyarat ASL menggunakan kombinasi CNN berbasis ResNet dan LSTM. Model CNN mencapai akurasi 89,07% dalam mengklasifikasikan gesture statis, sedangkan model LSTM mencapai akurasi 94,3% dalam mengenali rangkaian gesture dinamis dan mengungguli model GRU sebagai pembanding. Penelitian ini menjadi acuan pendekatan hybrid CNN-LSTM untuk menangani aspek statis sekaligus temporal dari gesture tangan.
5. Norré (2020) mengevaluasi sistem prediksi kata berbasis model bahasa N-gram (trigram) pada aplikasi Augmentative and Alternative Communication (AAC) untuk penyandang gangguan komunikasi. Hasil pengujian menunjukkan model trigram mampu menghemat lebih dari separuh keystroke dibandingkan tanpa prediksi kata. Penelitian ini menjadi acuan penerapan model N-gram untuk menyusun prediksi kata pada sistem komunikasi berbantuan bagi penyandang disabilitas.

2.2. Landasan Teori

2.1. Isyarat Tangan (*Hand Gesture*)

Isyarat tangan adalah bentuk komunikasi nonverbal menggunakan gerakan, posisi, dan konfigurasi jari untuk menyampaikan makna tertentu. Bagi penyandang tunarungu dan tunawicara, isyarat tangan merupakan media komunikasi utama. Pengenalan otomatis oleh komputer memerlukan analisis matematis posisi relatif landmark tangan, yang dapat dilakukan secara efektif menggunakan algoritma CNN (Mahardika et al., 2024). Oyedotun et al. (2017) menjelaskan bahwa gesture tangan dapat dikenali secara otomatis oleh sistem komputer melalui analisis pola visual jari dan telapak tangan pada citra, yang kemudian diklasifikasikan ke dalam

kategori tertentu menggunakan pendekatan pembelajaran mendalam. Paul et al. (2024) menambahkan bahwa gesture dalam konteks bahasa isyarat terdiri atas komponen statis berupa bentuk tangan diam dan komponen dinamis berupa rangkaian gerakan, sehingga sistem pengenalan perlu mampu menangani kedua aspek tersebut secara bersamaan.

Berdasarkan definisi di atas, penulis dapat menyimpulkan bahwa isyarat tangan merupakan bentuk komunikasi nonverbal yang memanfaatkan gerakan, posisi, dan konfigurasi jari untuk menyampaikan makna tertentu, baik dalam bentuk gesture statis maupun dinamis. Pengenalan isyarat tangan secara otomatis oleh komputer dilakukan dengan menganalisis pola visual dan posisi relatif landmark tangan pada citra atau video, yang kemudian diklasifikasikan ke dalam label gesture tertentu menggunakan algoritma pembelajaran mendalam seperti CNN. Kemampuan sistem dalam mengenali kedua jenis gesture tersebut menjadi dasar penting bagi pengembangan sistem penerjemah bahasa isyarat berbasis teknologi bagi penyandang tunarungu dan tunawicara.

2.2. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah arsitektur deep learning yang dirancang untuk memproses data grid seperti gambar. CNN terdiri dari tiga jenis layer utama: *Convolutional Layer* (mengekstraksi fitur melalui filter/kernel), *Pooling Layer* (mengurangi dimensi spasial), dan *Fully Connected Layer* (klasifikasi akhir). Menurut LeCun et al. (2015), kemampuan CNN mengekstraksi fitur secara hierarkis dari tepi, tekstur, hingga pola kompleks menjadikannya sangat efektif untuk pengenalan objek visual termasuk gesture tangan. Oyedotun dan Khashman (2017) menambahkan bahwa CNN dirancang khusus untuk mengenali

pola visual pada citra melalui proses pembelajaran fitur secara otomatis, sehingga tidak lagi memerlukan ekstraksi fitur secara manual sebagaimana pada metode klasifikasi citra konvensional.

Dalam konteks pengenalan bahasa isyarat, CNN memproses gambar atau frame video tangan sebagai input, mengekstraksi fitur visual landmark jari melalui beberapa lapisan konvolusi, dan menghasilkan klasifikasi gesture sebagai output. Model *HandPose TensorFlow.js* menggunakan arsitektur CNN ringan (MobileNet-based) yang dioptimalkan untuk inferensi real-time di browser.

Berdasarkan uraian di atas, penulis dapat menyimpulkan bahwa *Convolutional Neural Network* (CNN) merupakan arsitektur deep learning yang dirancang khusus untuk memproses data visual melalui operasi konvolusi, pooling, dan fully connected layer, sehingga mampu mengekstraksi fitur secara hierarkis dan otomatis tanpa memerlukan rekayasa fitur manual. Kemampuan inilah yang menjadikan CNN sangat efektif digunakan dalam sistem pengenalan gesture tangan, termasuk pada model *HandPose TensorFlow.js* yang diterapkan dalam penelitian ini.

2.3. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) merupakan salah satu jenis arsitektur *Recurrent Neural Network* (RNN) yang dirancang untuk mengatasi permasalahan vanishing gradient pada pemodelan data sekuensial. LSTM mampu menyimpan informasi dalam jangka panjang melalui mekanisme gates yang terdiri dari input gate, forget gate, dan output gate, sehingga sangat efektif dalam menangani data yang memiliki dependensi temporal seperti teks, suara, dan gesture dinamis.

Menurut Paul et al. (2024), LSTM terbukti mampu mengenali pola sekuensial pada gesture bahasa isyarat dengan tingkat akurasi yang lebih tinggi dibandingkan model sekuensial lainnya seperti GRU. Selanjutnya, Yadav et al. (2021) menjelaskan bahwa LSTM memiliki kemampuan dalam mempertahankan informasi historis yang relevan melalui mekanisme memori internal, sehingga banyak digunakan dalam pemrosesan bahasa alami (*Natural Language Processing*). Selain itu, Reddy et al. (2022) menyatakan bahwa LSTM sangat efektif dalam pemodelan data time-series karena mampu menangkap hubungan jangka panjang antar data secara lebih stabil dibandingkan RNN konvensional.

2.4. N-Gram

N-Gram merupakan metode dalam pemodelan bahasa (*language modeling*) yang digunakan untuk memprediksi kata berikutnya berdasarkan urutan kata sebelumnya. Model ini bekerja dengan menghitung probabilitas kemunculan suatu kata berdasarkan n kata sebelumnya, seperti unigram (1 kata), bigram (2 kata), dan trigram (3 kata).

Menurut Norré (2020), model N-gram (khususnya trigram) mampu meningkatkan efisiensi pengetikan dengan mengurangi jumlah keystroke dalam sistem prediksi kata pada aplikasi komunikasi. Selanjutnya, Jurafsky dan Martin (2021) menjelaskan bahwa N-gram merupakan pendekatan statistik sederhana namun efektif dalam pemodelan bahasa, khususnya untuk prediksi kata pada sistem NLP. Selain itu, Khan et al. (2022) menyatakan bahwa model N-gram masih banyak digunakan karena kemudahannya dalam implementasi serta performanya yang cukup baik pada dataset berukuran kecil hingga menengah.

Dalam penelitian ini, metode N-gram digunakan untuk menyusun prediksi kata berdasarkan hasil pengenalan gesture, sehingga sistem dapat menghasilkan kalimat yang lebih natural dan mudah dipahami.

2.5. TensorFlow.js

TensorFlow.js adalah *library* machine learning *open source* berbasis JavaScript dari Google yang memungkinkan pelatihan dan inferensi model CNN langsung di *browser* menggunakan akselerasi GPU melalui WebGL. Menurut Google (2023b), TensorFlow.js mendukung penggunaan model pretrained CNN yang dapat langsung dijalankan tanpa *server backend*. Smilkov et al. (2019) menjelaskan bahwa TensorFlow.js dirancang agar model machine learning dapat dilatih dan dijalankan langsung di browser maupun Node.js dengan performa yang dipercepat melalui WebGL, sehingga memungkinkan komputasi on-device tanpa mengirim data pengguna ke *server*. Dalam penelitian ini, TensorFlow.js digunakan untuk menjalankan model CNN *HandPose* yang melakukan inferensi pada setiap frame video dari kamera browser secara *real-time*.

Berdasarkan definisi di atas, penulis dapat menyimpulkan bahwa TensorFlow.js merupakan library machine learning berbasis JavaScript yang memungkinkan model CNN dilatih maupun dijalankan langsung di sisi klien (browser) tanpa memerlukan server backend, dengan memanfaatkan akselerasi GPU melalui WebGL. Karakteristik inilah yang menjadikan TensorFlow.js dipilih sebagai fondasi teknologi dalam penelitian ini, karena mendukung inferensi model CNN *HandPose* secara *real-time* langsung di browser pengguna.

2.6. Model HandPose TensorFlow.js

HandPose adalah model CNN pretrained dalam ekosistem TensorFlow.js yang mendeteksi tangan dan menghasilkan estimasi 21 titik landmark (keypoint) secara real-time (Google, 2023a). Model ini dikembangkan berdasarkan arsitektur CNN MediaPipe Hands (Zhang et al., 2020) dengan 21 landmark: titik 0 (pergelangan), titik 1-4 (jempol), titik 5-8 (telunjuk), titik 9-12 (tengah), titik 13-16 (manis), titik 17-20 (kelingking). Setiap titik memiliki koordinat $[x, y, z]$ yang menjadi input untuk klasifikasi *gesture*.

Berdasarkan uraian di atas, penulis dapat menyimpulkan bahwa Model HandPose merupakan model CNN pretrained dalam ekosistem TensorFlow.js yang dikembangkan berdasarkan arsitektur *MediaPipe Hands*, mampu mendeteksi tangan dan menghasilkan estimasi 21 titik landmark secara *real-time*. Setiap titik landmark yang dihasilkan memiliki koordinat tiga dimensi $[x, y, z]$ yang menjadi dasar bagi proses klasifikasi *gesture* pada tahap selanjutnya.

2.7. Klasifikasi Gesture Berbasis Landmark CNN

Setelah CNN *HandPose* menghasilkan 21 landmark, klasifikasi *gesture* dilakukan dengan *rule-based classification*: membandingkan posisi ujung jari (tip) dengan sendi tengah (PIP/MCP). Jari dinyatakan terbuka (1) jika tip lebih tinggi dari PIP (Y lebih kecil), tertutup (0) jika sebaliknya. Jempol menggunakan sumbu X karena orientasinya berbeda. Pola biner 5-digit [Jempol, Telunjuk, Tengah, Manis, Kelingking] dipetakan ke label *gesture* melalui *lookup table*.

Zhang et al. (2020) menjelaskan bahwa tahap klasifikasi pada sistem berbasis landmark umumnya memanfaatkan posisi relatif antar titik keypoint hasil deteksi CNN untuk menentukan konfigurasi tangan, tanpa memerlukan pelatihan ulang terhadap model deteksi utama. Paul et al. (2024) menambahkan bahwa

pendekatan berbasis aturan (*rule-based*) terhadap hasil ekstraksi fitur CNN dapat digunakan sebagai lapisan klasifikasi tambahan yang ringan secara komputasi untuk memetakan pola gesture statis ke label tertentu.

Berdasarkan definisi di atas, penulis dapat menyimpulkan bahwa klasifikasi gesture berbasis landmark CNN merupakan proses lanjutan setelah deteksi landmark tangan, di mana posisi relatif antar titik keypoint dianalisis menggunakan pendekatan *rule-based* untuk menentukan status terbuka atau tertutupnya setiap jari. Pendekatan ini dipilih karena bersifat ringan secara komputasi dan tidak memerlukan pelatihan ulang model CNN, sehingga sesuai diterapkan pada sistem yang berjalan secara *real-time* di *browser*.

2.8. Web Speech API

Web Speech API adalah API bawaan browser modern yang menyediakan sintesis suara (*Text-to-Speech*) melalui komponen *SpeechSynthesis*. Dalam penelitian ini, label gesture yang dihasilkan CNN dikonversi menjadi suara berbahasa Indonesia (id-ID) menggunakan *SpeechSynthesisUtterance*. Mekanisme *cooldown* 2.500ms diterapkan untuk mencegah pengulangan suara (W3C, 2023). Handayani et al. (2025) menegaskan bahwa teknologi berbasis suara pada sistem web *real-time* terbukti mampu meningkatkan aksesibilitas informasi bagi komunitas tunarungu, sehingga pemilihan output audio melalui *Web Speech* API relevan sebagai media komunikasi bagi pengguna yang tidak memahami bahasa isyarat.

Berdasarkan definisi di atas, penulis dapat menyimpulkan bahwa *Web Speech* API merupakan antarmuka bawaan browser modern yang memungkinkan konversi teks menjadi suara (*Text-to-Speech*) secara langsung tanpa memerlukan

layanan pihak ketiga. Pemanfaatan komponen *SpeechSynthesis* pada penelitian ini memungkinkan label gesture yang dihasilkan CNN dikonversi menjadi keluaran suara berbahasa Indonesia, sehingga berfungsi sebagai media komunikasi yang menjembatani penyandang tunarungu/tunawicara dengan masyarakat umum.

2.9. WebRTC dan Akses Kamera Browser

WebRTC memungkinkan akses kamera browser melalui *navigator.mediaDevices.getUserMedia()*. Data video tidak dikirim ke *server*, seluruh inferensi CNN berjalan di sisi klien sehingga menjamin privasi pengguna. Akses memerlukan HTTPS atau *localhost* (W3C, 2022). Mozilla (2026) menjelaskan bahwa WebRTC merupakan teknologi yang memungkinkan aplikasi web menangkap dan mempertukarkan data audio maupun video secara langsung antar browser tanpa perantara *server*, sehingga mendukung pemrosesan media secara *real-time* di sisi klien.

Berdasarkan definisi di atas, penulis dapat menyimpulkan bahwa WebRTC merupakan teknologi berbasis *browser* yang memungkinkan akses langsung ke perangkat kamera pengguna serta pemrosesan data video secara *real-time* di sisi klien, tanpa perlu mengirimkan data mentah ke *server*. Karakteristik ini sangat mendukung penerapan sistem pengenalan gesture berbasis CNN dalam penelitian ini, karena seluruh proses inferensi dapat dilakukan langsung pada perangkat pengguna sehingga privasi data video tetap terjaga.

2.10. Canvas API

Menurut Mozilla (2026), Canvas API menyediakan sarana untuk menggambar grafis melalui JavaScript pada elemen HTML `<canvas>`, yang dapat dimanfaatkan untuk animasi, visualisasi data, maupun pemrosesan video secara

real-time, dengan fokus utama pada grafis dua dimensi. Canvas API HTML5 memungkinkan penggambaran grafis 2D secara programatik. Dalam penelitian ini, Canvas digunakan untuk menggambar overlay skeleton 21 landmark CNN HandPose di atas video kamera, memberikan visualisasi real-time berupa garis hijau koneksi antar landmark dan titik-titik berwarna.

Berdasarkan definisi di atas, penulis dapat menyimpulkan bahwa Canvas API merupakan antarmuka pemrograman HTML5 yang memungkinkan penggambaran grafis dua dimensi secara dinamis melalui JavaScript. Dalam penelitian ini, kemampuan tersebut dimanfaatkan untuk menampilkan visualisasi real-time berupa overlay skeleton landmark tangan di atas video kamera, sehingga pengguna dapat melihat secara langsung titik-titik dan koneksi antar landmark yang dideteksi oleh model CNN *HandPose*.

2.11. Aksesibilitas *Web*

Aksesibilitas web mengacu pada pengembangan yang dapat digunakan semua orang termasuk penyandang disabilitas. W3C WCAG 2.1 menetapkan empat prinsip: *Perceivable*, *Operable*, *Understandable*, *Robust*. Sistem CNN pengenalan isyarat ini merupakan implementasi aksesibilitas yang mengkonversi informasi visual (*gesture*) ke audio (suara), membantu komunikasi tunarungu/tunawicara (W3C, 2018). *Rodríguez-Correa et al. (2023)* dalam kajian sistematis terhadap teknologi asistif bagi penyandang tunarungu menyatakan bahwa pengembangan teknologi yang mengonversi satu modalitas komunikasi ke modalitas lain, seperti dari visual ke audio maupun sebaliknya, terbukti berperan penting dalam meningkatkan aksesibilitas dan kemandirian komunikasi penyandang disabilitas pendengaran.

Berdasarkan definisi di atas, penulis dapat menyimpulkan bahwa aksesibilitas web merupakan prinsip pengembangan yang memastikan sistem dapat digunakan oleh semua kalangan pengguna, termasuk penyandang disabilitas, dengan berpedoman pada prinsip *Perceivable*, *Operable*, *Understandable*, dan *Robust* sebagaimana ditetapkan dalam WCAG 2.1. Sistem pengenalan bahasa isyarat berbasis CNN yang dikembangkan dalam penelitian ini merupakan salah satu bentuk implementasi aksesibilitas, karena mengonversi informasi visual berupa gesture tangan menjadi keluaran audio, sehingga membantu menjembatani komunikasi antara penyandang tunarungu/tunawicara dengan masyarakat umum.

2.12. Metode *Waterfall*

Metode *Waterfall* adalah model pengembangan perangkat lunak sekuensial dan linier. Pressman dan Maxim (2021) mendefinisikan 5 tahap: *Communication* (analisis kebutuhan), *Planning*, *Modeling* (perancangan), *Construction* (implementasi & pengujian), *Deployment*. Sommerville (2022) menambahkan bahwa model sekuensial ini paling sesuai diterapkan ketika kebutuhan sistem dapat didefinisikan secara lengkap dan stabil sejak awal proyek. Metode ini dipilih karena spesifikasi sistem CNN (8 *gesture*, *output* suara, *platform web*) telah terdefinisi jelas sejak awal penelitian.

Berdasarkan uraian di atas, penulis dapat menyimpulkan bahwa metode *Waterfall* merupakan model pengembangan perangkat lunak yang bersifat sekuensial dan linier, dengan tahapan yang berjalan secara berurutan mulai dari analisis kebutuhan hingga pemeliharaan sistem. Metode ini paling sesuai diterapkan pada proyek yang kebutuhan sistemnya telah terdefinisi secara lengkap dan stabil

sejak awal, sebagaimana kondisi pada penelitian ini di mana spesifikasi sistem CNN, jumlah gesture, dan output suara telah ditentukan sejak tahap perencanaan.

2.13. Black Box Testing

Black Box Testing adalah metode pengujian yang memverifikasi fungsionalitas sistem berdasarkan spesifikasi tanpa memperhatikan kode internal. Myers et al. (2022) menyatakan *Black Box Testing* efektif mendeteksi kesalahan fungsional, antarmuka, dan kinerja. Cholifah et al. (2018) dalam pengujian aplikasi berbasis *web* dan Android membuktikan bahwa *Black Box Testing* mampu memastikan seluruh fungsi sistem berjalan sesuai kebutuhan pengguna tanpa ditemukan kesalahan pada interface, sehingga metode ini dinilai andal untuk menguji fungsionalitas sistem berbasis *web*. Dalam penelitian ini digunakan untuk menguji 15 skenario uji meliputi seluruh fitur sistem CNN.

Berdasarkan definisi di atas, penulis dapat menyimpulkan bahwa *Black Box Testing* merupakan metode pengujian perangkat lunak yang berfokus pada verifikasi fungsionalitas sistem berdasarkan spesifikasi kebutuhan, tanpa memperhatikan struktur kode program di dalamnya. Metode ini dinilai efektif untuk mendeteksi kesalahan fungsional, antarmuka, maupun kinerja sistem, sehingga sesuai digunakan untuk menguji seluruh skenario fungsional pada sistem pengenalan bahasa isyarat berbasis CNN dalam penelitian ini.