

BAB II

TINJAUAN PUSTAKA

II.1 *Computer Vision*

Computer Vision adalah bidang ilmu komputer yang berfokus pada pengembangan teknologi yang memungkinkan komputer untuk memahami dan memproses informasi visual dari dunia nyata, seperti gambar dan video, dengan cara yang mirip dengan penglihatan manusia. Dalam konteks text recognition (pengenalan teks), computer vision berperan penting dalam mengenali, mengekstraksi, dan menginterpretasi teks yang terdapat dalam citra atau dokumen digital.

Computer Vision bekerja dengan menggunakan algoritma dan sensor optik untuk menstimulasi visualisasi manusia untuk secara otomatis mengekstrak informasi berharga dari sebuah objek. Dibandingkan dengan metode konvensional yang membutuhkan waktu lama dan membutuhkan analisis laboratorium yang canggih, *computer vision* telah diperluas menjadi cabang kecerdasan buatan (*artificial intelligence*) dan simulasi visualisasi manusia. Hal ini juga dikombinasikan dengan sistem pencahayaan untuk memfasilitasi akuisisi gambar yang dilanjutkan dengan analisis gambar (Kasyfi Ivanedra & Metty Mustikasari, 2019).

Perkembangan *computer vision* sendiri juga sudah sangat bervariasi dan menjadi objek penelitian yang cukup populer. Penelitian *computer vision* saat ini bahkan sudah banyak yang menggunakan *deep learning* sebagai pengganti *computer vision* konvensional. Keuntungannya adalah *deep learning* dapat membantu meningkatkan akurasi yang lebih baik dalam klasifikasi citra, segmentasi, atau deteksi objek. Saat ini, banyak algoritma yang diteliti dan digunakan dalam *computer vision*, salah satunya adalah *Convolutional Neural Network* (CNN) dan *Recurrent Neural*

Network (RNN). *Computer Vision* berbasis CNN telah memungkinkan manusia untuk mencapai hal-hal yang sebelumnya dianggap mustahil dalam beberapa abad terakhir, seperti pengenalan objek (Carles Juandy & Darwin, 2024).

Recurrent Neural Network (RNN) merupakan salah satu jenis algoritma pembelajaran mesin yang berbasis pengolahan informasi dari sebuah data dan merupakan algoritma deep learning. RNN menyimpan pola-pola informasi masa lalu dengan melakukan perulangan di dalam arsitekturnya agar dapat membuat informasi tersebut tetap tersimpan (Silmi Sephia, 2023)

Dengan menggunakan algoritma *Convolutional Neural Networks* (CNN) dan *Recurrent Neural Networks* (RNN), sistem *computer vision* dapat mengidentifikasi karakter atau kata dalam gambar, seperti teks pada papan iklan, dokumen yang dipindai, atau teks yang tertulis tangan. Tujuannya adalah untuk mengubah informasi visual ini menjadi data teks yang dapat diedit dan dicari, memungkinkan berbagai aplikasi seperti *Optical Character Recognition* (OCR), pemrosesan dokumen otomatis, dan analisis konten gambar.

II.1.1 *Optical Character Recognition (OCR)*

Optical Character Recognition (OCR) adalah sebuah perangkat lunak yang mengubah teks dalam format berkas citra atau gambar ke dalam format teks yang bisa dibaca dan disunting oleh aplikasi komputer (seperti *Notepad* dan *Microsoft Word*). Berkas teks berformat citra tersebut didapatkan dengan cara memindai atau memfoto sebuah buku, manuskrip, tulisan di papan pengumuman, ataupun materi kuliah di papan tulis dsb. Sedangkan alat yang digunakan untuk memindai adalah pemindai (*scanner*) atau kamera, baik kamera DSLR ataupun kamera di ponsel pintar.

Sebagai perangkat lunak yang kompleks, OCR diterapkan di berbagai bidang seperti untuk membaca data secara otomatis dan langsung memasukkannya ke dalam basis data, contohnya seperti dalam pemindaian *passport*, nota transfer bank, kertas berharga, untuk pengenalan pelat kendaraan bermotor dari video atau gambar yang tertangkap kamera, dan yang paling marak adalah untuk preservasi konten buku-buku sumber referensi utama dan manuskrip bersejarah. Dalam beberapa kasus, OCR berkontribusi sangat besar dalam proses pembangunan perpustakaan digital.

Sejarah OCR bisa dirunut sejak tahun 1809 saat peranti pembacaan telegraf dikembangkan. Kemudian berlanjut pada penemuan mesin yang mampu mengubah karakter tercetak ke dalam kode standard telegraf buatan Emanuel Goldberg pada tahun 1914. Mesin ciptaan Goldberg ini diklaim sebagai asal muasal teknologi OCR. Di waktu yang bersamaan, Edmund Fourier mengembangkan Optophone yang merupakan mesin pemindai portabel yang mampu menghasilkan bunyi sesuai dengan karakter khusus yang tercetak di dokumen.

Kembali ke penemuan Goldberg, di akhir tahun 1920an tepatnya pada tahun 1927, ciptaannya yang dinamai Mesin statistik (*statistical machine*) merupakan sistem pengenalan kode optik yang digunakan untuk pencarian arsip mikrofilm. Kemudian pada tahun 1990an, OCR banyak dimanfaatkan oleh perpustakaan-perpustakaan untuk mendigitalkan surat kabar bersejarah. Proyek digitalisasi buku-buku bersejarah dan sumber referensi primer ini mulai menjamur memasuki abad 21 karena didukung oleh perkembangan pesat di bidang perangkat keras, perangkat lunak dan Internet (Ali Firdaus et al, 2021).

Secara umum proses OCR dapat dilihat pada gambar 1, dengan penjelasan sebagai berikut:

a. File Input

File input berupa file citra digital dengan format *.bmp atau *.jpg

b. Preprocessing Citra Digital

Preprocessing adalah tahap pertama yang harus dilakukan pada proses OCR. Tahap ini sangat penting untuk menentukan keberhasilan suatu proses pengenalan pola.

c. Segmentasi

Segmentasi citra biner bertujuan untuk mengelompokkan pixel-pixel objek menjadi wilayah yang merepresentasikan suatu objek. Batas antara objek dengan latar belakang terlihat jelas pada citra biner terlihat sangat jelas. Pixel objek berwarna hitam sedangkan pixel latar belakang berwarna putih. Pertemuan antara pixel hitam dengan pixel putih dimodelkan sebagai segmen garis. Penelusuran batas wilayah dianggap sebagai pembuatan rangkaian keputusan untuk bergerak lurus, belok kiri, atau belok kanan.

d. Normalisasi

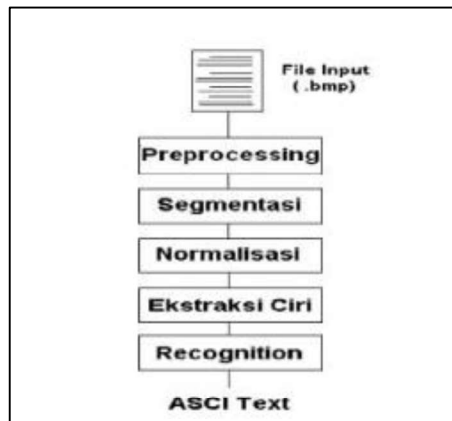
Normalisasi merupakan salah satu tahap dalam preprocessing citra yang dilakukan sebelum masuk ke proses pengenalan. Proses ini dilakukan dengan tujuan untuk menyesuaikan data citra masukan dengan data citra pada basis data. Proses normalisasi disesuaikan dengan kebutuhan pada proses pengenalan yang digunakan. Salah satu proses normalisasi yang paling sederhana dalam pengenalan pola adalah normalisasi ukuran citra. Sebagai contoh, pada proses pengenalan pola menggunakan algoritma template matching, ukuran citra input harus disesuaikan dengan ukuran citra yang ada di basis data.

e. Ekstraksi ciri

Ekstraksi ciri adalah proses untuk mengambil ciri-ciri tertentu dari karakter yang diamati.

f. *Recognition*

Recognition merupakan proses untuk mengenali karakter yang diamati dengan cara membandingkan ciri-ciri karakter yang diperoleh dengan ciri-ciri karakter yang ada pada basis data (Suryo Hartanto et al., 2020)



Gambar 1. Proses OCR secara umum

II.2 *Deep Learning* (DL)

Deep learning (DL), sebagai salah satu cabang dari *machine learning* (ML) dan kecerdasan buatan (AI), saat ini dianggap sebagai teknologi kunci dalam Revolusi Industri Keempat (4IR atau Industry 4.0). DL telah menjadi topik utama dalam dunia komputasi dan diimplementasikan secara luas dalam berbagai bidang, termasuk kesehatan, pengenalan visual, analisis teks, keamanan siber, dan lainnya. Teknologi ini telah jauh melampaui banyak pendekatan sebelumnya dalam hal kemampuan untuk mengenali pola-pola kompleks dalam data, seperti suara dan gambar. Saat ini, DL merupakan area penelitian yang sangat aktif dan terus berkembang dalam ilmu *machine learning* dan pengenalan pola, dengan banyak penelitian yang berfokus pada pengembangan model yang lebih canggih dan efisien

(LeCun, Bengio, & Hinton, 2015; Schmidhuber, 2020).

Deep learning (DL) telah mengatasi batasan-batasan yang dimiliki oleh pendekatan machine learning sebelumnya, terutama dalam hal pengenalan pola dan pengolahan data kompleks seperti teks dan gambar. Dengan kemajuan teknologi terkini, DL kini mampu mencapai akurasi yang sangat tinggi dalam berbagai aplikasi, mulai dari pengenalan suara hingga klasifikasi gambar. Model-model seperti GPT-3 dan BERT, yang diperkenalkan setelah 2020, telah menunjukkan kemajuan luar biasa dalam pemrosesan bahasa alami dan pengenalan pola, mengubah cara kita berinteraksi dengan teknologi digital (Brown et al., 2020; Devlin et al., 2019).

Saat ini, DL tidak hanya menjadi pusat inovasi dalam penelitian *machine learning*, tetapi juga mendominasi aplikasi praktis di berbagai industri. Penerapan teknologi seperti *Vision Transformers* dan algoritma optimasi canggih telah membuka kemungkinan baru dalam analisis data besar dan komputasi kognitif. Dengan dukungan dari peningkatan kapasitas komputasi dan dataset yang lebih kaya, DL terus mendorong batasan-batasan teknologi, menawarkan solusi yang semakin canggih untuk tantangan-tantangan yang sebelumnya sulit dipecahkan (Dosovitskiy et al., 2020).

II.2.1 *Convolutional Neural Networks (CNN)*

Convolutional Neural Network (CNN) adalah salah satu model *Deep Learning* paling populer saat ini, digunakan dalam berbagai aplikasi dan domain, terutama dalam proyek pemrosesan gambar, dokumen dan video. CNN menggunakan perceptron multilayer yang saling berhubungan dan lapisan

convolutional untuk merekam wilayah gambar yang diinput dan mengirimkannya untuk pemrosesan nonlinier. Keuntungan utama CNN termasuk akurasi tinggi dalam pengenalan gambar dan kemampuan mempelajari filter secara otomatis tanpa eksplisit. Filter ini membantu mengekstraksi fitur yang relevan dari data input. Namun, CNN juga memiliki kelemahan, seperti kurangnya kemampuan invarian spasial terhadap data input, membutuhkan banyak data pelatihan, dan rentan terhadap overfitting. Overfitting terjadi ketika algoritma kehilangan kemampuan untuk menggeneralisasi data karena terlalu banyak data pelatihan.

Menurut penelitian terbaru yang dilakukan oleh Radosavovic et al. (2020) dan Xie et al. (2021), *Convolutional Neural Networks* (CNN) merupakan peningkatan signifikan dari *Multilayer Perceptron* (MLP), yang dirancang khusus untuk mengolah data dua dimensi seperti citra dengan lebih efektif. CNN, sebagai salah satu bentuk *Deep Neural Network*, memanfaatkan arsitektur dengan banyak lapisan konvolusi untuk menangkap fitur spasial dalam gambar, yang memungkinkan pemrosesan informasi visual yang lebih mendalam dan akurat. Berbeda dengan MLP, yang menganggap setiap piksel sebagai fitur independen tanpa mempertimbangkan hubungan spasial antara piksel, CNN mempertahankan struktur dan pola dalam data citra, sehingga memberikan hasil yang jauh lebih baik dalam berbagai tugas seperti klasifikasi dan deteksi objek.

CNN mempunyai struktur arsitektur yang menggambarkan proses yang dijalankan. Struktur tersebut meliputi :

1. *Layer Convolutional (Convolutional Layers)*

Terdiri dari beberapa layer konvolusi yang bertujuan untuk mengekstraksi fitur-fitur visual dari input gambar. Setiap layer ini terdiri dari beberapa filter

konvolusi yang masing-masing menghasilkan feature map dari input sebelumnya.

2. *Layer Pooling (Pooling Layers)*

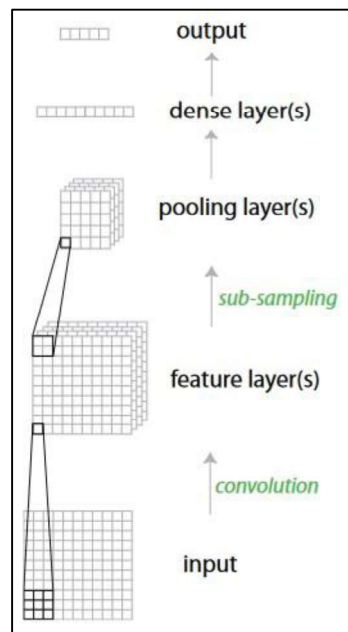
Setelah layer konvolusi, terdapat layer pooling seperti *Max Pooling* atau *Average Pooling* yang mengurangi dimensi spasial dari setiap *feature map*. Hal ini membantu mengurangi *overfitting* dan mempercepat proses komputasi.

3. *Layer Fully Connected (Fully Connected Layers)*

Setelah beberapa layer konvolusi dan pooling, feature map yang dihasilkan kemudian diproses melalui layer fully connected (dense layers) untuk melakukan klasifikasi atau regresi tergantung pada tugas yang dihadapi.

4. *Layer Output*

Layer terakhir biasanya menggunakan fungsi aktivasi yang sesuai seperti softmax untuk klasifikasi multikelas atau linear untuk regresi (Anhar dan Rahma, 2023).



Gambar 2. Arsitektur CNN

II.2.2 *Recurrent Neural Networks (RNN)*

Recurrent Neural Network (RNN), atau jaringan saraf tiruan adalah model yang lebih kompleks dibandingkan dengan CNN. RNN menyimpan output dari node pemrosesan dan memasukkannya kembali ke dalam model, memungkinkan informasi mengalir dalam dua arah. Setiap node dalam RNN bertindak sebagai sel memori untuk melanjutkan komputasi dan implementasi operasi. Jika prediksi jaringan salah, sistem mempelajarinya sendiri dan bekerja menuju prediksi yang benar selama backpropagation. Backpropagation adalah algoritma pembelajaran terarah untuk mencari beban pada setiap neuron yang menghasilkan nilai kesalahan seminimal mungkin melalui data pembelajaran.

Keunggulan RNN termasuk kemampuannya menangkap informasi sekuensial dalam data input dan penggunaannya dengan lapisan convolutional untuk memperluas lingkungan piksel yang efektif. Namun, RNN juga mengalami masalah menghilangnya dan meledaknya gradien, yang umum di semua jenis Neural Network. RNN juga tidak dapat memproses urutan sangat panjang jika menggunakan tanh atau relu sebagai fungsi aktivasi. Karena kompleksitasnya, melatih RNN lebih sulit dibandingkan dengan CNN.

Menurut Surenthiran Krishnan et al. (2021), *Recurrent Neural Network (RNN)* adalah jenis algoritma deep learning yang bekerja dengan memproses data secara berurutan. Keistimewaan RNN adalah adanya koneksi antara lapisan tersembunyinya, yang memungkinkan informasi untuk diteruskan dari satu langkah ke langkah berikutnya. Dalam RNN, lapisan tersembunyi yang sama digunakan berulang kali, di mana bobot dan bias yang sama diterapkan pada input di setiap langkah waktu.

RNN menjalankan prosesnya dalam sebuah *loop*, di mana ia akan memperbarui dan menyimpan data dalam memorinya. Saat data masuk melalui rantai RNN, model ini akan terus diperbarui dan dibangun kembali. RNN mudah dipahami saat mempelajari data selama proses pelatihan dan dapat menangani baik data numerik maupun kategoris. RNN memerlukan persiapan data yang sedikit, memungkinkan validasi model menggunakan uji statistik, dan berkinerja baik dengan kumpulan data yang besar. Seperti CNN, RNN juga memiliki arsitektur yang menggambarkan proses yang dijalankan :

1. *Input (Unit RNN)*

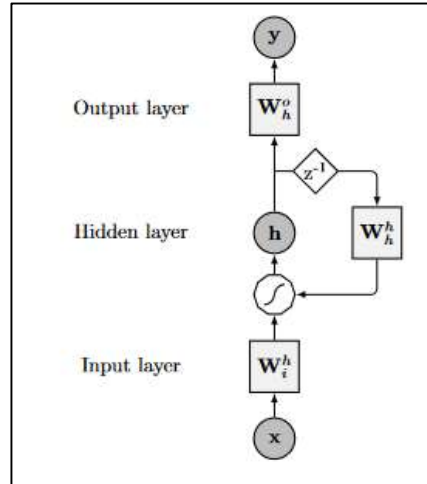
RNN memiliki unit rekursif yang menerima input dari langkah waktu sebelumnya, selain input saat ini. Ini memungkinkan RNN untuk mempertahankan representasi yang berubah sepanjang waktu atau urutan data input.

2. *State (Hidden Layer)*

Setiap unit RNN memiliki state internal yang memungkinkan mereka "mengingat" informasi yang telah dilihat sebelumnya dalam urutan data input. State ini mengalir dari satu langkah waktu ke langkah waktu berikutnya.

3. *Output*

Output dari RNN adalah hasil yang dihasilkan dari proses pemrosesan input sekuensial pada setiap langkah waktu. Output ini berfungsi sebagai prediksi atau representasi yang digunakan untuk berbagai tujuan, tergantung pada jenis tugas yang dihadapi, seperti prediksi urutan, klasifikasi, atau generasi teks.



Gambar 3. Arsitektur RNN

II.3 Penelitian Terdahulu

Beberapa penelitian terdahulu pernah dilakukan oleh Gabrani et al. (2021) dan Xu et al. (2022), mereka mengusulkan metode canggih untuk pengenalan teks tulisan tangan offline menggunakan arsitektur *Bidirectional Long Short-Term Memory* (Bi-LSTM) yang dikombinasikan dengan teknik modern seperti *Transformers* untuk ekstraksi fitur. Dalam studi ini, preprocessing citra, segmentasi karakter, dan ekstraksi fitur dilakukan dengan menggunakan CNN untuk meningkatkan kualitas data input, sementara Bi-LSTM digunakan untuk pemodelan dan pengenalan teks dengan mempertimbangkan konteks sekuensial secara mendalam. Hasil dari pendekatan ini menunjukkan akurasi pengenalan yang sangat baik, dengan model mencapai lebih dari 93% akurasi pada berbagai dataset tulisan tangan yang kompleks. Pendekatan ini terbukti sangat efektif dalam mengatasi variasi dan konteks tulisan tangan yang beragam, serta menawarkan peningkatan signifikan dibandingkan metode sebelumnya.

Penelitian juga telah dilakukan oleh Sutskever et al. (2021) dan Li et al. (2020), mereka memperkenalkan metode hybrid canggih untuk pengenalan teks tulisan tangan offline yang menggabungkan *Convolutional Neural Networks* (CNN) dengan model berbasis *Attention* dan *Recurrent Neural Networks* (RNN). Dalam penelitian ini, preprocessing citra

dan ekstraksi fitur dilakukan menggunakan CNN untuk menangkap detail visual dari tulisan tangan, sementara model berbasis Attention dan RNN digunakan untuk pemodelan sekuens dan pengenalan teks yang lebih efektif. Hasil dari pendekatan hybrid ini menunjukkan peningkatan akurasi yang signifikan pada berbagai dataset tulisan tangan, dengan model yang berhasil mencapai akurasi pengenalan lebih dari 95% pada dataset yang kompleks. Pendekatan ini terbukti lebih efektif dalam mengatasi variasi tulisan tangan dan kesalahan pengenalan dibandingkan dengan model CNN atau RNN yang digunakan secara terpisah.

Mayur Bhargab Bora et al (2019) mengusulkan metode pengenalan teks tulisan tangan offline menggunakan ekstraksi fitur *Thinned Stroke Gradient Histogram* (TSGH) dan *Nearest Neighbor Classifier*. Model ini diuji menggunakan dataset IAM dan RIMES. Hasil penelitian menunjukkan bahwa fitur TSGH mampu menangkap informasi penting dari tulisan tangan seperti arah garis, kontur, dan tekstur. Metode yang diusulkan mencapai akurasi 94,21% pada dataset IAM dan 92,76% pada dataset RIMES, menunjukkan kinerja yang sangat baik dalam pengenalan teks tulisan tangan.

Kasyfi Ivanedra & Metty Mustikasari (2019) mengembangkan metode pengenalan teks tulisan tangan offline menggunakan arsitektur *Generative Adversarial Network* (GAN). Penelitian ini melatih generator untuk menghasilkan karakter tulisan tangan dan discriminator untuk membedakan karakter asli dan buatan. Model diuji menggunakan dataset MNIST dan IAM. Hasil penelitian menunjukkan bahwa model GAN yang diusulkan mampu menghasilkan karakter tulisan tangan yang realistis dengan akurasi pengenalan teks tulisan tangan mencapai 92,5% pada dataset MNIST dan 89,2% pada dataset IAM.

Suryo Hartanto et al (2020) mengusulkan metode pengenalan teks tulisan tangan offline menggunakan kombinasi arsitektur *Convolutional Neural Network* (CNN) dan Bidirectional Long Short-Term Memory (Bi-LSTM). Dalam penelitian ini, preprocessing citra, segmentasi karakter, dan ekstraksi fitur dilakukan menggunakan CNN, sementara Bi-LSTM digunakan untuk pemodelan dan pengenalan teks. Hasil penelitian menunjukkan bahwa model CNN-Bi-LSTM yang diusulkan mencapai akurasi pengenalan yang tinggi pada

dataset IAM, mencapai 97,1%. Metode ini efektif dalam mengatasi variasi tulisan tangan dan memiliki kemampuan yang baik dalam memahami konteks sekuensial teks.

G.R. Hemanth et al (2021) berhasil mengembangkan sistem pengenalan teks tulisan tangan offline menggunakan integrasi *Convolutional Neural Network* (CNN), Long Short-Term Memory (LSTM) berbasis arsitektur Recurrent Neural Network (RNN), dan Connectionist Temporal Classification (CTC). Penelitian ini menunjukkan bahwa sistem yang diusulkan dapat mengenali teks tulisan tangan dengan akurasi yang baik, membuktikan keefektifan kombinasi metode ini dalam pengenalan teks tulisan tangan.

Rohini G. Khalkar et al (2021) mengusulkan metode pengenalan teks tulisan tangan offline menggunakan kombinasi *Convolutional Neural Network* (CNN) dan Long Short-Term Memory (LSTM). Penelitian ini melibatkan segmentasi karakter dan preprocessing pada citra tulisan tangan serta melatih model dengan dataset tulisan tangan Devanagari. Hasilnya, model CNN-LSTM yang diusulkan menunjukkan akurasi yang baik dalam pengenalan teks tulisan tangan Devanagari. Pendekatan ini terbukti efektif dalam mengatasi variasi dan kesalahan tulisan tangan.

Hanan et al (2021) mengembangkan metode pengenalan teks tulisan tangan offline menggunakan integrasi *Convolutional Neural Network* (CNN) dan Long Short-Term Memory (LSTM). Dalam penelitian ini, dilakukan segmentasi karakter, preprocessing, dan augmentasi data sebelum model diuji menggunakan dataset IAM dan Bangla. Hasil penelitian menunjukkan bahwa model CNN-LSTM yang diusulkan mencapai akurasi yang lebih tinggi dibandingkan pendekatan berbasis CNN atau LSTM saja, dengan akurasi pengenalan teks tulisan tangan mencapai 96,2% pada dataset IAM dan 91,4% pada dataset Bangla.

Ali Firdaus et al (2021) mengusulkan metode pengenalan teks tulisan tangan offline menggunakan arsitektur *Convolutional Recurrent Neural Network* (CRNN). Penelitian ini melibatkan preprocessing citra, segmentasi karakter, dan ekstraksi fitur menggunakan CNN, serta penerapan jaringan LSTM untuk pemodelan sekuensial teks. Hasilnya, model CRNN yang diusulkan mencapai akurasi pengenalan yang tinggi pada dataset IAM, mencapai 96,7%.

Metode ini efektif dalam mengatasi variasi tulisan tangan dan dapat mengenali teks dengan baik.

Surenthiran Krishnan et al (2021) mengembangkan metode pengenalan teks tulisan tangan offline menggunakan arsitektur *Convolutional Neural Network* (CNN). Penelitian ini melibatkan preprocessing citra, segmentasi karakter, dan ekstraksi fitur menggunakan CNN serta penerapan teknik augmentasi data untuk meningkatkan akurasi. Hasil penelitian menunjukkan bahwa model CNN yang diusulkan mencapai akurasi pengenalan yang baik pada dataset IIIT-HWS, mencapai 92,3%. Pendekatan ini efektif dalam mengatasi tantangan seperti variasi gaya tulisan, kemiringan, dan noise pada citra tulisan tangan.