

Optimalisasi Model Jaringan Syaraf Untuk Pengenalan CAPTCHA dengan Metode LeNet-5

Rocky Putra A¹, Rudi Kurniawan^{2,*}, Tri Hasanah Bimastari Aviani¹

¹ Fakultas Ilmu Teknik, Program Studi Informatika, Universitas Bina Insan, Lubuklinggau, Indonesia

² Fakultas Ilmu Teknik, Program Studi Rekayasa Sistem Komputer, Universitas Bina Insan, Kota Lubuklinggau, Indonesia

Email: ¹rockypa788@gmail.com, ^{2,*}rudi.kurniawan@univbinainsan.ac.id, ³tri_hasanahba@univbinainsan.ac.id

Email Penulis Korespondensi: rudi.kurniawan@univbinainsan.ac.id

Abstrak—Pada umumnya, CAPTCHA berupa image yang memuat huruf atau angka yang terdistorsi. Tes ini melibatkan pengguna untuk mengetikkan hasil tebakan huruf atau angka yang ada pada gambar terdistorsi sebagai keamanan sebelum pengguna bisa masuk atau mengakses sebuah website yang ingin dituju. Pada tulisan ini, telah dibuat sebuah simulasi sistem pembaca otomatis untuk CAPTCHA di mana huruf dan angka yang ada di dalam gambar dilakukan beberapa proses tahapan. Tahap awal melibatkan pelatihan menggunakan dataset EMNIST untuk melatih model dalam mengenali karakter huruf dan digit angka secara umum sebelum pengenalan pada gambar CAPTCHA. Selanjutnya, dilakukan proses pengenalan terhadap karakter huruf dan angka pada gambar CAPTCHA yang berbeda untuk membaca tulisan yang dimuat pada CAPTCHA tersebut. Model Convolutional Neural Network (CNN) metode LeNet-5 digunakan sebagai metode untuk pembaca huruf dan angka yang terdistorsi pada CAPTCHA dengan tingkat akurasi yang tinggi, yaitu sebesar 88,56%.

Kata Kunci: EMNIST; CAPTCHA; CNN; LeNet-5

Abstract—In general, CAPTCHA is an image containing distorted letters or numbers. This test involves users typing the results of guessing letters or numbers in the distorted image as security before users can enter or access a website they want to go to. In this paper, a simulation of an automatic reader system for CAPTCHA has been created where the letters and numbers in the image are carried out in several stages. The initial stage involves training using the EMNIST dataset to train the model to recognize general letter and digit characters before recognition in the CAPTCHA image. Furthermore, the process of recognizing letters and numbers in different CAPTCHA images is carried out to read the text contained in the CAPTCHA. The Convolutional Neural Network (CNN) model of the LeNet-5 method is used as a method for reading distorted letters and numbers in CAPTCHA with a high level of accuracy, achieving 88.56%.

Keywords: EMNIST; CAPTCHA; CNN; LeNet-5

1. PENDAHULUAN

Kemudahan dan efisiensi dalam berbagai aspek kehidupan mempengaruhi cara kita bekerja dan berinteraksi tetapi juga menciptakan berbagai peluang baru yang belum terbayangkan sebelumnya. Perkembangan teknologi informasi terutama dalam bidang jaringan komputer berkembang dengan pesat khususnya pengguna internet [1]. Oleh karena itu, perkembangan teknologi sangat penting bagi individu dan organisasi untuk memanfaatkan potensi teknologi secara maksimal di era modern ini. Penggunaan sistem komputer, perangkat keras, perangkat lunak, serta jaringan untuk mengelola, menyimpan, dan mengirimkan data. Dalam beberapa dekade terakhir perkembangan teknologi mengalami kemajuan pesat, seiring dengan inovasi yang terus-menerus di bidang perangkat keras dan perangkat lunak serta kemajuan dalam konektivitas global.

Di sisi lain, perkembangan teknologi informasi juga membawa tantangan baru seperti masalah keamanan data, privasi serta ketimpangan akses teknologi antara berbagai lapisan masyarakat. Keamanan menjadi aspek yang sangat krusial dalam dunia digital saat ini. Ancaman terhadap keamanan teknologi informasi dapat dari berbagai sumber, mulai dari serangan siber oleh hacker, perangkat lunak berbahaya (malware), hingga kesalahan manusia dalam mengelola data atau sistem. Jenis ancaman ini dapat menyebabkan kerugian finansial yang besar, kerusakan reputasi, atau bahkan dampak hukum yang serius. Oleh karena itu, keamanan teknologi sangat penting dari ancaman yang merusak, mencuri, atau memanipulasi informasi yang dimiliki oleh individu atau organisasi.

Pengguna menjadi prioritas utama dalam keamanan teknologi. Salah satu metode yang umum digunakan untuk menjaga keamanan adalah Completely Automated Public Turing Test to Tell Computers and Humans Apart (CAPTCHA). CAPTCHA dapat didefinisikan sebagai program komputer dengan kemampuan untuk menghasilkan dan untuk menilai serangkaian tes yang dapat membedakan manusia dari komputer [2]. CAPTCHA berfungsi untuk memverifikasi bahwa akses dilakukan oleh manusia, bukan oleh perangkat otomatis seperti bot. Hal ini penting dalam mencegah serangan cyber, penyalahgunaan data, dan berbagai ancaman lainnya yang dapat mengganggu keamanan platform.

Namun, di balik manfaatnya, CAPTCHA juga memiliki sejumlah kelemahan. Salah satu kelemahan utamanya adalah CAPTCHA sering kali mengganggu pengalaman pengguna, terutama jika tes yang diberikan sulit untuk diselesaikan. CAPTCHA yang berbasis teks terdistorsi dapat menyulitkan pengguna, termasuk mereka yang memiliki gangguan penglihatan atau disabilitas lainnya. Selain itu, CAPTCHA juga dapat menghambat akses pada perangkat seluler karena ukuran layar yang terbatas dan kontrol input yang kurang fleksibel. Lebih jauh lagi, perkembangan teknologi kecerdasan buatan telah memungkinkan bot untuk memecahkan CAPTCHA dengan tingkat keberhasilan yang tinggi, sehingga efektivitas CAPTCHA sebagai penghalang bot menjadi semakin dipertanyakan.

CAPTCHA sebagai security code adalah deretan karakter atau symbol yang ditampilkan secara acak pada halaman form berupa gambar. Pengguna diharuskan menulis deretan karakter tersebut kedalam sebuah field sebagai syarat untuk dapat mengirimkan atau melanjutkan aktifitas pengisian data pada form [3]. CAPTCHA sering kali mengganggu pengalaman pengguna, terutama jika tes yang diberikan sulit untuk diselesaikan. Pengguna dengan disabilitas, misalnya dapat merasa kesulitan dalam menyelesaikan CAPTCHA yang berbasis teks angka atau huruf. Beberapa CAPTCHA terkadang sangat sulit untuk diselesaikan, meskipun pengguna berusaha baik. Hal ini bisa menyebabkan frustrasi dan bahkan mengarah pada pengguna yang meninggalkan platform atau aplikasi yang meminta CAPTCHA.

Untuk menghadapi tantangan ini, dibutuhkan pendekatan yang lebih efektif dan akurat dalam mengenali CAPTCHA, terutama dalam bentuk gambar yang terdistorsi. Penelitian ini dipilih pendekatan Convolutional Neural Networks (CNNs) yang telah terbukti sebagai kerangka mesin pembelajaran yang sangat sukses untuk pengenalan objek [4]. CNN merupakan jenis jaringan syaraf tiruan yang dirancang untuk mengenali pola kompleks dalam data visual. Berbeda dengan metode konvensional, CNN mampu menangkap fitur visual dari gambar CAPTCHA dengan mengidentifikasi tepi, tekstur, dan pola yang sulit dikenali oleh model lain, menjadikannya sangat cocok untuk tugas pengenalan CAPTCHA yang kompleks.

Penelitian ini bertujuan untuk mengembangkan dan mengoptimalkan model CNN dengan metode LeNet-5 untuk mengenali CAPTCHA dengan tingkat akurasi yang tinggi dan waktu komputasi yang efisien. LeNet-5 adalah arsitektur CNN yang terdiri dari suatu jaringan berlapis banyak sehingga LeNet-5 memiliki jumlah parameter bebas atau jumlah lapisan yang lebih banyak daripada arsitektur lainnya [5]. Optimalisasi model CNN LeNet-5 diharapkan dapat menghasilkan sistem yang andal dalam membedakan CAPTCHA yang valid dan yang tidak, sekaligus memperkuat keamanan platform dari ancaman akses otomatis. Penelitian ini berperan penting dalam menyediakan solusi yang tangguh dan efisien guna mempermudah dalam mengidentifikasi karakter CAPTCHA dengan tingkat akurasi yang tinggi.

2. METODOLOGI PENELITIAN

2.1 Dataset dan Preprocessing

Dataset yang digunakan terdiri dari 1.070 gambar CAPTCHA yang dikumpulkan dari berbagai sumber daring. Preprocessing dilakukan untuk meningkatkan kualitas gambar, meliputi thresholding, noise removal, erosion, dan dilasi. Setelah preprocessing, karakter dalam CAPTCHA dipartisi menggunakan metode Gaussian untuk memudahkan segmentasi.

Model yang dibangun untuk memprediksi CAPTCHA menggunakan arsitektur sequential adalah sebuah jaringan saraf dalam (deep neural network) yang terdiri dari beberapa lapisan bertingkat. Gambar 4.9 menyajikan model summary dari sistem pengenalan captcha yang akan dibangun. Berikut adalah deskripsi mendetail mengenai struktur model ini berdasarkan informasi yang diberikan:

Deskripsi Struktur Model (Revisi dengan Pooling Layer):

Lapisan 1: Conv2D – menghasilkan 128 fitur dari input berukuran 20x10.

Lapisan 2: MaxPooling2D – mengurangi ukuran fitur untuk mencegah overfitting.

Lapisan 3: Conv2D – menghasilkan 32 fitur dengan ukuran lebih kecil.

Lapisan 4: MaxPooling2D – kembali mengurangi dimensi data.

Lapisan 5: Conv2D – menghasilkan 16 fitur fokus pada detail karakter.

Lapisan 6: MaxPooling2D – memperkecil fitur map lebih lanjut.

Lapisan 7: Flatten – meratakan fitur menjadi vektor satu dimensi.

Lapisan 8: Dense – lapisan fully connected dengan 32 neuron.

Lapisan 9: Dense – tambahan lapisan fully connected dengan 32 neuron.

Lapisan 10: Dense – lapisan output dengan 19 neuron, sesuai jumlah kelas karakter CAPTCHA.

Deskripsi Model

a. Arsitektur Umum

Model ini menggunakan pendekatan Sequential, yang berarti setiap lapisan dihubungkan secara berurutan. Hal ini adalah cara yang umum digunakan dalam membangun model jaringan saraf untuk tugas-tugas seperti pengenalan pola dan klasifikasi gambar.

b. Lapisan-lapisan dalam Model

Lapisan 1: sequential_1

1. Tipe: Sequential

2. Output Shape: (None, 20, 10, 128)

3. Parameter: 1,792

4. Deskripsi: Lapisan ini kemungkinan besar merupakan lapisan konvolusi (Conv2D) yang mengolah citra input dengan menghasilkan 128 fitur dari citra berukuran 20x10. Ini membantu dalam mengekstraksi fitur awal dari karakter-karakter dalam CAPTCHA.

Lapisan 2: sequential_2

1. Tipe: Sequential

2. Output Shape: (None, 10, 5, 32)

3. Parameter: 37,024

4. Deskripsi: Lapisan ini juga merupakan lapisan konvolusi yang mengurangi dimensi citra menjadi 10x5 sambil menghasilkan 32 fitur baru. Proses ini membantu dalam mendeteksi pola yang lebih kompleks dan mengurangi ukuran data.

Lapisan 3: sequential_3

1. Tipe: Sequential
2. Output Shape: (None, 5, 3, 16)
3. Parameter: 4,688
4. Deskripsi: Lapisan ini melanjutkan proses ekstraksi fitur dengan menghasilkan output berukuran lebih kecil (5x3) dengan 16 fitur. Ini menunjukkan bahwa model semakin fokus pada detail-detail penting dari karakter.

Lapisan 4: flatten

1. Tipe: Flatten
2. Output Shape: (None, 240)
3. Parameter: 0
4. Deskripsi: Lapisan ini mengubah output dari lapisan sebelumnya menjadi vektor satu dimensi. Dengan mengubah data menjadi bentuk datar, model dapat mempersiapkan data untuk diproses oleh lapisan-lapisan berikutnya.

Lapisan 5: sequential_4

1. Tipe: Sequential
2. Output Shape: (None, 32)
3. Parameter: 7,840
4. Deskripsi: Ini adalah lapisan dense (fully connected) yang menghubungkan semua neuron dari lapisan sebelumnya ke neuron di lapisan ini. Dengan output sebesar 32 neuron, lapisan ini berfungsi untuk menangkap hubungan non-linear antara fitur-fitur yang diekstrak.

Lapisan 6: sequential_5

1. Tipe: Sequential
2. Output Shape: (None, 32)
3. Parameter: 1,184
4. Deskripsi: Lapisan dense tambahan yang juga memiliki output sebesar 32 neuron. Ini memberikan kesempatan lebih bagi model untuk belajar representasi yang lebih kompleks.

Lapisan Terakhir: dense_2

1. Tipe: Dense
2. Output Shape: (None, 19)
3. Parameter: 627
4. Deskripsi: Ini adalah lapisan output yang menghasilkan prediksi akhir. Dengan output sebesar 19 neuron, ini menunjukkan bahwa model dirancang untuk mengenali hingga 19 kelas karakter atau simbol yang mungkin muncul dalam CAPTCHA.

Total Parameter

Model ini memiliki total parameter sebanyak:

1. Total params: 53,155
2. Trainable params: 52,675
3. Non-trainable params: 480

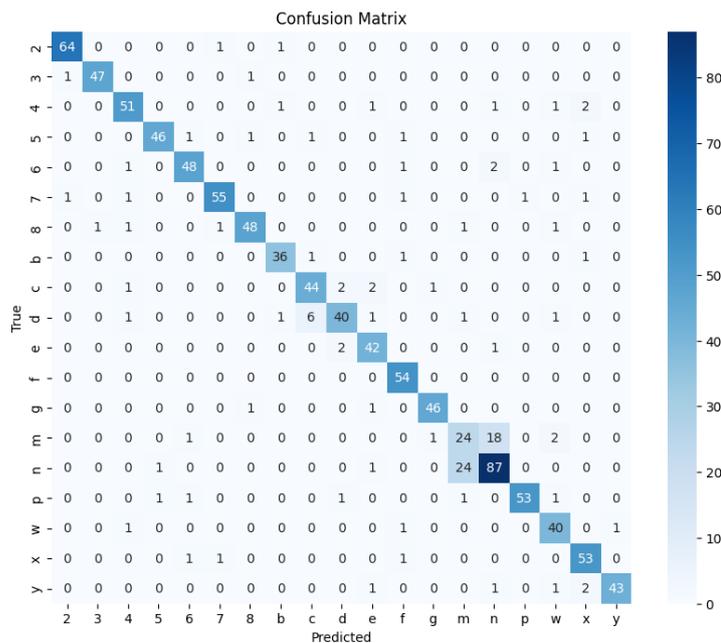
Tabel 1. Model sequential

Layer (type)	Output Shape	Param #
sequential_1 (Sequential)	(None, 20, 10, 128)	1,792
sequential_2 (Sequential)	(None, 10, 5, 32)	37,024
sequential_3 (Sequential)	(None, 5, 3, 16)	4,688
flatten (Flatten)	(None, 240)	0
sequential_4 (Sequential)	(None, 32)	7,840
sequential_5 (Sequential)	(None, 32)	1,184
dense_2 (Sequential)	(None, 19)	627

Model sequential ini dirancang dengan beberapa lapisan konvolusi dan dense untuk menangkap dan memproses informasi dari gambar CAPTCHA secara efektif. Dengan kombinasi teknik pengolahan citra dan arsitektur jaringan saraf yang mendalam, model ini bertujuan untuk meningkatkan akurasi dalam mengenali karakter-karakter di dalam CAPTCHA. Proses pelatihan akan melibatkan penyesuaian parameter untuk memaksimalkan kemampuan model dalam memprediksi karakter dengan benar.

2.2 Evaluasi Model

Evaluasi model antara lain melihat seberapa bagus model dalam mengidentifikasi setiap kelas yang ada. Evaluasi model ini meliputi confusion matrix, dan classification report. Gambar 1 dan Gambar 2 merupakan hasil confusion matrix dan Classification report



Gambar 1. Confusion Matrix

Accuracy : 0.885576923076923

	precision	recall	f1-score	support
2	0.97	0.97	0.97	66
3	0.98	0.96	0.97	49
4	0.89	0.89	0.89	57
5	0.96	0.90	0.93	51
6	0.92	0.91	0.91	53
7	0.95	0.92	0.93	60
8	0.94	0.91	0.92	53
b	0.92	0.92	0.92	39
c	0.85	0.88	0.86	50
d	0.89	0.78	0.83	51
e	0.86	0.93	0.89	45
f	0.90	1.00	0.95	54
g	0.96	0.96	0.96	48
m	0.47	0.52	0.49	46
n	0.79	0.77	0.78	113
p	0.98	0.91	0.95	58
w	0.83	0.93	0.88	43
x	0.88	0.95	0.91	56
y	0.98	0.90	0.93	48
accuracy			0.89	1040
macro avg	0.89	0.89	0.89	1040
weighted avg	0.89	0.89	0.89	1040

Gambar 2. Classification report

Gambar 2 menyajikan hasil dari confusion matrik. Confusion matrix tersebut menggambarkan kinerja model klasifikasi dalam memprediksi berbagai kelas. Berikut adalah analisis berdasarkan matriks yang ditampilkan:

- a. Diagonal utama:
 - Nilai pada diagonal utama (dari kiri atas ke kanan bawah) menunjukkan jumlah prediksi yang benar (True Positives). Nilai-nilai seperti 64, 47, 51, dan lainnya menunjukkan bahwa model berhasil memprediksi kelas dengan benar sejumlah angka tersebut.
- b. Kesalahan klasifikasi:
 - Nilai di luar diagonal utama menunjukkan jumlah kesalahan klasifikasi (misclassification). Contoh:
 1. Pada kelas 2, terdapat 1 prediksi salah ke kelas 3 dan 4.
 2. Kelas m memiliki beberapa kesalahan prediksi ke kelas g, n, dan lainnya.
 3. Beberapa kelas seperti x dan w memiliki kesalahan kecil, namun konsisten.
- c. Kelas dominan:
 - 1. Kelas p memiliki prediksi paling dominan, dengan 87 prediksi benar pada kelas tersebut.
 - 2. Beberapa kelas lain seperti y dan n memiliki prediksi benar yang cukup tinggi juga.
- d. Kesalahan kecil:

Sebagian besar nilai kesalahan di luar diagonal adalah angka kecil (0-2), menunjukkan bahwa model cukup baik dalam mengenali pola untuk setiap kelas.

Gambar 3 menyajikan classification report dari model klasifikasi. Hasil klasifikasi yang diberikan menunjukkan kinerja model dalam mengklasifikasikan 19 kelas yang berbeda. Berikut adalah penjelasan mendetail tentang metrik yang disajikan dalam laporan klasifikasi:

a. Akurasi Model

Akurasi: 0.8856 (atau sekitar 88.56%)

Akurasi ini menunjukkan bahwa model berhasil mengklasifikasikan dengan benar sekitar 88.56% dari total data yang diuji. Ini adalah indikasi yang baik bahwa model memiliki performa yang cukup tinggi dalam mengenali karakter-karakter CAPTCHA.

b. Metode Evaluasi

Classification report ini menyajikan tiga metrik evaluasi utama untuk setiap kelas:

1. Precision: Mengukur proporsi prediksi positif yang benar dari total prediksi positif.
2. Recall: Mengukur proporsi prediksi positif yang benar dari total aktual positif.
3. F1-score: Harmonik rata-rata dari precision dan recall, memberikan keseimbangan antara keduanya.

c. Analisis Kinerja per Kelas

Berikut adalah analisis untuk setiap kelas karakter yang diidentifikasi dalam laporan:

Kelas 2 hingga Kelas y:

1. Kelas 2, 3, 5, 6, 7, 8, f, g, p, y: Memiliki precision dan recall di atas 0.90, menunjukkan bahwa model sangat efektif dalam mengenali karakter-karakter ini dengan baik.
2. Kelas 4 dan e: Meskipun sedikit lebih rendah, keduanya masih menunjukkan performa yang baik dengan F1-score di atas 0.89.
3. Kelas c dan d:
 - a) Kelas c memiliki precision yang lebih rendah (0.85) tetapi recall yang lebih tinggi (0.88), menunjukkan bahwa meskipun ada beberapa kesalahan dalam prediksi positif, model cukup baik dalam menangkap sebagian besar karakter c yang sebenarnya.
 - b) Kelas d menunjukkan precision (0.89) dan recall (0.78) yang lebih rendah, dengan F1-score di 0.83, menandakan bahwa model sering kali salah mengklasifikasikan karakter ini.
4. Kelas m: Kelas m memiliki performa terendah dengan precision (0.47) dan recall (0.52), serta F1-score (0.49). Ini menunjukkan bahwa model kesulitan dalam mengenali karakter ini, dan mungkin memerlukan perhatian lebih lanjut dalam pelatihan atau pengolahan data.
5. Kelas n: Kelas n juga menunjukkan performa menengah dengan precision (0.79) dan recall (0.77), serta F1-score (0.78). Ini mengindikasikan bahwa meskipun model cukup baik dalam mengenali karakter ini, masih ada ruang untuk perbaikan.

d. Rata-rata Makro Tertimbang

Macro Average:

1. Precision: 0.89
2. Recall: 0.89
3. F1-score: 0.89

Ini menunjukkan rata-rata kinerja model di seluruh kelas tanpa mempertimbangkan jumlah sampel di setiap kelas.

Weighted Average:

1. Precision: 0.89
2. Recall: 0.89
3. F1-score: 0.89

Weighted average memperhitungkan jumlah sampel di setiap kelas dan memberikan gambaran kinerja keseluruhan model secara lebih realistis.

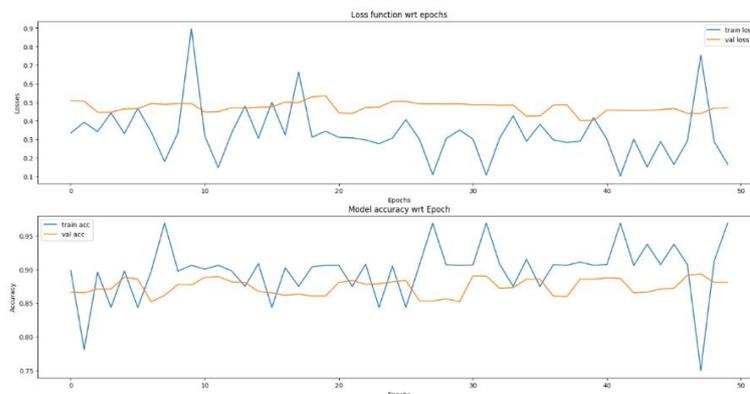
Secara keseluruhan, classification report ini menunjukkan bahwa model memiliki performa yang baik dalam mengenali sebagian besar karakter CAPTCHA dengan akurasi sekitar 88.56%. Namun, ada beberapa kelas seperti m dan n yang menunjukkan kinerja kurang memuaskan dan mungkin memerlukan perbaikan lebih lanjut melalui teknik augmentasi data atau penyesuaian parameter model untuk meningkatkan akurasi pengenalan karakter tersebut.

3. HASIL DAN PEMBAHASAN

3.1 Hasil

3.1.1 Grafik loss dan accuracy

Menyajikan data tentang grafik loss dan accuracy. Terlihat jelas bahwa model berjalan dengan baik yang menunjukkan model tidak terjadi overfitting.



Gambar 3. Grafik loss dan accuracy

3.1.2 Persiapan 1distribusi data

Dataset terdiri dari 1 folder utama, yaitu dengan nama 'samples' dimana terdiri dari kumpulan captcha yang berjumlah 1070 gambar. Gambar 4 menyajikan contoh gambar capcha yang ada.



Gambar 4. Contoh Captcha dari Dataset

3.1.3 Preprocessing data

Proses preprocessing dalam pengenalan CAPTCHA melibatkan beberapa teknik pemrosesan citra yang bertujuan untuk mempersiapkan gambar CAPTCHA agar dapat dikenali dengan lebih baik oleh sistem. Berikut adalah mengenai teknik-teknik yang digunakan antara lain :

a. Thresholding

Thresholding adalah teknik yang digunakan untuk memisahkan objek dari latar belakang dalam citra. Proses ini melibatkan pemilihan nilai ambang (threshold) yang membedakan antara piksel yang dianggap sebagai objek (misalnya, karakter dalam CAPTCHA) dan piksel yang dianggap sebagai latar belakang. Dengan menerapkan threshold, semua piksel dengan nilai intensitas di atas threshold akan diubah menjadi satu nilai (biasanya putih), sedangkan piksel di bawah threshold akan diubah menjadi nilai lain (biasanya hitam). Ini sangat berguna untuk mengurangi noise dan meningkatkan kontras antara karakter dan latar belakang. Gambar 4.2 menyajikan proses thresholding yang digunakan.



Gambar 5. Proses Thresholding Captcha

b. Erosion

Erosion adalah teknik morfologi yang digunakan untuk mengurangi ukuran objek dalam citra. Dalam konteks CAPTCHA, erosi membantu menghilangkan detail kecil yang tidak diinginkan atau noise yang mungkin mengganggu proses pengenalan karakter. Proses ini dilakukan dengan menggunakan struktur elemen (structuring element) yang bergerak di atas citra, menghapus piksel yang tidak sepenuhnya terhubung dengan objek utama. Hasilnya adalah pengurangan detail pada karakter, membuatnya lebih mudah untuk dipisahkan dari latar belakang. Gambar 6 menyajikan proses erosi.



Gambar 6. Proses Erosion Captcha

c. Dilation

Dilation adalah teknik morfologi yang berlawanan dengan erosi. Setelah proses erosi, dilasi digunakan untuk memperbesar objek dalam citra. Proses ini juga menggunakan struktur elemen yang sama dan membantu mengisi celah atau lubang kecil pada karakter yang mungkin muncul setelah erosi. Dengan menerapkan dilasi, karakter menjadi lebih solid dan lebih mudah dikenali dalam langkah-langkah berikutnya. Gambar 7 menyajikan proses dilation.



Gambar 7. Proses Dilation Captcha

d. Blurring

Blurring atau pengaburan adalah teknik yang digunakan untuk menghaluskan citra dengan cara mengurangi detail dan noise. Dalam preprocessing CAPTCHA, blurring membantu mengurangi ketajaman tepi karakter, sehingga membuat proses segmentasi karakter lebih efektif. Metode seperti Gaussian blur sering digunakan, di mana setiap piksel dalam citra diganti dengan rata-rata nilai piksel di sekitarnya, menghasilkan efek yang lebih halus. Gambar 8 menyajikan proses blurring.



Gambar 8. Proses Blurring Captcha

e. Partisi Karakter Menggunakan Gaussian

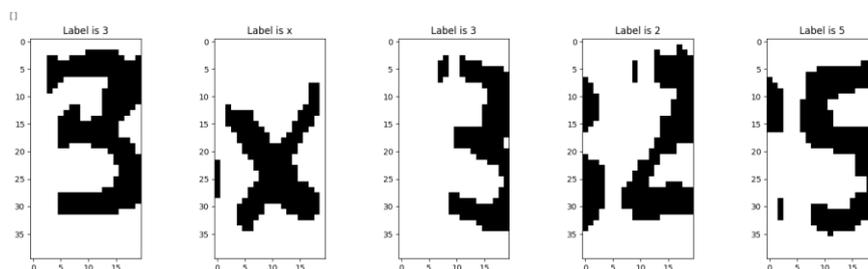
Setelah tahap-tahap sebelumnya, partisi karakter dilakukan untuk memisahkan setiap karakter individu dalam CAPTCHA. Teknik ini sering kali melibatkan penggunaan metode berbasis Gaussian untuk mendeteksi tepi dan memisahkan karakter dari latar belakang serta dari satu sama lain. Dengan memanfaatkan distribusi Gaussian, sistem dapat lebih akurat dalam menentukan batas-batas karakter, sehingga meningkatkan akurasi pengenalan.



Gambar 9. Partisi Karakter Gaussian

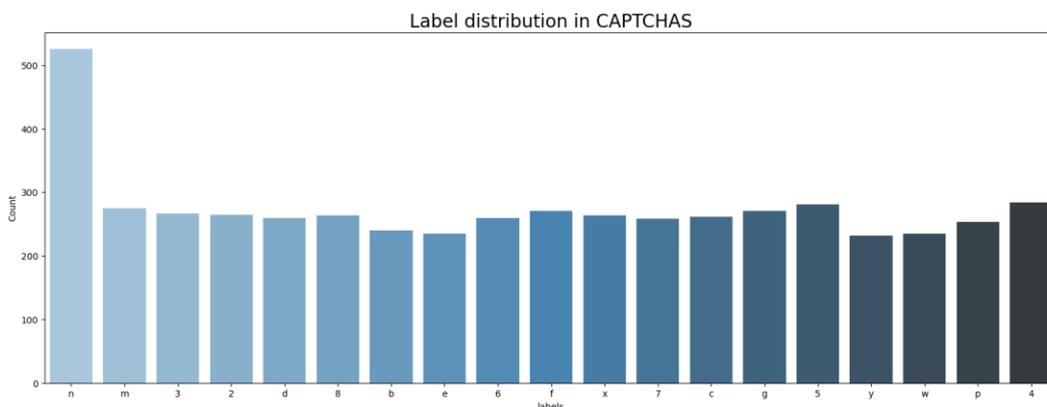
Proses preprocessing ini sangat penting dalam pengenalan CAPTCHA karena membantu meningkatkan kualitas citra sehingga memudahkan algoritma pengenalan karakter dalam mengenali dan mengekstraksi informasi dari gambar CAPTCHA dengan lebih efisien dan akurat. Setiap teknik memiliki perannya masing-masing dalam memastikan bahwa data yang dihasilkan siap untuk langkah-langkah selanjutnya dalam proses pengenalan.

Proses selanjutnya adalah melakukan segmentasi masing-masing gambar captcha (1070) menjadi masing-masing karakter berdasarkan proses preprocessing tersebut. dan memberikan label pada masing-masing karakter yang ada di captcha.



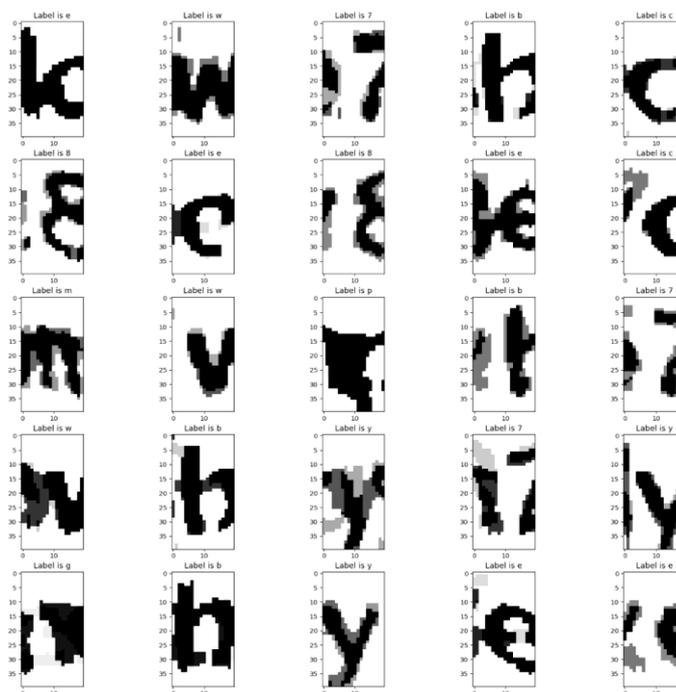
Gambar 10. Proses Pelabelan Segmentasi

Dari proses pelabelan, didapat jumlah keseluruhan data yang dihasilkan sebanyak 5200 karakter yang tersebar dalam 19 kelas. Gambar 11 menyajikan distribusi data yang dihasilkan dari proses pelabelan dataset.



Gambar 11. distribusi dari setiap label dari captcha

- f. Proses Persiapan Training dan Testing Data
Seterusnya, data yang berjumlah sebanyak 5200 karakter dibagi menjadi 2 bagian, yaitu data train dan data test dengan kombinasi 80 % (4160) data train dan 20 % (1040) data test.
- g. Proses SMOTE
Proses SMOTE dilakukan untuk mengatasi data yang tidak seimbang (imbalance). Dapat dilihat pada gambar dibawah terdapat satu kelas yang dominan (kelas 'n') dimana distribusinya hamper 2 kali lipat dari distribusi karakter yang lain.



Gambar 12. distribusi label data setelah proses SMOTE

3.2 Pembahasan

Hasil yang diperoleh dari model pengenalan CAPTCHA menunjukkan performa yang cukup baik, dengan akurasi keseluruhan mencapai 88.56%. Angka ini mencerminkan kemampuan model dalam mengklasifikasikan karakter-karakter dalam CAPTCHA dengan tingkat kesalahan yang relatif rendah. Dalam konteks pengenalan karakter, akurasi adalah metrik penting karena menunjukkan seberapa baik model dapat memisahkan karakter yang berbeda dalam gambar. Meskipun akurasi di atas 88% sudah tergolong baik, analisis lebih mendalam terhadap metrik-metrik lain seperti precision, recall, dan F1-score memberikan wawasan lebih lanjut mengenai kekuatan dan kelemahan model.

Precision adalah ukuran yang menunjukkan proporsi prediksi positif yang benar dari total prediksi positif. Dalam laporan, sebagian besar kelas karakter menunjukkan nilai precision di atas 0.90, yang berarti model sangat efektif dalam memprediksi karakter-karakter tersebut tanpa terlalu banyak menghasilkan false positives. Misalnya, kelas 2 dan 3

memiliki precision masing-masing 0.97 dan 0.98, menunjukkan bahwa hampir semua prediksi untuk kelas ini adalah benar. Hal ini sangat penting dalam aplikasi nyata di mana kesalahan dalam pengenalan karakter dapat mengakibatkan masalah, seperti ketika CAPTCHA digunakan untuk mengamankan situs web dari bot. Di sisi lain, recall mengukur proporsi prediksi positif yang benar dari total aktual positif. Ini memberikan gambaran tentang seberapa baik model dalam menangkap semua contoh dari suatu kelas tertentu. Kelas-kelas seperti f dan g juga menunjukkan nilai recall yang tinggi (masing-masing 1.00 dan 0.96), menandakan bahwa model mampu mengenali hampir semua karakter dari kelas tersebut. Namun, tidak semua kelas menunjukkan kinerja yang sama baiknya. Kelas m, misalnya, memiliki nilai precision dan recall yang rendah (0.47 dan 0.52), sehingga menandakan bahwa model sering kali gagal mengenali karakter ini dengan tepat. Hal ini menunjukkan bahwa ada tantangan tertentu dalam mengenali karakter m, mungkin disebabkan oleh kompleksitas visual atau variasi dalam gaya penulisan.

F1-score merupakan metrik gabungan antara precision dan recall, memberikan gambaran menyeluruh tentang keseimbangan antara keduanya. Model ini menunjukkan F1-score di atas 0.89 untuk sebagian besar kelas karakter, mencerminkan bahwa model tidak hanya mampu memprediksi dengan akurat tetapi juga mampu menangkap sebagian besar contoh dari setiap kelas. Namun, kelas m kembali menjadi perhatian utama dengan F1-score terendah (0.49), yang mengindikasikan bahwa perbaikan diperlukan untuk meningkatkan pengenalan karakter ini.

Rata-rata makro dan tertimbang juga memberikan wawasan tambahan mengenai kinerja keseluruhan model. Rata-rata makro menghitung rata-rata dari metrik tanpa mempertimbangkan jumlah sampel di setiap kelas, sedangkan rata-rata tertimbang memperhitungkan jumlah sampel tersebut. Kedua rata-rata menunjukkan hasil yang serupa di sekitar 0.89 untuk precision, recall, dan F1-score, menandakan konsistensi kinerja model di seluruh kelas meskipun ada beberapa kelas dengan performa lebih rendah.

Salah satu aspek penting dari hasil ini adalah adanya variasi dalam kinerja antar kelas karakter. Kelas-kelas seperti 2, 3, f, g, p, dan y menunjukkan performa tinggi dengan precision dan recall mendekati atau bahkan mencapai 1.00. Ini menunjukkan bahwa model dapat mengenali karakter-karakter ini dengan sangat baik dan dapat diandalkan untuk aplikasi praktis. Sebaliknya, kelas-kelas seperti m dan n menunjukkan bahwa ada ruang untuk perbaikan lebih lanjut. Untuk meningkatkan kinerja pada kelas-kelas ini, beberapa pendekatan dapat dipertimbangkan.

Pertama, augmentasi data dapat dilakukan untuk meningkatkan keragaman data pelatihan dengan menciptakan variasi baru dari gambar-gambar CAPTCHA yang ada. Misalnya, menambahkan noise atau melakukan rotasi pada gambar dapat membantu model belajar mengenali karakter dalam berbagai kondisi visual yang berbeda. Selain itu, teknik pemrosesan citra tambahan bisa diterapkan untuk meningkatkan kualitas gambar sebelum diproses oleh model.

Kedua, fine-tuning arsitektur model juga bisa menjadi langkah strategis untuk meningkatkan kinerja pengenalan karakter tertentu. Menambahkan lapisan konvolusi tambahan atau menggunakan teknik regularisasi seperti dropout dapat membantu model belajar lebih baik tanpa overfitting pada data pelatihan.

Ketiga, penggunaan transfer learning dengan menerapkan model yang telah dilatih sebelumnya pada dataset besar bisa menjadi solusi efektif untuk meningkatkan akurasi pengenalan karakter-karakter yang sulit dikenali seperti m dan n. Dengan memanfaatkan pengetahuan dari model lain yang telah terbukti efektif dalam tugas serupa, kita bisa mendapatkan hasil yang lebih baik tanpa memerlukan banyak data pelatihan tambahan.

Keempat, evaluasi lebih lanjut terhadap dataset juga penting dilakukan untuk memastikan bahwa data pelatihan mencakup variasi yang cukup dari setiap kelas karakter. Jika beberapa kelas memiliki jumlah contoh pelatihan yang jauh lebih sedikit dibandingkan dengan kelas lainnya (seperti m), ini bisa menyebabkan bias dalam pelatihan model.

Secara keseluruhan, hasil classification report ini memberikan gambaran positif tentang kemampuan model dalam mengenali CAPTCHA dengan akurasi tinggi secara keseluruhan namun juga menyoroti area-area spesifik di mana perbaikan diperlukan untuk mencapai performa optimal di semua kelas karakter.

Melakukan langkah-langkah perbaikan yang tepat berdasarkan analisis ini, kita dapat berharap untuk meningkatkan efektivitas sistem pengenalan CAPTCHA lebih lanjut di masa depan sehingga dapat berfungsi secara optimal dalam mengidentifikasi dan memverifikasi pengguna secara otomatis tanpa mengorbankan keamanan atau kenyamanan pengguna akhir. Hasil saat ini sudah cukup memuaskan dengan akurasi hampir mencapai 89%, masih ada tantangan tersisa terutama pada beberapa kelas tertentu yang perlu ditangani agar sistem pengenalan CAPTCHA dapat bekerja secara maksimal dalam berbagai situasi nyata di lapangan. Melalui pendekatan sistematis terhadap evaluasi hasil ini dan implementasi strategi perbaikan berbasis data serta teknologi terkini dalam machine learning dan computer vision, kita dapat terus meningkatkan kemampuan sistem pengenalan CAPTCHA untuk memenuhi kebutuhan keamanan digital saat ini dan masa depan.

4. KESIMPULAN

Hasil evaluasi model pengenalan CAPTCHA menunjukkan performa yang baik dengan akurasi mencapai 88.56%, mencerminkan kemampuan model dalam mengenali karakter-karakter dengan tingkat kesalahan yang relatif rendah. Metrik-metrik seperti precision, recall, dan F1-score menunjukkan bahwa sebagian besar kelas karakter dapat dikenali dengan efektif, meskipun terdapat beberapa kelas, seperti m dan n, yang memerlukan perhatian lebih lanjut. Secara keseluruhan, model ini menunjukkan potensi yang kuat untuk aplikasi di dunia nyata dalam mengidentifikasi dan memverifikasi pengguna secara otomatis. Kontribusi utama dari penelitian ini adalah pengembangan dan optimalisasi model Convolutional Neural Network (CNN) berbasis arsitektur LeNet-5 untuk pengenalan CAPTCHA berbasis gambar

terdistorsi. Dengan pendekatan ini, sistem mampu mencapai akurasi tinggi sebesar 88,56%, menunjukkan efektivitas penggunaan metode deep learning dalam menyelesaikan tantangan pengenalan karakter CAPTCHA secara otomatis. Selain itu, penelitian ini berhasil mengintegrasikan tahapan preprocessing gambar yang komprehensif seperti thresholding, noise removal, erosion, dilasi, dan partisi karakter Gaussian untuk meningkatkan kualitas data input, yang berkontribusi signifikan terhadap peningkatan kinerja model. Implikasi jangka panjang dari penelitian ini adalah potensi penerapan sistem pengenalan CAPTCHA berbasis deep learning dalam berbagai platform keamanan digital. Dengan keberhasilan model ini, penelitian membuka jalan untuk eksplorasi lebih lanjut dalam bidang keamanan siber, khususnya di bidang human-computer interaction (HCI) dan proteksi sistem berbasis verifikasi visual.

REFERENCES

- [1] G. Satya Nugraha, G. Pasek, S. Wijaya, F. Bimantoro, Y. Husodo, and F. Hamami, "INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION journal homepage : www.joiv.org/index.php/joiv INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION Arabic Character Recognition Using CNN LeNet-5." [Online]. Available: www.joiv.org/index.php/joiv
- [2] J. Zhang, X. Yu, X. Lei, and C. Wu, "A Novel Deep LeNet-5 Convolutional Neural Network Model for Image Recognition," *Computer Science and Information Systems*, vol. 19, no. 3, pp. 1463–1480, Sep. 2022, doi: 10.2298/CSIS220120036Z.
- [3] M. Rafly Alwanda, R. Putra, K. Ramadhan, and D. Alamsyah, "Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle," 2020.
- [4] A. H. Suherman, N. Ibrahim, H. Syahrian, V. P. Rahadi, and M. K. Prayoga, "KLASIFIKASI DAUN TEH GAMBUNG VARIETAS ASSAMICA MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK DENGAN ARSITEKTUR LENET-5," *JOURNAL OF ELECTRICAL AND SYSTEM CONTROL ENGINEERING*, vol. 4, no. 2, pp. 63–71, Feb. 2021, doi: 10.31289/jesce.v4i2.4136.
- [5] M. R. Alwanda, R. P. K. Ramadhan, and D. Alamsyah, "Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle," *Jurnal Algoritme*, vol. 1, no. 1, pp. 45–56, 2020, doi: 10.35957/algoritme.v1i1.434.
- [6] R. Bangun Pengaman Sistem Login Menggunakan Metode Captcha and D. Rusmana, "Rancang Bangun-Dadang Rusmana DESIGN AND BUILD LOGIN SYSTEM SECURITY USING CAPTCHA METHOD," vol. 10, no. 1, 2021.
- [7] D. Sutaji and D. N. Husenti, "DETEKSI KARAKTER PADA CITRA CAPTCHA LOGIN INTERNET BANKING MENGGUNAKAN TEMPLATE MATCHING," 2019.
- [8] M. A. Yasin, "MEMECAHKAN CAPTCHA-TEXT TERDISTORSI DENGAN CONVOLUTIONAL NEURAL NETWORK," 2024.
- [9] A. Mulyanto, E. Susanti, F. Rossi, W. Wajiran, and R. I. Borman, "Penerapan Convolutional Neural Network (CNN) pada Pengenalan Aksara Lampung Berbasis Optical Character Recognition (OCR)," *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 1, p. 52, 2021, doi: 10.26418/jp.v7i1.44133.
- [10] F. Ilham and N. Rochmawati, "Transliterasi Aksara Jawa Tulisan Tangan ke Tulisan Latin Menggunakan CNN," *J. Informatics Comput. Sci.*, vol. 1, no. 04, pp. 200–208, 2020, doi: 10.26740/jinacs.v1n04.p200-208.
- [11] Y. Brianorman and R. Muir, "Perbandingan Pre-Trained CNN: Klasifikasi Pengenalan Bahasa Isyarat Huruf Hijaiyah," *J. Sist. Info. Bisnis*, vol. 13, no. 1, pp. 52–59, 2023, doi: 10.21456/vol13iss1pp52-59.
- [12] F. Nuraeni et al., "IMPLEMENTASI ALGORITMA CONVOLUTIONAL NEURAL NETWORK (CNN) UNTUK PENGENALAN BACAAN TAJWID BERDASARKAN GAMBAR TULISAN DALAM AL- QUR ' AN," vol. 8, no. 5, pp. 11018–11025, 2024.
- [13] M. Malika and E. Widodo, "Implementasi Deep Learning Untuk Klasifikasi Gambar Menggunakan Convolutional Neural Network (Cnn) Pada Batik Sasambo," *Pattimura Proceeding Conf. Sci. Technol.*, pp. 335–340, 2022, doi: 10.30598/pattimurasci.2021.knmxx.335-340.
- [14] N. Fadlia and R. Kosasih, "Klasifikasi Jenis Kendaraan Menggunakan Metode Convolutional Neural Network (Cnn)," *J. Ilm. Teknol. dan Rekayasa*, vol. 24, no. 3, pp. 207–215, 2019, doi: 10.35760/tr.2019.v24i3.2397.
- [15] D. Dhelviana, T. Amelia, J. Sulaksono, and D. W. Widodo, "Sistem Pendeteksi Kekerasan Berbasis Cnn (Convolutional Neural Network)," vol. 2, pp. 457–462, 2023.
- [16] S. Prihatiningsih, N. S. M, F. Andriani, and N. Nugraha, "Analisa Performa Pengenalan Tulisan Tangan Angka Berdasarkan Jumlah Iterasi Menggunakan Metode Convolutional Neural Network," *J. Ilm. Teknol. dan Rekayasa*, vol. 24, no. 1, pp. 58–66, 2019, doi: 10.35760/tr.2019.v24i1.1934.
- [17] R. N. Wicaksono, H. Nugroho, and G. E. Yuliasuti, "Pengenalan Pola Citra Ekspresi Wajah Manusia Menggunakan Masker Dengan Metode Convolutional Neural Network (CNN)," *Pros. Semin. Nas. Sains dan Teknol. Terap.*, pp. 1–6, 2023, [Online]. Available: <http://ejurnal.itats.ac.id/sntekpan/article/view/5157%0Ahttp://ejurnal.itats.ac.id/sntekpan/article/download/5157/3571>
- [18] A. Fadhila, M. Mabe Parenreng, J. Teknik Elektro, and P. Negeri Ujung Pandang, "Pengenalan Tanaman Herbal Daun Merica dan Daun Sirih Menggunakan Metode Convolutional Neural Network (CNN)," *Pros. Semin. Nas. Tek. Elektro dan Inform.*, vol. 9, no. 1, pp. 109–113, 2023.
- [19] I. N. Pratama, T. Rohana, and T. Al Mudzakir, "Pengenalan Sampah Plastik Dengan Model," no. Ciastech, pp. 691–698, 2020.
- [20] S. N. Amartama, A. N. Hidayah, P. K. Sari, and R. A. Ramadhani, "Implementasi Convolutional Neural Network (CNN) dalam Pengenalan Pola Penulisan Tangan," *Semin. Nas. Teknol. Sains*, vol. 3, no. 1, pp. 133–138, 2024, doi: 10.29407/stains.v3i1.4155.