OPEN ACCESS

JURNAL ILMIAH SISTEM INFORMASI (JUSI)
VOL 4, No. 2, Month 2025, pp. 175 - 185
p-ISSN : 2803-1507 (media cetak) | e-ISSN : 2803-1531 (media online)
Link Publikasi: https://ejurnal.provisi.ac.id/index.php/JUISI/issue/view/134

Implementation of the K-Nearest Neighbor (KNN) Algorithm in Handwritten Digit Pattern Recognition Using the Zoning Method

Afilda Maharani Asmaraloka[1], Muhammad Ardi Hermansyah[2], Kharisatun Nisa[3], Fikri Hamdhan Dwi Saputra[4], Arif Setiawan[5]
[1,2,3,4,5] Universitas Muria Kudus
Jl. Lkr. Utara, 59327, Kudus, Jawa Tengah, Indonesia
e-mail: risanisa033@gmail.com[1], muhammadardi931@gmail.com[2], afilda088@gmail.com[3], fikrihamdhan13@gmail.com[4], arif.setiawan@umk.ac.id[5]

ARTICLE INFO

Article History:

Received: 10 Mei 2025
Revised: 21 Mei 2025
Accepted: 25 Mei 2025
Published: 31 Mei 2025

ABSTRACT

Handwritten digit recognition is one of the key challenges in the field of digital image processing and artificial intelligence, with significant potential in various applications such as automatic form input systems, handwritten data correction, and attendance systems based on handwriting. This study aims to develop a web-based information system capable of automatically recognizing handwritten digits using the K-Nearest Neighbors (KNN) classification method. The system is designed through several main stages, including image preprocessing (conversion to grayscale, thresholding, and image size normalization), feature extraction using the zoning technique, and classification using the KNN algorithm. This research utilizes the MNIST dataset, which contains thousands of handwritten digit images ranging from 0 to 9. The system is developed using the Google Colab platform, supported by Python libraries such as OpenCV, NumPy, and Scikit-learn. Test results show that the system can achieve an accuracy of over 90% at certain K values, indicating that the KNN method is quite effective and efficient in recognizing handwritten digit patterns. This system is expected to be applicable for various digitization needs of handwritten numbers in education, administration, and information technology sectors.

Keywords: *digit recognition, handwriting, K-Nearest Neighbors, digital image processing, Google Colab.*

Abstrak

Pengenalan angka tulisan tangan merupakan salah satu tantangan penting dalam bidang pengolahan citra digital dan kecerdasan buatan yang memiliki potensi besar dalam berbagai aplikasi, seperti sistem input formulir otomatis, koreksi data tertulis, hingga sistem absensi berbasis tulisan tangan. Penelitian ini bertujuan untuk mengembangkan sebuah sistem informasi berbasis web yang dapat mengenali angka tulisan tangan secara otomatis menggunakan metode klasifikasi *K-Nearest Neighbors (KNN)*. Sistem ini dirancang melalui beberapa tahapan utama, yaitu praproses citra (konversi ke *grayscale*, *thresholding*, dan normalisasi ukuran gambar), ekstraksi fitur menggunakan teknik *zoning*, serta klasifikasi menggunakan algoritma KNN.

DOI: https://doi.org/10.51903/kf6s5f56

p-ISSN : 2803-1507 e-ISSN : 2803-1531

Penelitian ini menggunakan dataset MNIST yang berisi ribuan data gambar angka tulisan tangan dari angka 0 hingga 9. Sistem dikembangkan menggunakan platform *Google Colab* dengan dukungan pustaka Python seperti *OpenCV, NumPy, dan Scikit-learn.* Hasil pengujian menunjukkan bahwa sistem mampu mencapai akurasi lebih dari 90% pada nilai K tertentu, yang menunjukkan bahwa metode KNN cukup efektif dan efisien dalam mengenali pola angka tulisan tangan. Sistem ini diharapkan dapat diterapkan dalam berbagai kebutuhan digitalisasi tulisan angka di bidang pendidikan, administrasi, dan teknologi informasi.
Kata Kunci: *pengenalan angka, tulisan tangan, K-Nearest Neighbors, pengolahan citra digital, Google Colab.*

1. PENDAHULUAN

Perkembangan teknologi digital telah mendorong transformasi di berbagai bidang, termasuk pendidikan dan administrasi publik. Salah satu bentuk transformasi tersebut adalah pengenalan tulisan tangan (*handwritten recognition*), khususnya angka tulisan tangan (*handwritten digit recognition*), yang berperan penting dalam otomatisasi input data, sistem absensi, hingga evaluasi berbasis tulisan. Dalam ranah ini, *Machine Learning* (ML) menjadi pendekatan yang banyak digunakan karena kemampuannya dalam mengenali pola visual. Metode *K-Nearest Neighbors* (KNN) dikenal luas sebagai salah satu algoritma klasifikasi yang sederhana namun cukup efektif, terutama dalam kasus klasifikasi citra berukuran kecil seperti angka tulisan tangan.

Dataset *Modified National Institute of Standards and Technology* (MNIST) merupakan standar umum yang digunakan untuk pelatihan dan pengujian sistem pengenalan angka tulisan tangan. Dataset ini terdiri dari ribuan gambar angka dari 0 hingga 9 yang telah diproses secara seragam, sehingga cocok digunakan untuk pengujian algoritma klasifikasi. Beberapa studi terdahulu telah memanfaatkan pendekatan *deep learning* seperti *Convolutional Neural Networks* (CNN) dengan hasil akurasi tinggi, namun metode tersebut menuntut sumber daya komputasi yang besar. Dalam konteks sistem ringan dan berbasis web, KNN tetap menjadi pilihan yang relevan karena efisiensi dan kemudahan implementasinya.

Studi terkini menunjukkan bahwa kombinasi KNN dengan teknik ekstraksi fitur seperti *zoning* dapat meningkatkan akurasi klasifikasi secara signifikan. Teknik *zoning* membagi citra menjadi beberapa blok kecil untuk mengekstraksi pola lokal, sehingga menghasilkan representasi numerik yang sesuai untuk klasifikasi. Penerapan *zoning* dalam pengolahan angka tulisan tangan meningkatkan akurasi sistem KNN hingga lebih dari 90%. Sistem berbasis KNN menggunakan Python dan MNIST, namun implementasinya masih bersifat lokal dan belum terintegrasi dalam sistem informasi berbasis web atau *platform cloud* seperti *Google Colab*.

Berdasarkan studi sebelumnya, masih terdapat gap dalam hal pengembangan sistem pengenalan angka tulisan tangan berbasis KNN yang dapat diakses secara daring melalui sistem informasi berbasis web. Penelitian ini bertujuan untuk mengembangkan sistem tersebut dengan tahapan utama praproses citra, ekstraksi fitur menggunakan *zoning*, dan klasifikasi menggunakan algoritma KNN. Sistem dikembangkan menggunakan *Google Colab* dan pustaka Python seperti *OpenCV* dan *Scikit-learn*, sehingga mudah diakses dan digunakan. Kontribusi utama penelitian ini adalah menghadirkan sistem yang tidak hanya akurat, tetapi juga portabel, ringan, dan dapat langsung dimanfaatkan dalam kebutuhan digitalisasi angka tulisan tangan di berbagai sektor.

2. TINJAUAN PUSTAKA

2.1. Kajian Teori dan Penelitian

Pengenalan pola tulisan tangan angka 0-9 merupakan salah satu aplikasi penting dalam bidang pengolahan citra dan pengenalan karakter. Dalam penelitian ini menerapkan algoritma K-Nearest Neighbor (KNN) untuk mengenali pola tulisan tangan, menunjukkan bahwa metode ini efektif dalam klasifikasi dengan akurasi yang memadai [1]. Pengenalan tulisan tangan merupakan tantangan yang kompleks, terutama karena variasi dalam gaya tulisan individu. Dengan menggunakan dataset yang terdiri dari 30 responden yang menulis 26 huruf kapital dan 26 huruf non-kapital, penelitian ini memberikan wawasan tentang bagaimana KNN dapat digunakan dalam pengenalan karakter tulisan tangan, serta membuka peluang untuk penelitian lebih lanjut dalam pengembangan sistem pengenalan tulisan tangan yang lebih akurat dan efisien [2].

Dalam penelitian ini menggunakan data sekunder berupa gambar jawaban ujian esai yang diunduh dari internet. Setiap jawaban diubah menjadi fitur numerik menggunakan metode ekstraksi fitur *Zoning*, yang memungkinkan algoritma KNN untuk memprediksi angka yang ditulis dengan lebih efektif. Hasil

p-ISSN : 2803-1507 e-ISSN : 2803-1531

penelitian menunjukkan bahwa penerapan KNN dapat meningkatkan efisiensi dan akurasi dalam penilaian jawaban ujian esai, memberikan kontribusi signifikan terhadap kualitas evaluasi akademik [3]. Metode ekstraksi fitur yang digunakan dalam penelitian ini adalah *Zoning*, yang mengubah gambar tweet menjadi fitur numerik yang dapat diproses oleh algoritma KNN. Dengan menggunakan pendekatan ini, penelitian ini berhasil mengidentifikasi kelas tweet berdasarkan kedekatan dengan tetangga terdekat, di mana dua dari tiga tetangga terdekat berada dalam kelas yang sama, sehingga meningkatkan keakuratan klasifikasi [4].

Pentingnya preprocessing dan rekonstruksi citra dalam proses pengenalan, di mana data yang diambil dari sebelas mahasiswa diolah untuk menghasilkan database citra asli, terkompresi, dan rekonstruktif. Hasil penelitian ini memberikan wawasan baru tentang bagaimana teknik kompresi dapat digunakan untuk mengurangi ukuran data tanpa mengorbankan kualitas citra, serta membuka peluang untuk penelitian lebih lanjut dalam pengenalan tulisan tangan yang lebih efisien [5].

Berdasarkan studi-studi tersebut, penelitian ini menawarkan kontribusi baru dengan membangun sistem informasi berbasis web yang mampu mengenali angka tulisan tangan menggunakan metode KNN. Sistem ini juga menggunakan teknik ekstraksi fitur *zoning* untuk memaksimalkan akurasi klasifikasi dan memanfaatkan platform *Google Colab* sebagai lingkungan pemrograman. Dengan demikian, penelitian ini memperluas cakupan implementasi sistem pengenalan angka tulisan tangan yang tidak hanya akurat, tetapi juga mudah diakses.

2.2. *Gambar dan Tabel*

Pengujian terhadap model K-Nearest Neighbors (KNN) dilakukan untuk mengetahui pengaruh variasi nilai K terhadap akurasi klasifikasi angka tulisan tangan. Hasil pengujian ini kemudian disajikan dalam Tabel 1 untuk menunjukkan perbandingan akurasi berdasarkan nilai K yang digunakan. Sumber data diolah dari hasil pengujian model KNN pada dataset MNIST menggunakan platform *Google Colab*.

Tabel 1. Hasil Akurasi Model KNN Berdasarkan Nilai K

No	Nilai K	Akurasi (%)
1	3	98.33
2	5	98.61
3	7	98.89
4	11	98.33
5	13	97.78
6	17	98.33

2.3. *Rumus dan Algoritma*

Proses klasifikasi dalam KNN dilakukan dengan menghitung jarak antara data uji dan seluruh data latih, kemudian memilih K data latih terdekat sebagai dasar pengambilan keputusan. Salah satu ukuran jarak yang paling umum digunakan dalam algoritma KNN adalah jarak Euclidean, karena sederhana dan efektif untuk data berdimensi rendah hingga sedang. Dalam penelitian ini, perhitungan jarak antara dua titik, yaitu titik data uji dan data latih, menggunakan rumus *Euclidean* yang dirumuskan sebagai berikut:

$$d(p,q) = \sqrt{\left\{\sum_{\{i=1\}}^{\{n\}}(p_i - q_i)^2\right\}} \qquad (1)$$

Algoritma 1. Proses Klasifikasi Data Menggunakan KNN

```
Keterangan:
    d(x,y) = jarak Euclidean antara dua titik vektor x dan y
    n = jumlah fitur
    Xi dan Yi = nilai fitur ke-i dari masing-masing vektor


Langkah-langkah:
1.  Ambil data gambar dan lakukan praproses (grayscale, thresholding, resize)if A[i] = x then
    return i
2.  Lakukan ekstraksi fitur menggunakan metode zoning
3.  Hitung jarak Euclidean antara data uji dan semua data latih
4.  Tentukan K tetangga terdekat berdasarkan nilai jarak terkecil
5.  Tentukan label berdasarkan mayoritas dari K tetangga
```

p-ISSN : 2803-1507 e-ISSN : 2803-1531

3. METODOLOGI PENELITIAN

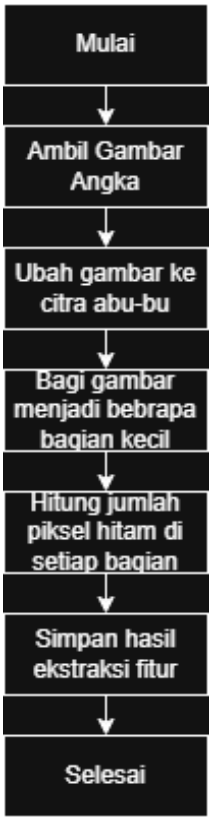
Tahapan dalam penelitian ini mencakup akuisisi data, pemrosesan dan ekstraksi ciri, pelatihan algoritma, serta evaluasi dan interpretasi performa model.

3.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan data sekunder yang berupa gambar angka 0 hingga 9 dalam format bitmap. Gambar-gambar tersebut diunduh dari internet, dengan total sebanyak 500 gambar, di mana setiap gambar memiliki resolusi 28x28 piksel.

3.2 Ekstraksi Fitur

Metode ekstraksi fitur yang digunakan adalah *Zoning*, yang berfungsi untuk mengubah gambar angka menjadi representasi numerik. Metode *Zoning* membagi gambar menjadi beberapa bagian kecil dan menghitung jumlah piksel hitam yang terdapat di setiap bagian. Hasil dari ekstraksi fitur ini akan menjadi input untuk algoritma KNN.



Gambar 1 Flowchart Metode Zoning

3.3 Pelatihan Model

Data yang telah dikumpulkan kemudian dipisahkan menjadi dua kelompok, yaitu data pelatihan dan data pengujian. Data pelatihan dimanfaatkan untuk membangun model KNN dengan menghitung jarak antara data pelatihan dan data pengujian. Model ini dilatih menggunakan berbagai nilai parameter K (jumlah tetangga terdekat) untuk menentukan konfigurasi yang paling optimal.

3.4 Pengujian Model

Setelah model selesai dilatih, dilakukan proses pengujian menggunakan data pengujian untuk mengevaluasi tingkat akurasi model KNN. Setiap gambar dalam data pengujian diklasifikasikan, lalu hasil prediksi dibandingkan dengan label sebenarnya untuk menghitung persentase keberhasilan klasifikasi.

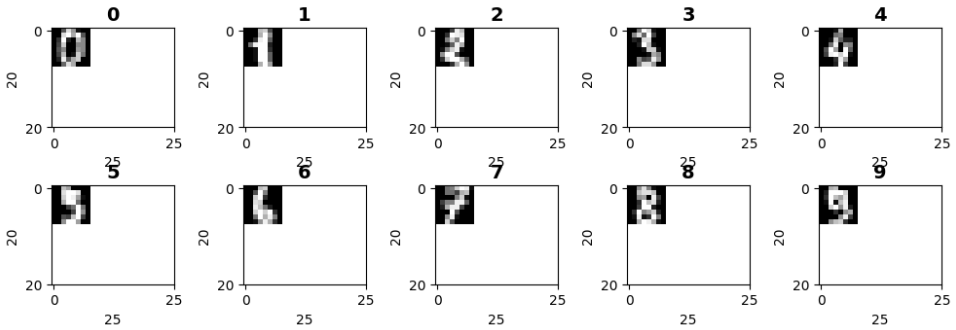
p-ISSN : 2803-1507 e-ISSN : 2803-1531

3.5 Analisis Data

Tahapan akhir dari penelitian ini adalah melakukan analisis terhadap hasil klasifikasi guna mengetahui faktor-faktor yang berpengaruh terhadap kinerja model. Beberapa aspek yang dianalisis meliputi pengaruh variasi nilai K serta sejauh mana metode ekstraksi fitur yang digunakan dapat meningkatkan akurasi model.

4. HASIL DAN PEMBAHASAN

Langkah awal yang dilakukan adalah mengambil data citra angka 0 hingga 9 dari direktori penyimpanan, kemudian mengonversinya ke dalam format citra abu-abu (*grayscale*) untuk mempermudah proses pengolahan. Setelah itu, dilakukan proses ekstraksi fitur menggunakan teknik *zoning*, yang membagi citra menjadi beberapa zona untuk menangkap karakteristik lokal dari masing-masing angka.



Gambar 2. Digit sesudah diubah ke citra abu-abu

Seluruh fitur numerik hasil ekstraksi bersama label kelasnya disimpan ke dalam variabel X dan y. Selanjutnya, dataset ini dibagi menjadi dua bagian, yaitu data latih dan data uji, dengan rasio pembagian sebesar 80:20.

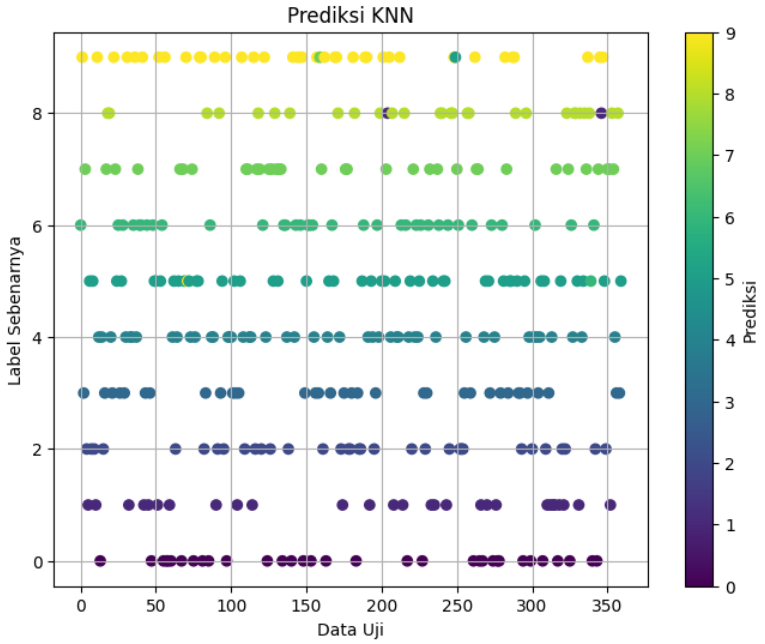
Model *K-Nearest Neighbor* (KNN) kemudian dilatih dengan menggunakan berbagai variasi nilai k untuk mengetahui pengaruhnya terhadap performa klasifikasi. Akurasi dari masing-masing model dicatat dan disimpan dalam sebuah daftar untuk dianalisis. Hasil dari eksperimen tersebut menunjukkan nilai akurasi yang bervariasi tergantung pada nilai k yang digunakan. Berikut hasil evaluasi model KNN:

```
k value: 3
Akurasi: 0.9833333333333333
Label Prediksi (10 data pertama): [6 9 3 7 2 1 5 2 5 2]
k value: 5
Akurasi: 0.9861111111111112
Label Prediksi (10 data pertama): [6 9 3 7 2 1 5 2 5 2]
k value: 7
Akurasi: 0.9888888888888889
Label Prediksi (10 data pertama): [6 9 3 7 2 1 5 2 5 2]
k value: 11
Akurasi: 0.9833333333333333
Label Prediksi (10 data pertama): [6 9 3 7 2 1 5 2 5 2]
k value: 13
Akurasi: 0.9777777777777777
Label Prediksi (10 data pertama): [6 9 3 7 2 1 5 2 5 2]
k value: 17
Akurasi: 0.9833333333333333
Label Prediksi (10 data pertama): [6 9 3 7 2 1 5 2 5 2]
```

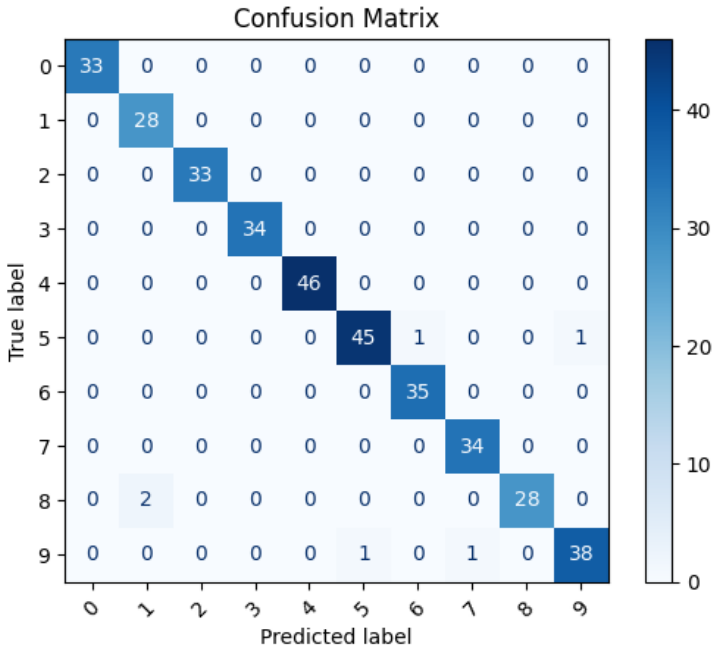
Berdasarkan hasil eksperimen, diperoleh bahwa nilai *best K* dalam algoritma K-Nearest Neighbors (KNN) adalah 7, dengan akurasi prediksi mencapai 0.9889 atau setara dengan 98,89%. Pemilihan nilai *k* dalam algoritma KNN memiliki peran penting dalam menentukan performa klasifikasi, karena nilai tersebut menentukan jumlah tetangga terdekat yang digunakan dalam proses prediksi. Dalam pengujian ini, nilai k = 7 terbukti memberikan akurasi tertinggi jika dibandingkan dengan k = 3 dan k = 5. Nilai *k* yang terlalu kecil,

p-ISSN : 2803-1507 e-ISSN : 2803-1531

seperti k = 3, cenderung menyebabkan *overfitting* karena sangat sensitif terhadap data yang mengandung noise. Sebaliknya, nilai *k* yang terlalu besar dapat menyebabkan *underfitting* karena mengaburkan batas antar kelas, sehingga mengurangi kemampuan model dalam membedakan pola secara tepat [6].



Gambar 3. Grafik Pengaruh Nilai K terhadap Akurasi



Gambar 4. Grafik Visualisasi Confusion Matrix terhadap Model KNN

Nilai k = 7 diduga menjadi titik keseimbangan antara bias dan varians, sehingga model dapat mengenali pola data secara lebih stabil tanpa terlalu bergantung pada outlier atau noise lokal. Beberapa penelitian terdahulu menunjukkan bahwa pada dataset dengan dimensi tinggi atau pola yang kompleks seperti MNIST, nilai k yang berada di antara 5 hingga 10 sering menghasilkan performa terbaik karena mempertimbangkan kedekatan fitur

p-ISSN : 2803-1507 e-ISSN : 2803-1531

dan distribusi kelas yang relatif seimbang [7]. Oleh karena itu, pemilihan nilai k = 7 dalam penelitian ini dianggap sesuai dengan karakteristik data dan sejalan dengan hasil dari studi-studi serupa. Untuk memberikan gambaran yang lebih jelas mengenai performa klasifikasi, dilakukan visualisasi *confusion matrix* seperti pada Gambar 4.

Confusion matrix tersebut menunjukkan bahwa model K-Nearest Neighbors (KNN) memiliki performa yang cukup baik dalam mengklasifikasikan data, namun masih terdapat pola kesalahan klasifikasi. Hal ini terlihat dari adanya beberapa nilai di luar diagonal utama (misalnya pada kelas 8 yang terkadang diklasifikasikan sebagai kelas 0 dan 9, serta kelas 9 yang salah diklasifikasikan sebagai kelas 4 dan 8). Ini mengindikasikan bahwa model KNN mengalami kebingungan antara kelas-kelas tertentu, yang bisa disebabkan oleh kemiripan fitur antar kelas tersebut.

Namun demikian, sistem yang dikembangkan dalam penelitian ini memiliki beberapa keterbatasan yang perlu diperhatikan. Pertama, ukuran dataset yang digunakan relatif kecil, yaitu hanya terdiri dari 500 gambar, sehingga representasi variasi tulisan tangan dalam data tersebut masih terbatas. Hal ini dapat memengaruhi kemampuan generalisasi model terhadap data yang lebih beragam. Kedua, penelitian ini belum menggunakan teknik validasi silang (*cross-validation*) yang lazim digunakan untuk mengurangi kemungkinan overfitting dan memberikan estimasi performa model yang lebih andal. Ketiga, model belum diuji terhadap gambar yang mengandung noise, distorsi, atau variasi pencahayaan, yang pada praktiknya sering ditemukan dalam data tulisan tangan dunia nyata. Keterbatasan ini menunjukkan bahwa performa model dalam kondisi yang lebih kompleks masih belum diketahui secara pasti, dan oleh karena itu, perlu dilakukan pengujian lanjutan dalam penelitian berikutnya [8].

Diskusi

Selain variasi nilai K, keberhasilan pengenalan angka tulisan tangan juga sangat dipengaruhi oleh kualitas data dan metode ekstraksi fitur yang digunakan. Penelitian oleh [9] menunjukkan bahwa zoning efektif dalam menangkap struktur lokal gambar angka, sehingga mampu meningkatkan akurasi klasifikasi KNN secara signifikan. [10] juga menegaskan pentingnya pengolahan awal citra dan pemilihan fitur yang tepat, terutama ketika data berasal dari tulisan tangan individu yang bervariasi. Dengan kombinasi kedua pendekatan tersebut, sistem klasifikasi dapat bekerja secara lebih stabil dan efisien.

Dalam konteks pendidikan dan evaluasi akademik, penerapan KNN untuk klasifikasi jawaban esai berbasis citra telah terbukti dapat meningkatkan efisiensi penilaian [11]. Hal ini juga didukung oleh penelitian [12] yang memanfaatkan zoning sebagai metode ekstraksi untuk memproses data tulisan tangan pada ujian. Pendekatan ini tidak hanya mempercepat proses koreksi, tetapi juga berpotensi mengurangi subjektivitas dalam penilaian manual. Dengan demikian, KNN relevan digunakan dalam sistem evaluasi otomatis di lingkungan pembelajaran daring.

Beberapa studi juga menyoroti keterbatasan KNN dalam menangani data dalam jumlah besar karena beban komputasi saat proses klasifikasi. Hal ini menjadi kendala terutama dalam sistem skala besar yang membutuhkan efisiensi waktu [13]. Untuk mengatasi tantangan ini, pendekatan seperti implementasi KNN pada platform cloud seperti Google Colab dinilai efisien dan praktis [14]. Dengan pemrosesan berbasis cloud, sistem tetap ringan namun mampu menangani komputasi intensif.

Kelemahan lainnya dari KNN adalah sensitivitasnya terhadap noise serta ketidakseimbangan data antar kelas. Ketika distribusi kelas tidak merata atau data mengandung banyak outlier, akurasi model dapat menurun drastis. Dalam hal ini, pendekatan hybrid atau ensemble seperti bagging dan boosting telah digunakan untuk memperkuat akurasi klasifikasi dan meningkatkan ketahanan model terhadap gangguan data [15], [16]. Strategi ini semakin dibutuhkan ketika model diterapkan pada data dunia nyata yang tidak selalu terstruktur rapi seperti dalam dataset MNIST.

Sebagai pembanding, algoritma lain seperti Convolutional Neural Network (CNN) memiliki performa unggul dalam mengenali karakter tulisan tangan karena kemampuannya dalam mengekstraksi fitur otomatis dari citra mentah. Meskipun demikian, CNN membutuhkan pelatihan yang lebih kompleks dan komputasi yang jauh lebih besar dibanding KNN [17]. Dalam sistem yang ditujukan untuk implementasi web atau perangkat terbatas, KNN masih menjadi pilihan yang rasional [18]. Hal ini mengingat sifatnya yang non-parametrik dan mudah diterapkan.

Studi oleh [19] mengaplikasikan KNN dalam klasifikasi tweet, menunjukkan bahwa metode ini juga dapat digunakan dalam pengolahan data media sosial. Hasil penelitian tersebut menunjukkan bahwa KNN dapat

p-ISSN : 2803-1507 e-ISSN : 2803-1531

memberikan akurasi di atas 80% dalam menangani teks singkat seperti cuitan Twitter. Pendekatan ini memperlihatkan fleksibilitas algoritma KNN dalam berbagai domain klasifikasi, baik citra maupun teks. Dengan demikian, KNN tidak hanya terbatas untuk pengenalan angka, tetapi juga untuk klasifikasi berbasis teks yang lebih kompleks.

Penggunaan KNN untuk mendeteksi karakter tulisan tangan dengan pendekatan berbasis Support Vector Machine (SVM) juga telah diteliti. Hasil studi tersebut menunjukkan bahwa meskipun SVM lebih presisi dalam beberapa kasus, KNN tetap kompetitif dalam hal kecepatan dan kesederhanaan [20], [21]. Di sisi lain, validitas hasil model KNN dapat ditingkatkan melalui teknik validasi silang seperti k-fold, yang telah terbukti efektif dalam memberikan evaluasi yang lebih stabil [22]. Teknik ini memungkinkan pembagian data menjadi beberapa subset sehingga model diuji lebih menyeluruh.

Dalam hal optimasi parameter, nilai K yang tepat menjadi kunci penting dalam menentukan performa akhir model. [23] menyarankan bahwa nilai K dalam rentang 5 hingga 10 memberikan hasil yang paling optimal untuk dataset MNIST. Penggunaan fitur kombinasi seperti zoning dan Histogram of Oriented Gradients (HOG) juga dilaporkan dapat meningkatkan performa klasifikasi secara signifikan [24]. Kombinasi ini memungkinkan model menggabungkan informasi bentuk lokal dan arah visual, yang penting untuk membedakan angka-angka yang mirip secara struktur.

Terakhir, penelitian oleh [25] menunjukkan bahwa variasi kondisi pencahayaan dan noise dapat memengaruhi hasil klasifikasi secara substansial. Hal ini menekankan pentingnya preprocessing lanjutan seperti filter citra, normalisasi, dan deteksi noise sebelum fitur diekstraksi. Dalam lingkungan nyata, tantangan seperti kualitas pemindaian dan perangkat input menjadi variabel penting yang memengaruhi output sistem. Oleh karena itu, sistem pengenalan tulisan tangan berbasis KNN perlu diuji lebih lanjut dalam skenario yang lebih kompleks dan dinamis.

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini menunjukkan bahwa algoritma *K-Nearest Neighbors* (KNN) dengan teknik ekstraksi fitur *zoning* mampu mengenali angka tulisan tangan dengan tingkat akurasi tinggi. Hasil terbaik diperoleh pada nilai K = 7 dengan akurasi sebesar 98,89%. Sistem yang dikembangkan menggunakan *Google Colab* dan pustaka Python terbukti efisien, ringan, dan cocok untuk diimplementasikan dalam sistem informasi berbasis web. Pendekatan ini menawarkan solusi praktis untuk digitalisasi angka tulisan tangan di berbagai bidang.

5.2 Saran

Untuk pengembangan lebih lanjut, sistem pengenalan angka tulisan tangan ini dapat ditingkatkan dengan memperluas cakupan dataset, misalnya menggunakan data tulisan tangan dari berbagai individu dengan gaya penulisan yang berbeda, guna meningkatkan kemampuan generalisasi model. Selain itu, integrasi sistem ke dalam aplikasi berbasis web atau mobile dengan antarmuka yang lebih interaktif dapat memperluas pemanfaatannya dalam dunia pendidikan dan administrasi.

Disarankan juga untuk membandingkan performa KNN dengan algoritma klasifikasi lain seperti *Support Vector Machine* (SVM), *Random Forest*, atau versi ringan dari *Convolutional Neural Network* (CNN) agar diperoleh evaluasi yang lebih menyeluruh terhadap efektivitas metode yang digunakan. Penggunaan teknik ekstraksi fitur tambahan selain *zoning*, seperti histogram piksel atau metode statistik lainnya, juga dapat diteliti guna meningkatkan akurasi klasifikasi. Terakhir, pengujian terhadap efisiensi waktu proses di perangkat dengan spesifikasi rendah dapat menjadi pertimbangan penting agar sistem benar-benar siap digunakan dalam skala luas dan beragam kondisi.

DAFTAR PUSTAKA

[1] H. Syahputra, R. Givent, and A. Simanjorang, “Penerapan Algoritma K-Nearest Neighbor (KNN) dalam Pengenalan Pola Tulisan Tangan Angka 0-9,” *Dinamik*, vol. 28, no. 2, p. 2023, 2023, doi: 10.35315/dinamik.v28i2.9360.

[2] V. Oktavia and N. Wijaya, “Pengenalan Tulisan Tangan Huruf Latin Bersambung Menggunakan Local Binary Pattern dan K-Nearest Neighbor,” 2022. doi: 10.14421/jiska.2022.7.3.211-225.

p-ISSN : 2803-1507 e-ISSN : 2803-1531

[3] M. I. Mubarok, C. Informatika, S. Karawang, J. H. Ronggo Waluyo, T. Timur, and I. Karawang, "Penerapan Algoritma K-Nearest Neighbor (KNN) Dalam Klasifikasi Penilaian Jawaban Ujisan Esai," 2023. doi: 10.36040/jati.v7i5.7676.

[4] C. Ramadhona Hassolthine, R. Sahara, F. Asy, S. N. Haq, and S. Abdullah, "Penerapan Algoritma K-Nearest Neighbour (K-NN) sebagai Klasifikasi Tweets pada Twitter," *JOISIE J. Inf. Syst. Informatics Eng.*, vol. 7, no. 2, pp. 2780–6189, 2023, doi: 10.35145/joisie.v7i2.3975.

[5] Z. A. Muhammad, G. Budiman, and S. Saidah, "Penggunaan Compressive Sensing Pada Pengenalan Huruf Dengan Tulisan Tangan Menggunakan Metode K-Nearest Neighbor," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 6, pp. 1237–1246, 2022, doi: 10.25126/jtiik.202295612.

[6] S. Sumarlin, "Implementasi Algoritma K-Nearest Neighbor Sebagai Pendukung Keputusan Klasifikasi Penerima Beasiswa PPA dan BBM Sumarlin STIKOM Uyelindo Kupang," *J. Sist. Inf. Bisnis*, vol. 5, no. 1, pp. 52–62, 2015, doi: 10.21456/vol5iss1pp52-62.

[7] A. Rajput and A. K. Singh, "Handwritten Digit Recognition Accuracy Comparison Using Knn,Cnn And Svm.," *Educ. Adm. Theory Pract.*, vol. 30, no. 2, pp. 638–643, 2024, doi: 10.53555/kuey.v30i2.1676.

[8] A. S. Wijaya, N. Chamidah, and M. M. Santoni, "Pengenalan Karakter Tulisan Tangan Dengan K-Support Vector Nearest Neighbor," *IJEIS (Indonesian J. Electron. Instrum. Syst.*, vol. 9, no. 1, p. 33, 2019, doi: 10.22146/ijeis.38729.

[9] Y. Mo, Z. Guo, R. Zhong, W. Song, and S. Cao, "Urban Functional Zone Classification Using Light-Detection-and-Ranging Point Clouds, Aerial Images, and Point-of-Interest Data," *Remote Sens.*, vol. 16, no. 2, p. 386, 2024, doi: 10.3390/rs16020386.

[10] D. Gagiu and D. Sendrescu, "Detection of Personality Traits Using Handwriting and Deep Learning," *Appl. Sci.*, vol. 15, no. 4, p. 2154, Feb. 2025, doi: 10.3390/app15042154.

[11] T. Matos, W. Santos, E. Zdravevski, P. J. Coelho, I. M. Pires, and F. Madeira, "A Systematic Review of Artificial Intelligence Applications in Education: Emerging Trends and Challenges," *Decis. Anal. J.*, vol. 15, p. 100571, Jun. 2025, doi: 10.1016/j.dajour.2025.100571.

[12] V. Jocovic, B. Nikolic, and N. Bacanin, "Software System for Automatic Grading of Paper Tests," *Electronics*, vol. 12, no. 19, p. 4080, Sep. 2023, doi: 10.3390/electronics12194080.

[13] Q. An, S. Rahman, J. Zhou, and J. J. Kang, "A Comprehensive Review on Machine Learning in Healthcare Industry: Classification, Restrictions, Opportunities and Challenges," *Sensors*, vol. 23, no. 9, p. 4178, 2023, doi: 10.3390/s23094178.

[14] A. R. Khalid, N. Owoh, O. Uthmani, M. Ashawa, J. Osamor, and J. Adejoh, "Enhancing Credit Card Fraud Detection: An Ensemble Machine Learning Approach," *Big Data Cogn. Comput.*, vol. 8, no. 1, p. 6, Jan. 2024, doi: 10.3390/bdcc8010006.

[15] M. M. Zahoor *et al.*, "A New Deep Hybrid Boosted and Ensemble Learning-Based Brain Tumor Analysis Using MRI," *Sensors*, vol. 22, no. 7, p. 2726, Apr. 2022, doi: 10.3390/s22072726.

[16] A. Zizaan and A. Idri, "Evaluating and Comparing Bagging and Boosting of Hybrid Learning for Breast Cancer Screening," *Sci. African*, vol. 23, p. 01989, Mar. 2024, doi: 10.1016/j.sciaf.2023.e01989.

[17] Z. A. Sejuti and M. S. Islam, "A Hybrid CNN–KNN Approach for Identification of COVID-19 with 5-Fold Cross Validation," *Sensors Int.*, vol. 4, p. 100229, Jan. 2023, doi: 10.1016/j.sintl.2023.100229.

[18] E. A. Abioye *et al.*, "Precision Irrigation Management Using Machine Learning and Digital Farming Solutions," *AgriEngineering*, vol. 4, no. 1, pp. 70–103, 2022, doi: 10.3390/agriengineering4010006.

[19] A. Huang, R. Xu, Y. Chen, and M. Guo, "Research on Multi-Label User Classification of Social Media Based on ML-KNN Algorithm," *Technol. Forecast. Soc. Change*, vol. 188, p. 122271, Mar. 2023, doi: 10.1016/j.techfore.2022.122271.

[20] M. Bansal, A. Goyal, and A. Choudhary, "A Comparative Analysis of K-Nearest Neighbor, Genetic, Support Vector Machine, Decision Tree, and Long Short Term Memory Algorithms in Machine Learning," *Decis. Anal. J.*, vol. 3, p. 100071, Jun. 2022, doi: 10.1016/j.dajour.2022.100071.

p-ISSN : 2803-1507 e-ISSN : 2803-1531

[21] A. Rana, P. Vaidya, and G. Gupta, "A Comparative Study of Quantum Support Vector Machine Algorithm for Handwritten Recognition with Support Vector Machine Algorithm," *Mater. Today Proc.*, vol. 56, pp. 2025–2030, Jan. 2022, doi: 10.1016/j.matpr.2021.11.350.

[22] M. S. Uzer, "Deep Learning-Based Classification Consisting of Pre-Trained Models and Proposed Model Using K-Fold Cross-Validation for Pistachio Species," *Appl. Sci.*, vol. 15, no. 8, p. 4516, Apr. 2025, doi: 10.3390/app15084516.

[23] R. F. Muktiadji, M. A. M. Ramli, and A. H. Milyani, "Twin-Delayed Deep Deterministic Policy Gradient Algorithm to Control a Boost Converter in a DC Microgrid," *Electronics*, vol. 13, no. 2, p. 433, 2024, doi: 10.3390/electronics13020433.

[24] A. Taner *et al.*, "Multiclass Apple Varieties Classification Using Machine Learning with Histogram of Oriented Gradient and Color Moments," *Appl. Sci.*, vol. 13, no. 13, p. 7682, 2023, doi: 10.3390/app13137682.

[25] R. A. Lashaki, Z. Raeisi, N. Razavi, M. Goodarzi, and H. Najafzadeh, "Optimized Classification of Dental Implants Using Convolutional Neural Networks and Pre-Trained Models with Preprocessed DATA," *BMC Oral Health*, vol. 25, no. 1, p. 535, 2025, doi: 10.1186/s12903-025-05704-0.