

STUDI ANALISIS PENGENALAN POLA TULISAN TANGAN ANGKA ARABIC (INDIAN) MENGGUNAKAN METODE *K- NEAREST NEIGHBORS* DAN *CONNECTED COMPONENT LABELING*

ANALYSIS STUDY OF PATTERN RECOGNITION ARABIC (INDIAN) HANDWRITING USE THE METHOD OF K-NEAREST NEIGHBORS AND CONNECTED COMPONENT LABELING

Roni Akbar^{*}, Eko Adi Sarwoko

*Email : roniakbar93@gmail.com

Jurusan Ilmu Komputer/Informatika Fakultas Sains dan Matematika Universitas Diponegoro, Semarang

Abstract — Tulisan tangan merupakan hasil menulis, dengan tangan (bukan ketikan). Gaya penulisan setiap orang tidak sama. Salah satu bahasa resmi Perserikatan Bangsa-Bangsa (PBB), bahasa Arab, memiliki penulisan angka yang dikenal sebagai angka *Arabic (Indian)*. Identifikasi ciri-ciri membantu manusia untuk membedakan suatu pola dengan pola lainnya. Pengelompokan pola ini dapat diterapkan kepada mesin untuk tujuan mengenali suatu objek tulisan pada citra. *Connected component labeling* digunakan untuk pemotongan karakter menjadi bagian yang tidak saling terhubung satu sama lain agar mudah dikenali. *K- nearest neighbors* (KNN) adalah metode yang digunakan untuk mencari seberapa besar kecocokan antara citra uji dengan citra latih berdasarkan pada kelas dari tetangga terdekat. Pada studi analisis ini telah dilakukan pengujian dengan menggunakan 100 citra uji. Tiga hasil klasifikasi terbaik dari pengenalan pola tulisan tangan angka *Arabic (Indian)* menggunakan metode *k- nearest neighbors* (KNN) adalah 86% saat nilai $k = 1$, 84% dengan nilai $k = 3$, dan 83% dengan $k = 5$.

Kata kunci: Pengenalan Pola, Tulisan Tangan, Angka Arabic (Indian), *Connected Component Labeling*, *K- Nearest Neighbors*.

Abstract — Handwriting refers to the result of writing by hand (not typed). The writing style of people are not the same. One of the United Nations official languages, Arabic, has a numerical system known as Arabic (Indian) numeral. The identification of feature help humans to be able distinguish the patterns. The grouping patterns can be applied to the machine for recognizing object in the image. Connected component labeling is used for separating characters to be easily recognizable. K- nearest neighbors is used to find the similarity value between query image and template images are based on the nearest neighbors class. This analytical study was tested using 100 test images. The top three classification results of Arabic (Indian) handwritten recognition use k- nearest neighbors (KNN) are 86% when $k = 1$, 84% when $k = 3$, and 83% with $k = 5$.

Keywords: Pattern recognition, handwriting, Arabic Number (Indian), Connected Component Labeling, K- Nearest Neighbors.

I. PENDAHULUAN

Tulisan tangan merupakan hasil menulis, barang yang ditulis ataupun cara menulis dengan tangan (bukan ketikan) [1]. Manusia memiliki kemampuan dalam pengenalan tulisan tangan melalui proses pembelajaran, namun ada beberapa kendala yang mungkin ditemukan saat pengenalan, sehingga tulisan tangan menjadi sulit dibaca.

Perserikatan Bangsa-Bangsa (PBB) menetapkan ada 6 bahasa sebagai bahasa resmi (*official languages*) di lembaga internasional tersebut, yakni China, Inggris, Prancis, Rusia, Spanyol, dan Arab [2]. Bahasa Arab, walaupun menggunakan sistem penomoran yang sama, namun bentuk karakter angka yang digunakan berbeda dari angka modern yang dipakai saat ini. Angka dalam bahasa Arab dikenal sebagai angka *Arabic (Indian)* [3].

Pengenalan tulisan tangan (*handwriting recognition*) adalah kemampuan komputer untuk dapat menerima dan menafsirkan input citra tulisan tangan yang dapat dimengerti dari sumber seperti dokumen kertas, foto, layar sentuh dan perangkat lainnya [4].

Pola merupakan entitas yang terdefinisi dan dapat diidentifikasi melalui ciri-cirinya. Ciri-ciri tersebut digunakan untuk membedakan suatu pola dengan pola lainnya. Ciri yang bagus adalah ciri yang memiliki daya pembeda yang tinggi, sehingga pengelompokan pola berdasarkan ciri yang dimiliki dapat dilakukan dengan keakuratan yang tinggi. Ciri pada suatu pola diperoleh dari hasil pengukuran terhadap objek uji [5].

Template matching adalah salah satu metode yang dapat digunakan dalam tahapan terakhir pengenalan pola, yakni tahapan klasifikasi. *Template matching* ini bekerja dengan cara mencocokkan tiap-tiap bagian dari suatu citra uji dengan citra yang menjadi *template* [6]. *Classifier k-nearest neighbors* (KNN) membantu menentukan kelas di mana k tetangga terdekat untuk melakukan klasifikasi sebuah objek [7].

Segmentasi digunakan untuk memisahkan karakter yang ada pada citra agar tidak bergabung satu sama lain. *Connected component labeling* merupakan metode segmentasi untuk memisahkan objek dan tidak terpengaruh pada kemiringan objek [8].

Pada tulisan ini dikaji studi analisis pengenalan pola tulisan tangan menggunakan metode *k-nearest neighbors* dan *connected component labeling*. Diharapkan studi analisis ini dapat mengenali dan mengklasifikasikan nilai tulisan tangan berupa objek dari bahasa negara lain, yakni angka *Arabi (Indian)* dengan metode *k-nearest neighbors*.

II. TINJAUAN PUSTAKA

A. Metode Pengembangan

Studi analisis ini menggunakan metode pengembangan model sekuensial linier. Tahap-tahap utama dari model ini memetakan kegiatan-kegiatan pengembangan dasar yaitu seperti di bawah ini [9].

1) Analisis

Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak. Untuk memahami sifat program yang dibangun, perancang perangkat lunak (analisis) harus memahami domain informasi, tingkah laku, unjuk kerja, dan antarmuka (*interface*) yang diperlukan.

2) Design

Proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda; struktur data, arsitektur perangkat lunak, representasi *interface*, dan detail (algoritma) prosedural. Proses desain menerjemahkan syarat/ kebutuhan ke dalam sebuah representasi perangkat lunak yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode.

3) Code

Desain harus bisa diterjemahkan ke dalam bentuk mesin yang bisa dibaca. Langkah pembuatan kode melakukan tugas ini. Jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat dilakukan secara mekanis.

4) Test

Proses pengujian berfokus pada logika internal perangkat lunak, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional yaitu mengarahkan pengujian untuk menemukan kesalahan-kesalahan dan memastikan bahwa *input* yang dibatasi memberikan hasil aktual yang sesuai dengan hasil yang dibutuhkan.

B. Pengolahan Citra

Pengolahan citra adalah pemrosesan citra, khususnya dengan menggunakan komputer, menjadi citra yang kualitasnya baik. Umumnya, operasi-operasi pada pengolahan citra diterapkan pada citra bila [5]:

- perbaikan atau memodifikasi citra perlu dilakukan untuk meningkatkan kualitas penampakan atau untuk menonjolkan beberapa aspek informasi yang terkandung di dalam citra
- elemen di dalam citra perlu dikelompokkan, dicocokkan atau diukur, dan
- sebagian citra perlu digabung dengan bagian citra yang lain.

C. Preprocessing Image

Tahapan *preprocessing* pada studi analisis ini antara lain di bawah ini.

1) Grayscale

Grayscale merupakan proses pengolahan citra untuk mengubah citra berwarna yang mempunyai nilai matrik masing-masing r , g , dan b menjadi citra *grayscale*. Berdasarkan pengujian sensitivitas mata manusia terhadap warna, *grayscale* dilakukan dengan *weighted method*, seperti dalam persamaan (1) [10].

$$Y = ((0.3 * r) + (0.59 * g) + (0.11 * b)) \quad (1)$$

Metode ini mengurangi kontribusi *red* dan *blue* yang muncul, dan memberikan kontribusi yang lebih besar pada *green*. Nilai kontribusi yang diberikan pada masing-masing warna adalah sebesar 30% *red*, 59% *green*, 11% *blue*.

2) Thresholding

Thresholding merupakan konversi dari citra hitam-putih ke citra biner dilakukan dengan operasi pengambangan. Operasi pengambangan citra mengelompokkan nilai derajat keabuan setiap piksel ke dalam 2 kelas, hitam dan putih. Pengambangan secara global, setiap piksel yang terdapat di dalam citra dipetakan menjadi dua nilai, yaitu 1 atau 0 dengan fungsi pengambangan dapat dilihat pada persamaan (2) [5].

$$f_B(i, j) = \begin{cases} 1, & f_g(i, j) < T \\ 0, & \text{lainnya} \end{cases} \quad (2)$$

3) Filtering

Filtering sebagai teknik untuk memodifikasi atau meningkatkan mutu citra. Sebagai contoh, teknik ini dapat memfilter sebuah citra untuk mengutamakan fitur tertentu dan membuang fitur yang lain (derau). Operasi *filtering* secara matematis ditunjukkan pada persamaan (3) [11].

$$g(x, y) = \text{median}(f(y-1, x-1), f(y-1, x), f(y-1, x+1), f(y, x-1), f(y, x), f(y, x+1), f(y+1, x-1), f(y+1, x), f(y+1, x+1)) \quad (3)$$

4) Auto cropping

Autocropping merupakan proses pemotongan citra pada koordinat tertentu pada area citra. Untuk memotong bagian dari citra digunakan dua koordinat, yaitu koordinat awal yang merupakan awal koordinat bagi citra hasil pemotongan dan koordinat akhir yang merupakan titik akhir koordinat dari citra hasil pemotongan. Sehingga membentuk bangun segi empat yang mana tiap-tiap piksel yang ada pada area koordinat tertentu dan disimpan dalam citra yang baru [5].

5) Scalling

Sebuah operasi geometri yang memberikan efek memperbesar atau memperkecil ukuran citra input sesuai dengan variabel penskalaan citranya. Ukuran baru hasil penskalaan didapat melalui perkalian antara ukuran citra input dengan variabel penskalaan [12].

6) Invert/Logical NOT

Invers atau logika *NOT* pada pengolahan citra memberikan keluaran hasil berupa kebalikan dari citra input. Penggunaan logika *NOT* pada citra biner sangat sederhana, apabila citra input bernilai 1 maka keluaran bernilai 0, sebaliknya input bernilai 0 menghasilkan keluaran bernilai 1 [12].

D. Connected Component Labeling

Connected component labelling merupakan teknik yang juga bisa digunakan untuk mengklasifikasikan *region* atau objek dalam citra digital. Teknik ini memanfaatkan teori *connectivity* piksel pada citra. Piksel-piksel dalam *region* disebut *connected* (ada konektivitasnya atau *connectivity*) bila mematuhi aturan *adjacency* atau aturan kedekatan piksel. Aturan kedekatan piksel ini memanfaatkan sifat ketetanggaan piksel. Dengan demikian piksel-piksel yang dikatakan *connected* pada dasarnya memiliki sifat *adjacency* satu sama lain karena mereka masih memiliki hubungan *neighborhood* atau ketetanggaan [12].

E. Template Matching

Template matching adalah suatu teknik untuk mengukur kesamaan dari dua gambar digital, untuk menentukan apakah kedua gambar sama atau tidak [13]. Teori ini didukung oleh pendapat Darma Putra yang menyatakan bahwa *template matching* merupakan proses mencari suatu objek (*template*) pada keseluruhan objek yang berada dalam suatu citra. *Template* dibandingkan dengan keseluruhan objek tersebut dan bila *template* cocok (cukup dekat) dengan suatu objek yang belum diketahui pada citra tersebut maka objek tersebut ditandai sebagai *template* [12].

F. K- Nearest Neighbors

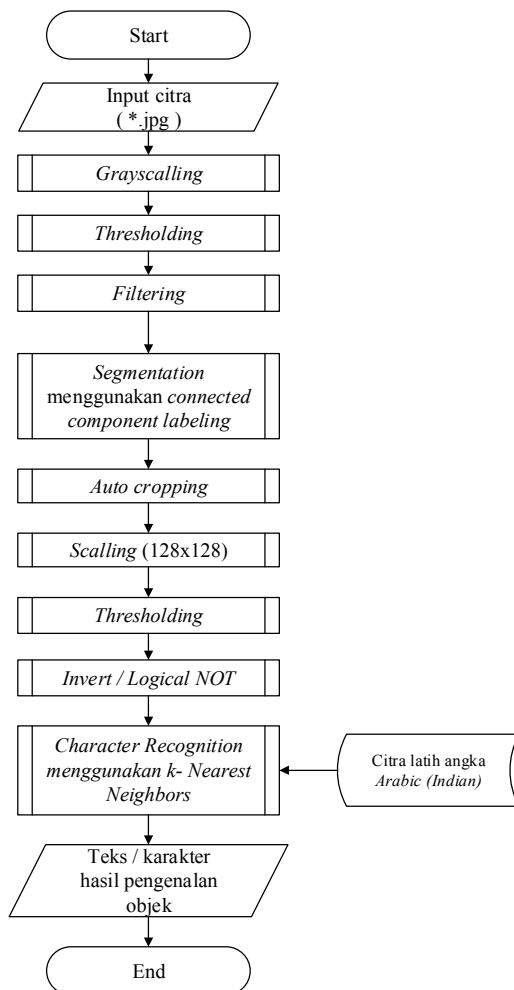
Ide dasar dari klasifikasi *k- nearest neighbours* (KNN) cukup sederhana, yaitu klasifikasi yang didasarkan pada kelas dari tetangga terdekat. Klasifikasi ini memungkinkan untuk mengambil lebih dari satu tetangga, teknik ini biasa disebut klasifikasi *k- nearest neighbors* (KNN) di mana *k* tetangga terdekat digunakan untuk menentukan kelas [7]. Melalui pendekatan yang lebih modern, klasifikasi KNN, mencari sebuah kelompok dari *k* objek di dalam *training set* yang paling dekat dengan objek yang diuji dan berdasarkan pada dominansi kelas di lingkungannya. Ada tiga elemen kunci dalam pendekatan ini yaitu sebuah kelompok objek berlabel, sebuah jarak atau persamaan

matematika untuk menghitung jarak antara objek-objek, dan nilai k (nilai yang menandakan banyaknya tetangga) [14].

III. ANALISIS DAN RANCANGAN

A. Proses Kerja Aplikasi

Studi analisis pengenalan pola tulisan tangan angka *Arabic (Indian)* menggunakan metode *k-nearest neighbors* (KNN) dan *connected component labeling* digunakan untuk mengenali dan mengklasifikasikan kelas angka modern dari angka *Arabic (Indian)* citra tulisan tangan. Pada pelaksanaan studi analisis ini, diperlukan analisis kebutuhan yang jelas sebagai tujuan utama agar memenuhi harapan dan rencana yang telah ditetapkan. Proses kerja aplikasi pada studi analisis ini terdiri dari beberapa langkah. Secara umum, proses keseluruhan aplikasi dapat dilihat pada Gambar-1.



Gambar-1. Diagram studi analisis pengenalan pola tulisan tangan angka *Arabic (Indian)* secara umum.

Citra *input* merupakan citra yang diuji. Pada citra *input* dilakukan proses perbaikan citra dengan *preprocessing*. Tahap pertama proses *preprocessing* dalam aplikasi ini adalah *grayscale* yang berguna untuk merubah citra berwarna menjadi citra keabuan. Tahap kedua yakni *thresholding*, proses ini mengubah citra keabuan menjadi citra biner. Tahap selanjutnya, proses *filtering* yang dapat memodifikasi citra dengan meningkatkan kualitas citra dan menghilangkan derau yang ada. Setelah citra uji di-*preprocessing*, *user* dapat memilih secara manual batas objek yang ingin dikenali. Batasan objek kemudian disegmentasi dengan tujuan agar karakter pada citra dapat dikenali satu persatu. Segmentasi menggunakan metode *connected component labeling*.

Objek yang dipisahkan kemudian ditampilkan pada *interface* aplikasi dan masuk pada *axes GUI*. Pada objek yang telah berada pada *axes GUI* dikenali proses *auto cropping*. Tujuan dari proses *auto cropping* ini untuk memotong piksel yang tidak termasuk objek. Selanjutnya, *scaling*, proses ini berguna untuk menyamakan ukuran citra setelah *auto cropping* selesai dilakukan.

Ukuran citra disamakan menjadi 128x128 piksel, sesuai dengan ukuran citra pada direktori penyimpanan. Setelah proses *scaling*, nilai intensitas piksel pada citra pun berubah. Sehingga perlu dilakukan proses *thresholding* kembali untuk mendapatkan nilai citra biner. Proses *thresholding* yang digunakan sama dengan proses *thresholding* setelah citra dikenali proses *grayscale*. Kemudian citra yang telah di-*thresholding* dikenali proses logika *NOT*, yang membalik warna citra hitam menjadi citra berwarna putih, dan sebaliknya.

Tahap akhir, pengenalan pola citra dilakukan dengan mengklasifikasikan jarak *euclidean* terdekat antara citra uji dan citra latihan menggunakan *classifier k- nearest neighbors* (KNN) sesuai nilai k yang ditentukan *user*.

B. Segmentasi Karakter

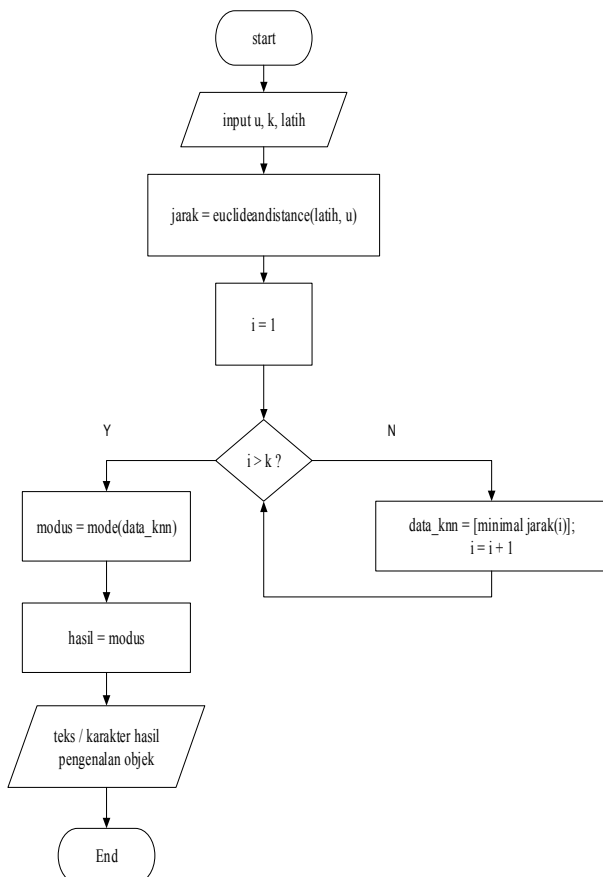
Pada proses segmentasi karakter ini dilakukan menggunakan metode *connected component labeling* pada citra angka *Arabic (Indian)* yang telah melalui proses negasi menjadi citra biner. Masing-masing *region* (kandidat yang menjadi karakter) memiliki label yang berbeda, sehingga dapat dilakukan pemotongan karakter menjadi bagian yang tidak saling terhubung satu sama lain agar mudah dikenali.

Penandaan komponen terhubung dilakukan dengan memeriksa suatu citra, piksel per-piksel (dari

kiri ke kanan dan atas ke bawah) untuk mengidentifikasi area piksel terhubung yaitu suatu area dari piksel berbatasan yang memiliki nilai intensitas sama atau nilai intensitas berada dalam suatu himpunan V , menghasilkan suatu *labeled image* dimana nilai tiap piksel-nya adalah nilai label dari *connected component*-nya. *Image* dipisahkan dengan memotong piksel sesuai label yang dimiliki.

C. Pengenalan Karakter

Pada proses ini dilakukan pengenalan terhadap angka *Arabic (Indian)* yang sudah terpisah melalui proses segmentasi. Masing-masing karakter dicocokkan menggunakan metode *k-nearest neighbors* (KNN), yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada KNN. *Classifier* tidak menggunakan model apapun untuk dicocokkan dan hanya berdasarkan pada memori. Data citra latih pada KNN yang digunakan adalah 100 pola yang mewakili karakter *numerik* (0 sampai 9) *Arabic (Indian)*. Detail alur pengenalan karakter menggunakan *k-nearest neighbors* (KNN) dapat dilihat pada Gambar-2.



Gambar-2. Flowchart proses *character recognition* menggunakan *k-nearest neighbours* (KNN).

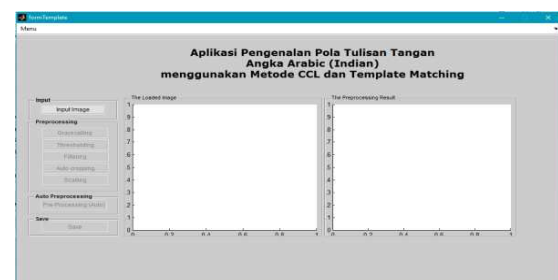
IV. HASIL DAN PEMBAHASAN

A. Implementasi

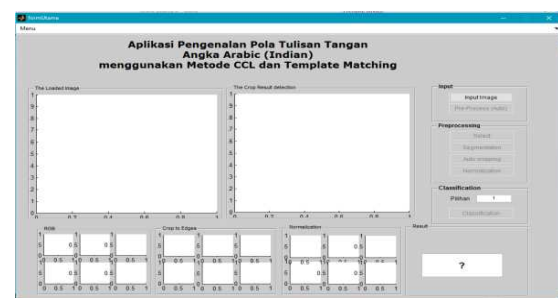
Studi analisis pengenalan pola tulisan tangan angka *Arabic (Indian)* menggunakan metode *k-nearest neighbors* (KNN) dan *connected component labeling* diimplementasikan menggunakan bahasa pemrograman MATLAB dan *database* Microsoft Access. Implementasi dapat dilihat pada Gambar-3 sampai Gambar-10.



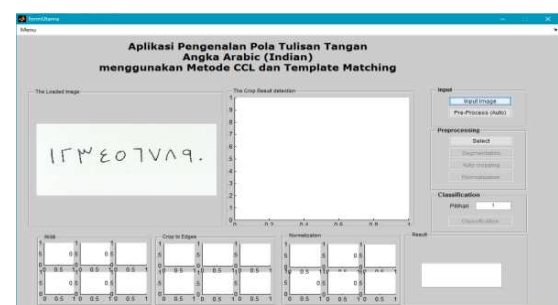
Gambar-3. Antarmuka halaman awal.



Gambar-4. Antarmuka form template.



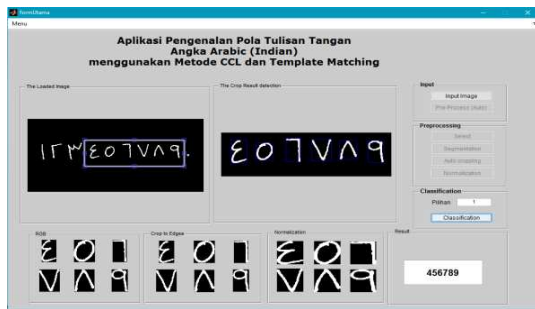
Gambar-5. Antarmuka form utama.



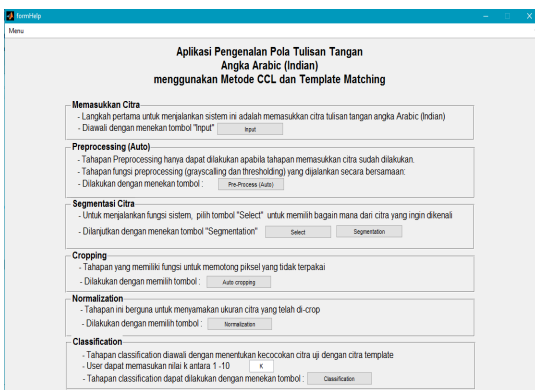
Gambar-6. Antarmuka form utama setelah proses input.



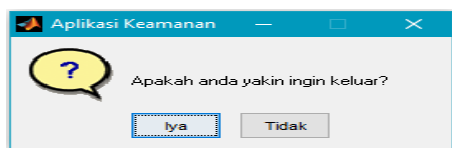
Gambar-7. Antarmuka form utama setelah proses normalisasi citra.



Gambar-8. Antarmuka form utama menampilkan hasil klasifikasi.



Gambar-9. Antarmuka form help.



Gambar-10. Antarmuka kotak dialog "Exit".

B. Pengujian

Hasil pengujian klasifikasi pola tulisan tangan angka Arabi (Indian) untuk angka sample 0 sampai angka sample 9, ditunjukkan oleh Tabel-1, Tabel-2 dan Tabel-3. Tabel berisi hasil klasifikasi angka Arabi (Indian) terbaik, dengan nilai $k=1$, $k=3$, dan $k=5$. Akurasi pengenalan angka Arabi (Indian) diperoleh dengan persamaan (4).

$$\text{Akurasi} = \frac{\text{banyaknyakarakterdikenali}}{\text{banyaknyakarakteruji}} * 100\% \quad (4)$$

Tabel-1. Hasil pengujian klasifikasi angka Arabi (Indian) dengan nilai $k=1$.

Nama Sampel	0	1	2	3	4	5	6	7	8	9	Success (%)
0	10	0	0	0	0	0	0	0	0	0	100
1	1	8	0	0	0	0	1	0	0	0	80
2	0	0	9	1	0	0	0	0	0	0	90
3	0	0	2	8	0	0	0	0	0	0	80
4	0	0	1	0	9	0	0	0	0	0	90
5	0	0	0	0	3	7	0	0	0	0	70
6	0	0	1	0	0	0	9	0	0	0	90
7	0	1	0	0	0	0	0	9	0	0	90
8	0	0	0	0	0	0	1	0	9	0	90
9	0	0	0	0	1	0	1	0	0	8	80
Total Akurasi Pengenalan Angka Arabi (Indian) = 86%											
Target Class											

Tabel-2. Hasil pengujian klasifikasi angka Arabi (Indian) dengan nilai $k=3$.

Nama Sampel	0	1	2	3	4	5	6	7	8	9	Success (%)
0	10	0	0	0	0	0	0	0	0	0	100
1	1	8	0	0	0	0	1	0	0	0	80
2	0	0	10	0	0	0	0	0	0	0	100
3	0	0	3	7	0	0	0	0	0	0	70
4	0	0	1	0	9	0	0	0	0	0	90
5	0	0	0	0	2	8	0	0	0	0	80
6	0	0	1	0	0	0	9	0	0	0	90
7	0	1	0	0	0	0	0	9	0	0	90
8	0	0	1	0	1	0	0	0	8	0	80
9	0	0	0	0	1	0	3	0	0	6	60
Total Akurasi Pengenalan Angka Arabi (Indian) = 84%											
Target Class											

Tabel-3. Hasil pengujian klasifikasi angka Arabi (Indian) dengan nilai $k=5$.

Nama Sampel	0	1	2	3	4	5	6	7	8	9	Success (%)
0	10	0	0	0	0	0	0	0	0	0	100
1	1	7	0	0	1	0	1	0	0	0	70
2	0	0	10	0	0	0	0	0	0	0	100
3	0	0	1	9	0	0	0	0	0	0	90
4	0	0	1	0	8	0	0	1	0	0	80
5	0	0	0	0	3	7	0	0	0	0	70
6	0	0	1	0	0	0	9	0	0	0	90
7	0	0	0	0	0	0	1	9	0	0	90
8	0	0	2	0	0	0	0	0	8	0	80
9	0	0	0	0	1	0	3	0	0	6	60
Total Akurasi Pengenalan Angka Arabi (Indian) = 83%											
Target Class											

Berdasarkan pengujian terhadap 10 (sepuluh) data sampel sejumlah seratus citra, didapatkan tiga hasil pengenalan terbaik dengan nilai $k=1$, $k=3$, dan $k=5$. Akurasi pengenalan angka Arabi (Indian) dari ketiga nilai k tersebut, secara berturut-turut sebesar 86%, 84%, dan 83%. Besarnya akurasi ini dipengaruhi oleh kualitas citra tulisan tangan angka Arabi (Indian), dan metode *template matching* yang digunakan untuk menghitung kecocokan piksel citra latih dengan citra uji. Pada studi analisis ini, penulis menemukan kesalahan klasifikasi yang ditemukan pada pengenalan angka 9. Angka 9 pada beberapa

uji coba diklasifikasikan sebagai angka 6. Contohnya, pada pengujian dengan nilai $k=3$ dan $k=5$, persentase angka 9 secara tepat dikenali sebesar 60%, 30% dikenali sebagai angka 6, dan 10% dikenali sebagai angka 4. Pada citra latihan angka *Arabic (Indian)* 6 dan 9 ditemukan adanya kemiripan bentuk, hal ini menjadi salah satu faktor yang menimbulkan jarak *euclidean* citra uji 9 lebih dekat dengan citra uji 6 pada nilai $k=3$ dan $k=5$.

V. PENUTUP

A. Kesimpulan

Penggunaan metode *connected component labeling* pada studi analisis ini dapat diandalkan, ditunjukkan oleh elemen citra yang dipilih dengan batas secara manual dapat terpisah sebagai elemen mandiri sehingga dapat diklasifikasikan oleh *k-nearest neighbors* (KNN). Dari tingkat akurasi yang diperoleh menunjukkan metode *k-nearest neighbors* dan *connected component labeling* dapat digunakan untuk melakukan pengenalan pola tulisan tangan angka *Arabic (Indian)*.

B. Saran

Saran yang dapat dilaksanakan untuk pengembangan lebih lanjut adalah dengan menggunakan perhitungan lain selain *euclidean distance* untuk menghitung kemiripan antara citra uji dan citra latihan, seperti *city block distance*, *manhattan distance*, atau lainnya. Menggunakan citra dan batas manual yang dinamis, sehingga data latihan dan penyeleksian objek dapat lebih beragam. Penambahan fitur kompresi citra, sehingga dapat memasukkan citra uji hasil dari pengambilan melalui kamera beresolusi tinggi. Menambahkan fitur agar aplikasi dapat mengenali objek lebih dari dua digit seperti puluhan, ratusan, ribuan, dan lain-lain.

DAFTAR PUSTAKA

- [1] KBBI [Online]. 2012. <http://kbbi.web.id/tulis>
- [2] UN Official Languages. 2014. <http://www.un.org/en/sections/about-un/official-languages/index.html>
- [3] Mahmoud, S. A. & Olatunji, S. O. Automatic Recognition of Off-line Hand-written Arabic (Indian) Numerals Using Support Vector and Extreme Learning Machines. *International Journal of Imaging (IJI)*. 2009; 2.
- [4] Handariningsih, R. P. *Application of Recognition and Analysis of Handwriting Character Using Freeman Chain Code Method*. 2011. <http://library.gunadarma.ac.id>
- [5] Munir, R., *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Bandung: Informatika. 2004.
- [6] Prayudi, Y. & Wardhana, A. W. *Penggunaan Metode Template Matching untuk Identifikasi Kecacatan Pada PCB*. s.l. Seminar Nasional Aplikasi Teknologi Informasi. 2008.
- [7] Cunningham, P. & Delany, S. J., "k-Nearest Neighbour Classifier", *Issue Technical Report UCD-CSI-4*. 2007.
- [8] Mardiana, T., (2011), Pengenalan Plat Nomor Kendaraan Menggunakan Metode Connected Labeling Dan K-Nearest Neighbor. [Online] Available at: <http://www.academia.edu/5432594/>
- [9] Pressman, R. *S.Rekayasa Perangkat Lunak*. Yogyakarta: Penerbit Andi. 2002.
- [10] Solomon, D., *The Computer Graphics Manual2*. New York: Springer. 2011.
- [11] Kadir, A. & Susanto, A., *Teori dan Aplikasi Pengolahan Citra*. Yogyakarta: Penerbit Andi. 2013.
- [12] Putra, D. *Pengolahan Citra Digital*. Yogyakarta: Penerbit Andi. 2010.
- [13] Brunelli, R. Technology of Vision Research Unit. 2008. [Online] Available at: http://tev.fbk.eu/TM/html/PENGENALAN_PLAT_NOMOR_KENDARAAN_MENGGUNAKAN_METODE_CONNECTED_COMPONENT_LABELING_DAN_K-NEAREST_NEIGHBOR.
- [14] Wu, X., Kumar, V. & Quinlan, J. R. *Top 10 Algorithm in Data Mining*. 2007; XIV: 1-37.