ARTIKEL

SISTEM PENGENALAN POLA TULISAN TANGAN HURUF HIJAIYAH DENGAN MENGGUNAKAN METODE K-NEAREST NEIGHBORS

Oleh:

MUKHAMAD KHOIRUL ZAMRONI

13.1.03.02.0337

Dibimbing oleh :

1. **Intan Nur Farida, M.Kom.**
2. **Julian Sahertian, S.Pd., M.T.**

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS NUSANTARA PGRI KEDIRI

TAHUN 2017

SURAT PERNYATAAN
ARTIKEL SKRIPSI TAHUN 2017

Yang bertanda tangan di bawah ini:

Nama Lengkap	: Mukhamad Khoirul Zamroni
NPM	: 13.1.03.02.0337
Telepun/HP	: 081315584644
Alamat Surel (Email)	: khoirulz19@gmail.com
Judul Artikel	: SISTEM PENGENALAN POLA TULISAN TANGAN HURUF HIJAIYAH DENGAN MENGGUNAKAN METODE K-NEAREST NEIGHBORS.
Fakultas – Program Studi	: Teknik – Teknik Informatika
Nama Perguruan Tinggi	: Universitas Nusantara PGRI Kediri
Alamat Perguruan Tinggi	: Jl. K.H. Ahmad Dahlan No.76, Mojoroto, Kediri, Jawa Timur 64112, Indonesia.

Dengan ini menyatakan bahwa :

a. artikel yang saya tulis merupakan karya saya pribadi (bersama tim penulis) dan bebas plagiarisme;
b. artikel telah diteliti dan disetujui untuk diterbitkan oleh Dosen Pembimbing I dan II.

Demikian surat pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari ditemukan ketidaksesuaian data dengan pernyataan ini dan atau ada tuntutan dari pihak lain, saya bersedia bertanggungjawab dan diproses sesuai dengan ketentuan yang berlaku.

Mengetahui		Kediri, 03 Agustus 2017
Pembimbing I	Pembimbing II	Penulis,
Intan Nur Farida, M.Kom. NIDN. 0704108701	Julian Sahertian, S.Pd., M.T. NIDN. 0707079001	Mukhamad Khoirul Zamroni 13.1.03.02.0337

SISTEM PENGENALAN POLA TULISAN TANGAN HURUF HIJAIYAH DENGAN MENGGUNAKAN METODE K-NEAREST NEIGHBORS

Mukhamad Khoirul Zamroni
13.1.03.02.0337
Fakultas Teknik – Prodi Teknik Informatika
E-mail : khoirulz19@gmail.com
Intan Nur Farida, M.Kom. dan Julian Sahertian, S.Pd., M.T.
UNIVERSITAS NUSANTARA PGRI KEDIRI

ABSTRAK

Huruf hijaiyah merupakan huruf-huruf yang digunakan sebagai dasar pembelajaran membaca Al-Quran, Penulisan huruf hijaiyah memiliki kesulitan tersendiri dibandingkan dengan huruf latin biasa, dikarenakan huruf hijaiyah memiliki bentuk-bentuk khusus yang setiap huruf mempunyai bentuk tulisan yang hampir sama. Hal ini yang biasanya membuat bingung terutama bagi anak-anak yang masih baru belajar huruf hijaiyah.

Rumusan masalah dalam penelitian ini adalah (1) Bagaimana merancang sistem pengenalan pola tulisan tangan huruf hijaiyah dengan menggunakan metode KNN? (2) Bagaimana performa sistem pengenalan pola tulisan tangan huruf hijaiyah dengan menggunakan metode KNN.

Tujuan dari penelitian ini adalah (1) Untuk mengetahui bagaimana cara merancang sistem pengenalan pola tulisan tangan huruf hijaiyah dengan menggunakan metode KNN. (2) Untuk mengetahui bagaimana performa dari sistem pengenalan pola tulisan tangan huruf hijaiyah dengan menggunakan metode KNN.

Penelitian ini menggunakan metode *k-nearest neighbors* dengan menggunakan dataset berupa tulisan tangan dari siswa Sekolah Dasar pada sebuah lembar kertas yang telah ditentukan dan dijadikan gambar digital dengan bantuan alat scanner. Data yang diambil akan dikonversi ke bentuk *grayscale* dan ke deteksi tepi *sobel*. Hasil dari konversi dibagi menjadi partisi. selanjutnya dihitung menggukan metode *k-nearest neighbors* untuk hasil identifikasi pengenalan pola.

Kesimpulan dari penelitian ini adalah dapat dibangunnya sebuah aplikasi yang bisa mengenali tulisan tangan huruf hijaiyah seseorang. *K-nearest neighbors* dapat digunakan sebagai metode pengenalan pola tulisan tangan huruf hijaiyah seseorang, akurasi yang didapatkan mencapai 79% dengan menggunakan data *training* sebanyak 196.

Kata Kunci : Huruf hijaiyah, Pengenalan pola, *K-nearest neighbors*.

I. LATAR BELAKANG

Seiring dengan berkembangan zaman yang secara tidak disadari diikuti oleh perkembangan teknologi sangat banyak memberi pengaruh terhadap perkembangan ilmu pengetahuan salah satunya adalah dalam hal pengenalan pola. Pengenalan pola adalah suatu ilmu untuk mengklasifikasikan atau menggambarkan sesuatu berdasarkan pengukuran kuantitatif fitur (ciri) atau sifat utama dari suatu obyek (Putra, 2010). Prinsip kerja pengenalan pola adalah dengan membandingkan kemiripan suatu data testing berdasarkan informasi yang sudah pernah diperoleh dari data training.

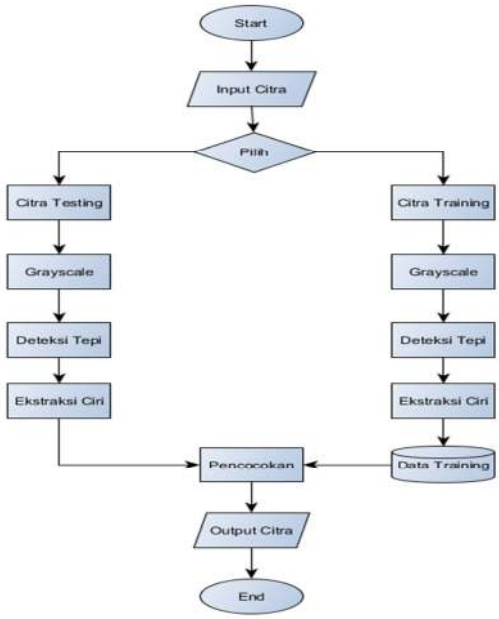
Huruf adalah bagian terkecil dari sebuah kalimat, untuk mendapatkan sebuah informasi dari suatu kalimat perlu dibutuhkan adanya penulisan huruf yang jelas dan mudah dipahami. Huruf hijaiyah merupakan huruf-huruf yang digunakan dalam penyusunan kata pada Al-Qur'an dan sebagai dasar pembelajaran membaca Al-Quran. Penulisan huruf hijaiyah memiliki kesulitan tersendiri dibandingkan dengan huruf latin biasa, dikarenakan huruf hijaiyah memiliki bentuk-bentuk khusus yang setiap huruf mempunyai bentuk tulisan yang hampir sama. Hal ini yang menjadikan komputer susah untuk mengenali tulisan tangan huruf *Hijaiyah* seseorang.

Berbagai penelitian dalam pengenalan pola huruf telah banyak dilakukan oleh para peneliti. Dan telah banyak juga metodologi yang telah dibangun untuk pengenalan pola huruf. Sebelumnya Fragil Putra (2015) telah meneliti dan didalam penelitiannya menjelaskan tentang Pengenalan Pola Huruf Hijaiyah dengan menggunakan metode *Principal Components Analysis (PCA)* dan *Euclidean Distance* untuk mencari nilai kedekatan, didalam penelitian tersebut didapat tingkat akurasi tertinggi yaitu 62%, tingkat akurasi dipengaruhi oleh nilai *threshold* (nilai ambang batas keabuan) dan juga perbandingan jumlah data *training* dan data *testing* yang digunakan.

Oleh karena itu maka penulis tertarik mencoba menyelesaikan masalah diatas dengan cara merancang sebuah sistem pengenalan pola tulisan tangan huruf hijaiyah seseorang menggunakan metode deteksi tepi *sobel* dan *K-Nearest Neighbors* (KNN) dengan harapan bisa menghasilkan tingkat akurasi yang lebih baik dari penelitian sebelumnya.

II. METODE

A. *Flowchart* Sistem



Gambar 1: *Flowchart* Sistem

Keterangan *Flowchart*

1. Start, mulainya sistem berjalan.
2. *Input* Citra, menginputkan citra gambar huruf hijaiyah sesuai prosesnya yaitu *testing* atau *training*.

3. *Grayscale*, proses merubah citra warna menjadi citra keabuan.
4. Deteksi tepi, proses merubah citra keabuan menjadi citra tepi untuk memperoleh bagian detail dari citra menggunakan operator *Sobel.*
5. Ekstraksi ciri, proses merubah hasil citra deteksi tepi sobel kedalam partisi 3x3.
6. Citra *training* disimpan ke data *training* dan selanjutnya.
7. Pencocokan, proses identifikasi citra *testing* yang dicocokkan dengan data *training* menggunakan metode KNN.
8. Output, hasil identifikasi keluar berupa teks bacaan huruf hijaiyah.
9. End, program berhenti.

B. *Grayscale*

Menurut Ahmad (2015) Citra *grayscale* memiliki pengertian sebagai berikut:

> Citra warna *grayscale* menggunakan warna tingkatan warna abu-abu. Warna abu-abu merupakan satu-satunya warna pada ruang *RGB* dengan komponen merah, hijau, dan biru mempunyai intensitas yang sama. Banyaknya warna yang ada tergantung pada jumlah *bit* yang disediakan di memori untuk menampung kebutuhan warna ini.

Untuk mengubah citra berwarna yang mempunyai nilai matrik masing-masing R, G dan B menjadi citra *grayscale* dengan nilai X, maka konversi dapat dilakukan dengan mengambil rata-rata dari nilai R, G dan B. Berikut ini adalah persamaan untuk menghitung citra *grayscale*:

$$X = \frac{R+G+B}{3}$$

Keterangan:

X = Nilai intensitas citra *grayscale*

R = Nilai intensitas warna merah asal

G = Nilai intensitas warna hijau

B = Nilai intensitas warna biru

C. Deteksi Tepi *Sobel*

"Deteksi tepi adalah proses untuk menemukan perubahan intensitas yang berbeda nyata dalam sebuah bidang citra"(Sutoyo, 2009).

Menurut Madenda (2015:110) menjelasan *Sobel* didalam bukunya sebagai berikut :

> *Sobel* mengembangkan sebuah operator dengan menggabungkan filter perata dan fungsi pertama turunan citra. Operator ini ditujukan agar dapat melakukan perataan atau minimalisasi *noise* oleh *filter* perata dan sekaligus melakukan pendeteksian tepi citra. Bila pemfilteran dilakukan dalam arah y maka turunan pertama dari hasil pemfilteran dilakukan dalam arah x.

Kelebihan dari metode *sobel* ini adalah kemampuan untuk mengurangi *noise* sebelum melakukan perhitungan deteksi tepi. Peninjauan pengaturan pixel di sekitar pixelnya (x,y) adalah (Setiawan, 2016):

$$\begin{matrix} p1 & p2 & p3 \\ p8 & (x,y) & p4 \\ p7 & p6 & p5 \end{matrix}$$

Operator *sobel* adalah magnitudo dari *gradien* yang dihitung dengan :

$$|M| = \sqrt{Sx^2+Sy^2}$$

Dalam hal ini turunan parsial dihitung dengan :

Sx = (p3 + cp4 + p5) − (p1 + cp8 + p7)

Sy = (p1 + cp2 + p3) − (p7 + cp6 + p5)

Dengan konstanta c = 2, sehingga terbentuk matrik operator *sobel* dapat di gambarkan seperti di bawah ini.

$$GX = \begin{bmatrix} -1 & 0 & 1 \\ 2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} GY = \begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

D. *K-Nearest Neighbors* (KNN)

"Algoritma *K-Nearest Neighbor* merupakan metode klasifikasi yang mengelompokan data baru berdasarkan jarak data baru itu kebeberapa data/tetangga (*neighbord*) terdekat" (Santoso, 2007). Salah satu cara untuk menghitung jarak dekat atau jauhnya tetangga menggunakan metode *euclidian distance*. *Euclidian Distance* berfungsi menguji ukuran yang bisa digunakan sebagai interpretasi kedekatan jarak antara dua obyek (Muhammad, 2015). di bawah ini merupakan rumus *Euclidian Distance*:

$$\boldsymbol{D(x,y) = \sqrt{\sum_{k=1}^{m} (x_{k-} y_k)^2}}$$

Keterangan,

Xk = nilai X pada *training* data

Yk = nilai Y pada *testing* data

m = batas jumlah banyaknya data

Jika semakin besar hasil nilai dari rumus di atas maka akan semakin jauh tingkat keserupaan antara data *training* dan data *testing* dan sebaliknya jika hasil nilainya semakin kecil maka akan semakin dekat tingkat keserupaan antar data *training* dan data *testing*. Langkah-langkah untuk menghitung algoritma K-NN adalah (Mukhoyyar, 2015):

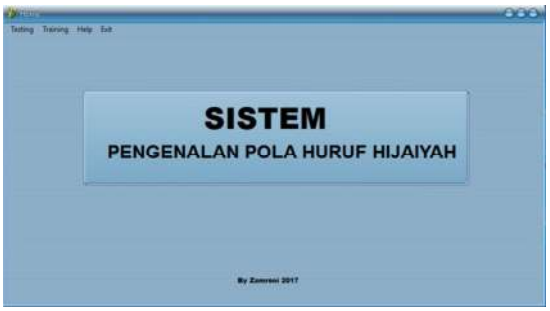
1. Menentukan nilai k.
2. Menghitung jarak masing-masing objek terhadap data latih yang diberikan dengan menggunakan metode *Euclidean Distance*.
3. Urutkan hasil perhitungan nilai kemiripan dari yang terkecil hingga yang terbesar.
4. Tentukan kelas dari data uji berdasarkan kelas yang paling kecil jaraknya.

III. HASIL DAN KESIMPULAN

A. Tampilan Program

1. Form Home

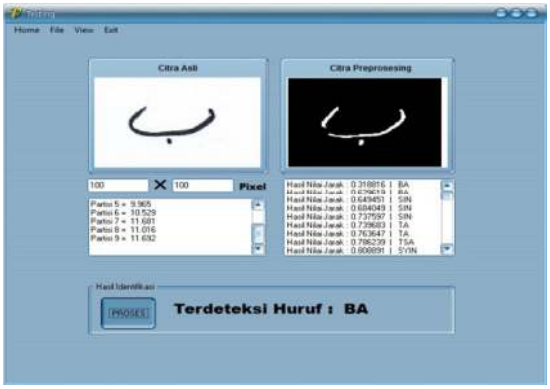
Form ini merupakan *form* yang muncul saat pertama kali aplikasi dijalankan oleh *user*. Tampilan *form home* dalam aplikasi pengenalan pola huruf Hijaiyah adalah sebagai berikut:



Gambar 2: *Form Home*

2. Form Testing

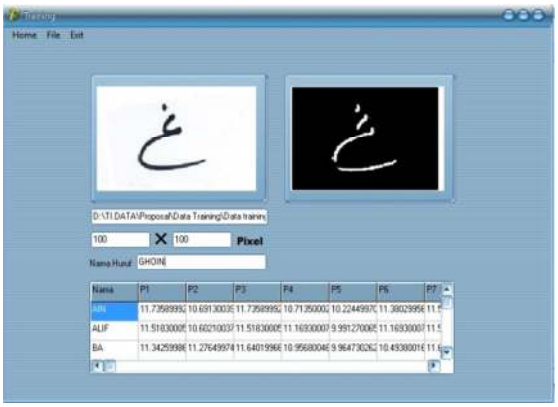
F*orm* ini merupakan *form* utama dari aplikasi yang berfungsi untuk mengidentifikasi citra huruf Hijaiyah, didalamnya terdapat tombol f*ile* - *open* untuk memasukan citra huruf hijaiyah. Kemudian tombol view - *preprocessing* untuk melakukan proses *grayscale* dan deteksi tepi *sobel* citra huruf hijaiyah. Dan tombol proses untuk melakukan identifikasi citra huruf hijaiyah yang dimasukan dan hasil identifikasi akan ditampilkan berupa bacaan huruh hijaiyah. Berikut ini adalah tampilan *form testing* dalam aplikasi pengenalan pola huruf *Hijaiyah*:



Gambar 4: *Form Testing*

3. Form Training

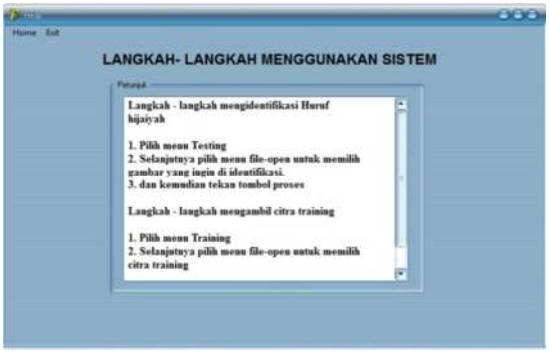
Form ini merupakan *form* untuk memasukan citra huruf hijaiyah ke data *training*, prosesnya sama seperti form *testing* hanya saja dalam *form training* akan ditemukan *data stringgrid* yang digunakan untuk menyimpan nilai – nilai partisi dari *input* citra *training*. Berikut ini adalah tampilan *form training* dalam aplikasi pengenalan pola huruf *Hijaiyah* :



Gambar 3: *Form Training*

4. Form Help

Form ini menampilkan tata cara menggunakan aplikasi untuk memberi petunjuk kepada *user*. Tampilan *form help* dalam aplikasi pengenalan pola huruf Hijaiyah adalah sebagai berikut:



Gambar 5: *Form Help*

B. Uji Coba

Dalam pengujian sistem pengenalan pola huruf hijaiyah ini dilakukan empat kali uji coba, dan disetiap

uji coba memiliki jumlah data *training* yang berbeda, data didapatkan dari tulisan tangan huruf hijaiyah siswa sekolah dasar yang berjumlah 196 data, yang nantinya dibagi menjadi 112 data *training* dan 84 data *testing*. Untuk memperoleh nilai akurasi pada setiap uji coba menggunakan rumus :

$$\frac{\text{Jml hasil benar}}{\text{Jml data training}} \text{ x } 100$$

Berikut ini adalah hasil uji coba dari sistem pengenalan pola huruf hijaiyah:

1. Uji Coba Pertama

Setelah dilakukan pengujian dengan menggunakan sistem pengenalan pola huruf hijaiyah, maka pada uji coba yang pertama menggunakan 112 data *training* didapatkan hasil seperti table dibawah ini:

Tabel 1: Hasil Uji Coba Pertama

Uji	Training	Testing	Hasil		Akurasi
			Benar	Salah	(%)
I	112	28	11	17	40
II	112	28	16	11	57
III	112	28	8	20	29

Dari tabel 1 didapatkan tingkat akurasi tertinggi yaitu 57% dengan menggunakan 112 data *training* dan uji coba ini dilakukan sebanyak 3 kali *testing* dengan 28 huruf hijaiyah yang berbeda karakternya.

2. Uji Coba Kedua

Pada uji coba kedua dilakukan dengan melakukan variasi jumlah data *training*, didapatkan hasil seperti tabel dibawah ini:

Tabel 2: Hasil Uji Coba Kedua

Uji	Training	Testing	Hasil		Akurasi
			Benar	Salah	(%)
I	140	28	11	17	40
II	140	28	17	11	61

Dari tabel 2 didapatkan tingkat akurasi tertinggi yaitu 61% dengan menggunakan 140 data *training* dan uji coba ini dilakukan sebanyak 2 kali *testing* dengan 28 huruf hijaiyah yang berbeda karakternya.

3. Uji Coba Ketiga

Pada uji coba ketiga dilakukan dengan melakukan variasi jumlah data *training* dan data *testing* di*rotasi* kekiri sekitar 10 drajat, didapatkan hasil seperti tabel dibawah ini:

Tabel 3: Hasil Uji Coba Ketiga

Uji	Training	Testing	Hasil		Akurasi
			Benar	Salah	(%)
I	168	28	4	24	14

Dari tabel 3 didapatkan tingkat akurasi sebesar 14% dengan menggunakan 168 data *training* dan 28 data *testing* yang sebelumnya di*rotasi* kekiri sekitar 10 drajat.

4. Uji Coba Keempat

Pada uji coba keempat dilakukan dengan melakukan variasi jumlah data

training, didapatkan hasil seperti tabel dibawah ini:

Tabel 4: Hasil Uji Coba Keempat

Uji	Training	Testing	Hasil		Akurasi (%)
			Benar	Salah	
I	196	28	22	6	79

Dari tabel 4 didapatkan tingkat akurasi sebesar 79% dengan menggunakan 196 data *training* dan uji coba ini dilakukan sebanyak 1 kali *testing* dengan 28 huruf hijaiyah yang berbeda karakternya. Dan ini merupakan hasil nilai akurasi tertinggi yang didapatkan setelah empat kali uji coba dilakukan dengan jumlah data training yang berbeda. Adapun hasil akurasi program yang dihasilkan dipengaruhi oleh jumlah data *training* yang diinputkan. Selain itu data *testing* yang digunakan pada penelitian ini merupakan tulisan tangan siswa sekolah dasar yang mana dari segi bentuk maupun ketepatan masing-masing penulisan hurufnya juga menjadi faktor penentu tinggi rendahnya akurasi program itu sendiri.

C. Kesimpulan

Berdasarkan implementasi program dan hasil evaluasi sistem pengenalan pola tulisan tangan huruf Hijaiyah dengan menggunakan metode K-Nearest Neighbors maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Aplikasi ini sudah mampu berjalan sesuai dengan perancangan sistem dan mampu mengenali karakter huruf hijaiyah.
2. Metode *K-Nearest Neighbors* bisa di implementasikan untuk sistem pengenalan pola tulisan tangan huruf Hijaiyah dan hasil uji coba pengenalan yang di dapat bisa mencapai akurasi 79%. Akurasi hasil pencocokan juga dipengaruhi oleh tulisan tangan huruf Hijaiyah itu sendiri. Baik itu dari segi posisi penulisan, ketebalan, maupun ketepatan dalam menulis huruf hijaiyah karena mengingat dalam karakter huruf hijaiyah memiliki huruf dengan pola penulisan yang hampir sama.

IV. DAFTAR PUSTAKA

Ahmad, U. 2015. *Pengolahan Citra Digital Dan Teknik Pemrogramannya*. Yogyakarta: Graha Ilmu.

Gestama, F.P. 2015. *Aplikasi Pengenalan Karakter Huruf Hijaiyah Untuk Pendidikan Anak Usia Dini (Paud)*. Skripsi. Tidak dipublikasikan. *Kediri: FT UNPGRI.*

Madenda, S. 2015. *Pengolahan Citra Dan Video Digital Teori Aplikasi Dan Pemrograman Menggunakan Matlab*. Jakarta: Erlangga.

Muhammad, G. 2015. *Analisis Sentimen Mengenai Berita Menggunakan*

Twitter Studi Kasus Detik.Com Dan Kompas.Com. (Online). tersedia: http://repository.widyatama.ac.id/xmlui/handle/123456789/6765, *diunduh 02 Desember 2016.*

Mukhoyyar, Z. 2015. Pengenalan Kata Aksara Jawa Menggunakan Algoritma K-Nearest Neighbor. (Online). tersedia: http://eprints.dinus.ac.id/16775/1/jurnal_15922.pdf, *diunduh 26 Oktober 2016*

Putra, D. 2010. *Pengolahan Citra Digital*. Yogyakarta : Andi.

Santoso, B. 2007. *Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis* (1 ed.). Yogyakarta: Graha Ilmu.

Setiawan, D. 2016. *Deteksi Tepi Berbasis Metode Sobel Untuk Peningkatan Kualitas Citra Medis*. Skripsi. Tidak dipublikasikan. Semarang: FIK UDINUS.

Sutoyo, T. dkk .2009. *Teori Pengolahan Citra Digital*. Yogyakarta: Andi.