

Ekstraksi Fitur Perpotongan dan Lengkungan untuk Mengenali Huruf Cetak

Limanyono Tanto¹, Liliana²

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra Jl. Siwalankerto 121 - 131 Surabaya 60236 Telp. (031)-2983455, Fax. (031)-8417658

E-mail: limanyonotanto@yahoo.com, lilian@petra.ac.id

ABSTRAK

Teknik manusia dalam memanipulasi gambar semakin banyak, terutama dalam hal pengenalan huruf. Berbagai jenis teknik memanipulasi ini pun memiliki kekurangan dan kelebihan. Salah satu teknik yang sering digunakan adalah menggunakan jaringan saraf tiruan. Sedangkan teknik pengenalan menggunakan fitur ekstraksi masih jarang digunakan.

Oleh karena itulah, dibuatlah sebuah aplikasi fitur ekstraksi untuk mengenali sebuah huruf balok pada gambar. Teknik yang akan digunakan adalah dengan mencari fitur perpotongan dan lengkungan dari gambar yang mengandung huruf balok sehingga dapat dilakukan pengenalan fitur dari huruf tersebut.

Melalui hasil pengujian, telah didapatkan contoh hasil pengenalan fitur perpotongan dan lengkungan yang disimpan pada matriks untuk dibandingkan dengan struktur dasar huruf. Matriks yang dihasilkan dari pengujian ini, untuk beberapa *font* tertentu hasilnya akan cukup memuaskan. Tetapi masih terdapat kekurangan dalam mengenali huruf dengan *font* yang tidak simetris.

Kata kunci:

Ekstraksi Fitur, Pengenalan huruf cetak, *Computer Vision*.

ABSTRACT

Human techniques in manipulating picture are more and more, especially in character recognition. Various types of techniques to manipulate it also has advantages and disadvantages. One technique that is often used is to use artificial neural network. While the recognition technique using the feature extraction is still rarely used.

Therefore, an application was made to recognize a feature extraction of capital letters in the image. The technique used is to find the intersection or edge and curvature features from images containing capital letters so that it can be done the recognition of features from such capital letters.

Through the test results, has obtained an example of the introduction of the intersection and curvature features stored in the matrix as compared to the basic structure of the letter. Matrix generated from this testing, for some particular font will be quite satisfactory result. But there are still shortcomings in recognizing letters in a font that is not symmetrical.

Keywords:

Feature Extraction, Letter Recognition, Computer Vision.

1. LATAR BELAKANG

Teknik manusia dalam hal manipulasi suatu citra/gambar semakin banyak, terutama dalam hal pengenalan alfabet. Kemampuan teknik manipulasi ini pun memiliki banyak macam

dan memiliki kekurangan dan kelebihan masing-masing. Hasil dari manipulasi ini pun dapat digunakan untuk melakukan pengenalan terhadap alfabet yang ada pada gambar. Teknik yang digunakan untuk melakukan pengenalan alfabet pun juga semakin banyak karena semakin berkembang.

Salah satu teknik pengenalan alfabet yang ada dan banyak digunakan adalah menggunakan jaringan saraf tiruan. Teknik ini menggunakan proses *training* untuk mendapatkan hasil maksimum dalam pengenalannya. Sebagai contoh adalah penelitian yang berjudul "Perancangan dan pembuatan perangkat lunak pengenalan huruf tulisan tangan dengan metode backpropagation" yang dibuat oleh mahasiswa bernama Setiawan Santoso dengan NRP 26406058. Hasil pengujian tertinggi menurut hasil pengujian penelitian ini, "Semua karakter sebesar 97%"[1]. Selain itu juga ada penelitian yang berjudul "Pembuatan perangkat lunak pengenalan huruf cetak menggunakan metode jaringan saraf tiruan" yang dibuat oleh mahasiswa bernama Steven Jaowry dengan NRP 26499163. Menurut penelitian ini, "Rata-rata hasil pengenalan untuk setiap jenis karakter yang dilatihkan adalah sama ($\pm 80\%$)."[2]. Namun teknik pengenalan menggunakan fitur ekstraksi masih jarang digunakan. Teknik ini dalam prosesnya untuk mengenali sebuah alfabet adalah dengan mendapatkan fitur yang dimiliki dari setiap alfabet. Setelah fitur berhasil di ekstraksilah baru dapat dikenali sebuah inputan tergolong alfabet yang mana.

Saat ini, sudah terdapat berbagai *research* serta aplikasi untuk melakukan pengenalan alfabet, tetapi metode pengenalan menggunakan fitur ekstraksi masih jarang digunakan. Hal ini terbukti dengan contoh yang telah disebutkan diatas, sehingga kurang di dalamnya teknik fitur ekstraksi ini.

Dari permasalahan yang dijabarkan di atas, pada skripsi ini akan dibuat aplikasi untuk melakukan ekstraksi fitur terhadap sebuah gambar yang memiliki objek alfabet terutama alfabet balok.

2. Metode Segmentasi

Segmentasi merupakan suatu proses untuk mempartisi sebuah gambar menjadi beberapa bagian yang sesuai dengan kriteria yang akan diambil untuk mempermudah proses analisa. Segmentasi adalah "*Segmentation subdivides an image into its constituent regions or objects.*"[3].

Sebelum proses segmentasi dilakukan diperlukan proses dasar gambar. Proses tersebut berupa *grayscale* dan *monochrome*. Dimana persamaan yang digunakan sesuai standart ITU-R BT.709 yang biasa digunakan untuk pengembangan HDTV pada proses *grayscale* pada sebuah gambar:[4]

$$Y' = 0.2126 * R + 0.7152 * G + 0.0722 * B \quad (1)$$

Setelah proses dasar dilakukan terhadap gambar yang didapatkan agar dapat melakukan pemilihan objek. Hal yang dapat dilakukan selanjutnya adalah proses segmentasi untuk memisah objek

keseluruhan pada gambar menjadi per hurufnya. Dari data objek yang telah dibinerkan melalui proses *Monochrome*, sehingga didapatkan gambar hitam putih yang dapat digunakan untuk proses segmentasi untuk mendapatkan bagian yang menunjukkan posisi dari huruf.

Proses segmentasi ini dilakukan melalui 2 proses utama:

1. Segmentasi horisontal, dimana pada proses ini dilakukan segmentasi terhadap baris dari data gambar.
2. Segmentasi vertikal, dimana pada proses ini dilakukan segmentasi terhadap kolom dari data gambar.

Dimana untuk proses segmentasi horisontal ini, dilakukan dengan melakukan pengecekan pada data gambar dari bagian paling atas kiri sampai bagian paling kanan bawah. Setelah pengecekan dilakukan maka akan didapatkan posisi paling atas dari data gambar dan juga tinggi dari data gambar pada baris tersebut. Kemudian data tersebut diambil untuk dilakukan proses segmentasi vertikal, maka proses ini akan diulangi lagi untuk mendapatkan baris data gambar selanjutnya sampai semua data pada gambar telah diambil semua.

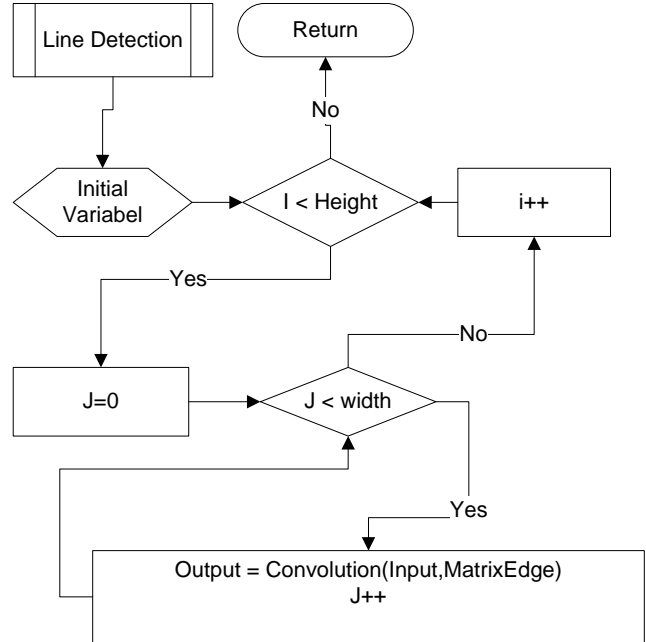
Proses segmentasi vertical ini juga akan dilakukan pengecekan pada data gambar dari bagian paling atas kiri sampai paling kanan bawah dari hasil segmentasi horisontal yang berupa sebaris data gambar. Kemudian akan didapatkan letak paling kiri dan lebar dari sebuah objek gambar. Letak tersebut akan menunjukkan letak dan ukuran dari sebuah data gambar huruf. Proses segmentasi horisontal ini akan diulangi lagi untuk mendapatkan semua objek huruf yang dapat ditemukan pada sebaris gambar huruf tersebut.

3. Pengenalan Fitur

Untuk pengenalan fitur huruf dari suatu gambar, prosesnya dilakukan pada setiap objek huruf yang ditemukan oleh proses segmentasi. Pengenalan Fitur ini akan berguna untuk *Template Matching*, karena "*Template matching is a simple filtering method of detection of a particular feature in an image.*"[6]. Sebuah proses pengenalan karakter dapat dilakukan dengan mendapatkan fitur dari karakter tersebut[5]. Sedangkan untuk proses pengenalan fitur untuk setiap objek tersebut dilakukan berdasarkan 2 hal yaitu pengenalan terhadap garis dan lengkungan.

3.1. Pengenalan Garis

Proses pengenalan garis dilakukan untuk mendapatkan posisi dari garis pada objek. Kemudian dari garis yang ditemukan dilakukan pengecekan perpotongan yang terjadi. Hasil dari pengecekan perpotongan yang terjadi digunakan untuk mengenal sebuah huruf, karena setiap huruf memiliki posisi perpotongan garis yang berbeda-beda meskipun beberapa huruf memiliki ciri yang sama. Proses pengenalan perpotongan ini akan dilakukan dengan melakukan perkalian matriks antara input objek yang didapat dengan beberapa matriks yang menunjukkan sebuah perpotongan. Hasil dari perkalian matriks ini akan menghasilkan matriks baru yang menunjukkan letak dari perpotongan yang ditemukan. Berikut rancangan sistem kerja pengenalan garis dalam *flowchart* pada Gambar 1.

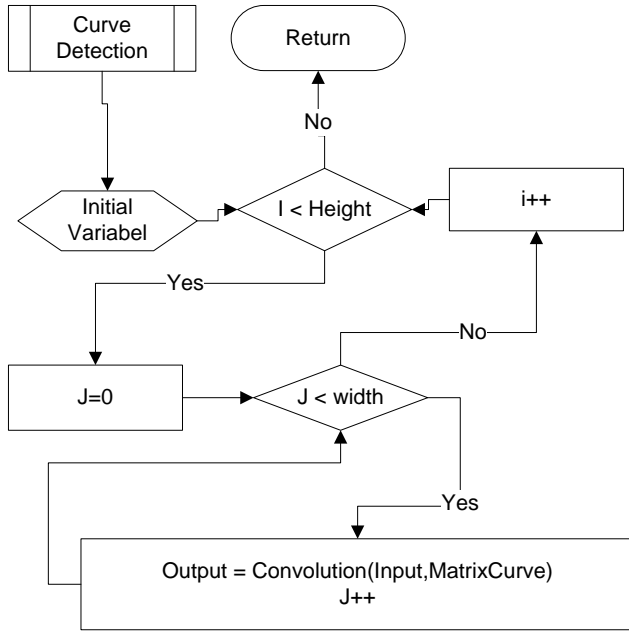


Gambar 1 *Flowchart* Pengenalan Garis

3.2. Pengenalan Lengkungan

Proses pengenalan lengkungan ini dilakukan untuk mendapatkan letak atau posisi dari lengkungan yang ada pada objek. Proses ini merupakan salah satu proses penting dalam melakukan pengenalan fitur dari sebuah objek huruf, karena pada sebuah huruf selain terdapat fitur garis yang berpotongan juga terdapat lengkungan. Fitur lengkungan ini akan memberikan ciri yang lebih khusus terhadap proses pengenalan sebuah huruf. Karena pada pengenalan perpotongan garis terdapat beberapa huruf yang memiliki hasil perpotongan garis yang sama. Sehingga hanya dengan pengenalan perpotongan garis saja tidak dapat digunakan mengenali sebuah huruf dengan baik.

Pengenalan lengkungan ini akan memberikan fungsinya untuk benar-benar membedakan antara satu huruf dengan huruf lainnya. Selain itu beberapa huruf juga tidak memiliki lengkungan pada cirinya. Hal ini cukup memberi dampak pada proses pengenalan huruf. Karena pada hasil kombinasi pengenalan fitur lengkungan ini dengan fitur perpotongan akan menunjukkan perbedaan antara satu huruf dengan huruf lainnya berdasarkan letak dari lengkungan dan perpotongan yang ditemukan. Berikut rancangan sistem kerja pengenalan lengkungan pada Gambar 2.



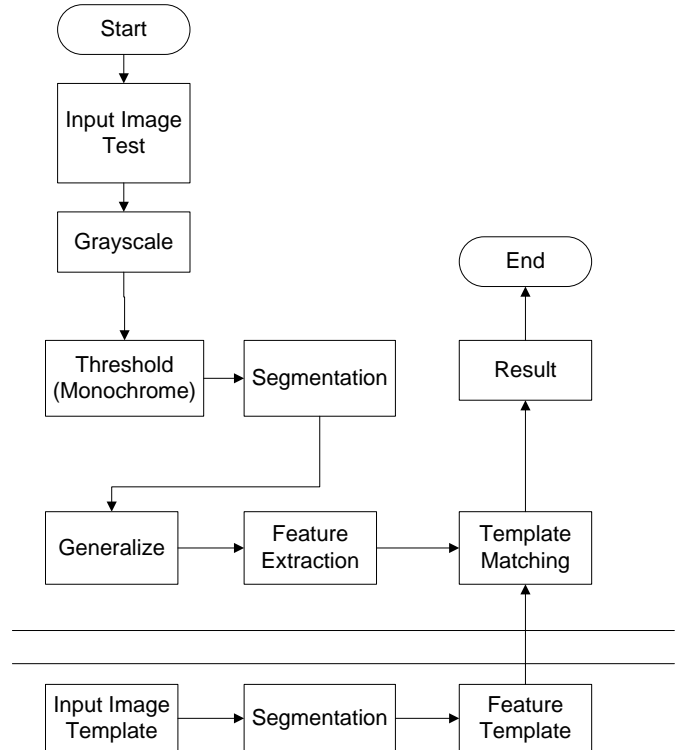
Gambar 2 Flowchart Pengenalan Lengkungan

4. Desain Sistem

Sistem kerja perangkat lunak untuk mengenali huruf dari gambar ini memiliki beberapa tahapan yang harus dilakukan. Pertama *user* diharuskan untuk memilih input gambar yang akan digunakan. Setelah *user* memilih inputan gambar, maka program akan langsung melakukan proses dasar untuk memproses sebuah gambar. Proses tersebut adalah *Grayscale* dan *Monochrome*. Setelah proses dasar tersebut dilakukan maka tampilan yang ditampilkan kepada *user* adalah tampilan hasil proses dasar tersebut.

Hasil proses dasar tersebut kemudian dilanjutkan dengan proses segmentasi objek huruf yang terdapat pada gambar. Sehingga proses pengenalan akan dilakukan pada tiap huruf yang telah dipisahkan dari gambar utama. Gambar objek huruf dari hasil segmentasi akan dilakukan proses *feature extraction*, dimana pada proses ini akan dilakukan 2 hal yaitu *curve detection* dan *edge detection*. Hasil dari *curve detection* akan disimpan pada matriks format yang telah ditentukan untuk mengetahui letak dari lengkungan pada gambar objek. Sedangkan hasil dari *edge detection* akan digunakan untuk mengetahui letak perpotongan antar garis yang terjadi pada objek gambar. Kemudian hasil dari kedua proses ini disatukan untuk menjadi satu matriks yang dapat digunakan untuk mengenali letak dari fitur pada objek gambar.

Proses selanjutnya setelah mendapatkan matriks data posisi fitur ini adalah proses *template matching*. Proses ini akan mencocokkan data objek tersebut sesuai dengan data huruf yang telah disediakan. Hasil dari proses penyesuaian ini akan dapat digunakan untuk mengenali objek huruf yang ada pada sebuah gambar. Berikut rancangan sistem kerja perangkat lunak secara garis besar digambarkan dalam *flowchart* pada Gambar 3.



Gambar 3 Flowchart Garis Besar Sistem Kerja Perangkat Lunak

5. Pengujian Sistem

Pengujian terhadap huruf yang bervariasi dilakukan dengan menggunakan huruf besar dan huruf kecil. Pengujian ini akan dilihat apakah semua huruf mampu dikenali sesuai yang diinginkan. Dimana Tabel 2 untuk contoh hasil yang presisi, Tabel 3 untuk contoh hasil yang bisa tertukar, dan Tabel 4 untuk contoh hasil yang tidak dikenali.

Tabel 1 Hasil Pengujian Keseluruhan

Hasil	List Huruf
Presisi	A,D,H,I,J,K,k,L,m,N,q,T,t,U,V,v,W,w,X,Y,y,Z
Tertukar	b,d,E,F,f,G,g,H,I,J,I,M,P,p,Q,R,r,u,x
Tidak Dikenali	a,B,C,c,e,n,O,o,S,s,z

6. KESIMPULAN

Berdasarkan hasil pengujian dapat disimpulkan beberapa hal sebagai berikut:

- Pendeteksian huruf sangat dipengaruhi oleh struktur dasar dari *font* yang digunakan. Karena dari hasil penelitian ini didapatkan tingkat keberhasilan terhadap *font Courier* adalah sebesar 23%. Untuk *font Arial* memiliki keberhasilan sebesar 73% dan untuk *font Times New Roman* memiliki tingkat keberhasilan 20%. Hal ini menunjukkan huruf yang memiliki struktur tambahan membuat proses pengenalan fitur kurang berfungsi dengan baik. Hal ini dikarenakan struktur tambahan tersebut akan menyebabkan pengenalan fitur mendeteksi area yang lebih atau fitur yang berlebih, sehingga dapat menyebabkan pendeteksian tidak berjalan dengan baik.

Pendeteksian huruf akan lebih akurat apabila *input font* yang dideteksi merupakan *font* yang fitur dan struktur dasarnya simpel dan simetris. Dan juga dari hasil yang dilakukan terhadap *font Times New Roman* memiliki masalah karena pada *font* ini terdapat bagian yang sangat tipis dan dapat mengakibatkan hilangnya fitur dari sebuah huruf.

- Ukuran dari tulisan yang terlalu kecil akan mengakibatkan proses deteksi gagal, karena data fitur yang tidak dapat didapatkan dengan baik untuk dideteksi. Sedangkan ukuran tulisan yang terlalu besar akan mengakibatkan ketebalan data yang terkadang akan membuat pendeteksian tidak berjalan dengan baik.

- Warna *input* yang berdasarkan nilai RGB cukup tinggi akan mengakibatkan tulisan tersebut untuk menjadi tidak terdeteksi. Hal ini dikarenakan adanya proses dasar pencitraan digital yaitu *monochrome* yang menggunakan sebuah nilai *threshold* untuk mengolah data pada gambar *input*. Nilai *threshold* ini akan membedakan nilai RGB yang tinggi dengan yang bernilai rendah. Sedangkan proses pendeteksian ini akan dapat dideteksi apabila objek yang di baca berada pada nilai RGB yang rendah.

- Untuk proses pengenalan antara huruf besar dan huruf kecil pada hasil pengujian ini menunjukkan tingkat keberhasilannya 78.85% berdasarkan Tabel 1. Dimana pada hasil pengujian didapatkan bahwa ada beberapa huruf besar yang dapat dikenali sebagai huruf kecilnya karena memiliki bentuk yang sama seperti huruf 'V' dan 'X'. Didapatkan juga pada hasil pengujian ini ada beberapa huruf yang juga memiliki struktur huruf lain pada huruf tersebut seperti huruf 'M' yang memiliki struktur huruf 'V' pada bagian tengahnya.

Tabel 3 Hasil Pengujian Huruf yang Bisa Tertukar

Huruf	Hasil
b	<div style="display: flex; justify-content: space-between; align-items: center;"> b <div style="border: 1px solid black; padding: 2px;">Detect Selected</div> Lb </div> <pre> 000000000000 000000000000 000000000000 000000000000 000000000000 000000000000 000000200000 000000000000 000033330000 202000000000 002000000000 000000000000 000000000000 000000000000 000000000000 000000000000 000000000000 000000000000 200000000000 200000000000 220000000000 000000000000 000002200000 0000 0020 2330 2000 0000 2000 2220 0000 0000 0255255255 25500255 0000 000255 0255255255 0000 0000 0255255255 25500255 0000 25500255 0255255255 </pre>

Tabel 2 Hasil Pengujian Huruf yang Dikenali dengan Presisi

Huruf	Hasil
A	<div style="display: flex; justify-content: space-between; align-items: center;"> A <div style="border: 1px solid black; padding: 2px;">Detect Selected</div> A </div> <pre> 0000000000000000 000000020020000000 000000020000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000202000000202000 000023000000030000 000233000000330000 000200000000020000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 000000000000000000 002200 000000 000000 000000 030030 020020 000020 000000 000000 000000 000000 00255000 00025500 000000 000000 00255000 00255000 000000 0025525500 00025500 000000 000000 </pre>

Tabel 4 Hasil Pengujian Huruf yang Tidak Dikenali

Huruf	Hasil
Z	<div style="display: flex; justify-content: space-between; align-items: center;"> Z <div style="border: 1px solid black; padding: 2px;">Detect Selected</div> TVv </div> <pre> 00002222222222000 00003333333332200 00022222233332200 02222222233342000 00000000000432200 00000000000322200 00000000000222000 00000000000200000 00000000000200000 00000000000000000 000000002000000000 000000020000000000 000000020000000000 000000200000000000 000002000000000000 000002000000000000 000200000000000000 024440000000000000 044430000000000000 022220000000000000 2222222222222200 033332 222342 000220 002000 022000 442000 442222 000000 000000 000000 000000 000000 02550000 000000 000000 000000 000000 000000 000000 02550000 000000 </pre>

7. REFERENCES

- [1] Santoso, Setiawan. 2010. Perancangan dan Pembuatan Perangkat Lunak Pengenalan Huruf Tulisan Tangan dengan Metode Backpropagation. Skripsi, Universitas Kristen Petra, Surabaya.
- [2] Jaowry, Steven. 2003. Pembuatan Perangkat Lunak Pengenalan Huruf Cetak Menggunakan Metode Jaringan Saraf Tiruan. Skripsi, Universitas Kristen Petra, Surabaya.
- [3] Gonzales, Rafael C., Woods, Richard E. 2002. Digital Image Processing Second Edition. Prentice Hall. Upper Saddle River, New Jersey 07458.
- [4] Wikipedia The Free Encyclopedia. (2013). Grayscale. Retrived August 12, 2013 from <http://en.wikipedia.org/wiki/Grayscale>.
- [5] Dileep, Dinesh. 2009. A Feature Extraction Technique Based on Character Geometry for Character Recognition. Department of Electronics and Communication Engineering, Amrita School of Engineering, Kollam, Kerala, India.
- [6] Bowman, M., Ballard, Dana H., Brown, Christopher M. 1982. Computer Vision. Department of Computer Science, University of Rochester, Rochester, New York, USA.