

PENGGUNAAN *TEMPLATE MATCHING* PADA *OPTICAL CHARACTER RECOGNITION* DENGAN *THRESHOLD*

TEMPLATE MATCHING IN OPTICAL CHARACTER RECOGNITION WITH THRESHOLD

Nadjmul Achyar¹, Kris Sujatmoko², Iwan Iwut Tritoasmoro³

^{1,2,3} Universitas Telkom, Bandung

nadjmulachyar@students.telkomuniversity.ac.id¹, krissujatmoko@telkomuniversity.ac.id²
iwaniwut@telkomuniversity.ac.id³

Abstrak

Optical character recognition adalah sebuah sistem komputer yang digunakan secara otomatis mengenali serangkaian karakter yang berasal dari mesin ketik, mesin cetak ataupun tulisan tangan. Dengan kata lain *optical character recognition* adalah proses pengalihan dokumen gambar menjadi file komputer tanpa harus pengeditan ulang, setiap karakter baik huruf, kata, kalimat dapat dikenali secara tepat dan dibaca oleh perangkat lunak yang lain, tanpa harus pengetikan ulang dan editing.

Untuk mengevaluasi performansi dari *method Template Matching* maka di gunakan pengaturan *threshold* agar mengetahui tingkat persentase ke berhasilan mengenali huruf-huruf pada citra gambar sehingga dapat mengenali karakter huruf -huruf dan pola huruf pada pixel yang berasal dari gambar. Proses pengujian penggunaan *optical character recognition* menggunakan *template matching* dan *threshold* pada 4 font huruf yang di uji.

Dari hasil pengujian system ini didapati hasil bahwa penggunaan algoritma *template matching* dan penggunaan *threshold* mendapat kan hasil 99.6% keberhasilan mengenali karakter dengan menggunakan *threshold*.

Kata kunci : Pengenal huruf otomatis, *Optical character recognition*, *Template Matching*, *Threshold*

Abstract

Optical character recognition is a computer system that is used to automatically recognize characters from a typewriter, printing press or handwriting. In other words, optical character recognition is the process of making image documents into computer files without re-searching, every character, both letters, words, sentences can be correctly recognized and read by other software, without having to retype and edit.

and to find the performance of the Template Matching method, a threshold setting is used to determine the percentage level of success in recognizing the letters in the image so that it can recognize the character of the pattern letters on the pixels that come from the image. The process of testing the use of optical character recognition uses template matching and threshold on the 4 tested fonts.

From the results of testing this system, it was found that the use of template matching algorithms and the use of thresholds resulted in 99.6% success in recognizing characters using thresholds.

Keywords: *Automatic letter recognition, Optical character recognition, Template Matching, Threshold*

1. Pendahuluan [10 pts/Bold]

Optical character recognition adalah sistem komputer yang dapat membaca huruf, baik yang berasal dari sebuah pencetak (printer atau mesin ketik) maupun yang berasal dari tulisan tangan[1]. Dengan adanya sistem *optical character recognition* maka pemakai dapat lebih leluasa

memasukkan data karena tidak perlu menggunakan papan ketik atau keyboard tetapi bisa menggunakan pena elektronik untuk menulis sebagaimana pemakai menulis di kertas. Adanya *optical character recognition* juga akan memudahkan penanganan pekerjaan yang memakai data input berupa tulisan seperti penyortiran surat di kantor pos, pemasukan data buku di perpustakaan, dan lain-lain. Adanya sistem *optical character recognition* yang cerdas akan sangat membantu usaha besar-besaran yang saat ini dilakukan banyak pihak yakni usaha digitalisasi informasi dan pengetahuan, misalnya dalam pembuatan koleksi pustaka digital, koleksi sastra kuno digital, dan lain-lain. *optical character recognition* merupakan solusi yang efektif untuk proses konversi dari gambar ke dalam bentuk dokumen digital. Permasalahan yang muncul dalam melakukan proses pengenalan huruf pada komputer adalah bagaimana sebuah sistem pengenalan dapat mengenali berbagai *font* huruf dengan ukuran, ketebalan, dan bentuk yang berbeda. Secara umum terdapat dua hal utama yang mempengaruhi proses *optical character recognition* yaitu: mekanisme ekstraksi ciri dan mekanisme pengenalan pola.

Mekanisme ekstraksi ciri dilakukan untuk mendapatkan bukti dari suatu karakter atau huruf. Proses pengenalan pola dilakukan setelah mekanisme ekstraksi ciri. Proses pengenalan pola bertujuan untuk mencocokkan *pattern* huruf yang berasal dari inputan dengan pola yang ada dalam basis pengetahuan. Beberapa algoritma yang dapat digunakan untuk proses pengenalan antara lain logika *fuzzy*, *k-Nearest Neighbor Algorithm*, *sequence alignment*, *template matching* dan lain-lain[2]. Dari setiap algoritma yang sudah di sebutkan, algoritma *template matching* merupakan salah satu algoritma yang efektif untuk diterapkan dalam sistem *OCR (optical character recognition)*. Algoritma pengenalan ini sangat sederhana dan dapat diterapkan dengan melakukan proses ekstraksi ciri terlebih dahulu, sehingga mempunyai hasil pengenalan yang tinggi.

2. Dasar Teori dan Metodologi

2.1 Optical Character Recognition

Optical character recognition adalah sebuah sistem pada komputer yang dapat mengolah gambar menjadi text. Gambar yang dimaksud dapat berupa hasil scan dokumen, hasil *captured* halaman web, hasil foto, dan lain-lain.

Optical character recognition adalah sistem yang sudah lama dikembangkan. Sistem *optical character recognition* terus dikembangkan hingga saat ini, sehingga dapat menghasilkan hasil akurasi yang lebih baik bahkan pada citra gambar yang dimana karakter huruf sulit untuk dikenali. Algoritma *template matching* pada *optical character recognition* memiliki 3 tahapan yaitu *pre-processing*, ekstraksi ciri dan klasifikasi karakter[1]. Yang memudahkan *optical character recognition* untuk membandingkan gambar input dengan *template*.

Character Recognition memiliki fungsi untuk mengenali tulisan, serta mengenali karakter tulisan dalam gambar dan merubahnya kedalam *American Standad Code for Information Interchange (ASCII)* atau bahasa mesin lainnya[2].

2.2 Template Matching

Template matching adalah metode untuk mencari dan menemukan bagian kecil dari gambar yang cocok dengan *template* gambar, *template matching* merupakan salah satu teknik dalam *image processing* yang berfungsi untuk mencocokkan setiap bagian dari suatu citra input dengan citra yang menjadi *template* (acuan). Citra input dibandingkan dengan citra *template* yang ada didalam basis data, kemudian dicari kesamaannya dengan menggunakan suatu aturan tertentu. Pencocokan citra yang menghasilkan tingkat kemiripan atau kesamaan yang tinggi menentukan suatu citra tersebut dikenali sebagai salah satu dari citra *template* [3].

Method template matching adalah salah satu *method* terapan dari teknik konvolusi. *Method* ini sering digunakan untuk mengidentifikasi citra karakter huruf, angka, sidik jari (*fingerprint*) dan aplikasi-aplikasi pencocokan citra lainnya [4]. Secara umum teknik konvolusi dapat didefinisikan sebagai suatu cara untuk mengkombinasikan dua buah baris angka yang menghasilkan baris angka ke tiga.

$$\min e = \sum_{(x,y) \in W} (I_{x,y} - T_{x,y})^2 \quad (1)$$

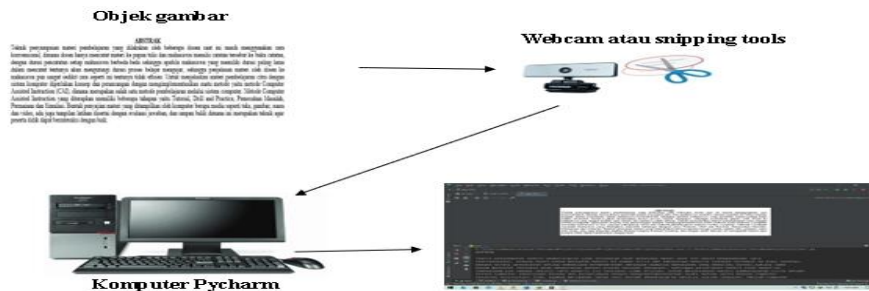
Rumus diatas digunakana untuk menghitung nilai error terkecil dengan membandingkan template acuan dengan gambar inputan, nilai yang digunakan yaitu nilai biner setiap gambar akan di rubah dulu ke citra biner sehingga dapat diketahui nilai biner dari setiap template acuan dengan gambar inputan.

2.3 Adaptive threshold

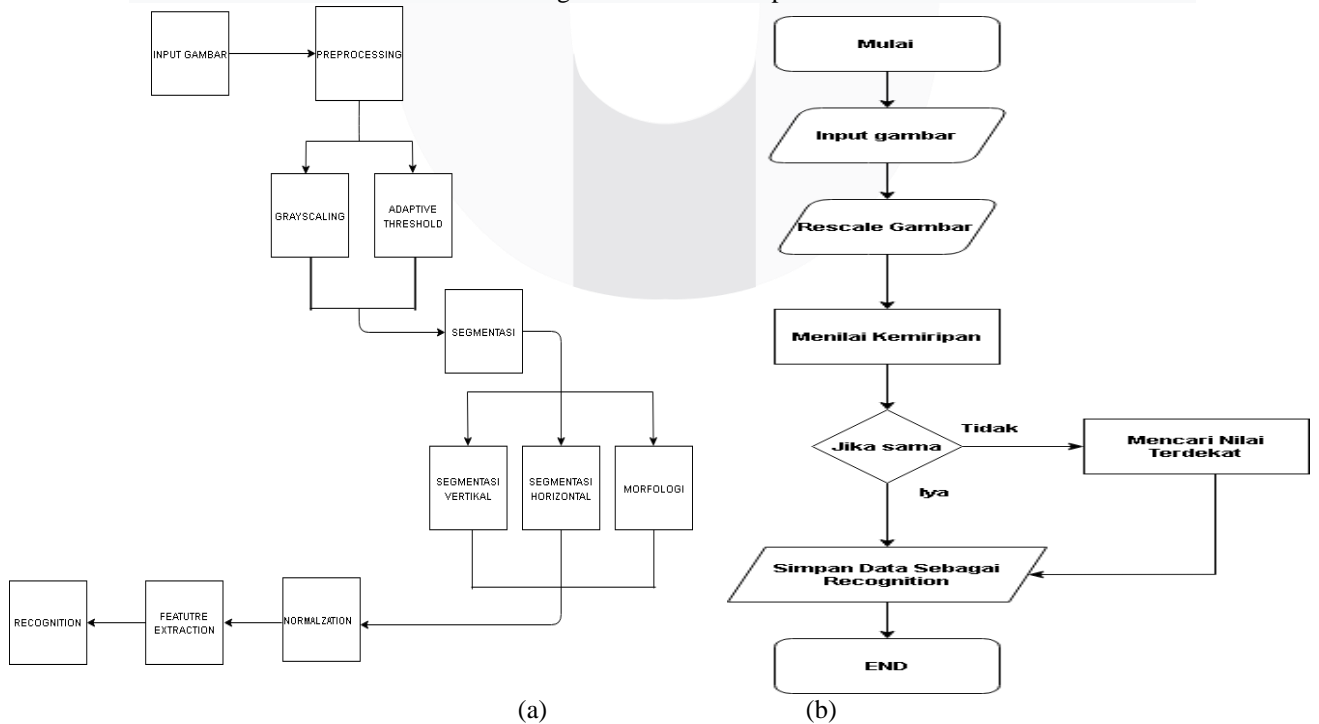
Adaptive thresholding adalah metode pengubahan citra gambar menjadi citra biner, tujuan dari penelitian ini adalah: untuk memudahkan ekstrasi citra atau informasi terpenting dalam citra. Sehingga proses selanjutnya seperti pengenalan citra atau *recognition*. *Adaptive thresholding* dilakukan terhadap daerah-daerah dalam citra. Citra dipecah menjadi bagian bagian kecil , kemudian proses pengembangan dilakukan secara adaptive. Dengan *adaptive thresholding* secara subjektif citra biner yang di hasilkan terlihat lebih cerah dan sedikit informasi yang hilang.

2.4 Desain sistem

Algoritma template matching digunakan pada sistem *Optical character recognition* agar memiliki hasil yang akurat , dikarenakan penggunaan *template matching* sangat sesuai untuk sistem *Optical character recognition*. Selain penggunaan algoritma template matching sistem *Optical character recognition* menggunakan *adaptive thresholding* dikarenakan saat melakukan *cropping* atau pengambilan citra gambar akan ada noise pada setiap citra yang kita gambar yang kita ambil disinilah peran *adaptive thresholding* diperlukan, yaitu *Thresholding* menghasilkan citra biner, di mana semua piksel dengan intensitas di atas atau di bawah nilai ambang diaktifkan, sementara semua piksel lainnya dimatikan.



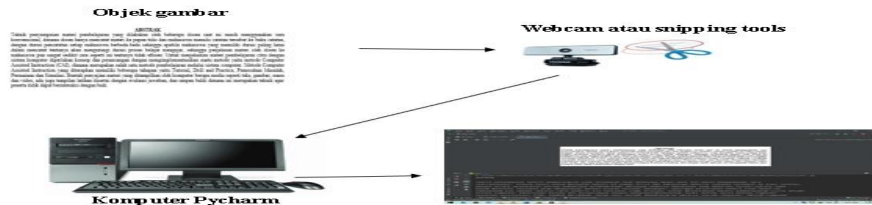
Gambar 1. gambaran umum implementasi



Gambar 2. (a) Block Diagram *Optical Character Recognition*, (b) Flow Chart Template Matching

2.5 Skematik Pengujian

Karena menggunakan pemrograman bahasa Python maka menggunakan aplikasi pycharm untuk menjalankan program *optical character recognition*, serta menggunakan aplikasi *snipping tools* untuk mengcapture citra gambar yang akan di ambil , maka diperlukan mengkonfigurasi *threshold* sebelum melakukan pengujian di karenakan setiap gambar pasti memiliki kesesuaian nilai *threshold* yang berbeda - beda

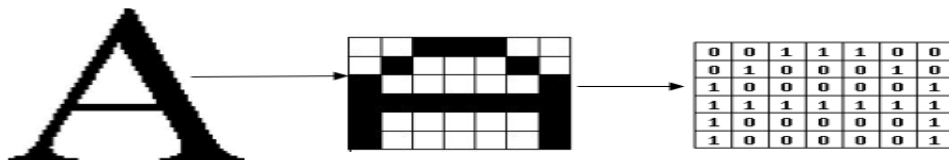


Gambar 3. gambaran umum implementasi

3. Hasil percobaan dan analisis

3.1. Pengujian

Sebelum melakukan pengujian perlu di ketahui pola pixel dari setiap masing masing huruf yang nantinya akan dihitung nilai error dari *template* huruf yang di pakai.



Gambar 4. Gambar nilai biner pola huruf A

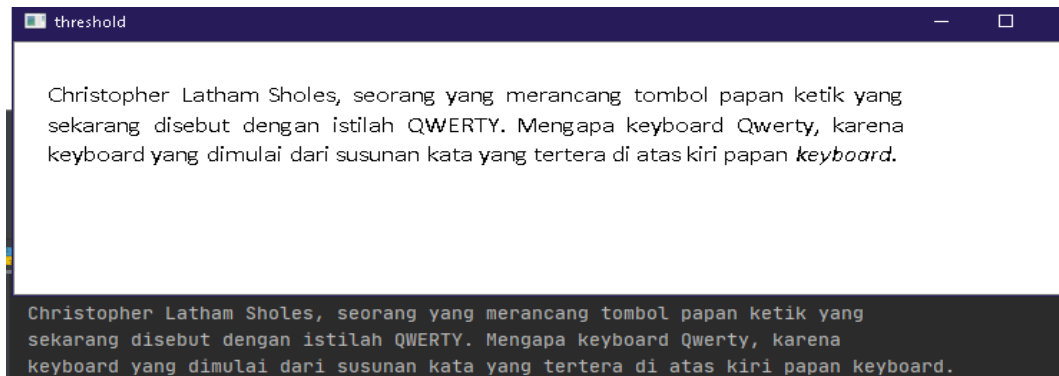
Setelah sebuah huruf sudah memiliki pola piksel maka dilakukan konversi ke biner sebagai nilai pola suatu huruf, dimulai dari kiri atas hingga ke kanan bawah. Nilai pola yang berwarna hitam akan direpresentasikan dengan nilai 1. Sedangkan pola yang berwarna putih akan direpresentasikan dengan nilai 0 berikut table nilai pola huruf.

Tabel 1. Pola Huruf dan nilai biner

	A	001110001000101000001111111110000011000001
	B	111000010010001110000100100010010001110000
	C	011110010000101000000100000010000100111100
	D	111110010000101000010100001010000101111100

Nilai error pola huruf pada citra latih dan *template* akan dihitung dengan deretan nilai pola biner pada citra *input* dan *template*. Citra *input* dan *template* dengan nilai error terkecil merupakan citra *template* yang paling sesuai dengan citra masukan menggunakan rumus persamaan (1). Pengujian

pada program ini di bagi menjadi 4 font yang di uji yaitu font calibri, times new roman, helvetica, verdana.



Gambar 5. hasil pengujian OCR

Tabel 2. Hasil pengujian font times new roman

Nama Gambar	Akurasi	Total Huruf	Error Rate	Waktu	Resolusi	Ukuran Font
TM1	100%	448	0%	4.5 Detik	1,927*2,655	12
TM2	100%	433	0%	4.0 Detik	1,927*2,655	12

Tabel 3. Hasil pengujian font calibri

Nama Gambar	Akurasi	Total Huruf	Error Rate	Waktu	Resolusi	Ukuran Font
CAL1	100%	448	0%	4.3 Detik	1,927*2,655	12
CAL2	99.8%	433	0.2%	4.0 Detik	1,927*2,655	12

Tabel 4. Hasil pengujian verdana

Nama Gambar	Akurasi	Total Huruf	Error Rate	Waktu	Resolusi	Ukuran Font
VER1	100%	448	0%	3.9 Detik	1,927*2,655	12
VER2	100%	433	0%	3.4 Detik	1,927*2,655	12

Tabel 5. Hasil pengujian font helvetica

Nama Gambar	Akurasi	Total Huruf	Error Rate	Waktu	Resolusi	Ukuran Font
HEL1	100%	448	0%	3.9 Detik	1,927*2,655	12
HEL2	100%	433	0%	3.8 Detik	1,927*2,655	12

3.2 Analisis

Dari hasil diatas dapat di analisis bahwa yang mempengaruhi waktu pemrosesan *optical character recognition* adalah total huruf resolusi citra gambar yang di uji, sedangkan yang mempengaruhi akurasi dari *optical character recognition* tingkat kecocokan citra uji dengan template yang ada, ekstrasi ciri huruf, dan nilai error dari citra uji dengan *template*.

4. Kesimpulan

Perancangan *optical character recognition* menggunakan metode template matching dan adaptive threshold dapat menghasilkan output pengenalan huruf atau *word recognition* yang akurat, metode template matching adalah metode yang cocok untuk melakukan *letter recognition* maupun *optical character recognition* dengan menggunakan template latih sebagai pembanding dengan temcitra uji, sedangkan *adaptive threshold* berguna mengurangi sebuah *noise* dan mengubah pola citra masukan

menjadi pola citra biner sehingga dapat memudahkan *optical character recognition* mengenali sebuah huruf maupun angka yang sudah ada di template latih.

REFERENSI

- [1] Ansari, S., & Sutar, U. (2016). Optimized and efficient feature extraction method for devanagari handwritten character recognition. *Proceedings - IEEE International Conference on Information Processing, ICIP 2015*, 11–15. <https://doi.org/10.1109/INFOP.2015.7489342>
- [2] Vamvakas, G., Gatos, B., Stamatopoulos, N., & Perantonis, S. J. (2008). A complete *optical character recognition* methodology for historical documents. *DAS 2008 - Proceedings of the 8th IAPR International Workshop on Document Analysis Systems*, (September), 525–532. <https://doi.org/10.1109/DAS.2008.73>
- [3] Liu, Z., Guo, Y., Feng, Z., & Zhang, S. (2019). Improved Rectangle Template Matching Based Feature Point Matching Algorithm. *Proceedings of the 31st Chinese Control and Decision Conference, CCDC 2019*, 2275–2280. <https://doi.org/10.1109/CCDC.2019.8833208>
- [4] Roy, A., Memon, N., & Ross, A. (2019). MasterPrint Attack Resistance: A Maximum Cover Based Approach for Automatic Fingerprint Template Selection. *2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems, BTAS 2019*. <https://doi.org/10.1109/BTAS46853.2019.9186010>

