

PEMBANGUNAN JENIS HURUF BARU PADA SISTEM KOMPUTER MELALUI PENDEKATAN STRING MATCHING

Andre Marga Pradja¹, Rimba Whidiana², Zk. Abdurahman Baizal³

¹Teknik Informatika, Fakultas Teknik Informatika, Universitas Telkom

Abstrak

Teknik manipulasi dan analisis bentuk karakter merupakan teknik yang dikembangkan dari pemrosesan gambar dan pengenalan bentuk. Permasalahannya adalah bagaimana mengenali suatu bentuk karakter dalam sistem komputer. Tugas akhir ini membahas bagaimana mengenali bentuk dan karakteristik dari suatu tipe huruf dan mendesain jenis huruf baru dari huruf yang telah ada dengan menggunakan pendekatan string matching. Tujuannya ialah membangun aplikasi dimana seseorang dapat menciptakan bentuk huruf baru tanpa melakukan proses desain, tetapi dengan modifikasi bentuk huruf yang sudah ada. Adapun algoritma string matching yang digunakan adalah Boyer-Moore dan tipe huruf yang dapat diproses hanya terbatas pada format TrueType Font.

Metode yang digunakan untuk penyelesaian masalah tersebut adalah Longest Common Substring (LCS) dan String Matching dengan algoritma Boyer-Moore. LCS digunakan untuk mencari elemen maksimal dari hasil perbandingan dua buah string. Elemen ini merupakan substring yang terdapat kedua kontur yang dibandingkan. Proses string matching digunakan untuk mencari kesamaan bentuk kontur dengan mencocokkan elemen satu dengan yang lainnya. Metode ini hanya dapat berjalan dengan baik pada bentuk kontur yang non-bidirectional tetapi tidak untuk kontur yang bidirectional.

Kata Kunci : analisis bentuk, string matching, Boyer-Moore, persamaan bentuk, TrueType font, longest common substring.

Abstract

Most known manipulation and analysis techniques originate from image processing and pattern recognition. The problem is how to recognize a shape at computer system. The final project studies how to identify and create new letter type through string matching process. The goal is developing an application that people can make new letter shape within modifying the old letter shapes and without design process. The string matching algorithm that is used is Boyer-Moore algorithm and the font format that can be processed is TrueType Font.

To solve the problems, the project uses Longest Common Substring (LCS) and Boyer-Moore string matching method. The LCS method is used to find maximal element from comparison of two strings. This element is a substring that appears in two comparison contours. The string matching process is used to find shape similarities. These methods work only in non-bidirectional shape but not in bidirectional shape.

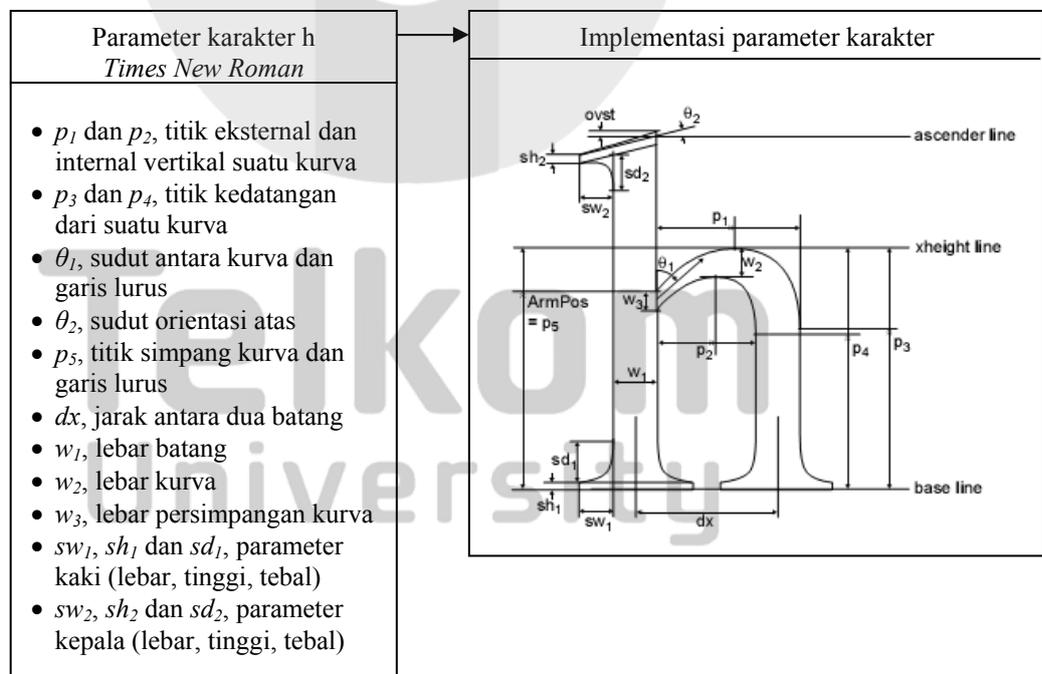
Keywords : shape analysis, shape modification, string matching, Boyer-Moore, shape similarities, TrueType font, longest common substring.

1. Pendahuluan

1.1 Latar Belakang

Huruf atau karakter merupakan simbol yang digunakan untuk menyusun kata. Huruf telah dikenal lama dan mengalami berbagai perubahan bentuk sejak awal keberadaannya. Bentuk huruf yang telah ada mungkin saja berbeda. Setiap bentuk karakter didisain berdasarkan pada aturan-aturan melalui pendekatan visual dimana manusia dapat mengenalinya. Di dalam dunia komputerisasi ini, karakter berkembang ke dalam bentuk yang bervariasi yang digunakan dalam berbagai aplikasi seperti *word processing*.

Karakter dirancang dengan mempertimbangkan gagasan dari desainer dan aturan-aturan yang berhubungan dengan tampilan visual yang disebut sebagai parameter. Parameter inilah yang menentukan bentuk dari suatu karakter (lihat gambar 1-1). Karakter-karakter ini telah menjadi suatu objek di dalam komputer yang dikembangkan programmer dan desainer untuk dapat digunakan pada bermacam piranti komputer seperti monitor dan printer. Masing-masing karakter dalam suatu jenis huruf digambarkan dengan sekumpulan garis. Ketika karakter tersebut dimasukkan ke dalam sistem pixel, garis-garis tersebut diskalakan sesuai dengan ukuran yang diinginkan, dan setiap pixel yang berada dalam garis tersebut diberi warna hitam.



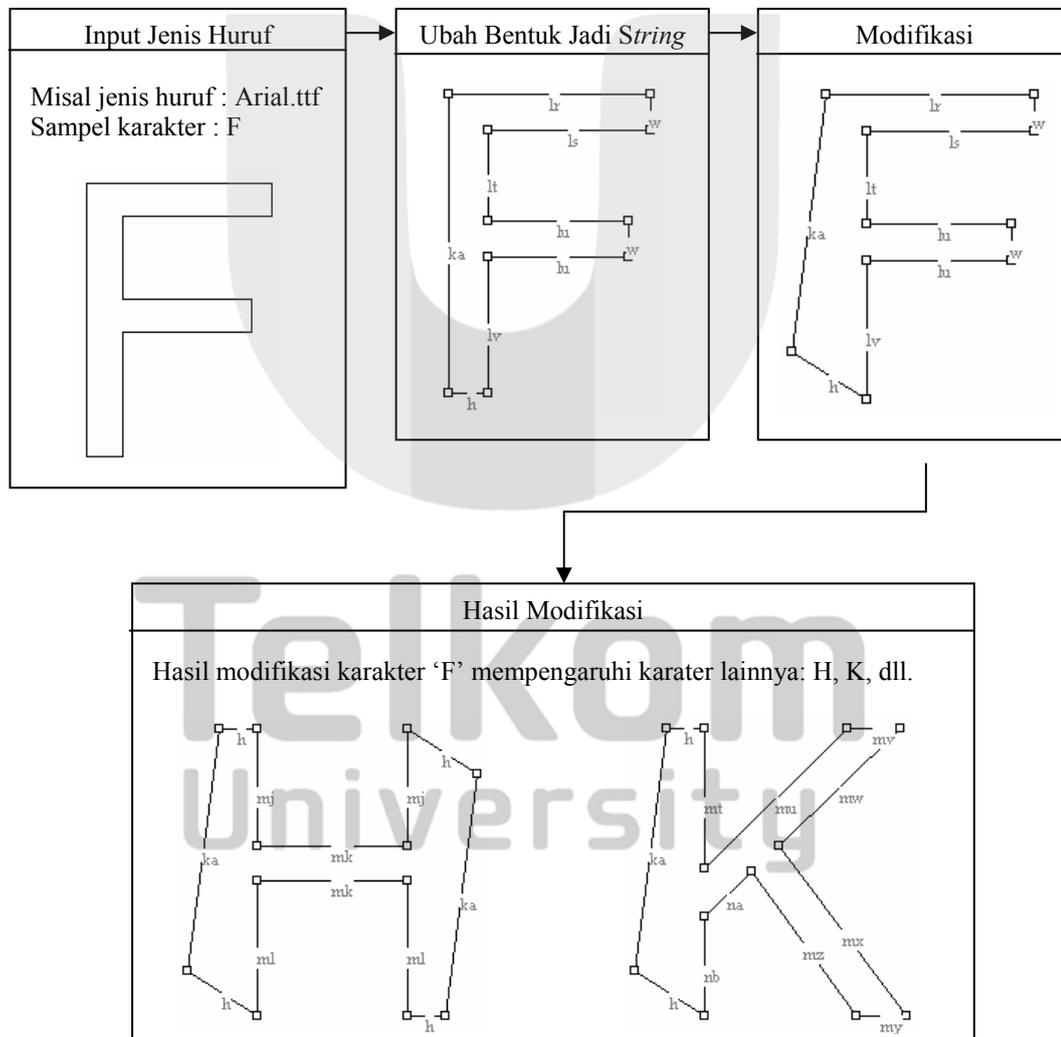
Gambar 1-1: Parameter pada karakter h *Times New Roman*

Tentunya akan lebih mudah membuat suatu jenis huruf apabila menyampingkan parameter tersebut. Namun permasalahannya adalah bagaimana membuat suatu jenis huruf tanpa memperhatikan parameternya tetapi masih sesuai dengan aturan desain huruf. Oleh karena itu diperlukan modifikasi bentuk suatu jenis huruf yang telah ada menjadi bentuk yang baru sehingga tidak perlu melakukan proses desain yang sedemikian rumit.

1.2 Perumusan Masalah

Permasalahan yang muncul dalam penulisan tugas akhir ini yaitu:

1. Bagaimana mengenali bentuk dan karakteristik dari suatu tipe huruf.
2. Bagaimana mendesain jenis huruf baru secara otomatis dari huruf yang telah ada dengan menggunakan pendekatan *string matching*.
3. Bagaimana membangun aplikasi untuk menerapkan poin (1) dan (2) sehingga dapat menghasilkan bentuk jenis huruf baru.



Gambar 1-2: Bagan sistem modifikasi suatu karakter

Adapun yang menjadi batasan masalah pada penulisan tugas akhir ini yaitu:

1. Algoritma *string matching* yang digunakan adalah algoritma Boyer Moore. Algoritma ini memiliki kompleksitas waktu yang baik, sehingga operasi yang dilakukan cukup cepat. Meskipun ada algoritma lain yang lebih cepat, namun untuk alasan praktis algoritma ini sudah cukup untuk digunakan dalam tugas akhir ini.
2. Aplikasi yang akan dibangun hanya digunakan pada sistem operasi Microsoft Windows.
3. Karakter yang diproses terbatas untuk 224 karakter ASCII, mulai dari ASCII ke-32 sampai ASCII ke-255.
4. Tipe huruf yang akan diproses adalah *True Type Font*.
5. Hasil modifikasi disimpan dalam bentuk file gambar berformat bitmap.

1.3 Tujuan

Tujuan dari penulisan tugas akhir ini yaitu membangun aplikasi dimana seseorang dapat menciptakan jenis huruf baru tanpa melakukan proses desain, tetapi dengan melakukan modifikasi bentuk karakter dari jenis huruf yang sudah ada.

1.4 Metodologi Penyelesaian Masalah

Metodologi penyelesaian masalah dalam penulisan tugas akhir ini yaitu:

1. Pengumpulan Data

Mencari informasi dari berbagai sumber seperti jurnal atau buku baik dari media elektronik maupun artikel-artikel mengenai topik karakteristik bentuk huruf, *string matching*, serta cara membangun suatu jenis huruf baru.

2. Perancangan Model

Mengidentifikasi dan mendefinisikan kebutuhan sistem melalui informasi dan data yang telah didapat. Kemudian merancang perangkat lunak yang akan dibangun dari segi fungsionalitas dan bentuk tampilan antar muka mengacu pada kebutuhan sistem yang telah didefinisikan.

3. Pembangunan dan Implementasi

Menterjemahkan tahap perancangan model di atas menjadi bahasa pemrograman. Adapun bahasa pemrograman yang digunakan adalah Pascal berbasis Delphi.

4. Pengujian dan Analisa Hasil

Pada tahap ini dilakukan pengujian terhadap berbagai jenis huruf dan melihat performansi sistem berdasarkan waktu. Hasil output dari aplikasi ini diimplementasikan untuk *word processing*.

5. Pembuatan Laporan

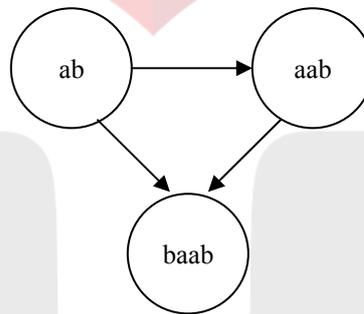
Membuat laporan serta menarik kesimpulan dari tahap-tahap diatas yang digunakan untuk penulisan laporan tugas akhir.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Kesimpulan yang dapat ditarik dari tugas akhir ini yaitu:

1. Modifikasi elemen hanya dapat dilakukan dari elemen minimum ke maksimum. Elemen minimum ini merupakan *substring* dari elemen maksimum. Artinya apabila elemen minimum dimodifikasi maka elemen maksimum ikut termodifikasi. Sedangkan apabila elemen maksimum dimodifikasi, elemen minimum tidak termodifikasi.



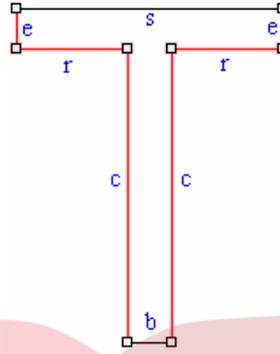
Gambar 5-1: Hubungan elemen minimum dengan maksimum

Hubungan elemen minimum dengan elemen maksimum berdasarkan gambar di atas:

Tabel 5-1: Hubungan elemen minimum dengan elemen maksimum

Elemen minimum	Elemen maksimum
ab	aab
ab	baab
aab	baab

2. Semakin banyak jumlah kontur, jumlah kurva dan kesamaan bentuk (elemen ekstraksi), maka semakin lama waktu yang diperlukan untuk memproses informasi karakternya.
3. Algoritma yang digunakan tidak dapat memproses bentuk yang *bidirectional*, yaitu elemen hasil pencerminan (refleksi). Misalnya elemen cre dan erc seharusnya dianggap sebagai elemen yang sama.



Gambar 5-2: Elemen refleksi dengan membandingkan *string* dengan kebalikannya

5.2 Saran

Untuk memperbaiki kekurangan perangkat lunak ini, diberikan saran sebagai berikut:

1. Gunakan kriteria lain sebagai penggolongan kurva agar kurva yang digolongkan tidak terlalu spesifik sehingga tidak terlalu banyak jenis kurva yang didapatkan. Misalnya kriteria kelengkungan, apakah kurva tersebut lengkung ke dalam atau keluar. Semakin banyak kurva yang digolongkan maka semakin lama waktu yang dibutuhkan untuk memprosesnya.
2. Gunakan proses *hinting* yang lain seperti *Model-Based Hinting*^[9] agar jumlah kurva yang didapatkan tidak terlalu banyak.
3. Pada perangkat lunak yang dikembangkan ini terbatas pada karakter ASCII-32 sampai ASCII-255. Diinginkan agar karakter yang diproses lebih dari itu, misalnya menggunakan karakter ANSII dan sebagainya.
4. Diinginkan agar hasil akhir dari penggunaan perangkat lunak ini berupa file huruf, bukan file bitmap.
5. Gunakan algoritma yang dapat memproses elemen hasil pencerminan (refleksi) seperti gambar 5-2 di atas.

Telkom
University

Daftar Pustaka

- [1] Adobe System Inc., *Adobe Type 1 Font Format*, Addison-Wesley, 1990.
- [2] Bambangwirawan, Pulus, *Grafik Komputer dengan C*, 2004.
- [3] Bezier Curve From Wikipedia, the free encyclopedia.
http://en.wikipedia.org/wiki/bezier_curve. Diakses tanggal 23 Juli 2007.
- [4] Boyer-Moore Algorithm. <http://www-igm.univ-mlv.fr/~lecroq/string>.
Diakses tanggal 21 Mei 2006.
- [5] Dempster, Kelly, *Focus on Curves and Surfaces*, 2003.
- [6] Dharma, Eddy Muntina, *Computer Graphic*, Bahan kuliah grafika citra STT Telkom, Bandung, 2006.
- [7] Gery, Ron, *Advanced TrueType: GetGlyphOutline*, Microsoft Developer Network Technology Group, 1992.
- [8] Hersch, Roger D, *Font Rasterization: The State of The Art*, Cambridge University Press, 1993.
- [9] Hersch, Roger D and Claude Betrisey, *Model-based Matching and Hinting of Fonts*. Computer Graphics Vol. 25., 1991.
- [10] Hertz, Jacky and Roger D. Hersch, *Analysing Character Shapes by String Matching Techniques*, Electronic Publishing, Vol. 6(3)., 1993.
- [11] Hsu, Wen-Jing, *New Algorithm for Comparing Symbol Sequences*, Departement of Computer Science, Michigan State University, 1987.
- [12] Microsoft Corporation, *Programmer's Guide to Pen Services For Microsoft Windows 95*, 1996.
- [13] Pressman, Roger S., *Software Engineering: A Practitioner's Approach*, Forth Edition, McGraw-Hill, 1997.