

# 1. Pendahuluan

## 1.1 Latar Belakang

*Optical Character Recognition* (OCR) adalah konversi mekanik atau elektronik dari gambar scan tulisan tangan, ketik, atau cetak ke dalam *encoded text* untuk komputer [10]. Kajian tentang OCR akhir-akhir ini memberikan metode dengan akurasi yang baik dan kecepatan komputasi yang semakin ringan. Komputasi yang semakin ringan memungkinkan mesin dengan spesifikasi rendah seperti *smartphone* sanggup mengerjakan OCR. Aplikasi OCR pada smart phone sudah banyak beredar di *marketplace* *smartphone* namun banyak yang diantaranya masih memakan waktu dan lama. Aplikasi yang cepat sangat penting untuk meningkatkan pengalaman pengguna.

Terdapat beberapa metode untuk OCR yaitu *Principal Component Analysis* (PCA), *Contour Analysis* (CA/ICF-ACF) [3], dan *Tesseract engine* [13]. *Tesseract* merupakan metode yang populer digunakan karena akurasinya yang tinggi dan bersifat open source. Namun *tesseract* membutuhkan waktu komputasi yang tidak cepat karena menggunakan *static classifier* [13]. *Contour Analysis* menggunakan metode *description extraction Intercorrelation function* (ICF) dan *Auto correlation function* (ACF) yang memiliki kompleksitas [3] lebih kecil dibanding kompleksitas metode lain diatas, juga *classifier* diskret yang berdasarkan template karakter. Beberapa kajian tentang *contour analysis* mempresentasikan kontur sebagai jarak tiap pixel *contour* ke sumbu x atau y [9] [15], namun untuk bisa menggunakan ACF dan ICF maka tiap piksel *contour* dipresentasikan dalam bilangan kompleks [3].

Perhitungan ACF dan ICF menjadi meningkat mengikuti banyaknya elemen *contour* dan sangat sensitif pada *noise*. Untuk mengatasi hal tersebut terdapat algoritma Ramer-Douglas-Peucker (RDP) yang merupakan algoritma penyerdehanaan *contour* dengan mengurangi *noise* dan membuang elemen yang melebihi toleransi [5]. Untuk meringankan perhitungan dapat digunakan *preselection* saat *matching* dimana dapat dilakukan perhitungan jarak sebelum dilakukan *matching* atau proses *preselection contour*[8]. *Descriptor* pada ACF dan ICF berbentuk serangkaian nilai yang dapat dilakukan *preselection* dengan algoritma Longest Common Sequence (LCS) [6] atau *euclidean distance*[7]. Algoritma LCS untuk *matching* membutuhkan nilai *threshold* tertentu dalam perhitungan kemiripan. LCS menunjukkan tangguh terhadap *noise*, *time shift* dan *data scaling* [10]

Pada Tugas Akhir ini diterapkan metode OCR, dengan menggunakan Contour Analysis dengan metode feature descriptor *Intercorrelation* dan *Autocorrelation* function di smartphone dengan optimasi contour menggunakan algoritma Ramer-Douglas-Peucker dan *preselection* menggunakan algoritma *longest common sequence* dan *euclidean distance*.

## 1.2 Perumusan Masalah

Berdasarkan masalah yang telah dikemukakan sebelumnya, maka masalah yang dirumuskan adalah sebagai berikut.

1. Bagaimana performansi OCR dengan pengurangan *noise contour* dengan algoritma Ramer-Douglas-Peucker ?
2. Bagaimana performansi OCR dengan diimplementasikan *preselection* sebelum *contour matching* dengan algoritma *longest common subsequence* dan perbandingannya dengan *euclidean distance* ?
3. Berapa akurasi OCR dengan metode *contour analysis* dengan ACF dan ICF?
4. Berapa lama waktu eksekusi?

Batasan dari studi kasus :

- a. Menggunakan *library fastcv* dari Qualcomm
- b. Karakter yang direkognisi adalah huruf latin dengan tipografi sans-serif.
- c. Sudut kemiringan teks dibatasi sampai 45°
- d. Ukuran teks pengambilan berkisar setinggi 24 – 56 piksel.

## 1.3 Tujuan

Tujuan dari tugas akhir ini adalah:

1. Menerapkan implementasi pengurangan noise contour dengan algoritma Ramer Douglas Peucker.
2. Membandingkan optimasi preselection menggunakan algoritma longest common sequence dengan euclidean distance.
3. Menganalisis performansi OCR dengan metode CA pada mobile phone.
4. Melakukan analisis waktu eksekusi OCR.

## 1.4 Hipotesa

Hipotesa dari tugas akhir ini adalah :

1. Contour Analysis dengan *feature extraction* dan descriptor ACF dan ICF dapat diimplementasikan ke perangkat mobile, karena kompleksitas tidak terlalu tinggi.
2. Menggunakan Contour Analysis dengan descriptor ACF dan ICF membandingkan nilai ACF dan ICF per template, maka akurasi akan tinggi untuk data train.

## **1.5 Metodologi penyelesaian masalah**

Metodologi yang digunakan untuk memecahkan masalah pada tugas akhir ini adalah

### **1.5.1 Identifikasi masalah**

Pada tahap ini, akan dilakukan brainstorming dan identifikasi mengenai fenomena yang dibahas di Tugas Akhir ini, yaitu proses OCR menggunakan metode CA dengan *descriptor* ACF dan ICF.

### **1.5.2 Studi literatur**

Pada tahap ini, akan dilakukan pencarian terhadap materi-materi guna mendukung penulisan tugas akhir ini. Refrensi yang dicari antara lain adalah jurnal, artikel, maupun buku mengenai *Contour Analysis*, OCR, fastCV, bahasa pemograman.NET, *Windows Phone* dan studi kasus yang berhubungan dengan OCR

### **1.5.3 Analisis dan perancangan kebutuhan sistem**

Pada tahap ini, akan dilakukan perancangan sebuah sistem OCR dengan menggunakan metode yang telah dijelaskan. Dan juga perancangan skenario interaksi sistem untuk menginput template dan memeriksa akurasi.

### **1.5.4 Implementasi sistem**

Pada tahap ini akan dilakukan coding atau implementasi berdasarkan rancangan yang telah dibuat sebelumnya.

### **1.5.5 Pengujian sistem**

Pada tahap ini akan dilakukan testing atau pengujian terhadap sistem yang telah diimplementasikan apakah sistem mampu mendeteksi dan membaca input karakter dengan baik sehingga berinteraksi dengan lingkungan OCR yang sudah terdefinisi. Pengujian sistem juga mencakup nilai-nilai konfigurasi LCS dan RDP untuk mencari akurasi terbaik.

#### **1.5.6 Analisis hasil pengujian dan pengambilan keputusan**

Sistem yang telah dirancang dengan menggunakan CA ini diuji dengan menggunakan hasil semua capture yang diambil dari media input(kamera smart phone). Lalu, dari hasil yang didapatkan akan dilakukan analisis. Analisis difokuskan pada performansi CA berdasarkan rata-rata waktu pemrosesan tiap frame dan ketepatan menentukan recognisi karakter.

#### **1.5.7 Perumusan kesimpulan dan penyusunan tugas akhir**

Pada tahap ini, akan dilakukan perumusan kesimpulan berdasarkan analisis dari hasil implementasi sistem yang telah dilakukan pada tahap sebelumnya. Setelah diambil kesimpulan, akan dilakukan penyusunan skripsi dan pengumpulan dokumentasi dengan mengikuti aturan-aturan yang telah ditetapkan oleh institusi.