BAB 1 PENDAHULUAN

1.1 Latar Belakang

Aksara Sunda merupakan bentuk penulisan dari bahasa Sunda yaitu bahasa daerah yang berasal dari Jawa Barat, salah satu aksara tradisi hasil karya ortografi masyarakat Sunda melalui perjalanan sejarahnya sejak 5 abad yang lalu hingga saat ini [1]. Aksara tersebut digunakan untuk sarana komunikasi yang dikenal dengan Bahasa Sunda, salah satu bahasa yang memiliki jenis huruf tersendiri yang di kenal dengan aksara Ngalagena atau Konsonan, aksara Swara atau Vocal, aksara Rarangken dan aksara Angka, yang saat ini kurang dipahami masyarakat Jawa Barat. Hal ini terjadi karena minimnya pengetahuan tentang pengenalan huruf tersebut.

Pengenalan aksara Sunda biasanya dimulai dari jenis aksara Ngalagena yang merupakan aksara dasar sebagai proses pembelajaran. Aksara Ngalagena terdiri dari 18 buah aksara yang merupakan huruf untuk bunyi dalam bahasa Sunda yaitu ka, ga, nga, ca, ja, nya, ta, da, na, pa, ba, ma, ya, ra, la, wa, sa, ha , fa, qa, va, xa, za , kha, sya. Setiap huruf memiliki perbedaan bentuk satu dengan lainnya namun ada beberapa huruf yang mempunya kesamaan. Dengan banyaknya jenis huruf dan kurangnya pembelajaran dalam pengenalan aksara Sunda dalam hal ini sangat menyulitkan sebagian masyarakat Jawa Barat untuk mengenali terutama yang bukan masyarakat Sunda. Untuk mempermudah pengenalan aksara Sunda Ngalagena dibutuhkan sebuah sistem aplikasi yang dapat membantu mengenali huruf yang satu dengan yang lainnya. Salah satu cara untuk menghasilkan sistem aplikasi dibutuhkan algoritma yang dapat membantu mengenalkan aksara Sunda yaitu algoritma *Template Matching*, adapula algoritma *Canny*.

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG

Algoritma *Template Matching* merupakan metode yang sederhana dan banyak digunakan untuk mengenali pola. Metode *Template Matching* ini merupakan metode yang sederhana. Suatu citra masukan yang mengandung *template* tertentu dibandingkan dengan *template* pada basis data. *Template* ditempatkan pada pusat bagian citra yang akan dibandingkan dan dihitung seberapa banyak titik yang paling sesuai dengan *template*. Langkah ini diulangi terhadap keseluruhan citra masukan yang akan dibandingkan. Nilai kesesuaian titik yang paling mendekati atau *error* minimum yang paling kecil antara citra masukan dan citra *template* menandakan bahwa *template* tersebut merupakan citra *template* yang paling sesuai dengan citra masukan [2].

Sedangkan cara kerja dari Algoritma *Canny* ini pertama-tama yaitu menghaluskan gambar dengan menyingkirkan *noise*. Kemudian ditemukannya *gradient* gambar untuk menyoroti daerah yang mempunyai *false space* yang tinggi. Algoritma canny ini kemudian mengambil jalur sepanjang ruang tersebut dan menekan piksel yang tidak maximum (penekanan non-maximum). Dengan menggunakan metode *Canny*, tepian yang dihasilkan lebih jelas, atau dapat dikatakan perbedaan tepian dengan *background* citra terlihat nyata [3].

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG

Dari penjelasan di atas, terdapat kesimpulan bahwa dibutuhkannya perbandingan algoritma untuk menentukan keakurasian dalam proyek aplikasi yaitu **"Perbandingan Algoritma *Template Matching* dan *Canny* untuk Pengenalan Aksara Sunda"**. Penggunaan algoritma *Template Matching* dan *Canny* ini karena keduanya merupakan metode yang sangat sederhana dan banyak digunakan untuk pengenalan pola.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang diuraikan di atas, dapat penyusun dapat merumuskan bahwa masalah yang timbul adalah:

1. Bagaimana mengimplementasikan algoritma *Template Matching* dan *Canny* agar dapat mengenali karakter Aksara Sunda?
2. Seberapa tingkat akurasi hasil pengenalan karakter Aksara Sunda yang menerapkan algoritma *Template Matching* dan *Canny* ?

1.3 Tujuan Penelitian

Tujuan dari penelitian ini untuk mengimplementasikan algoritma *Template Matching* dan *Canny* ke dalam sistem pengenalan Aksara Sunda serta mengetahui seberapa tingkat akurasi hasil pengenalan Aksara Sunda pada program yang telah menerapkan algoritma *Template Matching* dan *Canny*.

1.4 Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah :

1. *Input* berupa coretan tangan atau gestur.
2. Pengenalan pola karakter Aksara Sunda hanya terbatas pada 42 karakter Aksara.
3. Sistem yang dibuat berbasis *android*.
4. *Output* yang akan dikeluarkan hanya berupa “Benar” dan “Salah” bukan terjemahan dari Aksara Sunda.
5. Data inputan hanya per huruf.

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG

1.5 Kerangka Pemikiran

Adapun kerangka pemikiran dari penelitian tugas akhir ini yang digambarkan pada tabel 1.1

Tabel 1.1 Kerangka Pemikiran

Problems
– Bagaimana menentukan keakurasian metode *Tamplate Matching* dan *Canny* dalam pengenalan Aksara Sunda.

Opportunity
Membuat aplikasi perbandingan antara metode *Tamplate Matching* dan *Canny* untuk pengenalan Aksara Sunda.

↓

Approach	
Metode *Tamplate Matching* dan *Canny*	Teknologi : *Android*

↓

Software Development	
Metode pengembangan : *Prototype*	Alat analisis pengenalan : *UML*

↓

Software Implementation
Perbandingan metode *Tamplate Matching* dan *Canny* untuk pengenalan Aksara Sunda.

↓

Result

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG

Metode dengan tingkat akurasi yang akurat.

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG

Tabel 1.1 merupakan alur kerangka pemikiran yang dapat dijelaskan masalah yang ada yaitu bagaimana menentukan keakurasian metode *Template Matching* dan *Canny* dalam pengenalan Aksara Sunda. Dari permasalahan tersebut, dapat solusi untuk menyelesaikan masalah di atas. Yaitu dengan membuat aplikasi perbandingan antara metode *Template Matching* dan *Canny* untuk pengenalan Aksara Sunda. Metode yang digunakan yaitu *Template Matching* dan *Canny* untuk pengenalan Aksara Sunda, prototype untuk metode pengembangan dan UML untuk alat analisis. Dari penjelasan di atas, terdapat hasil dari semua proses yaitu "Perbandingan metode *Template Matching* dan *Canny* untuk pengenalan Aksara Sunda". Apabila dalam hasil tersebut kurang menjawab masalah dan solusi, maka kekurangan tersebut dimasukan kedalam masalah kembali, sampai didapatkan hasil yang sesuai.

1.5.1 Metode Penelitian

1. Teknik Pengumpulan Data

Teknik pengumpulan data dalam penelitian kali ini adalah sebagai berikut :

a. Studi pustaka yaitu dengan mendeskripsikan data yang diperoleh dari berbagai sumber referensi ang berhubugan dengan permasalahan ang sedang diteliti, untuk kemudian dianalisa, dan diuraikan dalam bentuk deskripsi yang dilatar belakangi oleh dana konsep – konsep dan teori yang dikemukakan dalam landasan teori.

b. *Observasi*, mencari dan mengumpulkan data, dimana data – data yang ada relevansinya dengan judul tugas akhir ini.

c. Media internet adalah salah satu media elektronik yang memiliki sumber bacaan yang berhubungan dengan judul tuugas akhir yang diambil.

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG

2. Tahap Penelitian

Penelitian ini dilakukan dengan langkah – langakah sebagai berikut :

a. Menguraikan konsep algoritma *Template Matching* dan *Canny* dalam pengenalan Aksara Sunda.

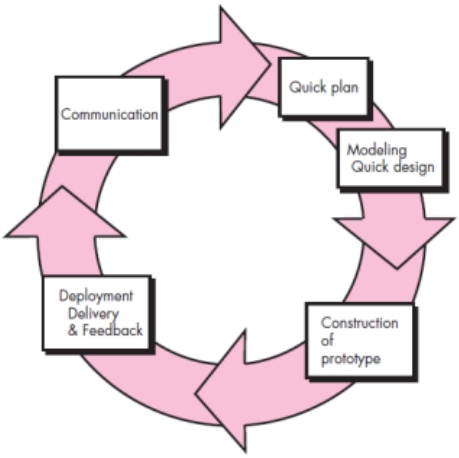
b. Mengimplementasikan algorittma *Template Matching* dan *Canny* ke dalam suatu program.

c. Melakukan analisa untuk membandingkan kinerja setiap algoritma berdasarkan kelebihan dan kemudahannya.

d. Membuat kesimpulan dan saran dari penelitian yang dilakukan.

3. Metode Pengembangan

Adapun metode pengembangan perangkat lunak yang akan dibuat yaitu menggunakan metodologi *Prototype*. Gambar 1.2 merupakan alur dari metode *Prototype*.



Gambar 1.1 Metode *Prototype* [4].

Metode *prototype* dimulai dari tahap komunikasi. Tim pengembang perangkat lunak melakukan pertemuan dengan para *stakeholder* untuk menentukan kebutuhan perangkat lunak yang saat itu diketahui dan untuk

menggambarkan area-area dimana definisi lebih jauh untuk iterasi selanjutnya.

Perencanaan iterasi pembuatan prototipe dilakukan secara cepat. Setelah itu dilakukan pemodelan dalam bentuk “rancangan cepat”. Pembuatan rancangan cepat berdasarkan pada representasi aspek-aspek perangkat lunak yang akan terlihat oleh para end user (misalnya rancangan antarmuka pengguna atau format tampilan). Rancangan cepat merupakan dasar untuk memulai konstruksi pembuatan *prototype*.

Prototype kemudian diserahkan kepada para *stakeholder* untuk mengevaluaasi *prorotype* yang telah dibuat sebelumnya dan memberika umpan balik yang akan digunakan untuk mempeluas spesifikasi kebutuhan. Iterasi akan terjadi pada saat *prototype* diperbaiki untuk memenuhi kebutuhan dari para *stakeholder*.

1.6 Sistematika Penulisan

Sistematika penulisan buku Tugas Akhir (TA) ini adalah sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini akan dijelaskan mengenai latar belakang masalah, manfaat dan tujuan penelitian, dan sistematika penulisan laporan.

BAB II DASAR TEORI

Pada bab ini akan dijelaskan mengenai uraian tinjauan pustaka dan landasan teori yang dapat dipergunakan sebagai pembanding atau acuan di dalam pembahasan masalah oleh penulis.

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG

BAB III ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini akan dijelaskan mengenai tahap-tahap perancangan sistem yang akan dibuat.

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan dijelaskan mengenai gambaran implementasi sistem dan juga hasil pengujian sistem.

BAB V KESIMPULAN DAN SARAN

Pada bab ini berisi kesimpulan penelitian secara keseluruhan dan saran yang diberikan untuk pengembangan sistem lebih lanjut.

LAMPIRAN

Berisi semua dokumen yang digunakan dalam proses penyusunan dan perancangan seperti source code, kelengkapan dokumen dan lain sebagainya

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG