

BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Adapun beberapa penelitian yang berkaitan dengan OCR dan *template matching* pada karakter alfabet latin maupun non alfabet adalah sebagai berikut:

Hartanto dkk. (2012) dengan judul penelitian *Optical Character Recognition Menggunakan Algoritma Template Matching Correlation*. Dalam penelitian ini menerapkan OCR menggunakan metode *template matching correlation* untuk mengenali paragraf yang terdiri dari 263 karakter alfabet latin dimana jenis huruf dan ukuran yang digunakan berbeda-beda. Proses pengujian pengenalan citra dilakukan sebanyak tiga kali dengan menggunakan ukuran huruf 12pt, 20pt, 36pt dan jenis huruf *Arial*, *Times New Roman*, *Comic Sans MS*, *Cambria* dan *Courier New*. Dari pengujian pertama didapatkan hasil persentase pengenalan sebesar 92, 24%, pada pengujian kedua 93, 54% dan pada pengujian ketiga sebesar 92, 93%. Didapatkan rata-rata tingkat keberhasilan pengenalan cukup tinggi yaitu 92, 20% walaupun jenis dan ukuran huruf yang digunakan sebagai penguji berbeda dengan *template*.

Dalam jurnal yang ditulis Hendry, J., dan Hidayat, R. (2011) yang berjudul *Template Matching Untuk Deteksi Obyek Citra Dengan Menggunakan Algoritma Korelasi* dikatakan peneliti melakukan pengenalan citra menggunakan *template matching correlation* terhadap citra RGB. Pengujian dilakukan terhadap beberapa kombinasi citra yang mungkin yaitu kombinasi berupa letak citra masukan yang berbeda dan adanya *noise* pada citra. Pada penelitian ini juga ditunjukkan pentingnya pemilihan nilai ambang yang tepat sebagai *threshold* pada waktu proses binerisasi. Dari penelitian juga dapat dikatakan bahwa dengan citra yang sama, semakin banyak *noise* maka semakin kecil nilai koefisien korelasi antara

citra *template* dengan citra masukan. Ini dikarenakan *noise* menyamarkan bentuk asli dari citra masukan, namun dengan mengubah nilai ambang maka citra masih dapat dideteksi, walaupun akan menyulitkan jika terdapat beberapa obyek dalam satu citra dengan bentuk hampir sama tetapi jenisnya berbeda.

Dalam penelitiannya yang berjudul Perbandingan Algoritma *Template Matching* dan *Feature Extraction* pada *Optical Character Recognition*, Bahri, R. S., dan Maliki, I. (2012) mengatakan algoritma *feature extraction* memiliki tingkat akurasi yang lebih tinggi dan waktu yang lebih singkat dibandingkan *template matching* tetapi *template matching* membutuhkan jumlah proses yang lebih sedikit karena tidak membutuhkan ekstraksi ciri terlebih dahulu. Pada penelitian tersebut *feature extraction* membutuhkan proses pendeteksian keterbukaan citra, pendeteksian perpotongan garis tengah citra, perhitungan jumlah piksel tiap blok, perhitungan jumlah garis vertikal dan horisontal dan proses klarifikasi sedangkan *template matching* hanya membutuhkan pemetaan intensitas piksel karakter yang akan dikenali, perhitungan eror dan pencarian nilai eror minimum.

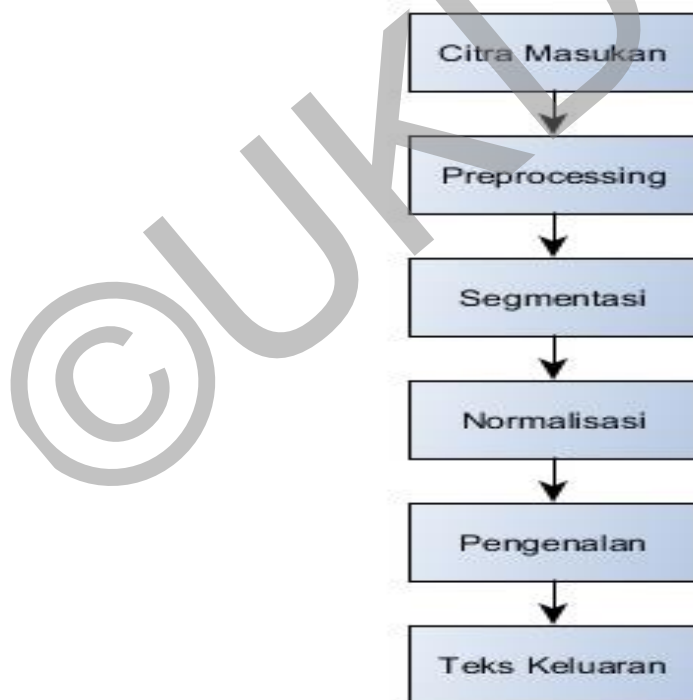
Putri dkk. (2013) dalam penelitiannya yang berjudul Klasifikasi Huruf Korea (Hangul) dengan Metode *Template Matching Correlation*, melakukan pengujian pengenalan citra huruf hangul Korea menggunakan *template matching correlation*. Dalam penelitiannya digunakan 5 jenis huruf hangul yaitu batang, *un jamu dotum*, *un pen*, *un shinmun* dan *un yetgul*. Dari ujicoba yang dilakukan didapatkan tingkat keakuratan cukup tinggi sebesar 83,33% dari 24 data uji yang digunakan dan disimpulkan bahwa posisi huruf dan jenis font yang digunakan akan sangat berpengaruh terhadap hasil klasifikasi.

2.2 Landasan Teori

2.2.1 Optical Character Recognition (OCR)

OCR adalah sebuah aplikasi komputer yang digunakan untuk mengenali citra huruf maupun angka untuk dikonversi ke dalam bentuk *file* tulisan. Sistem pengenalan huruf ini dapat meningkatkan fleksibilitas atau kemampuan dan kecerdasan sistem komputer. Sistem pengenalan huruf yang cerdas 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 (Hartanto, Sugiharto, & Endah, 2012).

Secara umum proses OCR yang digunakan dapat dilihat pada gambar 2.1.



Gambar 2.1. Proses OCR secara Umum

Pada tahap *preprocessing* dibagi menjadi 4 tahap yaitu *grayscale* dan binerisasi, setelah itu dilakukan segmentasi untuk memisahkan tiap karakter dan

normalisasi ukuran untuk menyesuaikan ukuran citra masukan dengan citra *template*. Terakhir dilakukan proses pengenalan untuk mencari citra *template* yang paling mirip dengan citra masukan.

2.2.2 Huruf Kanji

Huruf kanji adalah huruf yang berasal dari Cina pada tahun 1.500 SM lalu masuk ke Jepang sekitar abad ke-4 yang berjumlah sekitar 50.000 huruf kemudian dari beberapa kanji dikembangkan menjadi huruf hiragana dan katakana. Kanji adalah salah satu jenis huruf yang dipergunakan dalam bahasa Jepang dan mempunyai ciri tersendiri terutama dalam cara baca dan cara penulisannya, oleh karena itu kanji sering disebut sebagai huruf yang sangat rumit dan sukar untuk dipelajari namun demikian merupakan salah satu huruf yang sangat penting dalam bahasa Jepang karena setiap huruf menyatakan arti (Renariah, 2002).

Pada awal mulanya kira-kira abad ke-3 SM bangsa Jepang belum mempunyai bahasa tulisan sama sekali, namun mereka telah memiliki bahasa lisan sehingga pada akhirnya mereka meminjam sistem penulisan milik bangsa Cina yang merupakan bangsa tetangga mereka. Pada saat kanji masuk ke Jepang disertai dengan pengucapannya secara asli dalam bahasa Cina Kan yang disebut *on-yomi* tetapi kanji juga memiliki cara pengucapan yang merupakan terjemahan ke dalam bahasa Jepang asli yang disebut *kun-yomi* sehingga dalam satu huruf kanji bisa memiliki 1 atau 2 cara pengucapan.

Dalam kamus besar Kanji bahasa Jepang yaitu *kanwa daijiten* yang merupakan kamus kanji terlengkap di Jepang, di dalamnya terdapat sekitar 50.000 huruf kanji. Bagi orang Jepang sendiri jumlah tersebut merupakan jumlah yang sangat banyak, terlebih bagi pembelajar bahasa Jepang yang tidak memiliki latar belakang kanji, dimana mereka terbiasa dengan huruf alfabet yang berjumlah 26 huruf. Telah banyak diakui oleh pembelajar bahasa Jepang dan tidak sedikit yang merasakannya bahwa kanji adalah bidang yang sulit dipelajari dalam bahasa Jepang, terutama oleh para pembelajar bahasa Jepang yang tidak memiliki latar

belakang budaya kanji, yang dalam bahasa Jepang disebut *hikanjiken*. Hal tersebut disebabkan oleh banyak faktor, di antaranya adalah cukup banyak jumlah kanji yang harus diingat, cara baca kanji yang bervariasi, cara penulisan yang harus diperhatikan, serta pengetahuan kanji dalam menentukan dan pembentukan arti (Renariah, 2002).

一	丁	七	万	丈
井	丿	乂	乃	久
二	于	云	互	五

Gambar 2.2. Contoh huruf pada kanji Jepang

2.2.3 Grayscale

Menurut Basuki, A. (2005) *grayscale* digunakan untuk menyederhanakan model citra. Pada citra berwarna terdapat 3 *layer* matriks yaitu R-Layer, G-Layer dan B-Layer, R-Layer merupakan *layer red* atau warna merah, G-Layer merupakan *layer green* atau warna hijau dan B-Layer merupakan *layer blue* atau warna biru. Bila setiap proses perhitungan dilakukan menggunakan tiga *layer*, berarti dilakukan tiga perhitungan yang sama. Dengan demikian konsep itu dirubah dengan mengubah tiga *layer* diatas menjadi 1 *layer* matriks *grayscale* dimana hasilnya adalah citra *grayscale*. Dalam citra ini tidak ada lagi warna, yang ada adalah derajat keabuan.

Persamaan yang dapat digunakan untuk mengubah citra RGB menjadi *grayscale* dapat dilihat pada persamaan 2.1.

$$Grayscale = R \times 0,299 + G \times 0,587 + B \times 0,144 \quad [2.1]$$

Contoh penerapan persamaan 2.1 adalah sebagai berikut, misal sebuah piksel pada citra RGB memiliki nilai warna 24bit sebesar 15790335. Nilai

tersebut lalu dikonversi dari desimal menjadi biner dengan cara dilakukan operasi modulus dengan 2, dan didapatkan nilai biner sebagai berikut yaitu 111100001111000011111111. Dari nilai RGB yang sudah dikonversi menjadi biner tersebut akan dicari lagi nilai masing-masing *layer*-nya, dengan cara dilakukan operasi logika AND.

Nilai R = 111100001111000011111111

0000000000000000000011111111

----- AND

0000000000000000000011111111

Lalu untuk nilai G, geser 8 kali ke kanan dan lakukan lagi operasi logika AND.

Nilai G = 000000001111000011110000

0000000000000000000011111111

----- AND

0000000000000000000011110000

Lalu untuk nilai B, geser 8 kali ke kanan dan lakukan sekali lagi operasi logika AND.

Nilai B = 0000000000000000000011110000

0000000000000000000011111111

----- AND

0000000000000000000011110000

Sehingga diperoleh nilai masing-masing *layer* yaitu, 11111111 untuk R-Layer, 11110000 untuk G-Layer dan 11110000 untuk B-Layer. Jika dikonversi

dari biner menjadi desimal maka akan diperoleh triplet RGB (255, 240, 240). Dari nilai triplet RGB ini lalu dimasukkan ke dalam persamaan 2.1.

$$\text{Grayscale} = 255 \times 0,299 + 240 \times 0,587 + 240 \times 0,144 = 251$$

Dari perhitungan diatas, maka didapatkan nilai grayscale dari nilai RGB 15790335 yaitu 251.

2.2.4 Binerisasi

Pada tahap proses binerisasi, file citra digital dikonversi menjadi citra biner. Citra biner (*binary image*) adalah citra yang hanya memiliki dua nilai derajat keabuan, yaitu hitam dan putih. Pixel-pixel objek bernilai 1 dan pixel-pixel latar belakang bernilai 0. Pada waktu menampilkan gambar, 0 adalah putih dan 1 adalah hitam. Jadi, pada citra biner, latar belakang berwarna putih sedangkan objek berwarna hitam. Konversi dari citra hitam-putih ke citra biner dilakukan dengan menggunakan operasi *thresholding*, dimana akan dilakukan pengelompokan nilai derajat keabuan setiap pixel ke dalam 2 kelas, yaitu hitam dan putih (Hartanto, Sugiharto, & Endah, 2012).

Operasi *thresholding* dilakukan dengan menggunakan nilai T sebagai acuan untuk memutuskan sebuah piksel diubah menjadi hitam atau putih. Persamaan yang digunakan untuk menentukan T adalah $T = (f_{maks} + f_{min})/2$. f_{maks} adalah nilai intensitas maksimum pada citra dan f_{min} adalah intensitas minimum pada citra. Jika $f(x,y)$ diasumsikan sebagai nilai intensitas piksel pada koordinat (x, y) maka piksel tersebut diubah menjadi hitam atau putih berdasarkan kondisi berikut:

$$F(x, y) = 0, \text{ jika } f(x,y) \leq T \quad [2.2]$$

$$F(x, y) = 1, \text{ jika } f(x, y) > T$$

Sebagai contoh dari penerapan binerisasi pada sebuah piksel adalah sebagai berikut, diasumsikan pada sebuah citra *grayscale* memiliki intensitas maksimum 240 dan intensitas minimum 120 maka akan didapat nilai ambang atau T sebesar 180. Misal kita menggunakan nilai piksel hasil perhitungan dari sub-bab 2.2.3

yaitu 251, dan dimasukkan ke persamaan 2.2 maka akan didapatkan nilai biner 1 atau warna putih karena $251 > 180$.

2.2.5 Segmentasi

Langkah selanjutnya adalah melakukan *segmentation* atau segmentasi. Menurut Menurut Hartanto dkk. (2012) segmentasi bertujuan untuk mengelompokkan piksel pada suatu citra menjadi suatu wilayah yang merepresentasikan suatu objek. Pada penelitian ini digunakan metode *projection profiled based histogram*, dimana akan dilakukan segmentasi secara segmentasi baris berdasarkan histogram vertikal dan segmentasi karakter berdasarkan histogram horisontal. Hendry, J. (2011) dalam penelitiannya mengatakan metode ini mampu mengenali ruang antar baris dan kolom pada karakter sehingga dapat digunakan untuk melakukan segmentasi pada suatu citra teks yang memiliki karakter, jarak baris dan jarak kolom yang bervariasi, selain itu metode ini juga baik untuk karakter yang memiliki sub-karakter yang terpisah.

Cara kerja metode ini sangat sederhana, yaitu dengan menjumlahkan nilai intensitas tiap piksel bernilai 1 pada tiap baris untuk mendapatkan proyeksi vertikal dan tiap kolom untuk mendapatkan proyeksi horisontal. Persamaan yang digunakan adalah sebagai berikut:

$$VPP(y) = \sum_{i=0}^{w-1} f(i, y) \quad [2.3]$$

$$HPP(x) = \sum_{j=0}^{h-1} f(x, j) \quad [2.4]$$

Penjelasan persamaan 2.3 dan 2.4 adalah sebagai berikut:

$VPP(y)$ = *Vertical Projection Profile*.

$HPP(x)$ = *Horizontal Projection Profile*.

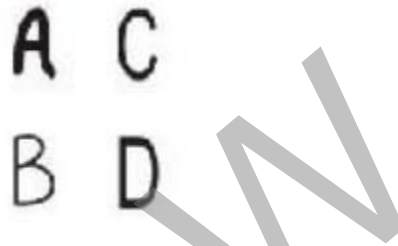
h = Tinggi citra.

w = Lebar citra.

$f(i,y)$ = Nilai piksel pada koordinat (i,y) .

$f(x,j)$ = Nilai piksel pada koordinat (x,j) .

Contoh penerapan projection profile adalah sebagai berikut, diasumsikan Gambar 2.3 sebagai citra masukan.



Gambar 2.3. Contoh Citra Masukan

Dikutip dari: Fatmawati, U. (2014). Contoh Citra Masukan.

<http://pta.trunojoyo.ac.id/uploads/journals/090411100128/090411100128.pdf>

Seperti yang sudah dijelaskan pada paragraf sebelumnya, segmentasi pada penelitian ini dilakukan dengan 2 tahap yaitu segmentasi baris dengan menghitung *VPP* tiap baris dan segmentasi karakter dengan menghitung *HPP* tiap kolom. Diasumsikan citra pada Gambar 2.3 sudah melalui tahap *grayscale* dan binerisasi sehingga menghasilkan matriks pada Tabel 2.1.

Tabel 2.1.

Matriks dari Gambar 2.3

0	0	1	1	1	0	0	0	0	1	1	1	0	0	6
0	1	1	0	1	1	0	0	1	1	1	1	0	0	8
0	1	1	1	1	1	0	0	1	1	0	0	0	0	7
0	1	0	0	0	1	0	0	1	1	1	1	0	0	6
0	1	0	0	0	1	0	0	0	1	1	1	0	0	5
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabel 2.1 (Sambungan)

0	1	1	1	1	0	0	0	1	1	1	1	0	0	8
0	1	0	0	1	1	0	0	1	0	0	0	1	0	5
0	1	1	1	1	0	0	0	1	0	0	0	1	0	6
0	1	0	0	1	1	0	0	1	0	0	0	1	0	5
0	1	1	1	1	0	0	0	1	1	1	1	0	0	8

Tabel paling kanan dari tabel 2.1 merupakan hasil perhitungan *VPP* dari tiap baris matriks. Berdasarkan hasil tersebut, maka dapat dihasilkan proyeksi vertikal seperti pada gambar 2.4. Berdasarkan proyeksi vertikal yang sudah diperoleh dapat dilakukan segmentasi baris sehingga menghasilkan 2 baris yaitu, baris pertama berisi huruf A dan C sedangkan baris kedua berisi huruf B dan D.



Gambar 2.4. Contoh Hasil Proyeksi Vertikal Citra Masukan

Dikutip dari: Fatmawati, U. (2014). Contoh Hasil Proyeksi Vertikal Citra Masukan.

(<http://pta.trunojoyo.ac.id/uploads/journals/090411100128/090411100128.pdf>)

Selanjutnya akan dilakukan segmentasi karakter pada tiap baris yang didapatkan dari segmentasi baris, pada contoh berikut hanya dilakukan pada baris pertama saja. Tabel 2.2 menunjukkan matriks dari baris pertama yang berisi karakter A dan C, dan tabel paling bawah dari tabel 2.2 merupakan hasil perhitungan *HPP* dari tiap kolom matriks.

Tabel 2.2.

Matriks dari Baris Pertama Gambar 2.3

0	0	1	1	1	0	0	0	0	1	1	1	0	0
0	1	1	0	1	1	0	0	1	1	1	1	0	0

Tabel 2.2 (Sambungan)

0	1	1	1	1	1	0	0	1	1	0	0	0	0
0	1	0	0	0	1	0	0	1	1	1	1	0	0
0	1	0	0	0	1	0	0	0	1	1	1	0	0

0	4	3	2	3	4	0	0	3	5	4	4	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---

Berdasarkan hasil *HPP* tersebut, maka akan dihasilkan proyeksi horisontal seperti pada gambar 2.5 yang dapat digunakan untuk melakukan segmentasi karakter pada baris pertama.



Gambar 2.5. Hasil Segmentasi Baris Pertama

Dikutip dari: Fatmawati, U. (2014). Hasil Segmentasi Baris Pertama.

(<http://pta.trunojoyo.ac.id/uploads/journals/090411100128/090411100128.pdf>)

2.2.6 Normalisasi

Normalization atau normalisasi merupakan salah satu tahap dalam *OCR* yang dilakukan sebelum masuk ke proses pengenalan. Proses ini dilakukan dengan tujuan untuk menyesuaikan data citra masukan dengan data citra pada basis data. Salah satu proses normalisasi yang paling sederhana dalam pengenalan pola adalah normalisasi ukuran citra. Sebagai contoh, pada proses pengenalan pola yang menggunakan algoritma *template matching*, ukuran citra input harus disesuaikan dengan ukuran citra yang ada di basis data (Hartanto, Sugiharto, & Endah, 2012).

Pada penelitian ini normalisasi dilakukan dengan metode *resize*, dimana seluruh citra karakter hasil dari proses segmentasi akan di-*resize* menjadi ukuran 45 x 35 piksel. Proses ini dilakukan supaya seluruh data citra menjadi seragam.

2.2.7 Template Matching

Tahapan dalam proses pengenalan citra dilakukan dengan menggunakan algoritma *template matching* yang berfungsi untuk mencocokkan tiap-tiap bagian dari suatu citra dengan citra yang menjadi *template* (acuan). Citra masukan dibandingkan dengan citra *template* yang ada di dalam basis data, kemudian dicari kesamaannya dengan menggunakan suatu aturan tertentu (Hartanto, Sugiharto, & Endah, 2012).

Pada penelitian ini digunakan algoritma template matching *Normalized Cross Correlation Coefficient* (NCC) untuk mengenali pola dari huruf kanji. Contoh penerapan proses pengenalan karakter alfabet dengan NCC, dimana piksel citra biner dengan warna hitam direpresentasikan dengan 1 sedangkan putih dengan 0. Persamaan yang digunakan adalah sebagai berikut:

$$\frac{\sum m \sum n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum m \sum n (A_{mn} - \bar{A})^2)(\sum m \sum n (B_{mn} - \bar{B})^2)}} \quad [2.5]$$

Penjelasan persamaan 2.5 adalah sebagai berikut:

A = Nilai piksel citra latih A pada posisi m, n.

A aksen = Nilai rata-rata citra latih A.

B = Nilai piksel citra uji B pada posisi m, n.

B aksen = Nilai rata-rata citra uji B.

Sebagai contoh terdapat 3 matriks citra sebagai berikut:

1	0	1
0	0	0
1	0	1

(a)

0	0	0
1	0	1
0	0	0

(b)

1	0	1
0	0	0
1	0	1

(c)

Gambar 2.6. Matriks citra contoh perhitungan NCC

Untuk mempermudah tiap matriks diubah menjadi bentuk vektor, yaitu matriks A (1 0 1 0 0 0 1 0 1), matriks B (0 0 0 1 0 1 0 0 0) dan matriks C (1 0 1 0 0 0 1 0

1). Matriks A diasumsikan sebagai data uji dan matriks BC sebagai data latih. Pertama-tama cari dahulu nilai rata-rata dari tiap citra dimana akan diperoleh, matriks A ($4/9 = 0,4$), matriks B ($2/9 = 0,2$) dan matriks C ($4/9 = 0,4$). Setelah didapatkan nilai rata-rata tiap citra hitung korelasi matriks A dengan matriks BC dengan menggunakan persamaan 2.5, hasil perhitungan korelasi tertinggi yang paling mendekati nilai 1 adalah nilai data latih yang paling mirip dengan data uji

2.2.8 Perhitungan Akurasi Hasil Pengenalan

Untuk menghitung akurasi keberhasilan pengenalan pada saat melakukan pengujian, digunakan persamaan berikut ini.

$$\text{Akurasi} = \frac{\text{Jumlah karakter yang dikenali dengan benar}}{\text{Jumlah karakter pada citra}} \times 100 \%$$

Untuk mendapatkan nilai akurasi pengenalan jumlah karakter yang dikenali dengan benar dibagi dengan jumlah total karakter pada pengujian, lalu hasilnya dikalikan dengan 100% sehingga dapat diperoleh persentase akurasi pengenalan.

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 Alat dan Bahan Penelitian

Untuk dapat mengerti dan mengidentifikasi kebutuhan dari sistem yang akan dibangun maka dibutuhkan analisis dan perancangan sistem terlebih dahulu. Salah satu kebutuhan sistem tersebut terdiri dari perangkat keras dan perangkat lunak yang akan digunakan untuk membangun sistem pengenalan huruf kanji.

3.1.1 Spesifikasi Perangkat Keras

Pada penelitian ini digunakan seperangkat laptop dan peralatan pendukung dengan spesifikasi sebagai berikut:

- Prosesor Intel Core i5-2450M 2nd-Gen Sandy Bridge 2.50 GHz.
- RAM 4 GB.
- Hard Disk Drive 2.5 Toshiba 500 GB.
- VGA Card Nvidia Geforce 410M 2 GB.
- Wireless Optical Mouse RAPOO.
- Printer dan Scanner HP-All in One Deskjet 1050 untuk mencetak dan memindai dokumen yang akan digunakan sebagai data uji dan data latihan pada penelitian ini.

3.1.2 Spesifikasi Perangkat Lunak

Pada penelitian ini digunakan beberapa perangkat lunak sebagai sarana penunjang penelitian, yaitu:

- Sistem operasi Microsoft Windows 7 Home Premium 64bit.

- Matlab R2015a digunakan untuk membangun sistem OCR pada penelitian ini.
- HPScan digunakan sebagai aplikasi untuk memindai dokumen.
- Wakan versi 1.67 digunakan sebagai aplikasi untuk memperoleh database arti dan variasi huruf dari huruf kanji.
- Xampp versi 1.5.5 digunakan sebagai *virtual database server*.
- Pencil versi 2.0.5 digunakan untuk merancang *mockup* antar muka.

3.2 Perancangan Sistem

Perancangan sistem yang akan dibangun pada penelitian ini akan dijelaskan melalui gambaran proses kerja sistem dan *flowchart*. Pada gambaran proses kerja sistem akan dijelaskan proses algoritma yang akan dilakukan sistem mulai dari awal hingga akhir dan untuk alur tiap proses akan dijelaskan menggunakan *flowchart*.

3.2.1 Gambaran Proses Kerja Sistem

Pada sistem yang akan dibangun terdapat proses kerja sistem yang akan dibagi menjadi 2 bagian, yaitu proses pelatihan dan proses pengujian. Proses pelatihan adalah proses di mana pengguna dapat memasukkan data berupa citra huruf kanji yang kemudian akan diproses oleh sistem dan kemudian hasilnya akan disimpan ke dalam *database* sebagai *template* yang akan digunakan sebagai data pembandingan pada proses pengujian.

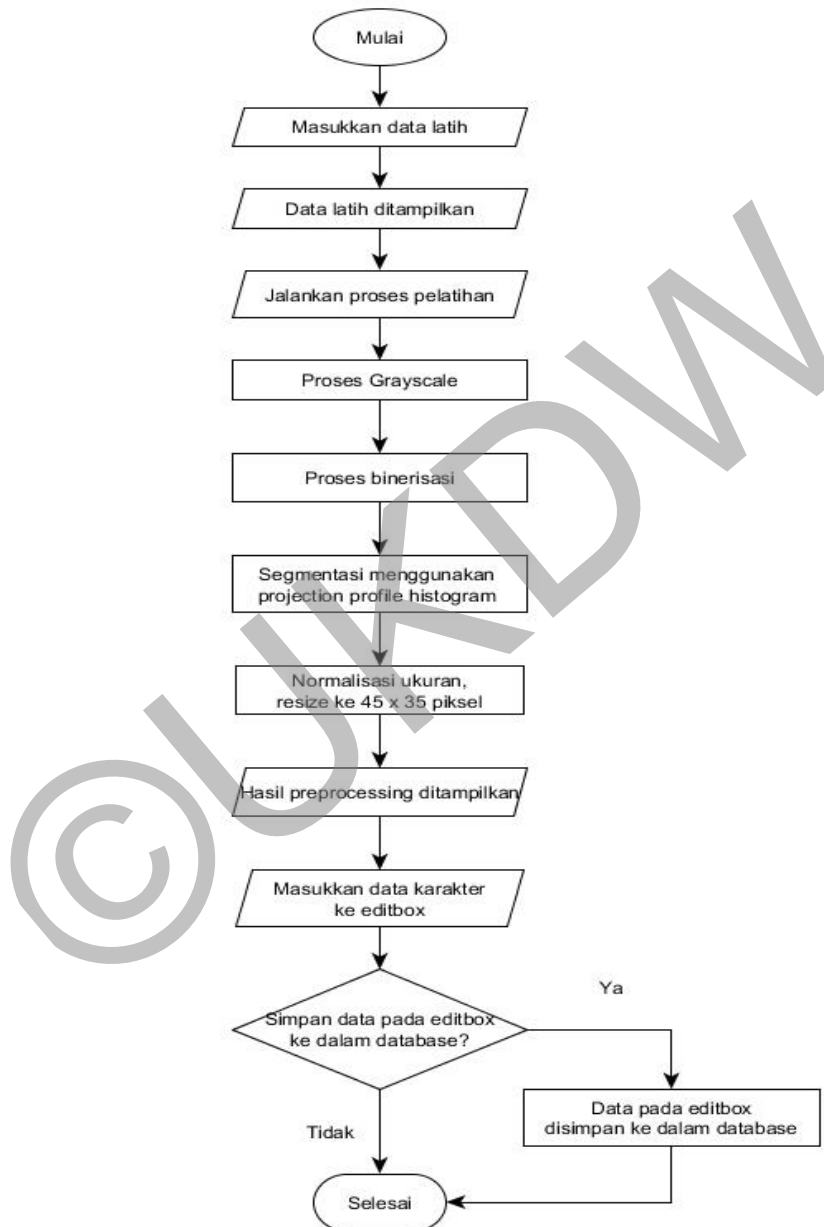
Proses pengujian merupakan proses di mana pengguna dapat memasukkan data berupa citra huruf kanji dan oleh sistem akan dibandingkan dengan seluruh *template* pada *database* menggunakan algoritma pengenalan pola *template matching*. Setelah citra masukan dan citra *template* dibandingkan maka sistem akan memutuskan *template* mana yang paling mirip dengan citra masukan.

- Algoritma Proses Pelatihan
 - Mulai
 - Data latih berupa citra huruf kanji dimasukkan.
 - Jalankan proses pelatihan.
 - Citra berwarna diubah menjadi citra *grayscale*.
 - Citra *grayscale* diubah menjadi citra biner. Nilai *threshold* yang digunakan tergantung pada nilai intensitas minimal dan maksimal piksel pada citra dan dilakukan secara otomatis oleh sistem.
 - Segmentasi citra menggunakan *projection profiled based histogram* secara vertikal dan horisontal.
 - Normalisasi citra tiap karakter hasil segmentasi dengan *resize* ukuran citra menjadi 45 x 35 piksel.
 - Masukkan data karakter pada kotak masukan yang disediakan, lakukan pada semua karakter.
 - Simpan data tiap karakter ke dalam database.
 - Selesai.

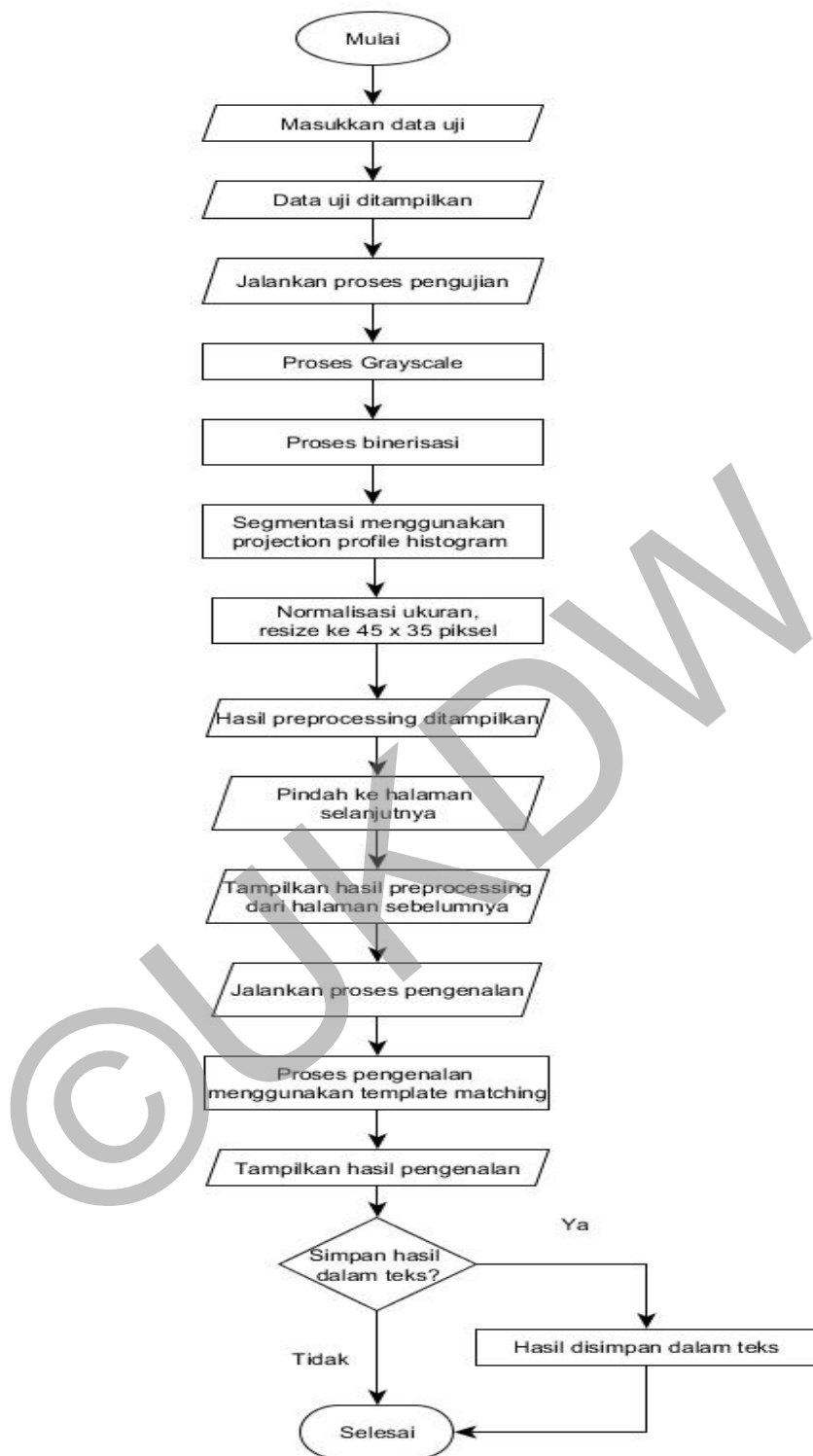
- Algoritma Proses Pengujian
 - Mulai
 - Data uji berupa citra huruf kanji dimasukkan.
 - Jalankan *preprocessing*.
 - Citra berwarna diubah menjadi citra *grayscale*.
 - Citra *grayscale* diubah menjadi citra biner. Nilai *threshold* yang digunakan tergantung pada nilai intensitas minimal dan maksimal piksel pada citra dan dilakukan secara otomatis oleh sistem.
 - Segmentasi citra menggunakan *projection profiled based histogram* secara vertikal dan horisontal.
 - Normalisasi citra tiap karakter hasil segmentasi dengan *resize* ukuran citra menjadi 45 x 35 piksel.
 - Lakukan pengenalan dengan algoritma *template matching*. Lakukan terhadap seluruh karakter pada citra masukan

- Tampilkan hasil pengenalan. Ulangi terhadap seluruh karakter pada citra masukan.
- Selesai.

3.2.2 Flowchart



Gambar 3.1 Flowchart proses pelatihan



Gambar 3.2. Flowchart proses pengujian

Pada gambar 3.1 dijelaskan alur proses sistem dari proses pelatihan menggunakan diagram alir. Pada proses ini data latih yang berupa citra RGB akan diubah menjadi citra *grayscale* lalu kemudian diubah menjadi citra biner sesuai dengan nilai *threshold* yang sudah ditentukan. Setelah citra biner diperoleh maka akan dilakukan segmentasi citra guna mendapatkan potongan tiap karakter, segmentasi dilakukan menggunakan *Projection Profile Histogram*. Pada proses segmentasi ini dilakukan dua sub-proses yaitu segmentasi baris untuk mendapatkan potongan karakter tiap baris dan segmentasi karakter untuk mendapatkan potongan tiap karakter secara individual. Setelah potongan tiap karakter didapatkan kemudian dilakukan normalisasi ukuran ke resolusi 45 x 35 piksel supaya seluruh data seragam. Proses pelatihan selesai setelah fitur biner masing – masing karakter diekstraksi dan kemudian dimasukkan ke dalam database.

Diagram alir pada gambar 3.2 menjelaskan proses pengujian di mana data uji yang berupa citra RGB akan menjalani proses yang sama dengan data latih pada proses pelatihan sampai dengan tahap normalisasi. Bedanya dengan proses pelatihan adalah setelah proses normalisasi akan dilakukan proses pengenalan karakter agar dapat menghasilkan keluaran berupa teks.

3.3 Pengumpulan Data

Pada penelitian ini dibutuhkan data-data yang nantinya akan digunakan sebagai data masukan pada proses pelatihan dan proses pengujian. Data yang digunakan untuk proses pelatihan akan disebut sebagai data latih dan pada proses pengujian akan disebut sebagai data uji. Huruf kanji yang digunakan dipilih sebanyak 100 karakter, yang merupakan 100 huruf kanji paling umum untuk digunakan sehari – hari di Jepang. Pemilihan 100 huruf tersebut berdasarkan dari artikel yang berisi 100 kanji yang paling sering digunakan pada surat kabar harian Jepang, artikel tersebut ditulis oleh Namiko Abe pada *website* “japanese – about” (Abe, Namiko; 2014; *The Top Most Frequently Used Kanji – Part 1*; <http://japanese.about.com/od/kan2/a/100kanji.htm>; diakses tanggal 15 Desember

2015). Jika dibandingkan dengan huruf pada buku – buku atau panduan pemula untuk belajar kanji, huruf kanji yang digunakan sebagian besar sama dan yang berbeda hanya beberapa karena buku – buku panduan tersebut biasanya memilih huruf yang lebih sering digunakan pada percakapan sehari – hari.

Pengumpulan data dibagi menjadi 2 jenis yaitu data karakter kanji cetak dan data karakter kanji tulisan tangan. Untuk kanji cetak dengan membuat sendiri karakter kanji menggunakan aplikasi pengolahan citra seperti Microsoft Paint dan Adobe Photosop. Jumlah *template* kanji cetak yang digunakan sebagai data latihan sebanyak 6 jenis *template* dengan total 600 karakter sedangkan untuk data uji sebanyak 6 jenis *template* dengan total 300 karakter. *Template* tersebut dapat dilihat pada tabel berikut ini.

Tabel 3.1
Contoh Template Data Latih Cetak

No	Contoh Template
1	日一 大年中
2	日一 大年中
3	日一 大年中
4	日一 大年中
5	日一 大年中
6	日一 大年中

Template nomor 1 pada table 3.1 merupakan *font* jenis DF-Kai SB, nomor 2 *font* jenis HG平成明朝体W3, nomor 3 *font* jenis HG平成角ゴシック体W3,

nomor 4 font jenis MingLiu, nomor 5 font jenis MS ゴシック dan nomor 6 font jenis MS 明朝.



Tabel 3.2
Contoh Template Data Uji Cetak

No	Contoh Template
1	九困思七山
2	生子分東三
3	生子分東三
4	九困思七山
5	九困思七山
6	生子分東三

Template nomor 1 pada tabel 3.2 merupakan font jenis Adobe Fangsong Std R, nomor 2 font jenis DotumChe, nomor 3 font jenis HGS平成明朝体W9, nomor 4 font jenis KaiTi, nomor 5 font jenis Meiryo UI dan nomor 6 font jenis SimSum. Seluruh karakter pada data latihan cetak dan data uji cetak menggunakan ukuran font 24.

Untuk karakter kanji tulisan tangan jumlah template yang digunakan sebagai data uji sebanyak 2 jenis template dengan total 200 karakter. Setiap template yang digunakan pada kanji tulisan tangan ditulis oleh sumber yang berbeda. Template tersebut dapat dilihat pada tabel berikut ini.

Tabel 3.3
Contoh Template Data Uji Tulisan Tangan

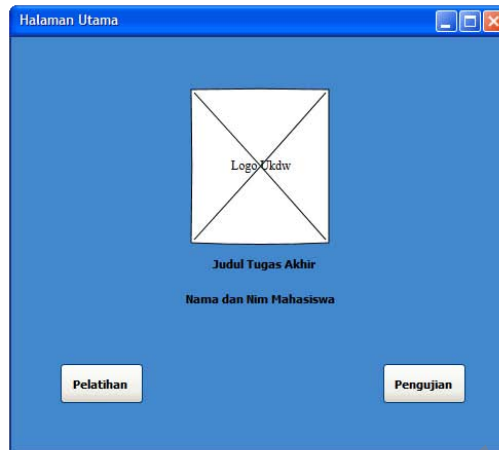
No	Contoh template
1	
2	

3.4 Perancangan Antar Muka

Pada aplikasi pengenalan huruf kanji yang akan dibangun, terdapat 4 form antar muka, yaitu form utama, form pelatihan, form pengujian dan form pengujian 2.

3.4.1 Form Halaman Utama

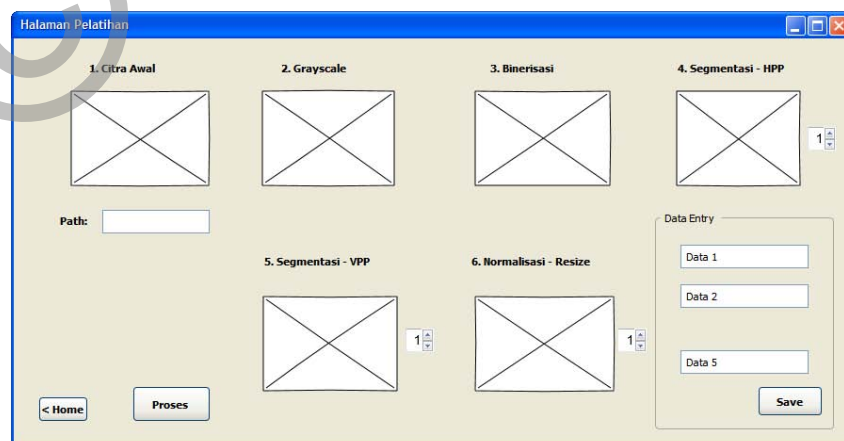
Form ini merupakan form awal dimana form ini akan ditampilkan pertama kali saat aplikasi dijalankan, form ini juga menjadi form utama karena merupakan penghubung bagi form yang lainnya. Pada form ini terdapat logo Ukdw, judul tugas akhir, nama, nim dan prodi mahasiswa, lalu pada bagian bawah terdapat 2 *button* yang akan membuka form lain dari aplikasi ini.



Gambar 3.3. Rancangan Antar Muka – Halaman Utama

3.4.2 Form Halaman Pelatihan

Form ini ditampilkan pada saat button “Pelatihan” ditekan, pada form ini pengguna dapat memasukkan data latih dan kemudian melakukan proses pelatihan terhadap data tersebut. Setelah proses pelatihan selesai pengguna dapat memasukkan keterangan dari huruf-huruf tersebut dan hasil dari pelatihan kemudian dapat disimpan ke dalam database.

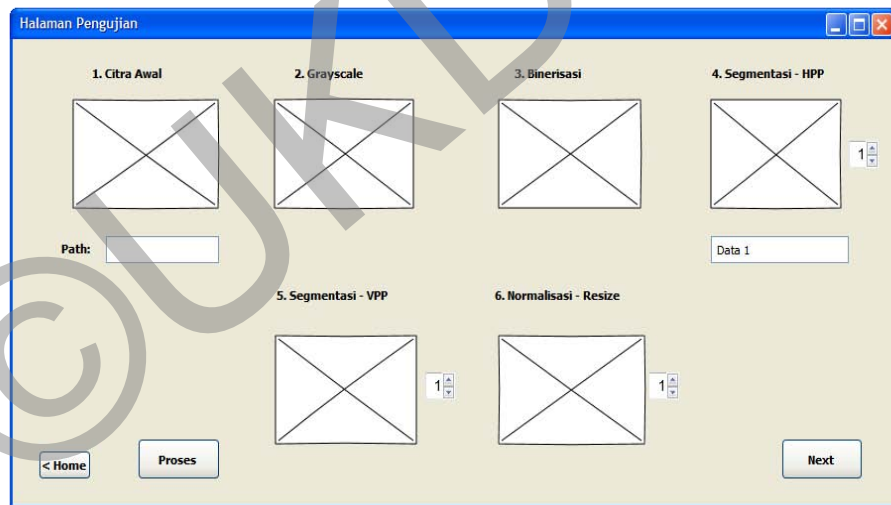


Gambar 3.4. Rancangan Antar Muka – Halaman Pelatihan

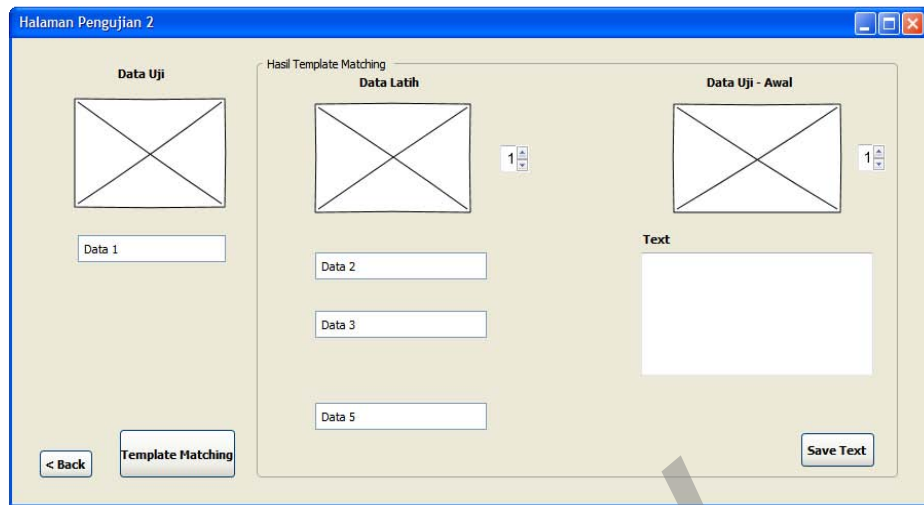
3.4.3 Form Halaman Pengujian

Pada form pengujian dapat dilakukan proses pengujian terhadap citra masukan, di mana citra uji tersebut akan dibandingkan dengan seluruh data latih pada database. Setelah proses pengujian selesai akan ditampilkan hasil pengujian, dan hasil pengujian yang paling tinggi merupakan data latih yang paling mirip dengan data uji. Hasil pengenalan kemudian juga dapat disimpan dalam format text.

Pada form pengujian dibagi menjadi dua bagian yaitu halaman pengujian pertama dan halaman pengujian kedua. Halaman pertama berguna untuk melakukan *preprocessing* data uji, sedangkan halaman kedua berguna untuk melakukan pengujian pengenalan pola terhadap data uji hasil *preprocessing* dari halaman pertama.



Gambar 3.5. Rancangan Antar Muka – Halaman Pengujian



Gambar 3.6. Rancangan Antar Muka – Halaman Pengujian 2

3.5 Kamus Data

Pada sistem yang akan dibangun pada penelitian ini hanya akan menggunakan satu buah tabel dengan struktur yang digunakan sebagai berikut:

Tabel 3.4.
Tabel ocr_kanji_45x35

Nama Field	Tipe Data	Aturan	Keterangan
index	int (11)	not_null, auto increment	index huruf, primary key
fitur_biner	longtext	not_null	fitur biner karakter kanji
karakter	varchar (255)	collation utf8_general_ci	karakter unicode kanji
arti	varchar (255)	not_null	arti bahasa indonesia dari karakter kanji
kunyomi	varchar (255)	not_null	cara baca jepang dari karakter kanji
onyomi	varchar (255)	not_null	cara baca cina dari karakter kanji

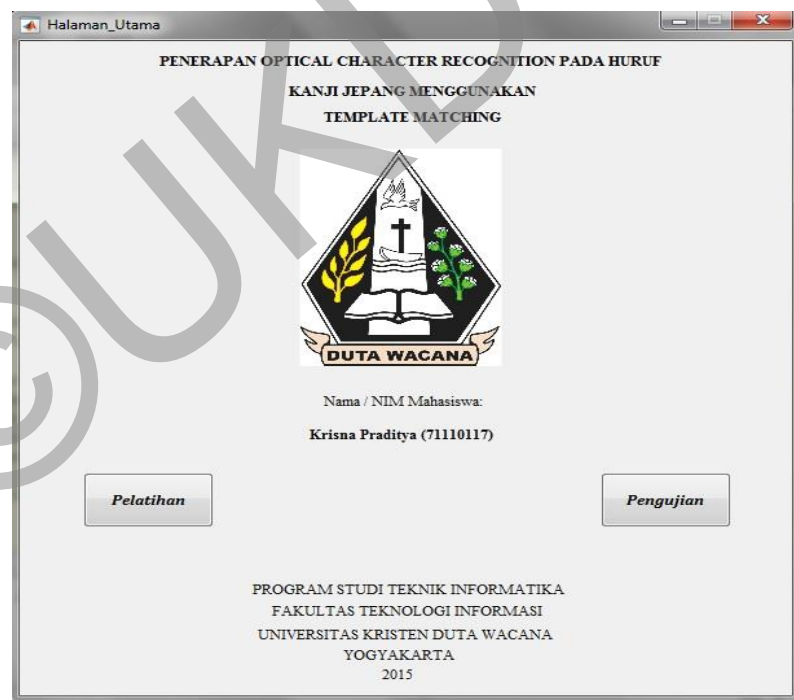
BAB 4

IMPLEMENTASI DAN ANALISIS SISTEM

4.1 Implementasi Sistem

Dalam implementasi sistem, terdapat empat buah form yang digunakan sebagai antarmuka sistem, satu form digunakan sebagai antarmuka form utama dan 3 form digunakan sebagai antarmuka form proses. Pada bagian ini akan dijelaskan mengenai empat form tersebut termasuk urutan cara penggunaannya.

4.1.1 Form Halaman Utama



Gambar 4.1. Form Halaman Utama

Halaman utama merupakan form pertama yang akan ditampilkan pada saat sistem dijalankan. Pada form ini terdapat logo Universitas Kristen Duta Wacana,

nama nim mahasiswa dan judul tugas akhir, selain itu juga terdapat dua buah *button* “Pelatihan” dan “Pengujian”. Kedua *button* tersebut akan menampilkan form lain sesuai dengan nama *button* dan sekaligus menutup form “Halaman Utama”. Tidak ada hal lain yang dapat dilakukan pada form ini selain menekan salah satu *button* tersebut.

4.1.2 Form Halaman Pelatihan

Form ini akan ditampilkan pada saat pengguna menekan *button* “Pelatihan” pada form “Halaman Utama”. Halaman ini berguna untuk melakukan serangkaian proses pelatihan terhadap data latih agar dapat digunakan sebagai template pada saat proses pengujian.

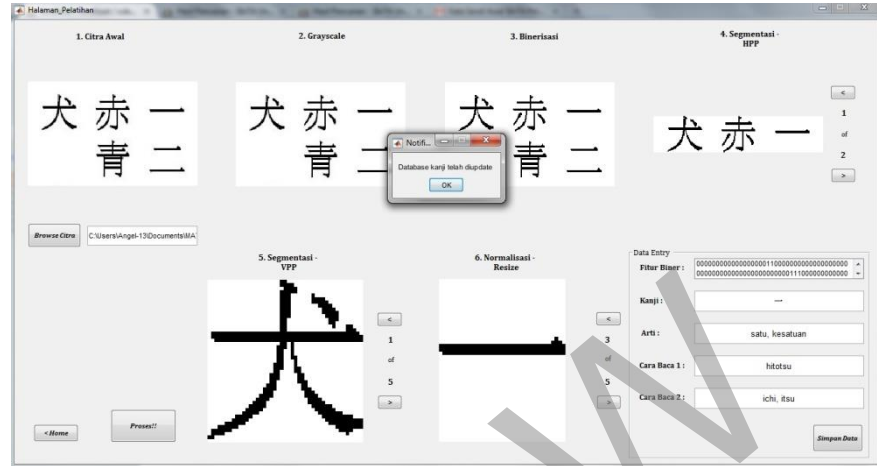


Gambar 4.2. Form Halaman Pelatihan

Untuk dapat menjalankan proses pelatihan, pengguna diharuskan memasukan data latih berupa citra RGB terlebih dahulu, dengan menekan *button* “Browse” yang kemudian akan membuka jendela untuk memilih file. Setelah data latih dimasukan, maka citra dan alamat dari file tersebut akan ditampilkan pada form.

Selanjutnya pengguna dapat menekan *button* “Proses” untuk menjalankan proses pelatihan, jika sebelumnya pengguna belum memasukan data latih maka

akan ditampilkan peringatan. Setelah proses pelatihan selesai maka hasil masing – masing *preprocessing* akan ditampilkan, selain itu fitur biner dari tiap karakter juga akan ditampilkan.



Gambar 4.3. Form Halaman Pelatihan – Input Data Berhasil

Setelah *preprocessing* seluruh karakter selesai pengguna dapat memasukkan data berupa arti, cara baca dan teks unicode dari tiap karakter untuk kemudian dimasukkan ke dalam database dengan menekan button “Simpan”. Jika seluruh proses sebelumnya belum dilakukan termasuk pengisian data maka akan ditampilkan peringatan, sedangkan jika proses memasukkan data ke dalam database berhasil maka akan ditampilkan notifikasi..

4.1.3 Form Halaman Pengujian 1 - Preprocessing

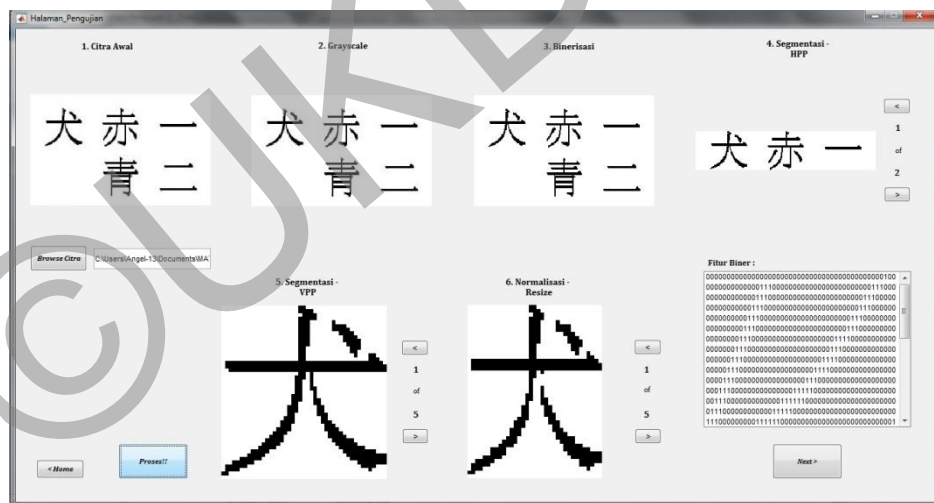
Form ini akan ditampilkan pada saat pengguna menekan button “Pengujian” pada form “Halaman Utama” dan pada halaman ini pengguna dapat melakukan pengujian pengenalan pola terhadap data uji. Form pengujian dibagi menjadi dua halaman yaitu ‘Halaman Pengujian’ dan “Halaman Pengujian 2”. “Halaman Pengujian 2” akan dijelaskan pada bagian 4.1.4.

Untuk dapat menjalankan proses pengujian, pengguna diharuskan memasukkan data uji berupa citra RGB terlebih dahulu, dengan menekan *button*

“Browse” yang kemudian akan membuka jendela untuk memilih file. Setelah data uji dimasukkan, maka citra dan alamat dari file tersebut akan ditampilkan pada form.



Gambar 4.4. Form Halaman Pengujian 1



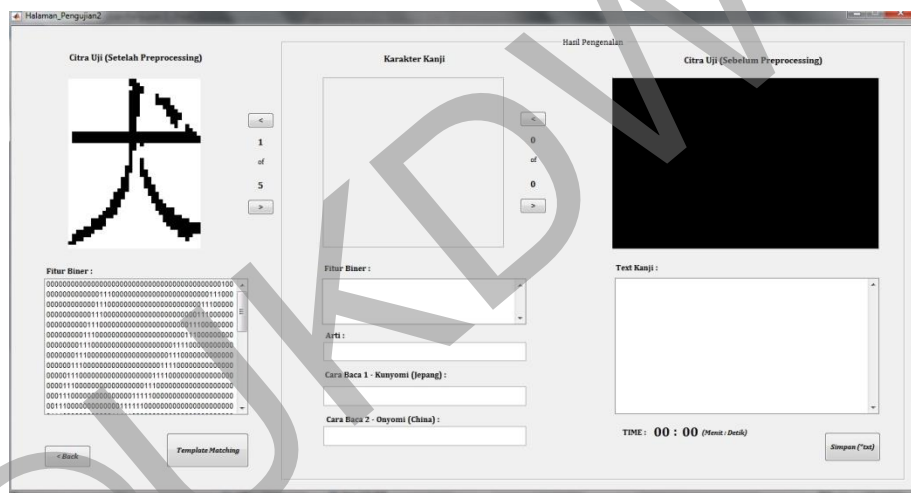
Gambar 4.5. Form Halaman Pengujian 1 – Setelah *Preprocessing*

Selanjutnya pengguna dapat menekan *button* “Proses” untuk menjalankan *preprocessing* citra, jika sebelumnya pengguna belum memasukkan data latih maka akan ditampilkan peringatan. Setelah proses selesai maka hasil masing – masing *preprocessing* akan ditampilkan, selain itu fitur biner dari tiap karakter

juga akan ditampilkan. Selanjutnya pengguna dapat menekan button “Next” untuk menampilkan form “Halaman Pengujian 2” dan melakukan proses pengujian.

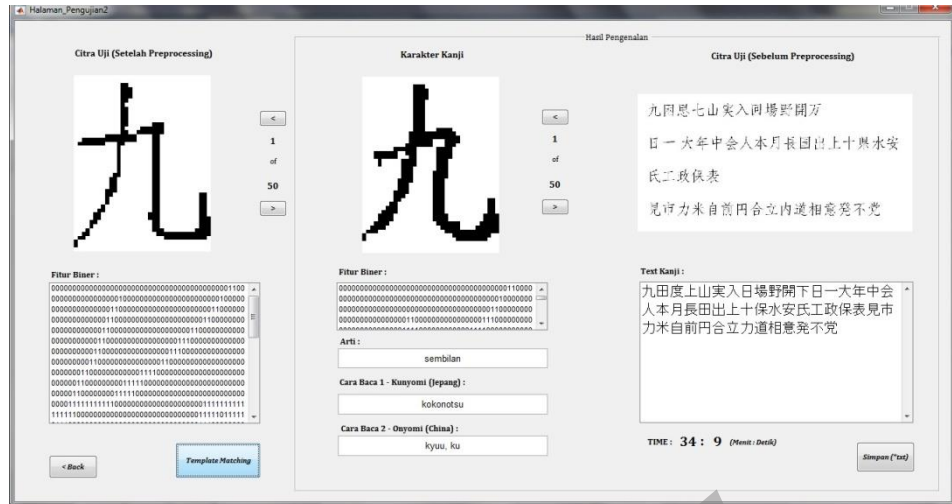
4.1.4 Form Halaman Pengujian 2 – Pengenalan Karakter

Form ini akan ditampilkan pada saat pengguna menekan *button* “Next” pada “Halaman Pengujian”, pada halaman ini pengguna dapat melakukan pengujian dengan melakukan proses pengenalan menggunakan algoritma *template matching* terhadap data uji yang sebelumnya sudah dilakukan proses *preprocessing* terlebih dahulu.



Gambar 4.6. Form Halaman Pengujian 2

Untuk melakukan pengujian pengenalan pengguna dapat menekan *button* “Template Matching” dan sistem akan menghitung nilai korelasi data uji dengan seluruh data latih yang berada di basis data. Setelah perhitungan selesai maka sistem akan mengambil nilai korelasi terbesar untuk setiap data uji, dan menampilkan hasil pengenalan tersebut berdasarkan indeks dari data latih pada basis data. Pengguna juga dapat menyimpan teks kanji hasil pengenalan ke dalam file dengan ekstensi *.txt* dengan menekan *button* “Simpan”, sistem akan menampilkan jendela untuk pengguna memilih nama dan *folder* tempat menyimpan file. Pada form juga ditampilkan waktu lamanya proses pengenalan.



Gambar 4.7. Form Halaman Pengujian 2 – Proses Pengenalan Selesai

4.2 Analisis dan Hasil Pengujian

4.2.1 Data Pengujian

Seperti yang sudah dibahas pada bagian 3.3 sub-bab “Pengumpulan Data” terdapat 2 jenis data uji yaitu data uji cetak dan data uji tulisan tangan. Jumlah data uji cetak sebanyak 300 karakter yang terbagi dalam 6 jenis *template* dan data uji tulisan tangan sebanyak 200 karakter yang terbagi dalam 2 jenis *template*.

4.2.2 Hasil Pengujian

Pengujian pertama dilakukan kepada data uji huruf kanji cetak, dilakukan enam kali pengujian sesuai dengan jenis *template* pada data uji.

Tabel 4.1

Hasil Pengujian Data Uji Cetak Adobe Fangsong Std R

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
1	九	九	Benar
2	困	田	Salah

Tabel 4.1 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
3	思	度	Salah
4	七	上	Salah
5	山	山	Benar
6	実	実	Benar
7	入	入	Benar
8	回	日	Salah
9	場	場	Benar
10	野	野	Benar
11	開	開	Benar
12	万	下	Salah
13	日	日	Benar
14	一	一	Benar
15	大	大	Benar
16	年	年	Benar
17	中	中	Benar
18	会	会	Benar
19	人	人	Benar
20	本	本	Benar
21	月	月	Benar
22	長	長	Benar
23	国	田	Salah
24	出	出	Benar
25	上	上	Benar
26	十	十	Benar

Tabel 4.1 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
27	県	保	Salah
28	水	水	Benar
29	安	安	Benar
30	氏	氏	Benar
31	工	工	Benar
32	政	政	Benar
33	保	保	Benar
34	表	表	Benar
35	見	見	Benar
36	市	市	Benar
37	力	力	Benar
38	米	米	Benar
39	自	自	Benar
40	前	前	Benar
41	円	円	Benar
42	合	合	Benar
43	立	立	Benar
44	内	力	Salah
45	道	道	Benar
46	相	相	Benar
47	意	意	Benar
48	発	発	Benar
49	不	不	Benar
50	党	党	Benar

Tabel 4.1 (Sambungan)

Jumlah Karakter Salah	8
Jumlah Karakter Benar	42
Persentase Akurasi Pengenalan	84%

Tabel 4.1 merupakan tabel hasil pengujian data uji karakter cetak dengan jenis *font* Adobe Fangsong Std R. Persentase hasil akurasi pengenalan adalah 84%, dimana terdapat 8 karakter yang dikenali secara salah dan 42 karakter dikenali dengan benar. 8 karakter yang salah dikenali adalah kanji 困, 思, 七, 回, 万, 国, 県, 内.

Tabel 4.2
Hasil Pengujian Data Uji Cetak DotumChe

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
1	生	生	Benar
2	子	子	Benar
3	分	分	Benar
4	東	東	Benar
5	三	三	Benar
6	行	行	Benar
7	同	同	Benar
8	今	今	Benar
9	高	高	Benar
10	金	金	Benar
11	時	時	Benar
12	手	手	Benar
13	新	新	Benar
14	明	明	Benar

Tabel 4.2 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
15	方	方	Benar
16	部	部	Benar
17	女	女	Benar
18	乙	乙	Benar
19	企	企	Benar
20	四	四	Benar
21	民	民	Benar
22	对	对	Benar
23	主	生	Salah
24	正	正	Benar
25	代	代	Benar
26	言	言	Benar
27	二	二	Benar
28	事	事	Benar
29	社	社	Benar
30	者	者	Benar
31	地	地	Benar
32	京	京	Benar
33	間	間	Benar
34	田	円	Salah
35	全	全	Benar
36	定	定	Benar
37	家	家	Benar
38	了	了	Benar

Tabel 4.2 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
39	六	六	Benar
40	問	問	Benar
41	話	話	Benar
42	文	文	Benar
43	動	動	Benar
44	度	度	Benar
45	体	体	Benar
46	学	学	Benar
47	下	下	Benar
48	目	目	Benar
49	五	五	Benar
50	後	後	Benar
Jumlah Karakter Salah			2
Jumlah Karakter Benar			48
Persentase Akurasi Pengenalan			96%

Tabel 4.2 merupakan tabel hasil pengujian data uji karakter cetak dengan jenis *font* DotumChe. Persentase hasil akurasi pengenalan adalah 96%, dimana terdapat 2 karakter yang dikenali secara salah dan 48 karakter dikenali dengan benar. 2 karakter yang salah dikenali adalah 主 salah dikenali sebagai 生 dan 田 salah dikenali sebagai 円.

Tabel 4.3
Hasil Pengujian Data Uji Cetak HGS平成明朝体W9

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
1	生	生	Benar

Tabel 4.3 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
2	子	子	Benar
3	分	分	Benar
4	東	東	Benar
5	三	三	Benar
6	行	行	Benar
7	同	同	Benar
8	今	今	Benar
9	高	高	Benar
10	金	金	Benar
11	時	時	Benar
12	手	手	Benar
13	新	新	Benar
14	明	明	Benar
15	方	方	Benar
16	部	部	Benar
17	女	女	Benar
18	乙	乙	Benar
19	企	企	Benar
20	四	四	Benar
21	民	民	Benar
22	对	对	Benar
23	主	主	Benar
24	正	正	Benar
25	代	代	Benar

Tabel 4.3 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
26	言	言	Benar
27	二	二	Benar
28	事	事	Benar
29	社	社	Benar
30	者	者	Benar
31	地	地	Benar
32	京	京	Benar
33	間	間	Benar
34	田	田	Benar
35	全	全	Benar
36	定	定	Benar
37	家	家	Benar
38	了	了	Benar
39	六	六	Benar
40	問	問	Benar
41	話	話	Benar
42	文	文	Benar
43	動	動	Benar
44	度	度	Benar
45	体	体	Benar
46	学	学	Benar
47	下	下	Benar
48	目	目	Benar
49	五	五	Benar

Tabel 4.3 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
50	後	後	Benar
Jumlah Karakter Salah			0
Jumlah Karakter Benar			50
Persentase Akurasi Pengenalan			100%

Tabel 4.3 merupakan tabel hasil pengujian data uji karakter cetak dengan jenis *font* HGS平成明朝体W9. Persentase hasil akurasi pengenalan adalah 100%, dimana seluruh karakter pada data uji dapat dikenali dengan benar. Dari keseluruhan pengujian pada data uji cetak, cetak pengujian ini merupakan pengujian dengan persentase akurasi pengenalan tertinggi.

Tabel 4.4
Hasil Pengujian Data Uji Cetak KaiTi

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
1	九	九	Benar
2	困	困	Benar
3	思	思	Benar
4	七	七	Benar
5	山	山	Benar
6	実	実	Benar
7	入	入	Benar
8	回	回	Benar
9	場	場	Benar
10	野	野	Benar
11	開	開	Benar
12	万	万	Benar

Tabel 4.4 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
13	日	日	Benar
14	一	一	Benar
15	大	大	Benar
16	年	子	Salah
17	中	中	Benar
18	会	会	Benar
19	人	人	Benar
20	本	本	Benar
21	月	月	Benar
22	長	長	Benar
23	国	国	Benar
24	出	出	Benar
25	上	正	Salah
26	十	十	Benar
27	県	中	Salah
28	水	本	Salah
29	安	安	Benar
30	氏	氏	Benar
31	工	工	Benar
32	政	政	Benar
33	保	保	Benar
34	表	表	Benar
35	見	九	Salah
36	市	市	Benar

Tabel 4.4 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
37	力	力	Benar
38	米	十	Salah
39	自	自	Benar
40	前	前	Benar
41	円	円	Benar
42	合	発	Salah
43	立	立	Benar
44	内	内	Benar
45	道	道	Benar
46	相	相	Benar
47	意	意	Benar
48	発	発	Benar
49	不	不	Benar
50	党	党	Benar
Jumlah Karakter Salah			7
Jumlah Karakter Benar			43
Persentase Akurasi Pengenalan			86%

Tabel 4.4 merupakan tabel hasil pengujian data uji karakter cetak dengan jenis *font* KaiTi. Persentase hasil akurasi pengenalan adalah 86%, dimana terdapat 7 karakter yang dikenali secara salah dan 43 karakter dikenali dengan benar. 7 karakter yang salah dikenali adalah kanji年, 上, 県, 水, 見, 米 dan 合.

Tabel 4.5
 Hasil Pengujian Data Uji Cetak Meiryo UI

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
1	九	九	Benar
2	困	困	Benar
3	思	思	Benar
4	七	七	Benar
5	山	山	Benar
6	実	実	Benar
7	入	入	Benar
8	回	回	Benar
9	場	場	Benar
10	野	野	Benar
11	開	開	Benar
12	万	万	Benar
13	日	円	Salah
14	一	十	Salah
15	大	大	Benar
16	年	年	Benar
17	中	中	Benar
18	会	会	Benar
19	人	人	Benar
20	本	本	Benar
21	月	月	Benar
22	長	長	Benar
23	国	国	Benar
24	出	出	Benar

Tabel 4.5 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
25	上	上	Benar
26	十	十	Benar
27	県	県	Benar
28	水	水	Benar
29	安	安	Benar
30	氏	氏	Benar
31	工	主	Salah
32	政	政	Benar
33	保	保	Benar
34	表	表	Benar
35	見	見	Benar
36	市	市	Benar
37	力	力	Benar
38	米	米	Benar
39	自	自	Benar
40	前	前	Benar
41	円	円	Benar
42	合	合	Benar
43	立	立	Benar
44	内	内	Benar
45	道	道	Benar
46	相	相	Benar
47	意	意	Benar
48	発	発	Benar

Tabel 4.5 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
49	不	不	Benar
50	党	党	Benar
Jumlah Karakter Salah			3
Jumlah Karakter Benar			47
Persentase Akurasi Pengenalan			94%

Tabel 4.5 merupakan tabel hasil pengujian data uji karakter cetak dengan jenis *font* Meiyo UI. Persentase hasil akurasi pengenalan adalah 94%, dimana terdapat 3 karakter yang dikenali secara salah dan 47 karakter dikenali dengan benar. 3 karakter yang salah dikenali adalah 日 salah dikenali sebagai 円, 一 salah dikenali sebagai 十 dan 工 salah dikenali sebagai 主.

Tabel 4.6
Hasil Pengujian Data Uji Cetak SimSun

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
1	生	生	Benar
2	子	子	Benar
3	分	分	Benar
4	東	東	Benar
5	三	三	Benar
6	行	行	Benar
7	同	同	Benar
8	今	金	Salah
9	高	高	Benar
10	金	全	Salah
11	時	時	Benar

Tabel 4.6 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
12	手	手	Benar
13	新	新	Benar
14	明	明	Benar
15	方	方	Benar
16	部	部	Benar
17	女	女	Benar
18	乙	乙	Benar
19	企	企	Benar
20	四	四	Benar
21	民	民	Benar
22	对	对	Benar
23	主	生	Salah
24	正	正	Benar
25	代	代	Benar
26	言	言	Benar
27	二	二	Benar
28	事	事	Benar
29	社	社	Benar
30	者	者	Benar
31	地	地	Benar
32	京	京	Benar
33	間	間	Benar
34	田	田	Benar
35	全	全	Benar

Tabel 4.6 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
36	定	定	Benar
37	家	家	Benar
38	了	了	Benar
39	六	六	Benar
40	問	問	Benar
41	話	話	Benar
42	文	文	Benar
43	動	動	Benar
44	度	度	Benar
45	体	体	Benar
46	学	学	Benar
47	下	下	Benar
48	目	目	Benar
49	五	五	Benar
50	後	後	Benar
Jumlah Karakter Salah			3
Jumlah Karakter Benar			47
Persentase Akurasi Pengenalan			94%

Tabel 4.6 merupakan tabel hasil pengujian data uji karakter cetak dengan jenis *font* Simsun. Persentase hasil akurasi pengenalan adalah 94%, dimana terdapat 3 karakter yang dikenali secara salah dan 47 karakter dikenali dengan benar. 3 karakter yang salah dikenali adalah 今 salah dikenali sebagai 金, 金 salah dikenali sebagai 全 dan 主 salah dikenali sebagai 生.

Pengujian selanjutnya dilakukan kepada data uji huruf kanji tulisan tangan, dilakukan dua kali pengujian sesuai dengan jenis *template* pada data uji.

Tabel 4.7
 Hasil Pengujian Data Uji Tulisan Tangan Template 1

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
1	日	日	Benar
2	一	一	Benar
3	大	一	Salah
4	年	三	Salah
5	中	下	Salah
6	会	目	Salah
7	人	人	Benar
8	本	市	Salah
9	月	市	Salah
10	長	合	Salah
11	国	国	Benar
12	出	出	Benar
13	上	正	Salah
14	十	生	Salah
15	生	工	Salah
16	子	子	Benar
17	分	一	Salah
18	東	東	Benar
19	三	五	Salah
20	行	市	Salah
21	同	円	Salah
22	今	社	Salah
23	高	月	Salah
24	金	一	Salah

Tabel 4.7 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
25	時	時	Benar
26	手	一	Salah
27	見	目	Salah
28	市	十	Salah
29	力	力	Benar
30	米	大	Salah
31	自	山	Salah
32	前	問	Salah
33	円	田	Salah
34	合	一	Salah
35	立	上	Salah
36	内	円	Salah
37	二	三	Salah
38	事	学	Salah
39	社	女	Salah
40	者	合	Salah
41	地	地	Benar
42	京	家	Salah
43	間	同	Salah
44	田	山	Salah
45	体	体	Benar
46	学	安	Salah
47	下	下	Benar
48	目	明	Salah

Tabel 4.7 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
49	五	三	Salah
50	後	行	Salah
51	新	大	Salah
52	明	円	Salah
53	方	万	Benar
54	部	開	Salah
55	女	万	Salah
56	乙	乙	Benar
57	企	正	Salah
58	四	自	Salah
59	民	見	Salah
60	対	新	Salah
61	主	十	Salah
62	正	十	Salah
63	代	社	Salah
64	言	言	Benar
65	九	分	Salah
66	困	十	Salah
67	思	一	Salah
68	七	政	Salah
69	山	山	Benar
70	実	実	Benar
71	入	本	Salah
72	回	回	Benar

Tabel 4.7 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
73	場	見	Salah
74	野	野	Benar
75	開	国	Salah
76	万	一	Salah
77	全	工	Salah
78	定	下	Salah
79	家	安	Salah
80	了	了	Benar
81	六	下	Salah
82	間	五	Salah
83	話	七	Salah
84	文	文	Benar
85	動	回	Salah
86	度	学	Salah
87	県	今	Salah
88	水	分	Salah
89	安	文	Salah
90	氏	代	Salah
91	工	工	Benar
92	政	政	Benar
93	保	氏	Salah
94	表	十	Salah
95	道	金	Salah
96	相	四	Salah

Tabel 4.7 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
97	意	安	Salah
98	発	合	Salah
99	不	工	Salah
100	党	安	Salah
Jumlah Karakter Salah			77
Jumlah Karakter Benar			23
Persentase Akurasi Pengenalan			23%

Tabel 4.7 merupakan tabel hasil pengujian data uji karakter tulisan tangan dengan *template* nomor 1. Persentase hasil akurasi pengenalan adalah 23%, dimana terdapat 77 karakter yang dikenali secara salah dan 23 karakter dikenali dengan benar.

Tabel 4.8
Hasil Pengujian Data Uji Tulisan Tangan Template 2

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
1	日	日	Benar
2	一	一	Benar
3	大	十	Salah
4	年	生	Salah
5	中	中	Benar
6	国	中	Salah
7	出	十	Salah
8	上	十	Salah
9	十	十	Benar
10	生	生	Benar
11	同	円	Salah

Tabel 4.8 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
12	今	十	Salah
13	高	一	Salah
14	金	金	Benar
15	時	時	Benar
16	自	日	Salah
17	前	乙	Salah
18	円	一	Salah
19	合	二	Salah
20	立	立	Benar
21	地	度	Salah
22	京	今	Salah
23	間	問	Salah
24	田	四	Salah
25	体	代	Salah
26	会	会	Benar
27	人	人	Benar
28	本	六	Salah
29	月	氏	Salah
30	長	長	Benar
31	子	子	Benar
32	分	万	Salah
33	東	道	Salah
34	三	年	Salah
35	行	代	Salah

Tabel 4.8 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
36	手	手	Benar
37	見	見	Benar
38	市	中	Salah
39	力	力	Benar
40	米	六	Salah
41	内	内	Benar
42	二	二	Benar
43	事	学	Salah
44	社	社	Benar
45	者	者	Benar
46	学	学	Benar
47	下	下	Benar
48	目	目	Benar
49	五	三	Salah
50	後	代	Salah
51	新	六	Salah
52	明	同	Salah
53	方	方	Benar
54	部	金	Salah
55	女	六	Salah
56	主	主	Benar
57	正	正	Benar
58	代	十	Salah
59	言	言	Benar

Tabel 4.8 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
60	九	九	Benar
61	入	入	Benar
62	回	回	Benar
63	場	保	Salah
64	野	野	Benar
65	開	開	Benar
66	六	六	Benar
67	問	問	Benar
68	話	話	Benar
69	文	文	Benar
70	動	動	Benar
71	工	困	Salah
72	政	代	Salah
73	保	保	Benar
74	表	三	Salah
75	道	道	Benar
76	乙	大	Salah
77	企	企	Benar
78	四	四	Benar
79	民	民	Benar
80	対	対	Benar
81	困	四	Salah
82	思	思	Benar
83	七	七	Benar

Tabel 4.8 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai	Hasil Pengenalan
84	山	山	Benar
85	実	実	Benar
86	万	万	Benar
87	全	全	Benar
88	定	家	Salah
89	家	下	Salah
90	了	下	Salah
91	度	道	Salah
92	県	党	Salah
93	水	本	Salah
94	安	安	Benar
95	氏	氏	Benar
96	相	相	Benar
97	意	意	Benar
98	発	発	Benar
99	不	不	Benar
100	党	実	Salah
Jumlah Karakter Salah			46
Jumlah Karakter Benar			54
Persentase Akurasi Pengenalan			54%

Tabel 4.8 merupakan tabel hasil pengujian data uji karakter tulisan tangan dengan *template* nomor 2. Persentase hasil akurasi pengenalan adalah 54%, dimana terdapat 46 karakter yang dikenali secara salah dan 54 karakter dikenali dengan benar.

4.2.3 Analisis Hasil Pengujian

Berdasarkan hasil pengujian pada bagian 4.2.2 terhadap data uji huruf kanji cetak dan kanji tulisan tangan dapat dirangkum sebagai berikut.

Tabel 4.9

Persentase Hasil Pengenalan Pengujian Kanji Cetak Karakter Set 1

No	Jenis Template Data Uji	Persentase Pengenalan
1	Adobe Fangsong Std R	84%
2	KaiTi	86%
3	Meiryo UI	94%
Rata – rata persentase pengenalan		88%

Tabel 4.10

Persentase Hasil Pengenalan Pengujian Kanji Cetak Karakter Set 2

No	Jenis Template Data Uji	Persentase Pengenalan
1	DotumChe	96%
2	HGS平成明朝体W9	100%
3	SimSum	94%
Rata – rata persentase pengenalan		96,67%

Tabel 4.11

Persentase Hasil Pengenalan Data Uji Kanji Tulisan Tangan

No	Jenis Template Data Uji	Persentase Pengenalan
1	Data Uji Tulisan Tangan 1	23%
2	Data Uji Tulisan Tangan 2	54%
Rata – rata persentase pengenalan		38,5%

Secara keseluruhan pengujian terhadap data uji kanji cetak diperoleh hasil pengujian yang cukup baik dengan rata – rata akurasi pengenalan sebesar 88% untuk pengujian pada karakter kanji set 1 dan sebesar 96,67% pada karakter kanji

set 2. Data uji HGS平成明朝体W9 mendapat hasil pengenalan sempurna yaitu 100% dikarenakan memiliki garis *font* yang paling tebal dibanding data uji lain, sedangkan data uji Adobe Fangsong Std R dan KaiTi mendapat hasil pengenalan lebih rendah dibanding Meiryu UI dikarenakan memiliki tipe garis *font* yang tidak lurus atau simetris. Kedua data uji tersebut memiliki tipe garis *font* mirip seperti goresan kuas.

Untuk pengujian terhadap data uji kanji tulisan tangan diperoleh rata – rata hasil pengujian yang lebih rendah yaitu 38,5%. Hal ini wajar karena data uji tulisan tangan memiliki variasi penulisan karakter yang jauh lebih banyak daripada data uji cetak. Data uji tulisan tangan nomor 2 mendapat hasil pengujian yang lebih tinggi dari data uji nomor 1 dikarenakan memiliki goresan karakter yang lebih tebal.

Tabel 4.12

Hasil Pengujian Data Uji kanji Cetak Berdasarkan Karakter Set 1

No	Huruf Kanji	Dikenali Sebagai			% Pengenalan
		1	2	3	
1	九	九	九	九	100%
2	困	田	困	困	66%
3	思	度	思	思	66%
4	七	上	七	七	66%
5	山	山	山	山	100%
6	実	実	実	実	100%
7	入	入	入	入	100%
8	回	日	回	回	66%
9	場	場	場	場	100%
10	野	野	野	野	100%
11	開	開	開	開	100%

Tabel 4.12 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai			% Pengenalan
		1	2	3	
12	万	下	万	万	66%
13	日	日	日	円	66%
14	一	一	一	十	66%
15	大	大	大	大	100%
16	年	年	子	年	66%
17	中	中	中	中	100%
18	会	会	会	会	100%
19	人	人	人	人	100%
20	本	本	本	本	100%
21	月	月	月	月	100%
22	長	長	長	長	100%
23	国	田	国	国	66%
24	出	出	出	出	100%
25	上	上	正	上	66%
26	十	十	十	十	100%
27	県	保	中	県	33%
28	水	水	本	水	66%
29	安	安	安	安	100%
30	氏	氏	氏	氏	100%
31	工	工	工	主	66%
32	政	政	政	政	100%
33	保	保	保	保	100%

Tabel 4.12 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai			% Pengenalan
		1	2	3	
34	表	表	表	表	100%
35	見	見	九	見	66%
36	市	市	市	市	100%
37	力	力	力	力	100%
38	米	米	十	米	66%
39	自	自	自	自	100%
40	前	前	前	前	100%
41	円	円	円	円	100%
42	合	合	発	合	66%
43	立	立	立	立	100%
44	内	力	内	内	66%
45	道	道	道	道	100%
46	相	相	相	相	100%
47	意	意	意	意	100%
48	発	発	発	発	100%
49	不	不	不	不	100%
50	党	党	党	党	100%

Tabel 4.12 merupakan tabel karakter hasil pengujian terhadap data uji kanji cetak. Berdasarkan tabel karakter yang memiliki persentase pengenalan terendah hanya 33% adalah kanji 県. Pada data uji kanji cetak kesalahan pengenalan terletak pada kemiripan bentuk beberapa karakter, seperti—salah dikenali sebagai 十 atau 水 salah dikenali sebagai 本.

Tabel 4.13

Hasil Pengujian Data Uji kanji Cetak Berdasarkan Karakter Set 2

No	Huruf Kanji	Dikenali Sebagai			% Pengenalan
		1	2	3	
1	生	生	生	生	100%
2	子	子	子	子	100%
3	分	分	分	分	100%
4	東	東	東	東	100%
5	三	三	三	三	100%
6	行	行	行	行	100%
7	同	同	同	同	100%
8	今	今	今	金	66%
9	高	高	高	高	100%
10	金	金	金	全	66%
11	時	時	時	時	100%
12	手	手	手	手	100%
13	新	新	新	新	100%
14	明	明	明	明	100%
15	方	方	方	方	100%
16	部	部	部	部	100%
17	女	女	女	女	100%
18	乙	乙	乙	乙	100%
19	企	企	企	企	100%
20	四	四	四	四	100%
21	民	民	民	民	100%
22	対	対	対	対	100%
23	主	生	主	生	33%

Tabel 4.13 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai			% Pengenalan
		1	2	3	
24	正	正	正	正	100%
25	代	代	代	代	100%
26	言	言	言	言	100%
27	二	二	二	二	100%
28	事	事	事	事	100%
29	社	社	社	社	100%
30	者	者	者	者	100%
31	地	地	地	地	100%
32	京	京	京	京	100%
33	間	間	間	間	100%
34	田	田	田	田	66%
35	全	全	全	全	100%
36	定	定	定	定	100%
37	家	家	家	家	100%
38	了	了	了	了	100%
39	六	六	六	六	100%
40	問	問	問	問	100%
41	話	話	話	話	100%
42	文	文	文	文	100%
43	動	動	動	動	100%
44	度	度	度	度	100%
45	体	体	体	体	100%
46	学	学	学	学	100%
47	下	下	下	下	100%

Tabel 4.13 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai			% Pengenalan
		1	2	3	
48	目	目	目	目	100%
49	五	五	五	五	100%
50	後	後	後	後	100%

Tabel 4.13 merupakan tabel karakter hasil pengujian terhadap data uji kanji cetak. Berdasarkan tabel karakter yang memiliki persentase pengenalan terendah hanya 33% adalah kanji主. Pada data uji kanji cetak kesalahan pengenalan terletak pada kemiripan bentuk beberapa karakter, seperti金 salah dikenali sebagai全 atau主 salah dikenali sebagai生.

Tabel 4.14

Hasil Pengujian Data Uji kanji Tulisan Tangan Berdasarkan Karakter

No	Huruf Kanji	Dikenali Sebagai		% Pengenalan
		1	2	
1	日	日	日	100%
2	一	一	一	100%
3	大	一	十	0%
4	年	三	生	0%
5	中	下	中	50%
6	会	目	会	50%
7	人	人	人	100%
8	本	市	六	0%
9	月	市	氏	0%
10	長	合	長	50%

Tabel 4.14 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai		% Pengenalan
		1	2	
11	国	国	中	50%
12	出	出	十	50%
13	上	正	十	0%
14	十	生	十	50%
15	生	工	生	50%
16	子	子	子	100%
17	分	一	万	0%
18	東	東	道	50%
19	三	五	年	0%
20	行	市	代	0%
21	同	円	円	0%
22	今	社	十	0%
23	高	月	一	0%
24	金	一	金	50%
25	時	時	時	100%
26	手	一	手	50%
27	見	目	見	50%
28	市	十	中	0%
29	力	力	力	100%
30	米	大	六	0%
31	自	山	日	0%
32	前	問	乙	0%
33	円	田	一	0%

Tabel 4.14 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai		% Pengenalan
		1	2	
34	合	一	二	0%
35	立	上	立	50%
36	内	円	内	50%
37	二	三	二	50%
38	事	学	学	0%
39	社	女	社	50%
40	者	合	者	50%
41	地	地	度	50%
42	京	家	今	0%
43	間	同	間	0%
44	田	山	四	0%
45	体	体	代	50%
46	学	安	学	50%
47	下	下	下	100%
48	目	明	目	50%
49	五	三	三	0%
50	後	行	代	0%
51	新	大	六	0%
52	明	円	同	0%
53	方	万	方	50%
54	部	開	金	0%
55	女	万	六	0%
56	乙	乙	大	50%

Tabel 4.14 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai		% Pengenalan
		1	2	
57	企	正	企	50%
58	四	自	四	50%
59	民	見	民	50%
60	对	新	对	50%
61	主	十	主	50%
62	正	十	正	50%
63	代	社	十	0%
64	言	言	言	100%
65	九	分	九	50%
66	困	十	四	0%
67	思	一	思	50%
68	七	政	七	50%
69	山	山	山	100%
70	実	実	実	100%
71	入	本	入	50%
72	回	回	回	100%
73	場	見	保	0%
74	野	野	野	100%
75	開	国	開	50%
76	万	一	万	50%
77	全	工	全	50%
78	定	下	家	0%
79	家	安	下	0%

Tabel 4.14 (Sambungan)

No	Huruf Kanji	Dikenali Sebagai		% Pengenalan
		1	2	
80	了	了	下	50%
81	六	下	六	50%
82	問	五	問	50%
83	話	七	話	50%
84	文	文	文	100%
85	動	回	動	50%
86	度	学	道	0%
87	県	今	党	0%
88	水	分	本	0%
89	安	文	安	50%
90	氏	代	氏	50%
91	工	工	困	50%
92	政	政	代	50%
93	保	氏	保	50%
94	表	十	三	0%
95	道	金	道	50%
96	相	四	相	50%
97	意	安	意	50%
98	発	合	発	50%
99	不	工	不	50%
100	党	安	実	0%

Tabel 4.14 merupakan tabel karakter hasil pengujian terhadap data uji kanji tulisan tangan. Berdasarkan tabel terdapat karakter yang sama sekali salah

dikenali tetapi juga ada karakter yang 100% dapat dikenali dengan benar, ini dikarenakan kurangnya data latih untuk data uji kanji tulisan tangan. Karakter hasil tulisan tangan cenderung memiliki ukuran, posisi karakter dan bentuk garis yang jauh lebih bervariasi dibanding karakter cetak sehingga dibutuhkan data latih dengan jumlah dan variasi *template* yang lebih banyak. Ketebalan goresan karakter juga mempengaruhi persentase pengenalan karena karakter yang memiliki goresan terlalu tipis cenderung kehilangan bentuknya setelah mengalami *preprocessing*. Hal ini juga berlaku untuk karakter kanji cetak dengan ukuran *font* terlalu kecil.

Kemiripan bentuk pada beberapa karakter juga menjadi penyebab terjadinya kesalahan pengenalan karakter. Seperti pada kanji 田 beberapa kali dianggap mirip dengan kanji 困 dan 国, kanji tersebut memiliki bentuk yang saling menyerupai jika dibandingkan dengan karakter kanji yang lain yaitu berbentuk kubus. Kemiripan juga terjadi pada kanji 十 dimana salah dikenali sebagai kanji 一 dan 米, dianggap adanya kemiripan oleh sistem dikarenakan kanji 十 memiliki fitur bentuk yang juga dimiliki oleh kedua kanji tersebut. Kanji 正 juga dianggap mirip dengan kanji 上, karena adanya kesamaan pada fitur bentuk yang dimiliki. Hasil dari *preprocessing* seperti misalnya normalisasi ukuran dimana dilakukan *resize* ke ukuran resolusi yang lebih kecil juga dapat merubah bentuk suatu karakter kanji sehingga mirip dengan karakter kanji lain, contohnya pada kanji 生 yang salah dikenali sebagai kanji 主. Keduanya memiliki bentuk yang sangat mirip dimana hanya dibedakan fitur bentuk pada bagian atas dan kiri atas karakter, sehingga jika kedua fitur tersebut berkurang kejelasannya atau bahkan terhapus maka kedua karakter tersebut sulit untuk dibedakan. Hal ini terjadi karena proses *resize* ke ukuran resolusi yang lebih kecil berarti terjadi pengurangan jumlah piksel, yang mana mempengaruhi kedua fitur bentuk kanji tersebut.

Penggunaan resolusi citra yang lebih besar pada saat proses *resize* dapat mengurangi terjadinya kerusakan pada bentuk karakter sehingga pada karakter kanji yang memiliki kemiripan, fitur bentuk yang membedakan tetap terjaga.

LAMPIRAN

©UKDW

LAMPIRAN A

LISTING PROGRAM

A.1 Form Halaman_Utama

```
function varargout = Halaman_Utama(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @Halaman_Utama_OpeningFcn, ...
    'gui_OutputFcn', @Halaman_Utama_OutputFcn, ...
    'gui_LayoutFcn', [] , ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function Halaman_Utama_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);
axes1_CreateFcn();

function varargout = Halaman_Utama_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

function axes1_CreateFcn(hObject, eventdata, handles)
logo = imread('ukdw_logo.jpg');
imshow(logo);

function pushbutton1_Callback(hObject, eventdata, handles)
close(Halaman_Utama);
Halaman_Pelatihan;

function pushbutton2_Callback(hObject, eventdata, handles)
close(Halaman_Utama);
Halaman_Pengujian;
```

A.2 Form Halaman_Pelatihan

```
function varargout = Halaman_Pelatihan(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @Halaman_Pelatihan_OpeningFcn, ...
                  'gui_OutputFcn', @Halaman_Pelatihan_OutputFcn, ...
                  'gui_LayoutFcn', [] , ...
                  'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function Halaman_Pelatihan_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = Halaman_Pelatihan_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)
handles.output = hObject;
[x, y] = uigetfile('*.*', 'Pilih file citra awal');
path = strcat([y, x]);
handles.citra_asli = imread([y, x]);
axes(handles.axes1);
imshow(handles.citra_asli);
set(handles.edit1, 'String', path);
guidata(hObject, handles);

function edit1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
```

```
    set(hObject,'BackgroundColor','white');  
end
```

```
function edit3_Callback(hObject, eventdata, handles)
```

```
function edit3_CreateFcn(hObject, eventdata, handles)  
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function edit4_Callback(hObject, eventdata, handles)
```

```
function edit4_CreateFcn(hObject, eventdata, handles)  
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function edit5_Callback(hObject, eventdata, handles)
```

```
function edit5_CreateFcn(hObject, eventdata, handles)  
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function edit6_Callback(hObject, eventdata, handles)
```

```
function edit6_CreateFcn(hObject, eventdata, handles)  
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function pushbutton2_Callback(hObject, eventdata, handles)
```

```
handles.output = hObject;  
cek = get(handles.edit1, 'String');  
if isempty(cek)  
    warn = msgbox('Masukkan citra terlebih dahulu!!', 'Error', 'Error');  
else  
    citra_gray = rgb2gray(handles.citra_asli);  
    axes(handles.axes2);  
    imshow(citra_gray);  
    threshold = graythresh(citra_gray);  
    citra_biner = im2bw(citra_gray, threshold);
```

```

axes(handles.axes3);
imshow(citra_biner);
citra_biner = ~citra_biner;
[baris, kolom] = size(citra_biner);
hpp = sum(citra_biner, 2);
batas_potong = zeros(1,baris);
iindex = 1;

for index = 2:baris - 1
    if hpp(index) == 0
        if hpp(index + 1) ~= 0
            batas_potong(iindex) = index + 1;
        elseif hpp(index - 1) ~= 0
            batas_potong(iindex) = index - 1;
        end
        iindex = iindex + 1;
    end
end

batas_potong=batas_potong(batas_potong ~= 0);
citra_biner = ~citra_biner;
bar_subplot = 0;
kol_subplot = 0;
handles.crop_hpp = cell(1, length(batas_potong) / 2);
handles.crop_vpp = {};
index_hpp = 1;
index_vpp = 1;

for iindex = 1:2:length(batas_potong);
    crop = imcrop(citra_biner,[1 batas_potong(iindex) kolom batas_potong(iindex)
+ 1) - batas_potong(iindex)]);
    bar_subplot = bar_subplot + 1;
    handles.crop_hpp{index_hpp} = crop;
    index_hpp = index_hpp + 1;
    crop_a = ~crop;
    vpp = sum(crop_a, 1);
    [baris2, kolom2] = size(crop_a);
    iindex2 = 1;
    batas_potong2 = zeros(1, kolom2);

    for index2 = 2: kolom2 - 1
        if vpp(index2) == 0
            if vpp(index2 + 1) ~= 0 && vpp(index2 - 1) == 0
                batas_potong2(iindex2) = index2 + 1;
            elseif vpp(index2 - 1) ~= 0 && vpp(index2 + 1) == 0
                batas_potong2(iindex2) = index2 - 1;
            end
        end
    end
end

```



```

        end
        iindex2 = iindex2 + 1;
    end
end

batas_potong2 = batas_potong2(batas_potong2 ~= 0);
crop_b = ~crop_a;

for iindex2 = 1:2:length(batas_potong2)
    crop2 = imcrop(crop_b,[batas_potong2(iindex2) 1 batas_potong2(iindex2 +
1) - batas_potong2(iindex2) baris2]);
    kol_subplot = kol_subplot + 1;
    handles.crop_vpp{index_vpp} = crop2;
    index_vpp = index_vpp + 1;
end

end
axes(handles.axes4);
imshow(handles.crop_hpp{1});
handles.counter = 1;
axes(handles.axes5);
imshow(handles.crop_vpp{1});
handles.counter2 = 1;
handles.counter_max = length(handles.crop_hpp);
handles.counter_max2 = length(handles.crop_vpp);
handles.crop_res = cell(1, length(handles.crop_vpp));
for i_res = 1:length(handles.crop_vpp)
    resize = imresize(handles.crop_vpp{i_res}, [45, 35]);
    handles.crop_res{i_res} = resize;
end
axes(handles.axes6);
imshow(handles.crop_res{1});
handles.counter3 = 1;
handles.counter_max3 = length(handles.crop_res);
set(handles.text8, 'String', '1');
set(handles.text15, 'String', handles.counter_max);
set(handles.text16, 'String', '1');
set(handles.text17, 'String', handles.counter_max2);
set(handles.text19, 'String', '1');
set(handles.text20, 'String', handles.counter_max3);
handles.fitur_biner = cell(1, length(handles.crop_res));

for i_fb = 1:length(handles.crop_res)
    handles.fitur_biner{i_fb} = sprintf('%d', ~handles.crop_res{i_fb}(:));
end
set(handles.edit2, 'String', handles.fitur_biner{1});

```

```

end
guidata(hObject, handles);

function pushbutton3_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter = handles.counter + 1;
axes(handles.axes4);

if handles.counter > handles.counter_max
    handles.counter = 1;
    imshow(handles.crop_hpp{handles.counter});
    set(handles.text8, 'String', handles.counter);
else
    imshow(handles.crop_hpp{handles.counter});
    set(handles.text8, 'String', handles.counter);
end
guidata(hObject, handles);

function pushbutton4_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter = handles.counter - 1;
axes(handles.axes4);

if handles.counter < 1
    handles.counter = handles.counter_max;
    imshow(handles.crop_hpp{handles.counter});
    set(handles.text8, 'String', handles.counter);
else
    imshow(handles.crop_hpp{handles.counter});
    set(handles.text8, 'String', handles.counter);
end
guidata(hObject, handles);

function pushbutton5_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter2 = handles.counter2 - 1;
axes(handles.axes5);

if handles.counter2 < 1
    handles.counter2 = handles.counter_max2;
    imshow(handles.crop_vpp{handles.counter2});
    set(handles.text16, 'String', handles.counter2);
else
    imshow(handles.crop_vpp{handles.counter2});
    set(handles.text16, 'String', handles.counter2);
end

```

```
guidata(hObject, handles);
```

```
function pushbutton6_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter2 = handles.counter2 + 1;
axes(handles.axes5);
```

```
if handles.counter2 > handles.counter_max2
    handles.counter2 = 1;
    imshow(handles.crop_vpp{handles.counter2});
    set(handles.text16, 'String', handles.counter2);
```

```
else
```

```
    imshow(handles.crop_vpp{handles.counter2});
    set(handles.text16, 'String', handles.counter2);
```

```
end
```

```
guidata(hObject, handles);
```

```
function pushbutton7_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter3 = handles.counter3 - 1;
axes(handles.axes6);
```

```
if handles.counter3 < 1
```

```
    handles.counter3 = handles.counter_max3;
    imshow(handles.crop_res{handles.counter3});
    set(handles.edit2, 'String', handles.fitur_biner{handles.counter3});
    set(handles.text19, 'String', handles.counter3);
    set(handles.edit3, 'String', "");
    set(handles.edit4, 'String', "");
    set(handles.edit5, 'String', "");
    set(handles.edit6, 'String', "");
```

```
else
```

```
    imshow(handles.crop_res{handles.counter3});
    set(handles.edit2, 'String', handles.fitur_biner{handles.counter3});
    set(handles.text19, 'String', handles.counter3);
    set(handles.edit3, 'String', "");
    set(handles.edit4, 'String', "");
    set(handles.edit5, 'String', "");
    set(handles.edit6, 'String', "");
```

```
end
```

```
guidata(hObject, handles);
```

```
function pushbutton8_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter3 = handles.counter3 + 1;
axes(handles.axes6);
```

```

if handles.counter3 > handles.counter_max3
    handles.counter3 = 1;
    imshow(handles.crop_res{handles.counter3});
    set(handles.edit2, 'String', handles.fitur_biner{handles.counter3});
    set(handles.text19, 'String', handles.counter3);
    set(handles.edit3, 'String', '');
    set(handles.edit4, 'String', '');
    set(handles.edit5, 'String', '');
    set(handles.edit6, 'String', '');
else
    imshow(handles.crop_res{handles.counter3});
    set(handles.edit2, 'String', handles.fitur_biner{handles.counter3});
    set(handles.text19, 'String', handles.counter3);
    set(handles.edit3, 'String', '');
    set(handles.edit4, 'String', '');
    set(handles.edit5, 'String', '');
    set(handles.edit6, 'String', '');
end
guidata(hObject, handles);
.
function pushbutton9_Callback(hObject, eventdata, handles)
handles.output = hObject;
cek1 = get(handles.edit2, 'String');
cek2 = get(handles.edit3, 'String');
cek3 = get(handles.edit4, 'String');
cek4 = get(handles.edit5, 'String');
cek5 = get(handles.edit6, 'String');

if isempty(cek1) || isempty(cek2) || isempty(cek3) || isempty(cek4) ||
isempty(cek5)
    warn = msgbox('Isi semua kolom pada Data Entry terlebih dahulu!!', 'Error',
'Error');
else
    conn = database('coba_ocr', 'root', '');
    tablename = 'demo_ocr';
    colnames = {'fitur_biner', 'karakter', 'arti', 'kunyomi', 'onyomi'};
    data = {get(handles.edit2, 'String'), get(handles.edit3, 'String'), get(handles.edit4,
'String'), get(handles.edit5, 'String'), get(handles.edit6, 'String')};
    insert(conn, tablename, colnames, data);
    notif = msgbox('Database kanji telah diupdate', 'Done');
    close(conn);
end
guidata(hObject, handles);
function pushbutton10_Callback(hObject, eventdata, handles)
close(Halaman_Pelatihan);
Halaman_Utama;

```

A.3 Form Halaman_Pengujian

```
function varargout = Halaman_Pengujian(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @Halaman_Pengujian_OpeningFcn, ...
                  'gui_OutputFcn', @Halaman_Pengujian_OutputFcn, ...
                  'gui_LayoutFcn', [], ...
                  'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function Halaman_Pengujian_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = Halaman_Pengujian_OutputFcn(hObject, eventdata,
handles)
varargout{1} = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)
handles.output = hObject;
[x, y] = uigetfile('*.*jpg', 'Pilih file citra awal');
path = strcat([y, x]);
handles.citra_asli = imread([y, x]);
axes(handles.axes1);
imshow(handles.citra_asli);
set(handles.edit1, 'String', path);
guidata(hObject, handles);

function edit1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_CreateFcn(hObject, eventdata, handles)
```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

.

function pushbutton2_Callback(hObject, eventdata, handles)
handles.output = hObject;
cek = get(handles.edit1, 'String');

if isempty(cek)
    warn = msgbox('Masukkan citra terlebih dahulu!!', 'Error', 'error');
else
    citra_gray = rgb2gray(handles.citra_asli);
    axes(handles.axes2);
    imshow(citra_gray);
    threshold = graythresh(citra_gray);
    citra_biner = im2bw(citra_gray, threshold);
    axes(handles.axes3);
    imshow(citra_biner);
    citra_biner = ~citra_biner;
    [baris, kolom] = size(citra_biner);
    hpp = sum(citra_biner, 2);
    batas_potong = zeros(1,baris);
    iindex = 1;

    for index = 2:baris - 1
        if hpp(index) == 0
            if hpp(index + 1) ~= 0
                batas_potong(iindex) = index + 1;
            elseif hpp(index - 1) ~= 0
                batas_potong(iindex) = index - 1;
            end
            iindex = iindex + 1;
        end
    end
    batas_potong=batas_potong(batas_potong ~= 0);
    citra_biner = ~citra_biner;
    bar_subplot = 0;
    kol_subplot = 0;
    handles.crop_hpp = cell(1, length(batas_potong) / 2);
    handles.crop_vpp = {};
    index_hpp = 1;
    index_vpp = 1;

    for iindex = 1:2:length(batas_potong);

```

```

    crop = imcrop(citra_biner,[1 batas_potong(iindex) kolom batas_potong(iindex
+ 1) - batas_potong(iindex)]);
    bar_subplot = bar_subplot + 1;
    handles.crop_hpp{index_hpp} = crop;
    index_hpp = index_hpp + 1;
    crop_a = ~crop;
    vpp = sum(crop_a, 1);
    [baris2, kolom2] = size(crop_a);
    iindex2 = 1;
    batas_potong2 = zeros(1, kolom2);

    for index2 = 2: kolom2 - 1
        if vpp(index2) == 0
            if vpp(index2 + 1) ~= 0 && vpp(index2 - 1) == 0
                batas_potong2(iindex2) = index2 + 1;
            elseif vpp(index2 - 1) ~= 0 && vpp(index2 + 1) == 0
                batas_potong2(iindex2) = index2 - 1;
            end
            iindex2 = iindex2 + 1;
        end
    end

    batas_potong2 = batas_potong2(batas_potong2 ~= 0);
    crop_b = ~crop_a;
    for iindex2 = 1:2:length(batas_potong2)
        crop2 = imcrop(crop_b,[batas_potong2(iindex2) 1 batas_potong2(iindex2 +
1) - batas_potong2(iindex2) baris2]);
        kol_subplot = kol_subplot + 1;
        handles.crop_vpp{index_vpp} = crop2;
        index_vpp = index_vpp + 1;
    end

    end
    axes(handles.axes4);
    imshow(handles.crop_hpp{1});
    handles.counter = 1;
    axes(handles.axes5);
    imshow(handles.crop_vpp{1});
    handles.counter2 = 1;
    handles.counter_max = length(handles.crop_hpp);
    handles.counter_max2 = length(handles.crop_vpp);
    handles.crop_res = cell(1, length(handles.crop_vpp));

    for i_res = 1:length(handles.crop_vpp)  resize =
    imresize(handles.crop_vpp{i_res}, [45, 35]);
        handles.crop_res{i_res} = resize;

```

```

end
axes(handles.axes6);
imshow(handles.crop_res{1});
handles.counter3 = 1;
handles.counter_max3 = length(handles.crop_res);
set(handles.text8, 'String', '1');
set(handles.text15, 'String', handles.counter_max);
set(handles.text16, 'String', '1');
set(handles.text17, 'String', handles.counter_max2);
set(handles.text19, 'String', '1');
set(handles.text20, 'String', handles.counter_max3);
handles.fitur_biner = cell(1, length(handles.crop_res));

for i_fb = 1:length(handles.crop_res)
    handles.fitur_biner{i_fb} = sprintf('%d', ~handles.crop_res{i_fb}(:));
set(handles.edit2, 'String', handles.fitur_biner{1});
end
end
guidata(hObject, handles);
.
function pushbutton3_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter = handles.counter + 1;
axes(handles.axes4);
if handles.counter > handles.counter_max
    handles.counter = 1;
    imshow(handles.crop_hpp{handles.counter});
    set(handles.text8, 'String', handles.counter);
else
    imshow(handles.crop_hpp{handles.counter});
    set(handles.text8, 'String', handles.counter);
end
guidata(hObject, handles);

function pushbutton4_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter = handles.counter - 1;
axes(handles.axes4);

if handles.counter < 1
    handles.counter = handles.counter_max;
    imshow(handles.crop_hpp{handles.counter});
    set(handles.text8, 'String', handles.counter);
else
    imshow(handles.crop_hpp{handles.counter});
    set(handles.text8, 'String', handles.counter);
end

```



```

end
guidata(hObject, handles);

function pushbutton5_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter2 = handles.counter2 - 1;
axes(handles.axes5);

if handles.counter2 < 1
    handles.counter2 = handles.counter_max2;
    imshow(handles.crop_vpp{handles.counter2});
    set(handles.text16, 'String', handles.counter2);
else
    imshow(handles.crop_vpp{handles.counter2});
    set(handles.text16, 'String', handles.counter2);
end
guidata(hObject, handles);

function pushbutton6_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter2 = handles.counter2 + 1;
axes(handles.axes5);

if handles.counter2 > handles.counter_max2
    handles.counter2 = 1;
    imshow(handles.crop_vpp{handles.counter2});
    set(handles.text16, 'String', handles.counter2);
else
    imshow(handles.crop_vpp{handles.counter2});
    set(handles.text16, 'String', handles.counter2);
end
guidata(hObject, handles);
.
function pushbutton7_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter3 = handles.counter3 - 1;
axes(handles.axes6);

if handles.counter3 < 1
    handles.counter3 = handles.counter_max3;
    imshow(handles.crop_res{handles.counter3});
    set(handles.edit2, 'String', handles.fitur_biner{handles.counter3});
    set(handles.text19, 'String', handles.counter3);
else
    imshow(handles.crop_res{handles.counter3});
    set(handles.edit2, 'String', handles.fitur_biner{handles.counter3});

```

```

    set(handles.text19, 'String', handles.counter3);
end
guidata(hObject, handles);

function pushbutton8_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter3 = handles.counter3 + 1;
axes(handles.axes6);

if handles.counter3 > handles.counter_max3
    handles.counter3 = 1;
    imshow(handles.crop_res{handles.counter3});
    set(handles.edit2, 'String', handles.fitur_biner{handles.counter3});
    set(handles.text19, 'String', handles.counter3);
else
    imshow(handles.crop_res{handles.counter3});
    set(handles.edit2, 'String', handles.fitur_biner{handles.counter3});
    set(handles.text19, 'String', handles.counter3);
end
guidata(hObject, handles);

function pushbutton9_Callback(hObject, eventdata, handles)
handles.output = hObject;
cek = get(handles.edit2, 'String');

if isempty(cek)
    warn = msgbox('Proses citra uji terlebih dahulu!!', 'Error', 'error');
else
    setappdata(0, 'tamp_crop_res', handles.crop_res);
    setappdata(0, 'tamp_fit_bin', handles.fitur_biner);
    setappdata(0, 'tamp_citra_asli', handles.citra_asli);
    Halaman_Pengujian('Visible', 'Off');
    Halaman_Pengujian2;
end
guidata(hObject, handles);

function pushbutton10_Callback(hObject, eventdata, handles)
close(Halaman_Pengujian);
Halaman_Utama;

```

A.4 Form Halaman_Pengujian2

```
function varargout = Halaman_Pengujian2(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @Halaman_Pengujian2_OpeningFcn, ...
    'gui_OutputFcn', @Halaman_Pengujian2_OutputFcn, ...
    'gui_LayoutFcn', [] , ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function Halaman_Pengujian2_OpeningFcn(hObject, eventdata, handles,
varargin)
handles.output = hObject;
handles.crop_res = getappdata(0, 'tamp_crop_res');
handles.fitur_biner = getappdata(0, 'tamp_fit_bin');
handles.citra_asli = getappdata(0, 'tamp_citra_asli');
axes(handles.axes1);
imshow(handles.crop_res{1});
handles.counter = 1;
handles.counter2 = 1;
handles.counter_max = length(handles.crop_res);
handles.counter_max2 = length(handles.crop_res);
set(handles.text1, 'String', '1');
set(handles.text3, 'String', handles.counter_max);
set(handles.edit1, 'String', handles.fitur_biner{1});
guidata(hObject, handles);

function varargout = Halaman_Pengujian2_OutputFcn(hObject, eventdata,
handles)
varargout{1} = handles.output;

function edit1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function edit2_Callback(hObject, eventdata, handles)
```

```
function edit2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function edit3_Callback(hObject, eventdata, handles)
```

```
function edit3_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function edit7_Callback(hObject, eventdata, handles)
```

```
function edit7_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function edit8_Callback(hObject, eventdata, handles)
```

```
function edit8_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function edit9_Callback(hObject, eventdata, handles)
```

```
function edit9_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function axes1_CreateFcn(hObject, eventdata, handles)
```

```
function pushbutton1_Callback(hObject, eventdata, handles)
close(Halaman_Pengujian2);
Halaman_Pengujian('Visible', 'On');
```

```

function pushbutton2_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter = handles.counter - 1;
axes(handles.axes1);

if handles.counter < 1
    handles.counter = handles.counter_max;
    imshow(handles.crop_res{handles.counter});
    set(handles.edit1, 'String', handles.fitur_biner{handles.counter});
    set(handles.text1, 'String', handles.counter);
else
    imshow(handles.crop_res{handles.counter});
    set(handles.edit1, 'String', handles.fitur_biner{handles.counter});
    set(handles.text1, 'String', handles.counter);
end
guidata(hObject, handles);
.
function pushbutton3_Callback(hObject, eventdata, handles)
handles.output = hObject;
handles.counter = handles.counter + 1;
axes(handles.axes1);

if handles.counter > handles.counter_max
    handles.counter = 1;
    imshow(handles.crop_res{handles.counter});
    set(handles.edit1, 'String', handles.fitur_biner{handles.counter});
    set(handles.text1, 'String', handles.counter);
else
    imshow(handles.crop_res{handles.counter});
    set(handles.edit1, 'String', handles.fitur_biner{handles.counter});
    set(handles.text1, 'String', handles.counter);
end
guidata(hObject, handles);

function pushbutton4_Callback(hObject, eventdata, handles)
handles.output = hObject;
conn = database('coba_ocr', 'root', '', 'Vendor', 'MySQL', 'Server', 'localhost');
tic;
sqlquery = 'select fitur_biner from demo_ocr';
curs = exec(conn, sqlquery);
curs = fetch(curs);
data_latih = curs.Data;
tamp_korelasi1 = zeros(1, length(data_latih));
tamp_korelasi2 = zeros(1, handles.counter_max);
nilai_korelasi = zeros(1, handles.counter_max);

```

```

handles.fb_kanji = cell(1, handles.counter_max);
handles.kanji = cell(1, handles.counter_max);
handles.arti_kanji = cell(1, handles.counter_max);
handles.cara_baca1 = cell(1, handles.counter_max);
handles.cara_baca2 = cell(1, handles.counter_max);
mean1 = zeros(1, length(handles.fitur_biner));

for index = 1:length(handles.fitur_biner)
    sum_pix = 0;
    data_uji_baris = handles.fitur_biner{index};
    for iindex = 1:length(data_uji_baris)
        sum_pix = sum_pix + str2double(data_uji_baris(iindex));
    end
    mean1(index) = sum_pix / length(data_uji_baris);
end

mean2 = zeros(1, length(data_latih));
for index = 1:length(data_latih)
    sum_pix = 0;
    data_latih_baris = data_latih{index};
    for iindex = 1:length(data_latih_baris)
        sum_pix = sum_pix + str2double(data_latih_baris(iindex));
    end
    mean2(index) = sum_pix / length(data_latih_baris);
end

B = zeros(1, length(data_latih));
for index = 1:length(data_latih)
    tamp_B = 0;
    data_latih_baris = data_latih{index};
    mean_latih = mean2(index);
    for iindex = 1:length(data_latih_baris)
        hitung_B = (str2double(data_latih_baris(iindex)) - mean_latih)^2;
        tamp_B = tamp_B + hitung_B;
    end
    B(index) = sqrt(tamp_B);
end

C = zeros(1, length(handles.fitur_biner));
for index = 1:length(handles.fitur_biner)
    tamp_C = 0;
    data_uji_baris = handles.fitur_biner{index};
    mean_uji = mean1(index);
    for iindex = 1:length(data_uji_baris)
        hitung_C = (str2double(data_uji_baris(iindex)) - mean_uji)^2;
        tamp_C = tamp_C + hitung_C;
    end
end

```

```

end
C(index) = sqrt(tamp_C);
end

for index = 1:length(handles.fitur_biner)
    data_uji_baris = handles.fitur_biner{index};
    mean_uji = mean1(index);
    tamp_AA = zeros(1, length(data_uji_baris));
    for iindex = 1:length(data_uji_baris)
        hitung_AA = str2double(data_uji_baris(iindex)) - mean_uji;
        tamp_AA(iindex) = hitung_AA;
    end

    for iindex = 1:length(data_latih)
        tamp_A = 0;
        data_latih_baris = data_latih{iindex};
        mean_latih = mean2(index);
        for iiindex = 1:length(data_uji_baris)
            hitung_A = (str2double(data_latih_baris(iiindex)) -
mean_latih)*tamp_AA(iiindex);
            tamp_A = tamp_A + hitung_A;
        end
        r = tamp_A / (B(iiindex) * C(index));
        tamp_korelasi1(iindex) = r;
    end
    [value, id] = max(tamp_korelasi1);
    tamp_korelasi2(index) = id;
    nilai_korelasi(index) = value;
end
cek = nilai_korelasi
set(handles.text14, 'String', '1');
set(handles.text16, 'String', handles.counter_max2);
axes(handles.axes3);
imshow(handles.citra_asli);

for index = 1:length(tamp_korelasi2)
    id_kanji = num2str(tamp_korelasi2(index));
    sqlquery = ['select fitur_biner from demo_ocr where `index` = ' '' id_kanji '''];
    curs = exec(conn, sqlquery);
    curs = fetch(curs);
    handles.fb_kanji(index) = curs.Data;
    sqlquery = ['select karakter from demo_ocr where `index` = ' '' id_kanji '''];
    curs = exec(conn, sqlquery);
    curs = fetch(curs);
    handles.kanji(index) = curs.Data;
    sqlquery = ['select arti from demo_ocr where `index` = ' '' id_kanji '''];

```

```

curs = exec(conn, sqlquery);
curs = fetch(curs);
handles.arti_kanji(index) = curs.Data;
sqlquery = ['select kunyomi from demo_ocr where `index` = ' id_kanji ''];
curs = exec(conn, sqlquery);
curs = fetch(curs);
handles.cara_baca1(index) = curs.Data;
sqlquery = ['select onyomi from demo_ocr where `index` = ' id_kanji ''];
curs = exec(conn, sqlquery);
curs = fetch(curs);
handles.cara_baca2(index) = curs.Data;
end
handles.citra_kor = cell(1, length(tamp_korelasi2));

for i = 1:length(tamp_korelasi2)
    tamp_citra_kor = data_latih{tamp_korelasi2(i)};
    pars_citra_kor = zeros(1, length(tamp_citra_kor));
    for ii = 1:length(tamp_citra_kor)
        pars_citra_kor(ii) = str2double(tamp_citra_kor(ii));
    end
    handles.citra_kor{i} = reshape(~pars_citra_kor, 45, 35);
end

axes(handles.axes2);
imshow(handles.citra_kor{1});
set(handles.edit3, 'String', handles.fb_kanji(1));
set(handles.edit7, 'String', handles.arti_kanji(1));
set(handles.edit8, 'String', handles.cara_baca1(1));
set(handles.edit9, 'String', handles.cara_baca2(1));
kanji_urut = char(handles.kanji);
kanji_urut = sprintf('%s', kanji_urut(:));
set(handles.edit2, 'String', kanji_urut);
time = toc;
if time > 60
    menit = fix(time/60);
    detik = round(mod(time, 60));
    menit = num2str(menit, '%02i');
    set(handles.text17, 'String', menit);
    set(handles.text19, 'String', detik);
else
    detik = round(time);
    set(handles.text19, 'String', detik);
end

close(curs);
close(conn);

```



```
warn = msgbox('Proses template matching selesai!!', 'Done');  
guidata(hObject, handles); %store ui data
```

```
function pushbutton5_Callback(hObject, eventdata, handles)  
cek = get(handles.edit2, 'String');  
if isempty(cek)  
    warn = msgbox('Lakukan proses pengenalan pola terlebih dahulu!!', 'Error',  
'error');  
else  
    text_kanji = get(handles.edit2, 'String');  
    [y, x] = uiputfile('*.txt', 'Save Text Kanji');  
    path_file = fullfile(x, y);  
    file_id = fopen(path_file, 'wt', 'n', 'UTF-8');  
    fprintf(file_id, text_kanji);  
end
```

```
function pushbutton6_Callback(hObject, eventdata, handles)  
handles.output = hObject;  
handles.counter2 = handles.counter2 - 1;  
axes(handles.axes2);  
  
if handles.counter2 < 1  
    handles.counter2 = handles.counter_max2;  
    imshow(handles.citra_kor{handles.counter2});  
    set(handles.edit3, 'String', handles.fb_kanji(handles.counter2));  
    set(handles.edit7, 'String', handles.arti_kanji(handles.counter2));  
    set(handles.edit8, 'String', handles.cara_baca1(handles.counter2));  
    set(handles.edit9, 'String', handles.cara_baca2(handles.counter2));  
    set(handles.text14, 'String', handles.counter2);  
else  
    imshow(handles.citra_kor{handles.counter2});  
    set(handles.edit3, 'String', handles.fb_kanji(handles.counter2));  
    set(handles.edit7, 'String', handles.arti_kanji(handles.counter2));  
    set(handles.edit8, 'String', handles.cara_baca1(handles.counter2));  
    set(handles.edit9, 'String', handles.cara_baca2(handles.counter2));  
    set(handles.text14, 'String', handles.counter2);  
end  
guidata(hObject, handles);
```

```
function pushbutton7_Callback(hObject, eventdata, handles)  
handles.output = hObject;  
handles.counter2 = handles.counter2 + 1;  
axes(handles.axes2);
```

```
if handles.counter2 > handles.counter_max2  
    handles.counter2 = 1;
```

```
imshow(handles.citra_kor{handles.counter2});
set(handles.edit3, 'String', handles.fb_kanji(handles.counter2));
set(handles.edit7, 'String', handles.arti_kanji(handles.counter2));
set(handles.edit8, 'String', handles.cara_baca1(handles.counter2));
set(handles.edit9, 'String', handles.cara_baca2(handles.counter2));
set(handles.text14, 'String', handles.counter2);
else
imshow(handles.citra_kor{handles.counter2});
set(handles.edit3, 'String', handles.fb_kanji(handles.counter2));
set(handles.edit7, 'String', handles.arti_kanji(handles.counter2));
set(handles.edit8, 'String', handles.cara_baca1(handles.counter2));
set(handles.edit9, 'String', handles.cara_baca2(handles.counter2));
set(handles.text14, 'String', handles.counter2);
end
guidata(hObject, handles);
```

©UKDWN

LAMPIRAN B
DATA UJI DAN DATA LATIH CETAK

B.1 DATA UJI CETAK

No.	Data Uji	Jenis Font
1.	九困思七山実入回場野開万 日一 大年中会人本月長国出上十県水安 氏工政保表 見市力米自前田合立内道相意発不党	Adobe Fangsong Std R
2.	生子分東三行同今高金時手新明方部 女乙企四民対 主正代言 二事社者地京間田 全定家了六問話文動度体学下目五後	DotumChe

No.	Data Uji	Jenis Font
3.	<p>生子分東三行同今高金時手新明方部</p> <p>女乙企四民対</p> <p>主正代言</p> <p>二事社者地京間田</p> <p>全定家了六問話文動度体学下目五後</p>	HGS平成明朝体W9
4.	<p>九困思七山実入回場野開万</p> <p>日一 大年中会人本月長国出上十県水安</p> <p>氏工政保表</p> <p>見市力米自前円合立内道相意発不党</p>	KaiTi

No.	Data Uji	Jenis Font
5.	<p>九困思七山実入回場野開万</p> <p>日一 大年中会人本月長国出上十県水安</p> <p>氏工政保表</p> <p>見市力米自前円合立内道相意発不党</p>	Meiryo UI
6.	<p>生子分東三行同今高金時手新明方部</p> <p>女乙企四民対</p> <p>主正代言</p> <p>二事社者地京間田</p> <p>全定家了六問話文動度体学下目五後</p>	SimSum

B.2 DATA LATIH CETAK

No	Data Latih	Jenis Font
1.	<p>日一 大年中会人本月長国出上十 生子分東三行同今高金時手 見市力米自前円合立内 二事社者地京間田 体学下目五後</p> <p>新明方部女乙企四民対主正代言 九困思七山実入回場野開万 全定家了六問話文動度 県水安氏工政保表 道相意発不党</p>	DF-Kai SB

No .	Data Latih	Jenis Font
2.	<p>日一 大年中会人本月長国出上十 生子分東三行同今高金時手 見市力米自前円合立内 二事社者地京間田 体学下目五後</p> <p>新明方部女乙企四民対主正代言 九困思七山実入回場野開万 全定家了六問話文動度 県水安氏工政保表 道相意発不党</p>	HG平成明朝体W3
3.	<p>日一 大年中会人本月長国出上十 生子分東三行同今高金時手 見市力米自前円合立内 二事社者地京間田 体学下目五後</p>	HG平成角ゴシック体W 3

No .	Data Latih	Jenis Font
	<p>新明方部女乙企四民对主正代言 九困思七山実入回場野開万 全定家了六問話文動度 県水安氏工政保表 道相意発不党</p>	
4.	<p>日一 大年中会人本月長国出上十 生子分東三行同今高金時手 見市力米自前円合立内 二事社者地京間田 体学下目五後</p>	MingLiu

No .	Data Latih	Jenis Font
	<p>新明方部女乙企四民对主正代言 九困思七山実入回場野開万 全定家了六問話文動度 県水安氏工政保表 道相意発不党</p>	
5.	<p>日一 大年中会人本月長国出上十 生子分東三行同今高金時手 見市力米自前円合立内 二事社者地京間田 体学下目五後</p>	MS ゴシック

No .	Data Latih	Jenis Font
	<p>新明方部女乙企四民対主正代言 九困思七山実入回場野開万 全定家了六問話文動度 県水安氏工政保表 道相意発不党</p>	
6.	<p>日一 大年中会人本月長国出上十 生子分東三行同今高金時手 見市力米自前円合立内 二事社者地京間田 体学下目五後</p> <p>新明方部女乙企四民対主正代言 九困思七山実入回場野開万 全定家了六問話文動度 県水安氏工政保表 道相意発不党</p>	MS 明朝

LAMPIRAN C
DATA UJI DAN DATA LATIH TULISAN TANGAN

©UKDW

Responden

Nama :

Status :

No :

Kanji 1-50

日 一 大 年 中	会 人 本 月 長
国 出 上 十 生	子 分 東 三 行
同 今 高 金 時	手 見 市 力 米
自 前 円 合 立	内 二 事 社 者
地 京 間 田 体	学 下 目 五 後

Kanji 51-100

新 明 方 部 女	乙 企 四 民 対
主 正 代 言 九	困 思 七 山 実
入 回 場 野 開	万 全 定 家 了
六 問 話 文 動	度 県 水 安 氏
工 政 保 表 道	相 意 発 不 党

日	一	大	年	中
会	人	本	月	長
国	出	上	十	生
子	分	東	三	行
同	今	高	金	時

手	見	市	力	米
白	前	円	合	立
内	二	事	社	者
地	京	間	田	体
学	下	目	五	後

新	明	方	部	女
乙	企	四	民	对
主	正	代	言	九
困	思	七	山	突
入	回	場	野	開

万	全	定	家	了
六	問	話	文	動
度	具	水	安	氏
工	政	保	表	道
相	意	兇	不	党

日 一 大 年 中
国 出 上 十 生
同 今 高 金 時
自 前 田 合 立
地 京 間 田 体

長行米者後
本月三力社五
本東市事目
人分貝二下
会子手内学

新 明 方 部 女
主 正 代 言 九
入 回 場 野 開
六 問 話 文 動
工 政 保 表 道

乙 企 四 氏 对
困 思 七 山 突
万 全 定 家 了
度 鼎 水 安 氏
相 意 癸 不 党

Kartu Konsultasi Tugas Akhir

Program Studi Teknik Informatika Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana Yogyakarta
Dr. Wahidin Sudrahusada 5-25 Yogyakarta, 55224. Telp. (0274)503929



: KRISNA PRADITYA
: PENERAPAN OPTICAL CHARACTER RECOGNITION PADA HURUF KANJI JEPANG
MENGUNAKAN TEMPLATE MATCHING

Pembimbing I : Dra. Widi Hapsari, M.T.

Tanggal: 7 10 2015	Paraf: 	2 Tanggal: 2-11-2015	Paraf:
bab 1-2		program.	
Tanggal: 20-11-2015	Paraf: 	4 Tanggal: 2-11-2015	Paraf:
program.		Laporan bab 1-5 revisi ana Gisa.	
Tanggal: 10-12-2015	Paraf: 	6 Tanggal: 26-02-2016	Paraf:
bab 4 revisi program dim 25 x 2		• analisis : diperlukan lagi, opy hasil hanya menaruh / dihapuskan label.	
Tanggal: 02-05-2016	Paraf: 	8 Tanggal:	Paraf:



Program Studi Teknik Informatika Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana Yogyakarta
Dr. Wahidin Sudirahusada 5-25 Yogyakarta, 55224. Telp. (0274)563929

FORMULIR CATATAN UJIAN SKRIPSI

(Diisi oleh Ketua Tim Penguji)

Pada hari ini : **Jumat, 27 Mei 2016**, telah dilakukan Ujian Skripsi untuk mahasiswa tersebut dibawah ini:

Nama Mahasiswa : **KRISNA PRADITYA**
No. Induk Mahasiswa : **71110117**
Judul Skripsi : **PENERAPAN OPTICAL CHARACTER RECOGNITION PADA HURUF KANJI JEPANG MENGGUNAKAN TEMPLATE MATCHING**
Dosen Pembimbing I : **Widi Hapsari, Dra. M.T.**
Dosen Pembimbing II : **Ignatia Dhian E K R, S.Kom, M.Eng**
Keterangan : **LULUS / ~~TIDAK LULUS~~**
(coret yang tidak terpilih)

Beberapa perubahan/catatan yang harus dilakukan oleh mahasiswa tersebut diatas terkait dengan skripsi yang dikerjakannya:

NO.	CATATAN PERBAIKAN
1	Pemilihan 100 data dituliskan di bab 3, sumber dari mana?
2	Data latihan tulisan tangan dituliskan
3	
4	
5	
6	
7	
8	
9	
10	

Perubahan diatas harus sudah diselesaikan paling lambat tanggal : **Senin, 27 Juni 2016**

Yogyakarta, 27 Mei 2016
Ketua Tim Penguji

Dra Widi Hapsari, M.T.

Catatan:
- Untuk lembar untuk mahasiswa
- Untuk lembar untuk arsip



Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana Yogyakarta
Dr. Wahidin Sudirahusada 5-25 Yogyakarta, 55224. Telp. (0274)563929

FORMULIR PERBAIKAN (REVISI) SKRIPSI
Strata-1 Program Studi Teknik Informatika

Yang bertanda tangan di bawah ini:

Nama : KRISNA PRADITYA
N I M : 71110117
Judul Skripsi : PENERAPAN OPTICAL CHARACTER RECOGNITION PADA HURUF
KANJI JEPANG MENGGUNAKAN TEMPLATE MATCHING
Tanggal Pendadaran : 27 Mei 2016 10:00 WIB

Telah melakukan perbaikan tugas akhir dengan lengkap.

Demikian pernyataan kami agar dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 9 Juni 2016

Dosen Pembimbing I

Widi Hapsari, Dra. M.T.

Dosen Pembimbing II

Ignatia Dhian E K R, S.Kom, M.Eng

Dibetok tanggal: 9 Juni 2016 20:58 WIB