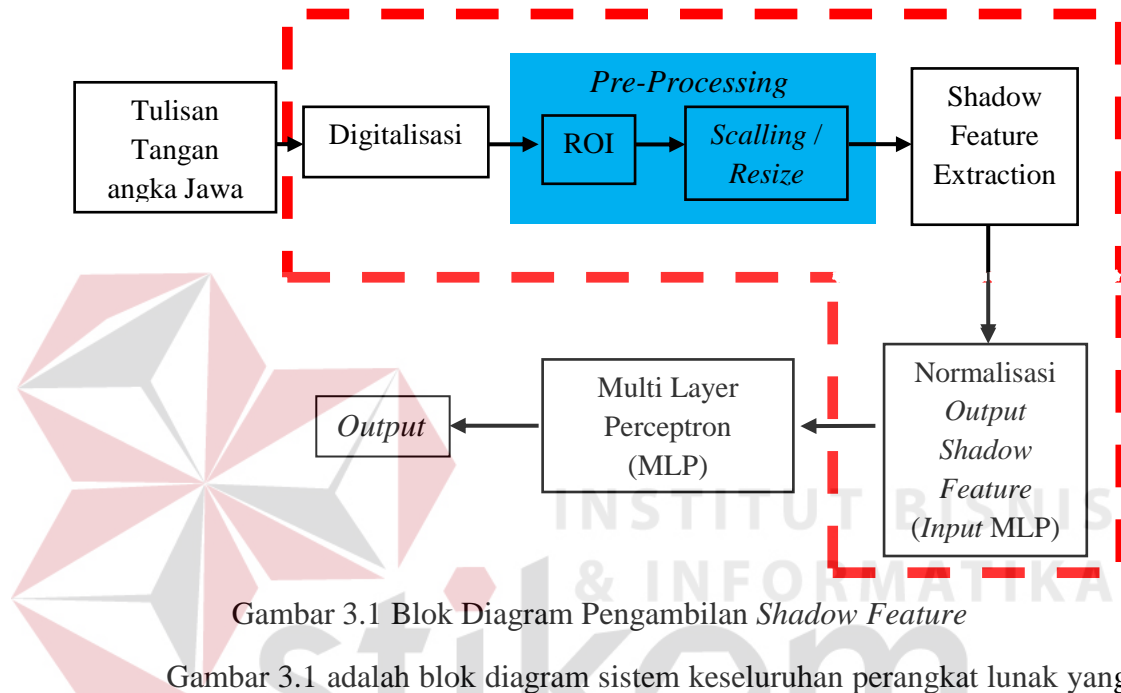


## BAB III

### METODE PENELITIAN

#### 3.1 Rancangan Penelitian



Gambar 3.1 Blok Diagram Pengambilan *Shadow Feature*

Gambar 3.1 adalah blok diagram sistem keseluruhan perangkat lunak yang dibuat, akan tetapi blok di dalam garis putus-putus berwarna merah merupakan fokus utama dari pengerjaan Tugas Akhir ini. Citra diambil dari media *input* interaktif yang memungkinkan untuk diberi tulisan tangan. Data citra dapat memiliki resolusi yang bervariasi.

Pertama-tama melakukan digitalisasi citra yang digambar ke biner, yang kedua adalah *preprocessing*, dan terakhir melakukan ekstraksi *shadow feature* sekaligus normalisasi nilai fitur. Ekstraksi ciri dilakukan dengan mengukur panjang bayangan dari tiap-tiap sisi pada masing-masing area. Namun sebelum diekstrak citra melalui tahap *preprocessing* dimana dalamnya ditentukan dulu ROI (*region*

*of interest*) dari citra dengan bentuk bujur sangkar dan kemudian *resize* ukuran (64 x 64 piksel).

### 3.2 Alat dan Bahan Penelitian

Alat dan bahan yang akan digunakan dalam penelitian ini terdiri dari beberapa alat pendukung untuk pembuatan dan pengujian program. Alat dan bahan yang akan digunakan sebagai berikut.

#### 3.2.1 Alat Penelitian

Alat yang digunakan untuk menyelesaikan penelitian ini adalah *Personal Computer* (PC)/ *Laptop*. Alat tersebut bertujuan untuk membuat simulasi 2 dimensi dengan menggunakan program *visual basic 6.0*.

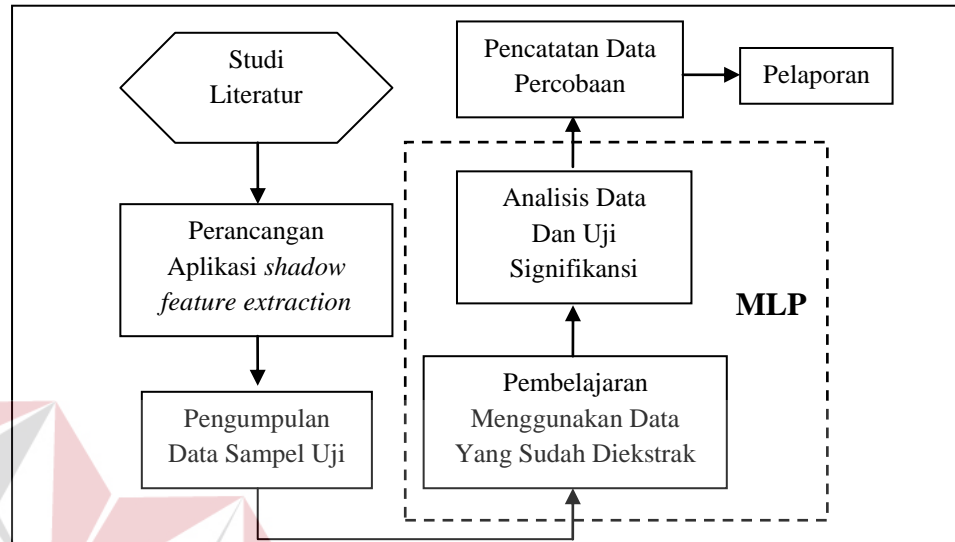
#### 3.2.2 Bahan Penelitian

Bahan yang akan diteliti oleh penulis dalam penelitian kali ini meliputi beberapa aspek sebagai berikut.

1. Pengambilan fitur bayangan menggunakan *software visual basic 6.0*.
2. Datasampel *set* citra dari 150 orang, 1 *set* terdiri 10 citra (angka 0-9) yang telah simpan pola penulisannya pada komputer. Untuk 100 *set* untuk keperluan training perangkat lunak dan 50 *set* untuk keperluan *testing*.
3. Microsoft excel 2007 digunakan untuk penyimpanan nilai hasil ekstraksi maupun sarana pengujian dan analisa keberhasilan metode yang diterapkan.
4. Matlab 7.0 yang digunakan untuk memetakan piksel citra menjadi bit-bit biner untuk keperluan saat pengujian secara manual.

### 3.3 Tahapan Penelitian

Langkah – Langkah yang digunakan dalam menyelesaikan tugas akhir ini adalah seperti pada Gambar 3.2:



Gambar 3.2 Diagram Alir Tugas Akhir

- a. Mengumpulkan literatur yang berhubungan dengan Tugas Akhir baik dari buku, jurnal, maupun dari *internet*. Serta memahami dengan rinci pengenalan pola tulisan menggunakan menggunakan *shadow feature extraction*.
- b. Melakukan perancangan perangkat lunak yang akan dibuat. Perancangan perangkat lunak dibagi menjadi beberapa tahap, yaitu :
  1. Merancang media untuk memasukkan pola huruf angka Jawa untuk keperluan sampel.
  2. Merancang proses ROI (*region of interest*) untuk keakuratan hasil ekstraksi.
  3. Merancang proses *resize* citra dengan dimensi 64 piksel x 64 piksel.

4. Merancang *shadow feature extraction* dan proses penyimpanan citra huruf yang sudah di-*input* agar tidak perlu *input* citra baru lagi untuk memudahkan saat pengujian.
  5. Merancang sistem proses evaluasi dan pencatatan.
- c. Mengambil sampel data pembelajaran pola huruf pada media yang dibuat pada perangkat lunak. Data pembelajaran harus bervariasi. Ada 10 pola huruf yang harus dikenali dan masing – masing pola diberi 100 variasi sampel data pembelajaran dan 50 sampel data untuk evaluasi.
  - d. Menjalankan proses pembelajaran sesuai dengan data pembelajaran yang diberikan point c, yang kemudian diekstrak menggunakan *shadow feature extraction*.
  - e. Proses pembelajaran menggunakan MLP yang sebelumnya citra sudah diekstrak fitur-fiturnya menggunakan *shadow feature extraction*.
  - f. Pencatatan data dan menjalankan sistem keseluruhan yang meliputi parameter-parameter yang sesuai dengan perumusan masalah yang dikemukakan.
  - g. Melakukan analisa, penarikan kesimpulan dan pengujian siginifikasi hasil yang didapat dari sampel yang diolah terlebih dahulu dengan mengekstrak cirinya dibandingkan dengan pembelajaran menggunakan data mentah.
  - h. Melaporkan dan mempublikasikan Tugas Akhir dalam bentuk jurnal.

### 3.3.1 Perancangan Perangkat Lunak

Melakukan perancangan perangkat lunak yang akan dibuat. Perancangan perangkat lunak dibagi menjadi beberapa tahap, yaitu :

## 1. Desain Kanvas

Pengambilan pola penulisan angka Jawa dari partisipan dilakukan pada media interaktif berupa kanvas yang digunakan untuk menangkap pola tulisan tangan dari partisipan. Kanvas tersebut akan dibuat dengan *software visual basic* menggunakan *tool picturebox*. Resolusi dari kanvas sebesar 256 x 256 piksel.

Dalam penggunaan kanvas tersebut partisipan cukup menggerakkan (*drag*) kursor sesuai pola angka Jawa yang akan digambar pada kanvas menggunakan *mouse/ mousepen* seperti pada Gambar 3.3 berikut.

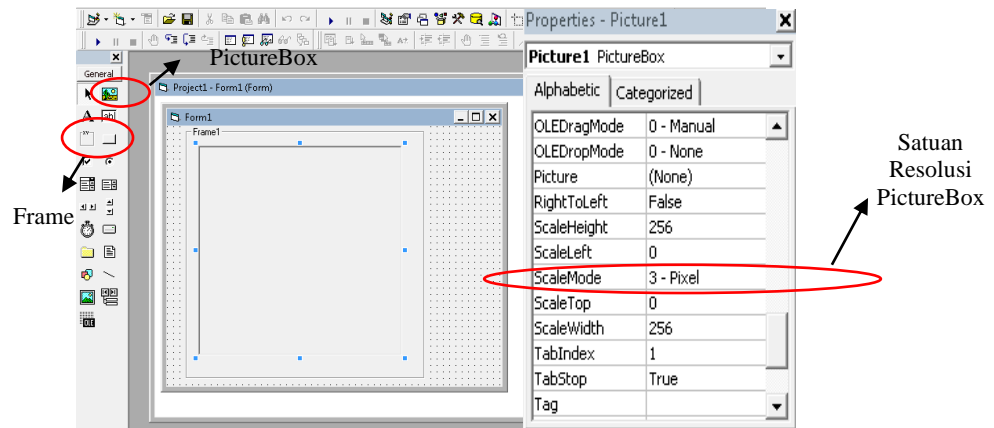


Gambar 3.3 Kanvas Kursor

Pola penulisan angka Jawa dari partisipan disimpan dengan format *bitmap* karena *bitmap* merupakan format gambar yang bisa dibaca oleh program grafis manapun. Langkah-langkah pembuatan kanvas adalah sebagai berikut:

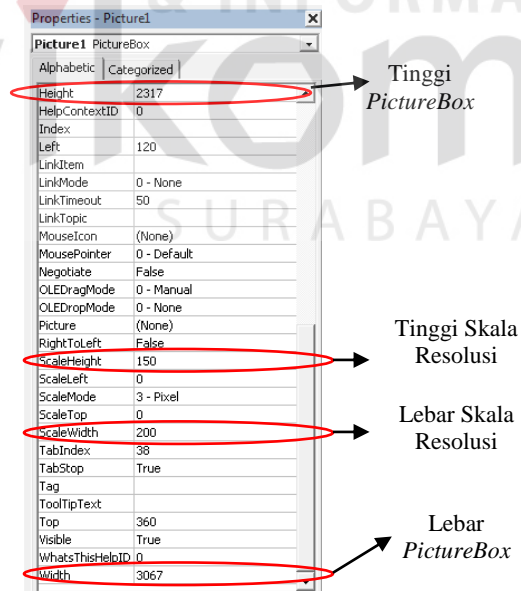
1. Arahkan kursor pada *toolbox*, kemudian pilih *frame*
2. Kemudian klik dan *drag* pada *form project*.
3. Kemudian arahkan lagi kursor pada *toolbox*, kemudian pilih *picturebox*.
4. Klik dan *drag picturebox* tapi kali ini kursor ada di area *frame*.

5. kemudian set satuan resolusi *picturebox* dalam piksel dengan mengatur *Scale Mode* pada *picturebox* dengan *pixel* seperti Gambar 3.4.



Gambar 3.4 Tampilan Kanvas dan Properti *PictureBox*

6. Kemudian setelah itu untuk mengatur lebar dan tinggi dari *picturebox* agar didapatkan skala resolusi dari *picturebox* (256 x 256), pada properti *picturebox* atur *Width* 3900 dan *Height* 3900. seperti Gambar 3.5 berikut:



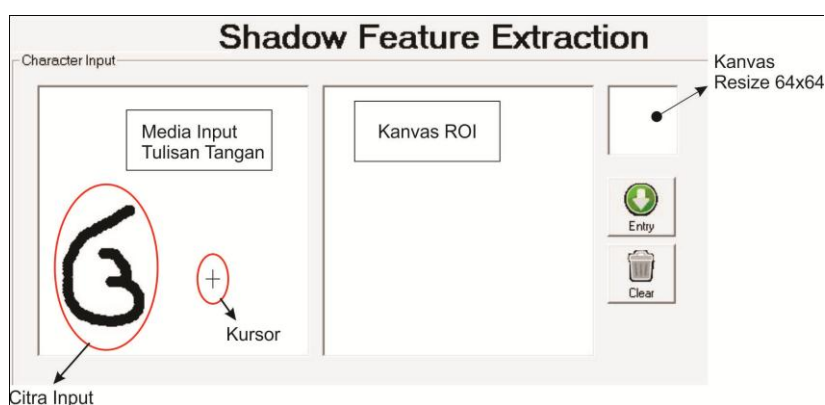
Gambar 3.5 Pengaturan Kanvas (*PictureBox*)

## 2. Digitalisasi

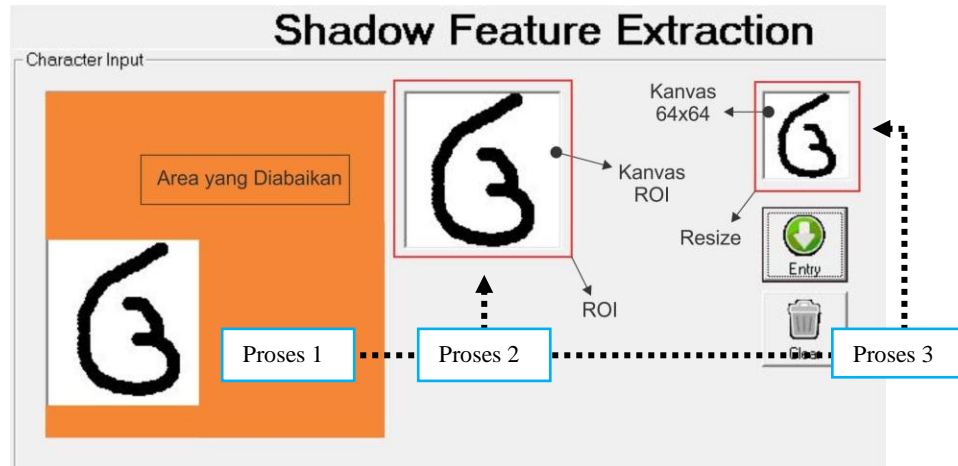
Digitalisasi adalah proses pembacaan warna piksel pada kanvas. Kanvas berwarna putih nilai biner 255 dan citra tulisan berwarna hitam nilai biner 0. Citra yang telah digambardi scan perbaris untuk dibaca warna pikselnya. Dalam scanning ini dilakukan pengklasifikasian terhadap piksel yang diambil, Koordinat piksel-piksel tersebut disimpan ke dalam *array* dua dimensi dengan format baris dan kolom.

## 3. Pembuatan ROI (*Region Of Interest*) dan *Scaling / Resize*

Proses ROI berfungsi untuk mengabaikan bagian (piksel) dari kanvas yang dirasa tidak perlu untuk diolah datanya. Selain itu setiap orang bisa berbeda-beda dimensi tulisan tangannya. Dalam program sebelum dilakukan proses ROI terlebih dahulu dilihat berapakah ukuran yang digambar oleh partisipan. Media *input* (*picturebox*) berukuran 256x256 (apabila citra diambil dari komputer) diambil kemudian digambar kembali ke kanvas ROI yang berukuran sama seperti Gambar 3.6, namun pada proses tersebut kanvas (*picturebox*) akan menyesuaikan dimensi dari tulisan tangan partisipan agar dapat terlihat perbedaan sebelum citra melalui ROI. Selanjutnya citra di-*resize* ke ukuran 64x64 piksel seperti Gambar 3.7,



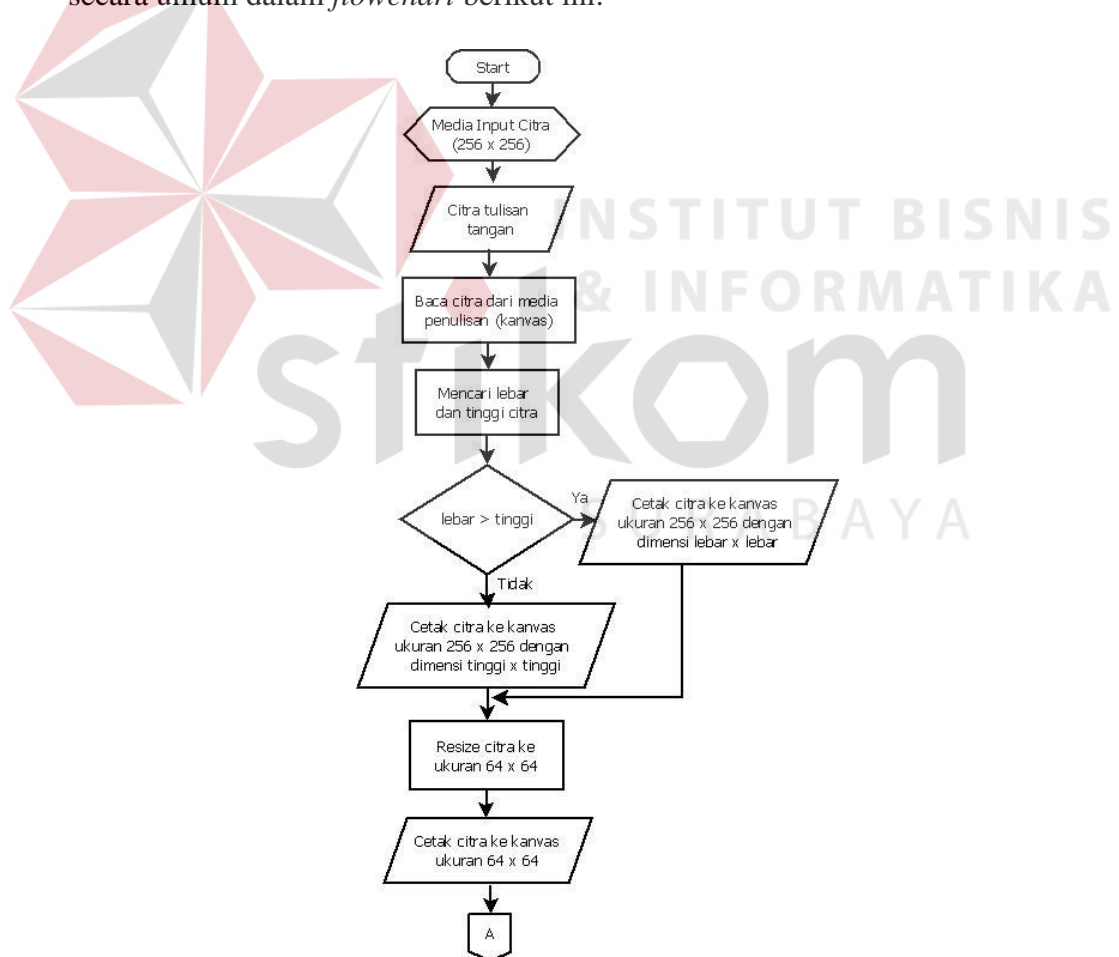
Gambar 3.6 Citra yang Akan Melewati Proses ROI dan *Scaling / Resize*



Gambar 3.7 Citra yang Telah Melewati Proses ROI dan *Scalling / Resize*

Langkah-langkah untuk melakukan ROI dalam program digambarkan

secara umum dalam *flowchart* berikut ini.



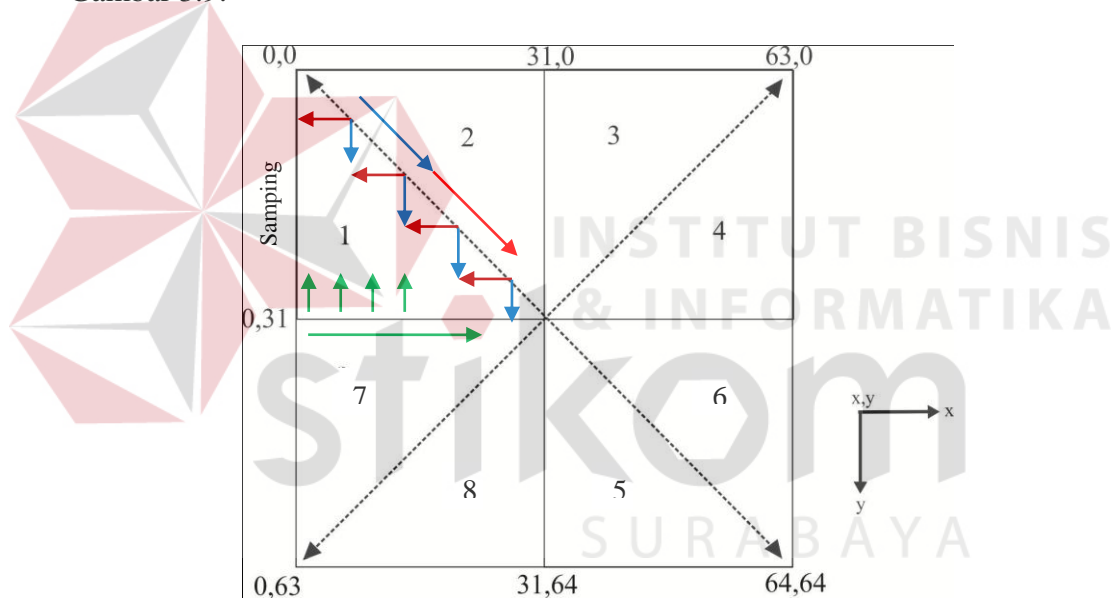
Gambar 3.8 *Flowchart* Preprocessing (ROI dan *Resize*)



#### 4. Ekstraksi Ciri Menggunakan *Shadow Feature*

Citra penulisan tangan dari *user* yang sudah melalui ROI (*region of interest*) dan *scalling / resize* baru dapat diolah untuk diekstrak. *shadow feature extraction* adalah pengenalan pola dari suatu citra dengan cara menghitung panjang bayangan dari bagian-bagian citra tertentu. Mula-mula citra digambar/dicetak kembali pada kanvas (*picturebox*) dimensi 64x64, kemudian kanvas tersebut dibagi menjadi delapan bagian yang masing-masing bagian mempunyai 3 sisi, jadi total fitur bayangan yang didapat berjumlah 24 fitur seperti

Gambar 3.9.



Gambar 3.9 Ilustrasi Pembagian Area Citra

Pada Gambar 3.9 telah tercantum koordinat-koordinat piksel kanvas (*picturebox*), nomer area, dan arah gerak proyeksi bayangan pada ke tiga sisi pada area 1. Bagian piksel dari citra yang berada di dalam area-area pada gambar diatas akan dihitung panjang bayangannya terhadap sisi yang menghadapnya.

Arah proyeksi fitur bayangan ke tiap sisi dari area 1 menunjukkan letak bayangan yang terproyeksi. Cara menghitung fitur bayangan dilakukan dengan

*scanning* pada tiap sisi-sisinya yaitu sisi samping, alas dan diagonal. *Scanning* berhenti jika scan pertama kali menemukan piksel yang berwarna hitam pada tiap baris atau kolom tergantung arah proyeksi bayangannya. Pergerakan dan arah *scanning* pada area 1 diilustrasikan oleh anak panah yang berwarna merah, biru dan hijau.

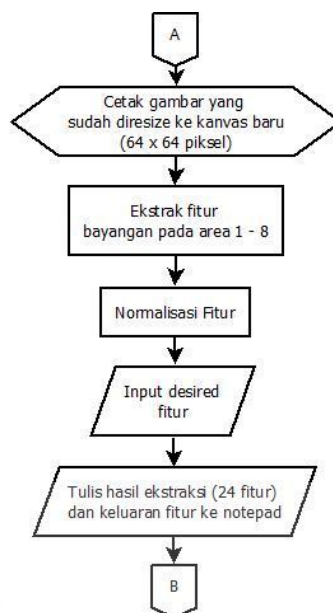
Ekstraksi fitur bayangan samping area 1 *scanning* bergerak sepanjang sisi diagonal ke bawah, sedangkan arah *scanning* dari sisi diagonal ke kiri (panah warna merah). Ekstraksi fitur alas pergerakan *scanning* sepanjang sisi diagonal ke bawah dan arah *scanning* dari sisi diagonal ke bawah (panah warna merah). Fitur bayangan diagonal pergerakan *scanning* dari samping ke kanan dan arah *scanningnya* dari alas ke atas (panah warna hijau). Seterusnya sampai dengan area 8 dengan menyesuaikan hadap masing area (segitiga) diatas.



Gambar 3.10 Tampilan Dalam *Software* Citra Yang Sudah Diekstrak

Langkah-langkah ekstraksi dijelaskan dengan *flowchart* pada Gambar 3.11

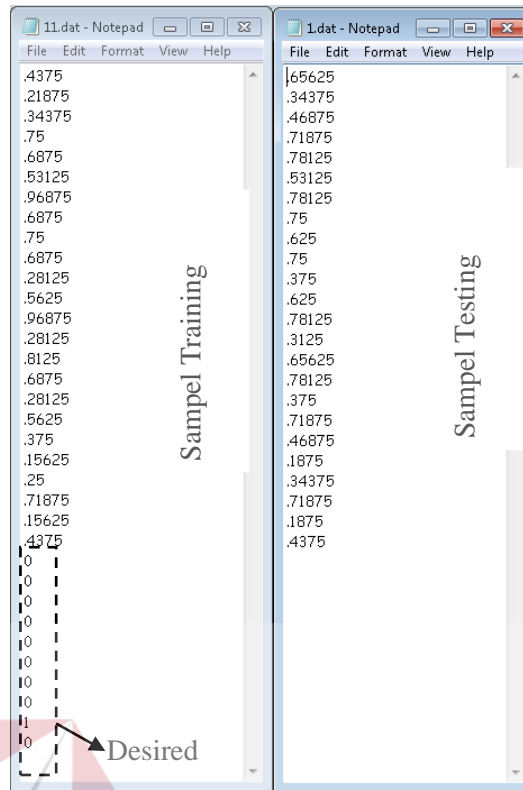
berikut:



Gambar 3.11 *Shadow Feature Extraction*

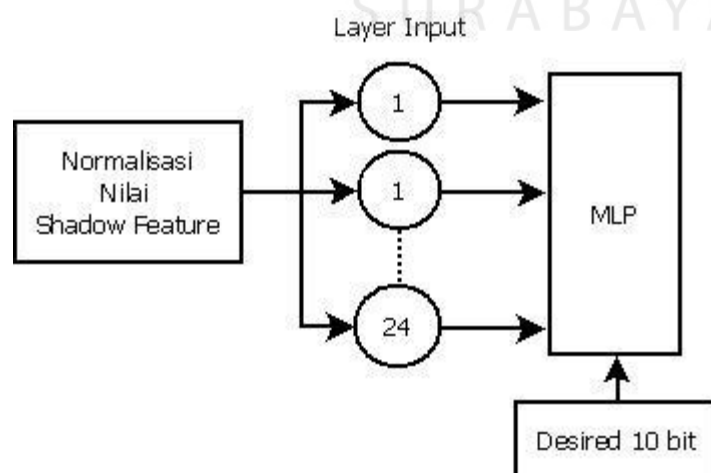
## 5. *Input Multi Layer Perceptron (MLP)*

Keluaran dari ekstraksi *shadow feature* adalah panjang bayangan fitur bayangan dari tiap-tiap sisi masing-masing area yang telah di normalisasi. Nilai fitur bayangan dari tiap area ini kemudian di normalisasi dengan cara dibagi dengan panjang total masing-masing sisi. Gambar 3.12 berikut ini adalah *capture* hasil ekstraksi yang sudah disimpan ke *notepad* masing-masing untuk keperluan sampel *training* dan sampel *testing*.



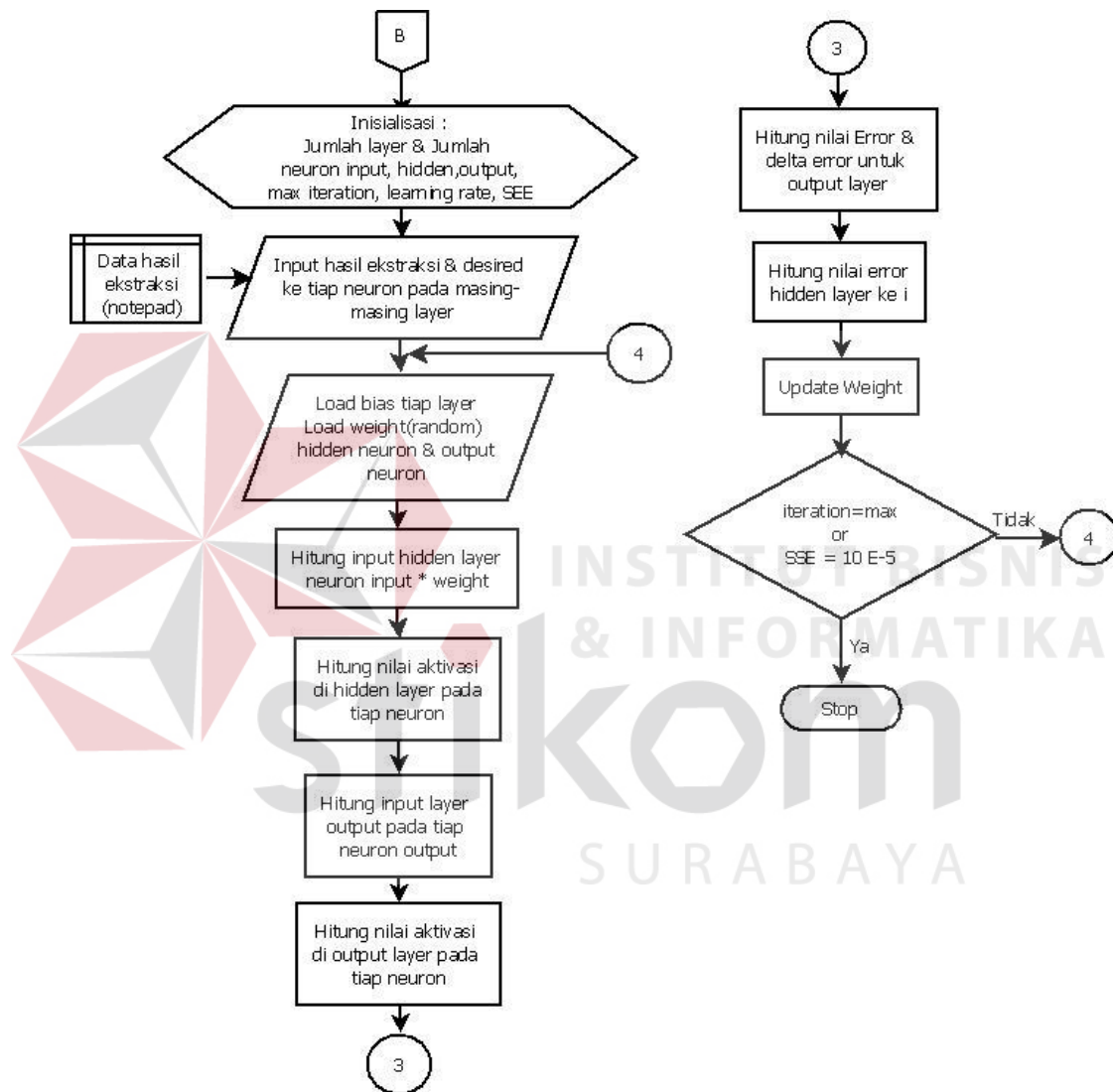
Gambar 3.12 Hasil Ekstraksi

Jumlah nilai normalisasi berjumlah 24 buah terdiri dari ekstraksi pada sisi area, dan *desired* berjumlah 10 buah total neuron layer input 34 buah seperti pada Gambar 3.13.

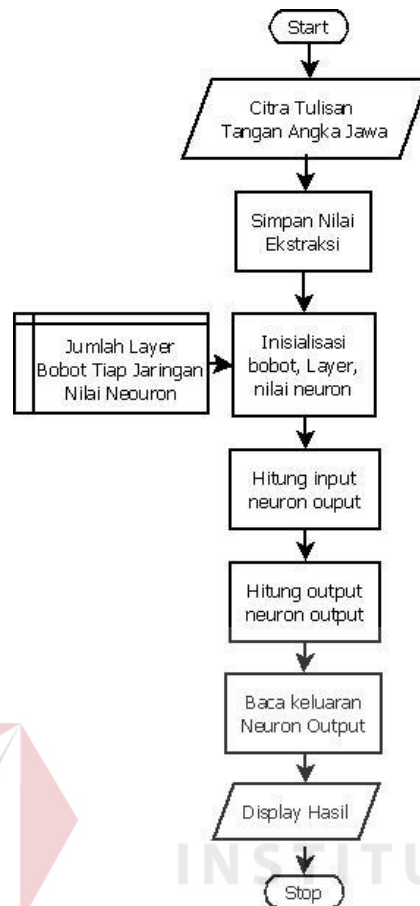


Gambar 3.13 Output Dari Shadow Feature Menjadi Input-an MLP

Langkah meng-*input*-kan nilai output ekstraksi (Gambar 3.14) dan pengujian MLP (Gambar 3.15) pengolahan data ekstraksinya diperlihatkan dalam *flowchart*.



Gambar 3.14 *Flowchart Training MLP*



Gambar 3.15 Flowchart Testing MLP

### 3.4 Teknik Pengumpulan & Analisis Data

#### 3.4.1 Pengumpulan Data

Data sampel primer untuk 1 *set* (10 pola angka) diperoleh dengan mengumpulkan hasil tulisan tangan dari orang yang berbeda. Karena ada 100 data sampel pembelajaran dan 50 data sampel uji untuk masing-masing angka, diperlukan 150 orang yang masing-masing memberikan 1 *set* sampel tulisan tangannya dalam angka Jawa.

Pengumpulan data sekunder dilakukan dengan melakukan percobaan di laboratorium setelah perangkat lunak sudah selesai dibuat dan sudah diperoleh bobot yang optimal seperti yang dijelaskan pada sub bab 3.3. Sistem MLP dengan

bobot optimal tersebut akan diuji menggunakan pola angka, baik dari sampel yang digunakan untuk pembelajaran, pengujian maupun citra yang baru digambarkan. Hasil pengujian antara sampel pelatihan, pengujian, dan data baru akan saling dibandingkan.

### 3.4.2 Pelatihan MLP

Aplikasi ini akan menghasilkan 2 buah file *log* untuk bobot yang dihasilkan. Bobot hasil akhir pembelajaran yang tercapai apabila syarat error minimum atau iterasi maksimum terpenuhi, akan disimpan ke dalam file *weight.dat*. Selain itu, ada file bernama *weight2.dat* untuk menyimpan informasi bobot setiap 100 iterasi. Hal ini diperlukan sebagai tindakan preventif bila terjadi hal-hal yang dapat menghentikan aplikasi sebelum mencapai akhir pembelajaran, yang bisa disebabkan karena komputer *hang* atau putusnya aliran listrik.

Proses pembelajaran dilakukan dengan konfigurasi sebagai berikut:

- Jumlah *hidden layer* : 3
- Jumlah *neuron input* : 24
- Jumlah *neuron layer 1* : 40
- Jumlah *neuron layer 2* : 40
- Jumlah *neuron layer 3* : 40
- Jumlah *neuron output* : 10
- Laju pembelajaran ( $\mu$ ) : 0.1
- *Error minimum* : 10E-4
- Iterasi maksimum : 10E+5

Proses pembelajaran dilakukan sebanyak 133.500 iterasi dengan menggunakan 100 *set* sampel *input* dengan nilai error akhir sebesar  $6.76168169753287E-04$ . Ini berarti bahwa proses pembelajaran selesai hingga target error minimum terhadap data sampel terpenuhi. Bobot yang dihasilkan akan disimpan pada file *weight.dat*.

### 3.4.3 Analisis Data

Data yang sudah terkumpul baik citra maupun hasil normalisasi ekstraksi (*notepad*) akan dilatihkan terhadap MLP dan dilakukan pengujian kembali pada data sampel yang diperoleh untuk menghitung nilai *error* pada perangkat lunak sesudah pembelajaran.

Setelah menganalisa data sampel untuk pelatihan berjumlah 100 *set* tulisan angka Jawa, proses selanjutnya adalah analisa data sampel pengujian, data sampel pengujian belum pernah dilatihkan pada MLP sebelumnya. Data tersebut diperoleh dari partisipan sebanyak 50 *set* tulisan angka Jawa, dan sudah disimpan hasil ekstraksinya. Kemudian pengujian secara langsung dengan citra tulisan angka Jawa yang baru diluar data sampel pelatihan dan sampel uji. Ada 3 parameter yang akan diperhatikan dalam pengujian yang akan dihasilkan pada tiap-tiap pengujian, yaitu apakah pengklasifikasian yang dihasilkan benar, salah, atau dikenali sebagai karakter yang lainnya.