**PENGENALAN TULISAN TANGAN HURUF HANGUL**

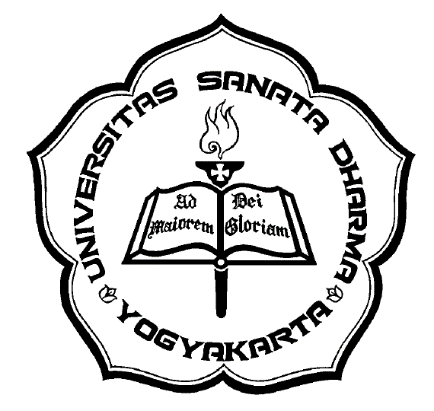
**DENGAN MENGGUNAKAN JARINGAN SARAF TIRUAN PROPAGASI BALIK**

**SKRIPSI**

Diajukan untuk Memenuhi Salah Satu Syarat

Memperoleh Gelar Sarjana Komputer

Program Studi Teknik Informatika



**Oleh :**

**Purbarini Sulysthian**

**135314125**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS SANATA DHARMA**

**YOGYAKARTA**

**PENGENALAN TULISAN TANGAN HURUF HANGUL**

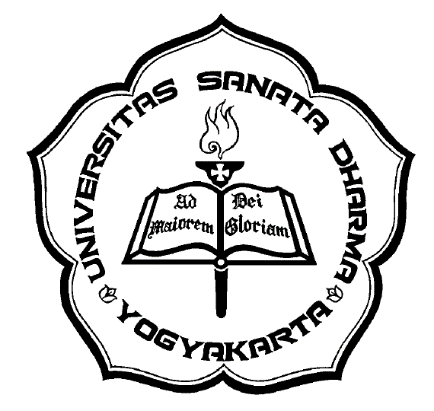
**DENGAN MENGGUNAKAN JARINGAN SARAF TIRUAN PROPAGASI BALIK**

**SKRIPSI**

Diajukan untuk Memenuhi Salah Satu Syarat

Memperoleh Gelar Sarjana Komputer

Program Studi Teknik Informatika



**Oleh :**

**Purbarini Sulysthian**

**135314125**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS SANATA DHARMA**

**YOGYAKARTA**

**2017**

# HALAMAN PERSETUJUAN

**SKRIPSI**

**PENGENALAN TULISAN TANGAN HURUF HANGUL**

**DENGAN MENGGUNAKAN JARINGAN SARAF TIRUAN PROPAGASI BALIK**

**Oleh:**

**Purbarini Sulysthian**

**135314125**

**Dosen Pembimbing**

**Dr. Anastasia Rita Widiarti Tanggal**

# DAFTAR ISI

[HALAMAN PERSETUJUAN ii](#_Toc501372894)

[DAFTAR ISI iii](#_Toc501372895)

[DAFTAR GAMBAR v](#_Toc501372896)

[DAFTAR TABEL vii](#_Toc501372897)

[BAB I 1](#_Toc501372898)

[1.1 Latar Belakang 1](#_Toc501372899)

[1.2 Rumusan Masalah 3](#_Toc501372900)

[1.3 Tujuan Penelitian 3](#_Toc501372901)

[1.4 Batasan Masalah 3](#_Toc501372902)

[1.5 Manfaat Penelitian 4](#_Toc501372903)

[1.6 Metode Penelitian 4](#_Toc501372904)

[1.7. Sistematika Penulisan Tugas Akhir 5](#_Toc501372905)

[BAB II 7](#_Toc501372906)

[2.1 Pengenalan Pola 7](#_Toc501372907)

[2.2 Huruf Hangul 7](#_Toc501372908)

[2.3 Pemrosesan Citra Digital 10](#_Toc501372909)

[2.4 Ekstraksi Fitur (Feature Extraction) 10](#_Toc501372910)

[2.4.1 Intensity of Character 10](#_Toc501372911)

[2.4.2 Mark Direction 10](#_Toc501372912)

[2.5 Jaringan Saraf Tiruan 11](#_Toc501372913)

[2.6 Jaringan Saraf Tiruan Propagasi Balik (*Backpropagation*) 11](#_Toc501372914)

[2.6.1 Pengertian Jaringan Saraf Tiruan Propagasi Balik 11](#_Toc501372915)

[2.6.2 Algoritma Jaringan Saraf Tiruan Propagasi Balik 12](#_Toc501372916)

[2.6.3 Simbol-simbol yang digunakan dalam Jaringan Saraf Tiruan Propagasi Balik 13](#_Toc501372917)

[2.6.4 Fungsi Aktivasi 14](#_Toc501372918)

[BAB III 18](#_Toc501372919)

[3.1 Bahan 18](#_Toc501372920)

[3.2 Peralatan Penelitian 18](#_Toc501372921)

[3.2.1 Hardware 18](#_Toc501372922)

[3.2.2 Software 18](#_Toc501372923)

[3.3 Tahap-tahap penelitian 19](#_Toc501372924)

[3.3.1 Pengumpulan data dan kebutuhan 19](#_Toc501372925)

[3.3.2 Pengolahan Data 20](#_Toc501372926)

[3.3.4 Analisis data 25](#_Toc501372927)

[3.4 Gambaran Umum Sistem 25](#_Toc501372928)

[3.5 Perancangan Program 27](#_Toc501372929)

[3.6 Perancangan Antarmuka Sistem 35](#_Toc501372930)

[3.7 Skenario Pengujian 37](#_Toc501372931)

[BAB IV 37](#_Toc501372932)

[4.1 Contoh Hasil Antarmuka 37](#_Toc501372933)

[4.2 Desain Jaringan 40](#_Toc501372934)

[4.3 Hasil Uji 41](#_Toc501372935)

[4.4 Testing Data Tunggal 59](#_Toc501372936)

[4.5 Analisis Data 60](#_Toc501372937)

[BAB V 63](#_Toc501372938)

[5.1 Kesimpulan 63](#_Toc501372939)

[5.2 Saran 63](#_Toc501372940)

[DAFTAR PUSTAKA 65](#_Toc501372941)

[LAMPIRAN 67](#_Toc501372942)

# DAFTAR GAMBAR

[Gambar 2. 1 Huruf Hangul 9](#_Toc501371480)

[Gambar 2. 2 Struktur Mark Direction (Surinta, 2010) 11](#_Toc501371481)

[Gambar 2. 3 Jaringan Saraf Tiruan Backpropagation dengan 1 Lapisan Tersembunyi (Andi, 2004) 12](#_Toc501371482)

[Gambar 2. 4 Binary sigmoid dengan jangkauan 0 dan 1 15](#_Toc501371483)

[Gambar 2. 5 bipolar sigmoid dengan jangkauan -1 dan 1 16](#_Toc501371484)

[Gambar 3. 1 Form Pengenalan Tulisan Tangan Huruf Hangul 19](#_Toc501371542)

[Gambar 3. 2 Contoh Form yang Telah Diisi 20](#_Toc501371543)

[Gambar 3. 3 Contoh Gambar yang Sudah di-Crop 20](file:///D:\rini\PENGENALAN%20POLA%20HURUF%20HANGUL%2013.docx#_Toc501371544)

[Gambar 3. 4 Contoh Gambar Citra yang sudah dibuah ke Citra Hitam Putih 21](#_Toc501371545)

[Gambar 3. 5 Conth Citra yang sudah di Resize 21](#_Toc501371546)

[Gambar 3. 6 Contoh Citra yang sudah ditipiskan 22](#_Toc501371547)

[Gambar 3. 7 Contoh Gambar yang sudah dibagi Menjadi 4 Segmen 22](#_Toc501371548)

[Gambar 3. 8 Contoh Citra yang sudah diekstrak Cirinya 23](#_Toc501371549)

[Gambar 3. 9 Contoh Program Jaringan Propagasi Balik 24](#_Toc501371550)

[Gambar 3. 10 Data Flow Diagram level 0 26](#_Toc501371551)

[Gambar 3. 11 Data Flow Diagram level 1 26](#_Toc501371552)

[Gambar 3. 12 Data Flow Diagram level 2 27](#_Toc501371553)

[Gambar 3. 13 Contoh Tampilan Utama Alat Uji 35](#_Toc501371554)

[Gambar 3. 14 Contoh Tampilan Panel Pelatihan Data 36](#_Toc501371555)

[Gambar 3. 15 Contoh Tampilan Panel Uji Data Tunggal 36](#_Toc501371556)

[Gambar 4. 1 Antarmuka Menu Utama Alat Uji 37](#_Toc501371570)

[Gambar 4. 2 Antarmuka Panel Pelatihan Data 38](#_Toc501371571)

[Gambar 4. 3 Contoh Hasil Pelatihan Data 38](#_Toc501371572)

[Gambar 4. 4 Contoh Men-load Data 39](#_Toc501371573)

[Gambar 4. 5 Antarmuka Panel Uji Data Tunggal 39](#_Toc501371574)

[Gambar 4. 6 Contoh Hasil Pengujian Data Tunggal 40](#_Toc501371575)

[Gambar 4. 7 Desain Jaringan 40](#_Toc501371576)

[Gambar 4. 8 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 1 42](#_Toc501371577)

[Gambar 4. 9 Hasil Perobaan 1 42](#_Toc501371578)

[Gambar 4. 10 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 2 43](#_Toc501371579)

[Gambar 4. 11 Hasil Percobaan 2 44](#_Toc501371580)

[Gambar 4. 12 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 3 45](#_Toc501371581)

[Gambar 4. 13 Hasil Percobaan 3 45](#_Toc501371582)

[Gambar 4. 14 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 4 46](#_Toc501371583)

[Gambar 4. 15 Hasil Percobaan 4 47](#_Toc501371584)

[Gambar 4. 16 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 5 48](#_Toc501371585)

[Gambar 4. 17 Hasil Percobaan 5 48](#_Toc501371586)

[Gambar 4. 18 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 6 49](#_Toc501371587)

[Gambar 4. 19 Hasil Percobaan 6 50](#_Toc501371588)

[Gambar 4. 20 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 7 51](#_Toc501371589)

[Gambar 4. 21 Hasil Percobaan 7 51](#_Toc501371590)

[Gambar 4. 22 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 8 52](#_Toc501371591)

[Gambar 4. 23 Hasil dari Percobaan 8 53](#_Toc501371592)

[Gambar 4. 24 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 9 54](#_Toc501371593)

[Gambar 4. 25 Hasil dari Percobaan 9 54](#_Toc501371594)

[Gambar 4. 26 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 10 55](#_Toc501371595)

[Gambar 4. 27 Hasil dari Percobaan 10 56](#_Toc501371596)

[Gambar 4. 28 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 11 57](#_Toc501371597)

[Gambar 4. 29 Hasil Percobaan 11 57](#_Toc501371598)

[Gambar 4. 30 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 12 58](#_Toc501371599)

[Gambar 4. 31 Hasil dari Percobaan 12 59](#_Toc501371600)

# DAFTAR TABEL

[Tabel 3. 1 Tabel Contoh Hasil Ekstrasi Ciri 23](#_Toc501371640)

[Tabel 4. 1 Tabel Hasil Testing Data Tunggal 60](#_Toc501371647)

# BAB I

**PENDAHULUAN**

## 1.1 Latar Belakang

Majunya dunia teknologi semakin memudahkan manusia untuk mengakses dan menerima informasi dari berbagai hal. Mendekatkan yang jauh adalah salah satu dampak nyata dari perkembangan dunia teknologi. Berbagai hal dapat kita akses dengan mudah, murah, dimana saja dan kapan saja. Salah satunya adalah kebudayaan, kita dapat mengetahui berbagai macam kebudayaan dari berbagai negara hanya dengan mengakses melalui jaringan internet saja. Berbicara tentang kebudayaan di seluruh dunia, tidak lengkap jika kita tidak berbicara tentang Korea Selatan, negara dengan berbagai kebudayaan, adat istiadat dan budaya K-Popnya. Budaya dari negara ini tidak henti-hentinya dibahas oleh anak muda indonesia yang sedang terserang virus *Hallyu wave* (gelombang korea). Berbicara mengenai budaya K-Pop dan kebudayaan Korea Selatan, maka berbicara juga tentang bahasa korea. Bahasa Korea berbeda dengan bahasa lain seperti Indonesia, Inggris, Jerman, dan negara-negara lainnya yang menggunkan alfabet ABCD untuk menulis. Bahasa Korea adalah bahasa yang menggunakan aksara dalam penulisannya sama seperti Mandarin dan Jepang. Huruf ini digunakan dalam kehidupan sehari-hari, dan mudah kita temukan dimana saja, seperti lagu-lagu korea, kebudayaan korea, drama korea dan lainnya. Dan hal inilah yang membuat banyak anak muda indonesia beramai-ramai untuk belajar aksara ini agar dapat mengerti dan dapat memahami kebudayaan Korea. Hal ini diperkuat dengan adanya data dari website *kto.visitkorea.or.kr* yang menunjukkan jumlah wisatawan Indonesia yang berkunjung ke negara gingseng ini mencapai 25,139 orang per Agustus 2016. Bahkan Indonesia menempati urutan ketujuh wisatawan terbanyak dari wisatawan seluruh dunia yang pernah berkunjung di negara Korea selatan. Dengan adanya fenomena ini, maka dibutuhkan suatu alat yang dapat

membantu wisatawan dan penggemar budaya Korea dalam memahami bahasa Korea, khususnya aksara Hangul.

Penelitian tentang pengenalan aksara sudah pernah dan banyak dilakukan oleh beberapa orang, mulai dari aksara Jepang, Hangul, Thailand, aksara Jawa, dll. Pada penelitian pengenalan aksara Hangul sebelumnya, yang dilakukan oleh Kim Sang Yoon, Kim Jin Wook dan Kim Hang Joon (1999) hasil keakuratan yang dihasilkan cukup tinggi, yaitu mencapai 96,5% pada 17500 data test dan 98,9% pada 19361 data training. Penelitian ini menggunakan metode Jaringan saraf tiruan dan Fuzzy Function, Jaringan Saraf Tiruan digunakan untuk mengklasifikan aksara dan Fuzzy Function digunakan untuk menyesuaikan sistem dengan kebiasaan indivu menulis aksara. Pada penelitian ini error disebabkan oleh tulisan yang tidak normal, hubungan posisi tulisan yang ambigu, karakter yang tidak dikenal dan penghubung yang terlalu banyak. Peneliti menggunakan citra dari pena elektronik yang digambarkan pada sebuah tablet dan pada proses prepocessing peneliti menghilangkan noise dan hook pada aksara. Penelitian tentang pengenalan aksara Hangul lainnya dengan menggunakan metode Jaringan Saraf Tiruan Propagasi Balik yang dilakukan oleh Siska Fathia (tanpa tahun), mahasiswa Teknik Informatika dari Universitas Dian Nuswantoro, Semarang, juga menghasilkan tingkat keakuratan yang tinggi, yaitu mencapai 99.7217% dan tingkat error sebesar 0.2783%. Penelitian lainnya yang menggunakan Jaringan Saraf Tiruan Propagasi Balik adalah penelitian yang dilakukan oleh Nicolaus Euclides Wahyu Utomo (2016), pada penelitian ini Sdr. Nicoulaus menggunakan Jaringan Saraf Tiruan Propagasi Balik untuk mengenali pola huruf Jepang Hiragana. Dan akurasi tertinggi yang didapat adalah 86,63% dengan menggunakan 2 alpisan tersembunyi.

Jaringan Saraf Tiruan adalah Jaringan Syaraf Tiruan (*Artificial Neural Network* (ANN)), atau juga disebut *Simulated Neural Network* (SNN), atau umumnya hanya disebut *Neural Network*(NN)), adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan jaringan syaraf manusia. Jaringan syaraf tiruan merupakan salah satu upaya manusia untuk memodelkan cara kerja atau fungsi sistem syaraf manusia dalam melaksanakan tugas tertentu. Pemodelan ini didasari oleh kemampuan otak manusia dalam mengorganisasikan sel-sel penyusunnya yang disebut neuron, sehingga mampu melaksanakan tugas-tugas tertentu, khususnya pengenalan pola dengan efektivitas yang sangat tinggi. Seperti halnya otak manusia, jaringan syaraf juga terdiri dari beberapa neuron, dan terdapat hubungan antara neuron-neuron tersebut. Neuron-neuron tersebut akan mentransformasikan informasi yang diterima, melalui sambungan keluarnya menuju ke neuron-neuron yang lain. Pada jaringan syaraf hubungan ini dikenal dengan nama bobot. Informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut.

Sedangkan jaringan saraf propagasi balik merupakan jaringan saraf yang menggunakan konsep jaringan berlapis jamak. Lapisan pertama adalah lapisan masukan (input) dan yang terakhir adalah lapisan keluaran (output). Lapisan diantara lapisan masukan dan lapisan keluaran disebut dengan lapisan tersembunyi (hidden).

Dengan adanya berbagai penelitian tentang pengenalan aksara dan penjelasan diatas, metode Jaringan Saraf Tiruan Propagasi Balik relevan dengan topik yang saya ambil. Karena dengan metode ini dapat mengenali aksara dengan baik dan hasil keakuratannya pun tinggi.

## 1.2 Rumusan Masalah

Berapa tingkat keakurasian yang dihasilkan dengan menggunakan metode Jaringan Saraf Tiruan Propagasi Balik ?

## 1.3 Tujuan Penelitian

Mengetahui performa alat uji dalam menterjemahkan aksara Hangul ke dalam huruf Latin berdasarkan tingkat keakurasian yang dihasilkan.

## 1.4 Batasan Masalah

1. Citra aksara yang digunakan dalam posisi normal.
2. Citra aksara yang digunakan adalah citra tulisan tangan aksara Hangul.
3. Aksara Hangul yang akan diuji adalah 14 huruf konsonan dasar.
4. Citra yang dipakai adalah citra yang dengan menggunakan pena.
5. Format citra yang akan digunakan adalah dalam bentuk jpg.
6. Citra digital diambil dengan cara di scan dengan melalui scanner.

## 1.5 Manfaat Penelitian

1. Membantu pengguna dalam mempelajari bahasa korea, khususnya dalam mempelajari huruf Hangul.
2. Memudahkan pengguna untuk membaca huruf Hangul dengan menterjemahkannya ke huruf latin.

## 1.6 Metode Penelitian

Metode penelitian yang digunakan adalah sebagai berikut:

1. Studi Pustaka

Tahap Studi adalah dasar dari penelitian ini, karena dengan adanya tahapan studi ini mahasiswa dibimbing dan dibekali dengan berbagai ilmu pengetahuan. Sehingga dalam proses studi ini mahasiswa diharapkan menyerap ilmu pengetahuan yang telah diberikan sebanyak-banyaknya dan kemudian mengimplementasikannya ke dalam tugas akhir yang akan diambil nantinya.

1. Pengumpulan Data

Pengumpulan data dalam penilitian ini diambil melalui form, form ini nantinya akan di scan untuk mendapatkan data digital. Citra yang digunakan adalah citra yang berbentuk tulisan tangan huruf Hangul. Kemudian citra yang sudah didapatkan dan diubah formatnya ke dalam bentuk jpg.

1. Pembuatan Alat Uji

Menyusun algoritma dari metode ekstraksi ciri dan metode Jaringan Saraf Tiruan Propagasi Balik untuk merancang alat uji kemudian mengimplementasikan algoritma tersebut ke dalam bahasa pemograman Matlab 2014.

1. Analisa Pembuatan Alat Uji

Analisa pembuatan alat uji ini dilakukan untuk menguji tingkat keakurasian alat uji yang telah dibuat. Pengujian dilakukan dengan menggunakan bahasa pemograman Matlab 2014.

1. Pembuatan Laporan

Pada tahap ini peneliti menulis semua kegiatan, algoritma, metode dan langkah-langkah pembuatan alat uji dll selama dalam penelitian. Pembuatan laporan ini bertujuan agar semua kegiatan dalam penilitian dapat didokumentasikan dengan baik dan rapi.

## 1.7. Sistematika Penulisan Tugas Akhir

BAB I Pendahuluan

Pada bab ini membahas latar belakang pemilihan judul tugas akhir, perumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metode penelitian dan sistematika penulisan proposal pada tugas akhir ini.

BAB II Landasan Teori

Bab ini berisi teori tentang huruf Korea atau Hangul, Pengenalan Pola, Ekstrasi Fitur *Zoning* , Ekstrasi ciri *Mark of Direction*, Ekstrasi ciri *Intensity of Character* dan juga teori tentang Jaringan saraf Tiruan Propagasi Balik.

BAB III Metode Penelitian

Bab ini akan membahas bagaimana citra gambar huruf Korea didapatkan atau dengan cara apa citra didapatkan, metode yang digunakan untuk mengolah citra yang sudah siap diuji, langkah-langkah pembuatan alat uji, cara mengolah data, cara pengujian dengan alat uji dan gambaran rancangan alat yang akan dibuat.

BAB IV Hasil Uji dan Analisis

Bab ini akan berisi hasil dari pengujian alat uji yang telah dibuat dan analisis dari seluruh pengujian.

BAB V Penutup

Bab ini berisi penutup yang didalamnya memuat kesimpulan dari hasil alat uji yang telah dibuat, kesimpulan dari seluruh pengujian yang telah dilakukan dan saran untuk peneliti yang akan melanjutkan penelitian ini.

# BAB II

**LANDASAN TEORI**

## 2.1 Pengenalan Pola

Pengenalan pola adalah suatu ilmu untuk mengklasifikasikan atau menggambarkan sesuatu berdasarkan pengukuran kuantitatif ciri atau sifat utama dari suatu obyek. Sedangkan pola adalah suatu entitas yang terdefinisi dan dapat diidentifikasi serta diberi nama. Pengenalan pola dapat dilakukan sebagai tindakan untuk mengolah data mentah dan membuat suatu aksi berdasar dari kategori dari pola data tersebut.

Pada dasarnya pengenalan pola terdiri dari 3 langkah utama yaitu pemrosesan awal, ekstraksi fitur dan klasifikasi. Pemrosesan awal merupakan langkah untuk memfokuskan obyek data yang akan dikenali dengan obyek lain yang tidak digunakan. Dalam hal ini peprosesan awal yang dilakukan terhadap obyek adalah dengan mengubah citra digital menjadi citra biner. Citra biner adalah citra digital yang hanya memiliki dua kemungkinan nilai piksel yaitu hitam dan putih (Sinaga, 2013).

## 2.2 Huruf Hangul

Hangul adalah sistem alfabet bahasa Korea. Hangul dibuat pada jaman Chosun tahun 1443 oleh raja Sejong dan beberapa ilmuan. Sebelum menggunakan Hangul, masyarakat Korea menggunakan huruf karakter Cina dalam kehidupan sehari-hari. Namun karena huruf karakter Cina sangat banyak jumlahnya, rumit, serta susah dipelajari, seringkali masyarakat Korea kesulitan dalam menggunakannya. Raja Sejong yang mengasihani rakyatnya membuat sistem alfabet Korea yang sesuai untuk melambangkan bunyi bahasa Korea serta mudah untuk dipelajari. Sistem alfabet ini disebut dengan “Hangul”. Saat Hangul diciptakan, Hangul disebut dengan ‘훈민정음(Hunminjeongeum)’ yang berarti “Bunyi Tepat untuk Mengajari Rakyat”.

Hangul terbagi sesuai dengan bunyinya, yakni vokal dan konsonan. Saat pertama kali dibuat Hangul terdiri atas 17 buah konsonan dan 11 buah vokal. Namun, pada perkembangannya, Hangul modern terdiri dari 40 buah alfabet, yakni 19 buah konsonan dan 21 vokal. Alfabet konsonan dibentuk berdasarkan teori bentuk dan jumlah coretan. Maksudnya, diantara konsonan terdapat konsonan dasar (ㄱ,ㄴ,ㅁ,ㅅ,ㅇ) yang merujuk pada bentuk titik artikulasi, serta 12 konsonan lainnya yang dibentuk dengan menambahkan coretan pada konsonan dasar berdasarkan keras tidaknya bunyi konsonan tersebut. Huruf vokal dibentuk dengan teori “천지인의 삼재(tiga struktur)”. Berdasarkan pada ㅇ, ㅡ, ㅣ, sedangkan vokal-vokal lain dibentuk dengan menggabungkan ketiga struktur diatas. Bentuk alfabet vokalㅇmerujuk pada bentuk langit yang melingkar, vokalㅡ merujuk pada bentuk tanah yang datar, dan vokalㅣ merujuk pada bentuk orang yang sedang berdiri (Hwa, 2008).



Gambar 2. 1 Huruf Hangul

Huruf vokal pada huruf Hangul terdiri dari 21 huruf vokal, 10 buah diantaranya adalah vokal dasar dan 11 buah adalah vokal perluasan bentukan dari bentuk vokal dasar. Vokal ditulis dengan urutan dari atas kebawah, kiri kekanan. Huruf vokal dasar dari huurf Hangul adalah ㅏ, ㅑ, ㅓ, ㅕ, ㅗ, ㅛ, ㅜ, ㅠ, ㅡ, ㅣ sedangkan huruf vokal perluasan adalah ㅐ, ㅒ, ㅔ, ㅖ, ㅘ, ㅙ, ㅚ, ㅝ, ㅞ, ㅟ, ㅢ.

Huruf konsonan pada huruf Hangul terdiri dari 19 buah konsonan, 14 diantaranya adalah konsonan dasar dan 5 lainnya adalah huruf konsonan ganda. Konsonan harus bertemu dengan vokal untuk membentuk suku kata agar dapat dilafalkan, konsonan tidak dapat dibunyikan bila berdiri sendiri. Konsonan dapat dilafalkan dengan satu atau lebih cara sesuai dengan vokal yang mengikutinya dan posisinya dalam suku kata. Huruf konsonan dasar dari huruf Hangul adalah ㄱ, ㄴ, ㄷ, ㄹ, ㅁ, ㅂ, ㅅ, ㅇ, ㅈ, ㅊ, ㅋ, ㅌ, ㅍ, ㅎ sedangkan konsonan ganda dari huruf Hangul adalah ㄲ, ㄸ, ㅃ, ㅆ, ㅉ (Hwa, 2008).

## 2.3 Pemrosesan Citra Digital

Sebuah citra dapat didefinisikan sebagai fungsi dua dimensi, f(x, y), x dan y merupakan koordinat spatial dan amplitudo dari f dengan pasangan koordinat(x,y) disebut itensitas citra atau gray level pada titik tersebut. Ketika x, y dan nilai amplitudo dari f adalah terbatas (finite), bernilai diskrit maka suatu citra dapat disebut sebagai citra digital (digital image). Citra digital ini terbentuk dari angka yang terbatas dan dapat diukur (finite) dari setiap elemennya yang memiliki lokasi dan nilai tertentu. Elemen-elemen tersebut sering kita sebut dengan picture elements, image elements, pels, dan pixels. Pemrosesan yang terkait dengan citra digital ini dapat didefinisikan sebagai suatu proses yang memiliki input dan output berupa image(citra) dan sebagai tambahan meliputi juga proses yang mengekstrasi atribut dari citra hingga sampai pada pengenalan masing-masing objek. (Gonzales dan Woods, 2008).

## 2.4 Ekstraksi Fitur (Feature Extraction)

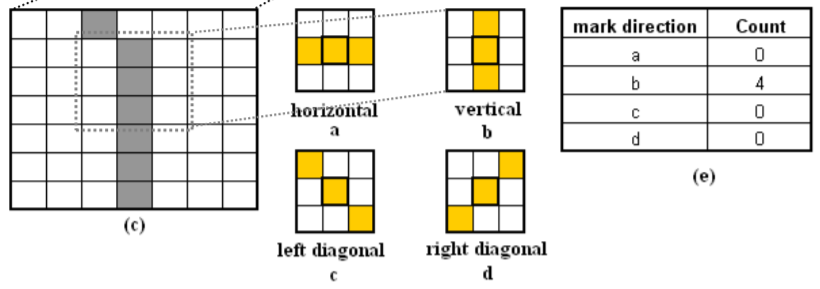
Ekstraksi fitur dilakukan untuk menyederhanakan citra dengan melakukan pengukuran fitur tertentu sehingga kita dapat mendapat atau melihat informasi dari citra tersebut (Sinaga, 2013). Pada hal ini, ekstrasi fitur yang saya gunakan adalah *Intensity Of Character dan Mark Direction*.

### 2.4.1 Intensity of Character

Intesitas karakter adalah ekstrasi ciri dengan cara menghitung jumlah piksel yang berwarna hitam.

### 2.4.2 Mark Direction

Tanda arah adalah ekstraksi ciri dengan cara mengitung banyaknya piksel yang memiliki tetangga yang berarah horizontal, vertical, diagonal ke kanan, dandiagonal ke kiri.



Gambar 2. 2 Struktur Mark Direction (Surinta, 2010)

## 2.5 Jaringan Saraf Tiruan

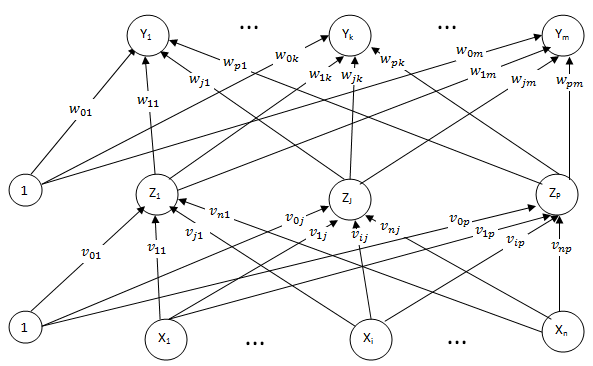
Jaringan saraf tiruan (artificial neural networks) atau disingkat JST adalah sistem komputasi dimana arsitektur dan operasi diilhami dari pengetahuan tentang sel saraf biologi dalam otak. JST dapat digambarkan sebagai model matematis dan komputasi untuk fungsi aproksimasi nonlinear, klasifikasi data, cluster dan regresi non parametik atau sebagai sebuah simulasi dari koleksi model saraf biologi.

Model saraf ditunjukan dengan kemampuannya dalam emulasi, analisa, prediksi, dan asosiasi. Berdasarkan kemampuan yang dimiliki JST dapat digunakan untuk belajar dan menghasilkan aturan atau operasi dari beberapa contoh, untuk menghasilkan output yang sempurna dari contoh atau input yang dimasukkan dan membuat prediksi tentang kemungkinan output yang akan muncul atau menyimpan karakteristik dari input yang disimpan kepadanya (Andri, 2004).

## 2.6 Jaringan Saraf Tiruan Propagasi Balik (*Backpropagation*)

### 2.6.1 Pengertian Jaringan Saraf Tiruan Propagasi Balik

Pelatihan sebuah jaringan yang menggunakan *backpropagation* terdiri dari 3 langkah, yaitu: pelatihan pola input secara *feedforward*, perhitungan dan backpropagation dari kumpulam kesalahan dan penyesuaian bobot. Sesudah pelatihan, aplikasi dalam jaringan dari fase *feedforward*. Bahkan jika pelatihan menjadi lambat, sebuah jaringan yang dilatih dapat menghasilkan outputnya sendiri secara cepat. Banyak variasi dari *backpropagation* yang dapat dibangun untuk meningkatkan kecepatan proses pelatihan (Andri, 2004).



Gambar 2. 3 Jaringan Saraf Tiruan Backpropagation dengan 1 Lapisan Tersembunyi (Andi, 2004)

Jaringan saraf banyak lapisan dengan satu lapisan dari unit yang tersembunyi dapat dilihat pada gambar diatas. Unit output (unit Y) dan unit yang tersembunyi juga mempunyai bias. Bias pada unit Yk disimbolkan dengan w0k dan bias pada unit tersembunyi Zj disimbokan dengan voj. Bias tersebut selalu bernilai 1. Dari gambar tersebut, dapat dilihat hanya aliran informasi langsung untuk fase forward pada operasi yang ditunjukkan (Andri, 2004).

### 2.6.2 Algoritma Jaringan Saraf Tiruan Propagasi Balik

Selama kondisi feedforward, masing-masing unit input input (Xi) menerima sebuah sinyal input dan mengirimkan sinyal tersebut ke masing-masing unit tersembunyi Z1,...,Zp. Masing-masing unit tersembunyi kemudian menghitung aktivasi dan mengirimkan sinyalnya (Zj) ke masing-masing unit output. Masing-masing unit output (Yk) kemudian menghitung aktivasinya (yk) untuk membentuk respon dalam jaringan yang diberi pola input.

Selama pelatihan, masing-masing unit output membandingkan aktivasi (yk) dengan nilai target (tk) kemudian menentukan kumpulan kesalahan untuk pola yang ada pada unit. Berdasarkan kesalahan tersebut, nilai faktor δk (k=1,...,m) dihitung kemudian δk digunakan untuk mendistribusikan kesalahan pada unit output (Yk) kembali ke semua unit pada lapisan sebelumnya. (unit tersembunyi dihubungkan dengan (Yk)). Kondisi semacam ini nanti akan digunakan untuk memperbaiki bobot diantara output dan lapisan tersembunyi. Untuk hal yang sama, nilai faktor δj (J=1,...,p) dihitung untuk masing-masing unit yang tersembunyi (Zj). Hal ini tidak diperlukan untuk menyebarkan kesalahan ke lapisan input, tetapi δj digunakan untuk memperbaiki bobot diantara lapisan yang tersembunyi dan dan lapisan input.

Sesudah δ ditentukan, bobot untuk semua lapisan disesuaikan secara simultan. Penyesuaian bobot wjk (dari unit tersembunyi Zj ke unit output Yk) didasarkan pada faktor δk dan aktivasi Zj pada unit tersembunyi Zj. Penyesuaian bobot vjk (dari unit input Xi ke unit tersembunyi Zj) didasarkan pada faktor δj dan aktivasi xi pada unit input (Andri, 2004).

### 2.6.3 Simbol-simbol yang digunakan dalam Jaringan Saraf Tiruan Propagasi Balik

Simbol-simbol yang digunakan pada algoritma pelatihan untuk jaringan backpropagation adalah sebagai berikut :

x input vektor pelatihan x=(x1, ..., xi, ..., xn)

t output vektor target t=(t1, ..., tk, ..., tm)

δk informasi tentang kesalahan pada unit Yk yang disebarkan kembali ke unit tersembunyi

δj informasi tentang kesalahan lapisan output ke unit tersembunyi Zj

α Learning rate

Xi Unit input 1

Untuk sebuah unit input, sinyal input dan sinyal output adalah sama yaitu x.

voj Bias pada unit tersembunyi j

Zj unit tersembunyi jInput jaringan ke Zj disimbolkan dengan 

Sinyal output(aktivasi) pada Zj disimbolkan dengan zj

Zj = ƒ(z\_inj)

vok Bias pada unit tersembunyi k

Yk unit tersembunyi lInput jaringan ke Yk disimbolkan dengan 

Sinyal output(aktivasi) pada Yk disimbolkan dengan yk

yk = ƒ(y\_ink)

### 2.6.4 Fungsi Aktivasi

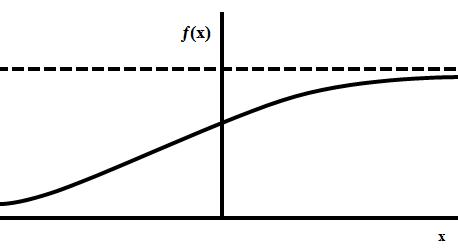
Fungsi aktivasi untuk sebuah jaringan backpropagation seharusnya mempunyai beberapa karakter yang peenting, berlanjut dan berbeda. Salah satu fungsi aktivasi yang biasa digunakan pada jaringan ini adalah fungsi binary sigmoid dimana mempunyai jangkauan 0 dan 1 dan didefinisikan dengan rumus :



Dengan



Fungsi tersebt dijelaskan pada gambar berikut ini :



Gambar 2. 4 Binary sigmoid dengan jangkauan 0 dan 1

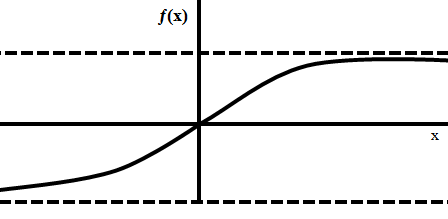
Untuk fungsi aktivasi lain adalah bipolar sigmoid yang mempunyai jangkauan (-1, 1) didefinisikan dengan rumus :



Dengan



Fungsi bipolar sigmoid dapat dilihat pada gambar berikut :



Gambar 2. 5 bipolar sigmoid dengan jangkauan -1 dan 1

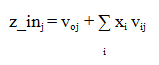
Algoritma Pelatihan

Algoritma dari jaringan backpropagation adalah sebagai berikut :

Langkah 1 inisialisasikan bobot(set ke nilai kecil secara acak)

Langkah 2 selama kondisi berhenti bernilai salah, kerjakan:

1. Untuk masing-masing pasangan pelatihan, lakukan : Feedforward
2. masing-masing unit input (Xi, i=1, ..., n) menerima sinyal input xi dan menyebarkan sinyal ini ke semua unit lapisan atas (unit tersembunyi)
3. masing-masing unit tersembunyi (Zj, j=1, ..., p) menjumlahkan bobot sinyal input



Dan mengaplikasikan fungsi aplikasi untuk menghitung sinyal output

Zj = ƒ(z\_inj)

Dan mengirimkan sinyal tersebut ke semua unit pada lapisan atas (unit output)

1. Masing-masing unit output ( Yk, k=1, ..., m) menjumlahkan bobot sinyal input



Dan mengaplikasikan fungsi aplikasi untuk menghitung sinyal output

yk = ƒ(y\_ink)

1. Untuk masing-masing pasangan pelatihan, lakukan : Backpropagation
2. Masing-masing unit output ( Yk, k=1, ..., m) menerima sebuah pola target yang bersesuaian dengan pola input pelatihan, menghitung informasi kesalahan,

δk = (tk - yk) ƒˈ (y\_ink)

kemudian menghitung koreksi bobot (digunakan untuk memperbaiki wjk)

∆wjk = αδk zj

Dan akhirnya menghitung koreksi bias (digunakan untuk memperbaiki w0k)

∆w0k = αδk

Setelah itu mengirimkan δk ke unit dalam lapisan yang paling atas

1. Masing-masing unit yang tersembunyi (Zj, j=1, ..., p) menjumlahkan input delta (dari unit lapisan atas)



Kalikan nilai ini dengan fungsi aktivasi untuk menghitung informasi kesalahan



Kemudian hitunglah koreksi bobot (digunakan untuk memperbaiki vij)



Setelah itu hitung koreksi bias (digunakan untuk memperbaiki v0j)



1. Perbaiki bobot bias

Masing-masing unit output output ( Yk, k=1, ..., m) memperbaiki bobot dan bias (j=0, ..., p)

Wjk (baru) = wjk (lama) + ∆ wjk

Masing-masing unit tersembunyi (Zj, j =1, ..., p) memperbaiki bobot dan bias (i=0, ..., n)

Vij (baru) = vij (lama) + ∆ vij

1. Tes kondisi berhenti

# BAB III

**METODE PENELITIAN DAN PERANCANGAN**

**PROGRAM**

## 3.1 Bahan

Data yang digunakan dalam tugas akhir ini adalah citra tulisan tangan huruf Hangul. Dari 40 Huruf Hangul, citra tulisan yang dipakai adalah 14 konsonan dasar huruf Hangul, yaituㄱ, ㄴ, ㄷ, ㄹ, ㅁ, ㅂ, ㅅ, ㅇ, ㅈ, ㅊ, ㅋ, ㅌ, ㅍ, ㅎ.

## 3.2 Peralatan Penelitian

### 3.2.1 Hardware

Hardware atau perangkat keras yang digunakan dalam penenlitian ini adalah :

1. *Processor* : Intel(R) Core(TM) i5-2450M CPU @2.50GHz 2.50GHz
2. *RAM*  : 4.00 GB
3. *Hard Drive* : 500 GB
4. *Graphic Interface* : NVIDIA GEFORCE 525M
5. *Scanner* : Canon Lide 120

### 3.2.2 Software

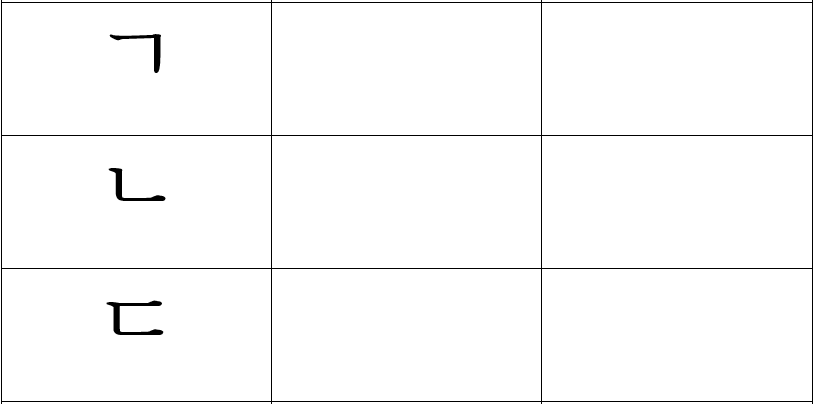
1. Microsoft Windows 10
2. Aplikasi Paint
3. Matlab R2017a

Sistem operasi Mircosoft Windows 10 digunakan agar Matlab R2017a dan Paint dapat berjalan

## 3.3 Tahap-tahap penelitian

### 3.3.1 Pengumpulan data dan kebutuhan

1. Membuat form untuk responden menulis data tulisan tangan huruf Hangul

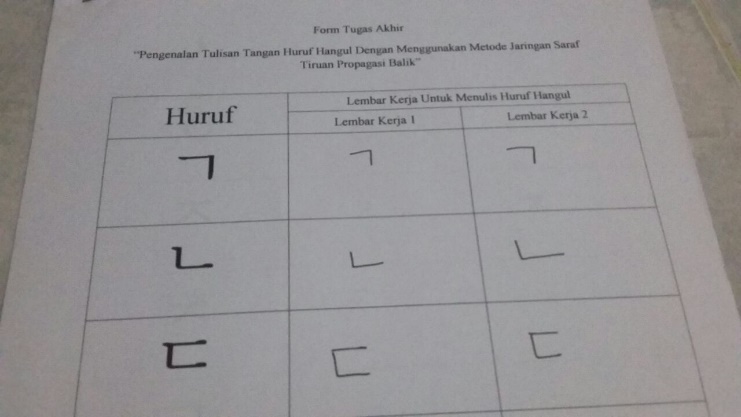


Gambar 3. 1 Form Pengenalan Tulisan Tangan Huruf Hangul

Data 14 konsonan tulisan tangan huruf Hangul didapatkan dari responden yang telah mengisi form dan responden menulis huruf sebanyak dua kali.

1. Pengumpulan data

Data didapatkan melalui form yang di bagikan kepada mahasiswa semester 5 jurusan Bahasa Korea, Universitas Gadjah Mada. Pada form yang dibagikan terdapat 2 kolom kosong disamping setiap 14 konsonan huruf Hangul. Mahasiswa diberi kesempatan untuk menulis tulisan tangan huruf Hangul sebanyak 2 x 14 data. Sehingga total yang diperoleh ada 25 x 2 x 14 data.



Gambar 3. 2 Contoh Form yang Telah Diisi

### 3.3.2 Pengolahan Data

Setelah mendapat data tulisan tangan Huruf Hangul, tahap selanjutnya adalah mengolah data, adapun tahap-tahap mengolah data antara lain :

1. Men-*scan* data

Setiap form yang didapatkan akan di scan melalui scanner, tujuan dari proses ini adalah mendapatkan data digital dari data tulisan tangan yang sebelumnya didapat pada form kuisoner. Setelah mendapat data tulisan tangan secara digital.

1. Meng-*crop* data digital

Kegiatan meng-*crop* data dilakukan dengan cara memotong form yang sudah di-*scan* untuk diambil data perhurufnya.



Gambar 3. 3 Contoh Gambar yang Sudah di-Crop

1. Pemrosesan Citra

Kegiatan pemrosesan citra dilakukan dengan tahap-tahap sebagai berikut :

1. Mengubah Citra Warna

Kegiatan ini bertujuan untuk mengubah citra yang tadinya berwarana menjadi citra keabuan, setelah mendapat citra keabuan langkah selanjutanya adalah mengubah citra keabuan menjadi citra hitam putih.



Gambar 3. 4 Contoh Gambar Citra yang sudah dibuah ke Citra Hitam Putih

1. *Resizing* Citra

Kegiatan ini dilakukan untuk mengubah ukuran dari citra, dari berbagai citra yang tidak sama ukurannya kemudian disamakan ukurannya menjadi 64 x 64 piksel. Hal ini dimaksudkan agar data yang diolah tidak terlalu besar dan dapat dibagi menjadi 4 segmen.



Gambar 3. 5 Conth Citra yang sudah di Resize

1. *Thinning* Citra

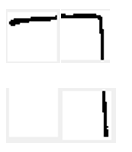
Kegiatan ini bertujuan untuk mengubah ketebalan citra menjadi 1 piksel



Gambar 3. 6 Contoh Citra yang sudah ditipiskan

1. Membagi Citra

Kegiatan ini bertujuan untuk membagi citra menjadi beberapa segmen yang lebih kecil. Tujuan dari kegiatan ini adalah lebih mempermudah dalam pencirian citra. Pada penelitia ini citra akan dibagi menjadi 4 segmen.



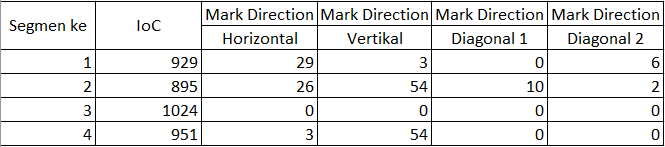
Gambar 3. 7 Contoh Gambar yang sudah dibagi Menjadi 4 Segmen

1. Mengambil Ciri Citra

Pada penelitian ini pengambila ciri citra menggunakan ekstrasi ciri *Intensity of Character* dan *Mark Direction*. *Intensity of Character* digunakan untuk mendapatkan jumlah piksel hitam dalam sebuah citra. Sedangkan *Mark Direction* digunakan untuk menghitung banyaknya piksel yang memiliki tetangga yang berarah horizontal, vertikal, diagonal ke kanan, dandiagonal ke kiri. Pada penelitian ini, *Intensity of Character* dan *Mark Direction* digunakan pada setiap segmen dari citra yang telah dibagi menjadi 4 segmen.



Gambar 3. 8 Contoh Citra yang sudah diekstrak Cirinya

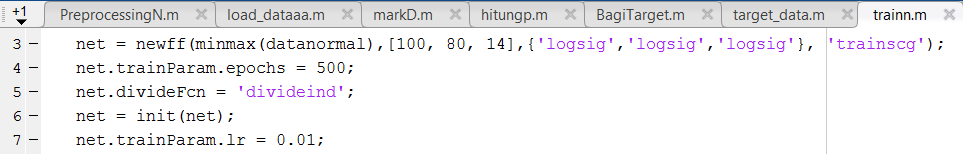


Tabel 3. 1 Tabel Contoh Hasil Ekstrasi Ciri

1. Mempersiapkan dan membuat alat uji algoritma *Backpropagation*

Langkah pertama dalam kegiatan ini adalah mengambil data ciri menggunakan script load\_dataaa.m dan menormalisasi data tesebut. Kemudian membuat target data menggunakan script target\_data.m dan dimasukkan ke variabel label. Kemudian mengubah vector target menjadi sebuah indeks dengan menggunakan script Bagi\_Target.m dan dimasukkan ke variabel l1 sampai l14.

1. Pelatihan Alat



Gambar 3. 9 Contoh Program Jaringan Propagasi Balik

Pada penilitian ini ada 43 data untuk setiap hurufnya dan ada 14 huruf dan total semua huruf ada 602 huruf. Untuk pelatihan alat digunakan data sebanyak 2/3 bagian untuk setiap hurufnya. Jadi total data yang digunakan dalam pelatihan alat 28 data huruf. Dikarenakan ada 602 huruf, maka total data dari semua huruf yang digunakan untuk data pelatihan sebanyak 392 huruf. Fungsi aktivasi yang digunakan dalam penilitian ini adalah logsig. Jumlah neuron yang dipakai ada beberapa, mulai dari 50 unit hingga 100 unit. Target yang telah disiapkan sebanyak 14 target karena terdapat 14 huruf Hangul yang berupa matriks.

Percobaan pelatihan alat dalam penelitian ini dilakukan sebanyak 12 kali. Pada setiap percobaan telah dilakukan perubahaan jumlah layer, jumlah neuron, atau fungsi aktivasi yang digunakan pada setiap layer untuk melihat peningkatan akurasinya.

1. Pengujian Alat

Pengujian alat dilakukan dengan menggunakan 6 kondisi ciri yang berbeda. Setiap percobaan dengan kondisi ciri yang berbeda, jumlah unit pada setiap hidden layer, atau jumlah iterasi yang dipakai telah dirubah dan terlihat ada peningkatan dan penurunan akurasi.

### 3.3.4 Analisis data

Pengujian dilakukan untuk menguji apakah pengenalan pola yang dilakukan sudah sesuai atau belum. Pengujian dalam penelitian ini adalah dengan menghitung keakurasian sistem yaiu dengan menghitung hasil jumlah data yang sesuai/benar dibagi dengan jumlah seluruh data. Pengujian dilakukan dengan rumus :

Akurasi (%) = Jumlah data benar x 100%

Jumlah semua data

## 3.4 Gambaran Umum Sistem

Rancangan sistem Pengenalan Tulisan Tangan Huruf Hangul dimulai dari data input yang berupa data ciri dari huruf Hangul yang akan dipakai. Untuk pengolahan di dalam hidden layer dipakai fungsi aktivasi logsig atau tansig . output dari sistem ini ada 14 karena targetnya berupa matriks 14 x1 yang akan dicocokan pada target masing-masing huruf. Rancangan ini sudah memiliki 3 komponen yaitu *input*, *hidden layer*, dan *output.*

Sistem ini akan digunakan untuk mengenali 14 konsonan huruf Hangul dengan menggunakan jaringan saraf tiruan propagasi balik.

Citra tulisan tangan huruf Hangul akan disimpan dengan format .jpg. kemudian akan dilakukan *preprocessing* Citra yang sudah dipotong sesuai dengan ukurannya kemudian akan diubah menjadi citra keabuaan dan citra hitam putih. Setelah menjadi citra biner maka langkah selanjutnya adalah mengubah ukuran citra menjadi 64 x 64 piksel. Kemudian citra dengan ukuran 64 x 64 ini akan ditipiskan untuk mendapatkan ketebalan citra 1 piksel.

Citra yang sudah ditipiskan kemudian akan dibagi menjadi 4 segmen. Dan dari setiap segmen akan diambil cirinya menggunakan ekstrasi ciri *Intensity of Character* dan *Mark Direction*. Ciri dari setiap segmen yang didapatkan akan dimasukkan kedalam sistem jaringan saraf tiruan. Citra tersebut akan dikenali jika memiliki kesamaan ciri dengan ciri dari data pelatihan yang dilatikan sebelumnya pada mesin *backpropagation*.

Penjelasan alur data dan kerja sistem yang telah dipaparkan diatas akan dijelaskan pada Data Flow Diagram berikut :



Gambar 3. 10 Data Flow Diagram level 0



Gambar 3. 11 Data Flow Diagram level 1



Gambar 3. 12 Data Flow Diagram level 2

## 3.5 Perancangan Program

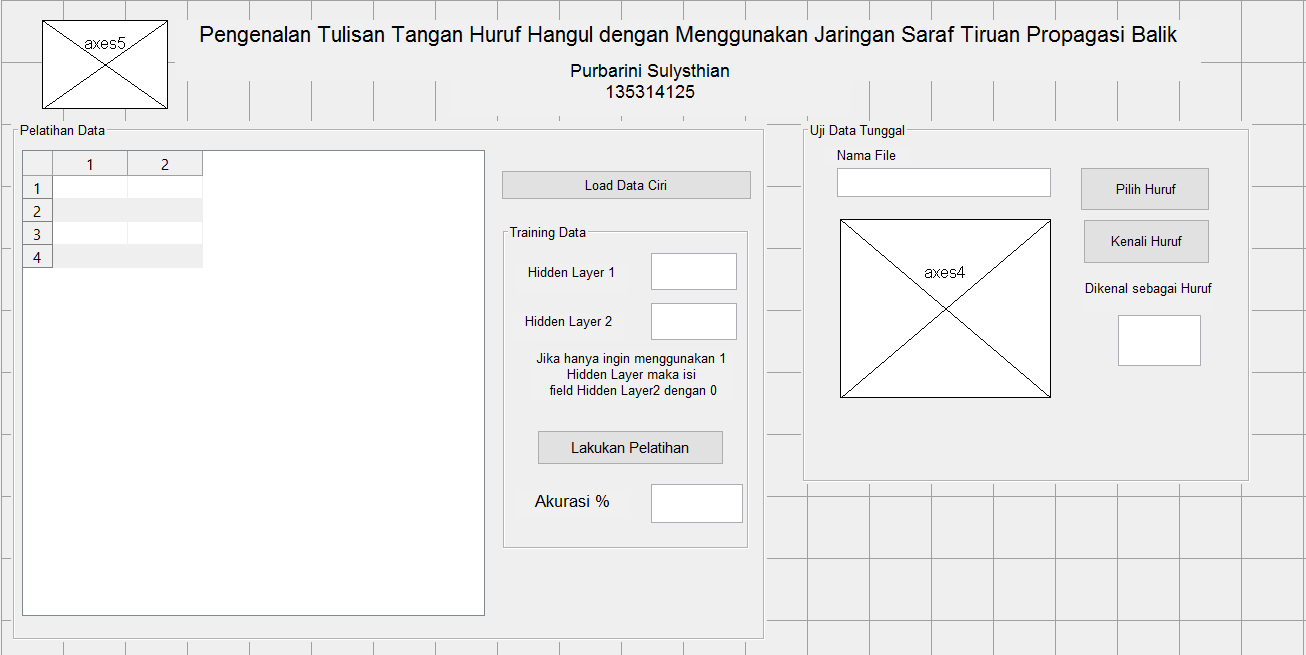
Pada penelitian ini telah dibuat beberapa scipt yang digunakan untuk megolah citra, membagi citra menjadi 4 segemen, mengambil ciri *Intensity of Character*, mengambil ciri *Mark Direction*, mengambil data ciri, membuat targaet, membagi target, pola jaringan, dan *testing* dan akurasi. Adapun algoritma dari setiap script yang telah dibuat sebagai berikut:

1. Algoritma Preprocessing
2. Mulai
3. Membaca file gambar citra huruf Hangul dengan menggunakan imread
4. Mengubah citra berwarna menjadi citra Grayscale dengan menggunakan fungsi rgb2gray
5. Mengubah citra keabuan menjadi citra hitam putih dengan fungsi im2bw
6. Mengubah ukuran citra menjadi 64x64 piksel dengan menggunakan fungsi imresize
7. Menipiskan citra dengan menggunakan Rosenfeld
8. Membagi huruf menjadi 4 segmen dengan ukuran 32x32 piksel
9. Mengekstrak ciri dengan Intensity of Character pada setiap segmen untuk menghitung jumlah piksel hitam pada setiap segmen
10. Memasukkan hasil ciri dari Intensity of Character kedalam array jh
11. Mengekstrak ciri dengan Mark Direction pada setiap segmen untuk menghitung banyaknya piksel yang memiliki tetangga yang berarah horizontal, vertikal, diagonal ke kanan, dandiagonal ke kiri
12. Memasukkan hasil ciri dari Mark Direction untuk arah horizontal kedalam array horz
13. Memasukkan hasil ciri dari Mark Direction untuk arah vertikal kedalam array vert
14. Memasukkan hasil ciri dari Mark Direction untuk arah diagonal 1 kedalam array dig1
15. Memasukkan hasil ciri dari Mark Direction untuk arah diagonal 2 kedalam array dig2\Membuat kombinasi ciri 1 dengan memasukkan variabel jh ke array ciri1
16. Membuat kombinasi ciri 2 dengan memasukkan variabel dig1 ke array ciri2
17. Membuat kombinasi ciri 3 dengan memasukkan variabel jh dan dig1 ke array ciri3
18. Membuat kombinasi ciri 4 dengan memasukkan variabel jh dan dig2 ke array ciri4
19. Membuat kombinasi ciri 5 dengan memmsukkan variabel dig1 dan dig2 ke array ciri5
20. Membuat kombinasi ciri 6 dengan memmasukkan variabel jh, dig1, dig2, horz, dan vert ke array ciri
21. Selesai
22. Algoritma BagiZona
23. Mulai
24. Membuat array baris & kolom yang merupakan ukuran baru dari citra
25. Membagi citra ke segmen pertama dari baris 1 sampai 32 dan kolom 1 sampai 32 lalu disimpan ke z1
26. Membagi citra ke segmen kedua dari baris 1 sampai 32 dan kolom 33 sampai 64 lalu disimpan ke z2
27. Membagi citra ke segmen ketiga dari baris 33 sampai 64 dan kolom 1 sampai 32 lalu disimpan ke z3
28. Membagi citra ke segmen keempat dari baris 33 sampai 64 dan kolom 33 sampai 64 lalu disimpan ke z4
29. Menyediakan keluaran berupa citra baru z1, z2, z3, dan z4
30. Selesai
31. Algoritma hitungp
32. Mulai
33. Mendapatkan jumlah baris dari matriks citra
34. Mendapatkan jumlah kolom dari matriks citra
35. Membuat jh dan menginisialisasi jh dengan 0
36. Untuk i=1 sampai kolom lakukan langkah h, jika sudah memenuhi sampai kolom lakukan langkah h
37. Untuk j=1 sampai baris lakukan langkah g, jika sudah memenuhi sampai baris lakukan langkah d
38. Jika new(i,j) sama dengan 1 tambahkan nili jh dengan 1
39. Menyediakan keluaran nilai jh
40. Selesai
41. Algoritma markD
42. Mulai
43. Mendapatkan jumlah baris dan kolom dari matriks citra
44. Membuat vert dan menginisialisasi vert dengan 0
45. Membuat horz dan menginisialisasi horz dengan 0
46. Membuat dig1 dan menginisialisasi dig1 dengan 0
47. Membuat dig2 dan menginisialisasi dig2 dengan 0
48. Untuk i=1 sampai kolom lakukan langkah h, jika sudah mmenuhi lakukan sampai kolom lakukan langkah selesai
49. Jika new(i,j) sama dengan 0 lakukan langkah i,k,m,o
50. Jika new(i,j) + new(i-1,j) + new(i+1,j) sama dengan 0 lakukan langkah j
51. Jika new(i-1,j-1) + new(i-1,j+1) + new(i,j-1) + new(i,j+1) + new(i+1, j) + new(i+1, j+1) sama dengan 6, maka tamabahkan vert = vert+1
52. Jika new(i,j) + new(i, j-1) + new(i,j+1) sama dengan 0 lakukan langkah l
53. Jika new(i-1,j-1) + new(i-1,j) + new(i-1,j+1) + new(i+1,j-1) + new(i+1, j) + new(i+1, j+1) sama dengan 6, maka tamabahkan horz = horz+1
54. Jika new(i,j) + new(i-1, j-1) + new(i+1,j+1) sama dengan 0 lakukan langkah m
55. Jika new(i-1,j) + new(i-1,j+1) + new(i,j-1) + new(i, j+1) + new(i+1, j-1) + new(i+1,j) sama dengan 6, maka tamabahkan dig1 = dig1+1
56. Jika new(i-1,j-1) + new(i-1, j+1) + new(i+1,j-1) sama dengan 0 lakukan langkah p
57. Jika new(i-1,j-1) + new(i-1,j) + new(i,j-1) + new(i, j+1) + new(i+1, j) + new(i+1,j+1) sama dengan 6, maka tamabahkan dig2 = dig2+1
58. Selesai
59. Algoritma load\_dataaa
60. Mulai
61. Membaca data ciri yang sudah siap dengan fungsi load
62. Menetapkan kolom mana saja yang menjadi ciri
63. Mernomalisasi data ciri dengan fungsi mapminmax
64. Selesai
65. Algoritma target\_data
66. Mulai
67. Membuat target l1 sampai dengan l14 dengan matriks 1x14 dan setiap matriksnya diulangan sebanyak 43 data sesuai dengan jumlah data perhurufnya dengan fungsi repmat dan kemudian ditranspose menjadi 14x1, jadi setiap l mempunyai matriks berukuran 14x43
68. Memasukkan matriks l1 sampai l14 ke matriks label
69. Selesai
70. Algoritma BagiTarget
71. Mulai
72. Mengambil banyak baris dan kolom pada target O (baris) dan N (kolom)
73. Membuat variabel labelData dengan membaca variabel target dari script target data yang telah dikonversi menjadi indeks dengan menggunakan fungsi vec2ind
74. Menginisialisasi id1 sampai id14 dengan niali baru yaitu nilai indeks. Id1 =1, id2 =2, id3 =3, id4 =4, id5 =5, id6 =6, id7 =7, id8 =8, id9 =9, id10 =10, id11 =11, id12 =12, id13 =13, id14 =14. Hal tersebut didapatkan dengan menjalankan id1=find(labelData==1)
75. Menginisialisasi variabel ind0 = 1:N
76. Selesai
77. Algoritma trainn
78. Mulai
79. Insialisasi variabel M dengan nilai 15 (banyaknya data testing)
80. Menginisialisasi control random number dengan nilai 0
81. Membuat struktur jaringan yang akan digunakan untuk perhitungan
82. Menginisialisasi banyak iterasi yang akan dilakukan saat jaringa dilakukan dijalankan
83. Menginisialisasi variabel net dengan jaringan yang telah dibuat tadi
84. Membagi data testing dan data training
85. Membuat array trnInd dengan indeks dari kolom terakhir tstInd+1 sampai N/14
86. Membuat array tstInd dengan indeks dari 1 sampai M
87. Menginisialisasi trainInd pada net dengan trnInd
88. Menginisialisasi testInd pada net dengan tstInd
89. Menjalankan jaringan JST dengan masukkan data ciri yang telah dinormalisasi dan targetnya
90. Selesai
91. Algoritma testing\_ akurasi
92. Mulai
93. Menjalankan jaringan untuk menguji data testing dengan memasukkan datanormal dari semua baris dan kolom tr.testInd dan dimasukkan ke dalam variabel ytst
94. Mengambil semua label dari semua baris dan kolom tr.testInd dan dimasukkan ke dalam variabel ttst
95. Membuat confusion matriks dengan variabel ytst dan ttst
96. Menghitung jumlah data diagonal pada confusion matriks untuk mengetahui berapa banyak data yang benar
97. Mengitung jumlah banyak data pada confusion matriks
98. Menghitung akurasi dengan rumus akurasi = (benar/jumlah total data)\*100
99. Selesai
100. Algoritma test\_data\_tunggal
101. Mulai
102. Melakukan pengambilan keputusan jika
103. Jika file\_data = ‘s1.data’ maka ciri = ciri1
104. Jika file\_data = ‘s2.data’ maka ciri = ciri2
105. Jika file\_data = ‘s3.data’ maka ciri = ciri3
106. Jika file\_data = ‘s4.data’ maka ciri = ciri4
107. Jika file\_data = ‘s5.data’ maka ciri = ciri5
108. Jika file\_data = ‘s6.data’ maka ciri = ciri6
109. Mengisi variabel read1 dengan nilai matriks dari varibel ciri
110. Menormalisasi variabel read1 kemudian di transpose, lalu hasilnya disimpan di variabel datanorm
111. Menjalankan jaringan dengan masukan datanorm lalu hasilnya disimpan di ytst1
112. Mengubah ytst1 dari vector menjadi indeks lalu hasilnya disimpan di variabel hasil
113. Melakukan pengambilan keputusan jika
114. Jika hasil = 1 maka hasil\_huruf = G
115. Jika hasil = 2 maka hasil\_huruf = N
116. Jika hasil = 3 maka hasil\_huruf = D
117. Jika hasil = 4 maka hasil\_huruf = R
118. Jika hasil = 5 maka hasil\_huruf = M
119. Jika hasil = 6 maka hasil\_huruf = B
120. Jika hasil = 7 maka hasil\_huruf = S
121. Jika hasil = 8 maka hasil\_huruf = NG
122. Jika hasil = 9 maka hasil\_huruf = J
123. Jika hasil = 10 maka hasil\_huruf = CH
124. Jika hasil = 11 maka hasil\_huruf = K
125. Jika hasil = 12 maka hasil\_huruf = T
126. Jika hasil = 13 maka hasil\_huruf = P
127. Jika hasil = 14 maka hasil\_huruf = H
128. Selesai
129. Algoritma tampilan menu utama
130. Menampilkan logo Universitas Sanata Dharma pada axes5
131. Menampilkan judul dari penelitian pada GUI bagian atas tengah
132. Menampilkan nama peneliti pada dibawah judul penelitian
133. Menampilkan GUI yang dibagi menjadi 2 panel yaitu Pelatihan Data dan Uji Data Tunggal
134. Selesai
135. Algoritma Panel Pelatihan Data
136. Memanggil script load data pada tombol Load Data Ciri
137. Menampikan data ciri yang sudah di load pada table
138. Pada panel Training Data
139. Memasukkan jumlah hidden layer 1 pada field Hidden Layer 1
140. Memasukkan jumlah hidden layer 2 pada field Hidden Layer 2
141. Melakukan pelatihan data dengan menjalankan script trainn pada tombol Lakukan Pelatihan
142. Menampilkan hasil dari script testing\_akurasi pada field Akurasi%
143. Selesai
144. Algoritma Panel Uji Data Tunggal
145. Membaca data huruf yang akan di uji pada tombol pilih Huruf
146. Menampilkan lokasi data huruf yang akan diuji pada field Nama Field
147. Menampilkan gambar yang akan di uji pada axes4
148. Memanggil script test\_data\_tunggal pada tombol Kenali Huruf
149. Menampilkan hasil pengenalan tulisan tangan huruf Hangul pada field Dikenal sebagai Huruf
150. Selesai

## 3.6 Perancangan Antarmuka Sistem

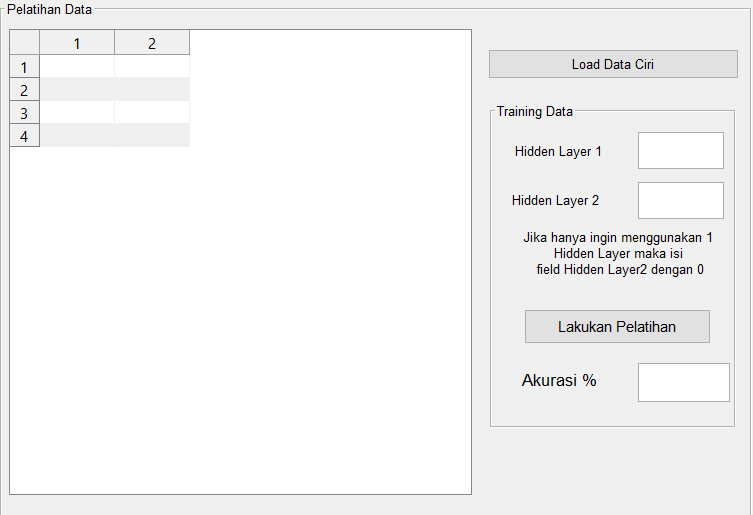
Pada penelitian ini dibuat juga *Graphic User Interface* yang dapat memudahkan pengguna lebih mudah melihat jalannya program dari alat uji. Rancangan dari GUI dapat dilihat pada gambar dibawah

1. Tampilan Utama



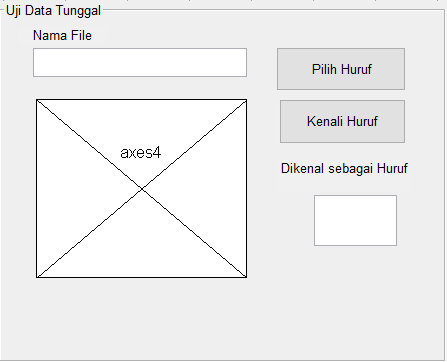
Gambar 3. 13 Contoh Tampilan Utama Alat Uji

1. Panel Pelatihan Data



Gambar 3. 14 Contoh Tampilan Panel Pelatihan Data

1. Panel Uji Data Tunggal



Gambar 3. 15 Contoh Tampilan Panel Uji Data Tunggal

## 3.7 Skenario Pengujian

Langkah pertama dari pengujian ini adalah membagi data training dan testing. Jumlah total data yang dipakai adalah 602 huruf dimana masing-masing data huruf ada 43 huruf. Pembagian data training adalah 2/3 dari seluruh data yang ada, sehingga data yang dipakai untuk pelatihan ada 28 data per huruf. Dan jumlah data yang dipakai untuk testing adala 1/3 dari seluruh data yang ada, sehingga data yang dipakai untuk testing ada 15 data.

Langkah selanjutnya adalah mengekstrak ciri dari masing-masing huruf yang ada. Jika semua data sudah diekstrak cirinya kemudian disimpan dalam bentuk .data. Dalam mengesktrak ciri data huruf, dilakukan 6 kombinasi dari kedua metode.

Setelah mendapat ciri dari huruf, kemudian membagi target dan mengubahnya menjadi vector target untuk digunakan dalam pelatihan jaringan. Pada pelatihan jaringan, telah disiapkan struktur jaringan mengunakan hidden layer, fungsi aktivasi dan target. Dalam penelitian ini menggunakan 90 hidden layer, fungsi aktivasi logsig, dan targetnya adalah berupa matriks berukuran 14x1.

Setelah proses training selesai maka selanjutnya adalah testing dan akurasi. Proses ini bertujuan untuk mengetahui bagaimana performa dari alat uji yang sudah dibuat. Kemudian akan dilihat apakah akurasi dari pengenalan data testing tersebut sudah memenuhi ≥ 70% atau belum. Apabila hasil akurasi yang dihasilkan belum mencapai 70% maka akan dilakukan pengubahan. Pengubahan yang dapat dilakukan antara lain, mengubah jumlah neuron pada *hidden layer*, mengubah fungsi aktivasi pada lapisan tersembunyi atau lapisan keluaran, mengubah fungsi pelatihan yang dipakai dalam struktur jaringan, mengubah jumlah lapisan tersembunyi dan mengubah banyaknya iterai yang dilakukan ketika menjalankan *backpropagation.* (Nugroho, 2016)

Langkah selanjutnya adalah pengenalan data huruf dengan mencocokan ciri dari data baru dengan data ciri yang sudah ada.

# BAB IV

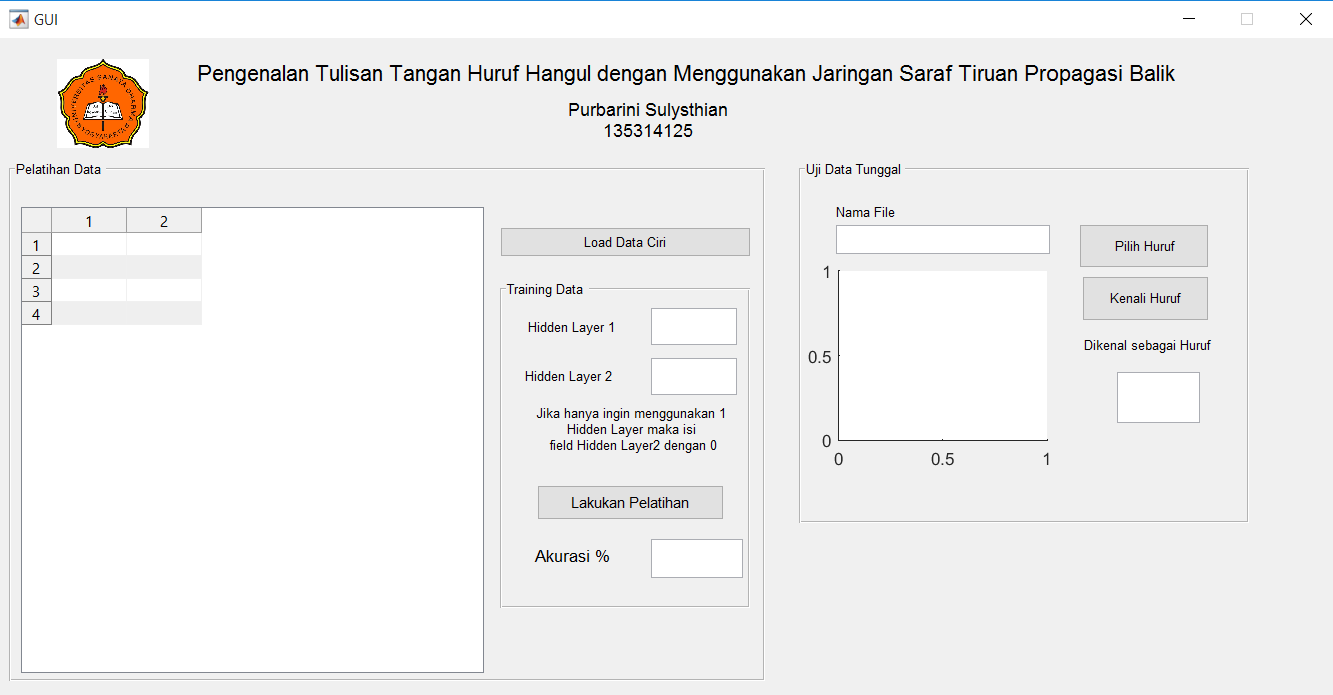
**HASIL UJI DAN ANALISIS**

## 4.1 Contoh Hasil Antarmuka

Desain antarmuka yang sudah dibuat dapat berjalan dengan baik, berikut adalah hasil dari running alat uji.

1. Menu Utama

Gambar dibawah ini menunjukkan tampilan utama dari desain antarmuka yang telah dibuat. Pada menu utama ini terdapat panel Pelatihan Data dan panel Uji Data Tunggal



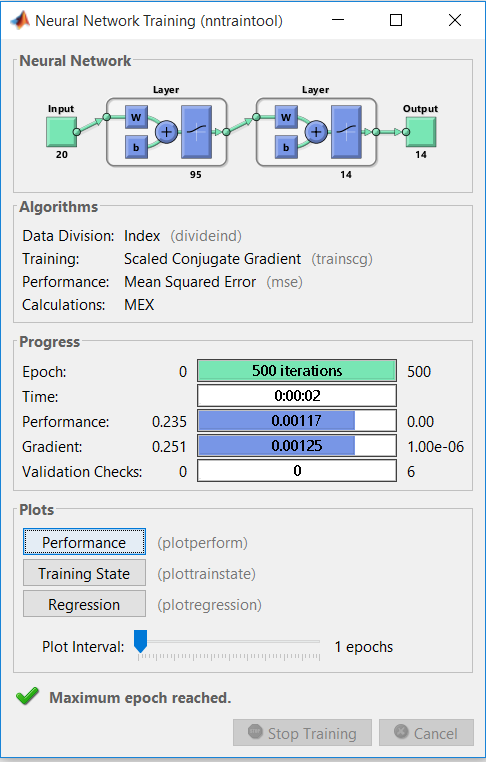
Gambar 4. 1 Antarmuka Menu Utama Alat Uji

1. Panel Pelatihan Data

Gambar dibawah menunjukkan tampilan dar panel Pelatihan Data. Panel ini berfungsi untuk me-*load* data ciri yang sudah ada dan melakukan pelatihan data.



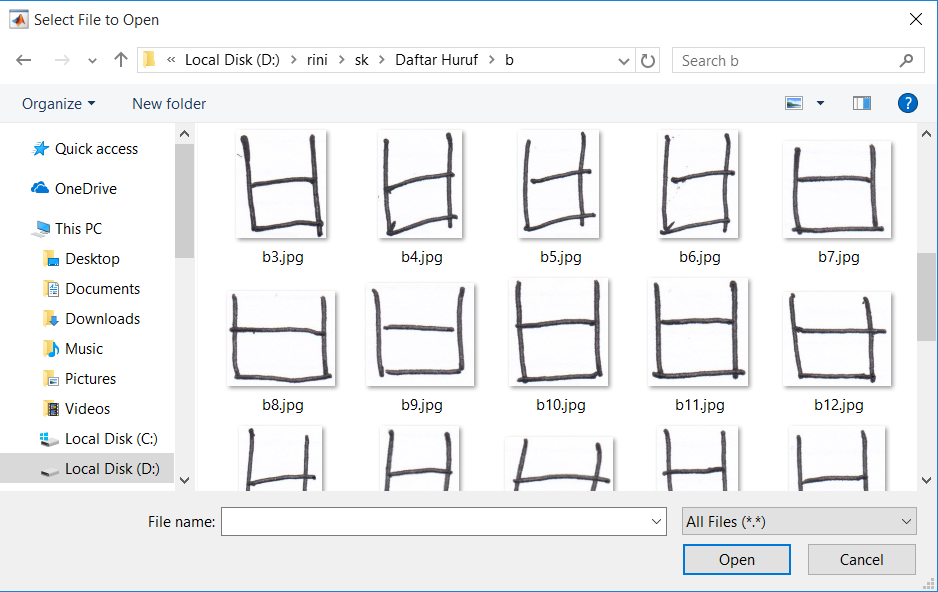
Gambar 4. 2 Antarmuka Panel Pelatihan Data



Gambar 4. 3 Contoh Hasil Pelatihan Data

1. Panel Uji Data Tunggal

Gambar dibawah menunjukkan panel Uji Data Tunggal. Panel ini berfungsi untuk melakukan pengujian data tunggal, proses kerja dari panel ini pertama pilih data yang akan diuji, kemudian tekan tombol Kenal Huruf.



Gambar 4. 4 Contoh Men-load Data



Gambar 4. 5 Antarmuka Panel Uji Data Tunggal



Gambar 4. 6 Contoh Hasil Pengujian Data Tunggal

## 4.2 Desain Jaringan

Pada penelitian ini penguji mendesain jaringan yang terdiri dari layer masukkan, layer tersembunyi, dan layer keluaran. Desain jaringan tersebut adalah sebagai berikut:



Gambar 4. 7 Desain Jaringan

## 4.3 Hasil Uji

Pada penelitian ini, peneliti melakukan dua belas percobaan dengan jumlah hidden layer dan fungsi aktivasi yang berbeda. Untuk setiap percobaan kombinasi ciri yang dipakai ada 6 kombinasi, yaitu:

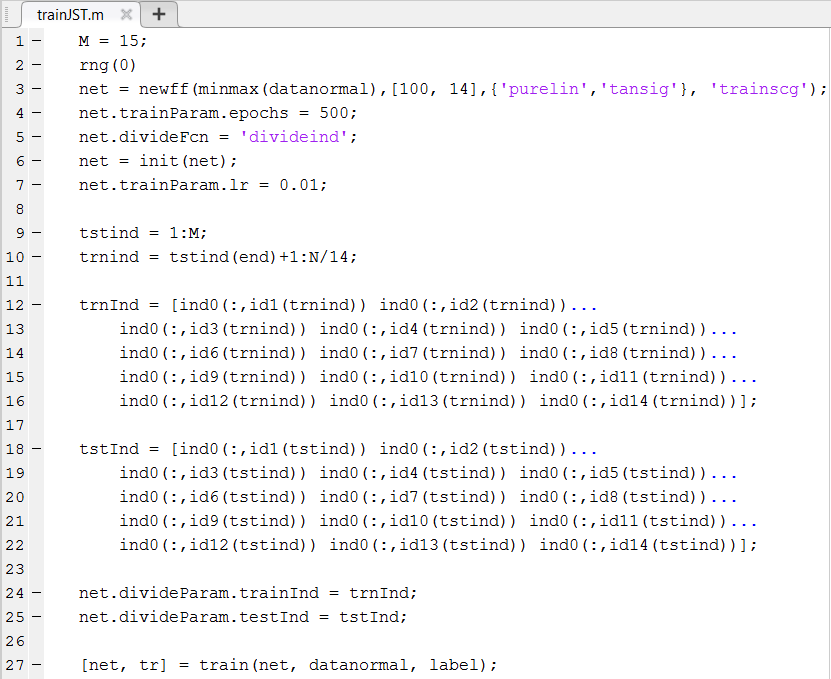
1. Kombinasi ciri 1(ciri dari *Intensity of Character*)
2. Kombinasi ciri 2 (ciri *dari Mark Direction* diagonal 1)
3. Kombinasi ciri 1 dan ciri 2
4. Kombinasi ciri 1 dan ciri 3 (ciri dari *Mark Direction* diagonal 2)
5. Kombinasi ciri 2 dan ciri 3
6. Kombinasi ciri 1, ciri 2, ciri 3, ciri 4 (ciri dari *Mark Direction* horisontal), dan ciri 5 (ciri dari *Mark Direction*)

Adapun hasil percobaan yang telah dilakukan dapat dilihat sebagai berikut**:**

1. Percobaan 1

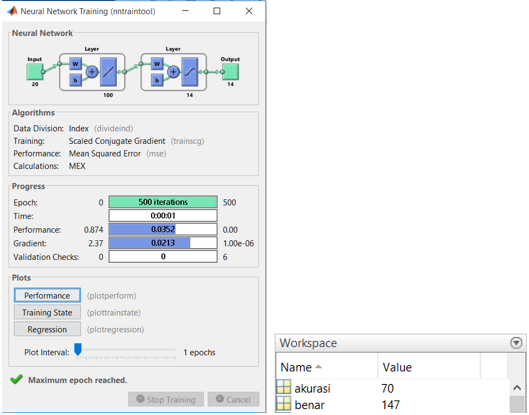
Pada percobaan 1, peneliti menggunakan 1 *hidden layer* dengan jumlah neuron pada *hidden layer* sebanyak 100, fungsi aktivasi pada hidden layer memakai *purelin,* fungsi pelatihan jaringan menggunakan trainscg, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *tansig.* Berikut adalah hasil dari percobaan 1

1. Program jaringan Backpropagation



Gambar 4. 8 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 1

1. Output



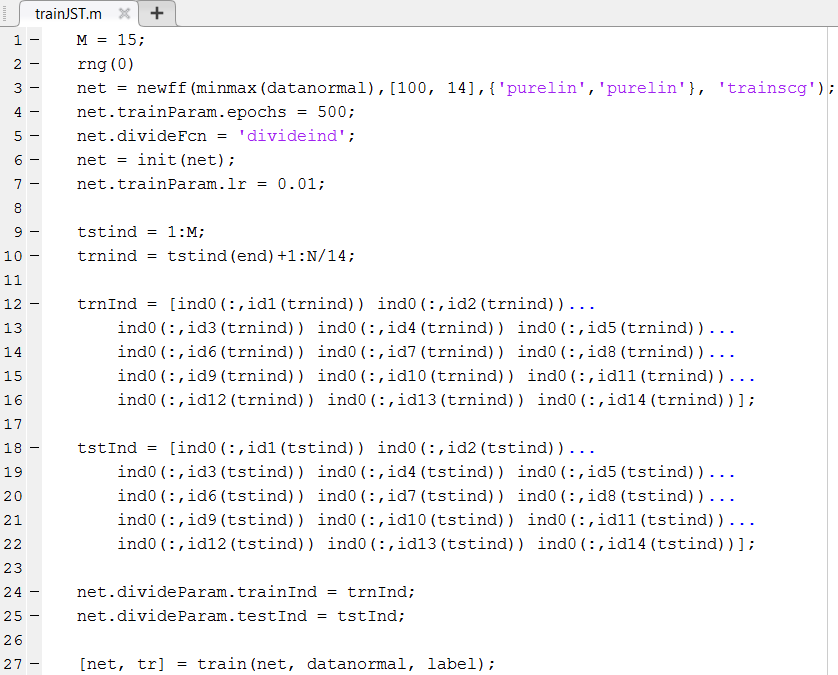
Gambar 4. 9 Hasil Perobaan 1

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 147 data dengan akurasi 70%. Waktu dari pelatihan tersebut 00:01, performanya 0.874 dan gradient sebesar 1.00E-06.

1. Percobaan 2

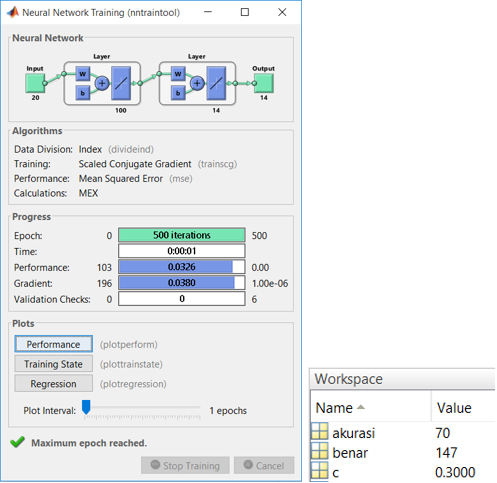
Pada percobaan 2, peneliti menggunakan 1 hidden layer dengan jumlah neuron pada hidden layer sebanyak 100, fungsi aktivasi pada hidden layer memakai *purelin,* fungsi pelatihan jaringan menggunakan trainscg, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *purelin.* Berikut adalah hasil dari percobaan 2

1. Program jaringan Backpropagation



Gambar 4. 10 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 2

1. Output



Gambar 4. 11 Hasil Percobaan 2

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 147 data dengan akurasi 70%. Waktu dari pelatihan tersebut 00:01, performanya 103 dan gradient sebesar 1.00E-6.

1. Percobaan 3

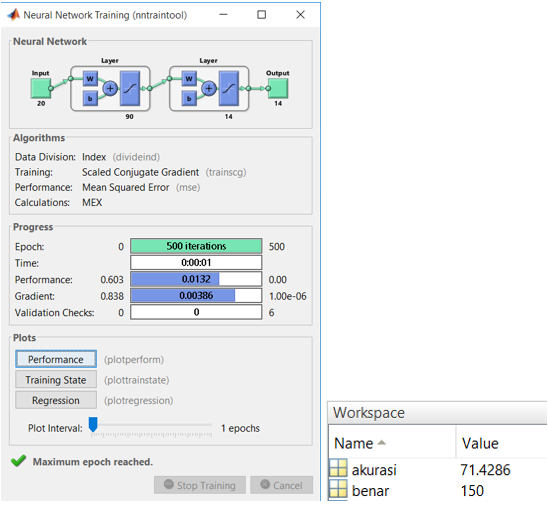
Pada percobaan 3, peneliti menggunakan 1 hidden layer dengan jumlah neuron pada hidden layer sebanyak 90, fungsi aktivasi pada hidden layer memakai *tansig,* fungsi pelatihan jaringan menggunakan trainscg, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *tansig.* Berikut adalah hasil dari percobaan 3

1. Program jaringan Backpropagation



Gambar 4. 12 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 3

1. Output



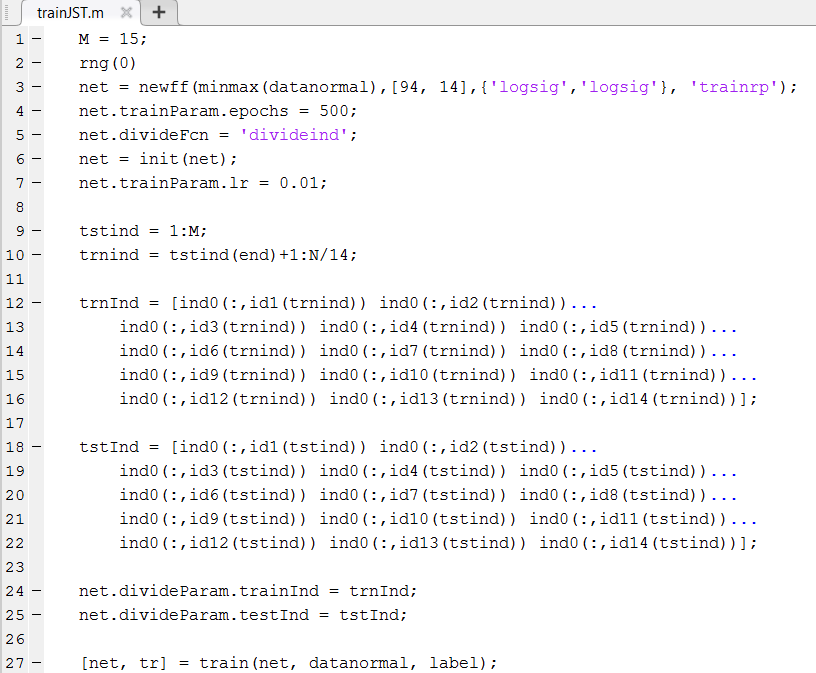
Gambar 4. 13 Hasil Percobaan 3

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 150 data dengan akurasi 71,4286%. Waktu dari pelatihan tersebut 00:01, performanya 0.603 dan gradient sebesar 1.00E-06.

1. Percobaan 4

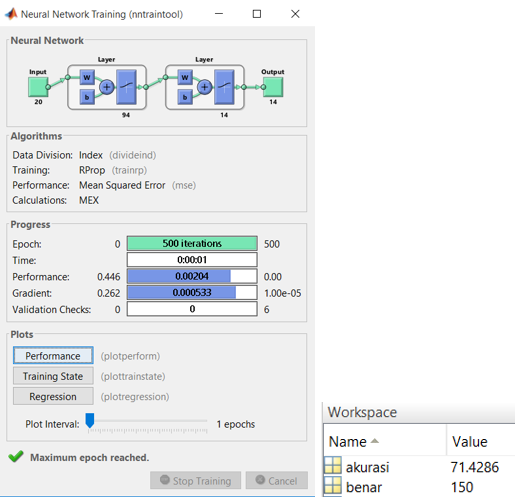
Pada percobaan 4, peneliti menggunakan 1 hidden layer dengan jumlah neuron pada hidden layer sebanyak 94, fungsi aktivasi pada hidden layer memakai *logsig,* fungsi pelatihan jaringan menggunakan trainrp, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *logsig.* Berikut adalah hasil dari percobaan 4

1. Program jaringan Backpropagation



Gambar 4. 14 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 4

1. Output



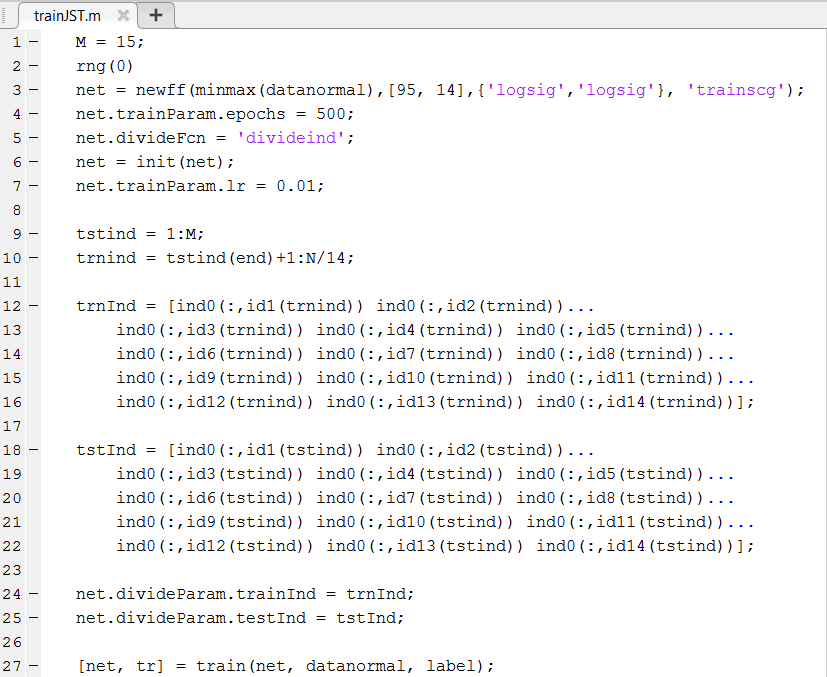
Gambar 4. 15 Hasil Percobaan 4

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 150 data dengan akurasi 71,4286%. Waktu dari pelatihan tersebut 00:01, performanya 0.446 dan gradient sebesar 1.00E-05.

1. Percobaan 5

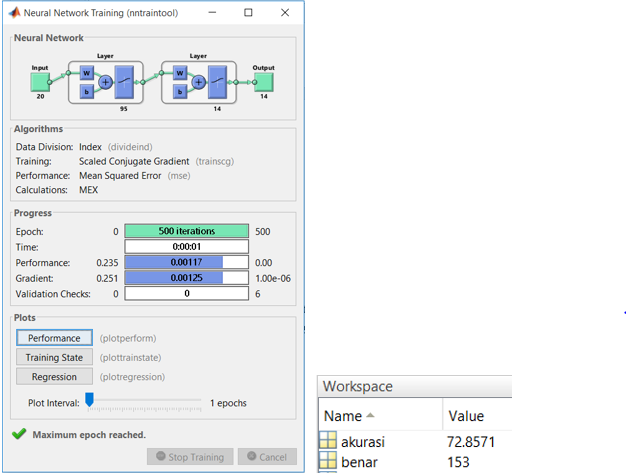
Pada percobaan 5, peneliti menggunakan 1 hidden layer dengan jumlah neuron pada hidden layer sebanyak 95, fungsi aktivasi pada hidden layer memakai *logsig,* fungsi pelatihan jaringan menggunakan trainscg, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *logsig.* Berikut adalah hasil dari percobaan 5

1. Program jaringan Backpropagation



Gambar 4. 16 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 5

1. Output



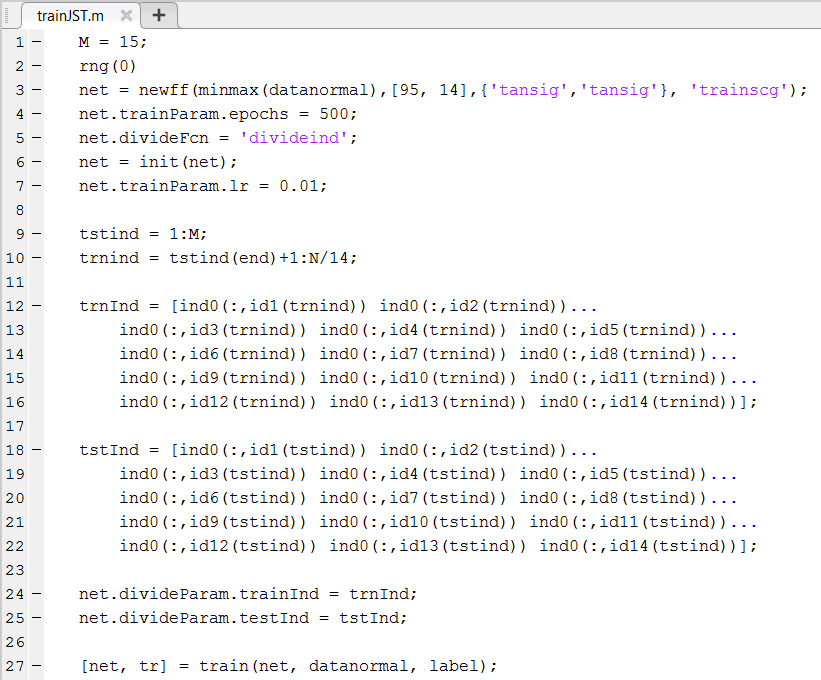
Gambar 4. 17 Hasil Percobaan 5

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 153 data dengan akurasi 72,8571%. Waktu dari pelatihan tersebut 00:01, performanya 0.235 dan gradient sebesar 1.00E-06.

1. Percobaan 6

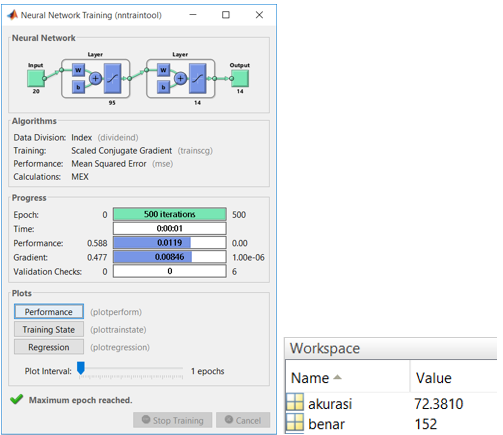
Pada percobaan 6, peneliti menggunakan 1 hidden layer dengan jumlah neuron pada hidden layer sebanyak 95, fungsi aktivasi pada hidden layer memakai *logsig,* fungsi pelatihan jaringan menggunakan trainscg, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *logsig.* Berikut adalah hasil dari percobaan 6

1. Program jaringan Backpropagation



Gambar 4. 18 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 6

1. Output



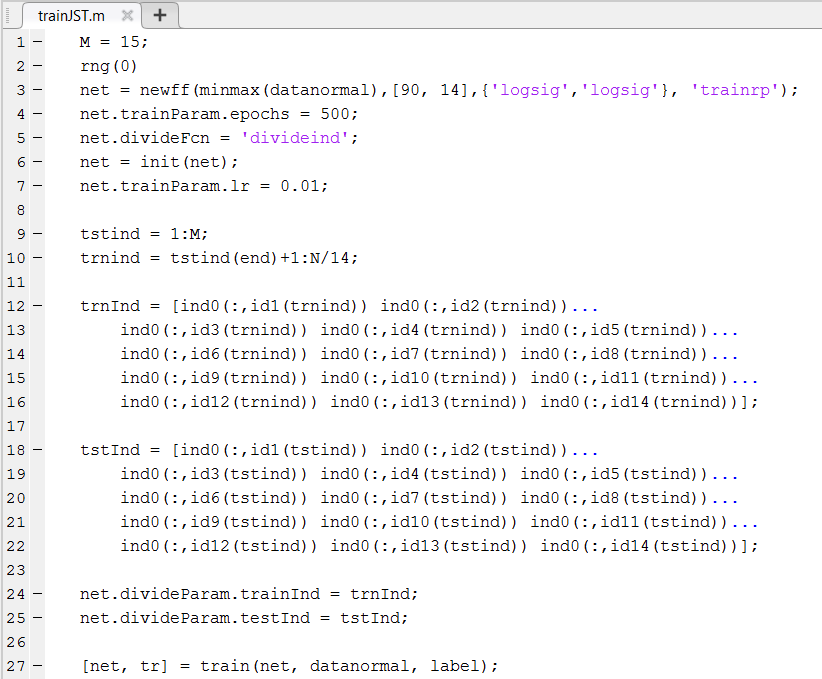
Gambar 4. 19 Hasil Percobaan 6

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 152 data dengan akurasi 72,3810%. Waktu dari pelatihan tersebut 00:01, performanya 0.588 dan gradient sebesar 1.00E-06.

1. Percobaan 7

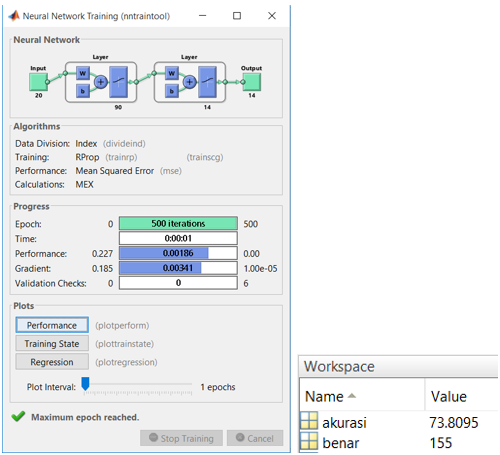
Pada percobaan 7, peneliti menggunakan 1 hidden layer dengan jumlah neuron pada hidden layer sebanyak 90, fungsi aktivasi pada hidden layer memakai *logsig,* fungsi pelatihan jaringan menggunakan trainrp, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *logsig.* Berikut adalah hasil dari percobaan 7

1. Program jaringan Backpropagation



Gambar 4. 20 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 7

1. Output



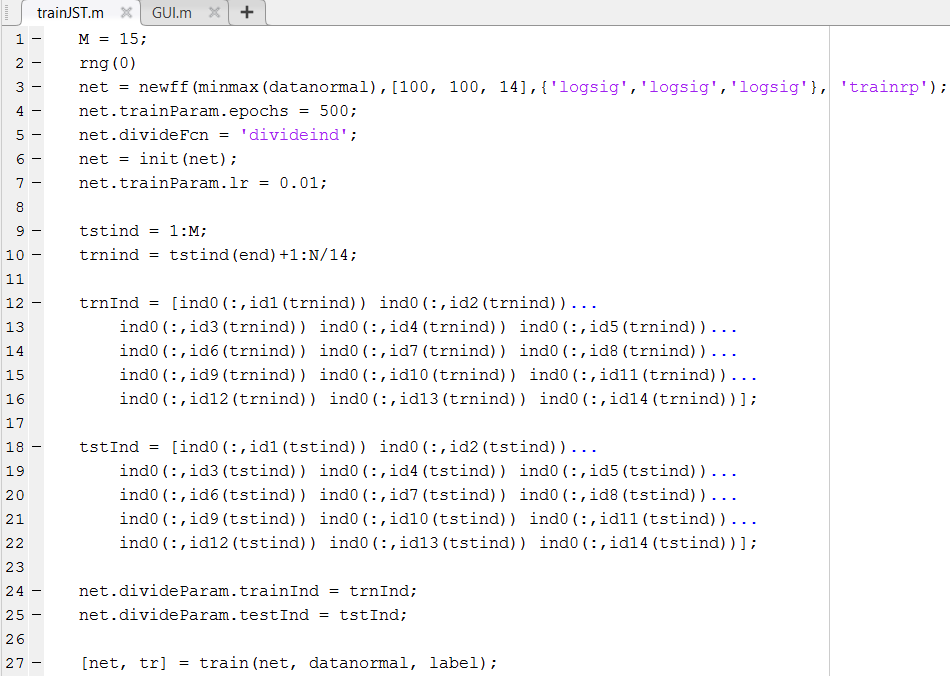
Gambar 4. 21 Hasil Percobaan 7

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 155 data dengan akurasi 73,8095%. Waktu dari pelatihan tersebut 00:01, performanya 0.277 dan gradient sebesar 1.00E-05.

1. Percobaan 8

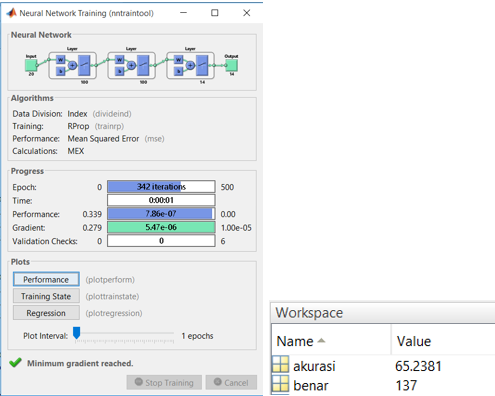
Pada percobaan 8, peneliti menggunakan 2 hidden layer dengan jumlah neuron pada hidden layer 1 sebanyak 100, hidden layer 2 sebanyak 100, fungsi aktivasi pada hidden layer 1 memakai *logsig,* fungsi aktivasi pada hidden layer 2 memakai *logsig,* fungsi pelatihan jaringan menggunakan trainrp, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *logsig.* Berikut adalah hasil dari percobaan 8

1. Program jaringan Backpropagation



Gambar 4. 22 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 8

1. Output



Gambar 4. 23 Hasil dari Percobaan 8

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 137 data dengan akurasi 65,2381%. Waktu dari pelatihan tersebut 00:01, performanya 0.339 dan gradient sebesar 1.00E-05.

1. Percobaan 9

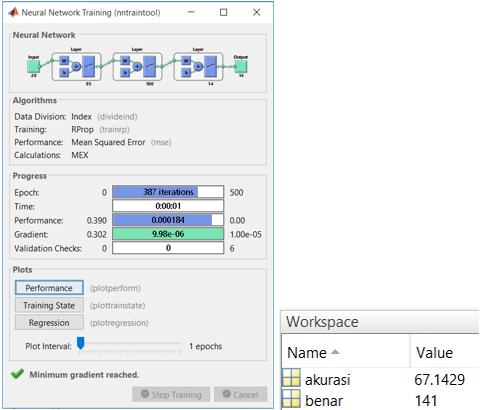
Pada percobaan 9, peneliti menggunakan 2 hidden layer dengan jumlah neuron pada hidden layer 1 sebanyak 95, hidden layer 2 sebanyak 100, fungsi aktivasi pada hidden layer 1 memakai *logsig,* fungsi aktivasi pada hidden layer 2 memakai *logsig,* fungsi pelatihan jaringan menggunakan trainrp, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *logsig.* Berikut adalah hasil dari percobaan 9

1. Program jaringan Backpropagation



Gambar 4. 24 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 9

1. Output



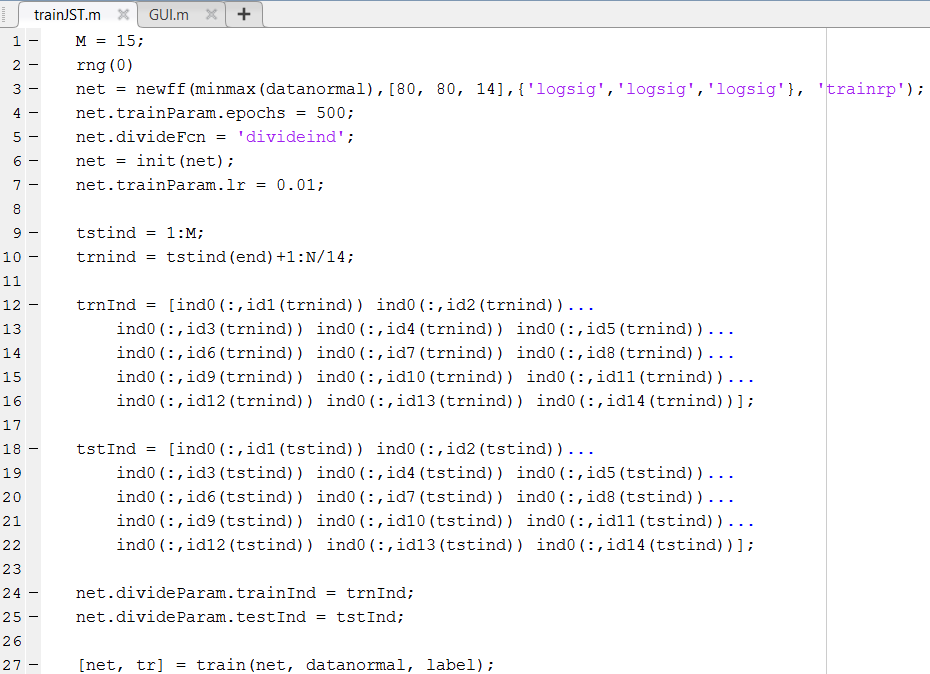
Gambar 4. 25 Hasil dari Percobaan 9

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 141 data dengan akurasi 67,1429%. Waktu dari pelatihan tersebut 00:01, performanya 0.390 dan gradient sebesar 1.00E-05.

1. Percobaan 10

Pada percobaan 10, peneliti menggunakan 2 hidden layer dengan jumlah neuron pada hidden layer 1 sebanyak 80, hidden layer 2 sebanyak 80, fungsi aktivasi pada hidden layer 1 memakai *logsig,* fungsi aktivasi pada hidden layer 2 memakai *logsig,* fungsi pelatihan jaringan menggunakan trainrp, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *logsig.* Berikut adalah hasil dari percobaan 10

1. Program jaringan Backpropagation



Gambar 4. 26 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 10

1. Output



Gambar 4. 27 Hasil dari Percobaan 10

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 144 data dengan akurasi 68,5741%. Waktu dari pelatihan tersebut 00:01, performanya 0.320 dan gradient sebesar 1.00E-05.

1. Percobaan 11

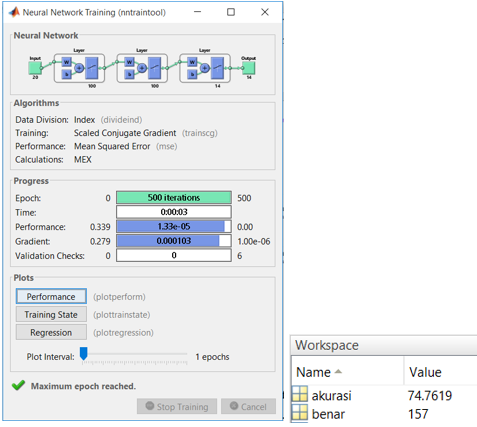
Pada percobaan 11, peneliti menggunakan 2 hidden layer dengan jumlah neuron pada hidden layer 1 sebanyak 100, hidden layer 2 sebanyak 100, fungsi aktivasi pada hidden layer 1 memakai *logsig,* fungsi aktivasi pada hidden layer 2 memakai *logsig,* fungsi pelatihan jaringan menggunakan trainscg, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *logsig.* Berikut adalah hasil dari percobaan 11

1. Program jaringan Backpropagation



Gambar 4. 28 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 11

1. Output



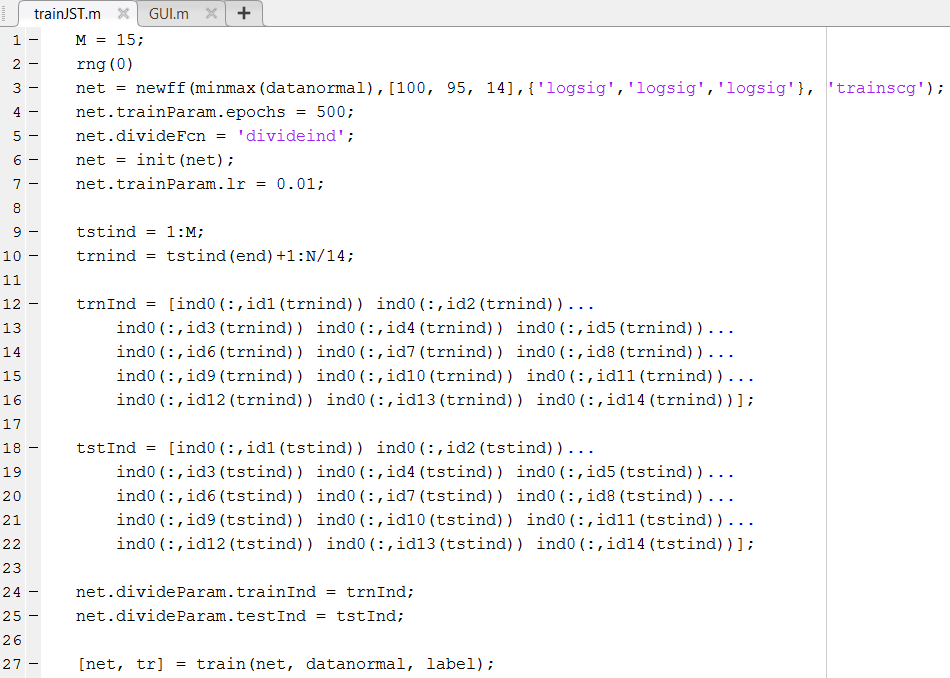
Gambar 4. 29 Hasil Percobaan 11

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 157 data dengan akurasi 74,7619%. Waktu dari pelatihan tersebut 00:03, performanya 0.339 dan gradient sebesar 1.00E-06.

1. Percobaan 12

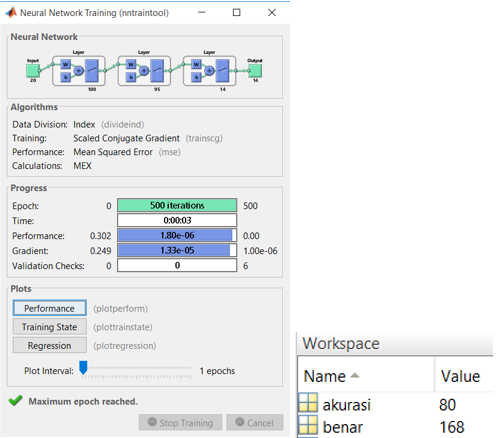
Pada percobaan 12, peneliti menggunakan 2 hidden layer dengan jumlah neuron pada hidden layer 1 sebanyak 100, hidden layer 2 sebanyak 85, fungsi aktivasi pada hidden layer 1 memakai *logsig,* fungsi aktivasi pada hidden layer 2 memakai *logsig,* fungsi pelatihan jaringan menggunakan trainscg, dan jumlah iterasi sebanyak 500 kali. Pada lapisan keluaran terdapat 14 unit neuron dengan fungsi aktivasi *logsig.* Berikut adalah hasil dari percobaan 12

1. Program jaringan Backpropagation



Gambar 4. 30 Program Pelatihan Jaringan Propagasi Balik pada Percobaan 12

1. Output

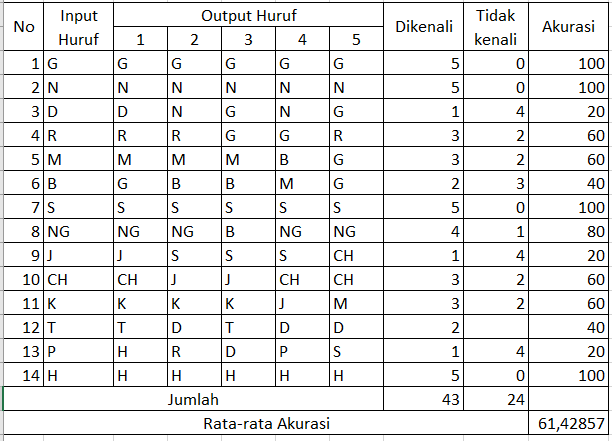


Gambar 4. 31 Hasil dari Percobaan 12

Dapat dilihat dari gambar diatas bahwa dari 620 data huruf, data benar yang dapat terbaca ada 168 data dengan akurasi 80%. Waktu dari pelatihan tersebut 00:03, performanya 0.302 dan gradient sebesar 1.00E-06.

## 4.4 Testing Data Tunggal

Pada penelitian ini dilakukan juga testing data tunggal dengan menggunakan 5 data pada setiap hurufnya, sehingga ada 70 data huruf yang dipakai untuk testing. Karena hasil akurasi terbaik didapatkan pada percobaan 12, maka pada pelatihan data tunggal ini akan menggunakan pola jaringan dan kombinasi ciri yang sama seperti pada percobaan 12. Berikut adalah tabel hasil testing data tunggal:



Tabel 4. 1 Tabel Hasil Testing Data Tunggal

Hasil percobaan pengujian data tunggal diata menunjukkan bahwa hasil yang didapat sangant beragam, total data yang dapat dikenali dengan baik 43 huruf sedangkan yang tidak dapat dikenali ada 24 huruf, dan rata-rata akurasi yang didapat sebesar 61,42857%. Hal ini disebabkan data yang ditulis kurang baik dan kurang jelas, sehingga saat diubah menjadi citra hitam putih bagian tidak jelasnya tidak bisa dikenali.

## 4.5 Analisis Data

Setelah melakukan 12 kali percobaan dengan mengubah banyaknya jumlah *hidden* layer, jumlah neuron pada *hidden* layer, fungsi aktivasi pada *hidden* layer dan fungsi aktivasi pada jaringan terlihat adanya perubahan pada akurasi alat uji.

Hasil terbaik dapat dilihat pada percobaaan 12, dengan banyak data benar/data yang dapat dibaca ada 168 data huruf dengan akurasi mencapai 80%. Dapat dilihat juga data huruf paling sedikit yang dapat dibaca ada pada percobaan 8, dimana akurasi yang didapat adalah 65,2381%. Dari hasil tersebut dapat dikatakan bahwa alat uji yang telah dibuat dapat mengenali tulisan tangan huruf Hangul.

Percobaan mengubah banyaknya jumlah *hidden* layer, jumlah neuron pada *hidden* layer, fungsi aktivasi pada *hidden* layer dan fungsi aktivasi pada jaringan ternyata juga berpengaruh dalam meningkatkan dan mengurangi tingkat keakurasian dari alat uji.

Selain itu juga telah dilakukan pengujian data tunggal untuk melihat hasil dari pelatihan jaringan. Pengujian data tunggal ini menggunakan 70 data huruf. Dari 70 data, data yang dikenali dengan baik ada 43 huruf dengan rata-rata akurasi yang dihasilkan mencapai 61,42857%. Akurasi yang rendah disebabkan data yang digunakan kurang jelas dan kurang baik dalam penulisannya.

Dari hasil yang diperoleh pada penelitian ini tentang pengenalan tulisan tangan huruf Hangul dengan menggunakan jaringan saraf tiruan dengan menggunakan jaringan saraf tiruan propagasi balik diketahui bahwa:

1. Pada penelitian ini terdapat 168 tulisan tangan huruf Hangul yang dapat terbaca, maka dapat dikatan bahwa algoritma *backpropagation* dapat mengenali tulisan tangan huruf Hangul.
2. Prosentasi akurasi dari beberapa percobaan dan fungsi yang telah dilakukan pada penelitian ini diperoleh akurasi terbaik sebesar 80%.

Maka penelitian ini dapat dikatakan berhasil karena sudah memenuhi tujuan penelitian dengan akurasi sudah melebihi 70% prosentase keberhasilan.

# BAB V

**PENUTUP**

## 5.1 Kesimpulan

Dari hasil penelitian yang telah dikakukan disimpulkan jika alat uji yang dibuat dengan menggunakan algoritma propagasi balik dapat digunakan untuk mengenali tulisan tangan huruf Hangul. Hal ini dapat dilihat dengan adanya beberapa huruf yang terbaca oleh alat uji yang menggunakan algotritma propagasi balik. Tingkat akurasi tertinggi didapat pada percobaan 12 sdengan akurasi sebesar 80% menggunakan kombinasi ciri 6 yang merupakan matriks kombinasi dari ciri *Intensity of Character*, *Mark Direction* horz, *Mark Direction* vert, *Mark Direction* dig1, *Mark Direction* dan dig2. Pola jaringan yang digunakan adalah jaringan dengan 2 hidden layer dan 1 lapisan keluaran, dengan jumlah unit neuron pada hidden layer 1 dan hidden layer 2 100, 95, dan pada lapisan keluaran jumlah unit neuronnya ada 14. Fungsi aktivasi yang digunakan pada hidden layer 1 dan hidden layer 2 adalah *logsig*, *logsig*, pada lapisan keluaran *logsig*, dan pada pelatihan jaringan menggunakan fungsi aktivasi *trainscg*. Iterasi yang terjasi sebesar 500 kali dengan waktu perhitungan 00:03 dan MSE sebesar 1.00E-06.

## 5.2 Saran

Dari hasil penelitian pengenalan tulisan tangan huruf Hangul ini, ada beberapa saran yang untuk memperbaiki kekurangan pada penelitian tersebut, antara lain:

1. Mencari data tulisan tangan huruf Hangul yang lebih banyak.
2. Tentukan Batasan-batasan dalam responden menulis tulisan tangan, jangan campurkan antara tulisan tangan huruf Hangul normal dan tulisan tangan huruf Hangul yang digayakan.
3. Mencari dan menggunakan metode ekstrasi ciri yang berbeda.
4. Mengubah susunan jaringan dengan menambah jumlah hidden layer, jumlah neuron pada hidden layer, fungsi aktivasi pada hidden layer, dan fungsi aktivasi pada jaringan.

# DAFTAR PUSTAKA

Keechul and Hang Joon Kim, 1999, “*On-line recognition of cursive Korean characters using graph representation*”. Taegu, South Korea

Handoyo, Erico Darmawan, Lydia Wiguna Susanto, 2011. *Penerapan Jaringan Syaraf Tiruan metode Propagasi Balik Dalam Pengenalan Tulisan Tangan Huruf Jepang Jenis Hiragana dan Katakana*. Universitas Kristen Manarata. Manarata University Pres : Bandung. ISSN 0216-4280

Nugroho, Nicolaus Euclides Wahyu, 2016. “*Pengenalan Pola huruf Jepang Hiragana Menggunakan Algoritma Backpropagation”.* Universitas Sanata Dharma, Yogyakarta.

Handoyo, Kasih, 2017. “ *Transliterasi Nama Jalan Beraksara Jawa*”. Universitas Sanata Dharma, Yogyakarta.

Gonzales, Rafael C. & Woods, Richard E., 2008. *Digital Image Processing*(3rd ed.). United States of America: Pearson Prentice Hall.

Surinta, Olarik, 2010. *“Overview of Handwritten Tahi Character Recogntion”.* Diakses dari http://www.ai.rug.nl/~molarik/APSMeeting/09-07-2010% 20Overview%20of%20Handwritten%20Thai%20Character%20Recognition.pdf pada tanggal 29 september 2017.

Siska Fathia, 2013.*”Penerapan Jaringan Saraf Tiruan Dalam Pengenalan Tulisan Tangan Huruf Korea (Hangul) Menggunakan Metode Propagasi Balik”.* Universitas Dian Nuswantoro, Semarang.

Korea, Monthly Statistics. [*http://kto.visitkorea.or.kr*](http://kto.visitkorea.or.kr)*.* Diakses pada tanggal 11 Oktober 2016

Hwa, Ahn Kyung, Cho Hyun Yong, Rura Ni Adinda, Agung Suray, Florian Hutagalung. 2008*.”Ayo Belajar Bahasa Korea dengan Bahasa Korea Terpadu Untuk Orang Indonesia”.* Korea Language Department, Korea Foundation. Secho-gu, Seoul.

Kristanto, Andri, 2004. *“Jaringan Syaraf Tiruan, Konsep Dasar, Algoritma dan Aplikasi”*. Gava Media. Yogyakarta.

Sibagariang, Swono, 2016. *”Klasifikasi Citra Mammogram dengan Metode Ekstraksi Ciri Zoning Menggunakan Ssvm”.* Universitas Sumatera Utara, Medan. Diakses dari [http://repository.usu.ac.id/xmlui /handle/123456789/57829](http://repository.usu.ac.id/xmlui%20/handle/123456789/57829) pada tanggal 5 Desember 2016.

Sinaga, Devi Yenni, 2013. *”Pengenalan Pola Plat Nomor Kendaraan Bermotor Dengan Menggunakan PCA dan Metode Propagasi Balik”.* Universitas Sanata Dharma, Yogyakarta.

# LAMPIRAN

1. Form Pengenalan Tulisan Tangan Huruf Hangul

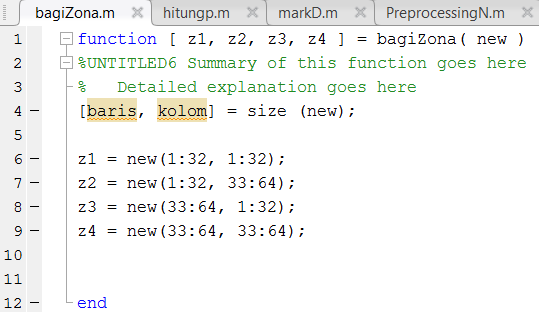
Form Tugas Akhir

“Pengenalan Tulisan Tangan Huruf Hangul Dengan Menggunakan Metode Jaringan Saraf Tiruan Propagasi Balik”

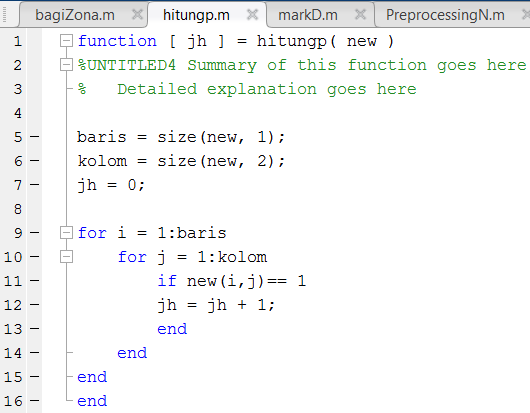
|  |  |  |
| --- | --- | --- |
| Huruf | Lembar Kerja Untuk Menulis Huruf Hangul | |
| Lembar Kerja 1 | Lembar Kerja 2 |
| **ㄱ** |  |  |
| **ㄴ** |  |  |
| **ㄷ** |  |  |
| **ㄹ** |  |  |
| **ㅁ** |  |  |
| **ㅂ** |  |  |
| **ㅅ** |  |  |
| **ㅇ** |  |  |
| **ㅈ** |  |  |
| **ㅊ** |  |  |
| **ㅋ** |  |  |
| **ㅌ** |  |  |
| **ㅍ** |  |  |
| **ㅎ** |  |  |

1. Source Code Program

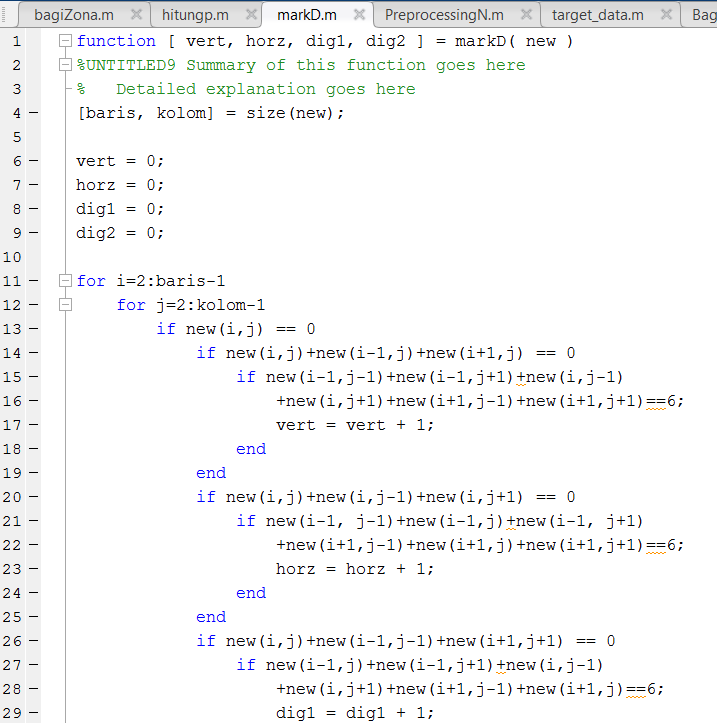
* Bagi Segmen

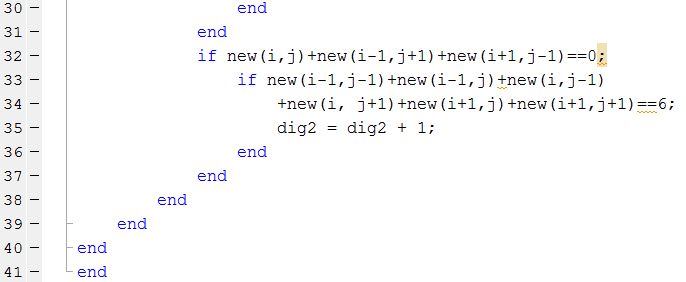


* *Intensity of Character*

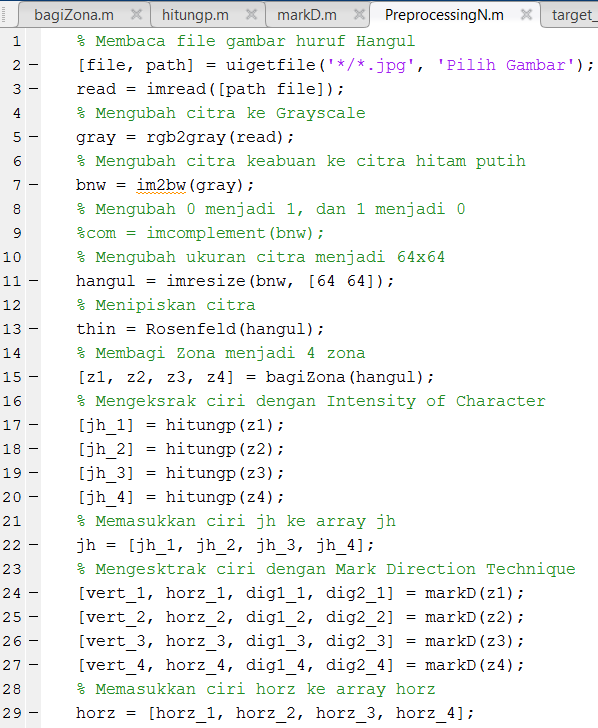


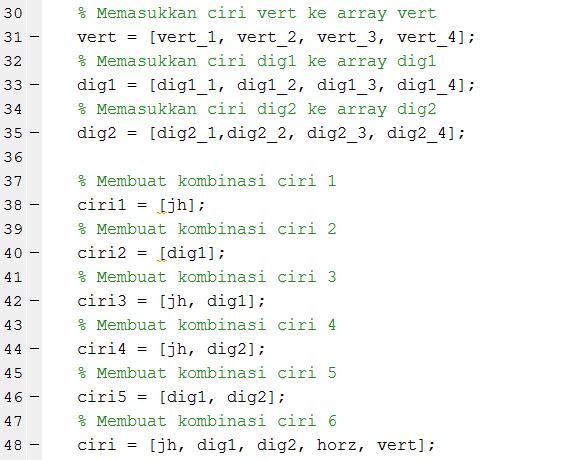
* Mark Direction



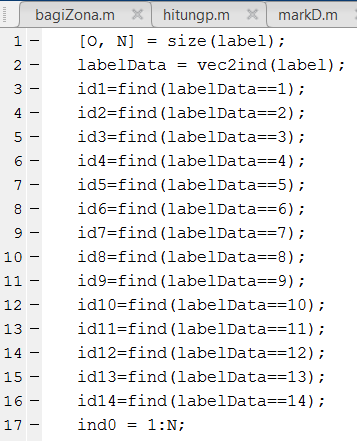


* Preprocessing

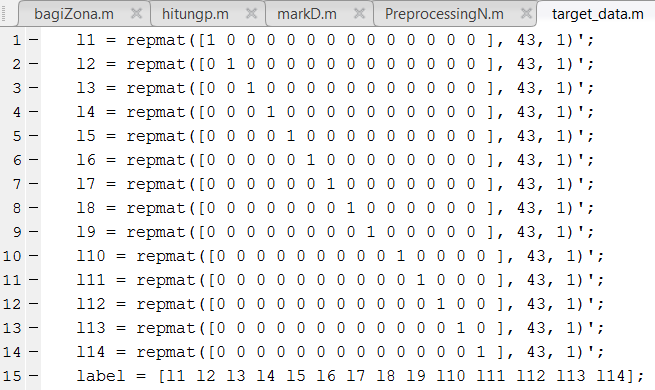




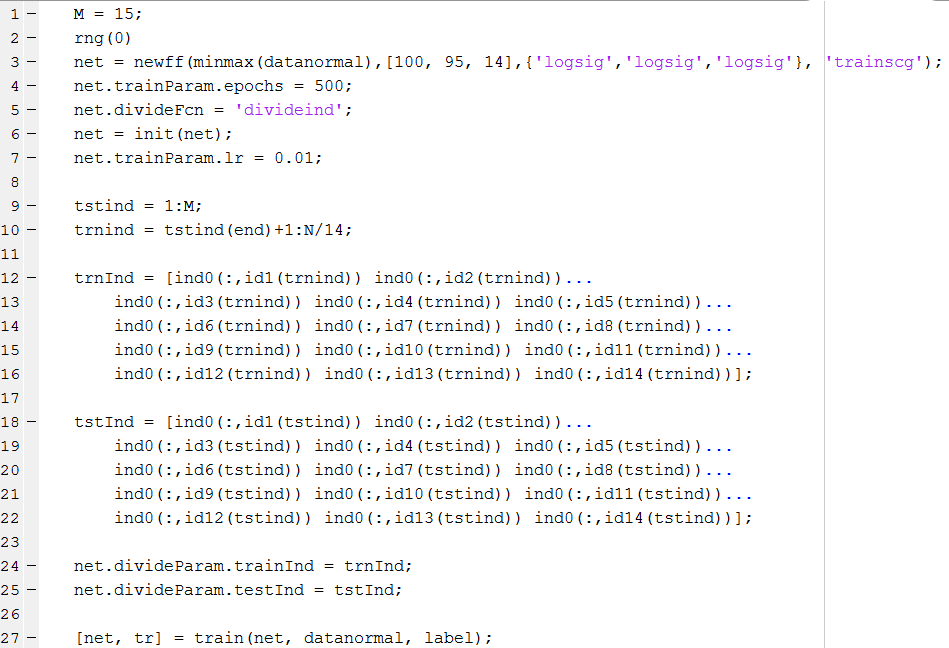
* Bagi Target



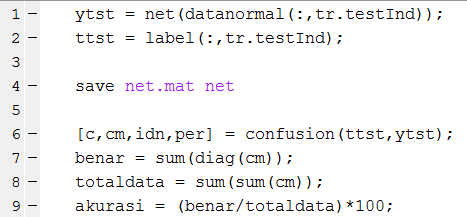
* Target Data



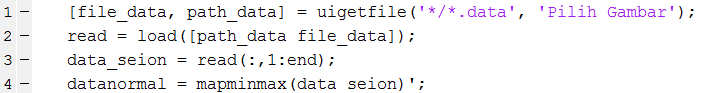
* Training JST



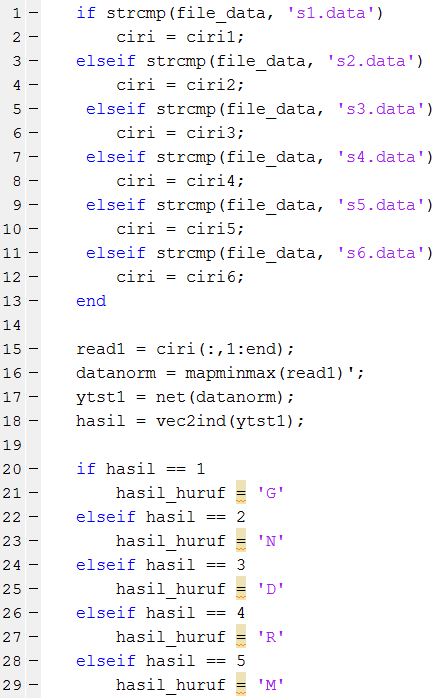
* Testing dan akurasi

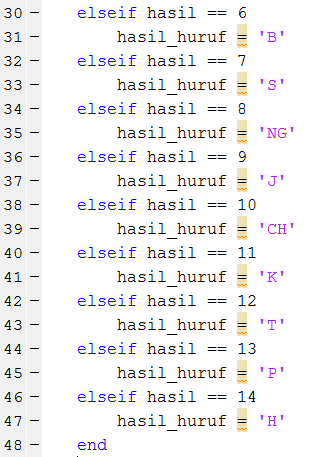


* Load Data



* Testing Data Tunggal





* GUI

