



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - TE 145561

**KLASIFIKASI GEMPA VULKANIK TIPE A DAN B PADA
MONITORING AKTIVITAS GUNUNG MERAPI
MENGUNAKAN *TOOLBOX NNTOOL* MATLAB**

Wisnu Waskitho Aji
NRP 10311500000003

Dosen Pembimbing
Ir. Arif Musthofa, M.T.
Dwi Lastomo, S.Si, M.T.

DEPARTEMEN TEKNIK ELEKTRO OTOMASI
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - TE145561

**KLASIFIKASI GEMPA VULKANIK TIPE A DAN B PADA
MONITORING AKTIVITAS GUNUNG MERAPI
MENGUNAKAN *TOOLBOX NNTOOL* MATLAB**

Wisnu Waskitho Aji
NRP. 1031150000003

Dosen Pembimbing
Ir. Arif Musthofa, MT.
Dwi Lastomo, S.Si, M.T.

DEPARTEMEN TEKNIK ELEKTRO OTOMASI
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT - TE145561

**CLASSIFICATION VOLCANIC EARTHQUAKES OF A AND B
TYPE ON MERAPI MOUNTAIN ACTIVITY MONITORING
USING MATLAB NNTOOL TOOLBOX**

Wisnu Waskitho Aji
NRP. 10311500000003

Supervisor
Ir. Arif Musthofa, MT.
Dwi Lastomo, S.Si, M.T.

*Electrical and Automation Engineering Department
Vocational Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2018*

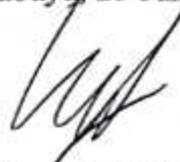
PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "**Klasifikasi Gempa Vulkanik Tipe A Dan B Pada *Monitoring* Aktivitas Gunung Merapi Menggunakan Toolbox Nntool Matlab**" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 25 Juni 2018



Wisnu Waskitho Aji
NRP 1031150000003

**KLASIFIKASI GEMPA VULKANIK TIPE A DAN B PADA
MONITORING AKTIVITAS GUNUNG MERAPI
MENGUNAKAN TOOLBOX NNTOOL MATLAB**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya Teknik
Pada
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing I

Dosen Pembimbing II

Ir. Arif Musthofa, M.T.
NIP. 196608111992031004

Dwi Lastomo, S.Si,M.T.
NIP. 19872017111059

**SURABAYA
JULI, 2018**

KLASIFIKASI GEMPA VULKANIK TIPE A DAN B PADA MONITORING AKTIVITAS GUNUNG MERAPI MENGUNAKAN *TOOLBOX NNTOOL* MATLAB

Wisnu Waskitho Aji
1031150000003

Dosen Pembimbing I : Ir. Arif Musthofa, M.T.
Dosen Pembimbing II : Dwi Lastomo, S.Si, M.T.

ABSTRAK

Gunung Merapi termasuk gunung berapi paling aktif di dunia. Aktivitas ke gempa pada Gunung Merapi dibagi menjadi gempa tektonik, vulkanik A dan B, guguran, dan multifase. Pada stasiun vulkanologi dan mitigasi di wilayah Gunung Merapi, getaran yang diterima oleh seismometer yang terpasang pada gunung kemudian ditransmisikan menuju stasiun untuk diinterpretasikan dalam bentuk seismograf. Penentuan jenis gempa di stasiun Gunung Merapi dilakukan secara manual dengan menganalisa bentuk gelombang yang di bentuk.

Dengan memanfaatkan bentuk gelombang gempa, dapat digunakan untuk menentukan jenis gempa menggunakan *toolbox nntool* pada matlab. Sehingga penentuan jenis gempa tidak perlu lagi dilakukan secara manual. Proses klasifikasi dimulai dengan melatih sistem untuk memahami kelas jenis gempa. Setelah sistem memahami data gempa yang dimiliki tiap kategori, maka proses klasifikasi dapat dilakukan. Jenis gempa yang dianalisis pada Tugas Akhir ini adalah gempa vulkanik tipe A dan B. Hasil klasifikasi tersebut digunakan sebagai penentu jenis gempa yang terjadi.

Pada Tugas Akhir ini, didapatkan hasil berupa pengenalan jenis gempa vulkanik tipe A dan tipe B yang dilakukan secara otomatis dengan akurasi total 91,83%. Akurasi pengenalan gempa tipe A sebesar 100%, akurasi gempa tipe B sebesar 84,21%, akurasi pengenalan gempa bukan tipe A&B sebesar 94,44%

Kata Kunci : Gempa Vulkanik, Klasifikasi, Matlab, Merapi

-----Halaman ini sengaja dikosongkan-----

**CLASSIFICATION VOLCANIC EARTHQUAKES OF A AND B
TYPE ON MERAPI MOUNTAIN ACTIVITY MONITORING
USING MATLAB NNTOOL TOOLBOX**

Wisnu Waskitho Aji
1031150000003

Supervisor I : Ir. Arif Musthofa, M.T.
Supervisor II : Dwi Lastomo, S.Si, M.T.

ABSTRACT

Mount Merapi is one of the most active volcanoes in the world. Seismic activity on Mount Merapi is divided into tectonic, volcanic A and B, falling, and multiphase earthquakes. At volcanology and mitigation stations in the Mount Merapi region, the vibrations are received by a mounted seismometer, then transmitted to the station for interpretation in seismographic form (called seismograph). Determination of the type of earthquake at Mount Merapi station is done manually by analyzing the shape of waveform.

By utilizing an earthquake waveform, it can be used to determine the type of earthquake using nntool toolbox on matlab. So that the determination of the type of earthquake is no longer need to be done manually. The identification process begins by training the system to understand the type of each earthquake category. After the system understands the earthquakes data of each category, then the classification process can be performed. The types of earthquakes analyzed in this Final Project are volcanic earthquakes of type A and B. The result of classification of such patterns is used as a determinant of the type of earthquake that occurred.

In this Final Project, the result of the introduction of type A and type B volcanic earthquakes is done with a total accuracy of 91.83%. Accuracy of Type A earthquake of 100%, type B quake accuracy of 84.21%, accuracy of earthquake introduction not type A & B by 94.44%

Keywords: *Classification, Matlab, Merapi, Volcanic Earthquakes*

-----Halaman ini sengaja dikosongkan-----

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT yang selalu memberikan rahmat dan hidayah-Nya sehingga Tugas Akhir ini dapat terselesaikan dengan baik. Shalawat serta salam semoga selalu dilimpahkan kepada Rasulullah Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Diploma-3 pada Program Studi Komputer Kontrol, Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya dengan judul:

KLASIFIKASI GEMPA VULKANIK TIPE A DAN B PADA MONITORING AKTIVITAS GUNUNG MERAPI MENGUNAKAN *TOOLBOX NNTOOL* MATLAB

Penulis mengucapkan terima kasih kepada Ibu dan Bapak penulis yang memberikan berbagai bentuk doa serta dukungan tulus tiada henti, Bapak Ir. Arif Musthofa, M.T. dan Dwi Lastomo, S.Si, M.T. atas segala bimbingan ilmu, moral, dan spiritual dari awal hingga terselesaikannya Tugas Akhir ini. Penulis juga mengucapkan banyak terima kasih kepada semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyelesaian Tugas Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, 25 Juni 2018

Penulis

----Halaman ini sengaja dikosongkan----

DAFTAR ISI

SAMPUL LUAR	i
SAMPUL DALAM	ii
PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PENGESAHAN	v
ABSTRAK.....	ix
<i>ABSTRACT</i>	xi
KATA PENGANTAR	xiii
DAFTAR GAMBAR.....	xvii
DAFTAR TABEL	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	3
1.5 Sistematika Laporan	3
1.6 Metodologi Penelitian	4
1.7 Relevansi.....	5
BAB II TEORI PENUNJANG	7
2.1 Gunung Merapi.....	7
2.1.1 <i>Monitoring</i> Aktivitas Gunung Merapi.....	8
2.1.2 <i>Monitoring</i> Kegempaan Gunung Merapi	9
2.2 Gempa Vulkanik.....	10
2.2.1 Gempa Vulkanik Dalam (Tipe A/VTA).....	11
2.2.2 Gempa Vulkanik Dangkal (Tipe B/VTB)	11
2.3 Klasifikasi Jenis Gempa	12
2.3.1 Penelitian Sebelumnya	12
2.3.2 Seismometer	12
2.4 Jaringan Saraf Tiruan (JST).....	14
2.5 Perangkat Lunak Matlab.....	18
2.5.1 Lingkungan Kerja Matlab	19
2.5.2 GUI.....	20
2.5.3 <i>M-File</i>	20
BAB III PERENCANAAN DAN PEMBUATAN SISTEM	23
3.1 Sistem Pemantauan Kegempaan Gunung Merapi	23
3.2 Desain Sistem Klasifikasi Jenis Gempa Vulkanik.....	23
3.2.1 Desain Metode Pelatihan JST Propagasi Balik	24
3.2.2 Proses Pelatihan	28

3.2.3	Proses Pengujian Sistem.....	30
3.3	Implementasi Metode	32
3.3.1	Pembangunan <i>Coding</i> Pada Matlab.....	32
3.3.2	Desain Antarmuka	34
3.4	Uji Coba Sistem Klasifikasi Gempa Vulkanik	36
3.4.1	Sumber Data	36
3.4.2	Lingkungan Uji Coba	36
3.4.3	Pelaksanaan Uji Coba.....	36
BAB IV PENGUJIAN ANALISIS SISTEM.....		41
4.1	Lingkungan Uji Coba	41
4.2	Data Uji Coba	41
4.3	Analisis Tampilan Sistem	43
4.4.1.	Antarmuka Menu Utama	43
4.4.2.	Antarmuka Menu Pelatihan.....	44
4.4.3.	Antarmuka Menu Pengujian.....	47
4.4	Pelatihan dan Pengujian Jaringan Saraf Tiruan	49
4.4.1	Proses Pelatihan Data Seismik Dengan JST Propagasi Balik.....	49
4.4.2	Proses Pengujian Data Seismik Dengan JST Propagasi Balik.....	57
4.5	Analisis Parameter Pelatihan JST	58
4.5.1.	Analisis Pengujian Data Uji Seismik	58
4.5.2.	Analisis Perulangan Pelatihan	60
4.5.3.	Analisis Perubahan <i>Epoch</i>	61
4.5.4.	Analisis Perubahan Jumlah Neuron Pada Layar Tersembunyi	61
BAB V KESIMPULAN DAN SARAN.....		63
5.1.	Kesimpulan.....	63
5.2.	Saran.....	64
DAFTAR PUSTAKA		65
LAMPIRAN A.....		67
LAMPIRAN B		69
RIWAYAT HIDUP PENULIS		71

DAFTAR GAMBAR

Gambar 2. 1	Gunung Merapi Tahun 2015	7
Gambar 2. 2	Metode Pemantauan Gunung Merapi.....	8
Gambar 2. 3	<i>Monitoring</i> Kegempaan Gunung Merapi	9
Gambar 2. 4	Mekanisme Gempa Vulkanik.....	10
Gambar 2. 5	Rekaman Gempa Vulkanik Tipe A	11
Gambar 2. 6	Rekaman Vulkanik Gempa Tipe B	11
Gambar 2. 7	Jaringan Pemantauan Merapi	13
Gambar 2. 8	Arsitektur Jaringan Saraf Tiruan Layar Tunggal	15
Gambar 2. 9	Arsitektur Jaringan Saraf Tiruan Propagasi Balik.....	16
Gambar 2. 10	Fungsi Aktivasi Sigmoid Biner	17
Gambar 2. 11	Fungsi Aktivasi Sigmoid Bipolar.....	17
Gambar 2. 12	Lingkungan Kerja Matlab	19
Gambar 2. 13	Jendela Kerja GUI.....	20
Gambar 2. 14	Fungsi <i>M-File</i>	21
Gambar 3. 1	Sistem Pemantauan Kegempaan Gunung Merapi	23
Gambar 3. 2	Proses Klasifikasi Jenis Gempa	24
Gambar 3. 3	Arsitektur JST <i>Backpropagation</i>	25
Gambar 3. 4	Kelas Setiap Jenis Gempa	26
Gambar 3. 5	Contoh Data Kelas Gempa.....	26
Gambar 3. 6	Cuplikan Hasil <i>Sampling</i> Data Gempa.....	27
Gambar 3. 7	<i>Plot</i> Data Rekaman Gempa	28
Gambar 3. 8	Target Pelatihan	28
Gambar 3. 9	Diagram Alir Proses Pelatihan JST <i>Backpropagation</i> ..	29
Gambar 3. 10	Diagram Alir Proses Pengujian	31
Gambar 3. 11	Cuplikan <i>Coding</i> Pelatihan	32
Gambar 3. 12	<i>Coding</i> Pembangunan JST	33
Gambar 3. 13	<i>Coding</i> Paramater dan <i>Training</i> Sistem	33
Gambar 3. 14	<i>Coding</i> Menampilkan Hasil Pelatihan	34
Gambar 3. 15	Desain Jendela Pelatihan Sistem	35
Gambar 3. 16	Desain Rencana Jendela Pengujian Sistem	35
Gambar 4. 1	Data Gempa	41
Gambar 4. 2	Tampilan Halaman Utama Sistem	44
Gambar 4. 3	Tampilan Jendela Pelatihan	45
Gambar 4. 4	Tampilan Jendela Pengujian	48
Gambar 4. 5	Tampilan Proses Pelatihan Sistem	51
Gambar 4. 6	<i>Performance</i> Pelatihan	52
Gambar 4. 7	Tampilan Dari <i>Training State</i>	53

Gambar 4. 8 Tampilan <i>Plot Regression</i>	54
Gambar 4. 9 Hasil Proses Pelatihan Sistem	55
Gambar 4. 10 Bobot Pada <i>Hidden Layer</i>	55
Gambar 4. 11 Bobot Bias Untuk <i>Hidden Layer</i>	56
Gambar 4. 12 Bobot Keluaran	56
Gambar 4. 13 Bobot Bias Keluaran	56
Gambar 4. 14 Contoh Tampilan Hasil Pengujian Data	57

DAFTAR TABEL

Tabel 3. 1	Jumlah Data Latih Dan Uji	26
Tabel 4. 1	Urutan Data Gempa	42
Tabel 4. 2	Hasil Pengujian	58
Tabel 4. 3	Perulangan Proses Pelatihan Dengan Parameter Tetap....	60
Tabel 4. 4	Perubahan <i>Epoch</i> Dengan Parameter Tetap.....	61
Tabel 4. 5	Perubahan Jumlah Neuron Pada Unit Tersembunyi	62

----Halaman ini sengaja dikosongkan----

BAB I

PENDAHULUAN

1.1 Latar Belakang

Gunung Merapi secara administratif terletak diantara Provinsi Daerah Istimewa Yogyakarta (DIY) dan Provinsi Jawa Tengah. Tepatnya, membatasi empat kabupaten yaitu Kabupaten Sleman dan Kabupaten Magelang, Kabupaten Boyolali dan Kabupaten Klaten. Dengan intensitas erupsi setiap 2-7 tahun, Gunung Merapi termasuk dalam gunung berapi paling aktif di dunia dan meningkatkan risiko bencana pada wilayah di sekitarnya. Badan Nasional Penanggulangan Bencana (BNPB) mencatat korban akibat erupsi 2010 mencapai 347 orang. Erupsi Gunung Merapi juga mengakibatkan kerugian materi pada berbagai sektor seperti permukiman, infrastruktur, telekomunikasi, listrik, dan air bersih [1]

Oleh karena itu, sistem mitigasi menjadi perhatian baik oleh pemerintah, organisasi non-pemerintah, dan orang-orang pada umumnya. Sistem mitigasi menurut UU Nomor 24 Tahun 2007 adalah serangkaian upaya untuk mengurangi risiko bencana, baik melalui pembangunan fisik maupun penyadaran dan peningkatan kemampuan menghadapi ancaman bencana [2]

Hulu sistem mitigasi adalah pemantauan aktivitas (*monitoring*) gunung berapi karena semua tindakan mitigasi dilakukan berdasarkan keadaan aktivitas gunung berapi dan skenario yang diprediksi. Prediksi aktivitas gunung berapi biasanya didasarkan pada dua aspek yaitu data historis dari aktivitas gunung berapi dan data aktivitas saat ini. Di sana, peran penting dari sistem pemantauan gunung berapi sangat diperlukan. Setiap sistem pemantauan aktivitas gunung berapi bertujuan untuk:

1. Mengidentifikasi tingkat aktivitas vulkanik saat ini (normal, siaga, waspada, awas)
2. Mengenal proses saat ini dan memperkirakan proses selanjutnya
3. Memprediksi potensi letusan meliputi waktu dan besarnya
4. Memahami aktivitas vulkanik / perilaku letusan dan siklusnya

Metode dan teknik yang umum diterapkan untuk memantau aktivitas gunungapi diantaranya penginderaan jauh, deformasi permukaan, pemantauan lahar, pemantauan gas dan pemantauan kegempaan.

Sampai sekarang, pemantauan aktivitas kegempaan (seismik) pada Gunung Merapi adalah metode yang paling kuat untuk mengetahui tingkat aktivitas Gunung Merapi. Pemantauan seismik dilakukan dengan seismometer yang terpasang pada 4 stasiun di sekeliling Merapi yaitu Pusunglondon, Klatakan, Plawangan dan Deles. Sinyal yang direkam seismometer dikirim ke stasiun penerima di Kantor BPPTKG Jogjakarta dengan telemetri radio VHF. Di stasiun penerima sinyal ini kemudian direkam pada kertas seismogram, dan juga disimpan dalam data digital untuk dianalisa [3]

Salah satu metode untuk memahami dan memprediksi aktivitas vulkanik adalah klasifikasi gempa dan menghitung jumlah masing-masing jenis gempa. Dalam Kasus erupsi Merapi, meningkatnya jumlah gempa *Volcano-tectonic* (VT) menjadi acuan meningkatnya aktivitas vulkanik

Klasifikasi gempa vulkanik berdasarkan bentuk dan frekuensi menurut Ratdomopurbo (1995) terbagi menjadi 7 jenis. Jenis gempa tersebut adalah Vulkanik tipe A, vulkanik tipe B, Multifase, Low Frequency Event (LF), *Low High Frequency Event* (LHF), dan Guguran [4]

Saat ini, pengklasifikasian gempa yang dilakukan oleh Balai Penyelidikan dan Pengembangan Teknologi Kebencanaan Geologi (BPPTKG) Yogyakarta masih manual, yakni dengan pengamatan langsung pada seismometer. Untuk memudahkan proses, dibutuhkan sistem yang dapat melakukan klasifikasi gempa secara otomatis.

Matlab merupakan perangkat lunak yang cocok dipakai sebagai alat komputasi yang melibatkan matrik dan kolom. Fungsi dari Jaringan Saraf Tiruan (JST) pada matlab dapat digunakan untuk pemrosesan informasi dengan karakteristik mirip dengan jaringan saraf biologi. Karena kemampuan yang mendekati saraf biologis manusia, jaringan saraf tiruan dapat diaplikasikan untuk melakukan klasifikasi. Kemampuan klasifikasi ini dapat dimanfaatkan untuk mengenali data getaran/sinyal gempa pada Gunung Merapi sehingga didapatkan hasil klasifikasi gempa *Volcano-tectonic*(VT) secara otomatis.

1.2 Rumusan Masalah

Untuk mengklasifikasi jenis gempa yang terjadi di Gunung Merapi masih dilakukan secara manual. Klasifikasi dapat digunakan untuk jenis-jenis gempa. klasifikasi dilakukan menggunakan Matlab

1.3 Batasan Masalah

Agar penulisan buku Tugas Akhir ini tidak menyimpang dan mengambang dari tujuan yang semula direncanakan sehingga mempermudah mendapatkan data dan informasi yang diperlukan, maka penulis menetapkan batasan-batasan masalah sebagai berikut :

1. Klasifikasi jenis gempa menggunakan *toolbox nntool* Matlab
2. Data berupa grafik amplitudo gempa Gunung Merapi dari tahun 2010 dalam bentuk digital
3. Data gempa dari 1 lokasi pemantauan rekaman seismometer (Pusunglondon)
4. *Software* yang dipakai adalah MATLAB R2015A
5. Jenis gempa yang diklasifikasi yaitu vulkanik A dan vulkanik B

1.4 Tujuan

Pembuatan Sistem Klasifikasi Jenis Gempa Pada Gunung Merapi memiliki tujuan untuk:

1. Mengotomatisasi klasifikasi jenis gempa vulkanik
2. Mengklasifikasikan gempa-gempa berdasarkan rekaman seismometer

1.5 Sistematika Laporan

Sistematika pembahasan Tugas Akhir ini terdiri dari lima bab, yaitu pendahuluan, teori penunjang, perencanaan dan pembuatan sistem, pengujian dan analisa sistem, serta penutup.

BAB I : PENDAHULUAN

Membahas tentang latar belakang, permasalahan, batasan masalah, maksud dan tujuan, sistematika laporan, serta relevansi.

BAB II : TEORI PENUNJANG

Berisi teori penunjang yang mendukung dalam perencanaan dan pembuatan sistem.

BAB III : PERANCANGAN DAN PEMBUATAN SISTEM

Membahas tentang perencanaan dan pembuatan sistem klasifikasi gempa yang meliputi pengenalan jenis gempa, pembuatan *coding*, dan penggunaan metode.

BAB IV : PENGUJIAN DAN ANALISA SISTEM

Membahas tentang pengujian dan penganalisaan terhadap sistem yang telah dibuat.

BAB V : KESIMPULAN DAN SARAN

Menjelaskan tentang kesimpulan dari Tugas Akhir ini dan saran-saran untuk pengembangan sistem ini lebih lanjut.

1.6 Metodologi Penelitian

Pengerjaan Tugas Akhir berupa Klasifikasi Gempa Vulkanik A dan B pada *Monitoring* Aktivitas Gunung Merapi Menggunakan Matlab dilakukan melalui beberapa tahapan, yaitu, tahap pengamatan, tahap studi literatur, tahap perancangan sistem, tahap pengujian sistem, dan tahap penyusunan laporan akhir.

Kegiatan penelitian diawali dengan studi literatur. Literatur yang digunakan berasal dari jurnal, buku ilmiah, dan beberapa artikel dari internet. Teori yang diperlukan sebagai penunjang perancangan sistem ini diantaranya teori mengenai karakteristik gempa pada Gunung Merapi, metode klasifikasi secara digital, dan jaringan saraf tiruan (JST). Studi literatur juga dilakukan melalui kerjasama dengan Balai Penelitian dan Pengembangan Teknologi Kebencanaan Geologi (BPPTKG) Yogyakarta. Kerjasama dilakukan untuk menambah informasi mengenai aktivitas kegempaan Gunung Merapi dan pengumpulan data getaran gempa Gunung Merapi.

Tahap kedua adalah perancangan sistem. Dimulai dengan pembacaan data getaran gempa dengan format di matlab (*matrix laboratory*). Setelah terbaca di matlab dalam bentuk pola grafik, data gempa tersebut dipecah setiap interval waktu tertentu. Data gempa hasil pemotongan grafik tiap interval waktu tersebut dijadikan sebagai data latih dan data uji. Kemudian merancang arsitektur jaringan saraf tiruan sebagai metode yang digunakan pada Tugas Akhir ini.

Tahap ke tiga yaitu pengujian dan analisis sistem. Sebelum diuji, jaringan saraf tiruan perlu dilatih. Pelatihan dilakukan menggunakan 165 data latih berupa pola gempa vulkanik tipe A, B, dan bukan kedua jenis gempa. Setelah didapatkan arsitektur yang sesuai, sistem diuji menggunakan 49 data uji. Jika hasil dari pengujian telah sesuai maka sistem dinyatakan berhasil.

Setelah dilakukan pengujian pada sistem dan hasilnya telah sesuai, ditarik kesimpulan mengenai penelitian dan tahapan terakhir

yaitu penyusunan laporan akhir sebagai bukti pernah dilakukan penelitian mengenai Klasifikasi Gempa Vulkanik A dan B pada Monitoring Aktivitas Gunung Merapi Menggunakan Klasifikasi Berbasis Jaringan Saraf Tiruan (JST).

1.7 Relevansi

Dengan adanya Tugas Akhir ini, diharapkan sistem dapat digunakan untuk mengetahui aplikasi Matlab dalam bidang mitigasi dan mengotomasikan penentuan jenis gempa yang terjadi pada Gunung Merapi sehingga dapat mengambil tindakan preventif menghadapi aktifitas Gunung Merapi

-----Halaman ini sengaja dikosongkan-----

BAB II

TEORI PENUNJANG

Pada bab ini membahas teori dasar dan teori penunjang dari komponen yang digunakan dalam pembuatan Tugas Akhir dengan judul *Klasifikasi Gempa Vulkanik Tipe A dan B Pada Monitoring Aktivitas Gunung Merapi Menggunakan Matlab*

2.1 Gunung Merapi [5][6]

Gunung Merapi secara administrasi terletak di kabupaten Sleman, Daerah Istimewa Yogyakarta untuk sisi selatan, sedangkan sisanya berada di provinsi Jawa Tengah yaitu untuk sisi barat gunung Merapi berada di kabupaten Magelang, untuk sisi utara dan timur berada di kabupaten Boyolali dan untuk sisi tenggara berada kabupaten Klaten. Saat ini ketinggian puncak gunung 2.930 m dpl, per 2010 yang telah mengalami perubahan kubah setelah meletus beberapa kali. Gunung Merapi memiliki rekahan yang semakin besar dan tidak lagi memiliki puncak seperti digambarkan pada Gambar 2.1



Gambar 2. 1 Gunung Merapi Tahun 2015

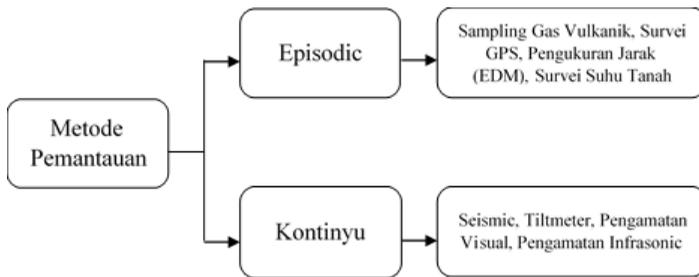
Gunung Merapi merupakan gunung api yang mengarah ke selatan dari Gunung Ungaran. Berdasarkan tektoniknya, terletak di zona subduksi (tempat terbentuknya deretan gunung berapi dan gempa bumi) yang mana Lempeng Indo-Australia menujam dibawah lempeng Eurasia atau

bergerak relatif ke arah utara sehingga menyebabkan aktivitas vulkanik disekitar Pulau Jawa.

2.1.1 *Monitoring* Aktivitas Gunung Merapi [4]

Kegiatan pemantauan (*Monitoring*) aktivitas Gunung Merapi sudah dilakukan sejak 1924. Alat pertama yang dipasang adalah seismometer mekanik *Wiechert* di lereng barat Gunung Merapi, kemudian tahun 1960 bekerjasama dengan Jepang dipasang seismometer *Hosaka* melengkapi seismometer yang sudah ada. Perkembangan terkini sistem pemantauan adalah menggunakan wahana satelit. Pemantauan disisi lain juga dilakukan seperti data *spectrometers* yang menunjukkan komposisi gas vulkanik, daya gravitasi, *remote sensing*, dan lain-lain.

Proses erupsi dan berbagai "tanda" yang muncul menjelang erupsi berbeda antara satu gunungapi dengan lainnya. Pemantauan aktivitas gunungapi, apalagi pada saat aktivitas gunung api meningkat harus melibatkan berbagai disiplin ilmu dengan berbagai macam peralatan. Apabila magma naik menuju ke permukaan maka tanda utama biasanya muncul sebagai indikasi menjelang erupsi yaitu meningkatnya gempa-gempa vulkanik, deformasi di permukaan akibat desakan magma, kenaikan gas-gas vulkanik dan adanya peningkatan suhu kawah



Gambar 2. 2 Metode Pemantauan Gunung Merapi

Gambar 2.2 menunjukkan skema pemantauan aktivitas Gunung Merapi. Metoda pemantauan berdasarkan cara mendapatkan datanya bisa dibagi atas dua kategori yaitu metoda pemantauan secara kontinyu yang memerlukan sistem pengiriman data melalui transmisi gelombang elektromagnetik. Secara episodik data diambil melalui survei lapangan pada waktu yang berlainan langsung di lokasi pengamatan. Pada Tugas

Akhir ini data didapatkan pada pemantauan secara kontinyu melalui pemantauan gempa.

2.1.2 *Monitoring Kegempaan Gunung Merapi* [4]

Kegempaan yang terjadi akibat adanya aktifitas suatu gunungapi merupakan salah satu parameter pemantauan yang sangat penting. Pemantauan kegempaan Gunung Merapi dilakukan oleh BPPTKG melalui data seismik yang diperoleh di lapangan.



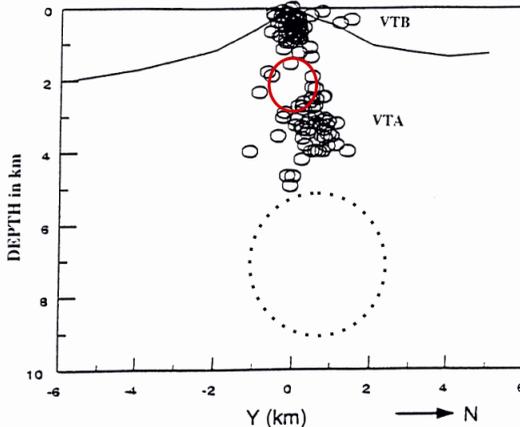
Gambar 2. 3 *Monitoring Kegempaan Gunung Merapi*

Gambar 2.3 merupakan sistem pemantauan kegempaan pada Gunung Merapi. Pemantauan kegempaan dengan telemetri seismik analog terdiri atas dua bagian yaitu sistem lapangan dan sistem penerima. Komponen utama sistem lapangan adalah seismometer (4) dan VCO-amplifier (5) yang berfungsi menguatkan sinyal dan merubah tegangan menjadi frekuensi yang akan ditumpangkan pada gelombang radio pembawa (6) dengan antena yagi (7). Adapun catu daya terdiri dari solar panel (1) dan regulator (2) untuk memutus dan menyambung arus dari aki (3) ke solar panel. Pada stasiun penerima sinyal akan diterima oleh radio receiver (a) yang kemudian diteruskan ke diskriminator (b).

Dari sini sinyal dapat disalurkan langsung ke recorder (seismograf) (d) atau disimpan dan ditampilkan secara digital di PC (e) dengan bantuan ADC (analog to digital converter) (c).

2.2 Gempa Vulkanik [7]

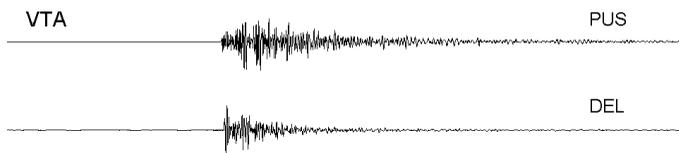
Gempa vulkanik adalah gempa yang disebabkan oleh pergerakan magma keluar dari dalam gunung berapi. Aktivitas magma tersebut menimbulkan getaran tanah hanya dirasakan di daerah sekitar gunung. Gempa vulkanik dapat dipakai sebagai tanda awal peningkatan aktivitas gunung api.



Gambar 2. 4 Mekanisme Gempa Vulkanik

Gambar 2.4 menjelaskan mengenai jenis gempa vulkanik tipe A dan B yang sering terjadi ketika gunung Merapi mendekati erupsi. Dapur magma primer Gunung Merapi diperkirakan terletak pada kedalaman lebih dari 5 kilometer dibawah puncak gunung dan dapur magma sekunder pada kedalaman antara 1,5 – 2,5 km. Pada Gunung Merapi gempa vulkanik dibagi berdasarkan data sinyal gempa dari jaringan stasiun seismig. Menurut Ratdomopurbo & Poupinet (2000), pengelompokan jenis gempa di gunung Merapi terdiri atas gempa vulkanik tipe A, vulkanik tipe B, gempa fase banyak (*multiphase*), gempa frekuensi rendah (*low frequency*), gempa LHF (*low high frequency*), dan gempa tremor.

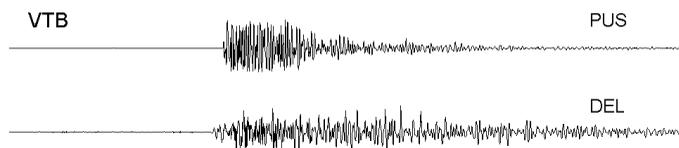
2.2.1 Gempa Vulkanik Dalam (Tipe A/VTA) [7]



Gambar 2. 5 Rekaman Gempa Vulkanik Tipe A

Gambar 2.5 merupakan contoh rekaman gempa vulkanik dalam pada stasiun pemantauan Pusunglondon (PUS) dan Deles (DEL). Gempa vulkanik dalam (tipe A) ditandai oleh hiposenter (pusat gempa) yang terjadi pada kedalaman lebih dari 2 km dibawah puncak gunung berapi, biasanya muncul pada Gunung api yang aktif. Penyebab gempa ini adalah adanya magma yang naik ke permukaan yang disertai rekahan-rekahan. Ciri utama dari gempa vulkanik tipe A adalah mempunyai waktu tiba gelombang primer (berbentuk longitudinal) dan sekunder (gelombang transversal) yang sangat jelas.

2.2.2 Gempa Vulkanik Dangkal (Tipe B/VTB) [3]



Gambar 2. 6 Rekaman Vulkanik Gempa Tipe B

Gambar 2.6 merupakan contoh rekaman gempa vulkanik tipe B pada stasiun pemantauan Pusunglondon (PUS) dan Deles (DEL). Gempa vulkanik dangkal (Tipe B) memiliki kedalaman kurang dari 2 km dan memiliki lebih banyak waktu tiba gelombang gempa pada seismometer yang muncul di stasiun yang letaknya jauh dari puncak. Untuk beberapa kejadian, gelombang sekunder (gelombang transversal) tidak dapat dibedakan dengan gelombang primer (gelombang longitudinal)

2.3 Klasifikasi Jenis Gempa [3]

Klasifikasi jenis gempa merupakan salah satu kegiatan untuk memantau aktivitas Merapi dan untuk mencari informasi mengenai proses yang sedang terjadi. Dalam Kasus erupsi Merapi, meningkatnya jumlah gempa MP dan VT (meliputi VTA dan VTB) adalah suatu tanda-tanda awal sebelum terjadinya letusan karena mencerminkan akumulasi stres akibat pergerakan magma menuju puncak gunung

2.3.1 Penelitian Sebelumnya

Ada banyak metode yang dapat digunakan untuk memantau aktivitas Gunung merapi. Seperti yang dikemukakan oleh R.S.J Sparks (2003), beberapa diantaranya adalah data kegempaan (*seismicity*), observasi satelit untuk deformasi tanah (*ground deformation*), data *spectrometer* yang menunjukkan komposisi gas vulkanik, daya gravitasi, *remote sensing*, dan jenis data lain. Penelitian ini hanya berupa tinjauan yang memaparkan beberapa metode tanpa rekomendasi metode tertentu yang memiliki pengaruh kuat dalam menentukan aktivitas.

Berdasarkan teori tersebut, peneliti memilih data kegempaan (*seismicity*) dan deformasi sebagai materi penelitian karena data tersebut memiliki pengaruh yang cukup kuat terhadap aktivitas Gunung Merapi sesaat sebelum letusan yang terjadi pada 2010.

Penelitian berikutnya oleh Nerisafitra (2015) menggunakan kombinasi metode jaringan saraf tiruan (JST) *backpropagation* dan logika fuzzy untuk menentukan kondisi aktivitas Gunung Merapi. Proses penentuan berdasarkan jumlah kejadian gempa setiap hari dari beberapa jeni gempa yaitu gempa vulkanik dalam (VTA), gempa vulkanik dangkal (VTB), gempa *low frequency* (LF), gempa *multiphase* (MP), dan guguran (RF).

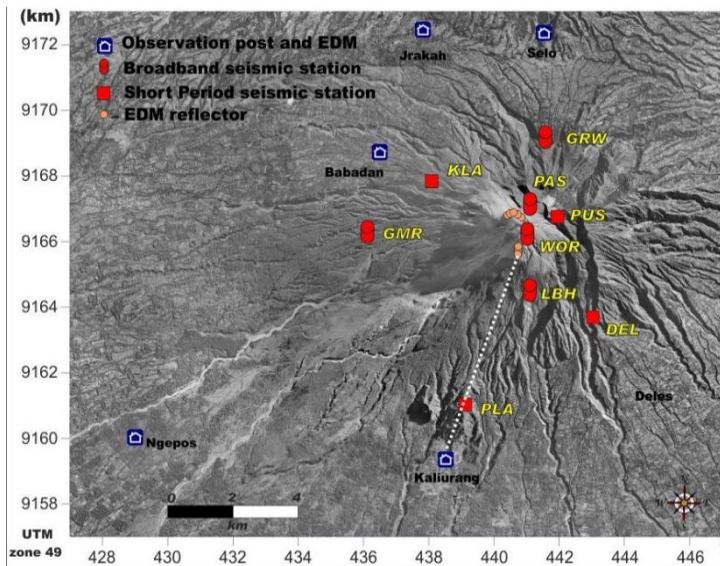
2.3.2 Seismometer[8][9]

Seismograf juga sering disebut dengan sebutan sismometer. Sismometer berasal dari bahasa Yunani yaitu *seimos*: gempa bumi dan *metero*: mengukur. Secara umum seismograf adalah alat atau sensor getaran, yang biasa digunakan untuk mendeteksi gempa bumi atau getaran pada permukaan tanah. Hasil rekaman dari alat ini disebut seismogram.

Berdasarkan frekuensinya, seismometer dibagi menjadi 3 jenis yaitu seismometer periode pendek (*Short Period Seismometer*) yang dapat merespon frekuensi 1 sampai 10 Hz, seismometer periode panjang

(*Long Period Seismometer*) dapat merekam frekuensi 0,01 sampai 0,1 Hz, dan seismometer broadband (*Broadband Seismometer*) yang merespon frekuensi 0,1 sampai 1 Hz.

Pada Gunung Merapi, terdapat 4 stasiun pemantauan yang dilengkapi dengan seismometer periode pendek. Getaran yang direkam seismometer ditransmisikan ke Yogyakarta (BPPTKG) melalui gelombang radio VHF (*Very High Frequency*). Kemudian sinyal dimodulasikan dan didigitalkan oleh sistem akuisisi *Guralp DM16S* pada tingkat 100 sampel per detik dengan akurasi 16 bit. Stasiun pemantauan periode pendek digunakan sebagai referensi dalam melakukan analisis kegempaan yang dilakukan secara rutin seperti melakukan klasifikasi dan perhitungan kejadian gempa.



Gambar 2. 7 Lokasi Pos Pemantauan Merapi

Gambar 2.7 adalah lokasi pos pemantauan merapi periode singkat. terdapat 4 stasiun pemantauan periode singkat yaitu Klatakan (KLA), Pusunglondon (PUS), Deles (DEL), dan Plawangan (PLA). Pada tugas akhir ini data berasal dari stasiun pengamatan Pusunglondon (PUS).

2.4 Jaringan Saraf Tiruan (JST) [10][11]

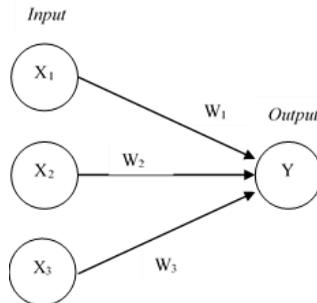
Jaringan saraf tiruan merupakan salah satu sistem pemrosesan informasi yang didesain dengan menirukan cara kerja otak manusia dalam menyelesaikan suatu masalah dengan melakukan proses belajar melalui perubahan bobot sinapsisnya. Jaringan saraf tiruan mampu melakukan pengenalan kegiatan berbasis data masa lalu. Data masa lalu akan dipelajari oleh jaringan saraf tiruan sehingga mempunyai kemampuan untuk memberikan keputusan terhadap data yang belum pernah dipelajari.

Setiap masukan dan keluaran yang diberikan kedalam lapisan jaringan saraf tiruan diproses dalam neuron. Menurut Puspitaningrum (2006), terdapat 3 jenis layar penyusun jaringan saraf tiruan yaitu :

1. Layar masukan (*input layer*)
Unit-unit dalam layar masukan disebut unit *input*. Unit *input* menerima pola masukan dari data luar
2. Layar tersembunyi (*hidden layer*)
Unit-unit dalam layar tersembunyi disebut unit tersembunyi. Unit ini menerima data dari unit *input* dan meneruskan ke unit *output*
3. Layar keluaran (*output layer*)
Unit-unit dalam lapisan *output* disebut unit *output*. Keluaran dari lapisan ini merupakan hasil yang diperoleh untuk menyelesaikan permasalahan

JST dibentuk sebagai generalisasi model matematika dari jaringan saraf biologi dengan asumsi bahwa pemrosesan informasi terjadi pada banyak elemen sederhana (neuron), sinyal dikirimkan diantara neuron-neuron melalui penghubung, penghubung antar neuron memiliki bobot yang akan memperkuat atau memperlemah sinyal. Kemudian untuk menentukan *output*, setiap neuron menggunakan fungsi aktivasi yang dikenalkan pada jumlahan *input* yang diterima. Besarnya *output* ini selanjutnya dibandingkan dengan suatu batas ambang. JST biasanya ditentukan oleh 3 hal yaitu:

1. Pola hubungan antar neuron
2. Metode untuk menentukan bobot penghubung (merupakan metode *learning*, atau algoritma)
3. Fungsi aktivasi



Gambar 2. 8 Arsitektur Jaringan Saraf Tiruan Layer Tunggal

Gambar 2.8 merupakan arsitektur jaringan saraf tiruan layer tunggal (tidak ada layer tersembunyi). Setiap unit (neuron) pada layer *input* hanya meneruskan sinyal ke unit *output*. Y menerima *input* dari neuron x_1 , x_2 , dan x_3 dengan bobot hubungan masing masing adalah w_1 , w_2 , dan w_3 . Ketiga sinyal dari neuron yang ada dijumlahkan dengan Persamaan 2.1.

$$net = x_1w_1 + x_2w_2 + x_3w_3 \dots \dots \dots (2.1)$$

Besarnya sinyal yang diterima oleh Y mengikuti fungsi aktivasi $Y = f(net)$. Apabila nilai masukan diatas fungsi aktivasi, maka sinyal diteruskan. Nilai fungsi aktivasi (model jaringan) dapat digunakan sebagai dasar merubah bobot.

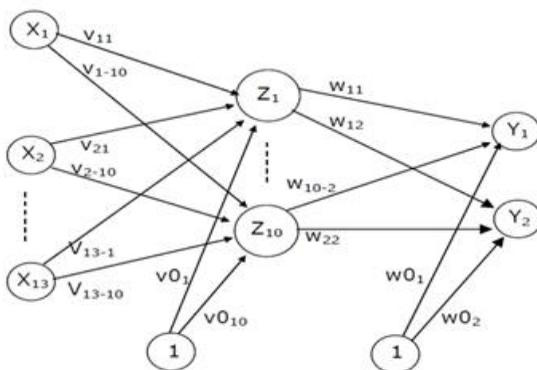
Penggunaan jaringan saraf tiruan meliputi beberapa hal seperti pengenalan pola, suara, angka, tanda tangan. Hal ini menarik dengan otak manusia yang masih mampu mengenali orang yang sudah beberapa waktu tidak dijumpai. Aplikasi lain JST adalah pengolahan sinyal untuk menekan *noise* dalam suara telepon. Dan yang sedang populer adalah meramalkan apa yang akan terjadi di masa mendatang berdasarkan pola kejadian yang ada di masa lampau. Ini dapat dilakukan mengingat kemampuan JST untuk mengingat dan membuat generalisasi dari apa yang sudah ada sebelumnya. Walaupun banyak aplikasi yang dapat dilakukan JST namun JST juga memiliki keterbatasan umum berupa keakuratan hasil yang dihasilkan dari pola yang terbentuk pada masukannya.

JST propagasi balik (*Backpropagation*) adalah sebuah metode sistematis untuk pelatihan JST layer banyak. Metode ini memiliki dasar matematis yang kuat, obyektik dan algoritma ini mendapatkan bentuk

Persamaan dan nilai koefisien dalam formula dengan meminimalkan jumlah kuadrat galat *error* melalui model yang dikembangkan.

JST dengan layar tunggal memiliki kelemahan dalam klasifikasi. Oleh karena itu ditambahkan satu atau beberapa layar yang disebut dengan layar tersembunyi (*hidden layer*) yang terletak diantara layar masukan dan layar keluaran. Meskipun penggunaan beberapa layar tersembunyi memiliki beberapa kelebihan, namun semakin banyak layar tersembunyi yang digunakan akan memerlukan waktu pelatihan yang lama.

JST propagasi balik melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan pada saat pelatihan dan untuk memberikan respon yang benar terhadap pola masukan yang serupa dengan pola yang dipakai pada proses pelatihan.



Gambar 2.9 Arsitektur Jaringan Saraf Tiruan Propagasi Balik

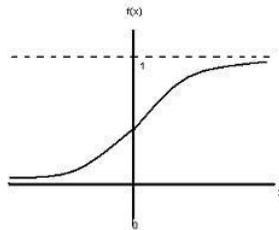
Gambar 2.9 menggambarkan arsitektur JST propagasi balik. Arsitektur jaringan saraf tiruan memiliki beberapa unit yang ada dalam satu atau lebih layar tersembunyi. Pada Gambar 2.9, v_{21} adalah bobot garis dari unit masukan x_2 ke unit layar tersembunyi z_1 dimana v_{01} adalah bobot garis yang menghubungkan bias di unit masukan ke unit layar tersembunyi z_1 . w_{11} merupakan bobot garis dari z_1 pada layar tersembunyi menuju ke y_1 di layar keluaran dengan w_{01} merupakan bobot garis dari unit bias di layar tersembunyi menuju ke y_1 pada unit keluaran.

Dalam JST propagasi balik, fungsi aktivasi yang digunakan harus memenuhi syarat yaitu kontinyu, terdiferensial dengan mudah dan merupakan fungsi yang tidak turun. Salah satu fungsi yang memenuhi persyaratan tersebut dan sering digunakan pada JST adalah fungsi sigmoid biner yang memiliki *range* (0,1). Persamaan 2.2 dan 2.3 menunjukkan fungsi untuk aktivasi sigmoid biner dan turunannya

$$f(x) = \frac{1}{1+e^{-x}} \dots \dots \dots (2.2)$$

Dengan fungsi turunannya:

$$f'(x) = f(x)(1 - f(x)) \dots \dots \dots (2.3)$$



Gambar 2. 10 Fungsi Aktivasi Sigmoid Biner

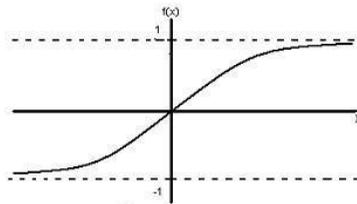
Gambar 2.10 merupakan fungsi aktivasi sigmoid biner dari rumus fungsi aktivasi sigmoid biner dan turunannya. [10]

Fungsi lain yang sering dipakai adalah fungsi sigmoid bipolar yang bentuk fungsinya mirip dengan fungsi sigmoid biner tapi dengan *range* (-1,1). Persamaan 2.4 dan 2.5 menunjukkan fungsi untuk aktivasi sigmoid bipolar dan turunannya

$$f(x) = \frac{2}{1+e^{-x}} - 1 \dots \dots \dots (2.4)$$

Dengan fungsi turunannya adalah

$$f'(x) = \frac{(1+f(x))(1-f(x))}{2} \dots \dots \dots (2.5)$$



Gambar 2. 11 Fungsi Aktivasi Sigmoid Bipolar

Dari rumus fungsi aktivasi sigmoid bipolar, didapatkan fungsi sigmoid bipolar yang ditunjukkan pada Gambar 2.11.

Fungsi sigmoid memiliki nilai maksimum 1, maka untuk pola dengan target lebih dari 1, pola masukan dan keluarannya harus terlebih dahulu ditransformasi sehingga semua polanya memiliki *range* yang sama seperti fungsi sigmoid yang dipakai. Alternatif lainnya adalah menggunakan fungsi aktivasi sigmoid hanya pada layar yang bukan keluaran. Pada layar keluaran, fungsi aktivasi yang digunakan adalah fungsi identitas, yaitu fungsi dimana $f(x) = x$.

2.5 Perangkat Lunak Matlab [12]

Matlab (Matrix Laboratory) adalah perangkat lunak untuk komputasi teknis dengan bahasa pemrograman level tinggi. Matlab mampu menyelesaikan masalah perhitungan dalam bentuk matriks. Matlab pertama kali dirilis pada tahun 1970 oleh Cleve Moler untuk menyelesaikan permasalahan aljabar linier. Saat ini, Matlab dikembangkan oleh MathWorks dengan menggabungkan proses pemrograman, komputasi, dan visualisasi melalui lingkungan kerja yang mudah digunakan.

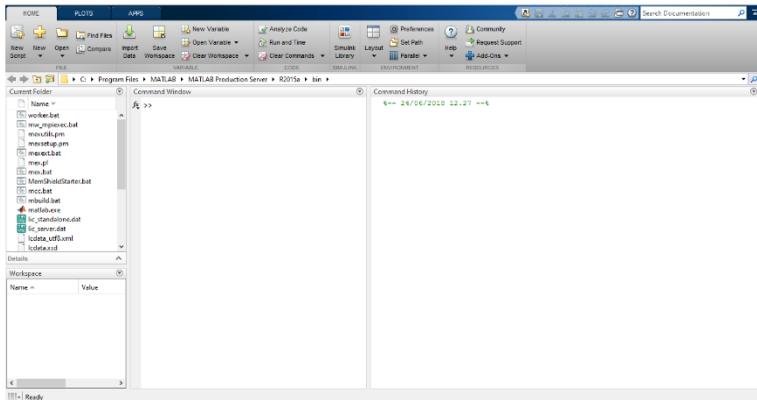
Matlab biasanya digunakan untuk pengembangan algoritma matematika dan komputasi Pemodelan, simulasi, pembuatan prototype dari penerimaan data analisa, eksplorasi, visualisasi data scientific dan engineering serta pengembangan aplikasi berbasis grafik dan pembuatan *Graphical User Interface (GUI)*.

Matlab memberikan sistem interaktif yang menggunakan konsep *array* tanpa membutuhkan pendeklarasian *array* seperti pada bahasa pemrograman yang lain. Selain itu, Matlab juga memberikan kemudahan bagi *programmer/developer program* yaitu untuk menjadi pembanding yang sangat handal, hal tersebut dapat dilakukan karena kekayaannya akan fungsi matematika, fisika, statistika, dan visualisasi.

Banyak model jaringan saraf tiruan menggunakan manipulasi matrik/vektor dalam iterasinya. Maka Matlab merupakan perangkat lunak yang cocok dipakai. Matlab menyediakan fungsi fungsi khusus untuk menyelesaikan model jaringan saraf tiruan. Pengguna hanya perlu menuliskan vektor masukan, target, model dan parameter yang diinginkan (laju pemahaman, threshold, bias).

2.5.1 Lingkungan Kerja Matlab [13]

Sebagaimana bahasa pemrograman lainnya, Matlab juga menyediakan lingkungan kerja terpadu yang sangat mendukung dalam membangun sebuah aplikasi. Pada setiap versi Matlab terbaru, lingkungan terpadu akan semakin dilengkapi. Lingkungan terpadu ini terdiri dari beberapa form yang memiliki kegunaan masing-masing. Setiap pertama kali membuka aplikasi Matlab, maka akan menyimpan *mode/setting* terakhir lingkungan kerja yang digunakan sebagai *mode/setting* lingkungan kerja pada saat membuka aplikasi Matlab di waktu berikutnya.



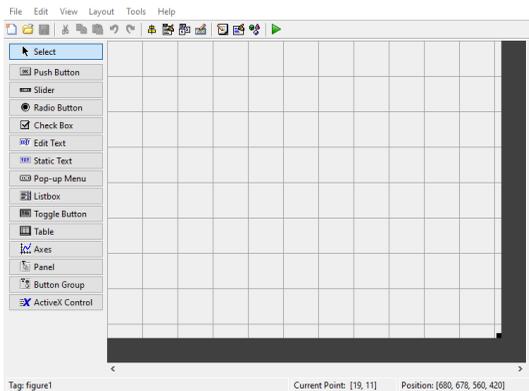
Gambar 2. 12 Lingkungan Kerja Matlab

Gambar 2.12 merupakan lingkungan kerja matlab yang terdiri dari beberapa bagian seperti menu Matlab, menu Plot, menu app di bagian atas jendela kerja. Bagian kanan dari Gambar 2.12 adalah *command history* yang berisikan perintah yang pernah dituliskan sebelumnya. pada sisi kiri dari Gambar 2.12 ada 2 bagian yang mana bagian atas adalah *current folder* yaitu posisi direktori yang saat ini dibuka. Sedangkan bagian bawah adalah *workspace window* yang menyediakan informasi mengenai variabel yang sedang aktif.

Pengguna perlu mengetikkan perintah yang diinginkan melalui jendela pada bagian tengah Gambar 2.12 yang disebut dengan *command window*. Disini terjadi interaksi antara pengguna dengan Matlab karena setiap perintah yang ditulis akan langsung dijalankan oleh Matlab.

2.5.2 GUI [13]

Dalam Matlab tersedia *tool* yang dapat digunakan untuk membuat *Graphical User Interface* (GUI) atau antarmuka pengguna grafis. GUI dibangun dengan objek grafis seperti tombol (*pushbutton*), *edit*, *slider*, *text*, *combo*, sumbu (*axes*), maupun menu dan lain-lain untuk kita gunakan. Sebagai contoh, ketika menggerakkan *slider*, maka akan bisa melihat perubahan sebuah nilai. Kemudian ketika menekan tombol OK, maka aplikasi akan dijalankan. Aplikasi yang menggunakan GUI umumnya lebih mudah dipelajari.



Gambar 2. 13 Jendela Kerja GUI

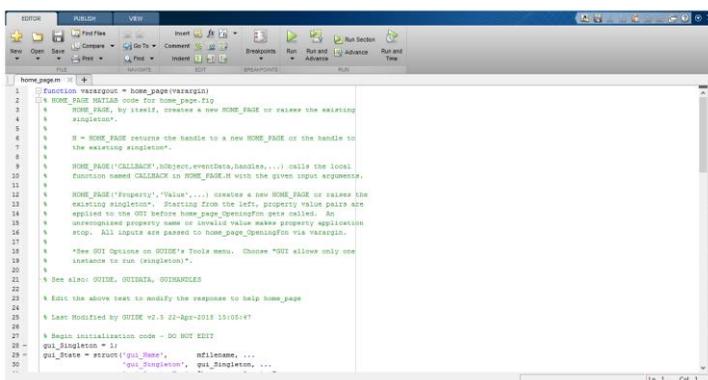
Gambar 2.13 merupakan jendela kerja untuk membuat GUI pada Matlab. Pada sisi kiri terdapat berbagai *tool* yang memiliki bermacam fungsi khusus. Ukuran file yang dihasilkan pada GUI relatif kecil sehingga cocok untuk aplikasi bidang sains dan teknik.

2.5.3 M-File [13]

Dalam membuat program GUI, matlab membuat program aplikasi berupa *M-file* yang menyediakan kerangka untuk mengontrol GUI. Kerangka dapat membantu membuat program menjadi lebih efisien dan sempurna. Semua kode termasuk kode *callback* dimasukkan dalam sebuah subfungsi dalam *M-file*. Pendekatan ini memungkinkan *M-file* memiliki titik masukan tunggal (*single entry point*) yang dapat

memanggil *callback* yang sesuai, atau sebuah subfungsi penolong yang ingin digunakan dalam GUI.

Fungsi *M-file* mirip dengan *script M-file* dimana keduanya merupakan file teks dengan ekstensi (.m). Sebagaimana *script M-file*, fungsi *M-file* tidak dimasukkan ke dalam jendela *command*, tetapi merupakan suatu file tersendiri yang dibuat dengan editor teks. Fungsi *M-file* menyediakan cara sederhana untuk menambah kemampuan *Matlab*, bahkan banyak fungsi standar *Matlab* yang merupakan fungsi *M-file*.



```
1 function varargout = home_page(varargin)
2 % HOME_PAGE MATLAB code for home_page.fig
3 % HOME_PAGE, by itself, creates a new HOME_PAGE or raises the existing
4 % singleton.
5 %
6 % If HOME_PAGE returns the handle to a new HOME_PAGE or the handle to
7 % the existing singleton.
8 %
9 % HOME_PAGE('CALLBACK', hObject,eventData,handles,...) calls the local
10 % function named CALLBACK in HOME_PAGE.M with the given input arguments.
11 %
12 % HOME_PAGE('Property','Value',...) creates a new HOME_PAGE or raises the
13 % existing singleton. Starting from the left, property value pairs are
14 % applied to the GUI before home_page_OpeningFcn gets called. An
15 % unrecognized property name or invalid value makes property application
16 % stop. All inputs are passed to home_page_OpeningFcn via varargin.
17 %
18 % *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
19 % instance to run (singleton)".
20 %
21 % See also: GUIDE, GUIDATA, GUIHANDLES
22
23 % Edit the above text to modify the response to help home_page
24
25 % Last Modified by GUIDE v2.5 22-Apr-2015 15:05:07
26
27 % Begin initialization code - DO NOT EDIT
28 gui_Singleton = 1;
29 gui_State = struct('gui_Name',       mfilename, ...
30                  'gui_Singleton',   gui_Singleton, ...
```

Gambar 2. 14 *M-File*

Gambar 2.14 merupakan *M-File* yang berisikan perintah untuk setiap objek yang telah dibuat pada GUI. Dengan menggunakan *M-File* pengguna dapat menyimpan program yang telah dibuat.

-----Halaman ini sengaja dikosongkan-----

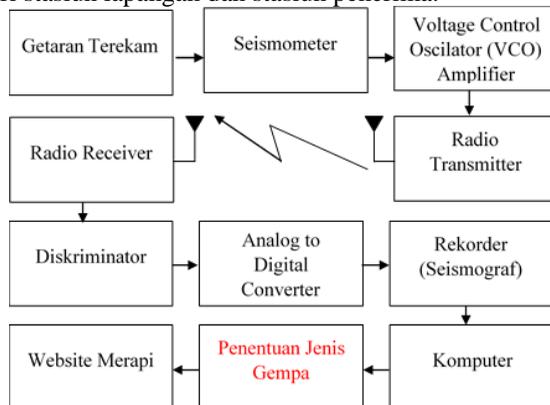
BAB III

PERENCANAAN DAN PEMBUATAN SISTEM

Pada bab ini dijelaskan mengenai tahapan yang dilakukan untuk menyelesaikan Tugas Akhir dengan judul Klasifikasi Gempa Vulkanik Tipe A dan B Pada *Monitoring* Aktivitas Gunung Merapi Menggunakan Matlab.

3.1 Sistem Pemantauan Kegempaan Gunung Merapi

Proses pemantauan gempa seperti dijelaskan pada SubBab 2.1.2 terdiri dari stasiun lapangan dan stasiun penerima.

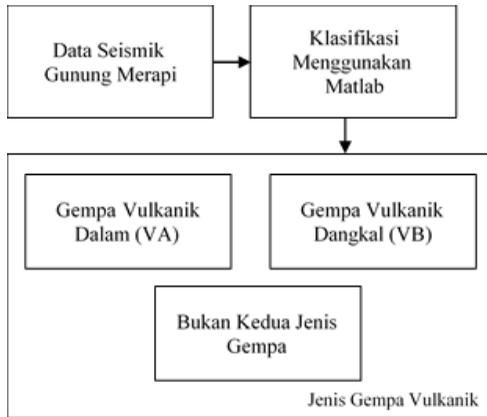


Gambar 3. 1 Sistem Pemantauan Kegempaan Gunung Merapi

Gambar 3.1 merupakan sistem pemantauan kegempaan Gunung Merapi. Pada Tugas Akhir ini, penulis membuat sistem untuk klasifikasi jenis gempa. Pengerjakan Tugas Akhir dilakukan setelah memperoleh data rekaman kegempaan yang tersimpan pada komputer.

3.2 Desain Sistem Klasifikasi Jenis Gempa Vulkanik

Pada bagian ini akan dijelaskan mengenai penggambaran alur proses yang terjadi dengan metode yang diusulkan untuk menghasilkan *output*. Penentuan desain aplikasi ini akan menggambarkan mengenai apa dan bagaimana metode yang diusulkan akan diterapkan dalam menyelesaikan masalah. Tahapan proses klasifikasi jenis gempa dapat dilihat pada Gambar 3.2.

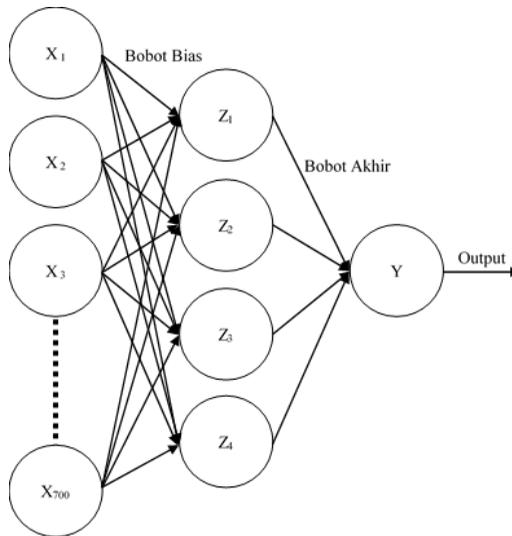


Gambar 3. 2 Proses Klasifikasi Jenis Gempa

Gambar 3.2 menjelaskan proses klasifikasi jenis gempa vulkanik pada pemantauan aktivitas gunung merapi. Dimulai dari data seismik data gunung merapi berupa data dari seismometer yang dikonversi menjadi data digital untuk diolah dengan *software* Matlab. Kemudian membangun *coding* untuk mengklasifikasi jenis gempa. Pada bagian keluaran diperoleh hasil berupa klasifikasi jenis gempa vulkanik pada gunung berapi yaitu gempa vulkanik dalam (VA), gempa vulkanik dangkal (VB), atau bukan kedua jenis gempa.

3.2.1 Desain Metode Pelatihan JST Propagasi Balik

Arsitektur jaringan saraf tiruan (JST) dipengaruhi oleh beberapa faktor, diantaranya adalah jumlah masukan, layer tersembunyi, bobot, dan keluaran. Pada Tugas Akhir ini, dilakukan pelatihan yang terawasi (*supervised learning*) terlebih dahulu agar dihasilkan pola dari setiap data pemantauan dengan tingkat akurasi tinggi.



Gambar 3. 3 Arsitektur JST *Backpropagation*

Gambar 3.2 merupakan arsitektur JST *backpropagation*. Arsitektur jaringan pada metode JST propagasi balik untuk klasifikasi jenis gempa Gunung Merapi ada 3 bagian utama yaitu *input* (X), *hidden layer* (Z), dan *output* (Y). Setiap neuron *input* (X) terhubung pada setiap neuron pada *hidden layer* (Z) dengan Bobot bias pada setiap neuron yang terhubung bernilai berbeda. Semua neuron pada lapisan tersembunyi (Z) terhubung pada lapisan *output* (Y) dengan bobot akhir berbeda setiap neuron. Perambatan dari *input* menuju *output* disebut perambatan maju (*feedforward*) kemudian jika nilai pada Y belum sesuai maka dilakukan perambatan balik (*backpropagation*) sehingga diperoleh nilai bobot yang sesuai untuk *output* berupa jenis gempa.

	1	2	3	4	5	6	7	8	9
1	49x1 struct	39x1 struct	38x1 struct	21x1 struct	21x1 struct	15x1 struct	14x1 struct	12x1 struct	5x1 struct

Gambar 3. 4 Kelas Setiap Jenis Gempa

Gambar 3.4 merupakan kelas setiap jenis gempa. Kolom 1 dan 2 adalah kelas bukan gempa tipe A dan B. Kolom 4, 5, dan 9 adalah kelas gempa tipe A. Kolom 3,6,7, dan 8 adalah kelas gempa tipe B.

Fields	scnl	Fs	start	data	units	version	misc_fields	misc_value
1	1x1 struct	50	7.3440e+05	700x1 double	'Counts'	1.2000	[]	
2	1x1 struct	50	7.3440e+05	700x1 double	'Counts'	1.2000	[]	
3	1x1 struct	50	7.3440e+05	700x1 double	'Counts'	1.2000	[]	
4	1x1 struct	50	7.3440e+05	700x1 double	'Counts'	1.2000	[]	
5	1x1 struct	50	7.3440e+05	700x1 double	'Counts'	1.2000	[]	
6	1x1 struct	50	7.3442e+05	700x1 double	'Counts'	1.2000	[]	
7	1x1 struct	50	7.3442e+05	700x1 double	'Counts'	1.2000	[]	
8	1x1 struct	50	7.3442e+05	700x1 double	'Counts'	1.2000	[]	
9	1x1 struct	50	7.3442e+05	700x1 double	'Counts'	1.2000	[]	
10	1x1 struct	50	7.3442e+05	700x1 double	'Counts'	1.2000	[]	
11	1x1 struct	50	7.3442e+05	700x1 double	'Counts'	1.2000	[]	
12	1x1 struct	50	7.3442e+05	700x1 double	'Counts'	1.2000	[]	

Gambar 3. 5 Contoh Data Kelas Gempa

Gambar 3.5 merupakan contoh kelas gempa. Wf {1,1} pada judul tabel menunjukkan baris pada kelas gempa (1) kemudian kolom kelas gempa yang dipilih (1) Pada kolom data berisi data amplitudo gempa yang telah *disampling* setiap 700 detik. Kolom *field* menunjukkan urutan kejadian gempa. Jumlah baris pada kelas gempa menunjukkan jumlah kejadian gempa pada setiap kelas gempa. Pada setiap data kelas gempa dibagi menjadi data latih dan data uji. Data latih digunakan untuk melatih sistem dan data uji digunakan untuk menguji sistem. Banyaknya data uji dapat diatur sesuai keinginan pengguna. Agar didapatkan sistem yang lebih akurat biasanya jumlah data latih lebih banyak dari jumlah data uji.

Tabel 3. 1 Jumlah Data Latih Dan Uji

Kelas Gempa	Jumlah Data Latih	Jumlah Data Uji
Wf {1,1}	40	9
Wf {1,2}	30	9

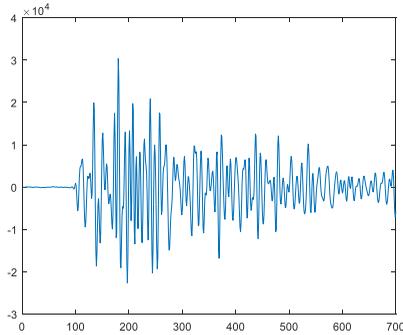
Kelas Gempa	Jumlah Data Latih	Jumlah Data Uji
Wf {1,3}	30	8
Wf {1,4}	15	6
Wf {1,5}	15	6
Wf {1,6}	10	5
Wf {1,7}	10	4
Wf {1,8}	10	2
Wf {1,9}	5	0
Total	165	49

Tabel 3.1 merupakan jumlah data latih dan data uji pada setiap kelas gempa. Jumlah data latih yang digunakan sebanyak 165 data dan data uji sebanyak 49 data sehingga total data yang digunakan pada Tugas Akhir ini sebanyak 214 data kejadian gempa

	1	2	3
1	-21.9689		
2	-23.5129		
3	-28.9709		
4	-31.9800		
5	-26.2822		
6	-10.2280		
7	12.4700		
8	34.4930		
9	48.2901		
10	50.5623		
11	44.9648		
12	39.3497		
13	38.7479		
14	40.8337		
15	38.6465		
16	27.2394		
17	7.1962		
18	-17.5310		
19	-42.1941		
20	-61.0443		
21	-66.4980		
22	-52.5858		
23	-21.2959		
24	15.3276		

Gambar 3. 6 Cuplikan Hasil *Sampling* Data Gempa

Gambar 3.6 merupakan cuplikan hasil *sampling* data gempa. Kolom 1 baris 1 menunjukkan besarnya amplitudo pada detik pertama. Jumlah baris menunjukkan banyaknya data untuk membentuk satu sinyal gempa. Pada tugas akhir ini banyaknya data yang digunakan untuk klasifikasi satu kejadian gempa adalah 700 data. Dari 700 data ini dijadikan unit masukan pada sistem klasifikasi jenis gempa menggunakan matlab.



Gambar 3. 7 *Plot* Data Rekaman Gempa

Gambar 3.7 merupakan *plot* data rekaman gempa. Bagian horisontal menunjukkan waktu untuk satu kejadian gempa sedangkan bagian vertikal menunjukkan besarnya amplitudo.

Hidden Layer merupakan variabel bebas sehingga pengguna dapat mengubah sesuai kebutuhan. Banyaknya hidden layer dapat mempengaruhi kemampuan klasifikasi sistem. Pada bagian keluaran dari JST merupakan data keluaran hasil klasifikasi yang dibandingkan dengan data target.

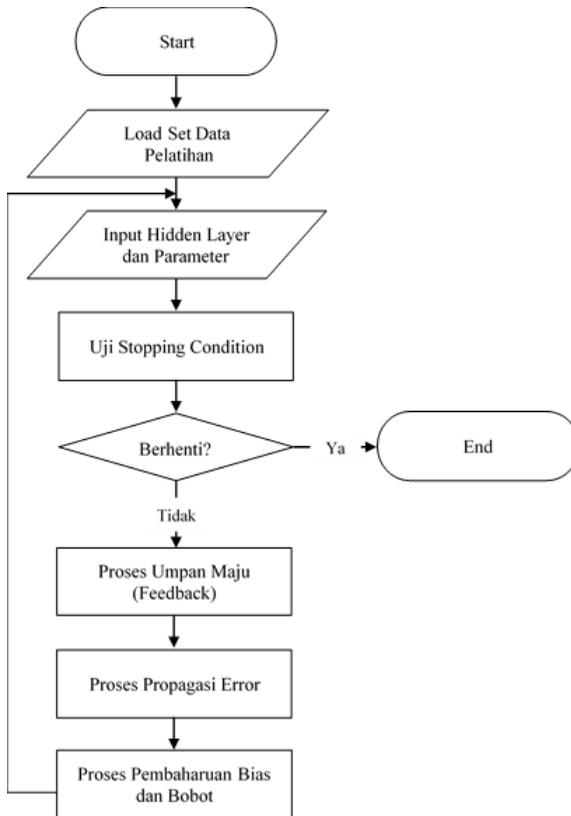
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Gambar 3. 8 Target Pelatihan

Gambar 3.8 merupakan target pelatihan. Target pelatihan merupakan kelas yang seharusnya di dapatkan pada unit keluaran. Nilai 1 merupakan inisialisasi kelas target untuk gempa tipe A, nilai 2 untuk inisialisasi gempa tipe B, dan nilai 0 untuk inisialisasi kelas bukan kedua jenis gempa.

3.2.2 Proses Pelatihan

Pelatihan merupakan prosedur awal yang harus dilakukan pada jaringan saraf tiruan (JST) untuk memperoleh kemampuan terbaik dari arsitektur JST. Alur pelatihan digambarkan seperti pada Gambar 3.9



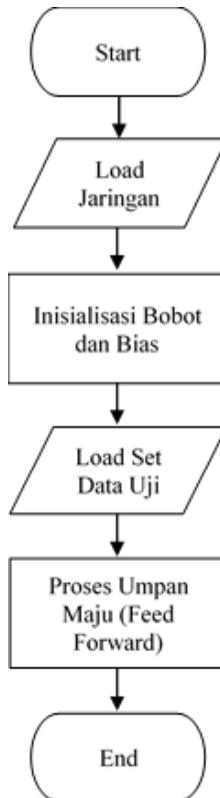
Gambar 3. 9 Diagram Alir Proses Pelatihan JST *Backpropagation*

Gambar 3.9 merupakan diagram alir proses pelatihan JST *backpropagation*. Proses diawali dengan memasukkan data pemantauan ke dalam sistem (di matlab). Kemudian memasukkan jumlah lapisan tersembunyi (*hidden layer*), jenis fungsi aktivasi, dan parameter yang meliputi *error goal*, *maximum epoch*, *learning rate* dan *training function*. Setelah itu dilakukan proses uji *stopping condition* yaitu pengujian untuk mendapatkan bobot terbaik pada setiap neuron sehingga didapat kemampuan terbaik jaringan. Pengujian diawali dengan perambatan maju, lalu sistem melakukan proses propagasi *error* yaitu penghitungan kesalahan pembacaan data. Jika *error* masih tinggi maka sistem akan melakukan proses pembaharuan bias dan bobot untuk

dilakukan uji *stopping condition* kembali. Proses *feedforward* dan JST propagasi balik berlangsung berulang-ulang hingga tercapai tujuan pelatihan, yaitu jumlah *epoch* (jumlah perulangan proses *feedforward* dan JST propagasi balik) lebih besar atau sama dengan *maksimum epoch*, atau nilai MSE (*Mean Square Error*) lebih kecil atau sama dengan target *error*. Selain itu, dalam jaringan saraf tiruan propagasi balik digunakan fungsi aktivasi yang berfungsi untuk menentukan keluaran dari suatu neuron atau jaringan. Bobot dan arsitektur jaringan hasil pelatihan yang telah mencapai akurasi tinggi nantinya digunakan untuk melakukan proses pengujian.

3.2.3 Proses Pengujian Sistem

Pengujian sistem dilakukan menggunakan jaringan saraf tiruan propagasi balik. Nilai bobot jaringan saraf tiruan yang digunakan yaitu nilai bobot pada waktu pelatihan. Proses pengujian sistem bertujuan untuk menguji akurasi dari proses pelatihan. Nilai bobot akhir pada proses pelatihan akan digunakan untuk menguji *set* data yang sudah dipersiapkan yaitu sebanyak 49 set data untuk gempa vulkanik tipe A, vulkanik tipe B, dan bukan gempa vulkanik tipe A maupun B.



Gambar 3. 10 Diagram Alir Proses Pengujian

Alur proses pengujian jaringan saraf tiruan propagasi balik seperti yang digambarkan pada Gambar 3.10. proses diawali dengan memuatkan arsitektur jaringan dari proses pelatihan. Kemudian sistem akan menginisiasikan bobot dan bias. Setelah itu memasukkan set data uji. Dilanjutkan dengan proses umpan maju pada proses pengujian yaitu sistem akan membaca data dengan arsitektur jaringan yang dibentuk.

Pada proses pengujian sistem ini akan dihasilkan tingkat akurasi sebagai tolok ukur keberhasilan sistem. Tingkat akurasi pada proses pengujian dapat dihitung melalui Persamaan:

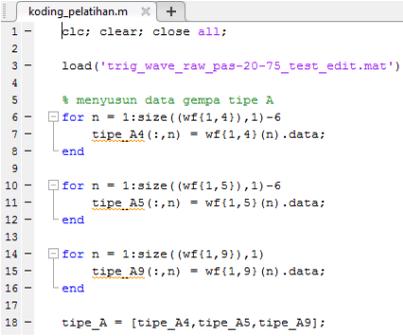
$$akurasi = \frac{keseluruhan\ jumlah\ data - jumlah\ data\ salah}{keseluruhan\ data} \times 100\% \dots\dots(3.1)$$

3.3 Implementasi Metode

Implementasi metode merupakan tahapan untuk menerapkan dari model algoritma yang telah dirancang ke dalam suatu bahasa pemrograman. Tahapan ini menghasilkan suatu program sebagai media yang dapat mewakili hasil dari algoritma yang diusulkan. Terdapat dua hal yang dilakukan pada langkah implementasi, yaitu pembangunan *coding* JST propagasi balik dan pembuatan desain antarmuka (interface) sehingga dapat dilakukan interaksi antara sistem dengan pengguna. Pada Tugas Akhir ini model algoritma akan diimplementasikan pada suatu lingkungan komputer dengan sistem operasi Windows menggunakan aplikasi Matlab versi R2015a.

3.3.1 Pembangunan *Coding* Pada Matlab

Pembuatan *coding* sistem klasifikasi gempa vulkanik dilakukan melalui pembangunan *coding* secara manual yang disimpan dalam *M-File* (.m). *Coding* dibangun sesuai jumlah jenis gempa yang akan digunakan pada tugas akhir ini. Selain itu *coding* untuk proses pelatihan dan pengujian dibedakan.



```
koding_pelatihan.m
1 -   clc; clear; close all;
2
3 -   load('trig_wave_raw_pas-20-75_test_edit.mat')
4
5   % menyusun data gempa tipe A
6 -   for n = 1:size(wf{1,4}),1)-6
7 -       tipe_A4(:,n) = wf{1,4}(n).data;
8 -   end
9
10 -  for n = 1:size(wf{1,5}),1)-6
11 -       tipe_A5(:,n) = wf{1,5}(n).data;
12 -   end
13
14 -  for n = 1:size(wf{1,9}),1)
15 -       tipe_A9(:,n) = wf{1,9}(n).data;
16 -   end
17
18 -  tipe_A = [tipe_A4,tipe_A5,tipe_A9];
```

Gambar 3. 11 Cuplikan *Coding* Pelatihan

Gambar 3.11 merupakan cuplikan *coding* pelatihan. Pembangunan *coding* pelatihan dimulai dengan membuka data gempa yang terdapat pada baris 3. Kemudian data gempa disusun antara tipe vulkanik A, vulkanik B, dan bukan kedua jenis gempa. Selain itu, data gempa dipisahkan antara data untuk pelatihan dan data untuk pengujian.

```

50 % Pembuatan JST
51 - data_latih = [tipe_A,tipe_B,tipe_bukan];
52 - target_latih = zeros(1,size(data_latih,2));
53 - target_latih(1:size(tipe_A,2)) = 1;
54 - target_latih(size(tipe_A,2)+1:size(tipe_B,2)) = 2;
55
56
57 % load bobot_awal.mat
58 - net = newff(minmax(data_latih),[10 1],{'logsig','purelin'},'traingdx');

```

Gambar 3. 12 Coding Pembangunan JST

Gambar 3.12 merupakan *coding* untuk pembangunan arsitektur JST menggunakan *toolbox* matlab *nntool*. Variabel `data_latih` merupakan data masukan kejadian gempa yang telah disusun. Variabel `target_latih` merupakan kelas target untuk setiap masukan. Target untuk tipe A memiliki nilai 1 dan menempati kolom target 1 sampai 35. Target untuk tipe B memiliki nilai 2 dan menempati kolom target 36 sampai 95. Target untuk tipe bukan A dan B memiliki nilai 0 dan menempati kolom target 96 sampai 165. Langkah selanjutnya adalah pembangunan jaringan pada matlab dengan sintaks: `net = newff(minmax(data_latih),[10 1],{'logsig','purelin'},'traingdx')`. Dimana `minmax (data_latih)` adalah nilai pada setiap unit masukan. 10 merupakan jumlah neuron pada layar tersembunyi. Banyaknya neuron pada unit tersembunyi dapat diubah. Sedangkan 1 merupakan jumlah neuron pada unit keluaran. Unit keluaran harus bernilai 1 karena pada sistem ini hanya memiliki satu keluaran yaitu jenis gempa. `logsig` adalah fungsi pelatihan pada layar tersembunyi berupa fungsi aktivasi sigmoid biner. Sedangkan `purelin` adalah fungsi pelatihan pada unit keluaran berupa fungsi aktivasi linier. `traingdx` adalah fungsi pelatihan jaringan.

```

64 % Memberikan nilai untuk mempengaruhi proses pelatihan
65 - net.performFcn = 'mse';
66 - net.trainParam.goal = 0.001;
67 - net.trainParam.epochs = 1000;
68 - net.trainParam.mc = 0.95;
69 - net.trainParam.lr = 0.1;
70
71 % Proses training
72 - [net_keluaran,tr,Y,E] = train(net,data_latih,target_latih);

```

Gambar 3. 13 Coding Paramater dan Training Sistem

Gambar 3.13 merupakan cuplikan *coding* untuk memberikan parameter pelatihan dan melakukan proses *training* pada sistem. Setelah jaringan dibangun, langkah selanjutnya adalah memasukkan parameter untuk proses pelatihan diantaranya jenis fungsi untuk mengukur kemampuan jaringan (baris 65), besarnya *error goal* (baris 66),

banyaknya *epoch* (baris 67), dan besarnya *learning rate* (baris 69). Langkah selanjutnya adalah melatih jaringan dengan jaringan yang telah dibangun (baris 72).

```
82 % Performa jaringan
83 - hasil_latih = round(sim(net_keluaran,data_latih));
84
85 - total_data = size(data_latih,2);
86 - [m,~] = find(hasil_latih==target_latih);
87 - jumlah_benar = numel(m);
88 - jumlah_salah = total_data-jumlah_benar;
89 - akurasi = numel(m)/total_data*100
```

Gambar 3. 14 Coding Menampilkan Hasil Pelatihan

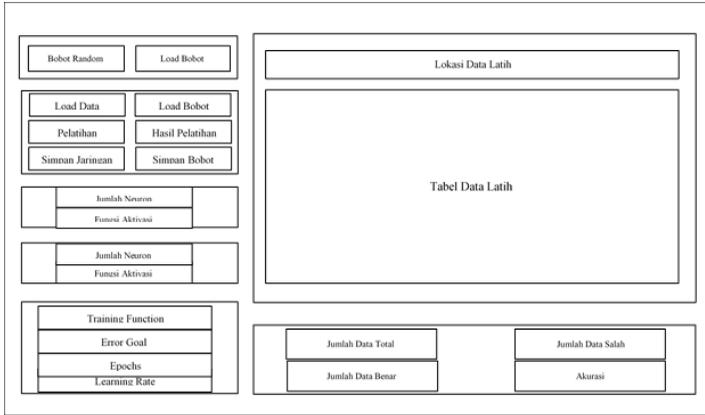
Gambar 3.14 merupakan cuplikan *coding* untuk menampilkan hasil pelatihan. Setelah jaringan dilatih, langkah selanjutnya adalah menampilkan hasil pelatihan. Apabila hasil pelatihan memiliki kelas target yang tepat maka klasifikasi gempu dinyatakan benar. Sedangkan klasifikasi yang tidak tepat dinyatakan salah. Akurasi didapatkan dengan menghitung banyaknya data benar dibagi jumlah data dikali 100% (baris 89). Setelah didapatkan jaringan yang sesuai, langkah selanjutnya adalah pembangunan *coding* untuk pengujian sistem.

Pembangunan *coding* pengujian sistem dimulai dengan membuka *file* data gempu. Langkah selanjutnya adalah menyusun gempu menjadi gempu tipe A, gempu tipe B, dan bukan kedua jenis gempu.

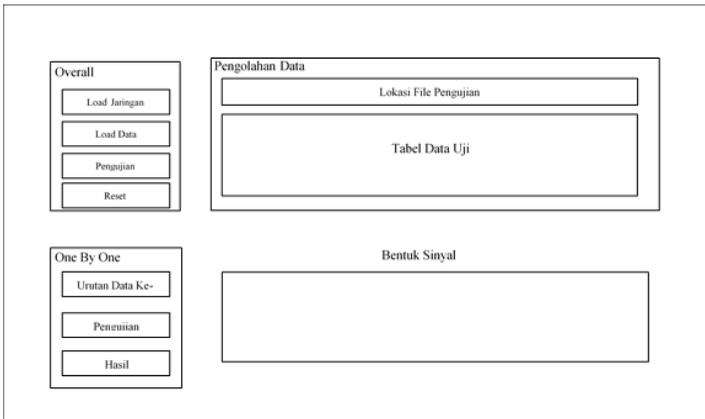
Setelah data dipisahkan antara pelatihan dan pengujian, dilakukan *load* jaringan yang tersimpan dari hasil pelatihan. Setelah dibuka, langkah selanjutnya adalah menuliskan kode untuk melakukan proses pengujian. Kemudian menampilkan hasil pengujian.

3.3.2 Desain Antarmuka

Interaksi dari pengguna dengan sistem diperlukan agar diperoleh hasil penelitian yang diharapkan. Oleh karena itu dibuatkan desain antarmuka. Desain antarmuka dibuat menggunakan fasilitas GUI (*Graphic User Interface*) yang terdapat pada Matlab. *Interface* yang dibuat untuk Tugas Akhir ini terdapat 3 jendela kerja yaitu bagian awal, jendela pelatihan, dan jendela pengujian sistem.



Gambar 3. 15 Desain Jendela Pelatihan Sistem



Gambar 3. 16 Desain Rencana Jendela Pengujian Sistem

Desain rencana antarmuka yang dibuat pada Tugas Akhir ini dapat dilihat pada Gambar 3.15 untuk rencana antarmuka pelatihan sistem dan Gambar 3.16 untuk rencana atarmuka pengujian sistem. Untuk melakukan perpindahan pada setiap jendela dapat dilakukan dengan men-klik pada bagian tab. Pada jendela pelatihan pengguna dapat melihat hasil pelatihan bobot pada bagian akurasi pelatihan sehingga bobot dapat disimpan untuk dilihat kembali hasil pelatihan bobot atau digunakan untuk proses pengujian. Pada jendela pengujian

sistem pengguna dapat melakukan uji bobot dengan cara load jaringan dilanjutkan load data kemudian men-klik pengujian. Tombol reset digunakan untuk memulai kembali proses pelatihan. Selain itu terdapat bagian fasilitas *one by one* yang memungkinkan pengguna melihat bentuk gelombang dan nama gempa dari setiap data uji.

3.4 Uji Coba Sistem Klasifikasi Gempa Vulkanik

Setelah sistem selesai dibuat maka langkah berikutnya adalah uji coba dan analisa sistem. Pada subbab ini akan dibahas mengenai cara melakukan uji coba sistem. Sedangkan hasil pengujian dan analisa hasil akan dijelaskan pada SubBab selanjutnya.

3.4.1 Sumber Data

Data yang digunakan pada Tugas Akhir ini adalah data seismik Gunung Merapi yang dimiliki oleh Balai Penelitian dan Pengembangan Teknologi Kebencanaan Geologi (BPPTKG) Yogyakarta. Data seismik berasal dari pos pemantauan Pusunglondon (PUS) pada bulan oktober 2009 sampai oktober 2010 yang tersebut meliputi:

- a. Data gempa dangkal (VA)
- b. Data gempa dalam (VB)
- c. Data Gempa Lain (*Low Frequency, Multiphase, Guguran*)

3.4.2 Lingkungan Uji Coba

Untuk melakukan pengujian sistem, program pada matlab harus dijalankan pada sebuah komputer. Adapun spesifikasi komputer yang digunakan untuk pengujian adalah:

- a. Processor Intel Core i3 2,0 GHz
- b. RAM 4GB
- c. Harddisk
- d. Software MATLAB R2015A

3.4.3 Pelaksanaan Uji Coba [14]

Pelaksanaan uji coba dilakukan dengan tahap pelatihan JST propagasi balik. Tahap pelatihan digunakan untuk memperoleh nilai bobot yang sesuai dengan target yang ingin dicapai sehingga sistem memiliki nilai akurasi yang tinggi antara proses pelatihan dan pengujian. Adapun langkah-langkah yang dilalui jaringan saraf tiruan propagasi balik adalah sebagai berikut:

1. Langkah 0: Penginisialisasian bobot dan bias dapat diatur secara acak, biasanya disekitar angka (0 dan 1 atau -1 (bias positif atau negatif))
2. Langkah 1: Apabila pada kondisi penghentian nilai yang didapat belum terpenuhi, maka ditempuh langkah 2-9
3. Langkah 2: Pada setiap data latih, ditempuh langkah 3 sampai 8
4. Umpan maju (*Feed forward*)

Langkah 3: Masing masing unit masukan ($x_i = i=1,2,\dots,n$) menerima sinyal masukan x_i . Sinyal masukan x_i dikirim ke seluruh unit pada layar tersembunyi. Masukan x_i yang dipakai adalah *input training data* yang sudah diatur skalanya. Nilai tertinggi dan terendah dari masukan yang dipakai dalam sistem kemudian dicari. Skala yang digunakan disesuaikan dengan fungsi aktivasinya. Bila menggunakan sigmoid biner dengan nilai terendah adalah 0 dan nilai tertingginya 1, maka nilai terendah masukan juga dianggap 0 sedangkan nilai tertinggi dianggap 1. Nilai-nilai diantaranya bervariasi antara 0 dan 1. Sedangkan bila menggunakan fungsi aktivasi sigmoid bipolar, *range* nilai yang digunakan bervariasi mulai dari -1 sampai nilai tertinggi adalah 1

5. Langkah 4: masing masing unit pada layar tersembunyi ($Z_j=1,2,\dots,p$) merupakan penjumlahan sinyal-sinyal *input* yang telah diberi bobot beserta biasnya, dengan Persamaan 3.2:

$$Zin = V_{0j} + \sum_{i=1}^n X_{ij}V_{ij} \dots \dots \dots (3.2)$$

Untuk menghitung nilai sinyal keluaran dari unit tersembunyi, digunakan fungsi aktivasi yang sudah dipilih. Dengan Persamaan 3.3:

$$Z_j = f(Zin_j) \dots \dots \dots (3.3)$$

Kemudian sinyal dari unit tersembunyi dikirim ke setiap unit di layar keluaran

6. Langkah 5: Masing-masing unit keluaran ($Y_k, k=1,2,\dots,m$) merupakan penjumlahan sinyal-sinyal masukan yang telah diberi bobot beserta biasnya, dengan Persamaan 3.4:

$$Yin_k = W_{0k} + \sum_{j=1}^p Z_j W_{jk} \dots \dots \dots (3.4)$$

Untuk menghitung nilai sinyal keluaran dari unit keluaran, digunakan fungsi aktivasi yang sudah dipilih, dengan Persamaan 3.5:

$$Y_k = f(Y_{in_k}) \dots \dots \dots (3.5)$$

7. Propagasi mundur (*Backpropagation*)

Langkah 6: Masing-masing unit *output* ($Y_k, k=1,2,\dots,m$) menerima suatu target (pola yang diinginkan) sesuai dengan pola masukan. Untuk menghitung besar *error* antara target dengan *output*, dengan Persamaan 3.6:

$$\delta_k = (t_k - Y_k) f'(Y_{in_k}) \dots \dots \dots (3.6)$$

Seperti masukan data latih, data keluaran hasil pelatihan (t_k) juga perlu diskalakan sesuai dengan fungsi aktivasi yang digunakan. Faktor δ_k berfungsi untuk menghitung koreksi *error* yang akan dipakai dalam pembaharuan nilai w_{jk} melalui persamaan 3.7:

$$\Delta W_{jk} = \alpha \delta_k Z_j \dots \dots \dots (3.7)$$

Koreksi bias (δ_k) yang akan dipakai dalam pembaharuan nilai w_{0k} juga dihitung dengan rumus 3.8:

$$\Delta W_{0k} = \alpha \delta_k \dots \dots \dots (3.8)$$

Faktor δ_k kemudian dikirim ke layar pada langkah 7

8. Langkah 7: *input* (dari layar pada langkah 6) yang diberi bobot, dijumlahkan pada masing-masing unit tersembunyi ($Z_j, j=1,2,\dots,p$) dengan Persamaan 3.9:

$$\delta_{in_j} = \sum_{k=1}^m \delta_k W_{jk} \dots \dots \dots (3.9)$$

Agar dapat menghasilkan faktor koreksi error j , hasil dari Persamaan diatas dikalikan dengan turunan fungsi aktivasi yang digunakan melalui Persamaan 3.10:

$$\delta_j = (\delta_{in_j}) f'(Z_{in_j}) \dots \dots \dots (3.10)$$

Faktor δ_j digunakan untuk menghitung koreksi *error* (Δ_{ij}) yang akan dipakai pada pembaharuan nilai v_{ij} , dengan Persamaan 3.11:

$$\Delta V_{ij} = \alpha \delta_j X_i \dots \dots \dots (3.11)$$

Koreksi bias (Δ_{0j}) yang akan dipakai pada pembaharuan nilai v_{ij} , juga dihitung dengan rumus 3.12:

$$\Delta V_{0j} = \alpha \delta_j \dots \dots \dots (3.12)$$

9. Perubahan bobot

Langkah 8: Masing-masing unit keluaran ($y_k, k=1,2,\dots,m$) akan dipakai pada pembaharuan nilai bias dan bobot dari setiap unit tersembunyi ($j=0,1,\dots,p$) melalui Persamaan 3.13:

$$w_{jk} = w_{jk}(\text{lama}) + \Delta V_{ij} \dots \dots \dots (3.13)$$

Masing-masing unit tersembunyi ($z_j, j=1,2,\dots,p$) juga akan dipakai pada pembaharuan nilai bias dan bobot setiap unit masukan ($i=0,1,\dots,n$) melalui Persamaan 3.14:

$$V_{ij}(\text{baru}) = V_{ij}(\text{lama}) + \Delta V_{ij} \dots \dots \dots (3.14)$$

10. Langkah 9: Pemeriksaan kondisi berhenti (*stopping condition*)

Setelah pelatihan selesai dilakukan, jaringan dapat digunakan untuk melakukan klasifikasi. Dalam hal ini, hanya propagasi maju (langkah 4 dan 5) saja yang dipakai untuk menentukan keluaran jaringan. Apabila fungsi aktivasi yang digunakan buka sigmoid biner, maka langkah 4 dan 5 harus disesuaikan. Demikian juga turunannya pada langkah 6 dan 7.

-----Halaman ini sengaja dikosongkan-----

BAB IV PENGUJIAN DAN ANALISIS SISTEM

4.1 Lingkungan Uji Coba

Dalam tugas akhir ini, uji coba dilakukan dengan menggunakan komputer jinjing dengan spesifikasi Processor Intel (R) Core (TM) i3 CPU 2,0 GHz dengan memori 4 GB. Program dibuat menggunakan software Matlab versi 2015a.

4.2 Data Uji Coba

Tugas akhir ini menggunakan data pada beberapa jenis gempa pada monitoring aktivitas gunung merapi. Data didapatkan melalui kegiatan magang di kantor Balai Penelitian dan Pengembangan Teknologi Kebencanaan Geologi (BPPTKG) Yogyakarta. Informasi gempa yang didapatkan berupa kejadian gempa gunung merapi pada rekaman seismometer Pusunglondon (PUS) bulan Oktober 2009 sampai Oktober 2010. Stasiun Pusunglondon (PUS) terletak 0,5 km dari puncak Gunung Merapi. Karena lokasi seismometer yang dekat dengan puncak gunung, rekaman setiap jenis gempa dapat dibedakan dengan jelas.



1	2	3	4	5	6	7	8	9
49x1 struct	39x1 struct	38x1 struct	21x1 struct	21x1 struct	15x1 struct	14x1 struct	12x1 struct	5x1 struct

Gambar 4. 1 Kelas Gempa

Gambar 4.1 merupakan kelas gempa hasil klasifikasi yang dilakukan oleh BPPTKG. Pada kolom 1 dan 2 berisikan data gempa bukan tipe A&B, kolom nomor 4,5, dan 9 adalah gempa tipe A, dan kolom nomor 3,6,7, dan 8 adalah gempa tipe B. Pada tugas akhir ini, data dibagi menjadi 2 jenis, yaitu data pelatihan sebanyak 165 data (77%) dan 49 data (22%) digunakan untuk data pengujian. Dari 49 data yang diuji, memiliki rincian sebagai berikut:

- Data Uji Gempa Tipe A : 12 data
- Data Uji Gempa Tipe B : 19 data
- Data Uji Gempa Bukan TipeA&B : 18 data

Kemudian seluruh data uji disusun berdasarkan jenis gempa yang ditentukan secara manual. Klasifikasi secara manual didasarkan pada hasil penentuan gempa yang dilakukan oleh pihak BPPTKG.

Tabel 4. 1 Urutan Data Gempa

Urutan Data Uji	Jenis Gempa
Ke-1	Tipe A
Ke-2	Tipe A
Ke-3	Tipe A
Ke-4	Tipe A
Ke-5	Tipe A
Ke-6	Tipe A
Ke-7	Tipe A
Ke-8	Tipe A
Ke-9	Tipe A
Ke-10	Tipe A
Ke-11	Tipe A
Ke-12	Tipe A
Ke-13	Tipe B
Ke-14	Tipe B
Ke-15	Tipe B
Ke-16	Tipe B
Ke-17	Tipe B
Ke-18	Tipe B
Ke-19	Tipe B
Ke-20	Tipe B
Ke-21	Tipe B
Ke-22	Tipe B
Ke-23	Tipe B
Ke-24	Tipe B
Ke-25	Tipe B
Ke-26	Tipe B
Ke-27	Tipe B
Ke-28	Tipe B
Ke-29	Tipe B
Ke-30	Tipe B
Ke-31	Tipe B
Ke-32	Bukan Tipe A&B
Ke-33	Bukan Tipe A&B
Ke-34	Bukan Tipe A&B
Ke-35	Bukan Tipe A&B
Ke-36	Bukan Tipe A&B
Ke-37	Bukan Tipe A&B
Ke-38	Bukan Tipe A&B
Ke-39	Bukan Tipe A&B

Urutan Data Uji	Jenis Gempa
Ke-40	Bukan Tipe A&B
Ke-41	Bukan Tipe A&B
Ke-42	Bukan Tipe A&B
Ke-43	Bukan Tipe A&B
Ke-44	Bukan Tipe A&B
Ke-45	Bukan Tipe A&B
Ke-46	Bukan Tipe A&B
Ke-47	Bukan Tipe A&B
Ke-48	Bukan Tipe A&B
Ke-49	Bukan Tipe A&B

Tabel 4.1 adalah urutan data jenis gempa yang digunakan untuk pengujian sistem. Langkah selanjutnya, data uji digunakan untuk membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan klasifikasi yang dilakukan oleh sistem.

4.3 Analisis Tampilan Sistem

Pada SubBab 3.3 telah dibahas mengenai implementasi metode yang berisi proses pembangunan sistem berdasarkan algoritma yang dibuat. Bagian analisis tampilan sistem membahas mengenai bagian-bagian GUI dan fungsinya.

4.4.1. Antarmuka Menu Utama

Antarmuka menu utama menampilkan sekilas informasi judul tentang program yang dibuat.



Gambar 4. 2 Tampilan Halaman Utama Sistem

Gambar 4.2 merupakan tampilan halaman utama sistem. Terdapat 2 sub-menu yang dapat dipilih pengguna sesuai kebutuhan. Pada sub “Menu Utama” apabila ditekan akan menampilkan pilihan “Pelatihan” dan “Pengujian”

4.4.2. Antarmuka Menu Pelatihan

Antarmuka pelatihan merupakan jendela yang digunakan untuk melakukan *input* data yang dibutuhkan untuk proses pelatihan pada sistem yang dibentuk.

Bobot
 Random Load Bobot

Load Data

Hidden Layer
 Jumlah Neuron:
 Fungsi Aktivasi:

Output Layer
 Jumlah Neuron:
 Fungsi Aktivasi:

Parameter
 Training Function:
 Error Goal:
 Epochs:
 Learning Rate:

Pengolahan Data
 Path:

	Kelas Target	Kelas Keluaran
1		
2		
3		
4		

Akurasi Pelatihan
 Jumlah Data Total: Jumlah Data Salah:
 Jumlah Data Benar: Akurasi:

Gambar 4. 3 Tampilan Jendela Pelatihan

Gambar 4.3 merupakan tampilan dari jendela pelatihan. Dalam jendela pelatihan terdapat beberapa bagian yaitu “Bobot”, “*Load Data*”, “*Hidden Layer*”, “*Output Layer*”, “Parameter”, “Pengolahan Data”, dan “Akurasi Pelatihan”. Berikut ini penjelasan mengenai setiap bagian dan cara pengoperasiannya.

A. **Bobot**

Bagian ini berisi 2 tombol pilihan yaitu “*Random*” untuk memunculkan bobot jaringan secara acak dan pilihan “*Load Bobot*” untuk memanggil bobot hasil pelatihan yang pernah disimpan sebelumnya.

B. ***Load Data***

Load Data memiliki 6 tombol pilihan yaitu “*Load Data*”, “*Load Bobot*”, “*Pelatihan*”, “*Hasil Pelatihan*”, “*Simpan Jaringan*”, dan “*Simpan Bobot*”. Pada saat tombol di pilihan bobot adalah *random* maka pada bagian *load data*, pilihan *button load data* yang aktif untuk memanggil data yang digunakan untuk pelatihan. Sedangkan apabila pada bagian bobot dipilih *load bobot* maka tombol *load bobot* akan aktif yang berfungsi untuk memanggil bobot hasil pelatihan. Tombol pelatihan digunakan untuk memulai proses pelatihan sistem. Tombol hasil pelatihan digunakan untuk menampilkan hasil pelatihan pada panel akurasi pelatihan. Tombol simpan jaringan berfungsi

untuk menyimpan arsitektur jaringan berupa parameter di panel *hidden layer*, *output layer*, dan panel parameter untuk dipanggil pada saat proses pengujian. Tombol simpan digunakan untuk menyimpan bobot hasil proses pelatihan.

C. *Hidden Layer*

Panel hidden layer berguna untuk mengatur parameter yang terdapat di layar tersembunyi. Pada *edit text* jumlah neuron, pengguna dapat memasukkan banyaknya neuron yang dimunculkan pada layar tersembunyi. Semakin banyak neuron yang dimasukkan akan memperlama waktu sistem untuk melakukan proses pelatihan. *Pop up menu* fungsi aktivasi mempunyai dua pilihan yaitu sigmoid biner dan sigmoid bipolar.

D. *Output Layer*

Berdasarkan arsitektur yang dibangun pada tugas akhir ini seperti pada Gambar 3.4, jumlah neuron pada layar keluaran hanya 1. Oleh sebab itu pada bagian *output layer* yang dapat diubah pengguna hanya jenis fungsi aktivasi *output layer*. *Pop up menu* fungsi aktivasi memiliki 3 pilihan yaitu linier, sigmoid biner, dan sigmoid bipolar.

E. Parameter

Setelah mengatur layar tersembunyi dan layar keluaran, sebelum proses pelatihan dilakukan ada baiknya mengatur parameter pelatihan. Bagian parameter terdapat 4 bagian yang dapat diatur oleh pengguna. Pertama adalah *training function* yaitu fungsi untuk meminimumkan kinerja. Terdapat pilihan *traingdx* (*gradient descent* dengan momentum dan *adaptive learning rate*), *traingd* (*gradient descent*), *trainlm* (*levenberg-marquardt backpropagation*), *trainrp* (*resilent propagation*), *traingda* (*gradient descent* dengan *adaptive learning rate*). Selanjutnya adalah pengaturan *error goal* atau jumlah kesalahan paling sedikit yang membuat sistem dinyatakan selesai dilatih. *Error goal* yang dapat dipilih pengguna dari $1e-6$ (0,000001) sampai $1e-3$ (0,001). *Edit text* berikutnya yang dapat diatur adalah jumlah rangkaian langkah dalam pembelajaran baik maju (*feedforward*) maupun (*backpropagation*) atau disebut *epoch*. Parameter

terakhir yaitu laju pembelajaran yang dapat diatur oleh pengguna (biasanya menggunakan 0,1 atau 0,01).

F. Pengolahan Data

Bagian pengolahan data merupakan bagian yang menampilkan data latih. Terdapat *inactive edit text* pada bagian *path* untuk menunjukkan direktori data latih pada laptop. Tabel pada bagian pengolahan data menampilkan data latih. Kolom tabel menunjukkan data untuk satu kejadian gempa dan baris menunjukkan jumlah kejadian gempa.

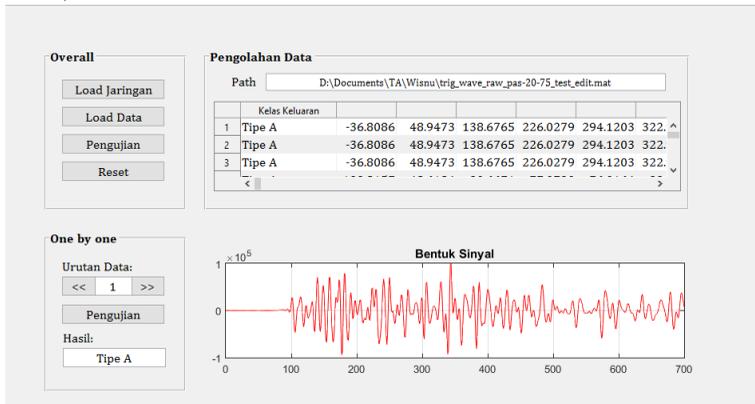
G. Akurasi Pelatihan

Akurasi pelatihan merupakan bagian yang menampilkan data hasil pelatihan setelah tombol hasil pelatihan pada *load data* ditekan. Semua *input text* bersifat *inactive* karena hanya berfungsi menampilkan data. Jumlah data total menunjukkan banyaknya data yang digunakan untuk melatih sistem. Jumlah data benar adalah banyaknya data yang memiliki kelas keluaran sama dengan kelas target. Sedangkan jumlah data salah adalah banyaknya data yang memiliki kelas keluaran berbeda dengan kelas target. Akurasi pelatihan menampilkan persentasi keakuratan sistem dalam mengenali pola yang dilatihkan. Akurasi diperoleh melalui Persamaan 4.1:

$$akurasi = \frac{jumlah\ data\ benar}{jumlah\ data\ salah} \times 100\% \dots \dots \dots (4.1)$$

4.4.3. Antarmuka Menu Pengujian

Antarmuka pengujian merupakan jendela yang digunakan untuk melakukan proses uji coba sistem. Proses pengujian dilakukan setelah proses pelatihan dilakukan.



Gambar 4. 4 Tampilan Jendela Pengujuan

Gambar 4.4 adalah tampilan jendela pengujuan setelah proses pengujuan selesai dilakukan. Terdapat 4 bagian pada jendela yaitu “Overall”, “Pengolahan Data”, “One by One”, dan “Bentuk Sinyal”. Berikut ini penjelasan mengenai setiap bagian dan cara pengoperasiannya.

A. *Overall*

Bagian *overall* berisi tombol untuk melakukan proses pengujuan. Ada 4 tombol yaitu “Load Jaringan”, “Load Data”, “Pengujuan”, dan “Reset”. Load jaringan merupakan tombol untuk memanggil arsitektur jaringan dan bobot yang dihasilkan. Arsitektur jaringan meliputi jumlah neuron, fungsi aktivasi, *epoch*, dan parameter lain yang terdapat pada proses pelatihan. Tombol load data digunakan untuk memanggil data yang akan dipakai untuk pengujuan sistem. Tombol pengujuan digunakan untuk melakukan proses pengujuan sistem. Dan tombol *reset* digunakan untuk mengatur kembali proses pengujuan.

B. Pengolahan Data

Pada bagian pengolah data, pada tulisan *path* terdapat *edit text* yang telah dinonaktifkan sehingga hanya memiliki fungsi untuk menampilkan lokasi data pengujuan diambil

pada laptop. Pada bagian pengolahan data juga menampilkan tabel data uji dimana pada kolom “Kelas Keluaran” merupakan hasil klasifikasi jenis gempa yang dilakukan oleh sistem. Baris pada tabel menunjukkan data yang digunakan untuk pengujian dan kolom menunjukkan data masukan pada sistem untuk satu kejadian gempa.

C. *One by One*

Bagian *one by one* berfungsi untuk menampilkan hasil pengujian satu per satu kejadian gempa. Pada bagian ini terdapat tombol push button kiri dan kanan untuk menggeser data gempa yang ditampilkan. *Inactive edit text* menunjukkan data gempa yang saat ini ditampilkan. Tombol “Pengujian” memiliki fungsi untuk melakukan pengujian pada satu data gempa dan pada *inactive edit text* menampilkan jenis gempa untuk satu data yang dipilih oleh pengguna.

D. Bentuk Sinyal

Bagian bentuk sinyal akan berisi sebuah tabel yang menampilkan plot bentuk gempa berdasarkan data pada tabel pengolahan data untuk satu kejadian gempa. Bentuk sinyal akan ditampilkan ketika tombol pengujian pada bagian *one by one* ditekan.

4.4 Pelatihan dan Pengujian Jaringan Saraf Tiruan

Pelatihan dan pengujian pada jaringan saraf tiruan meliputi proses pelatihan dan proses pengujian pada metode JST propagasi balik yang digunakan.

4.4.1 Proses Pelatihan Data Seismik Dengan JST Propagasi Balik

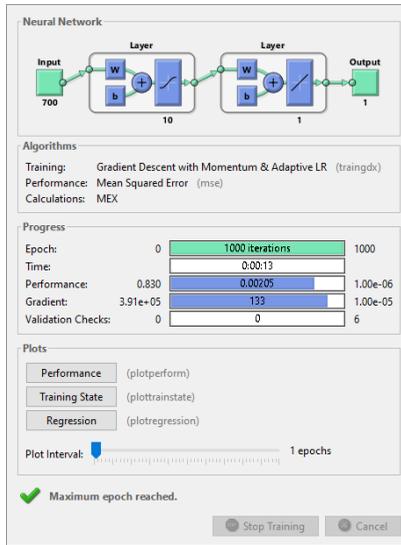
Dalam tugas akhir ini, metode JST Propagasi Balik dimulai dengan proses pelatihan terhadap data set pelatihan untuk mendapatkan bobot terbaik yang sesuai target. Bobot ini kemudian akan digunakan pada proses pengujian untuk mengetahui jenis gempa dalam metode pemantauan seismik. Proses pelatihan dilakukan pada seluruh jenis gempa yaitu gempa dalam (VA), gempa dangkal (VB) dan bukan kedua jenis gempa. Data yang akan dilatih merupakan jenis data yang berupa bilangan bulat jumlah kejadian gempa. Data yang didapatkan berjumlah 214 set data yang disimpan dalam bentuk file matlab(.mat). Dalam tahap

ini memerlukan pengaturan parameter dalam pembuatan sebuah jaringan seperti jumlah hidden layer, jumlah epoch, nilai *learning rate*.

Seperti dijelaskan pada bagian 3.2.1 mengenai desain metode pelatihan JST propagasi balik, pada sistem ini terdapat 700 neuron pada lapisan masukan dan 1 neuron pada lapisan keluaran. Proses pelatihan untuk mengklasifikasi gempa dilakukan melalui 165 set data. Dari 165 set data tersebut terdapat 35 data latih dengan kelas target gempa tipe A, 60 data latih dengan kelas target tipe B dan 70 data latih dengan kelas target bukan kedua jenis gempa. Awalnya jaringan dibangun dengan parameter berupa:

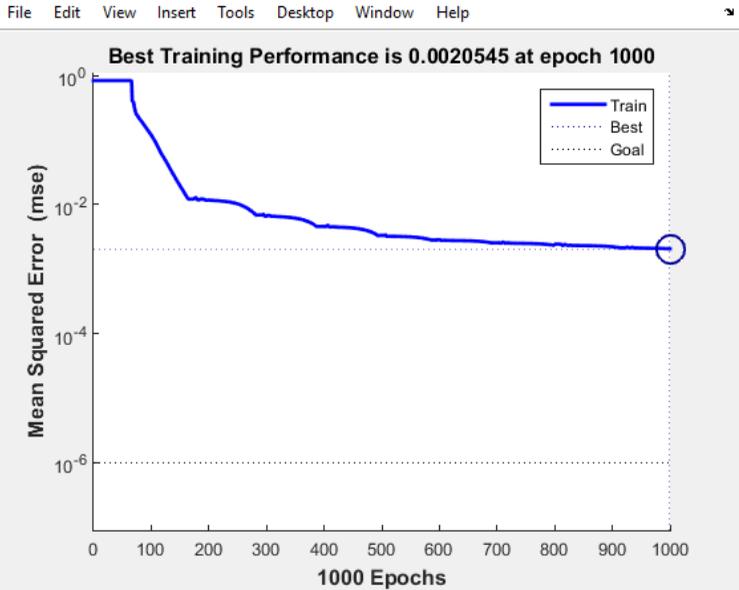
Jumlah neuron pada <i>hidden layer</i>	: 10
Fungsi aktivasi pada <i>hidden layer</i>	: sigmoid bipolar
Jumlah neuron pada <i>output layer</i>	: 1
Fungsi aktivasi pada <i>output layer</i>	: linier
Fungsi Pelatihan	: traingdx
<i>Error Goal</i>	: 1e-6
<i>Epochs</i>	: 1000
<i>Learning Rate</i>	: 0.1

Kemudian parameter tersebut dimasukkan ke dalam sistem untuk dilakukan pelatihan, setelah itu proses pelatihan dijalankan.



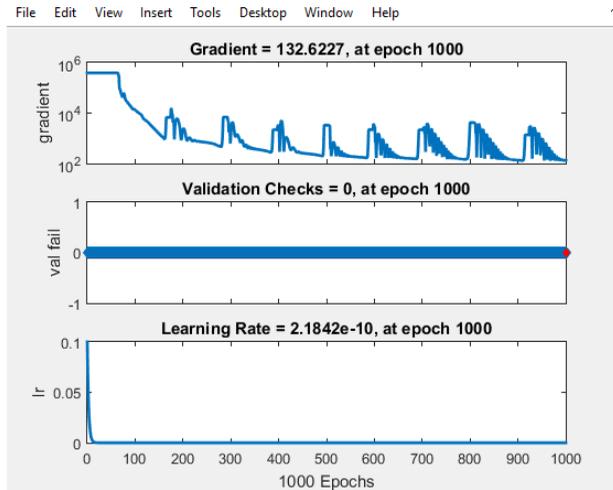
Gambar 4. 5 Tampilan Proses Pelatihan Sistem

Gambar 4.5 merupakan tampilan dari proses pelatihan sistem. Dapat dilihat bahwa sistem berhenti setelah mencapai *epoch* ke 1000 dan belum mencapai *error goal* yang ditentukan oleh pengguna yaitu $10e-6$. Pada percobaan ini, jenis pelatihan yang digunakan adalah *Gradient Descent with Momentum and Adaptive Learning* (traingdx). *Performance*, yang menyatakan metode untuk menghentikan proses jika sudah mendekati real adalah *Mean Squared Error* (MSE). *Epoch*, menyatakan jumlah perulangan pelatihan. *Time*, menyatakan waktu yang ditempuh oleh Matlab untuk melakukan pelatihan. *Performance*, menyatakan kualitas hasil pelatihan dimana semakin mendekati 0 maka kualitasnya semakin baik. *Gradient* merupakan kemiringan antara satu iterasi dengan iterasi berikutnya. *Validation Check* digunakan untuk memeriksa proses pelatihan mengarah ke arah yang tepat atau menyimpang. Pada bagian plot terdapat 3 tombol yaitu “*Performance*”, “*Training State*”, dan “*Regression*”. Ketiga tombol berguna untuk menampilkan grafik hasil pembelajaran.



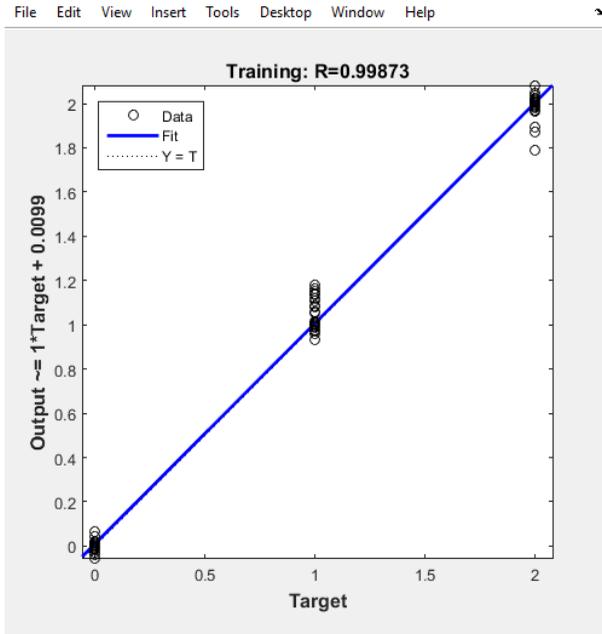
Gambar 4. 6 *Performance* Pelatihan

Gambar 4.6 merupakan grafik *performance* pelatihan. Grafik *performance* merupakan grafik hubungan antara *epoch* dengan *Mean Squarred Error* (MSE). Bentuk grafik yang menurun menunjukkan proses pelatihan berhasil dengan *error* terbaik adalah 0,0020545 pada saat *epoch* ke 1000



Gambar 4. 7 Tampilan Dari *Training State*

Gambar 4.7 adalah tampilan dari *training state*. Terlihat ada grafik hubungan antara *epoch* dengan *gradient*, grafik hubungan antara *epoch* dengan *val fail*, dan grafik hubungan antara *epoch* dengan Lr. Nilai gradien pada proses pelatihan ini sebesar 132,6227. Nilai *validation check* sebesar 0 dan nilai *learning rate* sebesar $2,1842e-10$.



Gambar 4. 8 Tampilan *Plot Regression*

Gambar 4.8 merupakan tampilan *plot regression*. Pada Gambar target dengan keluaran nilainya berkisar antara 0 – 2. Hal ini terjadi karena keluaran dengan target dilakukan proses normalisasi data sehingga data keluaran dan target berada pada *range* 0 – 2. Pada plot regresi ini mempunyai korelasi sebesar 0,998 yang menunjukkan korelasi yang baik antara data keluaran dengan target.

Bobot

Random Load Bobot

Load Data

Load Data	Load Bobot
Pelatihan	Hasil Pelatihan
Simpan Jaringan	Simpan Bobot

Hidden Layer

Jumlah Neuron:

Fungsi Aktivasi:

Output Layer

Jumlah Neuron:

Fungsi Aktivasi:

Parameter

Training Function:

Error Goal:

Epochs:

Learning Rate:

Pengolahan Data

Path:

	Kelas Target	Kelas Keluaran				
1	Type A	Type A	-39.1285	-15.5951	4.0976	7.30
2	Type A	Type A	-77.5957	-78.1658	-72.8554	-62.11
3	Type A	Type A	-98.4575	-59.6536	-11.8286	19.37
4	Type A	Type A	-17.0232	-26.9473	-19.2056	1.05
5	Type A	Type A	104.6794	133.8871	181.1749	198.49
6	Type A	Type A	-163.8887	-140.0820	-128.7956	-128.96
7	Type A	Type A	-42.2228	-52.3233	-27.9280	31.73
8	Type A	Type A	99.5127	28.9691	-93.5864	-190.56
9	Type A	Type A	-112.3105	-160.3248	-208.7328	-221.65
10	Type A	Type A	104.0585	34.7502	-49.2726	-120.09
11	Type A	Type A	133.6085	146.1820	143.0929	116.07
12	Type A	Type A	16.0845	51.5482	72.8083	87.74

Akurasi Pelatihan

Jumlah Data Total	<input type="text" value="165"/>	Jumlah Data Salah	<input type="text" value="0"/>
Jumlah Data Benar	<input type="text" value="165"/>	Akurasi	<input type="text" value="100%"/>

Gambar 4. 9 Hasil Proses Pelatihan Sistem

Setelah parameter diatur dan dilakukan proses pelatihan, hasil yang diperoleh ditampilkan pada Gambar 4.9 yaitu sistem dengan akurasi 100%.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	-8.8231e-07	-3.1840e-05	7.0002e-06	2.5250e-05	5.5532e-05	2.3330e-05	-8.8664e-05	9.7546e-05	-5.7990e-05	-5.7738e-05	-1.2844e-06	6.0357e-05	-3.7425e-05	3.3315e-05	3.3390e-05	-6.0305e-06	4.8815e-06	1.8735e-05
2	-4.4235e-05	4.1421e-05	-7.4630e-06	-2.7916e-05	-1.2791e-05	-2.4570e-05	-5.0598e-05	1.1077e-04	1.7079e-05	4.6035e-05	4.3070e-05	2.0249e-05	3.2515e-05	-1.1502e-05	8.0316e-05	-4.3377e-05	-7.8017e-05	1.6705e-05
3	8.8544e-05	-7.7460e-06	6.6031e-05	2.8529e-05	1.0350e-05	-1.4162e-05	7.2070e-05	2.5294e-05	2.9304e-05	-6.4930e-05	4.2701e-05	8.8662e-05	2.2027e-05	-1.9514e-05	7.0744e-05	-4.9515e-05	-5.8434e-06	6.6867e-05
4	8.5614e-05	1.6684e-05	8.8438e-05	7.7865e-05	9.4854e-05	5.3345e-05	1.2583e-05	1.0294e-04	-2.6762e-05	-6.7395e-05	-5.8059e-05	6.1395e-05	-1.1761e-05	-6.6205e-05	-6.2313e-05	-4.4286e-05	4.9817e-05	7.7312e-05
5	1.7893e-05	-4.3145e-06	8.9402e-05	-4.8334e-05	3.7474e-05	-5.9903e-05	-6.2520e-05	-3.8120e-05	-7.9540e-05	-4.2440e-05	-1.2401e-05	-7.6150e-05	9.2007e-05	8.3230e-05	6.8890e-05	7.4760e-05	5.7270e-05	-1.1547e-05
6	-9.7862e-05	1.0113e-05	4.4738e-05	-4.2905e-05	9.2706e-05	1.9505e-05	-1.6068e-05	9.1046e-05	-8.9074e-05	2.0377e-05	-8.5006e-05	9.3805e-05	6.3325e-05	-2.8018e-05	3.9904e-05	-1.8602e-05	1.0756e-05	1.5746e-05
7	7.8606e-05	-8.3120e-05	-5.7300e-05	-2.5668e-05	-8.0061e-05	-6.8660e-06	4.8959e-05	-1.1417e-04	-7.8917e-05	8.4743e-05	2.9465e-05	-4.3062e-05	-6.0518e-05	6.2442e-05	7.8080e-05	2.1434e-05	-4.1047e-05	7.2461e-05
8	3.5911e-05	1.6805e-05	-8.8130e-05	5.7407e-06	7.8402e-05	-3.7718e-05	9.5847e-05	-3.7984e-06	-7.6631e-05	8.8178e-05	5.2953e-05	8.5917e-05	-7.0248e-05	-8.4893e-05	4.0669e-05	-7.1421e-05	5.1255e-05	6.2380e-05
9	-4.3884e-05	-4.6890e-05	-5.8304e-05	4.4005e-05	-8.7199e-05	6.0894e-05	1.5402e-05	-8.0579e-05	-8.6813e-05	-2.7416e-05	8.8542e-05	3.3890e-05	-3.1776e-06	-1.4899e-05	5.3324e-05	-4.2304e-05	-8.6144e-06	1.6146e-05
10	9.7424e-05	-1.2518e-05	1.4736e-05	-4.0208e-05	-7.4166e-05	1.2836e-05	8.8413e-05	-4.2007e-05	8.4682e-05	-3.7048e-05	3.3737e-05	-2.2154e-05	1.9795e-05	-7.8201e-05	-6.2750e-05	3.5019e-05	1.0950e-06	2.8034e-05

Gambar 4. 10 Bobot Pada *Hidden Layer*

Pada Gambar 4.10 terdapat matrik berukuran 10x700 dimana pada bagian baris berjumlah 10 merupakan jumlah neuron pada hidden layer yang telah diatur sebelum melakukan pelatihan. 700 kolom menandakan jumlah neuron pada masukan data. Matrik 1x1 menandakan besarnya bobot pada neuron ke-1 di layar tersembunyi dari neuron masukan ke-1. Begitu seterusnya.

	1	2	3	4	5	6
1	3.8628					
2	2.9045					
3	-1.9563					
4	-1.2724					
5	-0.1257					
6	-1.4397					
7	1.5753					
8	1.8774					
9	-2.8018					
10	1.3650					

Gambar 4. 11 Bobot Bias Untuk *Hidden Layer*

Gambar 4.11 merupakan bobot dari neuron bias di lapisan masukan menuju ke neuron di lapisan tersembunyi. Karena hanya terdapat 1 neuron bias dan 10 neuron di lapisan tersembunyi, maka sistem menghasilkan bobot berupa matrik 10x1

	1	2	3	4	5	6	7	8	9	10
1	0.7110	-0.0514	-0.3515	-0.0799	0.3483	0.1481	0.6857	-0.8484	-0.4803	0.3751

Gambar 4. 12 Bobot Keluaran

Bobot keluaran merupakan bobot pada garis penghubung dari lapisan tersembunyi menuju ke lapisan keluaran. Gambar 4.4 menampilkan besarnya bobot dari lapisan tersembunyi menuju neuron keluaran. Elemen matrik 1x1 menunjukkan besarnya bobot dari neuron ke-1 di lapisan tersembunyi menuju ke neuron keluaran.

	1	2	3	4	5	6
1	-0.1350					

Gambar 4. 13 Bobot Bias Keluaran

Pada Gambar 4.13 merupakan matrik 1x1 dikarenakan neuron bias yang berasal dari lapisan tersembunyi hanya ada 1 dan neuron *output* juga hanya 1.

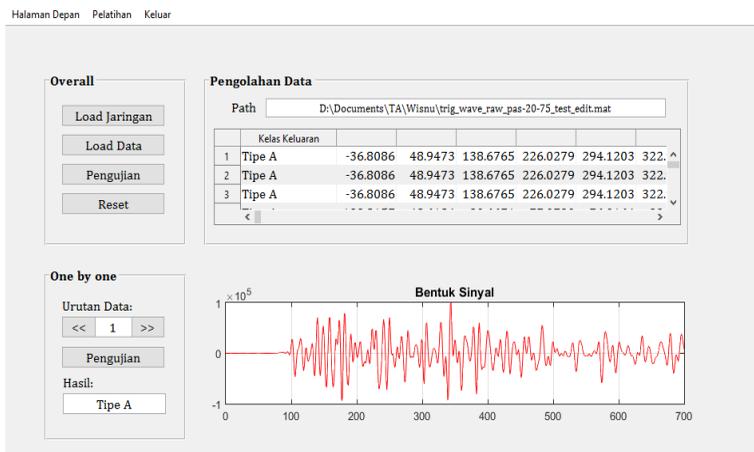
Dikarenakan sistem telah mencapai akurasi 100%, langkah selanjutnya adalah menyimpan bobot yang dihasilkan setelah pengaturan parameter dan dilakukan pelatihan. Bobot bobot tersebut digunakan pada saat proses pengujian data. Arsitektur jaringan yang berisi

parameter pelatihan juga disimpan untuk dipanggil pada proses pengujian.

4.4.2 Proses Pengujian Data Seismik Dengan JST Propagasi Balik

Berdasarkan proses pelatihan yang dilakukan, didapatkan bobot yang akan digunakan pada proses pengujian. Proses pengujian diawali dengan menyiapkan set data pengujian. Pada tugas akhir ini digunakan 49 set data uji yang terdiri dari gempa A, gempa B, dan bukan gempa A&B.

Pada proses pengujian, bobot yang didapatkan pada proses pelatihan dipanggil untuk digunakan pada set data pengujian. Hasil yang didapatkan berupa jenis gempa dari bobot terbaik yang didapatkan pada proses pelatihan.



Gambar 4. 14 Contoh Tampilan Hasil Pengujian Data

Gambar 4.14 menunjukkan proses pengujian data. Pada data ke-1, terdapat suatu sinyal gempa yang ditampilkan pada kolom bentuk sinyal. Bentuk sinyal tersebut dijadikan masukan pada arsitektur jaringan (700 neuron masukan). Kemudian melalui 10 neuron di lapisan tersembunyi dengan dikali bobot yang didapat sebelumnya. Dari lapisan tersembunyi, data diteruskan menuju neuron keluaran dengan dikali bobot pada lapisan tersembunyi. Di neuron keluaran tersebut, data diklasifikasikan menjadi 3 kelas keluaran yaitu gempa tipe A, gempa tipe B

dan bukan gempa tipe A dan B. Data gempa ke-1 setelah diuji dengan bobot terbaik yang dihasilkan pada proses pelatihan didapatkan hasil berupa gempa dengan tipe B (dangkal). Ini menunjukkan bahwa untuk bentuk sinyal yang menyerupai data gempa 1 maka dinyatakan gempa tipe B.

4.5 Analisis Parameter Pelatihan JST

Analisis parameter pelatihan JST dilakukan untuk mendapatkan bobot terbaik dengan cara mengubah parameter yang digunakan pada proses pelatihan atau dengan mengulangi proses pelatihan.

4.5.1. Analisis Pengujian Data Uji Seismik

Sistem diatur dengan parameter yang ditulis pada SubBab 4.4.1 dan pada proses pelatihan didapatkan akurasi sistem sebesar 100%. Kemudian dilakukan pengujian sistem yang dibangun dengan parameter tersebut, diperoleh hasil yaitu:

Tabel 4. 2 Hasil Pengujian

Urutan Data Uji	Kelas Target	Hasil Pengujian
Ke-1	Tipe A	Tipe A
Ke-2	Tipe A	Tipe A
Ke-3	Tipe A	Tipe A
Ke-4	Tipe A	Tipe A
Ke-5	Tipe A	Tipe A
Ke-6	Tipe A	Tipe A
Ke-7	Tipe A	Tipe A
Ke-8	Tipe A	Tipe A
Ke-9	Tipe A	Tipe A
Ke-10	Tipe A	Tipe A
Ke-11	Tipe A	Tipe A
Ke-12	Tipe A	Tipe A
Ke-13	Tipe B	Tipe B
Ke-14	Tipe B	Tipe B
Ke-15	Tipe B	Tipe B
Ke-16	Tipe B	Bukan Tipe A&B
Ke-17	Tipe B	Tipe B
Ke-18	Tipe B	Tipe B
Ke-19	Tipe B	Tipe B
Ke-20	Tipe B	Tipe A
Ke-21	Tipe B	Tipe B

Urutan Data Uji	Kelas Target	Hasil Pengujian
Ke-22	Tipe B	Tipe B
Ke-23	Tipe B	Tipe B
Ke-24	Tipe B	Tipe B
Ke-25	Tipe B	Tipe B
Ke-26	Tipe B	Tipe A
Ke-27	Tipe B	Tipe B
Ke-28	Tipe B	Tipe B
Ke-29	Tipe B	Tipe B
Ke-30	Tipe B	Tipe B
Ke-31	Tipe B	Tipe B
Ke-32	Bukan Tipe A&B	Bukan Tipe A&B
Ke-33	Bukan Tipe A&B	Tipe A
Ke-34	Bukan Tipe A&B	Bukan Tipe A&B
Ke-35	Bukan Tipe A&B	Bukan Tipe A&B
Ke-36	Bukan Tipe A&B	Bukan Tipe A&B
Ke-37	Bukan Tipe A&B	Bukan Tipe A&B
Ke-38	Bukan Tipe A&B	Bukan Tipe A&B
Ke-39	Bukan Tipe A&B	Bukan Tipe A&B
Ke-40	Bukan Tipe A&B	Bukan Tipe A&B
Ke-41	Bukan Tipe A&B	Bukan Tipe A&B
Ke-42	Bukan Tipe A&B	Bukan Tipe A&B
Ke-43	Bukan Tipe A&B	Bukan Tipe A&B
Ke-44	Bukan Tipe A&B	Bukan Tipe A&B
Ke-45	Bukan Tipe A&B	Bukan Tipe A&B
Ke-46	Bukan Tipe A&B	Bukan Tipe A&B
Ke-47	Bukan Tipe A&B	Bukan Tipe A&B
Ke-48	Bukan Tipe A&B	Bukan Tipe A&B
Ke-49	Bukan Tipe A&B	Bukan Tipe A&B

Tabel 4.1 merupakan tabel hasil pengujian. kolom urutan data menunjukkan urutan data uji. Kolom kelas target berisi jenis gempa yang seharusnya diklasifikasi sistem. Kolom hasil pengujian berisi hasil klasifikasi yang dilakukan oleh sistem. Klasifikasi dinyatakan benar apabila hasil pengujian sama dengan kelas target. Jumlah data yang benar digunakan untuk menghitung akurasi sistem. Dari Tabel 4.1 yaitu hasil pengujian sistem, diperoleh akurasi sebesar:

$$akurasi = \frac{\text{keseluruhan jumlah data} - \text{jumlah data salah}}{\text{keseluruhan data}} \times 100\% \dots (4.2)$$

$$akurasi \text{ total} = \frac{49-4}{49} \times 100\% = 91,83\% \dots (4.3)$$

$$\text{akurasi VTA} = \frac{12-0}{12} \times 100\% = 100\% \dots \dots \dots (4.4)$$

$$\text{akurasi VTB} = \frac{19-3}{19} \times 100\% = 84,21\% \dots \dots \dots (4.5)$$

$$\text{akurasi bukan A\&B} = \frac{18-1}{18} \times 100\% = 94,44\% \dots \dots \dots (4.6)$$

Dari pengujian yang dilakukan menggunakan bobot pelatihan dengan akurasi latih sebesar 100%, didapatkan hasil pengujian sistem dengan akurasi sebesar 91,83%, dimana sistem dapat mengenali gempa tipe A sebesar 100%, gempa tipe B sebesar 84,21%, dan gempa bukan tipe A&B sebesar 94,44%.

4.5.2. Analisis Perulangan Pelatihan

Dengan parameter yang sama seperti pada SubBab 4.4.1, proses pelatihan dilakukan secara berulang.

Tabel 4.3 Perulangan Proses Pelatihan Dengan Parameter Tetap

Nama Jaringan	Proses Pelatihan		Proses Pengujian	
	Jumlah Data Salah	Akurasi	Jumlah Data Salah	Akurasi
10	0	100%	9	81,63%
10a	1	99,39%	10	79,59%
10b	0	100%	10	79,59%
10c	4	97,57%	7	85,71%
10d	3	98,18%	12	75,51%
10e	0	100%	4	91,83%

Tabel 4.3 merupakan hasil yang diperoleh dari perulangan proses pelatihan dengan parameter yang sama. Kolom nama jaringan menunjukkan proses pelatihan berlangsung dimana nama jaringan 10 adalah proses pelatihan pertama, 10a merupakan proses pelatihan kedua, dan seterusnya. Pada kolom proses pelatihan berisi hasil dari pelatihan yang dilakukan. Untuk proses pelatihan pertama (jaringan 10) diperoleh akurasi pelatihan sebesar 100%. Dengan parameter yang sama, pada proses pelatihan yang dilakukan lagi (jaringan 10a) didapatkan akurasi pelatihan sebesar 99,39%. Kolom pengujian berisi hasil pengujian yang dilakukan setelah proses pelatihan. Pada jaringan 10 (proses pelatihan pertama) dengan akurasi pelatihan 100% diperoleh akurasi pengujian sebesar 81,63%. Parameter yang sama dapat mengubah akurasi pada proses pelatihan jika dilakukan secara berulang. Ini dikarenakan bobot yang dihasilkan pada setiap jaringan berbeda. Jaringan 10 dan jaringan

10b memiliki bobot yang berbeda meskipun memiliki akurasi pelatihan yang sama yaitu 100% sehingga pada proses pengujian memiliki akurasi yang berbeda. Hasil terbaik adalah jaringan 10e.

4.5.3. Analisis Perubahan *Epoch*

Dengan menggunakan parameter yang terdapat pada SubBab 4.4.1, parameter *epoch* diubah dari intercal 10 sampai 10000 dengan kenaikan *epoch* 10 kali.

Tabel 4. 4 Perubahan *Epoch* Dengan Parameter Tetap

Jumlah <i>Epoch</i>	Waktu Pelatihan	Proses Pelatihan		Proses Pengujian	
		Jumlah Data Salah	Akurasi	Jumlah Data Salah	Akurasi
10	00.00	130	21,21%	38	22,44%
100	00.01	27	83,63%	14	71,42%
1000	00.12	0	100%	4	91,83%
10000	01.50	0	100%	12	75,51%

Tabel 4.4 merupakan perubahan *epoch* dengan parameter tetap. Kolom jumlah *epoch* merupakan banyaknya *epoch* (jumlah perulangan proses *feedforward* dan JST propagasi balik). Pada baris pertama, *epoch* yang diberikan adalah 10 dan menghasilkan akurasi pelatihan 21,21% dan akurasi pengujian 22,44%. Jumlah *epoch* ditambah menjadi 100 dan menghasilkan akurasi pelatihan 83,63% dan akurasi pengujian 71,42%. *Epoch* kembali dinaikkan menjadi 1000 dan menghasilkan akurasi pelatihan 100% dalam waktu 12 sekon. Pada *epoch* 10000, akurasi pelatihan sebesar 100% dengan waktu pelatihan 1 menit 50 sekon. Jaringan akan memiliki akurasi yang optimal dengan jumlah *epoch* 1000 yaitu akurasi pengujian sebesar 91,83%. Dari Tabel 4.1, semakin banyak *epoch* diberikan akan membutuhkan waktu yang lama untuk melatih jaringan.

4.5.4. Analisis Perubahan Jumlah Neuron Pada Layar Tersembunyi

Jumlah neuron pada layar tersembunyi merupakan sebuah variabel bebas. Dengan parameter yang sama seperti pada SubBab 4.4.1, jumlah neuron diatur untuk mendapat akurasi pengujian terbaik

Tabel 4. 5 Perubahan Jumlah Neuron Pada Unit Tersembunyi

Jumlah Unit Tersembunyi	Proses Pelatihan		Proses Pengujian	
	Jumlah Data Salah	Akurasi	Jumlah Data Salah	Akurasi
2	130	21,21%	37	24,48%
4	10	93,93%	12	75,51%
6	7	95,75%	12	75,51%
8	114	30,90%	31	36,73%
10	0	100%	4	91,83%

Tabel 4.5 merupakan hasil yang didapat dengan mengubah banyaknya neuron pada unit tersembunyi. Pada kolom jumlah unit tersembunyi berisi jumlah neuron yang terdapat di unit tersembunyi. Berdasarkan penelitian sebelumnya dimana jumlah neuron ini diubah dari 2 sampai 10. Kolom proses pelatihan menunjukkan hasil dari perubahan jumlah neuron terhadap akurasi pelatihan. Pada jumlah unit 2, memiliki akurasi pelatihan sebesar 21,21%, dengan jumlah unit dinaikkan menjadi 10 maka akurasinya 100%. Kolom proses pengujian menunjukkan hasil pada proses pengujian. Dengan jumlah unit 2, akurasi pengujian yang didapat sebesar 24,48%. Ketika unit diubah menjadi 10, akurasi pengujian turun menjadi 91,83%. Dari perubahan unit pada layar tersembunyi, didapatkan hasil paling optimal dengan unit tersembunyi 10.

BAB V

KESIMPULAN DAN SARAN

Pada bab ini menjelaskan kesimpulan yang diperoleh pada tugas akhir dan saran untuk pengembangan kedepannya

5.1. Kesimpulan

Kesimpulan yang diperoleh melalui pembuatan tugas akhir ini adalah :

1. Jaringan saraf tiruan dapat digunakan untuk mengenali pola dengan cara melatih jaringan menggunakan pola lain (dengan bentuk pola menyerupai)
2. Parameter jaringan yang sesuai untuk mengklasifikasikan jenis gempa yaitu:
Jumlah neuron pada *hidden layer* : 10
Fungsi aktivasi pada *hidden layer* : sigmoid bipolar
Jumlah neuron pada *output layer* : 1
Fungsi aktivasi pada *output layer* : linier
Fungsi Pelatihan : traingdx
Error Goal : 1e-6
Epochs : 1000
Learning Rate : 0.1
3. Hasil dari klasifikasi gempa yang dilakukan oleh jaringan syaraf tiruan dapat diklasifikasikan sesuai dengan kelas jenis gempa dengan akurasi rata-rata 91,83%, dimana akurasi pengenalan gempa tipe A sebesar 100%, akurasi gempa tipe B sebesar 84,21%, akurasi pengenalan gempa bukan tipe A&B sebesar 94,44%
4. Parameter yang dimasukkan pada proses pelatihan mempengaruhi kemampuan sistem. Semakin tinggi nilai yang diberikan (jumlah neuron, jumlah *epoch*) maka proses pelatihan semakin lama dan belum tentu menghasilkan akurasi klasifikasi yang tinggi.
5. Pengulangan proses pelatihan dengan parameter yang sama dapat mempengaruhi akurasi klasifikasi. Karena bobot yang dihasilkan pada setiap pengulangan proses berbeda. Oleh karena itu, bobot yang menghasilkan akurasi paling tinggi harus disimpan untuk dijadikan acuan pada proses pengujian data.

5.2. Saran

Berdasarkan hasil pengujian sistem, dapat disarankan:

1. Klasifikasi jenis gempa lain (multifase, guguran, tektonik) dapat dilakukan oleh sistem
2. Tambahan data dari pos stasiun pengamatan lainnya

DAFTAR PUSTAKA

- [1] *Gunung Merapi*, pembahasan mengenai Gunung Merapi, <http://www.vsi.esdm.go.id/index.php/gunungapi/data-dasar-gunungapi/542-g-merapi> (diakses 12 April 2018)
- [2] Ariyadi Nugoho Susilo dan Iwan Rudiarto, "Analisis Tingkat Resiko Erupsi Gunung Merapi Terhadap Permukiman di Kabupaten Klaten", *Jurnal Teknik PWK vol: III no: 34-49*, 2014.
- [3] Agus Budi Santoso, "*The Seismic Activity Associated With The Large 2010 Eruption Of Merapi Volcano, Java: Source Location, Velocity Variation, And Forecasting*", *Thesis*, Université Grenoble Alpes, Grenoble, 2006
- [4] *Informasi Merapi Pemantauan*, pembahasan tentang informasi pemantauan Merapi, <http://merapi.bgl.esdm.go.id/pub/page.php?idf=11> (diakses 12 April 2018)
- [5] Indyo Pratomo, "Klasifikasi Gunung Api Aktif Indonesia, Studi Kasus dari beberapa Letusan Gunung Api dalam Sejarah", *Jurnal Geologi Indonesia, vol: I no: 209-227*, 2006.
- [6] Budi Joko Purnomo, "Analisa Deformasi Kawasan Vulkanik Aktif dengan menggunakan Data GPS September 2012-Maret 2014 (Studi Kasus: Gunung Merapi, DIY)", *Tugas Akhir*, Program S1 Teknik Geomatika FTSP-ITS, Surabaya, 2014
- [7] Sepry Dawid, Ferdy, dan Guntur Pasau, "Penentuan Lokasi Pergerakan Magma Gunung Api Soputan Berdasarkan Studi Sebaran Hiposenter Gempa Vulkanik Periode Mei 2013-Mei 2014", *Jurnal Ilmiah Sains vol: XV no: 88-93*, 2015
- [8] Cynthia Dorothy Hutapea, "Simulasi Sensor Seismometer Horizontal Menggunakan MEMS Accelerometer Berbasis Arduino UNO", *Skripsi*, Program S1 Fisika FMIPA-USU, Medan, 2017
- [9] Charles Ammon, *Gelombang, Seismogram, dan Seismometer*, pembahasan tentang seismogram dan seismometer, <http://eqseis.geosc.psu.edu/~cammon/HTML/Classes/IntroQuakes/Notes/seismometers.html> (diakses pada 23 Juni 2018)

- [10] Kiki dan Sri Kusumadewi, "Jaringan Saraf Tiruan Dengan Metode Backpropagation Untuk Mendeteksi Gangguan Psikologi", *Paper Media Informatika vol: II no: 1-11*, 2004
- [11] Paramitha Nerisafitra, "Deteksi Anomali Pemantauan Aktivitas Gunung Merapi Menggunakan Kombinasi Metode Jaringan Syaraf Tiruan Propagasi Balik dan Logika Fuzzy", *Thesis*, Program S2 Teknik Informatika FTIF-ITS, Surabaya, 2015
- [12] Amir Tjolleng, *Pengantar Pemrograman MATLAB*, Elex Media Komputindo, Jakarta, 2017
- [13] Gunaidi Abdia Away, *The Shortcut Of MATLAB Programming*, Penerbit Informatika, Bandung, 2014
- [14] Jok Jek Siang, *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan MATLAB*, Penerbit ANDI, Yogyakarta, 2004
- [15] Antonius Ratdomopurbo dan Georger Poupinet, "An overview of the seismicity of Merapi volcano (Java, Indonesia) 1983–1994", *Journal of Volcanology and Geothermal Research vol: 100 no: 193-214*, 2000.

LAMPIRAN A

A1. Dokumentasi Kegiatan Di BPPTKG Yogyakarta



A2. Perekaman Data Seismometer Oleh Seismograf



A3. Perakaman Data Seismometer Secara Digital



A4. Rekaman Gempa Pada Erupsi Gunung Merapi Tahun 2010

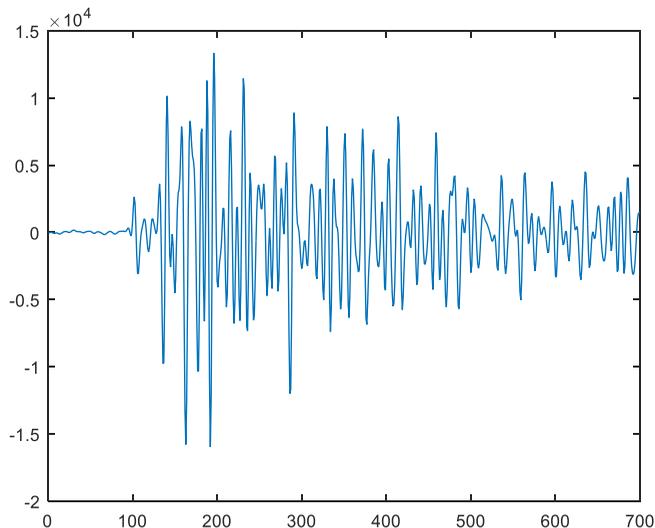


LAMPIRAN B

B.1 Cuplikan Data Kejadian Gempa Tipe A

Column 1 through 16	-0.0039	-0.0016	0.0004	0.0007	-0.0012	-0.0044	-0.0070	-0.0073	-0.0055	-0.0039	-0.0047	-0.0079	-0.0113	-0.0124	-0.0104	-0.0043
Column 17 through 32	-0.0018	0.0020	0.0047	0.0063	0.0067	0.0058	0.0037	0.0039	-0.0008	-0.0000	0.0036	0.0090	0.0139	0.0145	0.0141	0.0135
Column 33 through 48	0.0100	0.0071	0.0055	0.0032	0.0044	0.0051	0.0035	0.0008	-0.0019	-0.0030	-0.0017	0.0012	0.0042	0.0041	0.0070	0.0077
Column 49 through 64	0.0084	0.0083	0.0042	0.0020	-0.0031	-0.0068	-0.0073	-0.0044	-0.0004	0.0033	0.0050	0.0045	0.0026	-0.0002	-0.0035	-0.0076
Column 65 through 80	-0.0121	-0.0134	-0.0145	-0.0143	-0.0088	-0.0044	0.0005	0.0044	0.0047	0.0047	0.0043	-0.0003	-0.0057	-0.0102	-0.0123	-0.0115
Column 81 through 96	-0.0080	-0.0028	0.0022	0.0059	0.0075	0.0078	0.0079	0.0087	0.0092	0.0083	0.0045	0.0070	0.0138	0.0239	0.0336	0.0244
Column 97 through 112	-0.0030	-0.0265	-0.0083	0.0738	0.1808	0.2429	0.2134	0.0390	-0.1724	-0.3054	-0.3057	-0.2098	-0.0945	-0.0186	0.0225	0.0519
Column 113 through 128	0.0826	0.1015	0.0867	0.0318	-0.0445	-0.1160	-0.1459	-0.1168	-0.0396	0.0440	0.0950	0.1015	0.0754	0.0428	0.0115	-0.0091
Column 129 through 144	0.0140	0.1151	0.2461	0.3593	0.2479	-0.1152	-0.4096	-0.9745	-0.9715	-0.5193	0.1976	0.8165	1.0152	0.7345	0.2190	-0.1742
Column 145 through 160																

B.2. Cuplikan Hasil Plot Kejadian Gempa Tipe A



B3. Cuplikan Cuplikan Data Kejadian Gempa Tipe B

```

Column 1 through 14
-0.0011 -0.0142 -0.0155 -0.0081 -0.0004 0.0014 -0.0004 -0.0012 0.0034 0.0109 0.0142 0.0142 0.0126 0.0100 0.0111 0.0147

Column 17 through 32
0.0160 0.0108 -0.0008 -0.0134 -0.0211 -0.0201 -0.0123 -0.0034 0.0004 -0.0020 -0.0074 -0.0094 -0.0039 0.0074 0.0192 0.0241

Column 33 through 48
0.0184 0.0040 -0.0127 -0.0234 -0.0229 -0.0121 0.0021 0.0114 0.0104 -0.0005 -0.0141 -0.0297 -0.0340 -0.0325 -0.0182 0.0003

Column 49 through 64
0.0184 0.0241 0.0149 -0.0055 -0.0284 -0.0374 -0.0274 -0.0050 0.0149 0.0272 0.0231 0.0093 -0.0040 -0.0147 -0.0184 -0.0122

Column 65 through 80
-0.0009 0.0104 0.0171 0.0144 0.0019 -0.0149 -0.0242 -0.0232 -0.0055 0.0172 0.0305 0.0252 0.0041 -0.0205 -0.0352 -0.0345

Column 81 through 96
-0.0214 -0.0080 0.0019 0.0068 0.0081 0.0091 0.0092 0.0074 0.0081 0.0047 0.0079 0.0109 0.0033 -0.0178 -0.0435 -0.0474

Column 97 through 112
-0.0017 0.0055 0.1991 0.2381 0.1791 0.0400 -0.0745 -0.1147 -0.1085 -0.1280 -0.1947 -0.2490 -0.1938 -0.0141 0.1924 0.3514


Column 113 through 128
0.2990 0.2169 0.1403 0.1378 0.0469 -0.1400 -0.4440 -0.4724 -0.4512 -0.2119 0.3220 0.4319 0.4839 -0.1884 -0.8143 -1.0051

Column 129 through 144
-0.4482 0.6937 1.8085 2.2503 1.7184 0.4933 -0.7549 -1.4304 -1.3921 -0.9374 -0.4370 -0.0143 0.3807 0.4905 0.4723 -0.3277

Column 145 through 160

```

B4. Cuplikan Hasil Plot Kejadian Gempa Tipe B



RIWAYAT HIDUP PENULIS



Nama : Wisnu Waskitho Aji
TTL : Pacitan, 15 Maret 1997
Jenis Kelamin : Laki-laki
Agama : Islam
Alamat : Jalan D.I. Panjaitan RT 01
RW 02 Kabupaten Pacitan
Telp/Hp : 087758222997
E-mail : wisnu.waskitho15@mhs.ee
.its.ac.id

RIWAYAT PENDIDIKAN:

1. 2003 – 2009 : SD Negeri 1 PACITAN
2. 2009 – 2012 : SMP Negeri 1 PACITAN
3. 2012 – 2015 : SMA Negeri 1 PACITAN
4. 2015 – 2018 : Program Studi Komputer Kontrol, Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember, Surabaya

PENGALAMAN KERJA:

1. Kerja Praktek di PT Toyota Motor Manufacturing Indonesia (TMMIN), Sunter, Jakarta Utara

PENGALAMAN ORGANISASI:

1. Departemen Kajian Strategis HIMA D3TEKTRO 2016-2017
2. Departemen Kajian Strategis HIMA D3TEKTRO 2017-2018
3. Kementrian Inkubator Kajian BEM ITS 2017-2018
4. Anggota Lab. Elektronika Terapan 2017-2018