



**RANCANG BANGUN *PROTOTYPE* SISTEM KEAMANAN
BRANKAS UANG MENGGUNAKAN PENGENAL SUARA
DENGAN METODE JARINGAN SYARAF TIRUAN**

SKRIPSI

Oleh

**NURDIAN WAHYUDI
NIM: 081910201060**

**PROGRAM STUDI STRATA 1
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER**

2015



**RANCANG BANGUN *PROTOTYPE* SISTEM KEAMANAN
BRANKAS UANG MENGGUNAKAN PENGENAL SUARA
DENGAN METODE JARINGAN SYARAF TIRUAN**

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Program Studi Teknik Elektro (S1)
dan mencapai gelar Sarjana Teknik

Oleh

**NURDIAN WAHYUDI
NIM: 081910201060**

**PROGRAM STUDI STRATA 1
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER**

2015

PERSEMBAHAN

Syukur Alhamdulillah penulis panjatkan kehadiran Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul **“Rancang Bangun Prototype Sistem Keamanan Brankas Uang Menggunakan Pengenal Suara Dengan Metode Jaringan Syaraf Tiruan”** Skripsi ini disusun untuk memenuhi salah satu syarat dalam menyelesaikan pendidikan strata satu (S1) pada Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember.

Penulisan skripsi ini tidak lepas dari bantuan berbagai pihak, oleh karena itu penulis ingin menyampaikan ucapan terima kasih yang tiada terhingga kepada:

1. Allah SWT.
2. Rasulullah Muhammad SAW.
3. Ibuku tersayang Nurjuha yang sangat berjasa memberikan seluruh kasih sayang dan lantunan doa disetiap sujudnya. Terimakasih Ibu aku menyayangimu
4. Kakakku tercinta Yuyun Choiratul Anis S.Pd terimakasih selama ini sudah memberikan omelannya untuk diriku agar menjadi lebih baik dan bertanggung jawab. Aku menyayangimu kak.
5. Istriku tercinta Sungging Permatasari, terimakasih sayang engkau telah menamaniku dalam segala kondisi, melayaniku dan mendukungku dengan kebijakan dan keputusan yang aku lakukan. Aku sayang kamu, jangan pernah menyerah tuk bersama berubah menjadi lebih baik dihadapan Allah S.W.T
6. Semua Dosen Jurusan Teknik Elektro Fakultas Teknik Universitas Jember yang telah membimbing dan memberikan ilmu. Terutama Bapak., Satryo Budi Utomo, S.T., M.T. selaku DPU dan Bapak Sumardi ST.MT selaku DPA yang telah meluangkan waktu dan pikiran serta perhatiannya guna memberikan bimbingan dan pengarahan demi terselesaikannya skripsi ini. Dosen Penguji I Bapak. Bambang Supeno S.T.,M.T. dan Dosen Penguji II Bapak. Widjonarko, S.T., M.T

7. Pak Bambang Supeno S.T.,M.T, selaku Kombi terimakasih pak, akhirnya saya bisa menghadap dan menemukan dosen pembimbing yang sangat perhatian terhadap saya dan akhirnya saya kembali ke kampus setelah sempat vakum beberapa semester.
8. Teknik Elektro angkatan 08, kita bersaudara kita bisa. “ayo rek dulen maneh suwi g kumpul”
9. Aris Ardiansyah yang telah menyediakan tempat sanggar pramuka sebagai tempat pengerjaan laporan
10. Rijal Chaidir S.T.,M.T. Taufik Hidayat, Sebastian L,Dimas Agus S.T, Mirza Maulana S.T,Nopex S.T ,Roqiqul Ma’ani ST, Ibnu Arkham S.T, Musyafa Ali S.T Putu Adit Renandana S.T Zipo Priambodo S.T dan lainnya yang masih terus berkomunikasi sampai saat ini.
11. Teman – teman Teknomart Group Pak Askin, pak Ayyub, tim RyzqCom Jember, tim Multitech Jember, ITC Jember, Imtech Jember, Imtech Bondowoso trimakasih atas kesempatan untuk berkarir dan mengembangkan hobi sehingga di situlah kutemukan pendamping hidupku.
12. Wahyu Muldayani S.T.,M.T terimakasih kawan telah membantu proses penyelesaian skripsi ini dan memberikan masukan serta saran yang sangat berguna.
13. Terimakasih buat saudara-saudaraku di Lumajang Lek Dewi,Tika, Mbah Lik,Mbak Mimin, Mas Bahul, Mas Pipit,dan yang lupa disebutin terutama mas Agus yang dulu sering memberi tumpangan saat masih naik bus.
14. Terimakasih buat saudara-saudaraku di Sidodadi. Mbah Parni Mbah Tuk Ibu dan bapak mertua Bu Iin dan Pak Suroso Mbak Kom, Mas Arif, Ria Lismana, Helmalia, Berliana
15. Calon buah hati yang masih dikandung sang istri tercinta kau memberikan ayahmu semangat untuk menyelesaikan kuliahnya.
16. Terimakasih kepada pihak – pihak yang telah membantu terselesaikannya skripsi ini yang mungkin tidak sempat atau lupa disebut namanya.

MOTTO

“MANUSIA HANYA MAMPU BERUSAHA BIARKANLAH
ALLAH YANG MENENTUKAN HASILNYA”

(Nurdian Wahyudi)

(..”Sesungguhnya Allah tidak mengubah keadaan suatu kaum sehingga mereka mengubah keadaan yang ada pada diri mereka sendiri” ...)

(Al-Quran Surah Ar-Ra’du : 11)

”Beri kesempatan dirimu untuk merenungi kesalahan dan pelajaran apa yang kau dapat dari sebuah peristiwa yang telah kau alami”

(Nurdian Wahyudi)

“Saya datang, saya bimbingan, saya ujian, saya revisi dan saya menang!”

(Skripsi)

PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Nurdian Wahyudi

NIM : 081910201060

Menyatakan dengan sesungguhnya bahwa karya tulis yang berjudul “**Rancang Bangun *Prototype* Sistem Keamanan Brankas Uang Menggunakan Pengenal Suara Dengan Metode Jaringan Syaraf Tiruan**” adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, dan belum pernah diajukan pada institusi manapun, serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, 02 Januari 2015

Yang menyatakan,

Nurdian Wahyudi

NIM 081910201060

SKRIPSI

**RANCANG BANGUN *PROTOTYPE* SISTEM KEAMANAN BRANKAS UANG
MENGUNAKAN PENGENAL SUARA DENGAN METODE JARINGAN
SYARAF TIRUAN**

Oleh

Nurdian Wahyudi

NIM 081910201060

Pembimbing :

Dosen Pembimbing Utama : Satryo Budi Utomo, S.T., M.T

Dosen Pembimbing Anggota : Sumardi S.T.,M.T

PENGESAHAN

Skripsi berjudul “**Rancang Bangun *Prototype* Sistem Keamanan Brankas Uang Menggunakan Pengenal Suara Dengan Metode Jaringan Syaraf Tiruan**” telah diuji dan disahkan oleh Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember pada :

Hari :

Tanggal :

Tempat : Fakultas Teknik Universitas Jember

Menyetujui,

Dosen Pembimbing Utama

Dosen Pembimbing Anggota

Satryo Budi Utomo, S.T., M.T

Sumardi, ST.,MT

NIP 19850126 200801 1002

NIP 19670113 199802 1 001

Dosen Penguji I

Dosen Penguji II

Bambang Supeno, ST., MT

Widjonarko, ST., MT

NIP 19690630 199512 1 001

NIP 19710908 199903 1001

Mengesahkan

Dekan Fakultas Teknik,

Ir. Widyono Hadi, MT

NIP 19610414 198902 1 001

**RANCANG BANGUN *PROTOTYPE* SISTEM KEAMANAN BRANKAS
UANG MENGGUNAKAN PENGENAL SUARA DENGAN METODE
JARINGAN SYARAF TIRUAN**

Nurdian Wahyudi

Mahasiswa Jurusan Teknik Elektro,

Fakultas Teknik, Universitas Jember

ABSTRAK

Prototype sistem keamanan brankas uang menggunakan pengenalan suara dengan metode jaringan syaraf tiruan adalah sebuah *prototype* brankas yang cara membukanya dengan diberikan masukan perintah suara dari sang pemilik brankas. Pengenalan suara diproses dengan metode ekstraksi ciri suara yaitu MFCC (Mel Frequency Cepstrum Coeficient) dan JST (Jaringan Syaraf Tiruan). Brankas ini terdiri dari PC sebagai kontrol utama kemudian Arduino seri UNO R3 sebagai komunikasi dan penerjemah perintah dari PC ke rangkaian driver selenoid untuk mengaktifkan selenoid dan kemudian membuka pintu brankas. Pada penelitian ini brankas mampu mengenali suara dengan tingkat keberhasilan lebih dari 50%

Kata kunci : Brankas, Suara, MFCC, JST, Arduino.

**THE DESIGN OF PROTOTYPE STEEL SAFES SECURITY SYSTEM
USING SPEECH RECOGNITION WITH ARTIFICIAL NEURAL
NETWORK METHODE**

Nurdian Wahyudi

Mahasiswa Jurusan Teknik Elektro,

Fakultas Teknik, Universitas Jember

ABSTRACT

Prototype steel safes security system using voice recognition with neural network method is a way to open the safes were given voice command input from the owner safes . Metode processed with voice recognition feature extraction , namely voice MFCC (Mel Frequency Cepstrum coefficient) and ANN (Artificial Neural Network) . Safes consists of a PC as the primary control then Arduino UNO R3 series as communication and command interpreter from the PC to the solenoid driver circuit to activate the solenoid and then open the safe door . In this study safes able to recognize of the voice .a success rate of over 50 % .

Keywords : Sound , steel safes , MFCC , JST , Arduino

RINGKASAN

Rancang Bangun Sistem Keamanan Brankas Uang Menggunakan Pengenal Suara Dengan Metode Jaringan Syaraf Tiruan; Nurdian Wahyudi,

081910201060; 2015: 51 halaman; Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Brankas merupakan tempat penting untuk menyimpan barang berharga berupa uang, emas, ataupun perhiasan. Maka dari itu, brankas membutuhkan sistem keamanan yang canggih. Terdapat dua tipe brankas yang ada saat ini yaitu brankas dengan tipe mekanis dan tipe digital

Prototype sistem keamanan brankas dengan pengenalan suara menggunakan JST merupakan pengembangan dari sistem keamanan tipe digital. Sistem kerja brankas akan membuka pada saat diberi perintah suara. Suara yang diucapkan akan diproses menggunakan metode ekstraksi ciri suara yaitu MFCC (Mel-Frequency Cepstrum Coeficient). MFCC akan menghasilkan pola – pola yang nantinya digunakan sebagai masukan data pada JST. JST akan memproses masukan sesuai dengan target yang ditetapkan.

Kata yang diucapkan ditentukan sebagai kata sandi untuk membuka brankas sedangkan selain kata sandi tersebut maka brankas akan otomatis terkunci. Pada penelitian ini kata sandi ditentukan yaitu kata “abcdefg” sebagai kata untuk membuka brankas dan diuji selama 10x dengan tingkat keberhasilan sebesar 100%, pada pengujian dengan kata selain “abcdefg” kondisi brankas terkunci dan dari hasil pengujian pengenalan pola yang berhasil dikenali sebesar 90% dimana kondisi brankas tetap terkunci sesuai dengan yang diharapkan.

PRAKATA

Puji syukur kehadiran Allah SWT, atas hidayahnya dan rahmatnya sehingga kami dapat menyelesaikan skripsi ini sebagaimana mestinya. Shalawat serta salam semoga Allah SWT limpahkan kepada Nabi Muhammad SAW sebagai sumber inspirasi dan membuat kami lebih kuat dan menatap setiap hal yang penuh optimis dan berfikir positif, dalam menunjang kemampuan kami dalam menjalani persaingan globalisasi kerja nantinya.

Dalam pelaksanaannya kami tidak lepas dari kesulitan dan permasalahan dalam penyusunan skripsi ini, baik dari proses pembuatan proposal sampai penyusunan akhir skripsi, mengenai ilmu yang bermanfaat, moral dan sikap serta tanggung jawab dalam menyelesaikan skripsi ini. Dengan demikian kami mengucapkan terima kasih pada:

1. Ir.Widyono Hadi M.T.,selaku Dekan Fakultas Teknik Universitas Jember.
2. Dr.Triwahju Hardianto S.T.,M.T selaku ketua Jurusan Teknik Elektro, Fakultas Teknik Universitas Jember.
3. Bapak Satriyo Budi Utomo, S.T., M.T., selaku Dosen Pembimbing Utama yang dengan sabar mengarahkan dan memberikan saran dan masukan yang berguna dalam proses penyelesaian skripsi ini baik di kampus dan di rumah sehingga skripsi ini dapat terselesaikan.
4. Sumardi, S.T., M.T., selaku Dosen Pembimbing Anggota yang selalu mendukung saya dan memotifasi saya untuk selalu menjadi lebih baik sehingga saya lebih semangat dalam mngerjakan skripsi ini.
5. Bambang Supeno S.T., M.T., selaku penguji pertama dan Bambang Widjonarko, S.T., MT., selaku penguji kedua yang telah memberikan saran dan waktu.
6. H.R.B. .Ghozali S.T., M.T., selaku Dosen Pembimbing Akademik.
7. Seluruh Dosen Teknik Elektro Universitas Jember yang tidak dapat saya sebutkan satu-persatu, terima kasih atas bimbingan yang telah diberikan.

8. Keluargaku tercinta, atas dukungan dan motifasinya.
9. Semua teman-teman teknik elektro 2008 S1 buat suka, duka, canda dan tawa yang sudah kalian berikan pada saya.
10. Aris, Wahyu, Dimas, Rijal, Putu, Taufik, L. Sebastian yang telah membantu dengan dukungan moral.
11. Kepada seluruh pihak yang telah membantu menyelesaikan pendidikan di Universitas Jember ini yang tidak dapat saya sebutkan satu-persatu.

Dalam penyusunan skripsi ini tentunya masih banyak kekurangan baik dalam isi maupun analisisnya, oleh karena itu kami mengharapkan pada para pembaca dapat merevisi dan manjadikan lebih baik, kami berharap semoga skripsi ini dapat berguna bagi pembaca, terima kasih

Jember, 02 Januari 2016

Penulis

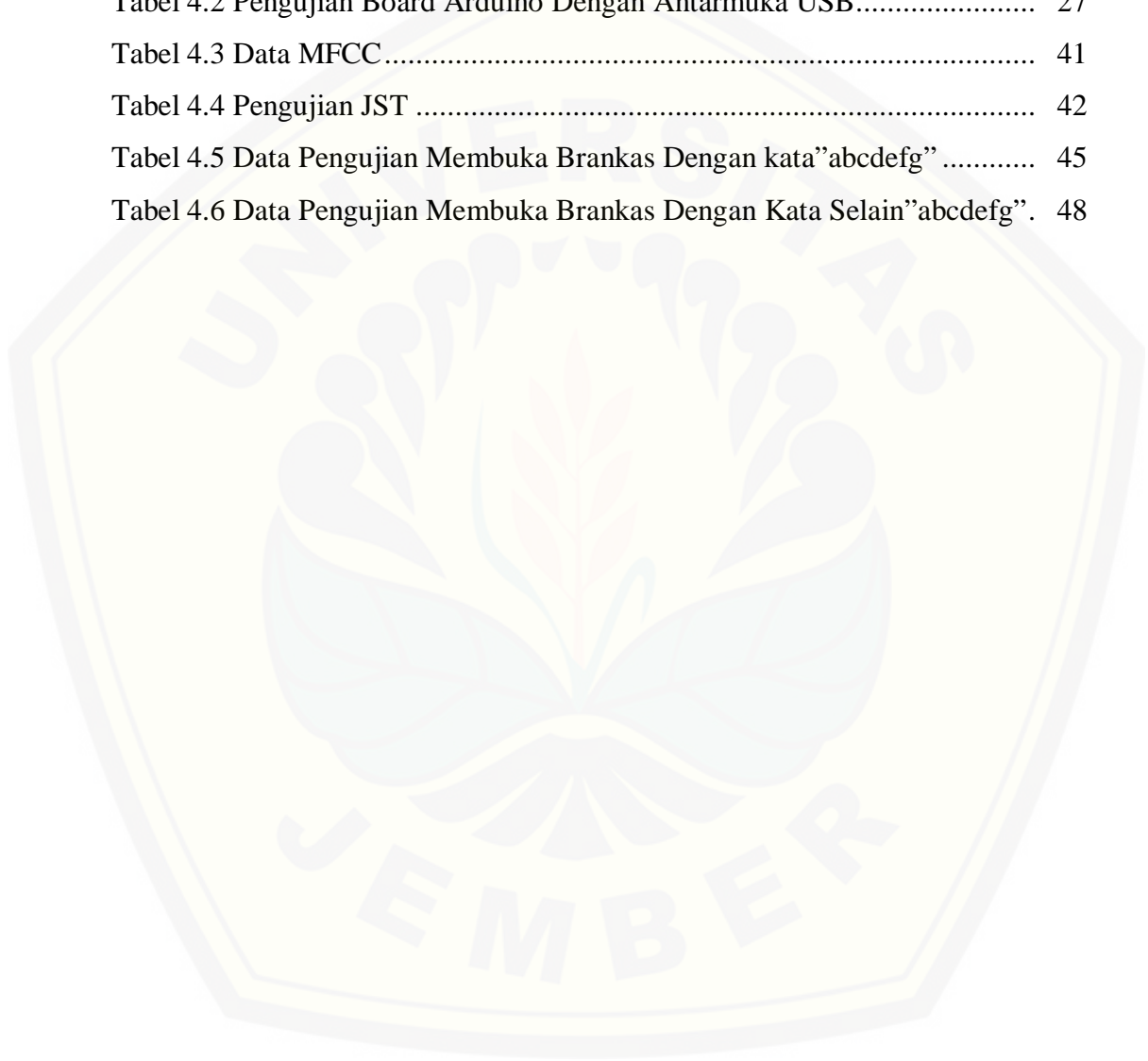
DAFTAR ISI

HALAMAN SAMPUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO.....	iv
HALAMAN PERNYATAAN.....	v
HALAMAN PENGESAHAN	vii
ABSTRAK	viii
ABSTRACT	ix
RINGKASAN.....	x
PRAKATA	xi
DAFTAR ISI.....	xiii
DAFTAR TABEL	xv
DAFTAR GAMBAR.....	xvi
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan.....	3
1.5 Manfaat.....	3
BAB 2. TINJAUAN PUSTAKA.....	4
2.1 Brankas	4
2.2 Sistem Pengenalan Perintah Suara.....	5
2.2.1 Voice Activity Detection (VAD).....	6
2.2.2 <i>Mel-Frequency Cepstrum Coefficients</i> (MFCC).....	9
2.3 Jaringan Syaraf Tiruan (JST).....	12
2.4 Arduino Uno R3.....	14
2.5 Relay	16
2.6 Selenoid	17
BAB 3. METODE PENELITIAN	18
3.1 Jadwal Penelitian	18
3.2 Alat dan Bahan.....	18

3.2.1 Perangkat Keras (<i>Hardware</i>)	18
3.2.2 Perangkat Lunak (<i>Software</i>).....	18
3.2.3 Perangkat Pendukung	18
3.3 Tahap Perencanaan	19
3.4 Perancangan Perangkat Keras.....	20
3.4.1 Rangkaian Driver Selenoid	21
3.5 Aplikasi Pengenalan Suara	22
3.6 Flowchart Sistem Kerja Alat Keseluruhan.....	23
3.7 Komunikasi Arduino Dengan Matlab PC menggunakan USB .	25
BAB 4. HASIL DAN PEMBAHASAN	26
4.1 Pengujian Perangkat Keras	26
4.1.1 Pengujian Selenoid	26
4.1.2 Pengujian Board Arduino Dengan Antarmuka USB.....	27
4.2 Pengujian Perangkat Lunak	27
4.2.1 Pengujian Pembentukan Pola Suara	28
4.2.2 Pengujian Pola Suara Dengan JST	42
4.3 Pengujian Alat Secara Keseluruhan.....	43
BAB 5 PENUTUP	51
5.1 Kesimpulan.....	51
5.2 Saran.....	51
DAFTAR PUSTAKA	52

DAFTAR TABEL

	halaman
Tabel 3.1 Input dan Output JST.....	25
Tabel 4.1 Hasil Pengujian Driver Selenoid.....	26
Tabel 4.2 Pengujian Board Arduino Dengan Antarmuka USB.....	27
Tabel 4.3 Data MFCC.....	41
Tabel 4.4 Pengujian JST	42
Tabel 4.5 Data Pengujian Membuka Brankas Dengan kata"abcdefg"	45
Tabel 4.6 Data Pengujian Membuka Brankas Dengan Kata Selain"abcdefg".	48



DAFTAR GAMBAR

	halaman
Gambar 1.1 Brankas Digital.....	1
Gambar 1.2 Brankas Mekanik.....	1
Gambar 2.1 Brankas Tipe Dinding	4
Gambar 2.2 Brankas Tipe Portable.....	4
Gambar 2.3 Brankas Tipe Ruangan.....	4
Gambar 2.4 Diagram Blok VAD.....	6
Gambar 2.5 Mel-Frequency Wrapping	11
Gambar 2.6 Tahap MFCC.....	12
Gambar 2.7 Arduino Uno R3	14
Gambar 2.8 Relay	16
Gambar 2.9 Selenoid.....	17
Gambar 3.1 Blok Diagram Sistem.....	20
Gambar 3.2 Brankas dan Kotak Arduino.....	21
Gambar 3.3 Rangkaian Driver Selenoid	21
Gambar 3.4 Aplikasi Pengenal Suara	22
Gambar 3.5 Flowchart Alat Keseluruhan.....	23
Gambar 3.6 Topologi JST	25
Gambar 4.1 Aplikasi Yang Dijalankan.....	28
Gambar 4.2a Hasil Perekaman Suara Kata “abcdefg”.....	29
Gambar 4.2b Grafik Frekuensi Terhadap Amplitudo.....	29
Gambar 4.3 Hasil Perekaman Suara Kata “lagi makan”	30
Gambar 4.4 VAD Kata “abcdefg”	30
Gambar 4.5 Akuisisi Data Kata “abcdefg”	31
Gambar 4.6 Windowing Kata “abcdefg”	32
Gambar 4.7 FFT Kata “abcdefg”	34
Gambar 4.8 Mel Wrapping Kata “abcdefg”	36
Gambar 4.9 Data MFCC	39
Gambar 5.0 Brankas Berhasil Dibuka	44
Gambar 5.1 Brankas Gagal Dibuka	44

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Brankas merupakan tempat penting untuk menyimpan barang berharga berupa uang, emas, ataupun perhiasan. Maka dari itu, brankas membutuhkan sistem keamanan yang canggih. Terdapat dua tipe brankas yang ada saat ini yaitu brankas dengan tipe mekanis dan tipe digital. Sistem keamanan brankas pada tipe mekanis yaitu dengan menggunakan model pemutar untuk membuka kunci pada brankas tersebut dari jarak dekat. Sistem tersebut memiliki kelemahan yaitu jika akurasi putaran kode kunci kurang sedikit atau hanya beberapa derajat saja, maka harus mengulang kode kunci brankas dari awal, dan harus benar-benar tepat untuk menentukan angka-angka kombinasi.

Brankas digital merupakan brankas yang memiliki kode kunci kombinasi dengan bantuan sistem elektronik berbasis digital. Brankas digital lebih mudah dioperasikan daripada brankas mekanik. Namun pada brankas tipe digital juga masih memiliki kekurangan yaitu jika brankas tersebut disinari lampu ultraviolet maka akan terlihat jelas sidik jari para tombol-tombol yang pernah atau sering di tekan. Berikut merupakan contoh tampilan dari kedua brankas tersebut.



Gambar 1.1 Brankas Digital



Gambar 1. 2 Brankas Mekanik

Berdasarkan masalah tersebut memotivasi penulis untuk merancang *prototype* sistem keamanan brankas yang lebih aman dan canggih serta fleksibel dibandingkan brankas yang sudah ada. Inovasi tersebut diwujudkan dengan merancang *prototype* sistem keamanan brankas dengan pengenalan suara untuk

membuka dengan menggunakan metode “Jaringan Syaraf Tiruan” yang digunakan untuk mengidentifikasi suara pemilik atau bukan.

1.2 Rumusan Masalah

Permasalahan yang mendasar dalam skripsi ini adalah :

1. Bagaimana mengimplementasikan pengenalan suara menggunakan metode jaringan syaraf tiruan ke dalam *prototype* sebuah brankas untuk sistem keamanan.
2. Bagaimana mendesain program untuk sistem keamanan brankas dengan suara menggunakan jaringan syaraf tiruan.
3. Bagaimana mendesain perangkat keras yang tepat untuk sistem keamanan brankas dengan suara menggunakan jaringan syaraf tiruan

1.3 Batasan Masalah

Pembatasan masalah sangat diperlukan untuk menghindari pembahasan yang menyimpang dari telah ditentukan. Dalam pembahasan ini ruang lingkup pembahasannya di fokuskan pada hal sebagai berikut :

1. Proses pengambilan suara dalam kondisi hening.
2. Menggunakan mikrofon laptop sebagai input suara
3. Jarak pengambilan suara dengan mikrofon maksimal 30cm
4. Menggunakan arduino seri Arduino Uno R3
5. Menggunakan *Notebook* atau PC sebagai pengendali utama.
6. Frekuensi pengambilan suara di lakukan pada frekuensi 8000hz dengan lama waktu 2 detik
7. Pembuatan program dilakukan dengan menggunakan bantuan bahasa pemrograman Matlab R2013a
8. Sistem menggunakan selenoid dan driver relay untuk penguncian pintu brankas
9. Perintah menggunakan sandi kata 'abcdefg' yang digunakan sebagai kata sandi untuk membuka berankas
10. Pengambilan suara dilakukan dalam kondisi normal, tidak dalam kondisi emosi.

1.4 Tujuan

Tujuan yang ingin dicapai dalam skripsi ini yaitu mengaplikasikan JST (Jaringan Syaraf Tiruan) sebagai proses identifikasi suara untuk keamanan brankas

1.5 Manfaat

Adapun manfaat yang ingin dicapai dari penelitian ini adalah :

1. Masyarakat atau vendor bisa mencipitakan brankas uang dengan teknologi pengenalan suara dan menjadikan pengenalan suara sebagai sebuah alternatif baru untuk sistem keamanan di masa kini.
2. Dapat dijadikan bahan referensi untuk pengembangan sistem pengenalan suara pada peralatan yang lain

BAB 2. TINJAUAN PUSTAKA

2.1 Brankas

Kata brankas berasal dari bahasa Belanda yaitu *branden* artinya membakar dan *kast* artinya lemari, jadi lemari tahan kebakaran. Sedangkan dalam bahasa Indonesia artinya lemari besi, yaitu lemari yang terbuat dari besi

Umumnya brankas berbentuk balok atau silinder. Berikut contoh gambar brankas:



Gambar 2.1

Brankas tipe dinding



Gambar 2.2

Brankas tipe portable



Gambar 2.3

Brankas tipe ruangan

Berdasarkan ketahanannya brankas di bagi menjadi 3 jenis:

- a. *Steel Safes* (brankas yang terbuat dari plat besi/baja saja). Brankas seperti ini tidak tahan api dan biasanya hanya di pergunakan untuk penyimpanan uang yang relatif sedikit untuk kebutuhan operasional usaha skala kecil.
- b. *Fire Safes* (brankas yang sudah di lengkapi dengan fitur tahan api). Brankas jenis ini terlihat dari ketebalan *body* brankas dan bila diketuk terasa ada isi dari *body* brankas terdubit.
- c. *Fire and Burglar Safes* (brankas yang masuk klasifikasi tahan api dan tahan dobra). Brankas jenis ini sangat di rekomendasikan untuk

digunakan karena sudah memiliki keamanan yang relatif tinggi. Biasanya sudah di lengkapi fitur relocking (penguncian otomatis yang di design khusus). Dari fisik brankas terlihat berding tebal, bahan plat juga lebih tebal dari jenis brankas tahan api.

Macam-macam kunci brankas sebagai berikut :

- a. Kunci kombinasi yang bisa di set berdasarkan keinginan pengguna dan ada kunci kombinasi yang sudah di set dari pabrik dan tidak bisa di set sesuai keinginan pengguna.
- b. Kunci Panjang. Kunci panjang biasanya menggunakan produk import ataupun lokal. Biasanya berbentuk panjang minimal 10 cm. Ada yang menggunakan *single bite* (1 sisi) dan *double bite* (2 sisi), orang sering menyebut dengan istilah 1 atau 2 bendera.
- c. Kunci kecil ada juga brankas analog atau digital yang menggunakan kunci tambahan kecil seperti kunci-kunci laci pada umumnya.

2.2 Sistem Pengenalan Perintah Suara

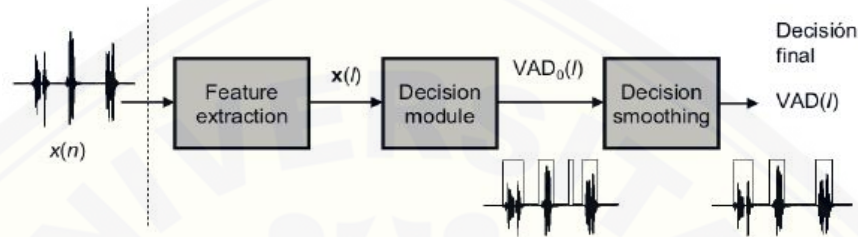
Suara adalah sebuah sinyal yang rumit sebagai sebuah hasil dari beberapa transformasi yang terjadi pada beberapa level yang berbeda dari semantik, linguistik, artikulasi (pengucapan) dan akustik. Perbedaan dalam transformasi ini tampak sebagai perbedaan anatomi yang melekat dalam vokal *tract* dan kebiasaan pengucapan yang dipelajari dari individu yang berbeda (Rabiner, 1993). Suara pada manusia digunakan untuk berkomunikasi atau berinteraksi antara manusia. Suara dapat digunakan sebagai data masukan sistem dengan menggunakan sensor suara.

Sensor Suara adalah sensor yang memiliki cara kerja merubah besaran suara menjadi besaran listrik. Pada dasarnya prinsip kerja pada alat ini hampir mirip dengan cara kerja sensor sentuh pada perangkat seperti telepon genggam, laptop, dan notebook. Sensor ini bekerja berdasarkan besar kecilnya kekuatan gelombang suara yang mengenai membran sensor yang menyebabkan Bergeraknya membran sensor yang memiliki kumparan kecil dibalik membran tersebut naik dan turun. Kecepatan gerak kumparan tersebut menentukan kuat

lemahnya gelombang listrik yang dihasilkannya. Salah satu komponen yang termasuk dalam sensor ini adalah *Microphone* atau Mic.

Mic adalah komponen elektronika dimana cara kerjanya yaitu membran yang digetarkan oleh gelombang suara yang akan menghasilkan sinyal listrik.

2.2.1 Voice Activity Detection (VAD)



Gambar 2.4 Diagram blok dari VAD (Ramirez, 2007)

Suara terdiri dari suara sebagai informasi dan suara sebagai noise. Dalam mendeteksi suara informasi/suara noise merupakan masalah kompleks dalam proses pengenalan suara. VAD adalah sebuah metode untuk membedakan segmen suara informasi dan suara noise. Hal ini sebagian banyak diaplikasi pada pemrosesan audio dan secara luas digunakan dalam bidang komunikasi suara untuk mencapai tinggi efisiensi dan bit rate rendah transmisi (Mousazadeh, 2013). Contohnya termasuk pengurangan kebisingan untuk perangkat bantu dengar digital, layanan komunikasi bergerak, sistem pengenalan suara, kompresi, dan pidato coding.

Voice Activity Detector adalah setiap kumpulan dari metode untuk segmen sinyal suara menjadi pengisi suara, tak bersuara, dan frame Noise. Secara umum, VAD memiliki tiga bagian komponen: *Feature Extractor*, *Decision Module*, dan *Decision Smoother* (Ramirez, 2007). Bagian ini masuk ke dalam sistem VAD keseluruhan sebagai berikut:

Struktur blok ini memberikan perancang peningkatan fleksibilitas dalam menciptakan solusi VAD. Bagian terbesar dari fleksibilitas ini berasal dari banyak fitur yang tersedia, masing-masing yang kuat dalam kondisi tertentu, dan batas-batas wilayah keputusan.

a. *Feature Extractor*

Fitur adalah setiap parameter dari sinyal suara yang dapat digunakan untuk membuat keputusan yang diperlukan. Fitur dikatakan baik apabila handal dan kuat terhadap noise. Tujuan dari proses ekstraksi ciri adalah untuk menghitung fitur *speech* diskriminatif untuk dideteksi. Sejumlah fitur suara kuat telah dipelajari antara lain: *full-band* dan *subband* energi (Woo dkk, 2000), tindakan spektrum perbedaan antara suara dan noise latar belakang (Marzinik dan Kollmeier, 2002), estimasi lapangan (Tucker, 1992), *zero crossing rate* dan statistik tingkat tinggi (Nemer dkk 2001., Ramirez dkk., 2006a., Górriz dkk, 2006a, Ramírez dkk., 2007). Sebagian besar metode VAD didasarkan pada pengamatan frame saat ini dan tidak mengangap informasi kontekstual. Namun, dengan menggunakan informasi suara (Ramírez dkk, 2004a., Ramírez dkk 2005a) telah menunjukkan manfaat yang signifikan bagi mendeteksi keberadaan suara di lingkungan kebisingan yang tinggi.

b. *Decision Module*

Decision Modul mendefinisikan metode untuk menetapkan kelas (suara atau diam) untuk fitur vektor x . Algoritma VAD berbasis pada kemungkinan statistik *test ratio* yang melibatkan vektor pengamatan tunggal (Sohn dkk., 1999). Metode ini dianggap sebagai hipotesis di mana aturan pengambilan keputusan optimal dengan meminimalkan kemungkinan kesalahan. Mengingat vektor observasi menjadi diklasifikasikan, masalah direduksi dengan memilih kelas (H_0 atau H_1) dengan posterior probabilitas $P(H_i | x)$ terbesar, seperti pada Persamaan (2.1).

$$P(H_1|x) \underset{H_0}{\overset{H_1}{\geq}} P(H_0|x) \quad (2.1)$$

Menggunakan aturan Bayes mengarah ke statistik uji kemungkinan rasio, seperti pada Persamaan (2.2).

$$\frac{P(x|H_1)}{P(x|H_0)} \underset{H_0}{\overset{H_1}{>}} \frac{P(H_0)}{P(H_1)} \quad (2.2)$$

Transformasi Fourier diskrit (DFT) koefisien *clean speech* (S_j) dan noise (N_j) diasumsikan variabel acak *asymptotically independent Gaussian*, seperti pada Persamaan (2.3) dan (2.4).

$$P(x|H_1) = \prod_{j=0}^{j-1} \frac{1}{\pi \lambda_N(j)} \exp \left\{ -\frac{|X_j|^2}{\lambda_N(j)} \right\} \quad (2.3)$$

$$P(x|H_1) = \prod_{j=0}^{j-1} \frac{1}{\pi [\lambda_s(j) + \lambda_N(j)]} \exp \left\{ -\frac{|X_j|^2}{\lambda_s(j) + \lambda_N(j)} \right\} \quad (2.4)$$

dimana X_j merupakan koefisien suara noise DFT, $\lambda_N(j)$, dan $\lambda_s(j)$ menunjukkan masing-masing varians N_j dan S_j untuk DFT, Dengan demikian, aturan pengambilan keputusan berkurang menjadi seperti Persamaan (2.5) dan (2.6).

$$\frac{1}{J} \sum_{j=0}^{j-1} \left[\frac{\gamma_j \xi_j}{1 + \xi_j} - \log (1 + \xi_j) \right] \underset{H_0}{\overset{H_1}{>}} \eta \quad (2.5)$$

$$\gamma_j = \frac{|X_j|^2}{\lambda_N(j)} \quad \xi_j = \frac{\lambda_s(j)}{\lambda_N(j)} \quad (2.6)$$

η mendefinisikan *decision threshold* dan J adalah urutan DFT. ξ_j dan γ_j menentukan *signal-to-noise ratio* SNRs. Dalam perkiraannya menggunakan *minimum mean-square error (MMSE) estimator*.

c. *Decision Smoother*

Sebagian besar VAD yang merumuskan aturan keputusan pada frame dengan frame sebagai dasar, menggunakan algoritma *decision smoothing* dalam rangka meningkatkan ketahanan terhadap noise. Motivasi pendekatan ini digunakan dalam proses produksi suara dan energi sinyal yang berkurang dari awal hingga akhir. *Decision Smoother* VAD digunakan untuk memulihkan periode suara yang tertutup oleh noise akustik.

2.2.2 *Mel-Frequency Cepstrum Coefficients* (MFCC)

Berbagai macam kemungkinan ada untuk *parametrically* mewakili sinyal suara untuk tugas pengenalan pembicara, seperti *Linear Prediction Coding* (LPC), *Mel-Frequency Cepstrum Coefficients* (MFCC) dan lain-lain (Leon, 2009). MFCC adalah mungkin yang paling dikenal dan paling populer, dan ini akan digunakan dalam penelitian ini. Input *speech* biasanya direkam pada *sampling rate* diatas 1000 Hz. Frekuensi sampling ini dipilih untuk meminimalkan efek konversi di analog ke digital. Sinyal-sinyal sampel dapat menangkap semua frekuensi hingga 5 kHz, yang menutupi sebagian besar energi suara yang dihasilkan oleh manusia. Tujuan utama dari prosesor MFCC adalah untuk meniru perilaku telinga manusia. Gambar 2.6 menunjukkan blok diagram dari prosesor MFCC. Proses ini dibagi menjadi 5 bagian, yaitu *Frame Blocking*, *Windowing*, *FFT*, *Mel-Frekuensi Wrapping dan Cepstrum* (Leon, 2009).

a. *Frame Blocking*

Dalam langkah ini sinyal suara kontinyu diblok menjadi frame N sampel, dengan frame yang berdekatan dipisahkan oleh M ($M < N$). Frame pertama terdiri dari N sampel pertama. Bingkai kedua dimulai M sampel setelah frame pertama, dan tumpang tindih dengan N-M sampel. Demikian pula, frame ketiga dimulai 2M sampel setelah frame pertama (atau M sampel setelah kedua bingkai) dan tumpang tindih dengan N- 2M sampel. Proses ini terus sampai semua suara dicatat dalam satu atau lebih frame.

b. *Windowing*

Langkah berikutnya dalam proses ini adalah memberikan jendela untuk setiap bingkai individu sehingga dapat meminimalkan diskontinuitas sinyal pada

awal dan akhir setiap frame. Konsep di sini adalah untuk meminimalkan distorsi spektral dengan menggunakan jendela untuk sinyal menuju nol pada awal dan akhir setiap frame. jika kita mendefinisikan jendela seperti $w(n)$, $0 \leq n \leq N - 1$, di mana N adalah jumlah sampel di setiap frame, maka hasil windowing adalah sinyal seperti dalam Persamaan (2.7).

$$y_l(n) = x_l(n)w(n), \quad 0 \leq n \leq N - 1 \quad (2.7)$$

Biasanya jendela *Hamming* digunakan untuk *windowing* proses.

c. Fast-Fourier Transform (FFT)

Langkah pengolahan berikutnya adalah *Fast Fourier Transform*, yang mengkonversi setiap frame N sampel dari domain waktu ke frekuensi domain. FFT adalah algoritma cepat untuk menerapkan *Fourier Transform Diskrit* (DFT) yang didefinisikan pada himpunan N sampel $\{x_n\}$ seperti pada Persamaan (2.8).

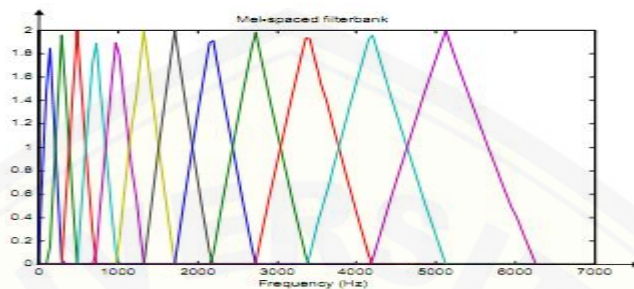
$$X_n = \sum_{k=0}^{N-1} X_k e^{-\frac{2\pi jkn}{N}}, \quad n = 0, 1, 2, \dots, N - 1 \quad (2.8)$$

Perhatikan bahwa kita menggunakan j di sini untuk menunjukkan satuan imajiner $j = \sqrt{-1}$. Secara umum X_n adalah bilangan kompleks. Hasilnya setelah langkah ini sering disebut sebagai spektrum atau periodogram.

d. Mel-Frequency Wrapping

Penelitian psikofisik telah menunjukkan bahwa persepsi manusia dari kandungan frekuensi suara untuk sinyal suara tidak mengikuti skala linier. Jadi untuk setiap nada dengan frekuensi aktual (f) diukur dalam Hz, subyektif lapangan diukur pada skala yang disebut 'mel' skala. Skala *mel-frequency* adalah frekuensi linier berada dibawah 1000 Hz dan bentuk logaritmik di atas 1000 Hz. Sebagai referensi titik, nada-nada 1 kHz, 40 dB di atas persepsi mendengar *threshold*, didefinisikan sebagai mels 1000. Salah satu pendekatan simulasi spektrum yaitu dengan menggunakan menyaring jarak pada skala mel (lihat Gambar 2.6). Filter mempunyai respon frekuensi *bandpass* segitiga, dan jarak serta *bandwidth* ditentukan oleh konstanta selang frekuensi mel. Spektrum modifikasi dari $S(Z)$ terdiri dari daya keluaran dari filter ini, dimana $S(Z)$ adalah

input. Perhatikan bahwa filter ini diterapkan dalam frekuensi domain, maka hanya berbentuk segitiga jendela pada spektrum (Gambar 2.8). Sebuah cara tentang hal filter ini adalah untuk melihat masing-masing filter sebagai histogram (dimana suara memiliki tumpang tindih) di domain frekuensi.



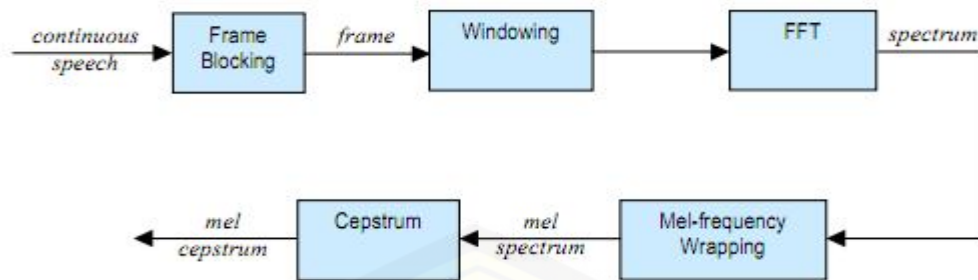
Gambar 2.5 *Mel-Frequency Wrapping*

e. *Cepstrum*

Dalam langkah terakhir ini, spektrum log mel dirubah kembali ke waktu. Hasilnya disebut *cepstrum* frekuensi mel koefisien (MFCC). Representasi cepstral dari spektrum pidato memberikan representasi yang baik dari lokal sifat spektral dari sinyal untuk analisis frame yang diberikan. Karena koefisien spektrum mel (sehingga mereka logaritma) adalah bilangan real, kita dapat mengkonversinya ke domain waktu menggunakan *the Cosine Transform Diskrit* (DCT). Oleh karena itu jika kita menunjukkan mereka mel koefisien spektrum daya yang merupakan hasil dari langkah terakhir adalah $\tilde{S}_k, k = 1, 2, \dots, K$, Kita dapat menghitung MFCC, seperti dalam Persamaan (2.9).

$$\tilde{C}_n = \sum_{k=1}^K (\log \tilde{S}_k) \cos \left[n \left(k - \frac{1}{2} \right) \frac{\pi}{K} \right], \quad n = 1, 2, \dots, K \quad (2.9)$$

Perhatikan bahwa kita mengecualikan komponen pertama, \tilde{C}_0 , dari DCT karena merupakan nilai rata-rata dari sinyal input yang dilakukan sedikit informasi spesifik *speaker*.



Gambar 2.6 Tahap MFCC (Leon, 2009)

2.3 Jaringan Syaraf Tiruan (JST)

Jaringan saraf tiruan memiliki banyak kesamaan dengan Markov model. Keduanya model statistik yang direpresentasikan sebagai grafik. Dimana model Markov menggunakan probabilitas bagi negara transisi, jaringan saraf menggunakan kekuatan koneksi dan fungsi. Perbedaan utama adalah bahwa jaringan saraf yang fundamental paralel sementara rantai Markov adalah serial. Frekuensi dalam pidato, terjadi secara paralel, sedangkan seri suku kata dan kata-kata pada dasarnya serial. Ini berarti bahwa kedua teknik yang sangat kuat dalam konteks yang berbeda.

Seperti dalam jaringan saraf, tantangannya adalah untuk mengatur bobot yang sesuai dari koneksi, model Markov tantangan adalah menemukan yang tepat transisi dan observasi probabilitas. Dalam banyak sistem pengenalan suara, kedua teknik diimplementasikan bersama-sama dan bekerja dalam hubungan simbiosis. Jaringan saraf berkinerja sangat baik pada belajar probabilitas fonem dari input audio yang sangat paralel, sementara model Markov dapat menggunakan probabilitas pengamatan fonem bahwa jaringan saraf menyediakan untuk menghasilkan urutan fonem paling mungkin atau kata. Ini adalah inti dari pendekatan *hybrid* untuk pemahaman bahasa alami.

Back propagation atau *error back-propagation* ialah teknik pembelajaran terawasi (*supervised learning*) yang digunakan untuk melatih jaringan syaraf tiruan. pertama kali dideskripsikan oleh Paul Werbos pada 1974, dan dikembangkan lebih lanjut oleh D.E. Rumelhart, G.E. Hilton, dan R.J. Williams pada 1986.

Algoritma ini berdasar pada aturan delta umum (*generalized delta rule*) Persamaan (2.10),

$$w_{ij} = w_{ij} + \Delta w_{ij} = w_{ij} - \eta \cdot \frac{\partial \varepsilon}{\partial w_{ij}} \quad (2.10)$$

dengan w_{ij} ialah bobot neuron pada layer-i ke layer-j, ialah laju belajar, dan ialah output SSE (jumlah kuadrat error). Biasanya jaringan syaraf tiruan belajar lebih cepat dengan kenaikan laju belajar, tetapi dapat menghasilkan osilasi bahkan kegagalan belajar. Untuk mereduksi osilasi pada nilai laju belajar yang tinggi, faktor momentum digunakan untuk menjaga kekonstanan kecenderungan proses belajar. Sehingga aturan delta umum untuk perbarahruan bobot menjadi Persamaan (2.11).

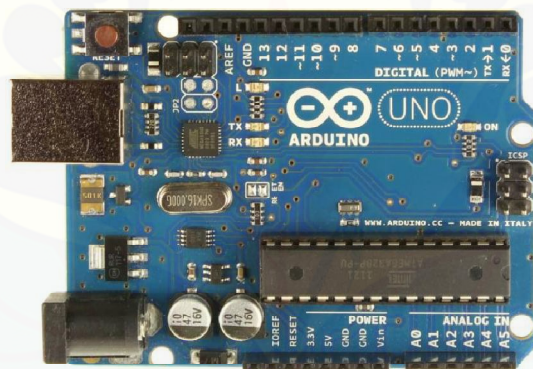
$$\Delta w_{ij}^{(n+1)} = -\eta \cdot \frac{\partial \varepsilon}{\partial w_{ij}} + \alpha \cdot \Delta w_{ij}^{(n)} \quad (2.11)$$

Pendekatan lain untuk mengklasifikasikan contoh pidato adalah untuk memanfaatkan *Radial Basis Function Network* (RBF). jaringan ini juga terdiri dari tiga lapisan: lapisan input, lapisan tersembunyi dan lapisan output. Perbedaan utama dari jenis jaringan adalah bahwa lapisan tersembunyi memiliki (Gaussian) pemetaan fungsi. Sebagian besar mereka digunakan untuk aproksimasi fungsi, tetapi mereka juga dapat memecahkan masalah klasifikasi.

Lapisan input mirip dengan lapisan input dari *Multilayer Feedforward* Jaringan digunakan. Jaringan RBF terdiri dari satu lapisan tersembunyi neuron dengan fungsi dasar. Pada masukan dari masing-masing neuron, jarak antara pusat neuron dan vektor input dihitung. Output dari neuron ini kemudian dibentuk dengan menerapkan dasar fungsi untuk jarak ini. Output jaringan RBF terbentuk dengan jumlah bobot output neuron dan bias kesatuan. Untuk perkiraan Frekuensi Mel Cepstrum *Coefficients*, 450 tersembunyi lapisan neuron yang diperlukan, yang jauh lebih dari 9 sigmoid tersembunyi neuron lapisan yang diperlukan dalam *Multilayer Feedforward Network* (Gevaert, 2010).

2.4 Arduino Uno R3

Arduino Uno merupakan papan mikrokontroler yang berbasis ATmega 328P. Mempunyai 14 digital input/output, yang 6 pin bisa digunakan sebagai keluaran PWM, 6 analog input, 16 MHz osilator Kristal, penghubung USB, power jack, ICSP header, dan tombol reset. Bagian ini sangat dibutuhkan untuk mendukung mikrokontroler. Contoh, menghubungkan Arduino ke komputer dengan kabel USB atau memberikan tegangan AC ke DC adaptor atau baterai untuk memulainya. Perbedaan mendasar dari sebelumnya adalah tidak menggunakan chip FTDI dan sebagai gantinya menggunakan Atmega8U2 yang diprogram sebagai converter USB-to-serial. Perubahan ini cukup membantu dalam instalasi *software* Arduino.



Gambar 2.7 Arduino Uno R3.

Mikrokontroler	:	ATmega328P
Tegangan operasi	:	5V
Tegangan masukan	:	7-12 V
Batas tegangan masukan	:	6-20 V
Digital I/O pin	:	14 (6 bisa digunakan sebagai keluaran PWM)
Pin masukan analog	:	6
Arus DC per pin I/O	:	40 mA
Arus DC untuk pin 3,3V	:	50 mA
		32 KB (ATmega 328) 0,5 KB digunakan untuk flash memory

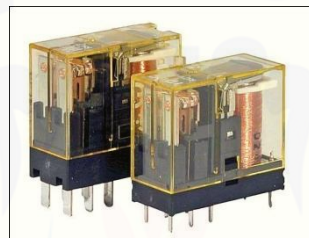
SRAM	:	2 KB (ATmega328)
EEPROM	:	1 KB (ATmega328)
Clock speed	:	16 MHz

Arduino merupakan sebuah *board minimum system* mikrokontroler yang bersifat *open source*. Di dalam rangkaian *board* arduino terdapat mikrokontroler AVR seri ATmega 328 yang merupakan produk dari atmel. Arduino memiliki kelebihan tersendiri dibanding *board* mikrokontroler lainnya selain bersifat *open source*, arduino juga mempunyai bahasa pemrograman berupa bahasa C. Selain itu dalam *board* arduino sendiri sudah terdapat *loader* yang berupa USB sehingga memudahkan pemakainya ketika memrogram mikrokontroler di dalam arduino.

Sedangkan pada kebanyakan *board* mikrokontroler yang lain masih membutuhkan rangkaian *loader* terpisah untuk memasukan program ketika memrogram mikrokontroler. *Port* USB tersebut bisa juga digunakan sebagai port komunikasi serial. Arduino menyediakan 20 pin I/O, yang terdiri dari 6 pin input analog dan 14 pin *digital input/output*. Untuk 6 pin analog sendiri bisa juga difungsikan sebagai output digital jika diperlukan output digital tambahan selain 14 pin yang sudah tersedia. Untuk mengubah pin analog menjadi digital cukup mengubah konfigurasi pin pada program. Dalam *board* kita bisa lihat pin digital diberi keterangan 0-13, jadi untuk menggunakan pin analog menjadi output digital, pin analog yang pada keterangan board 0-5 kita ubah menjadi pin 14-19. dengan kata lain pin analog 0-5 berfungsi juga sebagai pin output digital 14-16. Sifat *open source* arduino juga banyak memberikan keuntungan tersendiri untuk kita dalam menggunakan *board* ini, karena dengan sifat *open source* komponen yang kita pakai tidak hanya tergantung pada satu merek, namun memungkinkan kita bisa memakai semua komponen yang ada dipasaran. Bahasa pemrograman arduino merupakan bahasa C yang sudah disederhanakan *syntax* bahasa pemrogramannya sehingga mempermudah kita dalam mempelajari dan mendalami mikrokontroler. (www.arduino.cc)

2.5 Relay

Relay adalah komponen listrik yang dioperasikan sebagai saklar. Beberapa menggunakan elektromagnet untuk mengoperasikan pensaklaran secara mekanis, tetapi prinsip-prinsip operasi yang lain juga bisa digunakan. Relay diperlukan untuk mengendalikan rangkaian dengan sinyal daya rendah (dengan isolasi listrik yang lengkap antara kontrol dan rangkaian kontrol), atau di beberapa rangkaian yang harus dikontrol oleh satu sinyal. Gambar 2.16 di bawah ini adalah salah satu contoh relay yang banyak beredar di pasaran.



Gambar 2.8 Relay

Jenis relay yang dapat menangani daya tinggi yang dapat diperlukan untuk secara langsung mengendalikan motor listrik dan beban lainnya disebut kontaktor. Relay *solid-state* mengontrol rangkaian listrik tanpa menggerakkan komponen, ataupun dengan menggunakan perangkat semikonduktor untuk melakukan *switching*. Relay dengan yang dikalibrasi karakteristik dan koil pada nilai operasi biasanya beberapa digunakan untuk melindungi rangkaian listrik dari beban berlebih atau kondisi trip. Pada sistem tenaga listrik modern, fungsi-fungsi ini dilakukan oleh instrumen digital disebut "relay pelindung".

2.6 Solenoid

Solenoid adalah suatu alat dasar yang mengkonversi suatu sinyal listrik ke dalam gerakan mekanis, pada umumnya seperti garis. solenoid terdiri dari suatu kumparan dan alat pengisap. Pengisap tersebut mungkin adalah *freestanding*

atau dimuati pegas. Jika terdapat batang besi dan ditempatkan sebagian panjangnya di dalam solenoid, batang tersebut akan bergerak masuk ke dalam solenoid saat arus dialirkan. Hal ini dapat dimanfaatkan untuk menggerakkan tuas. Tuas inilah yang nantinya untuk menggerakkan katup. Kumparan mempunyai beberapa rating tegangan atau arus dan tipenya mungkin DC atau AC. Spesifikasi Solenoid meliputi rating listrik dan gaya pengisap menarik atau mendorong ketika yang diberi tegangan tertentu. Gaya ini mungkin dinyatakan dalam *newton* atau kilogram di dalam sistem *SI*, dan dalam pound atau ons dalam Sistem Inggris. Beberapa solenoid terbatas hanya untuk tugas sebentar-sebentar oleh karena batasan yang berkenaan dengan panas.

Dalam hal ini, *duty cycle* maksimum (persentase total waktu) akan ditetapkan. Solenoid digunakan ketika suatu gaya mendadak yang besar harus dipakai untuk melaksanakan beberapa pekerjaan.



Gambar 2.9 Solenoid

BAB 3. METODE PENELITIAN

3.1 Jadwal Penelitian

Pembuatan perangkat keras dan perangkat lunak dari rancang bangun prototype sistem keamanan brankas menggunakan pengenalan suara dengan JST dilakukan setelah pelaksanaan seminar proposal. Pelaksanaan kegiatan penelitian dilakukan di ITclinic servis jalan Kalimantan no.3a kavling 5 Sumbersari Jember.

3.2 Alat dan Bahan

Alat dan bahan yang digunakan dalam penelitian ini sebagai berikut :

3.2.1 Perangkat Keras (*Hardware*)

1. Mikrofon laptop
2. Kit Arduino
3. Kabel USB 1.5M
4. Suplai daya 6V
5. Suplai daya 9V
5. Relay
6. Selenoid dengan tegangan kerja 6v
7. Kotak Brankas

3.2.2 Perangkat Lunak (*Software*)

1. Matlab versi 2013
2. Arduino *Software Pack*

3.2.3 Peralatan pendukung

1. Laptop Asus A42F
2. Gergaji
3. Kabel
4. Solder
5. AVO meter
6. Kikir
7. Obeng

3.3 Tahap Perencanaan

Dalam pembuatan rancang bangun prototype sistem keamanan brankas menggunakan pengenalan suara dengan JST dibutuhkan langkah – langkah perancangan sebagai berikut :

1. Studi literatur

Tahap awal dari penelitian ini adalah mencari literatur dari hasil penelitian sebelumnya dan mencari literatur tentang penelitian yang sedang diteliti. Diharapkan dengan literatur yang didapat bisa memberikan keyakinan bahwa penelitian ini dapat dilaksanakan dan memberikan arahan untuk mengurangi kesalahan dalam penelitian.

2. Pembelian bahan pembuatan

Tahap kedua adalah pembelian material yang dibutuhkan untuk membuat prototype sistem keamanan brankas menggunakan pengenalan suara dengan JST yang meliputi solenoid sebagai pengunci pintu brankas, komponen-komponen elektronik aktif dan pasif yang dibutuhkan , kit Arduino dan perangkat pendukung lainnya.

3. Pengerjaan perangkat keras

Tahap ketiga adalah proses pengerjaan pembuatan prototype sistem keamanan brankas menggunakan pengenalan yang meliputi pembuatan perangkat mekanik dan elektronik dari bahan pembuatan yang sudah dibeli.

4. Pembuatan perangkat lunak

Tahap keempat adalah pembuatan perangkat lunak dengan software matlab 2013 sebagai pengolah pengenalan suara dan JST serta Arduino software untuk mengkompilasi file ke dalam board arduino Uno R3

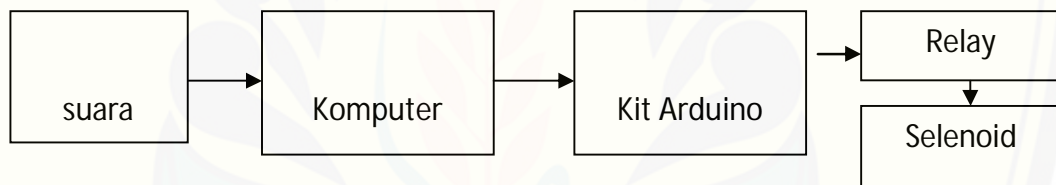
5. Pengujian alat dan Pengambilan data

Tahap kelima adalah pengujian alat dan pengambilan data, diharapkan dengan adanya tahap ini fungsi kerja baik perangkat keras maupun

perangkat lunak yang sudah dibuat dan sudah terintegrasi dalam suatu sistem dapat bekerja dengan baik.

3.4 Perancangan Perangkat Keras

Gambaran umum mengenai cara kerja *prototype* sistem keamanan brankas menggunakan pengenalan suara dengan JST yaitu pertama kata sandi ‘abcdefg’ direkam menggunakan mikrofon terlebih dahulu. Kemudian data suara dioleh pada laptop menggunakan software matlab versi 2013 untuk dilakukan pengolahan ekstraksi ciri suara menggunakan metode MFCC. Kemudian data dikirim pada *board* arduino menggunakan antarmuka USB, setelah itu digunakan sebagai data masukan atau perintah oleh relay yang diteruskan ke selenoid kunci pintu. Perintah atau masukan tersebut digunakan untuk membuka kunci pintu. Gambar 3.1 merupakan blok diagram cara kerja dari sistem.



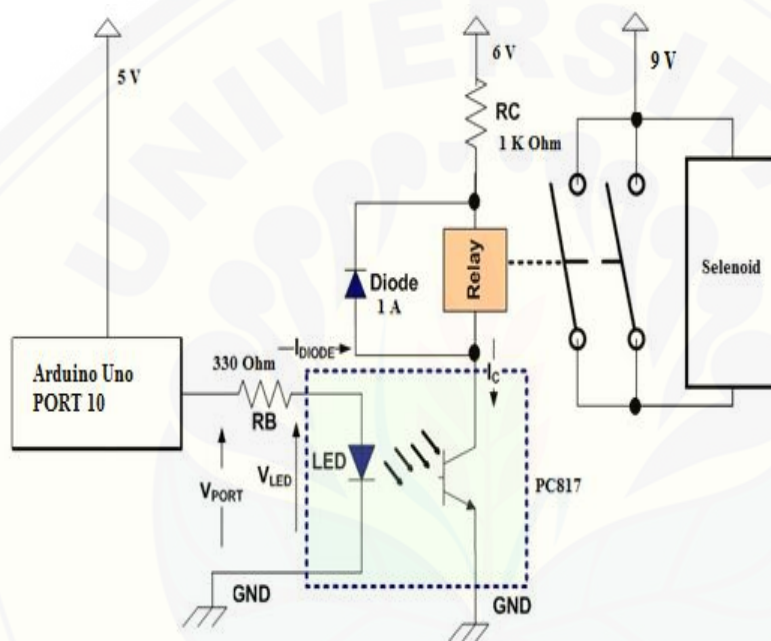
Gambar 3.1 Blok Diagram Sistem

Sedangkan pembuatan perangkat keras ini terdiri dari arduino, relay dan selenoid sebagai pengunci brankas yang ditunjukkan pada gambar 3.2



Gambar 3.2 Brankas dan kotak arduino

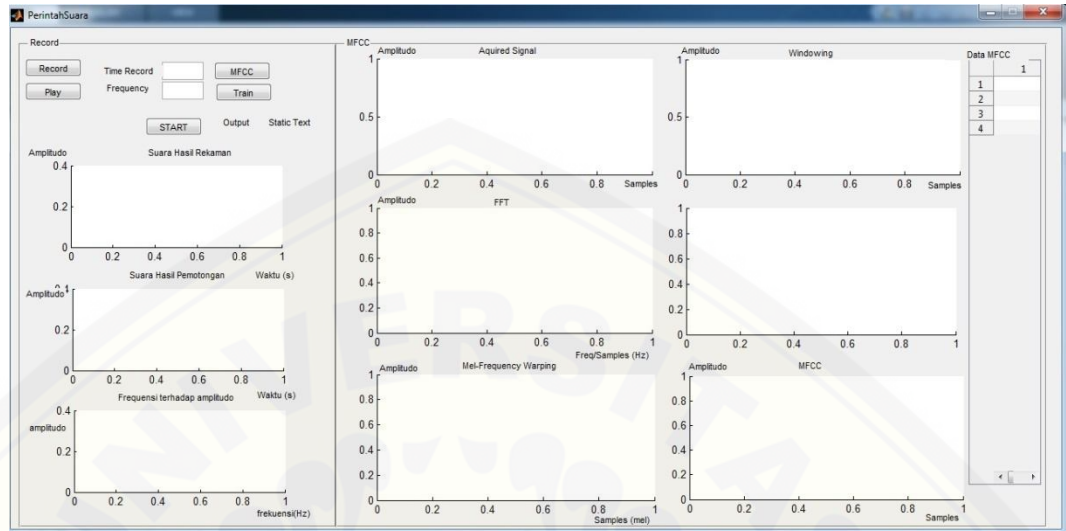
3.4.1 Rangkaian Driver Selenoid



Gambar 3.3 Rangkaian Driver Selenoid

Driver selenoid yang digunakan terdiri dari relay 6 V dan optocoupler PC817 sebagai pengaman mikrokontroler dari arus balik yang ditimbulkan relay. Input driver selenoid berupa tegangan 5 V yang dikeluarkan oleh mikrokontroler. Driver selenoid akan aktif atau brankas terbuka ketika driver selenoid menerima tegangan 5 V dari mikrokontroler. Sebaliknya driver selenoid tidak aktif atau brankas terkunci ketika menerima tegangan 0 V dari mikrokontroler.

3.5 Aplikasi Pengenalan Suara



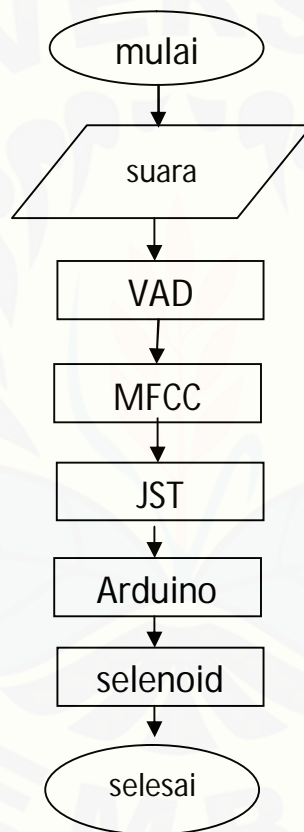
Gambar 3.4 Aplikasi Pengenalan suara

Aplikasi pengenalan suara ini dibuat untuk mendapatkan suara sebagai masukan terhadap jaringan syaraf tiruan yang diolah terlebih dahulu menggunakan metode *Mel-Frequency Cepstrum Coefficients* (MFCC). MFCC merupakan suatu metode yang cara bekerjanya seperti telinga manusia. MFCC mampu merubah suara analog dan membentuknya menjadi suatu pola yang dapat dikenali. Pola-pola dari setiap kata digunakan sebagai masukan jaringan syaraf tiruan sebagai data pembelajaran.

Aplikasi ini dibuat menggunakan *software* Matlab R2013a. Didalam aplikasi yang telah dibuat terdapat 5 fungsi utama yaitu *record*, *play*, *MFCC*, *Train* dan *Start*. fungsi *record* digunakan untuk merekam suara perintah yang diucapkan. Fungsi *play* digunakan untuk memutar ulang hasil suara yang telah direkam. Fungsi *MFCC* digunakan sebagai perubah suara yang telah direkam kedalam pola-pola yang nantinya digunakan sebagai masukan jaringan syaraf tiruan. Fungsi *Train* digunakan untuk melakukan pembelajaran jaringan syaraf tiruan dari hasil masukan dari MFCC yang ditentukan target keluaran sebelumnya pada jaringan syaraf tiruan. Fungsi *Start* digunakan untuk menguji dengan sebuah masukan perintah suara 'abcdefg' dengan mengisi

terlebih dahulu *time record* dan *frequency*, setelah itu aplikasi akan menampilkan pesan kondisi brankas apakah terbuka atau perintah tidak dikenali. Sedangkan fungsi masing-masing grafik menjelaskan sinyal yang diperoleh saat dilakukan MFCC sesuai dengan keterangan masing-masing grafik.

3.6 Flowchart Sistem Kerja Alat Secara Keseluruhan



Gambar 3.5 Flowchart alat keseluruhan

Sinyal suara terlebih dahulu diinisialisasi, prose penginisialisasian yaitu dengan merekam suara menggunakan mikropon yang ada pada komputer. Sinyal suara direkam selama 2 detik dengan fs sebesar 8000Hz. Sinyal suara yang telah direkam, kemudian ditampilkakan dikomputer.

Sinyal suara yang direkam kemudian dipotong untuk dideteksi bagian suara saja. VAD (*Voice Activity Detection*) digunakan pada bagian ini. Proses selanjutnya adalah menentukan jumlah frame dari sampel sinyal (n_{Frame}). Panjang frame yang digunakan sebesar 256. Sinyal yang digunakan sebagai proses awal pengenalan pola suara berdasarkan jumlah frame dari sampel sinyal selanjutnya adalah hamming window atau yang sering disebut dengan windowing. Windowing digunakan sebagai bentuk jendela dengan mempertimbangkan blok atau kata berikutnya dalam fitur rantai pengolahan ekstraksi dan mengintegrasikan semua lini frekuensi yang paling dekat. Fast Fourier Transform (FFT) digunakan mengkonversi sinyal dari domain waktu ke domain frekuensi. FFT dapat mengetahui besarnya respon frekuensi setiap frame.

Range frekuensi yang dihasilkan FFT spectrum terlalu lebar dan membuat sinyal suara tidak dapat mengikuti skalanya. Oleh karena itu, untuk setiap nada dengan frekuensi (diukur dalam Hz), pitch subjektif diukur pada skala yang disebut "mel" skala. Skala mel frekuensi adalah frekuensi linier berada dibawah 1000 Hz dan logaritmik diatas 1000Hz.

Spektrum subjektif disimulasikan dengan menggunakan filter bank, satu filter untuk setiap komponen frekuensi mel yang diinginkan. Filter bank memiliki band segitiga respon frekuensi dan jarak serta bandwidth ditentukan dengan interval frekuensi mel konstan.

Dalam langkah terakhir ini, spektrum log mel dirubah kembali ke waktu. Hasilnya disebut Mel Frequency Cepstrum Coefisien (MFCC). Karena koefisien spektrum mel adalah bilangan real, maka perlu mengkonversikannya ke domain waktu menggunakan Discrete cosine transform (DCT). DCT dilakukan untuk mengubah koefisien mel kembali ke domain waktu. DCT

Setelah proses MFCC maka dilakukan proses training pada JST dengan topologi seperti pada gambar 3.6

BAB 5. PENUTUP

Dari hasil perancangan, pembuatan, pengujian dan analisa sistem pada penelitian dengan judul “Rancang Bangun *Prototype* Sistem Keamanan Brankas Uang Dengan Pengenal Suara Menggunakan Metode Jaringan Syaraf Tiruan” ini, penulis mendapatkan kesimpulan serta memberikan saran yang akan berguna bagi pengembangan penelitian ini

5.1 Kesimpulan

1. Pada penelitian ini berdasarkan data pada pengujian keseluruhan alat didapatkan tingkat keberhasilan mencapai 100% dengan menggunakan kata “abcdefg” untuk membuka brankas.
2. Pada pengujian selanjutnya dilakukan pembukaan brankas menggunakan kata selain kata “abcdefg” dan dihasilkan tingkat keberhasilan sebesar 90% dengan acuan brankas tetap dalam kondisi terkunci sesuai dengan yang diinginkan peneliti.
3. Berdasarkan data yang diperoleh ketika brankas membuka terdapat kesamaan grafik pada FFT dan MFCC sehingga JST mengenali pola dengan baik dan menghasilkan output sesuai dengan target yaitu 10 dan 11 untuk membuka brankas.
4. Terdapat perbedaan grafik yang dihasilkan antara kata “abcdefg” dengan kata selain “abcdefg” sehingga output yang dihasilkan oleh JST sesuai target yaitu 01 dan 00 untuk kondisi brankas terkunci

5.2 Saran

1. Kualitas mikrofon mempengaruhi data yang diambil saat perekaman suara.
2. Metode ekstraksi ciri suara sangat berperan penting terhadap pola yang dibentuk dari suara yang ingin diujikan sehingga diharapkan dapat menggunakan metode lain untk pengembangan sistem keamanan brankas kedepannya.
3. Pada penelitian selanjutnya diharapkan *prototype* sistem keamanan pada brankas dapat menggunakan media lain selain suara misalnya sidik jari.

DAFTAR PUSTAKA

- Aditya, Reza. 2009. *Prototipe Pengenalan Suara Sebagai Penggerak Dinamo Starter Pada Mobil*. Depok . Jurusan Teknik Mesin Fakultas Teknologi Industri Universitas Gunadarma.
- Budhiharto, Widodo, Rizal, Gamayel. 2006. *12 Proyek Mikrokontroler untuk Pemula*. Jakarta : PT Elex Media Komputindo.
- Fadlisyah, Bustami, Ikhwanus. 2013. *Pengolahan Suara*. Yogyakarta: Graha Ilmu
- Febriana, Kurnia. 2004. *Aplikasi Jaringan Syaraf Tiruan Backpropogation Untuk Pengenalan Wajah*. Surabaya. Politeknik Elektronika Negeri Surabaya-ITS.
- .Furui, S, 1989. *Digital Speech Processing, Synthesis, and Recognition*, New York : Marcel Dekker Inc.
- Ludeman, L.C.1986, *Fundamentals of Digital Signal Processing*, Newyork: Happer & Row Publishers,
- Mustofa, Ali. 2007. *Sistem Pengenalan Penutur Dengan Metode Mel-Frequency Wrapping*. Malang : Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya
- Rabiner L.R dan Juang B. H, 1993. *Fundamentals of Speech Recognition*, Prentice Hall, Englewood Cliffs, N.J
- <http://www.youtube.com/user/18F4550videos/videos> [Diakses pada tanggal 19 September 2015]
- <http://youtube.com/2014/01/Artifiacial Neural Network #> [Diakses tanggal 20 September 2015]

LAMPIRAN – LAMPIRAN

Listig Program Keseluruhan.

```
function varargout = PerintahSuara(varargin)
% PERINTAHSUARA MATLAB code for PerintahSuara.fig
%     PERINTAHSUARA, by itself, creates a new PERINTAHSUARA or
raises the existing
%     singleton*.
%
%     H = PERINTAHSUARA returns the handle to a new PERINTAHSUARA
or the handle to
%     the existing singleton*.
%
%     PERINTAHSUARA('CALLBACK',hObject,eventData,handles,...)
calls the local
%     function named CALLBACK in PERINTAHSUARA.M with the given
input arguments.
%
%     PERINTAHSUARA('Property','Value',...) creates a new
PERINTAHSUARA or raises the
%     existing singleton*. Starting from the left, property
value pairs are
%     applied to the GUI before PerintahSuara_OpeningFcn gets
called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to PerintahSuara_OpeningFcn
via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help PerintahSuara

% Last Modified by GUIDE v2.5 17-Apr-2015 13:33:05

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @PerintahSuara_OpeningFcn,
                  ...
                  'gui_OutputFcn',  @PerintahSuara_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
```



```
if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before PerintahSuara is made visible.
function PerintahSuara_OpeningFcn(hObject, eventdata, handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to PerintahSuara (see
VARARGIN)

% Choose default command line output for PerintahSuara
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes PerintahSuara wait for user response (see UIRESUME)
% uiwait(handles.figure1);
clear all;
global a;
%
%
% a= arduino('COM4');
% a.pinMode(10,'output');
% a.pinMode(12,'output');
% a.pinMode(11,'output');

% --- Outputs from this function are returned to the command line.
function varargout = PerintahSuara_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

```

global signal
global signal1
global waktu
global fp

if isempty(waktu) || isempty(fp)
    error('cek salah satu atau keduanya');
    return;
else
    h= waitbar(0, 'mulai bicara sekarang...!');
    for i=1 :1
        data= wavrecord(fp*waktu,fp);
        waitbar(i/1,h)
    end
    waitbar(1,h, 'simpan sementara');
    close(h);
    signal1=0.99*data/max(abs(data));
    signal = myVAD(signal1);

    t1=[0:length(signal1)-1]/fp;
    t=fp*[0:length(signal1)-1]/length(signal1);
    plot(handles.axes1,t1,signal1)
    set(get(gca, 'XLabel'), 'String', 'Waktu (det)')
    xlabel('Waktu (det)'),ylabel('Amplitudo')
%
    t0=(1:length(signal))/fp;
    plot(handles.axes2,t0,signal);
    plot(handles.axes3,t,(signal1));
    xlabel('frek(hz)'),ylabel('Amplitudo')
% plotwav;
end
% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
global signal1
global fp
if ~isempty(signal1)
    wavplay(signal1,fp)
t=(0.1:length(signal1))/fp;

end

function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%        str2double(get(hObject,'String')) returns contents of
edit1 as a double

```

```
global waktu
    waktu=get(hObject,'String');
    waktu=str2double(waktu);

% --- Executes during object creation, after setting all
properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_Callback(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit2 as text
%       str2double(get(hObject,'String')) returns contents of
edit2 as a double
global fp
    fp=get(hObject,'String');
    fp=str2double(fp);

% --- Executes during object creation, after setting all
properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
```

```
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
global signal

i = 1;
j = 1;
x=signal;
[m,n] = size(x);
fs = 8000;
timeLength = m/fs; %About 1/2 a second
framelength = 256;

%overlapping
% nFrames = floor(n-m);
nFrames = floor((m-128)/(framelength-128));

%fclose(x);
% figure(1);
% plot(nFrame)
sampNo = 1;
for frame = 1 : nFrames,

    aquiredData = x(i:i+256-1);
    i = i + 128;
    %     figure(2);
    %     plot(aquiredData);
    %     %pause;

    plot(handles.axes5,aquiredData)
    % Hamming windowing
    windowing = (hamming(256).*aquiredData);
    %     figure(3);
    %     plot(windowing)
    plot(handles.axes6,windowing)
    % FFT
    FFTofData = fft(windowing);
    %     FFTofData1 = fft(windowing,fs);
    %     figure(4);
    %     plot(abs(FFTofData))
    plot(handles.axes7,abs(FFTofData))
    FFTofData = FFTofData(1:end/2);
    %     figure(1);
    %     plot(abs(FFTofData)) % First 128 frames of the sample
    plot(handles.axes8,abs(FFTofData))
    W=melFilterMatrix(8000,256,19);
    FilteredData = W*(abs(FFTofData));
    %     figure (6);
    %     plot(FilteredData)
    plot(handles.axes9,FilteredData)
    MFCC = dct(FilteredData);

    figure(7);
```

```
plot (MFCC)

    plot(handles.axes10, MFCC);
    myform = guidata(gcbo);
    set(myform.table1, 'data', MFCC);
end

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
global net
global c
global s
fileID=fopen('E:\skripsi_nurdian\abcd.txt');
c = fscanf(fileID, '%f', [19 inf]);
fclose(fileID);
fileID=fopen('E:\skripsi_nurdian\keluar.txt');
d = fscanf(fileID, '%f', [2 inf]);
fclose(fileID);

s= [20 2];
net =
newff(minmax(c),s,{'tansig','tansig'},'trainlm','learngdm','mse');

net.trainParam.epochs =1000;
net.trainParam.goal =0.001;
net.trainParam.show =1;
net.trainParam.lr=0.1;
net=train(net,c,d);

Ya = sim(net,c)

function edit3_Callback(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit3 as text
%        str2double(get(hObject,'String')) returns contents of
edit3 as a double

% --- Executes during object creation, after setting all
properties.
function edit3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
```



```
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton8 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
global signal
global signal1
global waktu
global fp
if isempty(waktu)||isempty(fp)
    errordlg('cek salah satu atau keduanya');
    return;
else
    h= waitbar(0,'mulai bicara sekarang....!');
    for i=1 :1
        data= wavrecord(fp*waktu,fp);
        waitbar(i/1,h)
    end
    waitbar(1,h,'simpan sementara');
    close(h);
    signal1=0.99*data/max(abs(data));
    signal = myVAD(signal1);
    t1=(1:length(signal1))/fp;
    plot(handles.axes1,t1,signal1)
    xlabel('Waktu (det)'),ylabel('Amplitudo')
    t2=(1:length(signal))/fp;
    plot(handles.axes12,t2,signal)
    xlabel('Waktu (det)'),ylabel('Amplitudo')
    t=fp*[0:length(signal1)-1]/length(signal1);
    plot(handles.axes13,t,signal1)
    xlabel('Freq (Hz)'),ylabel('Amplitudo')
    i = 1;
j = 1;
x=signal;
[m,n] = size(x);
fs = 8000;
timeLength = m/fs; %About 1/2 a second
framelength = 256;

%overlapping

nFrames = floor((m-128)/(framelength-128));
```



```
%fclose(x);

sampNo = 1;
for frame = 1 : nFrames,

    aquiredData = x(i:i+256-1);
    i = i + 128;
    figure(2);
    plot(aquiredData);
    %pause;

    plot(handles.axes5,aquiredData)
    % Hamming windowing
    windowing = (hamming(256).*aquiredData);
    figure(3);
    plot(windowing)
    plot(handles.axes6,windowing)
    % FFT
    FFTofData = fft(windowing);
    figure(4);
    plot(abs(FFTofData))
    plot(handles.axes7,abs(FFTofData))
    FFTofData = FFTofData(1:end/2);
    figure(1);
    plot(abs(FFTofData)) % First 128 frames of the sample
    plot(handles.axes8,abs(FFTofData))
    W=melFilterMatrix(8000,256,19);
    FilteredData = W*(abs(FFTofData));
    figure (6);
    plot(FilteredData)
    plot(handles.axes9,FilteredData)
    MFCC = dct(FilteredData);

    figure(7);
    plot (MFCC)

    plot(handles.axes10, MFCC);
    myform = guidata(gcbo);
    set(myform.table1,'data',MFCC);
end
end
global net
global c
global s
global Y
global a
F = MFCC;

weight = getx(net);

net =
newff(minmax(c),s,{'tansig','tansig'},'trainlm','learngdm','mse');
net = setx(net, weight);
```

```
Y = sim(net,F);
Y = round(Y)
set(handles.text5,'string',Y);
guidata(hObject,handles);

if Y == [1;0]
    z = msgbox('Barangkas Terbuka');
    a.digitalWrite(10,1);

else if Y == [0;1]
    z = msgbox('Berangkas Terkunci');
    a.digitalWrite(10,0);

else if Y == [1;1]
    z = msgbox('Berankas Terbuka');
    a.digitalWrite(10,1);
else
    z = msgbox('Berankas Terkunci');
    a.digitalWrite(10,0);

end
end
end
```