

## **IMPLEMENTASI PENGENALAN POLA SUARA MENGGUNAKAN *MEL-FREQUENCY CEPSTRUM COEFFICIENTS* (MFCC) DAN *ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM* (ANFIS) SEBAGAI KONTROL LAMPU OTOMATIS**

**Mada Sanjaya W.S<sup>\*1,2</sup>, Zabidin Salleh<sup>3</sup>**

<sup>1</sup>Bolabot Techno Robotic Institute,  
CV. Sanjaya Star Group, Bandung, INDONESIA

<sup>2</sup>Jurusan Fisika, Fakultas Sains dan Teknologi  
Universitas Islam Negeri Sunan Gunung Djati Bandung, INDONESIA

<sup>3</sup>Faculty of Science and Technology  
Universiti Malaysia Terengganu, Kuala Terengganu, MALAYSIA

\*E-mail : [madasws@gmail.com](mailto:madasws@gmail.com)

Halaman 43-54

Diterima: 1 Desember 2014, Direview: 12 Desember 2014 ,Dipublikasi: 25 Desember 2014

### **ABSTRAK**

Penelitian ini menggambarkan implementasi pengenalan pola suara untuk mengontrol nyala dan mati lampu AC secara otomatis. Metode pengenalan pola suara yang digunakan dalam penelitian ini adalah *Mel-Frequency Cepstrum Coefficients* (MFCC) dan *Adaptive Neuro-Fuzzy Inference System* (ANFIS). Metode LPC digunakan untuk ekstraksi ciri sinyal suara dan ANFIS digunakan sebagai metode pembelajaran untuk pengenalan pola suara. Data latih yang digunakan pada proses pembelajaran ANFIS sebanyak 6 ciri. Pengujian sistem pengenalan pola suara dilakukan menggunakan data suara terlatih dan data suara tak terlatih. Hasil pengujian menunjukkan tingkat keberhasilan untuk data suara terlatih sebesar 98,57 % dan data tak terlatih sebesar 95,90%. Sistem pengenalan pola suara ini telah diaplikasikan dengan baik untuk menyalakan dan mematikan lampu AC berbasis mikrokontroler Arduino.

**Kata Kunci:** Pengenalan suara, *Mel-Frequency Cepstrum Coefficients*, ANFIS, Sistem Kontrol.

### **PENDAHULUAN**

Sistem otomatisasi terutama robotika merupakan sistem yang dapat mempermudah pekerjaan manusia. Sistem robotika dapat dikendalikan secara langsung oleh manusia ataupun menggunakan program yang telah terdefiniskan oleh prosesor robot. Sistem kendali dengan menggunakan suara bisa menjadi salah satu alternatif yang mudah dan efektif terutama untuk para pengguna yang memiliki keterbatasan fisik. Namun penerapan sistem kendali menggunakan suara tidak mudah dilakukan oleh mesin. Diperlukan suatu metode pembelajaran pada mesin untuk dapat mengekstraksi dan mengenali ciri atau pola suara dengan mempelajari ciri-ciri sebelumnya.

Terdapat beberapa metode untuk mengekstrak ciri sinyal suara diantaranya menggunakan metode *Linear Predictive Coding* [1]–[6] dan *Mel-Frequency Cepstrum Coefficients* [7][1], [8]–[11] yang telah diaplikasikan pada beberapa bidang diantaranya, mendiagnosis suatu penyakit [3], [12], [13], kontrol motor DC [14], kontrol robot [15]–[21], kontrol otomatis kursi roda[22], [23], biomatriks [24], pengenalan pembicara [9], [11]. Untuk metode pembelajaran dan pengklasifikasian pola suara, telah banyak peneliti yang menggunakan metode *Hidden Markov Model* [7][6], [13], [20], *Artificial Neural Networks* [3]–[5], [25], [26], *Neuro-Fuzzy* [27], [28] dan metode *soft computing* lainnya[27].

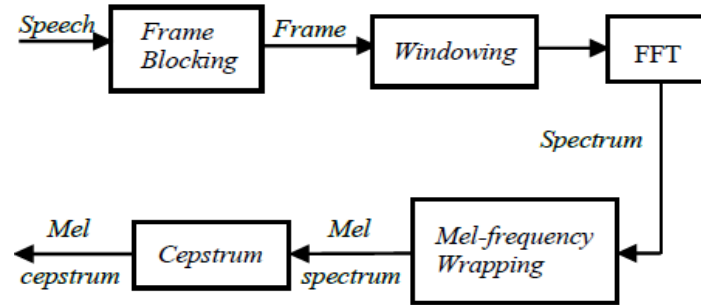
Dalam makalah ini, akan dibahas pengujian sistem ekstraksi ciri sinyal suara menggunakan *Mel-Frequency Cepstrum Coefficients* (MFCC) dan metode belajar serta klasifikasi pola suara menggunakan metode *Adaptive Neuro-Fuzzy Inference System* (ANFIS). Terakhir dalam makalah ini

juga dilakukan implementasi sistem pengenalan pola suara untuk otomatisasi kontrol lampu AC menggunakan mikrokontroler Arduino.

## LANDASAN TEORI

### Metode MFCC

MFCC merupakan cara yang paling sering digunakan pada berbagai bidang area pemrosesan suara, karena dianggap cukup baik dalam merepresentasikan sinyal. Cara kerja MFCC didasarkan pada perbedaan frekuensi yang dapat ditangkap oleh telinga manusia sehingga mampu merepresentasikan sinyal suara sebagaimana manusia merepresentasikannya. Blok diagram proses MFCC dapat dilihat pada gambar berikut:



Gambar 1. Blok diagram metode MFCC

### Preemphasis

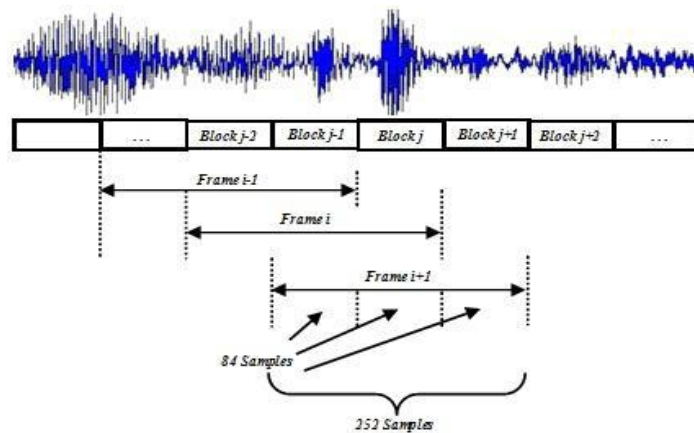
Dalam proses pengolahan sinyal wicara *pre emphasis* filter diperlukan setelah proses sampling. Tujuan dari pemfilteran ini adalah untuk mendapatkan bentuk *spectral* frekuensi sinyal wicara yang lebih halus. Dimana bentuk *spectral* yang relatif bernilai tinggi untuk daerah rendah dan cenderung turun secara tajam untuk daerah fekuensi diatas 2000 Hz. Filter *pre-emphasis* didasari oleh hubungan input/output dalam domain waktu yang dinyatakan dalam persamaan berikut:

$$y(n) = x(n) - ax(n - 1) \tag{1}$$

dimana  $a$  merupakan konstanta filter *pre-emphasis*, biasanya bernilai  $0.9 < a < 1.0$ .

### Frame Blocking

Pada proses ini, sinyal suara disegmentasi menjadi beberapa *frame* yang saling tumpang tindih (*overlap*). Hal ini dilakukan agar tidak ada sedikitpun sinyal yang hilang (*deletion*). Proses ini akan berlanjut sampai seluruh sinyal sudah masuk ke dalam satu atau lebih *frame* seperti yang diilustrasikan dalam Gambar 2.



Gambar 2. Ilustrasi *frame blocking* pada sinyal suara.

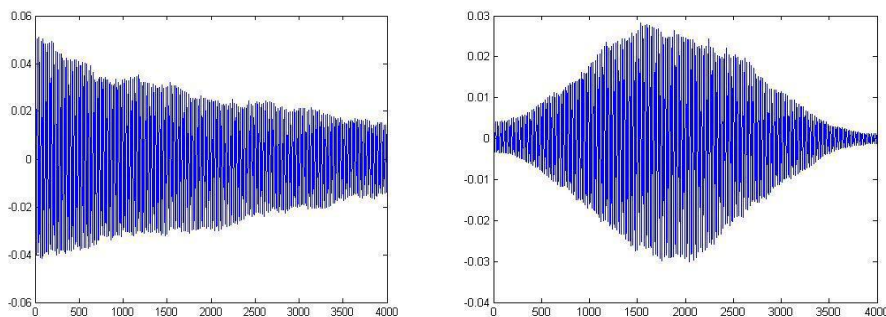
## Windowing

Sinyal analog yang sudah diubah menjadi sinyal digital dibaca *frame* demi *frame* dan pada setiap *frame*-nya dilakukan *windowing* dengan fungsi *window* tertentu. Proses *windowing* bertujuan untuk meminimalisasi ketidakberlanjutan sinyal pada awal dan akhir setiap *frame*. Jika kita definisikan *window* sebagai  $w(n)$ ,  $0 \leq n \leq N - 1$ , dimana  $N$  adalah jumlah sampel pada setiap *frame*-nya, maka hasil dari *windowing* adalah sinyal:

$$y_1(n) = x_1(n)w(n), 0 \leq n \leq N - 1 \quad (2)$$

dimana  $w(n)$  biasanya menggunakan *window* Hamming yang memiliki bentuk:

$$w(n) = 0.54 - 0.46 \cdot \cos\left(\frac{2\pi n}{N-1}\right), 0 \leq n \leq N - 1 \quad (3)$$



(a) Sebelum *windowing*                      (b) Setelah *windowing*

**Gambar 3.** Ilustrasi proses *windowing*

## Fast Fourier Transform (FFT)

FFT merupakan *fast algorithm* dari *Discrete Fourier Transform* (DFT) yang berguna untuk mengonversi setiap *frame* dengan  $N$  sampel dari domain waktu menjadi domain frekuensi, sebagaimana didefinisikan sebagai berikut

$$X_n = \sum_{k=0}^{N-1} x_k e^{-2\pi jkn/N} \quad (4)$$

dimana  $n = 0, 1, 2, \dots, N - 1$  dan  $j = \sqrt{-1}$ .

Hasil dari tahapan ini biasanya disebut dengan *spectrum* atau *periodogram*.

## Mel-Frequency Wrapping

Persepsi sistem pendengaran manusia terhadap frekuensi sinyal suara tidak dapat diukur dalam skala linear. Untuk setiap nada dengan frekuensi aktual,  $f$ , diukur dalam Hz, sebuah *subjective pitch* diukur dalam sebuah skala yang disebut “mel”. Skala *mel-frequency* ialah sebuah frekuensi rendah yang bersifat linear di bawah 1000 Hz dan sebuah frekuensi tinggi yang bersifat logaritmik di atas 1000 Hz. Persamaan berikut menunjukkan hubungan skala mel dengan frekuensi dalam Hz:

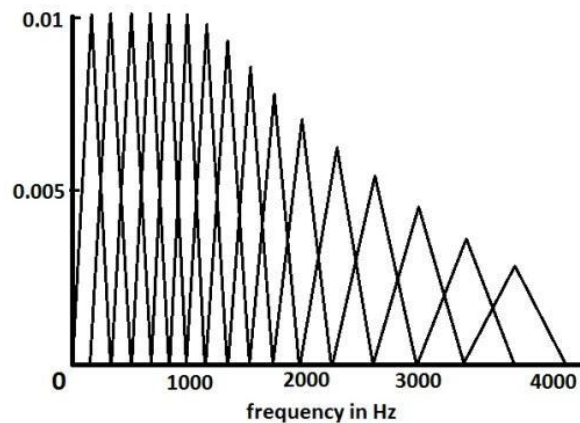
$$F_{mel} = \begin{cases} 2595 * \log_{10} \left( 1 + \frac{F_{HZ}}{700} \right), & F_{HZ} > 1000 \\ F_{HZ}, & F_{HZ} < 1000 \end{cases} \quad (5)$$

Proses *wrapping* terhadap sinyal dalam domain frekuensi dilakukan menggunakan persamaan berikut:

$$X_i = \log_{10} \left( \sum_{k=0}^{N-1} |X(k)| H_i(k) \right) \quad (6)$$

dimana  $i = 1, 2, 3, \dots, M$  ( $M$  adalah jumlah filter segitiga) dan  $H_i(k)$  adalah nilai filter segitiga ke- $i$  untuk

frekuensi akustik sebesar  $k$ .



Gambar 4. Filter frekuensi “mel”

### Cepstrum

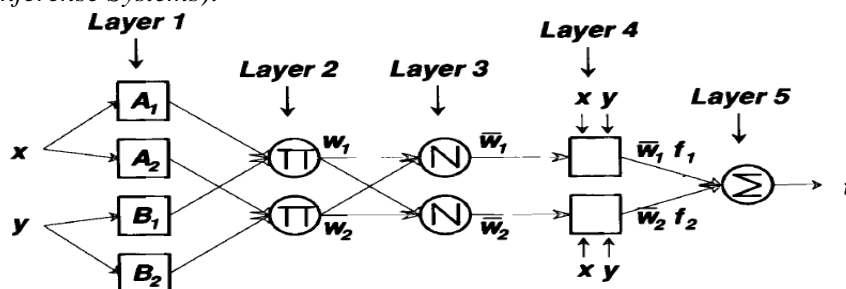
Pada tahap ini akan dikonversi *mel-spectrum* ke dalam domain waktu dengan menggunakan *Discrete Cosine Transform* (DCT). Hasilnya disebut dengan *mel-frequency cepstrum coefficient* (MFCC). Berikut adalah persamaan yang digunakan dalam transformasi cosinus:

$$C_j = \sum_{i=1}^M X_i \cos\left(j(i-1)/2 \frac{\pi}{M}\right) \quad (7)$$

dimana  $j = 1, 2, 3, \dots, K$  ( $K$  adalah jumlah koefisien yang diinginkan) dan  $M$  adalah jumlah filter.

### Adaptive Neuro Fuzzy Inferense System (ANFIS)

Neuro-fuzzy adalah gabungan dari dua sistem yaitu sistem logika fuzzy dan jaringan syaraf tiruan. Sistem neuro-fuzzy berdasar pada sistem inferensi fuzzy yang dilatih menggunakan algoritma pembelajaran yang diturunkan dari sistem jaringan syaraf tiruan. dengan demikian, sistem neuro-fuzzy memiliki semua kelebihan yang dimiliki oleh sistem inferensi fuzzy dan sistem jaringan syaraf tiruan. Dari kemampuannya untuk belajar maka sistem neuro-fuzzy sering disebut sebagai ANFIS (*Adaptive Neuro-Fuzzy Inferense Systems*).



Gambar 5. Contoh struktur ANFIS Roger Jang[27]

Salah satu bentuk struktur yang sudah sangat dikenal adalah seperti terlihat pada Gambar 5 yang merupakan struktur ANFIS. Dalam struktur ini, sistem inferensi fuzzy yang diterapkan adalah inferensi fuzzy model Takagi-Sugeno-Kang. Dalam sistem ANFIS sebagaimana Gambar 5 terdapat lima lapisan proses yang mana fungsi dan persamaan masing-masing lapisannya dijelaskan sebagai berikut:

#### Lapisan 1: Proses Fuzzyfication

Output dari *node*  $i$  pada *layer* 1 dinotasikan sebagai  $O_{1,i}$ . Setiap *node* pada *layer* 1 bersifat adaptif dengan output:

$$O_{1,i} = \mu_{A_i}(x), \quad i = 1,2 \quad (8)$$

$$O_{1,i} = \mu_{B_i}(y), \quad i = 1,2 \quad (9)$$

dimana  $x$  dan  $y$  adalah nilai-nilai input untuk *node* tersebut dan  $A_i$  dan  $B_i$  adalah himpunan *fuzzy*. Jadi, masing-masing *node* pada *layer* 1 berfungsi membangkitkan derajat keanggotaan.

### Lapisan 2: Lapisan Product

Dinotasikan dengan  $\pi$ . Setiap *node* pada *layer* ini berfungsi untuk menghitung kekuatan aktivasi (*firing strength*) pada setiap rule sebagai *product* dari semua input yang masuk atau sebagai operator *t-norm* (*triangular norm*):

$$O_{2,i} = w_i = \mu_{A_i}(x) \Delta \mu_{B_i}(y), \quad i = 1,2 \quad (10)$$

Sehingga

$$w_1 = \mu_{A_1}(x) \text{ AND } \mu_{B_1}(y) \quad (11)$$

$$w_2 = \mu_{A_2}(x) \text{ AND } \mu_{B_2}(y) \quad (12)$$

Output pada lapisan ini bertindak sebagai fungsi bobot.

### Lapisan 3: Lapisan Normalisasi

Dinotasikan dengan  $N$ . Setiap *node* pada lapisan ini bersifat *non-adaptif* yang berfungsi hanya untuk menghitung rasio antara *firing strength* pada rule ke- $i$  terhadap total *firing strength* dari semua rule:

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1,2 \quad (13)$$

### Lapisan 4: Lapisan Defuzzyfication

Setiap *node* pada lapisan ini bersifat adaptif dengan fungsi:

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i) \quad (14)$$

dimana  $\bar{w}_i$  adalah output pada *layer* 3 dan  $\{x + q_i y + r_i\}$  adalah himpunan parameter pada *fuzzy* model Sugeno orde pertama.

### Lapisan 5: Lapisan Total Output

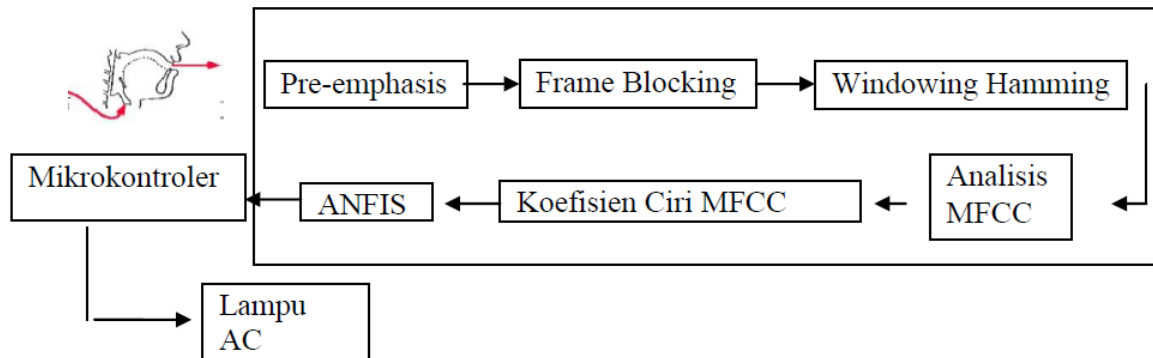
Satu *node* tunggal yang dilambangkan dengan  $\Sigma$  pada *layer* ini berfungsi mengagregasikan seluruh output pada *layer* 4 (penjumlahan dari semua sinyal yang masuk):

$$O_{5,i} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (15)$$

Sehingga secara keseluruhan, kelima *layer* tersebut akan membangun suatu *adaptive-networks* yang secara fungsional ekuivalen dengan *fuzzy* model Sugeno orde pertama.

## METODOLOGI DAN PERANCANGAN SISTEM

Peralatan yang digunakan pada penelitian yaitu *microphone* -54db+2 db dengan spesifikasi (impedansi 32 ohm at 1 kHz sensitivitas 102db/mW), laptop, mikrokontroler arduino, relay, lampu AC. Kabel USB dan komputer digunakan sebagai komunikasi data menggunakan *software* MATLAB 2012. Langkah – langkah penelitian secara garis besar dapat dilihat :

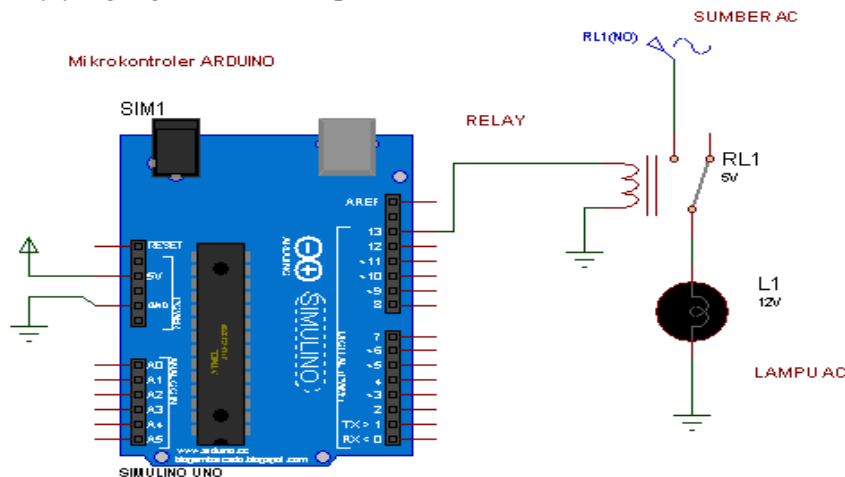


Gambar 6. Skema umum pengujian sistem

Terdapat dua bagian proses dalam penelitian ini, proses pertama adalah proses pembelajaran sistem yang didalamnya meliputi proses ekstraksi ciri menggunakan MFCC dan proses belajar menggunakan ANFIS sehingga terbentuk *database* berupa formulasi logika fuzzy. Proses kedua adalah proses pengujian sistem yang meliputi proses ekstraksi ciri MFCC, proses pengklasifikasian data ANFIS, serta implementasi berupa komunikasi data secara serial antara komputer dengan mikrokontroler Arduino untuk menyalakan atau mematikan lampu AC yang terhubung secara tak langsung menggunakan *relay* dengan Arduino.

### Perancangan Elektronik

Skema rangkaian elektronik kontrol lampu AC menggunakan mikrokontroler dapat dibuat menggunakan sistem relay maupun sistem optocoupler sebagai penghubung tak langsung antara komponen arus DC sebagai kontroler dan lampu AC. Berikut adalah skema kontrol lampu AC menggunakan relay yang digunakan dalam penelitian.



Gambar 7. Skema rangkaian mikrokontroler Arduino beserta relay dan lampu AC

Dari skema diatas terlihat bahwa input relay terkoneksi dengan pin13 dan ground dari mikrokontroler Arduino.Sedangkan pin output terhubung dengan sumber tegangan AC dan lampu AC. Bila pin 13 Arduino dalam kondisi HIGH maka saklar relay akan kondisi ON sehingga lampu AC akan menyala sedangkan jika pin 13 Arduino dalam kondisi LOW maka saklar relay akan kondisi OFF dan lampu AC akan mati. Kondisi HIGH dan LOW dari pin 13 Arduino inilah yang akan dikendalikan oleh sistem pengenalan suara di komputer yang terhubung secara serial dengan mikrokontroler Arduino.

### Perancangan Antarmuka

Antarmuka diperlukan untuk mempermudah penggunaan dari sistem yang dibuat. Pada penelitian ini telah dibuat antarmuka menggunakan GUI MATLAB. Antarmuka yang dibuat berisi setting komunikasi serial antara mikrokontroler dan komputer, nilai koefisien ekstraksi ciri menggunakan metode MFCC, grafik time series sinyal suara, tombol record suara, serta output ANFIS berupa pengenalan pola suara.

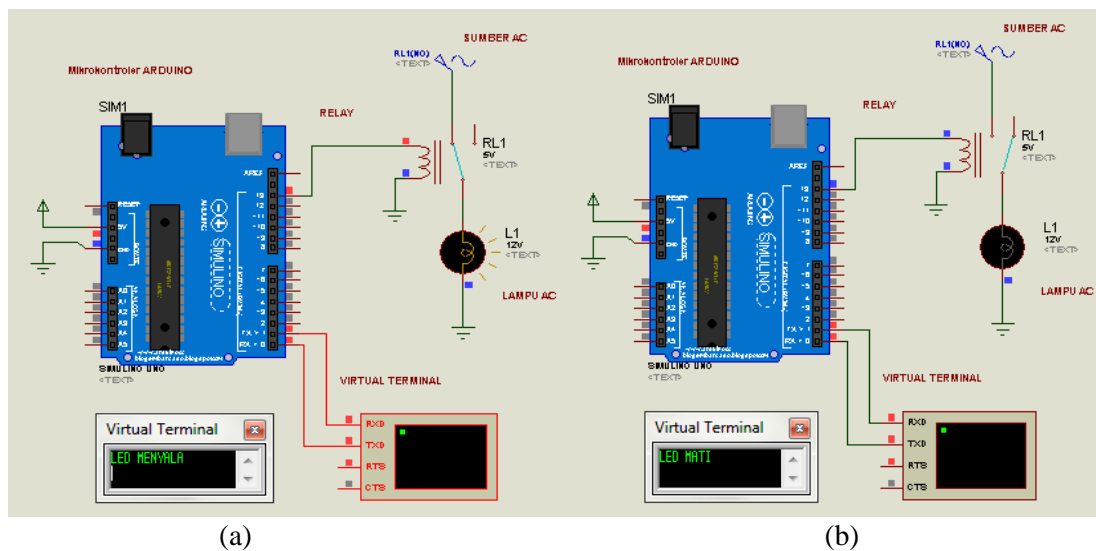


Gambar 8. Rancangan antarmuka

### HASIL DAN PEMBAHASAN

#### Pengujian Program IDE Arduino

Untuk menguji program IDE Arduino, telah dilakukan simulasi rangkaian menggunakan Proteus dan menunjukkan kesesuaian antara program dan output yang diharapkan.



Gambar 9. Pengujian program IDE Arduino kontrol lampu AC secara serial menggunakan komputer

#### Pembuatan Database Ekstraksi Ciri Menggunakan MFCC

Telah dilakukan pembuatan database 6 buah ciri /pola sinyal suara menggunakan metode MFCC terhadap seorang responden dengan mengucapkan dua buah kata “nyala” dan “mati” dengan masing-masing 10 kali pengulangan yang secara lengkap dapat dilihat pada tabel berikut:

**Tabel 1.** Hasil ekstraksi ciri MFCC dan target

| No | Ciri 1   | Ciri 2   | Ciri 3  | Ciri 4   | Ciri 5   | Ciri 6   | Target |
|----|----------|----------|---------|----------|----------|----------|--------|
| 1  | 8.51654  | 3.35706  | 1.96991 | -6.72733 | -1.40909 | -1.92471 | 1      |
| 2  | 4.44286  | 3.38998  | 3.20498 | -4.16123 | -2.05705 | -1.44783 | 1      |
| 3  | 5.20991  | 3.81779  | 2.08102 | -4.45775 | -2.39039 | -1.26394 | 1      |
| 4  | 2.61204  | 2.38381  | 1.66868 | -2.93557 | -1.69619 | -1.03670 | 1      |
| 5  | 6.24561  | 0.15679  | 2.11523 | -4.01245 | -1.74763 | -1.76225 | 1      |
| 6  | 7.39128  | 4.45092  | 2.34407 | -7.07256 | -0.80560 | -1.62316 | 1      |
| 7  | 3.38901  | 0.54230  | 1.89205 | -1.90688 | -2.94686 | -0.56638 | 1      |
| 8  | 5.26817  | 3.01692  | 2.47580 | -4.41381 | -2.05380 | -1.25286 | 1      |
| 9  | 2.98202  | 1.77261  | 1.23972 | -2.83055 | -0.95272 | -0.62437 | 1      |
| 10 | 1.50531  | 1.22102  | 1.03666 | -3.88563 | -1.03254 | -0.60021 | 1      |
| 11 | 1.20094  | -1.17752 | 0.51260 | -1.52752 | 1.54430  | -0.05302 | 0      |
| 12 | 0.84467  | -1.48441 | 0.08005 | -1.85271 | 1.88589  | -0.13855 | 0      |
| 13 | 3.12623  | -3.06120 | 0.00846 | -1.33876 | 3.67641  | -0.39186 | 0      |
| 14 | 6.28635  | -3.28296 | 1.56856 | -0.30715 | 5.35535  | -0.53721 | 0      |
| 15 | 0.49213  | -1.68155 | 0.24807 | 0.23254  | 0.32782  | -0.24038 | 0      |
| 16 | 3.68825  | -3.88391 | 0.57740 | 1.06655  | 2.54906  | -0.59497 | 0      |
| 17 | -0.04527 | -0.10768 | 0.11276 | -0.09129 | 0.05394  | -0.01695 | 0      |
| 18 | 6.48616  | 0.48602  | 2.55926 | -0.41534 | 3.63869  | 0.16244  | 0      |
| 19 | 1.21085  | -1.01180 | 0.28839 | -0.53142 | 1.06532  | -0.12306 | 0      |
| 20 | 3.73769  | 0.37519  | 0.39290 | 0.31807  | 0.18795  | 0.05907  | 0      |

Dari tabel di atas terlihat bahwa kata "nyala" dan "mati" memiliki pola ciri berbeda. Pola suara "nyala" akan di simbolkan dengan nilai target "1" sedangkan pola suara "mati" akan di simbolkan dengan nilai target "0". Pola ini kemudian akan menjadi input untuk metode pembelajaran ANFIS yang kemudian akan menghasilkan *database* berupa logika Fuzzy.

#### Pengujian Sistem Pengenalan Pola Suara

Hasil pengujian pengenalan ucapan menunjukkan bahwa total tingkat akurasi antara responden di dalam *database* sebesar 98,57 % dan responden di luar *database* menghasilkan akurasi sebesar 95,90%. Kesalahan-kesalahan pengenalan pola suara dapat dipengaruhi oleh aksent/intonasi kata masukan dan gangguan (*noise*) dari lingkungan mengingat waktu pengujian responden di dalam *database* dan responden di luar *database* yang berbeda.

**Tabel 2.** Hasil pengujian pengenalan ucapan responden di dalam *database*.

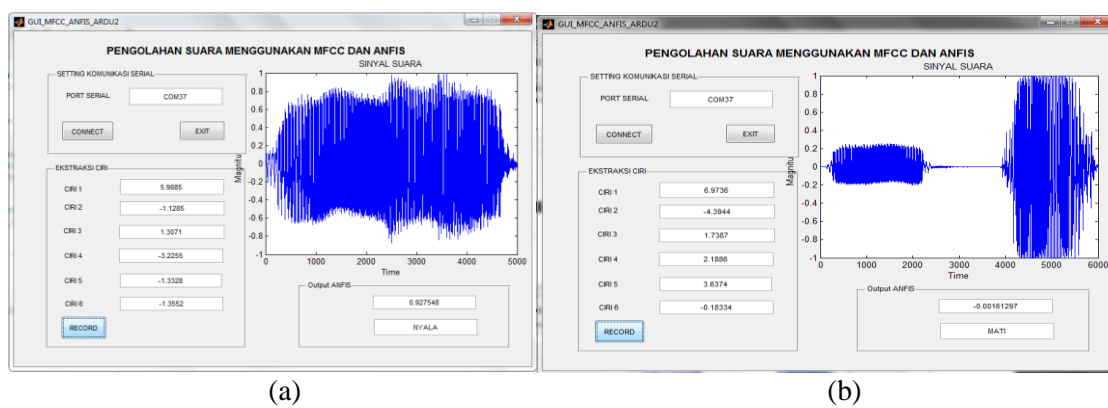
| Pengulangan            | Kata  | Jumlah |                   | Error     |
|------------------------|-------|--------|-------------------|-----------|
|                        |       | Target | Output Pengenalan |           |
| 1                      | Nyala | 1      | 1.02538           | -0.025380 |
|                        | Mati  | 0      | -0.0096827        | 0.009683  |
| 2                      | Nyala | 1      | 0.905543          | 0.094457  |
|                        | Mati  | 0      | -0.00332415       | 0.003324  |
| 3                      | Nyala | 1      | 1.03189           | -0.031890 |
|                        | Mati  | 0      | 0.0148632         | -0.014863 |
| 4                      | Nyala | 1      | 0.970055          | 0.029945  |
|                        | Mati  | 0      | -0.000731751      | 0.000732  |
| 5                      | Nyala | 1      | 0.914531          | 0.085469  |
|                        | Mati  | 0      | 0.00843288        | -0.008433 |
| <b>Error Rata-Rata</b> |       |        |                   | 0.014304  |



**Tabel 3.** Hasil pengujian pengenalan ucapan responden di luar *database*.

| Pengulangan            | Kata  | Jumlah |                   | Error    |
|------------------------|-------|--------|-------------------|----------|
|                        |       | Target | Output Pengenalan |          |
| 1                      | Nyala | 1      | 0.36648           | 0.36648  |
|                        | Mati  | 0      | -0.01032          | -0.01032 |
| 2                      | Nyala | 1      | 0.5               | 0.5      |
|                        | Mati  | 0      | -0.01418          | -0.01418 |
| 3                      | Nyala | 1      | -0.28539          | -0.28539 |
|                        | Mati  | 0      | 0.001404          | 0.001404 |
| 4                      | Nyala | 1      | -0.23661          | -0.23661 |
|                        | Mati  | 0      | -0.00046          | -0.00046 |
| 5                      | Nyala | 1      | 0.08913           | 0.08913  |
|                        | Mati  | 0      | -0.00015          | -0.00015 |
| <b>Error Rata-Rata</b> |       |        |                   | 0.040991 |

Berikut adalah tampilan GUI saat dilakukan pengujian terhadap kata "nyala" dan "mati". Dari Gambar 10 terlihat, saat kata "nyala" direcord maka sistem akan mengenali dengan output 0.927548 yang mendekati nilai "1" sehingga komputer akan mengirimkan data serial berupa karakter "1" ke mikrokontroler untuk menyalakan lampu AC. Sedangkan saat kata "mati" direcord maka sistem akan mengenali dengan output sebesar -0.00161297 yang mendekati nilai "0" sehingga komputer akan mengirimkan data serial berupa karakter "0" ke mikrokontroler untuk mematikan lampu AC. Sehingga secara umum alat berfungsi dengan baik 100% karena adanya pembulatan nilai pengenalan.



**Gambar 10.** Tampilan GUI MATLAB kontrol lampu AC menggunakan perintah suara

## KESIMPULAN

Berdasarkan hasil penelitian diatas dapat disimpulkan hasil pengujian gerak sistem menunjukkan bahwa sistem telah bekerja dengan baik sesuai dengan perintah yang diberikan. Dari hasil pengujian sistem pengenalan ucapan yang didapatkan penerapan metode MFCC dan ANFIS memiliki tingkat akurasi yang tinggi, yaitu 98,57 % untuk responden yang terdapat dalam data latih (*database*) dan 95,90% untuk responden diluar data latih (*database*). Dalam pengujian untuk mengontrol lampu AC, sistem dapat bekerja dengan baik 100% melalui pembulatan nilai output hasil pengenalan pola suara.

## DAFTAR PUSTAKA

- [1] N. Dave, "Feature Extraction Methods LPC , PLP and MFCC In Speech Recognition," *Int. J. Adv. Res. Eng. Technol.*, vol. 1, no. VI, pp. 1–5, 2013.
- [2] A. Vijayaraj and N. Velmurugan, "Limited Speech Recognition for Controlling Movement of Mobile Robot," *Int. J. Eng. Sci. Technol.*, vol. 2, no. 10, pp. 5275–5279, 2010.
- [3] A. Rizal, L. Anggraeni, and V. Suryani, "Pengenalan Suara Paru-Paru Normal Menggunakan LPC dan Jaringan Syaraf Tiruan Back-Propagation." Proceeding EECCIS 2006.
- [4] S. Wijoyo, "Speech Recognition Using Linear Predictive Coding and Artificial Neural Network for Controlling Movement of Mobile Robot," in *Proceedings of 2011 International Conference on Information and Electronics Engineering (ICIEE 2011)*, 2011, pp. 28–29.
- [5] S. Wijoyo, "Speech Recognition Using Linear Predictive Coding and Artificial Neural Network for Controlling Movement of Mobile Robot," *2011 Int. Conf. Inf. Electron. Eng.*, vol. 6, pp. 179–183, 2011.
- [6] Thiang and Wanto, "Speech Recognition Using LPC and HMM Applied for Controlling Movement of Mobile Robot," in *Seminar Nasional Teknologi Informasi 2010*, 2010.
- [7] A. F. Permana, "Implementasi Hidden Markov Model Untuk Aplikasi Pengenalan Ucapan Sebagai Kendali Gerak Robot Mobil," Institut Pertanian Bogor, 2011.
- [8] S. C. Joshi and A. N. Cheeran, "MATLAB Based Feature Extraction Using Mel Frequency Cepstrum Coefficients for Automatic Speech Recognition," *Int. J. Sci. Eng. Technol. Res.*, vol. 3, no. 6, pp. 1820–1823, 2014.
- [9] A. Mustofa, "Sistem Pengenalan Penutur dengan Metode Mel-Frequency Wrapping," *J. Tek. Elektro*, vol. 7, no. 2, pp. 88–96, 2007.
- [10] L. R. Mehta and A. S. Dabhade, "Comparison Study of MFCC and LPC for Marathi Isolated Word," *Int. J. Adv. Res. Electr. Electron. Instrum. Eng.*, vol. 2, no. 6, pp. 2133–2139, 2013.
- [11] V. Tiwari, "MFCC and Its Applications in Speaker Recognition," *Int. J. Emerg. Technol.*, vol. 1, no. 1, pp. 19–22, 2010.
- [12] L. Latumakulita, C. E. J. C. Montolalu, P. B. Alami, and S. S. Buatan, "Sistem Pakar Pendiagnosa Penyakit Ginjal," *J. Ilm. Sains*, vol. 11, no. 1, pp. 131–139, 2011.
- [13] A. P. Lestari, "Rancang Bangun Pengenalan Penyakit Darah Menggunakan Hidden Markov Model," Universitas Indonesia, 2008.
- [14] P. K. Sharma, B. R. Lakshminantha, and K. S. Sundar, "Real Time Control of DC Motor Drive Using Speech Recognition," *India Int. Conf. Power Electron. 2010*, pp. 1–5, Jan. 2011.
- [15] P. D. Duy, V. D. Lung, N. Q. D. Trang, and N. C. Toan, "Speech Recognition on Robot Controller Implemented on FPGA," *J. Autom. Control Eng.*, vol. 1, no. 3, pp. 274–278, 2013.
- [16] M. Z. Ozdemircan, "Robot Control with Voice Command," Yildiz Technical University, 2008.

- [17] S. D. Gosavi, U. P. Khot, and S. Shah, "Speech Recognition for Robotic Control," *Int. J. Eng. Res. Appl.*, vol. 3, no. 5, pp. 408–413, 2013.
- [18] A. S. Kumar, K. M. Rao, and A. B. Krishna, "Speech Recognition System For Controlling The Robot 1," *Int. J. Eng. Res. Technol.*, vol. 1, no. 7, pp. 1–8, 2012.
- [19] J. Norberto Pires, "Robot-by-Voice: Experiments on Commanding an Industrial Robot Using The Human Voice," *Ind. Robot An Int. J.*, vol. 32, no. 6, pp. 505–511, Dec. 2005.
- [20] N.-S. Pai, H.-J. Kuang, T.-Y. Chang, Y.-C. Kuo, and C.-Y. Lai, "Implementation of a Tour Guide Robot System Using RFID Technology and Viterbi Algorithm-Based HMM for Speech Recognition," *Math. Probl. Eng.*, vol. 2014, pp. 1–7, 2014.
- [21] N. Rai and B. Rai, "An ANN Based Mobile Robot Control Through Voice Command Recognition Using Nepali Language," *Int. J. Appl. Control. Electr. Electron. Eng.*, vol. 2, no. 4, pp. 13–22, 2014.
- [22] A. Kumar, P. Singh, A. Kumar, and S. K. P. P. Guide, "Speech Recognition Based Wheelchair Using Device Switching," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 4, no. 2, pp. 391–393, 2014.
- [23] S. Kundu, S. Sahni, S. Kumar, and A. Kumar, "Voice based Control Signal Generation for Intelligent Patient Vehicle," *Int. J. Inf. Comput. Technol.*, vol. 4, no. 12, pp. 1229–1235, 2014.
- [24] O. A. Mohamad, R. T. Hameed, and N. Țăpuș, "Access Control Using Biometrics Features with Arduino Galileo," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 4, no. 8, pp. 134–141, 2014.
- [25] E. S. Gopi, *Digital Speech Processing Using Matlab*. New Delhi, India: Springer, 2014.
- [26] T. M. Diansyah and R. Aulia, "Speech Recognition Berbasis Jaringan Syaraf Tiruan Menggunakan Algoritma FFT ( Fast Fourier Transform )," *Snastikom*, pp. 195–201, 2013.
- [27] R. Jang, C.-T. Sun, and E. Mizutani, "Neuro-Fuzzy and Soft Computing." Prentice-Hall, Inc., 1997.
- [28] E. Ronando and M. Isa, "Pengenalan Ucapan Kata Sebagai Pengendali Metode Linear Predictive Coding – Neuro Fuzzy," *J. Sains dan Seni ITS*, vol. 1, no. 1, pp. 51–56, 2012.