

## Pengenalan Jenis Beban Listrik menggunakan Fast Fourier Transform dan Neural Network

Fauzdan Sauri <sup>1</sup>, Wahyu Setyo Pambudi <sup>\*2</sup>, Riza Agung Firmansyah <sup>3</sup>, Syahri Muharom <sup>4</sup>

<sup>1,2,3,4</sup> Jurusan Teknik Elektro, Institut Teknologi Adhi Tama Surabaya (ITATS), Indonesia

e-mail: <sup>1</sup>[faudzansauri@gmail.com](mailto:faudzansauri@gmail.com), <sup>\*2</sup>[wahyusp@itats.ac.id](mailto:wahyusp@itats.ac.id), <sup>3</sup>[rizaagungf@itats.ac.id](mailto:rizaagungf@itats.ac.id),

<sup>4</sup>[syahrimuharom@itats.ac.id](mailto:syahrimuharom@itats.ac.id)

### Abstrak

Kondisi saat ini pemanfaatan energi di Indonesia sebesar 42% dikonsumsi oleh perumahan lebih tinggi dibandingkan golongan industri dan komersial. Mengatasi permasalahan ini bisa melalui manajemen konservasi energi yang dikhususkan untuk pelanggan rumah tangga. Proses awal konservasi energi adalah monitoring penggunaan beban energi listrik yang sedang terpakai. Monitoring jenis penggunaan beban energi listrik yang memiliki fitur low cost adalah Non Intrusive Load Monitoring (NILM). Metode yang bisa digunakan untuk monitoring beban energi listrik dengan NILM adalah gabungan Fast Fourier Transform (FFT)-Artificial Neural Network (ANN). Tingkat keberhasilan pengenalan jenis beban listrik ini tergantung dari besarnya epoch pada saat proses training ANN. Berdasarkan pengujian nilai keberhasilan mampu mencapai nilai mendekati 100% jika menggunakan learning rate 0.7, momentum 0.9 dengan epoch 10000, berbeda jika menggunakan epoch 500 nilai keberhasilan hanya sampai 30%. Hasil proses perhitungan menggunakan confusion matrix memiliki accuracy sebesar 0.5876 atau sebesar 58.76%, sedangkan nilai F1 sebesar 0.6928 atau sebesar 69.28%.

**Kata kunci**— Konservasi energi, NILM, FFT, ANN

### Abstract

The current condition of energy utilization in Indonesia is 42% consumed by housing, higher than the industrial and commercial sectors. Overcoming this problem can be done through energy conservation management specifically for household customers. The initial process of energy conservation is monitoring the use of electrical energy loads that are being used. Monitoring the type of use of electrical energy loads that have low-cost features is Non-Intrusive Load Monitoring (NILM). The method that can be used to monitor electrical energy loads with NILM is a combination of Fast Fourier Transform (FFT)-Artificial Neural Network (ANN). The success rate of recognizing this type of electrical load depends on the size of the epoch during the ANN training process. Based on testing the success value of being able to achieve a value of 100% if using epoch 10000, it is different if using epoch 500 the success is only up to 30%. The results of the calculation process using the confusion matrix have an accuracy of 0.5876 or 58.76%, while the F1 value is 0.6928 or 69.28%.

**Keywords**— Energy Conservation, NILM, FFT, ANN.

### 1. PENDAHULUAN

Ada dua cara utama untuk membuat energi lebih berkelanjutan, yaitu, menggunakan energi terbarukan (seperti energi matahari, energi angin, energi air) atau meningkatkan efisiensi pemanfaatan energi. Fakta pemanfaatan energi global bahwa 40% dikonsumsi oleh golongan

pelanggan rumah tangga [1], sedangkan di Indonesia penggunaan energi pelanggan rumah tangga sebesar 40% [2]. Permasalahan ini memerlukan manajemen penggunaan energi efektif penting dilakukan untuk meningkatkan efisiensi energi. Pemantauan beban memiliki potensi besar dalam banyak aplikasi yang berguna, misalnya, kesadaran energi dan konservasi energi, beban terkendali evaluasi kuantitatif, perilaku manusia dan prediksi beban listrik [3].

Mengetahui jenis penggunaan energi listrik yang sedang terpakai diperlukan monitoring dan pengukuran. Monitoring dan pengukuran ini merupakan bagian penting dari proses manajemen energi listrik. Pengukuran tersebut dapat dilakukan menggunakan alat ukur sederhana, tetapi proses pengukuran dan pencatatan masih manual sehingga data yang didapat tidak bisa real time dalam durasi waktu yang lama. Keterbatasan ini memerlukan sebuah alat monitoring yang dapat melakukan pengukuran pada satu titik sensing yang dapat mengetahui penggunaan jenis penggunaan beban listrik yang sedang terpakai [4][5].

Didalam sistem kelistrikan dalam rumah tangga, untuk mengidentifikasi status dan mengetahui penggunaan konsumsi energi peralatan listrik biasanya terdapat sensor tegangan dan sensor arus yang dipasang pada setiap peralatan listrik [6]. Hal ini disebut juga sistem pemantauan beban secara intrusive atau disebut juga *Intrusive Load Monitoring* (ILM), sedangkan sistem pemantauan beban dengan menggunakan satu instrumen sensor disebut *Non Intrusive Load Monitoring* (NILM). Sistem NILM lebih baik dibandingkan dengan sistem pemantauan menggunakan ILM dikarenakan tanpa memasang instrument apapun kedalam peralatan listrik rumah tangga yang berarti dapat mengurangi biaya pemasangan sensor yang sangat signifikan [7]. Pengukuran dan pemantauan beban listrik secara *non-intrusive* pada konsumsi energi listrik, yang diterapkan pada *smart power meter* NILM adalah teknologi terbaru untuk disagregasi energi. NILM dapat membedakan jenis peralatan listrik dan perilaku atau karakteristik pada peralatan listrik dengan hanya memasang satu titik sensor pada sumber listrik utama. Keunggulan dari NILM ini dapat menghemat biaya pembelian sensor dan perangkat keras lainnya. Alur didalam proses NILM terdiri dari akuisisi data, ekstraksi fitur dan pembelajaran data inferensi [8].

Fitur yang dapat diekstrak dari pembacaan NILM bisa berdasarkan frekwensi harmonik jenis beban listrik, dimana setiap jenis beban memiliki frekwensi harmonik yang berbeda [9][10]. Membaca frekwensi harmonik bisa melalui proses Fast Fourier Transform (FFT), dimana FFT ini adalah metode efisien untuk menghitung koefisien dari fourier diskrit menjadi suatu finite sekuen dari data yang kompleks [11]. Spektrum FFT perlu digabungkan dengan Artificial Neural Network (ANN) dengan pembelajaran *Back Propagation* (BP) untuk membaca karakter dari harmonik jenis beban listrik [10]. Pemilihan ANN untuk membaca harmonik beban listrik ini karena lebih baik lebih baik dibandingkan dengan model matematik untuk identifikasi pola sinyal [12]. Selain itu gabungan FFT dengan ANN juga mampu melakukan diagnosis generator maupun [13], serta estimasi tekanan darah [14].

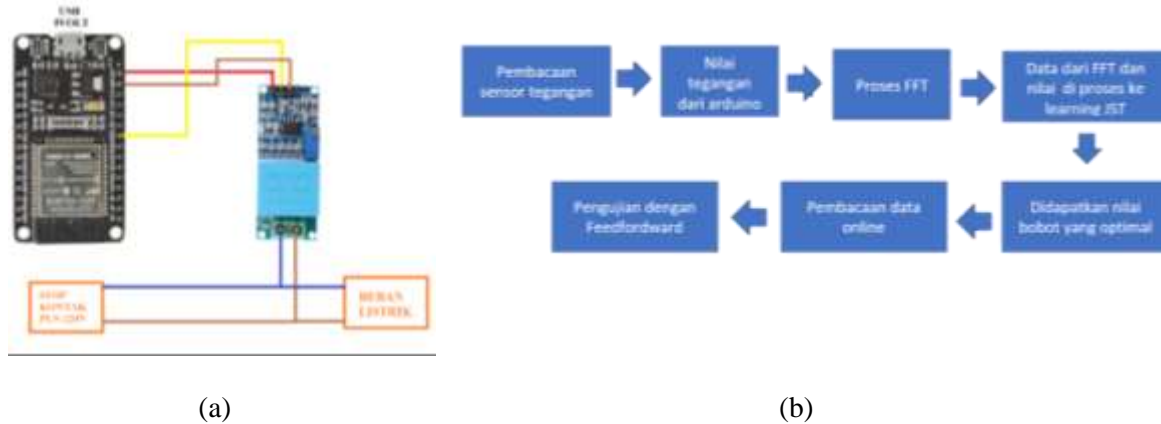
Berdasarkan latar belakang tersebut, penelitian ini membahas tentang pengenalan jenis beban listrik dengan pemasangan *single* sensor atau NILM menggunakan FFT-ANN dengan metode pembelajaran BP. Pembacaan spektrum tegangan menggunakan *embedded system* ESP32 sedangkan untuk ANN diproses didalam komputer dengan menggunakan program Visual Studio.

## 2. METODE PENELITIAN

### 2.1 Perancangan Sistem

Pada tahap Perancangan sistem adalah melakukan persiapan perancangan *hardware* dan *software*. Persiapan yang dilakukan dalam pembuatan *hardware* adalah melengkapi alat-alat yang digunakan seperti sensor tegangan dan arduino untuk proses pengambilan dataset. Perancangan *software* merupakan membuat interface pengolahan data FFT dan ANN yang dibuat pada visual studio. Rangkaian pengukuran tegangan seperti pada gambar 1(a), jenis sensor yang digunakan ZMPT101b, sensor tegangan ini memiliki tingkat kesalahan kurang dari

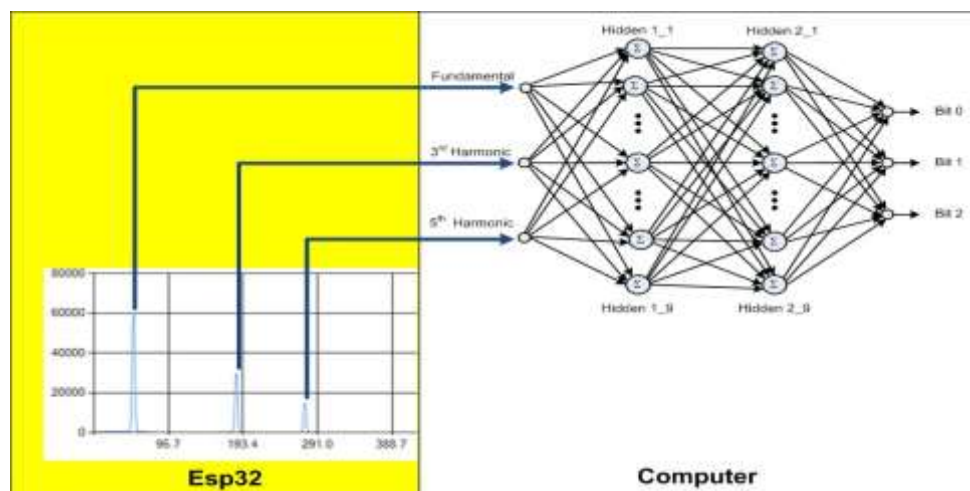
3% untuk tegangan pengukuran 220 Vac [6]. Pada penelitian ini beban peralatan rumah tangga yang akan digunakan adalah Televisi, kipas angin, pemanas dan blender yang menghasilkan nilai gelombang sinus dan terdapat harmonik dari beban. ESP32 pada gambar 1(a) berfungsi sebagai pengolah data untuk membaca nilai ADC dari tegangan ZMPT101b. Nilai ADC tersebut kemudian diproses dengan FFT untuk menampilkan spektrum frekwensi, sedangkan ANN ditanamkan pada *software* Visual Studio yang untuk pengenalan beban listrik yang sedang digunakan. Pada *software* Visual Studio berisi informasi spektrum frekwensi dari proses FFT yang telah diproses oleh ESP32 kemudian dikirimkan melalui komunikasi serial. Proses perancangan *software* yang telah dilakukan seperti pada gambar 1(b).



Gambar 1.(a). Perancangan *Hardware* (b). Blok Diagram *Software*

Pengambilan data tegangan ini dilakukan dengan untuk 4 jenis beban listrik yang mewakili beban resistif dan induktif, beban listrik tersebut adalah televisi, kipas angin, blender, dan pemanas. Hasil data ketiga beban menghasilkan nilai tegangan yang selanjutnya proses dengan FFT. FFT ini berfungsi untuk ekstraksi sinyal sinus tegangan 220 Vac dari setiap jenis beban listrik. Proses FFT ini menghasilkan nilai total *harmonic distortion* (THD), dimana nilai THD ini di proses dengan ANN.

Proses *training* pada penelitian ini menggunakan metode BP, dimana metode ini dapat menghasilkan nilai *training* yang optimal dengan nilai error pada saat learning sebesar 0.27131. Hasil dari *training* ini menghasilkan nilai bobot yang optimal dengan menggunakan learning rate 0.7 dan momentum 0.9, yang digunakan pada pengujian. Berdasarkan hasil pengujian menggunakan *feed forward* ANN dengan menggunakan nilai bobot yang diperoleh dari proses *learning* dengan input nilai harmonik frekwensi dari proses FFT, mampu menghasilkan prediksi nilai yang telah di inisialisasi pada masing masing beban listrik.



Gambar 2. Struktur Jaringan ANN

Pengambilan data tegangan menggunakan mikrokontroler ESP32 yang telah di program dengan arduino IDE. Teknik pengambilan data tegangan ini dengan cara membaca nilai ADC dari sensor ZMPT101b. Sinyal ADC yang diterima dari sensor ZMPT101b diproses FFT untuk menghasilkan spektrum frekwensi untuk membaca harmonik. Proses FFT adalah proses untuk mengubah sinyal yang ber domain waktu menjadi sinyal yang ber domain frekuensi [15]. Pada proses FFT ini didapatkan nilai amplitudo sehingga dapat memunculkan nilai THD. THD adalah rasio antara nilai rms dari seluruh komponen harmonik (sensor tegangan) dan nilai rms terdiri dari fundamental, harmonik ke-3 dan harmonik-5.

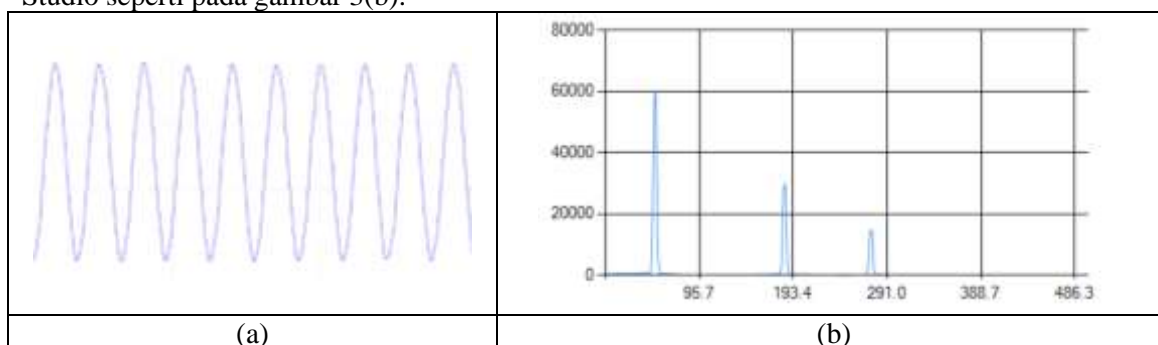
Tahapan pelatihan ini menerapkan proses *training* ANN dengan metode BP. Proses BP ini berfungsi sebagai proses pelatihan data untuk menghasilkan bobot yang optimal yang digunakan dalam tahap pengujian. Struktur jaringan ANN terdiri dari input layer, hidden layer dan output layer. Input menggunakan 3 node, hidden layer menggunakan 9 node, dan output keluaran menggunakan 3 node, seperti pada gambar 2.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Pengujian Tegangan

Proses kalibrasi awal pada sensor tegangan ZMPT101B dilakukan dengan cara memberikan tegangan AC ke sensor ZMPT101b. Bentuk sinyal yang dihasilkan berbentuk sinyal sinus seperti pada gambar 3(a), dimana sinyal ini dijadikan input proses FFT.

Sinyal sinus pada gambar 3(a) merupakan hasil pembacaan sinyal ADC dari sensor ZMPT101b. Berdasarkan sinyal gelombang sinus inilah yang kemudian proses dengan FFT. Output dari FFT adalah keluaran sinyal fundamental, harmonik ke-3, dan harmonik ke -5. Data ini dikirim kedalam software Visual Stuido melalui serial komunikasi yang selanjutn dilakukan *feed forward* ANN. Hasil keluaran proses FFT yang telah diterima program Visual Studio seperti pada gambar 3(b).



Gambar 3. (a). Sinyal sensor ZMPT101b (b). Grafik Hasil FFT

#### 3.2 Pengujian FFT Beban Listrik

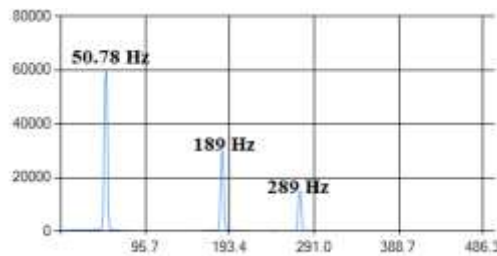
Pengujian FFT dengan beban listrik ini merupakan pengujian pembacaan hasil pemrosesan FFT dari masing-masing beban listrik yang digunakan. Hasil dari FFT ini menghasilkan nilai fundamental, harmonik ke-3, dan harmonik k-5. Beban listrik yang digunakan pada penelitian ini mewakili beban resistif dan induktif diantaranya adalah kipas angin, pemanas, blender dan televisi. Pengujian FFT ini dilakukan pada masing-masing beban tunggal dan dilakukan dengan beberapa kombinasi beban listrik seperti kipas angin dan televisi, pemanas dan blender. Berikut percobaan pengujian FFT dengan beban blender akan ditampilkan pada gambar 4(a).

Hasil dari pengujian ini menghasilkan spektrum sinyal FFT yang dihasilkan dari ESP32. Output dari FFT ini adalah sinyal magnitude dan frekwensi, output ini kemudian dikirimkan melalui serial ke komputer dan hasilnya tampilkan pada chart seperti pada gambar 4(b). Berdasarkan gambar 4(b) dibawah ini menunjukkan bahwa sinyal harmonik yang muncul dengan beban pemanas menunjukkan pada harmonik ke-3 muncul berada pada frekwensi 189 Hz, dan

harmonik ke-5 289 Hz. Pada penelitian ini dengan beban blender akan dilakukan beberapa percobaan pengambilan data FFT, hasil dari percobaan akan ditampilkan pada tabel 1 dibawah ini.



(a)



(b)

Gambar 4. (a). Pengujian FFT Beban Blender (b). Hasil FFT Beban Blender

Tabel 1. Percobaan FFT dengan beban blender

| No | Fundamental | Harmonik ke -3 | Harmonik ke -5 | Perangkat                   |
|----|-------------|----------------|----------------|-----------------------------|
| 1  | 50.7812     | 189.5469       | 289.9219       | Blender speed 2 percobaan 1 |
| 2  | 50.7812     | 169.9219       | 269.5313       | Blender speed 1 percobaan 1 |
| 3  | 50.7812     | 195.3125       | 287.0625       | Blender speed 2 percobaan 2 |
| 4  | 50.7812     | 171.875        | 271.4844       | Blender speed 1 percobaan 2 |

Selain menggunakan beban blender, pengujian FFT ini akan menggunakan beban kipas angin, pemanas dan televisi. Pengujian FFT ini menggunakan beberapa kombinasi seperti kipas angin speed1, speed2, speed3. Hasil pengujian dengan beban kipas angin, pemanas dan televisi akan ditampilkan pada tabel 2.

Tabel 2 Pengujian FFT dengan beban tunggal

| No | Fundamental (Hz) | Harmonik ke-3 (Hz) | Harmonik ke-5 (Hz) | Perangkat                                |
|----|------------------|--------------------|--------------------|--|
| 1  | 50.7812          | 150.3906           | 250                | Kipas speed 1 percobaan 1                |
| 2  | 50.7812          | 150.3906           | 250                | Kipas speed 2 percobaan 1                |
| 3  | 50.7812          | 154.2969           | 255.8594           | Kipas speed 3 percobaan 1                |
| 4  | 50.7812          | 150.3906           | 250                | Kipas speed 1 percobaan 2                |
| 5  | 50.7812          | 154.2969           | 250                | Kipas speed 2 percobaan 2                |
| 6  | 50.7812          | 154.2969           | 285.1563           | Kipas speed 3 percobaan 2                |
| 7  | 50.7812          | 150.3906           | 250                | Kipas speed 1 percobaan 3                |
| 8  | 50.7812          | 150.3906           | 250                | Kipas speed 2 percobaan 3                |
| 9  | 50.7812          | 150.3906           | 255.2594           | Kipas speed 3 percobaan 3                |
| 10 | 50.7812          | 150.3906           | 0                  | Teko pemanas memasak percobaan 1         |
| 11 | 50.7812          | 120.1406           | 0                  | Teko pemanas setelah 2 menit percobaan 1 |
| 12 | 50.7812          | 150.3906           | 0                  | Teko pemanas memasak percobaan 0         |
| 13 | 50.7812          | 119.1406           | 0                  | Teko pemanas setelah 2 menit percobaan 0 |
| 14 | 50.7812          | 80.191             | 125                | Televisi mode antena percobaan 1         |
| 15 | 50.7812          | 95.741             | 138.521            | Televisi mode vga percobaan 1            |
| 16 | 50.7812          | 80.191             | 125.851            | Televisi mode antena percobaan 2         |
| 17 | 50.7812          | 95.741             | 139.521            | Televisi mode vga percobaan 2            |

Pada tabel 2 terlihat pada beban kipas saat speed 1, speed 2 dan speed 3 menghasilkan sinyal harmonik yang berbeda. Percobaan dengan kipas angin dilakukan percobaan sebanyak 3 kali dengan speed yang berbeda-beda, seperti pada speed 1 percobaan 1 menghasilkan sinyal fundamental sebesar 50.7812 Hz, harmonik ke-3 150.3906 Hz, harmonik ke-5 250 Hz. Hasil percobaan speed 1 dan 2 menghasilkan sinyal harmonik yang sama, dan speed 3 percobaan 1 menghasilkan fundamental 50.7812 Hz, harmonik ke-3 154.2969 Hz, harmonik ke-5 255.8594 Hz. Selain menggunakan beban tunggal, pada penelitian ini juga menggunakan dengan kombinasi beban televisi dan kipas, blender dan pemanas. Hasil percobaan FFT dengan menggunakan beban kombinasi akan ditampilkan pada tabel 3.

Tabel 3 PengujianFFT dengan beban kombinasi

| No | Fundamental (Hz) | Harmonik ke-3 (Hz) | Harmonik ke-5 (Hz) | Perangkat                        |
|----|------------------|--------------------|--------------------|----------------------------------|
| 1  | 50.7812          | 183.5469           | 294.9219           | Blender speed 2 percobaan 1      |
| 2  | 50.7812          | 169.9219           | 269.5313           | Blender speed 1 percobaan 1      |
| 3  | 50.7812          | 185.3125           | 289.0625           | Blender speed 2 percobaan 2      |
| 4  | 50.7812          | 174.875            | 271.4844           | Blender speed 1 percobaan 2      |
| 5  | 50.7812          | 82.191             | 125                | Televisi mode antena percobaan 1 |
| 6  | 50.7812          | 93.741             | 138.521            | Televisi mode vga percobaan 1    |
| 7  | 50.7812          | 85.191             | 125.851            | Televisi mode antena percobaan 2 |
| 8  | 50.7812          | 91.441             | 139.521            | Televisi mode vga percobaan 2    |
| 9  | 50.7812          | 130.9341           | 251.941            | Blender dan pemanas percobaan 1  |
| 10 | 50.7812          | 124.9341           | 255.941            | Blender dan pemanas percobaan 2  |
| 11 | 50.7812          | 122.9341           | 252.941            | Blender dan pemanas percobaan 3  |

Hasil dari percobaan dengan beban kombinasi seperti pada tabel 3 diatas pada beban blender dan pemanas mendapatkan hasil yang berbeda,dikarenakan kombinasi beban saat blender speed 1 dan pemanas saat memasak, blender speed 2 saat pemanas tidak memasak.

Tabel 4. Input ANN Sebelum Normalisasi

| No | Sebelum Normalisasi |                    |                    | Setelah Normalisasi |                    |                    | Perangkat                                |
|----|---------------------|--------------------|--------------------|---------------------|--------------------|--------------------|--|
|    | Fundamental (Hz)    | Harmonik ke-3 (Hz) | Harmonik ke-5 (Hz) | Fundamental (Hz)    | Harmonik ke-3 (Hz) | Harmonik ke-5 (Hz) |  |
| 1  | 50.781              | 150.391            | 250.000            | 0.169               | 0.501              | 0.833              | Kipas speed 1 percobaan 1                |
| 2  | 50.781              | 150.391            | 250.000            | 0.169               | 0.501              | 0.833              | Kipas speed 2 percobaan 1                |
| 3  | 50.781              | 154.297            | 255.859            | 0.169               | 0.514              | 0.853              | Kipas speed 3 percobaan 1                |
| 4  | 50.781              | 150.391            | 250.000            | 0.169               | 0.501              | 0.833              | Kipas speed 1 percobaan 2                |
| 5  | 50.781              | 154.297            | 250.000            | 0.169               | 0.514              | 0.833              | Kipas speed 2 percobaan 2                |
| 6  | 50.781              | 154.297            | 285.156            | 0.169               | 0.514              | 0.951              | Kipas speed 3 percobaan 2                |
| 7  | 50.781              | 150.391            | 250.000            | 0.169               | 0.501              | 0.833              | Kipas speed 1 percobaan 3                |
| 8  | 50.781              | 150.391            | 250.000            | 0.169               | 0.501              | 0.833              | Kipas speed 2 percobaan 3                |
| 9  | 50.781              | 150.391            | 255.259            | 0.169               | 0.501              | 0.851              | Kipas speed 3 percobaan 3                |
| 10 | 50.781              | 150.391            | 0.000              | 0.169               | 0.501              | 0.000              | Teko pemanas memasak percobaan 1         |
| 11 | 50.781              | 120.141            | 0.000              | 0.169               | 0.400              | 0.000              | Teko pemanas setelah 2 menit percobaan 1 |
| 12 | 50.781              | 150.391            | 0.000              | 0.169               | 0.501              | 0.000              | Teko pemanas memasak percobaan 0         |
| 13 | 50.781              | 119.141            | 0.000              | 0.169               | 0.397              | 0.000              | Teko pemanas setelah 2 menit percobaan 0 |
| 14 | 50.781              | 185.547            | 294.922            | 0.169               | 0.618              | 0.983              | Blender speed 2 percobaan 1              |
| 15 | 50.781              | 169.922            | 269.531            | 0.169               | 0.566              | 0.898              | Blender speed 1 percobaan 1              |
| 16 | 50.781              | 195.313            | 289.063            | 0.169               | 0.651              | 0.964              | Blender speed 2 percobaan 2              |
| 17 | 50.781              | 171.875            | 271.484            | 0.169               | 0.573              | 0.905              | Blender speed 1 percobaan 2              |
| 18 | 50.781              | 80.191             | 125.000            | 0.169               | 0.267              | 0.417              | Televisi mode antena percobaan 1         |
| 19 | 50.781              | 95.741             | 138.521            | 0.169               | 0.319              | 0.462              | Televisi mode vga percobaan 1            |
| 20 | 50.781              | 80.191             | 125.851            | 0.169               | 0.267              | 0.420              | Televisi mode antena percobaan 2         |
| 21 | 50.781              | 95.741             | 139.521            | 0.169               | 0.319              | 0.465              | Televisi mode vga percobaan 2            |
| 22 | 50.781              | 130.934            | 253.941            | 0.169               | 0.436              | 0.846              | Blender dan pemanas percobaan 1          |
| 23 | 50.781              | 126.934            | 256.941            | 0.169               | 0.423              | 0.856              | Blender dan pemanas percobaan 2          |
| 24 | 50.781              | 121.934            | 254.941            | 0.169               | 0.406              | 0.850              | Blender dan pemanas percobaan 3          |
| 25 | 50.781              | 144.561            | 214.800            | 0.169               | 0.482              | 0.716              | Televisi dan kipas angin percobaan 1     |
| 26 | 50.781              | 146.821            | 213.214            | 0.169               | 0.489              | 0.711              | Televisi dan kipas angin percobaan 2     |
| 27 | 50.781              | 151.521            | 217.521            | 0.169               | 0.505              | 0.725              | Televisi dan kipas angin percobaan 3     |

Didalam pengujian ANN ini, dilakukan dengan 3 tahap. Tahap pertama yaitu memasukan data frekwensi kedalam input ANN, tahap kedua adalah proses *training* dengan masukan data FFT dari semua beban, tahap ketiga yaitu proses pengujian dengan *feed forward* dengan masukan data dari arduino. Input dari ANN adalah hasil dari proses FFT dari sensor tegangan. Proses FFT ini menghasilkan nilai Amplitudo dan frekwensi yang dikirim melalui serial komunikasi ke komputer kemudian diproses dengan ANN yang terdapat pada Program Visual Studio. Data serial yang diterima oleh Visual Studio dijadikan input dalam variable array. Sebelum di jadikan input, data dari serial di ambil 3 data, yaitu fundamental, harmonik ke-3 dan harmonik ke-5 dari masing masing beban.

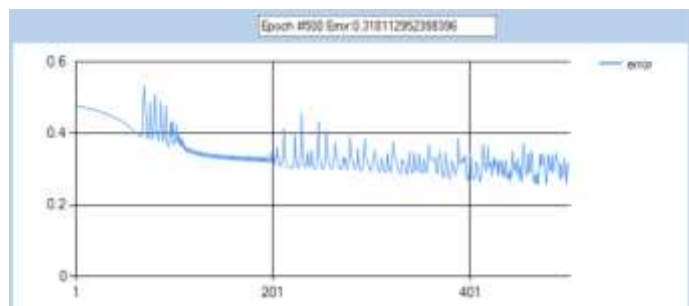
Setelah mengambil 3 data, maka dilakukan normalisasi untuk menjadikan nilai 0 hingga 1, karena fungsi aktivasi yang digunakan dari ANN antara 0 hingga 1. Nilai dataset yang belum di normalisasi dan setelah dinormalisasi seperti pada tabel 4. Normalisasi disini setiap nilai dari fundamental, harmonik ke-3 dan harmonik ke-5 masing-masing dibagi dengan 300, dimana nilai ini adalah nilai asumsi nilai maksimal dari frekwensi yang diukur.

Tabel 5. Nilai Kombinasi Biner Jenis Beban

| NO | Ideal-1 | Ideal-2 | Ideal-3 | Perangkat                                |
|----|---------|---------|---------|--|
| 1  | 0       | 0       | 1       | Kipas speed 1 percobaan 1                |
| 2  | 0       | 0       | 1       | Kipas speed 2 percobaan 1                |
| 3  | 0       | 0       | 1       | Kipas speed 3 percobaan 1                |
| 4  | 0       | 0       | 1       | Kipas speed 1 percobaan 2                |
| 5  | 0       | 0       | 1       | Kipas speed 2 percobaan 2                |
| 6  | 0       | 0       | 1       | Kipas speed 3 percobaan 2                |
| 7  | 0       | 0       | 1       | Kipas speed 1 percobaan 3                |
| 8  | 0       | 0       | 1       | Kipas speed 2 percobaan 3                |
| 9  | 0       | 0       | 1       | Kipas speed 3 percobaan 3                |
| 10 | 1       | 0       | 1       | teko pemanas memasak percobaan 1         |
| 11 | 1       | 0       | 1       | teko pemanas setelah 2 menit percobaan 1 |
| 12 | 1       | 0       | 1       | teko pemanas memasak percobaan 0         |
| 13 | 1       | 0       | 1       | teko pemanas setelah 2 menit percobaan 0 |
| 14 | 1       | 0       | 0       | Blender speed 2 percobaan 1              |
| 15 | 1       | 0       | 0       | Blender speed 1 percobaan 1              |
| 16 | 1       | 0       | 0       | Blender speed 2 percobaan 2              |
| 17 | 1       | 0       | 0       | Blender speed 1 percobaan 2              |
| 18 | 0       | 1       | 0       | Televisi mode antena percobaan 1         |
| 19 | 0       | 1       | 0       | Televisi mode vga percobaan 1            |
| 20 | 0       | 1       | 0       | Televisi mode antena percobaan 2         |
| 21 | 0       | 1       | 0       | Televisi mode vga percobaan 2            |
| 22 | 0       | 1       | 1       | Blender dan pemanas percobaan 1          |
| 23 | 0       | 1       | 1       | Blender dan pemanas percobaan 2          |
| 24 | 0       | 1       | 1       | Blender dan pemanas percobaan 3          |
| 25 | 1       | 1       | 0       | Televisi dan kipas angin percobaan 1     |
| 26 | 1       | 1       | 0       | Televisi dan kipas angin percobaan 2     |
| 27 | 1       | 1       | 0       | Televisi dan kipas angin percobaan 3     |

Data yang telah di normalisasi bertujuan agar input ANN yang menggunakan proses aktivasi Sigmoid bisa bernilai 0 hingga 1. Proses selanjutnya adalah memberikan input nilai kombinasi biner (Ideal-1, Ideal-2, Ideal-3) untuk pengenalan jenis beban. Nilai kombinasi biner adalah nilai yang ditentukan oleh peneliti untuk membedakan jenis beban yang telah ditargetkan. Input nilai kombinasi biner (Ideal-1, Ideal-2, Ideal-3) untuk pengenalan jenis beban di tampilkan pada tabel 5.

Pada *training* data ANN dilakukan sebanyak 20x proses *training* dengan mengubah nilai epoch untuk mendapatkan nilai *error* yang paling rendah. Setelah dataset diinputkan maka dilakukan proses training data menggunakan proses *backward* dan *feed forward*. Proses training data ini menggunakan berbagai percobaan nilai epoch yang berbeda beda untuk menghasilkan nilai erorr seminimal mungkin. Epoch yang akan digunakan pada penelitian ini akan menggunakan 3x percobaan dengan epoch yang berbeda-beda yaitu dengan epoch 500, 1500 dan 5000. Hasil percobaan pada epoch 500 nilai *Mean Square Error* (MSE) ditampilkan pada gambar 5 dibawah ini.



Gambar 5. Nilai MSE untuk Training 500 Epoch

Pada gambar 5 diatas merupakan hasil percobaan training data dengan menggunakan epoch 10000. *Training* data menggunakan epoch 500 didapatkan hasil nilai *error* sebesar 0.3181, dimana epoch 500 terlihat error yang cukup tinggi. Hal ini dikarenakan epoch terlalu kecil maka proses pembelajaran kurang maksimal.

Tabel 6. Epoch training data

| No | Epoch | Error   | Learning Rate |
|----|-------|---------|---------------|
| 1  | 500   | 0.33162 | 0.7           |
| 2  | 500   | 0.33141 | 0.7           |
| 3  | 500   | 0.3741  | 0.1           |
| 4  | 500   | 0.3791  | 0.1           |
| 5  | 500   | 0.4152  | 0.9           |
| 6  | 500   | 0.4194  | 0.9           |
| 7  | 500   | 0.3841  | 0.5           |
| 8  | 500   | 0.3892  | 0.5           |
| 9  | 1000  | 0.4164  | 0.7           |
| 10 | 1000  | 0.4211  | 0.7           |
| 11 | 1000  | 0.39934 | 0.1           |
| 12 | 1000  | 0.39841 | 0.1           |
| 13 | 1000  | 0.3416  | 0.9           |
| 14 | 1000  | 0.3492  | 0.9           |
| 15 | 1000  | 0.34819 | 0.5           |
| 16 | 1000  | 0.34841 | 0.5           |
| 17 | 2500  | 0.2911  | 0.7           |
| 18 | 2500  | 0.2985  | 0.7           |
| 19 | 2500  | 0.4039  | 0.1           |
| 20 | 2500  | 0.40942 | 0.1           |
| 21 | 2500  | 0.29836 | 0.9           |
| 22 | 2500  | 0.29124 | 0.9           |
| 23 | 2500  | 0.2999  | 0.5           |
| 24 | 2500  | 0.2931  | 0.5           |
| 25 | 5000  | 0.28546 | 0.7           |
| 26 | 5000  | 0.28724 | 0.7           |
| 27 | 5000  | 0.3415  | 0.1           |
| 28 | 5000  | 0.3387  | 0.1           |
| 29 | 5000  | 0.2941  | 0.9           |
| 30 | 5000  | 0.2851  | 0.9           |
| 31 | 5000  | 0.2985  | 0.5           |
| 32 | 5000  | 0.3014  | 0.5           |
| 33 | 10000 | 0.27393 | 0.7           |
| 34 | 10000 | 0.27131 | 0.7           |
| 35 | 10000 | 0.29421 | 0.1           |
| 36 | 10000 | 0.29753 | 0.1           |
| 37 | 10000 | 0.28523 | 0.9           |
| 38 | 10000 | 0.28142 | 0.9           |
| 39 | 10000 | 0.3131  | 0.5           |
| 40 | 10000 | 0.2942  | 0.5           |

Percobaan selanjutnya menggunakan epoch 5000 untuk *training*. Percobaan *training* data akan dilakukan percobaan sebanyak 25 percobaan proses *training* dengan menggunakan epoch 500, 1000 dan 5000. Hasil percobaan ditampilkan pada tabel 6. Pada tabel 6 ini merupakan hasil dari pengujian ANN dengan beberapa macam epoch yang digunakan.



Berdasarkan hasil percobaan pada tabel 6, pada epoch 10000, learning rate 0.7 dan momentum 0.9 memiliki hasil yang terbaik dan memiliki *error* sebesar 0.27131.

### 3.3 Pengujian

Proses pengujian ini menggunakan *feed forward*, proses *feed forward* ini dengan cara mengambil nilai bobot sementara dan membandingkan dengan data arduino. Proses pengujian ini menggunakan nilai epoch training terbaik dengan nilai *error* yang paling kecil yaitu sebesar 10000 epoch dan dilakukan percobaan dengan berbagai macam beban listrik.

Tabel 7. Hasil *Feed forward* Epoch 10000

| No | Output NN |          |          | Ideal |   |   | epoch | Perangkat           | Status Pengujian |
|----|-----------|----------|----------|-------|---|---|-------|---------------------|------------------|
|    | 1         | 2        | 3        | 1     | 2 | 3 |       |                     |                  |
| 1  | 0.8712    | 0.002632 | 0.9201   | 1     | 0 | 1 | 10000 | Pemanas             | Sesuai           |
| 2  | 0.9121    | 0.00027  | 0.9268   | 1     | 0 | 1 | 10000 | Pemanas             | Sesuai           |
| 3  | 0.8903    | 0.000206 | 0.8584   | 1     | 0 | 1 | 10000 | Pemanas             | Sesuai           |
| 4  | 0.9711    | 0.000169 | 0.9818   | 1     | 0 | 1 | 10000 | Pemanas             | Sesuai           |
| 5  | 0.9713    | 0.000184 | 0.9875   | 1     | 0 | 1 | 10000 | Pemanas             | Sesuai           |
| 6  | 0.0024    | 0.0027   | 0.8692   | 0     | 0 | 1 | 10000 | Kipas               | Sesuai           |
| 7  | 0.0003    | 0.0002   | 0.8901   | 0     | 0 | 1 | 10000 | Kipas               | Sesuai           |
| 8  | 0.0001    | 0.0003   | 0.8932   | 0     | 0 | 1 | 10000 | Kipas               | Sesuai           |
| 9  | 0.0003    | 0.0002   | 0.9702   | 0     | 0 | 1 | 10000 | Kipas               | Sesuai           |
| 10 | 0.0003    | 0.0002   | 0.9904   | 0     | 0 | 1 | 10000 | Kipas               | Sesuai           |
| 11 | 0.9248    | 0.001969 | 0.001021 | 1     | 0 | 0 | 10000 | Blender             | Sesuai           |
| 12 | 0.9017    | 0.000156 | 0.000274 | 1     | 0 | 0 | 10000 | Blender             | Sesuai           |
| 13 | 0.8855    | 0.0003   | 0.000277 | 1     | 0 | 0 | 10000 | Blender             | Sesuai           |
| 14 | 0.9727    | 0.000104 | 0.000149 | 1     | 0 | 0 | 10000 | Blender             | Sesuai           |
| 15 | 0.9771    | 0.000155 | 0.00018  | 1     | 0 | 0 | 10000 | Blender             | Sesuai           |
| 16 | 0.001706  | 0.8991   | 0.002815 | 0     | 1 | 0 | 10000 | TV                  | Sesuai           |
| 17 | 0.000171  | 0.866    | 0.000216 | 0     | 1 | 0 | 10000 | TV                  | Sesuai           |
| 18 | 0.000295  | 0.9186   | 0.000202 | 0     | 1 | 0 | 10000 | TV                  | Sesuai           |
| 19 | 0.000259  | 0.9924   | 0.000166 | 0     | 1 | 0 | 10000 | TV                  | Sesuai           |
| 20 | 0.000196  | 0.986    | 0.000279 | 0     | 1 | 0 | 10000 | TV                  | Sesuai           |
| 21 | 0.002207  | 0.9275   | 0.9089   | 0     | 1 | 1 | 10000 | Blender dan Pemanas | Sesuai           |
| 22 | 0.00029   | 0.8867   | 0.8795   | 0     | 1 | 1 | 10000 | Blender dan Pemanas | Sesuai           |
| 23 | 0.000118  | 0.8904   | 0.9154   | 0     | 1 | 1 | 10000 | Blender dan Pemanas | Sesuai           |
| 24 | 0.000213  | 0.9883   | 0.9872   | 0     | 1 | 1 | 10000 | Blender dan Pemanas | Sesuai           |
| 25 | 0.000216  | 0.9803   | 0.9773   | 0     | 1 | 1 | 10000 | Blender dan Pemanas | Sesuai           |
| 26 | 0.9087    | 0.866    | 0.001444 | 1     | 1 | 0 | 10000 | TV dan Kipas        | Sesuai           |
| 27 | 0.8921    | 0.8646   | 0.000188 | 1     | 1 | 0 | 10000 | TV dan Kipas        | Sesuai           |
| 28 | 0.8974    | 0.8591   | 0.000278 | 1     | 1 | 0 | 10000 | TV dan Kipas        | Sesuai           |
| 29 | 0.9978    | 0.9822   | 0.000168 | 1     | 1 | 0 | 10000 | TV dan Kipas        | Sesuai           |
| 30 | 0.9788    | 0.9947   | 0.00019  | 1     | 1 | 0 | 10000 | TV dan Kipas        | Sesuai           |

Proses pengujian ini langsung dilakukan dengan menguji data real dari pembacaan FFT yang dikirim melalui serial komunikasi. Data real merupakan data uji dari setiap masing-masing beban listrik yang akan digunakan. Visual Studio akan menutup serial komunikasi jika telah menerima 256 data serial dari arduino. Data serial yang dikirimkan adalah frekwensi fundamental, frekwensi harmonik ke-3 serta frekwensi harmonik ke-5. Setelah pengambilan data FFT dan setelah di training dengan datashet input yang telah ditetapkan kemudian dilakukan pengujian identifikasi menggunakan *feed forward* ANN. Hasil *feed forward* ini diatur jika hasil output lebih dari 0.5 akan di anggap 1, jika hasil *feed forward* kurang dari 0.5 akan di anggap 0. Pengaturan ini berfungsi untuk pembacaan jenis beban yang hasilnya dibandingkan dengan nilai kombinasi biner (Ideal-1, Ideal-2, Ideal-3) yang sudah di tetapkan seperti pada tabel 5. Percobaan yang dilakukan menggunakan epoch 10000 dengan masing-masing beban yang dilakukan percobaan sebanyak 5x. Hasil percobaan dengan epoch 10000 ditampilkan pada tabel 7. Pada tabel 7 diatas merupakan hasil dari pengujian *feed forward* dengan epoch 10000, dimana nilai output hasil *feed forward* ANN sudah mampu mendekati nilai ideal yang sudah di tentukan seperti pada tabel 5. Percobaan dengan epoch 10000 ini tidak terdapat kesalahan pembacaan jenis beban yang di uji. Pada percobaan pengujian kedua menggunakan epoch 5000, hasil dari pengujian dengan epoch 5000 ditampilkan pada tabel 8.

Tabel 8. Hasil *Feed forward* Epoch 5000

| No | Output NN      |                |               | Ideal    |          |          | epoch       | Perangkat           | Status Pengujian    |
|----|----------------|----------------|---------------|----------|----------|----------|-------------|---------------------|---------------------|
|    | 1              | 2              | 3             | 1        | 2        | 3        |             |                     |                     |
| 1  | 0.8525         | 0.02289        | 0.8727        | 1        | 0        | 1        | 5000        | Pemanas             | Sesuai              |
| 2  | 0.8974         | 0.01406        | 0.8502        | 1        | 0        | 1        | 5000        | Pemanas             | Sesuai              |
| 3  | 0.8767         | 0.01559        | 0.8759        | 1        | 0        | 1        | 5000        | Pemanas             | Sesuai              |
| 4  | 0.911          | 0.0189         | 0.8569        | 1        | 0        | 1        | 5000        | Pemanas             | Sesuai              |
| 5  | 0.8602         | 0.02233        | 0.8902        | 1        | 0        | 1        | 5000        | Pemanas             | Sesuai              |
| 6  | 0.0213         | 0.0178         | 0.9171        | 0        | 0        | 1        | 5000        | Kipas               | Sesuai              |
| 7  | 0.0233         | 0.0238         | 0.8594        | 0        | 0        | 1        | 5000        | Kipas               | Sesuai              |
| 8  | 0.0166         | 0.0153         | 0.9231        | 0        | 0        | 1        | 5000        | Kipas               | Sesuai              |
| 9  | 0.0286         | 0.0244         | 0.8572        | 0        | 0        | 1        | 5000        | Kipas               | Sesuai              |
| 10 | 0.0206         | 0.0189         | 0.8567        | 0        | 0        | 1        | 5000        | Kipas               | Sesuai              |
| 11 | <b>0.02431</b> | <b>0.8641</b>  | <b>0.8641</b> | <b>1</b> | <b>0</b> | <b>0</b> | <b>5000</b> | <b>Blender</b>      | <b>Tidak Sesuai</b> |
| 12 | 0.9088         | 0.01306        | 0.02338       | 1        | 0        | 0        | 5000        | Blender             | Sesuai              |
| 13 | 0.8645         | 0.02336        | 0.0171        | 1        | 0        | 0        | 5000        | Blender             | Sesuai              |
| 14 | 0.9004         | 0.0146         | 0.02338       | 1        | 0        | 0        | 5000        | Blender             | Sesuai              |
| 15 | <b>0.01842</b> | <b>0.01162</b> | <b>0.8727</b> | <b>1</b> | <b>0</b> | <b>0</b> | <b>5000</b> | <b>Blender</b>      | <b>Tidak Sesuai</b> |
| 16 | 0.02884        | 0.8718         | 0.01779       | 0        | 1        | 0        | 5000        | TV                  | Sesuai              |
| 17 | 0.019          | 0.9013         | 0.02284       | 0        | 1        | 0        | 5000        | TV                  | Sesuai              |
| 18 | 0.01101        | 0.8852         | 0.01842       | 0        | 1        | 0        | 5000        | TV                  | Sesuai              |
| 19 | 0.01708        | 0.8506         | 0.02431       | 0        | 1        | 0        | 5000        | TV                  | Sesuai              |
| 20 | 0.02266        | 0.9291         | 0.02819       | 0        | 1        | 0        | 5000        | TV                  | Sesuai              |
| 21 | 0.02916        | 0.8693         | 0.8529        | 0        | 1        | 1        | 5000        | Blender dan Pemanas | Sesuai              |
| 22 | 0.02038        | 0.8888         | 0.8641        | 0        | 1        | 1        | 5000        | Blender dan Pemanas | Sesuai              |
| 23 | 0.02414        | 0.8907         | 0.8838        | 0        | 1        | 1        | 5000        | Blender dan Pemanas | Sesuai              |
| 24 | 0.02752        | 0.8912         | 0.8887        | 0        | 1        | 1        | 5000        | Blender dan Pemanas | Sesuai              |
| 25 | 0.02808        | 0.9106         | 0.8929        | 0        | 1        | 1        | 5000        | Blender dan Pemanas | Sesuai              |
| 26 | 0.9179         | 0.8855         | 0.0171        | 1        | 1        | 0        | 5000        | TV dan Kipas        | Sesuai              |
| 27 | 0.9223         | 0.914          | 0.02165       | 1        | 1        | 0        | 5000        | TV dan Kipas        | Sesuai              |
| 28 | 0.8561         | 0.8934         | 0.02338       | 1        | 1        | 0        | 5000        | TV dan Kipas        | Sesuai              |
| 29 | 0.9014         | 0.8907         | 0.02288       | 1        | 1        | 0        | 5000        | TV dan Kipas        | Sesuai              |
| 30 | 0.885          | 0.8893         | 0.014         | 1        | 1        | 0        | 5000        | TV dan Kipas        | Sesuai              |

Tabel 9. Kesalahan pembacaan

| No | Epoch | Kesalahan Pembacaan | Percobaan Pengujian | Keberhasilan |
|----|-------|---------------------|---------------------|--------------|
| 1  | 10000 | 0                   | 10                  | 100 %        |
| 2  | 10000 | 0                   | 10                  | 100 %        |
| 3  | 10000 | 0                   | 10                  | 100 %        |
| 4  | 5000  | 2                   | 10                  | 80 %         |
| 5  | 5000  | 1                   | 10                  | 90 %         |
| 6  | 5000  | 1                   | 10                  | 90 %         |
| 7  | 2500  | 3                   | 10                  | 70 %         |
| 8  | 2500  | 3                   | 10                  | 70 %         |
| 9  | 2500  | 2                   | 10                  | 80 %         |
| 10 | 1000  | 5                   | 10                  | 50 %         |
| 11 | 1000  | 5                   | 10                  | 50 %         |
| 12 | 1000  | 4                   | 10                  | 60 %         |
| 13 | 500   | 7                   | 10                  | 30 %         |
| 14 | 500   | 7                   | 10                  | 30 %         |
| 15 | 500   | 9                   | 10                  | 10 %         |

Pada percobaan dengan epoch 5000 terjadi kesalahan pembacaan sebanyak dua kali. Terlihat pada nomor 11 dan 15 terjadi kesalahan pembacaan, dimana didapatkan hasil yaitu 0.02431, 0.8641, 0.8641 atau terbaca kondisi binernya 0,1,1 yang seharusnya untuk blender adalah 1,0,0. Nomor 15 juga terjadi kesalahan pembacaan, dimana data terbaca 0.01842, 0.01162, 0.8727 atau terbaca kondisi binernya 0,0,1 yang berarti beban kipas angin dimana seharusnya 1,0,0 untuk beban blender. Selain melakukan percobaan dengan epoch 10000 dan 5000 pada penelitian ini akan juga dilakukan dengan epoch 2500, 1000, 500, 100. Percobaan dengan berbagai macam epoch ini mencari tingkat akurasi dengan berbagai macam epoch. Hasil percobaan dengan menggunakan berbagai macam epoch di tampilkan pada tabel 9.

Pada tabel 9 merupakan kesalahan pembacaan dengan berbagai macam epoch. Proses pengujian ini dilakukan dengan mencoba semua jenis beban listrik yang digunakan dan dilakukan sebanyak 3 kali percobaan. Pada tabel 9 percobaan kesalahan pembacaan ini hasil terbaik yaitu dengan menggunakan epoch 10000 dengan *error* pembacaan 0 atau dengan kata lain memiliki tingkat keberhasilan 100% dan hasil terendah yaitu menggunakan epoch 500 yang

menghasilkan kesalahan pembacaan sebanyak 9 kali yang menghasilkan tingkat keberhasilan pembacaan sebesar 10%. Perhitungan performa pembacaan jenis beban listrik pada penelitian ini menggunakan *confusion matrix*, dimana hasil performa pembacaan jenis seperti pada tabel 10.

Tabel 10. Performa Pembacaan Jenis Beban

| No        | Jenis pengujian               | Jumlah data | TP | FP | TN | FN | Precision (PR) | Recall (PE) | Accuracy (Acc) | F1-Score (F1) |
|-----------|-------------------------------|-------------|----|----|----|----|----------------|-------------|----------------|---------------|
| 1         | Pengujian Kipas               | 60          | 36 | 28 | 2  | 0  | 0.5625         | 1           | 0.5758         | 0.7200        |
| 2         | Pengujian TV                  | 40          | 34 | 7  | 2  | 0  | 0.8293         | 1           | 0.8372         | 0.9067        |
| 3         | Pengujian Blender             | 40          | 16 | 11 | 3  | 1  | 0.5926         | 0.9412      | 0.6129         | 0.7273        |
| 4         | Pengujian Pemanas             | 40          | 21 | 14 | 4  | 1  | 0.6000         | 0.9545      | 0.6250         | 0.7368        |
| 5         | Pengujian Blender dan Pemanas | 40          | 11 | 19 | 7  | 3  | 0.3667         | 0.7857      | 0.4500         | 0.5000        |
| 6         | Pengujian Kipas dan TV        | 40          | 15 | 21 | 2  | 2  | 0.4167         | 0.8824      | 0.4250         | 0.5660        |
| Rata-rata |                               |             |    |    |    |    | 0.5613         | 0.9273      | 0.5876         | 0.6928        |

Berdasarkan hasil pada tabel 10 diatas nilai rata-rata accuracy sebesar 0.5876 atau sebesar 58.76%, sedangkan nilai F1 sebesar 0.6928 atau sebesar 69.28%. Nilai *accuracy* ini masih dibawah 60% karena proses pengujian dilakukan mulai dari epoch 500 yang memiliki tingkat keberhasilan maksimal hanya 30% (tabel 9) dari total 10 kali pengujian. Nilai *accuracy* bisa lebih tinggi jika yang diujikan epoch 10000 karena tingkat keberhasilannya bisa sampai 100%.

#### 4. KESIMPULAN

Berdasarkan dari pengujian dan analisa data yang telah dilakukan pada sistem pembacaan jenis beban listrik dengan metode FFT-ANN dapat diambil kesimpulan bahwa, dengan membaca nilai frekwensi fundamental, harmonik ke-3 dan harmonik ke-5 mampu membedakan jenis beban listrik. Proses FFT ditanamkan pada embedded system ESP32 yang dapat dikembangkan kearah *internet of thing* (IoT). ANN dengan metode pembelajaran BP dapat dibuat dengan aplikasi Visual Studio. Pada pengenalan jenis beban secara NILM ini tingkat kesalahan pembacaan jenis beban dipengaruhi oleh besarnya epoch pada saat proses training ANN, semakin besar nilai epoch nilai error semakin kecil. Nilai keberhasilan bisa sampai 100% jika dilakukan proses *training* sampai 10000 epoch berbeda dengan pada saat epoch 500 nilai keberhasilan paling tinggi hanya 30%. Secara keseluruhan pengujian dengan *training* epoch dari 500 – 10000 pada sistem pengenalan jenis beban listrik ini memiliki *accuracy* sebesar 0.5876 atau sebesar 58.76%, sedangkan nilai F1 sebesar 0.6928 atau sebesar 69.28%.

#### 5. SARAN

Sistem yang dikembangkan pada penelitian ini masih terdapat kekurangan, yang paling terlihat adalah tingkat *accuracy*. Perbaikan nilai *accuracy* salah satunya bisa dengan cara menaikkan nilai epoch pada saat *training* dataset atau dengan melakukan optimasi proses *training* ANN. Selain itu agar meningkatkan nilai *accuracy* bisa menggunakan ANN yang lebih kompleks seperti *Convolutional Neural Network* (CNN).

#### DAFTAR PUSTAKA

- [1] IESR, *Indonesia Clean Energy Outlook: Tracking Progress and Review of Clean Energy Development in Indonesia*. 2019. [Online]. Available: [www.iesr.or.id](http://www.iesr.or.id)
- [2] I. F. U. Muzayanah, H. H. Lean, D. Hartono, K. D. Indraswari, and R. Partama, "Population density and energy consumption: A study in Indonesian provinces," *Heliyon*, vol. 8, no. 9, p. e10634, 2022, doi: 10.1016/j.heliyon.2022.e10634.
- [3] J. Zhang, X. Chen, W. W. Y. Ng, C. S. Lai, and L. L. Lai, "New Appliance Detection for

- Nonintrusive Load Monitoring,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 8, pp. 4819–4829, 2019, doi: 10.1109/tii.2019.2916213.
- [4] A. Riantiarto, D. Suryadi, and Saifurrahman, “Rancang Bangun Alat Monitoring Arus Pada Beban Listrik Rumah Tangga Menggunakan WEB Berbasis Arduino UNO R3,” *J. Tek. Elektro Univ. Tanjungpura*, vol. 2, no. 1, pp. 1–9, 2019, [Online]. Available: <https://jurnal.untan.ac.id/index.php/jteuntan/article/download/35505/75676582854>
  - [5] I. Nirmalasari, A. E. Putra, and B. N. Prastowo, “Purwarupa Alat Ukur Daya Listrik Berbasis Netduino Plus,” *IJEIS (Indonesian J. Electron. Instrum. Syst.)*, vol. 5, no. 1, p. 21, 2015, doi: 10.22146/ijeis.7150.
  - [6] M. E. Lutfi and A. Rouf, “Purwarupa kWh Meter Prabayar Berbasis Sensor Network 1,” *Ijeis*, vol. 4, no. 2, pp. 147–156, 2014.
  - [7] S. Biansoongnern and B. Plangklang, “Nonintrusive load monitoring (NILM) using an Artificial Neural Network in embedded system with low sampling rate,” *2016 13th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol. ECTI-CON 2016*, 2016, doi: 10.1109/ECTICon.2016.7561398.
  - [8] B. Buddhahai, W. Wongseeree, and P. Rakkwamsuk, “An Energy Prediction Approach for a Nonintrusive Load Monitoring in Home Appliances,” *IEEE Trans. Consum. Electron.*, vol. 66, no. 1, pp. 96–105, 2020, doi: 10.1109/TCE.2019.2956638.
  - [9] T. Le, “Household Appliance Classification Using Lower Odd-Numbered Harmonics and the Bagging Decision Tree,” *IEEE Access*, vol. 8, pp. 55937–55952, 2020, doi: 10.1109/ACCESS.2020.2981969.
  - [10] D. S. Kumar, K. L. Low, A. Sharma, and W. L. Woo, “Non-Intrusive Load Monitoring using Feed Forward Neural Network,” *2019 IEEE PES Innov. Smart Grid Technol. Asia, ISGT 2019*, pp. 4065–4069, 2019, doi: 10.1109/ISGT-Asia.2019.8880801.
  - [11] P. Jacko and O. Kravets, “Spectral Analysis by STM32 Microcontroller of the Mixed Signal,” *Proc. Int. Conf. Mod. Electr. Energy Syst. MEES 2019*, pp. 342–345, 2019, doi: 10.1109/MEES.2019.8896545.
  - [12] M. Nikzad, K. Movagharnejad, and F. Talebnia, “Comparative Study between Neural Network Model and Mathematical Models for Prediction of Glucose Concentration during Enzymatic Hydrolysis,” *Int. J. Comput. Appl.*, vol. 56, no. 1, pp. 43–48, 2012, doi: 10.5120/8859-2818.
  - [13] J. Da Wu and J. M. Kuo, “An automotive generator fault diagnosis system using discrete wavelet transform and artificial neural network,” *Expert Syst. Appl.*, vol. 36, no. 6, pp. 9776–9783, 2009, doi: 10.1016/j.eswa.2009.02.027.
  - [14] X. Xing and M. Sun, “Optical blood pressure estimation with photoplethysmography and FFT-based neural networks,” *Biomed. Opt. Express*, vol. 7, no. 8, p. 3007, 2016, doi: 10.1364/boe.7.003007.
  - [15] M. Simic, “Nonintrusive identification of residential appliances using harmonic analysis,” *Turkish J. Electr. Eng. Comput. Sci.*, vol. 26, no. 2, pp. 780–791, 2018, doi: 10.3906/elk-1705-262.