

Mengidentifikasi Sinyal Suara Manusia Menggunakan *Metode Fast Fourier Transform* (Fft) Berbasis Matlab

Ade Riyani¹, Asyhar Nurrochman², Eko Sanjaya³, Putri Rizqiyah⁴, Apri Junaidi⁵

Institut Teknologi Telkom Purwokerto
Jl. D.I Panjaitan 128, Purwokerto, Indonesia

¹ 15102002@ittelkom.ac.id

² 15102011@ittelkom.ac.id

³ 15102015@ittelkom.ac.id

⁴ 15102031@ittelkom.ac.id

⁵ aprijunaidi@ittelkom.ac.id

Accepted on April 11, 2019

Abstract

Suara merupakan suatu hal yang unik dan memiliki range frekuensi tertentu dan intensitas suara yang bisa dan tidak bisa didengar oleh manusia. Satuan untuk mengukur intensitas suara tersebut adalah desibel (dB). kurangnya suatu teknologi yang dapat menerapkan pengenalan sinyal suara. Dengan adanya sistem pengenalan suara dapat mempermudah manusia dalam beraktivitas. Jadi dibutuhkan suatu sistem yang dapat mengenali karakteristik suara manusia yang nantinya dapat diterapkan dalam suatu aplikasi. Hasil yang didapatkan sistem pengenalan suara juga dapat mengenali suara dengan baik dengan suara yang dimainkan pada nilai *frame blocking* 16, 32, 64, 128 dan 256. Tingkat pengenalan suara ini turun jika nilai *frame blocking* yang digunakan di bawah 128.

Keywords: Suara, sinyal, Fast Fourier Transform

I. PENDAHULUAN

Perkembangan teknologi yang mulai pesat yang dimana dapat membantu manusia dalam memenuhi setiap kebutuhannya. Salah satu teknologi yang mulai berkembang adalah pengolahan sinyal suara. Pengolahan sinyal suara sudah banyak diaplikasikan dalam berbagai bidang seperti pengolahan suara, music, biomedical, navigasi, telekomunikasi, video dan gambar[1].

Suara merupakan suatu hal yang unik dan memiliki range frekuensi tertentu dan intensitas suara yang bisa dan tidak bisa didengar oleh manusia. Satuan untuk mengukur intensitas suara tersebut adalah desibel (dB) diambil dari nama penemunya yaitu Alexander Graham Bell yang dikenal sebagai penemu telepon, sedangkan satuan dari frekuensi suara adalah Hertz, diambil dari nama seorang Fisikawan, Heinrich Rudolf Hertz untuk menghargai jasa atas kontribusinya dalam bidang elektromagnetik[1].

Kurangnya suatu teknologi yang dapat menerapkan pengenalan sinyal suara. Dengan adanya sistem pengenalan suara dapat mempermudah manusia dalam beraktivitas. Jadi dibutuhkan suatu sistem yang dapat mengenali karakteristik suara manusia yang nantinya dapat diterapkan dalam suatu aplikasi. Seperti keamanan *smartphone* menggunakan suara, menjalankan perintah-perintah di *smartphone*, dan sebagainya. Dalam penelitian pengolahan sinyal suara ini menggunakan algoritma fast Fourier Transform(FFT) .

Algoritma Fast Fourier Transform (FFT) adalah suatu algoritma untuk menghitung Discrete Fourier Transform (DFT) yang digunakan untuk dua menghitung spektrum frekuensi sinyal dan FFT merupakan prosedur penghitungan DFT yang efisien sehingga akan mempercepat proses penghitungan DFT yang secara substansial dapat lebih menghemat waktu dari pada metoda yang konvensional. Algoritma fast fourier transform membagi frekuensi per priodenya, karena itu algoritma ini dapat berkerja dengan baik sehingga menghasilkan akurasi dengan cepat dan efisien [2].

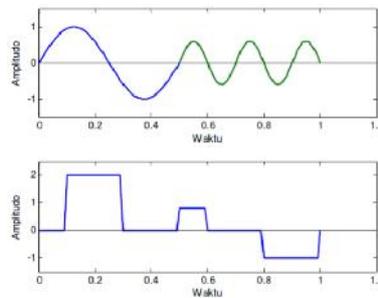
II. LANDASAN TEORI

2.1 Sinyal Suara

Audio diartikan sebagai suara atau reproduksi suara. Gelombang suara adalah gelombang yang dihasilkan dari sebuah benda yang bergetar[3].tempat dimana suara tak dapat merambat adalah ruangan hampa udara. Gelombang suara memiliki lembah dan bukit, satu buah lembah dan bukit menghasilkan satu periode. Siklus ini berlangsung berulang-ulang, yang membawa pada konsep frekuensi. Frekuensi adalah jumlah dari siklus yang terjadi dalam satu detik. Sinyal suara adalah sinyal yang dapat diterima oleh telinga manusia yaitu antara 20 Hz hingga 20KHz[1].

2.2 Jenis Sinyal

Sinyal dibedakan menjadi dua yaitu sinyal analog dan sinyal digital. Suatu sinyal dikatakan analog apabila amplitudo dari sinyal tersebut terus menerus ada dalam rentang waktu tertentu (kontinyu) fan memiliki variasi nilai amplitudo tak terbatas. Misalnya yang berasal dari suara tergolong sebagai sinyal analog. Sebaliknya data atau sinyal dikatakan digital apabila amplitudo dari sinyal tersebut tidak kontinyu dan memiliki variasi nilai amplitudo yang terbatas (diskrit)[4].



Gambar 2.1 sinyal analog dan sinyal digital

2.3 Fast Fourier Transform (FFT)

Fast Fourier Transform (FFT) adalah metode yang sangat efisien untuk menghitung koefisien dari Fourier diskrit ke suatu finite sekuen dari data yang kompleks. Karena substansi waktu yang tersimpan lebih dari pada metoda konvensional, fast fourier transform merupakan aplikasi temuan yang penting didalam sejumlah bidang yang berbeda seperti analisis spectrum, speech and optical signal processing, design filter digital[5]. Algoritma FFT berdasarkan atas prinsip pokok dekomposisi perhitungan discrete fourier transform dari suatu sekuen sepanjang N ke dalam transformasi diskrit Fourier secara berturut-turut lebih kecil[4]. Cara prinsip ini diterapkan memimpin ke arah suatu variasi dari algortima yang berbeda, di mana semuanya memperbandingkan peningkatan kecepatan perhitungan. Fast Fourier Transform, adalah suatu algoritma

untuk menghitung transformasi fourier diskrit dengan cepat dan efisien. Karena banyak sinyal-sinyal dalam sistem komunikasi yang bersifat kontinyu, sehingga untuk kasus sinyal kontinyu kita gunakan transformasi fourier. Transformasi Fourier didefinisikan oleh rumus:

$$S(f) = \int_{-\infty}^{\infty} s(t) e^{-j2\pi ft} dt \quad (1)$$

Dimana $s(f)$ adalah sinyal dalam domain frekuensi (frequency domain), $s(t)$ adalah sinyal dalam domain waktu (time domain), dan $e^{-j2\pi ft}$ adalah konstanta dari nilai sebuah sinyal, f adalah frekuensi dan t adalah waktu. FFT (Fast Fourier Transform) merupakan salah satu metode untuk transformasi sinyal suara dalam domain waktu menjadi sinyal dalam domain frekuensi, artinya proses perekaman suara disimpan dalam bentuk digital berupa gelombang spectrum suara yang berbasis frekuensi sehingga lebih mudah dalam menganalisa spectrum frekuensi suara yang telah direkam[4].

III. METODE PENELITIAN

Berikut adalah metodologi penelitian yang dilakukan pada penelitian ini:



Gambar 3.1 *flowchart* proses pengenalan suara

3.1 Perekaman Suara

Tahapan yang pertama yaitu perekaman suara, suara yang direkam adalah suara manusia yang direkam secara langsung tanpa, suara yang direkam akan langsung tanpa perantara *device* yang lainnya, suara yang sudah direkam akan disimpan dalam sebuah *file* dengan format *.wav*.

3.2 *Preprocessing*

Tahapan ini merupakan tahapan melakukan normalisasi suara yang sudah didapatkan pada tahapan sebelumnya, *preprocessing* terdiri dari

1. Normalisasi

Suara yang sudah disimpan pada tahapan sebelumnya dalam format *.wav* akan dinormalisasi dengan tujuan supaya amplitudo yang dihasilkan dapat maksimal.

2. Pemotongan sinyal.

Tahapan ini bertujuan untuk menghilangkan *noise* pada suara yang sudah terekam, sehingga kualitas suara dapat membaik.

3. *Frame blocking*

Tahapan ini bertujuan untuk memilih suara dari semua data yang ada sehingga data suara yang dipilih dapat mewakili seluruh data suara yang sudah terekam

4. *Windowing*

Tahapan ini berfungsi untuk menghilangkan diskontinuitas yang disebabkan karena tahapan *frame blocking*

3.3 *Fast Fourier Transform*

Tahapan ini merupakan tahapan mengekstrak ciri suatu data suara. Evaluasi FFT yang digunakan berdasarkan *frame blocking* yang didapatkan, setelah menentukan besar *frame blocking* nilai tersebut digunakan sebagai basis koefisiennya.

3.4 Penentuan nilai maksimum

Tahapan ini bertujuan untuk mengetahui nilai maksimum yang dihasilkan yang akan dijadikan sebagai penentuan hasil keluaran.

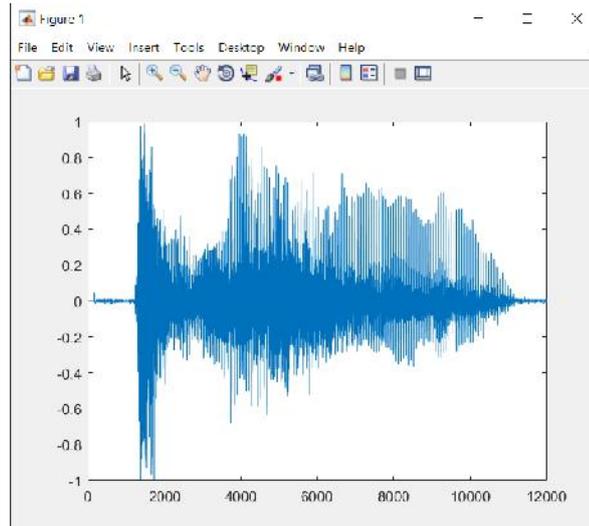
3.5 Penentuan teks suara

Penentuan hasil pengenalan suara ditentukan berdasarkan nilai-nilai maksimum yang dihasilkan dari tahapan sebelumnya.

IV. HASIL DAN PEMBAHASAN

4.1 Rekam

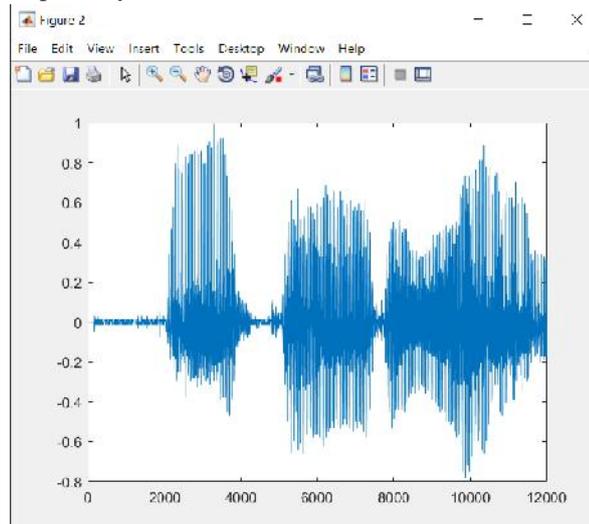
Semua sampel suara yang didapat akan dilakukan proses sampling terlebih dahulu sebelum masuk ke tahap berikutnya. Hasil dari proses rekaman ini nantinya akan berformat **.wav*. program perekaman suara memakai frekuensi sampling sebesar 4800 Hz dan durasi rekaman 1.5 detik. Untuk menampilkan gambar rekaman dilakukan perhitungan (frekuensi x durasi) sehingga menampilkan sinyal sebesar 7200 Hz. Selanjutnya suara disimpan dengan format **.wav*.



Gambar 4.1 Tampilan Plot Sinyal Perekaman

4.2 Normalisasi

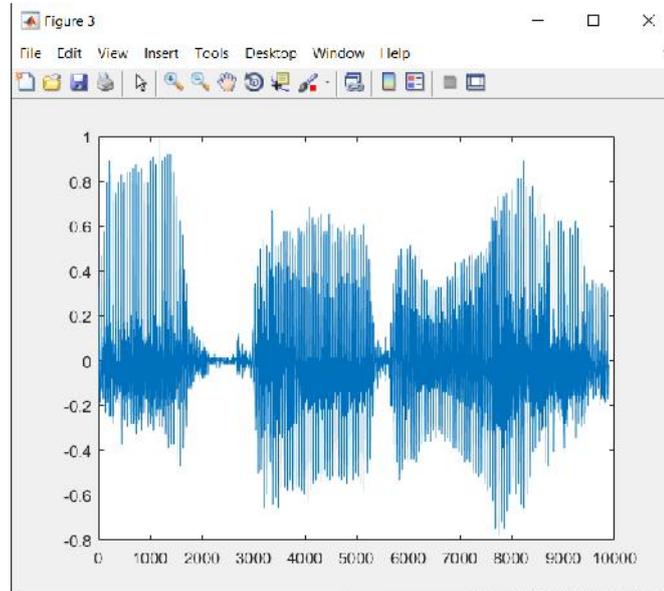
Setelah suara direkam dan disimpan, selanjutnya hasil rekaman tersebut dilakukan proses normalisasi. Pada proses ini, sinyal suara harus memiliki nilai sinyal maksimum. Normalisasi dilakukan dengan membagi data masukan (data sinyal suara) dengan nilai maksimum data tersebut. Berikut merupakan hasil plot sinyal hasil normalisasi.



Gambar 4.2 Tampilan Plot Sinyal Normalisasi

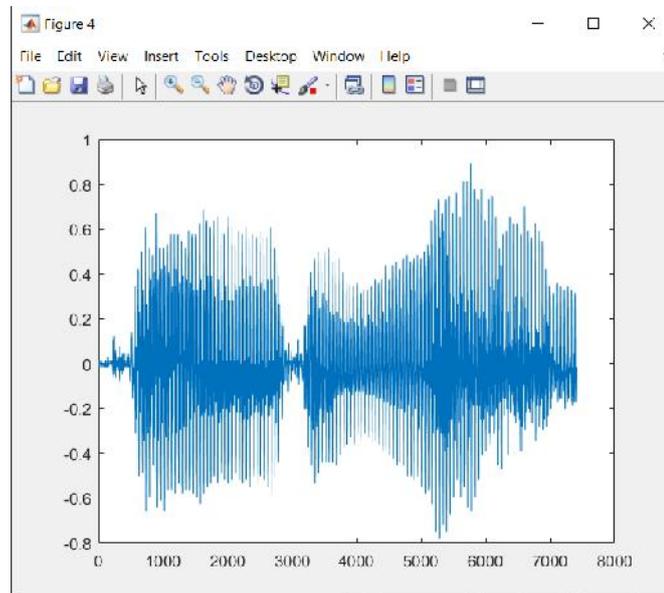
4.3 Pemotongan sinyal

Pada proses pemotongan sinyal dilakukan sebanyak dua kali dari sinyal yang telah dinormalisasi. Untuk pemotongan pertama dilakukan pada bagian *silence* atau bagian sinyal yang bukan bagian dari suara manusia. Pemotongan kedua dilakukan untuk memotong sinyal pada bagian transisi. Data yang tingginya lebih besar dari 0.3 dan lebih kecil dari -0.3 diinisialisasikan sebagai sinyal *silence*, sehingga sinyal tersebut akan dihilangkan. Berikut merupakan tampilan sinyal setelah sinyal *silence* dibuang.



Gambar 4.3 Tampilan Hasil Pemotongan Sinyal *Silence*

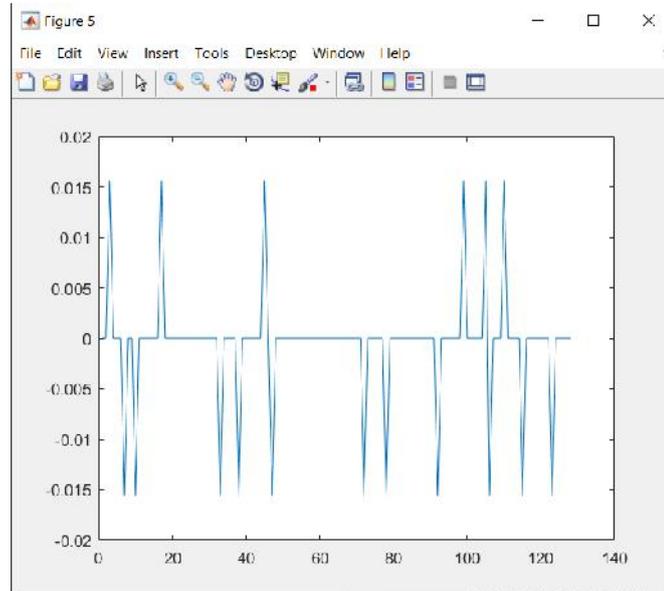
Selanjutnya adalah pemotongan bagian transisi dengan menghilangkan bagian sinyal yang terdapat pada bagian awal yang diinisialisasikan sebagai *bts* pada program. Sinyal tersebut dihilangkan karena hanya ingin mengambil sinyal suara manusia saja. Berikut merupakan hasil plot sinyal bagian transisi.



Gambar 4.4 Tampilan Hasil Pemotongan Sinyal Transisi

4.4 Frame Blocking

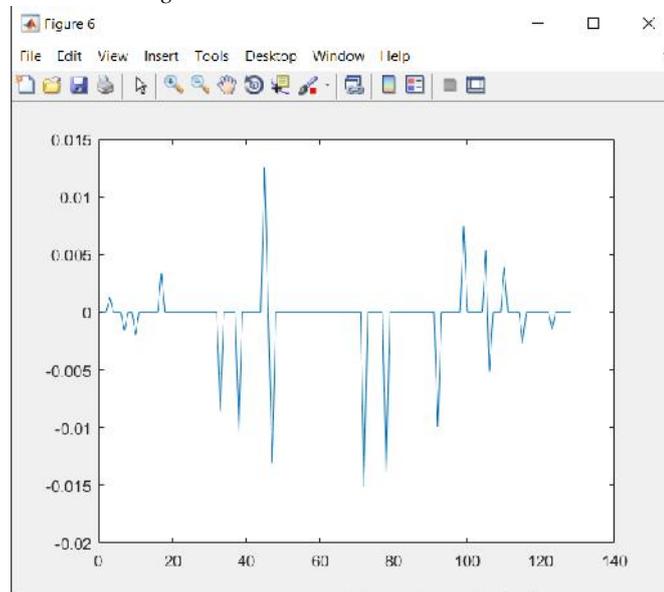
Proses *Frame Blocking* bertujuan untuk mengambil bagian dari data sesuai dengan Panjang nilai *frame blocking*. Nilai *frame blocking* yang digunakan adalah 16, 32, 64, 128 dan 256. Data ini mewakili seluruh data suara yang terekam. Berikut merupakan hasil dari proses *frame blocking*.



Gambar 4.5 Tampilan Hasil *Frame Blocking*

4.5 Windowing

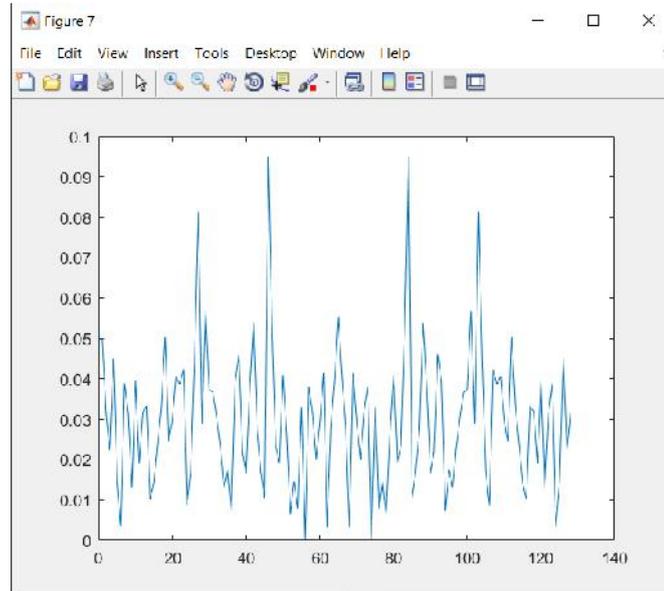
Windowing dilakukan untuk menghilangkan efek diskontinuitas yang diakibatkan dari proses *frame blocking* ketika sinyal ditransformasikan ke domain frekuensi. Sampel yang telah dibagi menjadi beberapa *frame* perlu dijadikan suara kontinu. Proses *windowing* ini menggunakan *hamming window*. Selanjutnya hasil perhitungan dari proses *frame blocking* akan dikalikan dengan *hamming*. Berikut adalah hasil dari *hamming window*.



Gambar 4.6 Tampilan Hasil *Hamming Window*

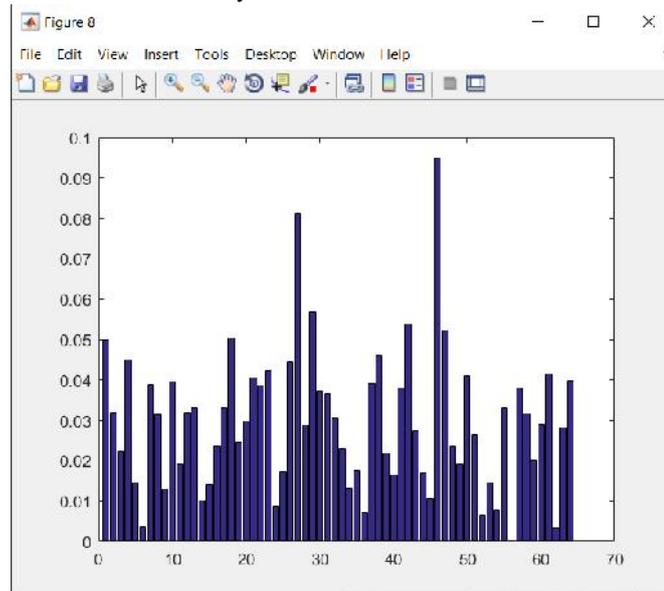
4.6 Ekstraksi Ciri Fast Fourier Transform (FTT)

Ekstraksi ciri FTT ini merupakan proses untuk mendapatkan besaran pada bagian sinyal yang terekam untuk menetapkan pola pembelajaran atau pola uji. Perhitungan ini bertujuan untuk mencari nilai *absolute* yang akan dianalisis untuk mengetahui suara. Proses ini dimulai dengan mencari nilai omputasi FFT yang diinisialisasikan sebagai *fft* dan dilanjutkan dengan mencari nilai *absolute*.



Gambar 4.7 Tampilan Hasil Ekstraksi Ciri FFT

Hasil komputasi tersebut kemudian dipotong setengah dari ukuran sinyal yang telah ditentukan. Selanjutnya adalah merubah dimensi sinyal.



Gambar 4.8 Tampilan Hasil Pemotongan Setengah Sinyal FFT

4.7 Pencarian Nilai Maksimum

Selanjutnya mencari nilai maksimum setelah mendapatkan sinyal FFT. Nilai maksimum tersebut dijadikan penentuan keluaran dari sistem pengenalan dengan menggunakan *look up table*. Berikut hasil dari pencarian nilai maksimum.

4.8 Look Up Table

Dalam sistem pengenalan suara ini digunakan 5 buah *look up table* untuk setiap *frame blocking*. Nilai yang digunakan pada proses ini diperoleh dari hasil pengambilan data pengujian dengan mencari nilai maksimum pada inyal FFT.

4.9 Penentuan Teks Suara

Setelah mengetahui nilai maksimum, maka diperoleh hasil keluaran pengenalan suara berupa teks suara. Penentuan keluaran digunakan *look up table*. Setelah didapatkan nilai maksimumnya maka nilai tersebut akan dicocokkan dengan nilai pada *look up table* untuk *frame blocking* yang dipilih.

4.10 Analisa Performa

Pengujian ini menggunakan seluruh nilai dari *frame blocking* yaitu 16, 32, 64, 128, 256. Data diambil sebanyak 10 kali untuk masing-masing suara. Tujuan dari pengujian ini adalah untuk mengetahui apakah program dapat bekerja dengan baik atau tidak.

V. KESIMPULAN

berdasarkan pengujian yang sudah dilakukan, sistem ini menghasilkan pengenalan suara dengan metode FFT sebagai berikut:

1. Sistem ini telah bekerja sesuai dengan perancangan. Sistem pengenalan suara juga dapat mengenali suara dengan baik dengan suara yang dimainkan pada nilai *frame blocking* 16, 32, 64, 128 dan 256.
2. Tingkat pengenalan suara ini turun jika nilai *frame blocking* yang digunakan dibawah 128. Penurunan ini diakibatkan oleh nilai indeks maksimum setiap suara berada pada range yang sama.

REFERENCES

- [1] R. A. L. Sibarani, *Identifikasi Sinyal Suara Menggunakan Metode Fast Fourier Transform (FFT) Berbasis Matlab*. 2012.
- [2] I. I. Tritoasmoro, L. V. Yovita, F. T. Elektro, and U. Telkom, "Implementasi tuner gitar berbasis fast fourier transform 1110700671," pp. 0–6, 2011.
- [3] T. Akhir and Y. R. Setyawan, "Pengenalan ucapan angka secara real time menggunakan ekstraksi ciri fft dan fungsi similaritas kosinus," *Progr. Stud. Teknik Elektro, Fak. Sains dan Teknol. Univ. Sanata Dharma*, 2014.
- [4] R. Y. Sipasulta, A. S. M. L. St, and S. R. U. A. Sompie, "Simulasi Sistem Pengacak Sinyal Dengan Metode FFT (Fast Fourier Transform)," *E-journal Tek. Elektro dan Komput.*, pp. 1–9, 2014.
- [5] H. Sujadi, I. Sopiandi, and A. Mutaqin, "Sistem Pengolahan Suara Menggunakan Algoritma FFT (Fast Fourier Transform)," *Pros. SINTAK*, pp. 101–107, 2017.