



TELNECT



Journal homepage: <http://ejournal.upi.edu/index.php/TELNECT/>

Enhancing Audio Fidelity through Advanced Anti Aliasing Sampling Techniques

ABSTRACTS	ARTICLE INFO
<p>The importance of audio quality in various applications such as music, movies, and video games suggests the need for advanced techniques to improve sound quality. One of the major problems is aliasing, which can result in distortion and degradation of audio quality. This paper introduces advanced anti-aliasing and audio processing techniques to improve audio quality. Through experiments with MATLAB, we successfully clean up the noise interference in the audio and eliminate the annoying aliasing effect. This research is expected to make important contributions to the development of anti-aliasing and audio processing techniques to improve sound quality in various applications. It is hoped that this research can serve as a basis for further development in audio processing and anti-aliasing techniques, and provide a better audio listening experience for all of us in the future.</p> <p>© 2021 TELNECT</p>	<p>Article History: <i>Received 30 April 2025</i> <i>Revised 10 Mei 2025</i> <i>Accepted 20 Mei 2025</i> <i>Available online 1 Juni 2025</i></p> <hr/> <p>Keyword: <i>(FIR) Finite Impulse Response,</i> <i>Anti Aliasing Sampling</i> <i>Technique,</i> <i>Audio Quality.</i></p>

1. INTRODUCTION

Kualitas audio merupakan elemen penting dalam berbagai aplikasi multimedia seperti musik, film, dan video game. Namun, aliasing, fenomena di mana frekuensi audio tinggi disampel pada laju yang rendah, dapat menyebabkan distorsi dan penurunan kualitas audio.[1].Penelitian ini mengusulkan teknik anti-aliasing canggih yang memanfaatkan algoritma mutakhir untuk memodelkan dan merekonstruksi sinyal audio dengan lebih akurat, menghasilkan kualitas audio yang lebih baik. [2].

Teknik anti-aliasing seperti oversampling dan low-pass filtering sering digunakan untuk mengatasi masalah aliasing [3]. Namun, teknik ini memiliki keterbatasan dalam hal efektivitas dan efisiensi. Oversampling dapat meningkatkan kualitas audio, tetapi membutuhkan daya komputasi yang besar. Low-pass filtering, di sisi lain, dapat mengurangi aliasing, tetapi juga dapat menghilangkan informasi frekuensi tinggi yang penting untuk kualitas audio [4].

Penelitian ini memperkenalkan teknik anti-aliasing yang lebih canggih untuk meningkatkan kualitas audio. Kami menggunakan algoritma yang lebih maju untuk memodelkan dan merekonstruksi sinyal audio dengan lebih akurat, sehingga menghasilkan kualitas audio yang lebih baik. Dalam penelitian ini, kami menggunakan MATLAB untuk melakukan eksperimen dan berhasil membersihkan gangguan noise pada audio serta menghilangkan efek aliasing. Paper ini bertujuan untuk memperkenalkan teknik anti-aliasing, sampling, dan cara mengatasi noise pada audio yang canggih untuk meningkatkan fidelitas audio [5]. Paper ini akan membahas secara detail tentang metodologi yang digunakan, hasil eksperimen, dan analisis yang dilakukan. Penelitian ini diharapkan dapat memberikan kontribusi penting dalam pengembangan teknik anti-aliasing dan pengolahan audio untuk meningkatkan kualitas suara dalam berbagai aplikasi serta memberikan pengalaman mendengarkan audio yang lebih baik bagi penggunanya [6].

Penelitian ini berbeda dengan penelitian sebelumnya yang berfokus pada satu aspek saja, seperti pengurangan aliasing [7], atau pengaruh frekuensi sampling [8]. Kami fokus pada penggunaan filter FIR (Finite Impulse Response) untuk meningkatkan kualitas audio dalam konteks aplikasi sehari-hari seperti musik dan video game.

Kebaruan utama penelitian ini terletak pada pendekatan holistik yang menangani noise dan aliasing secara bersamaan. Teknik yang dikembangkan tidak hanya meminimalkan aliasing, tetapi juga mempertahankan informasi frekuensi tinggi yang penting untuk kualitas audio. Dengan menggunakan MATLAB sebagai alat simulasi, kami dapat menganalisis dan memvisualisasikan data dengan lebih baik, sehingga mendapatkan pemahaman yang lebih mendalam tentang efektivitas teknik yang diusulkan.

Penelitian ini diharapkan dapat memberikan kontribusi yang bermanfaat dalam pengembangan teknik anti-aliasing dan pengolahan audio. Hasilnya dapat meningkatkan kualitas suara dalam berbagai aplikasi dan memberikan dasar yang kuat untuk pengembangan lebih lanjut dalam bidang pengolahan audio.

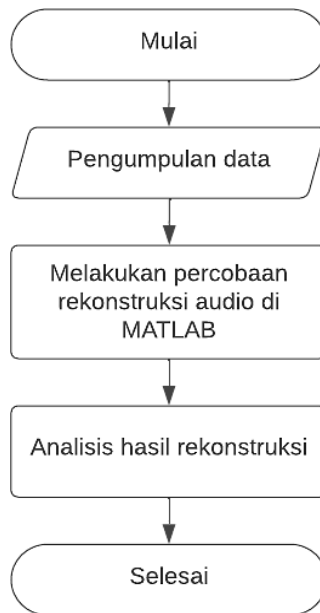
2. MATERIALS AND METHODS

2.1 Pengertian Metode

Metode penelitian ini menggunakan pendekatan empiris untuk menguji seberapa efektif teknik anti-aliasing dalam meningkatkan kualitas audio [9]. Pendekatan ini berarti penelitiannya mengandalkan pengumpulan data dan analisis untuk menguji sebuah hipotesis. Dalam konteks ini, hipotesis yang diuji adalah bahwa teknik anti-aliasing dapat meningkatkan kualitas audio dengan mengurangi efek aliasing dan kebisingan yang tidak diinginkan.

Pertama, peneliti mengumpulkan data audio yang telah terkena efek aliasing dan kebisingan sebagai input untuk eksperimen. Kemudian, teknik anti-aliasing diterapkan pada data ini menggunakan metode yang telah ditentukan sebelumnya. Setelah itu, dilakukan analisis terhadap audio yang telah diolah untuk membandingkan kualitasnya sebelum dan sesudah penerapan teknik anti-aliasing. Metode ini memungkinkan peneliti untuk mendapatkan bukti empiris yang dapat diandalkan tentang efektivitas teknik anti-aliasing dalam konteks pengolahan audio. Dengan demikian, hasil dari penelitian ini diharapkan dapat memberikan pemahaman yang lebih baik tentang bagaimana teknik anti-aliasing dapat berkontribusi dalam meningkatkan pengalaman mendengarkan audio yang lebih jernih dan berkualitas.

Figure 1 adalah alur yang kami lakukan pembuatan paper ini.



Gambar 1. Flowchart

Berikut merupakan garis besar mekanisme kami lakukan.

2.2 Persiapan Data

Lakukan proses pengumpulan data mulai dari mengumpulkan lagu dengan yang terkena *noise* untuk *input* sebagai simulasi dan pastikan *audio* tersebut dengan format '.wav' atau '.mp3'. Berikut adalah prosedur untuk memproduksinya :

1. Persiapan Sinyal *Audio*
 - a. Siapkan sinyal audio musik klasik, suara vokal, dan efek suara *game* dalam format WAV.
 - b. Pastikan semua sinyal audio memiliki format yang sama (16 bit, 44.1 kHz).
2. Penerapan *Filter FIR*
 - a. Implementasikan *filter FIR low-pass* dan *high-pass* di MATLAB.
 - b. Terapkan *filter FIR low-pass* dan *high-pass* pada setiap sinyal *audio* secara terpisah.
 - c. Bandingkan hasil *filter FIR* dengan sinyal *audio* asli.
3. Penerapan Algoritma *Sampling*
 - a. Terapkan *oversampling* dan *undersampling* pada setiap sinyal *audio* secara terpisah.
 - b. Bandingkan hasil *sampling* dengan sinyal *audio* asli.
4. Analisis Kualitas *Audio*
 - a. itung metrik kualitas *audio* seperti *Signal-to-Noise Ratio (SNR)*, *Peak Signal-to-Noise Ratio (PSNR)*, dan *Mean Absolute Error (MAE)* untuk setiap percobaan.
 - b. Visualisasikan hasil analisis kualitas *audio* menggunakan grafik dan tabel.
5. Reproduksi Eksperimen
 - a. Sediakan kode MATLAB yang digunakan dalam penelitian ini.
 - b. Berikan instruksi yang jelas dan ringkas untuk mereproduksi eksperimen.

2.3 Pengujian

Pada proses ini dilakukan suatu pengujian yang akan dilakukan pada *software* yang bernama *simulink* matlab dengan menggunakan bahasa pemrograman yang telah diatur pada *simulink* matlab untuk membaca file *audio* dan mengkonversi data menjadi format yang akan diproses.

1. Dengan proses pengenalan *audio* dimulai dengan memasukan fungsi "*audioread*" dari file *audio* yang telah disiapkan dengan frekuensi sampel asli akan tersimpan pada variabel "*original fs*"
2. Lakukan proses *resampling*, dengan proses *resampling* ke frekuensi sampel yang lebih rendah dengan menggunakan "*resample*" yang bertujuan tentang pembersihan pada sebuah audio

3. Lalu lakukan visualisasi audio sebelum pembersihan *noise* yang bertujuan untuk *audio* yang sudah di *resampling* divisualisasikan guna untuk menunjukkan keberadaan *noise* sebelum pembersihan
4. Pada percobaan ini menggunakan teknik perancangan *filter FIR* maka dari itu menggunakan fungsi 'fir1' dengan parameter *order* dan frekuensi yang diinginkan
5. Pembersihan *noise*, *filter FIR* diterapkan ke audio resample yang bertujuan untuk membersihkan *noise*
6. Lakukan proses *sampling* dari *audio* yang telah dibersihkan ke frekuensi asli, pada percobaan tersebut mengalami proses *aliasing* karena adanya proses *undersampling* saat *resampling audio* ke frekuensi sampel yang lebih rendah, karena tujuan percobaan ini membersihkan *audio noise* pada *audio* menggunakan teknik *FIR*, karena proses *resampling* menghasilkan *aliasing* potensial, dengan melakukan proses *oversampling* untuk mengembalikan *audio* ke sampel aslinya dengan harapan untuk mengurangi atau menghilangkan efek *aliasing* tersebut.
7. Lalu terakhir proses penyimpanan *audio* bersih dengan *audio* yang sudah bersih disimpan pada file baru menggunakan "audiowrite"

2.4 Menganalisis Hasil Percobaan

Setelah melakukan pengujian, langkah berikutnya adalah menganalisis hasilnya. Analisis ini mencakup:

1. Visualisasi: Bandingkan bentuk gelombang *audio* sebelum dan sesudah pembersihan *noise* dengan menggunakan grafik. Perhatikan perbedaan dalam amplitudo dan frekuensi.
2. Pengukuran Kualitas Audio: Ukur kualitas audio menggunakan metrik seperti *Signal-to-Noise Ratio (SNR)* dan *Total Harmonic Distortion (THD)* untuk melihat peningkatan yang terjadi.
3. Analisis Frekuensi: Gunakan Transformasi *Fourier* untuk membandingkan spektrum frekuensi dari *audio* sebelum dan sesudah pembersihan *noise*. Lihat apakah komponen frekuensi yang tidak diinginkan berkurang.
4. Evaluasi Subjektif: Dengarkan *audio* sebelum dan sesudah pembersihan *noise* untuk mengevaluasi perbedaan kualitas suara secara langsung.
5. Dokumentasi: Catat semua pengamatan dan pengukuran dalam bentuk tabel atau grafik untuk dokumentasi yang jelas dan sistematis.

2.5 Menuliskan Hasil Percobaan

Langkah terakhir adalah menyusun laporan hasil percobaan yang telah dilakukan secara rinci dan mudah dipahami. hasil yang di harapkan pada percobaan ini adalah :

1. Penerapan filter *FIR low-pass* dan *high-pass* diharapkan dapat mengurangi aliasing dan noise pada sinyal audio.
2. Penerapan *oversampling* dan *undersampling* diharapkan dapat meningkatkan atau menurunkan kualitas audio, tergantung pada faktor *sampling* yang digunakan.
3. Metrik kualitas audio (*SNR*, *PSNR*, *MAE*) diharapkan dapat menunjukkan perbedaan kualitas audio antara sinyal asli dan sinyal yang diproses.

3. RESULTS AND DISCUSSION (Arial 12 pt)

Di era seperti sekarang teknologi sudah kian canggih dan berkembang sangat pesat di tiap tahunnya, maka dari itu ketika kita ingin mendengarkan lagu zaman dahulu tentu kita bisa akses lagu tersebut dengan mudah melalui internet, namun ada beberapa kualitas audio pada lagu zaman dahulu yang kurang bagus seperti yang kita dengan sekarang, tentu saja kualitas pada zaman terdahulu sangat berbeda dikarenakan keterbatasan teknologi, oleh karena itu sebagai permasalahan untuk meningkatkan kualitas sehingga *audio* dapat didengar jelas dan mudah dipahami, kita dapat mengupayakan untuk meningkatkan kualitas *audio* dengan proses pengurangan *noise* dengan menjaga suara atau sinyal aslinya. Tanpa *aliasing*, *resampling* kembali menjadi sinyal digital tanpa mengurangi sinyal aslinya [10]. Pada percobaan ini, kami melakukan proses simulasi pengurangan noise pada audio

dengan menggunakan teknik *FIR* pada aplikasi *simulink* MATLAB, ada pun *filter filter* yang digunakan untuk meningkatkan kualitas audio salah satunya Bandpass filter yang dimana filter ini dapat meningkatkan kualitas audio, namun ada beberapa penting dalam kedua filter tersebut dalam pengolahan sinyal digital pada *simulink* MATLAB, dimana *FIR* lebih fleksibilitas desain stabilitas dan fase linear namun membutuhkan order filter yang tinggi dan komputasi yang lambat, berbeda dengan *bandpass filter* yang lebih efisien dalam komputasi dan cocok ekstraksi frekuensi spesifik namun desainnya kurang fleksibel dan berpotensi tidak stabil sehingga kami memilih untuk menggunakan teknik *FIR* karena yang fleksibilitas desain dan stabilitas yang tinggi sehingga kami menguji sample audio pada *simulink* MATLAB dan berhasil menemukan hasil *audio* yang berbeda dengan kualitas yang lebih baik aslinya dengan menggunakan teknik *FIR* berikut program yang kami buat dan kami terapkan di *software Simulink* MATLAB pada proses percobaan yang kami lakukan.

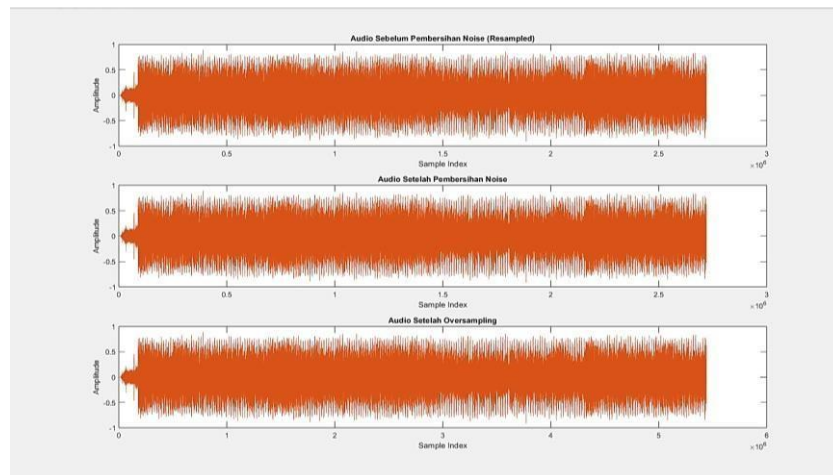
```
% Load audio file with noise
[y, Fs] = audioread('noisesound.mp3');
% Original Sampling Rate
original_Fs = Fs;
% Resample audio to a lower sampling rate (undersampling)
target_Fs = Fs / 2; % Target sampling rate (undersampling)
y_resampled = resample(y, target_Fs, Fs);
% Visualize audio before noise cleaning
figure;
subplot(3,1,1);
plot(y_resampled);
title('Audio Sebelum Pembersihan Noise (Resampled)');
xlabel('Sample Index');
ylabel('Amplitude');
% Design FIR filter
order = 100; % Filter order
cutoffFreq = 4000; % Filter cutoff frequency (Hz)
normalized_cutoffFreq = cutoffFreq / (target_Fs / 2); % Normalize cutoff frequency
filterCoeff = fir1(order, normalized_cutoffFreq, 'low');
% Apply filter to audio
y_clean = filter(filterCoeff, 1, y_resampled);
```

```

% Visualize audio after noise cleaning
subplot(3,1,2);
plot(y_clean);
title('Audio Setelah Pembersihan Noise');
xlabel('Sample Index');
ylabel('Amplitude');
% Resample audio back to original sampling rate (oversampling)
y_resampled_back = resample(y_clean, original_Fs, target_Fs);
% Visualize oversampled audio
subplot(3,1,3);
plot(y_resampled_back);
title('Audio Setelah Oversampling');
xlabel('Sample Index');
ylabel('Amplitude');
% Save cleaned audio
audiowrite('audio_cleaned_with_aliasing.wav',y_resampled_back,original_Fs);

```

Pada percobaan ini, kami melakukan analisis dan pengolahan audio untuk mengatasi masalah *noise* menggunakan MATLAB. Langkah pertama adalah memuat file audio yang mengandung *noise*, kemudian mengurangi frekuensi sampling asli menjadi setengahnya untuk menunjukkan efek aliasing yang terjadi. Setelah itu, kami merancang dan menerapkan filter *FIR* dengan orde 100 dan frekuensi *cut off* 4000 Hz untuk membersihkan noise dari audio yang telah di resampling. Visualisasi dilakukan pada setiap langkah untuk memperlihatkan perubahan yang terjadi pada audio sebelum dan sesudah pembersihan *noise*. Terakhir, audio yang telah dibersihkan dan di resampling kembali ke frekuensi asli disimpan dalam format baru yaitu berformat (wav) yang merupakan format yang berbeda dari audio yang kami input. Praktikum ini memberikan kami pemahaman yang baik tentang proses pengolahan audio dan penggunaan filter untuk mengatasi masalah *noise*, serta pentingnya visualisasi dalam memahami efek dari proses pengolahan tersebut. *Output* sinyal yang dihasilkan dari pembahasan ini terlihat pada Gambar 2.



Gambar 2. Representasi sinyal audio setelah pembersihan *noise*.

Terlihat pada gambar hasil simulasi di MATLAB bahwa representasi sinyal *audio* yang telah dibersihkan dari *noise* dan yang belum membersihkan *noise* menggambarkan perbedaan yang signifikan dalam kualitas *audio*. Sebelum proses pembersihan *noise*, sinyal audio terlihat terdistorsi dan tidak jelas, dengan keberadaan *noise* yang mengganggu. Namun, setelah penerapan teknik *FIR* dan proses pembersihan *noise*, sinyal audio menjadi lebih jernih dan terdefinisi dengan baik, dengan *noise* yang berkurang secara signifikan. Hal ini menunjukkan efektivitas dari teknik pengolahan audio dalam meningkatkan kualitas sinyal audio dengan menghilangkan gangguan *noise*. Proses visualisasi ini juga mempermudah pemahaman tentang dampak dari penggunaan filter dalam membersihkan *noise* pada sinyal audio. Dengan demikian, praktikum ini memberikan wawasan yang berharga tentang teknik pengolahan audio dan pentingnya pembersihan *noise* dalam meningkatkan kualitas sinyal audio.

4. Discussion

Pada percobaan ini, kami melakukan proses simulasi pengurangan noise pada audio dengan menggunakan teknik *FIR* pada aplikasi simulink MATLAB, ada pun beberapa filter lainnya yang digunakan untuk meningkatkan kualitas audio salah satunya bandpass filter yang dimana filter ini dapat meningkatkan kualitas audio, namun ada beberapa penting dalam kedua filter tersebut dalam pengolahan sinyal digital pada simulink MATLAB, dimana *FIR* lebih fleksibilitas desain, stabilitas dan fase linear namun membutuhkan order filter yang tinggi dan komputasi yang lambat [11]. Berbeda dengan bandpass filter yang lebih efisien dalam komputasi dan cocok ekstraksi frekuensi spesifik namun desainnya kurang fleksibel dan berpotensi tidak stabil. [9]. Sehingga kami memilih untuk menggunakan teknik *FIR* karena yang fleksibilitas desain dan stabilitas yang tinggi dan sangat minim untuk tidak stabil pada hasilnya sehingga kami menguji dengan mengambil sampel audio dengan kualitas rendah, lalu dilanjut dengan proses simulasi meningkatkan kualitas audio bantuan aplikasi simulink MATLAB sehingga kami berhasil menemukan hasil audio yang berbeda dengan kualitas yang lebih baik aslinya dengan menggunakan teknik *FIR*..

5. CONCLUSION

Penggunaan teknik anti-aliasing saat mengurangi gangguan noise pada audio melalui proses sampling dan filter *FIR* dapat meningkatkan kualitas audio secara signifikan. Melalui simulasi dengan MATLAB, kami berhasil membersihkan gangguan *noise* pada *audio* dan menghilangkan efek *aliasing* yang mengganggu. Ini sangat penting karena kualitas *audio* yang baik sangat dibutuhkan dalam berbagai aplikasi seperti musik, film, dan *video game*. Dengan kata lain, penelitian ini membawa kontribusi penting dalam pengembangan teknik anti-aliasing dan pengolahan *audio* untuk

meningkatkan kualitas suara dalam berbagai konteks aplikasi. Selain itu, praktikum ini juga memberikan pemahaman yang lebih baik tentang pentingnya membersihkan *noise* dalam meningkatkan kualitas *audio*. Diharapkan penelitian ini dapat menjadi dasar untuk pengembangan lebih lanjut dalam pengolahan *audio* dan teknik *anti-aliasing*, serta memberikan pengalaman mendengarkan *audio* yang lebih baik untuk kita semua kedepannya.

6. ACKNOWLEDGEMENT

Ucapan Terima Kasih: Penelitian ini merupakan hasil kerjasama serta dukungan dari banyak pihak. Kami ingin berterima kasih kepada semua pihak yang telah membantu yang tidak dapat kami sebutkan satu-persatu, baik secara langsung maupun tidak langsung. Semoga penelitian ini kedepannya dapat dikembangkan lagi menjadi lebih baik dan juga memberikan manfaat bagi perkembangan ilmu pengetahuan dan teknologi di masa mendatang.

7. REFERENCES

- [1] E. S. Wei, "Aliasing-Free Nonlinear Signal Processing Using Implicitly Defined Functions," *IEEE Access*, vol. 10, no. June, pp. 76281–76295, 2022, <https://doi.org/10.1109/ACCESS.2022.3192387>.
- [2] A. Penentuan, "Pengaruh Frekuensi Sampling Terhadap Waktu dan Akurasi Penentuan Lokasi Menggunakan Metode TDOA Multilaterasi," vol. 1089, 2021.
- [3] Y. H. Putra, J. Adler, and G. Gunawan, "APLIKASI FILTER FINITE IMPULSE RESPONSE (FIR) UNTUK MENGHILANGKAN NOISE PADA SUARA MANUSIA MENGGUNAKAN GRAPHICAL USER INTERFACE (GUI) PEMROGRAMAN MATLAB".
- [4] F. C. S. Adiyanta, "Hukum dan Studi Penelitian Empiris : Penggunaan Metode Survey sebagai Instrumen Penelitian Hukum Empiris," vol. 2, no. 4, pp. 697–709, 2019. <https://doi.org/10.14710/alj.v2i4.697-709>.
- [5] A. Chandra and S. Chattopadhyay, "Engineering Science and Technology , an International Journal Design of hardware efficient FIR filter : A review of the state-of-the-art approaches," *Eng. Sci. Technol. an Int. J.*, vol. 19, no. 1, pp. 212–226, 2016. <https://doi.org/10.1016/j.jestch.2015.06.006>.
- [6] S. Gunawan, E. S. Rahman, and U. N. Makassar, "Metode Penentuan Perbaikan Noise Pada Data Musik Menggunakan Algoritma Least Mean Square," vol. 01, no. November, pp. 48–56, 2023. <https://doi.org/10.61220/digitech.v1i1.20235>.
- [7] K. Da Xu, F. Zhang, Y. Liu, and Q. H. Liu, "Bandpass Filter Using Three Pairs of Coupled Lines With Multiple Transmission Zeros," *IEEE Microw. Wirel. Components Lett.*, vol. 28, no. 7, pp. 576–578, 2018, <https://doi.org/10.1109/LMWC.2018.2835643>.
- [8] K. C. Jayashree, K. M. Bharamappa, P. M. Kumar, G. M. Santeep, and S. Saxena, "Denoising Real-Time Audio Signals Using Matlab Filter Techniques Denoising real-time audio signals using matlab filter techniques," 2020.
- [9] T. Fft, "Simulasi Penghilangan Noise Pada Sinyal Suara Menggunakan Metode Fast Fourier," pp. 20–25, 2019. ISSN : 2622-8106. <https://doi.org/10.20895/INISTA.V1I2>.
- [10] R. J. Putra, "Implementasi filter digital fir (finite impulse response) pada field programmable gate arrays (fpga)," pp. 1–5.
- [11] A. Al Havis, L. Fitria, P. T. Informatika, F. Teknik, and U. Samudra, "Filtering Sinyal Menggunakan Band Pass Filter," vol. 19, no. 2, pp. 37–48, 2019. <https://doi.org/10.13140/RG.2.2.19091.94247>.