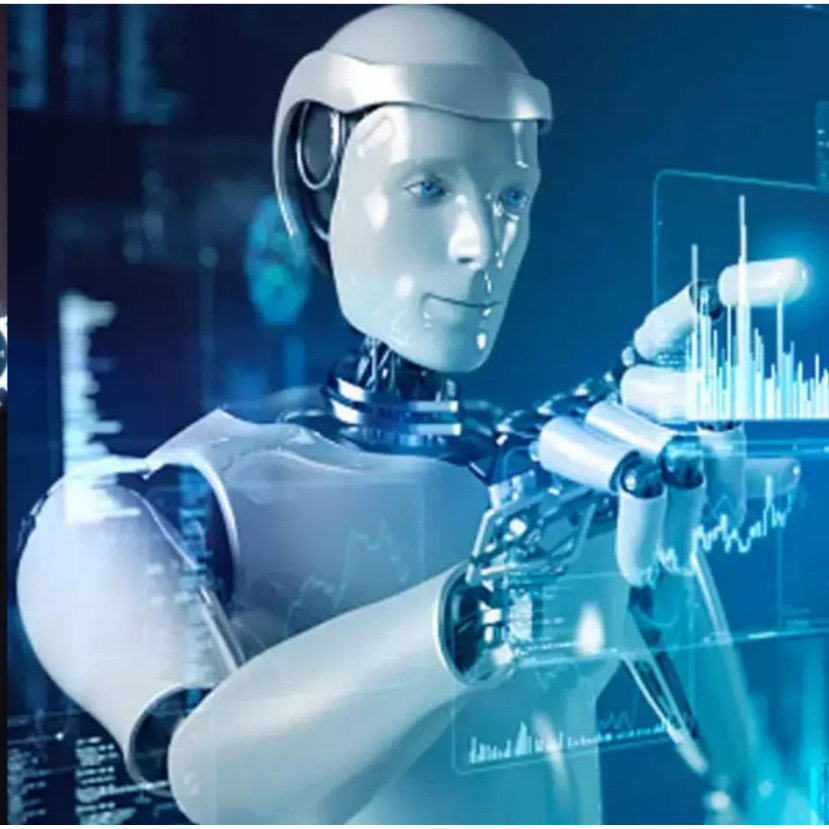
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 12, December 2025

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 12, December 2025

|DOI: 10.15680/IJIRSET.2025.1412075|

Audio Sound-Shield

Sindhu M G, Dr. Sanjay K S

P.G. Student, Department of Master of Computer Application, PES Institute of Technology and Management, Shivamogga, Karnataka, India

Associate Professor, Department of Master of Computer Application, PES Institute of Technology and Management, Shivamogga, Karnataka, India

ABSTRACT: These days, managing information isn't just an ice-to-have for organizations it's essential. Stick with old, manual systems and things get out of handfast. Suddenly you're dealing with duplicate files, errors, wasted time basically, ahead ache you didn't ask for. That's what makes this project so important .The goal here is to build automated software that cuts out all that hassle. It keeps your data accurate and safe, plus you get a solid digital platform you can count on. So, here's the deal: everything runs through a single hub. Users sign in, the system checks their credentials, and decides exactly what they can or can't do. With role-based access, only the right people get to sensitive info no more worrying about the wrong eye son your data .It's more secure ,and it's always clear who's responsible for what. The system keeps track of user sessions and makes sure everyone logs out safely, so your information stays protected the whole time. The whole setup has three main parts: frontend, backend, and database. The frontend is what people actually see and use it's built to be simple and user-friendly. The backend takes care of all the requests, runs the main logic, checks inputs, and connects securely with the database. And the database? That's the vault where everything gets stored. It's designed to keep your data organized, easy to find, and free from duplicates or mix-ups.

KEYWORDS: Sound Shield, Noise Detection, Sound Monitoring System, Audio Signal Processing, Noise Pollution Control, Real-Time Analysis, Data Management.

I. INTRODUCTION

Noise is everywhere these days. Cities keep growing, traffic never stops, construction is always humming in the background, and people just blast music and videos louder than ever. It all piles up. Most of the time, we barely notice how loud our world's gotten. But honestly, it's more than just annoying. Noise isn't just annoying—it wears you down. It messes with your sleep, shreds your focus, and honestly, it can mess up your hearing for good. Doesn't matter where you are—home, school, work, even the hospital—if there's too much noise, comfort goes out the window. Suddenly, everything just feels harder. That's where Sound Shield comes in. Instead of tuning things out, Sound Shield actually pays attention. It listens to your surroundings, breaks down the sounds, and tells you exactly how loud it is in decibels. If the noise level shoots up, you'll know right away. No guessing. You get real-time alerts, sharp sound analysis, actual feedback you can use. Sound Shield isn't just a gad get that tracks noise—it makes you notice what's going on, helps you care, and gives you the power to take back your space. Whether you're working, relaxing, or trying to sleep, it helps you feel comfortable again.

II. LITERATURESURVEY

This research digs in to how we're actually dealing with noise pollution these days, especially now that we've got all this new tech to track sound. Scientists aren't just complaining—they're warning us that noise does real harm to our health. So yeah, paying attention to the noise around us matters, whether you're downtown, stuck at your desk, or just trying to unwind at home. People have rolled out all sorts of fixes: digital signal processing, new kinds of sensors, even automatic alerts, all designed to help keep noise in check. It's necessary to take ac loser look at what has been

accomplished and what has been failed, which is exactly what leads to the development of something even better, such as Sound Shield. [1]. The authors focus on real-time data acquisition using microphones connected to embedded controllers. The system provides immediate feedback when noise crosses predefined limits, making it suitable for urban

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 12, December 2025

|DOI: 10.15680/IJIRSET.2025.1412075|

noise monitoring. The research highlights the importance of low-cost hardware solutions but also points out limitations in data storage and long-term analysis capabilities.[2]. This paper looks at an automated noise detection system built for cities. It uses sound sensors and a simple threshold-based approach to spot when noise gets too loud, the n send sound alerts. The authors care about keeping things fast and easy to set up, so the system responds in real time and doesn't take much work to install. Still, it's really just focused on catching noise in the moment there's not much here for long-term data, fancy visualizations, or digging into past trends. [3], the authors dive into audio signal processing. They break down different ways to analyze sound patterns things like intensity and frequency. Digital filtering and signal transformation come up a lot ;the techniques help measure noise more accurately.[4] suggests an IoT-based framework for monitoring noise. Sensors pick up sound at different spots, then send all that info to one central server. This setup makes remote monitoring and data storage possible, so you can use it in smart cities. But there's a catch. The authors point out that while IoT systems can scale up easily, they also get more complicated and expensive to build. [5] zooms in on noise monitoring where quiet really matters think hospitals or schools. The system is pretty straight forward :if noise spikes, it alerts staff so they can handle it right away. It works, but the study raises a fair criticism. These systems aren't one-size- fits-all. They don't always fit every environment, and people need more flexibility than what's currently offered.. Check out what's on the market, and you'll see the same thing over and over. There are plenty of sound monitoring systems, but most of them fail to impress. They struggle to catch noise fast enough, they're not all that user-friendly, and they don't handle data that well.

III. PROPOSEDMETHODOLOGY

The methodology for the Sound Shield solution will provide are liable and interpretable solution pipeline for identifying deepfakes within audio with structured machine learning approaches and detailed acoustic feature analysis. To start, a user will upload a WAV file via an easy-to-use and friendly interface. As soon as the user finishes uploading, the solution will start with immediate structured preprocessing. At this stage, a preprocessing procedure moves noise and refines audio signals by eliminating silence and resampling it all uniformly. After that, with processed and unified audio, it proceeds to feature extraction, which generates a set of acoustic features involving MFCCs, chroma features, spectral centroid, spectral roll-off, spectral band-width, zero-crossing rate, and energy. These act as mathematical signatures of corresponding acoustic signals. Once the extracted, they will be passed on for classification, where in a pre-trained SVM model will analyze these features. The pre-trained SVM model uses a set of pre-defined features and identifies classification markers based on pre-defined samples. Once it identifies classification markers, it continues with predictions. The result will identify either REAL or DEEPFAKE. It then enters visualization steps, which will provide waveform plots and feature graphs, among others. At this stage, it aims at enhancing transparency and explainability. At last, it will display all result outputs with accuracy interactive and clean solution offering user satisfaction. Also, it should be easier and more efficient with accuracy and transparency guaranteed. Thus, with structured methodology incorporated at all steps, it will be an efficient solution for addressing modern deepfakes within sound and an efficient solution with accuracy.

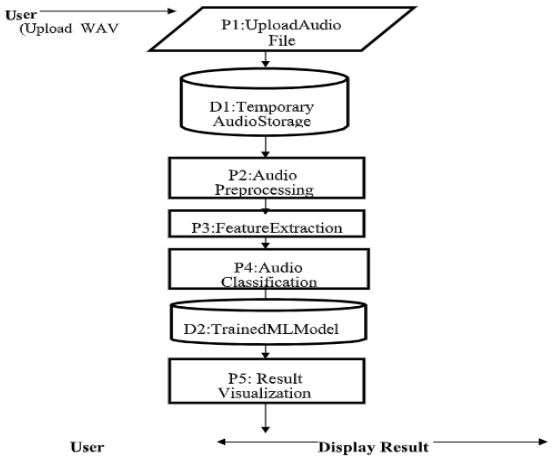


Fig1. Data Flow Diagram

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 12, December 2025

|DOI: 10.15680/IJIRSET.2025.1412075|

This diagram illustrates the complete dataflow of the Sound-Shield audio-analysis system, showing how user input travels through the processing pipeline to produce meaningful results. The process begins when the user uploads a WAV file, which enters P1: Upload Audio and is immediately stored in D1: Temporary Audio Storage for further operations. From here, the audio moves into P2: Audio Preprocessing, where noise reduction, normalization, and signal cleaning are performed to enhance quality. The refined audio is then passed to P3: Feature Extraction, where critical acoustic features such as MFCCs, chroma values, spectral centroid, and band width are computed. These extracted features flow into P4: Audio Classification, where the trained machine learning model predicts whether the audio is real or deepfake. The classifier relies on D2: Trained ML Model, which contains pre-learned patterns from previously labeled datasets. After the prediction is generated, the output proceeds to P5: Result Visualization, where waveforms, spectrograms, and classification labels are produced for user interpretation. Finally, the complete results are displayed back to the user in a clean interface, closing the data flow loop. Overall, the diagram capture show raw audio is transformed step-by-step into an intelligent and interpretable decision.

Algorithms Used: The Sound Shield system keeps things simple and actually works. Forget fancy math or complicated tricks just straightforward steps anyone can follow, all happening right as you use it.

1.Sound- Level- Calculation: The device begins by using a microphone or sensor to listen. The only thing it hears are the unprocessed, up-and-down waves. It transforms all of that noise into an actual decibel reading by chopping off a small amount of time just milliseconds and crunching the data for an average or RMS value.
2. Threshold-Based Decisions: The system must now determine if the noise is tolerable or becoming excessive. Its cutoff points are straight forward: Safe, Warning, and Danger. It compares the number to those boundaries every time it examines the sound. You're fine if the level is below the safe line. The system notifies you if it enters the warning zone. Go beyond the threat zone? The system prepares to notify you of the breach.
3. Threshold-Based Decisions: Now, the system has to figure out if the noise is okay or getting too loud. It uses clear cut off points: Safe, Warning, and Danger. Each time it checks the sound, it compares the number to those limits. If the level's under the safe line, you're good. If it pushes into the warning zone, the system gives you a heads-up. Go past the danger mark? That's aviolation, and the system gets ready to alert you.
4. Data Logging: Finally, the system keeps track of everything. Every so often, it writes down the decibel level, the time, and Whether things are Normal, Warning, or Danger. It saves all this in a file or database, so you can checkout graphs, run reports, or just see how noise changes over time.

Testing:
Were all put Sound Shield through the wringer. Every single part got its turn in the spot light—alone, and then aspart of the bigger picture. No short cuts. We tested everything: audio input, feature extraction, sound analysis, classification, the user interface, you name it. We bombarded it with all kinds of audio files. Some were perfectly clear, some were a mess, and a few were just bizarre. The goal? See how it handled both the usual stuff and the weird outliers. We wanted to make sure Sound Shield could pick up sounds accurately, tell different noises apart, and send out alerts right when you need them.
1. Unit Testing: We started with the basics, testing each piece on its own. Audio input, feature extraction, normalization, categorization—each one got its own turn. We threw test data at every part and watched to see if it did what it was supposed to.
2. Integration Testing: Once each part worked on its own, we hooked the together. We tracked how data moved from uploading, through feature extraction and the machine learning model, and finally to the user interface. We kept our eyes open for glitches or missing info, making sure nothing fell through the cracks.
3. System Testing: Then we tested the whole Sound Shield app as one unit. We loaded it with all sort so audio file stores how it analyzed, categorized, generated waveforms, and just performed overall.
4. FunctionalTesting:Wedidn't leave anything out. Uploading, playing audio, noisecateg orization, datadisplay—we checked that every feature worked the way it should.
5. Performance Testing: Speed counts. So we made sure the system could handle real-time audio with out breaking as weat. Nobody wants lag, so we pushed it hard to be sure it could analyze and categorize sounds fast.
6. Usability Testing: We paid attention to how easy it felt to use. Buttons, layout, instructions—they all had to make sense, whether you're a tech whiz or a total beginner.
7. Error Handling Testing: We even tried to break things on purpose. We uploaded corrupt files and weird formats just to see if Sound Shield would catch the problem, show a clear warning, and keep running instead of crashing.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 12, December 2025

|DOI: 10.15680/IJIRSET.2025.1412075|

Test Cases:

Testcase	Description	Input	Expected Output	Result
TC-01	Upload a valid way file and classify audio	User selects a valid. wavfile and clicks Classify	System accepts the file, processes it, and displays the result as REALorDEEPFAKE.	Pass
TC-02	Upload an invalid file format	Userselectsa.mp3/.txt/ unsupported file	System shows a warning/validation message asking the user to upload a valid wave file.	Pass
TC-03	No file uploaded and Classify button clicked	User clicks Classify without selecting any file	System displays a message like "Please upload an audio file to Begin and does not crash	Pass
TC-04	Display audio play back	User uploads a valid wavfile	System shows an audio player And allows the user to play the uploaded audio.	Pass
TC-05	Waveform visualization	User uploads a valid. wavfile and classification is done	System plots and displays the waveform of the uploaded Audio file using the wave form plotting function.	Pass
TC-06	Classify RealAudio	User uploads a known REAL audio sample from the dataset	System correctly predicts and displays the label as REAL	Pass
TC-07	Classify DEEP FAKE audio	User uploads a known DEEPFAKE audio sample From the dataset	System correctly predicts and displays the label as DEEPFAKE	Pass
TC- 08	Handle missing model file	Audio classification, deleted or not found	System shows an error message (or logs error) indicating that The model file is missing, With out crash	Pass

IV. EXPERIMENTAL RESULTS

The image series illustrates the complete flow process of a deepfake audio detection system named Sound Shield, demonstrating how a user can seamlessly upload an audio clip, analyze it, and achieve intelligent and interpretable results. The first image begins with an introduction to a clean and simple user interface, instructing a user on how to upload WAV files while pointing out its own prominent features, including MFCC, chroma, spectral feature extraction, and SVM classification. As soon as the user opens an internet file browser, orfileexplorer, ontheirdevice, then extimage illustratesa user selecting a sample from among their own collection, with an emphasis on easy and smooth connectivity with an internet file browser.

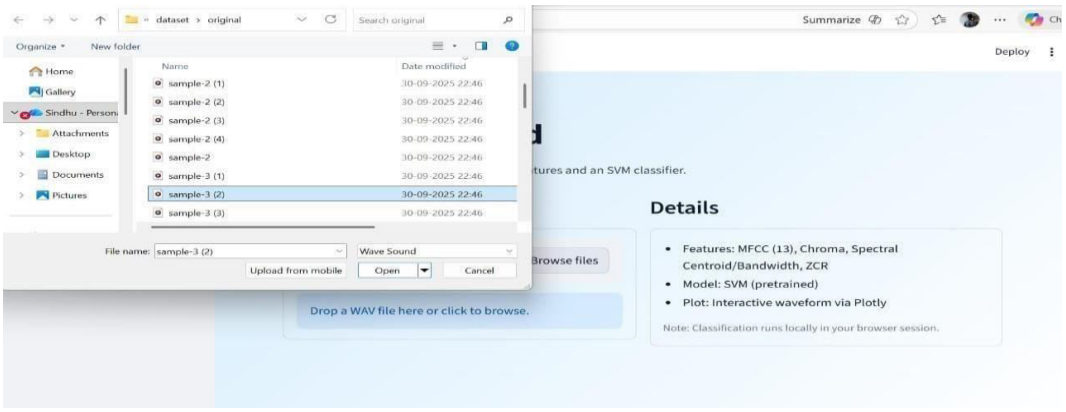
International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

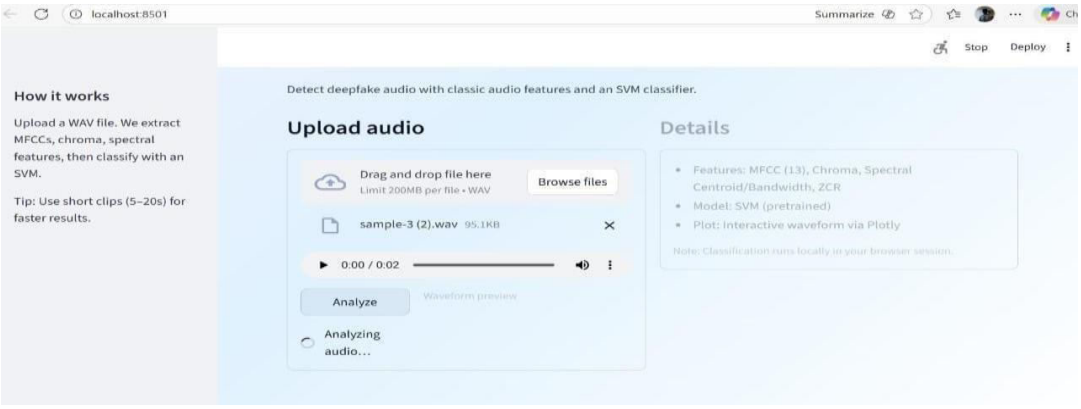
Impact Factor 8.699

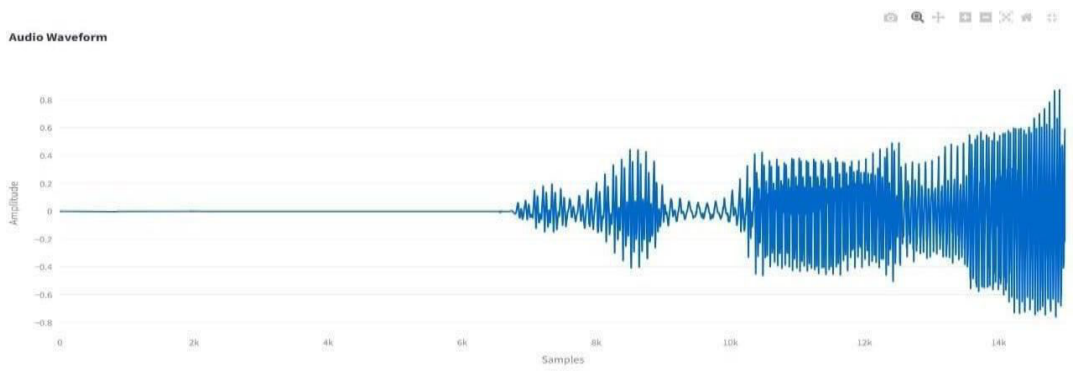
|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 12, December 2025

|DOI: 10.15680/IJIRSET.2025.1412075|







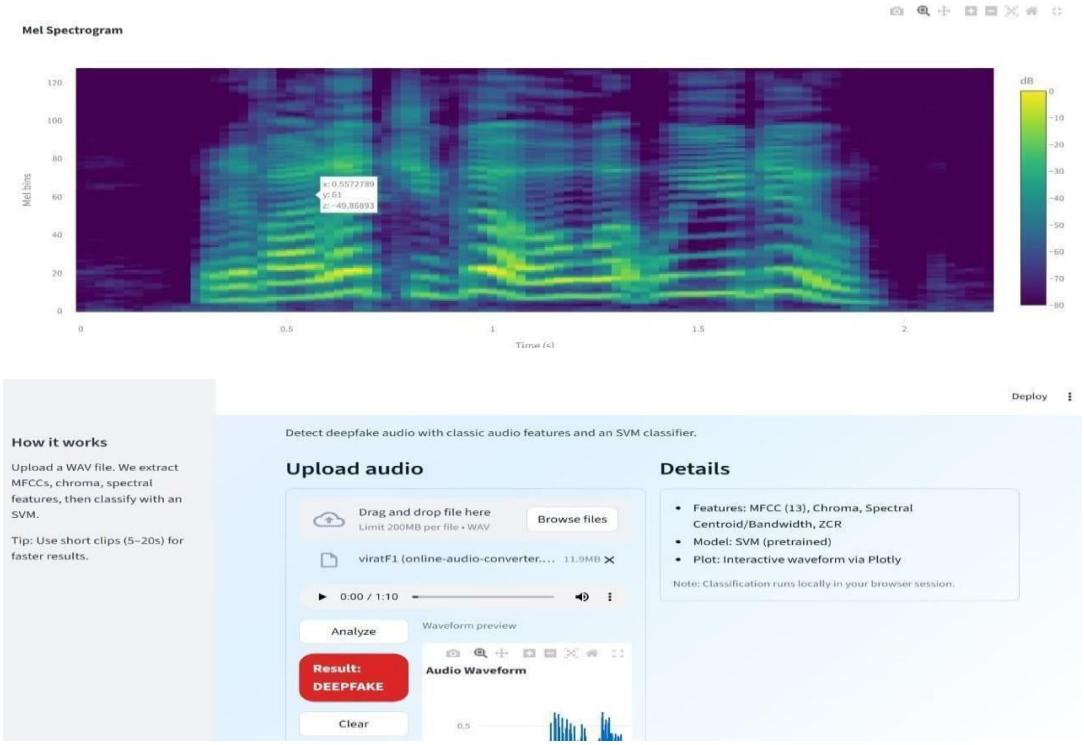
International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 12, December 2025

|DOI: 10.15680/IJIRSET.2025.1412075|



Once an audio clip is selected, the third picture shows uploaded file representation on the user interface with an inline link labeled as an "Analyze" button to seamlessly start computing. At that stage, it moves into an "Analyzing audio…" stage, indicating active computation on feature extraction and making model predictions. Moving forward to then extimage, itillustratesan"Analysis Complete result with a Result: REAL, implying confidence in an authentic result and visualization for better understanding. The next visualization illustrating amplitude changes and patterner presentation through an inline link labeled as Result: REAL depicts a detailed representation for understanding amplitude changes within an audio clip. When you look at the Mel Spectrogram, you get a whole new angle on the sound—it turns frequencies into colors, so you can actually see how the audio is built. Both the waveform and the spectrogram help you make sense of what's really happening in the audio. The last example shows a test where someone uploads an audio clip, and the system flags it as a DEEPFAKE. What stands out here is how clearly it separates real audio from fake. Basically, you get an inline link showing the deepfake result, along with a visual breakdown of your clip. There's also a link where you can check out both the waveform and the Mel spectrogram side by side, so you really get to see the full picture.

V. CONCLUSION

Sound-Shield isn't just another gadget. It actually listens to deepfake audio and figures out what's real. It digs into everything— zero-crossing rate, bandwidth, spectral centroid, chroma, MFCCs, Mel spectrograms. All those details bring together time and frequency info, so you get a fair look at what's going on in the audio. Right away, Sound-Shield sends those facts to a pretrained Support Vector Machine. That's what really draws the line between real and fake. There's no smoke and mirrors here. You can check out he wave form and spectrogram yourself and watch the whole thing playout live. There sults popup right next to the audio patterns. Gettingstarted is simple. Just upload your audio, hit play, and watch what happens. You don't need to be some kind of tech whiz— everything's built to be clear and straightforward. Sound-Shield isn't just about looking good. It's about being reliable and accurate, and it handles all kinds of audio testing. In a world where fake audio is everywhere, Sound-Shield helps you push back. The team reallyseesthisasproofthatmachinelearningmattersinaudioforensics.Andhonestly,thisisjustthebeginning.Soon,you'llseereal- time detection, bigger datasets, and sharper deep learning models.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 12, December 2025

|DOI: 10.15680/IJIRSET.2025.1412075|

REFERENCES

[1] Davis, S., &Mermelstein, P. (1980). Comparison of parametric representations for mono syllabic word recognition in continuously spoken sentences. IEEE Transactions on Acoustics, Speech, and Signal Processing.
[2] Müller,M.(2015).FundamentalsofMusicProcessing.Springer.(Coverschroma,spectralcentroid,bandwidth,and ZCR extraction).
[3] Cortes, C., & Vanik, V. (1995). Support-vector networks. Machine Learning.
[4] Yi,J.,Zhou,H.,&Baim.(2023).Audio Deepfake Detection
[5] Tahaoglu,G.(2025).Detecting deepfake audio using spectral features and machine learning architectures. Knowledge-Based Systems .
[6] Abdullah,A.,&Salih,O.M.(2024).DeepFakeAudioDetectionFrameworkusingMFCCs,ChromaFeatures,and Spectrogram Images. International Journal of Intelligent Systems and Applications in Engineering.
[7] Liu, X., Yamamoto, Y., & Kinnunen, T. (2022). ASVspoof 2021: Benchmark for Automatic Speaker Verification Spoofing and Deepfake Detection.
[8] Kinnunen, T., Sahidullah, M., Delgado, H., et al. (2017). Spoofing and countermeasures for speaker verification : A survey. Speech Communication.
[9] Chauhan, N. (2023). Deep Fake Audio Detection using MFCC and SVM (Open-source implementation). GitHub Repository.
[10] McKinney, W.et al.(2019). Plotly: A graphing library for interactive scientific visualizations. Plotly Technologies.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details