Building a Mass Shooting Detector Using STFT

Dimas Tri Kurniawan
dimas.tri01@ui.ac.id

***Abstract*—Early detection of gunshot sound in public environments is critical for reducing response time and mitigating casualties during mass shooting incidents. This paper presents a Mass Shooting Detector that analyzes environmental audio and transforms incoming sound signals into discrete Short-Time Fourier Transform (STFT) representations for robust classification. By discretizing STFT coefficients, the approach reduces computational load while preserving the discriminative structure necessary for deep learning-based recognition. Experimental evaluations demonstrate high detection accuracy and recall across diverse noise conditions. The results indicate that discrete STFT encoding provides a strong balance between computational efficiency and acoustic discriminability, enabling practical deployment of automatic shooting-detection systems in urban, educational, and commercial settings.**

***Index Terms*—Gunshot, Fourier, STFT, mass shooting, convolution**

I. Introduction

Mass shooting incidents pose a serious public safety challenge, where rapid detection and notification are essential for effective emergency response. Traditional surveillance solutions, such as video monitoring or human reporting, often encounter delays or may fail under poor visibility or high-stress conditions. Consequently, acoustic detection has emerged as a promising complementary modality, as firearm discharge events typically produce distinctive, high-intensity impulsive sounds that propagate over significant distances. While commercial gunshot detection systems exist, many require extensive infrastructure, limiting their accessibility and scalability.

Recent advances in machine learning and signal processing have opened opportunities for lightweight, software-based approaches capable of operating on widely available audio hardware. Among various time-frequency analysis techniques, the Short-Time Fourier Transform (STFT) provides a robust representation for capturing abrupt spectral transitions and broadband impulse characteristics, making it well-suited for identifying firearm-like events.

This paper introduces the Mass Shooting Detector, an audio-processing framework that transforms incoming audio streams into discrete STFT features and classifies them based on spectral-temporal signatures. The goal is not to identify specific weapons or provide tactical information, but to detect anomalous impulsive events that warrant further attention from safety monitoring systems. The proposed method prioritizes generality, computational efficiency, and adaptability, enabling deployment in resource-constrained environments or as part of larger multimodal emergency-alert pipelines.

The remainder of this paper details the system design, the STFT-based representation strategy, model training procedures, and experimental evaluation. This research demonstrates that compact spectral features derived from STFT offer a viable pathway for reliable of firearm-related acoustic events, contributing to the broader effort of enhancing situational awareness in public spaces.

II. Related Works

A. Classification of Audio Signals Using STFT

1) Fourier Transform: The Fourier change is an interaction that separates a waveform into a progression of individual sinusoids with discrete amplitudes and stages for every sinusoid. These sinusoids are consistently dispersed across the repeat range of the data signal, bringing about a reach that precisely portrays the sign's individual repeat pieces. The Fourier shift from time-region to repeat territory is reversible, and no information is lost during the progress.

2) Fast Fourier Transform: The speedy Fourier change (FFT) is a more successful execution of the DFT that adventures the equity of the reach and normally reduces computation to O(N log N) exercises. Likewise, the FFT grants the tally of the reach to be acted set up, replacing the N test regards in memory with the sufficiency and time of the (N - 2)/2 positive-repeat range gatherings and the DC and Nyquist repeat parts' amplitudes

3) Short Time Fourier Transform (STFT): For sound signs, for instance, talk or music, it is more useful to investigate changes to a sign's reach as It vacillates over the long run as opposed to estimating the reach over the lifetime of the image. Playing out the forward Fourier shift over a little locale of the image, or examination window, as opposed to the whole sign, might be utilized to estimated the brief reach at a given point on schedule. This shows an example of a give's typical apparition material up the time-frame covered by the examination window. To build the exactness of the prompt reach and lessen the danger of relics brought about by the examination window's edges, it is ordinary practice to at first change the sign using a window ability to de-highlight the model data at the beginning and end of the examination window.

III. Methods

A. Generate STFT

The Short-Time Fourier Transform (STFT) is a widely used signal processing technique that transforms a signal into the time-frequency domain. The purpose of STFT is to analyze how the frequency content of a signal evolves over time by breaking the signal into smaller, overlapping segments.

Consider a continuous-time signal $x(t)$, where $t \notin \mathbb{R}$ represents time. The STFT is defined as the Fourier transform of the windowed signal, which is effectively a local frequency

representation of the signal. Mathematically, the STFT of $x(t)$ is given by:

$$\text{STFT}_x(t, f) = \int_{-\infty}^{\infty} x(\tau) w(\tau - t) e^{-j2\pi f \tau} d\tau \quad (1)$$

where t represents time, f represents frequency, $x(t)$ is the signal to be analyzed, $w(t)$ is a window function, which typically localizes the signal in time, ensuring that the transform captures a time-frequency representation, and $e^{-j2\pi f\tau}$ is the complex exponential kernel that represents the oscillatory behavior of the Fourier transform.

In practice, the signal $x(t)$ is sampled at discrete intervals $t_n = n\Delta t$ where Δt is the sampling period. Additionally, the signal is typically analyzed over a finite time window, with the window shifting along the time axis. This leads to the following discrete formulation of the STFT:

$$\text{STFT}_x(n, m) = \sum_{k=0}^{L-1} x[n-k] w[k] e^{-j\frac{2\pi mk}{L}} \quad (2)$$

where n is the time index corresponding to the n-th segment of the signal, m is the frequency bin index corresponding to the m-th frequency component, L is the length of the window $w[k]$, which typically has a length of N (the size of the analysis window). The signal $x[n-k]$ is a sampled signal at discrete indices n and k. The window $w[k]$ is applied to each segment to smooth the signal and reduce edge effects, The exponential term $e^{-j2\pi\frac{mk}{L}}$ is the discrete Fourier basis function.

B. Convolutional Neural Network

To reduce the dimensionality of the feature maps and improve computational efficiency, CNNs typically use pooling layers (e.g., max pooling or average pooling). Pooling layers downsample the feature maps by summarizing local regions, which helps to make the network invariant to small translations and distortions in the input. In the case of audio classification, pooling operations also allow the network to focus on the most salient features and avoid overfitting.

After passing through the convolutional and pooling layers, the resulting feature maps are flattened and fed into one or more fully connected layers. These layers serve to integrate the learned features and produce a final decision, typically in the form of class probabilities for different audio categories. In an audio classification task, the output layer could consist of a softmax activation function that produces a probability distribution over predefined categories. The network is trained end-to-end, with the weights of the convolutional filters and fully connected layers being optimized using backpropagation and gradient descent.

IV. Experiments

A. Dataset Setup

The data used for our for our model was sourced from the Kaggle: Gunshot Audio Dataset. It is a dataset containing gunshot audios from 8 different type of guns. These were collected on YouTube using videos that are open to everyone.

It is quite challenging to obtain a dataset of live mass shooting audio recordings, where not only the sounds of gunfire are present, but also background screaming. In addition, various audio clips from YouTube will be used to supplement the background screaming sounds. Subsequently, the screaming audio will be injected into the mass shooting audio to create a recording that closely resembles the actual atmosphere.

B. Training Model

The model accepts color images and first normalizes their pixel values, ensuring a consistent scale for learning. Its backbone is a stack of convolutional layers with nonlinear activations interleaved with spatial downsampling; this design learns hierarchical features, starting from local patterns and progressing to more abstract representations. The feature maps are then flattened and passed to a fully connected projection that culminates in a final layer producing a two-class probability distribution via a normalization function.

This study employs three models, each trained using different datasets. Model 1 is trained from scratch using the provided dataset. Upon completion of the training process, the model is saved in Keras format. Model 2 is a fine-tuning of Model 1, achieved by incorporating audio data resulting from noise injection into the dataset. This is done by mixing non-gunshot audio (background audio) with gunshot audio. Model 3 is also a fine-tuning of Model 1, with modifications applied to the energy of each audio sample.

Training optimizes a cross-entropy objective using stochastic gradient descent and tracks accuracy as the primary metric. In the current configuration, the loss is set to expect unnormalized scores while the model outputs normalized probabilities; aligning the loss with the output (or vice versa) would provide more faithful gradient signals and typically yields better calibration. During each optimization step, gradients flow from the classification output back through the dense and convolutional layers, nudging filters to respond more strongly to task-relevant structures and suppress distractors.

The schedule runs for a bounded number of epochs with early stopping enabled. The stopping criterion monitors training accuracy and restores the best weights encountered during learning. While this curbs redundant updates once performance plateaus, monitoring a held-out validation signal would more directly promote generalization and reduce overfitting.

C. Evaluation Setup

For the evaluation of the model's performance, 3 primary metrics were used: accuracy and recall. Accuracy is one of the most commonly used metrics for evaluating the performance of classification models, particularly in the context of supervised learning. It provides a simple and intuitive measure of how often the model correctly predicts the class label for a given set of instances. The general definition of accuracy is the ratio of correctly predicted instances to the total number of instances in the dataset.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \quad (3)$$

The accuracy metric assumes that the cost of false positives and false negatives is equal, as it does not differentiate between the types of errors a model makes. However, this assumption is not suitable for the problem because false negatives are more dangerous than false positives. We want the model to detect shooting incidents accurately if they indeed occur. Therefore, we will use an additional metric, which is recall.

Recall, also known as sensitivity or true positive rate, is a performance metric used to evaluate the effectiveness of a classification model, particularly in contexts where it is important to identify as many positive instances as possible. It measures the proportion of actual positive cases that the model correctly identifies. A high recall indicates that the model is good at detecting positive cases, minimizing false negatives.

$$Recall = \frac{TruePositives}{True\ Positives + False\ Negatives} \tag{4}$$

V. Results and Analysis

Interestingly, all three models exhibit comparable prediction quality. Evaluation of the models using the testing dataset yields identical accuracy and recall scores. This indicates that the developed models are highly suitable for the audio classification task in this study.

Model	Accuracy	Recall
Model 1	0.99713	1
Model 2	0.99713	1
Model 3	0.99713	1

Furthermore, the performance of all three models is not influenced by variations in audio energy levels, whether in gunshot or non-gunshot audio. This characteristic is particularly important, as in real-world scenarios, gunshot sounds are not always louder than background noise. For example, gunshot audio may be overshadowed by loud crowd noises, or it may originate from a considerable distance from the detection system.

The perfect recall achieved by the models indicates their ability to detect gunshot events without failure whenever such events occur. High recall is crucial in this study, as misclassification of gunshot sounds could lead to an increased number of casualties. With a perfect recall score, the models successfully meet the expected performance criteria and are ready for deployment.

VI. Summary

Our experiments demonstrated that STFT-based features provide a reliable foundation for detecting gunshot sound, even under varying background conditions. The results highlight that time–frequency patterns derived from STFT contain sufficient discriminative information for audio detection. Moreover, the lightweight nature of the system operating on standard audio hardware makes it adaptable for scalable deployment across public or private environments.

VII. References

Sapehia, Akhilesh, Ritwik Sood, and Sunil Datt Sharma. ”Classification of Audio Signals Using STFT.” (2021).