National College of Ireland

Configuration Manual

"Hybrid Audio Event Recognition Using CNN and Transfer Learning: Comparative Study on Custom and Google Speech Commands Datasets"

MSc Research Project
MSCDAD_C

Hrushikesh Kumthekar
Student ID: 23313731

School of Computing
National College of Ireland

Supervisor: Eamon Nolan

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Hrushikesh Nitin Kumthekar
Student ID:	X23313731
Programme:	MSCDAD_C **Year:** 2024-25
Module:	MSC Research Practicum
Supervisor:	Eamon Nolan
Submission Due Date:	11-08-2025
Project Title:	Hybrid Audio Event Recognition Using CNN and Transfer Learning: Comparative Study on Custom and Google Speech Commands Datasets.
Word Count:	**773**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Hrushikesh Nitin Kumthekar

Date: 11-08-2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

1. INTRODUCTION

This document provides the configuration and step-by-step setup required to replicate the Keyword Spotting (KWS) framework developed to identify gunshots and distress words from audio input. The system integrates multiple deep learning models, including a custom CNN trained from scratch and transfer learning approaches (YAMNet + Logistic Regression/SVM/Dense NN), with a real-time graphical user interface (GUI) for detection.

2. HARDWARE REQUIREMENTS

2.1 Local Machine

- OS: Windows 10/11 (64-bit)
- Processor: Intel Core i5 (8th Gen or higher)
- RAM: Minimum 8 GB
- Storage: 20 GB free space (for datasets, models, and logs)
- Audio: Functional microphone for live detection

2.2 GPU Configuration (Optional)

- GPU: NVIDIA GPU with CUDA support (e.g., GTX 1650 or higher)
- Drivers: Latest NVIDIA CUDA Toolkit & cuDNN installed
- GPU recommended for faster training but not mandatory for inference.

3. SOFTWARE REQUIREMENTS

- **Python:** 3.8–3.10
- **IDE/Tools:**
  - Jupyter Notebook (via Anaconda) – for training and testing
  - Visual Studio Code – for code editing and debugging
  - PyInstaller (optional) – for packaging GUI application
- **Audio Tools:** sounddevice, librosa
- **GUI Toolkit:** tkinter
- **Machine Learning Libraries:**
  - tensorflow, tensorflow-hub
  - scikit-learn
  - numpy, pandas, matplotlib, seaborn
  - audiomentations (for data augmentation)

4. PACKAGE INSTALLATION

Run below command into dedicated anaconda environment in Anaconda Prompt where you want to execute project into.

```
pip install tensorflow tensorflow-hub scikit-learn librosa sounddevice audiomentations matplotlib seaborn joblib
```

5. DATASET DESCRIPTION

- **Custom Dataset:** Contains audio clips (WAV format) for multiple classes:
    - **Positive classes:** gunshot sounds, distress phrases
    - **Negative classes:** ambient noises, other non-critical sounds
- **Structure:**

bash
CopyEdit
audio-data/
  gunshot/
  help/
  noise/
  ...

- **Google Speech Commands Dataset (for testing baseline models)** – used in gui_google.py for generic KWS testing.

6. MODEL PREPARATION

6.1 Custom CNN Model (Spectrogram-based)

- Converts each audio clip into a 64-band mel-spectrogram.
- Architecture:
    - 3 convolutional layers + batch normalization + max pooling
    - Dense layer (128 neurons) + softmax output
- Trained using Adam optimizer and early stopping.
- Saved as: custom_model.keras
- Labels saved as: custom_labels.joblib

6.2 YAMNet Transfer Learning Models

- Extracts 1024-D embeddings from pre-trained YAMNet model.
- Classifiers: Logistic Regression, SVM, Dense NN, Small Dense NN.
- Saved as:
    - yamnet_logreg.joblib
    - yamnet_svm.joblib
    - yamnet_dense.keras
    - yamnet_small_dense.keras

7. GUI IMPLEMENTATION

7.1 Multi-Model GUI (multi_model_gui.py)

- **Features:**
    - Select model from dropdown (CNN, YAMNet+LR, YAMNet+SVM, YAMNet+Dense, YAMNet+Small Dense)
    - Live audio capture in 1-second chunks
    - Displays detections with timestamp, model used, predicted keyword, and confidence score
    - Configurable confidence threshold and debounce time

7.2 Google Speech Commands GUI (gui_google.py)

- **Purpose:** Baseline demonstration using Google Speech Commands dataset model.
- Detects predefined keywords like "yes", "no", "stop", "go".

8. STEPS TO RUN THE PROJECT

1. **Clone or copy project files** to your machine.
2. **Install required packages** as in Section 4.
3. **Ensure model files** (custom_model.keras, custom_labels.joblib, etc.) are in the same folder as the GUI scripts.
4. **Run Multi-Model GUI:**

bash
CopyEdit
python multi_model_gui.py

   - Select desired model from dropdown.
   - Click **Start** to begin detection.

5. **Run Google Speech Commands GUI (optional):**

bash
CopyEdit
python gui_google.py

6. Speak or play audio — detections will be displayed in real time.

9. REFERENCES

[1] Python Software Foundation (2025) *Python Language Reference, version 3.x*. Available at: https://www.python.org/ (Accessed: 8 August 2025).

[2] TensorFlow (2025) *TensorFlow – An end-to-end open source machine learning platform*. Available at: https://www.tensorflow.org/ (Accessed: 8 August 2025).

[3] Keras (2025) *Keras: Deep Learning for humans*. Available at: https://keras.io/ (Accessed: 8 August 2025).

[4] TensorFlow Hub (2025) *Reusable machine learning modules*. Available at: https://www.tensorflow.org/hub (Accessed: 8 August 2025).

[5] librosa (2025) *Python library for audio and music analysis*. Available at: https://librosa.org/ (Accessed: 8 August 2025).

[6] NumPy (2025) *NumPy: The fundamental package for array computing with Python*. Available at: https://numpy.org/ (Accessed: 8 August 2025).

[7] pandas (2025) *pandas: Python Data Analysis Library*. Available at: https://pandas.pydata.org/ (Accessed: 8 August 2025).

[8] Matplotlib (2025) *Matplotlib: Visualization with Python*. Available at: https://matplotlib.org/ (Accessed: 8 August 2025).

[9] Seaborn (2025) *Seaborn: Statistical data visualization*. Available at: https://seaborn.pydata.org/ (Accessed: 8 August 2025).

[10] scikit-learn (2025) *Machine Learning in Python*. Available at: https://scikit-learn.org/ (Accessed: 8 August 2025).

[11] SciPy (2025) *SciPy: Scientific computing tools for Python*. Available at: https://scipy.org/ (Accessed: 8 August 2025).

[12] joblib (2025) *joblib: Python serialization and pipelining*. Available at: https://joblib.readthedocs.io/ (Accessed: 8 August 2025).

[13] sounddevice (2025) *sounddevice: Play and record sound with Python*. Available at: https://python-sounddevice.readthedocs.io/ (Accessed: 8 August 2025).

[14] audiomentations (2025) *audiomentations: A Python library for audio data augmentation*. Available at: https://github.com/iver56/audiomentations (Accessed: 8 August 2025).

[15] tkinter (2025) *Tkinter: Python's de-facto standard GUI package*. Available at: https://docs.python.org/3/library/tkinter.html (Accessed: 8 August 2025).

[16] Google Speech Commands Dataset (2025) *Speech Commands Dataset*. Available at: https://www.tensorflow.org/datasets/catalog/speech_commands (Accessed: 8 August 2025).

[17] AudioSet/YAMNet (2025) *YAMNet model and AudioSet dataset*. Available at: https://tfhub.dev/google/yamnet/1 (Accessed: 8 August 2025).