

A2: Transfer Learning

Machine Learning and Sensing | Spring 2025

Assignment Overview

Goal: Explore transfer learning for acoustic activity recognition using Ubicoustics

- Evaluate the pretrained model on your dataset from Assignment 1
- Fine-tune the model using your own collected data
- Test performance on a holdout dataset
- Compare with best model from A1 (traditional ML)

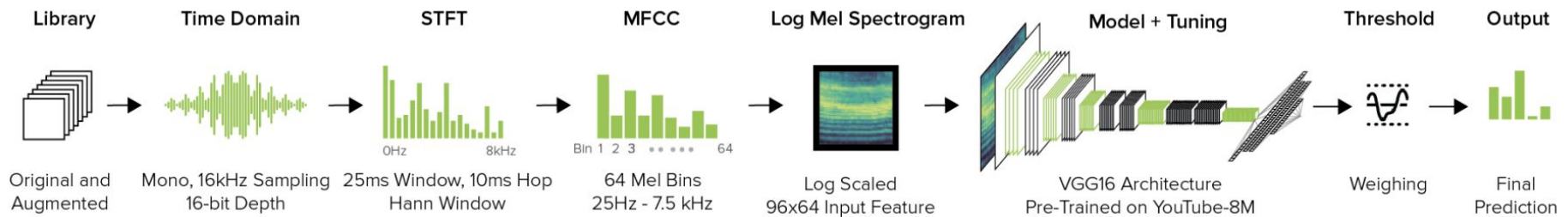
Submit:

- Colab Notebook (code executed sequentially)
- PDF Report

A2.0 – Ubicoustics Model

Objective: load the pretrained model, adjust the data mapping.

Link: <https://github.com/FIGLAB/ubicoustics>



A deep learning model (realtime and offline) detecting acoustic events in various environments based on Google's VGG- architecture.

Steps:

1. Install dependencies and [download the model](#)
2. run example scripts (optional) `$ python example_fileprediction_simple.py`
3. see notebook for more details

A2.1 – Evaluating Pre-trained Ubicoustics (5 pts)

Objective: Use pre-trained model to classify your A1 data

Steps:

- Map your 5 activities to Ubicoustics classes
- Filter output to 5 relevant classes
- Model inferencing

Report:

- Accuracy
- Confusion matrix
- Performance comparison: small vs. large room

Expected Output:

- [code] model loading, class mapping, inferencing
- [report] accuracy + confusion matrix

A2.2 – Fine-tuning with Environment-based Cross-validation (5 points)

Objective: Fine-tune Ubioustics on one environment, test on the other (data collected in A1).

Experiment Settings:

- Learning Rates: 0.01, 0.001, 0.0001
- Epochs: 10, 20, 30
- Layer Freezing: try multiple strategies

Repeat for:

- Small → Large Room
- Large → Small Room

Expected Output:

- [code] hyperparameters fine-tuning
- [report] accuracy + confusion matrix (comparison to A2.1 results)

A2.3 – Evaluate on Holdout Dataset (10 pts)

Objective: Use the best model (hyperparameters) you selected in A2.2. Then, evaluate the model on the holdout dataset provided on Dropbox (from A1).

Optimization (things to try):

- Windowing strategies
- Noise reduction
- etc

Expected Output:

- [code] evaluation on holdout dataset
- [report] final accuracy + confusion matrix
- [report] Summary of data processing, parameters fine-tuning
- [report] Justification for model selection

Summary: Performance Comparison & Recommendations

Things to deliver:

- A2.1 (pretrained)
- A2.2 (fine-tuned)
- A2.3 (final + holdout)
- Comparison to A1 (best traditional ML model)

Additional to hyperparameters fine-tuning (A 2.3):

- Windowing strategies
- Noise reduction
- Using only frames with detected sound
- etc

Note (for best model): use A2.1 if it outperforms fine-tuned models

Submission Checklist

- ☒ Colab notebook (executed sequentially)
- ☒ PDF report

Tips & Common Pitfalls

Correct model loading and class mapping

Do not train/fine-tune on hold-out dataset

Match plot labels/legends to classifier names

Make sure the notebook ran sequentially