TASK 1: GRANULAR FINGERPRINTING FOR TEMPORAL ALIGNMENT OF ARU RECORDINGS

Technical Report

Aditya Bhattacharjee

Centre for Digital Music,
Queen Mary University of London
a.bhattacharjee@qmul.ac.uk

ABSTRACT

This report presents a submission to the BioDCASE 2025 challenge task on temporal alignment of recordings from autonomous recording units (ARUs). We approach the problem as one of granular fingerprinting, learning invariant audio embeddings at a fine temporal resolution. Our method leverages a self-supervised contrastive framework designed to capture alignment-robust features in short overlapping audio segments across asynchronous sensor recordings. The contrastive setup is trained using the FSD50K dataset with artificial mixtures of "noisy" data points. The alignment is achieved in a zero-shot fashion by inferring the keypoints using a combination of cosine similarity-based lag calculation and linear regression.

***Index Terms*—** BioDCASE, multi-channel alignment, self-supervised learning, granular audio fingerprinting, ARU

1. INTRODUCTION

Temporal alignment of spatially distributed autonomous recording units (ARUs) is a critical prerequisite for many bioacoustic applications, including source localization, spatial filtering, and multi-sensor event detection. In field deployments, ARUs often operate independently without synchronization, resulting in temporal drift or offset due to unsynchronized clocks and varying recording start times. This desynchronization poses a significant challenge for downstream analysis, especially when recordings contain overlapping but misaligned acoustic events.

The BioDCASE 2025 Challenge Task 1 addresses this issue by benchmarking methods that estimate temporal alignment between such ARU recording pairs. Traditional solutions rely on signal-domain synchronization techniques such as waveform cross-correlation, which are often brittle in noisy, real-world conditions. Instead, we reframe the problem as one of granular audio fingerprinting.

Rather than attempting to align raw waveforms directly, we propose extracting a dense sequence of short-duration audio features, and learning invariant embeddings that are robust to domain-specific transformations such as variable source gains, background noise and partial content mismatch. By comparing these embeddings across recordings, we can estimate alignment through similarity-based matching, implicitly capturing the time offset without requiring any supervised alignment labels.

I want to thank my supervisor, Emmanouil Benetos, for his support and valuable feedback during this project

The Neural encoder architecture used is based on *GraFPrint* [1], a lightweight granular audio fingerprinting framework. We adapt this framework to the temporal alignment context by increasing the temporal resolution and training on synthetic mixtures that mimic time-shifted, noisy capture. This enables alignment in a zero-shot setting, directly comparing representations across recordings without needing task-specific fine-tuning.

2. SELF-SUPERVISED LEARNING FRAMEWORK

2.1. Input Features

During the training phase, we compute log-mel spectrograms from 1-second audio segments that are randomly sampled from the training data. This represents the window that is treated as an atomic unit for learning audio fingerprint representations.

2.2. Contrastive Objective

Our training objective is a contrastive learning task, where the goal is to learn audio embeddings that are invariant to gain perturbations, partial occlusions, and minor timing jitter. These conditions are frequently encountered in recordings from spatially distributed ARUs.

Each training sample is constructed as a synthetic mixture:

$$x = \sum_{i=1}^{K} g_i \cdot s_i, \tag{1}$$

where s_i denotes a randomly selected audio clip from the dataset, and $g_i \sim \mathcal{U}(g_{\min}, g_{\max})$ is a randomly sampled gain factor. This mixture x is an artificially created mixture of sources that serves as the *anchor* in the contrastive pair.

To create a *positive example*, we generate a second mixture using the same set of sources $\{s_i\}$ but with a different set of gain factors. Additionally, we introduce two key modifications:

- A random subset of sources $k_0 < K$ is muted (i.e., set $g_i = 0$) to simulate spatial dropouts; sources that may be present in one ARU but too faint or absent in another.
- Each source is subjected to a small, independent temporal shift $\delta_i \sim \mathcal{U}(-\epsilon, \epsilon)$. This jitter is chosen to be smaller than the feature hop size h_z [1], ensuring that positive pairs remain aligned at the feature level while introducing realistic variation.

[1] distinct from the spectrogram hop length

In addition to the above transforms, a randomly parameterized data augmentation pipeline is optionally applied to the data pair. Further details about this can be found in Section 4.

Both the anchor and the positive example are then passed through the same encoder to produce embeddings z and z^+, respectively. Negative samples are drawn from the rest of the batch, following the standard in-batch sampling strategy commonly used in contrastive learning [2]. A Normalized Temperature Cross-entropy (NTXent) function is used as the contrastive training objective. This formulation allows the model to learn representations that are both locally invariant and discriminative across acoustic content.

2.3. Encoder Architecture

In order to motivate the utility of audio fingerprinting systems for alignment, we adopt the encoder architecture from GraFPrint [1], which has demonstrated state-of-the-art performance in high-specific audio identification. The encoder is a graph neural network (GNN) applied to a dynamic k-nearest neighbour graph constructed over the time-frequency bins of the log-mel spectrogram. Each spectrogram frame is treated as a node, with edges connecting spectrally or temporally similar frames. We use the GraFPrint architecture without modification. For full architectural details and training configurations, we refer the reader to the original paper.

3. TEMPORAL ALIGNMENT METHOD

The goal of this method is to temporally align the embeddings (or fingerprints) generated by a pre-trained encoder from two audio channels. We describe two alignment methods: a constant-lag estimator based on global similarity maximization, and a drift-aware estimator that models slow temporal deviations over time. Both methods rely on the invariance properties learned through contrastive training; specifically, that corresponding segments across channels should yield embeddings with high cosine similarity. Thus, temporal alignment is reframed as a similarity maximization task over time-shifted embedding sequences.

3.1. Input Features

We evaluate our alignment framework on the development and evaluation sets provided as a part of the challenge. As detailed in Section 4, the data consists of dual-channel recordings with unsynchronized audio streams captured independently by separate devices. From each audio, we extract overlapping 1-second-long segments that are converted into log-melspectrograms. The feature hop size h_z is adjusted in different model submissions to emulate different feature rates, as described in Section 6. From the overlapping audio segments, we extract a time series of 128-dimensional embeddings using the trained encoder model.

3.2. Similarity-Based Lag Estimation

To compute the temporal offset between two audio channels, we perform a cross-similarity analysis between their feature sequences. For each candidate lag within a specified range, we align the two embedding sequences and compute the average framewise similarity using a dot product:

$$\text{score}(l) = \frac{1}{N_l} \sum_{i=1}^{N_l} \langle F_0^{(i)}, F_1^{(i+l)} \rangle \tag{2}$$

where $F_0^{(i)}$ and $F_1^{(i+l)}$ are the i-th frame embeddings from the two sequences, aligned with lag l, and N_l is the number of overlapping frames at that lag. The lag with the highest similarity score is selected as the optimal alignment offset. This basic approach assumes a constant time shift between the recordings. We observe that this simplifying assumption can lead to a more optimal temporal alignment in the presence of small clock drift.

3.3. Drift-Aware Alignment via Local Lag Regression

In practice, clock drift between devices can result in a non-constant temporal offset over time. To address this, we extend the lag estimation to model slow, continuous drift. Specifically, we estimate local lags around each keypoint by applying the similarity-based lag computation within a temporal window. This yields a set of timestamped local lag estimates.

To ensure robustness against noise and local mismatches, we compute the median of the local lags as a global anchor lag. The residual drift, defined as the deviation of each local lag from this global baseline, is then modelled using Huber regression [3]. The final predicted alignment at time t is given by:

$$k_1(t) = t + \underbrace{L_{\text{global}} \cdot \delta}_{\text{constant offset}} + \underbrace{\Delta(t)}_{\text{clock drift correction}} \tag{3}$$

where L_{global} is the median lag (in frames) and δ is the hop duration in seconds.

While this method estimates a linear clock drift using robust regression, we observe that a simpler global lag estimation can often yield more accurate alignments, particularly when the actual drift is small. In practice, when the estimated drift slope exceeds a small threshold (e.g., ± 0.05 seconds per second), it often reflects overfitting to noisy local lag estimates rather than true temporal deviation. To safeguard against such cases, we incorporate a fallback mechanism: if the estimated drift slope is deemed too large, the system reverts to using the global lag only, discarding the linear drift correction. This fallback ensures robustness and avoids catastrophic alignment errors in low-drift scenarios.

4. DATASET

We train our fingerprinting model using audio from the FSD50K dataset [4], a large-scale collection of sound events sourced from Freesound, encompassing over 51,000 audio clips across a wide variety of everyday sound classes [2]. The dataset includes both monophonic and polyphonic recordings, making it well-suited for learning robust, content-based audio representations that generalize across overlapping sources.

To simulate temporal and environmental variability during training, we apply a range of data augmentations designed to model transformations encountered in ARU recordings:

- **Colored noise mixing:** additive pink or brown noise at varying SNRs
- **Time-stretching:** to emulate small clock drift upt $\pm 5\%$.
- **Resampling jitter:** slight random changes in playback speed (e.g., $\pm$1–3%). This is an alternate strategy to mimic the clock drift.
- **Tanh distortion:** soft-clipping of waveform amplitude to simulate nonlinear microphone response.

[2] This is an external data resource

Since we do not use the training split of the provided dataset for training, the validation is performed on both the train and validation split of the `aru` and `zebra_finch` datasets. As a part of this submission, we also provide the inferred keypoints on the `eval` split of both datasets.

5. SUBMISSION DETAILS

We submit four variants of the alignment model, each differing slightly in temporal resolution, data configuration or alignment approaches. Common training hyperparameters are listed in Table 1, while Table 2 summarizes the differences across submissions.

Parameter	Value
Sample Rate	16kHz
Audio segment	1 sec
Batch size	256
Embedding dimension	128
Mel bins	64
Hop length	512
Window length	1024
Gain $(g_{\min}, g_{\max})$	$(0.2, 5.0)$
Shift ϵ	50ms

Table 1: Common hyperparameters used across all submissions.

Parameter	#1	#2	#3	#4[3]
Feature hop h_z	100 ms	100 ms	100 ms	50 ms
Time-stretch	Yes	Yes	Yes	No
Resample jitter	No	No	No	Yes
Color noise mix	No	No	No	Yes
Median lag est.	No	Yes	Yes	No
Lag window	–	20 s	20 s	–
Lag regression	–	Yes	No	–

Table 2: Submission-specific hyperparameter variations across model configurations.

All models were trained using a single NVIDIA A100 GPU. We used the Adam optimizer with a cosine annealing learning rate scheduler, with a maximum learning rate of 8*e*-5. We employ early stopping based on the validation set performance.

One major distinction between the submitted models lies in the temporal alignment methodology. Model #1 and #4 (excluded from submission) employ the global lag estimation described in Section 3.2. Model #2 uses the drift-aware alignment described in Section 3.3. In Model #3, we simplify the setup by getting rid of the regression-based correction and using just the median lag as the predicted offset.

6. RESULTS AND DISCUSSION

The results in Table 3 show that, among the submitted models, Model #3 achieves the lowest MSE on the *aru* dataset (0.258), despite using only the median lag without any regression. This indicates that the simplifying assumption that considers the clock drift to be negligible, leads to a better performance.

Method	*aru*	*zebra_finch*
`crosscor`	6.861	10.029
`deeplearning`	0.516	**1.262**
Model #1	0.391	10.200
Model #2	0.663	5.263
Model #3	**0.258**	5.739
Model #4[3]	1.555	2.635

Table 3: Mean squared error (MSE) of alignment across baselines and submitted models on the development and evaluation sets.

Model #2, which uses drift-aware alignment via regression, does not exhibit any significant improvement in performance, suggesting that explicit modelling of drift does not consistently lead to better alignment. Model #1, based on global lag estimation, performs well on the *aru* dataset (0.391) but fails to generalize to the *zebra_finch* recordings (10.200).

Model #4 [3] was trained with a larger variety of data augmentations and shows the most balanced performance across both datasets, albeit with higher overall MSE.

7. CONCLUSION

In this work, we presented a granular fingerprinting approach for temporal alignment of asynchronous ARU recordings, submitted as part of the BioDCASE 2025 Challenge Task 1. Our method reframes alignment as a contrastive similarity task, using embeddings learned via self-supervised training on synthetic audio mixtures from FSD50K.

We adopted the GraFPrint encoder, which applies a graph neural network to time-frequency spectrogram graphs, and demonstrated that this architecture when trained with a contrastive learning methodology. generalizes well to multichannel alignment tasks. Our experiments compared multiple alignment strategies, including constant lag estimation and drift-aware regression.

Results indicate that simple global lag estimation consistently outperforms more complex drift modelling, especially when clock offsets are small. In particular, Model #3 achieved the best performance on the ARU dataset, highlighting that overfitting to noisy local lag estimates can degrade accuracy.

8. REFERENCES

[1] A. Bhattacharjee, S. Singh, and E. Benetos, "Grafprint: A gnn-based approach for audio identification," in *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2025, pp. 1–5.

[2] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PmLR, 2020, pp. 1597–1607.

[3] P. J. Huber, "Robust estimation of a location parameter," in *Breakthroughs in statistics: Methodology and distribution*. Springer, 1992, pp. 492–518.

[3] Based on the special requirements of Task 1, this model has been excluded from the submission. However, we have maintained the above notes for the sake of completeness.

[4] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, "Fsd50k: An open dataset of human-labeled sound events," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 306–310.