

EnvId: A Metric Learning Approach for Forensic Few-Shot Identification of Unseen Environments

Denise Moussa*, Germans Hirsch*, Christian Riess
Friedrich-Alexander-Universität Erlangen-Nürnberg
 {denise.moussa, christian.riess}@fau.de

Abstract—Audio recordings may provide important evidence in criminal investigations. One such case is the forensic association of a recorded audio to its recording location. For example, a voice message may be the only investigative cue to narrow down the candidate sites for a crime. Up to now, several works provide supervised classification tools for closed-set recording environment identification under relatively clean recording conditions. However, in forensic investigations, the candidate locations are case-specific. Thus, supervised learning techniques are not applicable without retraining a classifier on a sufficient amount of training samples for each case and respective candidate set. In addition, a forensic tool has to deal with audio material from uncontrolled sources with variable properties and quality. In this work, we therefore attempt a major step towards practical forensic application scenarios. We propose a representation learning framework called EnvId, short for environment identification. EnvId avoids case-specific retraining by modeling the task as a few-shot classification problem. We demonstrate that EnvId can handle forensically challenging material. It provides good quality predictions even under unseen signal degradations, out-of-distribution reverberation characteristics or recording position mismatches.¹

Index Terms—audio forensics, representation learning, environment identification.

I. INTRODUCTION

Inferring the recording location from some audio material can be a highly relevant audio forensic task during criminal investigations. The analysis of audio material can directly contribute to evidence or at least provide investigative cues for the reconstruction of crime scenes and the course of events in a case [51].

A reverberant environment, which we also refer to as location, may leave characteristic reverberation traces in audio recordings. Reverberation, most generally described, results from reflections of sound waves in an environment. For example, when a recording is made of a sound that originates next to a wall (consider, *e.g.*, the breaking of a glass), then the microphone records the primary sound that travels directly from the origin to the microphone, and a reflection of that sound from the wall. In principle, any obstacle in the environment that interferes with sound waves, *i.e.*, in terms of absorption, transmission and reflection properties, contributes to its reverberation signature. A simple computational model for reverberation is the convolution of a signal with an acoustic impulse response (AIR).

Consequently, to identify a certain location, *e.g.*, a specific room in an apartment, one needs to invert the mapping of recorded reverberation traces to the underlying geometry and overall composition of the respective environment. This mapping is not unique, but from a practical point of view it is oftentimes sufficiently distinctive to support or reject hypotheses that a recording has been made in one out of a set of specific locations. Note that this task is different from acoustic scene classification, where the goal is to differentiate between certain types of environments like *public street* or *train station* that share similar background noise profiles [1], [13], [26].

So far, several analytic and deep learning (DL) methods have been proposed that exploit reverberation cues to characterize and/or identify a recording location. We discuss existing methods and their limitations w.r.t. forensic environment identification in Sec. I-A and Sec. I-B and describe our contribution to address these constraints in Sec. I-C.

A. Existing Methods

Many works address environment identification as a closed-set classification task in a supervised training paradigm [2], [4], [28]–[30], [34], [35], [37]. Such methods are able to identify a recording location that stems from a known and fixed set of candidate environments.

Some proposed approaches include traditional classifiers like Gaussian Mixture Models (GMMs) or Support Vector Machines (SVMs) trained on hand-crafted features that serve as acoustic fingerprints for the respective environments [4], [28]–[30], [34], [37]. Analytic features, however, oftentimes reach their limits in modeling the complexity of real-world audio conditions, such that DL tools have been increasingly explored for various audio forensic tasks [3], [6], [31], [48], including environment identification [2], [35]. Here, Papayianis *et al.* [35] show the superiority of deep convolutional recurrent neural network (CRNN) features as opposed to custom, analytic features. Additionally, Azimi *et al.* [2] use the deep features of pre-trained neural networks (NNs) for speaker identification as input to an SVM.

Other works focus on the estimation of specific environment parameters from single channel audio samples [8], [11], [12], [14]–[16], [19], [27]. Moore *et al.* [27] use analytical models to estimate the geometry of 2-D rectangular rooms while more recent approaches directly regress characteristic environment parameters with NNs. For example, many works target the

* Authors contributed equally to this work.

¹code: <https://fau1-github.cs.fau.de/mmsec/few-shot-recording-environment-identification>

estimation of RT_{60} , *i.e.*, the time until the AIR energy decays by 60db, or the volume of some space [8], [12], [14]–[16], [19]. However, also other parameters were investigated, like C_{50} , *i.e.*, the ratio between the first 50 ms of a signal and its remaining late energy [11], [16].

B. Limitations of Existing Methods

For forensic purposes, existing methods for environment identification are still too constrained in their practical applicability.

First, the proposed supervised closed-set classifiers are natural models for, *e.g.*, smart home environments, where it is possible to train on the same recording locations that are queried at test time [2], [35]. However, the forensic task of environment identification is by design a task that has to deal with unknown recording locations at test time. After all, the set of potential recording locations depends on the current case at hand. As a consequence, using supervised classifiers would require to collect extensive training data on a case-by-case basis which is expensive and impractical. In addition, the audio signal may stem from none of the potential candidates, such that open-set instead of closed-set classifiers are needed to reject a sample that does not belong to any class.

Second, location identification from estimated environment parameters also has some pitfalls for forensic applications. It is by design a two-stage process, and as such relatively prone to errors: the parameters themselves are only estimates and subject to possible confusions, overlap, or just estimation errors, and a subsequent environment identification may add further estimation errors.

Third, the robustness to signal degradations is an underexplored topic in research on environment identification. Several works experiment on quasi clean audio signals [4], [27], [29], [34], [35], [37]. Others simulate noise degradations, but use identical distributions in both the training and the test sets [2], [8], [11], [12], [14], [19], [30].

C. Contributions

With this work, we propose to overcome these limitations with several measures to take a further step towards robust forensic environment identification.

First, we approach the task as a few-shot classification task. This allows for the identification of recording environments by collecting only few reference audio samples from a set of case-dependent location candidates. Few-shot learning has already been successfully explored for audio classification to distinguish between signals of different content like, *e.g.*, background music, bird voices or human speech [18], [46], [47], [50], but not for the identification of audio recording environments.

Second, while we put our primary focus on few-shot learning to directly identify locations, we demonstrate that our framework is capable of environment parameter estimation. Despite the limits of using environmental parameter estimates for the identification task, they can still unveil important cues about a location. For example, estimated parameters can support or question the tool’s location prediction and thus assist in

the overall process. Also, the task may become relevant if the police does not have a hypothesis about candidate locations, but aims to characterize the recording location for further investigations, such as a coarse estimate of the volume of a room.

Third, we specifically set the focus on audio material in forensic investigations which is usually ‘in-the-wild’ data of uncontrolled characteristics. We thus evaluate under significant training and test set mismatches, as in practice, the audio signal might be heavily impacted by unknown degradation factors. We include varying noise distortions which can be expected from arbitrarily noisy environments or distortions introduced by low-cost acquisition devices. We also evaluate the practically important case of lossy audio compression and also re-compression, which both have not been addressed in previous works to our knowledge.

Additionally, we more closely analyze the impact of recording position mismatches for few-shot classification, since reverberation traces of a location also depend on the position of sound emitters and capturing devices. So far, some works on closed-set environment classification have evaluated training/test mismatches of microphone positions, however the actual positions are not documented [4], [30]. Here, we take a further step forward and give first insights into the influence of specific microphone/speaker positions on a custom dataset of synthetic four-sided rooms, where we differentiate between corridor, rectangular and square shaped room categories.

We denote the proposed framework as “EnvId”, as an abbreviation for “environment identification”. We hope that EnvId will set a new standard for environment identification from data in the wild, and set a baseline for further research in this direction.

The remainder of the paper is organized as follows. Sec. II explains the technical background on reverberant environments and Prototypical Networks [41]. In Sec. III, we present the proposed EnvId framework and the data generation pipeline for simulating challenging forensic scenarios. Sec. IV contains the experimental evaluation of EnvId and Sec. V concludes the work.

II. TECHNICAL BACKGROUND

Details on the acoustic modeling of reverberant in-the-wild audio signals are provided in Sec. II-A, and the concept of Prototypical Networks is presented in Sec. II-B.

A. Acoustic Model for Reverberant in-the-Wild Speech

Consider an audio signal $a(t)$ that propagates from an emitting source into an environment. During propagation, $a(t)$ can be reflected by various obstacles of different properties that influence the reflection behavior, like the orientation, shape or material of some object. Such reflections cause the signal to arrive at slightly different times at a recording microphone, similar to closely overlapping echos. This phenomenon is commonly referred to as reverberation and the reverberation signature of an environment can be measured as an AIR

signal $r(t)$. A reverberant audio signal $s(t)$ is then commonly modeled as convolution

$$s(t) = r(t) * a(t) \quad (1)$$

of an (ideally anechoic) audio signal $a(t)$ with the captured AIR $r(t)$, which is the de-facto standard model for reverberation [2], [4], [29], [30], [34], [35], [37]. Hence, one can separately record reverberation-free speech signals and AIRs, and freely combine those signals to simulate reverberant speech from various environments.

Additionally, an environment usually exhibits some kind of background noise, *e.g.*, from nature or urban surroundings. Also, microphones can introduce noise during the signal capturing process. Such noise is commonly modeled as an additive signal $n(t)$ scaled by scalar noise impact α [2], [8], [11], [12], [14], [19], [30]. The acoustic model can therefore be extended to

$$\tilde{s}(t) = r(t) * a(t) + \alpha \cdot n(t) . \quad (2)$$

Moreover, lossy compression is a recurring challenge for forensic analysts. Lossy compression might be performed directly by the capturing device, and/or later upon sharing content over the internet, as well as by editing and re-saving operations. For this reason, we further extend the signal formation to

$$\tilde{s}(t) = (f_1^C \circ \dots \circ f_N^C)(r(t) * a(t) + \alpha \cdot n(t)) , \quad (3)$$

where f_n^C with $n \in [1, N]$ describes one out of N lossy compression operations.

B. Prototypical Networks

Prototypical Networks [41] are a representation learning approach that enables few-shot classification, such that a classifier does not have to be retrained for classes that are not part of the training data. The approach has primarily been used for computer vision tasks, but some studies also demonstrate its effectiveness on audio signals [18], [46], [47], [50].

The idea of a Prototypical Network [41] is to learn a metric embedding space that generalizes to unseen classes. In that space, some class is represented by its prototype which is computed from few reference samples. Some input sample, the query, can then be assigned to a class by measuring the distance between its embedded representation to all classes' prototypes and choosing the closest one.

For the learning approach, the training is split into episodes that follow a N -way K -shot protocol. Thus, in each episode, a subset of $N \leq N_t$ classes is randomly sampled from the training set of N_t classes in total. For each class c in the subset, a support set \mathcal{S}_c of K reference samples $s_r \in \mathcal{S}_c$ is randomly chosen for prototyping, while the remaining samples serve as input queries $s_q \notin \mathcal{S}_c$. Per class, a prototype \mathbf{p}_c is computed as the average vector in the embedding space, formally

$$\mathbf{p}_c = \frac{1}{K} \sum_{\mathbf{x}_k \in \mathcal{S}_c} f(\mathbf{x}_k) , \text{ where} \quad (4a)$$

$$f : \mathbb{R}^D \rightarrow \mathbb{R}^E \quad (4b)$$

is a learnable function that projects the reference samples' feature vectors $\mathbf{x} \in \mathbb{R}^D$ to the dimension E of the metric space.

Then, the distance, typically the Euclidean distance, between each embedded query sample $\mathbf{q} \in \mathbb{R}^D$ and each prototype \mathbf{p}_c is computed as

$$d(\mathbf{p}_c, f(\mathbf{q})) = \|\mathbf{p}_c - f(\mathbf{q})\| . \quad (5)$$

The likelihood that the class label y of a query sample is c is further given by the softmax function over the distances to all classes' prototypes

$$p(y = c|\mathbf{q}) = \frac{\exp(d(\mathbf{p}_c, f(\mathbf{q})))}{\sum_{i=0}^{N-1} \exp(d(\mathbf{p}_i, f(\mathbf{q})))} . \quad (6)$$

The training loss is the negative log-likelihood

$$\mathcal{L}_{\text{class}} = -\log p(y = c|\mathbf{q}) , \quad (7)$$

which is minimized for the correct class c .

III. METHODS

The proposed framework for few-shot environment identification and parameter estimation is presented in Sec. III-A. The associated data generation pipeline is presented in Sec. III-B.

A. The EnvId Framework

EnvId is designed as an end-to-end trainable framework that handles joint few-shot location identification and environmental parameter regression.

First, at its core, EnvId implements the Prototypical Network approach (*cf.* Sec. II-B) to prevent case-specific retraining (*cf.* Sec. I-B). For our application, the goal is to learn a metric embedding space in which distances enable a distinction of recording locations. During inference, an audio sample in question (query) and K reference samples from each of N candidate locations can be projected to the metric space. The query can then be assigned to the closest location prototype via N -way K -shot classification.

Second, in case of criminal investigations, some given audio material might not stem from any of the candidate locations. Thus, EnvId supports open-set few-shot classification, *i.e.*, can reject the query as belonging to an *unknown* class (location), if the distance to all prototype candidates is too large.

Third, the query representations in the metric space are used to support environmental parameter estimation as a side task. The blind regression of scalar parameters can both act as an additional cue to the identification result and is still applicable if no candidate set of potential locations is available in a case.

An overview of the EnvId framework is shown in Fig. 1. In Sec. III-A1 – Sec. III-A5, the framework's individual components and inference process are presented in detail, and Sec. III-A6 explains the training procedure.

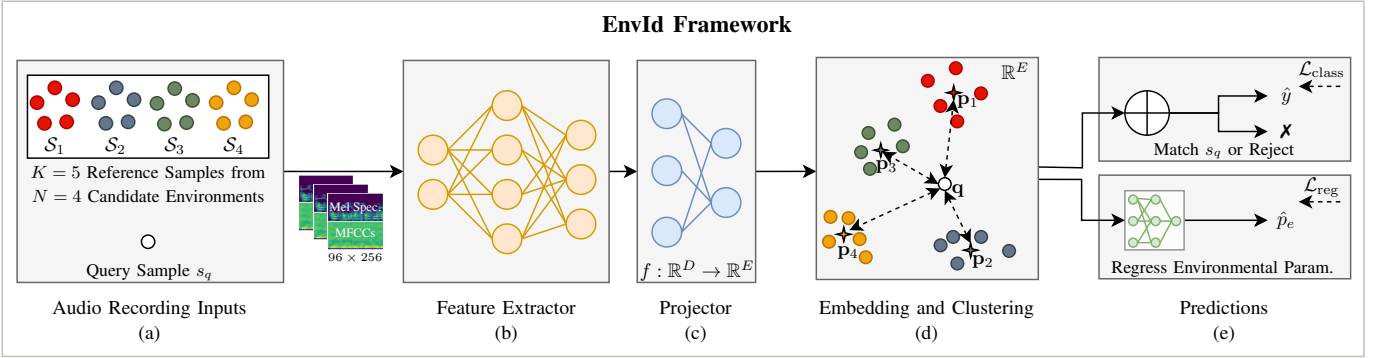


Fig. 1: Our end-to-end trainable EnvId framework for joint few-shot environment identification and blind parameter regression from audio recordings. The framework takes audio signals as inputs (a), and consists of a neural feature extractor (b) and projector (c) to process and map the input samples to the learnable, metric embedding space (d). The audio representations in the metric space can both be used for the identification of environments, and the regression of environmental parameters (e).

1) *Audio Recording Inputs (Fig. 1 (a))*: The input to EnvId consists of the query audio sample s_q to classify and a support set \mathcal{S}_c of K reference audio samples for each of N candidate locations. All audio signals are transformed to dense frequency representations, as it is commonly done in audio processing [3], [31], [35], [39]. In detail, we compute the Mel spectrogram [25] and the Mel frequency cepstral coefficients (MFCCs) [25] and concatenate the coefficients of both representations into one feature vector. The Mel spectrograms are computed with torchaudio [43] with a window size and FFT size of 1024 bins, a stride of 512, and 256 Mel filter banks. This results in a size of 96 for the time dimension and 256 for the frequency dimension. The MFCCs uses 20 coefficients and is computed from the spectrogram. Thus, the final concatenated feature input is of size 96×276 .

2) *Feature Extraction (Fig. 1 (b))*: Deep feature vectors of D dimensions are extracted from each input frequency representation. Here, we recommend a slim convolutional neural network (CNN) feature extractor that we refer to as Gamper*, *i.e.*, a variation of the CNN by Gamper *et al.* [12]: different from the original work, we use uniform 3×3 filter kernels in all layers instead of large kernel dimensions in the time domain. Gamper *et al.* [12] aim to capture mostly temporal information from the spectrogram input with a large receptive field in the time domain. However, we could not observe a performance gain for weighting temporal over spectral information on our task. Instead, uniform receptive fields on both dimensions showed to consistently improve the results.

Note that also any other custom end-to-end trainable feature extractor can be used for this stage, and we demonstrate the performances of various other networks in the associated evaluation in Sec. IV-D.

3) *Feature Projection (Fig. 1 (c))*: The feature projector f projects the deep features' dimension D to the metric embedding space's dimension E . For our studies, we use a simple fully connected layer for the mapping. While the embedding dimension can be varied, we use $E = 256$ throughout all experiment. In early studies with $2^6 \leq E \leq 2^9$, we observed that smaller E led to worse embedding space results while larger E did not show any advantage.

4) *Embedding and Prototype Computation (Fig. 1 (d))*: The projected features are embedded in the metric space. For each candidate location class c , a prototype \mathbf{p}_c is computed from its embedded support set (*cf.* Eq. 4). Then, the Euclidean distance from the embedded query to all prototypes is calculated (*cf.* Eq. 5).

5) *Location Classification and Parameter Regression (Fig. 1 (e))*: In the final step, the query sample is attributed to some location (Fig. 1 (e), top) and/or blind environmental parameter estimation is performed from its embedded representation (Fig. 1 (e), bottom).

During inference, a query can also be rejected if there is no prototype within a reasonably short Euclidean distance. Otherwise, the likelihood of the query sample belonging to a location is computed from the query-prototype distances in the metric space as defined in Eq. 6. The final classification is then performed by choosing the closest, *i.e.*, most likely location prototype as

$$\hat{y} = \underset{c}{\operatorname{argmax}}(p(y = c|\mathbf{q})) , \quad (8)$$

where \hat{y} is the predicted class label.

For the environmental parameter estimation side-task, a trainable function $g : \mathbb{R}^E \rightarrow \mathbb{R}$ regresses a scalar prediction \hat{p}_e from the embedded query \mathbf{q} , formally

$$\hat{p}_e = g(\mathbf{q}) , \quad (9)$$

where we use two linear layers with a hidden dimension of each 256 for $g(\cdot)$.

Note that during inference, it is possible to exclusively use EnvId for parameter regression if no set of candidate locations are available for few-shot identification in some forensic case. In our experiments in Sec. IV-I the regression task is evaluated for the environmental parameters volume and RT_{60} , *i.e.*, the time in which the signal energy decays by 60 dB.

6) *Training*: The EnvId framework is trained in episodes with a N -way K -shot protocol (*cf.* Sec. II-B), where the few-shot classification task is optimized using the negative log-likelihood loss $\mathcal{L}_{\text{class}}$ (Eq. 6). If environmental parameter regression is enabled, the metric space (Fig. 1 (d)) is optimized

to additionally represent scalar environmental characteristics in embedded samples with the regression loss

$$\mathcal{L}_{\text{reg}} = |p_e - \hat{p}_e|, \quad (10)$$

which penalizes the absolute deviation of the parameter label p_e from its regressed prediction \hat{p}_e . Naturally, training this side-task requires labeled audio samples with known environmental properties. For joint training, the total loss

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{class}} + \mathcal{L}_{\text{reg}} = -\log p(y = c|\mathbf{q}) + |p_e - \hat{p}_e| \quad (11)$$

is minimized.

B. Data Generation Framework

The quite unconstrained nature of in-the-wild data must be met with extensive training on diverse samples. Our data samples are hence created with a flexible generation pipeline that is built upon the acoustic model for reverberant speech in Sec. II-A. The proposed framework is visualized in Fig. 2.

In the first step (Fig. 2 (a)), clean reverberant speech samples are simulated from a set of AIRs $r(t)$ and a corpus of anechoic speech samples $s(t)$ according to Eq. 1.

In the second step, shown in Fig. 2(b), additive noise degradation is added using Eq. 2. This step is optional and can be skipped depending on the training or evaluation goal for EnvId. The operation requires a set of noise signals $n(t)$ and an interval of signal-to-noise ratios (SNRs) that controls the noise strength α . Per reverberant sample, the generator randomly chooses a noise configuration from those specifications.

In the third step, shown in Fig. 2(c), lossy compression is introduced as defined in Eq. 3. Correspondingly to noise degradation, this step is optional and the specific compression algorithm $f^{\mathcal{C}}(\cdot)$ with strength \mathcal{C} is randomly sampled from a customisable set for each incoming sample. Our pipeline natively supports the popular formats MP3, AMR-NB, GSM and Vorbis, and also the neural EnCodec [7] format. Re-compression is simulated by repeating this random compression process.

The pipeline is configured adaptively to generate data sets for our different training and evaluation scenarios presented in Sec. IV. The different pipeline settings are described in the respective experiment sections.

IV. EXPERIMENTS

The evaluation section is organized in nine parts. Initially, Sec. IV-A describes our overall setup for the experiments, Sec. IV-B gives an overview over the reported metrics and Sec. IV-C lists our sample sources for data set creation. The first of six experiments is described in Sec. IV-D which reports EnvId’s performance with different feature extractors for the comparison to related work. Sec. IV-E investigates open-set few-shot classification, and Sec. IV-F explores EnvId’s robustness towards unseen signal degradations. In Sec. IV-G, an ablation on the choice of K -shot for few-shot inference is provided. Sec. IV-H presents our study on the influence of specific recording positions in an environment. Finally, Sec. IV-I explores blind environmental parameter regression at the example of the RT_{60} and the volume.

A. Experimental Setup

We train EnvId with a 10-way 15-shot protocol in episodes (*cf.* Sec. II-B). Snell et al. [41] recommend to set N -way and K -shot to a value that is expected in practice during inference. Moreover, they report that training with a larger N can further increase the performance. Here, we see it as a practically feasible assumption that up to 15 reference recordings are collected from each of at maximum 10 candidate locations in a forensic case. During inference, we set $K = 15$ by default, but we also investigate the impact of collecting fewer K reference samples in Sec. IV-G, as collecting samples is a relevant cost factor in practice. Furthermore, results are reported for the most difficult case of $N = 10$ candidate classes, or the maximum number of available classes in case of data sets with fewer locations.

Additionally, note that our experiments exclusively evaluate few-shot classification with distinct locations in training and test. While the few-shot setting allows for classifying samples from both seen and unseen classes, our experiments only evaluate the identification of environments never seen in training. This challenging setting results from the nature of forensic investigations that, in practice, deal with highly case-specific candidate locations that are naturally unknown during the preceding training of the identification tool.

Training is conducted using the Adam [22] optimizer with learning rate $1e^{-4}$ for at maximum 300 epochs over the training set. Early stopping is applied if the validation accuracy for assigning audio samples to recording locations does not increase for 30 epochs.

Technically, the framework is implemented in PyTorch [36] v.1.10.2. Training is conducted on one consumer GPU, at most a RTX3090.

B. Classification Metrics

We report the accuracy, precision, recall and F1-score to evaluate EnvId’s environment identification capabilities.

The accuracy is reported over test sets of locations. It denotes the fraction of query audio input samples that are correctly appointed to their respective recording location. In some cases, the Top- n accuracies with $n \in [1, 3]$ are reported which includes softer accuracy scores where a prediction is counted as correct if the correct location is among the first n predictions.

The F1-score, precision and recall are reported for individual locations to analyze EnvId’s performance for different types of environments more closely. As by default, per location, the precision (P), *i.e.*, specificity, is computed by $\frac{T_p}{T_p + F_p}$, where T_p and F_p denote the true positives and false positives, respectively. The metric thus reports the correct proportion of all query samples that EnvId appoints to the respective location. Accordingly, the recall (R), *i.e.*, sensitivity per location is computed by $\frac{T_p}{T_p + F_n}$, where F_n denotes the false negatives. This defines EnvId’s correctly identified proportion of all query samples from the respective environment. The F1-score per location summarizes both capabilities in one score, defined as $\frac{2 \cdot P \cdot R}{P + R}$.

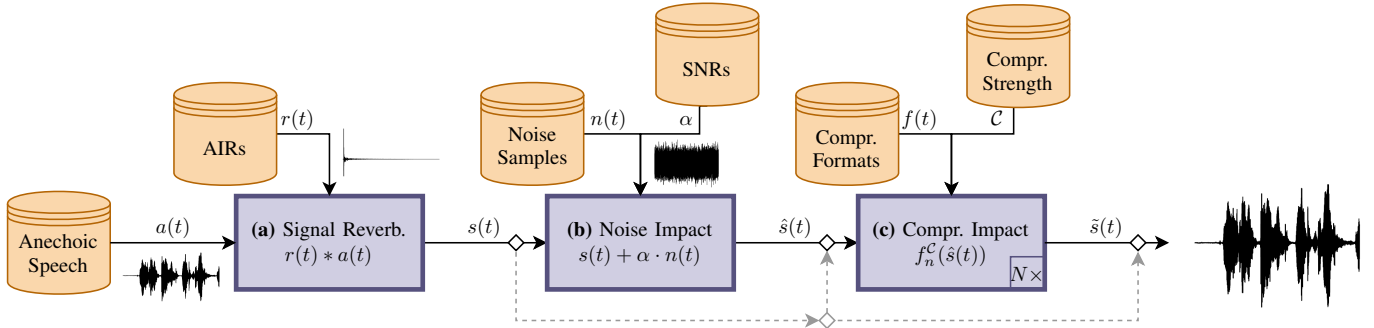


Fig. 2: Proposed pipeline for controlled simulation of real world audio recording and post-processing scenarios. Configurable sets of input signals, environments and degradations (orange) enable the creation of custom test cases. In 3 steps (purple), anechoic audio signals $a(t)$ pass various transformations and are output in frequency representation. Dashed arrows indicate skip connections to (randomly) enable and disable degradation transformations per sample.

TABLE I: Composition of the source data pools of anechoic speech samples and AIRs for data set generation.

Data Pool	Speech $a(t)$	AIR $r(t)$
	Source: # Samples	Source: # Samples
Training	ACE [10] : 239	MIT [44] : 200
Validation	ACE [10] : 64	MIT [44] : 20
Test	TSP [21] : 67	ACE [10] : 7 AAIR+REV [20], [23] : 13 MIT [44] : 20 OPENAIR [32] : 20

C. Speech and AIR Resources for Data Set Generation

For the simulation of reverberant speech (*cf.* Sec. III-B), anechoic single channel audio signals are collected from two freely available databases and AIRs are collected from a total of five. The AIR resources contain real AIR measurements and high-quality AIR simulations that represent a wide range of recording environments. Tab. I summarizes the source data pools that are further described below. Note that we additionally use custom synthetic AIRs to perform some evaluations under controlled settings. Those are later described in Sec. IV-H and Sec. IV-I.

1) *Anechoic Speech Pools.*: Anechoic speech signals $a(t)$ are taken from the ACE [10] corpus for training/validation and from the TSP [21] corpus for testing. From the ACE set [10], we use 33 speech samples from 4 female speakers and from 7 male speakers with a duration between 3 s and 97 s. From TSP, we use 87 utterances from 2 female speakers and from 2 male speakers with a duration between 1.70 s and 3.25 s. We train and test EnvId on audio snippets of 3 s, so, longer speech samples are split into 3 s segments. TSP also contains shorter samples, which are enlarged by concatenating speech from the same speaker at silent positions. This results in a total of 239/64 training/validation samples from the ACE [10] corpus, and 67 clean voice snippets from TSP for testing as shown in the left column of Tab. I.

2) *AIR Pools.*: The training/validation pools of AIR signals $r(t)$ consist of real 200/20 AIRs from the MIT Acoustical Reverberation Scene Statistics Survey [44] (MIT) data set. The MIT set provides diverse space categories. Small categories

are, *e.g.*, *cars* and *bathroom*, mid-size categories are, *e.g.*, *bar*, *train*, and *hallway*, and large categories are, *e.g.*, *theatre*, *atrium*, or *open air*. For testing we use four separate AIR pools from the remaining MIT AIRs and from smaller databases that are described below.

The ACE test pool consists of the ACE AIR set of 7 rooms that is oftentimes used in related work [2], [4], [8], [11], [12], [14], [15], [19], [30], [35]. The rooms are enclosed mid-size spaces of the category *lecture room*, *meeting room*, *office room* and *lobby*, with volumes between 47.3 m^3 and 370 m^3 .

The AAIR+REV test pool is created by combining the 7 real AIR measurements from the Aachen Impulse Response Database (AAIR) [20] with 6 high-quality simulations from the 2014 REVERB challenge (REV) [23]. AAIR provides measurements from a low-reverberant studio booth, a stairway, an office, a meeting room, two lecture rooms, and a church. It features diverse surface materials and furniture. The room volumes range between 11.9 m^3 and 370.8 m^3 , except for the church where only the floor area is reported with 570 m^2 . To enlarge the data set, we sample 6 high-quality simulations from the REV set following Kinoshita *et al.* [23], where 2 rooms are from each category *small*, *medium* and *large*.

The MIT and OPENAIR test pools contain 20 AIR measurements each. The MIT [44] database provides AIRs from variable space categories while the OPENAIR [32] data mainly covers large spaces like churches, halls, auditoriums and open spaces, where the volumes (excluding open air) range between 35.2 m^3 and 140000 m^3 .

For more details, we list the specific selected environments later in Tab. III for the OPENAIR pool. Analogously, the selected environments from the ACE, AAIR+REV and MIT test pools are listed in Tab. A.1, A.2 and A.3 in the supplemental material.²

D. Feature Extractor Benchmark for Few-Shot Identification

We train NN models from related work and popular standard architectures as feature extractors of the EnvId framework (Fig. 1 (b)) to be able to compare them to the proposed EnvId

²https://fau1-files.cs.fau.de/public/mmsec/moussa/2025TIFS/supplemental_material.pdf

TABLE II: Accuracy values (mean \pm standard deviation and maximum) of 5 training runs per feature extractor on the 4 test sets of noisy and single-compressed reverberant samples (*cf.* Sec. IV-D1). The best (bold), second best (underlined) and third best (italic) model results are highlighted. The parameter size includes the feature extractor and EnvId’s remaining trainable weights. We provide an own custom extractor (last row) as strong baseline for further research.

Feature Extractor	Params	ACE [10]		AAIR+REV [20], [23]		MIT [44]		OPENAIR [32]	
		$\mu \pm \sigma$	max	$\mu \pm \sigma$	max	$\mu \pm \sigma$	max	$\mu \pm \sigma$	max
Guessing chance		0.1430		0.1000		0.1000		0.1000	
Related Work — CRNNs:									
Papayiannis [35]	2.66 M	0.6768 \pm 0.0314	0.7122	0.6866 \pm 0.0166	0.7072	0.7134 \pm 0.0068	0.7239	0.5652 \pm 0.0274	0.5985
GamperCRNN [11]	1.25 M	0.7062 \pm 0.0336	0.7676	0.7362 \pm 0.0308	0.7738	0.7439 \pm 0.0181	0.7761	0.5551 \pm 0.0321	0.6157
Deng/Götz [8], [15]	5.49 M	0.8068 \pm 0.0271	0.8401	0.7614 \pm 0.0564	0.8427	0.6755 \pm 0.0647	0.7463	0.6204 \pm 0.0795	0.7373
Related Work — CNNs:									
VGGVox [2], [33]	11.65 M	0.4827 \pm 0.0079	0.4947	0.5362 \pm 0.0184	0.5637	0.5497 \pm 0.0065	0.5567	0.4727 \pm 0.0201	0.5037
Genovese/Ick [14], [19]	0.19 M	0.8196 \pm 0.1028	0.8721	0.7715 \pm 0.0841	0.8220	0.7340 \pm 0.1149	0.8022	0.6937 \pm 0.1064	0.7448
ThinResnet [2], [49]	11.72 M	0.8499 \pm 0.0178	0.8806	0.8583 \pm 0.0190	0.8783	0.8257 \pm 0.0054	0.8328	0.6907 \pm 0.0329	0.7388
Götz [16]	3.86 M	0.9416 \pm 0.0218	0.9616	0.9150 \pm 0.0355	0.9518	0.8236 \pm 0.0454	0.8679	0.8187 \pm 0.0501	0.8754
GamperCNN [12]	3.43 M	0.9467 \pm 0.0189	0.9638	0.9254 \pm 0.0291	0.9506	0.8345 \pm 0.0419	0.8687	0.8240 \pm 0.0486	0.8709
Standard Vision CNNs:									
RegNetY-400mf [38]	28.23 M	0.7211 \pm 0.0441	0.7868	0.7724 \pm 0.0307	0.8129	0.7334 \pm 0.0387	0.7851	0.6163 \pm 0.0400	0.6649
EffNet-B0 [42]	12.85 M	0.7318 \pm 0.0303	0.7868	0.8101 \pm 0.0157	0.8301	0.7939 \pm 0.0116	0.8067	0.6300 \pm 0.0191	0.6642
EffNet-B2 [42]	17.43 M	0.7548 \pm 0.0199	0.7932	0.8057 \pm 0.0278	0.8542	0.8078 \pm 0.0121	0.8291	0.6639 \pm 0.0209	0.6925
RegNetY-800mf [38]	43.75 M	0.7646 \pm 0.0187	0.7846	0.7929 \pm 0.0114	0.8140	0.7640 \pm 0.0213	0.7888	0.6436 \pm 0.0141	0.6604
ResNet-18 [17]	11.30 M	0.8171 \pm 0.0345	0.8593	0.8200 \pm 0.0276	0.8588	0.8124 \pm 0.0111	0.8269	0.6778 \pm 0.0348	0.7336
ResNet-50 [17]	24.03 M	0.8175 \pm 0.0113	0.8316	0.8361 \pm 0.0110	0.8485	0.8045 \pm 0.0106	0.8216	0.6840 \pm 0.0076	0.6925
ConvNeXt-Tiny [24]	28.01 M	0.8618 \pm 0.0112	0.8806	0.7986 \pm 0.0137	0.8129	0.7658 \pm 0.0190	0.8000	0.7022 \pm 0.0118	0.8000
Standard Transformers:									
Transf. Enc. [45]	19.19 M	0.4034 \pm 0.0394	0.4670	0.4693 \pm 0.0388	0.5385	0.4663 \pm 0.0141	0.4903	0.4227 \pm 0.0227	0.4545
ViT-Small [9]	86.08 M	0.4968 \pm 0.0350	0.5437	0.5146 \pm 0.0234	0.5408	0.4922 \pm 0.0113	0.5045	0.4813 \pm 0.0128	0.5015
Proposed Backbone:									
Gamper*	3.86 M	0.9557 \pm 0.0076	0.9701	0.9375 \pm 0.0047	0.9449	0.8712 \pm 0.0164	0.8903	<i>0.8131</i> \pm 0.0188	0.8358

configuration (*cf.* Sec. III-A2) on our task of forensic few-shot location identification.

The experiment is presented in four parts. First, we define the experimental setup, where Sec. IV-D1 presents the generated training and test data sets and Sec. IV-D2 summarizes the feature extractors for evaluation. Then, the experimental results are discussed in Sec. IV-D3, and Sec. IV-D4 provides an analysis of the generalization ability towards identifying out-of-distribution environments.

1) *Training and Test Data Sets:* The data generation pipeline from Sec. III-B is used to generate a training set, a validation set and four test sets from the respective speech and AIR pools in Tab. I. Note that for future reference, we name the test sets after their underlying environment pools, *i.e.*, as ACE, AAIR+REV, MIT and an OPENAIR test set. To create a set, each sample from the speech pool is convolved with each sample from the AIR pool, which leads to a uniform distribution of recording examples for each available environment (Fig. 2(a)). This experiment features the same common signal degradations in the training and test sets, while robustness towards unseen degradations is evaluated separately in Sec. IV-F. In detail, for each sample in a set, general background noise is simulated with additive white noise of random strength within a broad SNR range of $\alpha \in [-10, 50] \cup [\infty]$ db, where $\alpha = \infty$ corresponds to no noise being added (Fig. 2(b)). Single-compression (Fig. 2(c)), is then randomly applied with one of the popular formats MP3, AMR-NB, GSM with arbitrary and supported compression strength, or skipped. In detail, the set of available bitrate configurations to the pipeline

are $\mathcal{C}_{MP3} = \{8, 16, 24, 32, 40, 48, 56, 64, 80, 96, 112, 128\}$, $\mathcal{C}_{AMR-NB} = \{4.75, 5.15, 5.9, 6.7, 7.4, 7.95, 10.2, 12.2\}$ and $\mathcal{C}_{GSM} = \{13\}$, since GSM only operates with constant bitrate.

2) *Baseline Feature Extractors:* The comparison includes in total 17 feature extractors from the literature. This covers 8 neural feature extractors from related work (*cf.* Sec. I-A) that we implement from 11 corresponding published works. Three extractors are based on CRNNs, which we denote as Papayiannis [35], GamperCRNN [12], and Deng/Götz [8], [15]. Five extractors are based on CNNs, which we denote as VGGVox [2], [33], Genovese/Ick [14], [19], ThinResnet [2], [49], Götz [16], and GamperCNN [11]. Additionally, 9 standard CNNs from the RegNet [38], EffNet [42], ResNet [17] and ConvNext [24] families and two standard Transformer models [9], [45] are considered, because standard architectures oftentimes show good learning capabilities in various forensic and steganalytic tasks.

Each of the evaluated models is integrated into EnvId in two steps. The final classification/regression layers (if present) are removed, and the remaining layers (Fig. 1(b)) are linked to EnvId’s projector layer (Fig. 1(c)). The projector reduces the dimensionality of the model to $E = 256$, the size of EnvId’s metric embedding space (*cf.* Sec. III-A). Missing parameter configurations in related methods are supplemented with the respective standard values from PyTorch [36].

3) *Comparison of Feature Extractors:* Tab. II shows the evaluation results on all four test sets. For a representative comparison, we conduct 5 EnvId training runs with random weight initialization for each feature extractor and report the

mean accuracy and standard deviation, as well as the best model accuracy per test set. The first row lists the random guessing chance per test set for reference. Since we report 10-way few-shot results (*cf.* Sec. IV-A), this value corresponds to 0.1000, except for the small ACE set of 7 locations which enforces a 7-way setting with higher guessing chance of 0.1430. The second and following rows list the results per feature extractor and test set grouped by architecture type and origin, *i.e.*, related work vs. standard architectures. The rows in a group are sorted by the performance in ascending order. Our proposed backbone, *i.e.*, the variation of GamperCNN that we denote as Gamper* (*cf.* Sec. III-A) is shown in the last line.

Several insights can be drawn from these results. Overall, the proposed Gamper* performs best with an average accuracy score of 0.8944 over all sets and quite low standard deviations between 0.0047 and 0.0188 over the 5 training runs. The feature extractors GamperCNN and Götz perform second and third best with average accuracy values of 0.8826 and 0.8747 over all sets. Both yield less stable results over training runs than Gamper* with standard deviations between 0.0189 and 0.0486 for GamperCNN and between 0.0218 and 0.0501 for Götz. Notably, these three CNN feature extractors are relatively small, with less than 4 million parameters each. Nevertheless, they outperform the much more complex CRNN extractors and standard models that consist of a much larger number of parameters. It is particularly notable that the Transformer models achieve only a quite low performance.

We hypothesize that the benefits from large-scale vision nets and Transformers might potentially become accessible with even larger training data sets. However, due to the very promising results with efficient and slim CNNs models with short training times, we did not further investigate this direction.

Detailed performance results for each location class are listed for the ACE, AAIR+REV and MIT set in Tab. A.1, A.2 and A.3 of the supplemental material, for the proposed Gamper* feature extractor and the best baseline models Götz and GamperCNN.² The reports include the F1-score, precision, and recall for all individual recording locations. The corresponding scores on the most challenging OPENAIR set are separately discussed below.

4) *Generalization to Out-of-Distribution Locations:* The results in Tab. II show that OPENAIR is the most difficult test set. OPENAIR consists mainly of large, strongly reverberant locations like churches, cathedrals and halls. Such types of locations are only sparsely covered in the training set. In this section, we further characterize that sparsity and the associated challenge to generalize to out-of-distribution environments during few-shot classification.

One quantitative indication for the difficulty to generalize to the OPENAIR samples is provided by the Pearson correlation coefficient between the AIRs in the training and test sets. In detail, the Pearson correlation coefficients for the ACE, AAIR+REV, and MIT test sets are 0.5555, 0.3162 and 0.5844, whereas it is significantly lower for the OPENAIR test set with a value of 0.2375.

Especially at the level of individual AIRs, it is possible to observe the impact of differences in the training and test

distributions. To this end, we limit the examination to the three best performing backbones Gamper*, GamperCNN and Götz, and examine the F1-score for each location in the OPENAIR test set. The averaged performance scores over 5 training runs per model are listed in Tab. III in descending order w.r.t. the F1-score of the proposed Gamper* backbone. All three models perform worst on the same 4 cathedral and hall environments (bottom part of Tab. III). For these 4 worst environments, the F1-score is only 0.5917, 0.6590, and 0.6298 for Gamper*, GamperCNN and Götz. Without these 4 locations, the F1-scores for the remaining locations are much higher at 0.8637, 0.8661, and 0.8652, respectively.

The proposed Gamper* performs slightly worse than GamperCNN and Götz on OPENAIR locations, which shows the contradicting goals of specialization and generalization. However, the results are overall quite robust, given that the tested AIRs are considerably different from the training data.

E. Open-Set Matching: Rejection of Unknown Environments

In several important practical use cases, it is helpful if a system can indicate that an input does not match any of the reference locations. To our knowledge, this task is not addressed in related works. EnvId can be straightforwardly used for open-set few-shot classification by rejecting an input query if its distance to all available prototypes exceeds a threshold. The specific choice of rejection threshold depends on practical requirements, *i.e.*, whether a higher recall (true positive rate (TPR)) or lower false positive rate (FPR) is targeted.

For the evaluation, we only allow a test query’s recording location as reference candidate in 50% of the cases. Consequently, EnvId needs to reject half of the query samples as unknown. To show this general rejection ability, we calculate the average ROC over the 5 trained EnvId checkpoints with feature extractors Gamper*, GamperCNN and Götz from Tab. II on our 4 test sets of noisy and single compressed speech samples (*cf.* Sec. IV-D1).

The ROC results together with the area under the curve (AUC) score are plotted in Fig. 3. The TPR denotes the amount of samples correctly rejected as unknown and the FPR denotes the amount of samples wrongfully assigned to one candidate location. The proposed Gamper* feature extractor performs best across all test sets and has the lowest standard deviations over all runs. Götz and GamperCNN perform second and third best with AUC values that are on average 7.41 percentage points (pp) and 8.67 pp behind Gamper*’s performance. In line with the results in Tab. II, the best performance for all models is achieved on the ACE test set (Fig. 3a), followed by the AAIR+REV set (Fig. 3b). More in detail, the AUC score for Gamper* is 0.92 on ACE locations and 0.89 on AAIR+REV locations. The difference to the remaining feature extractors is largest for those two sets, where Götz yields scores of 0.83 and 0.81 and GamperCNN achieves AUC values of 0.80 and 0.79. Recognizing unknown locations from queries of the MIT and OPENAIR set is significantly harder (*cf.* Fig. 3c vs. Fig. 3d). Here, the AUC scores for Gamper* are 0.81 and 0.84. So, rejecting query samples from the OPENAIR

TABLE III: Mean F1-score, precision and recall per environment over 5 training runs of our EnvId framework on the OPENAIR [32] test set of noisy and single-compressed reverberant samples (*cf.* Sec. IV-D1). The results are reported for the Gamper*, GamperCNN [12] and Götz [16] feature extractor.

OPENAIR Class Label	Volume Space Category	Gamper*			GamperCNN			Götz		
		F1	P	R	F1	P	R	F1	P	R
Innocent Railway Tunnel Entrance	13000 m ³ -	0.9664	0.9877	0.9463	0.9548	0.9639	0.9463	0.9565	0.9591	0.9552
Live Room	35.2 m ³ Recording Studio	0.9604	0.9783	0.9433	0.9565	0.9589	0.9552	0.9704	0.9753	0.9672
Falkland Palace Bottle Dungeon	- Chamber	0.9548	0.9968	0.9164	0.9588	0.9781	0.9403	0.9307	0.9598	0.9045
Koli National Park Summer Stairwell	- Open Air - Hall	0.9424	0.9575	0.9284	0.9263	0.9891	0.8716	0.9182	0.9273	0.9104
Arthur Sykes Rymer Audit. Univ. York	1560 m ³ Auditorium	0.8998	0.8874	0.9134	0.8633	0.8532	0.8746	0.8892	0.8932	0.8866
Spokane Woman’s Club	1600 m ³ Auditorium, Ballroom, Hall	0.8911	0.8662	0.9224	0.8543	0.8160	0.9015	0.8466	0.8266	0.8687
Falkland Tennis Court	2300 m ³ Open Air, (Sports) Hall	0.8635	0.8488	0.8806	0.9084	0.8891	0.9313	0.8954	0.8702	0.9254
Dixon Studio Theatre Univ. York	908.23 m ³ Theater	0.8610	0.8870	0.8388	0.8069	0.8106	0.8060	0.8335	0.8407	0.8269
Council Chamber	1140 m ³ Chamber	0.8525	0.8049	0.9075	0.8865	0.9030	0.8716	0.9071	0.8992	0.9164
Lady Chapel St Alban’s Cathedral	- Cathedral	0.8524	0.8189	0.8896	0.8196	0.7936	0.8478	0.8374	0.8227	0.8537
Jack Lyons Concert Hall Univ. York	- Concert Hall	0.8153	0.7733	0.8627	0.8345	0.7910	0.8836	0.8234	0.8106	0.8388
Central Hall Univ. York	8000 m ³ Auditorium, Hall	0.8135	0.8052	0.8239	0.7977	0.7846	0.8149	0.7818	0.7622	0.8030
Alcuin College Univ. York	21000 m ³ Open Air	0.8047	0.7202	0.9134	0.8514	0.8341	0.8716	0.8577	0.8242	0.8955
Baptist Nashville Balcony	- Church	0.7947	0.7252	0.8806	0.8530	0.8593	0.8478	0.8425	0.8459	0.8418
Heslington Church	2000 m ³ Church	0.7780	0.7904	0.7672	0.7991	0.8287	0.7731	0.7612	0.7879	0.7403
Nuclear Reactor Hall	3500 m ³ Hall	0.7689	0.8170	0.7284	0.7874	0.8261	0.7552	0.7919	0.8211	0.7701
Terry’s Warehouse	4500 m ³ Hall	0.6832	0.6431	0.7313	0.6356	0.5547	0.7493	0.6258	0.5591	0.7134
York Minster	140000 m ³ Cathedral	0.5982	0.6904	0.5343	0.6449	0.6928	0.6060	0.6481	0.6693	0.6299
Sportscentre	9000 m ³ (Sports) Hall	0.5592	0.7172	0.4597	0.7374	0.8515	0.6507	0.7435	0.8391	0.6687
		0.5263	0.5939	0.4746	0.6180	0.6606	0.5821	0.5016	0.5597	0.4567

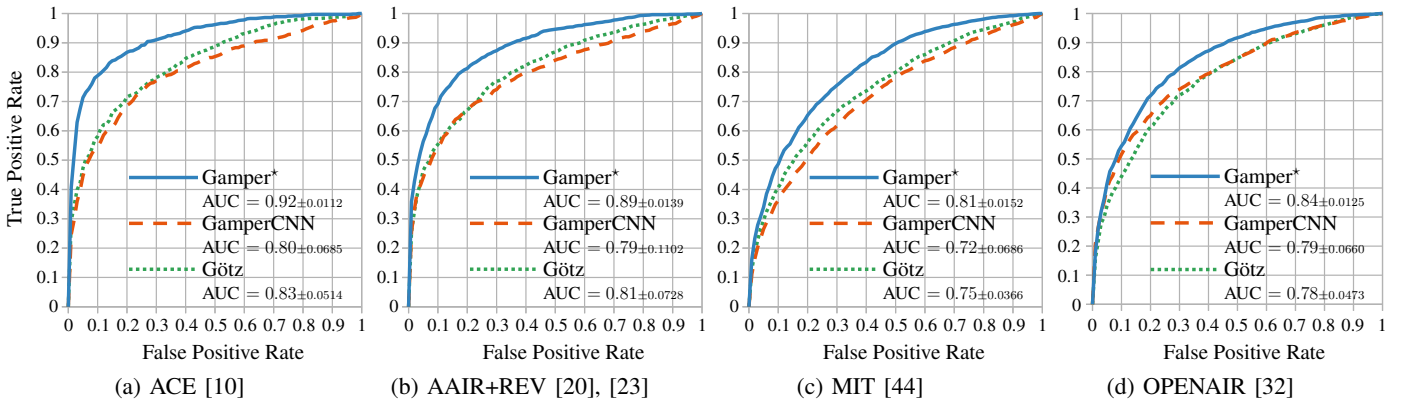


Fig. 3: Receiver operating characteristic (ROC) curves on our 4 test sets of noisy and single compressed reverberant speech from Sec. IV-D1 for rejecting samples that do not match any reference recording location. The results are reported for the Gamper*, GamperCNN [12] and Götz [16] feature extractor.

set is slightly easier for the classifier, even if the identification scores on these out-of-distribution environments is lower (*cf.* Sec. IV-D4). The same is true for Götz and GamperCNN with in total lower AUC scores, *i.e.*, AUC = 0.75 and AUC = 0.78 for Götz and AUC = 0.72 and AUC = 0.79 for GamperCNN.

To sum up, these findings show that EnvId is able to perform open-set few-shot identification of recording environments, but nevertheless there is still room for improvement. Therefore, we regard the matter as a possibility for future research.

F. Robustness towards Unseen Signal Degradations

This section provides experiments to characterize EnvId’s robustness towards signal degradations unseen during training. After all, unseen post-processing has to be expected for unconstrained in-the-wild audio data in practice. We here

include unseen multiple compression, compression codecs and background noise.

Equally to the previous evaluations, the experiments are performed with the trained EnvId checkpoints wrapping the three best performing feature extractors Gamper*, GamperCNN and Götz from Sec. IV-D. Recall that those models were trained on samples degraded by additive white noise of random strength and one compression run with random format and bitrate setting (*cf.* Sec. IV-D1). For the evaluation, we use variants of the MIT test set from Sec. IV-D1 with altered noise and compression degradation settings as described in the respective sections below. The MIT set is one of our larger sets and contains a diverse distribution of recording locations from small to large environments.

1) *Multiple Compression Runs*: When audio messages are shared over the internet, the signal might be recompressed multiple times. To analyze such situations, the data pipeline

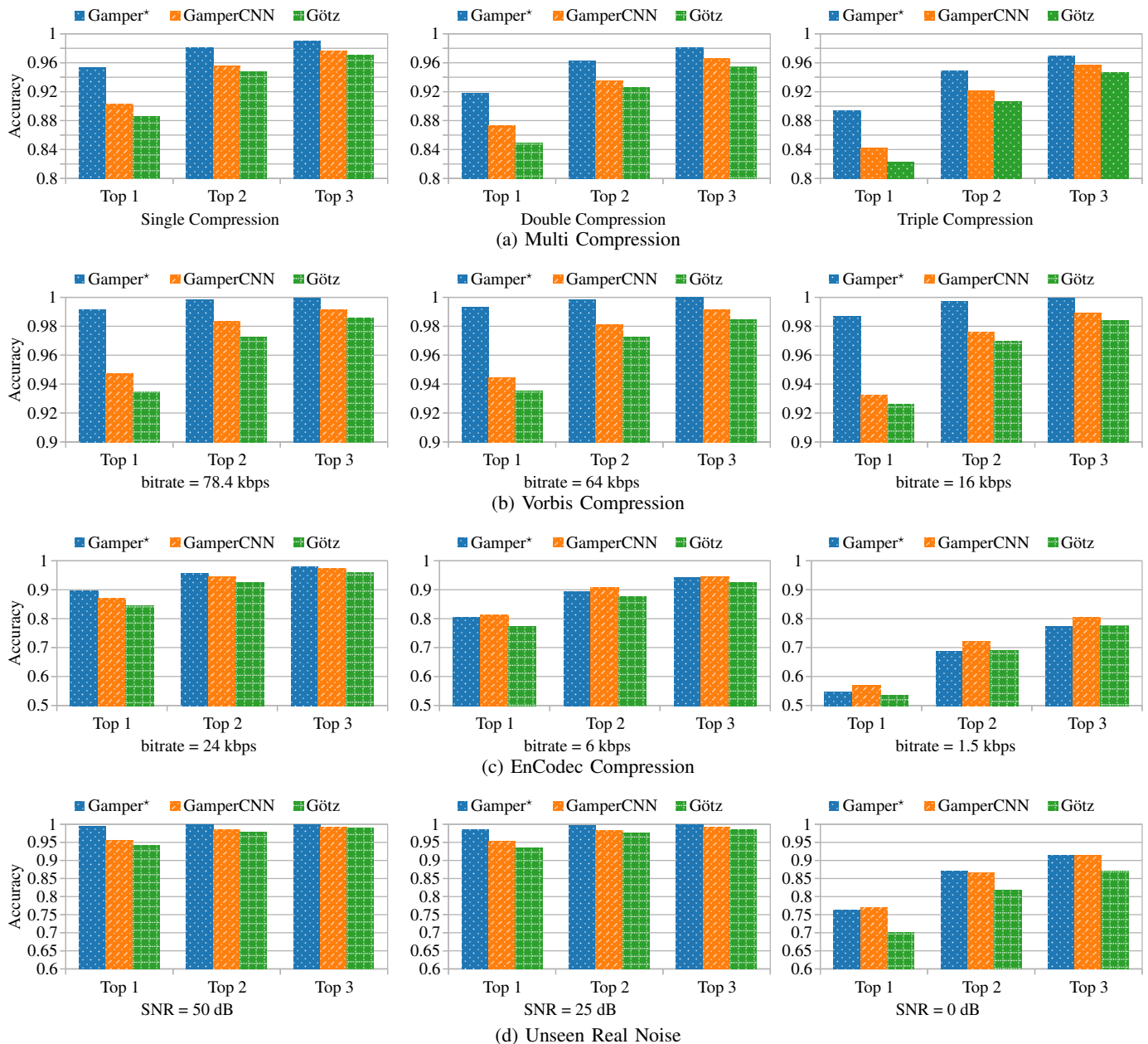


Fig. 4: Averaged Top- $\{1,2,3\}$ accuracy of 5 training runs for few-shot environment identification under degradation factors unseen during training. We provide benchmarks for the Gamper*, GamperCNN [12] and Götz [16] backbones on the MIT [44] test set for multi compression runs (Fig. 4a), high, mid and low quality settings of the unseen Vorbis (Fig. 4b) and neural EnCodec [7] compression codecs (Fig. 4c), and unseen real environmental background noise (Fig. 4d). Note that the y-axis is scaled differently for the individual degradation types to better highlight variations within each experiment.

is configured to construct two MIT test set versions, one with double compressed samples and one with triple compressed samples. The sets do not contain noise degradation and the compression configurations are sampled from the training distribution to isolate the influence of multiple runs. In detail, for each of $n \in \{2, 3\}$ compression runs per sample, the format and strength is randomly chosen from MP3, AMR-NB and GSM with valid bitrate settings (*cf.* Sec. IV-D1).

Fig. 4a shows the average Top- $\{1,2,3\}$ accuracy over 5 training runs per model for single, double and triple com-

pression. The proposed Gamper* feature extractor performs best in all cases, followed by GamperCNN and Götz. As expected, the accuracy degrades with increasing number of compressions for all backbones. However, the total performance remains high even for Top-1 predictions. For Gamper*, the accuracy is 0.9176 for double compression and 0.8937 for triple compression, which is only moderately lower than 0.9531 for single compression. For Top- $\{2,3\}$ predictions, the performance never falls below 0.9496. The difference to the remaining two backbones is the largest for the Top-1 accuracy.

Here, the scores of GamperCNN are on average lower by 4.92 pp, while Götz is on average even 6.91 pp behind Gamper*. All in all, the experiment thus shows EnvId’s good generalization ability to longer compression chains, especially with the proposed Gamper* backbone.

2) *Unseen Compression Codecs*: Fast progress in audio compression research might lead to the case that EnvId must perform inference on compression codecs that it has not been trained for.

We thus evaluate EnvId on signals compressed by two codecs that were not seen during training, namely Vorbis as an established analytic compressor and EnCodec [7] from Meta Research[©] as a relevant neural network-based compressor. The data generation pipeline is configured accordingly to produce in total 6 MIT test set variants, one for each codec with low, middle and high quality settings. No noise degradation is applied to the samples.

Fig. 4b shows the result for Vorbis, and Fig. 4c shows the results for EnCodec. The accuracy on Vorbis compression is very close to 1 for the best performing Gamper*, which is understandable, as it targets mid to high quality output where the lowest available bitrate is still as high as 16 kbps. In line with the results on multi-compression, GamperCNN and Götz perform second and third best, and the difference to Gamper* is most notable for the Top-1 accuracy, where, on average, GamperCNN and Götz perform lower than Gamper* by 4.92 pp and 5.89 pp.

In contrast to Vorbis, EnCodec has a higher impact on the performance. Overall, the Gamper* backbone has a slight advantage over the remaining two backbones on the high (24 kbps) quality setting, and performs second best but comparably to GamperCNN for the mid (6 kbps) and low (1.5 kbps) quality setting. More precisely, the Top-1 accuracies of Gamper* on the high (24 kbps) and mid (6 kbps) quality settings are 0.8951 and 0.8028, but it drops to 0.5478 for the low (1.5 kbps) quality setting. Also, GamperCNN only reaches slightly better Top-1 scores of 0.8131 and 0.5713 for mid and low qualities. The low accuracy on such an extremely low bitrate is not surprising: the lowest available bitrate in the training set are 4.75 kbps with AMR-NB, hence EnvId has to simultaneously deal with unseen compression artifacts at an unseen low bitrate.

In total, we report that EnvId is quite robust to unseen compression codecs, also at low bitrates of 24 kbps or even 6 kbps. Only EnCodec’s strongest compression of 1.5 kbps considerably degrades the identification accuracy.

3) *Unseen Noise Impact*: In practice, the background noise may considerably vary in the training and test data distribution. To evaluate EnvId’s generalization ability towards real environmental background noise, a selection of 4 publicly available noise sources is used that cover a passing train³, falling rain⁴, and sounds of a crowded airport⁵ and exhibition hall⁶. For the evaluation, 3 MIT test set versions with low (SNR = 50 dB), medium (SNR = 25 dB) and high (SNR = 0 dB) noise

degradation are generated. For each set version, one of the 4 background noise patterns is randomly selected per sample and added with the corresponding fixed test SNR.

The evaluation result is shown in Fig. 4d. In total, the Gamper* backbone again performs best, followed by GamperCNN and Götz. Solely for strong noise impact with SNR = 0 dB, GamperCNN* performs approximately equally well to Gamper*. Low (SNR = 50 dB) and medium (SNR = 25 dB) noise levels have little influence on EnvId’s Top-1 accuracy which is especially close to 1 for Gamper*. However, strongly degraded signals (SNR = 0 dB) cause the accuracy to fall to 0.7648, 0.8697 and 0.9134 for the Top-1, Top-2 and Top-3 predictions, respectively. Nevertheless, the results are remarkable when considering that the noise pattern is unseen and the distortion of the signal is extremely high.

G. Ablation on the Number of Samples per Prototypes

EnvId requires a support set of K reference samples for prototype computation per candidate recording environment. There are no strict constraints on the number of reference samples. From a theoretical perspective, one might expect that “more data is better”. However, in practice, meta-learning approaches like Prototypical Networks tend to overfit on the K value used in training [5]. Thus, K is typically chosen to be identical in training and testing as done in our experiments (*cf.* Sec. IV-A). Still, the question of a lower limit on the number of required samples is relevant for our scenario. After all, the collection of reference samples from locations in a forensic case is arguably constrained by the available resources for data acquisition. Therefore, we here evaluate the impact of testing with $K < 15$ using our EnvId framework trained with $K = 15$ samples (*cf.* Sec. IV-A).

Fig. 5 shows the accuracy of EnvId for K varying between 1 and 15, using the best performing feature extractor Gamper* on the 4 test sets from Sec. IV-D1. For our scenario, the results do not indicate a severe impact of a train/test mismatch of K for most test sets. In detail, for ACE and AIR+REV sets, already $K = 1$ provides a good prototype and locations of MIT can be stably identified with $K = 2$ reference samples. Only OPENAIR locations require at least $K = 10$ samples to yield stable prototypes. OPENAIR’s mostly large and strongly reverberant environments deviate the most from the training set distribution (*cf.* Sec IV-D4). Thus, we hypothesize that EnvId’s embedding space is less fit for representing these types of locations, such that a higher K is needed to form stable prototypes. Analogously to Sec IV-D4, these findings emphasize the importance of analyzing training data for weakly covered environmental characteristics to prevent such pitfalls.

H. Recording Position Mismatch

The actual position of the microphone in the room is a subtle, yet quite impactful issue that has been barely addressed in related works (*cf.* Sec. I-B).

If reference audio samples from candidate environments are collected during the investigation process, it must be assumed that the recording positions in the support set is at least somewhat different from the query signal. Therefore,

³<https://freesound.org/people/theplax/sounds/615849/>

⁴<https://freesound.org/people/straget/sounds/531947/>

⁵<https://freesound.org/people/arnaud%20coutancier/sounds/424362/>

⁶<https://freesound.org/people/BockelSound/sounds/487600/>

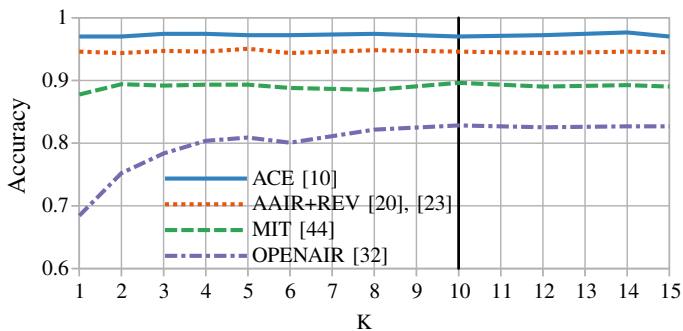


Fig. 5: Location identification accuracy w.r.t. the K -shot parameter for inference on the test sets from Sec. IV-D1. The out-of-distribution OPENAIR environments require the most K reference samples.

in this experiment, we provide a benchmark and data set with synthetic data of enclosed spaces of varying recording positions in differently shaped rooms.

For the evaluation, we fine-tune EnvId with the best performing Gamper* feature extractor from Sec. IV-D on a synthetic data set with annotated recording positions. The training hyperparameters remain unchanged for the fine-tuning process (*cf.* Sec. IV-A).

The experiment is split into four parts. In Sec. IV-H1, the method for simulating AIRs from synthetic rooms is described and Sec. IV-H2 summarizes the resulting training and test data sets of reverberant speech. The results on the influence of the microphone position is presented in Sec. IV-H3. Sec. IV-H4 further investigates the impact of noise and compression signal degradation, and Sec. IV-H5 discusses the results.

1) *Simulation of Synthetic AIRs*: To our knowledge, there is no data set that provides a grid of real AIR measurements with known positions for a large number of recording locations. Hence, we simulate enclosed spaces with pyroomacoustics [40]. We consider rooms within a length interval of $l_r \in [1, 50]$ m and height $h_r \in [2, 5]$ m in steps of 10 cm. Three common room shapes are *corridor*, *rectangle*, and *square* rooms, where we define the fraction f of width over length as $f \in [0.1, 0.3]$ for corridor, $f \in [0.4, 0.7]$ for rectangle, and $f \in [0.8, 1]$ for square.

Different floor and wall characteristics are simulated with an absorption coefficient $c_a \in [0.1, 0.8]$. AIR measurements are sampled from a equidistant 5×5 grid per room to uniformly cover potential microphone positions. The distance between the outer edges of the grid to the corresponding nearest wall is set to 30 cm. For this first study, we focus on the practically relevant scenario of some speaker during a telephone conversation or the recording of voice messages. The relative distance between the microphone and the speaker is thus set to a fixed size of 10 cm, and the microphone is always positioned 1.7 m above the ground floor. The speaker orientation is always set towards the room center and the microphone is set inversely. This preserves the comparability of different grid positions.

2) *Training and Test Data*: Our final training/validation includes samples from 45/15 rooms for each of the 3 shape

categories uniformly sampled from the whole volume range, which leads to 135/45 rooms per set. For each room, absorption coefficients are randomly sampled. AIR measurements are simulated for each of the 5×5 grid positions, yielding a total of 3375/1125 training/validation AIRs. Each AIR is convolved with each of 25/19 anechoic speech samples from a subset of the ACE training and validation pool (Tab. I). To approximate real world scenarios, the data generation pipeline is configured to randomly add white noise within a SNR range of $\alpha \in [-10, 50]$ and perform arbitrarily strong MP3, AMR-NB or GSM single compression for each sample. This degradation configuration is analogous to the training set from the previous experiments (*cf.* Sec. IV-D1). The final sets consist of about 84k/21k training/validation samples.

The test set is constructed with a volume range of $V \in [10, 3750]$ m³, as 3750 m³ is the maximum possible volume for a *corridor* room in our setting. We sample one room per shape category for 10 volumes uniformly distributed over the full range to yield 30 test rooms in total. The absorption coefficient is fixed to $c_a = 0.1$ for medium absorption [40]. For this experiment, the fixed absorption coefficient in the test set enables an isolated analysis of the impact of changing recording positions. Per room, AIRs measured from the 5×5 grid positions are convolved each with a subset of 16 samples from the TSP test pool (Tab. I), yielding 12k samples in total. The test set does not include noise or compression degradation, whose influence is analyzed individually in Sec. IV-H4.

3) *Recording Position Impact*: For the evaluation, the samples from one position (queries) are tested against prototypes that are calculated randomly from reference samples of all other recording positions. When averaging over all positions in all rooms per room category, then *corridor* rooms are easiest to identify with an average accuracy of 0.7715, followed by *rectangle* and *square* rooms with average accuracies of 0.7648 and 0.7608.

The results for each individual position are visualized in Fig. 6 for the three room shape categories *corridor* (left), *rectangle* (middle), and *square* (right). We additionally provide numerical values for each position in Tab. A.4 in the supplemental material.² Overall, the accuracy tends to increase at locations that are closer to walls. For example, the average center accuracies are 0.6375, 0.6875 and 0.7063 for *corridor*, *rectangle* and *square* rooms, but average accuracies at corners are 0.8453, 0.7750 and 0.7328. For *rectangle* and *square* rooms it is further notable that the most difficult line of positions is the one along the middle column, on average 0.7038 and 0.7113. For *corridor* rooms, the middle row is the most difficult grid line with 0.6800. Overall, the difference in accuracy between inner and outer positions is higher for the narrow *corridor* spaces than for the more uniform *rectangle* or *square* rooms. The larger variation is reflected by the standard deviation over all positions on the grid, which is 0.0748 for the category *corridor*, but only 0.0419 and 0.0438 for the category *rectangle* and *square*. These findings suggest that early reflections in the AIR for recordings close to walls lead to more characteristic features, which makes environment identification easier. This is in line with the observation that the overall accuracies are highest on *corridor* rooms, where

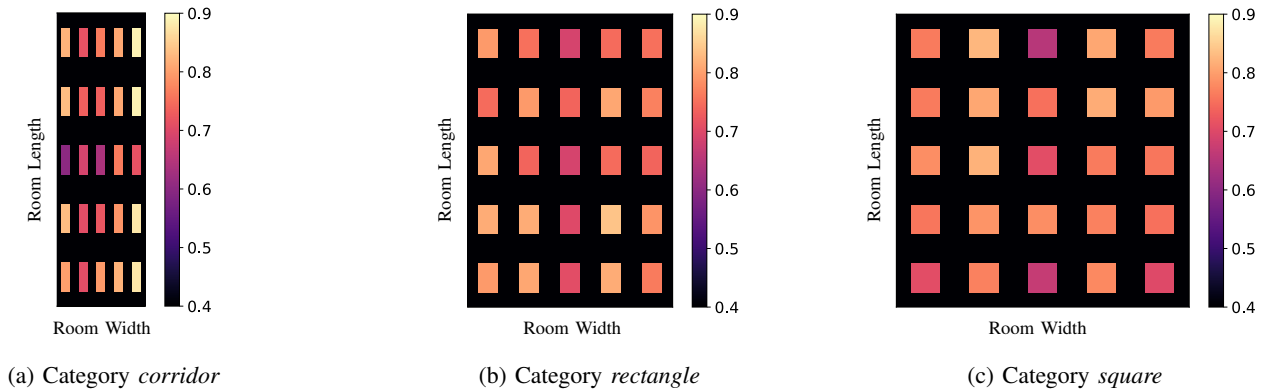


Fig. 6: Mean accuracy for specific microphone positions on a uniformly sampled 5×5 grid for the categories *corridor*, *rectangle* and *square*. Environments for audio samples recorded closer to boundaries are on average easier to identify. The total accuracy thus decreases for wider room widths from left to right.

the distances to the side walls are relatively short.

4) *Influence of Noise and Compression*: We also investigate the impact of compression and noise degradation on the identification scores. Therefore, two further versions of our test set of synthetic rooms from Sec. IV-H2 are generated. For the compression test set, the pipeline is again configured to randomly compress each sample once with MP3, AMR-NB or GSM and random compression quality settings, and for the noise test set, additive white noise of random SNR between -10 dB and 50 dB is chosen for each sample.

For compression, the identification scores for the three room categories, *i.e.*, the averaged accuracy over all recording positions drops to 0.6458 , 0.6163 and 0.6340 for *corridor*, *rectangle* and *square* rooms. This corresponds to an average decrease by 13.67 pp. Noise has an even stronger impact and decreases the values down to 0.5298 , 0.4898 and 0.5118 , *i.e.*, a decrease by 25.52 pp.

5) *Discussion*: The results indicate that recording position is indeed an important factor for environment identification that should be considered in future research and in practice. Especially speech that is recorded near walls is easier to identify due to nearby signal reflections. On average, this leads to better results on elongated rooms like corridors than on square rooms of the same volume. In future work, these findings remain to be more closely analyzed by acquiring a corpus of real-world AIR measurements from annotated positions.

I. Blind Environment Parameter Estimation

EnvId blindly regresses environmental parameters from the query’s embedding vector (*cf.* Sec. III-A). Such parameter estimates can act as an additional hint to the few-shot location predictions. Additionally, the regression can still provide valuable cues for forensic cases, where no set of candidate locations is available. In our evaluation, we demonstrate the regression capability at the example of the room volume V and the RT_{60} parameter, which are popular quantities in related work (*cf.* Sec. I-A).

We use the same training setup as for the previous experiment, *i.e.*, train the best EnvId configuration on our set of synthetic enclosed spaces (*cf.* Sec. IV-H2). For the synthetic data, both volume and RT_{60} labels are known, such that EnvId’s regression head can be jointly trained with the few-shot task (*cf.* Sec. III-A6).

1) *Comparison to Related Metric-Learning Approach*: We first compare EnvId’s regression capability to the recently proposed parameter estimator by Götzt *et al.* [16] who also apply representation learning techniques.

In detail, they use contrastive learning with an upstream encoder network to obtain embeddings for room volume and RT_{60} . These embeddings are fed to a downstream network to classify the room volume as small or large, and to regress the RT_{60} . With the help of the authors, we recreate the data set from the publication according to their protocol. The training set consists of simulated rooms of uniformly distributed volumes $V \in [27, 500]m^3$ and randomly sampled absorption coefficients. The test set consists of 1000 AIRs, *i.e.*, 10 AIRs from 100 simulated rooms as described in their work [16].

When running this experiment, EnvId achieves a root mean squared error (RMSE) of 0.2129 for RT_{60} and a remarkable volume classification accuracy of 0.9998 . Both values outperform the approach of Götzt *et al.*, who report a minimum RMSE of 0.2146 for RT_{60} and a maximum volume classification accuracy of 0.7422 .

To analyze the capabilities of EnvId more in detail, we additionally evaluate the model on our synthetic rooms with larger volume range $V \in [10, 3750]m^3$. To this end, we construct 3 test sets. One version with clean samples is generated like the test set for the previous experiment (*cf.* Sec. IV-H2), however with randomly sampled absorption coefficients c_a between 0.1 and 0.8 for all rooms. We also generate a noisy version and a version with single-compressed signals. The results are presented in the following two sections.

2) *Analysis of RT_{60} Estimations*: We first evaluate the influence of room shape on the estimation of the RT_{60} parameter. The results are shown in Tab. IV. Both the RMSE

TABLE IV: Results for the regression of RT_{60} on our 3 synthetic test sets of clean, single compressed and noisy signals. The RMSE and mean target RT_{60} are reported grouped by the room categories *corridor*, *rectangle*, and *square*.

Room Category	RMSE			Mean Targets
	Clean	Compr.	Noise	
Corridor	0.2640	0.2723	0.2748	0.6497
Rectangle	0.1692	0.1921	0.2157	0.4930
Square	0.1553	0.1777	0.2117	0.4125

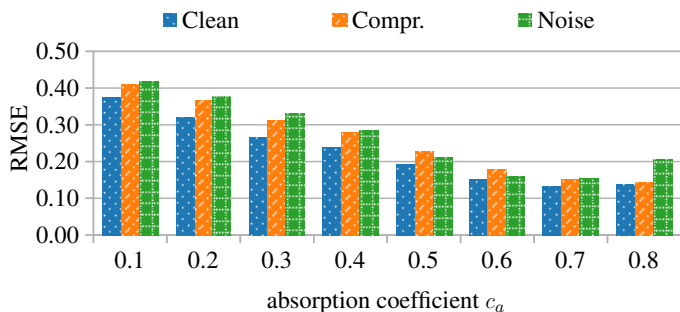


Fig. 7: RMSE results for RT_{60} grouped by absorption coefficient $c_a \in [0.1, 0.8]$ for clean, noisy and single compressed test samples.

and RT_{60} mean values of the targets are reported grouped by shape category. As is reflected by the mean target RT_{60} , the RT_{60} naturally increases for more elongated rooms in our set. Equally, the RMSE increases, such that *corridor* rooms exhibit higher RMSE scores than *rectangle* and especially *square* rooms. Compression and added noise slightly reduce the performance. The average RMSE increases by 1.79 pp for compression and by 3.79 pp for noise. These findings are in line with the results in the previous experiment (Sec. IV-H).

Additionally to room shape and size, the absorption coefficient c_a has a large impact on the RMSE predictions. The absorption characteristic for a specific room geometry strongly influences the energy decay of some emitted signal and thus the resulting RT_{60} value. Fig. 7 shows the average RMSE scores grouped by c_a and signal degradation in the test data. Note that for this experiment, we conduct in total $8 \cdot 3 = 24$ runs, *i.e.*, one run on each of 8 variants of the clean, noise and single compression test data sets with fixed $c_a \in [0.1, 0.2, \dots, 0.8]$. Evidently, the RMSE decreases with increasing absorption, *i.e.*, decreasing RT_{60} labels which complements the results in the last column of Tab. IV. Also, in line with previous experiments, single compression and noise degradation increase the prediction errors.

3) *Fine-Grained Volume Classification*: To provide an intuition about the quality of the volume predictions, we extend the evaluation approach by Götz *et al.* [16]. As previously discussed, they use the volume estimate to classify room sizes as small or large (*cf.* Sec. IV-I1). In this experiment, we generalize this approach beyond a binary task and evenly divide the volume range of our synthetic rooms in $n \in \{2, 3, 5, 10\}$ classes to test increasingly fine-grained versions of the prob-

TABLE V: Volume classification results on our 3 synthetic test sets with clean, single compressed and noisy samples. We report scores for classifying rooms in 2, 3, 5, or 10 size categories.

Signal Degradation	Accuracy			
	2-class	3-class	5-class	10-class
Clean	0.9706	0.9519	0.8566	0.7808
Compression	0.8872	0.8410	0.7045	0.6593
Noise	0.8298	0.7930	0.6605	0.5999

lem. Equivalently to Götz *et al.* [16], the classification is done via binning the volume estimates, *i.e.*, classifying the volume as belonging to one of the n size categories.

Tab. V shows the results on our 3 test set versions with clean, noisy and single compressed samples. The two class task, *i.e.*, $n = 2$ is solved with a high accuracy of 0.9706, also for our data set. Single compression decreases the accuracy to 0.8872, and added noise to 0.8298. Naturally, the accuracy decreases for increasing numbers of size categories. Nevertheless, for the finest problem granularity of $n = 10$ classes, EnvId still achieves accuracies of 0.7808, 0.6593 and 0.5999 for clean, compressed and noisy inputs.

4) *Discussion*: Overall, our findings show that EnvId is able to blindly extract relevant cues about recording location characteristics from the queries' embedded representations, also under signal impacting factors like noise or compression. Additionally, our experiments demonstrate that optimizing the regressor in EnvId as a side-task to few-shot identification can outperform a state-of-the-art approach for parameter estimation only.

V. CONCLUSION

DL tools are increasingly moving into the focus of police authorities to support criminal investigations. However, it remains an open challenge to develop tools that are flexible and robust enough to meet forensic requirements. In particular, the identification of recording environments from single audio samples is difficult due to the many practical challenges in this seemingly straightforward task.

In this work, we propose the end-to-end trainable EnvId framework as a step towards practically applicable recording environment identification. The proposed framework performs few-shot classification to identify case-specific environments not seen in training and supports environmental parameter estimation for recording location characterization.

Furthermore, EnvId addresses open-set identification and identification of data with unknown signal degradations, which is equally motivated by the challenges in forensic police work. We also explore the so-far neglected, but practically very important issue of a mismatch in recording positions within a location. Here, our experiments quantitatively show that it is notably easier to identify a room from mismatched recording position when they were done closer to the walls.

We see EnvId as a cornerstone for future work on these challenges. We aim to further explore the task of forensic recording location identification and improve our framework's

overall performance. For example, the efficiency on real-world reverberant speech samples from annotated locations is yet to be explored. Up to now, this is hindered by the lack of sufficiently large available data sets. Additionally, a protocol on how to collect an optimal support set of reference samples from recording locations for practice has to be established. After all, recording settings like the microphone position can influence the few-shot performance, as suggested by our experiments.

REFERENCES

- [1] Konstantinos Apostolidis, Jakob Abesser, Luca Cuccovillo, and Vasileios Mezaris. Visual and Audio Scene Classification for Detecting Discrepancies in Video: a Baseline Method and Experimental Protocol. In *Proceedings of the 3rd ACM International Workshop on Multimedia AI against Disinformation*, pages 30–36, 2024.
- [2] Mohammadreza Azimi and Utz Roedig. Room Identification with Personal Voice Assistants. In *European Symposium on Research in Computer Security*, pages 317–327. Springer, 2021.
- [3] Gianmarco Baldini and Irene Amerini. Microphone Identification based on Spectral Entropy with Convolutional Neural Network. In *International Workshop on Information Forensics and Security*, pages 1–6. IEEE, 2022.
- [4] Malte Baum, Luca Cuccovillo, Artem Yaroshchuk, and Patrick Aichroth. Environment Classification via Blind Roomprints Estimation. In *International Workshop on Information Forensics and Security*, pages 1–6. IEEE, 2022.
- [5] Tianshi Cao, Marc T Law, and Sanja Fidler. A Theoretical Analysis of the Number of Shots in Few-Shot Learning. In *International Conference on Learning Representations*, 2020.
- [6] Joon Son Chung, Jaesung Huh, Seongkyu Mun, Minjae Lee, Hee Soo Heo, Soyeon Choe, Chiheon Ham, Sunghwan Jung, Bong-Jin Lee, and Icksang Han. In Defence of Metric Learning for Speaker Recognition. In *INTERSPEECH*, pages 2977–2981, 2020.
- [7] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High Fidelity Neural Audio Compression. *Transactions on Machine Learning Research*, 2023.
- [8] Shuwen Deng, Wolfgang Mack, and Emanuël AP Habets. Online Blind Reverberation Time Estimation Using CRNNs. In *INTERSPEECH*, pages 5061–5065, 2020.
- [9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*, 2021.
- [10] James Eaton, Nikolay D Gaubitch, Alastair H Moore, and Patrick A Naylor. Estimation of Room Acoustic Parameters: The ACE Challenge. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(10):1681–1693, 2016.
- [11] Hannes Gamper. Blind C50 Estimation from Single-Channel Speech Using a Convolutional Neural Network. In *International Workshop on Multimedia Signal Processing*, pages 1–6. IEEE, 2020.
- [12] Hannes Gamper and Ivan J Tashev. Blind Reverberation Time Estimation Using a Convolutional Neural Network. In *16th International Workshop on Acoustic Signal Enhancement*, pages 136–140. IEEE, 2018.
- [13] Jürgen T Geiger, Björn Schuller, and Gerhard Rigoll. Large-scale Audio Feature Extraction and SVM for Acoustic Scene Classification. In *Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 1–4. IEEE, 2013.
- [14] Andrea F Genovese, Hannes Gamper, Ville Pulkki, Nikunj Raghuvanshi, and Ivan J Tashev. Blind Room Volume Estimation from Single-Channel Noisy Speech. In *International Conference on Acoustics, Speech and Signal Processing*, pages 231–235. IEEE, 2019.
- [15] Philipp Götz, Cagdas Tuna, Andreas Walthner, and Emanuël AP Habets. Blind Reverberation Time Estimation in Dynamic Acoustic Conditions. In *International Conference on Acoustics, Speech and Signal Processing*, pages 581–585. IEEE, 2022.
- [16] Philipp Götz, Cagdas Tuna, Andreas Walthner, and Emanuël AP Habets. Contrastive Representation Learning for Acoustic Parameter Estimation. In *International Conference on Acoustics, Speech and Signal Processing*, pages 1–5. IEEE, 2023.
- [17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Conference on Computer Vision and Pattern Recognition*, pages 770–778. IEEE, 2016.
- [18] Calum Heggan, Sam Budgett, Timothy Hospedales, and Mehrdad Yaghoobi. Metaaudio: A Few-Shot Audio Classification Benchmark. In *International Conference on Artificial Neural Networks*, pages 219–230. Springer, 2022.
- [19] Christopher Ick, Adib Mehrabi, and Wenyu Jin. Blind Acoustic Room Parameter Estimation Using Phase Features. In *International Conference on Acoustics, Speech and Signal Processing*, pages 1–5. IEEE, 2023.
- [20] Marco Jeub, Magnus Schafer, and Peter Vary. A Binaural Room Impulse Response Database for the Evaluation of Dereverberation Algorithms. In *International Conference on Digital Signal Processing*, pages 1–5. IEEE, 2009.
- [21] Peter Kabal. TSP Speech Database. *McGill University, Database Version*, 1(0):09–02, 2002.
- [22] Diederik P Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *3rd International Conference for Learning Representations*, 2015.
- [23] Keisuke Kinoshita, Marc Delcroix, Takuya Yoshioka, Tomohiro Nakatani, Emanuel Habets, Reinhold Haeb-Umbach, Volker Leutnant, Armin Sehr, Walter Kellermann, Roland Maas, et al. The REVERB Challenge: A Common Evaluation Framework for Dereverberation and Recognition of Reverberant Speech. In *Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 1–4. IEEE, 2013.
- [24] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *Conference on Computer Vision and Pattern Recognition*, pages 11976–11986. IEEE/CVF, 2022.
- [25] B Logan. Mel Frequency Cepstral Coefficients for Music Modelling. In *Proc. International Symposium on Music Information Retrieval*, 2000.
- [26] Manuel Milling, Andreas Triantafyllopoulos, Iosif Tsangko, Simon David Noel Rampp, and Björn Wolfgang Schuller. Bringing the Discussion of Minima Sharpness to the Audio Domain: A Filter-Normalised Evaluation for Acoustic Scene Classification. In *International Conference on Acoustics, Speech and Signal Processing*, pages 391–395. IEEE, 2024.
- [27] Alastair H Moore, Mike Brookes, and Patrick A Naylor. Room Geometry Estimation From a Single Channel Acoustic Impulse Response. In *21st European Signal Processing Conference*, pages 1–5. IEEE, 2013.
- [28] Alastair H Moore, Mike Brookes, and Patrick A Naylor. Roomprints for Forensic Audio Applications. In *Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 1–4. IEEE, 2013.
- [29] Alastair H Moore, Mike Brookes, and Patrick A Naylor. Room Identification Using Roomprints. In *54th International Conference: Audio Forensics*. Audio Engineering Society, 2014.
- [30] Alastair H Moore, Patrick A Naylor, and Mike Brookes. Room Identification Using Frequency Dependence of Spectral Decay Statistics. In *International Conference on Acoustics, Speech and Signal Processing*, pages 6902–6906. IEEE, 2018.
- [31] Denise Moussa, Germans Hirsch, Sebastian Wankerl, and Christian Riess. Point to the Hidden: Exposing Speech Audio Splicing via Signal Pointer Nets. In *Proc. INTERSPEECH 2023*, pages 5057–5061, 2023.
- [32] Damian T Murphy and Simon Shelley. OpenAIR: An Interactive Auralization Web Resource and Database. In *129th Audio Engineering Society Convention*. Audio Engineering Society, 2010.
- [33] Arsha Nagrani, Joon Son Chung, and Andrew Senior. VoxCeleb: A Large-Scale Speaker Identification Dataset. *INTERSPEECH*, pages 2616–2620, 2017.
- [34] Konstantinos Papayiannis, Christine Evers, and Patrick A Naylor. Discriminative Feature Domains for Reverberant Acoustic Environments. In *International Conference on Acoustics, Speech and Signal Processing*, pages 756–760. IEEE, 2017.
- [35] Konstantinos Papayiannis, Christine Evers, and Patrick A Naylor. End-to-end Classification of Reverberant Rooms Using DNNs. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:3010–3017, 2020.
- [36] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019.
- [37] Nils Peters, Howard Lei, and Gerald Friedland. Name That Room: Room Identification Using Acoustic Features in a Recording. In *20th ACM International Conference on Multimedia*, pages 841–844, 2012.

- [38] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, and Piotr Dollár. Designing Network Design Spaces. In *Conference on Computer Vision and Pattern Recognition*, pages 10428–10436. IEEE/CVF, 2020.
- [39] Davide Salvi, Mirco Pezzoli, Sara Mandelli, Paolo Bestagini, and Stefano Tubaro. Are you Really Alone? Detecting the use of Speech Separation Techniques on Audio Recordings. In *International Workshop on Information Forensics and Security*, pages 1–6. IEEE, 2023.
- [40] Robin Scheibler, Eric Bezzam, and Ivan Dokmanić. Pyroomacoustics: A Python Package for Audio Room Simulation and Array Processing Algorithms. In *International Conference on Acoustics, Speech and Signal Processing*, pages 351–355. IEEE, 2018.
- [41] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical Networks for Few-Shot Learning. *Advances in Neural Information Processing Systems*, 30, 2017.
- [42] Mingxing Tan and Quoc Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.
- [43] TorchAudio. torchaudio.transforms — TorchAudio 0.10.0 Documentation, accessed: 2023-05-25. <https://pytorch.org/audio/0.10.0/>.
- [44] James Traer and Josh H McDermott. Statistics of Natural Reverberation Enable Perceptual Separation of Sound and Space. *Proceedings of the National Academy of Sciences*, 113(48):E7856–E7865, 2016.
- [45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All You Need. *Advances in Neural Information Processing Systems*, 30, 2017.
- [46] Yu Wang, Nicholas J Bryan, Mark Cartwright, Juan Pablo Bello, and Justin Salamon. Few-Shot Continual Learning for Audio Classification. In *International Conference on Acoustics, Speech and Signal Processing*, pages 321–325. IEEE, 2021.
- [47] Piper Wolters, Chris Careaga, Brian Hutchinson, and Lauren Phillips. A Study of Few-Shot Audio Classification. *GHC*, 2020.
- [48] Ziyue Xiang, Paolo Bestagini, Stefano Tubaro, and Edward J Delp. Forensic Analysis and Localization of Multiply Compressed MP3 Audio Using Transformers. In *International Conference on Acoustics, Speech and Signal Processing*, pages 2929–2933. IEEE, 2022.
- [49] Weidi Xie, Arsha Nagrani, Joon Son Chung, and Andrew Senior. Utterance-Level Aggregation for Speaker Recognition in the Wild. In *International Conference on Acoustics, Speech and Signal Processing*, pages 5791–5795. IEEE, 2019.
- [50] Shilei Zhang, Yong Qin, Kewei Sun, and Yonghua Lin. Few-Shot Audio Classification with Attentional Graph Neural Networks. In *Interspeech*, pages 3649–3653, 2019.
- [51] James Zjalic. *Digital Audio Forensics Fundamentals: From Capture to Courtroom*. CRC Press, 2020.