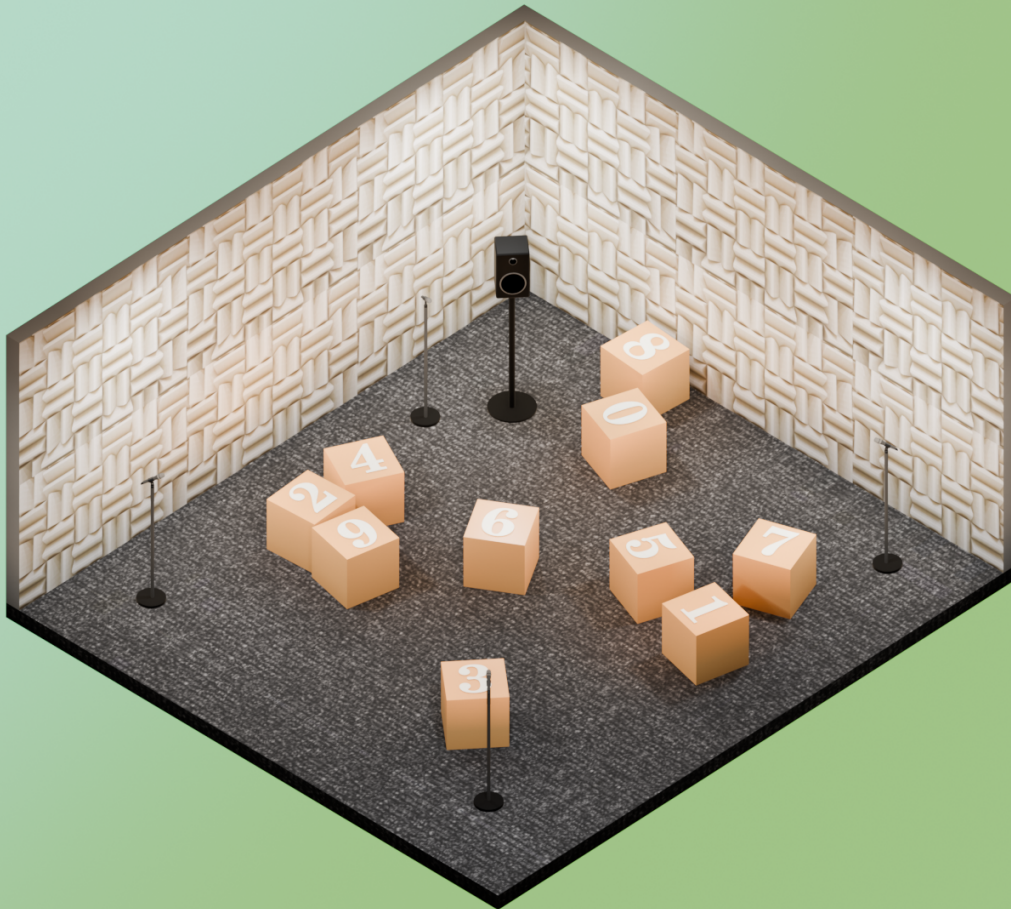


Change Detection of 3D Indoor Scenes using only Acoustic Sensors

MSc Thesis Report



by
Menthe Steensma
(4941381)

University Mentor:	J.F.P. Kooij (TU Delft)
Internship Supervisor:	S. Hornauer (Mines Paris - PSL)
Project Duration:	June, 2025 - March, 2026
Faculty:	Faculty of Mechanical Engineering, Delft

Change Detection of 3D Indoor Scenes using only Acoustic Sensors

Menthe Steensma

Faculty of Mechanical Engineering, TU Delft
Delft, The Netherlands

Abstract— Detecting changes in indoor environments is essential for applications such as automated inventory monitoring. We address this problem using only active acoustic sensing: a speaker emits a known signal, distributed microphones record the sound, the Room Impulse Responses (RIRs) are obtained through deconvolution, and a neural network determines which objects have been added or removed. We address two RIR-based tasks for passive objects: object count regression, which estimates the number of silent objects present in a room from multi-channel RIRs, and multi-label binary change detection, which predicts per-object presence changes between two observations. To the best of our knowledge, neither task has been previously addressed in the acoustic domain for passive objects. Both tasks are solved using a VGGish-based architecture operating on log-Mel spectrograms. To overcome the scarcity of real-world training data, we develop a simulation pipeline using SoundSpaces 2.0 and Blender to generate RIRs for arbitrary room configurations. We validate our approach through four incremental experiments. First, we reproduce a localization baseline on real RIRs from the SoundCam dataset, achieving a mean error of 15.5 cm with ten microphones at 48 kHz. Second, we quantify the sim-to-real domain gap, finding that in-domain simulated performance (18.2 cm) approaches the real baseline but cross-domain transfer fails, with errors exceeding the random-guess baseline. Third, in simulation, object count regression achieves a mean error of 0.38 boxes within the training distribution, well below the random baseline of 1.23, though extrapolation to unseen counts degrades (mean error of 0.67 boxes). Fourth, the binary change detection model correctly identifies changes for nine out of ten object locations, achieving a mean per-box accuracy of 0.935, well above the 0.5 random baseline. Direct-path occlusion reduces mean accuracy from 0.935 to 0.840, with failures concentrated at spatially clustered locations. Models trained on small change events (at most three simultaneous changes) generalize meaningfully to larger change magnitudes, retaining AUC above 0.9 for seven of ten locations. These results provide initial evidence for the feasibility of acoustic change detection for multiple silent objects, while highlighting the sim-to-real gap as the primary barrier to practical deployment.

I. Introduction

Automated inventory monitoring in warehouse environments requires knowing which boxes have been added or removed in stock locations between observations without relying on manual scanning. Periodic manual audits are labor-intensive, disruptive to operations [1], and performed only at fixed intervals, leaving discrepancies undetected for days or weeks. Drone-based audit systems can reduce manual effort but face safety, privacy, and regulatory constraints that currently limit their adoption [1], [2]. Camera-based approaches

are sensitive to the occlusion, dust, and variable lighting conditions common in dense storage environments.



Fig. 1: Possible warehouse setting [3].

Active acoustic sensing offers a complementary approach. Unlike passive acoustic methods, which depend on objects emitting sound, active sensing encodes object presence in the reflections of a known excitation signal, making it applicable to silent objects such as warehouse stock. A speaker and a set of microphones permanently installed at known positions emit and record a known excitation signal. The Room Impulse Response (RIR), obtained by deconvolving the recording with the excitation, characterizes how sound propagates through the room, encoding the direct path and all reflections off walls, floors, and objects [4]. Because stock locations are fixed, inventory changes manifest as the presence or absence of reflective objects at known positions rather than as arbitrary motion. The objects are passive and silent, so standard acoustic event detection does not apply. Changes are instead encoded in how the RIR differs between two observations.

This framing defines a specific task that has not been previously addressed: *multi-label binary change detection*, in which a model receives two sets of multi-channel RIR observations and outputs, for each known object location, whether an object was added or removed. This goes beyond scene-level binary detection (has anything changed?), single-object localization (where is one object?), and counting (how many objects are present?), all of which have been studied

in the acoustic literature, though primarily for sound-emitting subjects. Per-object change detection for multiple silent objects at fixed positions remains an open problem.

A key challenge is data availability. Systematically recording RIRs across all possible object presence/absence combinations is combinatorially infeasible at scale: ten binary object locations produce 1,024 configurations, each requiring a separate physical measurement. Acoustic simulation provides a scalable alternative, but simulated RIRs inevitably differ from real measurements, and the extent and character of this domain gap must be understood before simulation-trained models can be deployed on real data.

Our work addresses three research questions:

- RQ1: To what extent can models trained on simulated RIRs, generated using SoundSpaces 2.0, match the performance of models trained on real acoustic measurements, both within-domain and in cross-domain transfer?
- RQ2: Can a learned model infer the number of objects present in a room from a single multi-channel RIR observation, and how does performance degrade for counts outside the training distribution?
- RQ3: Can per-object presence changes be detected across multiple silent objects simultaneously from pairs of multi-channel RIR observations, and how do factors such as direct-path occlusion and the number of simultaneous changes affect detection performance?

The remainder of this work is organized as follows. Section II reviews related work on acoustic change detection, available datasets, and acoustic simulators. Section III presents the proposed approach, covering the task formulation, data preprocessing, model architecture, and training configuration. Section IV describes the experimental validation across four incremental experiments. Section V discusses the results, limitations, and directions for future work. Finally, Section VI concludes the report.

II. Related Work

We review prior work across three areas: existing approaches to acoustic change detection, the datasets available for this problem, and acoustic simulation platforms.

A. Acoustic Change Detection

We decompose acoustic change detection into several sub-tasks of increasing complexity: binary detection, counting and volume estimation, localization, and multi-object detection.

1) *Binary detection*: The simplest acoustic change detection task is determining whether a scene has changed at all. In the context of room acoustics, this has been studied primarily as occupancy detection. Model-based approaches compare signal energy recorded in a known empty room against a test recording and apply a predefined threshold [5], [6], [7]. These methods are limited to line-of-sight configurations and require a stable empty-room reference. Machine learning alternatives remove the need for manual thresholds: Alanwar et al. [8] use a random forest classifier, and Pham et al. [9] train an autoencoder on empty-room samples to flag anomalies. Most

relevant to our work, Wang et al. [10] adopt VGGish [11], an audio-adapted VGG architecture [12], and show that a multi-channel 48 kHz model outperforms the single-channel pretrained baseline. Importantly, the authors report that binary occupancy detection with an active source was “trivial”, suggesting the task is saturated and motivating the harder per-object formulation. A shared limitation of all these methods is that they output a single scene-level label: they cannot identify what changed, where, or by how much.

2) *Counting and volume estimation*: Extending binary detection to quantitative estimation, Hwang et al. [13] and Kim et al. [14] recover volumetric object information from ultrasonic echoes using U-Net-like architectures, and Liang et al. [15] regress room dimensions from binaural Mel spectrograms. These works confirm that acoustic signals encode spatial extent information, but none infers the count of silent objects within a room from audible-frequency RIRs, which is the task we address in Experiment 3.

3) *Localization*: Shibata et al. [16] reconstruct human pose from chirp echoes at approximately 30 cm accuracy, and Wang et al. [10] achieve similar accuracy using multi-channel VGGish, both in single-room settings. Lawrence [17] proposes a differential RIR approach, but target human subjects only. All localization methods address a single human, not a set of static objects.

4) *Multi-object detection*: Counting people via speech separation [18], [19], [20], [21] and multi-source localization [22] relies on the objects emitting sound. Shih et al. [23] correlate reverberation time with occupancy count using ultrasonic chirps, but produce only a count without identifying affected locations. We are not aware of prior work that performs per-object change detection across multiple silent objects at known fixed positions, nor infers object count from audible-frequency RIRs for passive objects. We address both gaps: a regression model that estimates object count from multi-channel RIRs (Experiment 3), and a multi-label binary classifier that predicts per-object presence changes between two RIR observations (Experiment 4).

B. Datasets

Two datasets contain RIRs from enclosed rooms with systematically varied object configurations. The Motus dataset [24] provides 3,320 ambisonic RIRs across 830 configurations of four furniture object types in a 60 m³ room. The SoundCam dataset [10] provides 3,800 RIRs from ten microphones across three rooms, with labels for binary detection, localization, and identity of a single human subject. Neither dataset provides the exhaustive presence/absence combinations needed to train a per-object change detection model of objects in a fixed location. This makes simulation the only scalable path to training data. We therefore develop a simulation pipeline using SoundSpaces 2.0 [25] and Blender [26] that generates RIRs for arbitrary room geometries and object configurations, and use it to produce all training data for Experiments 3 and 4. The resulting domain gap is analyzed systematically in Experiment 2.

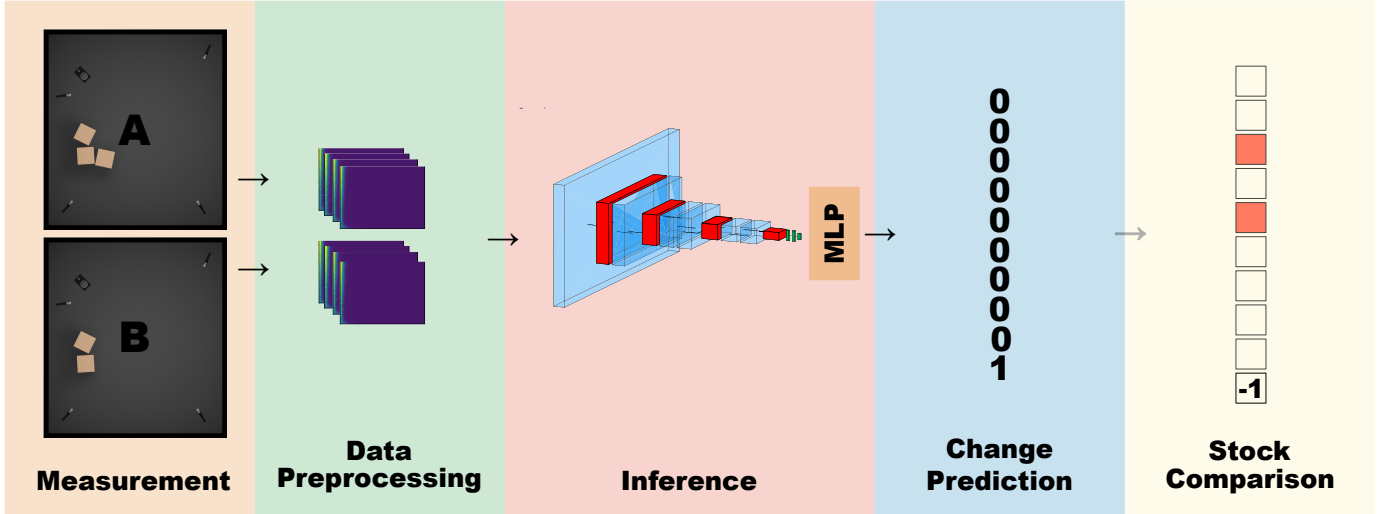


Fig. 2: Overview of the proposed binary change detection pipeline. Two configurations (A and B) of multi-channel RIRs are measured, converted to log-Mel spectrograms, and concatenated as input to the modified VGGish encoder followed by an MLP head. The model outputs a per-object binary change prediction vector, which can be compared against the digital stock records to flag discrepancies requiring manual verification.

C. Acoustic Simulation

Geometry-based simulators generate synthetic RIRs by modeling sound propagation on a room mesh. Pyroomacoustics [27], for example, implements the image-source method but is limited to shoebox room geometries. SoundSpaces 2.0 [25] uses ray tracing on arbitrary meshes via Habitat-Sim [28], supports configurable materials and source-receiver placements, and is the platform we adopt. Wave-based methods provide higher physical fidelity but are computationally prohibitive at room scale and high frequencies [29]. Hybrid approaches such as Treble [30] combine both the geometry- and wave-based method but are not open-source. Despite advances, discrepancies between simulated and real RIRs persist due to environmental factors, hardware imperfections, and modeling approximations [31]. We investigate this domain gap in Section IV-C.

III. Proposed Approach

This section presents the proposed method for active acoustic change detection. The model architecture and experimental setup build upon the localization implementation of SoundCam [10]. This section describes which components are reused and what adaptations are made for the tasks of object count regression and binary change detection. An overview of the pipeline is shown in Fig. 2.

A. Task Formulation

We consider an indoor room containing B objects at fixed positions. A speaker placed in a corner of the room emits a sine sweep, and m microphones distributed along the walls

record the arriving sound. The RIR for each microphone-speaker pair is obtained through deconvolution, yielding r RIRs per measurement ($r = m$ for a single speaker). Following the SoundCam implementation [10], log-Mel spectrograms computed from the RIRs serve as input to the model. This ensures consistency with the baseline we reproduce and aligns with the simulator, which outputs RIRs directly. Three tasks of increasing complexity are addressed. Localization validates that the architecture extracts spatial information from RIRs and provides a baseline for quantifying the sim-to-real gap [10] (RQ1). Object count regression extends the problem from a single object to multiple objects and tests whether multi-channel RIRs carry sufficient information to infer the number of objects present (RQ2). Binary change detection is the target task, requiring the model to jointly identify which of B objects have changed between two observations (RQ3). Each task builds on the preceding one: localization confirms spatial sensitivity, count regression confirms multi-object sensitivity, and change detection adds temporal comparison.

A learned approach is adopted rather than a handcrafted acoustic statistic. As shown in Fig. 3, RT60 values across all configurations do not separate cleanly by object count, confirming that a simple decay-time threshold is insufficient, and thus motivates a data-driven method.

For object count estimation, regression is preferred over classification because the count labels have ordinal structure that standard classification losses do not exploit [32], and because regression permits extrapolation beyond the observed label range in principle [33]. For change detection, a per-object binary output is preferred over a scalar count-difference

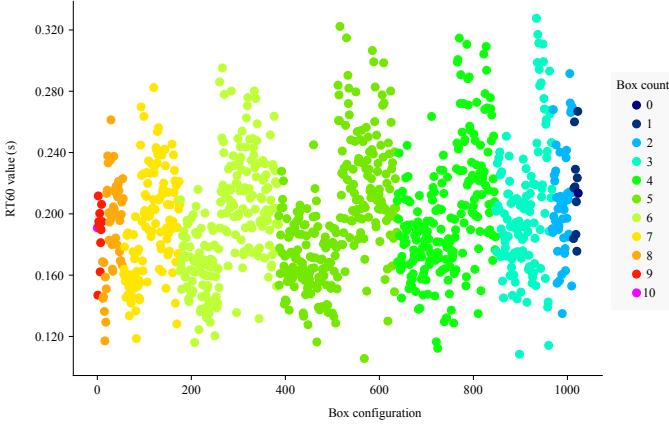


Fig. 3: RT60 values for the 1024 room configurations, grouped by the number of boxes present in the room. Note that the room is near-anechoic, yielding a different reflections from walls and present boxes than in a typical room.

output: the scalar formulation maps many distinct configuration pairs to the same target, causing severe overfitting (Section IV-D), and does not identify which locations are affected (Fig.4).

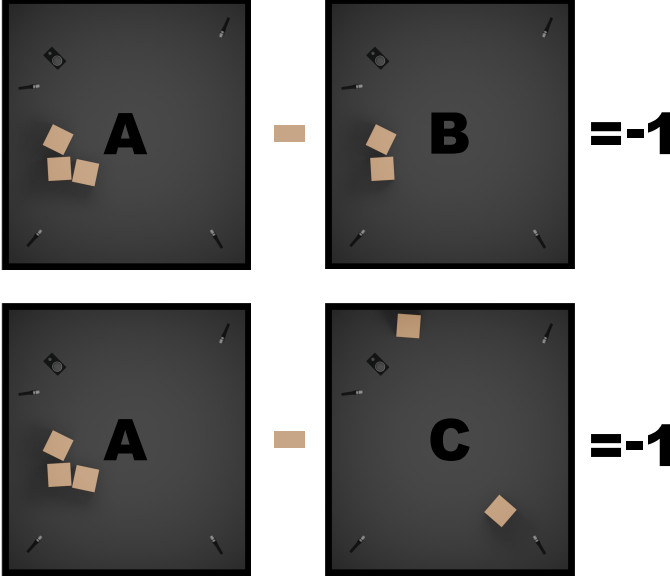


Fig. 4: Label ambiguity in pair-input object count difference formulation. The same configuration A compared with two very different situations B and C yield the same count difference.

1) *Localization Task*: The localization task takes as input the r log-Mel spectrograms from a single room state measurement and outputs the predicted xy-coordinates of one object in the room. This task is adopted from SoundCam [10], where a VGGish feature extractor followed by an MLP regresses the pelvis centroid of a human subject. We select localization rather than detection (binary classification of whether a person is present) for two reasons. First, detection requires empty-

room samples as negative examples. Since the simulator does not model the ambient noise present in real recordings, empty-room RIRs from the two domains would be trivially distinguishable, confounding any cross-domain evaluation. Second, localization provides a continuous error metric that offers finer-grained insight into model performance and the sim-to-real gap than a binary label would.

The loss function that is used for the localization is the mean squared error between the predicted and ground truth coordinates:

$$\mathcal{L}_{\text{loc}} = \frac{1}{N} \sum_{i=1}^N [(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2], \quad (1)$$

where $\hat{x}_i$ and $\hat{y}_i$ are the x- and y-coordinates of the i -th prediction, x_i and y_i are the corresponding ground truth coordinates, and N is the batch size.

2) *Object Count Regression Task*: The object count regression task takes the same input as localization (the r spectrograms from a single measurement) but outputs a single scalar: the predicted number of objects present in the room.

The VGGish architecture from the localization task is reused, with the output layer modified to produce a single scalar. The loss is the MSE between predicted and ground truth count:

$$\mathcal{L}_{\text{count}} = \frac{1}{N} \sum_{i=1}^N (\hat{C}_i - C_i)^2, \quad (2)$$

where $\hat{C}_i$ is the object count prediction for the i -th prediction, $C_i \in \{0, \dots, B\}$ is the corresponding ground truth label, and N is the batch size.

3) *Binary Change Detection Task*: The binary change detection model takes as input two sets of r spectrograms, corresponding to two measurements of the same room at different times, and outputs B binary predictions. For each object location $b \in \{1, B\}$, the ground truth label is $y_b = |p_b^{(1)} - p_b^{(2)}|$, where $p_b^t \in \{0, 1\}$ indicates whether object b is present in measurement t . A label of $y_b = 1$ indicates that object b was added or removed between the two observations. Because VGG-like architectures are not permutation-invariant [34], all ordered permutations of configuration pairs are used.

The model is trained using the binary cross-entropy loss with logits [35]:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{N} \sum_{i=1}^N [y_i \log(\sigma(z_i)) + (1 - y_i) \log(1 - \sigma(z_i))], \quad (3)$$

where z_i is the raw model output (logit) for the i -th prediction, $y_i \in \{0, 1\}$ is the corresponding ground truth label, N is the batch size, and $\sigma(z) = (1 + e^{-z})^{-1}$ is the sigmoid function. Binary predictions are obtained by thresholding $\sigma(z_i)$ at 0.5. The loss is computed per object location and averaged over all B locations and all samples in the batch. Each location is treated as a separate binary prediction, though all share the same encoder.

Data generation. Training data for the object count regression and binary change detection tasks are generated in simulation. A room containing B objects is modeled in Blender [26], with each object either present or removed, yielding $C = 2^B$ unique configurations. For each configuration, the room mesh is exported and RIRs are generated using SoundSpaces 2.0 [25]. The configurations are divided into training, validation, and test sets in an 80/10/10 ratio using stratified sampling [36] to maintain a balanced count distribution across splits. Input pairs for the binary change detection task are constructed from all ordered permutations of the C configurations, yielding $P = C \times (C - 1)$ pairs. The specific room geometry, object dimensions, material properties, and simulation parameters are detailed in Section IV.

B. Data Preprocessing

The raw RIR waveforms are transformed into log-Mel spectrograms before being input to the model. For the 48 kHz configuration, the Short-Time Fourier Transform (STFT) is computed [37] using a Hann window of 1200 samples with a hop size of 480 samples and an FFT length of 1200 samples. This produces a time-frequency representation with n_t time frames and K frequency bins. A Mel-spaced filter bank $\mathbf{M} \in \mathbb{R}^{n_f \times K}$ [38] with $n_f = 64$ filters spanning 125 to 22,500 Hz maps the frequency axis to perceptually motivated Mel bins, and a logarithmic compression is applied to obtain the log-Mel spectrogram of size $n_t \times n_f$. This preprocessing is applied independently to each of the m microphone channels, yielding r spectrograms per room state measurement. Fig. 5 shows an example of a real and simulated RIR log-Mel spectrogram that was created with the settings as specified.

16 kHz vs. 48 kHz. The SoundCam paper evaluates both a 48 kHz and a downsampled 16 kHz version, with the 48 kHz multichannel version outperforming in all cases except the single-channel setting. However, the released code only provides the 16 kHz implementation, which uses a window of 400 samples, a hop size of 160 samples, and a frequency range of 125–7,500 Hz. We create a 48 kHz reproduction by extending the upper frequency limit to 22.5 kHz and proportionally scaling (factor of three) the window, hop size, and input length to preserve a 0.96 s temporal window, resulting in spectrograms of size 96×64 for both versions. Both configurations are evaluated in Experiments 1 and 2.

C. Model Architecture

The VGGish architecture from SoundCam is adopted as the starting point for all tasks. Among existing learned acoustic sensing models, SoundCam is the only work that goes beyond binary scene-level detection to perform localization and identification of silent, non-moving subjects from active RIRs. Since its authors reported that binary occupancy detection with an active source was “trivial” [10], we hypothesized that the same architecture might extend to multi-object change detection. The aim of this work is not to optimize the model architecture, but to establish the feasibility of per-object acoustic change detection and to provide the experimental framework

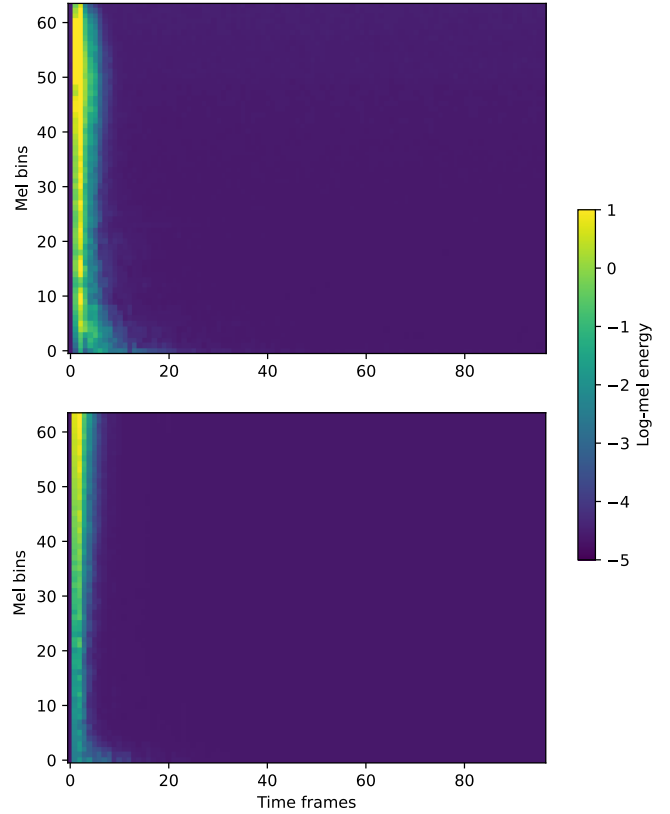


Fig. 5: Real RIR (left) and simulated RIR (right) for microphone 0, displayed as a log-Mel spectrogram of size 96×64 for a 48 kHz signal.

needed for future research. Architectural comparisons, such as evaluating lighter or attention-based alternatives, are left for future work once the feasibility and key challenges have been established. The VGGish model consists of three components: a feature extractor, an optional fusion stage, and a task-specific prediction head. For the localization task and object count regression we directly reuse model from SoundCam. For the binary change detection, we perform certain modifications that we describe below and can be seen in Fig. 6.

1) VGGish Feature Extraction: The feature extractor is based on VGGish [11], an adaptation of VGG [12] for audio. Compared to the original VGG, VGGish reduces the input dimensions from 224×224 to 96×64 to accommodate log-Mel spectrograms, removes the last convolutional and max-pooling layers, and compresses the output embedding to 128 dimensions.

For the localization and object count regression tasks, the VGGish feature extractor is used as described above. For binary change detection, two architectural modifications are applied. First, group normalization [40] is necessary after each convolution-ReLU pair, to stabilize training across the heterogeneous per-box prediction tasks and to avoid gradient explosion. Second, the weights are initialized using the Kaiming scheme [41], for faster convergence.

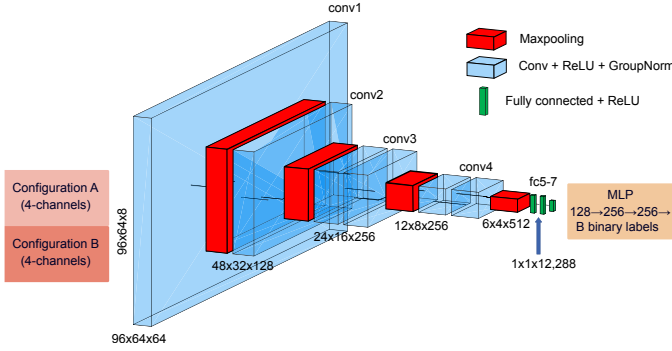


Fig. 6: Modified VGGish architecture for binary change detection. The feature extractor processes $2r=8$ concatenated log-Mel spectrograms (early fusion of two 4-microphone configurations) through four convolutional blocks with GroupNorm, producing a 12,288-dimensional flattened representation. This is projected to a 128-d embedding via fully connected layers, then passed to an MLP that outputs B per-object change logits. Adapted from [39].

2) *Input Fusion*: Two levels of fusion are relevant: microphone channel fusion (combining the r spectrograms from one measurement) and configuration pair fusion (combining two measurements for the change detection task).

For microphone channel fusion, two strategies are evaluated [42]. In early fusion, the r spectrograms are concatenated along the channel dimension before feature extraction, allowing the model to learn cross-channel relationships (e.g., inter-channel amplitude and time-of-arrival differences) directly. In late fusion, each channel is processed independently through the same shared-weight feature extractor, and the resulting embeddings are concatenated before the prediction head. Early fusion is adopted for consistency through experiments, as both fusion methods perform similarly.

For configuration pair fusion, which applies only to the binary change detection task, early and late fusion are both considered as well. Early fusion here means that the r spectrograms of both configurations are concatenated before feature extraction, leading to an input of $2r$ for the VGGish. Late fusion on the other hand processes the r spectrograms for both configurations independently through the shared-weight feature extractor, and concatenates the embeddings before the prediction head. Early fusion is adopted, as late fusion with shared weights collapses: the encoder converges to identical embeddings for both inputs, eliminating the change signal (Appendix C-H1).

3) *Prediction Head*: The VGGish encoder produces a flattened representation of 12,288 dimensions, which is projected to a 128-dimensional embedding through two fully connected layers (4096 and 4096 units, both followed by ReLU). The prediction head is an MLP with two hidden layers of 256 units each with ReLU activations, followed by a linear output layer. The output dimensionality varies by task: two values (xy-coordinates) for localization, one value (count) for regression,

and B values (per-object logits) for binary change detection.

D. Training Strategy

All models are trained with the Adam optimizer at a learning rate of 10^{-4} . The localization models use a batch size of 32, following the original SoundCam configuration, while for the count regression and change detection we increase the batch size to 64. We replace the fixed 1000-epoch schedule with early stopping to avoid unnecessary computation and reduce the risk of overfitting. We tune the patience to 100 for Experiment 1 and 2, 50 for Experiment 3, and 10 for Experiment 4. For reproducibility, all random number generator seeds are fixed, covering data shuffling, weight initialization, and CUDA operations. The original SoundCam implementation loaded all waveforms into memory simultaneously, which we replace with a PyTorch DataLoader and memory-mapped arrays to reduce memory usage. We use stratified sampling for the box configuration train/val/test split for Experiment 3 and 4, in an 80/10/10 manner. For the binary change detection baseline (Experiment 4), the full permutation set produces a balanced label distribution: across all pairs, each box changes in approximately half the cases. In the generalization sub-experiment, training is restricted to pairs with at most three simultaneous changes, which introduces a 3:1 imbalance favoring the no-change class. A loss-reweighted variant is therefore also evaluated (Section IV-E4).

IV. Experimental Validation

Four experiments are performed, progressing incrementally from localization on real data to binary change detection on simulated data. We first train and evaluate on the publicly available SoundCam RIR dataset, then create a simulated replica of the same scene to assess the domain gap. Exploiting the configurability of the simulator, we subsequently construct a novel scenario with multiple objects to perform object count regression and binary change detection. An overview of the experiments is provided in Table I.

A. Evaluation Setup

We describe the real and simulated RIR datasets used in this work and define the evaluation metrics for each task. Additionally, we introduce random baselines for comparison, especially as the object count regression and binary change detection tasks have not been previously addressed.

1) *Real Dataset*: The SoundCam dataset [10] is designed for human detection and localization in indoor environments. It contains RIRs from three rooms with different acoustic properties. For this work, the Treated Room is selected because it is an enclosed, empty, rectangular space ($4.9 \times 5.1 \times 2.7$ m) with defined acoustic materials, making it the most suitable candidate for simulation replication. The walls of the Treated Room are lined with melamine wedge panels (15 cm thick, Noise Reduction Coefficient (NRC) of 1.26). The ceiling consists of acoustic fiberglass tiles (NRC of 1.0), and the floor is covered with standard office carpet. Due to the highly absorptive wall treatments, the RT60 of the room is low

	Data Type	Freq. (kHz)	Mic. Channels	Object of Interest
Baseline Validation	Only real	16, 48	10, 4, 2, 1	1 human
Baseline Simulation	Real and sim	48	10, 4, 2, 1	1 human
Object Count Regression	Only sim	48	4	0–10 boxes
Binary Change Detection	Only sim	48	4	0–10 boxes

Table I: Overview of Experiments.

(Fig. 7), meaning that reflections die out quickly and the RIR is dominated by the direct sound and early reflections.

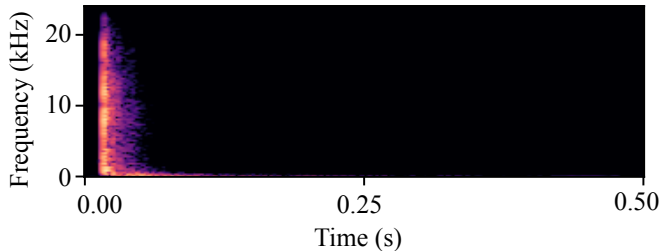


Fig. 7: RIR of the SoundCam Treated Room for microphone 1. The near-anechoic wall and ceiling materials reduce the reflections, causing fast sound decay and sound dying out after ~ 20 ms. Reprinted from [10].

A speaker is positioned in one corner at a height of 1.2 m, and ten microphones are distributed along the walls at the same height. Fig. 8 shows the physical setup. The selected part of the dataset comprises 1000 RIRs, each obtained by deconvolving a 10-second logarithmic sine sweep (20 Hz–24 kHz) recorded while a human subject stood at a specific location. An additional 4 seconds of the sound decay are recorded after each sweep. For each recording, the pelvis location is measured using three Azure Kinect DK RGB-D cameras, and the median of these measurements is stored as the ground truth centroid for the localization task. In all recordings, the subject stands upright with arms at their sides, facing a fixed direction.

2) *Simulated Dataset*: To generate simulated RIRs, a replica of the SoundCam Treated Room is constructed in Blender [26] and rendered using SoundSpaces 2.0 [25], a ray-tracing-based acoustic simulator built on top of Habitat-Sim [28]. The room geometry is specified as a mesh file, with distinct material identifiers assigned to the walls, floor, ceiling, and objects of interest. Sound sources and receivers are omnidirectional, and are modeled as massless point entities. Since SoundSpaces 2.0 does not support multiple distributed microphones in a single simulation pass, each microphone is simulated individually. The simulation uses a sampling rate of 48 kHz, a direct ray count of 50,000, an indirect ray count of 30,000, and six frequency bands. Material coefficients in the literature are specified only up to 4 kHz, and values for higher bands are set equal to those of the highest defined band [43]. Three acoustic properties are defined per surface: absorption α , scattering s , and transmission τ . Due to the limited availability of scattering and transmission data for the Treated Room



Fig. 8: The SoundCam Treated Room, reprinted from [10]. Visualizations of the microphone configurations used in this work are provided in Appendix A.

materials, values from the Matterport3D example implementation are reused [43], [44]. Absorption values are obtained from published measurement data. The full simulation parameter settings and material specifications are provided in Appendix B. For the localization experiments (Experiment 2), a human mesh is placed at the same 1000 centroid locations as in the real dataset, using the same train/val/test split for fair comparison. For the object count regression and change detection experiments (Experiments 3 and 4), the human is replaced with ten cubic boxes ($0.5 \times 0.5 \times 0.5$ m each), positioned by physics-based dropping in Blender to produce a fixed starting configuration (Fig. 9). Each box is either present or removed, yielding $C = 2^B = 1024$ unique configurations. The box material uses the Matterport3D “plastic” acoustic properties. Since the speaker and microphones are mounted at 1.2 m, the boxes do not obstruct the direct sound path. The configurations are divided into training, validation, and test sets with an 80/10/10 ratio using stratified sampling [36] to maintain a consistent count distribution across splits. The ground truth labels are the integer counts in $\{0, \dots, B\}$.

3) *Evaluation Metrics*: For the localization task (Experiments 1 and 2), the mean Euclidean distance between predicted and ground truth centroid coordinates is used, reported as mean (standard deviation) in cm. For object count regression (Experiment 3), the continuous object count predictions are evaluated using the mean absolute error (MAE) in boxes. Per-count confusion matrices are additionally reported after rounding the predictions to the nearest integer to obtain

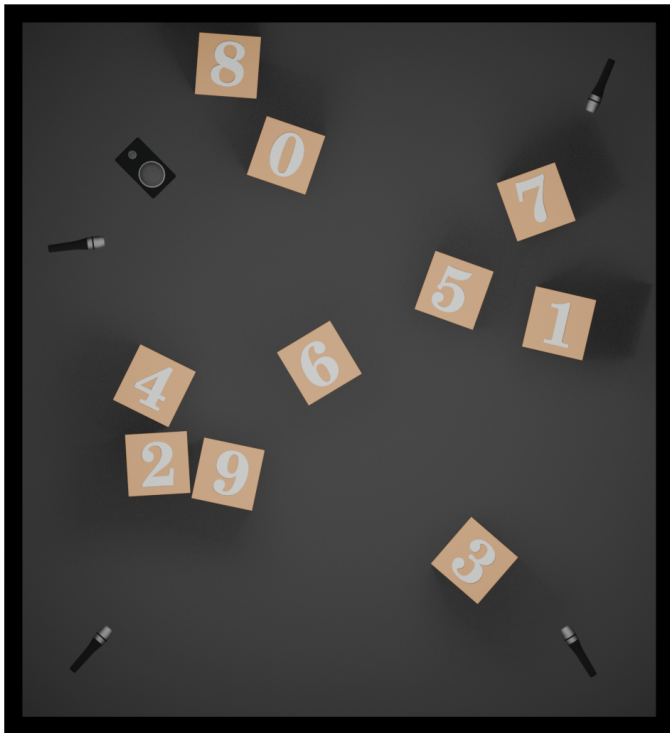


Fig. 9: Starting configuration with $B=10$ boxes for the object count regression and binary change detection tasks (Experiment 3 and 4). The speaker and microphone are added for context, but are not included in the mesh file that SoundSpaces uses for the RIR simulation. SoundSpaces models the speaker and microphone as omnidirectional, massless point entities.

classes. For binary change detection (Experiment 4), the primary metric is per-box accuracy. ROC-AUC curves are reported for the generalization sub-experiment.

4) *Random Baselines*: For localization, a simple baseline is to always predict the training set’s mean, in this case corresponding to approximately the room’s center coordinates (see Fig.15 in Appendix A). This constant predictor yields a mean Euclidean distance of 136.0 cm. We include this baseline to contextualize localization error magnitudes, while the primary comparison for localization with the simulated data remains with the reproduced SoundCam results. For object count regression, the optimal constant predictor is the median count [45], here equal to the mean count of 5. Given the binomial distribution of object counts with variance $\sigma_N^2 = 2.5$, this baseline yields an expected MAE of approximately 1.23 (see Appendix D for the derivation and data distribution). This serves as a reference point, since no prior results exist for this novel task. For binary change detection, a random predictor assigns each box a label of 0 or 1 with equal probability. For a balanced dataset, this results in an expected per-box accuracy of 0.5 and provides a baseline for comparison in the absence of previous results. For ROC-AUC, the random baseline is likewise 0.5.

B. Experiment 1: Baseline Validation and Reproduction

First, we reproduce the human localization task on the SoundCam real dataset. Table II shows the localization results for the 16 kHz and 48 kHz versions. The results slightly differ from the SoundCam published results, due to random-generated-number relying operations as the training shuffling, weight initialization, and CUDA operations. Nonetheless, all configurations outperform the random baseline of 136.0 cm, and a general trend is observed: performance improves with the number of microphone channels, with similar incremental gains across configurations.

1) *Anomaly Analysis and Seed Sensitivity*: There are two values that deviate from the expected trend: the original 48 kHz single-channel result and the reproduced 16 kHz two-channel result. In our reproduction, the 48 kHz single-channel model consistently outperforms its 16 kHz counterpart (mean error of 39.4 cm vs. 50.1 cm), unlike the original results (mean error of 71.6 cm vs. 49.4 cm). For the second anomaly, we assess the influence of random seed selection on the model performance. All seeds in the training pipeline are fixed, and training is repeated five times with different seed values. As shown in Table III, the mean error is 25.72 cm with a seed-to-seed standard deviation of ± 3.62 cm. This variability is comparable in magnitude to the inter-configuration performance differences, suggesting that seed variability may contribute to the discrepancy. In all subsequent experiments, we keep the seed fixed for reproducibility.

Seed	Error
<i>Reproduced VGGish, 4-channel (16 kHz)</i>	
1	25.8 (19.6)
2	23.4 (19.7)
3	27.5 (21.5)
4	21.3 (16.3)
1337	30.6 (29.8)
Across Seeds	25.72 (3.62)

Table III: Experiment 1.1 - Seed Sensitivity. Each entry reports mean localization error in cm, with within-run standard deviation in parentheses. The final row summarizes performance across seeds ($n=5$) as mean (std).

2) *Microphone Channel Fusion*: The effect of microphone channel fusion is assessed by comparing early and late fusion of the input RIRs. Table IX in Appendix C shows no significant performance difference between the two strategies. For consistency across all subsequent experiments, early fusion is used, following the original code implementation. Experiment 1 confirms that the reproduced VGGish model achieves localization accuracy comparable to the published SoundCam results, with the 48 kHz multichannel configuration performing best (15.5 cm with ten microphones). The seed sensitivity analysis reveals that weight initialization alone accounts for performance variations of up to ± 3.62 cm, a magnitude comparable to the differences between microphone configurations, underscoring the importance of fixed-seed eval-

<i>Model</i>	Number of Microphones			
	10	4	2	1
<i>VGGish (16 kHz)</i> [10]	18.6 (15.4)	21.8 (16.1)	26.8 (23.1)	49.4 (54.3)
<i>Reproduced VGGish (16 kHz)</i>	19.8 (17.4)	30.6 (29.8)	27.7 (20.9)	50.1 (50.6)
<i>VGGish (48 kHz)</i> [10]	18.1 (13.1)	17.2 (16.2)	20.4 (15.4)	71.6 (50.4)
<i>Reproduced VGGish (48 kHz)</i>	15.5 (13.3)	18.7 (18.2)	21.4 (19.8)	39.4 (40.6)
Random baseline	136.0			

Table II: Experiment 1 - Baseline Validation and Reproduction. Localization errors are given as mean (std) in cm. The results show a general trend of improved performance with more microphones.

uation in all subsequent experiments. Early and late microphone fusion perform similarly, and early fusion is adopted for consistency. These results establish the 48 kHz reproduced model as the reference for Experiment 2.

C. Experiment 2: Baseline Simulation

Moving to simulation, we retain the same training configuration as in Experiment 1, but replace the real RIRs with simulated RIRs. To ensure a controlled comparison, the ground-truth centroid locations as well as the train/validation/test splits remain identical. The reproduced 48 kHz results from Table II are used as the real-data reference, and a random-guess baseline of 136.0 cm is included for context. We evaluate performance in both within-domain and cross-domain settings, and motivate design choices for the object count regression and binary change detection tasks.

1) *Within-Domain Performance and Room Asymmetry*: The within-domain results (Table IV) show that models trained and tested on simulated RIRs (*Sim* $\rightarrow$ *Sim*) are consistently outperformed by their real-data counterparts (*Real* $\rightarrow$ *Real*), with the performance gap increasing as the number of microphones decreases. The largest degradation is in the single-channel case (118.0 cm vs. 39.4 cm), where the model collapses onto two lines (see Fig. 16 in Appendix C). We hypothesize that this behavior arises because a single RIR measurement provides limited spatial information: while it captures arrival times and amplitudes, it does not provide sufficient cues to uniquely determine direction of sound due to the absence of relative timing information. As a result, multiple human locations can produce similar RIRs at a single observation point. This ambiguity is magnified in nearly symmetric environments, such as the Treated Room replica, where mirror positions yield similar propagation distances and reflection patterns, leading to nearly indistinguishable RIRs.

We analyze the effect of room asymmetry on the simulated within-domain performance. In the real room, asymmetries arise from variations in material properties, sensor directivity, and the presence of measurement equipment, which may introduce additional spatial cues. To test whether such asymmetry improves localization in simulation, we introduce a localized perturbation in the simulated environment by placing an absorptive block near the farthest microphone, mimicking an object present in the real setup. The modified geometry and the corresponding results are reported

in Fig.17, Fig.16 and Table X in Appendix C. Contrary to expectations, the introduction of this asymmetry does not lead to consistent improvements in localization performance. The multi-channel configurations (2 and 4 microphones) show only minor changes that are small relative to the observed variance. The single-channel setting collapses into a single location prediction.

2) *Cross-Domain Performance and RIR Modification*: In the cross-domain setting, neither training on real data and testing on simulated data (*Real* $\rightarrow$ *Sim*) nor the reverse (*Sim* $\rightarrow$ *Real*) generalizes effectively. In most configurations, the resulting errors exceed the random-guess baseline, indicating a substantial domain mismatch. Notably, the *Sim* $\rightarrow$ *Real* condition exhibits particularly large errors (e.g., 427.6 cm for 10 microphones). Overall, these results highlight a significant domain gap between simulated and real RIRs, which limits both within-domain performance in simulation and cross-domain generalization.

To better understand the domain gap, we compare real and simulated RIRs. Fig. 18 in Appendix C shows an example from microphone 1, of a human in a position that does not block the direct path to the speaker. A clear difference is observed in the timing of the first prominent peak, which corresponds to the direct sound arriving at the microphone. For a source–receiver distance of 5.1 m and a sound speed of 343 m/s [46], the expected arrival time is approximately 14.9 ms. The simulated RIR exhibits an initial peak at 14.6 ms, consistent with this estimate. In contrast, the first peak in the real RIR appears approximately 2.7 ms later. This shift may arise from hardware-induced delays between speaker activation and the sound actually leaving the speaker, or preprocessing steps that suppress the direct-path component. The first peak is sometimes removed when the goal is to study the reflection structure of the RIRs independent from the direct sound [6]. To test whether these differences contribute to the domain gap, we apply to waveform level modifications to the simulated RIRs: 1) zero-padding to introduce a 2.7 ms delay, and 2) removal of the first peak. As shown in Table XI of Appendix C, these modifications do not improve cross-domain performance. In particular, *Sim* $\rightarrow$ *Real* errors remain high or increase.

3) *Object Geometry Influence*: We further evaluate the influence of object geometry on localization accuracy. We compare three shapes: a cuboid approximating human dimen-

Train Data → Test Data	Number of Microphones			
	10	4	2	1
<i>Reproduced VGGish (48 kHz)</i>				
Real → Real	15.5 (13.3)	18.7 (18.2)	21.4 (19.8)	39.4 (40.6)
Sim → Sim	18.2 (13.3)	28.3 (19.2)	37.0 (31.4)	118.0 (60.5)
Real → Sim	139.5 (66.4)	149.4 (64.1)	156.8 (75.7)	163.7 (75.9)
Sim → Real	427.6 (86.3)	186.8 (92.6)	140.0 (75.3)	159.1 (81.2)
Random baseline	136.0			

Table IV: Experiment 2 - Baseline in Simulation. Errors are given as mean (std) in cm.

sions, a human with arms spread out, and a human with arms at the sides. The results (Table XII) in Appendix C show that the cuboid yields the lowest error (13.4 cm in the 4-channel case). This experiment highlights that object geometry significantly influences the structure of simulated RIRs and, consequently, localization performance.

4) *Sampling Rate and Microphone Channel Selection:* Based on the results in Appendix C, the 48 kHz configuration is adopted for subsequent experiments. Although it shows a marginal within-domain disadvantage relative to 16 kHz (28.3 vs. 24.8 cm), this difference is not attributed to 48 kHz carrying less information. On real data, where the full acoustic response is physically present, 48 kHz consistently outperforms 16 kHz and is expected to better capture fine-grained spatial detail. A 4-channel microphone configuration is adopted for its efficiency-performance trade-off: it performs close to the 10-channel model on both real (18.7 vs. 15.5 cm) and simulated data (28.3 vs. 18.2 cm), while incurring substantially lower simulation cost.

Experiment 2 shows that in-domain simulated performance approaches the real-data baseline (18.2 vs. 15.5 cm with ten microphones), but that cross-domain transfer fails substantially in both directions, with errors exceeding the random-guess baseline in most configurations. Simple waveform-level corrections are not able to bridge the gap. Based on these results, the 48 kHz configuration and a 4-channel microphone arrangement are adopted for subsequent experiments, offering a favorable trade-off between spatial resolution and computational cost. The sim-to-real gap identified here remains the primary barrier to practical deployment.

D. Experiment 3: Object Count Regression

To extend from single-object localization to multi-object reasoning, we investigate whether the number of objects present in a room can be inferred from a single acoustic observation within the training distribution. We first evaluate a single-input regression model that predicts the total object count from one 4-channel measurement, demonstrating accurate interpolation within the training range while identifying saturation effects near the distribution boundaries. We then analyze a pair-input formulation that predicts the object count difference between two observations, and demonstrate that this formulation suffers from severe overfitting, motivating the binary change detection formulation introduced in Experiment 4.

1) *Single-input Object Count Regression:* The model is trained on configurations with object counts in the range $\{2, \dots, 8\}$. We evaluate generalization within this range and extrapolation to unseen counts $\{0, 1, 9, 10\}$. The latter is particularly relevant for real-world scenarios, where environments may not match the distribution observed during training.

a) *Within-range Count Labels:* On test configurations with counts in $\{2, \dots, 8\}$, the model achieves a MAE of 0.38 ± 0.47 , significantly outperforming the random baseline of 1.23. The prediction bias is small (-0.07), indicating a slight tendency to underestimate the number of boxes. Fig. 10 shows the confusion matrix over 100 test samples. Each row is normalized to percentages per true label. The results indicate that prediction accuracy degrades as the number of boxes increases. For higher counts, the model exhibits a systematic compression effect. In particular, for configurations with eight boxes, the model predicts seven boxes in approximately half of the cases. This suggests that the regression function becomes less sensitive at the upper end of the training range.

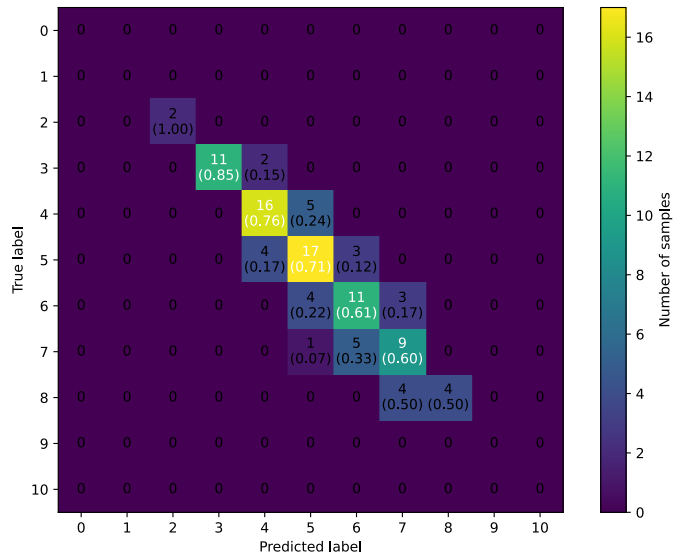


Fig. 10: Confusion matrix for single-input object count regression on 100 test samples of seen counts ($\{2, \dots, 8\}$). Rows correspond to ground truth labels and are normalized to percentages.

b) *Out-of-range Count Labels:* For object counts outside the training range, the MAE increases to 0.67 ± 0.78 , with a

stronger negative bias of -0.29 , indicating systematic underestimation. As shown in Fig. 11, predictions for high-count configurations (9 and 10 boxes) are strongly concentrated at 8 boxes, which is the maximum value observed during training. For low-count configurations, the model performs better: single-box cases are predicted correctly in 70% of instances, with most errors corresponding to overestimation by one unit.

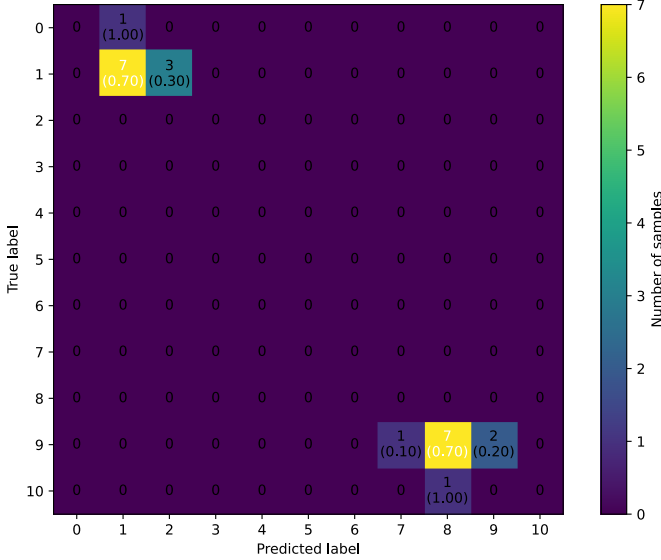


Fig. 11: Confusion matrix for single-input object count regression on 22 test samples with unseen counts ($\{0, 1, 9, 10\}$). Rows correspond to ground truth labels and columns to predicted values. Each entry shows the number of samples, with percentages normalized per row.

2) *Pair-input Object Count Change Regression*: To evaluate the object count change formulation discussed in Section III, we train a model that takes two configurations as input and predicts the total count difference $\Delta C \in [-10, 10]$. Input pairs are constructed using all ordered permutations, and the model is trained with an MSE loss on the count difference. Fig. 12 shows the training and validation loss curves. The model converges to near-zero training loss within four epochs, while the validation loss plateaus early at approximately 0.4, indicating severe overfitting. This behavior arises from the imbalance between input and output diversity. While the number of possible input pairs is large ($P = C \times (C - 1) = 1,047,552$), the output space is limited to $2B + 1 = 21$ distinct values. As a result, many different input pairs correspond to identical targets, enabling the model to memorize input-output associations without learning generalizable acoustic features. In contrast, the single-input regression model does not exhibit such rapid overfitting, suggesting that this issue is inherent to the pairwise regression formulation rather than model capacity alone. These findings empirically validate the limitations discussed in Section III.

Experiment 3 demonstrates that a single simulated multi-channel RIR observation is sufficient to estimate the total

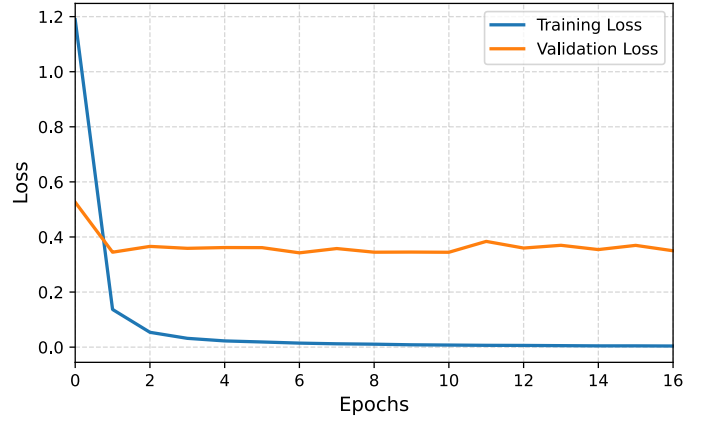


Fig. 12: Training and validation loss for the pair-input object count change regression model. The rapid convergence to near-zero training loss alongside stagnant validation performance indicates severe overfitting.

object count in a room, achieving a $\text{MAE of } 0.38 \pm 0.47$ within the training distribution, well below the random baseline of 1.23. Extrapolation to unseen counts degrades performance ($\text{MAE of } 0.67 \pm 0.78$), with predictions saturating at the boundary of the training range rather than generalizing beyond it. The pair-input regression formulation suffers from severe overfitting due to the mismatch between input diversity and output space, motivating the per-object binary classification approach pursued in Experiment 4.

E. Experiment 4: Binary Change Detection

Having established that the VGGish-based architecture can infer the number of objects present from multi-channel RIRs (Experiment 3), we now investigate whether the model can detect which objects have changed between two observations. As motivated in Section III-A, a per-object binary classification formulation is preferred over object count change regression, as it directly identifies affected locations and avoids the severe overfitting observed in the pair-input regression model. The model takes as input two sets of 4-channel log-Mel spectrograms corresponding to two room states and outputs one binary prediction per box location. Group normalization and Kaiming weight initialization are applied to stabilize training, as detailed in Section III-C. The necessity of these modifications is confirmed empirically in Appendix C-H2.

1) *Baseline Configuration*: In the baseline configuration, the speaker and microphones are mounted at a height of 1.20 m, ensuring that the direct sound path between speaker and microphones is never obstructed by the 0.50 m high boxes. The model achieves an overall mean per-box accuracy of 0.935. Table V reports the per-box breakdown. Box 1 alone remains at random-guess-level accuracy (0.497), while the remaining nine boxes are all predicted with accuracy above 0.94, well above the 0.5 random baseline.

The per-box training and validation loss curves in Fig. 13 reveal a structured learning process consistent across seeds. For the well-learned boxes, training loss converges to near

Box	Baseline	Seed 1	Seed 2	Seed 3	Mean $\pm$ Std
Box 0	1.000	1.000	1.000	1.000	1.000 $\pm$ 0.000
Box 1	0.497	0.512	0.512	0.495	0.504 $\pm$ 0.008
Box 2	0.986	0.978	0.974	0.975	0.978 $\pm$ 0.005
Box 3	0.981	0.997	0.994	0.979	0.988 $\pm$ 0.008
Box 4	0.997	0.997	0.993	0.988	0.994 $\pm$ 0.004
Box 5	0.984	0.957	0.971	0.987	0.975 $\pm$ 0.012
Box 6	0.997	0.997	0.998	0.988	0.995 $\pm$ 0.004
Box 7	0.979	0.972	0.971	0.969	0.972 $\pm$ 0.004
Box 8	0.992	0.992	0.994	0.988	0.991 $\pm$ 0.002
Box 9	0.940	0.943	0.928	0.947	0.940 $\pm$ 0.007
Mean	0.935	0.934	0.933	0.932	0.934 $\pm$ 0.001

Table V: Experiment 4 - Per-box accuracy for the baseline configuration and three additional random seeds, using a classification threshold of $\sigma(z) = 0.5$. The random-guess baseline is 0.5. Mean and standard deviation are computed across all four runs.

zero while validation plateaus below the random-guessing BCE level of 0.693, reflecting correct classification with moderate confidence. The model first converges on Boxes 0, 3, and 6, which lie close to the direct path between the speaker and microphone 1 (Fig. 9), and the remaining boxes converge sequentially as progressively weaker residual signals are extracted. This ordering varies across seeds: Box 7, for example, is learned at epoch 1 under Seeds 1 and 3 but only at epoch 23 under Seed 2. This variability does not affect the final accuracy, but reflects the randomness of gradient-based optimization under different initializations. Table V shows that the overall mean per-box accuracy is stable across runs (0.931–0.935), with per-box standard deviation not exceeding 0.012 for any box other than Box 1. The AUC values (Table XIV of Appendix C) confirm this stability is not an artifact of the 0.5 threshold: all learnable boxes achieve AUC above 0.97 under every seed.

Box 1 does not converge on either the training or validation loss, remaining at the random-guess level of 0.693 under three of the four seeds. Under Seed 2, training loss converges to zero but validation loss rises to 1.311, indicating that the model learned a correlation in the training pairs that inverts on the validation set. The cause of Box 1’s failure is examined in Section V.

2) *Architectural Modifications and Input Fusion*: Two modifications to the baseline VGGish architecture are required for the binary change detection task: group normalization after each convolution-ReLU pair, and Kaiming weight initialization. Ablation experiments confirm that removing either modification causes training to fail, though the exact failure mechanism for each could not be confirmed due to time constraints and is left as future work. For input pair fusion, only early fusion achieves meaningful performance (0.935 mean per-box accuracy). Late fusion collapses to random guessing, likely because the shared-weight encoder has no incentive to produce different embeddings for different inputs, leaving the classifier with no change signal. The full analysis for both experiments are provided in Appendix C.

Box	Accuracy (1.20 m)	Accuracy (0.40 m)
Box 0	1.000	1.000
Box 1	0.497	0.498
Box 2	0.986	0.503
Box 3	0.981	0.994
Box 4	0.997	0.996
Box 5	0.984	0.965
Box 6	0.997	0.992
Box 7	0.979	0.962
Box 8	0.992	0.994
Box 9	0.940	0.494
Mean	0.935	0.840

Table VI: Experiment 4 — Per-box accuracy for the baseline (1.20 m) and occluded (0.40 m) speaker–microphone configurations, using a classification threshold of $\sigma(z) = 0.5$. The random-guess baseline is 0.5.

3) *Effect of Direct Path Occlusion*: In the baseline configuration, the speaker and microphones are mounted above the boxes, so the direct sound path between speaker and microphones is never obstructed. To evaluate a more realistic scenario in which stock items can intercept acoustic paths, the speaker and microphones are lowered to 0.40 m, below the box height of 0.50 m, allowing boxes to fully or partially occlude the direct paths between the speaker and the microphones. The model is retrained from scratch on this modified dataset using the same hyperparameters as the baseline configuration.

The per-box accuracies for both configurations are reported in Table VI. The overall mean per-box accuracy decreases from 0.935 to 0.840. Of the three boxes for which accuracy collapses to random-guess level, Box 1 was already at random-guess level in the baseline configuration (0.497) and remains so under occlusion (0.498), indicating that lowering the sensor height does not further affect its performance, and that its failure is independent of direct-path occlusion. Boxes 2 and 9 represent the occlusion-induced failures: both achieve accuracy above 0.98 in the baseline but collapse to random-guess level (0.503 and 0.494, respectively) when the sensors are lowered. Inspecting the room layout (Fig. 9), Boxes 2 and 9 are spatially clustered in close proximity to each other and to Box 4. Section V looks into the interpretation of the failure. Unlike the baseline configuration, the occlusion experiment is conducted with a single seed. Whether the failures of Boxes 2 and 9 are stable across initializations is left for future work.

4) *Generalization to Unseen Numbers of Simultaneous Changes*: The baseline model trains and tests on all pair permutations, covering up to all ten boxes changing simultaneously. In practice, training data may only contain small change events, for example during an initial deployment period with limited restocking activity. To test generalization across change magnitudes, we train on pairs with at most three simultaneous changes (109,368 pairs, compared to 639,200 for the full permutation set) and evaluate on pairs with four to ten simultaneous changes. Limiting training to at most three changes creates a class imbalance: locations with no change outnumber changed locations at a ratio of 3:1. Two variants

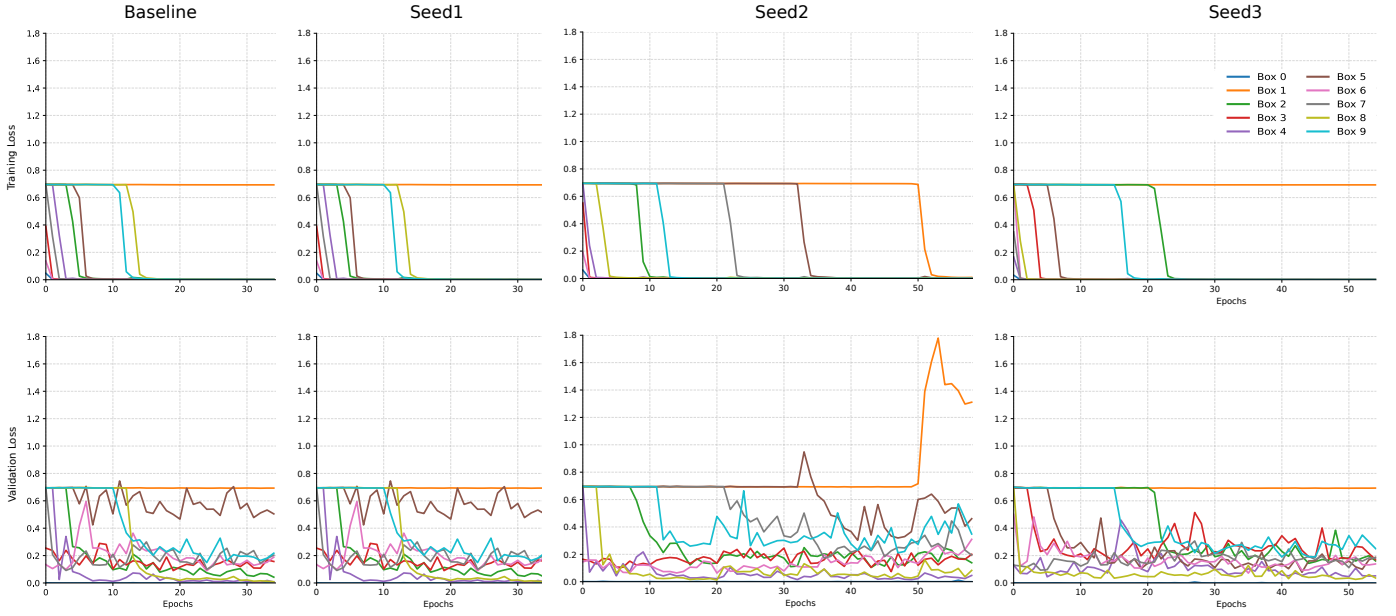


Fig. 13: Per-box training (top) and validation (bottom) loss curves for the baseline and three seeds. Most boxes converge consistently (training to 0, validation less than 0.693). Box 1 fails to converge, remaining at random or overfitting (Seed 2). Note that the number of epochs differs across runs due to early stopping.

are therefore trained: one with uniform loss weighting and one where positive-class errors are up-weighted by a factor of three. On the in-distribution test set (Fig. 14, top), all ten boxes achieve AUC above 0.94, with nine above 0.98. Notably, Box 1 is learned successfully in this setting (AUC of 0.947), in contrast to its consistent failure in the baseline configuration. One possible interpretation is that Box 1’s acoustic signature becomes detectable when training is restricted to smaller change magnitudes. Since a smaller training set (109,368 vs. 639,200 pairs) would generally be expected to hurt rather than help generalization, the improvement in Box 1 detection is more plausibly explained by the reduced number of simultaneously changing boxes: when fewer boxes change at once, Box 1’s acoustic contribution represents a larger fraction of the total RIR change, making it easier for the model to learn. A size-matched subsample of the full permutation set would nonetheless be needed to confirm this, and is left for future work. As the two loss-weighting variants perform similarly, it is confirmed that the 3:1 imbalance does not meaningfully affect in-distribution performance. On the out-of-distribution test set (Fig. 14, bottom), AUC drops for most boxes but remains above 0.9 for seven of ten, with all boxes above 0.71. The degradation is location-dependent: Boxes 3, 4, and 6 retain AUC above 0.95, while Boxes 1 and 2 drop the most (0.718 and 0.782). Loss reweighting produces only modest and inconsistent changes across boxes, indicating that the performance gap is not primarily attributable to the training class imbalance. These results show that the model generalizes meaningfully to change magnitudes not seen during training, though performance is location-dependent and degrades for some boxes.

Experiment 4 demonstrates that a VGGish-based architecture can perform per-object binary change detection from simulated multi-channel RIRs, achieving near-perfect per-box accuracy for the majority of locations in the non-occluded configuration. Box 1 is the single persistent failure case under full-permutation training, though its acoustic signature becomes detectable when training is restricted to smaller change magnitudes. In the single-seed occlusion experiment, mean accuracy decreases from 0.935 to 0.840. The two boxes whose accuracy collapses (Boxes 2 and 9) are spatially clustered near Box 4, suggesting that direct-path occlusion may affect closely spaced objects, though this pattern has not been confirmed across multiple initializations. The generalization experiment shows that per-object detection capability is largely preserved when tested on change magnitudes not seen during training, with seven of ten boxes retaining an AUC above 0.9.

V. Discussion

The experiments establish three principal findings. First, multi-channel RIRs encode sufficient information for object count regression and per-object binary change detection well above random baselines. Second, acoustic simulation can reproduce in-domain performance approaching the real-data baseline for multi-channel configurations, but fails in cross-domain transfer in ways that simple signal-level corrections cannot resolve. Third, per-object detection capability is largely preserved under change magnitudes and occlusion conditions not seen during training, with failure modes that are spatially localized and physically interpretable. The subsections below interpret each finding, identify their shared mechanisms, and draw out the practical implications.

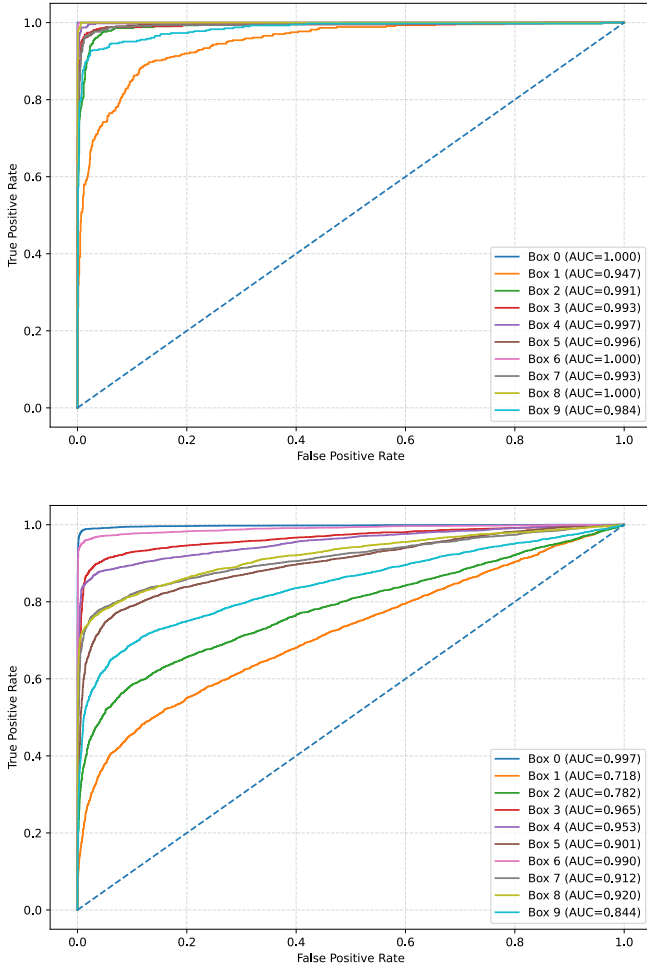


Fig. 14: ROC-AUC curves per box when trained on at most three simultaneous changes, and (top) tested on at most three, or (bottom) tested on four up to ten simultaneous changes.

A. Interpretation of the results

The results suggest plausible explanations, but their interpretation is constrained by methodological limitations and domain effects. The following paragraphs examine patterns and physically interpretable failure modes, starting with a note on the reliability of single-seed experiments.

Reliability and Sensitivity. Seed sensitivity analysis for the localization performance reveals that weight initialization alone accounts for performance variations of up to ± 3.62 cm, comparable in magnitude to the differences between microphone configurations and between fusion strategies. This sensitivity applies beyond localization: all sub-experiments of the per-object change detection and all object count regression results are based on single seeds. Consequently, the specific per-box accuracy values and failure cases reported should be interpreted as representative of one initialization rather than as stable estimates. Multi-seed evaluation of all experiments would be needed to establish the robustness of these findings.

Simulation Fidelity and the Sim-to-Real Gap. Within-domain simulation fidelity is sufficient for multi-channel localization configurations, but the gap widens as channel count decreases, collapsing entirely for a single channel. Single-microphone localization relies on fine-grained spectral cues such as frequency-dependent scattering, surface irregularity, and source/receiver directivity, which the simulator does not reproduce. With multiple microphones, this can be partially compensated by exploiting inter-channel timing and amplitude differences that the simulator does capture.

A fundamental limitation of the simulated RIRs is that material coefficients are defined only up to 4 kHz, corresponding to a wavelength of approximately 8.6 cm ($\lambda = c/f = 343/4000$). Material-dependent absorption, scattering, and transmission are therefore accurately represented only for geometric features on the order of 10 cm or larger. Finer spatial details such as edges, gaps, and surface irregularities produce frequency-dependent reflections above 4 kHz that are physically present in real recordings but absent in simulation, where higher-frequency bands are clipped to the 4 kHz coefficient value. Simulated 48 kHz RIRs consequently carry no additional material-specific information above 4 kHz relative to 16 kHz, which explains the marginal within-domain disadvantage of 48 kHz in simulation.

In cross-domain evaluation, errors exceed the random-guess baseline in most configurations, and neither waveform-level corrections (zero-padding, peak removal) nor the introduction of a localized geometric asymmetry in the simulated room reduce the gap, ruling out simple temporal misalignment and geometric dissimilarity as primary causes. A notable outlier is the ten-channel Sim→Real condition, which shows a disproportionately large error compared to the two- and four-channel configurations. A likely contributing factor is the geometry of the ten-channel layout (see Fig. 15 in Appendix A): several microphones are approximately aligned with the speaker, so some microphones partially or fully obstruct the direct acoustic path of its neighbors. Since SoundSpaces 2.0 models sources and receivers as massless point entities, these shadowing and diffraction effects are absent from the simulated data. The model therefore learns inter-channel relationships in simulation that do not hold in the real room, producing a large cross-domain penalty specific to this configuration. In the lower-channel configurations, each microphone is placed in a separate corner of the room, so mutual obstruction does not arise and the learned inter-channel structure is more transferable. Whether this is specific to the shadowing geometry of the ten-channel layout, or reflects a more general tendency for cross-domain error to grow with channel count, cannot be determined from four configurations alone and warrants dedicated investigation. The broader conclusion is that the domain gap reflects structural differences in spectral content and inter-channel relationships that are not addressable by signal-level corrections alone.

Object Count Inference. The degradation of count regression accuracy at higher box counts has two plausible contributing factors. First, the training data follows a binomial distribution

over the 1,024 configurations, peaking at five boxes and tapering toward the extremes, so higher counts are under-represented. Second, in the near-anechoic Treated Room, the highly absorptive walls ($\alpha \geq 0.99$ above 250 Hz) absorb most incident energy, while the plastic boxes ($\alpha \approx 0.02 - 0.05$) reflect it. Each added box therefore redirects energy back into the room that would otherwise be absorbed by the walls, increasing the total reverberant energy. While this property is beneficial for detection in principle, it also means that as more boxes are present, the reverberant field becomes increasingly complex and the marginal acoustic change introduced by each additional box becomes harder to distinguish from the contributions of the others. The present experimental design does not allow these two factors of data imbalance and increased reverberant field complexity to be disentangled. A count-balanced training set, in which each count is represented equally, would be needed to isolate the contribution of data imbalance from the acoustic saturation effect.

For counts outside the training range, the model does not extrapolate but instead saturates at the training boundary. This confirms that the model has learned a bounded mapping from spectrograms to counts rather than a physically grounded relationship, and that the full expected count range must be covered during training.

Per-Object Change Detection. The structured convergence ordering, where Boxes 0, 3, and 6 converge first across all seeds while the remaining boxes follow in a partially consistent sequence, is consistent with the hypothesis that per-box acoustic salience varies with spatial position. However, this interpretation remains indirect: the convergence order could also reflect properties of the loss landscape or gradient flow that are unrelated to acoustic signal strength. Establishing a direct link would require independently measuring per-box RIR energy differences and testing whether they correlate with convergence order across seeds.

Box 1 is the single persistent failure case under full-permutation training, remaining at random-guess-level accuracy across all seeds. The cause is not fully determined. Spatial proximity to neighbors is not a sufficient explanation: Box 1’s nearest neighbors are approximately 25 cm away, well above the 8.6 cm wavelength threshold, and more tightly clustered boxes (2 and 9) are successfully learned. One candidate explanation is that Box 1’s position falls in a region of low acoustic sensitivity relative to the four-microphone layout, such that the RIR changes induced by its presence or absence are too subtle for the model to detect. Computing per-box RIR differences would be needed to test this hypothesis. The generalization experiment provides indirect evidence: under restricted training (at most three simultaneous changes), Box 1 is learned successfully. One possible interpretation is that with fewer concurrent changes, Box 1’s contribution represents a larger fraction of the total RIR difference and is therefore easier to isolate. However, the restricted training set is also substantially smaller (109,368 vs. 639,200 pairs), and the effect of dataset size has not been controlled for. A size-matched subsample of the full permutation set would

be needed to distinguish between these explanations. More broadly, the generalization experiment shows that per-object detection transfers to change magnitudes not seen during training, suggesting that the learned representations capture acoustic differences that are not specific to the training pair distribution.

Direct-path occlusion reduces the accuracy of the model, with failures concentrated at spatially clustered locations. Box 1’s failure is independent of sensor height, confirming its cause is distinct from direct-path obstruction. Boxes 2 and 9 collapse to random-guess level when the sensors are lowered, while nearby Box 4 remains reliably detected. A plausible explanation is that the lower sensor height reduces the angular diversity of reflections from the Boxes 2–4–9 cluster, making their contributions harder to separate. The tight spatial clustering of these boxes may compound this effect: when reflections arrive from similar directions, distinguishing individual contributions would require fine spectral differences above 4 kHz, where material-dependent simulation accuracy is limited. The occlusion experiment was conducted with a single seed, so the stability of these results across initializations has not been verified.

Common Failure Mechanisms. The failure modes across experiments share a recurring pattern: in each case, the affected location is one where the acoustic change induced by the object’s presence or absence is small relative to competing signals: in comparison to simultaneously changing boxes (Box 1 under full-permutation training), reduced angular diversity at lower sensor height (Boxes 2 and 9 under occlusion), or boundary effects in the count distribution (saturation at high counts). This pattern is suggestive of a common underlying mechanism that detection depends on the relative rather than absolute acoustic contribution of each object, but confirming this would require measuring per-box signal strength directly and correlating it with detection performance across conditions. Without such measurements, the unifying interpretation remains a hypothesis.

A second recurring factor is the 4 kHz material coefficient truncation as discussed above. The lack of material-dependent frequency modeling could be an explanation for both the sim-to-real gap and the spatially localized detection failures, suggesting that extending material characterization to higher frequencies could yield compounding benefits.

Finally, the count saturation, bounded mapping, and per-box signal competition observed across experiments all point to the same underlying property: the model learns relative acoustic differences between room states rather than a physically grounded representation of object presence. This limits extrapolation beyond the training distribution in count regression and makes detection sensitive to the number and configuration of simultaneously changing objects in change detection.

B. Limitations

The sim-to-real gap is the most significant practical limitation and is not addressable by the strategies tested here. The gap arises from multiple sources beyond the 4 kHz

material coefficient truncation discussed above: the simulated RIRs are noise-free, hardware effects such as speaker delays and microphone directivity are absent, material properties are uniform across each surface rather than spatially varying, and environmental factors such as temperature and humidity are not modeled. Spectrogram augmentation or material randomization have not been applied here and closing the gap remains the primary prerequisite for practical deployment.

Furthermore, all simulated and real RIRs used in this work were obtained under low-noise conditions. Real warehouse environments introduce substantial ambient noise from ventilation, machinery, and personnel activity, which would degrade deconvolution quality and reduce the signal-to-noise ratio of the acoustic features the model relies on. The robustness of the approach to realistic noise levels has not been evaluated and constitutes a necessary step before deployment.

The experimental setting of a small, highly absorptive Treated Room differs from the target warehouse application in several important respects beyond room size. In the Treated Room, the high wall absorption creates a strong acoustic contrast with the reflective plastic boxes and reduce the total reverberant field. A typical warehouse with hard reflective surfaces (concrete, metal) has absorption coefficients comparable to or lower than the plastic boxes used here. In such an environment, adding stock would not introduce a reflective object against an absorptive background, but rather one reflective surface among many, substantially increasing the RIR complexity. Whether the approach remains viable under these conditions is an open question that cannot be answered from the present experiments alone.

All object count regression and change detection experiments use a single room geometry, a fixed starting configuration of ten uniformly sized boxes, and a single speaker-microphone arrangement. The localization experiments showed that object geometry significantly influences RIR structure, suggesting that varying box sizes in training would be important for generalization to real warehouse stock. Additionally, robustness to small positional displacements has not been assessed. In real restocking operations minor box shifts are unavoidable, and the current formulation with fixed positions could misinterpret them as additions or removals.

Finally, scalability presents an additional challenge. The number of configurations grows as 2^B with the number of object locations, and the number of training pairs grows as $2^B \times (2^B - 1)$. For the ten-box case studied here, this yields 1,024 configurations and over one million pairs, which is computationally tractable: the simulation takes about one hour per microphone. However, a realistic warehouse section would likely have more than ten stock positions, making simulation intractable. Strategies such as acoustic zoning, sparse sampling of the configuration space, or curriculum-based training would be needed to extend the approach beyond small-scale settings.

C. Future Work

Closing the sim-to-real gap is the primary priority. Two directions appear most promising based on the findings of

this work. First, extending material coefficient measurements above 4 kHz would address the most physically concrete identified cause of the gap. Second, the disproportionately large cross-domain error at ten channels suggests that inter-channel calibration to better match real recordings may offer more leverage than single-channel spectral corrections alone. More broadly, progressive realism strategies [47] that fine-tune on small amounts of real data, and training across randomized room geometries and material coefficients [48], would reduce reliance on simulation-specific artifacts.

Diagnosing Box 1’s persistent failure should precede any architectural intervention. Computing per-box RIR differences would establish whether Box 1 produces a weaker absolute signal than its neighbors, and analyzing the spatial pattern of prediction errors would clarify whether the failure reflects spatial confusion or a fundamental lack of acoustic signal. If the cause is insufficient microphone coverage at Box 1’s position, redistributing the microphones may resolve the failure without model changes.

Validating the approach in a more realistic acoustic environment is essential. The near-anechoic Treated Room provides a favorable detection regime that may not transfer to reverberant warehouse settings. Experiments in a room with harder, more reflective walls would test whether the detection mechanism generalizes beyond the conditions studied here.

Finally, scaling beyond the ten-object setting studied here also requires attention. Since the configuration space grows as 2^B , exhaustive simulation becomes intractable for realistic warehouse sections with substantially more stock positions. A zone-based formulation, in which the model detects changes at the level of spatial regions rather than individual object locations, would reduce the number of detection targets and could potentially generalize across object configurations within a zone, reducing the need to simulate all possible per-object combinations exhaustively. Whether such generalization holds is an open question.

VI. Conclusion

This work investigated active acoustic change detection for passive silent objects, proposing object count regression and multi-label binary change detection as two tasks addressable from multi-channel RIRs, and validating both through four incremental experiments on a simulated replica of the SoundCam Treated Room. In-domain simulated performance approaches the real-data baseline for multi-channel configurations (18.2 vs. 15.5 cm with ten microphones), but cross-domain transfer fails in both directions, with errors exceeding the random-guess baseline (RQ1). Additionally, a single multi-channel RIR observation is sufficient to estimate the number of objects within the training distribution (MAE of 0.38 against a random baseline of 1.23) (RQ2). Extrapolation to unseen counts degrades and saturates at the training boundary. Finally, per-object presence changes from pairs of multi-channel RIR observations can be detected with high accuracy for the majority of object locations, with nine of ten locations achieving accuracy above 0.94 (RQ3). In a single-seed

experiment, direct-path occlusion reduces mean accuracy from 0.935 to 0.840, with failures concentrated at spatially clustered locations. Models trained on small change events generalize meaningfully to larger simultaneous changes, retaining AUC above 0.9 for seven of ten locations. Closing the sim-to-real gap remains the primary barrier to practical deployment.

References

- [1] P. Rikhardsson, K. R. Thórisson, G. Bergthorsson, and C. Batt, “Artificial intelligence and auditing in small- and medium-sized firms: Expectations and applications,” *AI Magazine*, vol. 43, no. 3, pp. 323–336, Aug. 2022. DOI: 10.1002/aaai.12066. [Online]. Available: <https://doi.org/10.1002/aaai.12066>.
- [2] M. H. Christ, S. A. Emett, S. L. Summers, and D. A. Wood, “Prepare for takeoff: Improving asset measurement and audit quality with drone-enabled inventory audit procedures,” en, *Review of Accounting Studies*, vol. 26, no. 4, pp. 1323–1343, Dec. 2021, ISSN: 1573-7136. DOI: 10.1007/s11142-020-09574-5. Accessed: Mar. 5, 2026. [Online]. Available: <https://doi.org/10.1007/s11142-020-09574-5>.
- [3] PA SYSTEM SPEAKERS— SOUND SYSTEM SUPPLIERS IN DUBAI — PUBLIC ADDRESSING SYSTEM. [Online]. Available: <https://imtedad.com/public-address-system-pa-system.php>.
- [4] S. T. Neely and J. B. Allen, “Invertibility of a room impulse response,” en, *The Journal of the Acoustical Society of America*, vol. 66, no. 1, pp. 165–169, Jul. 1979, ISSN: 0001-4966, 1520-8524. DOI: 10.1121/1.383069. Accessed: Mar. 4, 2026. [Online]. Available: <https://pubs.aip.org/jasa/article/66/1/165/651493/Invertibility-of-a-room-impulse-response>.
- [5] C.-H. Lin, K.-M. Cheong, M.-C. Huang, M.-Y. Chen, C.-K. Chang, and T.-S. Chi, “Acoustic scene change detection by spectro-temporal filtering on spectrogram using chirps,” in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, Dec. 2016, pp. 1–5. DOI: 10.1109/APSIPA.2016.7820783. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/7820783>.
- [6] N. Yoneoka, Y. Arakawa, and K. Yasumoto, “Detecting Surrounding Users by Reverberation Analysis with a Smart Speaker and Microphone Array,” in *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Mar. 2019, pp. 523–528. DOI: 10.1109/PERCOMW.2019.8730674. [Online]. Available: <https://ieeexplore.ieee.org/document/8730674?arnumber=8730674>.
- [7] K.-M. Cheong, Y.-L. Shen, and T.-S. Chi, “Active acoustic scene monitoring through spectro-temporal modulation filtering for intruder detection,” *Journal of the Acoustical Society of America*, vol. 152, no. 4, pp. 2229–2242, 2022. DOI: 10.1121/10.0014957. [Online]. Available: <https://doi.org/10.1121/10.0014957>.
- [8] A. Alanwar, B. Balaji, Y. Tian, S. Yang, and M. Srivastava, “EchoSafe: Sonar-based Verifiable Interaction with Intelligent Digital Agents,” in *Proceedings of the 1st ACM Workshop on the Internet of Safe Things*, ACM, Nov. 5, 2017, pp. 38–43. DOI: 10.1145/3137003.3137014. [Online]. Available: <https://dl.acm.org/doi/10.1145/3137003.3137014>.
- [9] H. Q. V. Pham, H. D. Nguyen, and U. Roedig, “Gotcha: Physical intrusion detection with active acoustic sensing using a smart speaker,” in *Secure IT Systems – 29th Nordic Conference, NordSec 2024, Proceedings*, L. H. Iwaya, L. Kamm, L. A. Martucci, and T. Pulls, Eds., ser. Lecture Notes in Computer Science, vol. 15396, Cham: Springer, 2024, pp. 345–363. DOI: 10.1007/978-3-031-79007-2_18. [Online]. Available: https://doi.org/10.1007/978-3-031-79007-2_18.
- [10] M. Wang, S. Clarke, J.-H. Wang, R. Gao, and J. Wu, “Soundcam: A dataset for finding humans using room acoustics,” in *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36, Curran Associates, Inc., 2023, pp. 52 238–52 264. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/file/a4289154c9209b679ac761a50d5fec3a-Paper-Datasets_and_Benchmarks.pdf.
- [11] S. Hershey et al., “CNN architectures for large-scale audio classification,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, ISSN: 2379-190X, Mar. 2017, pp. 131–135. DOI: 10.1109/ICASSP.2017.7952132. Accessed: Mar. 5, 2026. [Online]. Available: <https://ieeexplore.ieee.org/document/7952132/>.
- [12] K. Simonyan and A. Zisserman, *Very Deep Convolutional Networks for Large-Scale Image Recognition*, arXiv:1409.1556 [cs], Apr. 2015. DOI: 10.48550/arXiv.1409.1556. Accessed: Mar. 5, 2026. [Online]. Available: <http://arxiv.org/abs/1409.1556>.
- [13] G. Hwang, S. Kim, and H.-M. Bae, “Bat-G net: Bat-inspired High-Resolution 3D Image Reconstruction using Ultrasonic Echoes,” in *Advances in Neural Information Processing Systems*, vol. 32, Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/hash/5352696a9ca3397beb79f116f3a33991-Abstract.html.
- [14] S. Kim, G. Hwang, and H.-M. Bae, “Bat-G2 Net: Bat-Inspired Graphical Visualization Network Guided by Radiated Ultrasonic Call,” *IEEE Access*, vol. 8, pp. 189 673–189 683, 2020. DOI: 10.1109/ACCESS.2020.3031297. [Online]. Available: <https://ieeexplore.ieee.org/document/9224899/>.
- [15] B. S. Liang et al., “Reconstructing room scales with a single sound for augmented reality displays,” *Journal of Information Display*, vol. 24, no. 1, pp. 1–12, Jan. 2, 2023. DOI: 10.1080/15980316.2022.2145377. [Online]. Available: <https://doi.org/10.1080/15980316.2022.2145377>.

- [16] Y. Shibata, Y. Kawashima, M. Isogawa, G. Irie, A. Kimura, and Y. Aoki, "Listening Human Behavior: 3D Human Pose Estimation With Acoustic Signals," presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 13 323–13 332. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2023/html/Shibata_Listening_Human_Behavior_3D_Human_Pose_Estimation_With_Acoustic_Signals_CVPR_2023_paper.html.
- [17] J. Lawrence, C. Tuna, A. Walther, and N. Peters, "Acoustic Position Estimation of a Silent Listener," in *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2025, pp. 1–5. DOI: 10.1109/ICASSP49660.2025.10888595. [Online]. Available: <https://ieeexplore.ieee.org/document/10888595/>.
- [18] R. Valle, "ABROA: Audio-based room-occupancy analysis using Gaussian mixtures and Hidden Markov models," in *2016 Future Technologies Conference (FTC)*, IEEE, Dec. 2016, pp. 1270–1273. DOI: 10.1109/FTC.2016.7821763. [Online]. Available: <http://ieeexplore.ieee.org/document/7821763/>.
- [19] S. Ghaffarzadegan, A. Reiss, M. Ruhs, R. Duerichen, and Z. Feng, "Occupancy Detection in Commercial and Residential Environments Using Audio Signal," in *Interspeech 2017*, ISCA, Aug. 20, 2017, pp. 3802–3806. DOI: 10.21437/Interspeech.2017-524. [Online]. Available: https://www.isca-archive.org/interspeech_2017/ghaffarzadegan17_interspeech.html.
- [20] Q. Huang, "Occupancy-Driven Energy-Efficient Buildings Using Audio Processing with Background Sound Cancellation," *Buildings*, vol. 8, no. 6, p. 78, 6 Jun. 2018. DOI: 10.3390/buildings8060078. [Online]. Available: <https://www.mdpi.com/2075-5309/8/6/78>.
- [21] L. Wang et al., "BERP: A Blind Estimator of Room Parameters for Single-Channel Noisy Speech Signals," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 2215–2231, 2025. DOI: 10.1109/TASLP.2025.3574849. arXiv: 2405.04476 [eess]. [Online]. Available: <http://arxiv.org/abs/2405.04476>.
- [22] T. N. T. Nguyen, W.-S. Gan, R. Ranjan, and D. L. Jones, "Robust Source Counting and DOA Estimation Using Spatial Pseudo-Spectrum and Convolutional Neural Network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2626–2637, 2020. DOI: 10.1109/TASLP.2020.3019646. [Online]. Available: <https://ieeexplore.ieee.org/document/9178434>.
- [23] O. Shih and A. Rowe, "Occupancy estimation using ultrasonic chirps," in *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems*, Association for Computing Machinery, Apr. 14, 2015, pp. 149–158. DOI: 10.1145/2735960.2735969. [Online]. Available: <https://dl.acm.org/doi/10.1145/2735960.2735969>.
- [24] G. Götz, S. J. Schlecht, and V. Pulkki, "MOTUS: A dataset of higher-order Ambisonic room impulse responses and 3D models measured in a room with varying furniture," in *2021 Immersive and 3D Audio: From Architecture to Automotive (I3DA)*, Sep. 2021, pp. 1–8. DOI: 10.1109/I3DA48870.2021.9610933. [Online]. Available: <https://ieeexplore.ieee.org/document/9610933/>.
- [25] C. Chen et al. "SoundSpaces 2.0: A Simulation Platform for Visual-Acoustic Learning," arXiv: 2206.08312 [cs]. [Online]. Available: <http://arxiv.org/abs/2206.08312>.
- [26] B. Foundation, *Blender - the free and open source 3D creation software — Blender.org*. [Online]. Available: <https://www.blender.org/>.
- [27] R. Scheibler, E. Bezzam, and I. Dokmanić, "Pyroomacoustics: A Python Package for Audio Room Simulation and Array Processing Algorithms," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2018, pp. 351–355. DOI: 10.1109/ICASSP.2018.8461310. [Online]. Available: <https://ieeexplore.ieee.org/document/8461310/>.
- [28] M. Savva et al., "Habitat: A Platform for Embodied AI Research," en, in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South): IEEE, Oct. 2019, pp. 9338–9346, ISBN: 978-1-7281-4803-8. DOI: 10.1109/ICCV.2019.00943. Accessed: Mar. 5, 2026. [Online]. Available: <https://ieeexplore.ieee.org/document/9010745/>.
- [29] T. Yoshida, T. Okuzono, and K. Sakagami, "A Parallel Dissipation-Free and Dispersion-Optimized Explicit Time-Domain FEM for Large-Scale Room Acoustics Simulation," en, *Buildings*, vol. 12, no. 2, p. 105, Feb. 2022, ISSN: 2075-5309. DOI: 10.3390/buildings12020105. Accessed: Mar. 5, 2026. [Online]. Available: <https://www.mdpi.com/2075-5309/12/2/105>.
- [30] T. Technologies, *Treble*. [Online]. Available: <https://www.treble.tech/>.
- [31] P. Srivastava, "Realism in virtually supervised learning for acoustic room characterization and sound source localization," Issue: 2023LORR0184, Theses, Université de Lorraine, Nov. 2023. Accessed: Mar. 5, 2026. [Online]. Available: <https://theses.hal.science/tel-04313405>.
- [32] W. Cao, V. Mirjalili, and S. Raschka, "Rank consistent ordinal regression for neural networks with application to age estimation," en, *Pattern Recognition Letters*, vol. 140, pp. 325–331, Dec. 2020, ISSN: 01678655. DOI: 10.1016/j.patrec.2020.11.008. Accessed: Mar. 13, 2026. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S016786552030413X>.
- [33] J. Wang, J. Chen, J. Liu, D. Tang, D. Z. Chen, and J. Wu, *A Survey on Ordinal Regression: Applications, Advances and Prospects*, arXiv:2503.00952 [cs], Mar.

2025. DOI: 10.48550/arXiv.2503.00952. Accessed: Mar. 13, 2026. [Online]. Available: <http://arxiv.org/abs/2503.00952>.
- [34] M. Kimura, R. Shimizu, Y. Hirakawa, R. Goto, and Y. Saito, *On permutation-invariant neural networks*, arXiv:2403.17410 [cs], Mar. 2024. DOI: 10.48550/arXiv.2403.17410. Accessed: Mar. 5, 2026. [Online]. Available: <http://arxiv.org/abs/2403.17410>.
- [35] PyTorch, *BCEWithLogitsLoss*, Jan. 2023. [Online]. Available: <https://docs.pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html>.
- [36] scikit-learn developers, *StratifiedKFold*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedKFold.html.
- [37] J. Allen, “Short term spectral analysis, synthesis, and modification by discrete Fourier transform,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 25, no. 3, pp. 235–238, Jun. 1977, ISSN: 0096-3518. DOI: 10.1109/TASSP.1977.1162950. Accessed: Mar. 5, 2026. [Online]. Available: <https://ieeexplore.ieee.org/document/1162950/>.
- [38] S. S. Stevens, J. Volkman, and E. B. Newman, “A scale for the measurement of the psychological magnitude pitch,” *The journal of the acoustical society of america*, vol. 8, no. 3, pp. 185–190, 1937.
- [39] G. Castel-Branco, G. Falcao, and F. Perdigão, “Puremic: A new audio dataset for the classification of musical instruments based on convolutional neural networks,” *Journal of Signal Processing Systems*, vol. 93, no. 9, pp. 977–987, 2021.
- [40] Y. Wu and K. He, “Group Normalization,” in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., Springer International Publishing, 2018, pp. 3–19. DOI: 10.1007/978-3-030-01261-8_1.
- [41] K. He, X. Zhang, S. Ren, and J. Sun, “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile: IEEE, Dec. 2015, pp. 1026–1034, ISBN: 978-1-4673-8391-2. DOI: 10.1109/ICCV.2015.123. Accessed: Mar. 5, 2026. [Online]. Available: <http://ieeexplore.ieee.org/document/7410480/>.
- [42] D. Wen et al., “Change Detection From Very-High-Spatial-Resolution Optical Remote Sensing Images: Methods, applications, and future directions,” *IEEE Geoscience and Remote Sensing Magazine*, vol. 9, no. 4, pp. 68–101, Dec. 2021, ISSN: 2168-6831. DOI: 10.1109/MGRS.2021.3063465. Accessed: Mar. 13, 2026. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/9395350>.
- [43] Facebookresearch, *sound-spaces/SoundSpaces2.md at main · facebookresearch/sound-spaces*. [Online]. Available: <https://github.com/facebookresearch/sound-spaces/blob/main/SoundSpaces2.md>.
- [44] A. Chang et al., *Matterport3D: Learning from RGB-D Data in Indoor Environments*, arXiv:1709.06158 [cs], Sep. 2017. DOI: 10.48550/arXiv.1709.06158. Accessed: Mar. 5, 2026. [Online]. Available: <http://arxiv.org/abs/1709.06158>.
- [45] S. Balkus, *The median minimizes the mean absolute error*, Sep. 2024. [Online]. Available: <https://statproofbook.github.io/P/med-mae.html>.
- [46] R. D. Blevins, *Formulas for dynamics, acoustics and vibration*. John Wiley & Sons, 2015.
- [47] J. M. Vera-Diaz, D. Pizarro, and J. Macias-Guarasa, “Towards End-to-End Acoustic Localization Using Deep Learning: From Audio Signals to Source Position Coordinates,” *Sensors*, vol. 18, no. 10, p. 3418, 10 Oct. 2018. DOI: 10.3390/s18103418. [Online]. Available: <https://www.mdpi.com/1424-8220/18/10/3418>.
- [48] P.-A. Grumiaux, S. Kitić, L. Girin, and A. Guérin, “A survey of sound source localization with deep learning methods,” *The Journal of the Acoustical Society of America*, vol. 152, no. 1, pp. 107–151, Jul. 5, 2022. DOI: 10.1121/10.0011809. [Online]. Available: <https://doi.org/10.1121/10.0011809>.
- [49] P. M. Morse and K. U. Ingard, *Theoretical acoustics*. Princeton university press, 1986.
- [50] *Understanding NRC Ratings in Acoustics and why some materials exceed an NRC of 1.0*. Accessed: Mar. 5, 2026. [Online]. Available: <https://www.baresque.com.au/articles/unlocking-acoustic-science-understanding-nrc-ratings-and-why-some-materials-go-beyond-1-0>.
- [51] J. A. Supplies, *Sound Absorption Coefficient Chart (125 Hz–4 KHz)*, 2025. [Online]. Available: <https://www.acoustic-supplies.com/absorption-coefficient-chart/>.
- [52] N. W. N. Control, “MelaMax Datasheet,” Net Well Noise Control, Tech. Rep. [Online]. Available: <https://www.controlnoise.com/wp-content/uploads/2017/04/MelaMax-Datasheet.pdf>.
- [53] U. ME, “Halcyon™ Acoustical Ceiling,” USG ME, Tech. Rep., Jan. 2018. [Online]. Available: https://www.usgme.com/media/mageworx/downloads/attachment/file/u/s/usg_me_halcyon_ceiling_technical_datasheet.pdf.
- [54] A. Exchange, *Acoustic Absorption Coefficient of Human Body in Octave Bands*, Jan. 2023. [Online]. Available: <https://www.auralexchange.com/knowledgebase/acoustic-absorption-coefficient-of-human-body-in-octave-bands/>.
- [55] W. G. Neubauer, *Acoustic reflection from surfaces and shapes*. Naval Research Laboratory, 1986.

Appendix A Microphone Configurations

The microphone configurations used in our experiments are illustrated in Fig. 15. Ten microphones are distributed along the walls of the SoundCam Treated Room at a height of 1.2 m. A speaker is placed in one of the corners at the same height. Table VII summarizes which microphones are used in each

configuration.

Table VII: Microphone selection for different channel configurations. Microphone numbers correspond to those shown in Fig. 15.

Configuration	Microphones Used
10-channel	1–10 (all)
4-channel	1, 2, 3, 4 (corner microphones)
2-channel	1, 2 (closest and farthest from speaker)
1-channel	1 (farthest from speaker)

In the 10-channel configuration, some microphones partially block the direct path from neighboring microphones to the speaker. The 4-channel configuration, used in Experiments 3 and 4, was selected based on the trade-off analysis described in Section IV. The 2-channel setup uses only the microphones closest and farthest from the speaker, while the single-channel setup employs the microphone farthest from the speaker.

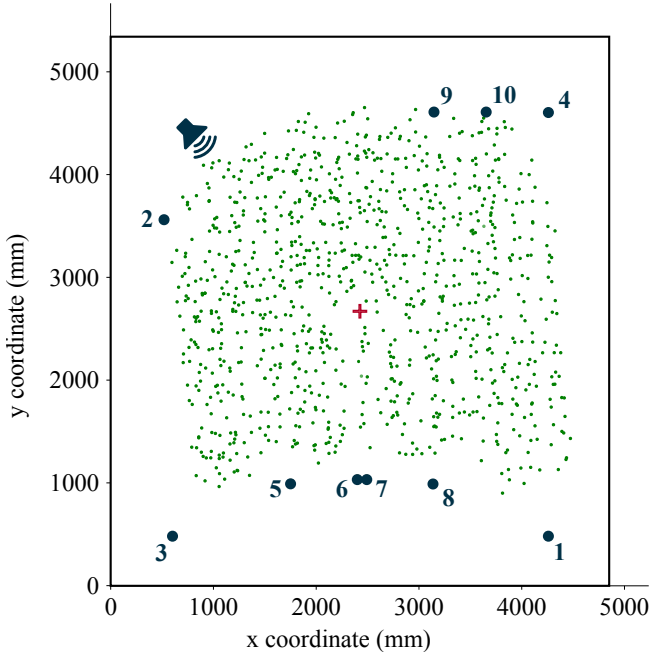


Fig. 15: Speaker and microphone (1–10) positions in the Treated Room ($4.9 \times 5.1 \times 2.7$ m). The green dots are the 1,000 centroid positions of the human in the localization task. The red + sign marks the center of the room. Microphone numbers correspond to Table VII. Adapted from [10].

Appendix B Simulation Details

This appendix provides the detailed acoustic parameter settings and material specifications used in the SoundSpaces 2.0 simulation.

A. Acoustic Parameters

A sampling rate of 48 kHz is used to match the real SoundCam recordings. The direct ray count (50,000) and

indirect ray count (30,000) control the number of rays used by the ray tracer. Higher counts improve fidelity at the cost of increased computation time. Six frequency bands are used. Although more bands would increase frequency resolution, the acoustic material coefficients available in the literature are specified only up to 4 kHz, and values for higher bands are set equal to those of the highest defined band [43]. The global volume is set to 1.0, as with the default value of 0.25 for the initial pressure value little sound energy remained for the reflections. All remaining parameters retain their default values as specified in the SoundSpaces 2.0 documentation [43].

B. Material Properties

Three coefficients must be specified for each surface. The *absorption coefficient* $\alpha \in [0, 1]$ quantifies the fraction of incident sound energy absorbed by the material rather than reflected [49]. In practice, measured absorption coefficients can exceed unity due to edge diffraction effects at the sample boundary and the contribution of the material’s side area, which is not included in the nominal surface area used for the calculation [50]. The *scattering coefficient* $s \in [0, 1]$ describes the fraction of reflected energy scattered diffusely rather than in the specular direction, ranging from fully specular ($s = 0$) to fully diffuse ($s = 1$). The *transmission coefficient* $\tau \in [0, 1]$ represents the fraction of incident energy that passes through the material, where $\tau = 0$ indicates complete blocking and $\tau = 1$ full transmission.

Due to the limited availability of scattering and transmission data for the specific materials in the Treated Room, values from the Matterport3D example implementation of SoundSpaces 2.0 are reused [43], [44]. Absorption values are obtained from published measurement data for the corresponding room materials [51], [52], [53], [54]. Table VIII lists the material assignments and their acoustic properties per frequency band.

Appendix C Additional Results

In this appendix, additional figures and more detailed results for the experimental validation are included.

A. Experiment 1.2: Microphone Channel Fusion

Table IX compares early and late fusion for microphone channel combination. In early fusion, the r spectrograms are concatenated along the channel dimension before feature extraction. In late fusion, the r spectrograms are processed independently through the same share weight feature extractor, after which the resulting embeddings are concatenated before the prediction head. No significant difference in performance is observed, and early fusion is adopted for all subsequent experiments.

Surface (<i>material, MP3D material</i>)	Coefficient	Source	Frequency Band (Hz)					
			125	250	500	1000	2000	4000
Floor (<i>office carpet, acoustic tile</i>)	α	[51]	0.500	0.700	0.600	0.700	0.700	0.500
	s	MP3D	0.100	0.150	0.200	0.200	0.250	0.300
	τ	MP3D	0.050	0.040	0.030	0.020	0.005	0.002
Walls (<i>melamine foam, acoustic tile</i>)	α	[52]	0.200	0.990	0.990	0.990	0.990	0.990
	s	MP3D	0.100	0.150	0.200	0.200	0.250	0.300
	τ	MP3D	0.050	0.040	0.030	0.020	0.005	0.002
Ceiling (<i>fiberglass tile, acoustic tile</i>)	α	[53]	0.400	0.900	0.990	0.950	0.990	0.990
	s	MP3D	0.100	0.150	0.200	0.200	0.250	0.300
	τ	MP3D	0.050	0.040	0.030	0.020	0.005	0.002
Human (<i>human, carpet</i>)	α	[54]	0.290	0.430	0.510	0.680	0.710	0.730
	s	MP3D	0.100	0.100	0.150	0.200	0.300	0.450
	τ	MP3D	0.004	0.008	0.006	0.002	0.001	0.001
Boxes (<i>plastic, plastic</i>)	α	MP3D	0.020	0.020	0.020	0.030	0.040	0.050
	s	MP3D	0.300	0.350	0.400	0.450	0.500	0.550
	τ	MP3D	0.010	0.010	0.010	0.010	0.005	0.005

Table VIII: Material properties per surface type and frequency band. The material types in brackets are descriptions of the surface material, and which MP3D material was used [44]. The coefficients are rounded to three decimals.

Fusion Type	Number of Microphones		
	10	4	2
<i>Reproduced VGGish (16 kHz)</i>			
Early Fusion	19.8 (17.4)	30.6 (29.8)	27.7 (20.9)
Late Fusion	17.1 (14.1)	18.6 (15.7)	26.6 (23.5)
<i>Reproduced VGGish (48 kHz)</i>			
Early Fusion	15.5 (13.3)	18.7 (18.2)	21.4 (19.8)
Late Fusion	17.1 (17.6)	16.2 (14.6)	21.3 (17.2)

Table IX: Experiment 1.2 - Microphone Channel Fusion. Errors are given as mean (std) in cm.

B. Experiment 2: Baseline Simulation

Figure 16 (top row) illustrates the localization results for the baseline simulated room. Consistent with the quantitative results in Table IV, performance degrades as the number of microphones decreases. In particular, the single-channel configuration exhibits a collapse of predictions onto a small set of spatial modes, reflecting the inherent ambiguity of localization from a single RIR. This effect is less pronounced for the 2- and 4-channel configurations, where additional inter-channel information can constrain the predictions.

C. Experiment 2.1: Room Asymmetry

To assess whether geometric asymmetry improves localization, we introduce a localized perturbation in the simulated environment by adding an absorptive block near the farthest microphone (Fig. 17). Due to computational constraints, only the 4-, 2-, and 1-channel configurations are trained. Table X and Fig. 16 (bottom row) show that this modification does not yield consistent improvements. For multi-channel configurations (2 and 4 microphones), differences are small relative to the observed variance, while single-channel performance degrades. These results suggest that simple geometric asymmetry alone is insufficient to reproduce the performance observed in real data, indicating that other factors (e.g., material properties,

sensor characteristics, or noise) may play a more significant role.



Fig. 17: Microphone 1 in the Treated Room (left) and in the simulated replica (right). In both rooms, a cuboid of absorptive material is placed on the right of the microphone, introducing room asymmetry. The left image is reprinted from [10].

Room Layout	Number of Microphones		
	4	2	1
<i>Reproduced VGGish (48 kHz)</i>			
"Symmetrical"	28.3 (19.2)	37.0 (31.4)	118.0 (60.5)
"Asymmetrical"	26.2 (17.5)	38.7 (36.3)	133.9 (47.4)

Table X: Experiment 2.1 - Room Asymmetry. Localization errors are given as mean (std) in cm. The "symmetrical" room is the rectangular room, the "asymmetrical" is the same room, but with an added absorptive cuboid on the right of microphone 1.

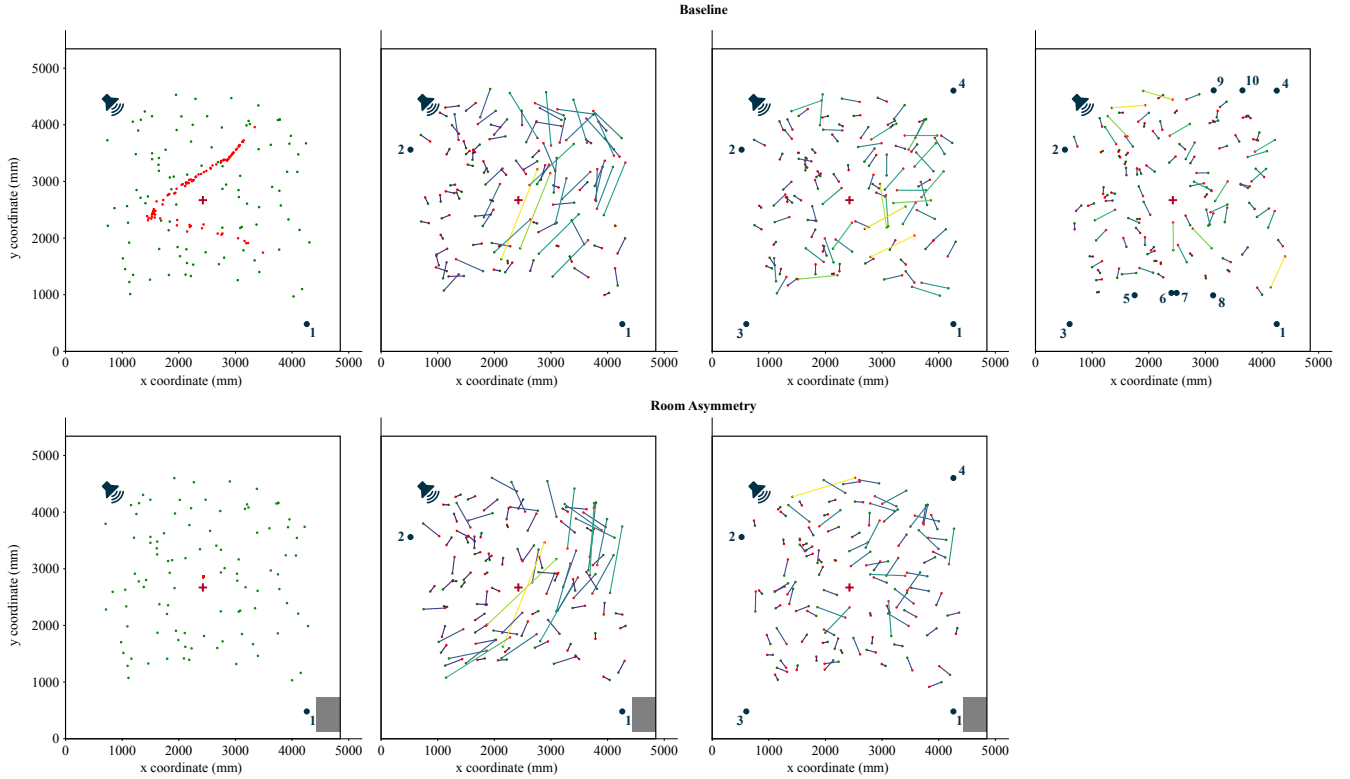


Fig. 16: Test results for localization in the baseline replicated room (Experiment 2, top row) and the “asymmetrical” room (Experiment 2.1, bottom row). For the 2-channel and 4-channel configurations, the ground truth (green) and predictions (red) are connected to visualize the localization error. For the 1-channel case, these connections are omitted, as the errors are large and would obscure the predictions. We refer to Section C-C for more details on the asymmetrical room experiment.

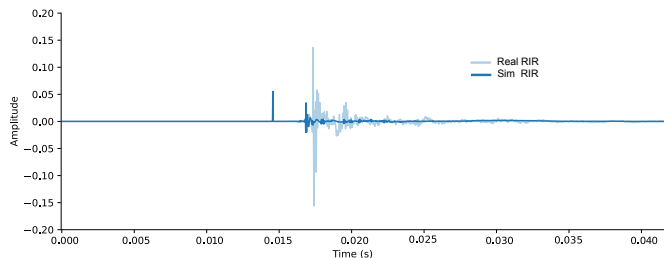


Fig. 18: Comparison of real and simulated RIR for Microphone 1. The real RIR shows a 2.7 ms delay in the first peak (direct path) relative to the simulated RIR, highlighting a temporal mismatch between domains.

D. Experiment 2.2: RIR Modification

We compare real and simulated RIRs and observe a difference in the timing of the first prominent peak (direct path) (Fig. 18). For a source–receiver distance of 5.1 m, the expected arrival time is approximately 14.9 ms. While the simulated RIR aligns with this estimate (14.6 ms), the measured RIR exhibits a delay of approximately 2.7 ms. To investigate whether this discrepancy contributes to the domain gap, we apply two waveform-level modifications to simulated RIRs: 1) zero-padding to introduce a delay, and 2) removal of the first peak. Table XI shows that these modifications do not

improve cross-domain performance. In particular, $Sim \rightarrow Real$ errors remain high or increase, suggesting that simple temporal alignment is insufficient to bridge the domain gap.

Modification	Train $\rightarrow$ Test	
	Sim $\rightarrow$ Sim	Sim $\rightarrow$ Real
<i>Reproduced VGGish (48 kHz), 4-channel</i>		
Unmodified	28.3 (19.2)	186.8 (92.6)
Zero-padding on the left (140 samples)	25.7 (20.0)	182.2 (87.5)
First peak removed	20.3 (15.2)	253.8 (100.2)

Table XI: Experiment 2.2 - RIR Modification. Errors are given as mean (std) in cm.

E. Experiment 2.3: Object Geometry

We evaluate the effect of object geometry by comparing a cuboid approximating human dimensions with two human mesh configurations (Fig. 19). As shown in Table XII, the cuboid yields the lowest localization error (13.4 cm in the 4-channel case). This may be attributed to its simpler reflection structure, whereas the human meshes introduce more complex scattering effects [55]. These results indicate that object geometry significantly influences simulated RIR characteristics and thus localization performance.

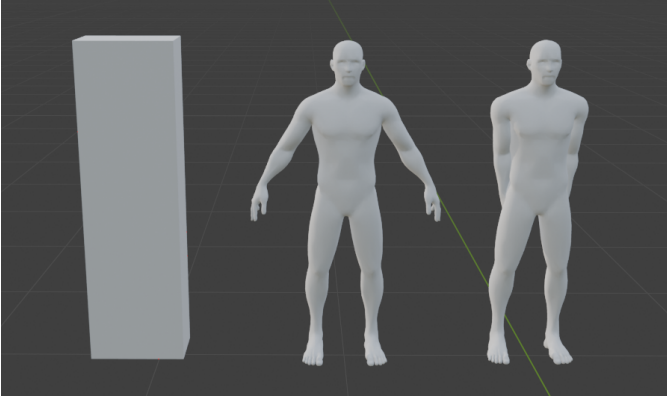


Fig. 19: Cuboid of average human size (1.7x0.5x0.3 m), human with arms spread out, and human with arms at the sides. The latter is used in the simulated replica for the localization task.

Object of interest	Number of Microphones		
	4	2	1
<i>Reproduced VGGish (48 kHz)</i>			
Cuboid of human size	13.4 (12.9)	27.7 (36.3)	80.3 (65.8)
Human arms spread out	31.8 (31.0)	33.7 (24.6)	106.7 (70.6)
Human arms at the sides	28.3 (19.2)	37.0 (31.4)	120.0 (60.4)

Table XII: Experiment 2.3 - Object Geometry. Errors are given as mean (std) in cm.

F. Experiment 2.4: Sampling Rate Comparison

The acoustic simulation is performed at a sampling rate of 48 kHz using six frequency bands. However, material absorption coefficients from the literature are typically specified only up to 4 kHz, and values for higher bands are set equal to those of the highest defined band [43]. Consequently, frequency-dependent behavior above 4 kHz is not accurately modeled, and higher-frequency reflections do not introduce additional material-specific variation. This limitation reduces the physical accuracy of the simulated high-frequency content, although the higher sampling rate still provides improved temporal resolution.

Table XIII compares localization performance for models trained at 16 kHz and 48 kHz on simulated RIRs in the 4-channel setting. The 16 kHz model achieves a slightly lower within-domain error (24.8 vs. 28.3 cm), while cross-domain performance (*Sim*→*Real*) remains nearly identical within the observed variance. The marginal within-domain disadvantage of 48 kHz may be explained by the modeling limitation described above: although higher frequencies can encode additional spatial cues, the lack of accurate material coefficients above 4 kHz limits the realism and variability of these components in the simulated RIRs. On real data, this limitation does not apply in the same way, as the microphones capture the full acoustic response of the room, which explains why 48 kHz consistently outperforms 16 kHz on real RIRs (Table II). The 48 kHz configuration is therefore adopted for

subsequent experiments, as it is expected to better capture fine-grained spatial detail in real acoustic conditions. Future work could address the simulation limitation by incorporating material measurements at higher frequencies.

Model	Train → Test	
	Sim → Sim	Sim → Real
<i>Reproduced VGGish, 4-channel</i>		
16 kHz	24.8 (19.6)	188.6 (92.8)
48 kHz	28.3 (19.2)	186.8 (92.6)

Table XIII: Experiment 2.4 — Sampling Rate Comparison. Errors are given as mean (std) in cm.

G. Experiment 2.5: Microphone Channel Selection

The choice of microphone channel count is guided by the results of Experiments 1 and 2, summarized in Table IV. On real data, the 4-channel model performs close to the 10-channel configuration (18.7 vs. 15.5 cm), and a similar trend is observed in simulation (28.3 vs. 18.2 cm). In contrast, the improvement from two to four channels is substantially larger than from four to ten in both settings. Given that SoundSpaces 2.0 simulates each microphone independently, increasing the number of channels incurs a proportional computational cost. The 4-channel configuration therefore provides a favorable trade-off between spatial resolution and simulation efficiency, and is adopted for Experiments 3 and 4.

H. Experiment 4: Baseline Configuration under Different Seeds

Table V presents the per-box AUC values for the baseline configuration across the original run and three additional random seeds. Overall, the mean per-box AUC is similar across seeds (0.947), with per-box standard deviations generally below 0.004, except for Box 1, which shows higher variability. These results suggest that the model’s performance is relatively consistent across different initializations and that the observed stability is not solely due to thresholding, as most boxes achieve high AUC values under all seeds. Box 1 remains a challenging case, with variable convergence and lower AUC values across seeds, consistent with the patterns observed in the training and validation loss curves. Overall, the findings indicate that the model’s behavior appears reasonably consistent across different initializations.

1) *Configuration Pair Fusion*: Two fusion strategies are evaluated: early fusion, where the spectrograms of both room states are concatenated before feature extraction, and late fusion, where each is encoded separately and the embeddings are concatenated before the classifier. Only early fusion achieves meaningful performance (0.935 mean per-box accuracy), as late fusion collapses to random guessing (0.501) from epoch 1. The loss remains at the random-guessing level of 0.693 throughout all training epochs, suggesting that the shared-weight encoder produces identical embeddings for both inputs, leaving the classifier with no change signal. A likely cause

Box	Baseline	Seed 1	Seed 2	Seed 3	Mean $\pm$ Std
Box 0	1.000	1.000	1.000	1.000	1.000 $\pm$ 0.000
Box 1	0.509	0.513	0.513	0.506	0.510 $\pm$ 0.003
Box 2	0.997	0.995	0.996	0.996	0.996 $\pm$ 0.001
Box 3	0.993	1.000	1.000	0.997	0.998 $\pm$ 0.003
Box 4	1.000	1.000	1.000	0.998	1.000 $\pm$ 0.001
Box 5	0.998	0.988	0.995	0.999	0.995 $\pm$ 0.004
Box 6	1.000	1.000	1.000	0.998	1.000 $\pm$ 0.001
Box 7	0.996	0.992	0.994	0.991	0.993 $\pm$ 0.002
Box 8	0.999	1.000	1.000	0.999	1.000 $\pm$ 0.001
Box 9	0.979	0.983	0.974	0.984	0.980 $\pm$ 0.004
Mean	0.947	0.947	0.947	0.947	0.947 $\pm$ 0.000

Table XIV: Experiment 4 - Per-box AUC values for the baseline configuration and three additional random seeds.

The random-guess baseline is 0.5. Mean and standard deviation are computed across all four runs.

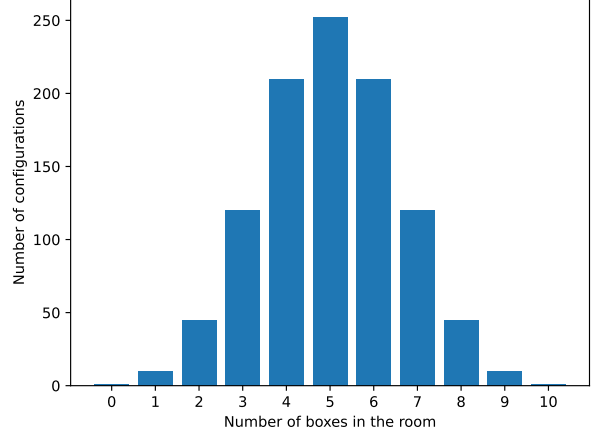


Fig. 20: Dataset distribution for the $B=10$ boxes in the object count regression and binary change detection task.

is that without a contrastive objective, the encoder has no incentive to produce different representations for different configurations. The gradient signal reaching it through the classifier disappears as soon as the classifier settles on a constant prediction. Early fusion avoids this by concatenating both room states before the encoder, forcing it to learn representations sensitive to their differences. Whether late fusion can be made to converge through hyperparameter tuning is left as future work.

2) *Ablation of Architectural Modifications*: The binary change detection model adds two modifications to the baseline VGGish: group normalization after each convolution-ReLU pair, and Kaiming weight initialization. To assess their individual contributions, three ablated variants are trained alongside the full model. Removing group normalization causes complete collapse to random guessing regardless of whether Kaiming initialization is used (0.501 and 0.497, respectively), suggesting that group normalization is the critical modification for training stability. Removing only Kaiming initialization while retaining group normalization also collapses to random guessing (0.499), though we hypothesize this reflects a less severe failure: without Kaiming, the initial weights are poorly scaled for a ReLU-after-GroupNorm network, which may cause slow convergence rather than a fundamental instability. Distinguishing between these failure modes would require tracking activation statistics and gradient norms during training, which is left as future work.

Appendix D

Random Baseline Details

Each of the $B = 10$ boxes is independently present or absent, so the total object count N follows a binomial distribution $N \sim \text{Bin}(10, 0.5)$, as shown in Fig. 20. This distribution has mean $\mu_N = 5$ and variance $\sigma_N^2 = 2.5$. Here, the mean and median coincide at 5. The optimal constant predictor under MAE is the median of the label distribution [45], which here equals 5. The expected MAE of this constant predictor is

$$\mathbb{E}[|N - 5|] = \sum_{k=0}^{10} |k - 5| \binom{10}{k} 2^{-10} = \frac{1260}{1024} \approx 1.23. \quad (4)$$