SRP 16-BAND UPLAM WITH MASK R-CNN FOR AUDIOVISUAL SELD

Technical Report

Rafał Foltyniewicz[1]*, Rafał Kaczmarek*[1]*, Michał Olejnik*[1]*, Iwan Ryzenkow*[1]*, Bogdan Jastrzębski*[1]*,*

[1] Samsung Research Poland,
{r.foltyniewi, r.kaczmarek, m.olejnik2, i.ryzenkow, b.jastrzebsk}@samsung.com

ABSTRACT

We present an audiovisual sound event localization and detection system based on a modified Mask R-CNN with a ResNet-50 Feature Pyramid Network backbone, taking 19-channel inputs composed of 3 RGB video channels and 16 UpLAM acoustic feature maps extracted from the tetrahedral microphone array. The model is trained with progressive backbone unfreezing, class-balanced sampling with inverse-frequency weights (cap=30), focal loss, and data augmentation including azimuth rotation, horizontal flip, RGB dropout, and audio band masking.

***Index Terms*—** sound event localization and detection, data augmentation, audio-visual machine learning, synthetic data generation

1. INTRODUCTION

DCASE 2026 Task 3 advances sound event localization and detection (SELD) through semantic acoustic imaging, focusing on the integration of spatial audio and audiovisual scenes. This challenge emphasizes the development of robust systems capable of identifying and localizing sound events in complex acoustic environments.

The backbone dataset for this challenge is STARSS23 [1] which offers multichannel audio and video data with spatiotemporal annotations for 13 sound events. STAIRS26 [2] complements it, introducing 32-channel recordings and acoustic energy maps for dense field estimation.

This report outlines the technical approach for participating in challenge with the primary aim of redesigning the UpLAM model to operate with 16 perceptually relevant Mel frequency bands and leveraging AudibleLight for scalable synthetic data generation.

2. SYSTEM ARCHITECTURE AND TRAINING PROCESS

This section describes modifications done to UpLAM, generation of additional synthetic training data and training process.

2.1. UpLam modification

The provided UpLAM [3] configuration produced 9 frequency bands linearly spaced between 1,500 Hz and 4,500 Hz, covering only a narrow 3 kHz window of the speech presence range. The upgraded 16-band UpLAM was proposed with a mel-scale frequency layout spanning 50 Hz to 4,500 Hz, extending low-frequency coverage that better reflecting the perceptually relevant spacing of spatial acoustic cues. Especially sounds of *steps*, *knocks* and low frequency features of *speech*. For training, STAIRS26 dataset was used. It is a 32-channel recording that pairs tetrahedral microphone captures with high-order reference signals in the exact target deployment domain. As a result, the backbone receives 19 input channels (16 acoustic + 3 RGB) instead of the original 12.

Model was trained using standard training pipeline from [4] and trained for 50 epoch. Best model was chosen by using lowest validation loss on STAIRS2026 dev-test, achieving 0.0011 average MSETVLoss, 13.27 Peak SNR and 3.7e-5 Abs Diff. This best model (based on ValidationLoss) was used for our sibmission.

2.2. Synthetic Data Generation

Given the growing use of synthetic data across different machine learning tasks and the limited amount of data in the original STARSS23 dataset, we chose to use AudibleLight [5] to generate additional data.

AudibleLight is a Python package that provides a unified API for generating synthetic soundscapes through ray-traced simulations and parametric room impulse responses (RIRs). It integrates frameworks like SpatialScaper, SoundSpaces, and Pyroomacoustics to ensure scalable acoustic diversity.

For the generation process, we set parameters to include 5 to 20 static events per 60-second scene, with no moving events. The maximum overlap was set to 4, and the reference decibel level was -50 dB. Events were positioned between 0.5 to 3 meters from the microphone, with a height limit of 0.3 meters relative to the microphone position, which was randomly chosen within a randomly selected mesh. We used the Eigenmike32 as microphone type parameter, with video resolution set to 1920 x 960 pixels at 10 fps. There was a 50% chance of adding noise, randomly selected from available options, and the video low power mode was disabled to maintain higher quality meshes.

After generation, the audio was resampled to 24 kHz and 4 appropriate tetrahedral channels were selected. The video was resized to 360:180 resolution, with frames saved as PNG files to meet task specifications.This process produced 122 recordings, totaling in 2 hours and 2 minutes of synthetic data.

2.3. Training Configuration and Optimization

The system was trained as a multi-task learning framework for joint sound event detection, localization, and distance estimation. The architecture utilizes the original ResNet-50 [6] backbone with a Feature Pyramid Network (FPN) [7], processing a 19-channel input consisting of 3 RGB channels and 16 acoustic feature maps derived from embeddings of the re-optimized UpLAM.

Optimization was performed using the AdamW [8] optimizer with a differentiated learning rate strategy across 12 parameter groups, ranging from $2 \times 10^{-}5$ for the box predictor to $2 \times 10^{-}4$

for the audio stem. To handle learning rate decay, a *ReduceLROnPlateau* scheduler was employed, reducing the learning rate by a factor of 0.5 after 3 epochs of validation loss stagnation. The total loss function is a weighted combination of the following components:

- **Classification:** Focal Loss (γ=2.0) with inverse-frequency class weighting to mitigate imbalance.
- **Box Regression:** Smooth L1 Loss.
- **Energy Mapping:** Weighted MSE loss for energy mask prediction (weight=5.0) and full-map prediction (weight=10.0).
- **Distance Estimation:** L1 Loss, normalized by a factor of 500.0.

2.4. Progressive Unfreezing and Stability

To prevent catastrophic forgetting of pretrained features while adapting the model to audio-visual modalities, a staged unfreezing schedule was implemented. Layer 2 remained permanently frozen to preserve low-level edge and texture detectors. The remaining layers were unfrozen as follows:

- *Epochs 1–2:* Layer 1 and input stems (initial vocabulary acquisition).
- *Epochs 3–6:* Addition of Layer 4 (high-level semantic adaptation).
- *Epochs 7+:* Addition of Layer 3 (mid-level feature refinement).

Each transition included a 5-epoch grace period where early stopping was suspended, and optimizer states were reset to allow the model to stabilize under the new parameter space.

2.5. Data Augmentation and Class Balancing

To improve generalization, we applied extensive audio-visual augmentations, including random azimuth rotation of acoustic maps, horizontal flipping (50% probability), RGB jitter (80% probability), and random occlusion of up to four frequency bands. We also introduced a modality dropout (15%) where RGB channels were replaced with the ImageNet mean to ensure the model did not overrely on visual cues. To address the inherent class imbalance in the STAIRS26 dataset, per-class weights were also employed.

3. EXPERIMENTAL METHODOLOGY AND RESULTS

This section describes approach to inferring model, generating results and evaluation of models.

3.1. Inference modification

We experimented with modifying inference script to generate better spatial representation of acoustic energy maps.

- **Dense extraction instead of sparse peaks**: The original baseline returned 20 sparse local maxima per detection. We replaced this with dense sampling: all pixels in the energy map above dense_thr × max are exported as triplets.
- **1D row-major sampling instead of 2D grid**: We experimented with 2D grid sampling (n_rows × n_cols lattice) but AP@0.75 dropped. We reverted to 1D: every N-th pixel in row-major order from the sorted pixel list. This keeps points concentrated at the blob center where intensity is highest.
- **Per-class score threshold**: We added a separate thresholds for some of individual classes. Class Knock (model class 13) was assigned 0.85 instead of the global value — without this, a single sequence's JSON file exceeded the competition's 20 MB per-file limit due to a flood of false positives.
- **Systematic inference tuning**: We run additional parameter search to check score_thr and min_age combinations across a subset of sequences. The optimal settings found were score_thr=0.70, min_age=2 and max_missed=2.

3.2. Evaluation method

The proposed system was evaluated using the STARSS23 development test split (78 clips) and STAIRS26 acoustic maps across 13 sound event classes. In accordance with DCASE2026 Task 3 guidelines, input data was restricted to 4-channel tetrahedral microphone array (MIC) recordings, while STAIRS26 provided the high-resolution maps used for training and validation. Performance was measured using two primary metrics: mean Average Precision (mAP), which quantifies localization and detection accuracy, and the Pearson correlation coefficient (Pearson r), which assesses the fidelity of the reconstructed acoustic energy distribution.

Our submission contains 3 runs. All models employed acoustic features extracted from our internally trained 16-band UpLAM model. The configurations for each run are as follows:

- **SRP-UPL-MRCNN (Run Index 1)**: Trained on a combination of the original STARSS23 dataset and generated synthetic data, utilizing a modified inference script. It used 15000 train frames and 3000 val frames per epoch.
- **SRP-UPL-MRCNN-2 (Run Index 2)**: Trained exclusively on the original dataset using the original inference script. It used whole train and eval datasets per epoch. Used as baseline 16-band UpLAM approach for comparison.
- **SRP-UPL-MRCNN-3 (Run Index 3)**: Trained exclusively on the original dataset utilizing a modified inference script. It used the same distribution of frames as first run.

The performance results for these three configurations are detailed in Table 1.

Table 1: Evaluation metrics with Macro and Micro averaging. The top values in each row represent Macro metrics, while the bottom values represent Micro metrics.

Run Index	AP@0.25	AP@0.5	AP@0.75	mAP	Pearson r
Baseline	0.0010	0.0000	0.0000	0.0003	0.1358
	0.0032	0.0000	0.0000	0.0011	0.4429
1	**0.0277**	**0.0069**	**0.0001**	**0.0116**	0.2796
	0.0279	0.0072	**0.0001**	0.0117	0.4270
2	0.0025	0.0000	0.0000	0.0009	**0.3455**
	0.0112	0.0000	0.0000	0.0037	0.4189
3	0.0212	0.0046	0.0000	0.0086	0.3287
	0.0308	**0.0080**	0.0000	**0.0129**	**0.4791**

4. CONCLUSION

Experiments shown that the usage of wider range of frequency bands in UpLAM improves all metrics. Additionally run with extra synthetic data in train lead to improvement of Macro mAP - showing better quality of detecting classes with smaller representation in original STARSS23 dataset.

Modifications in the inference script also had big impact on the quality of generated results. Dense extraction was the single biggest improvement — AP@0.50 jumped from near zero to useful values because the challenge evaluator (grid data interpolation) needs a dense point cloud to reconstruct a smooth Gaussian blob. Parameters tuning maximized AP@0.50 and Pearson r jointly while reducing total number of detections 3× in comparison to the naive threshold. Additionally, 1D sampling gave better soft-IoU at the stricter 0.75 threshold.

5. A NOTE ON NUMERICAL APPROACHES TO LAM

The LAM pipeline relies on APGD-computed acoustic intensity maps as ground truth for training. We investigated whether exact numerical solvers (Chambolle-Pock, subgradient descent, ADMM) could produce improved estimates. Figure 1 shows representative reconstructions on real LOCATA Eigenmike recordings.

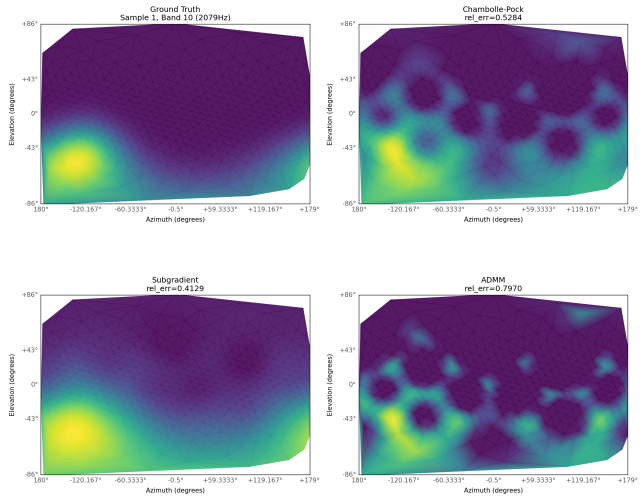

Figure 1: Acoustic map reconstructions on LOCATA data. Chambolle-Pock and ADMM converge to the exact minimizer but exhibit checkerboard artifacts. Subgradient descent and the APGD ground truth remain smooth due to early stopping.

ADMM successfully recovers the dominant acoustic source, confirming the mathematical validity of the imaging operator. However, the Hessian $H_{ij} = |a_i^H a_j|^2$ is singular: with 32 microphones mapping onto 484 spatial directions at sub-wavelength resolution, the system is fundamentally underdetermined. Adding quadratic regularization renders the problem well-posed, enabling convergence within tens of iterations.

The true minimizer of the loss, however, exhibits characteristic checkerboard and ringing artifacts (Fig. 1). ADMM, as the solver that achieves the lowest objective value, reveals these artifacts most clearly. The APGD-based ground truth used in the LAM pipeline produces smooth, visually plausible maps precisely because it is stopped early, before reaching this minimizer. The regularizer promotes piecewise constancy and sparsity but does not encode the physical structure of compact, contiguous acoustic sources on the sphere.

This has implications for training: the LAM model learns to replicate APGD's early-stopped estimates, which are smooth approximations to a loss function whose exact solution does not correspond to the desired acoustic map.

A further consequence is that early stopping precludes warm-starting across consecutive frames. In principle, since acoustic scenes change slowly, the solution from one time frame provides an excellent initialization for the next. This would amortize solver cost: rather than running hundreds of iterations per frame, a handful of fine-tuning steps would suffice, making exact solvers practical for streaming applications. However, if we rely on early stopping to suppress artifacts, warm-starting becomes self-defeating: it accelerates convergence toward the exact minimizer, which is precisely what early stopping was designed to avoid. Maintaining both a running estimate and artifact-free maps would require a frame-dependent stopping rule calibrated to the distance already traveled—an open problem under the current objective.

6. REFERENCES

[1] K. Shimada, A. Politis, P. Sudarsanam, D. Krause, K. Uchida, S. Adavanne, A. Hakala, Y. Koyama, N. Takahashi, S. Takahashi, T. Virtanen, and Y. Mitsufuji, "Starss23: An audio-visual dataset of spatial recordings of real scenes with spatiotemporal annotations of sound events," 2023. [Online]. Available: https://arxiv.org/abs/2306.09126

[2] I. R. Roman, A. Politis, K. Shimada, H. Cheston, P. Sudarsanam, D. DÍAZ-GUERRA APARICIO, Y. Sun, T. Shibuya, T. Shusuke, and Y. MITSUFUJI, "Stairs26: Sony-tau acoustic images of real-world scapes 2026," Apr. 2026. [Online]. Available: https://doi.org/10.5281/zenodo.18171005

[3] A. S. Roman, I. R. Roman, and J. P. Bello, "Latent acoustic mapping for direction of arrival estimation: A self-supervised approach," 2025. [Online]. Available: https://arxiv.org/abs/2507.07066

[4] A. S. Roman. (2024) Lam. GitHub repository. [Online]. Available: https://github.com/adrianSRoman/LAM

[5] H. Cheston, A. Stepien, J. Azcarreta, A. S. Roman, C. Chen, C. Bilen, and I. R. Roman, "Audiblelight: A controllable, end-to-end api for soundscape synthesis across ray-traced & real-world measured acoustics," in *DMRN+20: Digital Music Research Network One-Day Workshop 2025*, 2025.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[7] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," *CoRR*, vol. abs/1612.03144, 2016. [Online]. Available: http://arxiv.org/abs/1612.03144

[8] I. Loshchilov and F. Hutter, "Fixing weight decay regularization in adam," *CoRR*, vol. abs/1711.05101, 2017. [Online]. Available: http://arxiv.org/abs/1711.05101