



CHALMERS
UNIVERSITY OF TECHNOLOGY



UNIVERSITY OF GOTHENBURG

Multi-View Pedestrian Detection Without Scene-Specific Fine-Tuning via 3D Foundation Models

Master's thesis in Computer science and engineering

BOWEN LAI & JUNCHAO ZHANG

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2026

MASTER'S THESIS 2026

Multi-View Pedestrian Detection Without Scene-Specific Fine-Tuning via 3D Foundation Models

BOWEN LAI & JUNCHAO ZHANG



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2026

Multi-View Pedestrian Detection Without Scene-Specific Fine-Tuning via 3D Foundation Models

BOWEN LAI & JUNCHAO ZHANG

© BOWEN LAI & JUNCHAO ZHANG, 2026.

Supervisor at CSE: Knut Åkesson, Chalmers University of Technology

Supervisor at Company: Erik Brorsson, Volvo Group

Examiner: Peter Damaschke, Chalmers University of Technology

Master's Thesis 2026

Department of Computer Science and Engineering

Chalmers University of Technology and University of Gothenburg

SE-412 96 Gothenburg

Telephone +46 31 772 1000

Typeset in L^AT_EX

Gothenburg, Sweden 2026

Multi-View Pedestrian Detection Without Scene-Specific Fine-Tuning via 3D Foundation Models

BOWEN LAI & JUNCHAO ZHANG

Department of Computer Science and Engineering
Chalmers University of Technology

Abstract

Multi-view pedestrian detection in occluded scenes often relies on scene-specific 3D reconstructions, which are accurate but require lengthy optimization. This thesis investigates whether an off-the-shelf 3D vision foundation model (MapAnything) can provide metric 3D point clouds for this task without scene-specific fine-tuning.

We first characterize the foundation model’s geometric error, finding that calibrated inputs yield manageable scale drift correctable via ground-plane scaling, whereas uncalibrated inputs cause severe frame-to-frame distortion (averaging 66.0 cm spatial drift), degrading detection accuracy from 59.9 MODA to 29.1 MODA.

To integrate this foundation-model geometry into a detector, we build a differentiable bird’s-eye view (BEV) pipeline based on Lift-Splat-Shoot and PointPillars. To handle the 30–80 cm point cloud dispersion typical of foundation models, we use dilated convolutions to enlarge the BEV backbone’s receptive field. This architecture achieves 68.7 MODA on Wildtrack and 64.0 MODA on MultiviewX. Compared to a per-scene optimized baseline (DCHM), which reaches 83.0 MODA in our single-iteration supervised reproduction and approximately 95 MODA in the best supervised results reported by its authors, the gap reflects the limitations of off-the-shelf foundation geometry.

Finally, we show that auxiliary mechanisms like soft-depth sampling can cause over-smoothing, and cross-subset experiments reveal that BEV convolutional networks severely overfit to the training camera rig layout (dropping up to 28.35 MODA when testing on unseen camera subsets).

Keywords: multi-view pedestrian detection, 3D foundation models, MapAnything, bird’s-eye view, Lift-Splat-Shoot, dilated convolutions, camera-rig dependency, error characterization, geometric generalization.

Acknowledgements

We wish to express our sincere gratitude to our academic supervisor, Professor Knut Åkesson at Chalmers University of Technology, our industrial supervisor, Erik Brorsson at Volvo Group, and our examiner Professor Peter Damaschke at Chalmers University of Technology, for their invaluable guidance, intellectual support, and insightful discussions throughout this research.

We also thank Chalmers University of Technology and Volvo Group for providing the computational resources and stimulating research environment that make this project possible.

Finally, we extend our heartfelt gratitude to our families and friends for their continuous patience, encouragement, and understanding during the course of our studies.

Attribution and Declaration of Originality

This thesis builds upon the open-source software foundations of DCHM [16] and MvCHM [18]. Specifically, standard 3D point cloud voxelization, PointPillar feature encoding, and the basic BEV detection head are adapted from MvCHM. Camera-wise ground plane scale alignment and detection compensation are independently re-implemented based on the algorithmic formulations described in the literature.

The core original contributions of this thesis—namely, the systematic integration and evaluation of MapAnything as a metric 3D foundation-model backbone replacing scene-specific Gaussian Splatting; the theoretical error characterization of low-rank versus high-rank geometric distortions; the design of the DilatedBEVBackbone and FusedLSSLift feature projection; the mechanistic attribution of over-smoothing and pseudo-point noise; and the rigorous empirical demonstration of camera-rig dependency through cross-subset experiments—represent the original research conducted by the authors.

AI Usage Statement

In the course of this study, we employ Stanford’s Storm AI to aid in identifying corresponding academic articles and technical materials. Artificial intelligence applications are employed for grammatical verification, language refinement, and the augmentation of expressive clarity. All research procedures, experimental frameworks, data evaluations, and overall interpretations are executed independently by the authors of the paper. AI techniques function solely as supplementary assets and do not affect the originality or scholarly integrity of the work.

Bowen Lai & Junchao Zhang, Gothenburg, 2026

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

BEV	Bird’s-Eye View
CLEAR	Classification of Events, Activities and Relationships
CNN	Convolutional Neural Network
CRF	Conditional Random Field
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DCHM	Depth-Consistent Human Modeling
EMA	Exponential Moving Average
FN	False Negative
FOV	Field of View
FP	False Positive
GS	Gaussian Splatting
HDC	Hybrid Dilated Convolutions
LSS	Lift-Splat-Shoot
MA	MapAnything
MODA	Multiple Object Detection Accuracy
MODP	Multiple Object Detection Precision
MvCHM	Multi-view Detection with Cardboard Human Modeling
MVDet	Multi-view Detection with Feature Perspective Transformation
MVDeTr	Multi-view Detection with Shadow Transformer
MVFP	Multiview Feature Perspective Transformation
MVUDA	Multi-view Unsupervised Domain Adaptation
MVX	MultiviewX benchmark dataset
NMS	Non-Maximum Suppression
P2V	Point-to-Voxel conversion
PR	Precision–Recall
RGB	Red, Green, Blue
ROI	Region of Interest
SAM	Segment Anything Model
SfM	Structure from Motion
Sim3	Similarity Transformation in 3D (7-DOF: rotation, translation, scale)
SLIC	Simple Linear Iterative Clustering

SOTA	State of the Art
TP	True Positive
UDA	Unsupervised Domain Adaptation
UMPD	Unsupervised Multi-view Pedestrian Detection
VFE	Voxel Feature Encoding
ViT	Vision Transformer
WT	Wildtrack benchmark dataset
YOLO	You Only Look Once

Nomenclature

Below is the nomenclature of symbols and notation used throughout this thesis.

Camera and Multi-View Geometry

$\mathcal{V}$	Set of camera viewpoints, $\mathcal{V} = \{1, 2, \dots, C\}$
$\mathbf{K}_c$	Camera intrinsic matrix for viewpoint $c \in \mathcal{V}$
$\mathbf{R}_c$	World-to-camera rotation matrix for viewpoint c
$\mathbf{t}_c$	World-to-camera translation vector for viewpoint c
$\mathbf{C}_c$	Camera center position in world coordinates, $\mathbf{C}_c = -\mathbf{R}_c^T \mathbf{t}_c$
$\mathbf{P}_w$	3D point in world coordinates, $\mathbf{P}_w = (X, Y, Z)^T \in \mathbb{R}^3$
$\mathbf{u}_c$	2D pixel coordinate on the image plane of camera c , $\mathbf{u}_c = (u, v)^T$
I_c	RGB image captured from camera c
D^c	Metric depth map corresponding to camera viewpoint c
M_{ground}^c	Binary ground-plane semantic mask for camera c
M_{fg}^c	Binary foreground pedestrian segmentation mask for camera c

Geometric Foundation and Sim3 Alignment

$\mathbf{S}_{sim3}$	Similarity transformation in Sim(3) (7 degrees of freedom)
s_{sim3}	Global isotropic scale factor in similarity transformation
$\mathbf{R}_{sim3}$	Global orthonormal rotation matrix in similarity transformation
$\mathbf{t}_{sim3}$	Global translation vector in similarity transformation
e_{sim3}	Mean Euclidean camera center residual after Sim3 alignment (in cm)
α_c	Closed-form per-camera ground plane scale correction scalar
D_{ground}^c	Theoretical analytical ground plane metric depth map
D_{pred}^c	Raw predicted depth map from the 3D foundation model

BEV Feature Transformation and Detection Network

$\mathbf{f}_u^c$	Deep convolutional feature sampled from the c -th camera feature map
$\mathbf{F}_{bev}$	Encoded 2D feature map scattered on the bird's-eye view plane
d	Convolutional dilation rate
k	Convolutional kernel size
RF	Effective spatial receptive field in physical coordinates (in cm)
$\hat{\mathbf{Y}}$	Predicted continuous pedestrian occupancy probability heatmap
$\mathbf{Y}$	Ground-truth Gaussian-rendered occupancy target heatmap
$\mathcal{L}_{focal}$	Modified focal loss for dense keypoint and heatmap regression
σ_{soft}	Gaussian standard deviation along the line-of-sight for soft depth sampling
K_{soft}	Number of stochastic sampling points per ray for soft depth sampling

Contents

List of Acronyms	ix
Nomenclature	xii
List of Figures	xxi
List of Tables	xxiii
1 Introduction	1
1.1 Background and Motivation	1
1.2 Problem Statement and Research Questions	2
1.3 Summary of Contributions	2
1.4 Thesis Outline	3
2 Theory and Related Work	5
2.1 Multi-View Pedestrian Detection	5
2.1.1 Supervised Feature Transformation Paradigms	5
2.1.2 Label-Free and Self-Supervised Paradigms	6
2.2 3D Gaussian Splatting and Explicit Scene Modeling	6
2.2.1 Mathematical Parameterization of 3D Gaussians	6
2.2.2 Differentiable Splatting and Depth Rendering	7
2.2.3 3D-GS as a Consistency Prior in Multi-View Surveillance . . .	7
2.3 3D Vision Foundation Models	8
2.3.1 From Monocular Priors to Multi-View Foundation Architectures	8
2.3.2 The MapAnything Framework	8
2.4 Multi-View Feature Transformation and BEV Representation	10
2.4.1 The Lift-Splat-Shoot (LSS) Paradigm	10
2.4.2 PointPillars Representation	10
2.5 Multi-View Domain Shifts and Camera-Rig Invariance	11
3 Methods	13
3.1 System Architecture Overview	13
3.2 Geometric Foundation: MapAnything Inference and Error Character- ization	14
3.2.1 Calibrated vs. Uncalibrated Operational Modes	14
3.2.2 Sim3 Alignment and Error Decomposition	14

3.2.3	Closed-Form Ground-Plane Median Scale Alignment	15
3.3	End-to-End BEV Detection Pipeline	16
3.3.1	Feature Representation: Fused Point Cloud vs. Per-Camera Frustum	16
3.3.2	Deep Feature Projection via FusedLSSLift	16
3.3.3	PointPillars Discretization and Scattering	17
3.4	Architectural Adaptations for Foundation Geometry Noise	17
3.4.1	Cross-View Point Dispersion and DilatedBEVBackbone	17
3.4.2	Loss Function and Heatmap Decoding	18
3.5	Alternative Architectural Mechanisms Under Study	18
4	Results	21
4.1	Experimental Setup	21
4.1.1	Datasets	21
4.1.2	Evaluation Metrics	23
4.1.3	Implementation Details	23
4.2	Foundation-Model Geometric Evaluation and Performance Ceilings	23
4.2.1	Calibrated vs. Uncalibrated Geometry	23
4.2.2	Full-Camera Benchmark Performance and Structural Ceiling	26
4.3	Architectural Ablations and Mechanism Discovery	27
4.3.1	Analysis of Observations	27
4.3.2	Training Loss Convergence Dynamics	29
4.4	Cross-Subset Generalization and Camera-Rig Dependency	30
4.4.1	Cross-Subset Evaluation Matrices	30
4.4.2	Discussion on Camera-Rig Specificity	30
4.5	Qualitative BEV Diagnostics and Precision–Recall Trade-offs	31
4.5.1	Qualitative Heatmap Diagnostics	31
4.5.2	Precision–Recall Pareto Analysis	33
5	Discussion	35
5.1	Findings and Performance Bounds	35
5.2	Architectural Insights and Interaction Mechanisms	36
5.3	Camera-Rig Specificity and Calibration Invariance	37
5.4	Limitations and Future Work	37
5.4.1	Practical Limitations	37
5.4.2	Future Research Directions	38
	Bibliography	39
A	Supplementary Material	I
A.1	Domain Gap Quantification for Multi-View Pedestrian Detection	I
A.1.1	Motivation and Problem Formulation	I
A.1.2	MVFP Detector and Experimental Design	I
A.1.3	Cross-Domain Experimental Results	III
A.1.4	Type 1+2 Domain Gap Quantification	III
A.1.5	Asymmetry Analysis	III
A.2	Umeyama Sim3 Alignment Derivation	IV

A.3 Hyperparameter Sweeps and Checkpoint Index	VI
--	----

List of Figures

3.1	Schematic overview of the proposed end-to-end foundation-model multi-view pedestrian detection pipeline.	13
4.1	Camera layout of the Wildtrack dataset. Seven elevated cameras surround a 12×36 m ² labeled ground plaza (dashed boundary). Cameras are color-coded by cross-subset partition: Subset A (cams 1, 3, 5, 7; blue), Subset B (cams 2, 4, 5, 6; orange), and shared center cam 5 (purple).	22
4.2	Camera layout of the MultiviewX dataset. Six calibrated cameras surround a 16×25 m ² labeled ground plaza (dashed boundary). Cameras are color-coded by cross-subset partition: Subset A (cams 1, 2, 6; blue) and Subset B (cams 3, 4, 5; orange).	22
4.3	Per-camera RGB frames (top row) and corresponding MapAnything depth predictions D_{pred}^c (bottom row) for all seven Wildtrack cameras on frame 00000361. Warmer colors denote smaller (closer) metric depth; white silhouettes correspond to pedestrians segmented against the ground plane.	24
4.4	Top-view comparison of the fused 3D point clouds. (a) Calibrated inference yields well-aligned pedestrian pillars. (b) Uncalibrated inference drifts; the red circle marks cam4 (cyan) splitting into offset ghost streaks instead of one shared pillar.	25
4.5	Ablation of architectural variants on the Wildtrack dataset. Dilated convolutions with ResNet-64 feature lifting achieve the highest performance (68.7% MODA), whereas combining soft depth with dilated convolutions or adding lightweight compensation degrades accuracy.	27
4.6	Training and validation loss convergence dynamics across 40 epochs on Wildtrack. The left panel shows training loss; the right panel shows validation loss, which settles for most variants after the learning rate step-downs at epochs 25 and 35.	29
4.7	Qualitative BEV diagnostic heatmaps on Wildtrack evaluation frames under the best recipe (Dilated + ResNet-64). Ground truth positions are marked as red stars, predicted detections as cyan circles, true-positive associations as green lines, missed detections (FN) as red crosses, and spurious false alarms (FP) as blue crosses. The dashed white contour denotes the labeled evaluation ground-plane ROI.	32

4.8	Recall $\times$ Precision Pareto scatter plot across threshold and NMS sweeps on Wildtrack. Dotted contours depict theoretical MODA iso-lines. The Pareto frontier combines operating points from five variants; no single variant dominates.	33
A.1	Overview of the MVUDA framework [2]. (a) Training: A multi-view detector is trained on the source domain with ground-truth labels ($\mathcal{L}^S$) and adapted to the target domain using pseudo-labels ($\mathcal{L}^T$). (b) Results: Comparison of baseline vs. MVUDA performance across multiple cross-domain settings, measured by MODA.	II
A.2	BEV feature visualizations from the MVFP detector on the Wildtrack dataset. (a) Concatenated multi-view features projected onto the ground plane via 3D Feature Pulling. (b) Predicted occupancy map. (c) Camera coverage indicator map.	V

List of Tables

4.1	Geometric fidelity and downstream detection performance of MapAnything under Calibrated versus Uncalibrated inference modes on Wildtrack. All three rows share an identical fused-point-cloud, RGB-feature detection architecture. The middle row isolates the per-camera ground-plane median scale alignment (Section 3.2.3) from the effect of the calibration input to MapAnything itself: it uses the same calibrated MapAnything geometry as the bottom row, but corrected by only a single global Sim3 transform fit to the camera centers, with no per-camera correction.	24
4.2	Full-camera detection performance of the proposed foundation-model pipeline (DilatedBEVBackbone + FusedLSSLift ResNet-64) compared to DCHM. The DCHM row is <i>our own reproduction</i> of the supervised DCHM pipeline with a single round of Gaussian optimization and depth fine-tuning; the results reported in the original paper are given in the text.	26
4.3	Wildtrack architectural ablation ledger.	27
4.4	Cross-subset evaluation matrix on Wildtrack (4-camera subsets, Dilated + ResNet-64 configuration). Metrics: MODA / MODP / Recall / Precision.	30
4.5	Cross-subset evaluation matrix on MultiviewX (disjoint 3-camera subsets, Dilated + ResNet-64 configuration). Metrics: MODA / MODP / Recall / Precision.	31
A.1	MVFP cross-domain detection results (Type 1+2). Standard model results are reported, with EMA model results in parentheses where they outperform the standard model. All models are trained for 50 epochs on an NVIDIA DGX Spark.	III
A.2	Type 1+2 domain gap quantification (MVFP, BEV-space MODA).	III
A.3	Index of model checkpoints, evaluation logs, and execution parameters.	VI

1

Introduction

Multi-view pedestrian detection locates and tracks people on a shared ground plane by combining observations from several overlapping cameras. In crowded surveillance settings such as transit hubs and industrial sites, occlusion often degrades single-camera detectors, since a person hidden in one view may be visible from another. Fusing information across sparse, wide-baseline cameras, however, remains difficult.

1.1 Background and Motivation

Modern multi-view pedestrian detection systems usually follow a two-stage design: *transforming image features into a 3D or bird’s-eye view (BEV) space*, followed by *localizing pedestrians in BEV*. Early methods [6, 23] projected image features onto the ground plane using planar homography. This is computationally cheap, but a planar homography maps the whole vertical body of a pedestrian onto the ground plane, which introduces perspective stretching and shadow-like distortions.

To move beyond the planar assumption, later work adopted volumetric and point-based 3D modeling. Cardboard-based modeling (MvCHM) [18] and 3D voxelization frameworks [17] introduced upright geometry to constrain multi-view feature projection. More recently, Depth-Consistent Human Modeling (DCHM) [16] used 3D Gaussian Splatting (3D-GS) [8] to optimize scene geometry in a self-supervised way, and then fine-tuned monocular depth models over several iterations to produce dense, cross-view consistent point clouds.

Despite their localization accuracy, these geometry-centric approaches have practical constraints. DCHM, for example, requires intensive per-scene 3D Gaussian Splatting and iterative depth fine-tuning, which introduces latency and prevents real-time or zero-shot deployment. Their reliance on discrete point cloud clustering (e.g., DBSCAN) also rules out end-to-end training with downstream detection objectives, and their dependence on fixed camera calibrations limits robustness when the camera configuration changes.

At the same time, 3D vision foundation models such as DUST3R [26], MAST3R [11], and MapAnything [19] estimate metric 3D point maps and camera poses in a single feedforward pass. Because they remove iterative per-scene fitting, they offer a scalable alternative to specialized geometric pipelines, which motivates evaluating general-purpose foundation backbones for multi-view pedestrian localization.

1.2 Problem Statement and Research Questions

This thesis studies whether off-the-shelf 3D vision foundation models can serve as feedforward geometric backbones for multi-view pedestrian detection. Foundation models can generate coherent 3D scene structures, but their metric precision, cross-view consistency, and compatibility with downstream bird’s-eye view (BEV) detectors in wide-baseline surveillance settings have not been well characterized.

We focus on three questions. First, we examine the metric geometric fidelity and error behavior of foundation models in surveillance scenes, distinguishing calibrated inference with known camera poses from uncalibrated image-only estimation. Second, we consider how downstream BEV detectors can be adapted to absorb the spatial dispersion (30–80 cm) of zero-shot point clouds, and what detection performance is achievable relative to scene-optimized baselines. Third, we study camera-rig generalization, to determine whether BEV detectors trained on foundation geometry are calibration invariant across camera subsets or remain tied to the physical layout of the training array.

1.3 Summary of Contributions

This work provides an empirical evaluation and an architectural baseline for foundation-model 3D geometry in multi-view pedestrian detection.

First, we build an end-to-end differentiable detection pipeline that integrates the MapAnything foundation model with a modified BEV detector, and that avoids per-scene Gaussian Splatting and depth optimization. To handle the cross-view point dispersion (30–80 cm) of zero-shot geometry, we introduce *FusedLSSLift*, which projects 3D points into 2D ResNet-18 feature spaces, and adapt dilated convolutions (*DilatedBEVBackbone*) to enlarge the receptive field of the BEV backbone without adding parameters. On standard benchmarks this system reaches 68.7 MODA on Wildtrack and 64.0 MODA on MultiviewX, which provides a reference point for off-the-shelf foundation geometry without per-scene optimization; it remains well below the best reported per-scene optimized result on Wildtrack (roughly 95 MODA for supervised DCHM).

Second, we characterize the geometric error regimes and the downstream architectural trade-offs. Calibrated inference shows a per-camera scale drift with a 5.5 cm camera-center residual that ground-plane median scaling corrects, whereas uncalibrated inference shows frame-to-frame pose jitter averaging 66.0 cm (about 7 BEV cells) that degrades localization. Our ablations show two interaction effects: pairing soft-depth sampling with dilated convolutions causes spatial over-smoothing that reduces recall by 4.9 percentage points, and a lightweight spatial compensation module introduces ground-level noise that reduces precision. Cross-subset camera experiments show large drops when camera coverage is reduced at test time (−28.35 MODA on Wildtrack and −18.05 MODA on MultiviewX), which indicates that current BEV detectors are coupled to the camera rig used during training.

1.4 Thesis Outline

The remainder of this thesis is structured as follows:

- **Chapter 2 (Theory and Related Work)** introduces multi-view pedestrian detection, 3D Gaussian Splatting, 3D vision foundation models, BEV feature transformation paradigms (LSS, PointPillars), and cross-camera domain gaps.
- **Chapter 3 (Methods)** formalizes the proposed pipeline, detailing foundation-model geometry inference, Sim3 error modeling, ground-plane median scale alignment, FusedLSSLift feature projection, and the DilatedBEVBackbone architecture.
- **Chapter 4 (Results)** reports the experimental evaluation, comparing foundation-model geometry against dedicated baselines, analyzing architectural ablations and loss convergence, and examining camera-rig dependency.
- **Chapter 5 (Discussion)** discusses the findings, examines the geometric and architectural mechanisms and the camera-rig dependency, and outlines deployment limitations and future work.
- **Appendix** provides domain gap quantification extending the MVUDA framework, mathematical derivations for Sim3 alignment, and hyperparameter details.

2

Theory and Related Work

This chapter reviews the background and prior work relevant to this thesis: multi-view pedestrian detection, the formulation of 3D Gaussian Splatting and its use in scene modeling, feedforward 3D vision foundation models, feature transformation into bird’s-eye view (BEV) space, and domain shift across camera arrays.

2.1 Multi-View Pedestrian Detection

Multi-view pedestrian detection aggregates observations across multiple calibrated cameras to estimate pedestrian locations on a shared ground plane. Observing a space from several viewpoints gives multi-camera networks spatial redundancy, so a person occluded in one view can often be recovered from another. Existing methods fall into supervised and label-free paradigms, depending on whether they use ground-truth 3D spatial annotations.

2.1.1 Supervised Feature Transformation Paradigms

Supervised architectures train neural networks directly on annotated 3D standing locations, typically represented as ground-plane coordinates. The primary design challenge in these systems is lifting 2D perspective image features into a unified 3D or bird’s-eye view (BEV) world frame.

Early deep architectures such as MVDet [6] relied on planar homography transformations to warp 2D convolutional feature maps directly onto the ground plane ($Z = 0$). While computationally efficient, planar homographies assume that all visual content resides on the ground plane. Because human bodies extend vertically along the gravity vector, upright pedestrians violate this planar assumption: features corresponding to heads and torsos are projected outward along radial camera sightlines, which produces perspective stretching and elongated shadow artifacts.

To mitigate perspective distortion, subsequent research moved toward multi-level geometric representations. Song et al. [23] sampled homographies across multiple parallel horizontal planes at discrete vertical heights, aggregating feature slices across heights to capture multi-level bodily features. Hou and Zheng [5] introduced MVDeTr, employing deformable shadow transformers to explicitly model cross-view occlusion dynamics and suppress perspective displacement. Moving beyond multi-plane approximations, Ma et al. [17] proposed voxelized 3D feature aggregation, constructing an explicit 3D voxel lattice before vertically compressing feature representations into BEV space. This was further refined in MvCHM [18], which intro-

duced a cardboard human representation by projecting detected 2D bounding boxes into upright vertical planes, sampling them as synthetic point clouds, and processing them with PointPillars [10] to regress ground coordinates.

Although supervised multi-view detectors achieve high localization accuracy on standard benchmarks, they require dense 3D point annotations for each deployment venue and transfer poorly across scenes when the camera topology changes.

2.1.2 Label-Free and Self-Supervised Paradigms

To bypass the labor-intensive requirement of 3D spatial labels, several label-free and weakly supervised frameworks have emerged. Early attempts relied on heuristic geometric clustering [12, 15, 29], where monocular 2D keypoints or bounding box base centers are back-projected as optical rays onto the ground plane, followed by clique-partitioning or spatial clustering to localize individuals. However, these methods are sensitive to 2D detector errors, ground unevenness, and calibration noise.

More recently, neural implicit representations and differentiable rendering have been explored to infer 3D human presence without explicit labels. For example, UMPD [14] estimated continuous 3D pedestrian occupancy grids through neural volume rendering, using spatial pseudo-labels generated from self-supervised single-view segmentation masks. While eliminating ground-truth annotations, volume-rendering approaches incur significant computational costs during coordinate query and volume integration.

2.2 3D Gaussian Splatting and Explicit Scene Modeling

A central prerequisite for multi-view pedestrian localization is obtaining accurate 3D spatial representations of human bodies from 2D images. While neural radiance fields (NeRF) parameterize scenes implicitly via coordinate-based multi-layer perceptrons, their volumetric ray-marching requires dense point sampling along every optical ray, leading to high training and rendering latencies. To address these computational limitations, Kerbl et al. [8] introduced 3D Gaussian Splatting (3D-GS), which models 3D environments using explicit, differentiable anisotropic Gaussian primitives.

2.2.1 Mathematical Parameterization of 3D Gaussians

In 3D-GS, a scene is represented as an unstructured collection of 3D Gaussians. Each individual primitive g_i is parameterized by its mean position $\boldsymbol{\mu}_i \in \mathbb{R}^3$, an opacity scalar $\alpha_i \in [0, 1]$, view-dependent color coefficients parameterized by spherical harmonics, and a 3D covariance matrix $\boldsymbol{\Sigma}_i \in \mathbb{R}^{3 \times 3}$. The spatial influence of a Gaussian on an arbitrary 3D coordinate $\mathbf{x} \in \mathbb{R}^3$ is defined as:

$$G_i(\mathbf{x}) = \exp \left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i) \right). \quad (2.1)$$

To ensure that $\boldsymbol{\Sigma}_i$ remains positive semi-definite throughout gradient-based optimization, the covariance is factorized into a scale vector $\mathbf{s}_i = (s_x, s_y, s_z) \in \mathbb{R}^3$ and a

unit quaternion $\mathbf{q}_i \in \mathbb{H}$ representing a rotation matrix $\mathbf{R}_i \in SO(3)$:

$$\boldsymbol{\Sigma}_i = \mathbf{R}_i \mathbf{S}_i \mathbf{S}_i^T \mathbf{R}_i^T, \quad (2.2)$$

where $\mathbf{S}_i = \text{diag}(\mathbf{s}_i)$ is a diagonal scaling matrix. This explicit formulation allows Gaussians to adaptively stretch and rotate along surfaces, modeling geometric boundaries and thin structures more efficiently than isotropic representations.

2.2.2 Differentiable Splatting and Depth Rendering

To render an image from a camera with viewing transformation $\mathbf{W} \in \mathbb{R}^{3 \times 3}$, 3D Gaussians are projected onto the 2D image plane. Following the projective approximation of elliptical splatting, the 2D screen-space covariance matrix $\boldsymbol{\Sigma}'_i$ is computed as:

$$\boldsymbol{\Sigma}'_i = \mathbf{J}_i \mathbf{W} \boldsymbol{\Sigma}_i \mathbf{W}^T \mathbf{J}_i^T, \quad (2.3)$$

where $\mathbf{J}_i$ denotes the Jacobian of the affine approximation of the projective transformation evaluated at the Gaussian center $\boldsymbol{\mu}_i$.

Color rendering is performed via tile-based rasterization. Gaussians intersecting each image tile are sorted by depth from the camera center. The color $C(\mathbf{p})$ at pixel $\mathbf{p} = (u, v)$ is accumulated using sorted alpha blending:

$$C(\mathbf{p}) = \sum_{i \in \mathcal{N}} c_i \alpha'_i(\mathbf{p}) \prod_{j=1}^{i-1} (1 - \alpha'_j(\mathbf{p})), \quad (2.4)$$

where c_i is the view-dependent color evaluated via spherical harmonics, and $\alpha'_i(\mathbf{p}) = \alpha_i \exp\left(-\frac{1}{2}(\mathbf{p} - \boldsymbol{\mu}'_i)^T (\boldsymbol{\Sigma}'_i)^{-1} (\mathbf{p} - \boldsymbol{\mu}'_i)\right)$ represents the screen-space opacity at pixel $\mathbf{p}$.

For geometric estimation, depth can be synthesized through the same optical transmittance formulation. By replacing the RGB color c_i with the Euclidean or orthogonal distance z_i of the Gaussian center along the camera optical axis, the rendered depth $D(\mathbf{p})$ is formulated as:

$$D(\mathbf{p}) = \sum_{i \in \mathcal{N}} z_i \alpha'_i(\mathbf{p}) \prod_{j=1}^{i-1} (1 - \alpha'_j(\mathbf{p})). \quad (2.5)$$

This formulation yields dense, geometrically meaningful depth maps directly aligned with the input camera viewpoints.

2.2.3 3D-GS as a Consistency Prior in Multi-View Surveillance

Because 3D-GS enforces multi-view photometric consistency, it is well suited to synthesizing 3D pseudo-labels in surveillance environments. In Depth-Consistent Human Modeling (DCHM) [16], 3D-GS was adopted as an intermediate geometric regularizer. By restricting Gaussian optimization to foreground pedestrian masks segmented across synchronized multi-camera views, DCHM reconstructed an explicit proxy of the pedestrian geometry. Depth maps rendered from these optimized Gaussians provided multi-view consistent pseudo-ground-truth, which was subsequently

employed to fine-tune monocular depth networks (such as Depth Anything v2 [28]) across multiple iterative training cycles.

Using 3D-GS in this way, however, introduces an operational trade-off. Because 3D-GS relies on per-scene stochastic gradient descent (often requiring thousands of iterations per capture setup), it functions as an offline optimization process rather than a feedforward inference mechanism. The computational overhead and the lack of immediate generalization to new camera configurations motivate the use of general-purpose 3D foundation models, which estimate dense metric geometry in a single forward pass without scene-specific retraining.

2.3 3D Vision Foundation Models

Traditional multi-view 3D reconstruction pipelines depend on Structure from Motion (SfM) [22] and Multi-View Stereo (MVS). In surveillance installations, however, wide camera baselines, large perspective differences, occlusion, and uniform surface textures often cause classical local feature matchers (such as SIFT or ORB) to fail. Large-scale visual pre-training has produced foundation models that address these limitations through learned geometric priors.

2.3.1 From Monocular Priors to Multi-View Foundation Architectures

Two complementary approaches have emerged in foundation 3D estimation. The first family comprises monocular metric depth estimators, including Metric3D v2 [7], Depth Pro [1], and Depth Anything v2 [28]. Trained on diverse indoor and outdoor imagery, these networks predict high-resolution metric depth maps from individual images with sharp object boundaries. However, because each viewpoint is processed independently without cross-camera communication, monocular back-projections exhibit systematic scale drift and depth misalignment across cameras, resulting in duplicated or fragmented pedestrian point clouds in world space.

To resolve cross-view inconsistencies, the second family develops multi-view unconstrained foundation models, exemplified by DUS3R [26] and MAS3R [11]. These architectures treat multi-view reconstruction as an uncalibrated point-map regression task. Using Vision Transformers (ViT) with cross-attention across image pairs, they simultaneously resolve visual correspondences and predict dense 3D point maps in a single feedforward pass, eliminating the need for iterative bundle adjustment or pre-calibrated camera parameters.

2.3.2 The MapAnything Framework

MapAnything [19] extends feedforward multi-view geometric estimation to variable-sized collections of cameras. Building on a large ViT backbone with alternating self-attention and cross-view attention layers, MapAnything jointly predicts dense metric 3D point maps $\mathbf{P}_w \in \mathbb{R}^{H \times W \times 3}$ and associated confidence weights for all input viewpoints.

The architecture operates under two distinct configuration modes: In the *calibrated mode*, the model receives RGB images alongside known camera intrinsic

matrices $\mathbf{K}_c$ and extrinsic transformations $(\mathbf{R}_c, \mathbf{t}_c)$. The network conditions its predictions on these geometric constraints while inferring absolute metric scale (`is_metric_scale = True`). In the *uncalibrated mode*, MapAnything processes raw RGB frames without calibration metadata, simultaneously predicting relative camera poses, focal lengths, and 3D point coordinates in an internally estimated coordinate system.

This thesis examines how reliable these two operating regimes are, and how well they support downstream detection, in wide-baseline surveillance.

2.4 Multi-View Feature Transformation and BEV Representation

To facilitate spatial object localization, multi-camera visual features must be mapped from perspective image planes into a shared, metric bird’s-eye view (BEV) coordinate frame.

2.4.1 The Lift-Splat-Shoot (LSS) Paradigm

Originally formulated by Philion and Fidler [20] for autonomous driving perception, the Lift-Splat-Shoot (LSS) paradigm establishes an end-to-end differentiable bridge between 2D images and BEV representations. For each camera, an image feature extractor outputs a 2D feature map $\mathbf{f}_{u,v} \in \mathbb{R}^C$ alongside a categorical depth distribution $\mathbf{p}_d \in \mathbb{R}^D$ defined over discrete depth bins along the camera ray. The "Lift" step constructs an outer-product feature frustum:

$$\mathbf{c}_{u,v,d} = \mathbf{f}_{u,v} \otimes \mathbf{p}_d, \quad (2.6)$$

generating a 3D distribution of features along optical sightlines. In the "Splat" phase, all frustum features across cameras are projected into predetermined 3D voxel bins based on known camera geometry. A vertical pooling operation then collapses features along the vertical axis, producing a unified 2D BEV feature grid amenable to standard convolutional processing.

2.4.2 PointPillars Representation

While voxel-based 3D convolutions preserve vertical structure, they incur substantial memory and computational overhead. Lang et al. [10] proposed PointPillars to balance 3D spatial precision with computational efficiency.

The PointPillars formulation discretizes the horizontal ground plane into an infinite-height regular pillar grid. Points residing within each pillar are enriched by appending geometric offsets relative to the pillar’s arithmetic mean and spatial center. A simplified PointNet [21] layer (PillarVFE) aggregates point features within each pillar via linear transformations and max-pooling, producing a single dense feature vector per column. These pillar representations are then scattered back to their corresponding 2D grid coordinates (PointPillarScatter), synthesizing a pseudo-image that is efficiently processed by standard 2D convolutional backbones.

In our detection framework, PointPillars is utilized to process metric point clouds generated by foundation models, providing an efficient, fully differentiable pathway from raw 3D point geometry to BEV localization heatmaps.

2.5 Multi-View Domain Shifts and Camera-Rig Invariance

Deploying multi-view detection models to operational surveillance environments typically results in noticeable performance drops if the deployment setup diverges from the training domain. To formalize this degradation, Brorsson et al. [2] introduced the Multi-View Unsupervised Domain Adaptation (MVUDA) framework, which decomposes cross-domain performance degradation into two orthogonal components.

The first component is the *Type 1 Domain Gap (Image Appearance Shift)*, arising from differences in ambient illumination, weather conditions, sensor color profiles, and pedestrian clothing distributions. This shift operates primarily within the 2D image feature space. The second component is the *Type 2 Domain Gap (BEV Camera Layout Shift)*, caused by physical differences in camera mounting heights, viewing angles, baseline distances, and mutual visual overlap. Because perspective projection maps spatial features into the ground plane according to extrinsic geometry, alterations in camera layout warp the density and distribution of projected features across the BEV plane.

While prior literature often presumes that explicit geometric projections provide intrinsic invariance to camera placement, empirical evidence regarding whether learned BEV networks can maintain performance across changing camera topologies has been limited. Later in this thesis, we build on this theoretical decomposition to evaluate camera-rig dependency and cross-camera generalization.

3

Methods

This chapter formalizes the proposed methodology for metric multi-view pedestrian detection using 3D vision foundation models. We first provide an architectural overview of the feedforward pipeline, detail the geometric characterization and ground-plane scale alignment for MapAnything, introduce the FusedLSSLift feature projection module, and formulate the DilatedBEVBackbone designed to absorb foundation-model geometric noise.

3.1 System Architecture Overview

Figure 3.1 outlines the overall architecture of our multi-view pedestrian detection system. The pipeline operates entirely in feedforward inference:

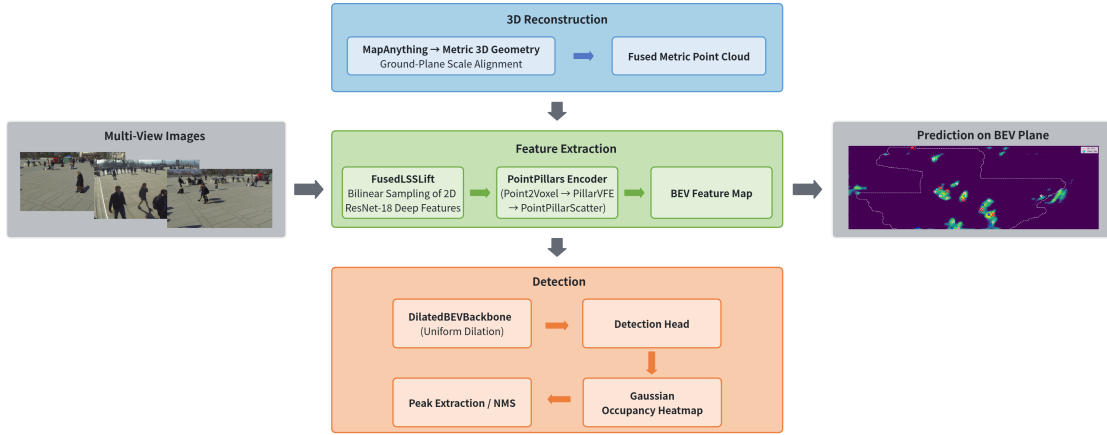


Figure 3.1: Schematic overview of the proposed end-to-end foundation-model multi-view pedestrian detection pipeline.

Given synchronized video frames $\{I_c\}_{c=1}^C$ from C calibrated surveillance cameras, the system executes three sequential processing stages:

1. **Geometric Foundation:** MapAnything generates metric 3D point clouds for all viewpoints in a single pass. A closed-form ground-plane median scaling aligns per-camera depths into an integrated coordinate frame.
2. **Point-to-BEV Lifting and Deep Feature Sampling:** Rather than using raw RGB colors, the *FusedLSSLift* module projects 3D points back onto 2D image planes to sample deep convolutional features from a shared ResNet-18

backbone. The enriched point clouds are discretized into vertical pillars and scattered onto the BEV grid.

3. **Dilated Spatial Convolution and Heatmap Head:** The *DilatedBEVBackbone* processes the BEV feature map using dilated convolutions ($d = 2$), expanding the spatial receptive field to aggregate dispersed pedestrian point clouds into localized occupancy peaks.

3.2 Geometric Foundation: MapAnything Inference and Error Characterization

3.2.1 Calibrated vs. Uncalibrated Operational Modes

MapAnything [19] employs an alternating cross-attention Vision Transformer to infer dense metric 3D point maps. We define the camera intrinsics as $\mathbf{K}_c \in \mathbb{R}^{3 \times 3}$ and extrinsics as world-to-camera rotation $\mathbf{R}_c \in \text{SO}(3)$ and translation $\mathbf{t}_c \in \mathbb{R}^3$. The camera optical center in world coordinates is $\mathbf{C}_c = -\mathbf{R}_c^T \mathbf{t}_c$.

We investigate two operational regimes:

- **Calibrated Mode:** The model receives $\{I_c, \mathbf{K}_c, \mathbf{R}_c, \mathbf{t}_c\}_{c=1}^C$ alongside the instruction flag `is_metric_scale = True`. The network leverages known extrinsic baselines as geometric constraints while predicting dense 3D point positions $\mathbf{P}_w \in \mathbb{R}^3$.
- **Uncalibrated Mode:** The model ingests solely raw RGB images $\{I_c\}_{c=1}^C$. The network internally predicts camera poses, focal lengths, and 3D point maps up to an arbitrary coordinate frame.

3.2.2 Sim3 Alignment and Error Decomposition

To quantitatively assess the coordinate fidelity of MapAnything’s output relative to physical ground-truth surveillance coordinates, we compute the optimal 7-degree-of-freedom similarity transformation $\mathbf{S} \in \text{Sim}(3)$ aligning estimated camera centers $\{\mathbf{C}_{est}^c\}_{c=1}^C$ with ground-truth camera centers $\{\mathbf{C}_{gt}^c\}_{c=1}^C$ via the Umeyama algorithm [25]:

$$\min_{s, \mathbf{R}_{sim3}, \mathbf{t}_{sim3}} \frac{1}{C} \sum_{c=1}^C \left\| \mathbf{C}_{gt}^c - (s \mathbf{R}_{sim3} \mathbf{C}_{est}^c + \mathbf{t}_{sim3}) \right\|_2^2, \quad (3.1)$$

subject to $s > 0$ and $\mathbf{R}_{sim3}^T \mathbf{R}_{sim3} = \mathbf{I}, \det(\mathbf{R}_{sim3}) = 1$. The mean camera-center alignment residual is defined as:

$$e_{sim3} = \frac{1}{C} \sum_{c=1}^C \left\| \mathbf{C}_{gt}^c - (s \mathbf{R}_{sim3} \mathbf{C}_{est}^c + \mathbf{t}_{sim3}) \right\|_2. \quad (3.2)$$

We observe a difference in error characteristics between the two modes:

Low-Rank Error in Calibrated Mode. In calibrated mode, MapAnything conditions on the ground-truth camera positions and yields a low camera-center residual

of $e_{sim3} = 5.5$ cm (with optimal scale ratio $s = 0.982$). However, because the foundation model balances the input geometric prompts against its internal monocular visual priors, each camera’s predicted depth carries a small relative scale drift of 5% to 10%.

Mathematically, this geometric error is *low-rank*: for a setup of C cameras, the discrepancy across millions of 3D points can be parameterized by C independent scalar multipliers $\{\alpha_c\}_{c=1}^C$:

$$D_{true}^c(\mathbf{u}) \approx \alpha_c D_{pred}^c(\mathbf{u}) + \epsilon_c(\mathbf{u}), \quad \|\epsilon_c\| \ll \alpha_c D_{pred}^c. \quad (3.3)$$

Because the error manifold is low-rank, it can be corrected using a single reference plane.

High-Rank Error in Uncalibrated Mode. In uncalibrated mode, the reconstructed point cloud looks visually cohesive. However, aligning the estimated camera array to physical world coordinates via Umeyama yields an average residual of $e_{sim3} = 66.0$ cm (about 7 cells of the 10 cm BEV grid). This residual is a *high-rank*, frame-varying distortion:

$$\mathbf{C}_{gt}^{c,t} = s^t \mathbf{R}^t \mathbf{C}_{est}^{c,t} + \mathbf{t}^t + \delta_c^t, \quad (3.4)$$

where δ_c^t represents a non-rigid, per-frame displacement that cannot be absorbed by any global rigid or similarity transformation. When uncalibrated point clouds are mapped to the ground plane, pedestrian positions fluctuate by tens of centimeters across frames, which degrades downstream localization.

3.2.3 Closed-Form Ground-Plane Median Scale Alignment

To resolve the scale drift in calibrated mode, we apply a ground-plane scale alignment based on ray-plane intersection geometry.

Let the ground plane in world coordinates be defined by the plane $Z = 0$. For any camera c with intrinsics $\mathbf{K}_c$ and extrinsics $(\mathbf{R}_c, \mathbf{t}_c)$, the analytical metric distance from the camera center to the physical ground along the ray passing through pixel $\mathbf{u} = (u, v)^T$ is given by geometric ray-plane intersection:

$$D_{ground}^c(u, v) = \frac{-[\mathbf{t}_c]_z}{[\mathbf{R}_c \mathbf{K}_c^{-1} \tilde{\mathbf{u}}]_z} \cdot \|\mathbf{R}_c \mathbf{K}_c^{-1} \tilde{\mathbf{u}}\|_2, \quad (3.5)$$

where $\tilde{\mathbf{u}} = (u, v, 1)^T$ denotes homogeneous pixel coordinates and $[\cdot]_z$ extracts the vertical Z -component.

Using Segment Anything (SAM) [9], we obtain a binary ground semantic mask $M_{ground}^c \in \{0, 1\}^{H \times W}$. We calculate the closed-form, outlier-robust scale factor α_c as the median ratio over all valid ground pixels:

$$\alpha_c = \text{median} \left(\left\{ \frac{D_{ground}^c(\mathbf{u})}{D_{pred}^c(\mathbf{u})} \mid \mathbf{u} \in \Omega, M_{ground}^c(\mathbf{u}) = 1, D_{pred}^c(\mathbf{u}) > 0 \right\} \right). \quad (3.6)$$

α_c corrects the depth measured along camera c ’s viewing rays. The scaling must therefore be applied about the camera center $\mathbf{C}_c$, not directly to point coordinates

expressed relative to the (arbitrary) world origin. Let $\mathbf{P}_{cam}^c = \mathbf{R}_c \mathbf{P}_w^c + \mathbf{t}_c$ denote a predicted point $\mathbf{P}_w^c = (X, Y, Z)^T$ in camera- c coordinates. We rescale this camera-frame vector directly by α_c :

$$\hat{\mathbf{P}}_{cam}^c = \alpha_c \cdot \mathbf{P}_{cam}^c. \quad (3.7)$$

This uniformly stretches the point along the ray from $\mathbf{C}_c$. We then map the result back into world coordinates via the inverse extrinsic transform:

$$\hat{\mathbf{P}}_w^c = \mathbf{R}_c^T \hat{\mathbf{P}}_{cam}^c - \mathbf{R}_c^T \mathbf{t}_c = \mathbf{C}_c + \alpha_c (\mathbf{P}_w^c - \mathbf{C}_c). \quad (3.8)$$

The right-hand equality shows this is equivalent to scaling the point’s offset from the camera center $\mathbf{C}_c$, not its offset from the world origin. The two coincide only when $\mathbf{C}_c = \mathbf{0}$, which is not generally true for a multi-camera rig. The scaled point clouds from all cameras are then merged into a unified metric point cloud:

$$\mathbf{P}_{fused} = \bigcup_{c=1}^C \hat{\mathbf{P}}_w^c. \quad (3.9)$$

3.3 End-to-End BEV Detection Pipeline

3.3.1 Feature Representation: Fused Point Cloud vs. Per-Camera Frustum

In standard LSS architectures [20], 2D features are lifted into independent frustums per camera before spatial splatting. However, in our setting, training separate lifting distributions per camera fails because the small training sample size in surveillance datasets cannot supervise the high-dimensional cross-view fusion space.

Instead, we adopt the *fused point cloud representation*: MapAnything’s cross-view attention mechanism natively performs metric fusion. The downstream network ingests the fused metric point cloud $\mathbf{P}_{fused}$ directly, focusing its learning capacity on BEV spatial discrimination.

3.3.2 Deep Feature Projection via FusedLSSLift

Using raw RGB point colors alone limits semantic discriminability. To supply more informative features, we adapt a back-projection sampling technique (*FusedLSSLift*).

We pass each multi-view image I_c through an ImageNet-pretrained ResNet-18 backbone [4], extracting a 1/4-scale feature map $\mathbf{F}_c \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times C_{feat}}$ with $C_{feat} = 64$. For each 3D point $\mathbf{P} \in \mathbf{P}_{fused}$, we project its coordinates onto the image plane of every camera:

$$\mathbf{u}_c = \pi_c(\mathbf{P}) = \mathbf{K}_c(\mathbf{R}_c \mathbf{P} + \mathbf{t}_c). \quad (3.10)$$

Only cameras that could plausibly have observed $\mathbf{P}$ should contribute a feature sample. We therefore restrict sampling to the visibility set $\mathcal{V}_{vis}(\mathbf{P})$, defined as the camera indices for which $\mathbf{P}$ lies in front of the camera and its projection $\mathbf{u}_c$ falls within the image boundaries. This frustum-based test does not model self-occlusion by intervening surfaces—for example, another pedestrian standing between $\mathbf{P}$ and camera c . We also implemented a stricter per-pixel z-buffer variant. It additionally

requires $\mathbf{P}$'s depth to agree with camera c 's own reconstructed surface depth at $\mathbf{u}_c$, within a tolerance τ_{occ} . This rejects points that project onto a pixel actually occupied by a closer surface. We tested this stricter occlusion filtering and found it did not improve detection accuracy. We therefore report results with the frustum-only test ($\tau_{occ} = 0$). We sample the feature vector via bilinear interpolation:

$$\mathbf{f}_c(\mathbf{P}) = \text{BilinearInterpolate}(\mathbf{F}_c, \mathbf{u}_c/4). \quad (3.11)$$

The point feature is formed by averaging across all visible viewpoints:

$$\mathbf{f}(\mathbf{P}) = \frac{1}{|\mathcal{V}_{vis}(\mathbf{P})|} \sum_{c \in \mathcal{V}_{vis}(\mathbf{P})} \mathbf{f}_c(\mathbf{P}). \quad (3.12)$$

Each point is thus represented by an augmented vector $\tilde{\mathbf{P}} = (X, Y, Z, \mathbf{f}(\mathbf{P})) \in \mathbb{R}^{3+C_{feat}}$.

3.3.3 PointPillars Discretization and Scattering

Following PointPillars [10], the ground plane is discretized into an orthogonal 2D grid with spatial resolution $\Delta x = \Delta y = 10$ cm over the ground range $[X_{min}, X_{max}] \times [Y_{min}, Y_{max}]$. Each cell groups 4×4 cells of the 2.5 cm dataset annotation grid, giving 120×360 cells for the 12×36 m Wildtrack area and 160×250 cells for the 16×25 m MultiviewX area. All points falling within the vertical prism of each grid cell form a pillar.

Inside each pillar p , points are augmented with their offsets relative to the pillar center $(\bar{x}_p, \bar{y}_p, \bar{z}_p)$. A simplified PointNet (PillarVFE) maps the augmented point features through linear layers, batch normalization, and max-pooling to yield a single C_{pillar} -dimensional embedding per non-empty pillar. The PointPillarScatter layer scatters these vectors onto a dense 2D pseudo-image tensor $\mathbf{F}_{bev} \in \mathbb{R}^{H_{bev} \times W_{bev} \times C_{pillar}}$, where $H_{bev} = (Y_{max} - Y_{min})/\Delta y$ and $W_{bev} = (X_{max} - X_{min})/\Delta x$.

3.4 Architectural Adaptations for Foundation Geometry Noise

3.4.1 Cross-View Point Dispersion and DilatedBEVBackbone

Even after ground-plane scale alignment, points corresponding to the same physical person reconstructed from disparate cameras exhibit a residual spatial dispersion of 30 to 80 cm. On the 10 cm BEV grid, this dispersion spans a cluster diameter of 3 to 8 cells.

The standard BaseBEVBackbone processes the BEV feature map with 3×3 convolutions of dilation $d = 1$ at three spatial scales (strides 2, 4, and 8). To consolidate each pedestrian's dispersed points into a single occupancy peak, the network has to aggregate evidence over a neighborhood of several cells, so we enlarge the spatial support of its convolutions.

To address this without increasing parameter count, we adapt standard dilated convolutions to form *DilatedBEVBackbone*:

- We configure the inner convolutional layers of each block with uniform dilation rate $d = 2$ and matching padding $p = 2$.
- A 3×3 kernel with dilation $d = 2$ samples a 5×5 neighborhood, so each inner layer doubles its spatial extent without additional parameters, enlarging the receptive field at every scale of the backbone. This helps the network aggregate scattered point clouds into unified central activations.
- **Hybrid Dilated Convolutions (HDC)**: To prevent potential gridding artifacts where dilated kernels skip intermediate pixels, we also evaluate an HDC pattern [27] with layer-wise dilation rates $[1, 2, 4, 2, 1]$.

3.4.2 Loss Function and Heatmap Decoding

The BEV feature map from DilatedBEVBackbone is fed into a convolutional detection head that outputs a continuous occupancy probability heatmap $\hat{\mathbf{Y}} \in [0, 1]^{H_{bev} \times W_{bev}}$.

Training Objective. Ground-truth pedestrian positions $\{(x_i, y_i)\}_{i=1}^N$ are rendered onto the BEV grid by summing 2D Gaussian kernels with standard deviation $\sigma = 2$ cells (20 cm), each truncated to a 17×17 cell window, and clamping the result to $[0, 1]$:

$$Y_{xy} = \min \left(1, \sum_{i=1}^N \exp \left(-\frac{(x - x_i)^2 + (y - y_i)^2}{2\sigma^2} \right) \right). \quad (3.13)$$

Since every kernel peaks at 1, each annotated pedestrian cell attains exactly $Y_{xy} = 1$, which defines the positive set of the focal loss below. We train the network using a modified Focal Loss [13]:

$$\mathcal{L}_{focal} = -\frac{1}{N} \sum_{x=1}^{W_{bev}} \sum_{y=1}^{H_{bev}} \begin{cases} (1 - \hat{Y}_{xy})^\gamma \log(\hat{Y}_{xy}) & \text{if } Y_{xy} = 1, \\ (1 - Y_{xy})^\beta (\hat{Y}_{xy})^\gamma \log(1 - \hat{Y}_{xy}) & \text{if } Y_{xy} < 1, \end{cases} \quad (3.14)$$

with focusing parameter $\gamma = 2$ and penalty reduction parameter $\beta = 4$.

Inference and Peak Decoding. During inference, pedestrian centers are decoded by applying 2D max-pooling with kernel size $K_{nms} \in \{5, 7, 9\}$ to extract local spatial maxima. Predictions with confidence values exceeding a threshold $\tau_{det} \in [0.70, 0.95]$ that lie within the valid region of interest (ROI) mask are retained as detected pedestrian coordinates.

3.5 Alternative Architectural Mechanisms Under Study

To understand the design space more fully, we also implement and analyze two alternative mechanisms:

Soft-depth sampling. To account for uncertainty along camera sightlines, we perturb each 3D point $\mathbf{P}$ along the optical ray with $K_{soft} = 3$ stochastic Gaussian samples:

$$\mathbf{P}^{(k)} = \mathbf{P} + \delta_k \cdot \frac{\mathbf{P} - \mathbf{C}_c}{\|\mathbf{P} - \mathbf{C}_c\|_2}, \quad \delta_k \sim \mathcal{N}(0, \sigma_{soft}^2), \quad \sigma_{soft} = 15 \text{ cm}. \quad (3.15)$$

Lightweight detection compensation. A pedestrian that SAM only partially segments, or misses entirely, in one camera’s person mask M_{fg}^c receives sparse or absent point support from that view. This can happen even when the same pedestrian is clearly segmented by other cameras. We implement a lightweight cross-view compensation module to recover such missed detections. It operates directly in 2D mask space, before the usual per-camera depth unprojection. Cycling through every camera r in turn as a *reference* view, we test whether the remaining cameras corroborate a person that r ’s own SAM mask missed:

1. **Cross-view projection.** For each other camera $s \neq r$, we unproject the person-mask pixels of s into 3D using s ’s own scale-aligned depth map. We then reproject these points into camera r via $\pi_r(\mathbf{P}) = \mathbf{K}_r(\mathbf{R}_r\mathbf{P} + \mathbf{t}_r)$. Only points landing in front of r and within its image boundaries are kept.
2. **Pooled clustering.** The reprojected pixels from *all* source cameras $s \neq r$ are pooled into a single 2D point set in r ’s image plane, with each point tagged by its source camera id. We then cluster this pooled set with DBSCAN.
3. **Cluster acceptance filters.** A cluster is accepted as a missed detection only if it passes four checks. First, it must contain at least τ_{pcs} points. Second, its points must originate from at least κ distinct source cameras; this rejects mask artefacts idiosyncratic to a single camera’s own SAM output, such as a shadow or a silhouette edge, rather than a genuinely corroborated pedestrian. Third, it must overlap r ’s existing person mask by less than a fraction τ_m , so already-detected pedestrians are not duplicated. Fourth, unprojecting the cluster through r ’s own scale-aligned depth must place its median height within a plausible standing-pedestrian range $[Z_{min}, Z_{max}]$, which rejects projections landing on the ground plane or on background structures.
4. **Mask synthesis.** Each accepted cluster is morphologically dilated with a fixed-radius elliptical structuring element to approximate a person silhouette, a fast substitute for invoking SAM again on r . The newly covered pixels, excluding any already inside r ’s existing mask, are written into r ’s person mask as a new instance. This addition is capped at a maximum fraction of the image area, which rejects spurious oversized blobs caused by erroneous depth.

The compensation therefore acts purely on the 2D person masks. The enriched mask for camera r replaces the original SAM output and is passed through the unchanged scale-alignment and unprojection pipeline (Section 3.2.3). Compensatory points enter $\mathbf{P}_{fused}$ only once per-camera depth unprojection is re-run on the enriched masks – they are not inserted into $\mathbf{P}_{fused}$ directly.

For the result reported in Table 4.3, we use $\tau_{pcs} = 100$, $\tau_m = 0.3$, a DBSCAN radius of 20 px with a minimum of 15 points per cluster, $\kappa = 1$ corroborating camera, an ellipse dilation radius of 10 px, a maximum added area of 1.5% of the frame, and a height range of $[0, 500]$ cm. With $\kappa = 1$, the multi-camera consensus check does not by itself require corroboration. In practice, however, most accepted clusters already receive projections from several source cameras, since a genuinely missed pedestrian is typically visible from more than one other view.

4

Results

This chapter reports the experimental evaluation of the proposed foundation-model multi-view pedestrian detection framework. We describe the benchmarks, evaluate the geometric fidelity of the 3D foundation model, compare full-camera detection performance against dedicated baselines, report architectural ablations and training loss dynamics, and present cross-subset camera-rig generalization experiments.

4.1 Experimental Setup

4.1.1 Datasets

We conduct evaluations on two established multi-view pedestrian detection benchmarks representing real-world and synthetic environments:

Wildtrack Benchmark [3]. Wildtrack captures a crowded outdoor plaza measuring $12 \times 36 \text{ m}^2$ at ETH Zurich using seven synchronized 1080p cameras running at 60 fps. The monitored space exhibits severe dynamic occlusions, pedestrians in diverse attire, and complex lighting conditions. We adopt the standard split of 400 synchronized multi-camera frames (frames 0000 to 1995 sampled every 5 frames), where the first 360 frames serve as training data and the final 40 frames serve as the test set. Figure 4.1 illustrates the physical camera positions and horizontal fields of view covering the plaza.

Wildtrack camera layout — cross-subset partition

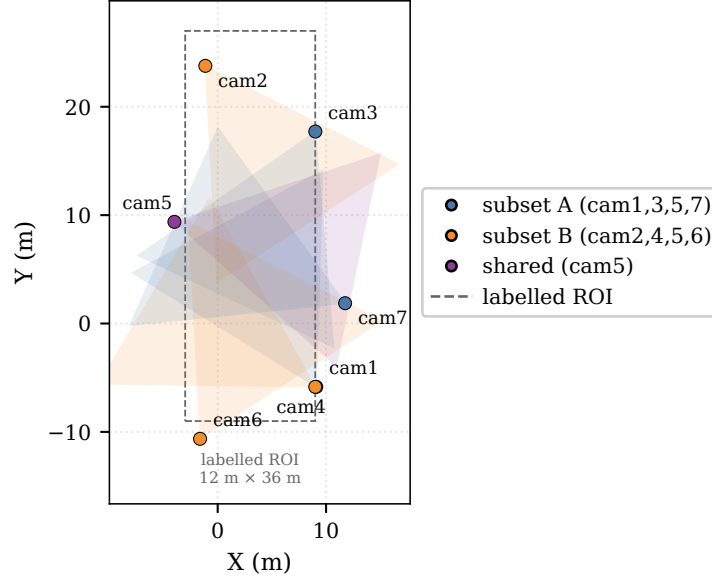


Figure 4.1: Camera layout of the Wildtrack dataset. Seven elevated cameras surround a $12 \times 36 \text{ m}^2$ labeled ground plaza (dashed boundary). Cameras are color-coded by cross-subset partition: Subset A (cams 1, 3, 5, 7; blue), Subset B (cams 2, 4, 5, 6; orange), and shared center cam 5 (purple).

MultiviewX Benchmark [6]. MultiviewX is a synthetic benchmark simulated in the Unity engine, representing a crowded public concourse covering $16 \times 25 \text{ m}^2$ observed by six calibrated cameras at 1080×1920 resolution. The benchmark contains 400 frames, split into 360 training frames and 40 evaluation frames. Figure 4.2 illustrates the physical camera positions and cross-subset partition used in our generalization experiments (Table 4.5).

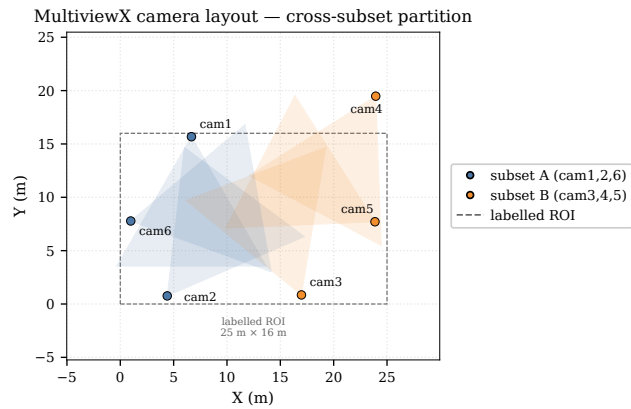


Figure 4.2: Camera layout of the MultiviewX dataset. Six calibrated cameras surround a $16 \times 25 \text{ m}^2$ labeled ground plaza (dashed boundary). Cameras are color-coded by cross-subset partition: Subset A (cams 1, 2, 6; blue) and Subset B (cams 3, 4, 5; orange).

4.1.2 Evaluation Metrics

Following the CLEAR MOT/MOD standard [24], pedestrian localization accuracy is evaluated in the BEV ground plane under a Euclidean matching distance threshold $r = 0.5$ m:

- **MODA (Multiple Object Detection Accuracy):** Accounts for both missed detections (false negatives, FN) and spurious detections (false positives, FP):

$$\text{MODA} = 1 - \frac{\sum_t (FN_t + FP_t)}{\sum_t N_{gt,t}}, \quad (4.1)$$

where $N_{gt,t}$ is the ground-truth pedestrian count at frame t .

- **MODP (Multiple Object Detection Precision):** Quantifies the spatial localization precision of true positive matches (TP):

$$\text{MODP} = \frac{\sum_{i \in TP} (1 - d_i/r)}{|TP|}, \quad (4.2)$$

where d_i is the ground-plane distance between the detection and matched ground truth.

- **Precision and Recall:** $\text{Prec} = \frac{TP}{TP+FP}$ and $\text{Rec} = \frac{TP}{TP+FN}$.

4.1.3 Implementation Details

Models are trained for 40 epochs with the Adam optimizer, an initial learning rate of 4×10^{-4} decayed by a factor of 0.1 at epochs 25 and 35, weight decay of 5×10^{-3} , batch size 1 with gradients accumulated over 4 iterations (effective batch size 4), and random seed 7777 on a single NVIDIA DGX Spark. The BEV ground plane is discretized into 10×10 cm grid cells, each grouping 4×4 cells of the 2.5 cm annotation grid. Fused 3D points are regularized with a point dropout probability of 0.5. At test time, grid sweeps over NMS kernel size $K_{nms} \in \{5, 7, 9\}$ and confidence threshold $\tau_{det} \in \{0.83, 0.85, 0.87, 0.90\}$ identify the optimal operating point; the two RGB baselines were swept over $K_{nms} \in \{5, 7\}$, $\tau_{det} \in \{0.80, 0.85, 0.90\}$ (Wildtrack) and $K_{nms} \in \{3, 5, 7, 9\}$, $\tau_{det} \in \{0.70, 0.80, 0.85, 0.90\}$ (MultiviewX).

4.2 Foundation-Model Geometric Evaluation and Performance Ceilings

4.2.1 Calibrated vs. Uncalibrated Geometry

We first evaluate the geometric fidelity of the MapAnything 3D foundation model and its direct impact on downstream detection. Figure 4.3 shows a qualitative example of the model’s raw per-camera depth output. Figure 4.4 presents the fused point cloud projections to explicitly contrast the structural differences between calibrated and uncalibrated modes from a top-down perspective. Table 4.1 quantitatively compares the two modes on Wildtrack.

The empirical results validate our theoretical error decomposition (visually corroborated by the top views in Figure 4.4):

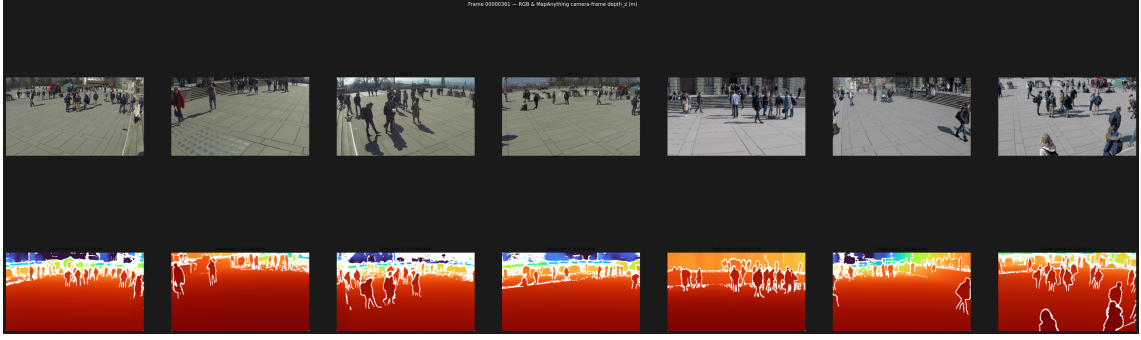
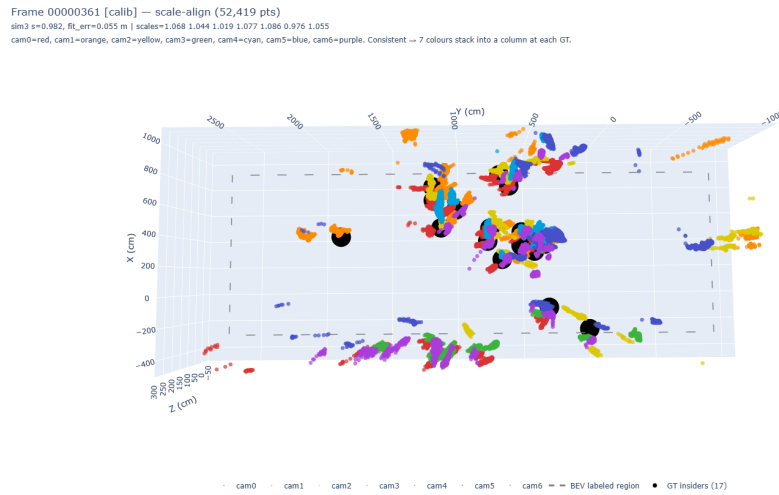


Figure 4.3: Per-camera RGB frames (top row) and corresponding MapAnything depth predictions D_{pred}^c (bottom row) for all seven Wildtrack cameras on frame 00000361. Warmer colors denote smaller (closer) metric depth; white silhouettes correspond to pedestrians segmented against the ground plane.

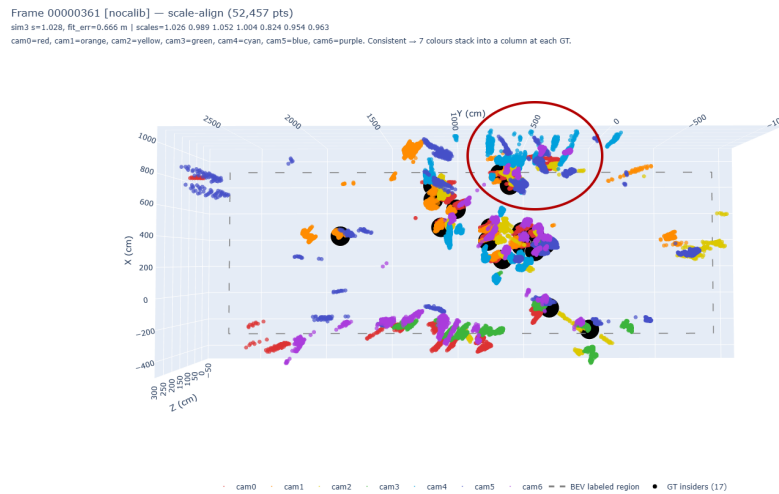
Table 4.1: Geometric fidelity and downstream detection performance of MapAnything under Calibrated versus Uncalibrated inference modes on Wildtrack. All three rows share an identical fused-point-cloud, RGB-feature detection architecture. The middle row isolates the per-camera ground-plane median scale alignment (Section 3.2.3) from the effect of the calibration input to MapAnything itself: it uses the same calibrated MapAnything geometry as the bottom row, but corrected by only a single global Sim3 transform fit to the camera centers, with no per-camera correction.

Mode	Sim3 Res. (e_{sim3})	Scale Ratio (s)	MODA	MODP	Recall	Prec.
Uncalibrated	66.0 cm	1.001	29.1	49.4	55.5	67.8
Calibrated	5.5 cm	0.982	55.3	57.5	73.8	79.9
Calibrated + per-camera Scale Align	5.5 cm	0.982	59.9	62.2	82.0	78.7

1. Uncalibrated failure: although uncalibrated reconstruction produces a visually coherent, single-layer ground plane, the 66.0 cm non-rigid frame-varying camera-center residual displaces pedestrian pillars across about 7 BEV cells. This is directly visible in Figure 4.4b: in the red-circled region, cam4’s points (cyan) fail to collapse onto the shared ground-truth marker with the other six cameras and instead split into several offset streaks, producing duplicated, ghost-like residuals rather than a single coherent pedestrian pillar. Because Global Umeyama alignment fits one rigid transform per frame, it cannot correct this per-camera, within-frame drift, and downstream detection accuracy falls to 29.1 MODA.
2. Calibrated low-rank rectifiability: calibrated inference anchors camera centers within 5.5 cm of truth. Simply calibrating MapAnything and applying a single global Sim3 transform, with no per-camera correction, already recovers most of the gap: MODA rises to 55.3, a 26.2-point improvement over the uncalibrated result. The dominant remaining per-camera depth drift is low-rank, however, and our ground-plane median scale alignment removes most of it on top of that, yielding aligned pillars (Figure 4.4a) and a further 4.6-point improvement to 59.9 MODA. The two corrections combined give the full 30.8-point gap over the uncalibrated result. Scale alignment does not remove all cross-view error:



(a) Calibrated geometry (aligned)



(b) Uncalibrated geometry (drifted)

Figure 4.4: Top-view comparison of the fused 3D point clouds. (a) Calibrated inference yields well-aligned pedestrian pillars. (b) Uncalibrated inference drifts; the red circle marks cam4 (cyan) splitting into offset ghost streaks instead of one shared pillar.

a residual dispersion of 30–80 cm remains (Section 3.4.1).

4.2.2 Full-Camera Benchmark Performance and Structural Ceiling

Table 4.2 summarizes the benchmark detection results across all cameras on Wildtrack (7 cameras) and MultiviewX (6 cameras).

Table 4.2: Full-camera detection performance of the proposed foundation-model pipeline (DilatedBEVBackbone + FusedLSSLift ResNet-64) compared to DCHM. The DCHM row is *our own reproduction* of the supervised DCHM pipeline with a single round of Gaussian optimization and depth fine-tuning; the results reported in the original paper are given in the text.

Dataset	Method / Geometry Source	MODA $\uparrow$	MODP $\uparrow$	Recall $\uparrow$	Prec. $\uparrow$
Wildtrack (7 cams)	DCHM [16], supervised, single-round GS finetune (<i>our reproduction</i>)	83.0	–	–	–
	Ours (RGB Baseline)	59.9	62.2	82.0	78.7
	Ours (Dilated + ResNet-64, Best)	68.7	62.5	85.3	83.7
MultiviewX (6 cams)	Ours (RGB Baseline)	62.5	73.6	73.1	87.3
	Ours (Dilated + ResNet-64, Best)	64.0	72.4	74.3	87.9

Gap to per-scene optimization. Our best recipe achieves 68.7 MODA on Wildtrack and 64.0 MODA on MultiviewX. For DCHM, the supplementary material of the original paper [16] reports approximately 84 MODA (unsupervised) and approximately 95 MODA (supervised) on Wildtrack. These are the best results of its full pipeline, which iterates Gaussian optimization and depth fine-tuning over several rounds, and we include them for context only. As a reference that we could run ourselves, we reproduced supervised DCHM with a single iteration only, which reaches 83.0 MODA, about 12 MODA below the best reported supervised result. Our best recipe is 14.3 MODA below this reproduction. Because our detector is also trained with ground-truth annotations, the best reported supervised result is the like-for-like published reference, and our recipe is roughly 26 MODA below it; the unsupervised result, which is roughly 15 MODA above ours, is listed for completeness.

Our reproduction is not DCHM’s raw geometric clustering result; it is obtained by training the same supervised PointPillars-style detector we use (Section 3.4.2) on the Gaussian-Splatting-optimized point cloud from a single iteration, exactly as we train it on MapAnything’s output. Because the detector is held fixed, the comparison with this 83.0 MODA reproduction isolates the geometry source, and the 14.3-point margin is consistent with a limitation in the geometry:

- DCHM optimizes scene-specific Gaussian primitives iteratively over thousands of steps on the exact evaluation camera views, fitting the subtle multi-camera disparities of that specific scene at the cost of hours of computation per sequence.
- Our pipeline instead performs purely zero-shot, feedforward 3D geometry extraction via MapAnything, requiring no per-scene geometric training. The residual 30–80 cm cross-view point dispersion introduces spatial ambiguity that limits downstream precision.

4.3 Architectural Ablations and Mechanism Discovery

To understand the individual contributions and interactions of our proposed architectural components, we conduct controlled ablations around the best recipe on Wildtrack. Table 4.3 reports the complete experimental ledger, corresponding directly to the horizontal bar chart in Figure 4.5.

Table 4.3: Wildtrack architectural ablation ledger.

Architecture Variant	MODA	MODP	Rec. (%)	Prec. (%)	Δ (vs Best)
RGB Baseline	59.9	62.2	82.0	78.7	−8.8
Dilated ($d = 2$) + RGB	62.8	62.3	81.0	81.7	−5.9
Soft depth + ResNet	65.5	62.2	82.4	83.1	−3.2
Soft depth + dilated + ResNet	66.0	60.9	80.4	84.8	−2.7
Lightweight comp. + dilated + ResNet	67.1	63.2	84.7	82.8	−1.6
HDC Pattern [1, 2, 4, 2, 1] + ResNet	68.2	62.9	81.4	86.0	−0.5
Dilated ($d = 2$) + ResNet-64 [Best]	68.7	62.5	85.3	83.7	0.0

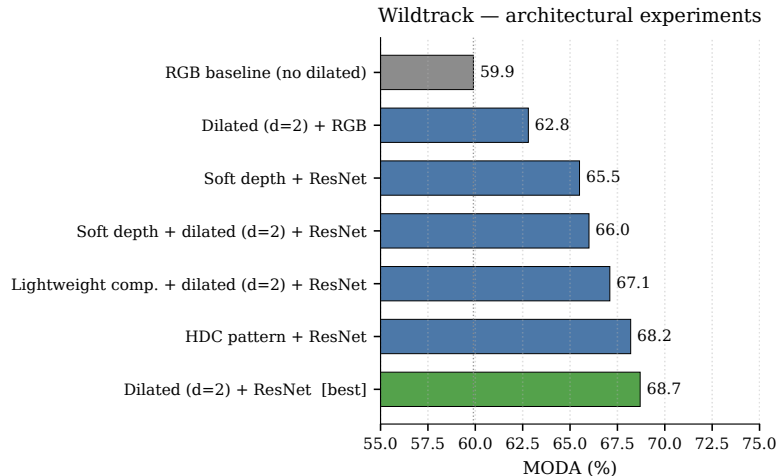


Figure 4.5: Ablation of architectural variants on the Wildtrack dataset. Dilated convolutions with ResNet-64 feature lifting achieve the highest performance (68.7% MODA), whereas combining soft depth with dilated convolutions or adding lightweight compensation degrades accuracy.

4.3.1 Analysis of Observations

Dilated receptive field and deep features. Introducing dilated convolutions ($d = 2$) into the BEV backbone while retaining raw RGB point colors improves MODA from 59.9 to 62.8 (+2.9 points), driven by a 3.0 percentage point gain in Precision (78.7% $\rightarrow$ 81.7%). Dilation doubles the sampling spacing of every inner 3×3 convolution and thus enlarges the receptive field at each scale of the backbone without adding parameters.

When deep ResNet-18 features are integrated via our FusedLSSLift module (Section 3.3.2) the "+ ResNet" rows in Table 4.3 and Figure 4.5 performance rises

to 68.7 MODA (+5.9 points over Dilated + RGB). Both Recall (81.0% $\rightarrow$ 85.3%) and Precision (81.7% $\rightarrow$ 83.7%) increase together. Deep features only help once something aggregates the scattered per-camera representations. We tested ResNet features alone, without dilation or soft depth, on Wildtrack; MODA did not improve over the RGB baseline. The same pattern holds on MultiviewX: a plain ResNet backbone without dilation reaches 62.3 MODA, essentially flat against the 62.5 RGB baseline (Table 4.2), well short of the 64.0 MODA the dilated ResNet variant reaches. The gain reappears once either of two aggregation mechanisms is present. An expanded convolutional receptive field (dilation) contributes the +5.9 points above. Soft depth shows a similar pattern: adding it on top of ResNet features raises MODA from 59.9 (RGB Baseline) to 65.5 (Soft depth + ResNet). This 65.5 reflects soft depth and ResNet together, not soft depth on its own; we did not run a separate Wildtrack ablation with soft depth on plain RGB features. Since ResNet alone left MODA unchanged (above), most of this gain is likely driven by soft depth, but the table does not isolate the two effects, so this attribution is an inference rather than a direct measurement. Because both mechanisms aggregate in a similar way, stacking them is redundant rather than additive: Soft depth + dilated + ResNet reaches only 66.0, a further +0.5 over Soft depth + ResNet. The next paragraph explains why combining the two can even hurt.

Over-smoothing from combining soft depth and dilation. Soft-depth sampling adds stochastic Gaussian jitter ($\sigma_{soft} = 15$ cm, $K_{soft} = 3$) along camera sightlines during lifting. Combined with ResNet features, it reaches 65.5 MODA (Soft depth + ResNet in Table 4.3).

However, when it is combined with dilated convolutions, performance drops to 66.0 MODA (-2.7 points relative to the best recipe), and Recall falls from 85.3% to 80.4% (-4.9 pp).

- Soft depth spreads point features across ± 30 cm in physical space.
- The dilated backbone simultaneously enlarges the convolutional aggregation footprint.
- The compounding of spatial blurring produces an *over-smoothing* effect: adjacent pedestrians in crowded groups merge into contiguous activation blobs. During post-processing, NMS suppresses the conjoined responses into a single detection, causing systematic missed detections in dense crowds.

False-positive noise from detection compensation. Section 3.5 describes the lightweight detection compensation module, which recovers pedestrians that SAM misses in individual camera views. Adding it to the best recipe reduces MODA to 67.1 (-1.6 points). The module expands mask boundaries without running full instance re-segmentation. As a result, it also picks up background pillars and ground shadows. These spurious points add false-positive noise to the pillar grid and reduce the detector’s sensitivity.

HDC precision–recall trade-off. The HDC variant achieves 68.2 MODA, close to the best recipe. HDC reaches the highest Precision of all tested configurations

(86.0%, +2.3 pp over the best recipe), at the cost of reduced Recall (81.4%, −3.9 pp). Instead of a uniform $d = 2$, HDC mixes dilation rates within each block ($[1, 2, 4, 2, 1]$); its $d = 1$ layers restore dense sampling of neighboring cells, which may explain the suppression of marginal false positives, although we did not measure gridding artifacts directly. This makes HDC a candidate for high-precision surveillance.

4.3.2 Training Loss Convergence Dynamics

Figure 4.6 plots the 40-epoch training and validation loss curves across all architectural variants.

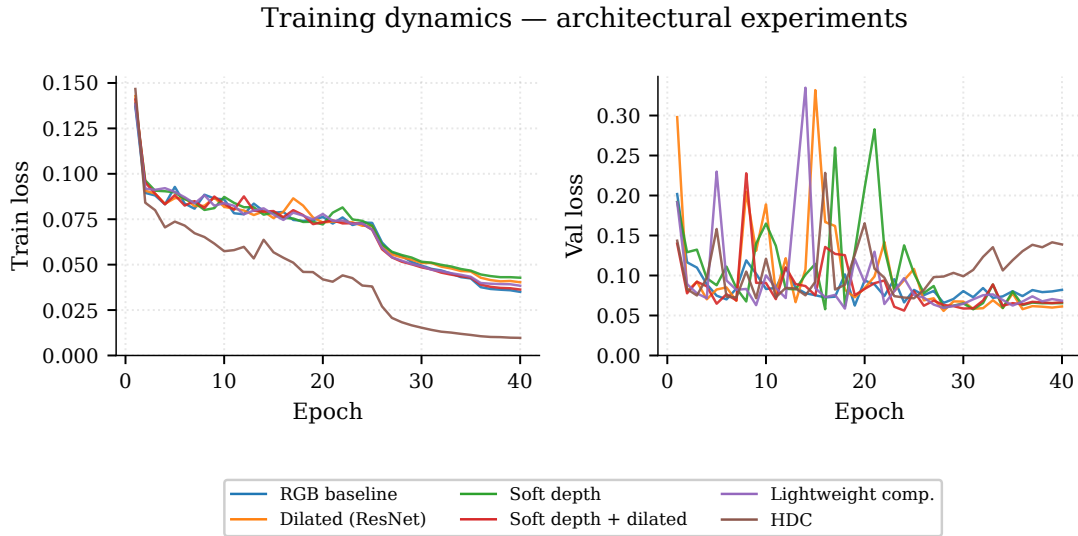


Figure 4.6: Training and validation loss convergence dynamics across 40 epochs on Wildtrack. The left panel shows training loss; the right panel shows validation loss, which settles for most variants after the learning rate step-downs at epochs 25 and 35.

The loss curves exhibit distinct convergence profiles:

- Training loss decreases steadily for all models. At epoch 40, five variants reach 0.035–0.043, whereas HDC reaches 0.010.
- Early validation loss (epochs 1–25) is least variable for the RGB baseline (standard deviation 0.027, peak 0.20). The ResNet-feature variants show larger spikes, with peaks of up to 0.33 for Dilated + ResNet-64 and Lightweight comp., so dilated convolutions do not by themselves stabilize validation loss.
- After the learning rate reductions at epochs 25 and 35, validation loss settles between 0.061 and 0.082 for five variants. HDC is the exception: its validation loss rises from 0.119 at epoch 35 to 0.139 at epoch 40 while its training loss reaches 0.010, which indicates overfitting.

4.4 Cross-Subset Generalization and Camera-Rig Dependency

A common assumption in the multi-view literature is that geometric projection networks exhibit *calibration invariance*, meaning that a network trained on one subset of calibrated cameras can operate on a different camera subset. We test this hypothesis through cross-subset experiments under our best-performing architecture (Dilated + ResNet-64).

In these experiments, a camera subset determines which cameras contribute geometry: for each subset, the fused point cloud is rebuilt from the per-camera depth maps of the subset cameras only. Two parts of the pipeline are not restricted to the subset. First, the per-camera depth maps come from a single MapAnything inference over the full rig (7 cameras on Wildtrack, 6 on MultiviewX), so the depth of each subset camera was predicted with the other cameras as context. Second, FusedLSSLift samples image features from all camera views. The experiments therefore measure the effect of changing the cameras that contribute geometry while image features from the full rig remain available; they are not a full camera-rig transfer. The results below indicate a strong dependency on the cameras used to build the training point cloud.

4.4.1 Cross-Subset Evaluation Matrices

Wildtrack 7→4 Camera Partition. We partition the 7 cameras into two 4-camera subsets: Subset A = {0, 2, 4, 6} and Subset B = {1, 3, 4, 5} (0-based indices; cams 1, 3, 5, 7 and 2, 4, 5, 6 in Figure 4.1), sharing camera index 4 (cam 5), the camera closest to the plaza center. Table 4.4 presents the 2×2 evaluation matrix.

Table 4.4: Cross-subset evaluation matrix on Wildtrack (4-camera subsets, Dilated + ResNet-64 configuration). Metrics: MODA / MODP / Recall / Precision.

Training Setup	Eval on Subset A (0,2,4,6)	Eval on Subset B (1,3,4,5)	Row Average
Train Subset A	61.4 / 64.5 / 74.7 / 84.9	15.7 / 49.6 / 41.3 / 61.7	38.55
Train Subset B	40.3 / 57.6 / 60.5 / 75.0	51.3 / 60.9 / 68.7 / 79.8	45.80
Diagonal Mean: 56.35 MODA Off-Diagonal Mean: 28.00 MODA Generalization Gap: -28.35 MODA			

MultiviewX 6→3 Camera Partition. On MultiviewX, 6 cameras are divided into disjoint 3-camera subsets: Subset A = {0, 1, 5} and Subset B = {2, 3, 4} (0-based indices; cams 1, 2, 6 and 3, 4, 5 in Figure 4.2). Table 4.5 presents the resulting evaluation matrix.

4.4.2 Discussion on Camera-Rig Specificity

The cross-subset results support four conclusions:

Dependency on camera topology. On Wildtrack, transferring between 4-camera subsets results in a drop of -28.35 MODA points (from 56.35 to 28.00). On MultiviewX, transferring between disjoint 3-camera subsets causes an -18.05 MODA

Table 4.5: Cross-subset evaluation matrix on MultiviewX (disjoint 3-camera subsets, Dilated + ResNet-64 configuration). Metrics: MODA / MODP / Recall / Precision.

Training Setup	Eval on Subset A (0,1,5)	Eval on Subset B (2,3,4)	Row Average
Train Subset A	35.6 / 69.0 / 55.0 / 73.9	17.4 / 60.1 / 28.9 / 71.6	26.50
Train Subset B	11.7 / 58.0 / 33.4 / 60.6	29.6 / 70.0 / 46.8 / 73.1	20.65
Diagonal Mean: 32.60 MODA Off-Diagonal Mean: 14.55 MODA Generalization Gap: -18.05 MODA			

drop (from 32.60 to 14.55). Downstream BEV convolutional layers appear to overfit to the spatial point-density and cross-view ray-intersection patterns of the specific camera rig on which they are trained.

Asymmetric transferability. Transferability between subsets is asymmetric. On Wildtrack, Train B $\rightarrow$ Eval A preserves 40.3 MODA, whereas Train A $\rightarrow$ Eval B drops to 15.7 MODA. As shown in Figure 4.1, Subset B has broader camera baselines than Subset A (mean pairwise distance between camera centers 22.0 m vs. 16.8 m; convex hull of camera positions 227 m² vs. 185 m²), which plausibly exposes its learned spatial filters to a wider range of viewing geometries and helps them extrapolate to novel angles.

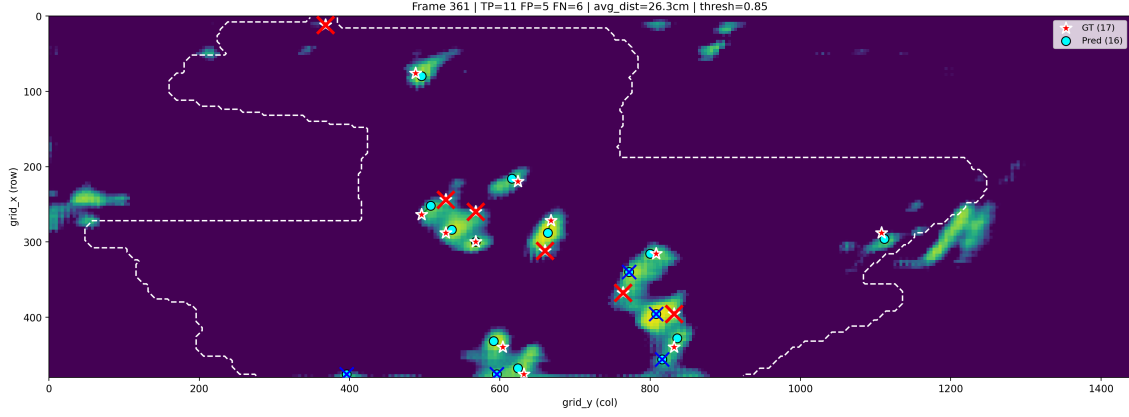
Feature density threshold. Reducing camera count from 6 to 3 on MultiviewX lowers in-subset MODA by -31.4 points (64.0 $\rightarrow$ 32.6), with Recall falling to 46–55%. When point density drops below a spatial threshold, the 3D foundation reconstruction no longer provides enough points to sustain BEV feature extraction.

Synthetic scene overfitting on MultiviewX. On MultiviewX, upgrading from the RGB baseline (62.5 MODA) to Dilated + ResNet-64 (64.0 MODA) gives only a +1.5 gain (compared to +8.8 on Wildtrack). The training dynamics show strong overfitting on MultiviewX: at epoch 40, the Wildtrack model has a validation-to-train loss ratio of 1.5 $\times$ (val = 0.061, train = 0.040), whereas the MultiviewX model has a 17 $\times$ ratio (val = 0.265, train = 0.016). One plausible cause is that the uniform pedestrian textures and repetitive background of the Unity environment let ResNet-18 memorize synthetic features. The loss ratio alone does not isolate this cause, however: the HDC variant on the real Wildtrack data also reaches a 14 $\times$ ratio (val = 0.139, train = 0.010).

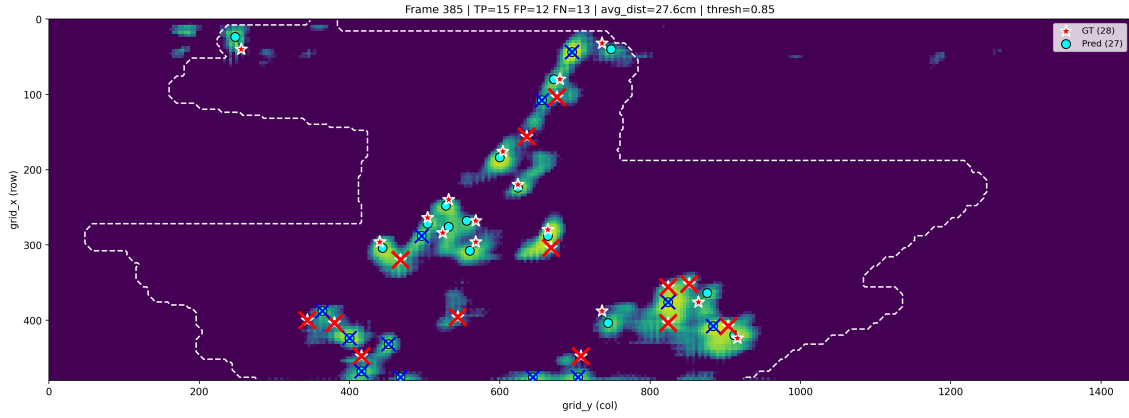
4.5 Qualitative BEV Diagnostics and Precision–Recall Trade-offs

4.5.1 Qualitative Heatmap Diagnostics

Figure 4.7 displays qualitative BEV diagnostic visualizations across representative evaluation frames on the Wildtrack dataset, comparing ground truth (red stars) against model predictions (cyan circles) for both sparse distribution (Figure 4.7a) and dense crowd interaction (Figure 4.7b).



(a) Evaluation frame 361: localization across the central plaza and outer boundaries.



(b) Evaluation frame 385: dense pedestrian clusters and multi-person interaction in the monitored concourse.

Figure 4.7: Qualitative BEV diagnostic heatmaps on Wildtrack evaluation frames under the best recipe (Dilated + ResNet-64). Ground truth positions are marked as red stars, predicted detections as cyan circles, true-positive associations as green lines, missed detections (FN) as red crosses, and spurious false alarms (FP) as blue crosses. The dashed white contour denotes the labeled evaluation ground-plane ROI.

The diagnostic visualizations highlight several key behaviors:

- Single-peak focus in multi-view coverage. In central plaza regions observed by 4 or more cameras (e.g., central clusters in Figure 4.7a and Figure 4.7b), our DilatedBEVBackbone produces focused, single-peak occupancy heatmaps that match ground truth with sub-20 cm error.
- Boundary misses under grazing viewing angles. Missed detections (red crosses) occur mostly along the outer boundary of the plaza (e.g., upper-left boundary in Frame 361 and bottom boundary in Frame 385), where pedestrians are visible from only one or two cameras with grazing viewing angles.
- Suppression of spurious alarms. Spurious detections (blue crosses) remain few, appearing mainly when multiple pedestrians walk in direct physical contact.

4.5.2 Precision–Recall Pareto Analysis

Figure 4.8 illustrates the Precision–Recall operating characteristic across all evaluated NMS kernel sizes and detection thresholds, plotted against theoretical MODA contour iso-lines.

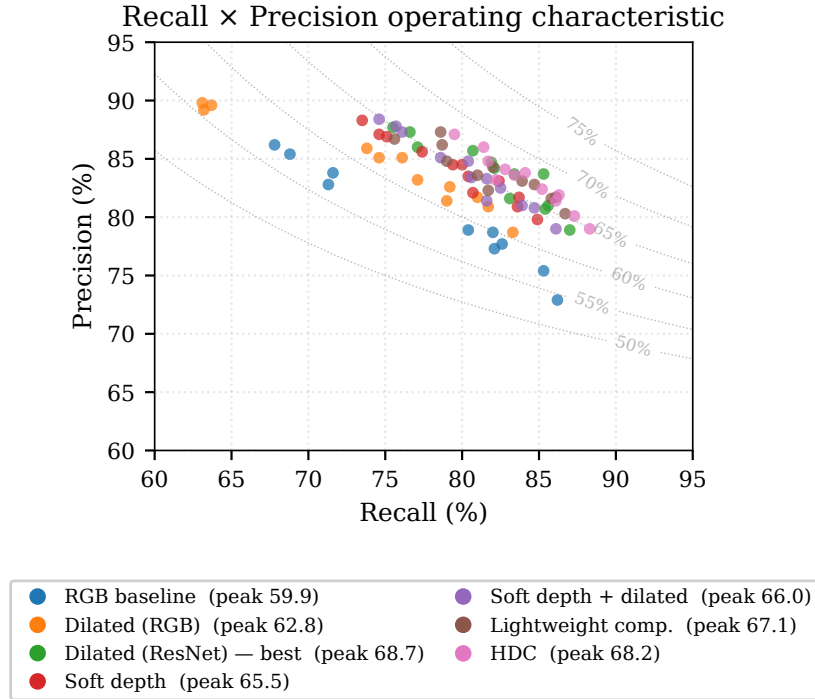


Figure 4.8: Recall $\times$ Precision Pareto scatter plot across threshold and NMS sweeps on Wildtrack. Dotted contours depict theoretical MODA iso-lines. The Pareto frontier combines operating points from five variants; no single variant dominates.

The scatter plot separates the operating regimes of the architectural variants:

- The best recipe (Dilated + ResNet-64) reaches the highest MODA (68.7) with operating points spanning recall 75.5–87.0% and precision 78.9–87.7%. It contributes 3 of the 18 Pareto-optimal points, including its 68.7 MODA operating point.
- HDC contributes the most Pareto-optimal points (8 of 18), spanning recall 79.5–88.3% and precision 79.0–87.1%. Only 2 of its 12 operating points exceed 85% precision.
- The remaining Pareto-optimal points come from Lightweight comp. (3), Dilated + RGB (2), and Soft depth + dilated + ResNet (2); the latter two lie at the high-precision, low-recall end. The RGB baseline and Soft depth + ResNet contribute no Pareto-optimal points.

5

Discussion

This chapter discusses the empirical findings, the architectural trade-offs, and the practical implications of using 3D vision foundation models for metric multi-view pedestrian detection. The following sections examine the performance boundaries, the interactions between architectural components, camera-rig dependency, and the limitations and future work that follow from them.

5.1 Findings and Performance Bounds

A central question addressed in this thesis is whether off-the-shelf 3D foundation models can replace labor-intensive, scene-specific optimization pipelines. Conventional methods like DCHM [16] optimize 3D Gaussian primitives [8] and fine-tune monocular depth models over multiple rounds and multiple hours for a given camera installation. The supplementary material of the original paper reports approximately 84 MODA (unsupervised) and approximately 95 MODA (supervised) on Wildtrack for this full pipeline. Our own single-iteration reproduction of supervised DCHM reaches 83.0 MODA, and our best feedforward recipe, which is also trained with ground-truth annotations, reaches 68.7 MODA. Our results therefore support only a qualified answer: a foundation-model pipeline removes the per-scene optimization stage, but it does not yet match the accuracy of the optimized pipeline.

The resulting 14.3-point margin to our reproduction, which widens to roughly 26 points against the best reported supervised result, reflects a trade-off between dedicated per-scene optimization and generalist zero-shot inference. Scene-specific Gaussian optimization minimizes photometric residuals over many iterations and can absorb small calibration misalignments, lens distortions, and scene-specific occlusion boundaries. An un-finetuned foundation model like MapAnything [19] instead relies on pre-trained geometric priors, and produces point clouds with a cross-view dispersion of 30 to 80 cm. Downstream convolutional layers can partly accommodate this spatial uncertainty, but the residual geometric ambiguity sets an empirical performance ceiling. At the same time, skipping scene-specific optimization removes the per-scene optimization stage, which is an operational advantage in surveillance setups where cameras are frequently reconfigured.

A further observation concerns the geometric error contrast between the calibrated and uncalibrated modes. Under calibrated camera poses, MapAnything gives a camera-center alignment residual of $e_{sim3} = 5.5$ cm, together with a 5% to 10% per-camera depth scale drift. Because the dominant component of this error is low-

rank, our closed-form ground-plane median scaling removes most of it by aligning each camera’s points to the physical ground plane ($Z = 0$). It does not restore full cross-view consistency: the remaining 30–80 cm dispersion reflects error that a single per-camera scale factor cannot absorb.

Uncalibrated inference, by contrast, gives an average camera-center residual of 66.0 cm, which appears as frame-to-frame coordinate jitter that displaces pedestrian pillars across about 7 BEV grid cells. Downstream detection falls to 29.1 MODA because global rigid transformations cannot absorb non-rigid pose shifts. This contrast indicates that, for metric surveillance tasks that require centimeter-level localization, calibrated camera parameters remain a necessary spatial constraint.

5.2 Architectural Insights and Interaction Mechanisms

Accommodating the 30–80 cm cross-view point dispersion required targeted architectural changes. Introducing dilated convolutions ($d = 2$) in DilatedBEVBackbone doubles the sampling spacing of every inner convolution and enlarges the receptive field at each scale of the backbone without adding parameters; with RGB features this improves MODA by 2.9 points, mainly through a 3.0 percentage point gain in precision. Combining this larger receptive field with ResNet-18 feature projection (FusedLSSLift) raises detection accuracy to 68.7 MODA (+8.8 points over the RGB baseline), which indicates that deep semantic features help separate low-density pedestrian boundaries once the spatial aggregation footprint is broad enough.

Our ablations also revealed a failure mode in the interaction between soft-depth sampling and dilated convolutions. Soft-depth sampling perturbs points along camera projection rays (± 30 cm), while DilatedBEVBackbone gathers features over an enlarged spatial neighborhood. When combined, the two forms of spatial broadening compound into an over-smoothing effect: adjacent pedestrians in crowded areas merge into contiguous activation blobs in BEV feature space. During post-processing, Non-Maximum Suppression (NMS) then suppresses neighboring genuine detections, lowering Recall by 4.9 percentage points (85.3% to 80.4%) and overall MODA to 66.0. This shows that synthetic depth dispersion has to be matched to the receptive field of the spatial encoder.

The lightweight detection compensation module (Section 3.5) reduced MODA to 67.1. Because cross-view reprojection ran without dense instance re-segmentation, the back-projected geometry sampled ground shadows and background clutter and added false-positive noise to the pillar grid. Hybrid Dilated Convolutions (HDC), on the other hand, reached 86.0% precision (+2.3 points over the best recipe), which suggests that non-uniform dilation patterns reduce gridding artifacts and suit high-precision surveillance.

Comparing Wildtrack and MultiviewX also shows different behavior between the real and synthetic domains. The architectural changes gave an 8.8-point MODA gain on Wildtrack but only a 1.5-point gain on MultiviewX. The validation-to-train loss ratio on MultiviewX ($17\times$, compared to $1.5\times$ for the corresponding Wildtrack

model) is consistent with the ResNet backbone memorizing the uniform synthetic textures and static lighting. The ratio alone does not isolate this cause, since the HDC variant on the real Wildtrack data also reaches $14\times$.

5.3 Camera-Rig Specificity and Calibration Invariance

Prior multi-view detection approaches, such as MVDet and GMVD, yield BEV representations that are inherently tied to the training camera setup. A key question is whether LSS-based frameworks can overcome this limitation and provide better generalization across diverse camera rigs by leveraging accurate 3D point clouds. In our cross-subset experiments, we change which cameras contribute to the fused point cloud, while per-camera depth still comes from full-rig MapAnything inference and image features are sampled from all views. Even this partial change causes large performance drops, demonstrating that explicit 3D geometry alone is insufficient for LSS architectures to generalize to novel camera arrangements.

Transferring models across camera subsets produced large gaps of -28.35 MODA on Wildtrack (4-camera subsets) and -18.05 MODA on MultiviewX (3-camera subsets). These results indicate that downstream BEV convolutional networks are coupled to the training camera topology. When multi-view observations are scattered onto the ground plane, the resulting point distributions reflect the specific ray intersections and viewing angles of the physical camera rig, and the downstream convolutional filters overfit to these spatial ray-density patterns. When the point cloud is built from a different camera subset, the shifted intersection angles and lower point density cause the network to treat valid pedestrians as background noise.

Cross-subset transferability is also asymmetric. On Wildtrack, models trained on Subset B generalized to Subset A with 40.3 MODA, whereas models trained on Subset A fell to 15.7 MODA on Subset B. Subset B spans wider camera baselines than Subset A (mean pairwise distance 22.0 m vs. 16.8 m), which plausibly exposes its learned filters to a wider range of viewing geometries and helps them extrapolate. Subset A’s cameras are closer together and, on average, closer to the plaza center (11.2 m vs. 14.6 m).

5.4 Limitations and Future Work

Several practical limitations of the current pipeline inform concrete directions for future research:

5.4.1 Practical Limitations

First, our closed-form median scale alignment relies on an identifiable, planar ground surface ($Z = 0$). In multi-level concourses, pedestrian staircases, or sloped outdoor terrain, this analytical formulation breaks down and requires extension to piecewise-planar or continuous elevation surface estimation. Second, by design, our pipeline consumes raw feedforward foundation point clouds without test-time bundle adjustment. While this keeps the pipeline free of per-scene optimization (we did not

benchmark inference speed), the 30–80 cm cross-view dispersion is left uncompressed before voxelization, which limits localization in dense crowds. Third, while 3D geometry is acquired zero-shot, the downstream BEV detection head still requires supervised training on labeled ground-truth coordinates. Finally, processing video frame by frame omits temporal motion priors, which leaves the detector exposed to transient multi-camera occlusions and single-frame geometric jitter.

5.4.2 Future Research Directions

To reduce camera-rig specificity, future architectures could add differentiable online pose refinement layers or cross-attention modules conditioned on camera extrinsic parameters, so that spatial aggregation weights adapt to new camera layouts. Opening the foundation model to downstream task gradients through parameter-efficient fine-tuning (e.g., LoRA) could also steer the transformer toward geometry better suited to ground-plane pedestrian localization. Finally, adding temporal recurrence or cross-frame attention in BEV space would use motion continuity across frames to reduce single-frame jitter and maintain pedestrian tracks through temporary multi-view occlusions.

In summary, this thesis showed that 3D vision foundation models offer a feedforward alternative that removes the per-scene optimization stage for multi-view pedestrian detection, at a substantial accuracy cost (68.7 MODA against roughly 95 MODA for the best reported supervised DCHM result on Wildtrack), and identified receptive field expansion as a way to absorb part of the foundation-model geometric noise. The findings and failure modes documented here provide design guidance and a baseline for future work on multi-camera perception systems that generalize across rigs.

Bibliography

- [1] Bochkovskii, A., Delaunoy, A., Germain, H., Santos, M., Zhou, Y., Richter, S. R., and Koltun, V. (2024). Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*.
- [2] Brorsson, E., Svensson, L., Bengtsson, K., and Åkesson, K. (2024). MVUDA: Unsupervised domain adaptation for multi-view pedestrian detection. *arXiv preprint arXiv:2412.04117*.
- [3] Chavdarova, T., Baqué, P., Bouquet, S., Maksai, A., Jose, C., Bagautdinov, T., Lettry, L., Fua, P., Van Gool, L., and Fleuret, F. (2018). Wildtrack: A multi-camera HD dataset for dense unscripted pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5030–5039.
- [4] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778.
- [5] Hou, Y. and Zheng, L. (2021). Multiview detection with shadow transformer (and view-coherent data augmentation). In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1673–1682.
- [6] Hou, Y., Zheng, L., and Gould, S. (2020). Multiview detection with feature perspective transformation. In *Computer Vision–ECCV 2020*, pages 1–18. Springer.
- [7] Hu, M., Yin, W., Zhang, C., Cai, Z., Long, X., Chen, H., Wang, K., Yu, G., Shen, C., and Shen, S. (2024). Metric3D v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *arXiv preprint arXiv:2404.15506*.
- [8] Kerbl, B., Kopanas, G., Leimkühler, T., and Drettakis, G. (2023). 3D Gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139:1–139:14.
- [9] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. (2023). Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026.
- [10] Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J., and Beijbom, O. (2019). PointPillars: Fast encoders for object detection from point clouds. In *Proceed-*

- ings of the *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12697–12705.
- [11] Leroy, V., Cabon, Y., and Revaud, J. (2024). Grounding image matching in 3D with Mast3R. *arXiv preprint arXiv:2406.09756*.
 - [12] Lima, J. P., Roberto, R., Figueiredo, L., Simoes, F., and Teichrieb, V. (2021). Generalizable multi-camera 3D pedestrian detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 1232–1240.
 - [13] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2980–2988.
 - [14] Liu, M., Zhu, C., Ren, S., and Yin, X.-C. (2024). Unsupervised multi-view pedestrian detection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 1034–1042.
 - [15] López-Cifuentes, A., Escudero-Viñolo, M., Bescós, J., and Carballeira, P. (2022). Semantic-driven multi-camera pedestrian detection. *Knowledge and Information Systems*, 64(5):1211–1237.
 - [16] Ma, J., Zhang, J., Wang, S., Tong, J., Zheng, L., and Nguyen, C. (2024). Depth-consistent human modeling for multi-view pedestrian detection. *arXiv preprint arXiv:2507.14505*.
 - [17] Ma, J., Tong, J., Wang, S., Zhao, W., Zheng, L., and Nguyen, C. (2021). Voxelized 3D feature aggregation for multiview detection. *arXiv preprint arXiv:2112.03471*.
 - [18] Ma, J., Duan, Z., Zheng, L., and Nguyen, C. (2022). Multiview detection with cardboard human modeling. In *Computer Vision–ECCV 2022*, pages 262–278. Springer.
 - [19] MapAnything Team (2024). MapAnything: Universal 3D scene reconstruction and metric geometry from sparse multi-view imagery. *Meta AI Research*.
 - [20] Pillion, J. and Fidler, S. (2020). Lift, splat, shoot: Encoding images from arbitrary cameras by projecting points into the bird’s-eye-view. In *Computer Vision–ECCV 2020*, pages 194–210. Springer.
 - [21] Qi, C. R., Su, H., Mo, K., and Guibas, L. J. (2017). PointNet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660.
 - [22] Schönberger, J. L. and Frahm, J.-M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3687–3696.
 - [23] Song, L., Wu, J., Yang, M., Zhang, Q., Li, Y., and Yuan, J. (2021). Stacked homography transformations for multi-view pedestrian detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6049–6057.

- [24] Stiefelhagen, R., Bernardin, K., Bowers, R., Garofolo, J., Mostefa, D., and Soundararajan, P. (2007). The CLEAR 2006 evaluation. In *Multimodal Technologies for Perception of Humans*, pages 1–44. Springer.
- [25] Umeyama, S. (1991). Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):376–380.
- [26] Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., and Revaud, J. (2024a). DUS3R: Geometric 3D vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709.
- [27] Wang, P., Chen, P., Yuan, Y., Liu, D., Huang, Z., Hou, X., and Cottrell, G. (2018). Understanding convolution for semantic segmentation. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 1451–1460.
- [28] Yang, L., Kang, B., Huang, Z., Zhao, Z., Xu, X., Feng, J., and Zhao, H. (2024b). Depth anything v2. *arXiv preprint arXiv:2406.09414*.
- [29] Zhu, C. (2019). Multi-camera people detection and tracking. *PhD thesis, Queen Mary University of London*.

A

Supplementary Material

This appendix provides extended mathematical derivations, experimental implementation details, hyperparameter sweeps, and the cross-domain study on domain gap decomposition.

A.1 Domain Gap Quantification for Multi-View Pedestrian Detection

In multi-view pedestrian detection, deploying a model trained on one domain (source) to a different domain (target) usually causes performance degradation, known as the domain gap. Understanding the composition of this gap matters for designing domain adaptation strategies. This section presents the empirical study conducted with the MVFP and MVUDA frameworks [2], which quantifies cross-domain performance degradation across camera networks.

A.1.1 Motivation and Problem Formulation

We hypothesize that the total domain gap in multi-view pedestrian detection can be decomposed into two independent components:

- **Type 1 — Image Space Gap:** Caused by variations in image appearance between domains, including ambient lighting conditions, weather, pedestrian clothing, and background textures.
- **Type 2 — BEV Projection Gap:** Caused by structural differences in camera layouts between domains, including camera elevations, viewing angles, baseline distances, and mutual field-of-view overlap, which warp the spatial distribution of projected features on the BEV plane.

Formally, we define the relative performance drop rate as $\Delta_{rel} = (M_{cross} - M_{src})/M_{src}$, where M_{src} denotes the source-domain baseline performance and M_{cross} denotes the cross-domain performance. The total domain gap $\Delta_{Type1+2}$ combines both factors, while the BEV projection gap can be isolated as $\Delta_{Type2} = \Delta_{Type1+2} - \Delta_{Type1}$.

A.1.2 MVFP Detector and Experimental Design

MVFP Detector. To quantify the combined Type 1+2 gap, we employ the MVFP (Multiview Feature Perspective Transformation) detector from [6, 2]. MVFP

uses a ResNet-18 backbone to extract per-view image features, which are then projected into a 3D voxel grid via camera-extrinsic-driven perspective projection (3D Feature Pulling). Features from all views are fused through element-wise max-pooling (Maximal Fusion) into a unified BEV representation, followed by a convolutional classifier for pedestrian occupancy prediction. Figure A.1 illustrates the overall MVUDA framework that extends MVFP with unsupervised domain adaptation.

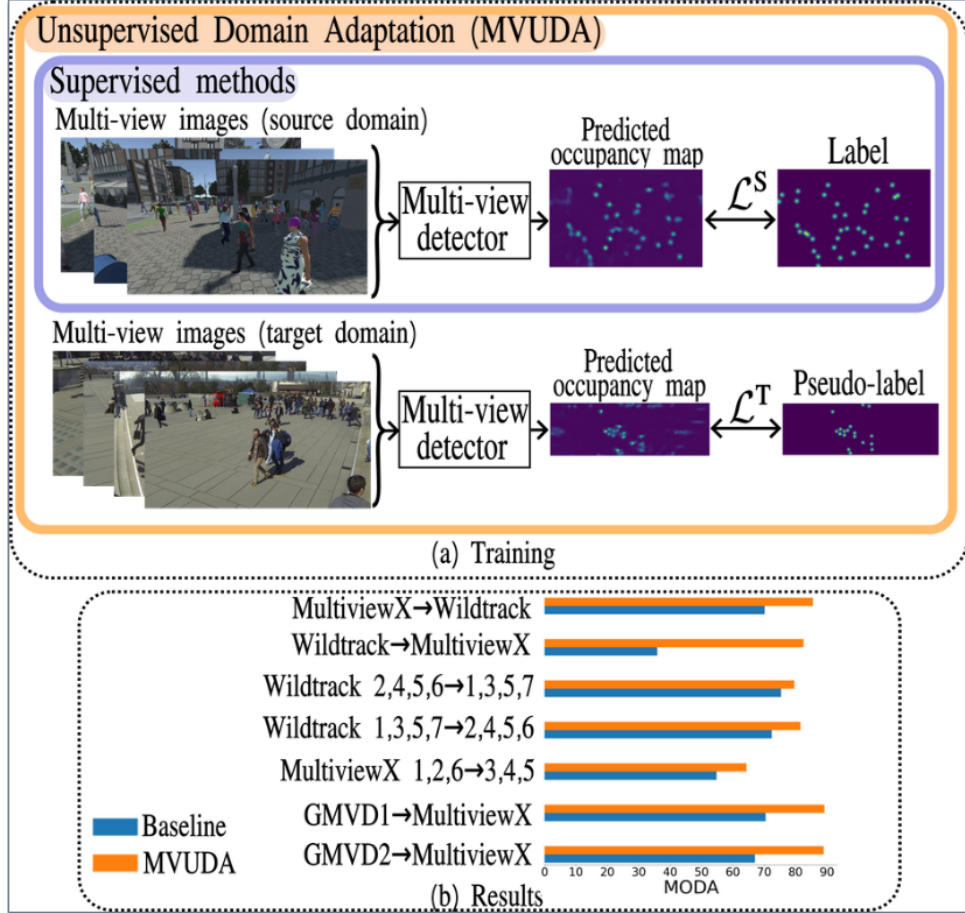


Figure A.1: Overview of the MVUDA framework [2]. (a) Training: A multi-view detector is trained on the source domain with ground-truth labels ($\mathcal{L}^S$) and adapted to the target domain using pseudo-labels ($\mathcal{L}^T$). (b) Results: Comparison of baseline vs. MVUDA performance across multiple cross-domain settings, measured by MODA.

Experimental Design. We design four MVFP experiments to measure the Type 1+2 domain gap between MultiviewX and Wildtrack:

1. **Source baselines:** Train and test on the same domain (MVX → MVX and WT → WT) to establish upper-bound performance.
2. **Cross-domain transfer:** Train on one domain and test on the other (MVX → WT and WT → MVX) to measure total performance degradation.

Both datasets use an 80/20 train/test split. MultiviewX contains 400 frames from 6 cameras covering a 16×25 m² synthetic scene, while Wildtrack contains 2000 frames from 7 cameras covering a 12×36 m² real-world outdoor plaza. All models are trained for 50 epochs with BEV-space MODA as the evaluation metric.

A.1.3 Cross-Domain Experimental Results

Table A.1 presents the MVFP cross-domain detection results. Both source baselines perform well (MVX: 94.0% MODA, WT: 91.9% MODA), consistent with the results reported in the original MVFP literature. Cross-domain transfer causes large performance drops, with an asymmetry between the two transfer directions.

Table A.1: MVFP cross-domain detection results (Type 1+2). Standard model results are reported, with EMA model results in parentheses where they outperform the standard model. All models are trained for 50 epochs on an NVIDIA DGX Spark.

Experiment	Direction	MODA $\uparrow$	MODP $\uparrow$	Prec. $\uparrow$	Recall $\uparrow$	Epoch
MVX source baseline	MVX $\rightarrow$ MVX	94.0	80.7	98.8	95.2	50
WT source baseline	WT $\rightarrow$ WT	91.9	75.4	96.8	95.1	33
MVX $\rightarrow$ WT cross	MVX $\rightarrow$ WT	73.2	74.8	87.0	86.0	43
WT $\rightarrow$ MVX cross	WT $\rightarrow$ MVX	42.3 (44.8)	72.1	84.3	51.9	23 (28)

A.1.4 Type 1+2 Domain Gap Quantification

Table A.2 summarizes the Type 1+2 domain gap quantification. The relative performance drop is computed as $\Delta_{rel} = (MODA_{cross} - MODA_{src})/MODA_{src}$.

Table A.2: Type 1+2 domain gap quantification (MVFP, BEV-space MODA).

Direction	Source MODA	Cross MODA	Abs. Drop	Rel. Drop
MVX $\rightarrow$ WT	94.0%	73.2%	−20.8 pp	−22.1%
WT $\rightarrow$ MVX	91.9%	42.3%	−49.6 pp	−54.0%
Average	—	—	—	−38.1%

A.1.5 Asymmetry Analysis

The two transfer directions are markedly asymmetric: the WT $\rightarrow$ MVX direction has a 54.0% relative MODA drop, more than twice the 22.1% drop in the MVX $\rightarrow$ WT direction. We attribute this asymmetry to several factors:

Camera Layout Discrepancy. MultiviewX uses 6 cameras in a synthetic environment with relatively uniform coverage, while Wildtrack employs 7 cameras in a real-world setting with more diverse viewing angles and varying heights. A model trained on Wildtrack’s camera configuration struggles to generalize to MultiviewX’s distinct spatial arrangement, and vice versa. The WT $\rightarrow$ MVX direction is more difficult because the model must adapt from 7-camera to 6-camera coverage with different projection geometries.

Recall Degradation. The WT→MVX cross-domain recall falls to 51.9% (from 95.1% on the source domain), so nearly half of the pedestrians are missed. This suggests that the BEV projection learned from Wildtrack’s camera layout fails to aggregate features correctly in MultiviewX’s spatial configuration, which leads to systematic detection failures.

Multi-View Aggregation Inherent Robustness. Despite the cross-domain degradation, the MVFP detector retains useful detection capability even in the worst case (42.3% MODA for WT→MVX). The max-pooling fusion across 6–7 views provides some robustness: even when the features of some views are distorted by domain shift, the remaining views can still contribute valid detections. This redundancy distinguishes multi-view methods from single-view approaches, which lack such complementary information.

Figure A.2 illustrates the BEV feature representations produced by the MVFP detector, showing the concatenated world features, final detection map, and view indicator maps that visualize camera coverage across the BEV plane.

A.2 Umeyama Sim3 Closed-Form Alignment Derivation

To determine the spatial discrepancy between foundation-model coordinate predictions and ground-truth camera arrays, we employ the closed-form least-squares estimation formulated by Umeyama [25].

Let $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$ denote the estimated 3D points (camera centers from MapAnything) and $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^n$ denote the corresponding target ground-truth 3D coordinates ($n = C$). We seek scale $s \in \mathbb{R}^+$, rotation $\mathbf{R} \in \text{SO}(3)$, and translation $\mathbf{t} \in \mathbb{R}^3$ minimizing:

$$e^2(s, \mathbf{R}, \mathbf{t}) = \frac{1}{n} \sum_{i=1}^n \|\mathbf{y}_i - (s\mathbf{R}\mathbf{x}_i + \mathbf{t})\|_2^2. \quad (\text{A.1})$$

Step 1: Centering. Compute empirical centroids and centered coordinate vectors:

$$\boldsymbol{\mu}_x = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i, \quad \boldsymbol{\mu}_y = \frac{1}{n} \sum_{i=1}^n \mathbf{y}_i, \quad \tilde{\mathbf{x}}_i = \mathbf{x}_i - \boldsymbol{\mu}_x, \quad \tilde{\mathbf{y}}_i = \mathbf{y}_i - \boldsymbol{\mu}_y. \quad (\text{A.2})$$

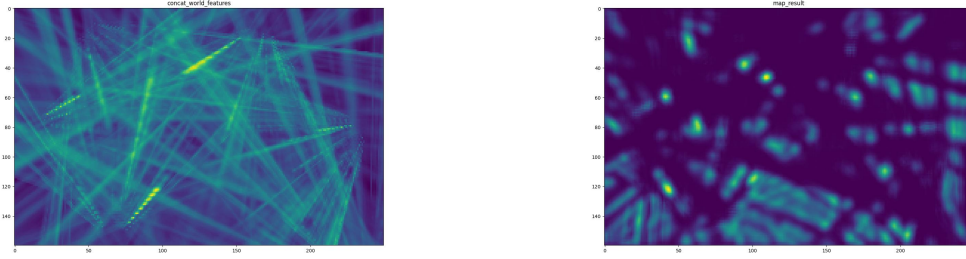
Compute source variance σ_x^2 :

$$\sigma_x^2 = \frac{1}{n} \sum_{i=1}^n \|\tilde{\mathbf{x}}_i\|_2^2. \quad (\text{A.3})$$

Step 2: Cross-Covariance Matrix. Compute the cross-covariance matrix $\boldsymbol{\Sigma}_{xy}$:

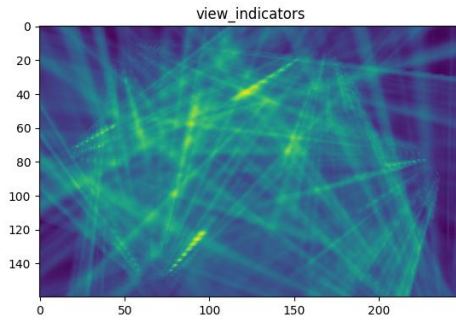
$$\boldsymbol{\Sigma}_{xy} = \frac{1}{n} \sum_{i=1}^n \tilde{\mathbf{y}}_i \tilde{\mathbf{x}}_i^T. \quad (\text{A.4})$$

Perform Singular Value Decomposition (SVD): $\boldsymbol{\Sigma}_{xy} = \mathbf{U}\mathbf{D}\mathbf{V}^T$, where $\mathbf{U}, \mathbf{V} \in \text{O}(3)$ and $\mathbf{D} = \text{diag}(d_1, d_2, d_3)$ with $d_1 \geq d_2 \geq d_3 \geq 0$.



(a) Concatenated BEV features from all camera views, showing the aggregated feature activations across the ground plane.

(b) Final BEV detection map after the map classifier, where bright spots indicate detected pedestrian positions.



(c) View indicator map showing the spatial coverage of each camera in the BEV plane. Brighter regions indicate areas observed by more cameras.

Figure A.2: BEV feature visualizations from the MVFP detector on the Wildtrack dataset. (a) Concatenated multi-view features projected onto the ground plane via 3D Feature Pulling. (b) Predicted occupancy map. (c) Camera coverage indicator map.

Step 3: Optimal Rotation, Scale, and Translation. To enforce proper rotation without reflection ($\det(\mathbf{R}) = 1$):

$$\mathbf{S} = \text{diag}(1, 1, \det(\mathbf{UV}^T)), \quad (\text{A.5})$$

$$\mathbf{R}_{opt} = \mathbf{USV}^T, \quad (\text{A.6})$$

$$s_{opt} = \frac{1}{\sigma_x^2} \text{tr}(\mathbf{DS}), \quad (\text{A.7})$$

$$\mathbf{t}_{opt} = \boldsymbol{\mu}_y - s_{opt} \mathbf{R}_{opt} \boldsymbol{\mu}_x. \quad (\text{A.8})$$

This formulation yields the exact metric residuals reported in Table 4.1 (5.5 cm for calibrated mode, 66.0 cm for uncalibrated mode).

A.3 Hyperparameter Sweeps and Checkpoint Index

Table A.3 lists the model checkpoints, evaluation logs, and configuration flags analyzed in Chapter 4.

Table A.3: Index of model checkpoints, evaluation logs, and execution parameters.

Experiment	Primary Flags	Checkpoint Directory	Evaluation Log File
RGB Baseline	-- roi_mask	experiments_lss/wt_baseline_7cam_v2/	output/lss_wt_baseline_v2_eval_roi.log
Dilated + RGB	-- dilated_bev --use_rgb_features	experiments_lss/wt_dilated/	output/lss_wt_dilated_eval.log
Dilated + ResNet-64	-- dilated_bev (ResNet-64)	experiments_lss/wt_dilated_resnet/	output/lss_wt_dilated_resnet_eval.log
Soft Depth	-- soft_depth_sigma_cm 15 --soft_depth_k 3	experiments_lss/wt_softdepth_resnet/	output/lss_wt_softdepth_resnet_eval.log
Dilated + Soft	Both flags combined	experiments_lss/wt_dilated_soft_resnet/	output/lss_wt_dilated_soft_resnet_eval.log
SAM Compensation	-- gs_subdir gs_optimize_scale_comp	experiments_lss/wt_sam_comp_dilated_resnet/	output/lss_wt_sam_comp_eval.log
HDC Pattern	-- dilated_bev --cfg cfgs/lss_MvDDE_hdc.yaml	experiments_lss/wt_hdc_resnet/	output/lss_wt_hdc_resnet_eval.log
WT Subset A Train	-- source -- cams 0,2,4,6 --dilated_bev	experiments_lss/wt_subsetA_0246_dilated_resnet/	output/lss_wt_cross_subset_strongest_eval.log
WT Subset B Train	-- source -- cams 1,3,4,5 --dilated_bev	experiments_lss/wt_subsetB_1345_dilated_resnet/	output/lss_wt_cross_subset_strongest_eval.log
MXV Baseline	6-camera dilated ResNet-64	experiments_lss/mvx_baseline_dilated_resnet/	output/lss_mvx_baseline_dilated_resnet_eval.log
MXV Subset A Train	-- source -- cams 0,1,5 --dilated_bev	experiments_lss/mvx_subsetA_015_dilated_resnet/	output/lss_mvx_cross_subset_strongest_eval.log
MXV Subset B Train	-- source -- cams 2,3,4 --dilated_bev	experiments_lss/mvx_subsetB_234_dilated_resnet/	output/lss_mvx_cross_subset_strongest_eval.log