

# Bringing Generalization to Deep Multi-View Pedestrian Detection

Jeet Vora<sup>1</sup>, Swetanjal Dutta<sup>1</sup>, Kanishk Jain<sup>1</sup>, Shyamgopal Karthik<sup>2</sup>, Vineet Gandhi<sup>1</sup>

**Abstract**—Multi-view Detection (MVD) is highly effective for occlusion reasoning in a crowded environment. While recent works using deep learning have made significant advances in the field, they have overlooked the generalization aspect, which makes them *impractical for real-world deployment*. The key novelty of our work is to *formalize* three critical forms of generalization and *propose experiments to evaluate them*: generalization with i) a varying number of cameras, ii) varying camera positions, and finally, iii) to new scenes. We find that existing state-of-the-art models show poor generalization by overfitting to a single scene and camera configuration. To address the concerns: (a) we propose a novel Generalized MVD (GMVD) dataset, assimilating diverse scenes with changing day-time, camera configurations, varying number of cameras, and (b) we discuss the properties essential to bring generalization to MVD and propose a barebones model to incorporate them. We perform a comprehensive set of experiments on the WildTrack, MultiViewX and the GMVD datasets to motivate the necessity to evaluate generalization abilities of MVD methods and to demonstrate the efficacy of the proposed approach. The code and the proposed dataset can be found at <https://github.com/jeetv/GMVD>

## I. INTRODUCTION

“Essentially all models are wrong, but some are useful.”

— George E. P. Box

In this work, we pursue the problem of Multi-View Detection (MVD), a mainstream solution for dealing with occlusions, especially when detecting humans/pedestrians in crowded settings. The input to MVD methods is images from multiple calibrated cameras observing the same area from different viewpoints with an overlapping field of view. The predicted output is an occupancy map [1] in the ground plane (bird’s eye view). The solutions of MVD has evolved from classical methods [1], [2], [3], to hybrid approaches [4] to end-to-end trainable deep learning architectures [5]. Expectedly, the current landscape of MVD is dominated by end-to-end trainable deep learning methods [5], [6], [7]. We argue that by *training and testing on homogeneous data*, current deep MVD methods have overlooked critical fundamental concerns, and to render them *useful*, the focus should shift towards their generalization abilities.

Ideally, three forms of generalization abilities are essential for the practical scalability and deployment of MVD methods, which is illustrated in Fig. 1:

- 1) *Varying number of cameras*: The model should adapt to a varying number of cameras (a network trained on six camera views, should work on a setup with five cameras).

<sup>1</sup> CVIT, KCIS, International Institute for Information Technology, Hyderabad, India [jeet.vora@research.iiit.ac.in](mailto:jeet.vora@research.iiit.ac.in)

<sup>2</sup>University of Tübingen, Germany

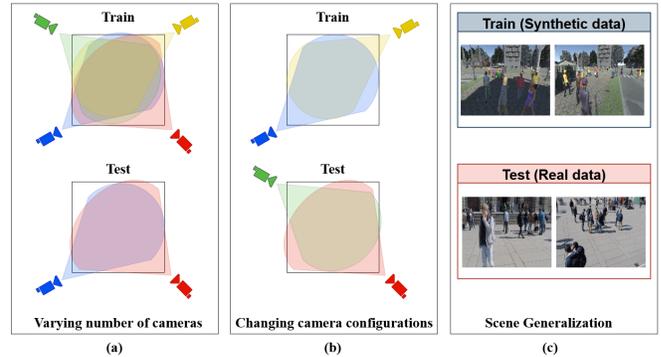


Fig. 1. Three forms of generalization required in MVD: (a) varying number of cameras, (b) different camera configurations, and (c) generalizing to new scenes.

- 2) *Varying configuration*: The model should not overfit to the specific camera configuration. The performance should be similar even with altered camera positions, as long as they span the dedicated area.
- 3) *Varying scenes*: Models trained on one scene should work on another (model trained on a traffic signal should work on a setup inside a university).

Surprisingly, the existing deep learning-based MVD methods are primarily trained and tested with the same camera configuration, on the same scene, using the same number of cameras. Even the environmental conditions (time, weather, etc.) are similar across train and test splits. For instance, the most commonly used Wildtrack dataset [8] includes a 200 second recording from all cameras, where the first 3 minutes are used for training and the rest of the 20 seconds are used for testing. We argue that the current State Of The Art (SOTA) methods are seriously hindered from the deployment perspective. The current models [5], [6], [7] will break if a camera malfunctions. They will need retraining if a camera needs to be added to the setup. Furthermore, our experiments show that the performance significantly drops if the camera positions or the scene is varied. The SOTA models also seem to overfit to the order in which the cameras are sent to the model.

The absence of a diverse dataset is a major shortcoming. The available datasets: Wildtrack (real) and MultiViewX (synthetic), comprise a single short sequence, where initial frames are used for training and later for testing. In Figure 2, we show that the evaluation strategy in both datasets is unreliable and prone to overfitting. To this end, we propose a novel Generalized MVD (GMVD) dataset. Given the privacy concerns, COVID restrictions, hardware setup



Fig. 2. The train and test sets of Wildtrack (first row) and MultiViewX datasets (second row) have significant overlap. We show the last image of the training set (left) and the first image of the test set (right). In both datasets, the appearance of several pedestrians is already seen in the training set. In Wildtrack, there are many static pedestrians as well.

difficulties, the requirement of manual annotations, etc., we believe curating a sizeable synthetic dataset is the right way forward. Henceforth, we use Unity and the GTA game environment to capture the GMVD dataset. It includes about 53 sequences captured in 7 different scenes with significant variations in camera configuration, weather, lighting conditions, pedestrian appearance, etc. The number of cameras also varies across scenes. We use 6 scenes for training and 1 scene for testing. The proposed GMVD dataset sets up a new benchmark for evaluating MVD with generalization. It further allows reserving valuable real-world footages [8] directly for testing.

Furthermore, we suggest a set of design guidelines to ensure practical usability of Deep MVD methods. We demonstrate that permutation invariance, transfer learning, and regularization are vital for generalization. We improve the baseline architecture [5] with appropriate changes and establish SOTA generalization for MVD. We want to emphasize that we do not claim any major architectural novelty, and our work focuses on the barebone baseline architecture. Overall, our work makes the following contributions:

- 1) We conceptualize and emphasize the importance of generalization in MVD and propose a novel GMVD dataset for the same.
- 2) We highlight the shortcomings of the current evaluation methodology and propose novel experimental setup on existing datasets.
- 3) We adapt the baseline architecture to bring generalization to deep MVD. We show that permutation invariance is crucial for MVD and average pooling is one minimal way to achieve it. We propose a novel drop view regularization.
- 4) We back our claims using an extensive set of experiments and ablation studies. We show staggering improvements in scene and configuration generalization, paving the way for a practicable MVD.

## II. RELATED WORK

### A. Classical Methods

Seminal work by Fleuret *et al.* [1] cast MVD as predicting occupancy probabilities over a discrete grid, an idea which has stood the test of time. The classical methods in MVD rely on background subtraction to compute likelihood over a fixed set of anchor boxes derived using scene geometry, project them on the top view and adopt conditional random field (CRF) or mean-field inference for spatial aggregation [1], [2], [3]. The classical methods, however observe a gradual degradation in detection performance with increased crowds, as the background subtraction becomes less effective with increase in crowds and clutter. Some methods do away with background subtraction and rely on handcrafted classifiers [9] instead.

### B. Anchor based MVD

Anchor based MVD methods replace background subtraction with anchor-based deep pedestrian detectors like Faster R-CNN [10], SSD [11] and YOLO [12]. Some of these methods process each view separately [13] and some process them simultaneously [14], [15]. The inaccuracies in the pre-defined anchor boxes [4] limit the performance of these methods. Even if the boxes are correct, locating the exact ground point to project in each 2D bounding box presents a challenge and leads to a significant amount of errors. Moreover, some of the Anchor based methods still rely on operations outside of Convolutional Neural Networks (CNNs), requiring to work out a balance between different potential terms [14].

### C. End-to-end Deep MVD

MVDet [5] is a recent anchor-free approach that aggregates multi-view information by perspective transformation and concatenating multi-view feature map onto the ground plane and then performs large kernel convolution for spatial aggregation. It overcomes limitations of manual tuning of CRF potentials, reliance on pre-defined 3D anchor boxes and projection errors from monocular detectors. It aggregates projected features from a ResNet [16] backbone using three convolutional layers to predict the final occupancy map. MVDet achieves notable improvement over the preceding anchor based methods (over 14% improvement on the WildTrack dataset [8]). The idea from [5] was further enhanced by using deformable transformers [17] to improve the feature aggregation in MVDeTr [6]. More recently, SHOT [7] introduced a combination of homographies at multiple heights to improve the quality of the projections.

## III. PROPOSED DATASET

We propose a new MVD dataset incorporating the three forms of generalization discussed above (Figure 1). Some example frames from the proposed Generalized Multi-View Detection (GMVD) dataset are illustrated in Figure 3. The GMVD dataset contains diverse non-overlapping scenes within and across training and test sets. In contrast, the existing MVD datasets Wildtrack and MultiViewX include noticeable overlap across train and test splits (single scene,

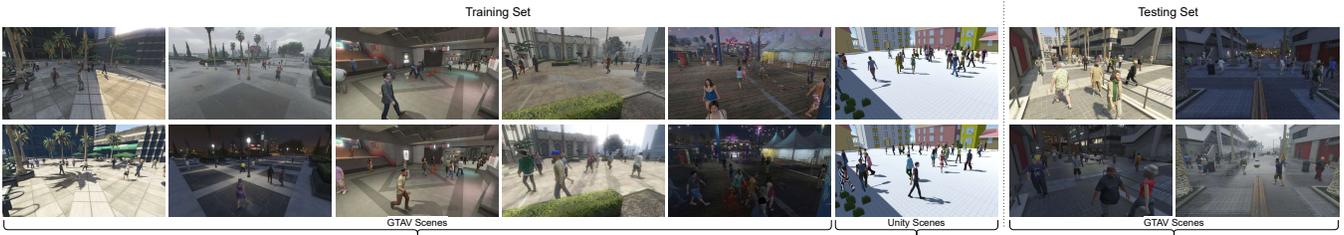


Fig. 3. The proposed GMVD Dataset includes seven scenes. Each column illustrates frames from one of the views from two different sequences of the same scene. The first six scenes are used for training and the last scene with two configurations are reserved for testing. Additionally, there are noticeable lighting and weather variations within each scene.

TABLE I

DATASET STATISTICS FOR VARIOUS MVD DATASETS. OUR PROPOSED GMVD DATASET IS THE LARGEST AND MOST DIVERSE DATASET ON A VARIETY OF METRICS. AVG. COVERAGE REFERS TO THE AVERAGE NUMBER OF CAMERAS THAT COVER EACH POINT ON THE GROUND PLANE.

Dataset	Track Labels	IDs	# Scenes	# Training Frames	# Testing Frames	# Cameras	# Sequences	Avg. Coverage
WildTrack	✓	313	1	360	40	7	1	3.74
MultiViewX	✓	350	1	360	40	6	1	4.41
GMVD (Ours)	✓	2800	7	4983	1012	3, 5, 6, 7, 8	53	2.76 - 6.4

pedestrians appearance, and location), encouraging existing MVD methods to overfit the dataset-specific aspects and thus hindering their practicality. The GMVD dataset, by its design, prevents overfitting from happening by keeping a clear separation in train and test splits.

Capturing a real-world MVD dataset is difficult, primarily because of privacy concerns. The COVID restrictions also restrict crowded human capture. Moreover, such a dataset requires significant manual annotation effort. Consequently, we curate the GMVD dataset using synthetic environments. The GMVD dataset is curated using Grand theft Auto V (GTAV) and Unity Game Engine. We employ two different environments to avoid overfitting to a single synthetic data generation source. This reasoning is aligned with recent works [18], [19] which utilize multi-source datasets to improve generalization performance. The GMVD dataset includes seven distinct scenes, one indoor (subway) and six outdoors. One of the scenes are reserved for the test split. We vary the number of total cameras in each scene and provide different camera configurations within a scene.

Additional salient features of GMVD include daytime variations (morning, afternoon, evening, night) and weather variations (sunny, cloudy, rainy, snowy). We generate multiple short sequences for each scene while randomly varying the daytime and the weather. The generation of multiple random sequences ensures diversity, as different pedestrians (with different clothing and appearance) are picked in each case. The dataset also includes significant variations in lighting conditions. Local illumination sources come into play due to the presence of indoor and night scenes. We compare our dataset with the existing ones in Table I. Avg. Coverage represents the average amount of cameras observing each location. For GMVD, avg. coverage varies from 2.76-6.4 cameras depending on the scene. In addition to the discussed variations, GMVD is advantageous due to the dataset size, especially in terms of the total number of individual sequences.

Thereby, we propose the GMVD dataset as a new benchmark for MVD. We further encourage future methods to train on the GMVD dataset and test their performance on sparsely available, difficult to capture real-world datasets like WildTrack .

**Dataset Generation:** We used Script Hook V [20] library to interface with the GTAV environment. For each scene, camera positioning and orientation were determined manually so as to increase the camera coverage. All the cameras were positioned above the humans’ average height. Due to hardware limitation, it is commonplace to have a small synchronization delay in real-world multi-camera setups. To emulate such realistic scenario, we induce a small synchronization error (20-100 ms) between different camera views [21]. A ground plane was defined for each location, partially overlapping with each camera’s field of view. Only pedestrians inside the ground plane were considered for multi-view detection. We relied on the GTA’s navigational AI engine to avoid collision and to obtain realistic pedestrian behavior.

In Unity environment, the scene is manually curated by putting together 3d models of street, buildings and other props. We used the PersonX [22] 3d human models to create the pedestrians. To avoid collision errors (which are present in MultiViewX dataset), pedestrians were spawned at random locations within the region of interest, for every frame. Since both the environments are synthetic, the 3D-2D correspondences were directly available from the game engines. We use similar procedure as [5] for camera calibration.

**Track Labels:** Our work focuses on a comprehensive analysis of the problem of Multi-View Detection. However, the proposed dataset can also be useful for the task of multi-view pedestrian tracking. To this end, for the sequences generated from the GTAV environment, we collect the track labels while capturing the data. While we do not use track labels in this work, we provide them with the dataset, which will be beneficial for the community in the future. We provide a total

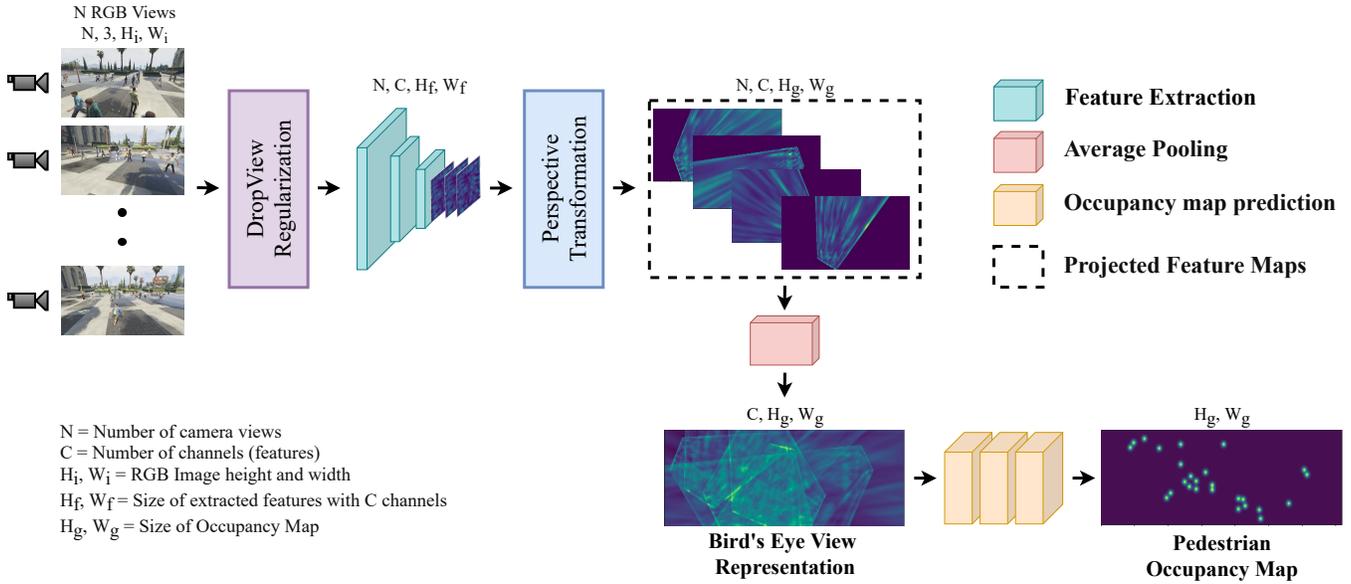


Fig. 4. Our proposed architecture: ResNet features are extracted from the input views, which are then projected to the top view. Following this, the projected features across views are pooled and then the final occupancy map is predicted. The use of average pooling across views is crucial in ensuring that our proposed architecture can work for an arbitrary number of views.

of 125000 frames with track labels. The GTAV frames for the GMVD dataset are regularly sampled from these densely annotated sequences.

#### IV. PROPOSED METHOD

We propose an anchor free deep MVD method along the lines of [5], [6], [7] specifically tailored to improve the generalization abilities by modifying the training objective and making use of an average pooling strategy on the projected feature maps. The overall architecture is shown in Fig. 4. The input to our pipeline are multiple calibrated RGB cameras with overlapping fields of view, and the expected output is the occupancy map for pedestrians.

##### A. Feature Extraction and Perspective Transformation

**Feature Extractor:** We use a ResNet18 [16] backbone as a feature extractor replacing last three strided convolutions with dilated convolutions to have a high spatial resolution of the feature maps. Given  $N$  camera views of image size  $(3, H_i, W_i)$ , where  $H_i$  and  $W_i$  corresponds to height and width of images,  $C$ -channel features are extracted for  $N$  camera views which corresponds to size  $(N, C, H_f, W_f)$ , where  $H_f$  and  $W_f$  represents the height and width of the extracted features.

**Perspective Transformation:** The extracted features from the feature extractor are then projected onto the ground plane using a perspective transformation, where  $(H_g, W_g)$  corresponds to the height and width of the ground plane grid. Considering the calibrated cameras,  $K$  represents the intrinsic camera parameters and  $[R|t]$  represents the extrinsic camera parameters ( $R$  is the rotation matrix and  $t$  is the translation vector).

In the world coordinate system, the ground plane corresponds to  $Z = 0$ , i.e.,  $W = (X, Y, 0, 1)^T$ . A pixel of an image

$I = (x, y)^T$  is transformed to the ground plane as follows:

$$I = s \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = K[R|t] \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = P \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (1)$$

where  $s$  is a scaling factor and  $P$  is a perspective transformation matrix.

##### B. Spatial Aggregation

**Average Pooling:** We first project the ResNet feature maps from each viewpoint on to the bird's eye view using the perspective transformation to obtain the projected feature maps  $f m_i$  (where,  $i = 1, 2, \dots, N$ ). Following this, we average pool the projected feature maps  $f m_i$  to obtain the final bird's eye view feature representation  $F$  of size  $(C, H_g, W_g)$ , which is written as,

$$F = \frac{\sum_{i=1}^N f m_i}{N}. \quad (2)$$

While there can be many other alternatives to average pooling, we opt for this solution, primarily because it is permutation-invariant. Unlike MVDet, where the camera views ideally need to be input in the same order as training during inference, our proposed solution can accept arbitrary number of views in an arbitrary order. Furthermore, the average pooling solution is free from any learnable parameters which ensures that there is no overfitting introduced due to this operation. The projected feature maps for  $N$  cameras of size  $(N, C, H_g, W_g)$  after average pooling, reduces to  $(C, H_g, W_g)$ , thus removing the dependency over the number of camera views thereby allowing the model to take an arbitrary number of views as input.

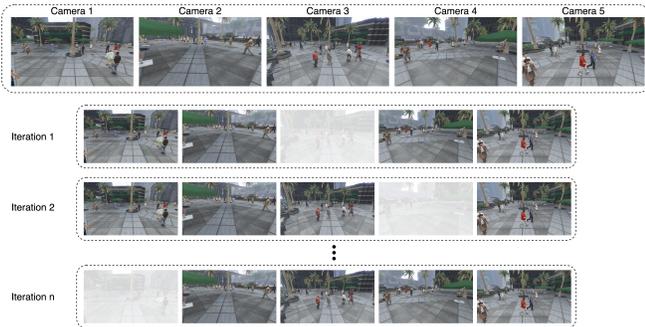


Fig. 5. An illustration of our proposed DropView regularization

**DropView Regularization:** Inspired by Dropout [23] as well as work on self-supervised learning which drops color channels to prevent the model from memorization [24], [25], we propose the DropView regularization technique. For each sample, we randomly select one view to discard during training iterations, as illustrated in Fig 5. The occupancy map prediction step is done with all the remaining views. We provide a detailed analysis of the effect of this regularization strategy in our experiments.

**Occupancy Map Prediction:** Similar to MVDet [5], we use 3 dilated convolutional layers to predict the occupancy map of size  $(H_g, W_g)$ .

### C. Loss Function

The loss function compares the output probabilistic occupancy map ( $p$ ) with the ground-truth ( $g$ ). Inspired by the work on saliency estimation in images and videos [26], [27], [28], we use the combination of Kullback–Leibler Divergence (KLDiv) and Pearson Cross-Correlation (CC) metrics as a loss function. The final loss function can be written as:

$$L(p, g) = \frac{\sigma(p, g)}{\sigma(p) \times \sigma(g)} - \sum_i g_i \log \left( \frac{g_i}{p_i} \right), \quad (3)$$

where  $\sigma(p, g)$  is the covariance of  $p$  and  $g$ ,  $\sigma(p)$  is the standard deviation of  $p$  and  $\sigma(g)$  is the standard deviation of  $g$ . The loss function was selected empirically using the scene generalization experiment, i.e. training on MultiViewX and testing on WildTrack, where using KLDiv+CC gave best results (compared with MSE, CC or KLDiv alone).

## V. EXPERIMENTS

### A. Experimental setup

**Datasets:** In addition to our proposed GMVD dataset, we use the WildTrack and MultiViewX datasets. The *WildTrack* dataset consists of 7 static calibrated cameras with overlapping fields of view, covering an area of  $12 \times 36 m^2$ . The dataset comprises a single 200 second sequence annotated at 2 fps. The image resolution is  $1080 \times 1920$  pixels. The ground plane grid is discretized into a  $480 \times 1440$  grid, where each grid cell is  $2.5 cm$  square. On average, the dataset captures 23.8 persons per frame. The *MultiViewX* dataset is a synthetic dataset which has similar configurations

as the *WildTrack* dataset. However, it consists of 6 static calibrated cameras with overlapping fields of view and 400 synchronized frames of resolution  $1080 \times 1920$  annotated at 2 fps for ground-truth covering an area of  $16 \times 25 m^2$ . The ground plane grid is discretized into a  $640 \times 1000$  grid, where each grid cell is  $2.5 cm$  square. On average, the dataset captures 40 persons per frame. For both datasets, we use the first 90% frames in training and the last 10% frames for testing, as done in previous work [5], [8].

**Evaluation metrics:** We use the standard evaluation metrics proposed in [8]. *Multiple Object Detection Accuracy* (MODA) is the primary performance indicator that accounts for normalized missed detections and false positives, i.e., it considers both false negatives and false positives. *Multiple Object Detection Precision* (MODP) assesses the localization precision [31]. *Precision* and *Recall* is calculated by  $Precision = TP/(TP+FP)$  and  $Recall = TP/(TP+FN)$  respectively; where TP, FP and FN are True Positives, False Positives, False Negatives. A threshold of 0.5 meters is used to determine the true positives.

**State of the Art comparisons:** We compare against nine different methods. The set includes one monocular object detection baseline (referred to as RCNN clustering [13]); a classical probabilistic occupancy map method [1]; four anchor based methods [30], [14], [15], [29] and three recent end-to-end trainable deep MVD approaches [5], [6], [7]. For generalization experiments, we only compare against the recent state-of-the-art methods MVDet [5], MVDetr [6] and SHOT [7].

### B. Implementation Details

Down sampled images of  $720 \times 1,280$  pixels serve as an input to the model. The feature extracted from ResNet-18 has  $C = 512$  channel features, which is bilinearly interpolated to get the shape of  $270 \times 480$ . These  $(N, C = 512, H_f = 270, W_f = 480)$  extracted features are projected onto top view to obtain  $(N, 512, H_g, W_g)$  sized features for  $N$  viewpoints, which are average pooled to obtain the ground plane grid shape of  $(512, H_g, W_g)$ .  $H_g$  and  $W_g$  vary from scene-to-scene, depending on the area of ground plane.

The spatial aggregation has three layers of dilated convolution with a  $3 \times 3$  kernel size and dilation factor of 1, 2, and 4. Training is done for ten epochs with early stopping; we set batch size as 1, SGD optimizer with momentum = 0.9 has been used with one-cycle learning rate scheduler. A probability of  $\tau$  or more on the occupancy grid is considered a detection. For GMVD experiments,  $\tau$  is determined using MultiViewX as a validation set, and for other experiments, we use  $\tau = 0.4$  in alignment with the previous works. Non-Maximal Suppression (NMS) is applied with a spatial resolution of 0.5m. All training and testing have been performed on a single Nvidia GTX 1080 Ti GPU. Unless specifically mentioned, we always use pre-trained ImageNet [32] weights while training our proposed model.

TABLE II

COMPARISON AGAINST THE STATE-OF-THE-ART METHODS. OUR METHOD REFERS TO THE PROPOSED MODEL IN SECTION IV. WE MADE FIVE RUNS FOR SOME OF THE EXPERIMENTS AND THE VARIANCES ARE PRESENTED IN THE BRACKET.

Method	ImageNet (pre-train)	WildTrack				MultiViewX			
		MODA	MODP	Prec	Recall	MODA	MODP	Prec	Recall
RCNN Clustering [13]	×	11.3	18.4	68.0	43.0	18.7	46.4	63.5	43.9
POM-CNN [1]	×	23.2	30.5	75.0	55.0	-	-	-	-
Lopez-Cifuentes <i>et al.</i> [29]	×	39.0	55.0	-	-	-	-	-	-
Lima <i>et al.</i> [30]	×	56.9	67.3	80.8	74.6	-	-	-	-
DeepMCD [15]	×	67.8	64.2	85.0	82.0	70.0	73.0	85.7	83.3
Deep-Occlusion [14]	×	74.1	53.8	95.0	80.0	75.2	54.7	97.8	80.2
MVDet [5]	×	88.2	75.7	94.7	93.6	83.9	79.6	96.8	86.7
MVDeTr [6]	✓	91.5	82.1	97.4	94.0	93.7	91.3	99.5	94.2
SHOT [7]	×	90.2	76.5	96.1	94.0	88.3	82.0	96.6	91.5
Ours	×	87.2(±0.6)	74.5(±0.4)	93.8(±1.6)	93.4(±1.8)	78.6(±0.9)	78.1(±0.4)	96.8(±0.5)	81.3(±0.9)
Ours	✓	85.4(±0.4)	76.7(±0.2)	95.2(±0.4)	89.9(±0.8)	86.9(±0.2)	79.8(±0.1)	97.2(±0.2)	89.6(±0.2)
Ours (DropView)	✓	86.7(±0.4)	76.2(±0.2)	95.1(±0.3)	91.4(±0.6)	88.2(±0.1)	79.9(±0.0)	96.8(±0.2)	91.2(±0.1)

TABLE III

RESULTS FOR EVALUATING WITH A VARYING NUMBER OF CAMERAS. THE MODEL IS TRAINED ON ALL 7 CAMERAS ON WILDTRACK, AND IS TESTED ON 2 DIFFERENT SETS OF 4 CAMERAS EACH.

Method	Inference on {1,3,5,7}				Inference on {2,4,5,6}			
	MODA	MODP	Prec	Recall	MODA	MODP	Prec	Recall
MVDet	38.9	71.5	<b>93.8</b>	41.6	16.2	47.6	80.3	21.4
MVDeTr	55.8	<b>76.7</b>	80.8	73.2	34.6	69.2	68.6	63.8
SHOT	66.6	75.1	91.0	73.9	46.3	67.8	88.2	53.5
Ours	76.5	74.0	91.7	84.0	<b>79.3</b>	71.4	<b>91.1</b>	87.9
Ours (DropView)	<b>77.0</b>	74.5	90.3	<b>86.2</b>	79.2	72.5	88.6	<b>90.9</b>

### C. Results

Like prior works, we evaluate our approach on the WildTrack and MultiViewX datasets in Table II. We find that our proposed models attains satisfactory performance on the test sets of both WildTrack (best MODA score of 87.2) and MultiViewX (best MODA score of 88.2). This is slightly worse than the recently proposed methods [6], [7], but is far superior to the performance of the classical and the anchor-based MVD methods. However, we would like to highlight that the traditional evaluation protocol is highly misleading since the train and test sets have significant overlap, thereby encouraging overfitting. Therefore, we emphasize the evaluation across a varying number of cameras, changing camera configurations, and on new scenes.

**Generalization to Varying Number of Cameras:** An interesting scenario that can potentially occur in practical scenarios is the loss of some camera feeds due to various issues. In this case, a model trained with 7 cameras, may need to be able to perform inference with just 4 cameras. To simulate this setting, we train all the models (MVDet, MVDeTr, SHOT and Ours) on all 7 cameras and test them on 2 different sets of 4 cameras ( $\{1, 3, 5, 7\}, \{2, 4, 5, 6\}$ ) in Table III. Our proposed model is able to naturally work in this setting without any issues. For MVDet, MVDeTr, and SHOT, we randomly duplicate 3 of these views to ensure that 7 views are available. We observe that the performance of MVDet, MVDeTr, and SHOT degrades drastically when evaluated in this setting. When trained with the DropView

TABLE IV

SCENE GENERALIZATION : EVALUATION OF OUR METHOD WHILE TRAINING ON SYNTHETIC DATASET (MULTIVIEWX) AND TESTING ON REAL DATASET (WILDTRACK). CAMERA 7 OF THE WILDTRACK DATASET WAS DISCARDED FOR THE EXPERIMENTS IN THE FIRST FIVE ROWS.

Method	Inference on total cameras	ImageNet (pre-train)	MODA	MODP	Prec	Recall
MVDet	6	×	17.0	65.8	60.5	48.8
MVDeTr	6	✓	50.2	69.1	74.0	77.3
SHOT	6	×	53.6	72.0	75.2	79.8
Ours	6	✓	60.1	72.1	75.6	<b>88.7</b>
Ours (DropView)	6	✓	66.1	72.2	82.0	84.7
Ours	7	✓	69.4	72.96	<b>83.7</b>	86.14
Ours (DropView)	7	✓	<b>70.7</b>	<b>73.8</b>	<b>89.1</b>	80.6

regularization, our proposed model outperforms these methods by a huge margin (MODA of 77.0 vs 66.6 and 79.2 vs 46.3). This experiment clearly illustrates the need for the architectures to automatically work with an arbitrary number of views. Furthermore, since MVDet, MVDeTr, and SHOT learn a separate spatial aggregation module for each view, the spatial aggregation module overfits to the order of input cameras (indicated by the significant performance variations across the two sets). Future works should ensure that the model has permutation invariance to the order of input views in addition to working with an arbitrary number of views.

**Generalization to New Camera Configurations:** Another practical scenario that we explore is when the camera po-

TABLE V  
EXPERIMENTS ON THE WILDTRACK DATASET WITH CHANGING CAMERA CONFIGURATIONS

		Inference on {2,4,5,6}				Inference on {1,3,5,7}				
		Method	MODA	MODP	Prec	Recall	MODA	MODP	Prec	Recall
Trained on camera set	{2,4,5,6}	MVDet	<b>85.2</b>	72.2	92.6	<b>92.5</b>	43.2	68.2	<b>94.6</b>	45.8
		MVDeTr	75.4	<b>79.5</b>	<b>96.9</b>	77.9	41.7	<b>73.7</b>	92	45.7
		SHOT	81.9	74.1	94.1	87.4	51.4	72.5	94.4	54.6
		Ours	81.8	73.5	93.5	87.9	66.5	71.4	94.3	70.8
		Ours (DropView)	84	72.9	92.4	91.6	<b>75.1</b>	71.1	94.3	<b>79.9</b>
	{1,3,5,7}	MVDet	27.8	<b>68.7</b>	<b>90.8</b>	31.0	78.2	73.6	89.5	<b>88.6</b>
		MVDeTr	5.6	65.5	62.4	14.0	72.5	<b>78.9</b>	95	76.5
		SHOT	15.3	62.9	89.2	17.4	79.7	76.4	<b>95.7</b>	83.5
		Ours	52.4	67.4	81	68.5	76.4	74.6	91.5	84.1
		Ours (DropView)	<b>62.6</b>	67.4	86.7	<b>73.9</b>	<b>80.8</b>	74.0	94.2	86

TABLE VI  
CHANGING CONFIGURATION AND SCENE GENERALIZATION  
EXPERIMENT ON THE SETTING INTRODUCED IN [7]

Method	MODA	MODP	Prec	Recall
MVDet	33.0	76.5	64.5	73.4
MVDeTr	56.5	70.8	85.0	68.6
SHOT	49.1	<b>77.0</b>	73.3	<b>77.1</b>
Ours	57.8	76.5	88.7	66.3
Ours (DropView)	<b>66.1</b>	75.8	<b>89.3</b>	75.2

TABLE VII  
EVALUATION WHEN TRAINED ON GMVD TRAINING SET: FIRST ROW  
SHOWS THE RESULT ON GMVD TEST SET AND SECOND ROW IS WHEN  
TESTED ON WILDTRACK DATASET.

Inference on	MODA	MODP	Prec	Recall
GMVD	68.2	76.3	91.5	75.5
WildTrack	80.1	75.6	90.9	89.1

sitions are varied between the train and test sets. We train all the models on two sets of camera views and then test the trained models on both sets. The results are provided in Table V. When the models are evaluated on the same camera configuration, all the models have satisfactory performance. However, when evaluated on the different camera configuration, MVDet, MVDeTr, and SHOT see a huge degradation in performance. Our model is fairly robust to the changing camera configuration. Especially when trained with DropView regularization, the resulting model outperforms all other models by over 20 percentage points.

**Scene Generalization:** Finally, an important concern with the practical utility of MVD methods is that since real-world data is scarce, a trained model should be able to generalize to new scenes. We first evaluate the scene generalization abilities of the MVD methods by training them on MultiViewX and evaluating them on WildTrack in Table IV. Our proposed model is able to utilize the extra camera present in the WildTrack dataset and achieves a MODA score of 70.7. This further highlights the benefits of an architecture that works with arbitrary number of views, since the performance during inference can be enhanced by adding more view. However,

even without the additional view, our model achieves a MODA score of 66.1, which is much higher than SHOT which only achieves a MODA score of 53.6.

In addition to this, we perform the scene generalization experiment proposed in [7] where the MultiViewX scene is split into two halves, and each half is covered using 3 cameras each. In this setting as well (Table VI), our proposed approach with DropView regularization has a MODA score of 66.1, which is significantly higher than both SHOT (49.1) and MVDeTr (56.5).

**GMVD Benchmark:** Having shown that our proposed model is capable of comprehensive generalization abilities, we benchmark our proposed approach on the GMVD dataset (Table VII). We train our model on the training set of the GMVD dataset and use MultiViewX dataset for validation. Since each sequence in the training set has a different number of cameras, *none* of the existing methods can be adapted to this setting, since they can be trained only on a *fixed* set of cameras. When evaluated on WildTrack, our model is able to achieve a MODA score of 80.1, which is a significant improvement over the results from training on MultiViewX. Notably, this shows that training on our synthetic dataset, we can nearly attain the same performance as training on WildTrack itself. When evaluated on GMVD test set, our model achieves a MODA score of 68.2. The results empirically suggest the difficulty of the GMVD test set, compared to WildTrack and MultiViewX, resulting from a distinct train-test split and the presence of extensive variations. We believe that our dataset can serve two important purposes. The first is as a diverse, synthetic dataset from which a model can be adapted to real-world data. The second is that the GMVD dataset itself can be a challenging benchmark to evaluate the generalization capabilities of MVD methods. In this setting, MultiViewX being used for validation is ideal, since this ensures that no information from the test set is leaked during training.

## VI. DISCUSSION AND FUTURE WORK

The biggest limitation in the field of Multi-View Detection is that real-world capture of data is extremely challenging due to the difficulty in collecting a dataset with people in addition to the challenges involved in the hardware setup

and annotations. The absence of a large, diverse benchmark significantly hampers the progress of this topic. Therefore, the existing WildTrack dataset is extremely valuable for the community. However, due to its limited size and variety, it is not suitable for training and should only be used to evaluate the generalization abilities of the models. In this regard, we hope that our proposed dataset and our barebone model serves as a useful tool in bridging the gap between the theory and real-world application of MVD methods. In our work, we have not explored the use of unsupervised domain adaptation techniques to bridge the gap between the feature distributions of the synthetic and real datasets and the direction is left for exploration in the future work.

## VII. CONCLUSION

We find the current Multi-View Detection setup severely limited and encouraging models to overfit the training configuration. Therefore, we conceptualize and propose novel experimental setups to evaluate the generalization capabilities of MVD models in a more practical setting. We find the state-of-the-art models to have poor generalization capabilities on our proposed setups. To alleviate this issue, we introduce changes to the feature aggregation strategy, loss function, as well as a novel regularization strategy. With the help of comprehensive experiments, we demonstrate the benefits of our proposed architecture. In addition to this, we propose a diverse, synthetic, but realistic dataset which can be used both as an evaluation benchmark, as well as a training dataset for various MVD methods. Overall, we hope our work plays a crucial role in steering the community towards more practical Multi-View Detection solutions.

## REFERENCES

- [1] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multicamera people tracking with a probabilistic occupancy map," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 267–282, 2008.
- [2] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 9, pp. 1806–1819, 2011.
- [3] A. Alahi, L. Jacques, Y. Boursier, and P. Vandergheynst, "Sparsity driven people localization with a heterogeneous network of cameras," *Journal of Mathematical Imaging and Vision*, vol. 41, no. 1, pp. 39–58, 2011.
- [4] T. Kong, F. Sun, H. Liu, Y. Jiang, L. Li, and J. Shi, "Foveabox: Beyond anchor-based object detection," *IEEE Transactions on Image Processing*, vol. 29, pp. 7389–7398, 2020.
- [5] Y. Hou, L. Zheng, and S. Gould, "Multiview detection with feature perspective transformation," in *ECCV*, 2020.
- [6] Y. Hou and L. Zheng, "Multiview detection with shadow transformer (and view-coherent data augmentation)," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 1673–1682.
- [7] L. Song, J. Wu, M. Yang, Q. Zhang, Y. Li, and J. Yuan, "Stacked homography transformations for multi-view pedestrian detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6049–6057.
- [8] T. Chavdarova, P. Baqué, S. Bouquet, A. Maksai, C. Jose, T. M. Bagautdinov, L. Letry, P. Fua, L. Gool, and F. Fleuret, "Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5030–5039, 2018.
- [9] G. Roig, X. Boix, H. B. Shitrit, and P. Fua, "Conditional random fields for multi-camera object detection," *2011 International Conference on Computer Vision*, pp. 563–570, 2011.

- [10] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 1137–1149, 2015.
- [11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C.-Y. Fu, and A. Berg, "Ssd: Single shot multibox detector," in *ECCV*, 2016.
- [12] J. Redmon, S. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2016.
- [13] Y. Xu, X. Liu, Y. Liu, and S.-C. Zhu, "Multi-view people tracking via hierarchical trajectory composition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4256–4265.
- [14] P. Baqué, F. Fleuret, and P. Fua, "Deep occlusion reasoning for multi-camera multi-target detection," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 271–279, 2017.
- [15] T. Chavdarova and F. Fleuret, "Deep multi-camera people detection," *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 848–853, 2017.
- [16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.
- [17] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.
- [18] R. Gong, D. Dai, Y. Chen, W. Li, and L. Van Gool, "mdalu: Multi-source domain adaptation and label unification with partial datasets," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 8876–8885.
- [19] Y. Zhao, Z. Zhong, F. Yang, Z. Luo, Y. Lin, S. Li, and N. Sebe, "Learning to generalize unseen domains via memory-based multi-source meta-learning for person re-identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 6277–6286.
- [20] A. Blade, "Script Hook V," <http://www.dev-c.com/gtav/scripthookv/>, 2008, [Online; accessed 19-July-2008].
- [21] P. Kohl, A. Specker, A. Schumann, and J. Beyerer, "The mta dataset for multi-target multi-camera pedestrian tracking by weighted distance aggregation," in *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.
- [22] X. Sun and L. Zheng, "Dissecting person re-identification from the viewpoint of viewpoint," in *CVPR*, 2019.
- [23] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [24] S. Jenni and P. Favaro, "Self-supervised feature learning by learning to spot artifacts," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2733–2742.
- [25] Z. Lai and W. Xie, "Self-supervised learning for video correspondence flow," *arXiv preprint arXiv:1905.00875*, 2019.
- [26] Z. Bylinskii, T. Judd, A. Oliva, A. Torralba, and F. Durand, "What do different evaluation metrics tell us about saliency models?" *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 3, pp. 740–757, 2018.
- [27] N. Reddy, S. Jain, P. Yarlagadda, and V. Gandhi, "Tidying deep saliency prediction architectures," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10 241–10 247, 2020.
- [28] S. Jain, P. Yarlagadda, S. Jyoti, S. Karthik, R. Subramanian, and V. Gandhi, "Vinet: Pushing the limits of visual modality for audio-visual saliency prediction," *arXiv preprint arXiv:2012.06170*, 2020.
- [29] A. López-Cifuentes, M. Escudero-Viñolo, J. Bescós, and P. Carballeira, "Semantic driven multi-camera pedestrian detection," *ArXiv*, vol. abs/1812.10779, 2018.
- [30] J. Lima, R. Roberto, L. Figueiredo, F. Simões, and V. Teichrieb, "Generalizable multi-camera 3d pedestrian detection," *ArXiv*, vol. abs/2104.05813, 2021.
- [31] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. S. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, and J. Zhang, "Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 319–336, 2009.
- [32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

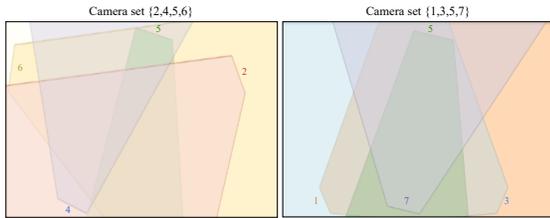


Fig. 6. Camera splits of WildTrack dataset for changing camera configuration experiment.

## APPENDIX

Method	ImageNet (pre-train)	MODA	MODP	Prec	Recall
MSE	✓	57.3( $\pm 0.2$ )	72.6( $\pm 0.0$ )	75.6( $\pm 0.1$ )	84.5( $\pm 0.05$ )
CC	✓	55.5( $\pm 5.5$ )	<b>74.2</b> ( $\pm 0.4$ )	72.1( $\pm 4.4$ )	<b>89.5</b> ( $\pm 2.6$ )
KL	✓	62.5( $\pm 0.1$ )	73.4( $\pm 0.04$ )	<b>89.1</b> ( $\pm 0.0$ )	71.3( $\pm 0.0$ )
KLCC	✓	<b>69.4</b> ( $\pm 0.6$ )	72.96( $\pm 0.2$ )	83.74( $\pm 0.5$ )	86.14( $\pm 0.3$ )

TABLE VIII

CHOICE OF LOSS FUNCTION: WE PRESENT AN ABLATION STUDY FOR OUR PROPOSED METHOD ON THE SCENE GENERALIZATION EXPERIMENT. OVERALL, THE MODEL TRAINED WITH BOTH KL-DIVERGENCE AND CROSS-CORRELATION ACHIEVES THE BEST PERFORMANCE.

We ablate the choice of the loss function in Table VIII for the scene generalization experiment. We consider the Mean Squared Error (MSE), KL-Divergence(KL), Pearson Cross-Correlation (CC), as well as our chosen loss function (KL+CC). We find that the combination of KL-Divergence and Pearson Cross-Correlation achieves significantly better results than any other loss function.

First we show the predicted occupancy maps of MVDet, MVDeTr, SHOT and our method and compare them with the ground truth, in the traditional setting. Subsequently, qualitative results are shown w.r.t to three generalization abilities obtained from both the WildTrack and MultiViewX datasets.

### A. WildTrack Dataset

The traditionally evaluated results which contains occupancy maps of ground truth, our method, MVDet, MVDeTr and SHOT are shown in Fig. 7. The occupancy map from our method which uses average pooling, KLCC loss function and ImageNet pretraining gives us more accurate localization as compared to the base MVDet architecture. The results (maps) are competitive when compared to SHOT and MVDeTr. The maps obtained using MVDeTr are sharper and focused, however, it also has more false positives.

**Varying number of cameras:** The output occupancy map for varying number of cameras are shown in Fig. 8. WildTrack consists of seven cameras, we show the results inferred with three cameras upto six cameras. As the number of views are increasing, we get an accurately localized occupancy map.

**Changing camera configurations:** The output occupancy map for cross subset evaluation are shown in Fig. 10. Here, we have the occupancy maps for a model trained on one set and tested on other set. For example, trained on camera views one, three, five and seven and tested on cameras two, four, five and six or vice-versa like the camera splits shown in Figure 6. Clearly the pre-training is improving localization in both the methods. Furthermore, our method with average pooling is better at disambiguating the occlusions and also giving brighter outputs (resulting in sharp maxima's).

### B. MultiViewX Dataset

In this subsection the qualitative results for MultiViewX dataset are been shown. We consider similar configurations as in the Wildtrack dataset. The obtained results clearly indicates the improvements our method brings over the MVDet, MVDeTr and SHOT model and observations are similar to that of the Wildtrack dataset. Fig. 7 shows the traditionally evaluated results.

**Varying number of cameras:** The output occupancy map for varying number of cameras are shown in Fig. 11. MultiViewX consists of six cameras, we show the results inferred with three cameras upto five cameras. As the number of views are increasing, we get an accurately localized occupancy map.

**Changing camera configurations:** The output occupancy map for cross subset evaluation are shown in Fig. 12. Here, we have the occupancy maps for a model trained on one set and tested on other set. For example, trained on camera views one, three, and four and tested on cameras two, five and six or vice-versa, the camera splits are shown in Figure 9 and their results are shown in Table IX.

Method	Inference on {1,3,4}				Inference on {2,5,6}				
	MODA	MODP	Prec	Recall	MODA	MODP	Prec	Recall	
Trained on camera set {1,3,4}	MVDet	72	76.1	93.5	77.4	46.3	66.4	94.5	49.1
	MVDeTr	<b>77.4</b>	<b>85.1</b>	97.9	79	60.4	71.3	95.4	63.5
	SHOT	74.3	76.3	94.1	<b>79.3</b>	37.3	67	67.5	<b>72.1</b>
	Ours	67.7	76.4	96.2	70.5	59.6	73.4	94.7	63.2
	Ours (DropView)	67.3	75.3	<b>98.4</b>	68.5	<b>62.9</b>	<b>73.6</b>	<b>96.3</b>	65.4
	Trained on camera set {2,5,6}	MVDet	34.3	66.2	93.8	36.7	77.6	77.4	93.8
MVDeTr		51.1	72.1	<b>94.9</b>	54	<b>83.1</b>	<b>87.1</b>	<b>97.8</b>	<b>85</b>
SHOT		47.3	<b>73</b>	94.2	50.3	80.7	78.7	96.1	84.1
Ours		45.8	71.8	94.5	48.6	76.1	78.7	95.9	79.5
Ours (DropView)		<b>53.4</b>	71.6	88.2	<b>61.6</b>	75.2	77.4	92.8	81.5

TABLE IX

EXPERIMENTS ON THE MULTIVIEWX DATASET WITH CHANGING CAMERA CONFIGURATIONS

### C. Scene Generalization

The qualitative results of output occupancy map for cross-dataset evaluation are shown in Fig. 13, when we train on synthetic dataset (MultiViewX ) and test on real dataset (WildTrack ). First four occupancy maps are the outputs of MVDet, MVDeTr, SHOT and our method when tested on only 6 views of WildTrack dataset for having a fair comparison with other methods. We also show the output occupancy map when tested on all the views of WildTrack dataset. Our method provides accurately localized occupancy maps and disambiguate the occlusions as compared to other methods.

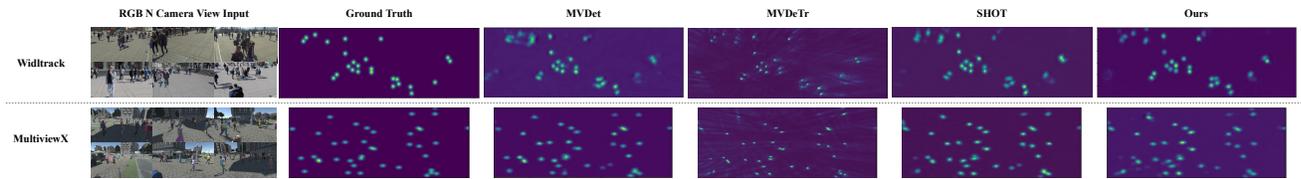


Fig. 7. Sample frames from WildTrack and MultiViewX dataset with corresponding occupancy maps of ground truth, our result MVDet, MVDeTr and SHOT for comparison. We can see the clusters forming in the MVDet predictions, in contrast our method gives much sharper and distinct predictions.

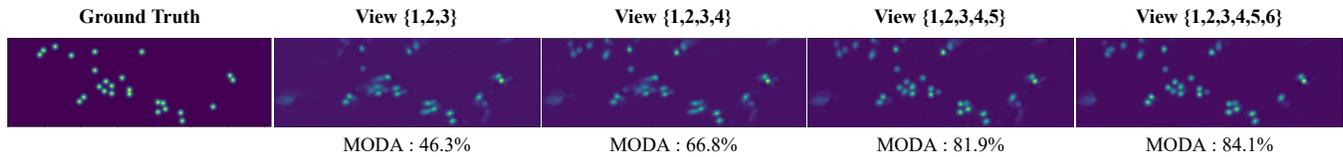


Fig. 8. Occupancy maps for varying number of cameras on WildTrack dataset when trained on seven cameras and tested on varying subsets of the cameras.

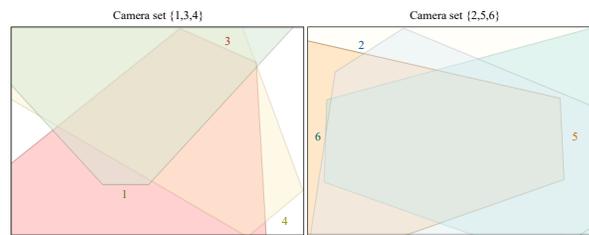


Fig. 9. Camera splits of MultiViewX dataset for changing camera configuration experiment shown in Table IX.

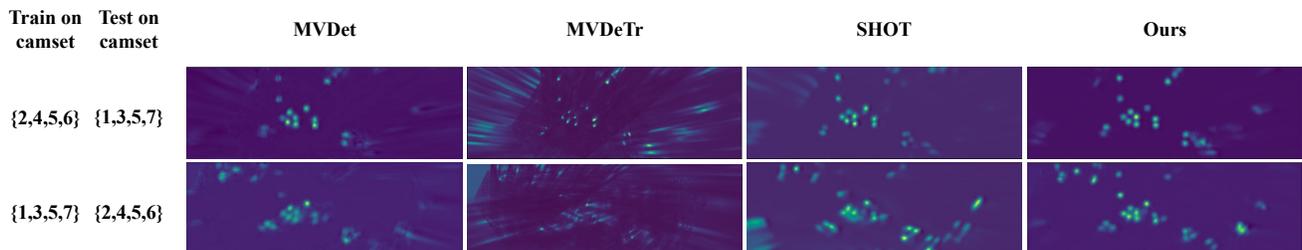


Fig. 10. Result occupancy maps for cross subset evaluation from WildTrack dataset.

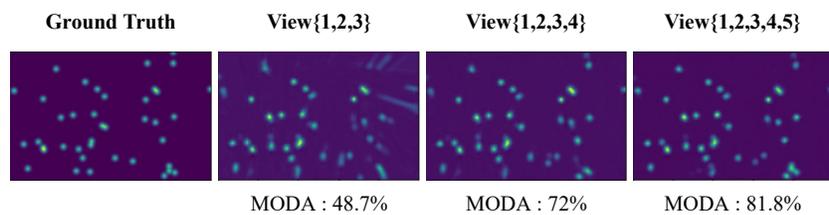


Fig. 11. Occupancy maps for varying number of cameras on MultiViewX dataset when trained on seven cameras and tested on varying subsets of the cameras.

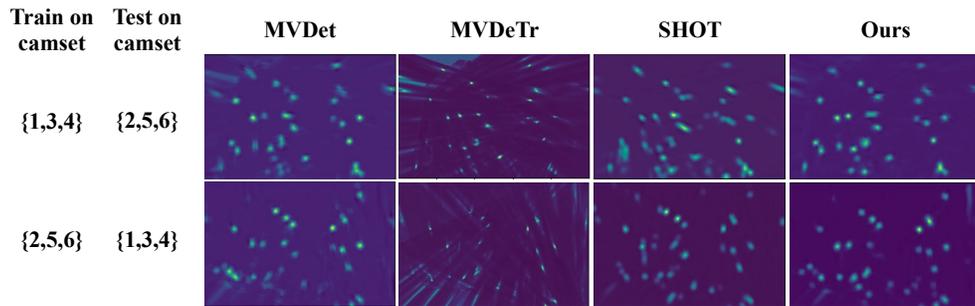


Fig. 12. Result occupancy maps for cross subset evaluation from WildTrack dataset.

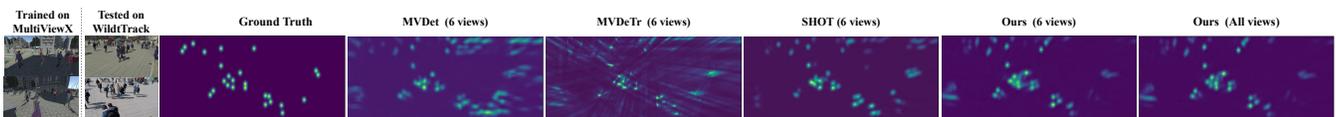


Fig. 13. Occupancy maps obtained on inference from WildTrack dataset where the models were trained on the synthetic dataset (MultiViewX).