Geometry-Aware Video Object Detection for Static Cameras

Dan Xu
danxu@robots.ox.ac.uk
Weidi Xie
weidi@robots.ox.ac.uk
Andrew Zisserman
az@robots.ox.ac.uk

Visual Geometry Group,
Department of Engineering Science,
University of Oxford,
Oxford, UK

Abstract

In this paper we propose a geometry-aware model for video object detection. Specifically, we consider the setting that cameras can be well approximated as static, *e.g.* in video surveillance scenarios, and scene pseudo depth maps can therefore be inferred easily from the object scale on the image plane.

We make the following contributions: *First*, we extend the recent anchor-free detector (CornerNet [17]) to video object detections. In order to exploit the spatial-temporal information while maintaining high efficiency, the proposed model accepts video clips as input, and only makes predictions for the starting and the ending frames, *i.e.* heatmaps of object bounding box corners and the corresponding embeddings for grouping. *Second*, to tackle the challenge from scale variations in object detection, scene geometry information, *e.g.* derived depth maps, is explicitly incorporated into deep networks for multi-scale feature selection and for the network prediction. *Third*, we validate the proposed architectures on an autonomous driving dataset generated from the Carla simulator [6], and on a real dataset for human detection (DukeMTMC dataset [28]). When comparing with the existing competitive single-stage or two-stage detectors, the proposed geometry-aware spatio-temporal network achieves significantly better results.

1 Introduction

In the Deep Learning era, we always expect the deep networks to learn all the required world knowledge given sufficient training data. However, as images are essentially a 2D projection of the 3D world, and the depth information has been lost during the image formation [12], the scene geometry, *e.g.* the depth, plays an essential role in resolving the ambiguities from scale variations and object occlusion in images. Despite the great success achieved in video object detection [11, 15, 22, 35], detecting objects under different scales or occlusions has only partially been tackled via learning multi-scale features in a brute-force fashion [2] or utilizing aggressive data augmentation [23]. This may indicate that these detectors actually have not learnt the scene geometry well.

These existing approaches mostly work on videos collected from dynamic cameras or internet video streams, such as the ImageNet VID dataset [29], and thus the varying scene

(a) A False Positive Detection Case (b) Height in Pixels of Objects (c) Pseudo Depth Map of Humans

Figure 1: An illustration of the motivation of using scene geometry for detection: (a) a false positive detection of vehicle with a wrong scale; (b) the height of objects in different classes; (c) the geometry priors derived from (b), *i.e.* 2.5D pseudo depth maps, able to provide useful geometric constraints on the object scales for learning a geometry-aware detector.

geometry is difficult to incorporate explicitly. However, there are many real-world applications, for instance the video surveillance, where the cameras can be well-approximated as static settings, *i.e.* the camera sits at a fixed position, and the relative depth of an object in the world coordinate system can also be determined by its height on the image plane [13]. In this paper, we propose a deep model for video object detection under static cameras, where the relative scene depth information can be estimated effectively (as shown in Figure 1), and is further used as strong geometric constraint for learning scale-aware object detector.

To investigate the effectiveness of using scene geometry in video object detection under the static camera settings, we first design a compact video object detector. While the two-stage anchor-based object detectors (*e.g.* Faster-RCNN [27]) have achieved impressive accuracy on image-wise object detection, the efficiency of the model is usually upper-bounded by the region proposal process. The generation of proposals would also significantly increase the complexity of the model especially when we deal with object detection in videos. In this paper, we extend the more efficient single-stage anchor free and single-frame object detection model CornerNet [17] by incorporating spatio-temporal information, the proposed geometry-aware spatio-temporal network is termed as GAST-Net. Our main contribution is therefore three-fold: (i) we design a spatio-temporal corner network structure, which accepts video clips (image sequences) as input. As far as we know, this is the first use of a corner-based scheme for object detection in videos. The network utilizes a spatio-temporal convolutional backbone to encode appearance and motion representations, which are further used to predict and group corners only for the first frame and the last frame of the sequences. By doing so, we are able to capture the long temporal dependencies in videos. (ii) We explore how the scene geometry derived from static cameras can be employed as priors for multi-scale feature selection and for the network prediction, therefore helping to tackle the challenges from scale variations in video object detection. (iii) Extensive experiments have been conducted on a synthetic automous driving dataset generated with Carla [5], and on pedestrian detection on the DukeMTMC dataset [28]. On both datasets, we demonstrate great benefits of incorporating the scene geometry, and show that the proposed GAST-Net significantly outperforms existing competitive single-stage and two-stage detectors.

2 Related Work

Object Detection in Static Images. Two families of detectors are currently popular: First, two-stage detectors, *e.g.* R-CNN [9], Fast R-CNN [8], Faster R-CNN [27] and R-FCN [6].

The main idea of these detectors is to train a small sub-network for generating proposals that potentially contain objects, and then learn a classification network to predict the existence and categories of the objects. Second, one-stage detectors that predict object bounding boxes in one step such as YOLO [26], SSD [20] and CornerNet [17]. Our model is also an one-stage detector which first predicts corner heatmaps and embeddings, then groups the corners as individual objects similar to CornerNet, while we extend it by building up a spatio-temporal corner network to capture temporal information for video object detection.

Object Detection in Videos. As an important research topic, video object detection has drawn significant attention [1, 14, 16, 24, 33, 35, 36, 38]. To take advantage of existing image-based detectors, several works focus on post-processing class scores from image-based detectors, and enforce temporal consistency on the scores. For instance, tubelet proposals are generated in [15] via applying a tracker to frame-based bounding box proposals. The class scores along the tubelet are further re-scored by a 1D CNN model. Unlike the detection problem in static images, videos naturally contain temporal coherence with objects changing smoothly in time. Zhu *et al*. [39] thus consider a motion-based model that applies a detection net only on key frames, and an optical flow net is used for propagating deep features to the rest of the frames. To further simplify dense prediction in optical flow, a recent work [7] proposes to simultaneously learn detection and tracking. In this paper, we consider a common case of video object detection, where the camera is static. Under this situation, all the derived geometry information can therefore be applied to help the CNN with geometry-aware learning, in order to eliminate the scale ambiguities in 2D images.

Object Detection from RGB-D data. Another line of research is about using RGB-D data where the scene depth information has been demonstrated beneficial for various computer vision tasks [19, 30], and is also widely used for object detection [6, 10, 32]. Among the existing works, Gupta *et al*. [10] proposed a joint framework for object detection and semantic segmentation, and the depth maps are encoded with a geocentric encoding approach to provide complementary features to the RGB representations. Spinello *et al*. [32] also explored using RGB-D data as input for people detection. There also exists some works exploring using depth data for 3D object detection [25, 31]. Qi *et al*. [25] developed a 3D point cloud deep representation model for effective 3D object bounding box prediction. However, these works require the explicit depth data captured from depth sensors, which are not always available in many application scenarios. Our work targets deriving the scene depth information from the RGB data, and thus does not require additional depth sensors other than standard RGB cameras.

Geometry-Aware Deep Learning. Scene geometry is considered as important prior information for computer vision tasks [21, 37]. Leibe *et al*. [18] explored joint object tracking and detection using geometry assumptions within a traditional non-deep-learning framework. In crowd counting [21], as the camera usually sits on a fixed position and the variance between people's height is small, it is easy to obtain the homography between the image and the head plane. By incorporating this information in the model, it becomes possible to directly predict the crowd density in the physical world. Moreover, previous works [13] also considered to place the local object detection in the context of the overall 3D scene, by directly modelling the interdependence of objects, surface orientations, and camera viewpoint. In our work, we aim to explore using scene geometry for the task of video object detection in CNN. Instead of estimating accurate 3D geometry, we consider deriving and utilizing scene-specific coarse depth as well as image-plane coordinates, and enforce the convolutional operations to be conditioned on the object scales and positions, leading to a geometry-aware deep learning.

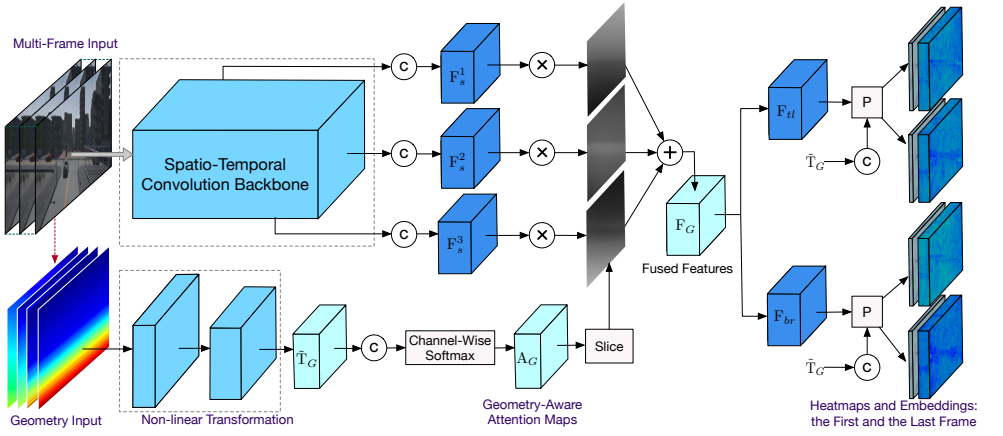


Figure 2: Framework of the proposed geometry-aware spatio-temporal corner network for video object detection from static cameras. It accepts multiple frames as input, and predicts heatmaps and embeddings of the first and the last frame for detection. The geometry input contains 2D image-plane coordinates and 2.5D pseudo depth maps, which can be directly derived from training data. **P** denotes a prediction module. The symbols ⓒ, $\otimes$ and $\oplus$ denote convolution, element-wise multiplication and element-wise addition operation, respectively.

3 Geometry-Aware Spatio-Temporal Corner Network

Figure 2 depicts a framework overview of the proposed GAST-Net. It consists of two main components. The first is the proposed spatio-temporal network that accepts video clips as input, and outputs multi-heads feature representations with both appearance and motion information at different scales. The second component is a geometry-aware module that first encodes the inferred relative depth maps (*i.e.* the pseudo depth maps), and further used for selecting the features dynamically based on the geometry information. Intuitively, given the depth for all pixels on the image plane, in order to detect objects that are close to the camera, features from a large receptive field should be used. In our case, 2D image-plane coordinates and the pseudo depth maps are used to represent the image and scene geometry, which can both be derived from the training data. Eventually, the corner heatmaps and embeddings are predicted from the fused feature representation, and bounding boxes are obtained by grouping the corners. We introduce the details of the proposed GAST-Net in the following.

3.1 Spatio-Temporal Corner Network

Single-Frame Corner Network. In contrast to the traditional anchor-based detectors, such as SSD [20] and Faster-RCNN [27], CornerNet [17] is an one-stage and anchor free detector in a bottom-up fashion. The main idea is to regress heatmaps for the top-left and the bottom-right corners of the objects and predict their embeddings. The bounding boxes that outline the objects are later generated via grouping corners from the embeddings.

Proposed Multi-Frame Spatio-Temporal Corner Network. For efficient object detection in videos, we extend the CornerNet architecture by exploiting the spatio-temporal information. To capture the temporal relationship between adjacent frames, the input to our model is video clips with multiple frames, and a 3D convolution-based backbone is used as an en-

coder (variants of VGG and ResNet50 in our experiments). The video clip representation F_G (Figure. 2) is further projected to two separate feature maps F_{tl} and F_{br} (corresponding to the first and last frame respectively), and later decoded as heatmaps and embeddings of the top-left and bottom-right corners for the object bounding boxes. In order to improve the efficiency and reduce the computational overhead, supervision is only added on the first and the last frame. Multi-scale context is enabled by taking feature maps from different layers from the encoder, and the spatio-temporal representation F_G is calculated by fusing the multi-scale features with the proposed geometry-aware network module.

3.2 Geometry-Aware Network Module

Scene Geometry from Static Cameras. Static cameras are used in a wide range of real-world applications, where an important task is to detect and track all the cars and pedestrians. In these scenarios, as the variations of the objects' physical height tend to be small, the depth information will be directly related to their sizes on image plane. For instance, the further objects will have smaller scales according to the perspective projection of the camera [13]. Therefore, the geometry information from static cameras *i.e.* the relative depth, can be directly estimated from the training data, as shown the psudo depth map in Figure 1.

In this work, we mainly consider two types of geometry information, *e.g.* image geometry and scene geometry. The image geometry considers the image-plane 2D coordinates as auxiliary information, which can be treated as a means to enable position-dependent convolutions. We generate a set of two coordinate maps $\{G_x, G_y\}$ for the x and the y dimension respectively. In $G_x \in \mathbb{R}^{H\times W}$, with H and W as the height and the width of the images, each column is the x dimension coordinate, while in $G_y \in \mathbb{R}^{H\times W}$, each row is the y dimension coordinate. G_x and G_y are normalized in the range of [0,1]. The other type of geometry information is the scene geometry, *i.e.* relative scene depth maps. For a fixed camera viewpoint v_m, the object height and its depth are inversely proportional [13]. Given the bounding boxes of all the objects from the training data, we are able to estimate a coarse relative depth map by calculating the mean of the maximal and the minimal height of the bounding boxes for each row on the map. The rows without any objects are bilinearly interpolated using the values of adjacent rows. For each object class c_n and each camera viewpoint v_m, we estimate such a pseudo depth map, and for the whole training data, we have a set of class- and scene-specific pseudo depth maps, $\{D_{v_m,c_n} \in \mathbb{R}^{H\times W}\}(m = 1,...,M, n = 1,...,N)$, where N and M are the number of object classes and camera viewpoints, respectively.

Non-Linear Transformation of Geometry Information. Given a camera view v_m, we concatenate all the coordinate maps with the pseudo depth maps, and perform non-linear transformations with two Convolution-Batch Norm-ReLU blocks denoted as Conv-BN-ReLU$_2(\cdot)$. Then the transformation operation is formulated as follows:

$$\tilde{T}_G = \text{Conv-BN-ReLU}_2(\text{concat}(G_x, G_y, D_{v_m,c_1}, ..., D_{v_m,c_N})), \quad (1)$$

where concat$(\cdot)$ is a concatenation operation. After that, we obtain a fine-grained geometry distribution $\tilde{T}_G$. In our framework, $\tilde{T}_G$ is used to guide the multi-scale feature selection with an attention mechanism, and is also used to guide the prediction of heatmaps and embeddings for later grouping corners.

Geometry-Aware Multi-Scale Feature Fusion. In order to detect objects of different scales in the image plane, the geometry information is used to modulate the multi-scale features

with an attentional process. Given a set of S multi-scale features $\{F_s^i\}_{i=1}^S$, we correspondingly learn a set of S geometry-aware attention maps A_G, $A_G = \{A_G^i\}_{i=1}^S$. Our intuition of using $\tilde{\mathrm{T}}_G$ for attention generation is that the geometry information, *e.g.* pseudo depth map, essentially has strong constraints to the object scales on image plane. Formally, we generate the attention maps as follows:

$$\mathrm{A}_G = \mathrm{Softmax}(W_G \tilde{\mathrm{T}}_G + b_G), \tag{2}$$

$\mathrm{Softmax}(\cdot)$ is computed along the channels, and $\{W_G, b_G\}$ are the convolution parameters. Then the set of attention maps $\{A_G^i\}_{i=1}^S$ is used to select and fuse features in different scales as follows:

$$\mathrm{F}_G = \mathrm{A}_G^1 \otimes \mathrm{F}_s^1 + \cdots + \mathrm{A}_G^S \otimes \mathrm{F}_s^S, \tag{3}$$

where the symbol $\otimes$ denotes an element-wise multiplication operation.

Geometry-Aware Prediction. The geometry distribution $\tilde{\mathrm{T}}_G$ is also used to guide the prediction of heatmaps and embeddings in the spatio-temporal corner network. The prediction part has four independent convolutional layers, corresponding to the top-left and bottom-right corners of the first frame and the last frame, respectively. For each prediction layer, it accepts features from both the image sub-network and from the geometry sub-network, *i.e.* $\tilde{\mathrm{T}}_G$. A separate 1×1 convolution is applied on $\tilde{\mathrm{T}}_G$ to adjust the feature dimensions. In our setting, the number of geometry feature channels is set to $1/4$ of the image one. Then these two parts of features are concatenated and input into the prediction convolutional layer.

3.3 Network Optimization and Inference

The overall network architecture uses a combination of two types of losses. One is a regression loss on the corner heatmaps, and the other is an embedding loss for both the first frame and the last frame. Similar to [17], the heatmap regression uses a focal loss, since the number of corner pixels is much fewer than background pixels. The embedding loss employs a pull-push loss that aims to train the network to group the corners by a pull loss, and to separate the corners by a push loss. During the inference, we supply the network with testing video clips and the geometry input derived in the training phase, where we assume that the training and the testing data are collected under the same camera scenes, which is usually a common setting in applications with static cameras. For each frame in the videos, it could be the first frame or the last frame of the input clips, and thus each frame is predicted twice. We collect grouped bounding boxes from the two-times predictions and apply an NMS operation to get the final bouding box output of that frame.

4 Experiments

4.1 Experimental Setup

Datasets. We conduct the experiments on two different datasets: (i) a synthetic dataset generated from an open-sourced self-driving simulator Carla [5], termed as Carla-Vehicle-Pedestrian dataset. We collected around 48 scenes with in total 60K images. The resolution of each image is of 720×1280. Among them 40 scenes with around 50k images are used for training and the rest for testing. The dataset contains two classes of pedestrian and vehicle.

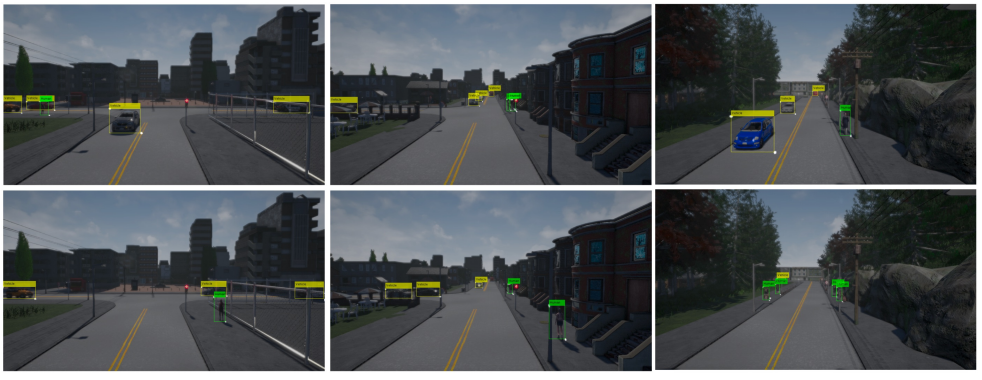


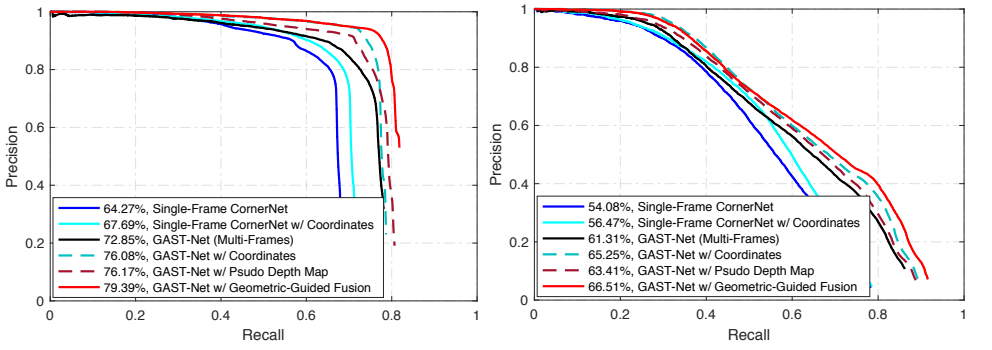
Figure 3: Qualitative detection results of humans and vehicles under three different scene views on Carla-Vehicle-Pedestrian dataset. The detected corners are visualized as grey blobs.

This dataset is very challenging as it has been generated with many small-scale pedestrians and vehicle objects. The frame rate of this dataset is round 9 fps. (ii) the DukeMTMC [28] dataset that was originally created for object tracking and person identification. The dataset contains long videos with 9 different static cameras. We use the video data from camera 1 to 5, and create a dataset of around 720K images. The frame rate of the DukeTMTC is 60 fps. The image resolution is 1080×1920. Among them 70% are used for training and the rest for testing. In the training, we sample images at every 6th frame. Several qualitative detection samples on the two datasets are shown in Figure 3 and Figure 5.

Parameter Setting and Evaluation Metrics. In training, the images are resized to a resolution of 360×640 for Carla-Vehicle-Pedestrian, and 270×480 for DukeMTMC. The number of input frames is set to 4 for both datasets. The batch size is set to 8 and 16, and the network is trained with 30 and 20 epochs for the two datasets respectively. We used Adam [4] for optimization; and the weights for the regression focal loss, the push loss, and the pull loss, are set as 1, 0.1 and 0.1 respectively. The learning rate is initialized as 10^{-4} for both datasets. The detection performance is evaluated with the metric of average precision at IoU 0.5 (AP^{50}) and at IoU 0.75 (AP^{75}), and also with mAP, which is calculated by taking the average over the two APs.

4.2 Experimental Results

Baseline models. To demonstrate the effectiveness of different components in the proposed GAST-Net, we conduct experiments on several different models: (i) Single-Frame CornerNet, which we follow [20] while replacing their hourglass backbone with a VGG-11 structure for fair comparison; (ii) GAST-Net (multi-frame), which is our base spatio-temporal corner network. We use a conv-3D network structure (*e.g.* C3D [34]) as the spatio-temporal convolutional backbone. The representations from the backbone are used to separately decode for the first and the last frame. This model does not employ any geometry information. (iii) Single-Frame CornerNet w/ 2D coordinates or 2.5D pseudo depth map in prediction, which uses 2D coordinates or 2.5D pseudo depth map in the network prediction module via the geometry network branch, and using the encoded $\tilde{\mathrm{T}}_G$ to help the network prediction as we described in the Sec. 3.2. This baseline model is built upon the Single-



(a) PR-Curve at IoU 0.5 on Vehicle Class (b) PR-Curve at IoU 0.5 on Pedestrian Class

Figure 4: Comparison of Precision-Recall Curves of different variants of the proposed approach on the Carla-Vehicle-Pedestrian dataset.

Method	Vehicle-Class		Pedestrian-Class	
	AP^{50}	AP^{75}	AP^{50}	AP^{75}
Single-Frame Corner Net [20]	64.27%	52.20%	54.08%	28.63%
GAST-Net (multi-frame)	72.85%	62.37%	61.31%	54.82%
Single-Frame Corner Net w/ 2D coordinates in prediction	67.69%	55.83%	56.47%	31.56%
Single-Frame Corner Net w/ 2.5D psudo depth map in prediction	67.91%	54.75%	56.67%	30.83%
GAST-Net (multi-frame) w/ 2D coordinates in prediction	76.08%	69.02%	65.25%	57.54%
GAST-Net (multi-frame) w/ 2.5D psudo depth map in prediction	76.17%	66.52%	63.41%	56.11%
GAST-Net (multi-frame) w/ geometry-guided feature fusion	**79.39%**	**71.95%**	**66.51%**	**59.06%**

Table 1: Quantitative comparison of different variants of the proposed approach on the Carla-Vehicle-Pedestrian dataset. We use a backbone structure of C3D [34], which utilizes a VGG-11 structure while replacing all 2D convolution/pooling with 3D convolution/pooling.

Frame CornerNet, *i.e.* the model (i); (iv) GAST-Net (multi-frame) w/ 2D coordinates or 2.5D pseudo depth map in prediction, which uses 2D coordinates or 2.5D pseudo depth map in the prediction module in a means similar to (iii) via utilizing the encoded geometry distribution $\tilde{\mathrm{T}}_G$ for the network prediction. This model is directly built upon the model GAST-Net (multi-frame), *i.e.* the model (ii); (v) GAST-Net (multi-frame) w/ geometry-guided feature fusion (our full model). It further uses the geometry information to guide the multi-scale feature fusion upon the model (iv) that uses the geometry only for the network prediction. All the models are learned in the same training setting as described in Sec. 4.1 for a fair performance comparison.

Effectiveness of multi-frame spatio-temporal corner prediction. We conduct ablation study on the Carla-Vehicle-Pedestrian dataset. A quantitative comparison of different baseline models is shown in Table 1. PR-Curves of the different approaches are shown in Figure 4. It can be observed that GAST-Net (multi-frame) significantly outperforms Single-Frame CornerNet on all the metrics by a large margin. In terms of AP^{50}, GAST-Net (multi-frame) is around 8.6 and 7.0 points better than Single-Frame CornerNet on the vehicle and the pedestrian class, respectively. The performance gain is even higher on the more strict metric of AP^{75}, demonstrating the effectiveness of incorporating temporal relationship in the

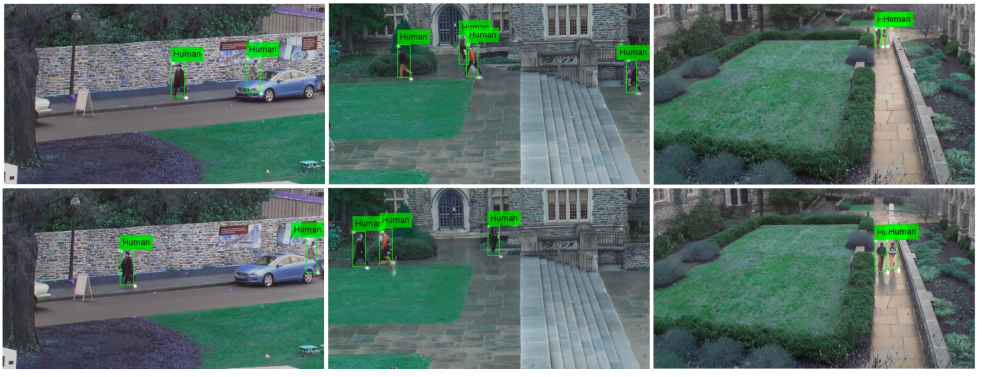


Figure 5: Qualitative detection results of humans under three different camera view points on the DukeMTMC dataset. The detected corners are visualized in grey blobs.

Method	Backbone	mAP	AP^{50}	AP^{75}
Faster RCNN [27]	VGG	63.85%	80.56%	47.15%
Single-Shot Detector (SSD) [20]	VGG	59.06%	73.87%	44.26%
Single-Frame Corner Net [17]	VGG	61.49%	72.65%	50.34%
GAST-Net (full model)	VGG	**68.26%**	**78.65%**	**57.87%**
Faster RCNN [27]	ResNet-50	70.68%	81.73%	59.64%
Single-Frame Corner Net [20]	ResNet-50	68.71%	75.18%	62.25%
GAST-Net (full model)	ResNet-50	**74.42%**	**80.64%**	**68.21%**

Table 2: Quantitative comparison with competitive one-stage and two-stage detectors on the DukeMTMC dataset. Among the comparison methods, Faster RCNN [27] is a two-stage anchor-based detector, while the rest are all one-stage detectors.

proposed video object detection architecture.

Effectiveness of geometry guided prediction. When comparing the performance of GAST-Net (multi-frame) w/ 2D coordinates or 2.5D pseudo depth map with GAST-Net (multi-frame), it is clear that the geometry priors, *i.e.* both the 2D image-plane coordinates and the 2.5D pseudo depth maps, are beneficial for improving the detection performance. On the vehicle class, GAST-Net with 2D coordinates improves AP^{75} around 6.7 points, meaning that the coordinates are especially beneficial for the network to learn better localization of corners. We also use the geometry information for Single-Frame CornerNet, and consistent performance gains can be observed.

Effectiveness of geometry guided multi-scale feature fusion. In this section, we use the learned geometry distribution to guide multi-scale feature fusion. As shown from Table 1, GAST-Net w/ Geometry-guided fusion further achieves better performance than model (iv) on all the metrics and on all the classes, verifying our initial motivation of encoding the geometry information into deep network for geometry-aware scale perception and learning.

Comparison with existing one-stage and two-stage detectors. We compare the proposed architecture with representative one-stage and two-stage object detectors, including Single-Shot MultiBox Detector (SSD) [20], Faster-RCNN [27], and Single-Frame CornerNet [17]

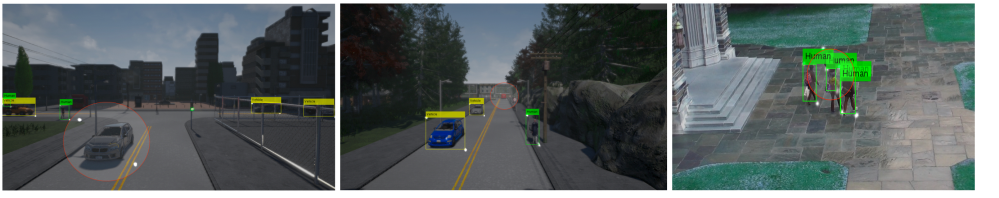


Figure 6: Failure examples on the CVP and the DukeTMTC datasets. The object detections that fail are marked with red circles. The missing (*e.g*. the first two examples), or inaccurate *grouping* (*e.g*. the last example) of the detected top-left and bottom right corners is an important factor affecting the final detection performance.

on the DukeMTMC dataset. The comparison experiments are performed with two differnet backbone network structures, *e.g*. VGG-11 and ResNet50. Quantitative comparisons are shown in Table 2. GAST-Net achieves the best performance among these competitors. Specifically, to compare with the one-stage detectors, ours is 7.1 points better than Single-Frame CornerNet, and 8.1 points better than SSD on the mAP metric with VGG backbone. Ours is also around 4.4 points better than the two-stage Faster-RCNN approach. It can be also noted that, our corner-based framework has much better performance than anchor-based SSD and Faster-RCNN on AP^{75}, which is probably because that the dense prediction of corners is more powerful in accurate bounding box localization than using sparse anchor based proposal generation.

Discussion. The proposed GAST-Net is an one-stage based approach, which detects the top-left and the bottom-right corners, and learns to group the corresponding corners to bounding boxes. The final detection performance is thus affected by the grouping capability. In our experiments, we observed that the detector is able to produce very good detection and localization on the object corners *w.r.t* the Percentage of Correct Keypoints (PCK) recall metric. However, the grouping fails in some cases, for instance, for extreme scale of objects, or for crowded cases with dense occlusion, as shown in Figure 6, leading to lower recall on the object bounding boxes. Possible solutions to tackle the grouping issues could be investigating a scale-aware network structure with long-term tracking.

5 Conclusion

We have presented a geometry-aware spatio-temporal network (GAST-Net) for video object detection from static cameras. GAST-Net consists of two main parts. One is the spatio-temporal corner network that aims to perform object detection from corner estimation and grouping with video clips as input. The other part is the designed geometry-aware network module which utilizes the scene geometry derived from static cameras for multi-scale feature selection and fusion. Extensive experiments on two challenging datasets demonstrate the superior performance of the proposed approach, and show the great advantage of using geometry in deep networks for the video object detection task. The geometry-aware network module is also potentially beneficial for other computer vision tasks affected by scale issues, such as object tracking and semantic segmentation.

Acknowledgement. This research work was supported by the EPSRC Programme Grant Seebibyte EP/M013774/1.

References

[1] G. Bertasius, L. Torresani, and J. Shi. Object detection in video with spatiotemporal sampling networks. In *ECCV*, 2018.

[2] L. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *CVPR*, 2016.

[3] J. Dai, Y. Li, K. He, and J. Sun. R-fcn: Object detection via region-based fully convolutional networks. In *NIPS*, 2016.

[4] D. P. Diederik and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[5] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. CARLA: An open urban driving simulator. In *CoRL*, 2017.

[6] Andreas Eitel, Jost Tobias Springenberg, Luciano Spinello, Martin Riedmiller, and Wolfram Burgard. Multimodal deep learning for robust rgb-d object recognition. In *IROS*, 2015.

[7] C. Feichtenhofer, A. Pinz, and A. Zisserman. Detect to track and track to detect. In *ICCV*, 2017.

[8] R. B. Girshick. Fast R-CNN. In *ICCV*, 2015.

[9] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv preprint arXiv:1311.2524*, 2013.

[10] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from rgb-d images for object detection and segmentation. In *ECCV*, 2014.

[11] W. Han, P. Khorrami, T. L. Paine, P. Ramachandran, M. Babaeizadeh, H. Shi, J. Li, S. Yan, and T. S. Huang. Seq-nms for video object detection. *arXiv preprint arXiv:1602.08465*, 2016.

[12] R. Hartley and A. Zisserman. Multiple view geometry in computer vision. Cambridge university press, 2003.

[13] D. Hoiem, A. Efros, and M. Hebert. Putting objects in perspective. In *CVPR*, 2006.

[14] A. Joulin, K. Tang, and F. Li. Efficient image and video co-localization with frank-wolfe algorithm. In *ECCV*, 2014.

[15] K. Kang, W. Ouyang, H. Li, and X. Wang. Object detection from video tubelets with convolutional neural networks. In *CVPR*, 2016.

[16] S. Kwak, M. Cho, I. Laptev, J. Ponce, and C. Schmid. Unsupervised object discovery and tracking in video collections. In *ICCV*, 2015.

[17] H. Law and J. Deng. Cornernet: Detecting objects as paired keypoints. In *ECCV*, 2018.

[18] B. Leibe, K. Schindler, N. Cornelis, and V. G. Luc. Coupled object detection and tracking from static cameras and moving vehicles. *TPAMI*, 30(10):1683–1698, 2008.

[19] Dahua Lin, Sanja Fidler, and Raquel Urtasun. Holistic scene understanding for 3d object detection with rgbd cameras. In *ICCV*, 2013.

[20] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, and A.C. Berg. SSD: Single shot multibox detector. In *ECCV*, 2016.

[21] W. Liu, K. Lis, M. Salzmann, and P. Fua. Geometric and physical constraints for head plane crowd density estimation in videos. *arXiv preprint arXiv:1803.08805*, 2018.

[22] Y. Lu, C. Lu, and C. K. Tang. Online video object detection using association lstm. In *ICCV*, 2017.

[23] B. Pepikj, M. Stark, P. Gehler, and B. Schiele. Occlusion patterns for object class detection. In *CVPR*, 2013.

[24] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012.

[25] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *CVPR*, 2018.

[26] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016.

[27] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2016.

[28] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCV*, 2016.

[29] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and F. Li. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[30] Max Schwarz, Anton Milan, Arul Selvam Periyasamy, and Sven Behnke. Rgb-d object detection and semantic segmentation for autonomous manipulation in clutter. *IJRR*, 37 (4-5):437–451, 2018.

[31] Shuran Song and Jianxiong Xiao. Deep sliding shapes for amodal 3d object detection in rgb-d images. In *CVPR*, 2016.

[32] Luciano Spinello and Kai O Arras. People detection in rgb-d data. In *IROS*, 2011.

[33] P. Tang, C. Wang, X. Wang, W. Liu, W. Zeng, and J. Wang. Object detection in videos by high quality object linking. *TPAMI*, 2019. doi: 10.1109/TPAMI.2019.2910529.

[34] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.

[35] S. Tripathi, Z. C. Lipton, S. Belongie, and T. Nguyen. Context matters: Refining object detection in video with recurrent neural networks. *arXiv preprint arXiv:1607.04648*, 2016.

[36] T. Vu, A. Osokin, and I. Laptev. Tube-cnn: Modeling temporal evolution of appearance for object detection in video. *arXiv preprint arXiv:1812.02619*, 2018.

[37] F. Wang, L. Zhao, X. Li, X. Wang, and D. Tao. Geometry-aware scene text detection with instance transformation network. In *CVPR*, 2018.

[38] J. L. Xiao, F.and Yong. Video object detection with an aligned spatial-temporal memory. In *ECCV*, 2018.

[39] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Wei Y. Deep feature flow for video recognition. In *CVPR*, 2017.