Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI-23)
Sister Conferences Best Papers Track

Translating Images into Maps (Extended Abstract) *

Avishkar Saha[1] ‡, **Oscar Mendez**[1], **Chris Russell**[2] and **Richard Bowden**[1]
[1]Centre for Vision Speech and Signal Processing, University of Surrey, Guildford, UK
[2]Amazon, Tubingen, Germany
{a.saha, o.mendez, r.bowden}@surrey.ac.uk, cmruss@amazon.com

Abstract

We approach instantaneous mapping, converting images to a top-down view of the world, as a translation problem. We show how a novel form of transformer network can be used to map from images and video directly to an overhead map or bird's-eye-view (BEV) of the world, in a single end-to-end network. We assume a 1-1 correspondence between a vertical scanline in the image, and rays passing through the camera location in an overhead map. This lets us formulate map generation from an image as a set of sequence-to-sequence translations. This constrained formulation, based upon a strong physical grounding of the problem, leads to a restricted transformer network that is convolutional in the horizontal direction only. The structure allows us to make efficient use of data when training, and obtains state-of-the-art results for instantaneous mapping of three large-scale datasets, including a 15% and 30% relative gain against existing best performing methods on the nuScenes and Argoverse datasets, respectively.

1 Introduction

Many tasks in autonomous driving are substantially easier from a top-down, map or bird's-eye view (BEV). As many autonomous agents are restricted to the ground-plane, an overhead map is a convenient low-dimensional representation, ideal for navigation, that captures relevant obstacles and hazards. For scenarios such as autonomous driving, semantically segmented BEV maps must be generated on the fly as an instantaneous estimate, to cope with freely moving objects and scenes that are visited only once.

Inferring BEV maps from images requires determining the correspondence between image elements and their location in the world. Multiple works guide their transformation with dense depth and image segmentation maps [Sengupta *et al.*2012, Pan *et al.*2020, Liu *et al.*2020, Wang *et al.*2019, Schulter *et al.*2018], while others [Lu *et al.*2019, Mani *et al.*2020, Roddick and Cipolla2020, Philion and Fidler2020, Saha *et al.*2021] have developed approaches which resolve depth and semantics implicitly. Although some exploit the camera's geometric priors [Roddick and Cipolla2020, Philion and Fidler2020, Saha *et al.*2021], they do not explicitly learn the interaction between image elements and the BEV-plane.

Unlike previous approaches, we treat the transformation to BEV as an image-to-world translation problem, where the objective is to learn an alignment between vertical scan lines in the image and polar rays in BEV. The projective geometry therefore becomes implicit to the network. For our alignment model, we adopt transformers [Vaswani *et al.*2017], an attention-based architecture for sequence prediction. With its attention mechanisms, we explicitly model pairwise interactions between vertical scanlines in the image and their polar BEV projections.

The contributions of our paper are (1) We formulate generating a BEV map from an image as a set of 1D sequence-to-sequence translations. (2) By physically grounding our formulation we construct a data-efficient transformer network that is convolutional with respect to the horizontal x-axis, yet spatially-aware. (3) We show how axial attention improves performance by providing temporal awareness and demonstrate state-of-the-art results across three large-scale datasets.

2 Related Work

BEV object detection: Early approaches detected objects in the image and regressed 3D pose parameters [Mousavian *et al.*2017, Kehl *et al.*2017, Simonelli *et al.*2019, Poirson *et al.*2016, Palazzi *et al.*2017, Chen *et al.*2016]. OFTNet [Roddick *et al.*2019] generated 3D features from a projected voxel grid for 3D object detection. Our approach decouples the relationship between the distance from the camera and the context available to each voxel, allowing each BEV position to access the entire vertical axis of the image.

Inferring semantic BEV maps: Current state-of-the-art approaches can be categorized as either 'compression' [Roddick and Cipolla2020, Saha *et al.*2021] or 'lift' [Philion and Fidler2020, Hu *et al.*2021] approaches. 'Compression' methods vertically condense image features and expand into BEV, implicitly relating an object's depth to its available context. However, they may ignore small, distant objects. 'Lift' approaches expand each image into a frustum of features to learn pixel-wise depth distribution but lack spatial awareness

*Paper initially published in the *International Conference on Robotics and Automation (ICRA) 2022*.

‡Extended abstract done while at Amazon.

and may overfit. We address this by (1) maintaining image spatial structure for alignment with the BEV-plane and (2) introducing spatial awareness to assign image context across the ray space based on content and position.

Encoder-decoder transformers: Transformers [Vaswani *et al.*2017] through their attention mechanisms [Bahdanau *et al.*2015] have produced state-of-the-art performance in many tasks [Devlin *et al.*2019,Dosovitskiy *et al.*2021]. Like us, the 2D detector DETR [Carion *et al.*2020] performs decoding in a spatial domain through attention. However, their predicted output sequences are sets of permutation invariant object detections without any spatial order. In contrast, our predicted BEV ray sequences is inherently spatial and so we need permutation equivariance in our decoding.

3 Method

Our goal is to learn a model $\mathbf{\Phi}$ that takes a monocular image $\mathbf{I}$ and produces a semantically segmented birds-eye-view map of the scene $\mathbf{Y}$. Formally, given an input image $\mathbf{I} \in \mathbb{R}^{3\times H\times W}$ and its intrinsic matrix $\mathbf{C} \in \mathbb{R}^{3\times 3}$, our model predicts a set of binary variables $\mathbf{Y}^k \in \mathbb{R}^{X\times Z}$ for each class $k \in K$: $p(\mathbf{Y}^k|\mathbf{I},\mathbf{C}) = \mathbf{\Phi}(\mathbf{I},\mathbf{C})$, where Φ is a neural network trained to resolve both semantic and positional uncertainties. The design of our network rests on our novel transformation between the image-plane $\mathbb{P}^I$ and BEV-plane $\mathbb{P}^{BEV}$. Our end-to-end approach, as shown in Fig. 1a, is composed of the following subtasks: (1) constructing representations in the image-plane which encode semantics and some knowledge of depth, (2) transforming the image-plane representation to BEV and (3) semantically segmenting the BEV-representation.

3.1 Image-to-BEV Translation

Transforming from image to BEV requires a mapping which determines the image pixel correspondence to BEV polar ray. As camera geometry dictates a 1-1 correspondence between each vertical scanline and its associated ray, we treat the mapping as a set of sequence-to-sequence translations. With reference to Fig. 1b, we want to find the discretized radial depths of elements in the vertical scan line of an image, up to r metres from the camera: we have an image column $S^I \in \mathbb{R}^H$, and we want to find its BEV ray $S^{\phi(BEV)} \in \mathbb{R}^r$, where H is the height of the column and r represents the radial distance from the camera. We propose learning the alignment between input scanlines and output polar rays through an attention mechanism [Bahdanau *et al.*2015]. We employ attention in two ways: (1) *inter-plane attention* as shown in Fig.1b, which initially assigns features from a scanline to a ray and (2) *polar ray self-attention* that globally reasons about its positional assignments across the ray. We motivate both uses below, starting with inter-plane attention.

Inter-plane attention: Consider a semantically segmented image column and its corresponding polar BEV ground truth. Here, alignment between the column and the ground truth ray is 'hard', *i.e.* each pixel in the polar ray corresponds to a single semantic category from the image column. Thus, the only uncertainty that must be resolved to make this a hard-assignment is the depth of each pixel. However, when making this assignment, we need to assign features that aid in resolving semantics and depth. Hence, a hard assignment would be detrimental. Instead, we want a soft-alignment, where every pixel in the polar ray is assigned a combination of elements in the image column, *i.e.* a *context* vector. Concretely, when generating each radial element $S_i^{\phi(BEV)}$, we want to give it a *context* c_i based on a convex combination of elements in the image column S^I and the radial position r_i of the element $S_i^{\phi(BEV)}$ along the polar ray. This need for context assignment motivates our use of soft-attention between the image column and its polar ray, as illustrated in Fig. 1.

Formally, let $\mathbf{h} \in \mathbb{R}^{H\times C}$ represent the encoded "memory" of an image column of height H, and let $\mathbf{y} \in \mathbb{R}^{r\times C}$ represent a *positional query* which encodes relative position along a polar ray of length r. We generate a context $\mathbf{c}$ based on the input sequence $\mathbf{h}$ and the query $\mathbf{y}$ through alignment $\boldsymbol{\alpha}$ between elements in the input sequence and their radial position. First, the input sequence $\mathbf{h}$ and positional query $\mathbf{y}$ are projected by matrices $W_Q \in \mathbb{R}^{C\times D}$ and $W_K \in \mathbb{R}^{C\times D}$ to the corresponding representations Q and K:

$$Q(\mathbf{y}_i) = \mathbf{y}_i W_Q, \quad K(\mathbf{h}_i) = \mathbf{h}_i W_K. \tag{1}$$

Following common terminology, we refer to Q and K as 'queries' and 'keys' respectively. After projection, an unnormalized alignment score $e_{i,j}$ is produced between each memory-query combination using the scaled-dot product [Vaswani *et al.*2017]:

$$e_{i,j} = \frac{\langle Q(\mathbf{y}_i), K(\mathbf{h}_j)\rangle}{\sqrt{D}}. \tag{2}$$

The energy scalars are then normalized using a $\mathrm{softmax}$ to produce a probability distribution over the memory:

$$\alpha_{i,j} = \frac{\exp(e_{i,j})}{\sum_{k=1}^{H}\exp(e_{i,k})}. \tag{3}$$

Finally, the context is computed as a weighted sum of K:

$$c_i = \sum_{j=1}^{H} \alpha_{i,j} K(\mathbf{h}_j). \tag{4}$$

Generating the context this way allows each radial slot r_i to independently gather relevant information from the image column; and represents an initial assignment of components from the image to their BEV locations. Such an initial assignment is analogous to lifting a pixel based on its depth. However, it is lifted to a distribution of depths and thus should be able to overcome common pitfalls of sparsity and elongated object frustums. This means that the image-context available to each radial slot is decoupled from its distance to the camera. Finally, to generate BEV feature $S_i^{\phi(BEV)}$ at radial position r_i, we globally operate on the assigned contexts for *all* radial positions $\mathbf{c} = \{c_1, ..., c_r\} : S_i^{\phi(BEV)} = g(\mathbf{c})$, where $g(.)$ is a nonlinear function reasoning across the *entire* polar ray. We describe its role below.

Polar ray self-attention: The need for the non-linear function $g(.)$ as a global operator arises out of the limitations brought about by generating each context vector c_i independently. Given the absence of global reasoning for each context c_i, the spatial distribution of features across the ray is

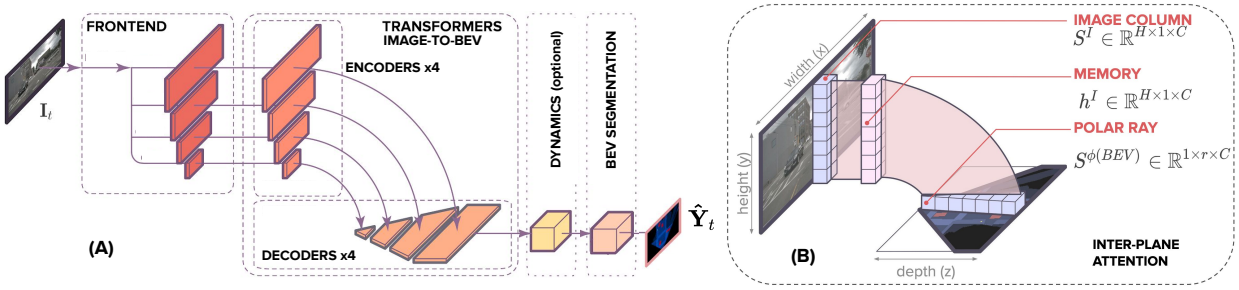

Figure 1: (A) Our model architecture. The **Frontend** extracts spatial features at multiple scales. **Encoder-decoder transformers** translate spatial features from the image to BEV. An optional **Dynamics Module** uses past spatial BEV features to learn a spatiotemporal BEV representation. A **BEV Segmentation Network** processes the BEV representation to produce multi-scale occupancy grids. (B) Our inter-plane attention mechanism. In our attention-based model, vertical scan lines in the image are passed one by one to a transformer encoder to create a 'memory' representation which is decoded into a BEV polar ray.

unlikely to be congruent with object shape, locally or globally. Therefore, we need to operate globally across the ray to allow the assigned scanline features to reason about their placement within the context of the entire ray, and thus aggregate information in a manner that generates coherent object shapes. Global computation across the polar ray is computed much like soft-attention outlined in Eq. (1) - (4), except that the self-attention is applied to the ray only. Eq. (1) is recalculated with a new set of weight matrices with inputs to both equations replaced with the context vector c_i.

Extension to transformers: Our inter-plane attention can be extended to attention between the encoder-decoder of transformers by replacing the key $K(\mathbf{h}_j)$ in Eq. (4) with another projection of the memory $\mathbf{h}$, the 'value'. Similarly, polar-ray self-attention can be placed within a transformer-decoder by replacing the key in Eq. (4) with a projection of the context c_i to represent the value.

3.2 Model Architecture

We build an architecture that facilitates our goal of predicting a semantic BEV map from a monocular image around this alignment model. As shown in Fig. 1, it contains three main components: a standard CNN backbone which extracts spatial features in the image-plane, encoder-decoder transformers to translate features from the image-plane to BEV and finally a segmentation network which decodes BEV features into semantic maps.

2D Multi-scale feature learning in $\mathbb{P}^I$: Reconstructing an image in BEV requires representations which can detect scene elements at varying depths and scale. Like prior object detection methods [Roddick and Cipolla2020, Saha *et al.*2021], we handle this scale variance using a CNN backbone with a feature pyramid to produce feature maps $\mathbf{f}^I_{t,s} \in \mathbb{R}^{C\times h_s\times w_s}$ at multiple scales $u \in U$.

1D Transformer encoders in $\mathbb{P}^I$: This component encodes long-range vertical dependencies across the input features through self-attention, using an encoder for each scale u of features (second left block of Fig. 1a). Each scale of features $\mathbf{f}^I_{t,u}$ is first reshaped into its individual columns, creating w_u sequences of length h_u and dimension C. The U encoders each produce a memory $\mathbf{h}^I_{t,u} \in \mathbb{R}^{w_u\times h_u\times C}$.

1D Transformer decoders in $\mathbb{P}^{BEV}$: This component generates sequences of BEV features along a polar ray through attention across the encoder memory. As shown in Fig. 1, there is one transformer decoder for each transformer encoder. Every encoded image column $\mathbf{h}^I \in \mathbb{R}^{h_u\times C}$ is transformed to a BEV polar ray $\mathbf{f}^{\phi(BEV)} \in \mathbb{R}^{r_u\times C}$, where r_u is the radial distance along the ray. The U decoders each output w_u BEV sequences of length r_u along the ray, producing a polar encoding $\mathbf{f}^{\phi(BEV)} \in \mathbb{R}^{w_u\times r_u\times C}$. Finally we concatenate along the ray to obtain a single 2D polar feature map and convert to a rectilinear grid, to create our BEV representation $\mathbf{f}^{BEV}_t \in \mathbb{R}^{C\times Z\times X}$.

Dynamics with axial attention in $\mathbb{P}^{BEV}$: This optional component (Fig. 1a) incorporates temporal information from past estimates to build a spatiotemporal BEV representation of the present using axial-attention.

Segmentation in $\mathbb{P}^{BEV}$: To decode our BEV features into semantic occupancy grids, we adopt a convolutional encoder-decoder structure used in prior segmentation networks [Yu *et al.*2018, Saha *et al.*2021]. The aggregated module structure (right block of Fig. 1a), takes BEV features $\mathbf{f}^{BEV}_t \in \mathbb{R}^{C\times Z\times X}$ and outputs occupancy grids $\mathbf{m}^{BEV}_{t,u} \in \mathbb{R}^{classes\times x_u\times z_u}$ for scales $u \in U$.

Loss in $\mathbb{P}^{BEV}$: As the training signal provided to the predicted occupancy grids must resolve both semantic and positional uncertainties, we use the same multi-scale Dice loss as [Saha *et al.*2021]. At each scale u, the mean Dice Loss across classes K is:

$$\mathcal{L}^u = 1 - \frac{1}{|K|}\sum_{k=1}^{K}\frac{2\sum_i^N \hat{y}_i^k y_i^k}{\sum_i^N \hat{y}_i^k + y_i^k + \epsilon}, \tag{5}$$

where y_i^k is the ground truth binary variable grid cell, $\hat{y}_i^k$ the predicted sigmoid output of the network, and ϵ is a constant used to prevent division by zero.

4 Experiments and Results

We compare our approach to current state-of-the-art approaches on the **nuScenes [Caesar *et al.*2020], Argoverse**

Method	Drivable	Crossing	Walkway	Carpark	Bus	Bike	Car	Cons.Veh.	Motorbike	Trailer	Truck	Ped.	Cone	Barrier	Mean
VPN [Pan *et al.*2020]	58.0	27.3	29.4	12.3	20.0	4.4	25.5	4.9	5.6	**16.6**	17.3	7.1	4.6	10.8	17.5
PON [Roddick and Cipolla2020]	60.4	28.0	31.0	18.4	20.8	9.4	24.7	12.3	7.0	**16.6**	16.3	8.2	5.7	8.1	19.1
STA-S [Saha *et al.*2021]	71.1	31.5	32.0	28.0	22.8	14.6	34.6	10.0	7.1	11.4	18.1	7.4	5.8	10.8	21.8
Our Spatial	**72.6**	**36.3**	**32.4**	**30.5**	**32.5**	**15.1**	**37.4**	**13.8**	**8.1**	15.5	**24.5**	**8.7**	**7.4**	**15.1**	**25.0**
STA-ST [Saha *et al.*2021]	70.7	31.1	32.4	**33.5**	29.2	12.1	36.0	12.1	**8.0**	13.6	22.8	8.6	6.9	14.2	23.7
Our Spatiotemp.	**74.5**	**36.6**	**35.9**	31.3	**32.8**	**14.7**	**39.7**	**14.2**	7.6	**13.9**	**26.3**	**9.5**	**7.6**	**14.7**	**25.7**

Table 1: IoU(%) on the nuScenes validation split and baseline results of [Roddick and Cipolla2020].

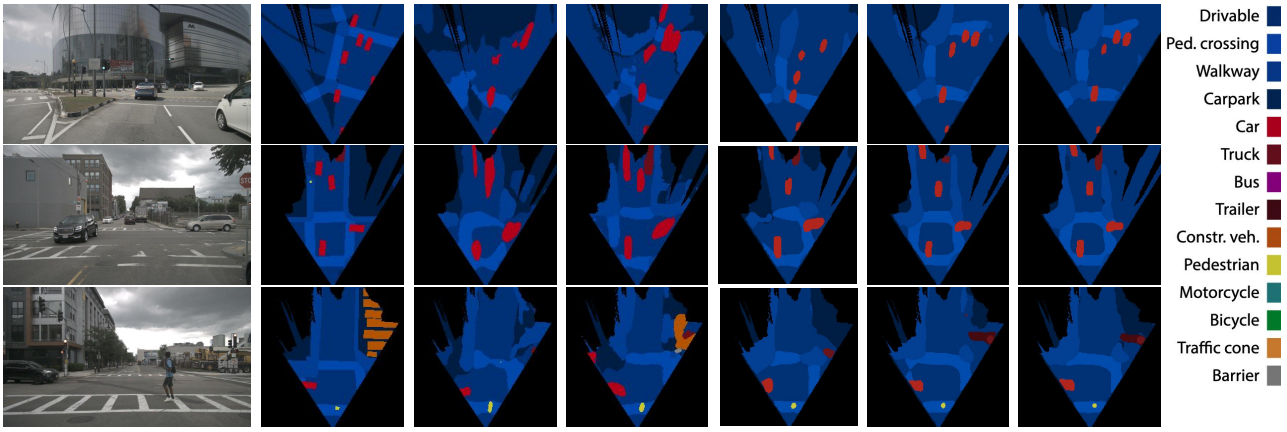

Figure 2: Qualitative results on the nuScenes validation set of [Roddick and Cipolla2020].

[Chang *et al.*2019] and **Lyft [Kesten *et al.*2019]** datasets.

Implementation: Our frontend uses a pretrained ResNet-50 [He *et al.*2016] with a feature pyramid [Lin *et al.*2017] on top. BEV feature maps built by the transformer decoder have a resolution of 100×100 pixels, with each pixel representing 0.5m^2 in the world. Our spatiotemporal model takes a 6Hz sequence of 4 images, where the final frame is the time step we make the prediction for. We train our network end-to-end with an Adam optimizer, batch size 8 and initial learning rate of $5e{-}5$, which we decay by 0.99 every epoch for 40 epochs.

Results: We begin our comparison against 'compression' approaches [Roddick and Cipolla2020, Saha *et al.*2021] on nuScenes and Argoverse using the train/val splits of [Roddick and Cipolla2020]. We then compare against the 'lift' approach of [Philion and Fidler2020, Hu *et al.*2021] on nuScenes and Lyft. In Table 1, our spatial model outperforms the current state-of-the-art compression approach of STA-S [Saha *et al.*2021]. It is the smaller dynamic classes in particular on which we show significant improvement. This is supported by our qualitative results in Fig. 2. Our results on the Argoverse dataset in Table 2 demonstrate similar patterns, where we improve upon PON [Roddick and Cipolla2020] by a relative 30%. In Table 3 we outperform LSS [Philion and Fidler2020] and FIERY [Hu *et al.*2021] on nuScenes and Lyft (FIERY [Hu *et al.*2021] uses the 'lift' approach of [Philion and Fidler2020]). One of the avenues for future work is improving localisation accuracy for distant objects. Finally, our approach is easily transferrable to indoor mobile robotics applications once ground truth has been collected to train the models.

	Driv.	Veh.	Ped.	L.Veh.	Bic.	Bus.	Trail.	Mot.	Mean
PON [Roddick and Cipolla2020]	65.4	31.4	**7.4**	11.1	3.6	11	0.7	5.7	17.0
Ours	**75.9**	**35.8**	5.7	**14.9**	**3.7**	**30.2**	**12.2**	2.6	**22.6**

Table 2: IoU(%) on the Argoverse validation split of [Roddick and Cipolla2020].

	nuScenes			Lyft		
	Driv.	Car	Veh.	Driv.	Car	Veh.
(S) LSS	72.9	32.0	32.0	-	43.1	44.6
(S) FIERY	-	37.7	-	-	-	-
(S) Ours	**78.9**	**39.9**	**38.9**	**82.0**	**45.9**	**45.4**
(ST) FIERY	-	39.9	-			
(ST) Ours	**80.5**	**41.3**	**40.2**			

Table 3: IoU(%) for spatial (S)/spatiotemporal (ST) methods.

5 Conclusion

We proposed a novel use of transformer networks to map from images and video sequences to an overhead map or bird's-eye-view of the world. We combine our physical-grounded and constrained formulation, with ablation studies that make use of progress in monotonic attention to confirm our intuitions whether context above or below a point is more important for this form of map generation. Our novel formulation obtains state-of-the-art results for instantaneous mapping of three well-established datasets.

Acknowledgements

This project was supported by the EPSRC project ROSSINI (EP/S016317/1) and studentship 2327211 (EP/T517616/1).

References

[Bahdanau *et al.*, 2015] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[Caesar *et al.*, 2020] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11621–11631, 2020.

[Carion *et al.*, 2020] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*, 2020.

[Chang *et al.*, 2019] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, et al. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8748–8757, 2019.

[Chen *et al.*, 2016] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2147–2156, 2016.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, 2019.

[Dosovitskiy *et al.*, 2021] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[Hu *et al.*, 2021] Anthony Hu, Zak Murez, Nikhil Mohan, Sofía Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. FIERY: Future instance segmentation in bird's-eye view from surround monocular cameras. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[Kehl *et al.*, 2017] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1521–1529, 2017.

[Kesten *et al.*, 2019] R Kesten, M Usman, J Houston, T Pandya, K Nadhamuni, A Ferreira, M Yuan, B Low, A Jain, P Ondruska, et al. Lyft level 5 av dataset 2019. *urlhttps://level5. lyft. com/dataset*, 2019.

[Lin *et al.*, 2017] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[Liu *et al.*, 2020] Buyu Liu, Bingbing Zhuang, Samuel Schulter, Pan Ji, and Manmohan Chandraker. Understanding road layout from videos as a whole. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4414–4423, 2020.

[Lu *et al.*, 2019] Chenyang Lu, Marinus Jacobus Gerardus van de Molengraft, and Gijs Dubbelman. Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks. *IEEE Robotics and Automation Letters*, 4(2):445–452, 2019.

[Mani *et al.*, 2020] Kaustubh Mani, Swapnil Daga, Shubhika Garg, Sai Shankar Narasimhan, Madhava Krishna, and Krishna Murthy Jatavallabhula. Monolayout: Amodal scene layout from a single image. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 1689–1697, 2020.

[Mousavian *et al.*, 2017] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3d bounding box estimation using deep learning and geometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7074–7082, 2017.

[Palazzi *et al.*, 2017] Andrea Palazzi, Guido Borghi, Davide Abati, Simone Calderara, and Rita Cucchiara. Learning to map vehicles into bird's eye view. In *International Conference on Image Analysis and Processing*, pages 233–243. Springer, 2017.

[Pan *et al.*, 2020] Bowen Pan, Jiankai Sun, Ho Yin Tiga Leung, Alex Andonian, and Bolei Zhou. Cross-view semantic segmentation for sensing surroundings. *IEEE Robotics and Automation Letters*, 2020.

[Philion and Fidler, 2020] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Proceedings of the European Conference on Computer Vision*, 2020.

[Poirson *et al.*, 2016] Patrick Poirson, Phil Ammirato, Cheng-Yang Fu, Wei Liu, Jana Kosecka, and Alexander C Berg. Fast single shot detection and pose estimation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 676–684. IEEE, 2016.

[Roddick and Cipolla, 2020] Thomas Roddick and Roberto Cipolla. Predicting semantic map representations from images using pyramid occupancy networks. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[Roddick *et al.*, 2019] Thomas Roddick, Alex Kendall, and Roberto Cipolla. Orthographic feature transform for

monocular 3d object detection. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2019.

[Saha *et al.*, 2021] Avishkar Saha, Oscar Mendez, Chris Russell, and Richard Bowden. Enabling spatio-temporal aggregation in birds-eye-view vehicle estimation. In *Proceedings of the International Conference on Robotics and Automation*, 2021.

[Schulter *et al.*, 2018] Samuel Schulter, Menghua Zhai, Nathan Jacobs, and Manmohan Chandraker. Learning to look around objects for top-view representations of outdoor scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 787–802, 2018.

[Sengupta *et al.*, 2012] Sunando Sengupta, Paul Sturgess, L'ubor Ladickỳ, and Philip HS Torr. Automatic dense visual semantic mapping from street-level imagery. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 857–862. IEEE, 2012.

[Simonelli *et al.*, 2019] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1991–1999, 2019.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[Wang *et al.*, 2019] Ziyan Wang, Buyu Liu, Samuel Schulter, and Manmohan Chandraker. A parametric top-view representation of complex road scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10325–10333, 2019.

[Yu *et al.*, 2018] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2403–2412, 2018.