Bird's Eye View Projection Networks for Self-Driving Obstacle Tracking

Andrii Dobroshynskyi [1] **Anuj Menta** [1] **Cole Smith** [1]

Abstract

In this paper, we explore different approaches for Bird's Eye View projection of surrounding obstacles in self-driving scenarios. We propose two distinct generative models for capturing top-down bounding boxes of surrounding cars and other objects, a Two-Head Semantic Grid Encoder-Decoder (ED) and a Residual Variational Autoencoder. Of these models, the Semantic Grid model outperforms the method of variational binary-mask reconstruction for object detection, and road detection tasks.

1. Introduction

To successfully navigate the surroundings and perform correctly as well as safe actions, autonomous vehicle driving systems require a good representation of their current true environment. It is usually extremely important to extract detailed information from the surrounding scenes to make reasonable decisions. One particular approach for creating a spatial scene representation to aid the system in navigation is creating birds-eye-view (BEV) maps that capture the objects and environment boundaries around the vehicle that are important for recognizing danger and for taking action.

Constructing a BEV map is useful as most of the directional control that is done by the system will most likely revolve around navigation on the ground plane (i.e. forward-backward and left-right navigation), but the challenge is to accurately create such maps from monocular images alone. When approaching this from a task of learning useful representations, the task entails understanding the road layout and concepts such as turns and intersections, as well as active objects in the scene such as cars around the vehicle.

In this work we outline and describe the approaches we attempted for the competition task of predicting the BEV road map layout as well as the positional coordinates of bounding boxes representing the objects present in the scene. We show some results and visualizations of the representations that the individual networks can learn from a set of 6 monocular images taken from the view of our ego vehicle.

[1]Courant Institute of Mathematical Sciences, New York University, New York, New York, USA. Correspondence to: Andrii Dobroshynskyi <andrii.d@nyu.edu>, Anuj Menta <menta@nyu.edu>, Cole Smith <css@nyu.edu>.

2. Related Works

There have been several works and approaches to the problem of predicting the top-down view representations from images alone. From a coordinate mapping perspective, (Roddick et al., 2018) introduced an orthographic feature transform for more accurate reasoning about surrounding 3D space from images where scales are correct, and (Roddick & Cipolla, 2020) showed that using pyramid occupancy networks in addition to a series of spatial transformers can generate good results for predicting the presence of objects, by transforming segments of road in Cartesian coordinates and stitching them together for a complete representation.

Learning to segment the scene by encoding semantic meaning for objects has been attempted by (Lu et al., 2018) who used variational encoder-decoders to predict semantic occupancy grids for 4 object classes at pixel level from a single front-facing image taken from the vehicle. (Pan et al., 2019) applied a similar approach and proposed a View-Parsing Network (VPN) to aggregate the scene surround information from multiple angles and make predictions for semantic occupancy at the pixel level.

Finally, more generative model approaches have shown very promising results with (Zhu et al., 2018) proposing a Bridge-GAN for generating a BEV view via an intermediate homography view, and more recently, (Mani et al., 2020), who also treat the scene layout as a multi-channel semantic occupancy grid, proposed MonoLayout that can predict the BEV scene layout from a single image as well as "hallucinate" objects that may be occluded in the input image. In this work we incorporate various ideas for treating the scene as a semantic occupancy map and attempt various approaches to mapping the 6-image input into a road and object BEV map.

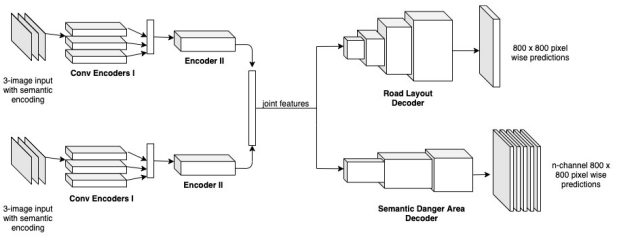

Figure 1. Two-head semantic encoder-decoder taking images split by front and back then reconstructing into a BEV roadmap layout and a BEV semantic occupancy predictions for the presence of objects

3. Two-Head Semantic Grid Encoder-Decoder (ED)

In this section we describe a model architecture of a mirrored two-branch or "head" encoder-decoder for the learning of both spatial road features as well as semantic occupancy maps by reconstructing areas of "high danger". The architecture allows for simultaneous training for the two tasks end-to-end, as well as splitting the two branches into separate sub models for the road BEV reconstruction and the semantic occupancy predictions also via reconstructions and training individually.

3.1. Architecture

The architecture proposed is a two-branch convolutional encoder that splits the six input images into a front-facing set of 3 images and a back-facing the set of 3 images. These are fed through a series of 2×2 encoder heads each for feature extraction and after each head the output representations are concatenated. The final concatenated deep feature map from both heads are then fed into a two-branched decoder that reconstructs the output via a series of convolutions and deconvolutions generating per-pixel probabilities for road vs. non-road occupancy, as well as the per-pixel predictions for rough areas matching the semantic encoding. Figure 1 shows the proposed architecture combined into a branched network.

Road map layout estimation: The road layout task is treated as a binary classification problem and the encoder-decoder is trained to reconstruct the road pixel-wise and is evaluated via binary cross-entropy as the loss function. The idea behind using two main heads for feature extraction is the idea that generally the objects that the car can perceive of behind it (back 3 images) versus the objects in front are likely not related for the roadmap reconstruction task - for instance, useful information about the positioning of a roundabout ramp or intersection is likely to come from front-facing cameras.

Bounding box prediction: The proposed architecture treats the presence of objects in the scene as the presence of "high danger" areas that are semantically encoded into the input images. To reconstruct quality BEV maps the model is required to learn complicated concepts of depth which have been shown difficult with only monocular images. The idea of the model is then to represent information about bounding boxes of various objects with a semantic encoding that creates our input images.
Input images can be interpreted as grids split into reasonably sized boxes forming such a grid since it is likely that objects in the scenes generally adhere to similar shape and size (i.e. cars, buses, bikes). Information about the objects in the scene can then be encoded by generating a 1 channel image of input size with each pixel representing a category of the object that is present here. Since we are training the model to predict high danger *areas*, we can relax the exact coordinates of the objects and encode a larger area around the grid cell where the object is located, which helps our decoder capture the positioning.

Using an encoder-decoder architecture we then can train the semantic occupancy branch or subnetwork to reconstruct the various areas of danger via a similar mechanism as the road map binary prediction as we can control the output of the decoder to be a $n \times W \times H$ size output for various danger area object categories. At inference, using the output of the model for the danger area predictions we then can set a threshold and do post-processing by predicting bounding boxes in the grid cells that are predicted as having a high chance of being a danger area. This model can also be trained end-to-end with cross-entropy and has shown good convergence.

4. Residual Variational Network

We propose a multi-headed Variational Autoencoder using residual networks, with the overall goal of the network to generate valid reconstructions of both raw input images and segmentation masks. In this way we train the network in a semi-supervised manner. A prior upon the encoder is formed by optimizing the network to reconstruct its raw input images. Using this prior, we then fine-tune the encoder in a supervised manner with distinct decoder networks to generate segmentation masks for the road layout and objects.

4.1. Architecture

4.1.1. Encoder

A summary of the architecture is outlined in figure 2. The encoder architecture using a ResNet-18 network. The output of this network is a vector of length 512, which is then projected to an encoded length of 1024. The encoded image

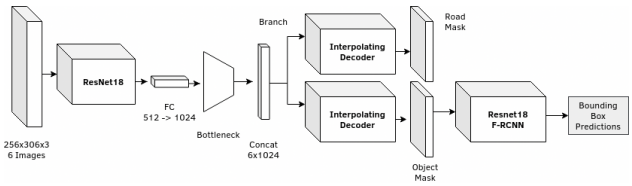

Figure 2. The Res-Var network (1) intakes a single-image though its encoder a total of six times. (2) We then use a variational bottleneck and (3) branch into two replicated decoders to generate the two segmentation masks. (4) The object mask is given to F-RCNN, and the bounding-box gradients pass through into the encoder.

is then passed through a fully-connected Bottleneck layer and reparameterized into a latent space of size 512. This emits the three variables: z, μ, and $log(var)$.

In our experiments, each segmentation mask for the road and objects share the same encoder. We found that this improves reconstruction as compared to using a different encoder instance per output map.

4.1.2. Decoder

Each of the six images taken around the ego-car is encoded, and their latent representations are stacked before decoding. The decoder up-samples the latent representation to a 2D segmentation mask of size $(1, 400, 400)$ using a series of interpolation and 2D convolutions. We use an up-sampling technique proposed in (Odena et al., 2016), by performing bilinear interpolation on the intermediate feature maps, and progressively down-sampling to the required output shape. According to (Odena et al., 2016), this helps to avoid checkerboard patterns common with deconvolution approaches. The final output shape is then interpolated to a shape of $(1, 800, 800)$ using the nearest pixel.

We use one decoder network for each map output. Originally, we used the same decoder base for each output map with only replicating the last convolutional layer, but we found that this approach led to the poor reconstruction of the object mask.

4.2. Loss

The encoder and decoder are trained end-to-end using a class-weighted pixel-wise Binary Cross-Entropy loss over the map reconstructions,

$$l(x, y) = \frac{1}{N}\sum_{n=0}^{N} l_n,\ BCE(x, y) = l(x, y)$$

$$l_n = -w_n[y_n \cdot logx_n + (1 - y_n) \cdot log(1 - x_n)]$$

Figure 3. BEV road layout reconstruction with the semantic encoder-decoder. Ground truth (left) vs. reconstruction (right)

To generate the encoder priors, we reconstruct singular input images using pixel-wise Mean Squared Error over RGB targets and reconstructions,

$$l_n = (x_n - y_n)^2,\ MSE(x, y) = l(x, y)$$

For each loss, we minimize the Kullback-Lieber Divergence over μ and $log(var)$ to enforce a Gaussian latent space. The contribution of the KL-Divergence is originally zero on the first training iteration and ramps linearly up to one after 500 iterations, or about 1 epoch of training.

$$l_n = y_n \cdot (logy_n - x_n),\ KLD(x, y) = l(x, y)$$

After the first epoch, we then introduce the bounding-box regression loss from the ResNet-18 Faster-RCNN network. We ignore the classification loss, as our segmentation mask is binary, and its reconstruction loss is already included. The gradients from this network flow through the entire network, into the shared encoder. Our hypothesis for doing so is that *these gradients will elicit better object-mask reconstructions from the decoder*.

Our final supervised loss after 1 Epoch is defined as,

$$L_{obj} = BCE(x_o, \hat{x_o}) + KLD(\mu_o, log(var)_o)$$

$$L_{road} = BCE(x_r, \hat{x_r}) + KLD(\mu_r, log(var)_r)$$

$$Loss = L_{obj} + L_{road} + L_{box}(y_o, \hat{y_o})$$

5. Results

5.1. Semantic Occupancy Grid and Road Reconstruction

The model was able to achieve around **.74** threat score for the road layout task and average around **0.003** for the object bounding box prediction task. Figure 3 shows some visualized results on the unlabeled dataset for the road map layout prediction (unseen data), and figure 4 shows a sample

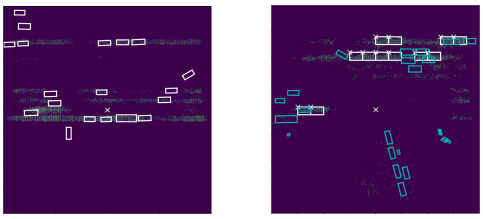

Figure 4. Scene occupancy prediction with the semantic encoder-decoder predicting areas of danger (left) and example bounding box at inference (right)

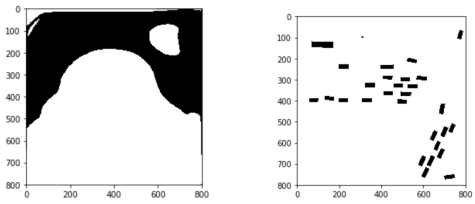

Figure 5. Object Segmentation Map reconstruction (left) and Ground Truth Map (right). Object Segmentation Map was made binary using a threshold of 0.03.

visualization of the network's predictions for danger areas per the semantic grid as well as an example of predicting bounding boxes at inference with a simple naive method based on separating the image into a grid. We can see that the encoder architecture and the decoder can capture the road map layout quite well including things such as an intersection that may not necessarily be in the car's direct field-of-view.

For the bounding box prediction task, the network struggled to get precise predictions/reconstructions for the areas that a specific object is expected to be located in. However, the general areas did seem to be predicted correctly, in other words, the network was able to learn to recognize groups of cars as danger areas and predict per-pixel accordingly. After thresholding and drawing the bounding boxes, we can see that we get correct areas, so with more fine-tuning of the grid-based semantic prediction method, we believe even better and more localized predictions can be made.

5.2. Variational Reconstruction

Our single-image reconstruction results from the Variational Autoencoder proved to be too blurry for useful results. While we used this exercise as only a pre-training task, the supervised mask reconstructions suffered from a similar issue.

In figure 5, the bound box mask reconstruction is compared to its ground-truth. The reconstruction is a 1-dimensional 800x800 image, which is processed through a Sigmoid function to constrain the range of the pixel values to $[0, 1]$. In 5, the binary mask is created by thresholding the pixel values at 0.03. We find that this model can capture *regions* of higher obstacle probability, but we hypothesize the blurry results exhibited in the pre-training task are an indication that the same might be happening in the reconstructed masks.

We also tried constraining the gradients of certain parts of the training: Freezing the encoder after pre-training, after 1 Epoch of supervised training, not passing through the F-RCNN gradients, and passing the F-RCNN gradients though for the full length of training, these methods did not lend different results. Overall, the segmentation masks overfit to the background class after about 5 epochs of training.

6. Source Code

You can find the full source code for running and training all the models listed in this paper at the following links

GitHub: Semantic Encoder-Decoder + other models

GitHub: Residual Variational Autoencoder

References

Lu, C., van de Molengraft, M. J. G., and Dubbelman, G. Monocular semantic occupancy grid mapping with convolutional variational encoder-decoder networks. *arXiv preprint arXiv:1804.02176*, 2018.

Mani, K., Daga, S., Garg, S., Shankar, N. S., Jatavallabhula, K. M., and Krishna, K. M. Monolayout: Amodal scene layout from a single image, 2020.

Odena, A., Dumoulin, V., and Olah, C. Deconvolution and checkerboard artifacts. *Distill*, 2016. doi: 10.23915/distill.00003. URL http://distill.pub/2016/deconv-checkerboard.

Pan, B., Sun, J., Andonian, A., Oliva, A., and Zhou, B. Cross-view semantic segmentation for sensing surroundings. *arXiv preprint arXiv:1906.03560*, 2019.

Roddick, T. and Cipolla, R. Predicting semantic map representations from images using pyramid occupancy networks, 2020.

Roddick, T., Kendall, A., and Cipolla, R. Orthographic feature transform for monocular 3d object detection. *arXiv preprint arXiv:1811.08188*, 2018.

Zhu, X., Yin, Z., Shi, J., Li, H., and Lin, D. Generative adversarial frontal view to bird view synthesis. In *2018 International conference on 3D Vision (3DV)*, pp. 454–463. IEEE, 2018.