Augmentation of a Monocular Image

Ankita Victor

Abstract

This project dealt with the of monocular images, given no other information, while maintaining realistic occlusion. An important consideration of this project was to minimize manual intervention. The approach taken was to use an unsupervised neural net that estimated depth for every pixel of the monocular image. Using this depth map, the monocular image is rendered as a 3-D image and occlusion is taken care of. While not perfect in terms of the object boundaries, the neural that has been trained on the KITTI dataset generalizes well for other kinds of images as opposed to supervised learning techniques that rely on having ground truth depth data.

1 Introduction

The goal of this project was to introduce a virtual object into a monocular image given no other cues and using as little manual intervention. What makes this problem statement challenging is that reconstructing an RGBD image, where D refers to depth, given an RGB image. Augmented Reality (AR), software typically works by deriving some real world coordinates—image registration—using different methods of computer vision, very often related to video tracking.

The first stage in any live, AR application is to detect interest points or markers from the live camera feed. This step uses a variety of feature detection methods like blob detection, edge detection or image processing methods. The second stage reconstructs a 3-D, real world coordinate system from the data obtained in the previous stage. When using markers, the scene's 3-D structure is pre-calculated—the size and orientation of the object are pre-defined with respect to the marker. If the scene is unknown, techniques like simultaneous localization and mapping (SLAM), GPS, 3D cameras, and multi camera technology are used to estimate depth.

In case of augmenting images, there is no opportunity to extract depth information directly from the image. The predecessor to this project involved billboarding specific objects with the user selecting their boundaries and then placing them at different depth. The purpose of this project was to automate the depth giving process. For the same purpose different machine learning techniques were evaluated. A key point of consideration was unsupervised learning—to allow the solution to generalize across any image.

2 Depth Extraction

Recovering 3-D depth from images is a basic problem in computer vision, and has important applications in robotics, scene understanding and 3-D reconstruction. Most

work on visual 3-d reconstruction has focused on binocular vision and on other algorithms that require multiple images, such as structure from motion. Even though humans perceive depth by seamlessly combining stereo and monocular cues, most work on depth estimation has focused on stereo-vision. Depth estimation from a single still image is a difficult task, since depth typically remains ambiguous given only local image features [1]. The following subsections describe monocular depth estimation techniques.

2.1 3-D Depth Reconstruction from a Single Still Image

The authors of this paper consider the task of 3-D depth estimation from a single still image using a supervised learning approach by collecting a training set of monocular images (of unstructured indoor and outdoor environments which include forests, sidewalks, trees, buildings, etc.) and their corresponding ground-truth depth maps. They then apply supervised learning to predict the value of the depth map as a function of the image. They say that depth estimation is a challenging problem, since local features alone are insufficient to estimate depth at a point, and one needs to consider the global context of the image. They use a hierarchical, multi-scale Markov Random Field (MRF) that incorporates multi-scale local- and global-image features, and models the depths and the relation between depths at different points in the image [1].

The authors do note that the monocular cues their algorithm learns rely on prior knowledge, learned from the training set, about the environment. Thus, the monocular cues may not generalize well to images very different from ones in the training set [1]. Keeping in mind the need for generalizability, this technique was discarded.

2.2 High Speed Obstacle Avoidance using Monocular Vision and Reinforcement Learning

The authors consider the task of driving a remote control car at high speeds through unstructured outdoor environments and present an approach in which supervised learning is first used to estimate depths from single monocular images. The learning algorithm is trained either on real camera images labeled with ground-truth distances to the closest obstacles, or on a training set consisting of synthetic graphics images. The resulting algorithm is able to learn monocular vision cues that accurately estimate the relative depths of obstacles in a scene. Reinforcement learning is used to learn a control policy that selects a steering direction as a function of the vision system's output [2].

The authors use a dataset of several thousand images, each correlated with a laser range scan that gives the distance to the nearest obstacle in each direction. After training on this dataset (using the laser range scans as the ground truth target labels), a supervised learning algorithm is then able to accurately estimate the distances to the nearest obstacles in the scene. Again, the reliance on dataset makes the technique unsuitable.

2.3 Single Image Depth Estimation From Predicted Semantic Labels

The authors consider the problem of estimating the depth of each pixel in a scene from a single monocular image. They first perform a semantic segmentation of the scene and use the semantic labels to guide the 3D reconstruction. The authors state

the following advantages—by knowing the semantic class of a pixel or region, depth and geometry constraints can be easily enforced (e.g., sky is far away and ground is horizontal). In addition, depth can be more readily predicted by measuring the difference in appearance with respect to a given semantic class. For example, a tree will have more uniform appearance in the distance than it does close up. Finally, the incorporation of semantic features allows good results with a significantly simpler model than previous works [3].

The authors label pixels as one of sky, tree, road, grass, water, building, mountain, and foreground object. The first seven classes cover a large portion of background regions in outdoor scenes while the last class captures the eclectic set of foreground objects such as cars, street signs, people, animals [3]. However, the authors hand-annotate the training images with semantic class labels—supervised training which makes the unsuitable.

2.4 Make3D: Learning 3D Scene Structure from a Single Still Image

Here, the authors consider the problem of estimating detailed 3-D structure from a single still image of an unstructured environment. For each small homogeneous patch in the image, they use a Markov Random Field (MRF) to infer a set of plane parameters that capture both the 3-D location and 3-D orientation of the patch. The MRF models both image depth cues as well as the relationships between different parts of the image [4].

The algorithm attempts to segment an image into many small planar surfaces. Using a superpixel segmentation algorithm, we they attempt to find an oversegmentation of the image that divides it into many small regions (superpixels). Because we use an over-segmentation, planar surfaces in the world may be broken up into many superpixels and each superpixel is likely to (at least approximately) lie entirely on only one planar surface. For each superpixel, the MRF model then tries to infer the 3-d position and orientation of the 3-d surface that it came from [4].

The authors say that other than assuming that the environment is made up of a number of small planes, their model makes no explicit assumptions about the structure of the scene which enables the algorithm to capture much more detailed 3-D structure. However again, the MRF model is trained to estimate depths using a training set in which the ground-truth depths were collected using a laser scanner [4], which makes it unsuitable.

2.5 Unsupervised Monocular Depth Estimation with Left-Right Consistency

In this paper the authors start off by saying that most existing approaches treat depth prediction as a supervised regression problem and as a result, require vast quantities of corresponding ground truth depth data for training. Just recording quality depth data in a range of environments is a challenging problem. They replace the use of explicit depth data during training with easier-to-obtain binocular stereo footage [5].

They propose a training objective that enables a convolutional neural network to learn to perform single image depth estimation, despite the absence of ground truth depth data. Exploiting epipolar geometry constraints, they generate disparity

images by training the network with an image reconstruction loss. Their training loss enforces consistency between the disparities produced relative to both the left and right images, leading to improved performance and robustness. By posing monocular depth estimation as an image reconstruction problem, the authors solve for the disparity field without requiring ground truth depth.

Given a single image I at test time, the posed goal is to learn a function f that can predict the per-pixel scene depth, $d^* = f(I)$. Most existing learning based approaches treat this as a supervised learning problem, where they have color input images and their corresponding target depth values at training. It is presently not practical to acquire such ground truth depth data for a large variety of scenes. Even expensive hardware, such as laser scanners, can be imprecise in natural scenes featuring movement and reflections. As an alternative, they instead pose depth estimation as an image reconstruction problem during training.

The intuition is that, given a calibrated pair of binocular cameras, if a function that is able to reconstruct one image from the other can be learned, then the model has learned something about the 3D shape of the scene that is being imaged. At training time, the neural net is given access to two images I^l and I^r, corresponding to the left and right color images from a calibrated stereo pair, captured at the same moment in time.

Instead of trying to directly predict the depth, the neural net finds the dense correspondence field d^r that, when applied to the left image, would enable reconstruction of the right image, $I^l(d^r)$ or I^{*r}. Similarly, the left image is estimated given the right one, $I^{*l} = I^r(d^l)$. Assuming that the images are rectified, d corresponds to the image disparity - a scalar value per pixel that the model will learn to predict. Given the baseline distance b between the cameras and the camera focal length f, depth d^* is given from the predicted disparity as, $d^* = \frac{bf}{d}$

Figure 1: Sample Test Image

Figure 2: Depth Map

3 Adaptation

The image used for augmentation is shown in Figure 3. A pre-trained model was provided as input to momodepth_simple.py [6]. The corresponding depth map is shown in Figure 4. As can be seen, given that neural net was trained on images from the KITTI and Cityscapes dataset which contain pictures of European style streets, the model works well for an Indian road.

Figure 3: Original Image

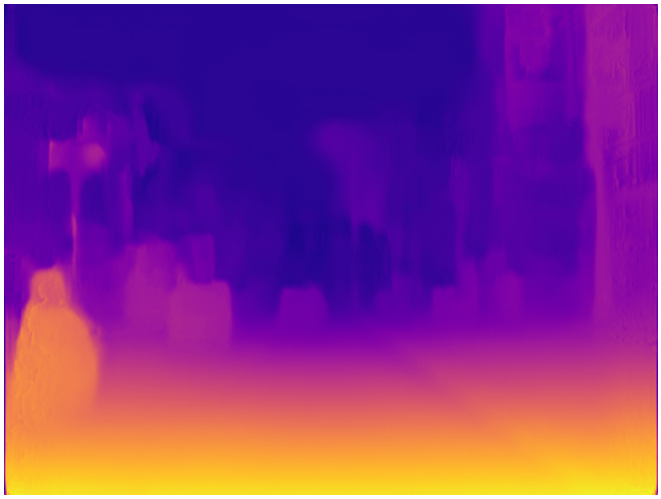

Figure 4: Output of the Neural Net

The two outputs of the model are an image and a numpy file containing the values. The latter is then fed into an OpenGL renderer as depth for every pixel. The image is displayed in 3D by rendering a texture as a set of width x height points with position (x, y, z) given by the pixel location and depth from the depth map. On rendering in 3D one can also see a ground-horizon folding happening.

Figure 5: Augmented With A Sphere

Figure 6: Augmented With A Cow

4 Bounding Boxes

To get a feel for the 3Dness of the output, image processing techniques were selected to draw 3D bounding boxes over the elements in place of supervised segmentation. The three algorithms considered were Watershed algorithm, Normalized Cuts, and Mean Shift Segmentation. The out the latter two are seen in Figures 7 and 8. The Mean Shift Segmentation algorithm gives an oversegmented output in comparsion to Normalized Cuts. The Normalized Cuts was chosen as each segment had a single color and the number of segments was fewer.

2D bounding boxes over the image were found by iterating over each pixel color and finding the minimum and maximum x and y extents (see Figure 9). This was then extended to 3D by inferring the depth map values (see Figure 10). Certain 'stray' pixels cause an incorrect fit of the 3D boxes.

Figure 7: Normalized Cut

Figure 8: Mean Shift Clustering

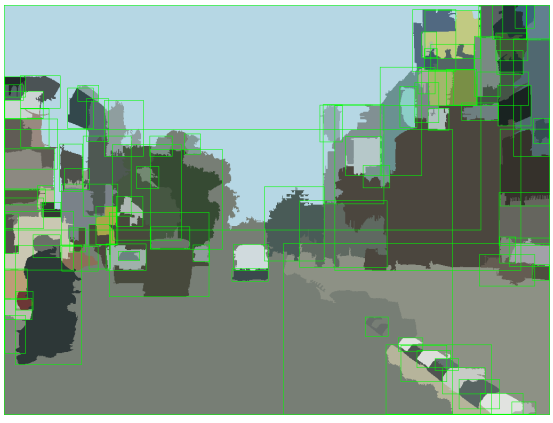

Figure 9: 2D Bounding Boxes

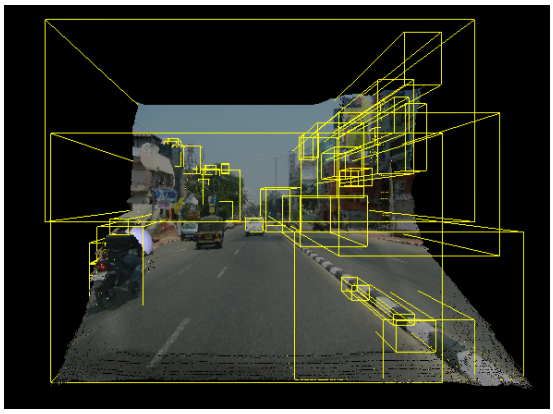

Figure 10: 3D Bounding Boxes

5 Limitations and Future Work

Image boundaries are not necessarily perfect. Considering the trade-off between manually marking out regions and supplying depth, the deep learning technique is a useful substitute. Similarly the 3D bounding boxes are dependent on values provided by the depth map.

Getting perfect object boundaries by doing some sort of superpixel oversegmentation as referred to in Make3D [4], followed by a consensus of depth values obtained using [5] of pixels in one segment can be used to 'clean up' the 3D image.

References

[1] Saxena, Ashutosh, Sung H. Chung, and Andrew Y. Ng. "3-d depth reconstruction from a single still image." International journal of computer vision 76.1 (2008): 53-69.

[2] Michels, Jeff, Ashutosh Saxena, and Andrew Y. Ng. "High speed obstacle avoidance using monocular vision and reinforcement learning." Proceedings of the 22nd international conference on Machine learning. ACM, 2005.

[3] Liu, Beyang, Stephen Gould, and Daphne Koller. "Single image depth estimation from predicted semantic labels." Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010.

[4] Saxena, Ashutosh, Min Sun, and Andrew Y. Ng. "Make3d: Learning 3d scene structure from a single still image." IEEE transactions on pattern analysis and machine intelligence 31.5 (2009): 824-840.

[5] Godard, Clément, Oisin Mac Aodha, and Gabriel J. Brostow. "Unsupervised monocular depth estimation with left-right consistency." CVPR. Vol. 2. No. 6. 2017.

[6] https://github.com/mrharicot/monodepth