

000
001
002
003
004
005
006
007
008
009
010
011
012
013
014
015054
055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107

SmallDepthMask: Large DataSet Generation for Monocular Depth Estimation and Foreground Segmentation from few Internet Images



Anonymous CVPR 2021 submission

Paper ID ****

Abstract

Segmentation of the desired object along with depth estimation is useful in various applications like robotics and autonomous navigation. Any deep learning workflow to segment the desired foreground object in a scene require significant training data. The data generation process usually involve expensive hardware like RGB-D sensors, Laser Scanners or significant manual involvement. This paper presents a novel way to utilize only a small number of readily available png images with transparency for the foreground object, and representative background images from the internet and combine them to generate a huge dataset for deep learning utilizing current state of the art monocular depth estimation and segmentation techniques. Few example applications show the efficacy of the training data on detecting cattle on road for autonomous driving application, etc. The baseline models exhibit strong generalization to real scenarios.

1. Introduction

Depth estimation and semantic segmentation are often used together in many vision tasks [10] like autonomous navigation of agents, augmented reality, self driving cars and other robotics applications. In all these application, identification of desired objects precisely in the scene and its depth estimate from the camera are crucial for safe and effective navigation. Modern RGB-D sensors like OAK-

D are capable of simultaneously running advanced neural networks while providing depth from two stereo cameras and color information from a single 4K camera in the center. Deep learning based techniques using convolution neural nets have effective solution in both depth estimation and semantic segmentation. In general for high accuracy outcomes, a deep learning network is dependent on large training dataset availability. To gather such data itself incur high cost and time. For specific applications requiring several foreground objects against variety of backgrounds become even more challenging in terms of simulating those scenarios. Synthetic datasets using Virtual Reality have been proposed to that end.

Recent research indicates effective use of readily available images on the internet to curate training data. This paper introduces SmallDepthMask, a way to curate custom dataset containing millions of images by multiplexing desired foreground objects over representative background scenes, while also generating corresponding depth and foreground mask images. This significantly reduces the cost and time overheads. The authors also experiment by creating baseline models for several application contexts and show that the generated data successfully generalizes to detect relevant objects in real scenes. Multiplexing, combined with random cropping, scaling and translation, make the data-generation fast and effective. With only 100 pair of background and foreground images, the authors generate 4 million image triplets and effectively leverage existing SOTA models for depth estimation and semantic segmentation.

- 108 The main contributions of this paper are the following:
109
110 1. A novel effort to mix and match foreground and back-
111 ground images reducing the need for complex scene
112 generation for data curation.
113
114 2. Curate large dataset to effectively train models for cus-
115 tom applications of detecting depth and mask for spe-
116 cific foreground objects over any target background,
117 from a limited input of ready available internet images.
118
119 3. Combine image, depthmap and foreground mask in a
120 single dataset using current SOTA models for depth
121 estimation and semantic segmentation.
122
123 4. To release curated dataset and the trained models mak-
124 ing them publicly available. Researchers can use this
125 single dataset to do segmentation, train models to pre-
126 dict depth, or to predict both depth and mask.

127 Figure ?? represents an example from the generated
128 dataset to help detect cattle on road, which is very common
129 on Indian roads, leading to several accidents involving loss
130 of life and property. The generated datasets and the trained
131 models are publicly available.

132 2. Related Work

133 A depth image is an image channel in which each pixel
134 relates to a distance between the image plane and the corre-
135 sponding object in the RGB image. Monocular depth gives
136 information about depth and distance and the Monocular
137 Depth Estimation is the task of estimating scene depth us-
138 ing a single image[1]. Image Segmentation is the process of
139 partitioning an image into multiple segments and it can be
140 used for locating objects, boundaries [3]. RGBD image is
141 a combination of a RGB image and its corresponding depth
142 image[19].

143 Depth information is integral to many problems in
144 robotics including mapping, localization and obstacle
145 avoidance for terrestrial and aerial vehicles, autonomous
146 navigation, and in computer vision, including augmented
147 and virtual reality[11]. RGBD datasets usually collected us-
148 ing depth sensors, monocular cameras and LiDAR scanners
149 are expensive and data collection is a time consuming job.
150 The wellknown datasets for monocular 3D object detection
151 are Context-Aware MixEd ReAlity (CAMERA), Objectron,
152 Kitty3D, Cityscape3D, Synthia, etc. [] and these datasets
153 have limitations like indoor only images, small number of
154 training examples and sparse sampling.

155 To address the issues usage of expensive devices and
156 small number of training examples, this paper proposes
157 a technique to come up with a custom dataset by us-
158 ing existing accurate depth predictor models, like High
159 Quality Monocular Depth Estimation via Transfer Learn-
160 ing(nyu.h5) [2].

161 A variety of RGBD datasets in which images are paired
162 with corresponding depth maps(D) have been proposed
163 through the years. Some of the frequently used RGBD
164 datasets are the Kitti dataset [7], the Synthia dataset [14],
165 Make3D dataset [15], NYU dataset [16]

166 The dataset Kitti [7] is the well known RGB-D dataset
167 collected using a vehicle equipped with a sparse Velodyne
168 VLP-64 LiDAR scanner and RGB cameras, and features
169 street scenes in and around the German city of Karlsruhe.
170 The Primary application of this dataset involves percep-
171 tion tasks in the context of self-driving. Synthia [14] is a
172 street scene dataset with depth maps of synthetic data, re-
173 quiring domain adaptation to apply to real world settings.
174 Cityscapes [5] provides a dataset of street scenes, albeit
175 with more diversity than KITTI. Sintel [12] is another syn-
176 thetic dataset which mainly comprises of outdoors scenes.

177 Megadepth [9] is a large-scale dataset of outdoor images
178 collected from internet, with depth maps reconstructed us-
179 ing structure-from-motion techniques, but this dataset lacks
180 in ground truth depth and scale. The RedWeb [18] dataset
181 provide depth maps generated from stereo images which are
182 freely available in large-scale data platforms such as Flickr.
183 The datasets MegaDepth and RedWeb can be easily com-
184 puted with the existing MVS methods.

185 Make3D [15] provides RGB and depth information for
186 outdoor scenes. The NYUv2 dataset [16] is widely used for
187 monocular depth estimation in indoor environments. The
188 data was collected with a Kinect RGBD camera, which pro-
189 vides sparse and noisy depth returns. These returns are
190 generally in-painted and smoothed before they are used for
191 monocular depth estimation tasks. As a result, while the
192 dataset includes sufficient samples to train modern machine
193 learning pipelines, the “ground-truth” depth does not neces-
194 sarily correspond to true scene depth.

195 Most of the existing datasets consists of indoor images,
196 or outdoor images of city streets. For every specific ap-
197 plication, like detecting animals roaming on roads for self
198 driving or assisted driving cars, or people inside a room for
199 autonomous room cleaners etc. researchers need to curate
200 specific dataset to train relevant deep learning models. The
201 present work makes the task of curating dataset extremely
202 simple and cost effective.

203 3. Method

204 The curated dataset must have following objectives:

- 205 1. dataset which dedicatedly includes foreground object.
- 206 2. dataset should drive deep learning models and gener-
207 alize.
- 208 3. dataset should provide accurate dense depth maps.
- 209 4. dataset should provide foreground mask

216 5. dataset can be stored offline or generated online during
 217 training phase dynamically
 218

219 3.1. Data Acquisition 220

221 The first step to curate data is to determine a target application scenario and thus determine the foreground object(s) and the representative background context. At the same time the dataset must have sufficient variability to include majority of the types and views that the trained deep network may see when deployed.
 222

223 We propose to download or take RGB image of n foreground object(s) and m background images (we used $n = m = 100$) balancing the types and views. For example, for cattle on roads dataset, we chose several cow, bull and calf types, individual or in group, sitting, standing or walking, and from various angles. Similarly for background, we chose backgrounds of streets, storefronts, main roads, highways, markets, railway tracks, landscapes, garbage piles etc. PNG images with transparency are readily available on the internet for almost any desired foreground object. Such images will easily allow to generate foreground mask from non-transparent pixels. If not, tools like GIMP [8], combined with deep learning foreground extractors¹ that uses a combination of Image based techniques . and DNN to separate foreground from background) can help generate the required PNG foreground images.
 224

225 Fig. 1 shows few of the sample scene and foreground images used for the creation of this dataset.
 226

227 3.2. Multiplexing and Depth Generation 228

229 This step is to place each foreground object on to several background images generating a fg-bg image and the corresponding mask corresponding to the foreground placement and scale. Depth is computed from the fg-bg image via the model proposed by Ibraheem Alhashim et al. in their paper titled "High Quality Monocular Depth Estimation via Transfer Learning" [2]². This model takes 448×448 size images as input, hence we resize all background images to this size while maintaining their aspect ratio.
 230

231 The data generation process is completely online and produces one batch of images for training a deep model. By repeating one foreground object k times for each background image and repeating another k times with horizontally flipped version of the same foreground, one can generate $2kmn$ fg-bg images. For $k = 20$ and $n = m = 100$ this becomes 400,000 fg-bg images. Algorithm 1 describes the data generation process
 232

233 ¹<https://www.remove.bg/>

234 ²source code for depth estimation model:
 235 <https://github.com/ialhashim/DenseDepth/blob/master/DenseDepth.ipynb>

236 4. Experiments 237

238 In this section, we provide a baseline for monocular depth estimation and foreground segmentation on the *SmallDepthMask* dataset. Convolutional Neural Networks(CNN) are progressive in exploring structural features and spatial image formation. To come up with baseline, we started training simple CNN, Resnet, and Unet++. The state-of-the-art models for image segmentation are variants of U-Net and fully convolutional networks (FCN)[6]. long skip connections are used to skip features from the contracting path to the expanding path in order to recover spatial information lost during downsampling [20]. Short skip connections can be used to build deep FCNs.
 239

240 4.1. Detecting Cattle on Road 241

242 DISCUSS WHY? NEED FOR THIS APPLICATION
 243 HERE...
 244

245 Previous sections describe as an example the dataset generation for this application. The dataset is available for public use. Every image (fgbg, mask, depth) in the dataset are of size 224×224 . The distribution of fgbg, mask and depth values for XYZ dataset are shown in fig. (WHAT?). The dataset has 400K records. A 70 – 30 train-test-split 280K and 120K train/test images respectively. A sample record in the dataset contains paths to all the images as shown below.
 246

```
247        ('./data/bgimages/bgimg099.jpg',
 248        './data/out2/images/fgbg392483.jpg',
 249        './data/out2/masks/mask392483.jpg',
 250        './data/out2/depth/fgbg392483.jpg')
```

251 4.2. Model 252

253 By using both long and short skip connections we proposed a light weight model following U-Net architecture with two decoder networks meant for segmentation mask prediction and depth prediction. The architecture of the model is shown in Fig 3. The total number of parameters of this model are 5,525,568 including both the decoders.
 254

255 The encoder part of network is comprised of four DownSampling units. Every downsampling unit compresses the input scene image with the help of a series of convolutional operations. In our implementation, the source image of size 128×128 , changed into $64 \times 64 \rightarrow 32 \times 32 \rightarrow 16 \times 16 \rightarrow 8 \times 8$. This model has DepthDecoder and MaskDecoder and each of them is comprised of four upsampling units. The compressed source image is expanded with the help of Atrous and Transposed convolution operations. The encoder outcome 8×8 is expanded into $16 \times 16 \rightarrow 32 \times 32 \rightarrow 64 \times 64 \rightarrow 128 \times 128$. As shown in model architecture Fig.3 the outcome of encoder downsampling units were added to outcomes of decoder upsampling units.
 256

257 We have trained this model on the entire *SmallDepthMask* dataset from scratch with train-test-split of 70 – 30%.
 258



Figure 1. Scene and foreground object images

Algorithm 1: Generate Dataset([bgimages], [fgimages], k, b)

input : m Background Image paths, 2n Foreground Image paths, multiplexing factor k, batch size b must be multiple of k

output: Yield $2kmn$ fg-bg, mask and depth images in batches of size b

for offline use it creates 3 folders with fg_bg, mask and depth each having $2kmn$ images;

for $bg \leftarrow 1$ to m do

 for $fg \leftarrow 1$ to $2n$ do

 for $i \leftarrow 1$ to k do

$croppedbg \leftarrow$ take maximal random crop of 448×448 from bg without affecting the aspect ratio;
 randomly pick a center point (x, y) in range $[0, 447]$;
 randomly pick a scale in range $[0.3, 0.6]$ (ratio of area fg covers bg);
 create $fg - bg$ image by resizing the fg to scale and place it on top of $croppedbg$ centered at x, y calculated;
 calculate binary mask from current placement of fg by thresholding transparency channel. save $fg - bg$ image and mask add $fg - bg$ image to a batch;

 if b new $fg - bg$ images generated then

 run depth model on batch and save corresponding depth images. Rescale saved images to 224×224

During training the network is trained with the batch size of 64 for 10 epochs using SGD optimizer[4]. Every epoch took one hour of time on GPU because of the huge training data. We have used OneCycleLR scheduler [17] with a maximum Learning rate of 0.1. This made the initial learning rate as 0.0099. The Deep Convolutional Neural Networks encoder is fed with a image ($128X128$) and the first decoder outputs a mask image and second decoder outputs a depth image. To reduce overfitting[13], and achieve generalization this work employed the augmentation techniques, Random Rotation, Random Grayscale, Color Jitter, random horizontal flips and random channel swaps.

4.3. Loss function

Deciding a universal loss function is not possible for complex objectives like Image segmentation and depth prediction. Based on the survey done by Shruti Jadon [?] we have picked L1 loss and SSIM(Structural Similarity Index) loss [?]. Their work also suggested to use penalty term which helps the network to focus towards hard-to-segment boundary regions. The Loss is calculated with the help of

L1 and SSIM at both the decoders and employed regularization for weight penalty.

For training our network with two decoders, we defined the same loss function L for depth and mask prediction, between y and \hat{y} as the weighted sum of two loss function values.

$$L(y, \hat{y}) = \lambda L_{term1}(y, \hat{y}) + (1 - \lambda) L_{term2}(y, \hat{y}) \quad (1)$$

The first loss term $L_{term1}(y, \hat{y})$ is the point-wise L1 loss defined on the predictions of Mask Decoder and Depth Decoder units of the network.

$$L_{term1}(y, \hat{y}) = \frac{1}{n} \sum_{x=1}^n |y_i - \hat{y}_i| \quad (2)$$

The second loss term $L_{term2}(y, \hat{y})$ uses a commonly used metric for image reconstruction task i.e., SSIM. Many recent toady depth prediction CNNs employed this metric. The loss term is redefined as shown in equation as SSIM has an upper bound of one.



Figure 2. Three images resulted from the algorithm used. (top) A scene image on which a foreground object is positioned at random location with random scale, (middle) respective mask for the scene image, and (bottom) calculated depth by using a model.

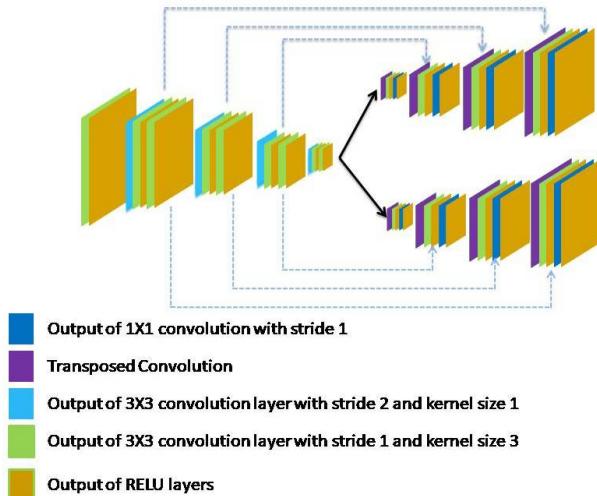


Figure 3. Network Architecture

$$L_{term1}(y, \hat{y}) = \frac{1 - SSIM(y, \hat{y})}{2} \quad (3)$$

Different weight parameters λ were tried and we have ended with a value $\lambda = 0.84$. The final loss function is as follows.

$$L(y, \hat{y}) = 0.84 * L_{term1}(y, \hat{y}) + 0.16 * L_{term2}(y, \hat{y}) \quad (4)$$

4.4. Optimizer and Learning Rate

MENTION DETAILS HERE

4.5. Results

The model is trained on the entire dataset and obtained significant accuracy and minimal loss. The outcome of the model on validation dataset is shown in Fig 4. and on the unseen data is shown in Fig 5. The unseen data fed to the model is a real picture and not one curated as in *SmallDepthMask* where foreground is placed on background.

From the obtained segmentation masks and depths, It is very clear that the model generalized well. Few exceptions like the spots on cow and two calves not having very good detection indicate the non-prediction of such examples in the training set. However, it should be straightforward to introduce few more examples to make the application more robust.

5. Conclusion and Future Work

WRITE MORE HERE HOW WE FULFILLED THE CLAIMS Robust dataset for Image Segmentation and depth generation is proposed whose implementation cost is low. A lightweight baseline model to infer both mask and depth is proposed.

Due to random scaling and placement the fg-bg images are often not so realistic. Figure 5 middle show sample images with wrong placement and scale. Regardless of this, the trained model is generalizing well. However, we experimented with detection of ground and sky regions from semantic segmentation. [ONE LINE WITH REFERENCE TO THE WORK WE USED FOR THIS] That combined with the depth information of the background gives necessary cues as to where to place the foreground image and at what scale. Figure 5 shows the outcomes. We further experimented with adding occlusions from semantic segmentation information where non ground/sky pixels that belong to regions starting below the top margin of the foreground location, are placed on top of the foreground. Figure 5 illustrates this. The effect of such data on training is yet to be explored. At the same time, the generated images by such informed scaling and placement are also not free from artifacts. The orientation and unknown size of foreground object pose a challenge, leading to use of several experimental constants in the transformation process. We would like to formalize that and analyze its outcome on training as future work.

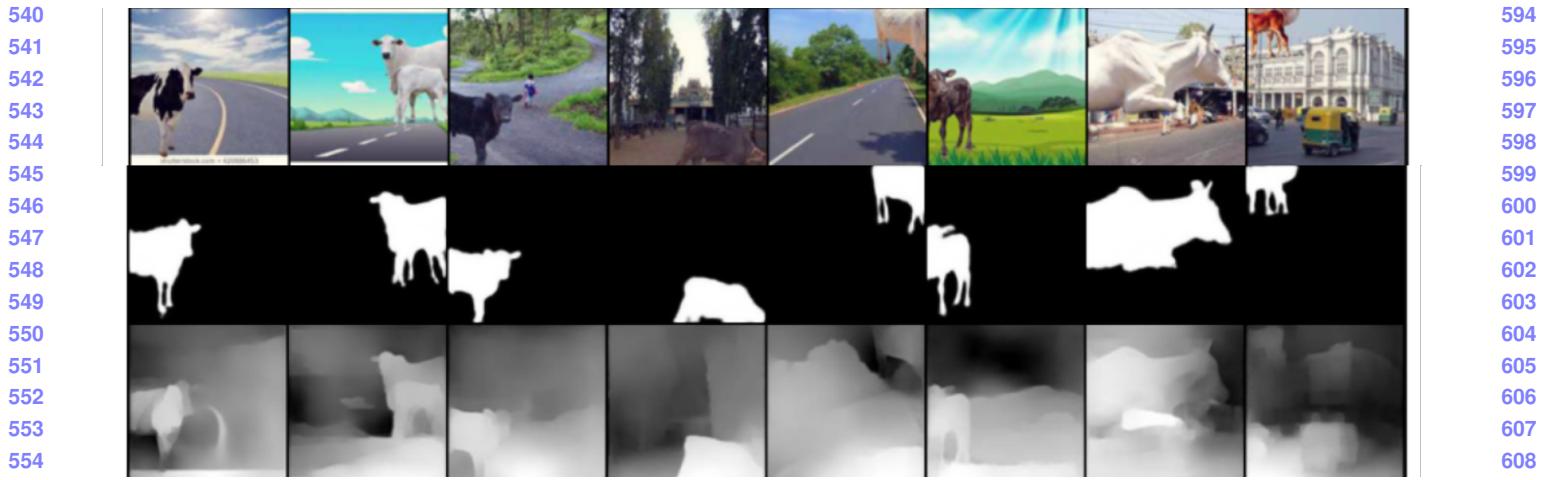


Figure 4. Segmentation mask and depth inference on validation data

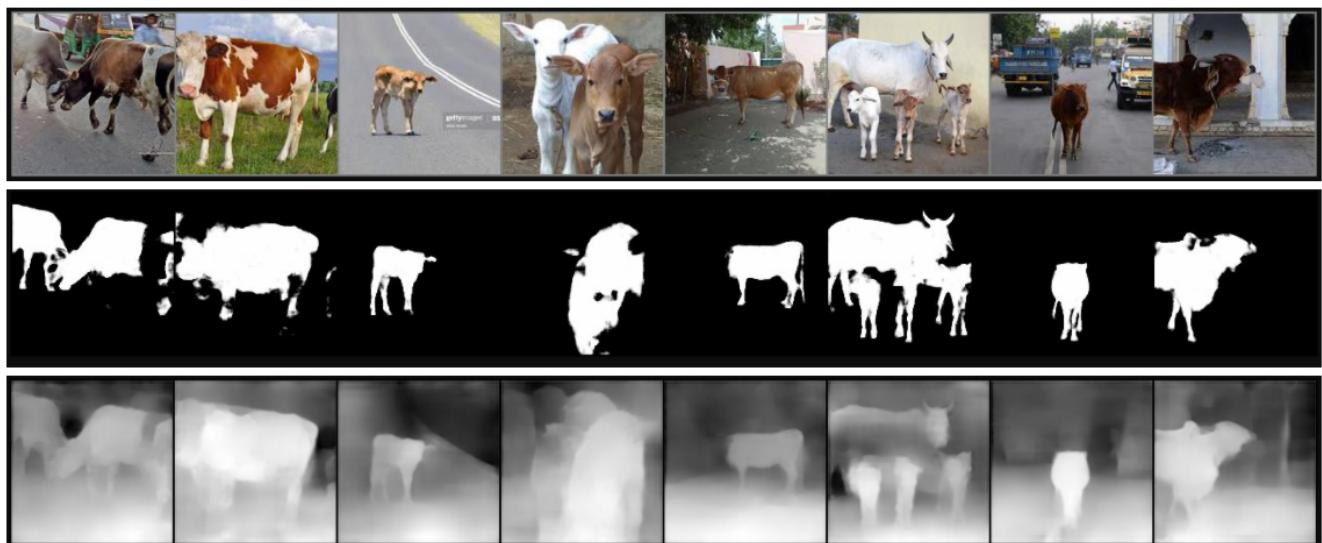


Figure 5. Segmentation mask and depth inference on unseen data.

6. Acknowledgement

Authors are grateful to Vishnu Institute of Technology for providing necessary infrastructure, and to Rohan Sravan of The School of AI for initiating the idea and providing necessary support to carry out the research.

References

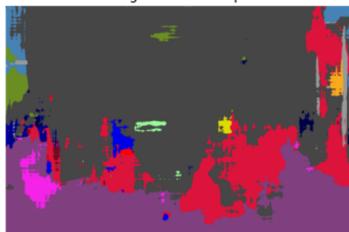
- [1] Abdullah Abuolaim and Michael S Brown. Defocus deblurring using dual-pixel data. In *European Conference on Computer Vision*, pages 111–126. Springer, 2020. [2](#)
- [2] Ibraheem Alhashim and Peter Wonka. High quality monocular depth estimation via transfer learning. *arXiv preprint arXiv:1812.11941*, 2018. [2, 3](#)
- [3] Catalin Amza. A review on neural network-based image segmentation techniques. *De Montfort University, Mechanical and Manufacturing Engg., The Gateway Leicester, LE1 9BH, United Kingdom*, pages 1–23, 2012. [2](#)
- [4] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT’2010*, pages 177–186. Springer, 2010. [4](#)
- [5] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. [2](#)
- [6] Michal Drozdal, Eugene Vorontsov, Gabriel Chartrand, Samuel Kadoury, and Chris Pal. The importance of skip connections in biomedical image segmentation. In *Deep learn-*

648

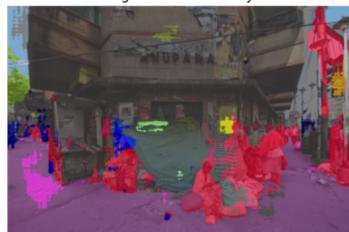
input image



segmentation map



segmentation overlay



702

649

703

650

704

651

705

652

706

653

707

654

708

655

709

656

710

657

711

658

712

659

713

660

714

661

715

662

716

663

717

664

718

665

719

666

720

667

721

668

722

669

723

670

724

671

725

672

726

673

727

674

728

675

729

676

730

677

731

678

732

679

733

680

734

681

735

682

736

683

737

684

738

685

739

686

740

687

741

688

742

689

743

690

744

691

745

692

746

693

747

694

748

695

749

696

750

697

751

698

752

699

753

700

754

701

755

Figure 6. Using Semantic Segmentation for adaptive placement and scaling. (Top) semantic segmentation. (Middle) Example of poor scaling and placement in proposed approach. (Bottom) Attempted correct placement and scaling

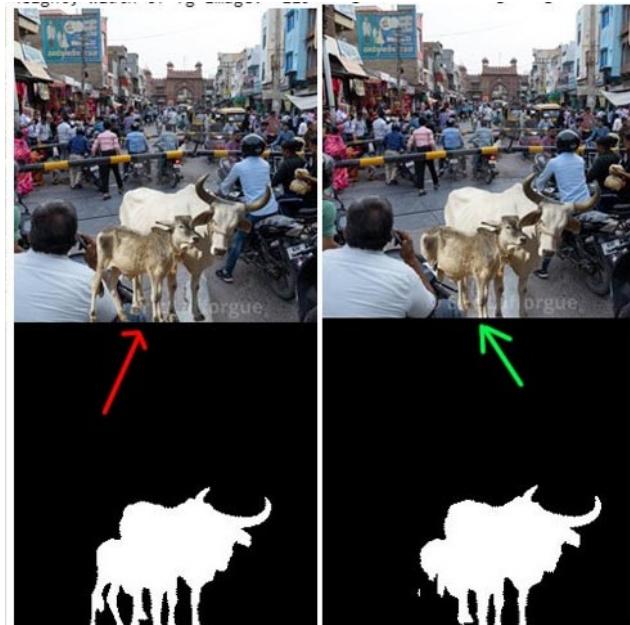


Figure 7. Restoring occlusion from semantic segmentation

ing and data labeling for medical applications, pages 179–187. Springer, 2016. 3

[7] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2

[8] Ian M Howat, A Negrete, and Benjamin E Smith. The green-

land ice mapping project (gimp) land classification and surface elevation data sets. *The Cryosphere*, 8(4):1509–1518, 2014. 3

[9] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018. 2

[10] Zhe Lin, Scott D Cohen, Peng Wang, SHEN Xiaohui, and Brian L Price. Joint depth estimation and semantic segmentation from a single image, July 10 2018. US Patent 10,019,657. 1

[11] Eric Marchand, Hideaki Uchiyama, and Fabien Spindler. Pose estimation for augmented reality: a hands-on survey. *IEEE transactions on visualization and computer graphics*, 22(12):2633–2651, 2015. 2

[12] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016. 2

[13] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017. 4

[14] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3234–3243, 2016. 2

[15] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3d: Depth perception from a single still image. In *AAAI*, vol-

- ume 3, pages 1571–1576, 2008. 2

[16] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012. 2

[17] Leslie N Smith. A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay. *arXiv preprint arXiv:1803.09820*, 2018. 4

[18] Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, Yang Xiao, Ruibo Li, and Zhenbo Luo. Monocular relative depth perception with web stereo data supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 311–320, 2018. 2

[19] Yinda Zhang and Thomas Funkhouser. Deep depth completion of a single rgbd image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 175–185, 2018. 2

[20] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: Redesigning skip connections to exploit multiscale features in image segmentation. *IEEE transactions on medical imaging*, 39(6):1856–1867, 2019. 3