

Data-driven Depth Inference from a Single Still Image

Kyunghee Kim
Computer Science Department
Stanford University
kyunghee.kim@stanford.edu

Abstract

Given an indoor image, how to recover its depth information from one single image? This problem has been studied before for many years. But previous research mainly focused on using manually designed features, heuristics, or structure information. Lacking enough training data limits the methods that can be used to deal with this problem. However, with Kinect, it is now much cheaper to get ground truth depth information for indoor images. The purpose of this project is to use a lot of training data to obtain a more data-driven approach for recovering depth information given a single image. As a result we obtained a depth image of a single RGB image from other training images and this reconstructed image showed a similar pattern in its ground truth labels.

Future Distribution Permission

The author(s) of this report give permission for this document to be distributed to Stanford-affiliated students taking future courses.

1. Introduction

Depth estimation from images and reconstruction of 3D structure of the images has been of interest to computer vision researchers for many years. Saxena et al. [1][2] used Markov Random Field (MRF) to model the depths and relation between depths at different parts of the image. Scharstein and Szeliski [3] produced a dense disparity map using two-frame stereovision. Torralba and Oliva [4] proposed a way to obtain the properties of the structure in the image from Fourier spectrum and infer the depth from this information. Saxena, Chung, and Ng [5] inferred depth from monocular image features. This project will use a MAP-MRF approach similar to [1], [2] and [6] and use massive amount of indoor images collected with Kinect [7] to infer the depth from a single image.

2. Preliminary Experiment

We will formulate the problem as an energy minimization problem as in [1], [2], [6] and [8] and before writing an energy function which consists of the unary term that models the relationship between the features in each pixel to the depth information and the pair-wise term that models the relationship between two neighboring pixels and depth information, we run some preliminary experiment to examine the properties of the unary term.

2.1 Data

The images were collected with Microsoft Kinect [7] RGB camera and depth camera that contains the indoor images with 4 scene categories i.e., office, kitchen, bedroom, and living room. We collected 200 RGB images of these indoor sceneries and each RGB image has a corresponding ground truth depth image created with the Kinect depth camera. Since the Kinect depth camera measures the depth information accurately within ~5 meters, indoor images seem to be more proper for this experiment than out door images that usually can have objects more than 5 meters away.

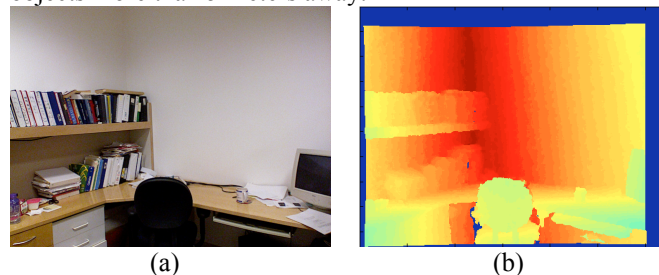


Figure 1: **One sample image.** The image on the left is an RGB image of ‘office’ scene category and the image of the right is the corresponding ground truth depth image.

2.2 Unary term experiment

Experiment Procedure In this experiment we infer the depth of a test image from training images and compare the inferred depth with the ground truth depth. We select one image among 200 images to use it as a test image and use the remaining 199 images as training images. And we

scan the test image with 3 by 3 window size patch from the first pixel of the image until the end of the image without any overlapping patch. While this window slides over the entire pixels on the test image, for each patch we select a patch of the same size i.e., 3 by 3 patch from training images that shares the most similar features with the patch from the test image. As features we use RGB color by calculating how much the RGB values in test patch are different from the path in selected among the training patch. From training images we randomly select 1000 patches for each 3 by 3 patch from the test image and finally choose only one patch among 1000 patches from training images as the best match to the test patch. We hypothesize that since these images are from similar indoor scenes if the patches have similar features in RGB images then their depth information would also share quite a lot of similarities. To compare the RGB similarity between test patch and the training patch we use the formula (1)

$$similarity = \sum_{x=1}^i \sum_{y=1}^i \sum_{C=R,G,B} (C_{test}(x,y) - C_{training}(x,y))^2 \quad (1)$$

where i by i is the size of the patch and C_{test} is the color value i.e., RGB in test patch and $C_{training}$ is the color value in training patch. The training patch is chosen to be best match when it has the minimum value of formula (1) amongst all the training patches.

Experiment result Figure 2. shows the result of this experiment. The image on the left is the ground truth depth of the test image, the same image previously shown in Figure 1 on the right side and the image on the right side of the Figure 2 is the result obtained from this experiment. This result image was generated by concatenating patches from the training images that matched the best with each patch on the test image while the 3 by 3 patch was sliding over the test image. As we can see in Figure 2 we can find the patterns of the original image in Figure 2 (a) in the reconstructed image from the experiment in Figure 2 (b). Figure 2 (b) shows the shape of the chair and table from the original image. This result gives us intuitions that the depth information of a single still image previously unseen by training images can be inferred by selecting the depth from the similar patches in training images.

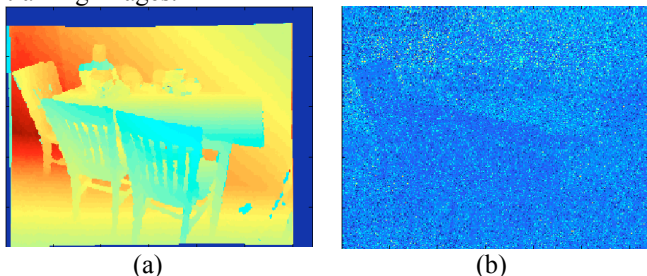


Figure 2: Ground truth depth image on the left and the result image inferred on the right.

We also use the norm-2 measurements of all the pixels in the ground truth depth image and the result image as in (2) to evaluate the result more quantitatively.

$$diff = \sqrt{\sum_{(x,y) \in G, G(x,y) \neq 0} \frac{(G(x,y) - P(x,y))^2}{\# of (x,y) \in G, G(x,y) \neq 0}} \quad (2)$$

where $G(x,y)$ means the ground truth depth at pixel (x,y) and $P(x,y)$ is the inferred depth at pixel (x,y) in the result image such as Figure 2 (b). We should notice that in (2) we are summing over all the pixels except for the ones with depth values are zeros. This is because in the ground truth depth image such as Figure 3 there exist some pixels with error marked with black circles and ellipses in the Figure 3 and also the boundary of the ground truth image is surrounded with depth zeros regardless of what the ground truth depth values are. We exclude these pixels in our formula (2) to calculate the difference between the ground truth depth and the predicted depth from the training images. For the result image in Figure 2 (b) we obtained 921.7675 as an estimate using the formula (2).

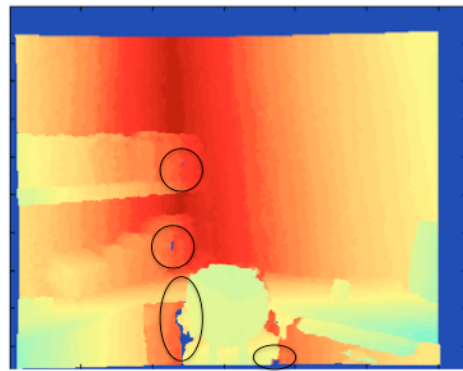


Figure 3. The pixels with error in depth information are marked with black circles and ellipses.

3. Control Experiments

From the preliminary experiment with one test image we could observe what we need to improve to make our energy minimization approach to work. We run control experiments for more test images by varying the patch size and the number of training patches. We use 3 images for each scene as test images. Since there are 4 scenes, i.e., office kitchen bedroom and living room, we use 12 images as test images. For each test image we run three kinds of experiments. In the first control experiment we use all the other 199 remaining images as training images. In the second control experiment we will use only the images in the same scene category as training images. Lastly we use the images in the different scene category as training images. We expect to see the second control experiment performed the best and the third one performs the worst. We also change the number of training patches from 1000 to 3000 whereas we used only 1000 training patches in the preliminary experiment. And we also vary the size of the

super pixels from 3 by 3 to 11 by 11 whereas we used only the 3 by 3 size patch in the preliminary experiment.

Implementation The preliminary experiment was performed with MATLAB m-files and it took 5 hours to get one result image in Figure 2 (b) for one test image. Since we want to run more experiments with more images and run several control experiments we need to speed up the code. Therefore we implement the code in mex files that make the computation more efficient when we use for-loops and it takes only 3 minutes for each test image. For each control experiment the code was implemented using mex file and the specific implementation for each experiment is explained next.

3.1 Control experiment 1: Depth inference from training images in all scene categories

Implementation We implement the first control experiment using the algorithm illustrated in Algorithm 1. The controlled parameter is the scene category. Therefore in this experiment we use all images in the training images regardless of their scene category.

Algorithm 1. Depth inference of a single still image from training patches in all scene categories

Input a still RGB image without ground truth depth information

Output inferred depth information of the input image

Algorithm

for each pixel (x,y) in the input image, make an i by i patch including neighboring pixels

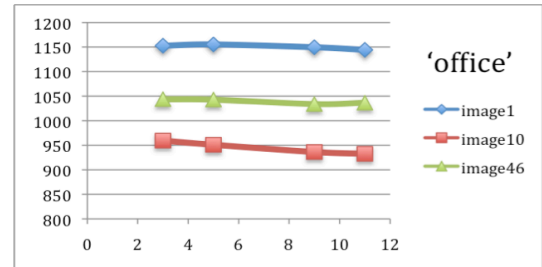
for this each patch obtained from the test image, randomly select the same size patch from training images that do not include the test image and *from all scene categories* and compare their RGB values using formula (1)

Select the best match whose similarity using formula (1) has the minimum value amongst all the training patches
Construct a depth image by concatenating the depth value of the center pixel in the training patch selected in the procedure above.

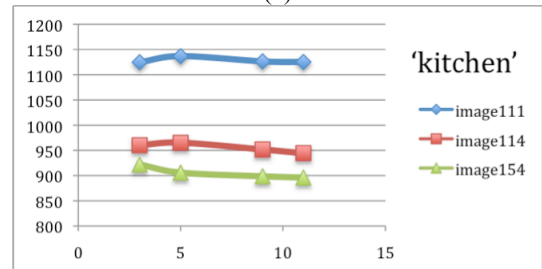
Return the constructed depth image

Experiment result In this controlled experiment, we use 3 by 3, 5 by 5, 9 by 9 and 11 by 11 as patch sizes and fix the number of training patches as 1000 in Algorithm 1 and obtain the reconstructed depth image and measure the estimate using the formula (2). The results are shown in Figure 4. In the results in Figure 4 the x-axis indicates the patch size and the y-axis indicates the estimate. We measure the estimate for each scene category using the formula (2) and the estimates of two images i.e., image1 and image10 from ‘office’ scene category tend to decrease

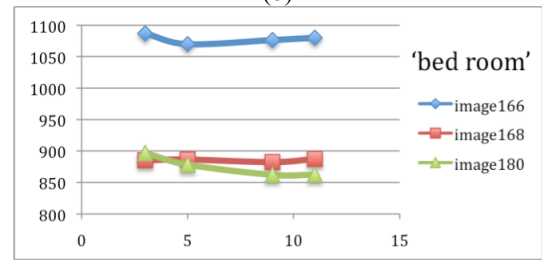
as the patch size increases, the estimates of two images i.e., image114 and image154 from ‘kitchen’ scene category tend to decrease as the patch size increases, the estimate of one image, image180 from ‘bed room’ scene category decreases and the estimates of two images, image34 and image156 from ‘living room’ scene category tends to decrease as the patch size increases.



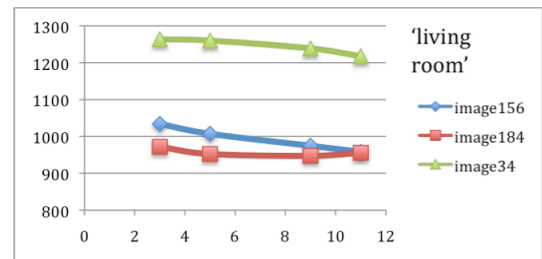
(a)



(b)



(c)



(d)

Figure 4. Estimate of each test image of each scene category by varying the patch size (a) ‘office’ scene category (b) ‘kitchen’ scene category (c) ‘bed room’ scene category (d) ‘living room’ scene category

This result agrees with our expectation since we expected that if the patch size becomes larger than the selected training patch is more similar to our test image patch and

it is more probable that it comes from the similar part in their original images and it makes their depth more similar. Therefore 7 images out of 12 test images produced the result that agrees with our expectation. But 5 images out of 12 test images were increasing or repeating to increase or decrease and did not keep decreasing as the patch size increases. So we examine these unexpected results more carefully and image 166 seems to be a good candidate to examine this property but by mistake during the experiment we did not save the reconstructed image of image 166 and we only recorded its estimate so we examine image 111 and image 46 instead which we saved the reconstructed image successfully. For example, here we show the reconstructed depth image from image 34 which produces a very good performance in its estimate measurements.

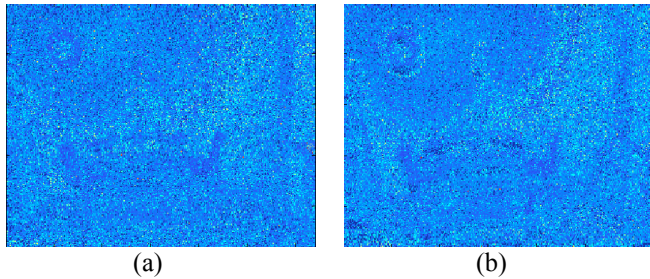


Figure 5: The reconstructed image with patch size 3 by 3 for image 34 in ‘living room’ category on the left and the reconstructed image with patch size 11 by 11 for image 34 in ‘living room’ category on the right¹

In Figure 5 we observe that for image 34 the reconstructed image with patch size 11 by 11 produces a better result which is much more similar to the ground truth depth image than the result with patch size 3 by 3. In Figure 5 (b) we can see the shape of the chair and table more clearly than Figure 5 (a). Figure 5 (b) shows even more distinguished depth information of the clock on the wall than the ground truth depth in Figure 6 (a).

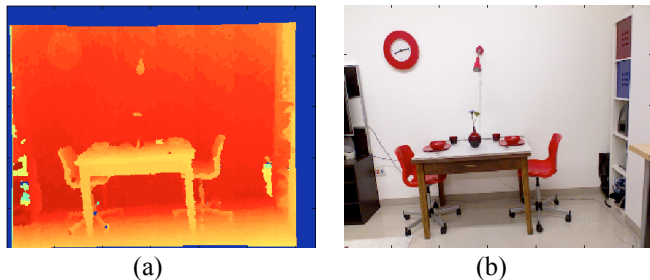


Figure 6: The ground truth depth of image 34 in ‘living room’ category on the left and the corresponding RGB image on the right.

¹ Since the figure size attached in Figure 5 is too small to observe the result in original image size we attach the bigger size image in the Appendix.

We compare this result with the estimates of image 46 which stayed almost the same for patch size 3 by 3, 5 by 5 and 11 by 11 and only slightly decreased for patch size 9 by 9. Figure 7 shows the result obtained from image 46 in ‘office’ scene category. Clockwise from Figure 7 (a) shows the reconstructed depth image from 3 by 3 patch size, 5 by 5, 9 by 9 and 11 by 11 patch size.

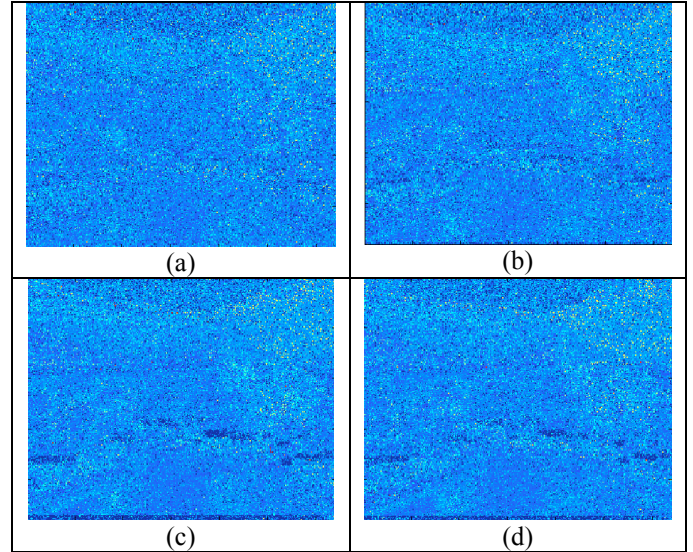


Figure 7: The reconstructed depth image for image 46 in ‘office’ category. (a) 3 by 3 patch size result (b) 5 by 5 patch size result (c) 11 by 11 patch size result (d) 9 by 9 patch size result.

Compared to the reconstructed images for image 34 previously from Figure 7 we cannot guess how the ground truth depth image would look like by looking at these result images since we cannot see any patterns here. Therefore we can conclude that regardless of the size of the patch the depth inference was not successful for this image. Figure 8 shows the ground truth depth image of image 46 and its original RGB image.

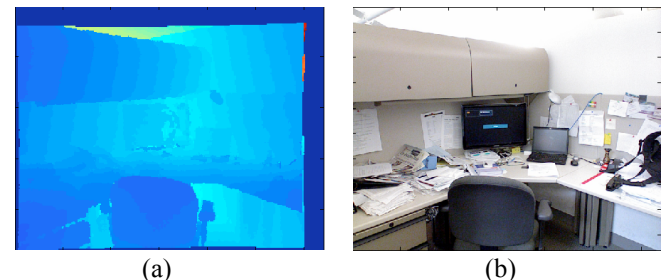


Figure 8: The ground truth depth of image 46 in ‘office’ category on the left and the corresponding RGB image on the right.

One obvious difference we can observe is that from ground truth depth image of image 46 in Figure 8 (a) we

can see that most of its pixels are blue whereas in Figure 7 (a) most of the pixels are red or yellow. Therefore one assumption could be the range of the depth affect the performance of the depth inference. When we calculated the average ground truth depth value in Figure 6(a) the mean was 2295.3 and for Figure 8(a) the mean was 1611.4. Therefore low ground depth value might not be good depth inference. However comparing two images is not enough to make this conclusion so we compare two more images in the same scene category, 'office' and for image 1 and image 10 as in Figure 9 we could observe most of yellow or red pixels for these two images as well and these two images showed decreasing tendency in its estimate when the patch size was increasing. Therefore the depth range might be the reason for the poor performance in image 46 but still comparing 4 images is too small size of images to make this conclusion. We need to further look into this property. As another assumption other features in the image such as corner or orientation of the patch might be the reason for the performance therefore it would be worthwhile to examine this property using SIFT features [10].

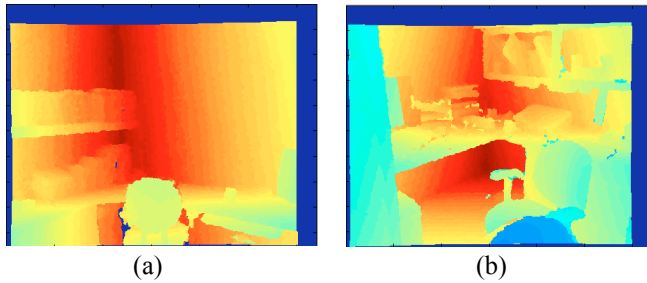
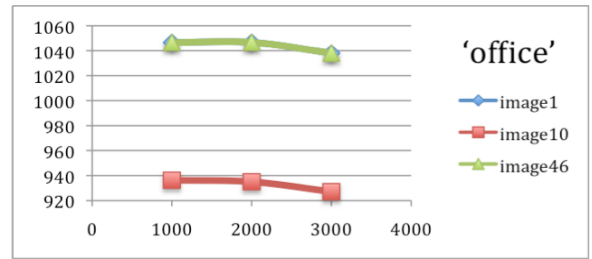


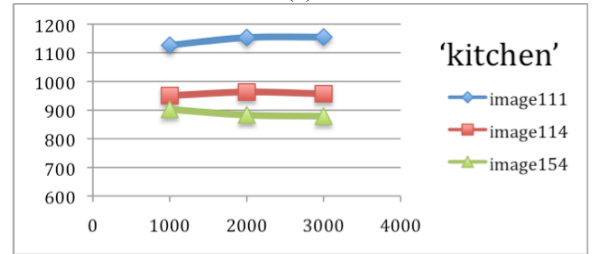
Figure 9: The ground truth depth of image 1 in 'office' category on the left and the ground truth depth of image 10 in 'office' category on the right.

We also vary the number of training patch size from 1000 to 3000 fixing the patch size as 9 by 9. We tested the number of training patch size 1000, 2000 and 3000. And the results are shown below. In the results in Figure 10 the x-axis indicates the number of training patch and the y-axis indicates the estimate. We hypothesized that the estimate will decrease as the number of training patch size increases since as there are more number of patches to be the candidate for being a good match it is more probably to get better inference for the depth. All of the three test images in 'office' scene category produce the result that agrees with our hypothesis. In 'kitchen' category the estimate of image111 is increasing as the number of training patches increases and the estimate of image114 increased slightly in the number of training patches is 2000 but it is almost constant. For 'bed room' scene category the estimate of image 168 is increasing and image 184 in 'living room' category is also increasing. But the reconstructed images for image 168 and 184 were not

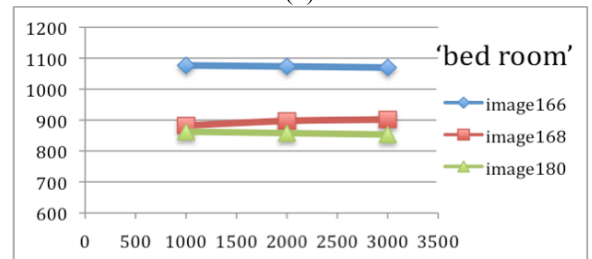
saved during the experiment by mistake and we only recorded its estimate and image111 is the one obviously



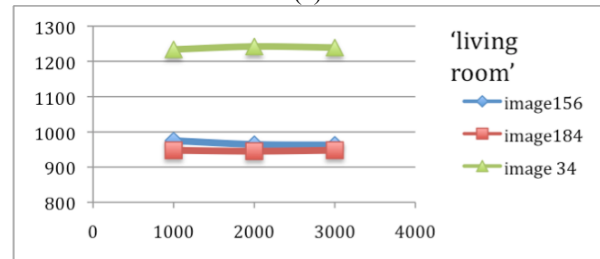
(a)



(b)



(c)



(d)

Figure 10. Estimate of each test image of each scene category by varying the number of training patches (a) 'office' scene category (b) 'kitchen' scene category (c) 'bed room' scene category (d) 'living room' scene category.

increasing as the number of training patches increase so we analyze the reconstructed depth image of the image 111 in detail. Even if we obtained decreasing estimate in Figure 10 (b) as the number of training patches increases for image 111 from Figure 11 we notice that the actual result of the reconstructed image gets better as the number of the training patches increases. When the number of the training patches is 3000 we can see the shape of the oven most clearly. This result seems to show that our estimate

calculated using formula (2) is not always robust. Although it could be a good estimate to measure the difference in the original ground truth depth and the

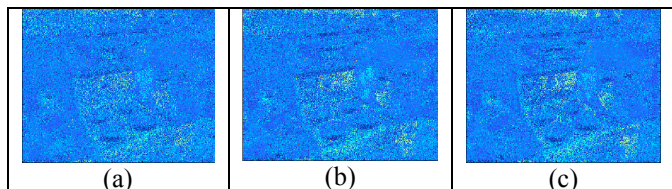


Figure 11. The reconstructed depth image for image 111 in ‘kitchen’ scene category. (a) the number of the training patches is 1000 (b) the number of the training patches is 2000 (c) the number of the training patches is 3000.²

reconstructed depth image in general it does not always give the best comparison since for example, in this case for image 111 the difference might have been large for other pixels in the regions that the inference was not good but for the regions with good inference such as the oven area, the result with the number of training patches with 3000 was the best inference result when we actually observed the reconstructed depth image.

3.2 Control experiment 2: Depth inference from training images in the same scene categories

Implementation We implement the second control experiment using the algorithm illustrated in Algorithm 1. The controlled parameter is the scene category. Therefore in this experiment we use the images in the same scene category in the training images.

Algorithm 2. Depth inference of a single still image from training patches in the same scene category

Input a still RGB image without ground truth depth information

Output inferred depth information of the input image

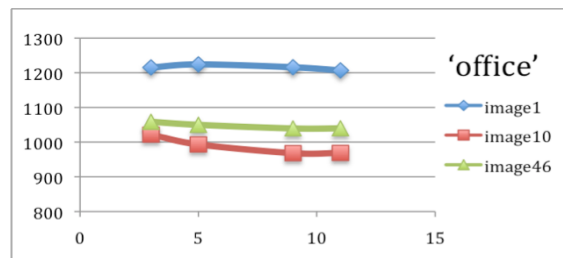
Algorithm

for each pixel (x,y) in the input image, make an i by i patch including neighboring pixels
for this each patch obtained from the test image, randomly select the same size patch from training images that do not include the test image and *from the same scene categories* and compare their RGB values using formula (2)
Select the best match whose similarity using formula (2) has the minimum value amongst all the training patches
Construct a depth image by concatenating the depth value of the center pixel in the training patch selected in the procedure above.

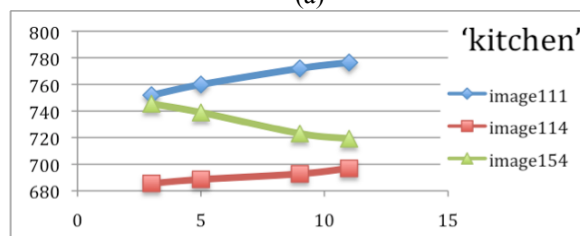
² Since the figure size attached in Figure 11 is too small to observe the result in original image size we attach the bigger size image in the Appendix.

Return the constructed depth image

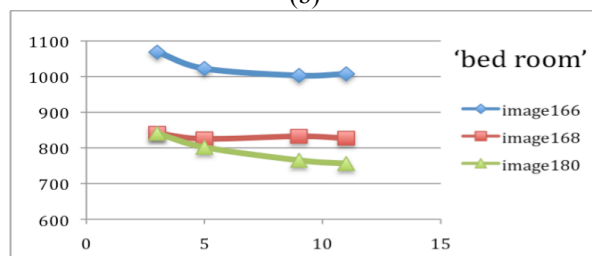
Experiment result We use 3 by 3, 5 by 5, 9 by 9 and 11 by 11 as patch sizes and fix the number of training patches as 1000 in Algorithm 2 and obtain the reconstructed depth image and measure the estimate using the formula (2). The results are shown in Figure 12. In the results in Figure 12 the x-axis indicates the patch size and the y-axis indicates the estimate. We measure the estimate for each scene category using the formula (2).



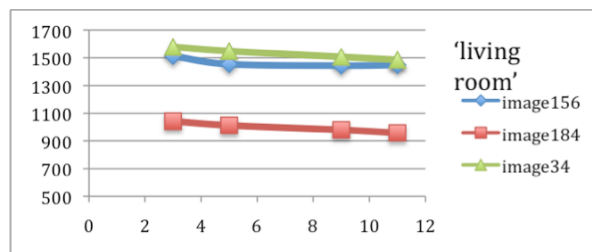
(a)



(b)



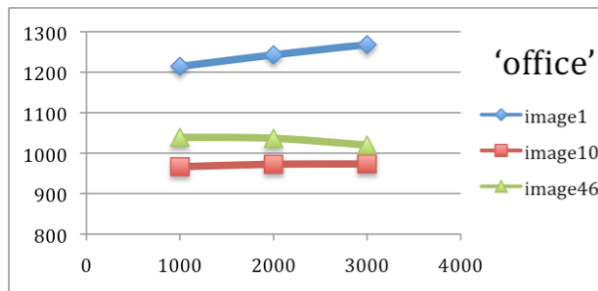
(c)



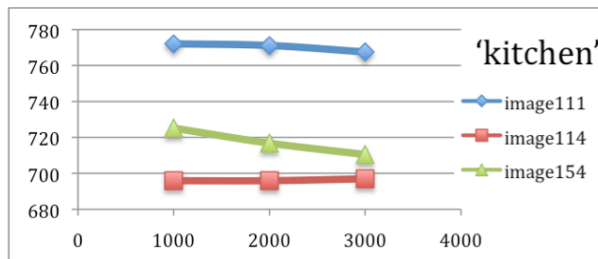
(d)

Figure 12. Estimate of each test image of each scene category by varying the patch size (a) ‘office’ scene category (b) ‘kitchen’ scene category (c) ‘bed room’ scene category (d) ‘living room’ scene category

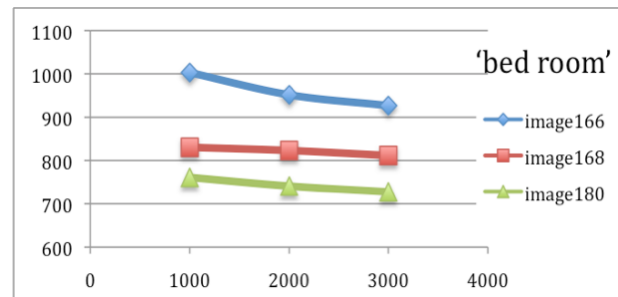
When the training images are from the same scene category i.e., when we find the most similar patch and reconstruct the depth image we consider only the patches in the same scene category we expected to see the lowest estimate using formula (2). However as we can compare the estimates in Figure 12 to the estimates in Figure 4 the experiment with all scene categories and also the estimates in Figure 14 of the experiment with different scene categories, only the estimates for ‘kitchen’ scene category were lower than the estimated from other two experiments. This seems to be because other three categories, ‘office’, ‘bedroom’, and ‘living room’ share common indoor structure or furniture such as desks or chairs. Even if ‘kitchen’ scene category has a table similar to the desk and chair it also contains unique items such as oven, utensil, bottles, etc. Another interesting observation is that in this same scene category training images experiment the number of test images that showed the decreasing estimates as the patch size increases was the most. In all the test images the estimates show this tendency except for image111 and image 114.



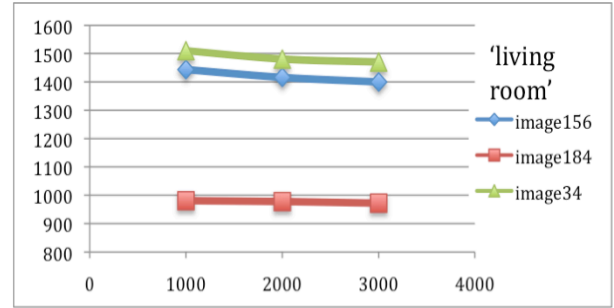
(a)



(b)



(c)



(d)

Figure 11. Estimate of each test image of each scene category by varying the number of training patches (a) ‘office’ scene category (b) ‘kitchen’ scene category (c) ‘bed room’ scene category (d) ‘living room’ scene category.

3.3 Control experiment 3: Depth inference from training images in different scene categories

We implement the third control experiment using the algorithm illustrated in Algorithm 3. In this experiment we use all images in the training images in different scene category.

Algorithm 3. Depth inference of a single still image from training patches in different scene category

Input a still RGB image without ground truth depth information

Output inferred depth information of the input image

Algorithm

for each pixel (x,y) in the input image, make an i by i patch including neighboring pixels

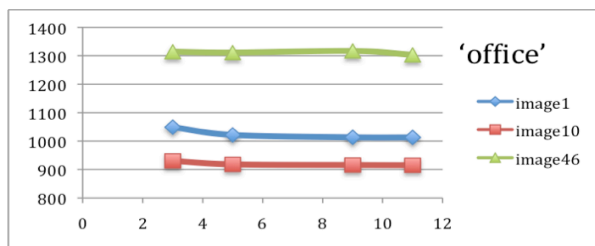
for this each patch obtained from the test image, randomly select the same size patch from training images that do not include the test image and *from different scene categories* and compare their RGB values using formula (2)

Select the best match whose similarity using formula (2) has the minimum value amongst all the training patches Construct a depth image by concatenating the depth value of the center pixel in the training patch selected in the procedure above.

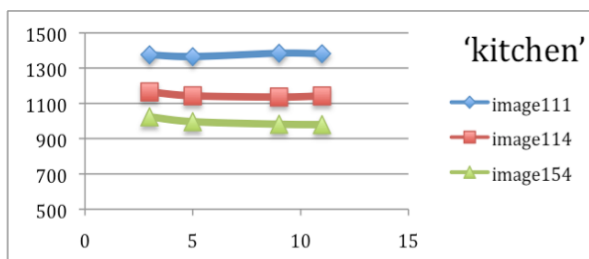
Return the constructed depth image

Experiment result We use 3 by 3, 5 by 5, 9 by 9 and 11 by 11 as patch sizes and fix the number of training patches as 1000 in Algorithm 3 and obtain the reconstructed depth image and measure the estimate using the formula (2). The results are shown in Figure 14 and Figure 15. In the results in Figure 12 the x-axis indicates the patch size and the y-axis indicates the estimate. We measure the estimate for each scene category using the formula (2). In the results in Figure 15 the x-axis indicates the number of training patches and the y-axis indicates the estimate. As we

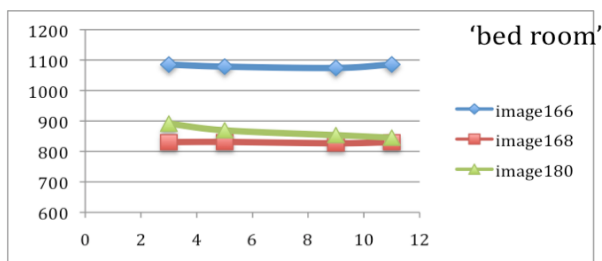
increase the number of training patches the estimate from the control experiment 3 decrease in 'kitchen' scene data and also in this experiment the estimates in 'kitchen' scene data are lower than the estimates in the control experiment 1 or control experiment 2, which agree with our hypothesis.



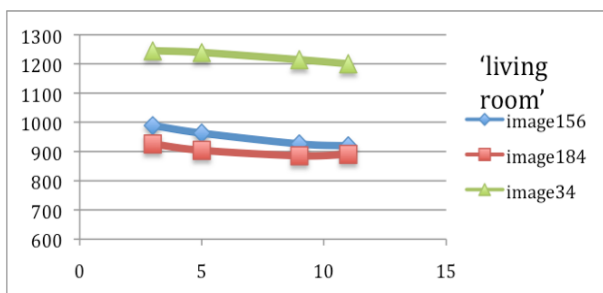
(a)



(b)

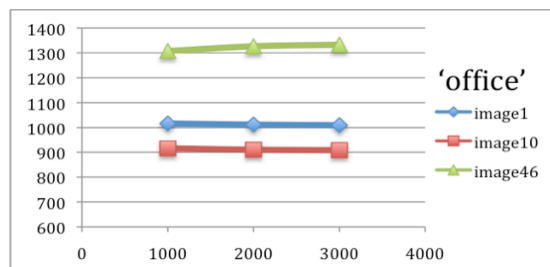


(c)

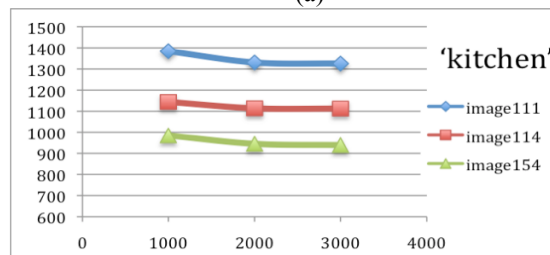


(d)

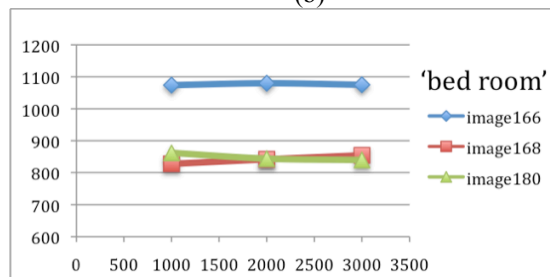
Figure 14. Estimate of each test image of each scene category by varying the patch size (a) 'office' scene category (b) 'kitchen' scene category (c) 'bed room' scene category (d) 'living room' scene category



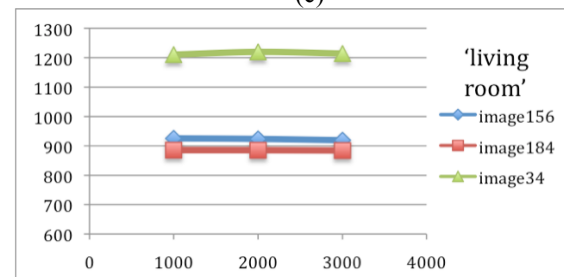
(a)



(b)



(c)



(d)

Figure 15. Estimate of each test image of each scene category by varying the number of training patches (a) 'office' scene category (b) 'kitchen' scene category (c) 'bed room' scene category (d) 'living room' scene category.

4. Conclusion and Discussion

We provided a method to infer depth information from a single still image without any ground truth labels. By running the experiments explained in Sec 2 and 3 we obtained a depth image for this RGB image and this image showed a similar pattern to the ground truth depth. We could confirm this result because our test image originally had ground truth labels but we did not use these labels for the depth inference and used these labels for the performance evaluation. As a future work we can try using

SIFT[10] features to get the similar patches in the experiment. As a way to speed up our method constructing a tree using hierarchical clustering when we find the best match patch will be helpful when there are millions of data.

References

- [1] Ashutosh Saxena, Sung Chung, and Andrew Ng. 3-D Depth Reconstruction from a Single Still Image. IJCV 2008.
- [2] Ashutosh Saxena, Min Sun, Andrew Ng. Make3D: Learning 3D Scene Structure from a Single Still Image. PAMI 2008.
- [3] Daniel Scharstein and Richard Szeliski. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. IJCV 2002.
- [4] Antonio Torralba and Aude Oliva. Depth Estimation from Image Structure. IEEE Trans Pattern Analysis and Machine Intelligence (PAMI), vol. 24, no. 9, pp.1-13, 2002.
- [5] Ashutosh Saxena, Sung H. Chung, and Andrew Y. Ng. Learning Depth from Single Monocular Images. Neural Information Processing Systems (NIPS) 2005.
- [6] Sara Vicente, Vladimir Kolmogorov, and Carsten Rother. Joint Optimization of Segmentation and Appearance Models. ICCV 2009.
- [7] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-Time Human Pose Recognition in Parts from a Single Depth Image. CVPR 2011.
- [8] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. GrabCut: Interactive Foreground Extraction using Iterated Graph Cuts. SIGGRAPH 2004.
- [9] James Hays and Alexei Efros. Scene Completion using Millions of Photographs. SIGGRAPH 2007.
- [10] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints, IJCV 2004.

5. Appendix

Figure 5 (a)

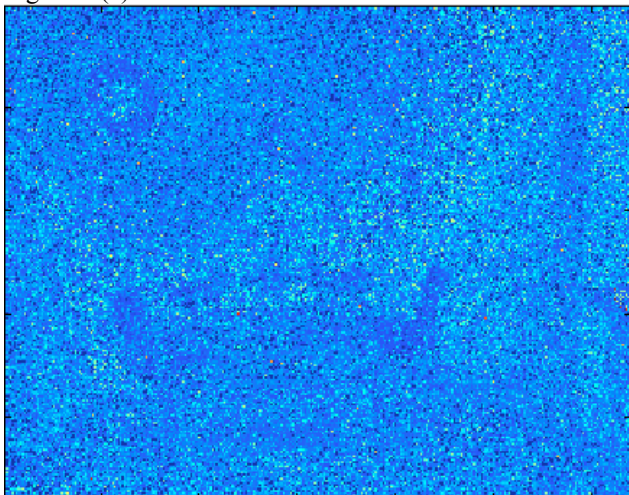


Figure 5. (b)

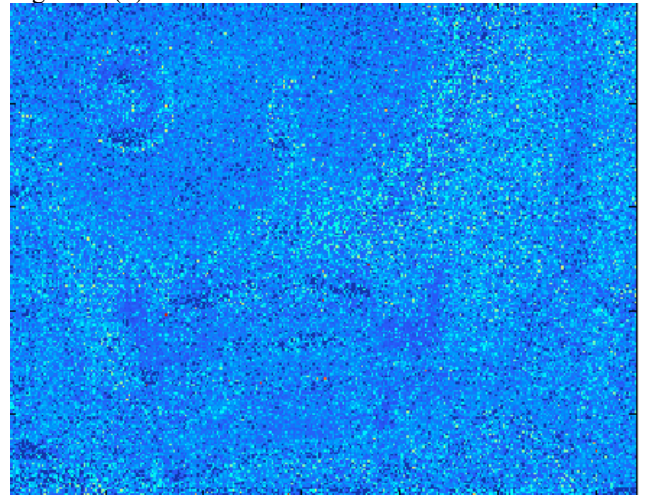


Figure 6(a)

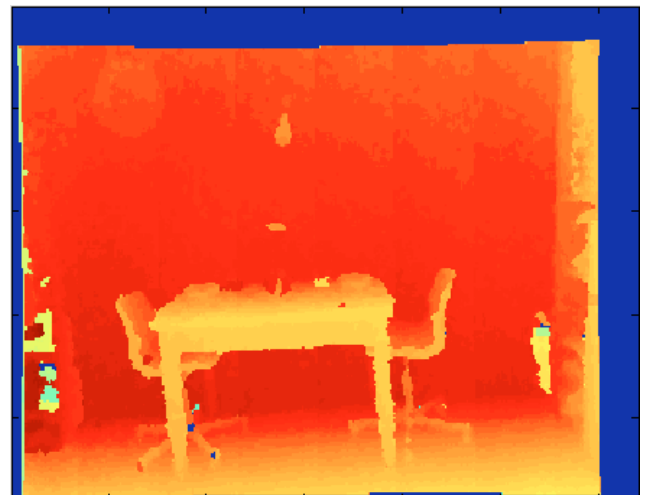


Figure 11 (a)

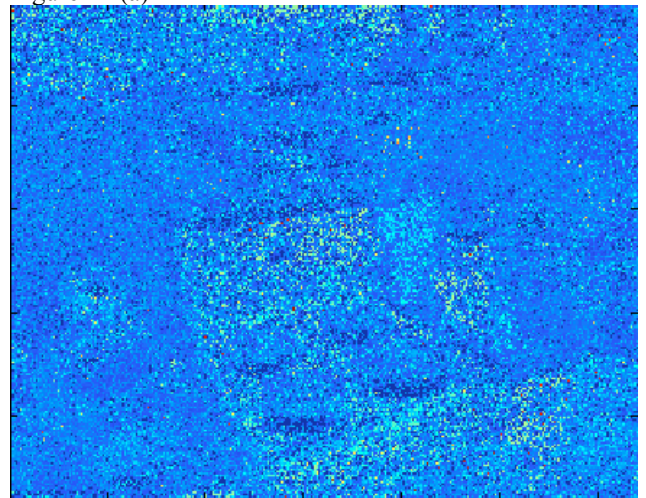


Figure 11 (b)

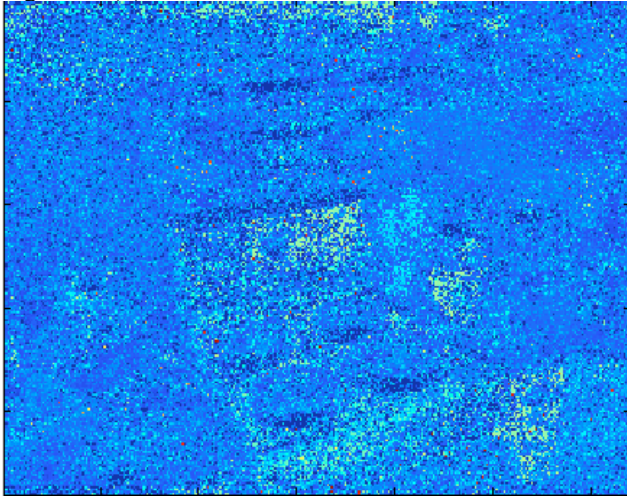


Figure 11 (c)

