Discrete-Continuous Depth Estimation from a Single Image

Miaomiao Liu, Mathieu Salzmann, Xuming He
NICTA* and CECS, ANU, Canberra
{miaomiao.liu, mathieu.salzmann, xuming.he}@nicta.com.au

Abstract

In this paper, we tackle the problem of estimating the depth of a scene from a single image. This is a challenging task, since a single image on its own does not provide any depth cue. To address this, we exploit the availability of a pool of images for which the depth is known. More specifically, we formulate monocular depth estimation as a discrete-continuous optimization problem, where the continuous variables encode the depth of the superpixels in the input image, and the discrete ones represent relationships between neighboring superpixels. The solution to this discrete-continuous optimization problem is then obtained by performing inference in a graphical model using particle belief propagation. The unary potentials in this graphical model are computed by making use of the images with known depth. We demonstrate the effectiveness of our model in both the indoor and outdoor scenarios. Our experimental evaluation shows that our depth estimates are more accurate than existing methods on standard datasets.

1. Introduction

In this paper, we address the problem of scene depth estimation from a single image. Estimating the depth of a general scene from a monocular, static viewpoint is a very challenging task, since no reliable cues, such as stereo correspondences, or motion, can be exploited.

In recent years, much progress has been made towards accurate 3D scene reconstruction from single images. For instance, simple geometric assumptions (i.e., box models) have proven effective to estimate the layout of a room [9, 17, 27]. Similarly, for outdoor scenes, the Manhattan, or blocks world, assumption has been utilized to perform 3D scene layout estimation [7]. These box models, however, are limited to represent simple structures, and are therefore ill-suited to obtain detailed 3D reconstructions.

*NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the ARC through the ICT Centre of Excellence program.

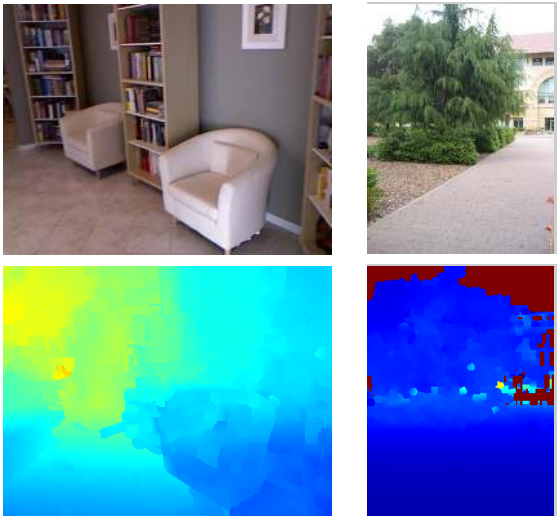


Figure 1. Depth estimation from a single image: Input images and depth maps estimated by our method.

In contrast, several methods have been proposed to directly estimate the depth of image (super)pixels [24, 25]. In this context, it was shown that exploiting additional sources of information, such as user annotations [22], semantic labels [18], or the presence of repetitive structures [30], could help improving reconstruction accuracy. Unfortunately, such additional information is not available in general. Recently, nonparametric approaches were therefore introduced to handle this case [13, 15, 16]. Given an input image, these approaches proceed by retrieving similar images in a pool of images for which the depth is known. The depths of the retrieved candidates are then employed in conjunction with smoothness constraints to estimate a depth map. While this has achieved some success, as suggested in [31] in the context of stereo, the gradient-aware smoothing strategy often poorly reflects the real 3D scene observed in the image.

In this paper, we introduce a method that addresses this issue by modeling depth estimation as a discrete-continuous optimization problem. In particular, in addition to the standard continuous variables that encode the depth of the superpixels in the input image, we make use of discrete variables that allow us to model complex relationships between

neighboring superpixels. Depth estimation can then be expressed as inference in a higher-order, discrete-continuous graphical model.

More specifically, given an input image, we make use of a nonparametric approach to retrieve similar images in a dataset of images with known depth. We exploit the depths of these images to construct data terms for the continuous variables in our model. Furthermore, we employ discrete variables to encode the occlusion relationships between neighboring superpixels. The interactions of several discrete variables can then be expressed with junction potentials, which define invalid configurations. These discrete occlusion variables also let us define smoothness constraints that better reflect realistic scenes. We make use of particle belief propagation [12, 20] to perform inference in the resulting higher-order, discrete-continuous graphical model.

In contrast to most existing methods which typically consider either indoor scenes, or outdoor ones, we demonstrate the effectiveness of our model in these two scenarios. Our experiments show the benefits of our discrete-continuous formulation, which yields state-of-the-art accuracies on the NYU v2 indoor scenes dataset [28] and on Make3D [25].

2. Related Work

Estimating the depth of a scene from images is one of the major goals of computer vision. Therefore, it has attracted a lot of attention over the years. Here, we focus on the advances that have been made in the recent years.

A classical approach to reconstructing 3D scenes consists in exploiting video. In this context, 3D scene flow is one of the most popular and mature approaches to depth estimation [3, 4, 29]. Similarly, structure-from-motion [5] and SLAM [19] have now reached the stage where existing systems can efficiently handle huge amounts of images. Therefore, they are now being integrated into 3D scene understanding methods that jointly detect, or segment objects while performing 3D reconstruction [6, 8, 23]. While here we tackle the monocular, static case, it was shown that the depth maps obtained from single images could also have a beneficial impact on video-based 3D reconstruction [13].

When it comes to single images, 3D reconstruction methods have not yet attained the same degree of maturity as video-based techniques. Nonetheless, much progress has been made in recent years. In particular, for indoor scenes, effective techniques have been proposed to estimate the layout of rooms. These methods typically rely on box-shaped models, and try to fit the box edges to those observed in the image [9, 17, 27]. The same simple geometric prior, blocks world, was exploited in outdoor scenes [7]. In [10], a more accurate geometric model was employed, but the results remain only a rough estimate of surface normals.

The simple geometric models described above do not allow us to obtain a detailed 3D description of the scene. In contrast, several methods have proposed to directly estimate the depth of image (super)pixels. Since a single image does typically not provide enough information to estimate depth, other sources of information have been exploited. In particular, in [22], depth was predicted from user annotations. In [18], this was achieved by making use of semantic class labels. Alternatively, the presence of repetitive structures in the scene was also employed for 3D reconstruction [30]. With the recent popularity of depth sensors, sparse depths have also been used to estimate denser depth maps [2].

In this work, however, we focus on the scenario where no such sources of information are available. In this setting, supervised learning techniques were the first to provide realistic results by learning the parameters of a Markov random field [24, 25]. More recently, several nonparametric approaches were introduced [13, 15, 16]. These methods exploit the availability of a set of images for which the depths are known. Depth in the input image is then estimated by first retrieving similar images in this set, and optionally warping their depth using SIFT flow. These (warped) depth maps are then utilized in the objective function of a nonlinear optimization problem that encourages the resulting depth to be smooth.

Our work is close in spirit to that of [13, 15, 16] in the sense that we also make use of a nonparametric approach to retrieve candidate depth maps. However, we avoid the warping process of [13, 15], which is computationally expensive and does not necessarily improve the quality of the candidates. More importantly, we introduce the use of discrete variables that allow us to model more complex relationships between neighboring superpixels, and formulate depth estimation as inference in a discrete-continuous graphical model. As evidenced by our results, this formulation is beneficial in terms of accuracy of the estimated depth, and proved effective for both the indoor and outdoor scenarios.

3. Discrete-Continuous Depth Estimation

We now describe our approach to depth estimation from a single image. To this end, we first derive the Conditional Random Field (CRF) that defines our problem, and discuss the inference method that we use. We then define the different potentials utilized in our model.

3.1. Discrete-Continuous CRF

Our goal is to estimate the depth of the pixels observed in a single image depicting a general scene. We formulate this problem in terms of superpixels, making the common assumption that each superpixel is planar. The pose of a superpixel is then expressed in terms of the depth of its centroid and its plane normal. Furthermore, we make use of additional discrete variables that encode the relationship of two neighboring superpixels. In particular, here, we con-

sider 4 types of relationships encoding the fact that the two superpixels (i) belong to the same object; (ii) belong to two different but connected objects; (iii) belong to two objects that form a left occlusion; and (iv) belong to two objects that form a right occlusion. Here, the notion of left and right occlusions follows the formalism of [11] based on edge directions. Given these variables, we express depth estimation as an inference problem in a discrete-continuous CRF.

More specifically, let $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_S\}$ be the set of continuous variables, where each $\mathbf{y}_i \in \mathbb{R}^4$ concatenates the centroid depth and plane normal of superpixel i, and where S is the total number of superpixels in the input image. Furthermore, let $\mathbf{E} = \{\mathbf{e}_p\}_{p \in \mathcal{E}}$ be the set of discrete variables, where each $\mathbf{e}_p \in \{so, co, lo, ro\}$, which indicates same object (so), connected but different objects (co), left occlusion (lo) and right occlusion (ro), respectively. $\mathcal{E}$ is the set of pairs of superpixels that share a common boundary.

Given these variables, we then form a CRF, where the joint distribution over the random variables factorizes into a product of non-negative potentials. This joint distribution can be written as

$$p(\mathbf{Y}, \mathbf{E}) = \frac{1}{Z} \prod_i \Psi_i(\mathbf{y}_i) \prod_\alpha \Psi_\alpha(\mathbf{y}_\alpha, \mathbf{e}_\alpha) \prod_\beta \Psi_\beta(\mathbf{e}_\beta) \ ,$$

where Z is a normalization constant, i.e., the partition function, Ψ_i is a unary potential function over the continuous variables that defines the data term for depth, and Ψ_α and Ψ_β are potentials over mixed variables and discrete variables, respectively, which encode the smoothness and consistency between depth and edge types.

Inference in the graphical model is then performed by computing a MAP estimate. By working with negative log potential functions, e.g., $\phi_i(\mathbf{y}_i) = -\ln(\Psi_i(\mathbf{y}_i))$, this can be expressed as the optimization problem

$$(\mathbf{Y}^*, \mathbf{E}^*) \tag{1}$$
$$= \operatorname*{argmin}_{\mathbf{Y}, \mathbf{E}} \sum_i \phi_i(\mathbf{y}_i) + \sum_\alpha \phi_\alpha(\mathbf{y}_\alpha, \mathbf{e}_\alpha) + \sum_\beta \phi_\beta(\mathbf{e}_\beta) \ .$$

The potentials that we use here are discussed in Section 3.2.

To handle mixed discrete and continuous variables, we make use of particle (convex) belief propagation (PCBP) [20], which lets us obtain an approximate solution to the optimization problem (1). More specifically, PCBP proceeds by iteratively solving the following steps:

1. Draw N_s random samples $\mathbf{y}_i^j$, $1 \leq j \leq N_s$ around the previous MAP solution for each variable $\mathbf{y}_i$.

2. Compute the (approximate) MAP solution of the discrete CRF formed by the discrete variables $\{\mathbf{e}_p\}$ and by utilizing the random samples $\{\mathbf{y}_i^j\}$ as discrete states for the variables $\{\mathbf{y}_i\}$.

In practice, we draw samples for the plane normal of the superpixels according to a Fisher-Bingham distribution, which forces them to have unit norm. Samples for the depth of the centroid of each superpixel are drawn according to a Gaussian distribution. At each iteration, we tighten the sampling around the previous MAP solution. The approximate MAP of the discrete CRF is obtained by distributed convex belief propagation [26].

In this work, we make use of a nonparametric approach to obtain a reasonable initialization for the algorithm. In particular, we retrieve the K images most similar to the input image from a set of images for which the depth is known. To this end, we perform a nearest-neighbor search based on concatenated GIST, PHOG and Object Bank features and directly make use the depth of the retrieved images, i.e., in contrast to [13, 15], we do not warp the depth of the retrieved images. The retrieved K depth maps then directly act as states in the first round of PCBP, i.e., no random samples are used in this round.

In the next section, we describe the specific potentials used in the optimization problem (1).

3.2. Depth and Occlusion Potentials

The objective function in (1) contains three different types of potentials involving, respectively, continuous variables only, discrete and continuous variables, and discrete variables only. Below, we discuss the functions used in these three different types of potentials.

Potentials for continuous variables:
The potentials involving purely continuous variables are unary potentials, and are of two different kinds. For the first one, we exploit the K candidates retrieved by the image-based nearest-neighbor strategy mentioned in the previous section. The first potential encodes the fact that the final depth should remain close to at least one candidate. To this end, we make use of the squared depth difference. More specifically, assuming a calibrated camera, the depth $d_i^{\mathbf{u}_j}$ of pixel $\mathbf{u}_j = (u_j, v_j)$ in superpixel i can be obtained by intersecting the visual ray passing through $\mathbf{u}_j$ with the plane defined by $\mathbf{y}_i$. This lets us write the potential

$$\phi_i^c(\mathbf{y}_i) = \min_{k=1}^{K} \frac{1}{N_i^p} \sum_{j=1}^{N_i^p} (d_i^{\mathbf{u}_j}(\mathbf{y}_i) - d_{k,i}^{\mathbf{u}_j})^2 \ , \tag{2}$$

where N_i^p is the number of pixels in superpixel i, and $d_{k,i}^{\mathbf{u}_j}$ denotes the depth of the k^{th} candidate for superpixel i at pixel $\mathbf{u}_j$. In practice, instead of directly using the candidate depth, we fit a plane to the candidate superpixels and use the intersection of this plane with the visual rays. This provides some robustness to noise in the candidates.

As a second unary potential for the continuous variables, we also make use of the candidate depths, but in a less di-

rect manner. More specifically, we train 4 different Gaussian Process (GP) regressors, each corresponding to one dimension of the variable $\mathbf{y}_i$. The input to each regressor is composed of the corresponding measurement of the candidates for superpixel i. We found these inputs to be more reliable than image features. For each GP, we used an RBF kernel with width set to the median squared distance computed over all the training samples. For more details on GP regression, we refer the reader to [21]. Given the regressed value $\mathbf{y}_i^r$ for superpixel i, we compute the depth $d_{r,i}^{\mathbf{u}_j}$ at each pixel $\mathbf{u}_j$ in the same manner as before, and write the potential

$$\phi_i^r(\mathbf{y}_i) = \frac{w_r}{N_i^p} \sum_{j=1}^{N_i^p} (d_i^{\mathbf{u}_j}(\mathbf{y}_i) - d_{r,i}^{\mathbf{u}_j})^2 \ , \tag{3}$$

where w_r is the weight of this potential relative to $\phi_i^c(\mathbf{y}_i)$. In practice, we also use the regressed value $\mathbf{y}_i^r$ as a state for superpixel i in the first round of PCBP where no sampling is performed.

Potential for mixed variables:
Our model also exploits a potential that involves both continuous and discrete variables. In particular, we define a potential that encodes the compatibility of two superpixels that share a common boundary and the corresponding discrete variable. This potential can be expressed as

$$\phi_{i,j}^m(\mathbf{y}_i, \mathbf{y}_j, \mathbf{e}_{i,j}) = w_m \times \begin{cases} g_{i,j}\|\mathbf{n}_i - \mathbf{n}_j\|^2 & \\ + \frac{1}{N_{i,j}^b} \sum_{m=1}^{N_{i,j}^b} (d_i^{\mathbf{u}_m}(\mathbf{y}_i) - d_j^{\mathbf{u}_m}(\mathbf{y}_j))^2 & \text{if } \mathbf{e}_{i,j} = so \\ \frac{1}{N_{i,j}^b} \sum_{m=1}^{N_{i,j}^b} (d_i^{\mathbf{u}_m}(\mathbf{y}_i) - d_j^{\mathbf{u}_m}(\mathbf{y}_j))^2 & \text{if } \mathbf{e}_{i,j} = co \\ \phi_{i,j}^o(\mathbf{y}_i, \mathbf{y}_j, \mathbf{e}_{i,j}) & \text{otherwise,} \end{cases}$$

where w_m is the weight of this potential, $\mathbf{n}_i$ is the plane normal of superpixel i, i.e., 3 components of $\mathbf{y}_i$, $N_{i,j}^b$ is the number of pixels shared along the boundary between superpixel i and superpixel j, and $g_{i,j}$ is a weight based on the image gradient at the boundary between superpixel i and j, i.e., $g_{i,j} = \exp(-\mu_{i,j}/\sigma)$, with $\mu_{i,j}$ the mean gradient along the boundary between the two superpixels. To handle the occlusion cases, the function $\phi_{i,j}^o(\mathbf{y}_i, \mathbf{y}_j, \mathbf{e}_{i,j})$ assigns a cost 0 if the two superpixels are in a configuration that agrees with the state of $\mathbf{e}_{i,j}$, i.e., left occlusion or right occlusion, and a cost θ_{max} otherwise. While this potential depends on three variables, it remains fast to compute, since $\mathbf{e}_{i,j}$ can only take four states.

Potentials for discrete variables:
Finally, we use two different potentials that only involve discrete variables. The first one is a unary potential that makes use of a classifier trained to discriminate between occlusion (i.e., $lo \cup ro$) and non-occlusion (i.e., $so \cup co$) cases. To this end, we utilize the image-based occlusion cues introduced in [11] and employ a binary boosted decision tree classifier. Given the prediction of the classifier $\hat{\mathbf{e}}_p$, our potential function takes the form

$$\phi_p^t(\mathbf{e}_p) = \begin{cases} -\theta_t & \text{if } \mathbf{e}_p \text{ agrees with } \hat{\mathbf{e}}_p \\ \theta_t & \text{otherwise} \ , \end{cases} \tag{4}$$

where θ_t is a parameter of our model. Note that distinguishing between all four types of edge variables proved too unreliable, which motivated our decision to only consider occlusion vs. non-occlusion.

The second purely discrete potential is similar to the junction feasibility potential used in [31] for stereo. More specifically, it encodes information about whether the junction between three edge variables is physically possible, or not. Therefore, this potential takes the form

$$\phi_{p,q,r}^t(\mathbf{e}_p, \mathbf{e}_q, \mathbf{e}_r) = \begin{cases} \theta_{max} & \text{if impossible case} \\ 0 & \text{otherwise} \ . \end{cases} \tag{5}$$

Here, we employed the same impossible cases as in [31] for our 4 states, assuming that co typically form a hinge, while so are mostly coplanar. Note that, here, we only considered junctions of three superpixels, since junctions of four occur very rarely. However, 4-junctions could easily be introduced in our framework.

4. Experimental Evaluation

We now present our experimental results on depth estimation in outdoor and indoor scenes. In particular, we evaluated our method on two publicly available datasets: the Make3D range image dataset [25] and the NYU v2 Kinect dataset [28]. For both datasets, we compare our results with those of the depth transfer method of [13], which represents the current state-of-the-art for depth estimation from a single image. In addition to the baseline [13], we also evaluate the results of our unary terms only and of our GP depth regressors on their own, as well as the results of our model without discrete variables and with the same pairwise term as the $e_{i,j} = so$ case and of the first approximate MAP in our model obtained before sampling particles in PCBP.

For our quantitative evaluation, we report errors obtained with the three following commonly-used metrics:

- average relative error (**rel**): $\frac{1}{N} \sum_{\mathbf{u}} \frac{|g_{\mathbf{u}} - d_{\mathbf{u}}|}{g_{\mathbf{u}}}$,
- average $\mathbf{log}_{10}$ error: $\frac{1}{N} \sum_{\mathbf{u}} |log_{10} g_{\mathbf{u}} - log_{10} d_{\mathbf{u}}|$,
- root mean squared error (**rms**): $\sqrt{\frac{1}{N} \sum_{\mathbf{u}} (g_{\mathbf{u}} - d_{\mathbf{u}})^2}$,

where $g_{\mathbf{u}}$ is the ground-truth depth at pixel $\mathbf{u}$, $d_{\mathbf{u}}$ is the corresponding estimated depth, and N denotes the total number of pixels in all the images.

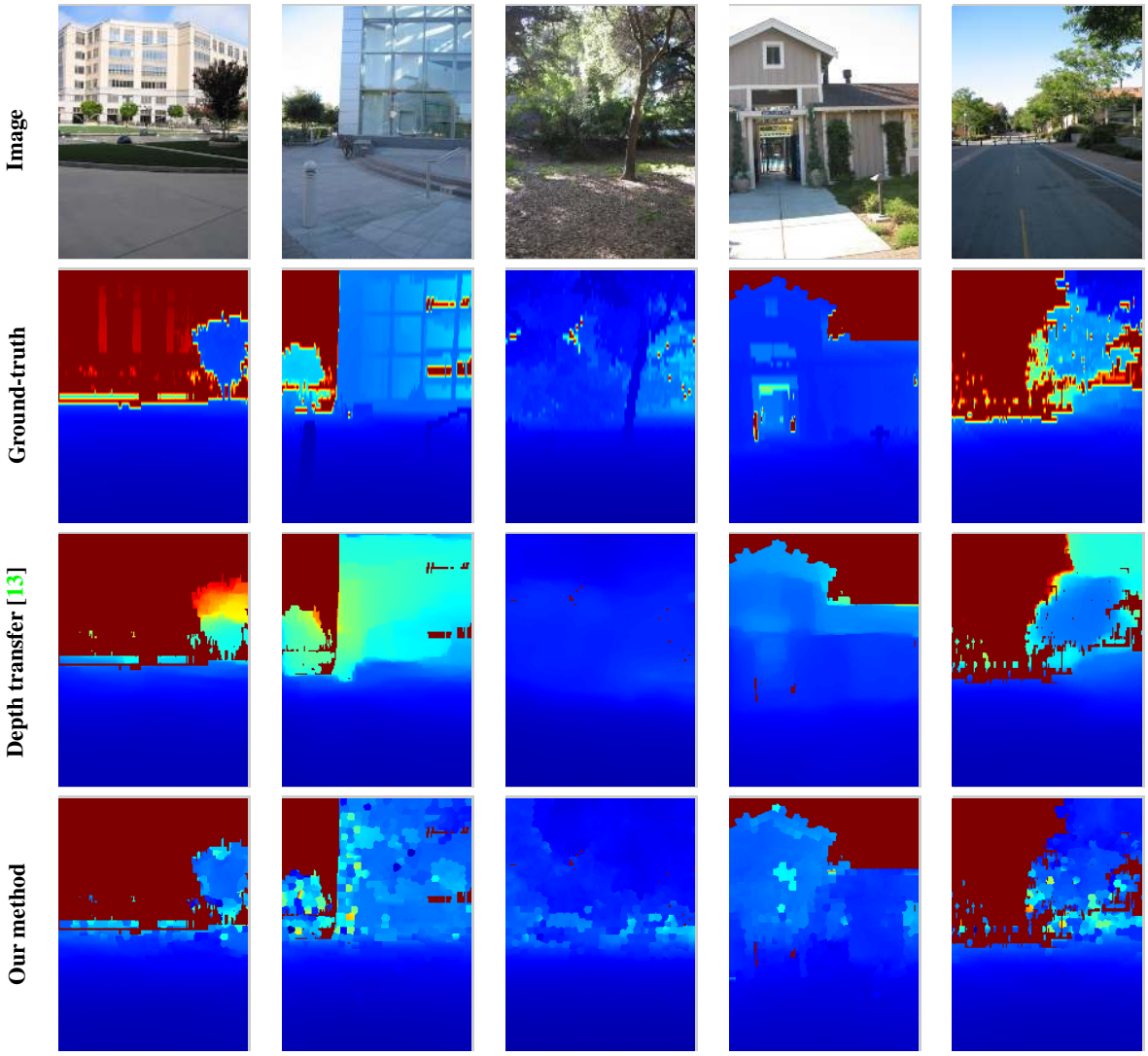


Figure 2. Qualitative comparison of the depths estimated with depth transfer [13] and with our method on the Make3D dataset. Color indicates depth (red is far, blue is close).

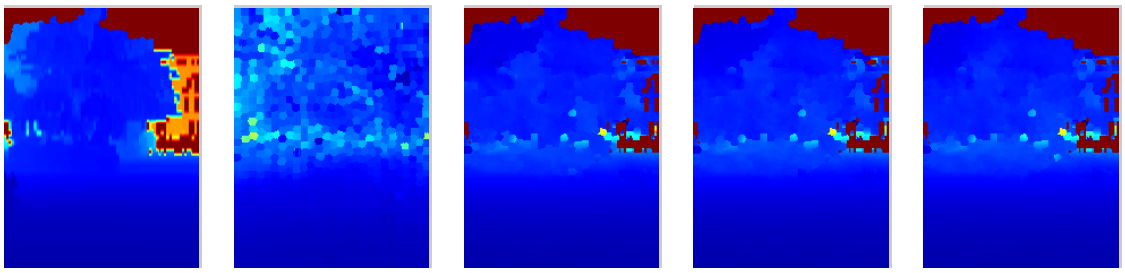
Method		rel	$\log_{10}$	rms
Depth transfer [13]	$\mathbf{C_1}$	0.355	**0.127**	**9.2**
	$\mathbf{C_2}$	0.361	0.148	15.1
Our method	$\mathbf{C_1}$	**0.335**	0.137	9.49
	$\mathbf{C_2}$	**0.338**	**0.134**	**12.6**

Table 1. Depth reconstruction errors on the Make3D dataset for depth transfer [13] and for our method evaluated on two criteria ($\mathbf{C_1}$ and $\mathbf{C_2}$, see text for details.)

Method	rel	$\log_{10}$	rms
Unary	0.352	0.142	9.61
GP regression	0.547	0.175	10.5
No discrete variables	0.326	0.147	9.932
No sampling	0.337	0.139	9.54
Full model	0.335	0.137	9.49

Table 2. Make3D: Comparison of our final results with those obtained with unary terms only, with our GP depth regressors only, using a model without discrete edge type variables, and after the first round of PCBP where no sampling is involved.

In both experiments, we used SLIC [1] to compute the superpixels. For each test image, we retrieved $K = 7$ candidates from the training images. The parameters of our model were selected using a small validation set of 10 images from the NYU v2 dataset and kept the same in both experiments. The specific values were $w_r = 1$, $w_m = 10$, $\theta_t = 10$, and $\theta_{max} = 20$. Note that these parameters could, in principle be learned. However, our approach proved robust enough for this to be unnecessary. We performed two iterations of PCBP with $N_s = 20$ particles at each iteration.



Ground-truth | Regression | No sampling | Sampling round 1 | Final depth

Figure 3. Make3D: Depth maps at different stages of our approach.

Method	**rel**	**$\log_{10}$**	**rms**
Depth transfer [13]	0.374	0.134	1.12
Our method	**0.335**	**0.127**	**1.06**

Table 3. Depth reconstruction errors on the NYU v2 dataset for depth transfer [13] and for our method using the training/test partition provided with the dataset.

Method	**rel**	**$\log_{10}$**	**rms**
Depth fusion (no warp) [16]	0.371	0.137	1.3
Depth fusion [15]	0.368	0.135	1.3
Depth transfer	0.350	0.131	1.2
Our method	**0.327**	**0.126**	**1.08**

Table 4. Comparison of the depth estimation errors on the NYU v2 dataset using a leave-one-out strategy.

4.1. Outdoor Scene Reconstruction: Make3D

The Make3D dataset contains 534 images with corresponding depth maps, partitioned into 400 training images and 134 test images. All the images were resized to 460×345 pixels in order to preserve the aspect ratio of the original images. Since the true focal length of the camera is unknown, we assume a reasonable value of 500 for the resized images. Due to the limited range and resolution of the sensor used to collect the ground-truth, far away pixels, were arbitrarily set to depth 80 in the original dataset. To take this, as well as the effect of interpolation when resizing the images, into account in our evaluation, we report errors based on two different criteria: ($\mathbf{C_1}$) Errors are computed in the regions with ground-truth depth less than 70; ($\mathbf{C_2}$) Errors are computed in the entire image. In this second scenario, to reduce the effect of meaningless candidates in sky regions, we used a classifier to label sky pixels and for the depth of the corresponding superpixels to take the value $(0, 0, 1, 80)$. Note that the same two criteria ($\mathbf{C_1}$ and $\mathbf{C_2}$) were used to evaluate the baseline.

In Table 1, we compare the results of our approach with those obtained by depth transfer [13]. Note that, using criteria $\mathbf{C_1}$, we outperform the baseline in terms of relative error and perform slightly worse for the other metrics. Using criteria $\mathbf{C_2}$, we outperform the baseline for all metrics.

Method	**rel**	**$\log_{10}$**	**rms**
Unary	0.350	0.132	1.11
GP regression	0.431	0.151	1.21
No discrete variables	0.354	0.141	1.20
No sampling	0.339	0.129	1.08
Full model	0.335	0.127	1.06

Table 5. NYU v2: Comparison of our final results with those obtained with unary terms only, with our GP depth regressors only, using a model without discrete edge type variables, and after the first round of PCBP where no sampling is involved.

Fig. 2 provides a qualitative comparison of our depth maps with those estimated by depth transfer [13] for some images of the dataset. Note that depth transfer tends to oversmooth the depth maps and, e.g., merge foreground objects with the background. Thanks to our discrete variables, our approach better respects the discontinuities in the scene. In Table 2, we show the results obtained with some of the parts of our model. Note that, even though the sampling in PCBP does not seem to have a great impact on the errors, it helps smoothing the depth maps and thus makes them look more realistic. This is evidenced by Fig. 3, where we show the depth maps at different stages of our approach. Note that the influence of each stage is more easily seen with NYU v2 (see Fig. 5) for which the overall depth range is smaller.

4.2. Indoor Scene Reconstruction: NYU v2

The NYU v2 dataset contains 1449 images, partitioned into 795 training images and 654 test images. All the images were resized to 427×561 pixels, while simultaneously respecting the masks provided with the dataset. In this case, the intrinsic camera parameters are given with the dataset. We evaluated the depth transfer code provided by [13] to obtain baseline results on the training/test provided with the dataset and compare these results with those obtained with our approach in Table 3. To be able to compare our results with those reported in [14], we also applied our method in a leave-one-out manner on the full dataset. The results are reported in Table 4. Note that, in both cases, we outperform the baselines for all metrics. These error metrics were computed over the valid pixels (non-zero depth) in the ground-

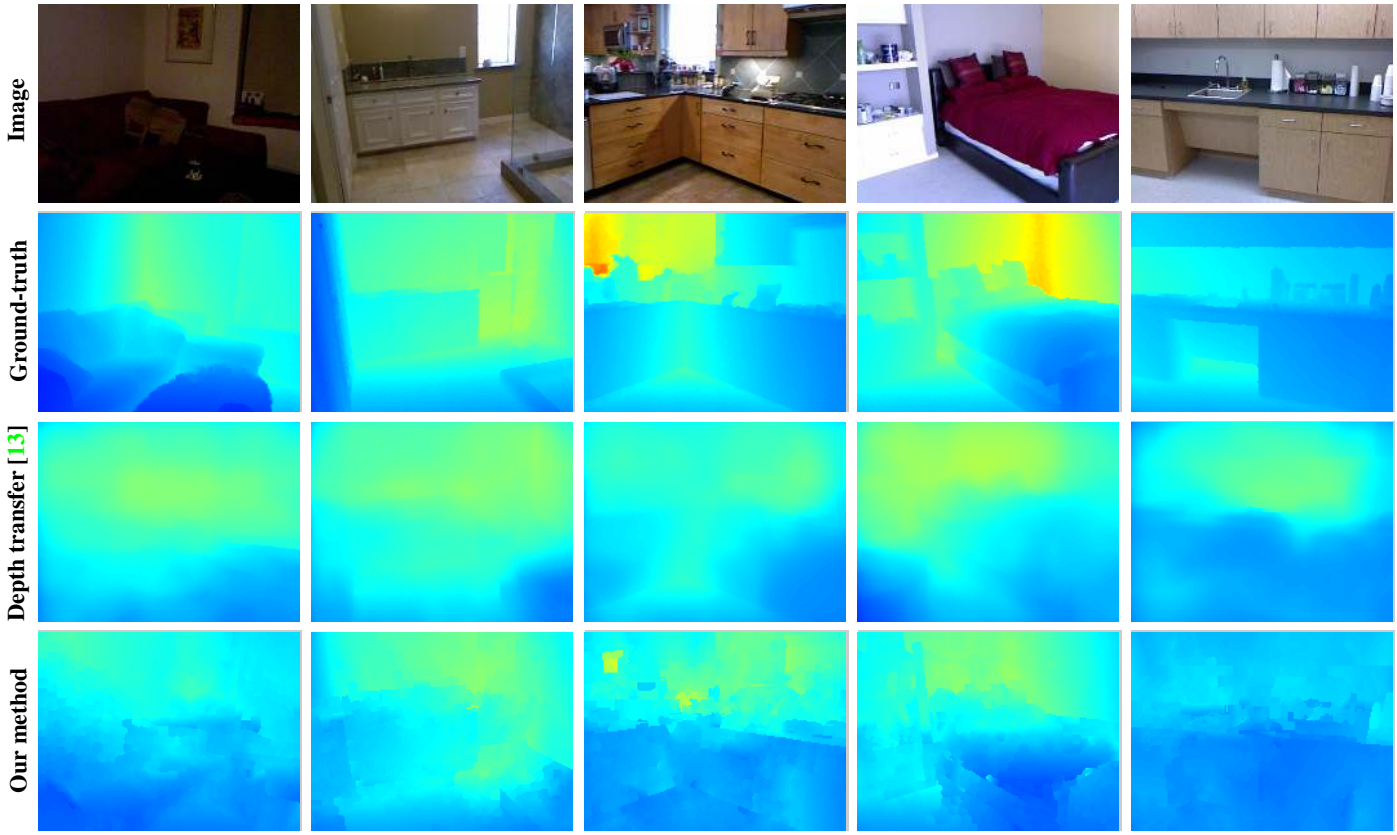


Figure 4. Qualitative comparison of the depths estimated with depth transfer [13] and with our method on the NYU v2 dataset.

Figure 5. NYU v2: Depth maps at different stages of our approach.

truth depth maps.

In Fig. 4, we provide a qualitative comparison of our results with those of [13] for some images. Note that the oversmoothing of the depth maps generated by depth transfer is even more obvious in the short depth range scenario. In contrast, our approach still yields a realistic representation of the scene. In Table 5, we show the influence of the different parts of our model. Note that all the components contribute to our final results. Fig. 5 depicts the depth maps at different stages of our approach. While sampling smoothes the depth map, it still respects the image discontinuities.

In addition to the estimated depth, our model can also predict the boundary type of the superpixel edges. In particular, the occlusion boundaries are useful cues for spatial reasoning. We qualitatively evaluate the occlusion boundary prediction by showing typical results in Fig. 6 for both indoor and outdoor scenarios. Note that our model captures most of the occlusion edges.

5. Conclusion

In this paper, we have presented an approach to estimating the depth of a scene from a single image. To this end, we have employed continuous variables to represent the depth of image superpixels, and discrete ones to encode relationships between neighboring superpixels. As a result, we have formulated depth estimation as inference in a higher-order, discrete-continuous graphical model, which we have performed using particle belief propagation. Our experiments

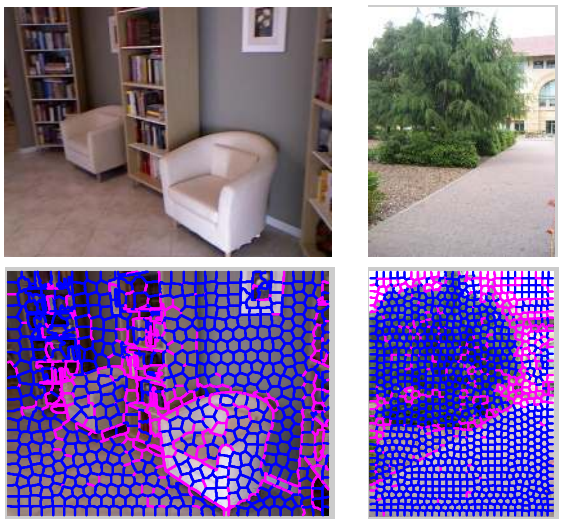


Figure 6. Estimated boundary occlusion map. The top row shows the input image and the bottom row shows the estimated boundary occlusion map. The superpixel boundaries are drawn in blue. Pixels in magenta denote the estimated occlusion boundaries.

have shown that this model let us effectively reconstruct general scenes from still images in both the indoor and outdoor scenarios. In the future, we intend to study how this model can be exploited in 3D scene understanding by, e.g., jointly performing semantic labeling and depth estimation.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Suesstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *PAMI*, 2012. 5

[2] O. M. Aodha, N. D. Campbell, A. Nair, and G. Brostow. Patch based synthesis for single depth image superresolution. In *ECCV*, 2012. 2

[3] N. Brikbeck, D. Cobzas, and M. Jaegersand. Depth and scene flow from a single moving camera. In *3DPVT*, 2010. 2

[4] N. Brikbeck, D. Cobzas, and M. Jaegersand. Basis constrained 3d scene flow on a dynamic proxy. In *ICCV*, 2011. 2

[5] D. Crandall, A. Owens, N. Snavely, and D. Huttenlocher. Discrete-continuous optimization for large-scale structure from motion. In *CVPR*, 2011. 2

[6] A. Geiger, C. Wojek, and R. Urtasun. Joint 3d estimation of objects and scene layout. In *NIPS*, 2011. 2

[7] A. Gupta, A. Efros, and M. Hebert. Blocks world revisited: Image understanding using qualitative geometry and mechanics. In *ECCV*, 2010. 1, 2

[8] C. Haene, C. Zach, A. Cohen, R. Angst, and M. Pollefeys. Joint 3d scene reconstruction and class segmentation. In *CVPR*, 2013. 2

[9] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. In *ECCV*, 2010. 1, 2

[10] D. Hoiem, A. Efros, and M. Hebert. Geometric context from a single image. *ICCV*, 2005. 2

[11] D. Hoiem, A. Stein, A. Efros, and M. Hebert. Recovering occlusion boundaries from a single image. In *ICCV*, 2007. 3, 4

[12] A. Ihler and D. McAllester. Particle belief propagation. In *AISTATS*, 2009. 2

[13] K. Karsch, C. Liu, and S. B. Kang. Depth extraction from video using non-parametric sampling. In *ECCV*, 2012. 1, 2, 3, 4, 5, 6, 7

[14] K. Karsch, C. Liu, and S. B. Kang. Depthtransfer: Depth extraction from video using non-parametric sampling. *PAMI*, 2014. 6

[15] J. Konrad, G.Brown, M. Wang, P. Ishwar, C. Wu, and D. Mukherjee. Automatic 2d-to-3d image conversion using 3d examples from the internet. In *SPIE Stereoscopic Displays and Applications*, 2012. 1, 2, 3, 6

[16] J. Konrad, M. Wang, and P. Ishwar. 2d-to-3d image conversion by learning depth from examples. In *3DCINE*, 2012. 1, 2, 6

[17] D. C. Lee, A. Gupta, M. Hebert, and T. Kanade. Estimating spatial layout of rooms using volumetric reasoning about object and surfaces. In *NIPS*, 2010. 1, 2

[18] B. Liu, S. Gould, and D. Koller. Single image septh estimation from predicted semantic labels. In *CVPR*, 2010. 1, 2

[19] R. A. Newcombe, S. Lovegrove, and A. J. Davison. Dtam: Dense tracking and mapping in real-time. In *ICCV*, 2011. 2

[20] J. Peng, T. Hazan, D. McAllester, and R. Urtasun. Convex max-product algorithms for continuous mrfs with applications to protein folding. In *ICML*, 2011. 2, 3

[21] C. E. Rasmussen and C. K. Williams. *Gaussian Process for Machine Learning*. MIT Press, 2006. 4

[22] B. Russell and A. Torralba. Building a database of 3d scenes from user annotations. In *CVPR*, 2009. 1, 2

[23] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. J. Kelly, and A. J. Davison. Slam++: Simultaneous localisation and mapping at the level of objects. In *CVPR*, 2013. 2

[24] A. Saxena, S. H. Chung, and A. Y. Ng. 3-d depth reconstruction from a single still image. *IJCV*, 2007. 1, 2

[25] A. Saxena, M. Sun, and A. Y. Ng. Make3d: Learning 3d scene structure from a single still image. *PAMI*, 2009. 1, 2, 4

[26] A. G. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun. Distributed message passing for large scale graphical models. In *CVPR*, 2011. 3

[27] A. G. Schwing and R. Urtasun. Efficient exact inference for 3d indoor scene understanding. In *ECCV*, 2012. 1, 2

[28] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 2, 4

[29] C. Vogel, K. Schindler, and S. Roth. 3d scene flow estimation with a rigid motion prior. In *ICCV*, 2011. 2

[30] C. Wu, J. M. Frahm, and M. Pollefeys. Repetition-based dense single-view reconstruction. In *CVPR*, 2011. 1, 2

[31] K. Yamaguchi, T. Hazan, D. McAllester, and R. Urtasun. Continuous markov random fields for robust stereo estimation. In *ECCV*, 2012. 1, 4