Automatic Scene Inference for 3D Object Compositing
Supplemental file

Kevin Karsch[1], Kalyan Sunkavalli[2], Sunil Hadap[2],
Nathan Carr[2], Hailin Jin[2], Rafael Fonte[1],
Michael Sittig[1] David Forsyth[1]
[1]University of Illinois
[2]Adobe Research

List of figures/tables (in order of appearance)

Geometric priors for depth inference

Here, we describe the new terms we incorporate at inference time to infuse geometric information into our estimated depth.

Camera model. We compute a simple pinhole camera (focal length, f and camera center, (c_0^x, c_0^y)) and extrinsic parameters from three orthogonal vanishing points [Hartley and Zisserman 2003] (obtained in the "Geometric reasoning" step). We use this camera model as our projection operator, which is necessary for computing surface normals from depth:

$$K = \begin{bmatrix} f & 0 & c_0^x \\ 0 & f & c_0^y \\ 0 & 0 & 1 \end{bmatrix}$$

Surface normals from dense depth. First, it is important to noise that we can recover a surface normal at each pixel given dense depth. We could use plane fitting to estimate the surface orientation, but for computational reasons, we use a local operator ($N : \mathbb{R} \to \mathbb{R}^3$) that considers the change in nearby depth values:

$$P(\mathbf{D}) = \mathbf{D}(x,y)K^{-1}[x,y,1]^T,\ \forall (x,y) \in \text{pixels}, \quad (1)$$

$$V_x(\mathbf{D}) = \nabla_x \mathbf{P}(\mathbf{D}),\ V_y(\mathbf{D}) = \nabla_y \mathbf{P}(\mathbf{D}), \quad (2)$$

$$N(\mathbf{D}) = \frac{V_x(\mathbf{D}) \times V_y(\mathbf{D})}{||V_x(\mathbf{D}) \times V_y(\mathbf{D})||}, \quad (3)$$

where $\times$ is the cross product operator. Intuitively, V_x and V_y are estimates of unique surface tangents, and their normalized cross product is thus the surface normal.

Manhattan world prior. Under the Manhattan World assumption, patches of a scene should always be oriented along one of the three dominant directions. These three directions are defined by the vanishing points we detect, which encode a rotation matrix $R = (R_x, R_y, R_z)^T$ defined as the rotation that takes the identity to the set of rescaled, unprojected vanishing points ($R * I \propto K^{-1}[\text{vp}_x, \text{vp}_y, \text{vp}_z]$). To enforce such a prior, we add a penalty for surface normals not lying in parallel or perpendicular to one of these three directions:

$$pp(N,V) = \frac{1}{2} - |\,|N^T V| - \frac{1}{2}| \quad (4)$$

$$E_m(N(\mathbf{D})) = \sum_{i \in \text{pixels}} pp(N_i, R_x) + pp(N_i, R_y) + pp(N_i, R_z).$$

The function pp is a negated and translated absolute value function that is small if the input vectors are either parallel or perpendicular, and large otherwise.

Orientation constraints. We also have a good idea of the orientation of some surfaces in the scene from our geometric reasoning step, and we incorporated this knowledge as a soft constraint on surface normals in regions which we have high confidence of the surface orientation. Let $\mathcal{O}$ be the set of pixels for which we can confidently predict surface orientation, and O_i^{map} is the predicted orientation at the i^{th} pixel:

$$E_o(N(\mathbf{D})) = \sum_{i \in \mathcal{O}} 1 - |N_i^T O_i^{\text{map}}|. \quad (5)$$

3D smoothness. In real scenes, not all planes will align with one of the three dominant directions. So, we incorporate a simple smoothness prior, but we enforce smoothness in 3D rather than in the image plane. We encourage nearby normals to be pointing in the same direction, unless there are strong edges in the input image (assumed to be potential discontinuities in the normal field). We model this term as

$$E_{gs}(N(\mathbf{D})) = \sum_{i \in \text{pixels}} s_i^x ||\nabla_x N_i|| + s_i^y ||\nabla_y N_i||, \quad (6)$$

where ∇_x and ∇_y are horizontal and vertical gradients in the image domain, and $s^x = (1 + e^{(||\nabla_x I|| - 0.05)/.01})^{-1}$ and $s^y = (1 + e^{(||\nabla_y I|| - 0.05)/.01})^{-1}$ are soft thresholds (sigmoidal functions) of input image (I) derivatives.

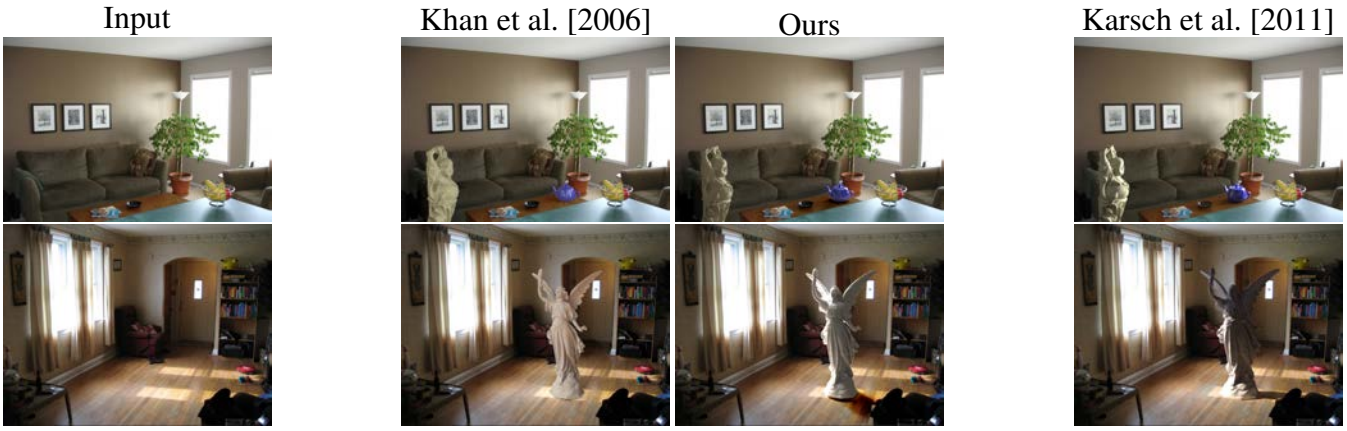


Fig. 1. Comparison of several techniques for estimating illumination from a single image. From left to right: the image-wrapping method of Khan et al. [2006], and our method. These methods are all fully automatic, compared to the semiautomatic method of Karsch et al. [2011] (right). Our automatic method is able to produce more visually appealing results than existing automatic approaches, and is comparable to methods which require a good deal of user interaction. Best viewed in color at high-resolution.

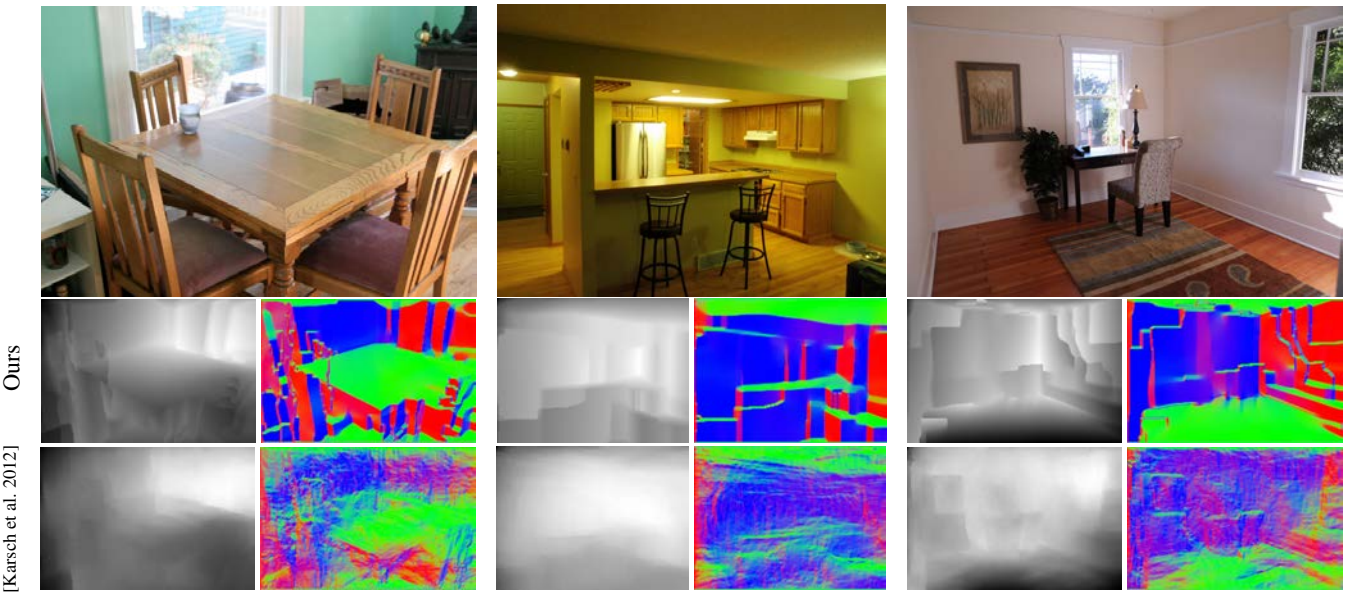


Fig. 2. Several results comparing our single image, geometric-based depth algorithm to the original Depth Transfer algorithm [Karsch et al. 2012]. Each result shows the input image above, followed below by our estimated depth and surface normals, and the estimated depth and surface normals using Depth Transfer on bottom. Surface normal images are computed by mapping the absolute value of each normal (per pixel) to the RGB channels respectively (normals are first globally rotated such that a normal pointing in any of the three dominant scene directions is either red, green, or blue). Notice that the addition of geometric-based priors significantly improve the estimated depth, and allow for piecewise planar reconstructions.

Fig. 3. In some cases, our light intensity optimization can fail (middle), but a user can manually correct these intensities on the fly using our interface (given that source positions have been estimated with moderate accuracy, and only the intensity has been misestimated).

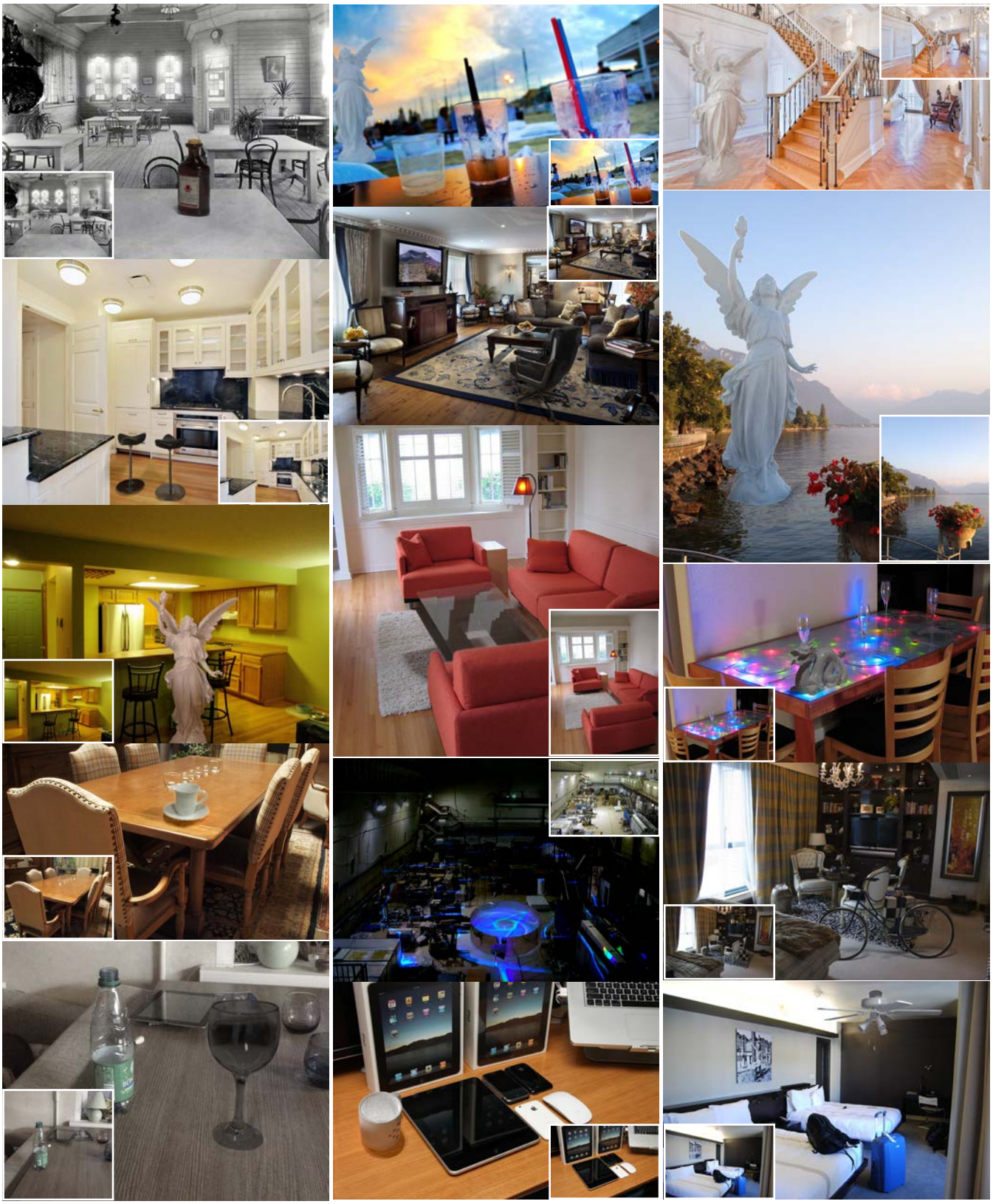


Fig. 4. Additional results. Our method achieves varying degrees of quality, but is automatic and can be used for many types of images.

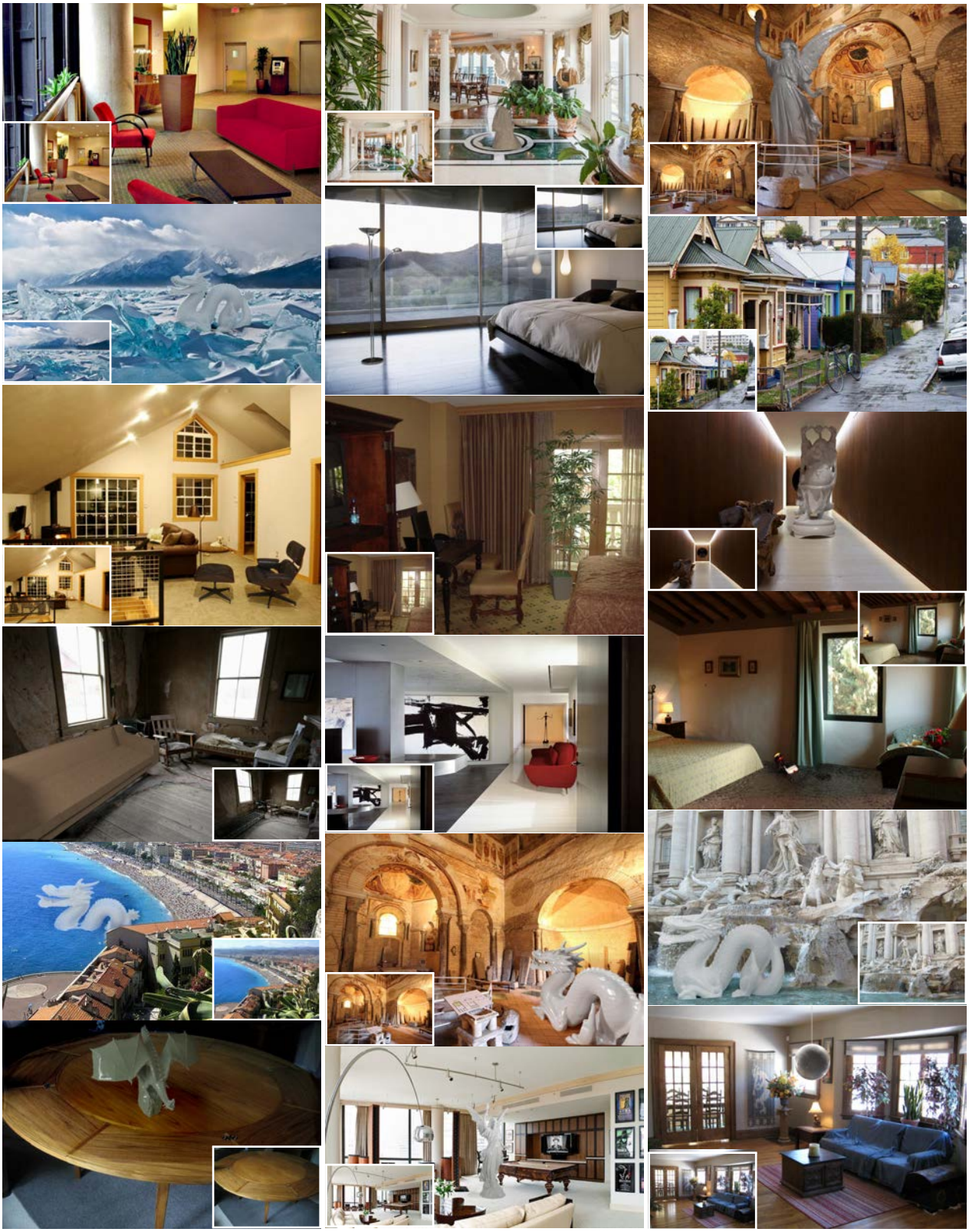


Fig. 5. Additional results. Our method achieves varying degrees of quality, but is automatic and can be used for many types of images.

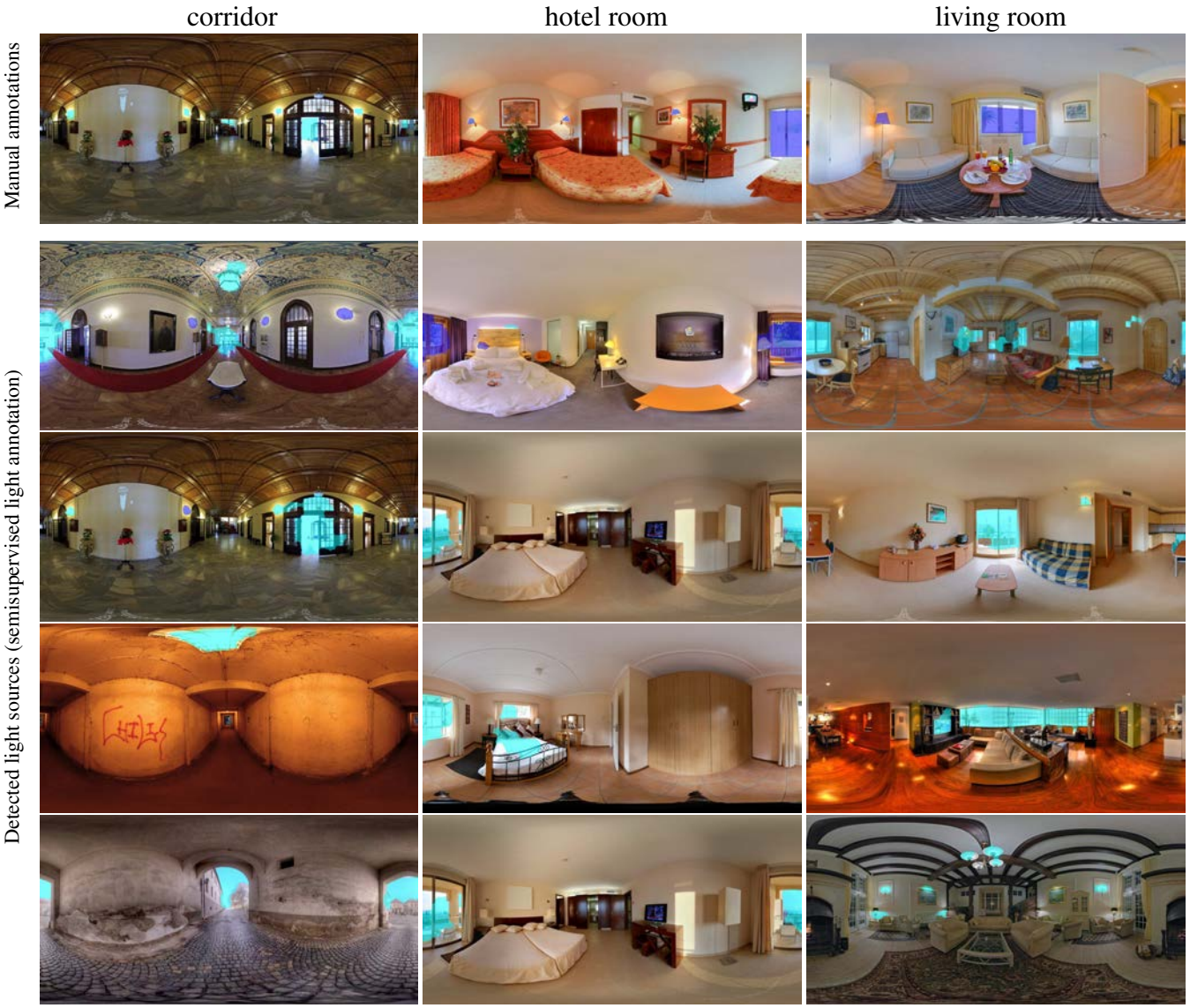


Fig. 6. Example classes from the SUN360 panorama dataset [Xiao et al. 2012]. We manually annotate light sources for several images in each class (top), train a classifier to predict light source location and distance. We then use this classifier to annotate all other images in the SUN360 dataset (bottom). Blue annotations indicate a "near" source (1-5m), and teal indicates a "medium" proximity; most sources fell into these categories for indoor scenes. Our classifier is typically robust to false positives like strong specular reflections and shafts of light; however, several true light sources are undetected by our classifier.

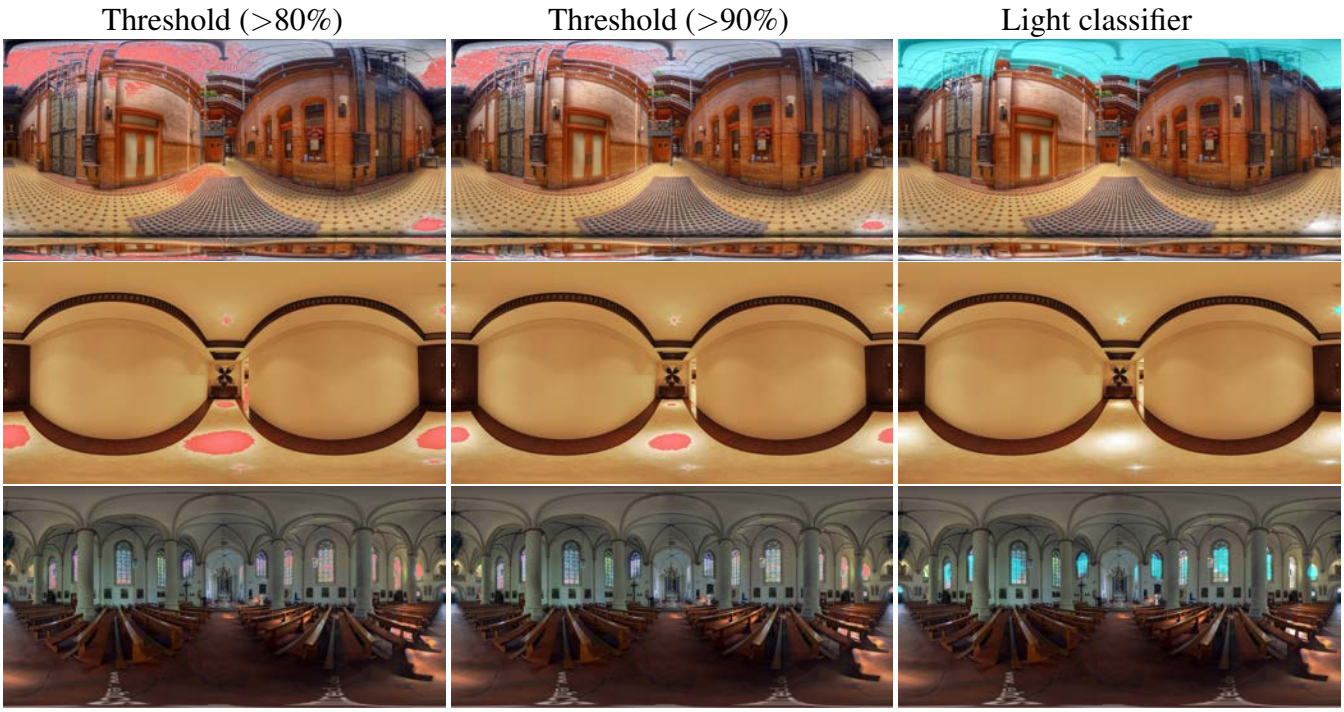


Fig. 7. Comparison of thresholding an image to detect light sources (left and middle) versus our light classification method. In many scenes, thresholding poses issues because of bright image points that are not truly sources of illumination, and also because tonemapping can mis-represent the true radiance values in a photo. Our classifier is typically robust to false positives like strong specular reflections and shafts of light (top and middle), and has success in detecting light sources that are not saturated (bottom). Light source detections are displayed in red (unknown distance from camera) and teal (automatically classified as "medium" distance); with thresholding, we cannot estimate how far the light is from the camera, whereas our classifier predicts distance as one of four discrete labels (close, medium, far, and infinite).

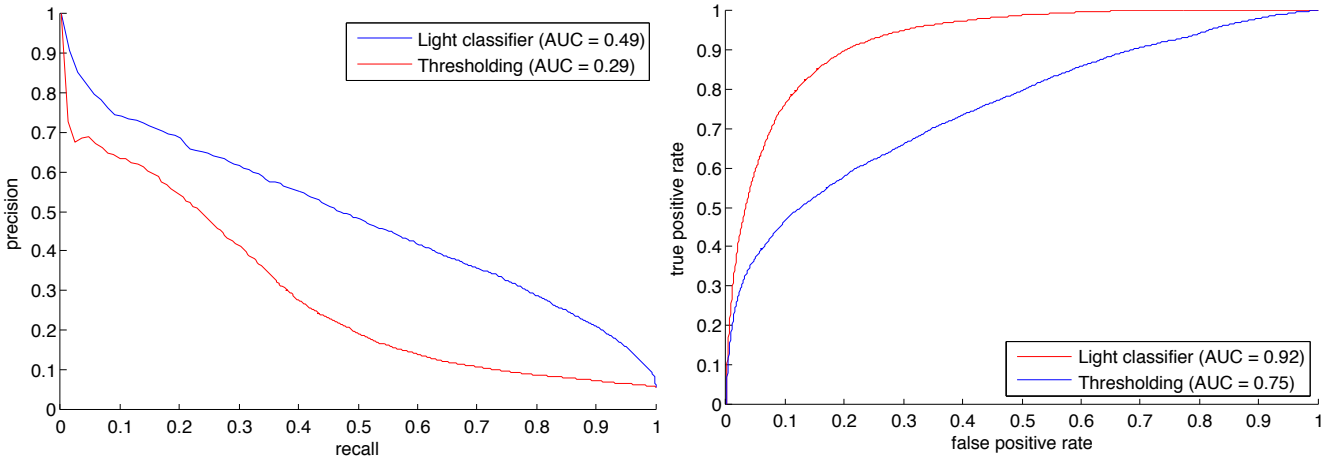


Fig. 8. Precision-recall and ROC curves for our light classifier and thresholding baseline. Our classifier steadily outperforms simple thresholding methods.

Table I. Light Detection Evaluation

Panorama class	Intersection / Union >80%	Intersection / Union >90%	Intersection / Union Our classifier	Distance accuracy
cave	38.52	39.41	48.85	81.49
church	15.39	13.41	27.68	99.10
corridor	28.63	28.55	44.68	81.81
hotel room	10.78	14.87	31.48	97.30
living room	16.53	15.83	38.81	88.65
lobby atrium	26.60	25.93	42.67	97.06
museum	20.49	20.58	34.96	80.77
old building	41.42	46.77	73.46	93.67
restaurant	26.88	20.67	42.79	99.55
shop	12.72	26.34	33.19	96.98
subway station	8.71	14.31	31.26	85.25
theater	14.35	15.22	26.66	96.80
train interior	32.57	25.08	56.04	98.92
workshop	26.59	31.05	46.83	99.36
all	22.87	24.14	41.38	92.62

Quantitative analysis of our light detection algorithm. For five panoramas in each class, we hand-label the pixels/directions which contribute significant light to the panorama, including the distance the source is from the camera (labelled as one of the following categories: near, medium, far, infinite). From these annotations, we build a light classifier, and report the classification accuracy of our classifier and baseline classifier (thresholding the image at the $\{80, 90\}^{th}$ intensity percentiles). We report the intersection over union (true positive rate divided by total number of detections) for these methods; for each class, our detector significantly outperforms the baselines. Our classifier also estimates the distance of the source, and achieves distance classification accuracy much higher than chance.

Table II. Depth Error on Synthetic Study Scenes

	Lighting	Our method	[Karsch et al. 2012]
bedroom	A	0.0344	0.0383
	B	0.0394	0.0421
	C	0.0376	0.0398
bedroom mean		0.0371	0.0400
corridor	A	0.0478	0.0442
	B	0.0443	0.0502
	C	0.0443	0.0502
corridor mean		0.0455	0.0489
table	A	0.0290	0.0249
	B	0.0489	0.0491
	C	0.0481	0.0482
table mean		0.0420	0.0407
outdoor	A	0.0275	0.0260
	B	0.0319	0.0300
	C	0.0525	0.0515
outdoor mean		0.0373	0.0359
mean over all scenes		0.0405	0.0414

We compute depth error as the pixel-wise norm (squared) between two depth maps, up to a scale and translation (since the depth from the synthetic scenes and our estimates do not have consistent units). Formally, the depth in this table is computed as $\min_{s,t} ||D_{gt} - sD_{est} - t||^2$, where D_{gt} is the ground truth depth, and D_{est} is the depth estimated by one of the above methods. Quantitatively, we see that our depth maps are slightly better than the method of Karsch et al. over all images.

Table III. Depth Error on Make3D Dataset (Outdoors)

Method	**rel**	**$\log_{10}$**	**RMS**
Make3D [Saxena et al. 2009]	0.370	0.187	N/R
Semantic Labels [Liu et al. 2010]	0.375	**0.148**	N/A
θ-MRF [Li et al. 2011]	N/R	N/R	**15.0**
Depth Transfer [Karsch et al. 2012]	0.361	**0.148**	15.1
Depth Transfer+GP (ours)	**0.352**	0.149	15.3

Errors computed using the standard split (400 training images, 134 test). Our method achieves state of the art results for relative error (**rel**), and performs comparably to other methods for log base 10 and root mean squared error.

Table IV. Depth Error on NYUv2 Dataset (Indoors)

Method	**rel**	**$\log_{10}$**	**RMS**
Depth Transfer [Karsch et al. 2012]	0.382	0.137	**1.2**
Depth Transfer+GP (ours)	**0.362**	**0.133**	1.3

Errors computed on the NYUv2 dataset [Silberman et al. 2012] containing 1449 indoor images. For evaluation, 100 images were randomly selected for testing; the rest were used for training.

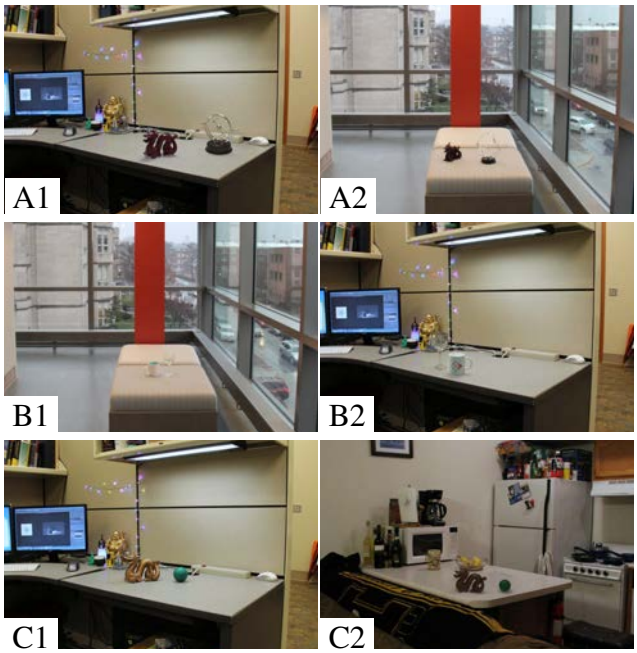


Fig. 9. Three example trials from our user study. In the study, users were shown two side-by-side pictures; one photograph is real, and the other has synthetic objects inserted into it. Users were instructed to choose the picture from the pair that looked the most realistic. For each row, which of the pair would you choose? (Answers written below; best viewed in color at high-resolution.)

A1: synthetic, A2: real; B1: real, B2: synthetic; C1: real, C2 synthetic

Table V. Absolute Illumination Error on Synthetic Study Scenes

Condition		Our method	Matching	[Khan et al. 2006]	[Lalonde et al. 2009]	Lighting notes
bedroom	A	0.1379	0.1052	0.0652	-	sunlight through window
	B	0.1774	0.1492	0.1031	-	overcast skylight with room lamps
	C	0.0360	0.0562	0.0601	-	indoor lamps only
bedroom mean		0.1171	0.1035	0.0761	-	
corridor	A	0.1625	0.3115	0.2158	-	sunlight through windows
	B	0.1142	0.2343	0.1285	-	dusk sunlight
	C	0.1563	0.1304	0.0470	-	overhead diffuse lights; night time
corridor mean		0.1443	0.2254	0.1304	-	
table	A	0.0521	0.0587	0.0741	-	overhead lamps only
	B	0.0821	0.2202	0.1594	-	diffuse light from a distance
	C	0.1229	0.1382	0.0898	-	overhead + distant lights, cool hue
table mean		0.0857	0.1390	0.1078	-	
outdoor	A	0.0736	0.1709	0.0725	0.1746	direct sunlight
	B	0.0711	0.1220	0.1061	0.2684	overcast
	C	0.0701	0.2270	0.1062	0.2004	dusk sunlight
outdoor mean		0.0716	0.1733	0.0949	0.2145	
mean over all scenes		0.1047	0.1603	0.1023	0.2145	

We compute error by rendering nine randomly placed objects (with varying materials; see Fig 10) into two scenes (e.g. a ground truth synthetic scene, and the corresponding scene produced by our method). The final error in each cell is computed as the absolute pixel-wise difference between the two renderings, averaged over all nine objects/pixels. Interestingly, but perhaps not surprisingly, the results are not very consistent with the people's preferences in our "synthetic image" user study – for example, on average, the method of Khan et al. achieves slightly lower error than our method, but in the user study, our method saw nearly a 5% gain (over the method of Khan et al.) in confusion.

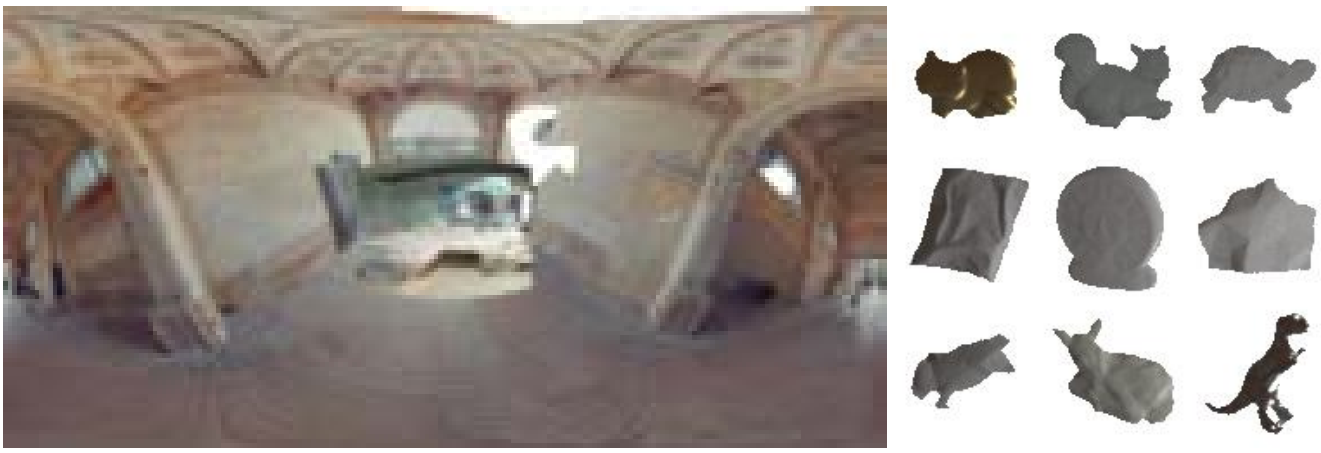
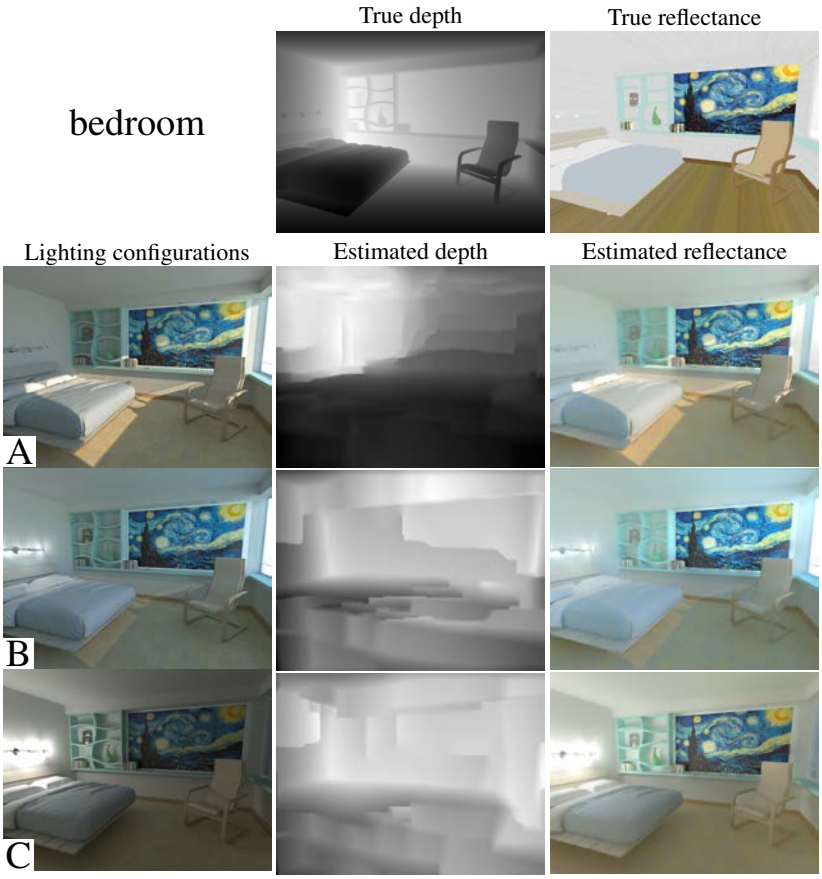


Fig. 10. The nine shapes/materials (from the MIT intrinsic dataset [Grosse et al. 2009]) used to measure quantitative illumination error (in Fig V), and also used for our training loss metric (Sec 5.2 in the main paper). In the above images, the objects on the left were rendered into the illumination on the right (corresponding to our estimated illumination for the "bedroom A" scene; see Fig 11). Each of the objects are placed randomly in the scene so that no part of the object is occluded or out-of-view, then rendered with a given illumination environment and cut out from the background (error is only computed on pixels occupied by the object). Each object contains one of LuxRender's preset, physically based materials (gold, glossy, matte, velvet, etc).

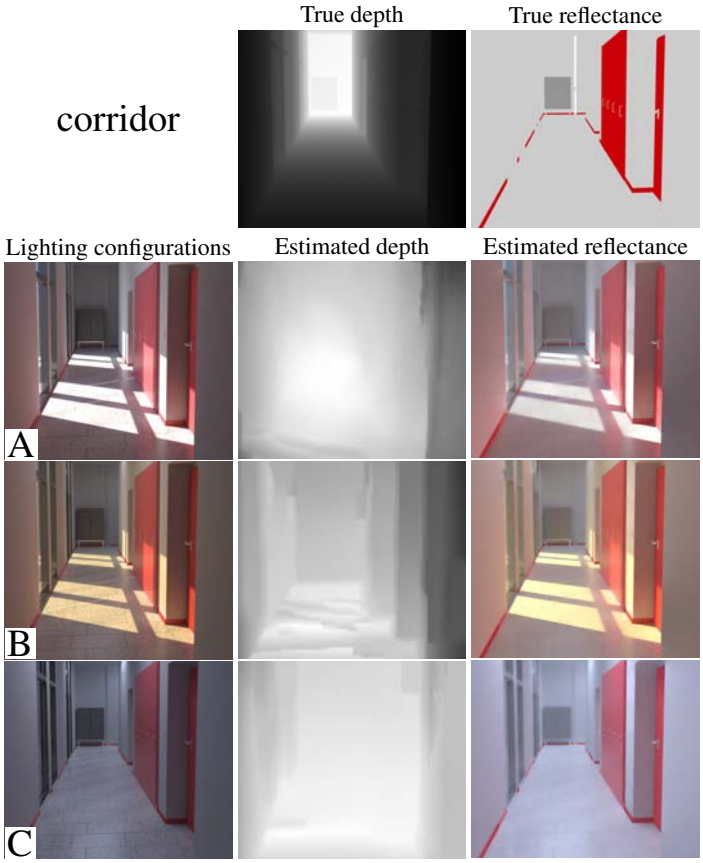


Fraction of times subjects chose a synthetic result over the true insertion (bedroom scene).

Condition	N	ours	Khan	match
direct light	99	.434	.480	.521
diffuse light	345	.443	.453	.291
large coverage	444	.441	.459	.343
small coverage	0	-	-	-
total	444	.441	.459	.343

Fig. 11. Automatic scene estimates compared to the true depth, diffuse reflectance and illumination for the "bedroom" scene.

Fig. 12. Example results from our user study. New objects have been inserted into the synthetic scene using the approaches tested in our study.



Fraction of times subjects chose a synthetic result over the true insertion (corridor scene).

Condition	N	ours	Khan	match
direct light	197	.335	.242	.355
diffuse light	247	.389	.250	.231
large coverage	296	.338	.216	.243
small coverage	148	.419	.308	.371
total	444	.365	.247	.286

Fig. 13. Automatic scene estimates compared to the true depth, diffuse reflectance and illumination for the "corridor" scene.

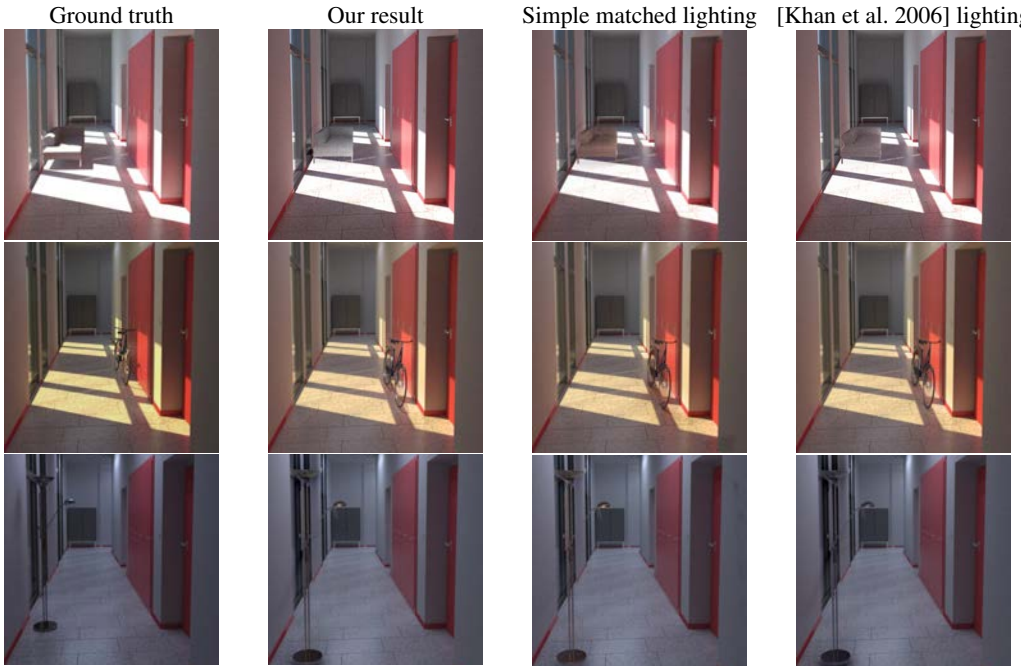


Fig. 14. Example results from our user study. New objects have been inserted into the synthetic scene using the approaches tested in our study.

Fraction of times subjects chose a synthetic result over the true insertion (table scene).

Condition	N	ours	Khan	match
direct light	0	-	-	-
diffuse light	444	.324	.226	.207
large object	148	.324	.226	.257
small object	296	.324	.226	.182
total	444	.324	.226	.207

Fig. 15. Automatic scene estimates compared to the true depth, diffuse reflectance and illumination for the "table" scene.

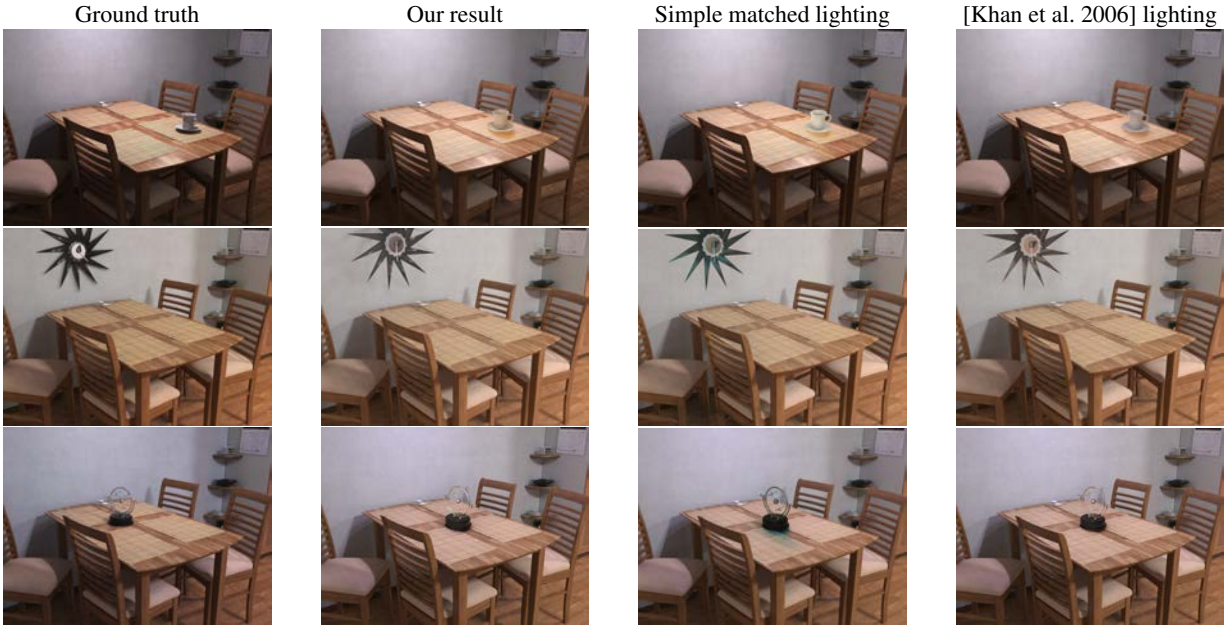
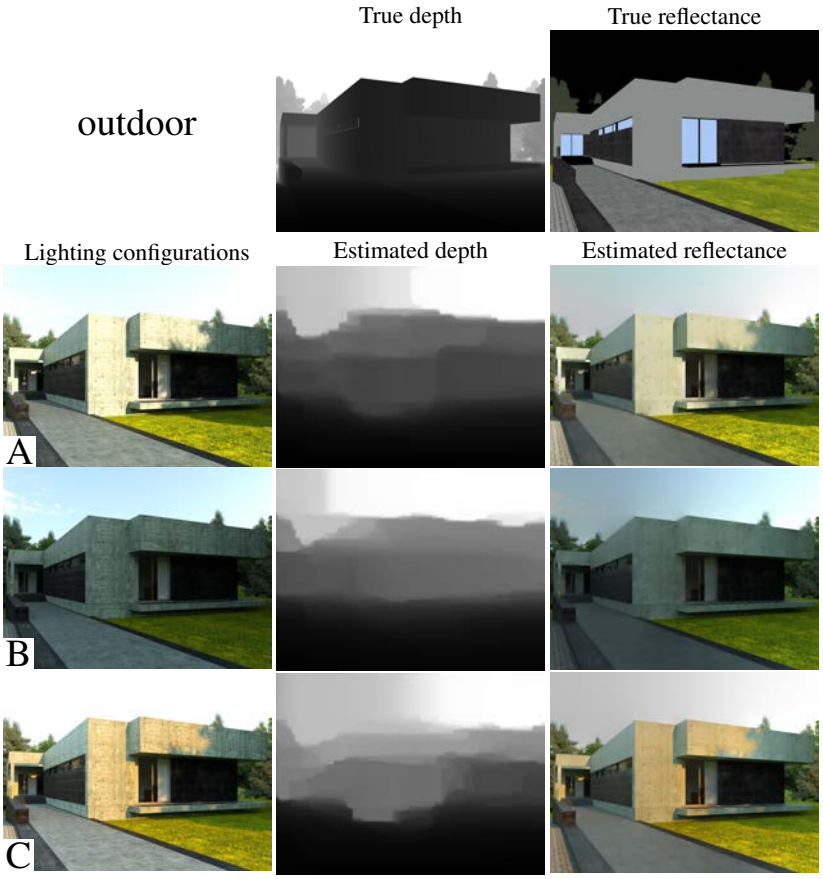


Fig. 16. Example results from our user study. New objects have been inserted into the synthetic scene using the approaches tested in our study.



Fraction of times subjects chose a synthetic result over the true insertion (outdoor scene).

Condition	N	ours	Khan	Lalonde
direct light	296	.253	.192	.250
diffuse light	148	.358	.507	.221
large coverage	444	.288	.297	.240
small coverage	0	-	-	-
total	444	.288	.297	.240

Fig. 17. Automatic scene estimates compared to the true depth, diffuse reflectance and illumination for the "outdoor" scene.

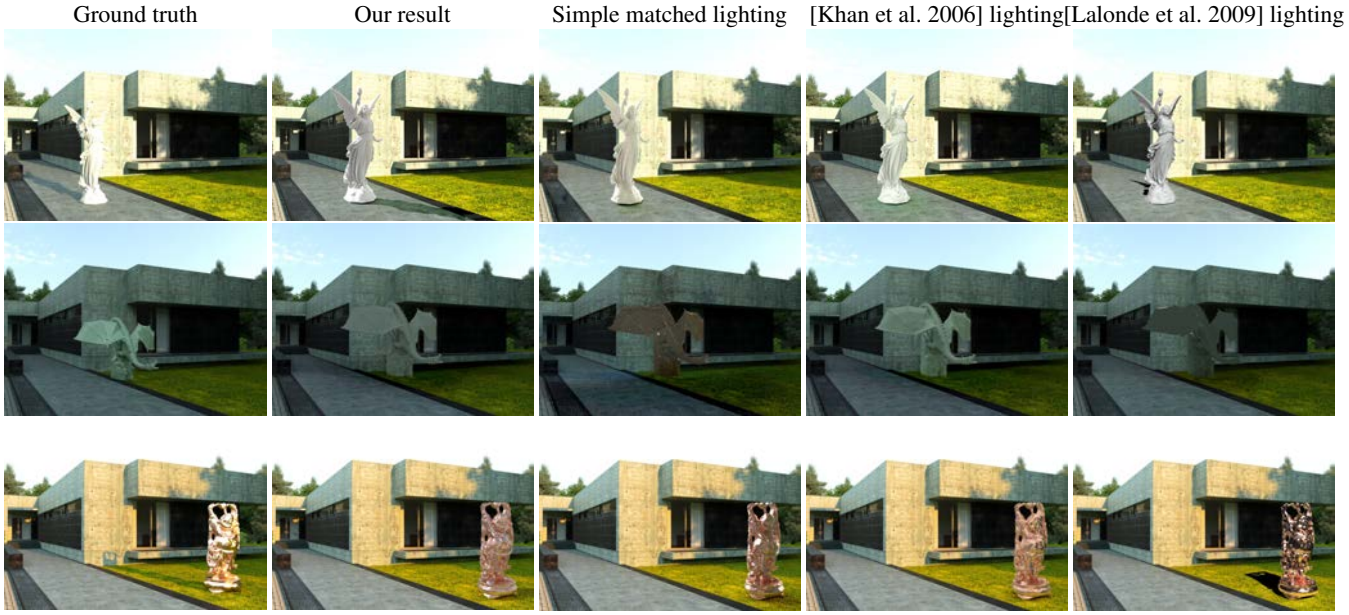


Fig. 18. Example results from our user study. New objects have been inserted into the synthetic scene using the approaches tested in our study.

Fig. 19. We conducted a preliminary study to measure how realistic our synthetic study scenes appeared to people. In the study, pairs of real and synthetic images were shown to subjects (rows in the above figure), and subjects were asked to choose the image they felt looked most realistic. The percentage next in each row shows how often users preferred the synthetic image to the actual, real photo. Averages were obtained with 38 subjects (50 in total, but 12 were discarded either because the subject failed the in-test qualification, or indicated he/she had seen one or more of the photos prior to the study). All real photos are shown on the right for demonstration, but placement/ordering was permuted in the study.

Fig. 20. Some of the most and least realistic results from our study (in terms of how many times they were confused as real). Each row shows either the best or worst result from our method (left) or the Khan baseline (right), indicated by the text in each row. E.g. the first row shows our best result and the corresponding Khan result for comparison.

ACKNOWLEDGMENTS

We are thankful to Alexey Trofimov and Moyan Brenn for making their photos available, as well as Flickr users xerostomia, salvadonica, brianteutsch, denniswong, jamiejohn, sneakerdog, tiarescott, salvadonica, smemon, pocait, state-records-nsw, mr_t_in_dc, vxla, smomashup1, parisonponce, waytru, oskay, pascalrossini, kurmanka, jmrosenfeld, lirumlar, glenbledsoe, wonderlane, ivyfield, jeofficial, and jurvetson. This work was funded in part by the NSF GRFP, NSF Award IIS 0916014, ONR MURI Award N00014-10-10934, and an NSF Expeditions Award, IIS-1029035. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or of the Office of Naval Research.

REFERENCES

GROSSE, R., JOHNSON, M. K., ADELSON, E. H., AND FREEMAN, W. 2009. Ground truth dataset and baseline evaluations for intrinsic image algorithms. *ICCV*.

HARTLEY, R. AND ZISSERMAN, A. 2003. *Multiple view geometry in computer vision*.

KARSCH, K., HEDAU, V., FORSYTH, D., AND HOIEM, D. 2011. Rendering synthetic objects into legacy photographs. In *SIGGRAPH Asia*. 157:1–157:12.

KARSCH, K., LIU, C., AND KANG, S. B. 2012. Depth extraction from video using non-parametric sampling. In *ECCV*.

KHAN, E. A., REINHARD, E., FLEMING, R. W. W., AND BÜLTHOFF, H. H. 2006. Image-based material editing. In *ACM SIGGRAPH*.

LALONDE, J., EFROS, A. A., AND NARASIMHAN, S. 2009. Estimating Natural Illumination from a Single Outdoor Image. *ICCV*.

LI, C., SAXENA, A., AND CHEN, T. 2011. θ-mrf: Capturing spatial and semantic structure in the parameters for scene understanding. In *NIPS*. 549–557.

LIU, B., GOULD, S., AND KOLLER, D. 2010. Single image depth estimation from predicted semantic labels. In *CVPR*. 1253–1260.

SAXENA, A., SUN, M., AND NG, A. Y. 2009. Make3D: Learning 3D Scene Structure from a Single Still Image. *IEEE Transactions on Pattern Analysis and Machine Intelligence 31,* 5, 824–840.

SILBERMAN, N., HOIEM, D., KOHLI, P., AND FERGUS, R. 2012. Indoor segmentation and support inference from rgbd images. In *ECCV*.

XIAO, J., EHINGER, K. A., OLIVA, A., AND TORRALBA, A. 2012. Recognizing scene viewpoint using panoramic place representation. In *CVPR*.