Depth Estimation using Monocular and Stereo Cues

Ashutosh Saxena, Jamie Schulte and **Andrew Y. Ng**
Computer Science Department
Stanford University, Stanford, CA 94305
{asaxena,schulte,ang}@cs.stanford.edu

Abstract

Depth estimation in computer vision and robotics is most commonly done via stereo vision (stereopsis), in which images from two cameras are used to triangulate and estimate distances. However, there are also numerous monocular visual cues—such as texture variations and gradients, defocus, color/haze, etc.—that have heretofore been little exploited in such systems. Some of these cues apply even in regions without texture, where stereo would work poorly. In this paper, we apply a Markov Random Field (MRF) learning algorithm to capture some of these monocular cues, and incorporate them into a stereo system. We show that by adding monocular cues to stereo (triangulation) ones, we obtain significantly more accurate depth estimates than is possible using either monocular or stereo cues alone. This holds true for a large variety of environments, including both indoor environments and unstructured outdoor environments containing trees/forests, buildings, etc. Our approach is general, and applies to incorporating monocular cues together with any off-the-shelf stereo system.

1 Introduction

Consider the problem of estimating depth from two images taken from a pair of stereo cameras (Fig. 1). The most common approach for doing so is stereopsis (stereo vision), in which depths are estimated by triangulation using the two images. Over the past few decades, researchers have developed very good stereo vision systems (see [Scharstein and Szeliski, 2002] for a review). Although these systems work well in many environments, stereo vision is fundamentally limited by the baseline distance between the two cameras. Specifically, the depth estimates tend to be inaccurate when the distances considered are large (because even very small triangulation/angle estimation errors translate to very large errors in distances). Further, stereo vision also tends to fail for textureless regions of images where correspondences cannot be reliably found.

Beyond stereo/triangulation cues, there are also numerous *monocular* cues—such as texture variations and gradients, defocus, color/haze, etc.—that contain useful and important depth information. Even though humans perceive depth by seamlessly combining many of these stereo and monocular cues, most work on depth estimation has focused on stereo vision, and on other algorithms that require multiple images such as structure from motion [Forsyth and Ponce, 2003] or depth from defocus [Klarquist *et al.*, 1995].

Figure 1: Two images taken from a stereo pair of cameras, and the depthmap calculated by a stereo system. Colors in the depthmap indicate estimated distances from the camera.

In this paper, we look at how monocular cues from a single image can be incorporated into a stereo system. Estimating depth from a single image using monocular cues requires a significant amount of prior knowledge, since there is an intrinsic ambiguity between local image features and depth variations. In addition, we believe that monocular cues and (purely geometric) stereo cues give largely orthogonal, and therefore complementary, types of information about depth. Stereo cues are based on the difference between two images and do not depend on the content of the image. The images can be entirely random, and it will generate a pattern of disparities (e.g., random dot stereograms [Blthoff *et al.*, 1998]). On the other hand, depth estimates from monocular cues is entirely based on prior knowledge about the environment and global structure of the image. There are many examples of beautifully engineered stereo systems in the literature, but the goal of this work is not to directly improve on, or compare against, these systems. Instead, our goal is to investigate how monocular cues can be integrated with any reasonable stereo system, to (hopefully) obtain better depth estimates than the stereo system alone.

Depth estimation from monocular cues is a difficult task, which requires that we take into account the global structure of the image. [Saxena *et al.*, 2006a] applied supervised learning to the problem of estimating depth from single monocular images of unconstrained outdoor and indoor environments. [Michels *et al.*, 2005] used supervised learning to estimate 1-D distances to obstacles, for the application of driving a remote controlled car autonomously. Methods such as shape from shading [Zhang *et al.*, 1999] rely on purely photometric properties, assuming uniform color and texture; and hence are not applicable to the unconstrained/textured images that we consider. [Delage *et al.*, 2006] generated 3-d models from an image of indoor scenes containing only walls and floor. [Hoiem *et al.*, 2005] also considered monocular 3-d reconstruction, but focused on generating visually pleasing graphical images by classifying the scene as sky, ground, or vertical planes, rather than accurate metric depthmaps.

Building on [Saxena *et al.*, 2006a], our approach is based on incorporating monocular and stereo cues for modeling depths and relationships between depths at different points in the image using a hierarchical, multi-scale Markov Random Field (MRF). MRFs and their variants are a workhorse of machine learning, and have been successfully applied to numerous applications in which local features were insufficient and more contextual information must be used.[1] Taking a supervised learning approach to the problem of depth estimation, we designed a custom 3-d scanner to collect training data comprising a large set of stereo pairs and their corresponding ground-truth depthmaps. Using this training set, we model the posterior distribution of the depths given the monocular image features and the stereo disparities. Though learning in our MRF model is approximate, MAP posterior inference is tractable via linear programming.

Although depthmaps can be estimated from single monocular images, we demonstrate that by combining both monocular and stereo cues in our model, we obtain significantly more accurate depthmaps than is possible from either alone. We demonstrate this on a large variety of environments, including both indoor environments and unstructured outdoor environments containing trees/forests, buildings, etc.

2 Visual Cues for Depth Perception

Humans use numerous visual cues for 3-d depth perception, which can be grouped into two categories: Monocular and Stereo. [Loomis, 2001]

2.1 Monocular Cues

Humans have an amazing ability to judge depth from a single image. This is done using monocular cues such as texture variations and gradients, occlusion, known object sizes, haze, defocus, etc. [Loomis, 2001; Blthoff *et al.*, 1998; Saxena *et al.*, 2006a]. Some of these cues, such as haze (resulting from atmospheric light scattering) and defocus (blurring of objects not in focus), are local cues; i.e., the estimate of depth is dependent only on the local image properties. Many objects' textures appear different depending on the distance to them. Texture gradients, which capture the distribution of the direction of edges, also help to indicate depth.[2]

Some of these monocular cues are based on prior knowledge. For example, humans remember that a structure of a particular shape is a building, sky is blue, grass is green, trees grow above the ground and have leaves on top of them, and so on. These cues help to predict depth in environments similar to those which they have seen before. Many of these cues rely on "contextual information," in the sense that they are global properties of an image and cannot be inferred from small image patches. For example, occlusion cannot be determined if we look at just a small portion of an occluded object. Although local information such as the texture and colors of a patch can give some information about its depth, this is usually insufficient to accurately determine its absolute depth. Therefore, we need to look at the *overall* organization of the image to estimate depths.

Figure 2: The filters used for computing texture variations and gradients. The first 9 are Laws' masks, and the last 6 are oriented edge filters.

2.2 Stereo Cues

Each eye receives a slightly different view of the world and stereo vision combines the two views to perceive 3-d depth. An object is projected onto different locations on the two retinae (cameras in the case of a stereo system), depending on the distance of the object. The retinal (stereo) disparity varies with object distance, and is inversely proportional to the distance of the object. Thus, disparity is not an effective cue for small depth differences at large distances.

3 Features

3.1 Monocular Features

In our approach, we divide the image into small rectangular patches, and estimate a single depth value for each patch. Similar to [Saxena *et al.*, 2006a], we use two types of features: *absolute* features—used to estimate the absolute depth at a particular patch—and *relative* features, which we use to estimate relative depths (magnitude of the difference in depth between two patches).[3] We chose features that capture three types of local cues: texture variations, texture gradients, and color, by convolving the image with 17 filters (9 Laws' masks, 6 oriented edge filters, and 2 color filters, Fig. 2). [Saxena *et al.*, 2006b]

We generate the *absolute* features by computing the sum-squared energy of each of these 17 filter outputs over each

[1] Examples include text segmentation [Lafferty *et al.*, 2001], image labeling [He *et al.*, 2004; Kumar and Hebert, 2003] and smoothing disparity to compute depthmaps in stereo vision [Tappen and Freeman, 2003]. Because MRF learning is intractable in general, most of these model are trained using pseudo-likelihood.

[2] For textured environments which may not have well-defined edges, texture gradient is a generalization of the edge directions. For example, a grass field when viewed at different distances has different distribution of edges.

[3] If two neighbor patches of an image display similar features, humans would often perceive them to be parts of the same object, and to have similar depth values.

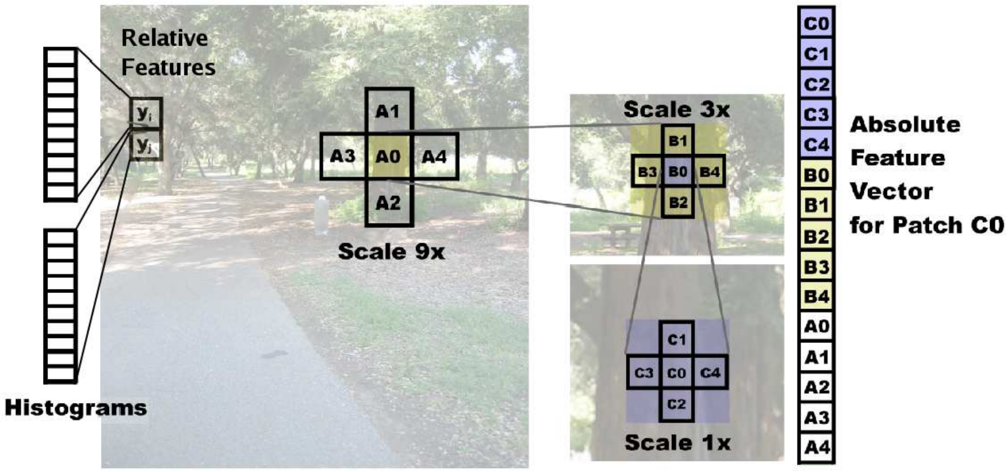

Figure 3: The absolute depth feature vector for a patch, which includes the immediate neighbors, and the distant neighbors in larger scales. The relative depth features for each patch compute histograms of the filter outputs.

patch. Since local image features centered on the patch are insufficient, we attempt to capture more global information by using image features extracted at multiple spatial scales[4] for that patch as well as the 4 neighboring patches. (Fig. 3) This results in a *absolute* feature vector of $(1 + 4) * 3 * 17 = 255$ dimensions. For *relative* features, we use a 10-bin histogram for each filter output for the pixels in the patch, giving us $10 * 17 = 170$ values y_i for each patch i. Therefore, our features for the edge between patch i and j are the difference $y_{ij} = |y_i - y_j|$.

3.2 Disparity from stereo correspondence

Depth estimation using stereo vision from two images (taken from two cameras separated by a baseline distance) involves three steps: First, establish correspondences between the two images. Then, calculate the relative displacements (called "disparity") between the features in each image. Finally, determine the 3-d depth of the feature relative to the cameras, using knowledge of the camera geometry.

Stereo correspondences give reliable estimates of disparity, except when large portions of the image are featureless (i.e., correspondences cannot be found). Further, the accuracy depends on the baseline distance between the cameras. In general, for a given baseline distance between cameras, the accuracy decreases as the depth values increase. This is because small errors in disparity then translate into huge errors in depth estimates. In the limit of very distant objects, there is no observable disparity, and depth estimation generally fails. Empirically, depth estimates from stereo tend to become unreliable when the depth exceeds a certain distance.

Our stereo system finds good feature correspondences between the two images by rejecting pixels with little texture, or where the correspondence is otherwise ambiguous.[5] We use the sum-of-absolute-differences correlation as the metric score to find correspondences. [Forsyth and Ponce, 2003] Our cameras (and algorithm) allow sub-pixel interpolation accuracy of 0.2 pixels of disparity. Even though we use a fairly basic implementation of stereopsis, the ideas in this paper can just as readily be applied together with other, perhaps better, stereo systems.

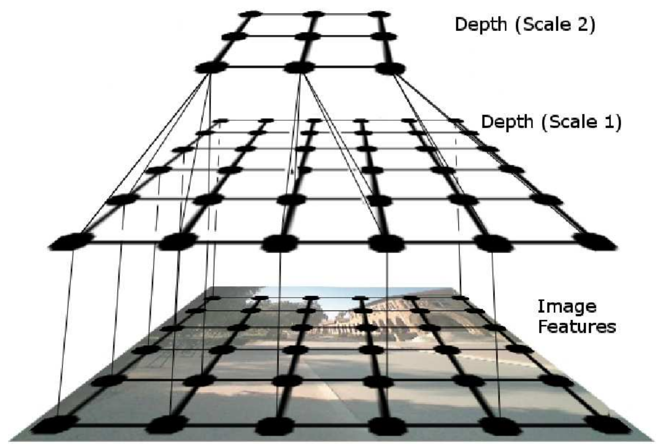

Figure 4: The multi-scale MRF model for modeling relation between features and depths, relation between depths at same scale, and relation between depths at different scales. (Only 2 out of 3 scales, and a subset of the edges, are shown.)

4 Probabilistic Model

Our learning algorithm is based on a Markov Random Field (MRF) model that incorporates monocular and stereo features, and models depth at multiple spatial scales (Fig. 4). The MRF model is discriminative, i.e., it models the depths d as a function of the image features X: $P(d|X)$. The depth of a particular patch depends on the monocular features of the patch, on the stereo disparity, and is also related to the depths of other parts of the image. For example, the depths of two

[4]The patches at each spatial scale are arranged in a grid of equally sized non-overlapping regions that cover the entire image. We use 3 scales in our experiments.

[5]More formally, we reject any feature where the best match is not significantly better than all other matches within the search window.

$$P_G(d|X;\theta,\sigma) = \frac{1}{Z_G}\exp\left(-\frac{1}{2}\sum_{i=1}^{M}\left(\frac{(d_i(1)-d_{i,\text{stereo}})^2}{\sigma^2_{i,\text{stereo}}} + \frac{(d_i(1)-x_i^T\theta_r)^2}{\sigma^2_{1r}} + \sum_{s=1}^{3}\sum_{j\in N_s(i)}\frac{(d_i(s)-d_j(s))^2}{\sigma^2_{2rs}}\right)\right) \quad (1)$$

$$P_L(d|X;\theta,\lambda) = \frac{1}{Z_L}\exp\left(-\sum_{i=1}^{M}\left(\frac{|d_i(1)-d_{i,\text{stereo}}|}{\lambda_{i,\text{stereo}}} + \frac{|d_i(1)-x_i^T\theta_r|}{\lambda_{1r}} + \sum_{s=1}^{3}\sum_{j\in N_s(i)}\frac{|d_i(s)-d_j(s)|}{\lambda_{2rs}}\right)\right) \quad (2)$$

adjacent patches lying in the same building will be highly correlated. Therefore, we model the relation between the depth of a patch and its neighbors at multiple spatial scales.

4.1 Gaussian Model

We first propose a jointly Gaussian MRF (Eq. 1), parameterized by θ and σ. We define $d_i(s)$ to be the depth of a patch i at scale $s \in \{1,2,3\}$, with the constraint $d_i(s+1) = (1/5)\sum_{j\in\{i,N_s(i)\}} d_j(s)$. I.e., the depth at a higher scale is constrained to be the average of the depths at lower scales. Here, $N_s(i)$ are the 5 neighbors (including itself) of patch i at scale s. M is the total number of patches in the image (at the lowest scale); x_i is the absolute feature vector for patch i; $d_{i,stereo}$ is the depth estimate obtained from disparity;[6] Z_G is the normalization constant.

The first term in the exponent in Eq. 1 models the relation between the depth and the estimate from stereo disparity. The second term models the relation between the depth and the multi-scale features of a single patch i. The third term places a soft "constraint" on the depths to be smooth. We first estimate the θ_r parameters in Eq. 1 by maximizing the conditional likelihood $p(d|X;\theta_r)$ of the training data; keeping σ constant. [Saxena *et al.*, 2006a] We then achieve selective smoothing by modeling the "variance" term σ^2_{2rs} in the denominator of the third term as a linear function of the patches i and j's relative depth features y_{ijs}. The σ^2_{1r} term gives a measure of uncertainty in the second term, which we learn as a linear function of the features x_i. This is motivated by the observation that in some cases, depth cannot be reliably estimated from the local monocular features. In this case, one has to rely more on neighboring patches' depths or on stereo cues to infer a patch's depth.

Modeling Uncertainty in Stereo

The errors in disparity are modeled as either Gaussian [Das and Ahuja, 1995] or via some other, heavier-tailed distribution (e.g., [Szelinski, 1990]). Specifically, the errors in disparity have two main causes: (a) Assuming unique/perfect correspondence, the disparity has a small error due to image noise (including aliasing/pixelization), which is well modeled by a Gaussian. (b) Occasional errors in correspondence causes larger errors, which results in a heavy-tailed distribution for disparity. [Szelinski, 1990]

We estimate depths on a log scale as $d = \log(C/g)$ from disparity g, with camera parameters determining C. If the standard deviation is σ_g in computing disparity from stereo images (because of image noise, etc.), then the standard deviation of the depths[7] will be $\sigma_{d,\text{stereo}} \approx \sigma_g/g$. For our stereo system, we have that σ_g is about 0.2 pixels;[8] this is then used to estimate $\sigma_{d,\text{stereo}}$. Note therefore that $\sigma_{d,\text{stereo}}$ is a function of the estimated depth, and specifically, it captures the fact that variance in depth estimates is larger for distant objects than for closer ones.

When given a new test image, MAP inference for depths d can be derived in closed form.

4.2 Laplacian Model

In our second model (Eq. 2), we replace the L_2 terms with L_1 terms. This results in a model parameterized by θ and by λ, the *Laplacian spread* parameters, instead of Gaussian variance parameters. Since ML parameter estimation in the Laplacian model is intractable, we learn these parameters following an analogy to the Gaussian case. [Saxena *et al.*, 2006a] Our motivation for using L_1 terms is three-fold. First, the histogram of relative depths $(d_i - d_j)$ is close to a Laplacian distribution empirically, which suggests that it is better modeled as one. Second, the Laplacian distribution has heavier tails, and is therefore more robust to outliers in the image features and errors in the training-set depthmaps (collected with a laser scanner; see Section 5.1). Third, the Gaussian model was generally unable to give depthmaps with sharp edges; in contrast, Laplacians tend to model sharp transitions/outliers better. (See Section 5.2.) Given a new test image, MAP posterior inference for the depths d is tractable, and is easily solved using linear programming (LP).

5 Experiments

5.1 Laser Scanner

We designed a 3-d scanner to collect stereo image pairs and their corresponding depthmaps (Fig. 6). The scanner uses the SICK laser device which gives depth readings in a vertical column, with a $1.0°$ resolution. To collect readings along the other axis (left to right), we mounted the SICK laser on a panning motor. The motor rotates after each vertical scan to collect laser readings for another vertical column, with a $0.5°$ horizontal angular resolution. The depthmap is later reconstructed using the vertical laser scans, the motor readings

[6] In this work, we directly use $d_{i,stereo}$ as the stereo cue. In [Saxena *et al.*, 2007], we use a library of features created from stereo depths as the cues for identifying a grasp point on objects.

[7] Using the delta rule from statistics: $\text{Var}(f(x)) \approx (f'(x))^2\text{Var}(x)$, derived from a second order Taylor series approximation of $f(x)$.

[8] One can also envisage obtaining a better estimate of σ_g as a function of a match metric used during stereo correspondence, [Scharstein and Szeliski, 2002] such as normalized sum of squared differences; or learning σ_g as a function of disparity/texture based features.

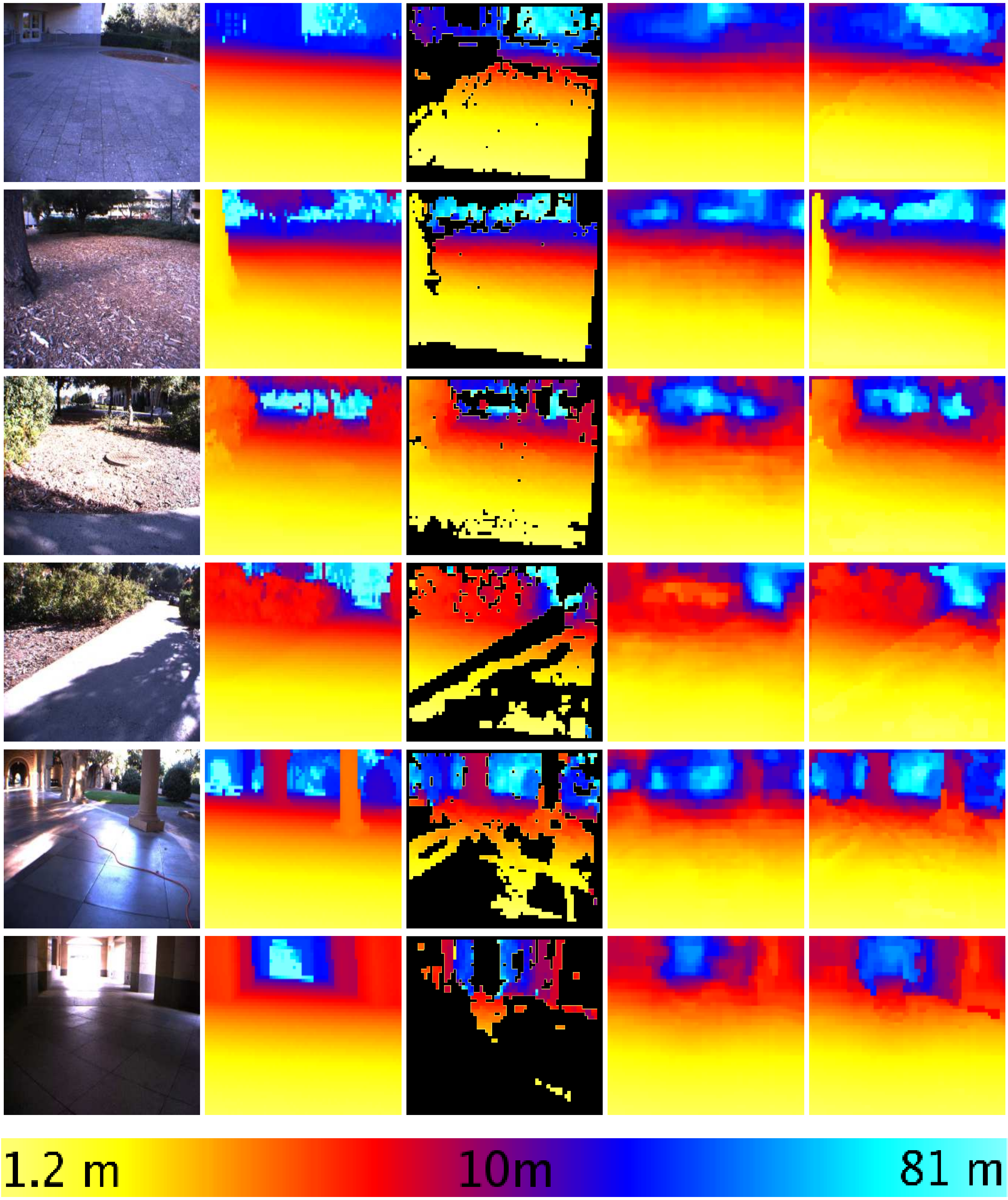

Figure 5: Results for a varied set of environments, showing one image of the stereo pairs (column 1), ground truth depthmap collected from 3-d laser scanner (column 2), depths calculated by stereo (column 3), depths predicted by using monocular cues only (column 4), depths predicted by using both monocular and stereo cues (column 5). The bottom row shows the color scale for representation of depths. Closest points are 1.2 m, and farthest are 81m. (**Best viewed in color**)

Figure 6: The custom 3-d scanner to collect stereo image pairs and the corresponding depthmaps.

and known position and pose of the laser device and the cameras. The laser range finding equipment was mounted on a LAGR (Learning Applied to Ground Robotics) robot. The LAGR vehicle is equipped with sensors, an onboard computer, and Point Grey Research Bumblebee stereo cameras, mounted with a baseline distance of 11.7cm.

We collected a total of 257 stereo pairs+depthmaps, with an image resolution of 1024x768 and a depthmap resolution of 67x54. In the experimental results reported here, 75% of the images/depthmaps were used for training, and the remaining 25% for hold-out testing. The images consist of a wide variety of scenes including natural environments (forests, trees, bushes, etc.), man-made environments (buildings, roads, trees, grass, etc.), and purely indoor environments (corridors, etc.). Due to limitations of the laser, the depthmaps had a maximum range of 81m (the maximum range of the laser scanner), and had minor additional errors due to reflections and missing laser scans. Prior to running our learning algorithms, we transformed all the depths to a log scale so as to emphasize multiplicative rather than additive errors in training.

5.2 Results and Discussion

We evaluate the performance of the model on our test-set comprising a wide variety of real-world images. To quantitatively compare effects of various cues, we report results from the following classes of algorithms that use monocular and stereo cues in different ways:
(i) **Baseline**: The model, trained without any features, predicts the mean value of depth in the training depthmaps.
(ii) **Stereo**: Raw stereo depth estimates, with the missing values set to the mean value of depth in the training depthmaps.
(iii) **Stereo (smooth)**: This method performs interpolation and region filling; using the Laplacian model without the second term (which models depths as a function of monocular cues) in Eq. 2, and also without using monocular cues to estimate λ_2 as a function of the image.
(iv) **Mono (Gaussian)**: Depth estimates using only monocular cues, without the first term in the exponent of the Gaussian model.
(v) **Mono (Lap)**: Depth estimates using only monocular cues, without the first term in the exponent of the Laplacian model.
(vi) **Stereo+Mono**: Depth estimates using the full model.

Table 1: The average errors (RMS errors gave similar results) for various cues and models, on a log scale (base 10).

Algorithm	All	Campus	Forest	Indoor
Baseline	.341	.351	.344	.307
Stereo	.138	.143	.113	.182
Stereo (smooth)	.088	.091	.080	.099
Mono (Gaussian)	.093	.095	.085	.108
Mono (Lap)	.090	.091	.082	.105
Stereo+Mono (Lap)	**.074**	**.077**	**.069**	**.079**

Table 1 shows that the performance is significantly improved when we combine both mono and stereo cues. The algorithm is able to estimate depths with an error of .074 orders of magnitude,[9] which represents a significant improvement over stereo (smooth) performance of .088.

Fig. 5 shows that the model is able to predict depthmaps (column 5) in a variety of environments. It also demonstrates how the model takes the best estimates from both stereo and monocular cues to estimate more accurate depthmaps. For example, in row 6 (Fig. 5), the depthmap generated by stereo (column 3) is very inaccurate, however, the monocular-only model predict depths fairly accurately (column 4). The combined model uses both sets of cues to produce a better depthmap (column 5). In row 3, stereo cues give a better estimate than monocular ones, and again we see that using our combined MRF model, which uses both monocular and stereo cues, results in an accurate depthmap (column 5), correcting some mistakes of stereo, such as some far-away regions which stereo predicted as close.

We note that monocular cues rely on prior knowledge learned from the training set about the environment. This is because monocular 3-d reconstruction is an inherently ambiguous problem. Thus, the monocular cues may not generalize well to images very different from ones in the training set, such as underwater images or aerial photos. In contrast, the stereopsis cues we used are are purely geometric, and therefore should work well even on images taken from very different environments. To test the generalization capability of the algorithm, we also tested the algorithm on images (e.g. containing trees, buildings, roads, etc.) downloaded from the Internet (images for which camera parameters are not known). The model (using monocular cues only) was able to produce reasonable depthmaps on most of the images. (However, not having ground-truth depthmaps and stereo images for images downloaded from the Internet, we are unable to give quanti-

[9] Errors are on a $\log_{10}$ scale. Thus, an error of ε means a multiplicative error of 10^{ε} in actual depth. E.g., $10^{.074} = 1.186$, which thus represents an 18.6% multiplicative error.

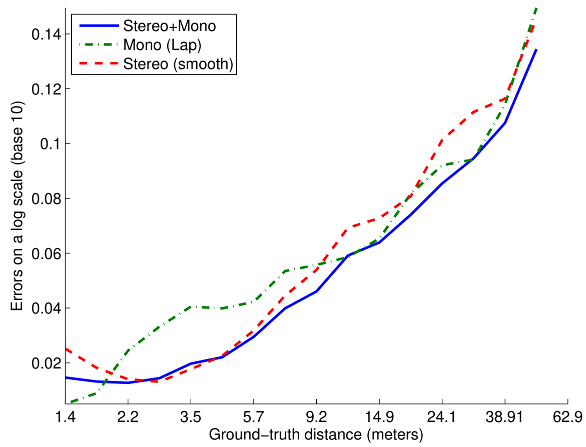

Figure 7: The average errors (on a log scale, base 10) as a function of the distance from the camera.

tative comparisons for these images.[10])

In Fig. 7, we study the behavior of the algorithm as a function of the 3-d distance from the camera. At small distances, the algorithm relies more on stereo cues, which are more accurate than the monocular cues in this regime. However, at larger distances, the performance of stereo degrades; and the algorithm relies more on monocular cues. Since, our algorithm models uncertainties in both stereo and monocular cues, it is able to combine stereo and monocular cues effectively.

We also carried out an error analysis, to identify the cases when the algorithm makes mistakes. Some of its errors can be attributed to limitations of the training set. For example, the maximum value of the depths in the training and test set is 81m; therefore, far-away objects are all mapped to the one distance of 81m. The monocular algorithm fails sometimes to predict correct depths for objects which are only partially visible in the image (e.g., Fig. 5, row 2: tree on the left). For depth at such a point, most of its neighbors lie outside the image area, hence the relations between neighboring depths are not effective. However, in those cases, stereo cues often help produce the correct depthmap (row 2, column 5).

6 Conclusions

We have presented a hierarchical, multi-scale MRF learning model for capturing monocular cues and incorporating them into a stereo system so as to obtain significantly improved depth estimates. The monocular cues and (purely geometric) stereo cues give largely orthogonal, and therefore complementary, types of information about depth. We show that by using both monocular and stereo cues, we obtain significantly more accurate depth estimates than is possible using either monocular or stereo cues alone. This holds true for a large variety of environments, including both indoor environments and unstructured outdoor environments containing trees/forests, buildings, etc. Our approach is general, and applies to incorporating monocular cues together with any off-the-shelf stereo system.

[10]Results on internet images are available at: `http://ai.stanford.edu/~asaxena/learningdepth`

Acknowledgments

We thank Andrew Lookingbill for help in collecting the stereo pairs. We also thank Larry Jackel and Pieter Abbeel for helpful discussions. This work was supported by the DARPA LAGR program under contract number FA8650-04-C-7134.

References

[Blthoff *et al.*, 1998] I. Blthoff, H. Blthoff, and P. Sinha. Top-down influences on stereoscopic depth-perception. *Nature Neuroscience*, 1:254–257, 1998. 1, 2

[Das and Ahuja, 1995] S. Das and N. Ahuja. Performance analysis of stereo, vergence, and focus as depth cues for active vision. *IEEE Trans PAMI*, 17:1213–1219, 1995. 4

[Delage *et al.*, 2006] Erick Delage, Honglak Lee, and Andrew Y. Ng. A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image. In *CVPR*. 2006. 2

[Forsyth and Ponce, 2003] David A. Forsyth and Jean Ponce. *Computer Vision : A Modern Approach*. Prentice Hall, 2003. 1, 3

[He *et al.*, 2004] X. He, R. Zemel, and M. Perpinan. Multiscale conditional random fields for image labeling. In *CVPR*, 2004. 2

[Hoiem *et al.*, 2005] D. Hoiem, A.A. Efros, and M. Herbert. Geometric context from a single image. In *ICCV*, 2005. 2

[Klarquist *et al.*, 1995] W.N. Klarquist, W.S. Geisler, and A.C. Bovik. Maximum-likelihood depth-from-defocus for active vision. In *IEEE Int'l Conf on Intell Robots and Systems*, 1995. 1

[Kumar and Hebert, 2003] S. Kumar and M. Hebert. Discriminative fields for modeling spatial dependencies in natural images. In *NIPS 16*, 2003. 2

[Lafferty *et al.*, 2001] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001. 2

[Loomis, 2001] J.M. Loomis. Looking down is looking up. *Nature News and Views*, 414:155–156, 2001. 2

[Michels *et al.*, 2005] Jeff Michels, Ashutosh Saxena, and Andrew Y. Ng. High speed obstacle avoidance using monocular vision and reinforcement learning. In *ICML*, 2005. 2

[Saxena *et al.*, 2006a] Ashutosh Saxena, Sung H. Chung, and Andrew Y. Ng. Learning depth from single monocular images. In *NIPS 18*, 2006. 1, 2, 4

[Saxena *et al.*, 2006b] Ashutosh Saxena, Justin Driemeyer, Justin Kearns, Chioma Osondu, and Andrew Y. Ng. Learning to grasp novel objects using vision. In *10th International Symposium on Experimental Robotics, ISER*, 2006. 2

[Saxena *et al.*, 2007] Ashutosh Saxena, Justin Driemeyer, Justin Kearns, and Andrew Y. Ng. Robotic grasping of novel objects. To appear in *NIPS*, 2007. 4

[Scharstein and Szeliski, 2002] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47:7–42, 2002. 1, 4

[Szelinski, 1990] R. Szelinski. Bayesian modeling of uncertainty in low-level vision. In *ICCV*, 1990. 4

[Tappen and Freeman, 2003] M.F. Tappen and M.T. Freeman. Comparison of graph cuts with belief propagation for stereo, using identical mrf parameters. In *ICCV*, 2003. 2

[Zhang *et al.*, 1999] Ruo Zhang, Ping-Sing Tsai, James Edwin Cryer, and Mubarak Shah. Shape from shading: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):690–706, 1999. 2