2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Towards detection of orthogonal planes in monocular images of indoor environments

Branislav Mičušík[1], Horst Wildenauer and Markus Vincze[2]

***Abstract*— In this paper, we describe the components of a novel algorithm for the extraction of dominant orthogonal planar structures from monocular images taken in indoor environments. The basic building block of our approach is the use of vanishing points and vanishing lines imposed by the frequently observed dominance of three mutually orthogonal vanishing directions in man-made world. Vanishing points are found by an improved approach, taking no assumptions on known internal or external camera parameters. The problem of detecting planar patches is attacked using a probabilistic framework, searching for the maximum a posteriori probability (MAP) in a Markov Random Field (MRF). For this, we propose a novel formulation fusing geometric information obtained from vanishing points and features, such as rectangles and partial rectangles, together with a color-homogeneity criteria imposed by an image over-segmentation.**

The method was evaluated on a set of images exhibiting largely varying characteristics concerning image quality and scene complexity. Experiments show that the method, despite the variations, works in a stable manner and that its performance compares favorably to the state-of-the-art.

I. Introduction

In the last years the interest in designing mobile robots for domestic tasks has been rapidly growing within the robotics community. Besides being an important field of its own right, building scalable and affordable platforms in response to the diverse application scenarios targeted at by industry represents a tempting goal for robotics research.

In this context, solutions solely based on visual sensory input are moving still more into the center of interest. On one side there is the economical factor pushing down prices of robots by avoiding expensive sensors, on the other hand, images or video acquired by cameras already contain rich information to harvest for tasks such as scene understanding, localization, and navigation. Consequently, during the last years the work on vision-based systems has emerged as a very challenging area from practical and scientific point of view. There is an enormous effort, partially propelled by the cognitive vision research field, to perceive and understand a scene just from visual information.

In this paper, we describe a novel approach devised to help a robot to understand the content of a scene, given a single image. To be more specific, we propose a method for decomposing a single monocular image, possibly stemming from a non-calibrated camera, into orthogonal planes, see Fig. 1. Finding these planes in the image can significantly aid a robot in self localization, navigation and further recognition of objects or landmarks dominating indoor environments, such as windows, doors, tables, chairs, etc. A priori, we design a method for non-calibrated acquisition settings to be able to also handle cases for which either the internal camera parameters are unknown, or are likely to be imprecise. In experiments it is shown that the method is able to extract a significant amount of structural information from a single monocular image. However, a later merging of entire image sequences will greatly contribute to a stabilization of the whole process.

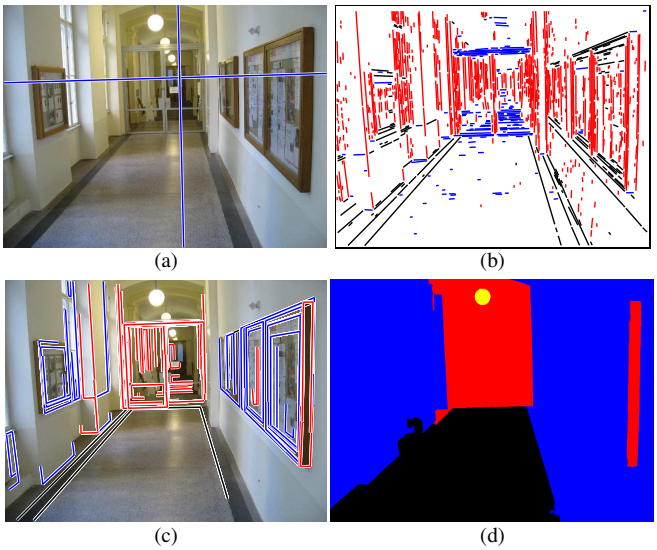

Fig. 1. Proposed sequential chain leading to detection of orthogonal planes in a monocular image. (a) The input image (844×1126 pixels) with vanishing lines depicted. (b) Detected lines consistent with three automatically estimated orthogonal vanishing points. (c) Detected partial and complete quadrilaterals utilizing the vanishing points and lines pointing to them. (d) Final segmentation of planes based on a Markov Random Field formulation employing vanishing points, lines, and quadrilateral segments.

The general concept of the proposed chain is related to previous approaches [1], [2], [3], [4]. However, we formulate the problem in a probabilistic graph-based framework allowing to solve it on a more global level than before. The paper is in its spirit and goals most similar to the recent state-of-the-art work of Hoiem et.al. [4]. They use learnt appearance models based on various geometric, color, and texture cues to partition an image into coarse 3D surface entities. We show that even without learning and by applying less cues we can still compete with their method.

[1]B. Mičušík is with the Computer Science Department, George Mason University, USA, bmicusik@gmu.edu. Most of the research was carried out during his stay at the institution of other two authors.

[2]H. Wildenauer and M. Vincze are with the Automation and Control Institute, Faculty of Electrical Engineering and Information Technology, Vienna University of Technology, Austria, {wildenauer, vincze}@acin.tuwien.ac.at.

The novelty of the paper is two-fold. First, an adopted RANSAC-based line clustering stage for detecting vanishing points and lines consistent with them is described, improving in stability over previous techniques. Second, we formulate the problem of detecting planes in a monocular image using the estimated vanishing points in a probabilistic framework based on searching for maximum a posteriori probability (MAP) of a Markov Random Field (MRF).

In our approach we partially exploit the so-called *Manhattan world* assumption. I.e., the frequently observed dominance of three mutually orthogonal vanishing directions in man-made environments [5]. Motivated by ideas presented in [6], we adopted a RANSAC-based line clustering technique which is able to find dominant vanishing directions in a robust manner. The suggested method takes constraints imposed by a calibrated camera into account; however internal camera parameters do not have to be known a priori as they are estimated during the clustering process. The vanishing point estimation is followed by a search for perspectively distorted rectangles - basic landmarks in man-made environments that are helping further to set the priors for our MRF-based plane detection method. We propose how the estimated vanishing points should be utilized for a suitable setting of weights for edges and vertices of the graph representing the MRF we are operating on.

The method is intent to be applied on mobile platforms where real-time, or at least close to real-time performance, is required. The proposed steps are designed with respect to that, so they can be efficiently coded to fulfill such a requirement.

The structure of the paper is as following. First, the estimation of vanishing points and lines pointing to them is explained in Sec. II followed by Sec. III with a short description of the detection of quadrilateral structures. An Explanation of our MRF-based approach for final localization of planes in an image is given in Sec. IV. We summarize the entire algorithm in Sec. V and report experimental results in Sec. VI.

II. Vanishing point detection

Man-made environments generally exhibit strong regularity in structure and often many parallel lines are present. In such settings, vanishing points provide useful visual cues for deducing information about the 3D structure of an imaged scene. Furthermore, if two or more vanishing points are found of which the underlying structure's orientations are assumed to be orthogonal, then, taking mild assumptions, internal camera parameters can be estimated.

In the following sections, a brief outline of the involved processing stages and the line error model in use is given.

A. Line detection

Initially, connected edge segments are found utilizing a Canny-edge detector with subpixel accurate non-maxima suppression and adaptive hysteresis-thresholding. Following directional edge linking, line candidates are extracted using the iterative subdivision scheme from [7]. The resulting line segments are refined by a Total Least Squares fit to the edge segments pixel coordinates and short lines or lines with low fitting quality are rejected [8].

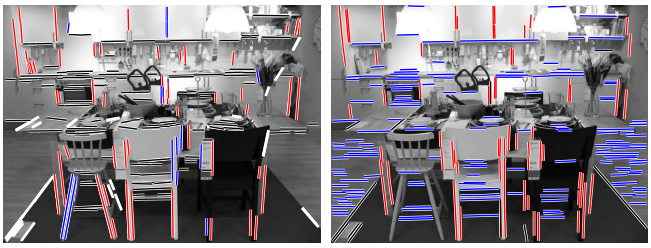

Fig. 2. Comparison of the method [8] and our proposed algorithm on an image of a cluttered scene. Line sets corresponding to each of three detected vanishing points, differentiate by color, are shown. Notice that the orthogonal set of vanishing points, depicted by memberships of lines to them, was estimated incorrectly by the method [8], but correctly by our algorithm. White lines in the left image correspond to noisy lines, not associated with any vanishing point.

For images with low resolution a substantial increase in the number and quality of detected line segments can be achieved by up-sampling the image by factor two prior to edge detection [9].

B. RANSAC-based line clustering

In this stage vanishing point hypotheses are repeatedly generated through the intersection of lines. The intersection points having a large enough set of lines pointing towards them are likely to be true vanishing points and are reconsidered in further processing stages.

1) Line segment error: To quantify the error of a line segment meeting a vanishing point, an ideal line from the segment's midpoint to the vanishing point is constructed and the normal distance of one segment endpoint to this line is measured. Formally, this distance can be written as $d^2(\mathbf{a}_i, \bar{\mathbf{a}}_i)$, where $\mathbf{a}_i$ is the measured line segment endpoint, and $\bar{\mathbf{a}}_i$ is its root point on the ideal line. The described model is based on the assumption that there is little variation in the midpoint of the line segment, as it is the mean of the involved pixel positions. Other error models can be found in [10], [11].

2) Iterative RANSAC: Since the actual mixture fraction of lines belonging to different vanishing points is unknown, we adopt the adaptive variant proposed in [12]. Specifically, we run the algorithm several times over the dataset and successively remove the largest found inlier set from the data before the next trial. After each trial, the vanishing point position is refined by applying Kanatani's *renormalization* scheme [13] to the respective consensus set. We reject newly detected vanishing points if they lie within the uncertainty of previously detected ones utilizing the test statistics proposed in [6]. Here, however, we adopted the vanishing point covariance matrices obtained by *renormalization*. The iteration is stopped, if no more consensus sets with a cardinality above a predefined threshold are found.

3) Candidate selection & camera calibration: Depending on the complexity of the scene the described clustering typically results in numbers of three up to ten vanishing point

candidates. From this set we exhaustively select vanishing point triples and retain only those with approximately orthogonal projective rays. Finally, from the remaining triples the one having the largest total consensus set is chosen as the final estimate of the dominant orthogonal structure.

In the case of unknown internal camera parameters, the camera calibration necessary for the orthogonality test can be carried out individually for each triple of vanishing points. For this we have chosen the composite calibration method described in [13], assuming square pixels and the camera's principle point to be located in the center of the image. Our experiments have shown that a further refinement of its position often caused unstable calibration results, thus we did not consider it further.

C. Comparison to other known methods

In preliminary experiments, we compared our method to implementations of two state-of-the-art methods [8], [14] provided by the authors. We found our algorithm to give qualitatively comparable results to the latter, however usually running five to ten times faster, see Fig. 2 for comparison.

III. Quadrilateral detection

Human made environments contain many rectangular structures. These, depending on occlusions and the camera's field of view, are projected as complete quadrilaterals or incomplete parts (e.g., U- or L-shaped features) thereof. Such features represent strong visual cues for the detection of planar surfaces and consequently are of aid to the task of scene reconstruction and understanding.

In our work we use a perspective rectangle detection method related to the approach of [1], however, applying a probabilistic graph-based method. Line segments compatible with a vanishing line, i.e., the two vanishing points generating it, are grouped by principles of proximity and continuity and a probabilistic inference is used to find hypotheses for quadrilateral-shaped structures in the graph. On of the major advantages of our approach is that it does not only detect perspectively distorted rectangles, but also sub-parts if they are compatible with the initial plane-hypothesis. For an example of the features found, see Fig. 1.

As this method is currently under a reviewing process, further technical details will be omitted here. However, it can be easily replaced with other techniques, such as the one presented in [2], [15].

IV. MRF based plane detection

Having detected vanishing points and lines pointing to them we want to assign to each pixel in an image its 3D orientation w.r.t. to a camera coordinate system. As we assume a Manhattan world structure, this is equivalent to assign one of three labels, where each label corresponds to one of three orthogonal planes.

To solve the problem on a global level, i.e. to allow to take into account prior information about possible pixel orientations and relations between neighboring pixels simultaneously, we formulate the problem in a fully probabilistic framework; as searching for a maximum posterior (MAP) configuration of the Markov Random Field (MRF) [16]. It has been shown [17] that the solution can be found as a Gibbs distribution with maximal probability, i.e., by solving the so called labeling or Max-sum problem of second order - maximizing a sum of bivariate functions of discrete variables.

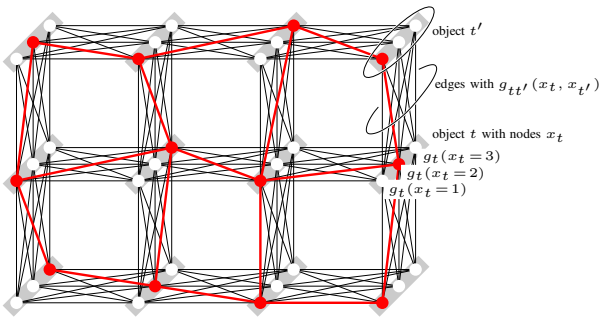

Fig. 3. An example 3×4 grid graph $\mathcal{G}$ for $|\mathcal{X}| = 3$ labels with symbols explained in the text. A labeling $\mathcal{L}$, i.e. solution, from Eq. (2) is shown by a red thick subgraph. Image provided by courtesy of T. Werner [17].

We assume an MRF, i.e., a graph $\mathcal{G} = \langle \mathcal{T}, \mathcal{E} \rangle$, consisting of a discrete set $\mathcal{T}$ of objects (in the literature also called sites, or locations) and a set $\mathcal{E} \subseteq \binom{|\mathcal{T}|}{2}$ of pairs of those objects. Each object $t \in \mathcal{T}$ is assigned a label $x_t \in \mathcal{X}$ where $\mathcal{X}$ is a discrete set. A *labeling* is a mapping that assigns a single label to each object, represented by a $|\mathcal{T}|$-tuple $\mathbf{x} \in \mathcal{X}^{|\mathcal{T}|}$ with components x_t.

An instance of the Max-sum problem is denoted by the triplet $(\mathcal{G}, \mathcal{X}, \mathbf{g})$, where the elements $g_t(x_t)$ and $g_{tt'}(x_t, x_{t'})$ of $\mathbf{g}$ are called *qualities*. The quality of a labeling $\mathbf{x}$ is defined as

$$F(\mathbf{x} \,|\, \mathbf{g}) = \sum_t g_t(x_t) + \sum_{\{t,t'\}} g_{tt'}(x_t, x_{t'}). \qquad (1)$$

Solving the Max-sum problem means finding the set of optimal labellings

$$\mathcal{L}_{\mathcal{G},\mathcal{X}}(\mathbf{g}) = \operatorname*{argmax}_{\mathbf{x} \in \mathcal{X}^{|\mathcal{T}|}} F(\mathbf{x} \,|\, \mathbf{g}). \qquad (2)$$

Fig. 3 depicts the symbols and the problem in a more intuitive way on a simple grid graph. Recently, very efficient algorithms for solving this problem through linear programming relaxation and its Lagrangian dual, originally proposed by Schlesinger in 1976 [18], has been reviewed [19], [17], [20].

A. Graph entities

Generally, the most difficult problem and art connected to MRF based methods is to encode all possible priors about objects being labeled (e.g., orientation, texture, color, shape, appearance) into a graph, i.e., a MRF, while still keeping the problem tractable. The priors we utilized lead to partitioning an image into geometrically and color coherent regions as Fig. 1 shows.

We build a graph on an over-segmented image, i.e., on superpixels, see Fig. 4, to keep the running time in reasonable bounds. The idea is to locally merge pixels with similar color together. The use of superpixels significantly reduces

the number of objects in the graph, still preserving texture information. Simply reducing the image size and building an MRF on pixels to avoid the large complexity as implemented in many approaches leads to losing details and high texture frequencies. In this paper, we use the fast Minimum Spanning Tree based method by Felzenszwalb [21], giving us, by appropriate setting of parameters, 500-800 regions on average. However, any other over-segmentation can be used.

The graph entities are the following. The superpixels represent objects, i.e. the set $\mathcal{T}$, in the graph and edges, i.e. the set $\mathcal{E}$, are established between each two neighboring superpixels. The number of nodes (labels) K is 4, that is, we use one label for each orthogonal plane and one label for "undecided" to allow the solver mark the places where there is not enough information to decide which plane the superpixel belongs to.

Each edge $g_{tt'}(x_t, x_{t'})$ and each object node $g_t(x_t)$ is set accordingly to the smoothness and data term respectively, described in the following sections. After building and setting the graph, the Max-sum solver [17] is run to obtain a particular label x_t for each superpixel t.

B. Smoothness term

The smoothness term $g_{tt'}(x_t, x_{t'})$ controls the mutual bond of neighboring superpixels. In our case we take into account the color difference between superpixels and the straightness of the common boundary. This can be written as follows

$$g_{tt'}(x_t, x_{t'}) = \exp\left(\alpha \|\mathbf{u}_t - \mathbf{u}_{t'}\|^2\right) - \beta\, S^{\text{st}}_{tt'}, \qquad (3)$$

where $\mathbf{u}_t$ is a 3-element color vector of the t-th superpixel (mean color of all pixels belonging to that superpixel) and $\alpha < 0$ is a parameter pre-set to -10. We represent $\mathbf{u}_t$ in the Lab color space because of the perceptual non-uniformity of the standard RGB space. $S^{\text{st}}_{tt'} = \frac{\sum_i^N \text{length\_line}_i}{\text{length\_boundary}}$ is a sum of lengths of N lines fitted to the shared boundary between two superpixels t and t' (longer than 20 pixels), see Sec. II-A, normalized by the length of the boundary. The parameter β controlling the influence of the smoothness term, was set to 0.5 in our experiments.

The proposed smoothness term in Eq. (3) tends to merge superpixels with similar color and jagged boundaries. Such jagged boundaries are usually produced accidentally due to weak gradients [21] and therefore do not correspond to real splits of two superpixel patches in the scene.

C. Data term

The data term $g_t(x_t)$ encodes the quality of assigning a label x from the set $\mathcal{X}$ to an object/superpixel t in the graph. The quality measures how the superpixel itself suits to particular class models, in our case, to lie on one of the orthogonal planes.

For each superpixel 4 numbers are needed to be set, i.e., how likely is that the superpixel is marked by one of four labels. The first three labels stand for the belief that a superpixel lies on one of the three orthogonal planes; the forth label encodes the level of "undecidedness".

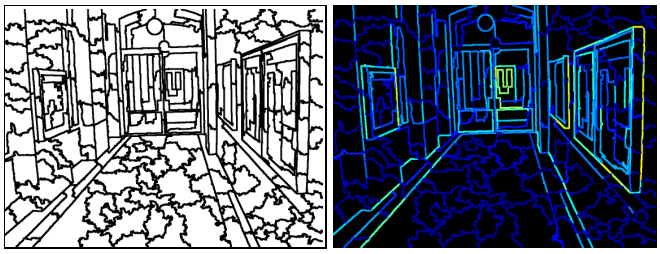

Fig. 4. Left: Superpixels detected in the image from Fig. 1. Each region corresponds to one object in the constructed graph. Right: The smoothness term. Boundary-color encodes the penalty set in the graph between the objects corresponding to two neighboring superpixels. Darker coloring denotes less penalization. Note, that straight boundary segments are penalized stronger.

The consistency of a superpixel to a plane is expressed via a deviation of gradient orientations of the pixels along the boundary of the superpixel to two vanishing points corresponding to that plane. For computation of the gradient orientations we use the 5-component gradient mixture model described in [5]. For each image pixel, the model gives the probability of the pixel lying on an edge, the membership to one of the three vanishing points, and the probability of being noise. We take into account only those pixels having a probability of being on an edge above a certain threshold. Then, a normalized histogram $h_t(y)$ with four bins $y = \{1, 2, 3, 4\}$ is computed from vanishing point memberships of all pixels lying along the t-th superpixel boundary. The fourth bin accumulates points classified to be on an edge, however, not consistent with any vanishing point direction.

Finally, the consistency of the superpixel with each label is set as

$$g_t(x) = \begin{cases} \sum_{\substack{i=1 \\ i \neq x}}^{3} h_t(i) & \text{if } x = \{1, 2, 3\}, \\ h_t(x) & \text{if } x = 4. \end{cases} \qquad (4)$$

In the data term, two additional priors are utilized. One stemming from the position of ideal lines and one from detected quadrilateral segments. The ideal line is defined as a line passing through two vanishing points and is a projection of an intersection of a 3D plane with a plane at infinity [12]. It gives us the constraint that a superpixel detected in the image cannot cross the ideal line of the plane it belongs to. The data term of such superpixels is set to zero to decrease the belief of them to lie on a particular plane. Fig. 1 shows two ideal lines where one corresponds to a ground plane. Notice that this line, called a horizon, is completely above the ground plane and therefore superpixels on that plane are not allowed cross the horizon.

The second prior comes from the fact that all superpixels behind detected quadrilateral segments, see Sec. III, have to lie on the plane where the segments are detected. The data term of such superpixels is increased or set to a high value in order to strengthen the belief of them to lie on that particular plane.

V. Algorithm

We shortly summarize the main steps leading to the final detection of orthogonal planes in a monocular image. The

algorithm consists of sequential steps for the detection of

1) lines and vanishing points coming from their intersections as the largest total consensus sets corresponding to orthogonal directions.
2) quadrilaterals or their parts corresponding to rectangles or their parts in a scene.
3) orthogonal planes in a scene based on an MRF framework formulated on over-segmented image; utilizing vanishing points, ideal lines and quadrilaterals.

VI. Results

We evaluate the proposed method on large variety of indoor images downloaded from the Internet. Some of the most representative are shown in Fig. 5. The images are approximately 1 Mpixel large and their quality varies since they were taken by different, to us unknown, cameras under different illumination conditions. The results show feasible and stable performance, although light reflections, shadows, jpg-artifacts, and occlusions, are present in the images.

Fig. 5 shows each image segmented into 4 labels, three for each orthogonal plane and one for "undecided". We compare our method to the state-of-the-art method [4] aiming at exactly the same goal, i.e. at recovering surface layout from a single image. To produce the results of [4] the publicly available code[1] was used in combination with a provided indoor classifier. The presented results show comparable performance of our method and often achieving better results. Moreover, the run-time of our method was shorter, 1 min on average, while the method of [4] took 3 min using the same Pentium 4@2.8 GHz.

The proposed method is currently mostly implemented in unoptimized MATLAB and many of the routines and functions can be re-implemented in a much more efficient way in C/C++. For finding the MAP of the MRF we use a publicly available[2] C++ implementation of the Max-sum solver [17].

It can be seen in Fig. 5 that at some places, especially at connections of planes, our result is not always correct. This is caused by either superpixel missing the true boundary and thus overlapping two planes. Or, there is an occlusion present, i.e., one plane partially occludes the other. In the second case, the incorrect behavior comes from the data term formulation, Eq. (4), as the superpixel is expected to contain two strong gradient directions only. In the case of the occlusion, e.g. a table leg touching a floor, the superpixel covering a part of the floor and touching the leg contains pixels at its boundary which are pointing to a vertical vanishing point. This may cause that the superpixel is incorrectly assigned to one of the vertical planes. The resulting inconsistency, depending on neighboring superpixels, cannot always be solved by the smoothness term.

VII. Conclusions

We have presented a novel algorithm for the extraction of dominant orthogonal planar structures from monocular images, taken in indoor environments. We have shown that even without learning by using basic geometric cues we can still compete with the state-of-the-art-method aiming at the same goal. Although the algorithm is a priori designed to handle occlusion-free environments, it still provides reasonable results in cluttered scenes. The presented framework is intend on being a part of a robot's "decision unit" for understanding a surrounding scene and to further support navigation.

[1] http://www.cs.cmu.edu/~dhoiem/projects/software.html

[2] http://cmp.felk.cvut.cz/cmp/software/maxsum/

VIII. Acknowledgments

The research leading to these results has received funding from the European Community's Sixth Framework Programme (FP6/2003-2006) under grant agreement No. FP6-2006-IST-6-045350 (robots@home) and US National Science Foundation Grant No. IIS-0347774.

References

[1] J. Košecká and W. Zhang, "Extraction, matching and pose recovery based on dominant rectangular structures," *Computer Vision and Image Understanding (CVIU)*, vol. 100, no. 3, pp. 174–293, 2005.

[2] J. Lim, C. McCarthy, D. Shaw, L. Cole, and N. Barnes, "Insect inspired robots," in *Proc., Australasian Conference on Robotics and Automation (ACRA)*, 2006.

[3] M. Rous, H. Lupschen, and K. Kraiss, "Vision-based indoor scene analysis for natural landmark detection," in *Proc., Intl. Conference on Robotics and Automation (ICRA)*, 2005, pp. 4642–4647.

[4] D. Hoiem, A. Efros, and M. Hebert, "Recovering surface layout from an image," *International Journal of Computer Vision (IJCV)*, vol. 75, no. 1, Oct. 2007.

[5] J. M. Coughlan and A. L. Yuille, "Manhattan world: Orientation and outlier detection by bayesian inference." *Neural Computation*, vol. 15, no. 5, pp. 1063–1088, 2003.

[6] R. Pflugfelder and H. Bischof, "Online auto-calibration in man-made worlds," in *Proc., Digital Image Computing: Techniques and Applications*, 2005, pp. 519–526.

[7] P. Kovesi, "Matlab and octave functions for cumputer vision and image processing," School of Computer Science & Software Engineering, University of Western Australia.

[8] J. Košecká and W. Zhang, "Efficient computation of vanishing points," in *Proc., Intl. Conference on Robotics and Automation (ICRA)*, 2002, pp. 223–228.

[9] U. Köthe, "Edge and junction detection with an improved structure tensor," in *Proc., Symposium of German Association for Pattern Recognition (DAGM)*, 2003, pp. 25–32.

[10] C. Rother, "A new approach to vanishing point detection in architectural environments," *Image Vision Computing (IVC)*, vol. 20, no. 9-10, pp. 647–655, 2002.

[11] D. Liebowitz, "Camera calibration and reconstruction of geometry from images," Ph.D. dissertation, University of Oxford, 2001.

[12] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[13] K. Kanatani and Y. Sugaya, "Statistical optimization for 3-d reconstruction from a single view," *IEICE Transactions on Information and Systems*, vol. E88-D, no. 10, pp. 2260–2268, 2005.

[14] H. Wildenauer and M. Vincze, "Vanishing point detection in complex man-made worlds," in *Proc., Intl. Conference on Image Analysis and Processing (ICIAP)*, 2007.

[15] J. Hayet, F. Lerasle, and M. Devy, "Visual landmarks detection and recognition for mobile robot navigation," *Image and Vision Computing*, vol. 25, no. 8, pp. 1341–1351, August 2007.

[16] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Constructing free-energy approximations and generalized belief propagation algorithms," *IEEE Trans. on Information Theory*, vol. 51, no. 7, pp. 2282–2312, 2005.

[17] T. Werner, "A linear programming approach to Max-sum problem: A review," *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, vol. 29, no. 7, pp. 1165–1179, 2007.

[18] M. Schlesinger, "Syntactic analysis of two-dimensional visual signals in noisy conditions," *Kibernetika*, no. 4, pp. 113–130, 1976, in Russian.

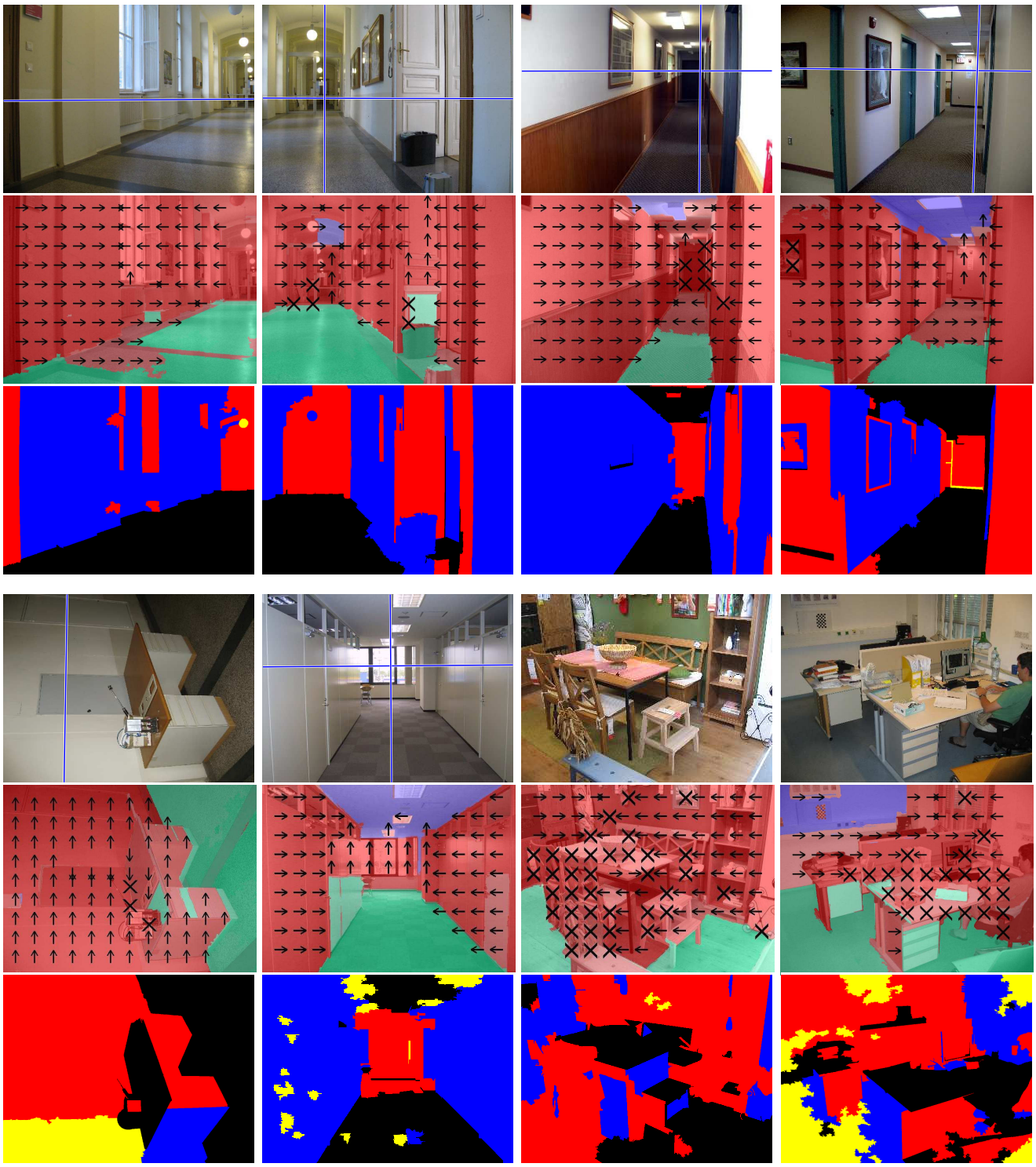

Fig. 5. Results of planes-detection in single indoor images arranged in triplets. Top: Input image with in-plotted ideal lines estimated by our method. Middle: The method by Hoiem et.al. [4] segmenting images into ground plane and vertical planes. Arrows stand for plane orientations to the left/up/right, markers 'o' and 'x' for porous and solid materials, respectively. Bottom: Our proposed method segmenting images into three orthogonal planes. Each plane is depicted by a different color, where yellow color stands for "undecided" pixels.

[19] M. Schlesinger and B. Flach, "Analysis of optimal labelling problems and application to image segmentation and binocular stereovision." in *Proc., East-West-Vision Workshop*, 2002, pp. 65–81.

[20] V. Kolmogorov, "Convergent tree-reweighted message passing for energy minimization," *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, vol. 28, no. 10, pp. 1568–1583, 2006.

[21] P. Felzenszwalb and D. Huttenlocher, "Efficient graph-based image segmentation," *Int. Journal of Computer Vision (IJCV)*, vol. 59, no. 2, pp. 167–181, 2004.