Monocular Depth Ordering Using T-Junctions and Convexity Occlusion Cues

Guillem Palou, *Student Member, IEEE*, and Philippe Salembier, *Fellow, IEEE*

***Abstract*—This paper proposes a system that relates objects in an image using occlusion cues and arranges them according to depth. The system does not rely on *a priori* knowledge of the scene structure and focuses on detecting special points, such as T-junctions and highly convex contours, to infer the depth relationships between objects in the scene. The system makes extensive use of the binary partition tree as hierarchical region-based image representation jointly with a new approach for candidate T-junction estimation. Since some regions may not involve T-junctions, occlusion is also detected by examining convex shapes on region boundaries. Combining T-junctions and convexity leads to a system which only relies on low level depth cues and does not rely on semantic information. However, it shows a similar or better performance with the state-of-the-art while not assuming any type of scene.**

As an extension of the automatic depth ordering system, a semi-automatic approach is also proposed. If the user provides the depth order for a subset of regions in the image, the system is able to easily integrate this user information to the final depth order for the complete image. For some applications, user interaction can naturally be integrated, improving the quality of the automatically generated depth map.

***Index Terms*—Binary partition tree (BPT), convexity, monocular depth, occlusion cues, T-junction estimation.**

I. Introduction

HUMANS are known for their ability to recognize objects and determine the scene structure in many distinct situations. Our capacity to retrieve a coherent depth interpretation of the environment seems to be robust and reliable in the majority of cases, with the exception of some optical illusions. The ability to perceive a 3-D world in humans is mainly due to binocular vision, where each eye provides a different image of the scene and disparity is subconsciously inferred. However, in monocular situations, perception is affected but still, depth information can be perceived. The scientific community has tried to mimic the human behavior to determine the depth structure of scenes. To this day, human performance is still much better than computer based approaches in both time and accuracy, but the evolution of 3-D visualization hardware encourages researchers devote efforts to estimate depth from visual content.

With the decrease of stereo camera costs, depth estimation in stereo/multiview systems is gaining more importance. Most of the state of the art systems on depth estimation take profit of multiple points of view to infer disparity. However, most of the acquired content has only one point of view. The huge amount of photos or movies obtained with conventional cameras makes monocular depth estimation an attractive research area and an rather pressing need for the 3-D media industry. As 2-D to 3-D conversion is a relatively new field, many systems still rely on semi-supervised approaches to correct estimation errors. For example, converting monocular content to 3-D to some extend has been an objective for many industrial actors such as Microsoft [1], Disney [2] or Prime Focus (a post-production company for Hollywood Studios) with View-D software [3]. Monocular depth systems are not able to estimate a perfect depth map, but, in practice, a rough representation may suffice for humans to perceive a 3-D effect [4]. Additionally, monocular depth estimation can be used as an input to other systems such as object editing by depth (foreground/background removal, or example) or as a a rough depth estimation for a full 3-D system.

Although current commercial products (such as the ones previously mentioned) heavily rely on human interaction to derive a correct depth interpretation, there is a need for an automated system to reduce both time and costs. To this end, many research institutions [5], [6] have proposed several monocular depth estimation systems. These systems base their reasoning on finding monocular depth cues in images. Although these cues are easily identified by humans, they are a detection challenge for state of the art computer algorithms. Since they are an important part of the proposed algorithm, Section I-A is devoted to describe their role on human depth perception. Section I-B discusses the current literature on monocular depth estimation, followed by a brief description of the structure and innovations of our system.

A. Monocular Low Level Depth Cues

Perceptual organization is a widely known area of study in the Gestalt Psychology with its Laws of Organization. According to this field, the set of cues that humans use to infer depth in a single image [7], are, among others: brightness, shading, blurring, occlusion, convexity, vanishing points, texture gradient and familiar size. Even though humans make extensive use of stereo vision to perceive *absolute* depth, when only one point of view is available, we are also capable to infer certain depth relationships. Many monocular cues also rely on

Manuscript received January 2, 2012; revised December 17, 2012; accepted January 5, 2013. Date of publication January 14, 2013; date of current version March 14, 2013. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Kenneth Lam.

G. Palou and P. Salembier are with the Signal Theory and Communications Department at the Technical University of Catalunya, Barcelona 08034, Spain (e-mail: guillem.palou@upc.edu; philippe.salembier@upc.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2013.2240002

1057-7149/$31.00 © 2013 IEEE

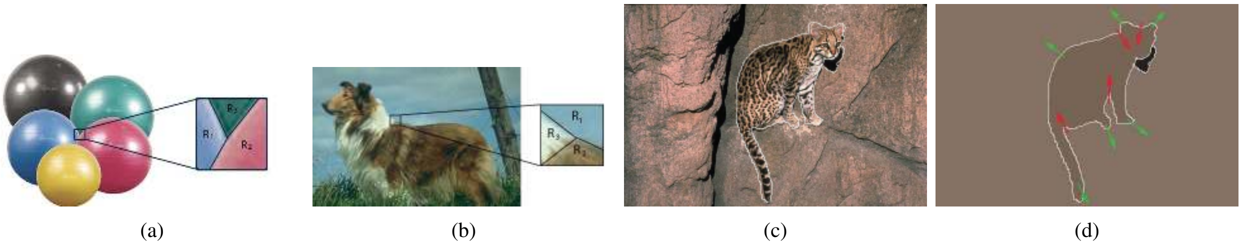


Fig. 1. (a) T-junction example. Locally, region R_2 is the one forming the largest angle, appearing to be over R_1 and R_3. (b) Inverted T-junction example. Locally, region R_1 is the one forming the largest angle, but corresponds to the sky region, which is behind the dog. (c) and (d) Points of high convexity are cues to determine the relative depth. Local cues at each boundary point should be averaged to decide the correct sign of convexity.

the overall scene knowledge and previously known situations such as the sun being on top or people standing upright on the ground. A priori knowledge of the scene structure, like the approximate size of a person or the shape of a tree may also help to infer depth in natural scenes. Other cues, such as occlusion, are local and related to specific image points. At these points, the image structure may offer good signs of depth discontinuity. Produced by the projection of the 3-D scene into the image plane, occlusion is specially observed in two cases: T-junctions and convexity.

T-junction points appear when an object is in front of other two objects, see Fig. 1. T-junctions are formed by three regions and, locally, one of them is forming an almost flat angle (i.e. 180 degrees). The other two regions may form two arbitrary angles, but if any falls below 30-40 degrees, the perception of occlusion falls rapidly [8]. Although T-junctions are clear signs of occlusion, the relative depth order of the intervening regions cannot be determined exclusively by examining the local angle configuration. In *normal* T-junctions, the region forming the largest angle is likely to be the occluding region (and thus closer to the viewer). However, in the case of *inverted* T-junctions, the same region can correspond to the background (and thus occluded and further away to the viewer). According to [8], junction detection is difficult even for humans but, once detected, the occluding side is easily identified, see Fig. 1(b). Computers, however have much more difficulties in determining the depth order and a global reasoning on other T-junctions is needed.

The second case of occlusion is produced when a single object is lying in front of other regions. In small neighborhoods of the object boundaries, convex shapes appear to be in front of their background, whether or not other cues are present. When humans deal with natural shapes, local decisions at points of object boundaries are averaged along the entire object contour to arrive at a global interpretation. For example, in Fig. 1(c), a feline standing in front of a wall is shown. If convexity is interpreted along the boundaries as shown in Fig. 1(d), there may be parts of the contour indicating one depth order (green arrows) and other parts with opposite sign (red arrows). In such cases, humans partially use convexity cues to decide that the feline is in front of the wall. As a priori information, recognizing the different parts of the image (the feline and the wall) immediately restricts the scene structure. Humans know that a feline cannot be visible and behind a wall at the same time. Nevertheless, occlusion cues help to enforce depth relationships even in known situations.

Humans not only make use of local cues to derive the depth structure of an image, but other global reasonings take place. Therefore, it is unlikely that a system for depth perception defined only on T-junction and convexity detection can compete with human vision. However, in this paper, we are interested in studying the performances and limitations of such a system. In cases where the system cannot achieve the correct depth interpretation, user interaction can be used. Since humans can easily identify depth planes in an image, the proposed system incorporates the possibility to accept depth information on a limited set of regions provided with markers defined by the user. Markers are widely used in image processing: image segmentation [9], reconstruction [10] or 2-D to 3-D reconstruction [4]. In this work, markers set the depth relations on image regions.

B. Related Work

Monocular depth perception is a fairly new field of study in computer vision. One of the first works trying to recover the image structure was presented in [11], but reference points were needed to reconstruct lines and planes. In [12], instead of the overall depth organization, a computation of the scale of the image (i.e. mean depth) was proposed. Focusing on algorithms that recover absolute/relative depth of regions in the image, two main approaches are found. The ones that use high level information and the ones that operate over the image structure finding special points indicating some depth cues. In the former class, [5] and [6] oversegment the image and gather for each region color, texture, vertical and horizontal features to use them in a conditional random field, trained a priori with a ground truth data set, for absolute depth estimation. The main drawback of high-level information approaches is that they are limited to the kind of images they have been trained for. The latter type of systems, where [13] can be included, use focus on the detection of relative depth cues such as occlusion to order the objects in the scene. Occlusion does not permit to infer absolute depth as high-level information may offer, but is more generic as it does not assume anything about the type of scene.

Depth ordering the regions of an image permits to determine immediately the occlusion boundaries. An occlusion boundary is defined as the border between two different depth planes. Moreover, the nearest side of the boundary is defined as the owner (or figure, foreground) of the boundary. Similarly, its further side is called the background or simply, ground, of the boundary. The problem to detect on occlusion boundaries which side is figure and which side is ground has been addressed by works in [14]– [16]. Similar to depth estimation systems, some of these approaches rely also on low level cues such as shapemes [14], convexity or parallelism [15]. The main drawback of these systems is that they do not provide closed partitions and only single contours are labeled.

Following the computational vision model of [17], [18], our system tries to integrate the estimation of depth cues and the segmentation process. In this work, the segmentation is understood as a two step process: A construction of a region-based, hierarchical representation of the image, and a selection of the regions in this hierarchy to compose the final depth ordered partition. Inside this framework, T-junction and convexity cues are estimated iteratively during the first step of the system. The second stage proposes an optimization scheme to produce a depth ordered partition from the depth relations determined by the previously mentioned cues.

This architecture differs from the one of [13], which consists of an estimation of T-junction points, followed by image segmentation, convexity detection and a final depth ordering stage. Here we integrate the estimation of low level depth cues into the construction of the BPT. As the framework is region-based, our aim is to increase the performance and robustness of the cue detection compared to [13] where the T-junction detection is performed using a modification of the [19] pixel-based detector. A part from the architecture, another fundamental difference with [13] is that the T-junction model is extended to include both *normal* and *inverted* depth orders. Finally, the last major difference deals with the pruning of the BPT, formulated here as an optimization problem on the tree.

Although the system can provide an automatic depth ordered segmentation of the image, an optional strategy involving user interaction is also discussed. The purpose of incorporating human interaction is to help the system to decide in challenging situations, where the assumptions related to low level depth cues are not fulfilled. To interact with the system, a few depth markers can be provided as an additional input. If that is the case, the depth relations introduced by the user are naturally combined with the detected depth cues in the second stage of the system. This form of interaction is suitable owing to the fact that the first step of the system is computationally more costly. As a result, with little computation overload, the user is able to easily refine the markings in case the algorithm does not provide a sufficiently accurate solution. The work in [4] also proposes algorithms for semi-automatic 2-D to 3-D reconstruction, for videos and single images. These systems offer absolute depth maps (up to a scale), while our system outputs relative depth orders.

Two major conclusions can be drawn from this work: First, it is shown that using only low-level (and very local) cues a global depth ordering of the image can be obtained. Second, it is shown that even if the algorithm does not rely on a training phase, results are of similar or better quality than approaches of the state of the art [5], [6] that rely on high level specific (even semantic) cues.

The following sections describe the system architecture. First, the system models are exposed in Section II, as well as how the hierarchical image representation is built. Section III is devoted to the estimation of occlusion cues. Section IV describes the process of finding a suitable depth ordered partition from the set of estimated cues. Finally, experimental results are presented in Section V for both the automatic and the semi-automatic proposed systems. Comparison with other systems is also performed qualitatively against [5], [6], and [13] and quantitatively with [14], [15] by evaluating the performance on occlusion boundaries.

II. System Models

An important part of the system relies on the Binary Partition Tree (BPT). The BPT is a structured hierarchical representation of the image regions that can be obtained iteratively from an initial partition [20], [21]. At each iteration, pairs of adjacent regions are iteratively merged to form a parent region containing the two merged ones [20]. The pair of regions to be merged are the two most similar according to a similarity measure. In this project, the BPT is used with two objectives:

1) *Region-based representation of the image:* Pixels can be thought as the basic unit of image information. Many times working with pixels is limiting and another image representation is needed. In our case the final objective is to have a depth ordering of objects/regions in the image so, a region representation is needed. Going from pixels to regions is carried out using the BPT algorithm
2) *Solution space:* When the BPT is constructed, the leaves of the tree represent the regions belonging to the initial partition and the root node refers to the entire image support. The remaining tree nodes represent the intermediate regions formed during the merging process. Many partitions can be formed by combining regions represented in this hierarchical structure. This process can be seen as a tree pruning. In summary, the BPT defines a partition solution space.

A. Region Model

To define similarity, region models are needed, along with a distance measure between them. The chosen color space to represent the image is the $CIE\ Lab$ because of the perceptual nature of color difference metrics in this space. Region color distribution is modeled using adaptive three-dimensional histograms (signatures), [22]. Previous region merging algorithms use a simpler region model, considering only the mean color [21] or monodimensional histograms [23]. 3-D-histograms do not loose correlation information between channels. Unfortunately, their representation is very costly in memory usage. To overcome this drawback, adaptive signatures as in [22] are chosen. In practice, 8 dominant colors are a good choice to represent the whole image [24], so the same number is chosen to describe each region, but depending on the region color homogeneity, a lower number may suffice. Each signature $\boldsymbol{s}_i$ is characterized by a set of ordered pairs $\{(p_1, \boldsymbol{c}_1), (p_2, \boldsymbol{c}_2) \dots (p_n, \boldsymbol{c}_n)\}$ with $n \leq 8$. Each pair i is composed of a representative color vector $\boldsymbol{c}_i$ and its probability of appearance p_i. Since in a BPT construction, some regions belong to the initial partition and some others are created by merging, the estimation of these dominant colors is depending on the nature of the region. If the initial partition is formed by individual pixels, the dominant color for each region is simply the pixel color. If a segmentation is available as input, the dominant colors of the regions containing many pixels are estimated using a quantization algorithm as in [25]. In our case, the proposed system has no segmentation input, therefore the initial partition is formed by the pixels. On the other hand, when a region is the result of a merging process, another approach can be followed to reduce the computational burden.

Hierarchical Signature Estimation: To approximate a joint signature $\boldsymbol{s}$ from two signatures $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$, the following algorithm is proposed: When two regions are merged, a new signature $\boldsymbol{s}$ is created for the parent region by joining the two underlying signatures, $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$. While the number of representative colors exceeds the maximum (that is 8, here), the two most similar colors are merged and replaced by their average color, until $\boldsymbol{s}$ contains at most 8 colors.

The distance d_{ij} chosen to measure the difference between two colors i and j of signature $\boldsymbol{s}$ is $d_{ij} = (p_i + p_j)c_{ij}$. Where the c_{ij} term is perceptually defined as in [22] which is based in [26]:

$$c_{ij} = \left(1 - e^{-\frac{\Delta_{ij}}{\gamma}}\right) \tag{1}$$

with Δ_{ij} being the euclidean distance between Lab-colors $\boldsymbol{c}_i$ and $\boldsymbol{c}_j$. The decay parameter γ indicates a soft threshold of distinguishable colors and is set to 14.0 as in [22].

B. Region Similarity Measure

The construction of the BPT is done by merging neighboring regions iteratively. The order in which these regions are merged is defined by a similarity measure. Usually, this measure is based on low-level features of the regions such as color, area, or shape [21]. In this work, however, depth information based on T-junctions is also introduced to contribute to this measure. The formal expression used to measure the similarity between two adjacent regions R_1 and R_2 is:

$$d(R_1, R_2) = d_a\left(\alpha d_c + (1-\alpha)d_s\right)d_d \tag{2}$$

d_a stands for the area distance. d_c and d_s are the color and shape measures respectively. α is the weighting factor between shape and color and its value was experimentally set to $\alpha = 0.7$, giving color much more importance than shape. d_d is the newly introduced depth measure. These four contributions (area,color,shape and depth) are considered to be key characteristics to define regions.

Color has been proven to be the most important feature. In practice, however, objects in the real world have more or less compact and round shapes. The exclusive use of color distance d_c lead to regions with unnatural shapes so a measure evaluating the region shape d_s is introduced. Moreover, relevant objects in a scene present similar areas so a term addressing region size d_a is also included. Since the goal of this work is to estimate depth planes, the inclusion of a depth measure d_d attempts to differentiate different levels of depth already during the BPT construction.

To measure color similarity $d_c(R_1, R_2)$ between signatures, the earth mover's distance (EMD) [27] is chosen:

$$d_c(R_1, R_2) = EMD(\boldsymbol{s}_1, \boldsymbol{s}_2). \tag{3}$$

Although in [22] this distance is used locally to detect corners and junctions, there is no knowledge that it has been used for a complete segmentation process. The EMD distance is defined to be the minimum cost to transport a certain probability masses f_{ij} to transform one signature $\boldsymbol{s}_1$ to another $\boldsymbol{s}_2$, according to some costs between signature colors. Formally, the EMD is defined as:

$$EMD(\boldsymbol{s}_1, \boldsymbol{s}_2) = \min \sum_i \sum_j f_{ij} c_{ij} \tag{4}$$

$$\text{subject to:} \quad f_{ij} \geq 0, \quad \sum_i f_{ij} = p_{2j}, \quad \sum_j f_{ij} = p_{1i}. \tag{5}$$

The costs c_{ij} are defined as the distance (1) between signature colors. The constraints (5) impose that the probability masses f_{ij} should be non-negative and should transform the probabilities of occurrence in $\boldsymbol{s}_1$ $(p_{11}, p_{12}, \ldots, p_{1n})$ to the ones in $\boldsymbol{s}_2$, $(p_{21}, p_{22}, \ldots, p_{2n})$. The minimization of (4) subject to (5) is performed using linear programming [28]. The shape distance is the relative increase of perimeter of the new region with respect to the biggest one [21]:

$$d_s(R_1, R_2) = \max\left(0, \frac{\min(P_1, P_2) - 2P_{1,2}}{\max(P_1, P_2)}\right) \tag{6}$$

where P_1, P_2 and $P_{1,2}$ are the two region perimeters and the common perimeter respectively. d_s is only applied when both region shapes are meaningful (i.e. at least 50 pixels in area).

The area distance is defined as:

$$d_a(R_1, R_2) = \log\left(1 + \min(A_1, A_2)\right) \tag{7}$$

with A_1, A_2 the respective region areas in pixels. There is no general consensus about which area weighting distance should be used during BPT construction. In [21], [23], either no weighting and linear weighting are performed. Generally, area weighting is used to encourage the merging of small and semantically unimportant regions before the large regions merge. In this work, a logarithmic weighting is chosen.

As a final similarity measure, depth information d_d is introduced using T-junction candidate points. The idea is to increase the region distance if two adjacent regions do not belong to the same depth plane, according to a set of T-junctions. To determine the probability that a region R_1 occludes R_2, common T-junction candidate points are examined. Candidates arise where three regions meet, see Fig. 2.

If R_1 and R_2 share a common neighbor R_3, at least a T-junction candidate n is present at the contact point(s) of the three regions. For each candidate n a probability $p_{i,n}$, $i = 1, 2$, of the occluding region to be R_i is computed as described in Section III. Common candidates between R_1 and R_2 determine the probability of occlusion:

$$p_x = \left(1 - \prod_n^{N_x}\left(1 - p_{x,n}\right)\right)\prod_n^{N_y}\left(1 - p_{y,n}\right) \tag{8}$$

where $(x, y) = (1, 2)$ or vice versa. Therefore, p_1, p_2 is the probability of R_1, R_2 being the occluding region respectively.

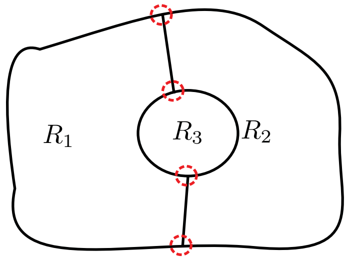


Fig. 2. Common T-junctions between R_1 and R_2. Red circles show the T-junction points that should be evaluated when measuring the similarity between R_1 and R_2.

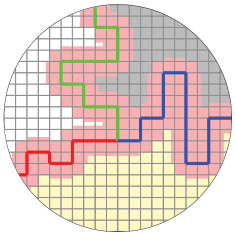


Fig. 3. T-junction boundary presents boundary pixels (pink), which may introduce bias in mean and variance estimation. The three different regions are marked with white, gray, and yellow.

N_1 and N_2 are the number of T-junctions indicating whether R_1 or R_2 is the occluding region. The probability p_1 can be seen in the following way: if R_1 is occluding R_2, all the $p_{2,n}$ should be false and at least one $p_{1,n}$ should be true. A similar reasoning can be applied to p_2. The final depth distance between regions is defined as:

$$d_d(R_1, R_2) = \frac{1}{(1 - |p_1 - p_2|)}. \tag{9}$$

The measure $d_d(R_1, R_2) \approx 1$ either when there are conflicting T-junctions indicating both R_1 and R_2 as occluding regions ($p_1 \approx p_2$), or when the T-junction confidences are low ($p_1, p_2 \approx 0$). $d_d(R_1, R_2) \gg 1$ when the occlusion relation is very likely, that is, either $p_1 \gg p_2$ or vice versa.

Introducing depth information into the region distance allows us to favor the merging of regions belonging to the same depth plane, leaving different depth planes to be merged at the top of the tree. To evaluate (8) and, as a consequence, (9), $p_{1,n}$ and $p_{2,n}$ need to be estimated for each candidate point. This is the subject of the following section.

III. Occlusion Depth Cues Estimation

Two subsystems are designed to detect the two considered depth cues: T-junctions and convexity. For the first class of cues, each point of the image is assigned a confidence value, indicating the probability to be indeed a true T-junction. The confidence computation is performed, as in [29], during the BPT construction so as to introduce depth information into the region similarity measure (2).

The second class of occlusion cues are convexity cues. They relate two adjacent regions by their common boundary shape and gradient intensity. Convexity is only reliable on long contours which only appear at the very top of the BPT structure. To this end, convexity cues are estimated for the final depth ordering but do not affect the region similarity measure.

A. T-Junction Candidates Estimation

Several approaches can be found on the literature about T-junctions estimation but, unlike the proposed system, many of them rely on a hard threshold to detect these points [13], [22], and [30].

In this section,we assume that we are analyzing a candidate local configuration n_o in which R_1 may be on the top of R_2 and R_3; that is, we want to estimate the value of p_{1,n_o} of equation (8). Extension to p_{2,n_o} is straightforward. To simplify the notation, we call p this value of p_{1,n_o}. To estimate the confidence value p of a T-junction, color difference, angle structure and boundary curvature confidence are evaluated at each candidate point within a centered circular window ($R = 10$), except for the angle. Color contributes to differentiate between contrasted regions, angle helps to infer the depth relationship and curvature detects if the junction has clearly defined boundaries. Since they are independent features, $p = p_{\text{color}} p_{\text{angle}} p_{\text{curve}}$.

1) Color: When a T-junction is formed in an image at a location $\boldsymbol{p_t}$, it may have some color characteristics that indicate a discontinuity on depth. The analysis of the color characteristics is limited to a local neighborhood $\Omega(\boldsymbol{p_t})$, see Fig. 3. In this local window, the three regions can be modeled with a three dimensional histogram, similar to the one proposed as region model in Section II-A. Since the analysis is done in a local neighborhood, $n = 3$ representative colors proved to be sufficient. As shown in Fig. 3, the included pixels for color confidence(s) evaluation are the ones which are not neighbors of the other two regions. Due to the blurring of contours, all region boundary pixels are discarded to avoid a bias in the signature calculation.

Define h_i $i = 1, 2, 3$ to be the histogram of region R_i near the T-junction candidate. Since a distance measure can only be applied to a histogram pair at a time, a total of three color distances are computed. λ_{ij}, $i < j$, $i, j = 1, 2, 3$, represents the distance between region R_i and region R_j. Distances are also computed using the EMD, as for the region color similarity used for the BPT construction. Each distance gives a value $0 \leq \lambda_{ij} \leq 1$:

$$\lambda_{ij} = EMD(\boldsymbol{h}_i, \boldsymbol{h}_j). \tag{10}$$

If $\lambda_{ij} \approx 0$, the two regions do not seem different in a local neighborhood. Conversely, if $\lambda_{ij} \approx 1$ a strong contrast is present between R_i and R_j. The color confidence p_{color} for the pixel $\boldsymbol{p}_t$ is obtained by:

$$p_{\text{color}} = \frac{2\lambda_{\min}\lambda_{\max}}{\lambda_{\min} + \lambda_{\max}} \tag{11}$$

with $\lambda_{\max} = \max(\lambda_{12}, \lambda_{13}, \lambda_{23})$. $\lambda_{\min}$ is computed similarly. The measure (11) is motivated by the Harris corner detector [31], [32] and $p_{\text{color}} \approx 1$ only when all $\lambda_{ij} \approx 1$.

2) Angle: The angle is a fundamental local cue to determine the depth order of the three regions meeting at a T-junction, see Fig. 1(a) and (b). Within the BPT construction, the angles of a T-junction point are determined by the region boundaries. Information at the junction center is considered to be unclear, so all the boundaries falling within a small circle of radius 3 are neglected. Region boundaries around T-junctions are locally considered to be straight lines corrupted by noise. The boundary coordinates can me modeled by:

$$\boldsymbol{b}_{ij}(n) = \boldsymbol{t} + n\boldsymbol{\varphi}_{ij} + z(n) \tag{12}$$

where $\boldsymbol{t} = (t_x, t_y)$ is a vector containing the T-junction coordinates. $\boldsymbol{\varphi}_{ij} = (\varphi_x, \varphi_y)$ is a vector indicating the main direction of the boundary and $z(n)$ represents the noise. The tangent vector at each boundary point is approximated with finite differences as $\boldsymbol{\tau}_{ij}(n) = \boldsymbol{b}_{ij}(n) - \boldsymbol{b}_{ij}(n-1)$. To estimate each branch b_{ij} orientation $\boldsymbol{\varphi}_{ij}$, the average tangent vector $\widehat{\boldsymbol{\varphi}}_{ij}$ is found by means of an exponential weighted mean.

$$\widehat{\boldsymbol{\varphi}}_{ij} = \frac{\sum_{n=0}^{N_{ij}-1} \lambda(n)\,\boldsymbol{\tau}_{ij}(n)}{\sum_{n=0}^{N_{ij}-1} \lambda(n)} = \frac{\sum_{n=0}^{N_{ij}-1} \lambda_0^n \boldsymbol{\tau}_{ij}(n)}{\sum_{n=0}^{N_{ij}-1} \lambda_0^n}. \tag{13}$$

The total number of considered points for a branch is N_{ij} and depends directly on the damping factor λ_0. The points near the junction have more importance (and thus are weighted by a larger factor) than the points being further away. Since contour points lie between pixels of integer coordinates, there is a finite number of values for the tangent vectors $\boldsymbol{\tau}_{ij}(n) = (\pm 1, \pm 1)$. This finite set of values introduces high frequency changes in the mean estimation. Therefore, the estimator (13) should attenuate these high variations while keeping the angle estimation as local as possible. The parameter λ_0 controls both the locality of the estimator and frequency selectivity. Typical values are in the range $\lambda_0 = 0.9$–0.99.

Once the three average tangent vectors are available, each region angle θ_i is estimated and the junction angle characteristics evaluated. Considering the angles, ideal shaped T-junctions have a maximum angle of π and a minimum angle of $\frac{\pi}{2}$. Two measures are then proposed:

$$\Delta\theta_{\max} = \|\theta_{\max} - \pi\| \qquad \Delta\theta_{\min} = \|\theta_{\min} - \frac{\pi}{2}\| \tag{14}$$

where $\theta_{\max}$ and $\theta_{\min}$ refer to the maximum and minimum of the three angles respectively. To obtain the confidence value, $\Delta\theta_{\min}$ and $\Delta\theta_{\max}$ are considered to be Rayleigh distributed. With this assumption, two confidences can be obtained using:

$$\Theta_{\max} = \exp\left(-\frac{\Delta\theta_{\max}}{\sigma^2}\right) \tag{15}$$

$\Theta_{\min}$ is computed similarly. $\sigma = \frac{\pi}{6}$. This value is obtained from [8], as the perception of occlusion on T-junctions drops rapidly when angle variations are greater than 30–40 degrees from the ideal angle configuration. By combining these two values, p_{angle} is obtained similarly to (11):

$$p_{\text{angle}} = \frac{2\Theta_{\min}\Theta_{\max}}{\Theta_{\min} + \Theta_{\max}}. \tag{16}$$

3) Curvature: Although curvature is not as important as color and angle, it serves to measure the branch straightness. If boundaries are highly curved, the point may not be perceived as a junction and, instead, only erratic and noisy boundaries are seen. The curvature of the boundaries is measured using the level sets theory. The process of curvature confidence calculation is shown in Fig. 4.

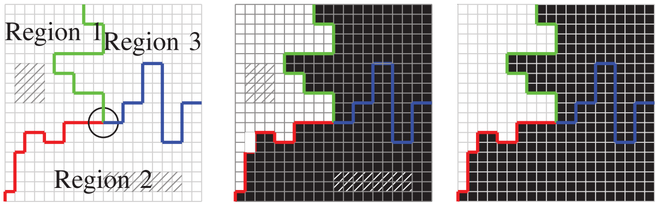


Fig. 4. Process to calculate the curvature. Left: local window with the three regions and some outliers (diagonal striped pixels, belonging to other regions). Center: binary image where Region 1 has been isolated. Right: reconstructed image without outliers.

Each region R_i is isolated creating a binary image of the local window. Note that since the regions may have arbitrary shapes, other regions than R_1, R_2, R_3 may be present in the local window. To eliminate possible interferences from these outliers, a reconstruction process is performed where, from the boundaries, the binary markers are extended eliminating the holes that may be present. The second and third steps in Fig. 4 illustrate this hole filling process. Finally, the mean absolute value $\overline{|\kappa|}_i$ of the curvature $\kappa(x_l, y_l)$ of the two branches forming a region R_i is computed at the boundary points (x_l, y_l) in the binary image using the level sets theory [33]. Each of the $\overline{|\kappa|}_i$ measures (one for each region) are also assumed to be Rayleigh distributed to obtain:

$$\Upsilon_i = \exp\left(-\frac{\overline{|\kappa|}_i}{\sigma_c^2}\right). \tag{17}$$

Similar to color and angle, curvature confidence p_{curve} is obtained by finding $\Upsilon_{\max}$ and $\Upsilon_{\min}$:

$$p_{\text{curve}} = \frac{2\Upsilon_{\min}\Upsilon_{\max}}{\Upsilon_{\min} + \Upsilon_{\max}}. \tag{18}$$

4) Local Depth Gradient Determined by T-Junctions: Previous work on T-junctions [13] imposed unique depth configuration for these kind of cues: the region forming the largest angle was always assumed to lie closer to the viewer. However, experience shows that T-junction may also indicate the opposite depth relation. Since, locally, all kinds of junctions are similar, deciding whether T-junctions are *normal* or *inverted* should be done by looking at other characteristics than intrinsic color, angle and curvature local features. As a result, to determine the sign of the depth gradient of each T-junction, relation with other T-junction configurations are considered, see Fig. 1.

T-junctions actually indicate depth discontinuities but the sign of the discontinuity proved to be rather uncertain. Normally, if an object is really occluding other objects in the background, more than one T-junction is likely to be formed in the image, and all these T-junctions may have the same region/object as the occluding region. This is why a global reasoning is helpful. Moreover, it is possible to detect a T-junction even though no real occlusion relation exists. False detections often occur due to color or texture variations. In our case, as a starting point to the global reasoning explained

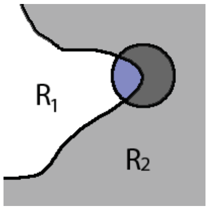


Fig. 5. Normally, convex shapes present less area in small neighborhoods centered on contour points. Convex regions, as in R_1 here, are perceived as foreground while R_2 is perceived as background.

in Section IV-D, we will assume that detected T-junction are not false detection and are of type *normal*.

This initial guess has a low confidence and will be allowed to change when estimating the global depth ordering of the scene. That is, in some circumstances, the depth gradient of a T-junction will be changed if there are many other occlusion relations indicating the opposite depth relationship.

B. Convexity Estimation

Convexity depth cues are defined locally at region boundaries. A region R_1 is convex with respect to R_2 if, on average, the curvature vector on the common boundary is pointing towards R_1. If R_1 appears to be convex, it is perceptually seen as the foreground region (and thus, closer to the viewer). Since computing derivatives to estimate the curvature may lead to imprecise results, an alternative approach is presented. Generally, when examining boundary pixels, if R_1 presents less area than R_2 in a local neighborhood, R_1 may be seen as convex, see Fig. 5. Formally, the overall boundary convexity is obtained from the combinations of two measures:

$$\zeta_c(R_1, R_2) = \sum_{(x,y)\in\Gamma} \frac{\alpha(x,y)}{L} \sum_{(x,y)\in\Gamma} \frac{w(x,y)}{L} \tag{19}$$

with $\alpha(x, y) = 1$ if the area of R_1 is greater than the area of R_2 in $\Omega(x, y)$, and $\alpha(x, y) = -1$ otherwise. The function $0 \leq w(x, y) \leq 1$ is a weighting function of the points and it is chosen to be the normalized Sobel gradient of the image, although other gradient operators work too. L is the number of points where the measure $\alpha(x, y)$ is calculated. The overall convexity confidence of a boundary is:

$$\zeta(R_1, R_2) = 1 - \exp\left(-\frac{1}{\gamma_c} \|\zeta_c(R_1, R_2)\|\right) \tag{20}$$

γ_c has been determined experimentally and set to $\frac{1}{12}$. If the result $\zeta_c(R_1, R_2)$ is positive, R_1 is considered to be convex and, therefore, on top of R_2 with confidence $\zeta(R_1, R_2)$. The converse indicates that R_2 is on top of R_1. To make the measure as scale invariant as possible, the neighborhood $\Omega(x, y)$ of a pixel is chosen to be a circular window with a radius of about the 5% of the contour length. Points lying near junctions, image borders and other regions are discarded for the measure. Contours having small lengths ($L < 100$ pixels) are considered to be non-significant for convexity cues.

IV. Depth Ordering

Once the BPT has been constructed as described in sections II and III, a further processing is required to obtain the relative depth order. The depth map is constructed by selecting some of the regions represented by the BPT and the process can be formulated as a BPT pruning because the leaves of the pruned tree represent the regions belonging to a partition. The pruning is optimal in the sense that it minimizes a cost function. To this end, an initial depth partition is obtained by an initial BPT pruning and the tree is iteratively pruned, reducing the number of regions. During this process, for each examined partition, the depth relations of T-junctions and convexity cues are used to determine the region depth order by means of a probabilistic framework. Since several cues may indicate opposite depth relationships, a conflict resolution is necessary. The final output of the system is the depth partition with minimum cost.

A. Initial Partition/T-Junction Selection

Prior to entering the minimization process, the BPT is pruned to simplify the solution search space. Ideally, all possible partitions resulting from BPT pruning should be examined, but the high dimensionality of the problem encourages to cut the search range to a few solutions. Restricting the solution space does not prevent to get to the optimal solution, as long as the true solution remains after the restriction. Since humans partially interpret scenes by reasoning from characteristic points, it seems logical that the remaining solution space should contain the most prominent estimated cues. To this end, the initial BPT pruning is done by preserving the more confident estimated T-junction points.

During the BPT construction, every point of the image is assigned a T-junction confidence value $0 < p < 1$. Determining which T-junctions could indicate occlusion is performed by thresholding. Discarded candidates are the ones with $p < 0.1$ or $p < 0.2p_{\max}$, where $p_{\max}$ is the maximum confidence value found during the BPT construction. Discarding points with $p < 0.1$ attempts to eliminate some false alarms, although this threshold leaves practically untouched prominent T-junctions. The relative threshold $0.2p_{\max}$ is chosen to eliminate low-contrasted T-junctions compared to the overall image contrast. About 10–30 T-junctions are preserved on average per image.

The initial BPT pruning is performed by choosing the minimum amount of regions preserving the remaining T-junctions. An example of an initial partition is shown in Fig. 6.

B. Criterion Definition

After an initial partition selection, the algorithm performs a minimization process on the BPT structure. The goal of this process is to retrieve the 'best' depth order partition D. The criterion to determine the best solution is defined by equation (21). This criterion relies on three notions: First, estimated cues (T-junctions and convexity relations) are supposed to be reliable. That is, the algorithm should try to accept as many

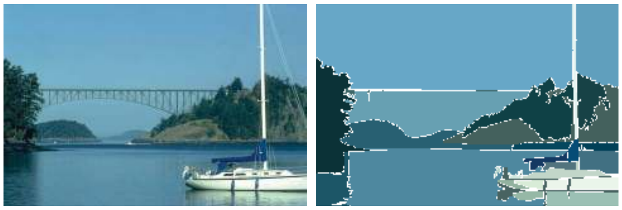


Fig. 6. Original image (left) and the partition generated after T-junction selection (right). Regions are represented by their mean color.

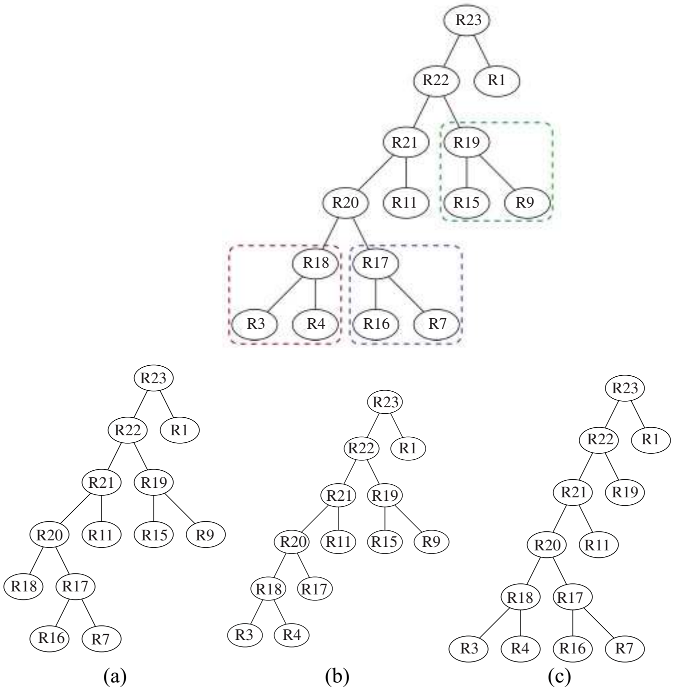


Fig. 7. Three allowed prunings of a given BPT. For each pruning, the framed leaf nodes are merged to their parent, reducing the number of BPT nodes by one. All other possible prunings in this BPT reduce the number of leaves by more than one. (a)–(c) Results of the prunings (red, blue, and green) are shown at the bottom.

estimated depth cues as possible. Second, natural images can be decomposed with few depth planes/regions. Third, regions are expected to have at least one depth relationship with their neighbors. Behind these three intuitions, the following criterion for D can be defined:

$$C(D) = \sum_{i \in R} p_i + \gamma_N N + \gamma_u U \tag{21}$$

where R is the set of rejected depth cues (T-junctions and convexity relations) for a particular solution. p_i is the confidence of a T-junction or the confidence of a convexity relation between region boundaries. γ_N and γ_u are the weights for N and U. U refers to the number of isolated regions, that is, regions which do not have any depth relationship with any other in the final depth image. Finally, N stands for the number of regions composing the final depth partition.

The value of γ_u is set rather high to efficiently minimize the number of isolated regions. In practice, values $\gamma_U > 2$ produce good results. The value γ_N is chosen depending on the values of the confidences found for T-junction and convexity cues. Since γ_N weights the number of regions N, setting a high value encourages the final solution to have few regions. If the value is high enough, the system output behaves like a foreground/background segregation system, separating the front-most depth plane from the deeper regions. Usually, $p_{\min} < \gamma_N < p_{\max}$, with $p_{\min}$ and $p_{\max}$ being the minimum and maximum confidences found in the image respectively. $\gamma_N = p_{\min}$ throughout the experiments of this paper.

C. Minimization Process

The adopted scheme assumes that the final depth ordered partition D is the one minimizing the criterion $C(D)$. Seeking for the global minimum of this criterion starting from the initial partition alone has proven to be difficult, as $C(D)$ is extremely non-convex, with many local minima. The work in [29] attempts to find a global minimum by searching for the optimal solution with a RANSAC-style algorithm. For images with few T-junctions, the solution found can be near the optimal. Nevertheless, since the complexity of [29] is exponential with the number of T-junctions, the process turned out to be unfeasible for relatively complex images. In this work, the BPT is used to explore greedily a subset of solutions and to minimize the criterion. The approach follows a strategy that gradually prunes the tree until the root node is reached.

The initial pruned tree B_0 is obtained by the initial T-junction selection. At each iteration t, for each tree B_t, a set of K feasible solutions B_t^k, $k = 1 \ldots K$, are generated by considering all the possible prunings that reduce the number of leaves in B_t by exactly one. In the example of Fig. 7, three such prunings are possible. Since the leaves of each pruned BPT define a partition, the depth ordered partition D_t^k is obtained for each B_t^k. With all D_t^k available, the next tree B_{t+1} is the tree corresponding to the partition of minimum cost:

$$D_{\min}^k = \arg\min_{D_i^k} \left(C(D_1^k), C(D_2^k), \ldots, C(D_s^k) \right). \tag{22}$$

The pruning process is applied successively, obtaining at each iteration B_t and D_t, $t = 1 \ldots T$. At the final iteration T, the tree has only one leaf and cannot be further pruned. The final depth ordered partition is:

$$D_{\min} = \arg\min_{D_{\min}^k} \left(C(D_{\min}^1), C(D_{\min}^2), \ldots, C(D_{\min}^T) \right). \tag{23}$$

As can be seen in the previous minimization procedure, a depth ordered partition has to be generated from each pruned tree B. To this end, local depth cues should propagate their depth information through regions by means of a Depth Order Graph (DOG). Since conflicts may appear, a probabilistic scheme to resolve these conflicts is proposed.

D. Probabilistic Framework for Depth Ordering

Since the initially computed cues are merely local, a global reasoning should be done to arrive at a consistent solution for the whole image. To this end, a Depth Order Graph (DOG) is constructed for each partition extracted from the BPT. Nodes in the graph represent regions of the partition extracted from the leaves of the BPT. The depth relations are represented in the DOG by directed weighted edges, going from the foreground region to the background one. There is exactly one edge going from region R_1 to R_2 if there is a depth cue i (T-junction or convexity) stating that R_1 is in front of R_2. The weight of this edge is the cue confidence, p_i.

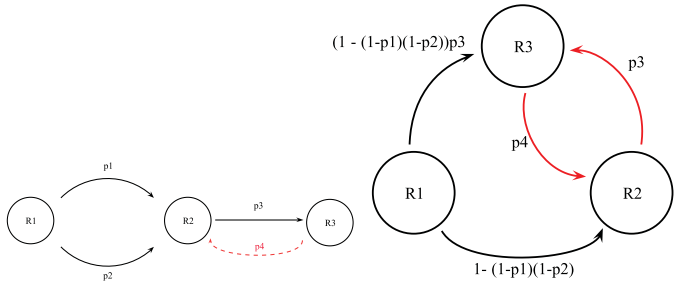
To order the regions according to depth, the DOG should be acyclic (with no conflicts). To achieve such a graph structure, the DOG has to be modified. To this end, it is interpreted as a network of reliable links [34]. Each edge in the DOG associated with a cue i is reliable with probability p_i. A region R_j is reachable from R_i if there exists at least one directed path that goes from the former region to the latter. The probability of existence of this path ρ_{ij} is defined as reliability in [34], and referred in this work as probability of precedence (PoP) due to its nature. That is, the PoP ρ_{ij} is the probability of a region R_i to be in the foreground of R_j. The proposed solution to create a direct acyclic graph from the DOG can be summarized as follows:

1) Compute the PoP, ρ_{ij}, for every pair of regions (nodes), R_i and R_j, that is, the probability that R_i is in the foreground of R_j.
2) Examine all pairs ρ_{ij} and ρ_{ji}. If a cycle is present, both R_i and R_j can be foreground and, therefore, both $\rho_{ij}, \rho_{ji} \neq 0$.
3) In case of conflict, modify one of the paths from R_i to R_j or vice versa to eliminate the cycle.

The probability ρ_{ij} can be calculated exactly by the inclusion–exclusion principle [34]. Nevertheless, its computation cost encourages to find approximate solutions. Since the exact value of ρ_{ij} is not the ultimate goal of the conflict resolution step, an upper bound proved to give reasonable results. The PoP is computed using a variant of the Floyd–Warshall algorithm [35]:

```
for $j$=1...$|V|$ do
  for $i$=1...$|V|$ do
    for $k$=1...$|V|$ do
      $\rho_{ik}^{n+1} = \rho_{ik}^{n} + \rho_{ij}^{n}\rho_{jk}^{n} - \rho_{ik}^{n}\rho_{ij}^{n}\rho_{jk}^{n}$
    end for
  end for
end for
```

The computation of all the pairs ρ_{ij} leads to a new graph, DOG$^+$, which is the transitive closure of the DOG, see Fig. 8. The transitive closure of a graph G is a graph G^+ with the same nodes of G but that contains a direct edge (possibly weighted) from node R_i to R_j if there exists a path P_q in G that connects both nodes. In our case, the transitive closure of the DOG contains edges with weights ρ_{ij}. The graph G^+ allows to easily detect cycles as paths with arbitrary lengths are reduced to direct edges. It is known that identifying all cycles in a graph G is an NP problem [36], meaning that there is no efficient solution. Instead, making use of the DOG$^+$, cycles can be easily detected by direct comparison of ρ_{ij} and ρ_{ji}.



(a) (b)

Fig. 8. Simple DOG example (a) and its transitive closure (b). Edge weights are shown close to each edge. Red edges form a conflict. The red stripped edge in the left graph breaks a cycle if it is deleted.

A conflict may occur mainly because of two factors. The first may be because of some false T-junction or convexity depth relations, false alarms have been introduced. The second may be because self occlusion actually exists in the image. Assuming that self-occlusion is rather difficult to find in natural images, the conflicts are said to come from bad depth cue selection. Translating this reasoning to the DOG, each time a conflict is found, either ρ_{ij} or ρ_{ji} is assumed to have been wrongly estimated.

The proposed approach aims to break low-confident depth relations. Assuming $\rho_{ij} < \rho_{ji}$, some modifications on the paths that go from R_i to R_j should be done by deleting or turning some edges (and thus possibly breaking the cycle). For each conflicting path P, the modified cue is the one corresponding to the edge with lowest confidence. Two different cases appear. First, if the edge represents a convexity depth cue, the cue is discarded and the corresponding edge removed. Second, if the edge nature comes from a T-junction, a slightly different approach is used. Since the occlusion relation in a T-junction is not clear, the edge is first reverted, thus changing a *normal* T-junction to an *inverted* one. If it still creates a conflict, it is discarded.

This process is repeated until no cycles in the DOG are found. When an acyclic graph is available, the depth order of each region can be computed using a topological partial sort to obtain the depth ordered partition D.

E. Depth Ordering

The depth ordering of the regions/nodes forming the DOG is performed using a topological partial ordering [35]. Since in a depth image two different regions may have the same depth order, $R_1 = R_2$, (i.e. do not have any depth relationship between them), strict ordering of the elements is not suitable. Instead, partial order permits that two elements of a set have the same order when sorted. After the depth for each region is computed, the criterion (21) can be evaluated. When a region doe not share any occlusion cue with its neighbors, its depth is chosen to be the depth of the most similar adjacent region according to the distance (2).

V. Experimental Results

The proposed depth ordering system is compared with state of the art systems on f/g labeling and on depth estimation. Note that our system defines the depth information on a region basis, whereas the f/g algorithms [14], [15] output is a labeling on points of image contours that may not be closed. Therefore f/g algorithms do not allow to create complete depth order partitions. Nevertheless, the existence of a ground truth f/g database makes the comparison with these

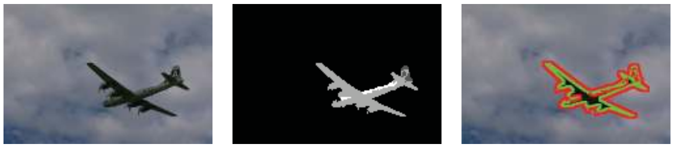


Fig. 9. From left to right. Original image, depth order partition, and f/g labeling on contours. In the depth order partition, white regions are closer to the viewer. In the f/g labels, green pixels belong to the foreground region, the red belong to the ground region.

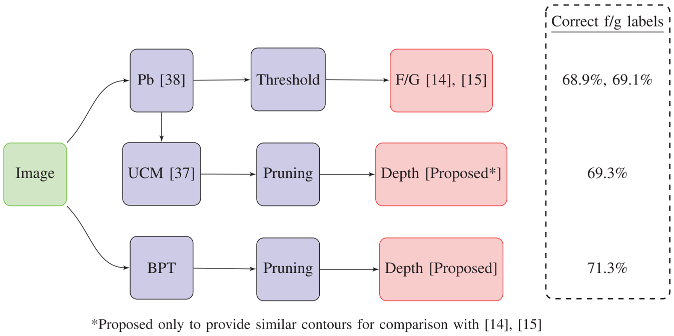


Fig. 10. First row: systems [14], [15] threshold the output of the soft contour detector [38] and infer the figure/ground labels on the resulting contours. Second row: the ultrametric contour maps (UCMs) approach [37] uses the contours of [38] as an input and the proposed optimization strategy estimates the depth order map. Third row: complete proposed scheme using a BPT image representation and the optimization strategy.

systems attractive. The evaluation can be done quantitatively, determining the number of contour points where the labeling is correctly estimated. Depth ordering systems are more difficult to compare, as no ground truth database exists with relative depth ordering. Instead, a qualitative evaluation is presented by examining the overall image depth structure.

A. Quantitative Performance on Figure/Ground Assignment

One of the byproduct of depth order estimation can be seen as a solution to the boundary ownership problem. At the depth region boundaries, the region lying closer to the viewer is considered to be the owner the of the boundary, or figure side. The further region is considered the ground side. To obtain the f/g labels, the contour points are extracted from the final depth order partition at points where the depth gradient is not null, assigning either a figure or ground label depending on the local depth configuration, as shown in Fig. 9. To evaluate the system, ground truth f/g labels of the dataset [37] are used.

Fig. 10 shows an outline of the proposed comparison. The first row of Fig. 10 represents the state of the art in f/g labeling [14], [15]. It consists in extracting non closed contours on the original image [38], then in thresholding these contours and in assigning the f/g relationships on the contours following [14], [15]. In figure 10, the % of correct assignment is indicated. As our algorithm involves two major parts: 1) the image representation by means of a BPT and 2) an optimization strategy extracting depth order partitions from the BPT, two experiments are done.

The goal of the experiment represented by the second row of figure 10 is to demonstrate the interest of the proposed optimization strategy running on an image region-based representation and to compare it with the strategies described in [14] and [15]. As we need an image region-based representation, we have chosen to construct it by means of the ultrametric contour map (UCM) described in reference [37] which rely on the contours [38] used in the first experiment. The use of UCM allows us to construct a tree based representation of the image as the BPT and we can use our optimization strategy to extract the depth order. As can be seen in Fig. 10, the results of this strategy slightly outperform the results in [14], [15] and therefore demonstrate the interest of the optimization strategy proposed here.

TABLE I

TABLE SUMMARIZING THE EVOLUTION OF F/G LABELS CORRECTNESS WHEN THREE FACTORS ARE CHANGED IN THE SYSTEM. BPT±d_d INDICATES WHETHER DEPTH IS CONSIDERED OR NOT IN THE BPT CONSTRUCTION. ROWS ON THE TABLE INDICATE, FROM TOP TO BOTTOM, WHEN ONLY T-JUNCTION, CONVEXITY, OR BOTH DEPTH RELATIONS ARE USED IN THE OPTIMIZATION STRATEGY

	$BPT+d_d$	$BPT-d_d$
Only T-junctions	62.9%	64.6%
Only convexity	65.45%	64.9%
T-junction+convexity	71.3%	67.1%

Finally, the last row of Fig. 10 shows the last experiment, the goal of which is to demonstrate the interest of the BPT construction compared to the UCM approach as a region-based representation of the image. Here a clear improvement can be observed in the context of the f/g assignment and demonstrate the interest of the proposed BPT tool.

The next set of experiments study the system behavior when some factors are excluded. Table I shows the f/g labeling performance when the system is run with or without considering: depth region distance in the BPT construction, T-junction depth relations and convexity depth relations in the optimization strategy. Following intuition, the best behavior is when all factors are used, including the depth distance. Table I confirms that the introduction of a depth factor into the BPT construction contributes positively to the final result. Results also show that, in the optimization strategy, T-junction depth relations are less reliable than convexity cues, possibly because junction detection is still a difficult challenge [32].

Results obtained on depth ordering and f/g labeling are shown in Figures 11 and 12. Although f/g labeling performance is similar to current state of the art algorithms, it has to be noticed that the proposed system gives much more information than simple labels on contours. For instance, the depth order image can be considered as a possible segmentation. In contrast to our approach, f/g labeling systems [14], [15] operate only on boundaries, discarding region information. To our knowledge, the only work that proposes a joint segmentation and f/g labeling system is found in [16], but no quantitative comparison is available. Note that a region-based system is able to apply global reasoning and to resolve possible depth conflict providing more robustness to the estimation process.

B. Qualitative Analysis

High level information systems in [5], [6] can be compared with the proposed algorithm as they produce a complete

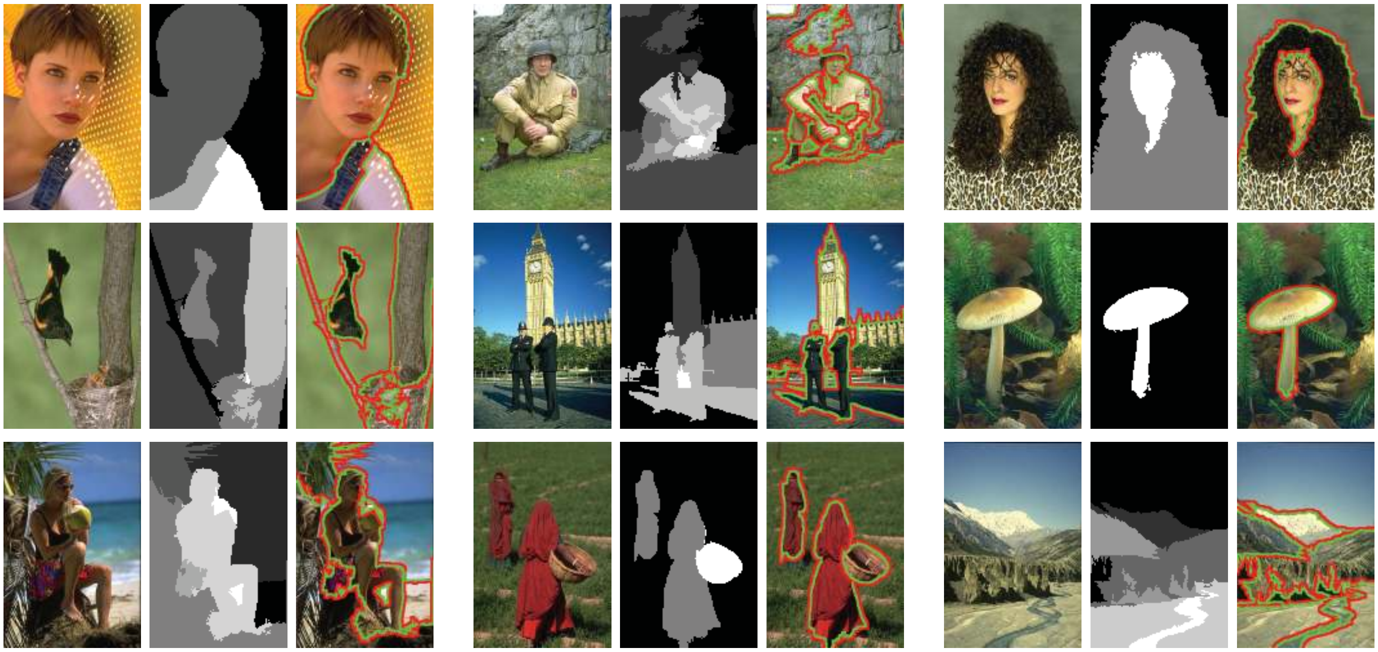


Fig. 11. Results on depth estimation and f/g assignment on some of the BSDS500 images. From left to right, for each column: original image, depth order estimation, and f/g boundaries.

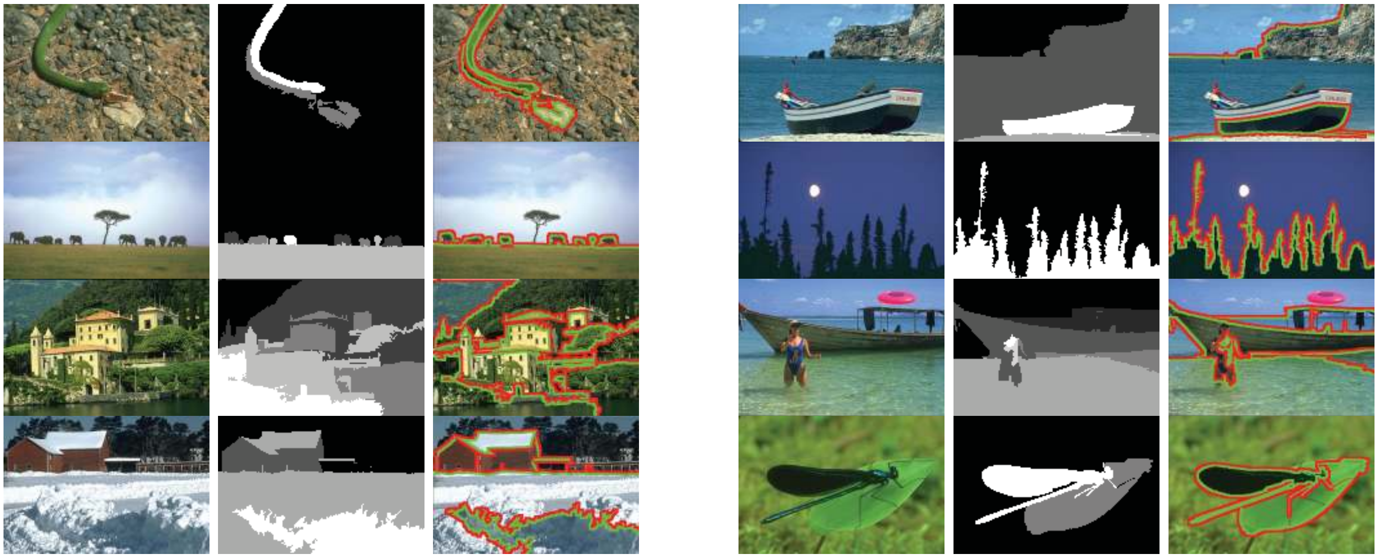


Fig. 12. More results on the BSDS500 dataset. From left to right, for each column: original image with estimated T-junctions, depth order estimation, and f/g boundaries overlaid in the original image.

depth partition. The comparison of the systems should be done qualitatively, noting that [5], [6] estimate the absolute depth of the image while the proposed approach only offers relative depth order. Despite the differences, results can be contrasted by looking at the major structure of the final depth partitions.

Results in Fig. 13 show that the proposed system generates clearer boundaries in most cases. However, [6] and [5] permit arbitrary surface orientation, leading to smooth depth gradients which our system is unable to obtain. However, most of their results present the same general image structure, being the lower regions the ones that normally are closer to the viewer, specially in [6]. This can be a drawback if non-typical pictures are presented to the system as the algorithm will try to fit the learned model into the input image. Of course, this can be overcome by a more extensive training dataset but the variety of scenes that must be presented may turn this process unfeasible.

Our algorithm does not make any assumption on the type of images and relies only on low-level information. Obviously, trusting only pixel information, without any previous knowledge of the scene can be limiting, but it also has its positive points. In particular, there is no real restriction on the scene type. This makes the algorithm work in more situations such as landscape, indoor or portrait images. Such examples can be found in Fig. 14 where a variety of scenes are presented.

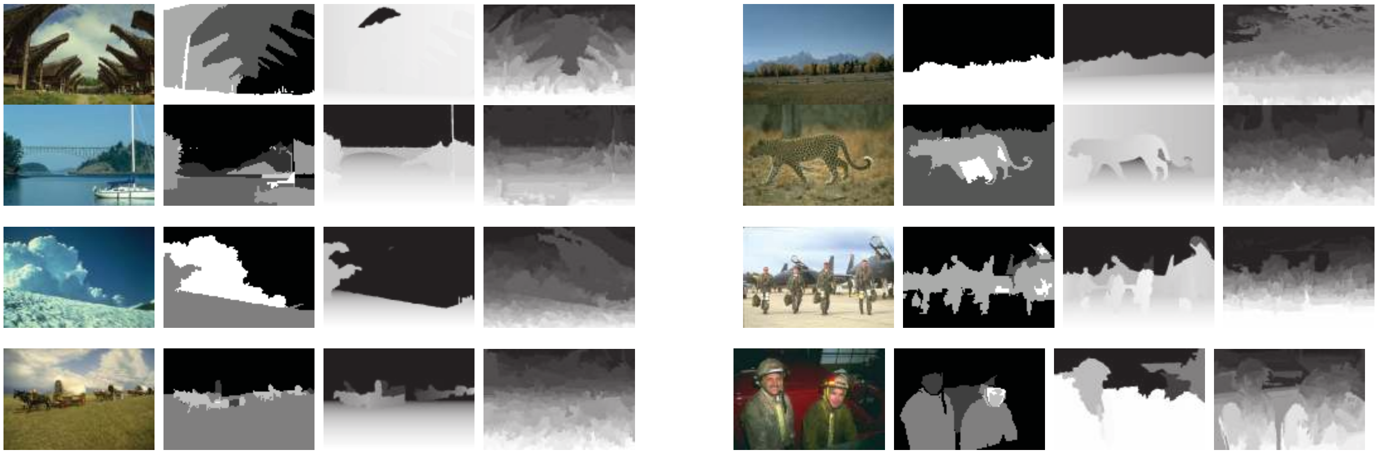


Fig. 13. From left to right. Original image, results of the proposed system, results from [5], and results from [6]. White regions are closer to the viewer, while black ones are further away.

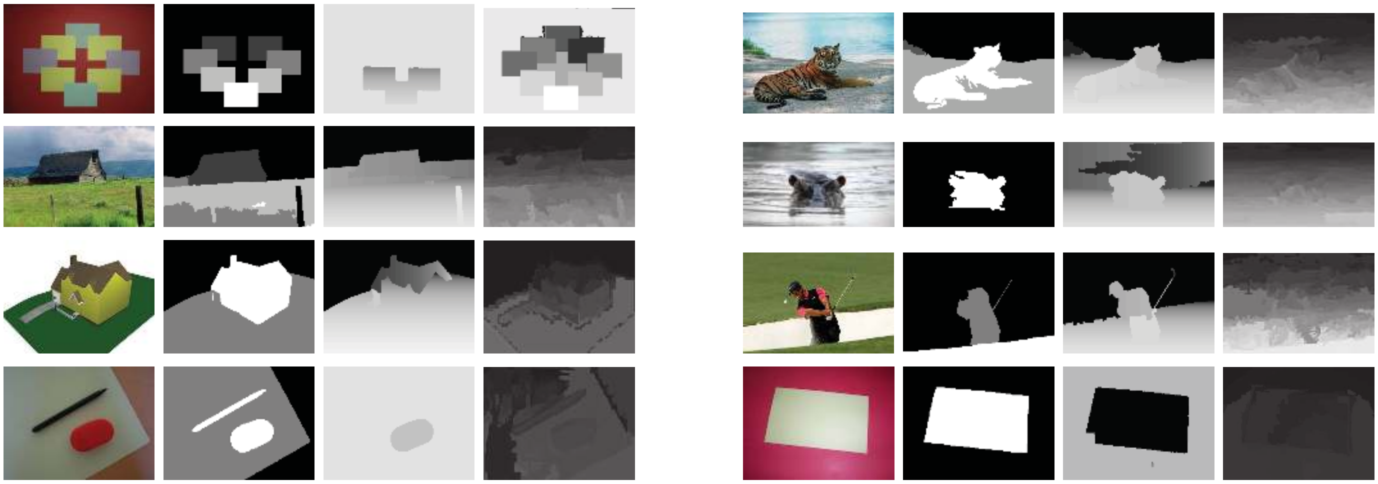


Fig. 14. Proposed algorithm runs on several situations. For each combination of four columns, from left to right: original image, proposed system results, results from [5], and results from [6]. Even in such different scenes, the considered low level cues (occlusion and convexity) remain valid, obtaining reasonable depth order maps. High level information approaches always show a similar structure.

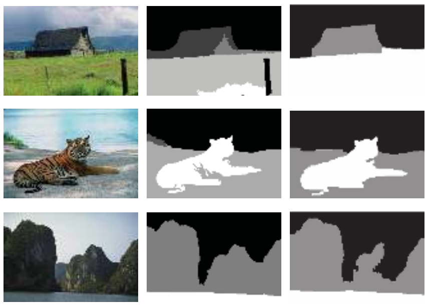


Fig. 15. From left to right. Original image, results of the proposed system, and results of the region-based approach in [13].

Some of these scenes were downloaded from the Internet, some were taken with a camera and some others are from the Corel database.

Finally, in Fig. 15 a comparison with the region-based system discussed in [13] is shown. Since both systems rely on the same low-level cues, a similar performance is expected. However, the integration of cue estimation and segmentation helps to retrieve more depth cues resulting in clearer boundaries and more detailed depth maps.

C. Introducing User Interaction

For some applications, user interaction can naturally be integrated in the working flow and can be used to improve the quality of the depth order map. With very little modification, the unsupervised system we have described can be adapted to accept user input. If the user introduces some depth markers in the images, the given information can be used to force some depth relations. There could be many situations where this extension is desirable. For example, user information may overcome some system limitations. Moreover, user may be interested in accurately ordering some parts of the image, leaving all the other regions to be ordered automatically. Since the proposed system is originally designed to perform in an unsupervised way, unlike [4], it is able to infer extra depth planes other than the ones introduced by the user. Markers can be simply defined by roughly marking areas of the image with gray levels. To integrate this information

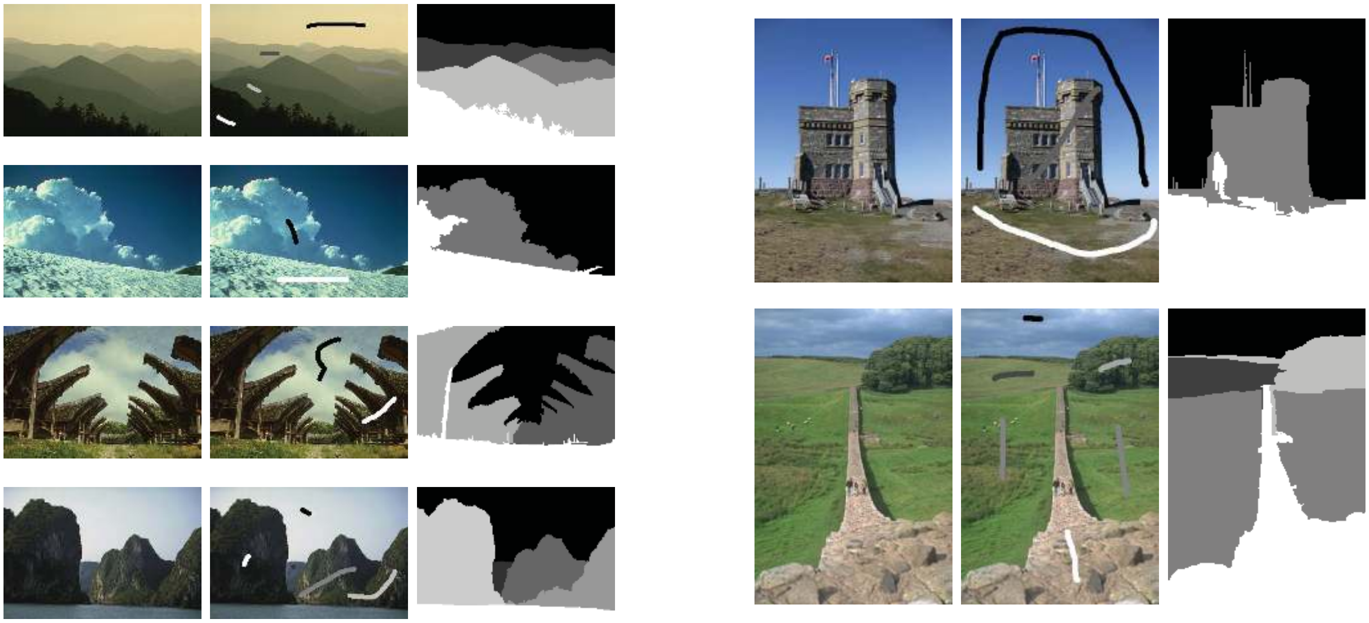


Fig. 16. From left to right, for both columns. Original image, image with user defined markers, and retrieved depth ordered partition.

with the depth ordering stage, two little changes are proposed: one concerns the initial BPT pruning and the other the DOG construction.

a) Initial BPT pruning: A part from preserving the most important T-junction at the initial partition, the pruning must also preserve user input markers at different regions.

b) Depth ordering: Each pair markers from two different regions introduce a fully confident depth relation. That is, these edges are assigned the maximum confidence $p = 1$, making sure that no edge is deleted in the conflict resolution step and the final depth ordered partition contains all the user markers.

Examples of the system accepting user interaction are shown in Fig. 16 showing that, with little user information, accurate orderings can be obtained.

VI. Conclusion

This paper has proposed a system which relies only on low level image cues for monocular image segmentation and depth ordering. Despite the simplicity of the used depth cues, the algorithm offered results comparable to other approaches which base their reasoning on higher level information. The proposed system involves several contributions compared to existing algorithm. The most important innovations are the joint T-junction estimation and BPT construction which adds depth information to the process as well as the tree pruning algorithm which minimize a global criterion. Additionally, the proposed region color model and region distance were not used before on a BPT construction.

Moreover, we have shown how user interaction can be easily and naturally integrated in the processing architecture. Projecting the 3-D world to a 2-D plane implies an inherent loss of information which cannot be recovered completely using a single image. The generated depth ordering can nevertheless be used in several environments.

1) Visualization of images giving a pseudo-depth impression
2) Object editing by depth (foreground/background removal)
3) Rough depth representation as input for full 3-D systems

Moreover, as stated in Section I the huge amount of 2-D content that already exists: videos and photos, encourage the development of systems to perform the inverse operations. While generating accurate depth maps is a challenging task, defining depth planes is proven to be possible. In fact, using occlusion cues does only permit to provide constant depth regions, which in some cases are not able to describe accurately the geometry of the scene. Nevertheless, the depth perception of the scene is preserved even with this strong restriction.

References

[1] B. Ward, S. Bing Kang, and E. Bennett, “Depth director: A system for adding depth to movies,” *IEEE Comput. Graph. Appl.*, vol. 31, no. 1, pp. 36–48, Jan.–Feb. 2011.

[2] O. Wang, M. Lang, M. Frei, A. Hornung, A. Smolic, and M. Gross, “Stereobrush: Interactive 2-D to 3-D conversion using discontinuous warps,” in *Proc. 8th Eurograph. Symp. Sketch-Based Inter. Model.*, 2011, pp. 47–54.

[3] C. Bond, “System and process for transforming two-dimensional images into three-dimensional images,” U.S. Patent 0 050 864, Mar. 3, 2011.

[4] R. R. Raymond Phan and D. Androutsos, “Semi-automatic 2D to 3D image conversion using a hybrid random walks and graph cuts based approach,” in *Proc. Int. Conf. Acoust. Speech Signal Process.*, May 2011, pp. 897–900.

[5] D. Hoiem, A. A. Efros, and M. Hebert, “Recovering occlusion boundaries from an image,” *Int. J. Comput. Vis.*, vol. 91, no. 3, pp. 328–346, 2011.

[6] A. Saxena, A. Ng, and S. Chung, “Learning depth from single monocular images,” in *Neural Information Processing Systems*, vol. 18, 2005.

[7] C. Swain, “Integration of monocular cues to create depth effect,” in *Proc. Int. Conf. Acoust. Speech Signal Process.*, vol. 4. Apr. 1997, pp. 2745–2748.

[8] J. McDermott, “Psychophysics with junctions in real images,” *Perception*, vol. 33, no. 9, pp. 1101–1127, 2004.

[9] J. Cutrona and N. Bonnet, "Two methods for semi-automatic image segmentation based on fuzzy connectedness and watersheds," in *Proc. Int. Conf. Visualizat., Imag. Image Process.*, Sep. 2001, pp. 1–5.

[10] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.

[11] A. Criminisi, I. Reid, and A. Zisserman, "Single view metrology," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, vol. 1. Sep. 1999, pp. 434–441.

[12] A. Torralba and A. Oliva, "Depth estimation from image structure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 9, pp. 1226–1238, Sep. 2002.

[13] M. Dimiccoli, "Monocular depth estimation for image segmentation and filtering," Dept. Signal Theory Commun., Ph.D. dissertation, Univ. Politecnica de Catalunya, Barcelona, Spain, 2009.

[14] X. Ren, C. Fowlkes, and J. Malik, "Figure/ground assignment in natural images," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 614–627.

[15] I. Leichter and M. Lindenbaum, "Boundary ownership by lifting to 2.1d," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep.–Oct. 2009, pp. 9–16.

[16] M. Maire, "Simultaneous segmentation and figure/ground organization using angular embedding," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 450–464.

[17] D. Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. San Francisco, CA, USA: W.H. Freeman, 1982.

[18] S. H. Schwartz, *Visual Perception: A Clinical Orientation*, 3rd ed. New York, USA: McGraw-Hill, May 2004.

[19] S. M. Smith and J. M. Brady, "Susan-a new approach to low level image processing," *Int. J. Comput. Vis.*, vol. 23, pp. 45–78, May 1997.

[20] P. Salembier and L. Garrido, "Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 561–576, Apr. 2000.

[21] V. Vilaplana, F. Marques, and P. Salembier, "Binary partition trees for object detection," *IEEE Trans. Image Process.*, vol. 17, no. 11, pp. 2201–2216, Nov. 2008.

[22] M. A. Ruzon and C. Tomasi, "Edge, junction, and corner detection using color distributions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1281–1295, Nov. 2001.

[23] F. Calderero and F. Marques, "Region merging techniques using information theory statistical measures," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1567–1586, Jun. 2010.

[24] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, and A. Yamada, "Color and texture descriptors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 6, pp. 703–715, Jan. 1998.

[25] M. Orchard and C. Bouman, "Color quantization of images," *IEEE Trans. Signal Process.*, vol. 39, no. 12, pp. 2677–2690, Dec. 1991.

[26] R. N. Shepard, "Toward a universal law of generalization for psychological science," *Science*, vol. 237, no. 4820, pp. 1317–1323, 1987.

[27] Y. Rubner, C. Tomasi, and L. Guibas, "A metric for distributions with applications to image databases," in *Proc. 6th Int. Conf. Comput. Vis.*, Jan. 1998, pp. 59–66.

[28] F. Hillier and G. Lieberman, *Introduction to Mathematical Programming*. New York, USA: McGraw-Hill, 1990.

[29] G. Palou and P. Salembier, "Occlusion-based depth ordering on monocular images with binary partition tree," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2011, pp. 1093–1096.

[30] R. Bergevin and A. Bubel, "Detection and characterization of junctions in a 2D image," *Comput. Vis. Image Understand.*, vol. 93, no. 3, pp. 288–309, 2004.

[31] C. Harris and M. Stephens, "A combined corner and edge detection," in *Proc. 4th Alvey Vis. Conf.*, 1988, pp. 147–151.

[32] M. Maire, P. Arbelaez, C. Fowlkes, and J. Malik, "Using contours to detect and localize junctions in natural images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[33] F. Guichard, L. Moisan, and J.-M. Morel, A review of P.D.E. models in image processing and image analysis," *J. Phys. IV*, vol. 12, no. 1, pp. 137–154, Mar. 2002.

[34] R. Terruggia, "Reliability analysis of probabilistic networks," Ph.D. dissertation, Dept. Comput. Sci., Univ. degli Studi di Torino, Turin, Italy, 2010.

[35] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction Algorithms*, 2nd ed. Cambridge, MA, USA: MIT Press, Sep. 2001.

[36] P. Mateti and N. Deo, "On algorithms for enumerating all circuits of a graph," *J. Comput. Soc. Ind. Appl. Math.*, vol. 5, no. 1, pp. 90–99, 1976.

[37] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," EECS Dept, Univ. California, Berkeley, USA, Tech. Rep. UCB/EECS-2010-17, Feb. 2010.

[38] D. Martin, C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530–549, May 2004.

Guillem Palou (S'11) received the degree in electrical engineering from the Technical University of Catalonia (UPC), Barcelona, Spain, and the final degree from the Massachusetts Institute of Technology, Boston, MA, USA, both in 2009. He is currently pursuing the Ph.D. degree with the Image Processing Group, UPC.

His current research interests include monocular depth perception, image segmentation, and feature detection.

Mr. Palou was a recipient of a Scholarship from the Generalitat de Catalunya.

Philippe Salembier (M'96–SM'09–F'12) received the degree from the École Polytechnique, Paris, France, and the degree from the École Nationale Supérieure des Télécommunications, Paris, in 1983 and 1985, respectively, and the Ph.D. degree from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 1991.

He was a Post-Doctoral Fellow with the Harvard Robotics Laboratory, Cambridge, MA, USA, in 1991. From 1985 to 1989, he was with the Laboratoires d'Electronique Philips, Limeil-Brevannes, France, where he was involved in research on digital communications and signal processing for HDTV. In 1989, he joined the Signal Processing Laboratory, EPFL, where he was engaged in research on image processing. In 1991, he was with the Harvard Robotics Laboratory and then with the Technical University of Catalonia, Barcelona, Spain, where he is currently a Professor of digital signal and image processing. His current research interests include image and sequence coding, compression and indexing, segmentation, video sequence analysis, mathematical morphology, level sets, and nonlinear filtering.

Dr. Salembier was an Area Editor of the *Journal of Visual Communication and Image Representation* (Academic Press) from 1995 to 1998 and an AdCom Officer of the European Association for Signal Processing (EURASIP) from 1994 to 1999. He was a Guest Editor of special issues of *Signal Processing* on mathematical morphology in 1994 and on video sequence analysis in 1998. He was a Co-Editor of a special issue of *Signal Processing: Image Communication on MPEG-7 Technology* in 2000. He was the Co-Editor-In-Chief of *Signal Processing* from 2001 to 2002. He was a member on the Image and Multidimensional Signal Processing Technical Committee of the IEEE Signal Processing Society from 2000 to 2006 and was the Technical Chair of the IEEE International Conference on Image Processing 2003 in Barcelona. He was an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING from 2002 to 2008 and the *IEEE Signal Processing Letters* from 2005 to 2008. He is currently an Associate Editor of the *EURASIP Journal on Image and Video Processing, Signal Processing: Image Communication* (Elsevier), and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. He was involved in the definition of the MPEG-7 standard (Multimedia Content Description Interface) as the Chair of the Multimedia Description Scheme Group from 1999 to 2001.