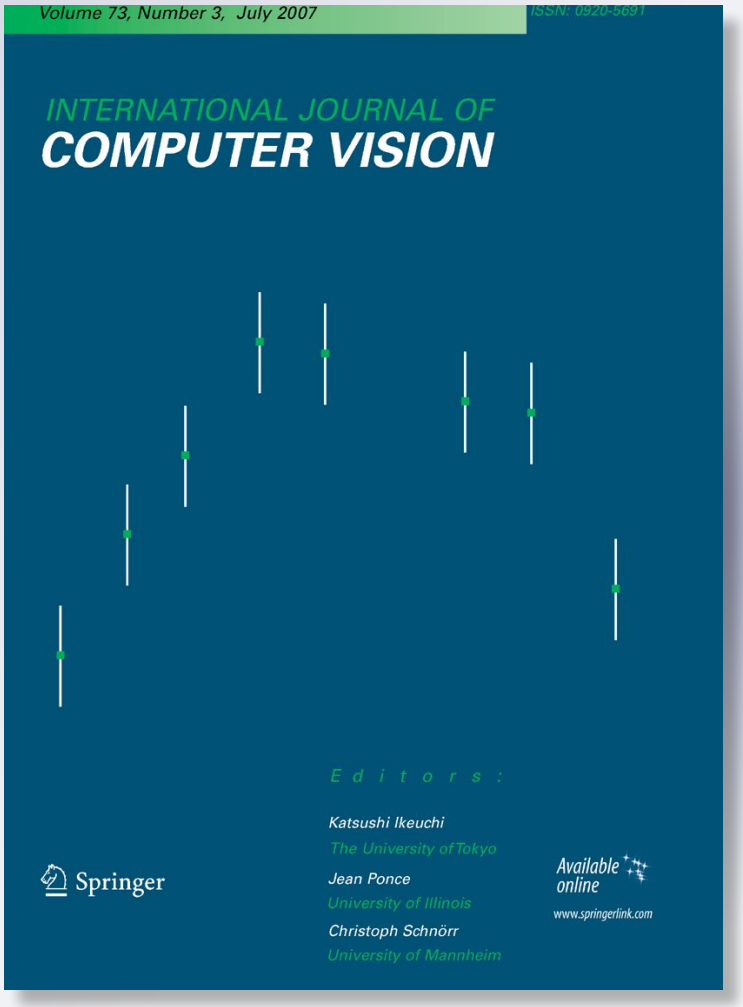
Geometric Image Parsing in Man-Made Environments

Elena Tretyak, Olga Barinova, Pushmeet Kohli & Victor Lempitsky

International Journal of Computer Vision

ISSN 0920-5691
Volume 97
Number 3

Int J Comput Vis (2012) 97:305-321
DOI 10.1007/s11263-011-0488-1



Springer

Int J Comput Vis (2012) 97:305–321
DOI 10.1007/s11263-011-0488-1

Geometric Image Parsing in Man-Made Environments

Elena Tretyak · Olga Barinova · Pushmeet Kohli · Victor Lempitsky

Received: 3 March 2011 / Accepted: 29 July 2011 / Published online: 8 September 2011

Abstract We present a new optimization based parsing framework for the geometric analysis of a single image coming from a man-made environment. This framework models the scene as a composition of geometric primitives spanning different layers from low level (edges) through mid-level (lines segments, lines and vanishing points) to high level (the zenith and the horizon). The inference in such a model thus jointly and simultaneously estimates (a) the grouping of edges into the line segments, (b) the grouping of line segments into the straight lines, (c) the grouping of lines into parallel families, and (d) the positioning of the horizon and the zenith in the image. Such a unified treatment means that the uncertainty information propagates between the layers of the model. This is in contrast to most previous approaches to the same problem, which either ignore the middle levels (line segments or lines) all together, or use the bottom-up step-by-step pipeline.

For the evaluation, we consider a publicly available York Urban dataset of "Manhattan" scenes, and also introduce a new, harder dataset of 103 urban outdoor images containing many non-Manhattan scenes. The comparative evaluation for the horizon estimation task demonstrate higher accuracy and robustness attained by our method when compared to the current state-of-the-art approaches.

Keywords Geometry estimation · Scene understanding · Man-made environment · Vanishing points estimation

1 Introduction

Recent years have seen a growing interest in the geometric analysis of a scene based on as little as a single image of this scene. Often the image of interest comes from a man-made environment, e.g. when the image is taken indoors or on a city street. In this case, the image is highly likely to contain a certain number of straight lines, which can be identified in the edgemap of the image, and which often can be further grouped into parallel families. The presence of such lines and their parallelism are known to be valuable cues for the geometric analysis.

When a family of parallel lines is projected on the image, their projections are known to intersect in a single point in the image plane called *vanishing point*. The vanishing point uniquely characterizes the 3D direction of those lines (given the camera). When 3D directions of several families are coplanar, the respective vanishing points belong to the same line. Such situation occurs frequently for man-made environments, as there often exist several families with different horizontal directions. In this case, the line containing their vanishing points is called the *horizon*. Most of the remaining lines of the scene are typically vertical. As such,

Tretyak Elena, Barinova Olga and Victor Lempitsky are supported by Microsoft Research programs in Russia. Victor Lempitsky is also supported by EU under ERC grant VisRec no. 228180.

E. Tretyak (✉) · O. Barinova
Lomonosov Moscow State University, Moscow, Russia
e-mail: tretiak.elena@gmail.com

O. Barinova
e-mail: olga.barinova@gmail.com

P. Kohli
Microsoft Research Cambridge, Cambridge, UK
e-mail: pkohli@microsoft.com

V. Lempitsky
University of Oxford, Oxford, UK
e-mail: victorlempitsky@gmail.com

they are parallel to each other and their projections intersect in the vanishing point called the *zenith*.[1]

The environments where horizontal lines fall into two orthogonal families, are known as "Manhattan" worlds. A considerable number of previous works investigated the Manhattan case, and the particular simplifications that it brings to the geometric analysis. The parsing framework suggested in this work may be adapted to the Manhattan case, however our work focuses on the non-Manhattan case, assuming the presence of the horizon and the zenith but not the two orthogonal horizontal directions. Surprisingly, very few previous works have paid attention to such scenario (most notably Schindler and Dellaert 2004), although we would argue that such assumptions about the scene strike a good balance between the generality and the robustness of the estimation.

In general, several computer vision and image processing tasks can benefit from the ability to extract the geometric information from a single image. E.g. the knowledge about the location of the horizon may be used to rectify the user photograph with inclined horizon, to facilitate the dense single-view reconstruction and "auto pop-up" (Hoiem et al. 2005a, 2005b); this knowledge may also greatly improve semantic segmentation, scene understanding, and object detection (Hoiem et al. 2008) as well as video stabilization (Duric and Rosenfeld 1996). Other geometric primitives can also be useful for different computer vision tasks. E.g. line segments, their grouping into parallel families and corresponding vanishing points are actively used for 3D structure analysis of indoor scenes: in Yu et al. (2008) this information is used for the extraction of depth-ordered planes. In Lee et al. (2009), Flint et al. (2010) line segments and vanishing points are used for 3D structure recovery, i.e. finding corners, walls, ceiling and floor. Also Hedau et al. (2009, 2010) and Lee et al. (2010) use these geometric primitives to estimate the box layout of the room, where the walls the floor and the ceiling are found and then further employed to solve more difficult tasks such as localization of objects and their fitting into cuboids. The abundance of applications thus motivates the research into better method of geometric analysis of single images leading to more accurate and robust algorithms.

1.1 Related Work

Conceptually, the process of line-based geometric analysis of a single image is well investigated, and typically involves several bottom-up steps. The process is generally initialized with the edge map of an image computed with some edge detector (a standard Canny detector is used in this work). Then, the bottom-up pipeline (McLean and Kotturi 1995; Tuytelaars et al. 1998; Cipolla et al. 1999; Antone and Teller 2000; Almansa et al. 2003; Aguilera et al. 2005; Wildenauer and Vincze 2007; Tardif 2009) involves grouping edges into lines, grouping lines into line families and finding the respective vanishing points, and, finally, fitting the horizon and the zenith or the Manhattan directions, depending on the assumptions about the world.

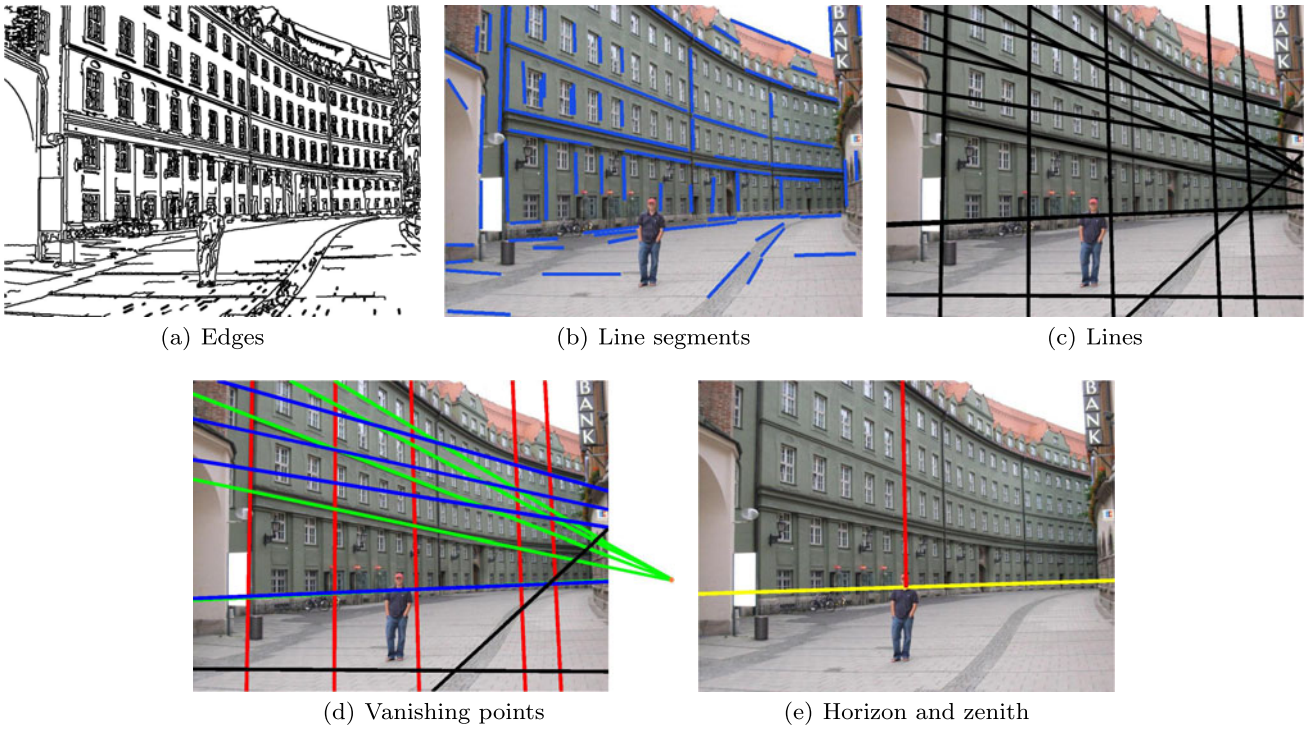
The problem with the step-by-step approach is, however, that neither of the steps can be performed with 100% accuracy and reliability. As the edge maps are always noisy and contaminated with spurious edge pixels not coming from straight lines, the line detection step would miss some of the straight lines and, even worse, detect some spurious lines that do not exist in the scene. Due to these errors, the parallel line grouping step would often group together lines from different families or create groups containing spurious lines (leading to spurious vanishing points) or split actual line families into several (reducing the accuracy of the respective vanishing point estimation). Finally, given an imperfect set of vanishing point, contaminated with outliers, horizon and zenith estimation may lead to gross errors. Also step by steps methods fail to carry over the uncertainty associated with estimates from the previous steps to the later steps in a principled fashion.

Previous works address the challenges associated with each step through several classes of techniques, including robust statistical inference (Collins and Weiss 1990), clustering (McLean and Kotturi 1995; Antone and Teller 2000; Kosecká and Zhang 2002), the RANSAC algorithm (Aguilera et al. 2005; Schaffalitzky and Zisserman 2000), various kinds of Hough transforms (Tuytelaars et al. 1998; Antone and Teller 2000; Almansa et al. 2003), stochastic model fitting (Rother 2000; Tardif 2009) as well as seeking user supervision (Cipolla et al. 1999). While different approaches possess different strengths and weaknesses, neither results in perfect accuracy and robustness, leading to the accumulation of errors towards higher stages of the pipeline.

A group of methods (Coughlan and Yuille 1999; Deutscher et al. 2002; Schindler and Dellaert 2004; Denis et al. 2008) goes beyond this pipeline paradigm, as they bypass the line extraction step altogether and directly fit the low-parametric high-level model of the frame (the Manhattan frame (Coughlan and Yuille 1999; Deutscher et al. 2002) or a set of Manhattan frames (Schindler and Dellaert 2004)) to the low-level edge map or even to the dense set of image gradients. The joint optimization nature of these methods is similar to our philosophy. However, the simplicity of the model and lack of the edges-to-lines grouping stage limits the accuracy and robustness of their approach as compared to a well-engineered full pipeline approach such as Tardif (2009).

In our preceding conference paper (Barinova et al. 2010b) the geometric parsing was performed by simultane-

[1] Strictly speaking, when this vanishing point lies below the horizon, it should be called the *nadir*. For brevity, we use the term *zenith* in this case as well.



(a) Edges (b) Line segments (c) Lines

(d) Vanishing points (e) Horizon and zenith

Fig. 1 (Color online) Geometric primitives of different levels for an example "non-Manhattan" image. (**a**)—edge pixels, (**b**)—line segments, (**c**)—straight lines, (**d**)—lines are grouped in parallel families (color indication used), (**e**)—the horizon and the zenith (shown with the direction in *red*). Our framework aims at joint estimation of primitives at the latter four levels given the former one (edge pixels)

ous line detection, parallel lines grouping, vanishing point detection, as well as the zenith and the horizon estimation. At the same time, several previous works (Tardif 2009; Kosecká and Zhang 2002) have demonstrated that it may be beneficial to base the parsing on finite-length line segments rather than on straight lines directly. Here, we demonstrate that line segment detection can be incorporated in our parsing framework in a form of an additional layer that groups edge pixels into finite-level line segments. Such line segments are then grouped into straight lines. The experimental comparison with the model from Barinova et al. (2010b) in Sect. 4 demonstrates that incorporating such an additional layer indeed improves the parsing accuracy.

1.2 Overview of Our Method

In this work, we investigate the *geometric parsing* approach to the geometric analysis. By geometric parsing here, we understand the process, when the geometric elements at different levels of complexity (Fig. 1), as well as the intra-level grouping relations are explicitly recovered through the joint optimization process. Note, that the term *parsing* is used in a similar meaning in such works as (Tu et al. 2005), where semantic primitives of different levels (e.g. body parts, individual humans, crowd) as well as the intra-level grouping relations are recovered. In our case, the primitives at different levels are edge pixels, line segments, lines, horizontal vanishing points, the zenith and the horizon.

Our work thus differs from works that employ a single bottom-up pass, as the inference in our case is performed jointly, allowing the information from top levels resolve the ambiguities on the lower levels (and vice versa). Our work also differs from the works that bypass the line detection, as the lines in our method are detected explicitly. To the best of our knowledge, the method presented here is the first that integrates line segment and line detection, vanishing point location, and higher-level geometric estimation (the horizon and the zenith in our case) in a single optimization framework.

There are several design choices and assumptions in our model that are motivated by the applicability and tractability. Firstly, unlike the majority of previous works, we do not make a Manhattan-world assumption. Instead, we consider a less-restrictive non-Manhattan scenario similar to the "Atlanta world" of Schindler and Dellaert (2004) that will be detailed below in Sect. 2. Regarding the camera parameters, we assume that the principal point is known (if unknown, we assumed it to be in the center of the frame); we also assume that pixels are square. This assumption holds approximately for the vast majority of cameras in real life, and it makes the inference in our model much easier. We also do

York Urban dataset (Denis et al. 2008) The new "Eurasian cities" dataset

Fig. 2 While the York Urban dataset (Denis et al. 2008) contains images of "Manhattan" worlds, our framework uses less restrictive scene assumptions that are met by non-Manhattan images in the new dataset that we introduce. Our framework is evaluated on both datasets

not model radial distortion explicitly, which is perhaps a bigger shortcoming of our model, although the robust nature of our algorithm means that considerable distortion might be tolerated without explicit modeling. Finally, we assume the focal length unknown. Theoretically, locations of the horizon and the zenith allow to estimate the focal length of the camera directly from the results of the parsing, however the accuracy of such estimation is hindered by the degeneracy that occurs when the horizon passes near the principal point, which in practice happens very often.

In a sequel, we detail our energy model in Sect. 2, and discuss the optimization procedure in Sect. 3. We then perform the experimental validation on two datasets (Fig. 2). The first one is the York Urban dataset presented in Denis et al. (2008), where several approaches were benchmarked. This dataset has been recently also used for the evaluation in Tardif (2009), where improved results have been reported. The second dataset was collected by ourselves and, unlike Urban, contains a lot of more challenging non-Manhattan outdoor scenes. The experimental comparison in Sect. 4 demonstrates the competitiveness of the parsing approach.

2 The Model for Geometric Parsing

We now explain the energy model of the world within our method. We assume an image to be defined by the set of its edge pixels. The main assumptions about the world are (a) that a considerable part of edge pixels may be explained by a set of line segments, (b) that a considerable part of those line segments may be explained by a set of lines (c) that a considerable part of those lines fall into several parallel line families. It is further assumed that (d) one of these families is a set of vertical (in 3D) lines converging (in the image plane) to the *zenith* and (e) all other families consist of horizontal (in 3D) lines converging (in the image plane) to a set of *horizontal* vanishing points, that all lie close to a single line in the image plane known as the *horizon*. The model thus encompasses the edge pixels, the line segments, the lines, the zenith, and the horizontal vanishing points, as well as the grouping relations of edge pixels into line segments, line segments into lines as well as lines into parallel families.

2.1 Energy Formulation

We now introduce the notation and the energy model. The edge pixels are denoted $\mathbf{p} = \{p_i\}_{i=1..P}$. The line segments and lines present in the scene are denoted $\mathbf{s} = \{s_i\}_{i=1..S}$ and $\mathbf{l} = \{l_i\}_{i=1..L}$ respectively. As the model involves grouping of lines into parallel families, we denote with z the vanishing point of the vertical line family (the zenith) and with $\mathbf{h} = \{h_i\}_{i=1..H}$ the set of vanishing points of the horizontal families. The points $h_1, h_2, \ldots, h_H$ thus have to lie close to a line in the image plane (we will refer to this fact as the *horizon constraint*).

The energy function in our model consists of five parts. The first three parts describe the connection between different geometric primitives. The first part includes edges and line segments, the second—line segments and lines, the third—lines and vanishing points. The fourth part is responsible for the *horizon constraint*, and the fifth imposes MDL (Minimum Description Length)-like prior. Now we sequentially describe each part.

The first three parts include the individual energy terms corresponding to the (pseudo-)likelihood of each edge pixel, each line segment and each line.

The edge pixel energy term is defined as:

$$E_{edge}(p|\mathbf{s}) = \min\Big(\theta_{bg}, \min_{i=1..S} \theta_{dist} \cdot d(p, s_i) + \theta_{grad} \cdot d_{angle}(p, s_i)\Big), \quad (1)$$

where $d(p, s_i)$ denotes the Euclidean distance in the image plane between the pixel p and the line segment s_i (the minimum distance between the edge pixel p and the line segment s_i), $d_{angle}(p, s_i)$ denotes the angular difference between the local edge direction at pixel p and the direction of the line

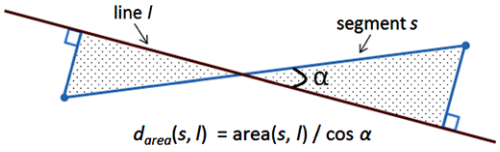


Fig. 3 Distance between line segment s and line l

segment s_i, θ_{bg} is the constant, corresponding to the likelihood of the background clutter, and θ_{dist} and θ_{grad} are the constants corresponding to the spread of edge pixels generated by a particular line segment around that line segment. Thus, the energy term for an edge pixel p is small if this edge pixel is well explained by some line segment from the set $\mathbf{s}$ and is large otherwise. The largest possible value is θ_{bg}, which corresponds to an edge pixel generated by the background clutter.

The line segment energy term is defined as:

$$E_{segment}(s|\mathbf{l}) = \min\Big(\mu_{bg} \cdot length(s), \min_{i=1..L} \mu_{dist} \cdot d_{area}(s, l_i)\Big), \quad (2)$$

where $d_{area}(s, l_i)$ denotes the distance in the image plane between the segment s and the line l_i, defined as the area of the figure between the line and the segment divided by the cosine of the angle $d_{angle}(s, l_i)$ between the line and the line segment (Fig. 3), in fact, this formula describes the value of the integral of the Euclidean distance function from line segment to line along the line segment; μ_{bg} is the constant, corresponding to the likelihood of the background clutter, and μ_{dist} is the constant corresponding to the spread of line segments generated by a particular line around that line. Thus, the energy term for a line segment s is small if this segment is well explained by some line from the set $\mathbf{l}$ and is large otherwise. The largest possible value is $\mu_{bg} \cdot length(s)$, which corresponds to a line segment which doesn't correspond to any lines. Such description of the energy term gives an explanation of each line segment as a set of edge pixels, that form this line segment.

The line energy terms are defined as

$$E_{line}(l|\mathbf{h}, z) = \min\Big(\eta_{bg}, \min_{i=1..H}\big(\eta_{dist} \cdot \phi(l, h_i), \eta_{dist} \cdot \phi(l, z)\big)\Big), \quad (3)$$

where ϕ denotes the distance on the Gaussian sphere (Barnard 1983) between the projection of the line l and projection of the respective vanishing point (h_i or z). η_{bg} is the constant, corresponding to the likelihood of lines that are neither horizontal nor vertical, and η_{dist} is the constant, corresponding to the spread of lines in their families around the respective vanishing points. Thus, the energy term for a line l is small if this line is well explained by (i.e. passes close to) a vanishing point from the set $\mathbf{h} \cup \{z\}$ and is large otherwise. The largest possible value is η_{bg}, which corresponds to a line that is neither vertical nor horizontal.

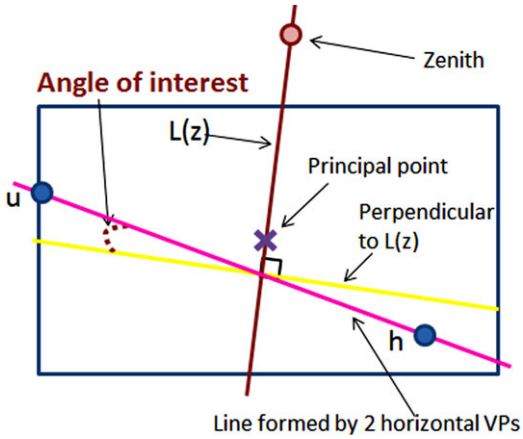


Fig. 4 Explanation of the *horizon constraint*. Here u and h are two horizontal vanishing points

According to horizontal constraint introduced above all vanishing points except the zenith have to lie close to a line in the image plane. How can we enforce this constraint? Should a separate variable for the position of the horizon be introduced? It turns out (Beardsley and Murray 1992) that under our assumption about internal camera parameters (square pixels and known principal point) this is not necessary. Under these assumptions, the horizon is perpendicular to the radius vector between the line $L(z)$ connecting the zenith and the principal point, and we enforce this perpendicularity with the following energy term:

$$E_{horizon}(u, h|z) = \kappa_{hor} \cdot \tan\psi\big(u - h, L(z)\big). \quad (4)$$

where ψ is the absolute angle between the vector $u - h$ and a perpendicular to $L(z)$, and κ_{hor} is a constant. The graphical explanation is given on Fig. 4. The tan in (4) was chosen because it imposes significant penalty (upto $+\infty$) on strong non-orthogonality between the horizon and $L(z)$.

The last part of the model describes the MDL prior

$$E_{prior}(\mathbf{s}, \mathbf{l}, \mathbf{h}) = \lambda_{line}|\mathbf{l}| + \lambda_{vp}|\mathbf{h}| + \lambda_{segment} \sum_{i=1..S} length(s_i). \quad (5)$$

This term penalizes the number of line segments $|\mathbf{s}| = S$, the number of lines $|l| = L$ and the number of horizontal vanishing points $|\mathbf{h}| = H$, thus favouring simpler explanations of the scene ($\lambda_{segment}$, λ_{line} and λ_{vp} are the constants regulating the strength of this prior). For the line segments we also multiply the term corresponding to each line segment by its length. Such formulation assigns bigger penalty for longer line segments in order to balance energy on different layers of the model: a long line segment can be included

to the model only if there are enough edge pixels that correspond to this segment (the segment is the closest for these edge pixels).

The final energy is thus defined as:

$$\begin{aligned} &E_{total}(\mathbf{s},\mathbf{l},\mathbf{h},z|\mathbf{p}) \\ &= \sum_{i=1..P} E_{edge}(p_i|\mathbf{s}) \\ &\quad + \sum_{i=1..S} E_{segment}(s_i|\mathbf{l}) + \sum_{i=1..L} E_{line}(l_i|\mathbf{h},z) \\ &\quad + \sum_{1\le i<j\le H} E_{horizon}(h_i,h_j|z) + E_{prior}(\mathbf{s},\mathbf{l},\mathbf{h}). \end{aligned} \tag{6}$$

The energy (6) thus ties together the different-level components in the image of a non-Manhattan environment, and the line-based parsing of such an image may be performed through the minimization of (6).

2.2 Interpretation of the Model

Some of the components of our model may be easily formulated with the language of probabilities. Thus, three bottom layers of our model related to grouping edges into line segments, grouping line segments into lines and grouping lines into parallel families allow probabilistic interpretation. The energy terms corresponding to each of these layers E_{edge}, $E_{segment}$ and E_{line} can be viewed as log-posterior of the probabilistic model derived in Barinova et al. (2010a).

It is unclear, however, if the $E_{horizon}$ term in (4) admits a probabilistic interpretation, as it apparently involves some overcounting of the orthogonality cues. In practice, this non-probabilistic nature does not present a problem, as we train our model discriminatively by tuning the constants θ_{bg}, θ_{dist}, θ_{grad}, μ_{bg}, μ_{dist}, η_{bg}, η_{dist}, κ_{hor}, $\lambda_{segment}$, λ_{line}, λ_{vp} on the hold-out validation set.

Some of existing vanishing points estimation pipelines allow interpretation in terms of energy minimization similarly to our work. For example the method (Boulanger et al. 2006) can be viewed as an optimization of the energy function that consists of a sum of distances from lines to vanishing points. This energy function roughly corresponds to the term E_{line} of our energy. However there is no exact correspondence between the energy functions implicitly minimized by existing approaches and individual terms of our model because to the best of our knowledge this work is the first one to use both line segments and lines. Another important difference between our energy and the energy of other pipelines for vanishing points estimation is the soft constraint on the horizon (last term $E_{horizon}$). This term is in some sense related to the constraint used under the Manhattan World assumption (Coughlan and Yuille 1999) but allows handling a richer class of the scenes.

3 Inference

The minimization of (6) is a hard computation problem that necessitates the use of approximations. One possible way would be to minimize it greedily in a layer-by-layer fashion, first choosing the set of lines segments given the edges, then choosing the set of lines given line segments, then choosing the set of vanishing points given lines, then fitting the horizon and the zenith into the chosen lines. Such an approach would correspond to the traditional bottom-up pipeline from previous methods. Its results might be improved with reiteration of the process through the EM-algorithm, although in practice that suffers from the local minima problem and often gets stuck close to the initial greedy approximation.

3.1 Discretization of the Model

A different approach taken in this work is to derive a *discrete approximation* to the original energy that is easier to minimize. To achieve that, we do three steps of the bottom-up pipeline, namely line segment detection, line detection and vanishing point detection, with very low acceptance thresholds, ensuring that an extensive set of X line segments $\hat{s}_1..\hat{s}_X$, extensive set of Y lines $\hat{l}_1..\hat{l}_Y$ and an extensive set of V vanishing points $\hat{h}_1..\hat{h}_V$ are detected. On practice, one may use any approach that detects line segments, lines and any approach that detect a set of vanishing points. We detail our choices in the experimental section (see also Fig. 7).

The task of the approximate minimization of (6) may then be reduced to the minimization of the energy of discrete variables $\mathbf{x} = \{x_i\}_{i=1..X}$, $\mathbf{y} = \{y_i\}_{i=1..Y}$, $\mathbf{v} = \{v_i\}_{i=1..V}$, and z. Here, each variable x_i is binary and decides whether a candidate line segment $\hat{s}_i$ is present ($x_i = 1$) or absent ($x_i = 0$) in the image. Similarly, each variable y_i is binary and decides whether a candidate line $\hat{l}_i$ is present ($y_i = 1$) or absent ($y_i = 0$) in the image and each variable v_i is binary and decides whether a candidate vanishing point $\hat{h}_i$ is a *horizontal* vanishing point that is present ($v_i = 1$) or absent ($v_i = 0$) in the image. Finally, the variable z is, as defined above, a 2D point in the image plane corresponding to the zenith. The set of its possible locations is however restricted to discrete set of candidate vanishing points. For computational efficiency, we may further prune the set of possible locations for z by removing candidate vanishing points that correspond to the horizon inclinations of more than 7.5 degrees. This can be regarded as an additional hard prior on z in our original energy.

The discrete approximation to the energy (6) is then defined by the requirement:

$$\begin{aligned} &E_{discrete}(\mathbf{x},\mathbf{y},\mathbf{v},z|\mathbf{p}) \\ &\quad \equiv E_{total}\big(\{\hat{s}_i\}_{i:x_i=1},\{\hat{l}_j\}_{j:y_j=1},\{\hat{h}_k\}_{k:v_k=1},z|\mathbf{p}\big). \end{aligned} \tag{7}$$

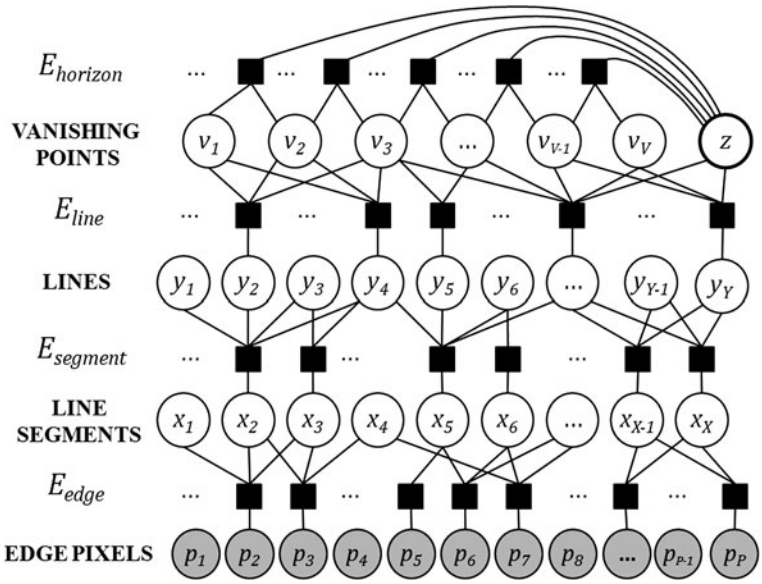


Fig. 5 The graphical model for the discrete approximation of the energy (6). The variables $x_1 \dots x_X$, $y_1 \dots y_Y$ and $v_1 \dots v_V$ are binary and correspond to the existence or the absence of candidate segments, lines and horizontal vanishing points. z stands for the location of the zenith and takes the value in a precomputed discrete set of 2D points in the image plane. The unary cliques corresponding to $x_1 \dots x_X$, $y_1 \dots y_Y$ and $v_1 \dots v_V$ are omitted for clarity. The *shaded nodes* (edge pixels) are observed both during training and at test time. Please, see text for more details

In other words, the discrete energy is defined as the continuous energy of the appropriate subsets of candidate line segments, lines and vanishing points. So, all the candidates, for which the corresponding value of binary variable is zero, are not included into the discrete energy.

In more detail, the discrete energy defined in (7) can be written as:

$$\begin{aligned} &E_{discrete}(\mathbf{x}, \mathbf{y}, \mathbf{v}, z|\mathbf{p}) \\ &= \sum_{i=1..P} E_{edge}\big(p_i|\{\hat{s}_j\}_{j:x_j=1}\big) \\ &\quad + \sum_{i=1..X} x_i \cdot E_{segment}\big(\hat{s}_i|\{\hat{l}_j\}_{j:y_j=1}\big) \\ &\quad + \sum_{i=1..Y} y_i \cdot E_{line}\big(\hat{l}_i|\{\hat{h}_k\}_{k:v_k=1}, z\big) \\ &\quad + \sum_{1\le i<j\le V} v_i \cdot v_j \cdot E_{horizon}(\hat{h}_i, \hat{h}_j|z) \\ &\quad + \sum_{i=1..X} \lambda_{segment} \cdot x_i \cdot length(\hat{s}_i) \\ &\quad + \sum_{i=1..Y} \lambda_{line} \cdot y_i + \sum_{i=1..V} \lambda_{vp} \cdot v_i. \end{aligned} \tag{8}$$

Note, that the different-level candidates in our model are not treated equally. Each edge pixel can exist in two states: it either belongs to background clutter, then the value of θ_{bg} is added to the energy, or it belongs to a line segment candidate, then the distance to the closest line segment is added to the energy. As edge pixels represent the observed data, there are no binary variables associated with them and all the edge pixels are always switched on. Each line segment candidate can exist in three states: (1) the corresponding binary variable x_i equals 0, then the line segment candidate is switched off and it does not affect the energy, (2) the corresponding binary variable x_i equals 1 and the line segment belongs to the background, then the value of $\mu_{bg} \cdot length(\hat{s}_i)$ is added to the energy (3) the corresponding binary variable x_i equals 1 and the line segment belongs to a line candidate, then the distance to this line is added to the energy. Similarly, each line exists in one of the three states: (1) the corresponding binary variable y_i equals 0, then the line candidate is switched off and it does not affect the energy, (2) the corresponding binary variable y_i equals 1 and the line belongs to background, then the value of η_{bg} is added to the energy (3) the corresponding binary variable y_i equals 1 and the line belongs to a vanishing point, then the distance to this vanishing point is added to the energy. Each vanishing point candidate can be in two states: (1) the value of v_i equals 0, then it does not affect the energy (2) the value of v_i equals 1, then this vanishing point makes contribution into the horizon constraint term. Also for each switched on line segment, line and vanishing point we add an additional term to the energy (MDL prior).

Although, switched off candidates are not penalized in our model, the minimum of the energy is not achieved, when all values of variables $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{v}$ are equal to 0. In that case the energy value is equal to

$$\sum_{i=1..P} E_{edge}\big(p_i|\{\hat{s}_j\}_{j:x_j=1}\big) = \sum_{i=1..P} \theta_{bg}.$$

That value obviously can be minimized, if a set of appropriate line segments, lines and vanishing points is chosen.

The factor graph for the formula (8) is shown in Fig. 5. Note, that due to the truncation effect of the constants θ_{bg} and θ_{dist} in the definition of E_{edge}, the connections between the E_{edge} factors and the line segment variables as well as

between the $E_{segment}$ factors and the lines variables and between the E_{line} factors and the vanishing points variables are sparse. Each E_{edge} factor is connected only to the line segments that pass nearby that edge pixel and, likewise, each $E_{segment}$ factor is connected to the line variables that lie near that line and each E_{line} factor is connected to the vanishing point variables that lie near that line.

Since the values of $\mathbf{p}$ are observed, very big efficiency gains may be easily obtained by merging (summing up) the E_{edge} factors that are connected to the same (or nested) sets of line segment variables. Since E_{edge} terms constitute the vast majority of terms in (8), this trick dramatically reduces the number of energy terms in the model. It permits us to use quite a simple and brute-force optimization scheme, while still allowing short optimization runtime of several seconds for a typical photograph.

In more detail, we exhaustively search through the zenith candidate set (which typically includes less then a dozen of candidates). Given a fixed z, we then perform optimization over the binary variables x, y and v through the modified simulated annealing algorithm with the randomized node visiting order. In the beginning the temperature is set to the value of the parameter κ_{hor}. We iteratively try to randomly change the values of variables v, y and x. If the difference between the new and the old energy values is not bigger than the current temperature, we accept new values of variables. As usual with simulated annealing, after each iteration the temperature is multiplied by a fixed parameter, which is responsible for the convergence speed. The process is repeated until the energy difference between two last iterations falls below the threshold (the value of the temperature becomes too small). During our experiments 300 steps were required.

3.2 Manhattan Directions Detection

For some tasks it is useful to detect three orthogonal directions on an image. And although our model was designed to solve more general problem of detecting an arbitrary number of vanishing points, it can be easily modified to solve the problem of finding three orthogonal vanishing points.

At first, during the inference we can fix the number of horizontal points (to one or two). This will lead to detection of the three points most supported by data, without any constraint on their orthogonality. Furthermore, we can extend our horizon constraint on vanishing points to a new constraint on the orthogonality of three vanishing points, which is represented on Fig. 6. In order to fulfill this constraint we should additionally penalize the angle between the line, connecting one vanishing point h and the zenith, and perpendicular to the line $L(u)$, connecting principal point and the other vanishing point u, and the angle defined in the same way for the second horizontal vanishing point. In more details our new orthogonal constraint term can be written as:

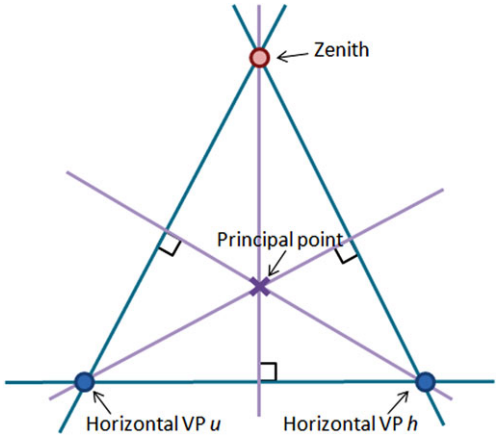


Fig. 6 Constraint on three orthogonal vanishing points: principal point should be the orthocenter of the triangle, formed by these points (in the case when image skew is zero and the aspect ratio is one (Hartley and Zisserman 2003))

$$
\begin{aligned}
&E_{ortho}(u,h|z)\\
&\quad=\kappa_{hor}\cdot\big(\tan\psi\big(u-h,L(z)\big)\\
&\qquad+\tan\psi\big(h-z,L(u)\big)+\tan\psi\big(u-z,L(h)\big)\big),
\end{aligned}
\tag{9}
$$

which should be included into the energy function instead of the $E_{horizon}$ term.

4 Experiments

4.1 Technical Details

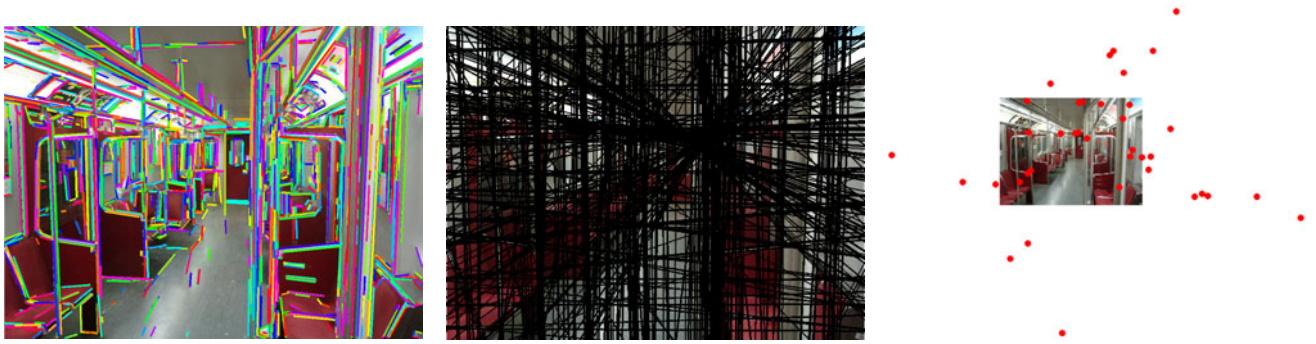
In our experiments we used the following strategy for choosing candidate line segments, candidate lines and candidate vanishing points.

Candidate line segments were calculated using Line Segment Detector algorithm (Morel et al. 2008). To get an excessive number of line segments candidates, we used a pyramid of images with three different scales and detected line segments for each scale. In total we usually obtained around 2000 line segment candidates for each pyramid of images.

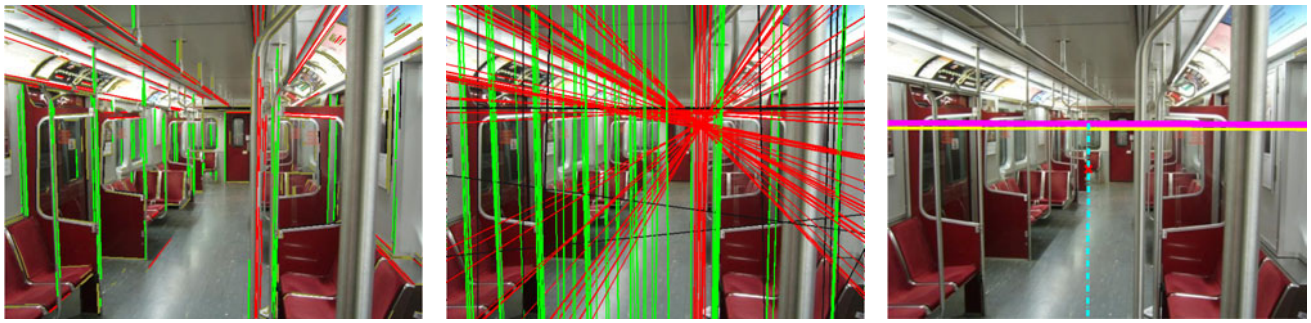
For the line detection the probabilistic version of Hough transform (Barinova et al. 2010a) was used. As Barinova et al. (2010a) provides the confidence measure for each detected line, we fixed the number of candidates to 500 and for each image took 500 lines with the highest confidence. Figure 7 gives an example of what the candidate set typically looks like.

The candidates for vanishing points were chosen using the J-linkage procedure, described in Tardif (2009). This method is based on random sampling, so we ran it several times starting from different random initializations. Usually we got from 50 to 100 candidates for vanishing points. The

(a) Input image (b) Edge pixels



(c) Candidate line segments (d) Candidate lines (e) Candidate vanishing points



(f) Result of the parsing(line segments) (g) Result of the parsing(lines) (h) Result of the parsing(horizon and zenith)

Fig. 7 (Color online) Sample image from the York Urban dataset: (**a**)—the input image, (**b**)—edge pixels, (**c**)—all candidate line segments superimposed, (**d**)—all candidate lines superimposed, (**e**)—candidate vanishing points (without vertical candidate vanishing points) (**f**), (**g**), (**h**)—the result of the parsing. On (**f**) and (**g**) coloring reflects grouping into parallel families. *Black lines* are lines, that are presented on the image, but do not belong to any vanishing point. Coloring of a line segment is the same as the coloring of the corresponding line. Line segments, which do not lie close to any lines are colored with *olive*. On (**h**) *pink* and *yellow thick lines* correspond to the found and the ground truth horizons respectively, *cyan line* shows the direction to zenith and the cross corresponds to the principal point position

set of candidate vanishing points is divided into two parts: the zenith candidates and the horizontal vanishing points candidates. Zenith candidates are chosen according to the two constraints: (1) the absolute value of the y-coordinate is bigger than the threshold thr_y (2) the deviation from the vertical line is not greater than the threshold thr_{ang}. Usually there are less than 10 zenith candidates, whereas all the other points are treated as horizontal vanishing points candidates. After performing the inference in our model we usually got from 2 to 5 vanishing points and groups of lines supporting each of them.

In the experiments on York Urban dataset we exploited the coordinates of principal point provided, in the experiments on the new dataset we assumed the principal point to lie in the center of the image frame.

The code was mostly implemented in C++ (compiled in MEX functions) and in Matlab. It can be downloaded from the project homepage.[2]

[2] http://graphics.cs.msu.ru/en/science/research/3dreconstruction/geometricparsing.

4.2 Training the Model

The parameters for our models were tuned on the hold-out validation set. Fully automatic learning of the parameters is quite difficult for two reasons. First, standard approaches for structural learning are not applicable to our energy. Second, with available ground truth data we can learn parameters to find a good horizon on the validation set, but even with a good estimation of the horizon the overall result can be poor, as the output also includes line segments, lines and vanishing points. In general learning would require ground truth labeling of all primitives in the training image dataset.

As each parameter has some intuitive meaning,it is possible to find a good initial guess of the parameters. Having an initial guess we can refine them using standard optimization methods. Below we describe this approach.

At first, the ratio between layers should be determined. For this the values of the terms E_{edge}, $E_{segment}$ and E_{line} should be adjusted accordingly. Parameters λ_{line}, λ_{vp} and $\lambda_{segment}$ help to control the number of corresponding primitives: the smaller are these parameters, the more primitives are found. The parameter κ_{hor} is responsible for the strength of the horizon constraint. Then the parameters inside the first three terms should be adjusted. For example, in the term E_{line} the ratio of the parameters $\eta_{bg}/\eta_{dist} = thr_{line}$ effectively constitutes a threshold for switching "on" a line. If the distance to the closest vanishing point for a given line is greater, than this threshold, then this line is treated as background clutter, otherwise it is labeled as belonging to the closest vanishing point. That means, the less this ratio thr_{line} is, the more lines will be classified as background clutter. Similar interpretation of the parameters is possible for the other two terms.

When a good initial estimate is found, the parameters can be further optimized using local optimization techniques. In this work the matlab implementation of the version of the Nelder-Mead simplex search algorithm was used (`fminsearch`) with the objective function being the area under curve statistics for the horizon estimation (see below). The optimization was performed over all parameters. Objectives based on other geometric primitives can be also used for parameters tuning, if some additional properties are required. In general, the parameters should be tuned according to particular dataset and particular purpose.

As the optimal values of parameters are determined by the geometric constraints of the perspective projection, the method should work well with the same values of parameters on the most pictures of man-made environments. The values of the parameters tuned using this approach are shown below.[3] Note that they are very close for both datasets. This fact confirms that similar values of parameters can be used for different pictures of man-made environments.

4.3 Datasets

Our approach is evaluated on two datasets (Fig. 2):

1. The *York Urban* dataset (Denis et al. 2008) contains 102 images of outdoor and indoor scenes taken within the same location with the same camera. Most of the scenes meet the Manhattan world assumption, as the lines available in the scene mostly fall into the three orthogonal families.
2. The *Eurasian cities*[4] dataset is a new set of 103 outdoor urban images. The images come from the cities of different cultures, hence with different line statistics. They were also taken with different cameras. The main difference of the dataset is the abundance of scenes that fit poorly the Manhattan assumption. During the annotation, we manually specified several most distinctive lines per each distinctive parallel line family in each image (with the interactive tool similar to that of Denis et al. (2008)). This allows to estimate the horizon with good accuracy and we use it as ground truth in the comparative evaluation. For each image we provided several families of parallel line segments, a vanishing point for each family (calculated using function introduced in Tardif (2009)) and a horizon (calculated using least square fit over horizontal vanishing points).

4.4 Competing Methods

We have compared our approach against the two previously published methods:

1. *The method of Tardif* (Tardif 2009) is a pipeline approach which reported the top performance on the York Urban dataset. For the experiments on the York Urban dataset we used the author code (with the exception of the EM process that was not published and that we reimplemented by carefully following the text of Tardif (2009)). For York Urban dataset in cases where more than 3 vanishing points were detected, we chose 3 most orthogonal of them as described in the paper (Tardif 2009). The coordinates of principal point provided by the authors of the dataset were used during orthogonalization. For the experiments in the Eurasian Cities we did not choose most orthogonal points because the dataset contains non-Manhattan scenes. Parameters of EM were chosen on the test set.

[3] Through the validation, the parameters for our method for York/Eurasian cities were set to: $\theta_{bg} = 6.82 \times 10^{-4}/6.82 \times 10^{-4}$, $\theta_{dist} = 5.4 \times 10^{-4}/5.4 \times 10^{-4}$, $\theta_{grad} = 5.4 \times 10^{-4}/5.4 \times 10^{-4}$, $\mu_{bg} = 4.1 \times 10^{-4}/4.1 \times 10^{-4}$, $\mu_{dist} = 4.1 \times 10^{-7}/4.1 \times 10^{-7}$, $\eta_{bg} = 1.2 \times 10^{-2}/9.6 \times 10^{-3}$, $\eta_{dist} = 0.56/0.65$, $\kappa_{hor} = 4.65/4.65$, $\lambda_{segment} = 7.0 \times 10^{-5}/2.3 \times 10^{-6}$, $\lambda_{line} = 4.5 \times 10^{-3}/3.5 \times 10^{-3}$, $\lambda_{vp} = 2.3 \times 10^{-2}/3.5 \times 10^{-2}$. All angular differences were measured in radians.

[4] Available at http://graphics.cs.msu.ru/files/tmp/EurasianCitiesBase.zip.

2. *The method of Kosecká and Zhang* (Kosecká and Zhang 2002) is an approach based on the EM-algorithm, alternating between the two stages: estimation of vanishing point coordinates given distribution of corresponding line segments and re-estimation of distribution of line segments according to positions of vanishing points. The process starts with clustering line segments according to their orientation which results in excessive number of clusters. During EM the clusters with close vanishing points are merged together. Also clusters that have little support are pruned. We took the code from implementation of the Automatic Photo Pop Up system (Hoiem et al. 2005a), which uses that method for vanishing points estimation. Parameters of the method were tuned on the test set.

On top of the edge detection and line segment detection steps that we discuss below, the baseline methods have one parameter to tune (the parameter of the EM algorithm, which specifies the minimum number of line segments in a cluster). In contrast, our method has 11 parameters, which gives it more flexibility compared to competitors, and potentially allows better adjusting to a particular dataset. So we adjusted the parameters for our method on the hold-out validation set for each dataset. We gave the baseline methods some handicap by tuning their parameters on the test set, so we report their best performance over the range of parameters.

Importantly, to put all the methods on an equal footing, we made sure that all three algorithms are provided with the same Canny edge map (we used the parameters suggested by Tardif (2009)). Both baseline methods use line segments, so we use the line segments detection implementation by Tardif (2009) for both of them.

4.5 Horizon Estimation

After running each method we obtain the zenith, as well as a number of vanishing points corresponding to the parallel families of the line segments (for baseline methods) or lines (for our method). We use this information to estimate the position of the horizon in an image. The horizon is estimated in the same way for all methods. Thus, we restrict it to be perpendicular to the line connecting principal point and zenith. So the slope of horizon is given by zenith and we estimate only its position along the 1D axis. To do this last step, we perform the weighted least squares fit, where the weight of each detected horizontal vanishing point equals the number of corresponding lines (or line segments). In the case when no horizontal vanishing points or zenith were found, the horizon was drawn strongly horizontally in the middle of the image. In our approach the case when no zenith is found is impossible, as zenith is the part of our model. In contrast, in the other competing methods the horizontal vanishing points and the zenith are treated equally.

4.6 Accuracy Measure

While all the considered approaches essentially output both low-level and high-level primitives, comparing the accuracy of the low-level description of the scenes (e.g. set of lines) is problematic, as the ground truth available for the datasets do not provide full set of lines and line segments. Thus, if a line segment or a line or a vanishing point is present in the output that is missing in the ground truth, it is unclear whether this is due to the error of the algorithm or due to the incompleteness of the ground truth. Also different low-level geometric primitives are used in algorithms: some methods take into account line segments, other methods—only lines,

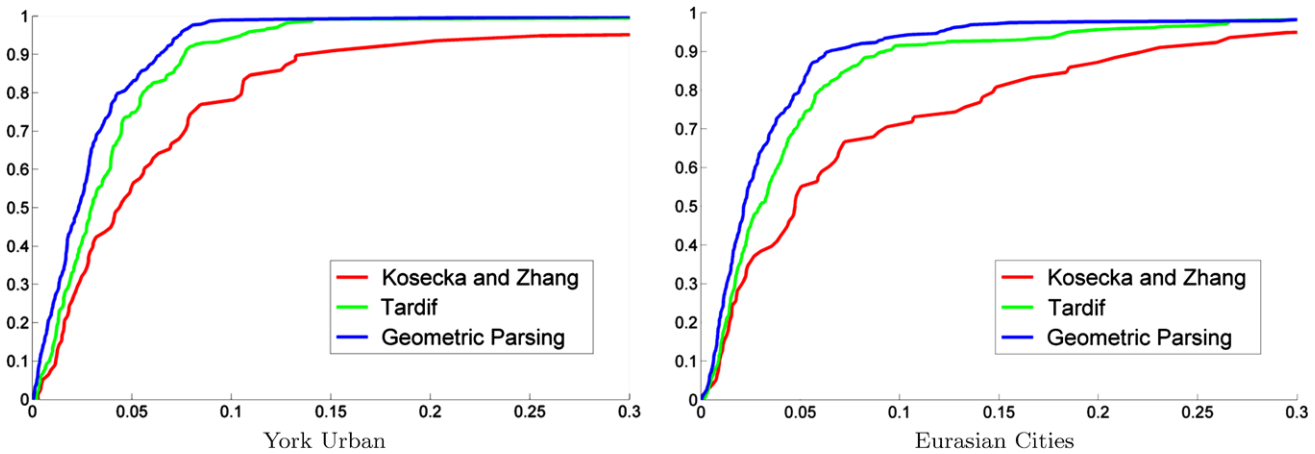


Fig. 8 The results of the comparison of the cumulative statistics for the accuracies of the proposed framework along with the methods of Tardif (2009) and Kosecká and Zhang (2002). The x-axis corresponds to the horizon estimation error measure (see text for more details). The y-axis corresponds to the share of the images in the test set that has the error less than the respective x value. In both cases, the proposed framework obtains higher accuracy than the competitors

that makes the comparison of these approaches more difficult.

We therefore focused on the accuracy of the horizon estimation. Assume that the horizon is given as a (linear) function $H(x)$ of a pixel x-coordinate. Assume that $H_0(x)$ and $H_1(x)$ are the ground truth and the estimated horizon. We define the estimation error as the maximum Euclidean distance between the lines $H_0(x)$ and $H_1(x)$ within the image domain ($0 < x <$ image width), divided by the image height. To represent the error over the dataset, we plot the share of the images with the error less than τ for each τ.

Different metrics were proposed in literature for Manhattan-World images. The first publically available dataset and metric were presented in Denis et al. (2008). The proposed metric calculates the average Manhattan-frame orientation estimation error. Further in Tardif (2009) two other metrics were proposed. The first one measures the consistency of the ground truth line segments with the estimated vanishing points and the second is the accuracy of the estimated focal length using the vanishing points. All these metrics imply the predefined number of vanishing points for each image or detailed marking of line segments. In contrast, in our work we avoid any constraints on the number of points and our method is able to find an arbitrary number of points. Our metric, which is based on the accuracy of the horizon estimation, is more suitable for more general, non-Manhattan case. On the other hand horizon-based metric has some drawbacks due to the ignorance of the lower-level primitives. For example, it adds no penalty if not all the vanishing points presented on the image are detected by the algorithm.

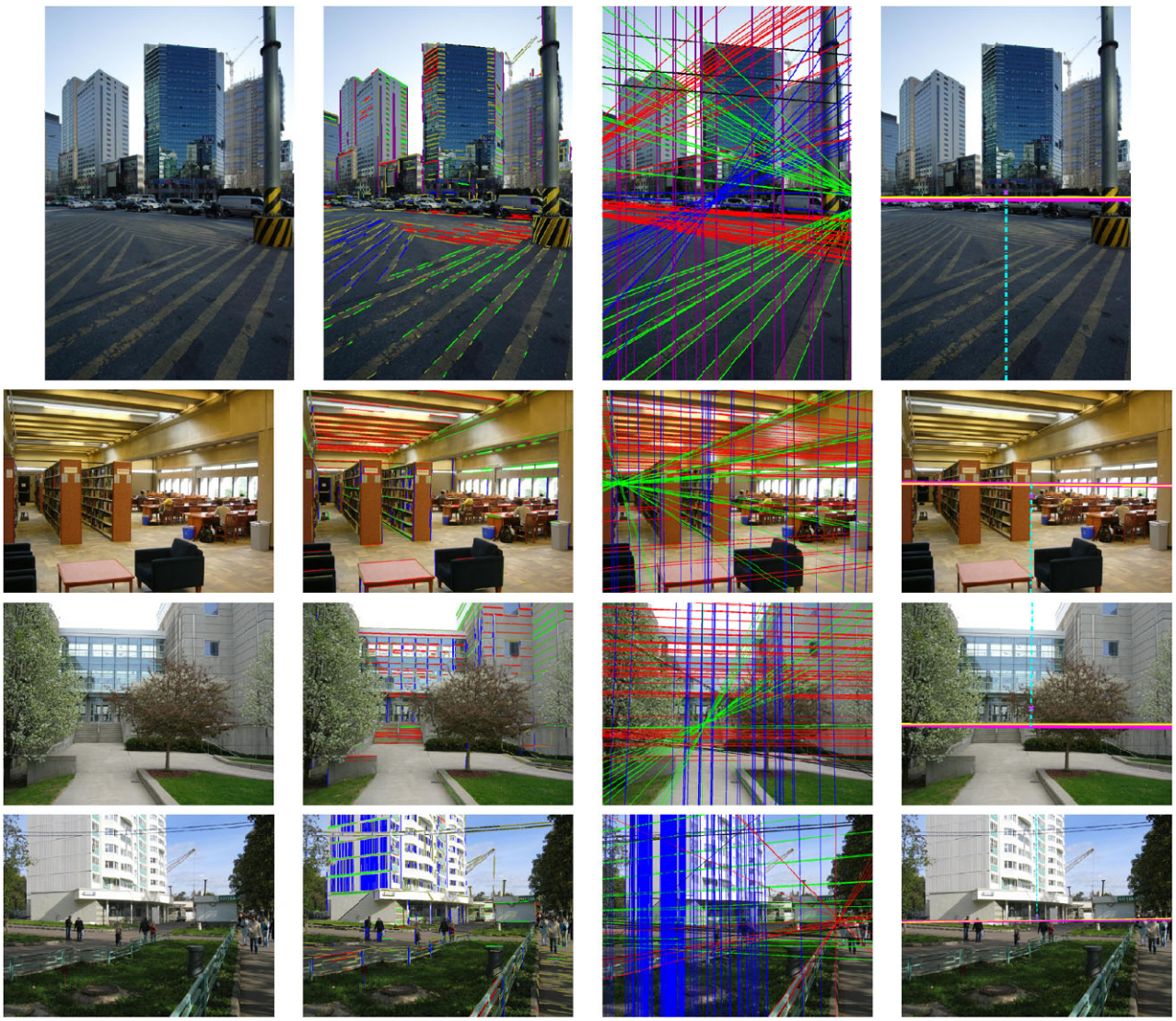
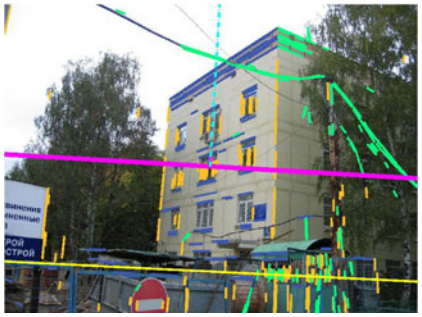


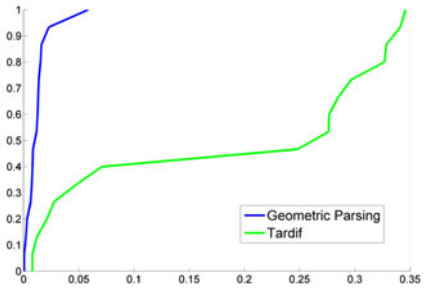
Fig. 9 (Color online) Sample results of the proposed framework from both datasets. For each we give the input image and the output of the parsing: line segments and lines, grouped into parallel families, zenith and horizon. The coloring is the same as in Fig. 7

(a) Input Image

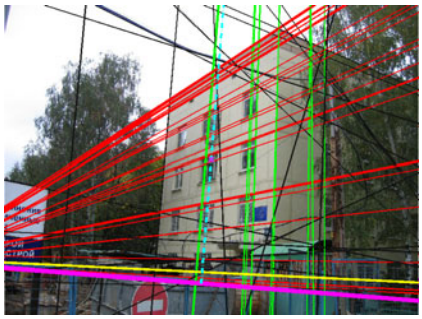


(b) Worst Result, Tardif

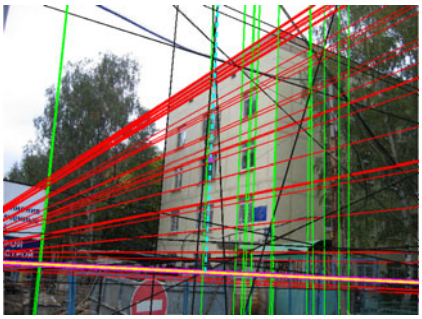
(c) Best Result, Tardif



(d) Cumulative histogram after 15 launches



(e) Worst Result, Geometric Parsing



(f) Best Result, Geometric Parsing

Fig. 10 (Color online) Comparison of our Geometric Parsing approach with the method of Tardif. The best and the worst results for both methods after 15 launches are shown in the figures (**b**)–(**c**) and (**e**)–(**f**) as well as the cumulative histogram of the horizon estimation error. Coloring of line segments and lines reflects grouping into parallel families. *Pink* and *yellow thick lines* correspond to the found and the ground truth horizons respectively, *cyan line* shows the direction to zenith and the cross corresponds to the principal point position

4.7 Results

Quantitative results are given in Fig. 8, while in Figs. 7 and 9, we present some qualitative examples from both datasets for our framework. On Fig. 13 some failure cases are shown. Note that we used the first 25 images of each dataset as a held-out set for the parameter validation of our models, and for the other competing methods test set was used for validation. During the validation, the area under curve statistics on the validation set was optimized. The accuracy measures in the plots in Fig. 8 thus reflect the best performance on the test set for methods of Tardif (2009) and Kosecká and Zhang (2002), and the performance of our models on the images not seen during the validation. As the method of Tardif (2009) and our approach are randomized, these methods were launched 5 times on each image on both datasets.

As can be seen, the method presented in the paper outperforms both competing methods on Eurasian Cities and York Urban datasets. The latter is all the more important, given the fact the stronger competing method of Tardif (2009) makes explicit use of the Manhattan assumption that is very appropriate for the York dataset, while our method worked with the more general non-Manhattan world model. At the same time, our current implementation is much slower than the competing methods (approximately half a minute per image vs. few seconds per image on a modern PC). The time for our method is dominated by the candidate (line segments, lines and VPs) generation and graph construction, and can be reduced significantly if less exhaustive number of candidates would be considered.

In Fig. 10 the comparison between Geometric Parsing approach and the method of Tardif (2009) on one of the images from our dataset is presented. For both methods the cumulative histogram after 15 launches is shown. As can be seen from this example, the method of Tardif gives quite unstable results. In the case of this image it usually wrongly clusters the line segments coming from wires into a parallel family. Our approach is also randomized due to several reasons: it uses the method of Tardif on the candidate calculation step, and also the simulated annealing algorithm used for the inference is randomized. But due to the usage of the model which incorporates the information about the geometry of the scene, the Geometric Parsing approach shows more stable performance.

Figure 11 presents a qualitative comparison of line segment detection performance. Our method provides better grouping of line segments into parallel families than Tardif's method and is better at distinguishing between line seg-

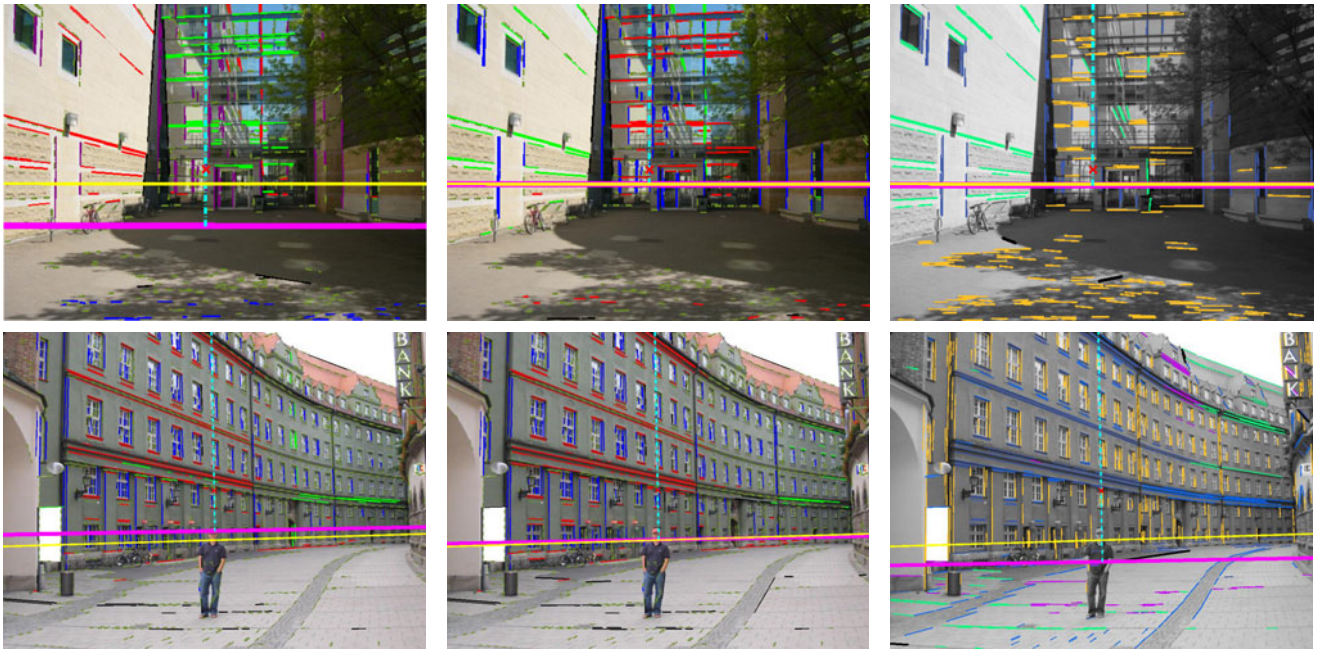


Fig. 11 (Color online) Comparison of the results in horizon and line segments estimation for our Geometric Parsing approach, modified Geometric Parsing approach, which does not include the horizon constraint, and the method of Tardif (Tardif 2009). Coloring of line segments and lines reflects grouping into parallel families. *Pink* and *yellow thick lines* correspond to the found and the ground truth horizons respectively, *cyan line* shows the direction to the zenith and the cross corresponds to the principal point position. In the first and the second columns olive line segments represent line segments that were classified as not belonging to any line. Black line segments correspond to lines that do not belong to any vanishing points. Tardif's method classifies more clutter line segments as belonging to some vanishing point (mostly, line segments on the ground). Omitting the horizon constraint (first column) results in the incorrect grouping of segments into a parallel family (*top*—blue family on the ground, *bottom*—green family on the walls)

Table 1 Time required for each step of the geometric parsing approach: geometric primitives *candidates* detection and parsing (in sec)

Database	Line Segment Detection	Line Detection	VP Detection	Parsing	Total
York Urban	1.01 ± 0.19	9.46 ± 2.97	3.72 ± 2.10	3.00 ± 2.30	17.20 ± 6.49
Eurasian Cities	1.19 ± 0.21	12.58 ± 1.82	4.76 ± 2.96	4.00 ± 2.70	22.53 ± 6.41

ments that belong to vanishing points from background clutter. Note how the method (Tardif 2009) tends to form spurious vanishing points from the background clutter and attach clutter lines to detected clusters.

In addition to our main error measure (horizon accuracy), we also estimated the error of the zenith estimation on York urban dataset (where ground truth Manhattan geometry allows accurate localization of the zenith). We measured the errors as the angle between directions from the principle point to the ground truth zenith and the estimated zenith (in radians). The error for our method is (0.0056 ± 0.0188), for the method (Tardif 2009) is (0.0052 ± 0.0057) which is smaller than for Kosecká and Zhang (2002) (0.0144 ± 0.0707).

Also in the Table 1 we show the mean time and the standard deviation for each step of our framework.[5] As can be seen from this table, candidates calculation steps constitute the considerable part of the time required. It can be reduced by the calculation of less candidates or by the optimization of the code. Also the parameters of the algorithms can be tuned for faster candidates detection. In this work we used unoptimized code and did not focus on time optimization.

[5]The program was tested on the computer Intel Core 2 Quad CPU Q8200 2.34 GHz, 2.00 GB of RAM.

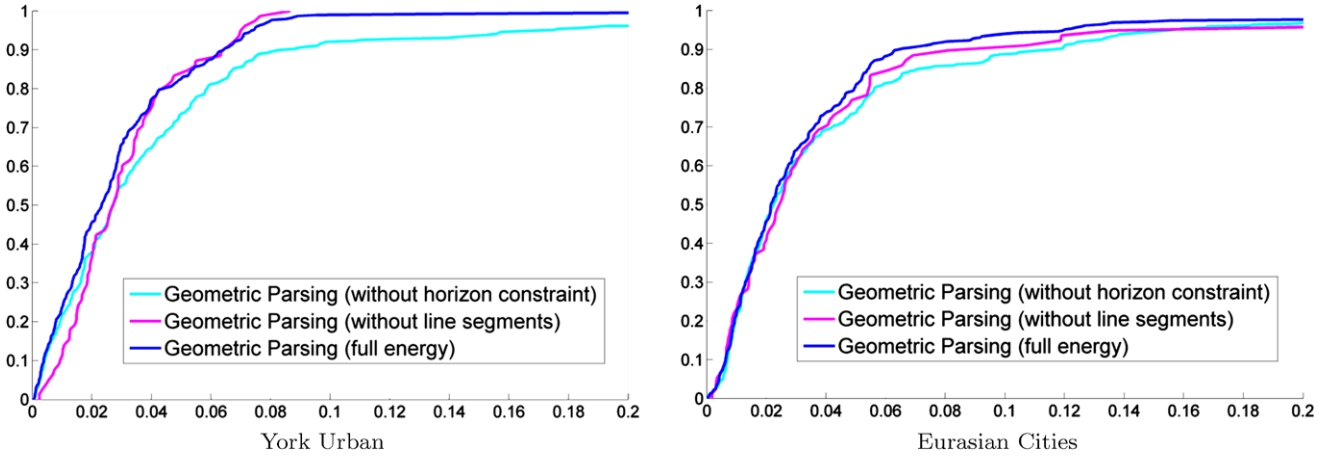


Fig. 12 Cumulative plots comparing the full model with the two truncated models: the model without line segment layer (Barinova et al. 2010b) and the model with omitted horizon constraint. The x-axis corresponds to the horizon estimation error measure (see text for more details). The y-axis corresponds to the share of the images in the test set that has the error less than the respective x value. For both datasets the full model achieves higher accuracy than the truncated models

4.8 Modifications of the Model

In order to find how the terms of the energy influence the result, we compared the original model with two modifications, one of which does not include the horizon constraint term and the other does not include the layer with line segments, so that the edge pixels layer is directly connected to the lines layer (Barinova et al. 2010b). The result of the comparison is shown on Fig. 12.

As the experiments show, the usage of the additional layer and additional relations between geometric primitives allows more accurate estimation of the horizon. Qualitatively, we observed that the layer of line segments helps to choose lines more adequately. And as in our model the inference is performed simultaneously for the whole model, this layer affects the final choice of the vanishing points and horizon.

The horizon term directly influences the result of the horizon estimation. It is especially useful in the case, when there are groups of lines present in the image, which are not horizontal. This effect can also be seen in the qualitative comparison in Fig. 11. This constraint is responsible for more accurate selection of vanishing points, which all lie on the horizon line, orthogonal to the line, connecting the zenith and the principal point.

We have also tested the described modification for the detection of three orthogonal points (see Sect. 3.2) on York dataset. This modification has not significantly improved the performance in horizon estimation as our model already includes strong horizon constraint, which works well on relatively simple York dataset. Such modification however may still be useful for the tasks that require the detection of the three "Manhattan" vanishing points.

5 Summary and Discussion

We formulated the problem of geometric analysis of a single image in an optimization framework. Given a set of observed edge pixels, the framework jointly infers groupings of edge pixels into line segments, line segments into lines, parallel lines, vanishing points and geometric concepts such as the zenith and the horizon. The experimental comparison suggests that such a joint inference results in higher accuracy and robustness compared to bottom-up estimation.

The current framework ignores appearance information from the scene elements. For instance, parallel lines arising due to a railway track or a road might have similar appearance which may provide additional cues for grouping lines and inferring the location of the zenith and the horizon. This information can produce better results and is a topic of a future work. Another interesting direction of work is the incorporation of an uncertainty measure in the presence of edges.

In general, we have shown that incorporating within one model different-level geometric primitives can be beneficial for scene geometry estimation. Similar idea can be used for other tasks if they allow the extraction of several layers of elements. In particular, one can treat detectable objects characteristic to the environment of interest as primitives. Thus, Hoiem et al. (2008) demonstrated how locating such objects (cars and pedestrians in their case) jointly with the estimation of geometric parameters of the scene can benefit both object detection and geometry estimation. In the same way, the candidate detections produced by a conventional object detector may be added as yet another group of variables into our model.

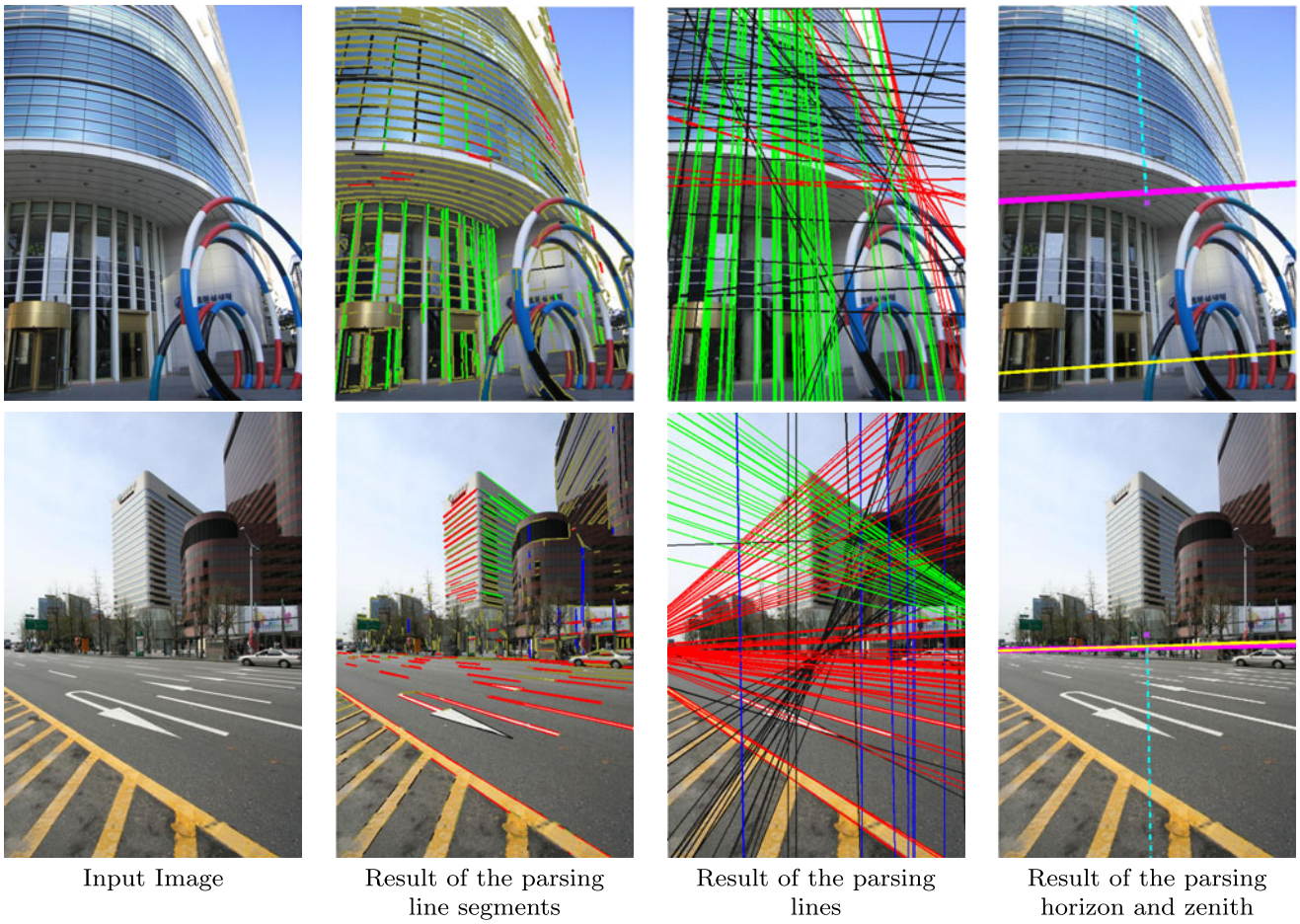


Fig. 13 Failure cases of the algorithm. The notation of the primitives is the same as on the Fig. 7. In the cases when the input image does not satisfy all the conditions we require (we assume that the buildings consist mostly of straight lines), the method can fail. For example, on the first image the most part of the elements on the image are not straight, so there is no strong presence of parallel families of lines. On the second image the method shows good performance according to our horizon metric, even though not all the vanishing points are detected. Using our metric it is quite difficult to learn the parameters of the algorithm to detect all the vanishing points

References

Aguilera, D. G., Lahoz, J. G., & Codes, J. F. (2005). A new method for vanishing points detection in 3d reconstruction from a single view. In *Proc. of ISPRS Commission V*.

Almansa, A., Desolneux, A., & Vamech, S. (2003). Vanishing point detection without any a priori information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(4), 502–507.

Antone, M. E., & Teller, S. J. (2000). Automatic recovery of relative camera rotations for urban scenes. In *CVPR* (pp. 2282–2289).

Barinova, O., Lempitsky, V., & Kohli, P. (2010a). On detection of multiple object instances using hough transforms. In *CVPR*.

Barinova, O., Lempitsky, V., Tretiak, E., & Kohli, P. (2010b). Geometric image parsing in man-made environments. In *ECCV*.

Barnard, S. (1983). Interpreting perspective images. *Artificial Intelligence*, *21*(4), 435–462.

Beardsley, P. Murray, D. (1992). Camera calibration using vanishing points. In *BMVC* (pp. 416–425).

Boulanger, K., Bouatouch, K., & Pattanaik, S. (2006). Atip: A tool for 3d navigation inside a single image with automatic camera calibration. In *EG UK theory and practice of computer graphics*.

Cipolla, R., Drummond, T., & Robertson, D. P. (1999). Camera calibration from vanishing points in image of architectural scenes. In *BMVC*.

Collins, R. T., & Weiss, R. S. (1990). Vanishing point calculation as a statistical inference on the unit sphere. In *ICCV* (pp. 400–403).

Coughlan, J. M., & Yuille, A. L. (1999). Manhattan world: Compass direction from a single image by Bayesian inference. In *ICCV* (pp. 941–947).

Denis, P., Elder, J. H., & Estrada, F. J. (2008). Efficient edge-based methods for estimating Manhattan frames in urban imagery. In *ECCV (2)* (pp. 197–210).

Deutscher, J., Isard, M., & MacCormick, J. (2002). Automatic camera calibration from a single Manhattan image. In *ECCV (4)* (pp. 175–205).

Duric, Z., & Rosenfeld, A. (1996). Image sequence stabilization in real time. *Real-Time Imaging*, *2*(5), 271–284.

Flint, A., Mei, C., Reid, I., & Murray, D. (2010). Growing semantically meaningful models for visual slam. In *Proc. IEEE conference on computer vision and pattern recognition* (pp. 467–474). Los Alamitos: IEEE Computer Society.

Hartley, R., & Zisserman, A. (2003). *Multiple view geometry in computer vision*. Cambridge: Cambridge University Press.

Hedau, V., Hoiem, D., & Forsyth, D. (2009). Recovering the spatial layout of cluttered rooms. In *ICCV* (pp. 1849–1856).

Hedau, V., Hoiem, D., & Forsyth, D. (2010). Thinking outside the box: using appearance models and context based on room geometry. In *ECCV* (pp. 224–237).

Hoiem, D., Efros, A. A., & Hebert, M. (2005a). Automatic photo pop-up. *ACM Transactions on Graphics*, *24*(3), 577–584.

Hoiem, D., Efros, A. A., & Hebert, M. (2005b). Geometric context from a single image. In *ICCV* (pp. 654–661).

Hoiem, D., Efros, A. A., & Hebert, M. (2008). Putting objects in perspective. *International Journal of Computer Vision*, *80*(1), 3–15.

Kosecká, J., & Zhang, W. (2002). Video compass. In *ECCV (4)* (pp. 476–490).

Lee, D. C., Hebert, M., & Kanade, T. (2009). Geometric reasoning for single image structure recovery. In *CVPR*.

Lee, D. C., Gupta, A., Hebert, M., & Kanade, T. (2010). Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In *NIPS*.

McLean, G. F., & Kotturi, D. (1995). Vanishing point detection by line clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *17*(11), 1090–1095.

Morel, J.-M., Randall, G., Grompone von Gioi, R., & Jakubowicz, J. (2008). Lsd: A fast line segment detector with a false detection control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*, 722–732.

Rother, C. (2000). A new approach for vanishing point detection in architectural environments. In *BMVC*.

Schaffalitzky, F., & Zisserman, A. (2000). Planar grouping for automatic detection of vanishing lines and points. *Image and Vision Computing*, *18*, 647–658.

Schindler, G., & Dellaert, F. (2004). Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments. In *CVPR (1)* (pp. 203–209).

Tardif, J.-P. (2009). Non-iterative approach for fast and accurate vanishing point detection. In *ICCV*.

Tu, Z., Chen, X., Yuille, A. L., & Zhu, S. C. (2005). Image parsing: Unifying segmentation, detection, and recognition. *International Journal of Computer Vision*, *63*(2), 113–140.

Tuytelaars, T., Van Gool, L. J., Proesmans, M., & Moons, T. (1998). A cascaded hough transform as an aid in aerial image interpretation. In *ICCV* (pp. 67–72).

Wildenauer, H., & Vincze, M. (2007). Vanishing point detection in complex man-made worlds. In *ICIAP* (pp. 615–622).

Yu, S., Zhang, H., & Malik, J. (2008). Inferring spatial layout from a single image via depth-ordered grouping. In *POCV*.