Image and Vision Computing 29 (2011) 569–579

Contents lists available at SciVerse ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Toward coherent object detection and scene layout understanding ☆

Sid Yingze Bao *, Min Sun, Silvio Savarese

Univsersity of Michigan at Ann Arbor, Ann Arbor, MI, 48105, USA

ARTICLE INFO

Article history:
Received 3 May 2011
Received in revised form 25 July 2011
Accepted 4 August 2011

Keywords:
Object detection
Scene layout
Focal length estimation
Supporting surface estimation

ABSTRACT

Detecting objects in complex scenes while recovering the scene layout is a critical functionality in many vision-based applications. In this work, we advocate the importance of geometric contextual reasoning for object recognition. We start from the intuition that objects' location and pose in the 3D space are not arbitrarily distributed but rather constrained by the fact that objects must lie on one or multiple supporting surfaces. We model such supporting surfaces by means of hidden parameters (i.e. not explicitly observed) and formulate the problem of joint scene reconstruction and object recognition as the one of finding the set of parameters that maximizes the joint probability of having a number of detected objects on K supporting planes given the observations. As a key ingredient for solving this optimization problem, we have demonstrated a novel relationship between object location and pose in the image, and the scene layout parameters (i.e. normal of one or more supporting planes in 3D and camera pose, location and focal length). Using a novel probabilistic formulation and the above relationship our method has the unique ability to jointly: i) reduce false alarm and false negative object detection rate; ii) recover object location and supporting planes within the 3D camera reference system; iii) infer camera parameters (view point and the focal length) from just one single uncalibrated image. Quantitative and qualitative experimental evaluation on two datasets (desk-top dataset [1] and LabelMe [2]) demonstrates our theoretical claims.

1. Introduction

When we observe a complex scene such as an office or a street, it is easy for our visual system to recognize the objects and infer their spatial organization in the environment. Objects do not appear in arbitrary locations: it is very unlikely to observe a monitor floating in the air or a car hanging from a building. Objects are rather organized in the physical space in consistent geometrical configurations: their locations and poses obey the law of physics (objects lie on supporting planes in stable configurations) and follow the conventions of human behavior. It is clear that when humans observe the environment, such constraints will help reinforce the process of joint recognition and scene layout recovery [3,4]. The recognition of objects with the estimate of their locations, scales and poses helps infer the spatial properties of the environment (e.g., the location and orientation of the surface where objects lie), and in turn the scene layout provides strong spatial contextual cues as for where and how objects are expected to be found. Contributions in computer vision for the past decade have followed the common paradigm of recognizing objects in isolation [5–9], regardless of the geometrical contextual cues. It is indeed true that objects can be in general recognized even when no information about the scene layout is provided. However, we claim that joint object recognition and scene reconstruction are critical if one wants to obtain a coherent understanding of the scene as well as minimize the risk of detecting false positive examples or missing true positive ones. This ability is crucial for enabling higher level visual tasks such as event or activity recognition and in applications such as robotics, autonomous navigation, and video surveillance.

The intuition that recognition and reconstruction are mutually beneficial has been initially explored in early works of computer vision [10–15], and recently revitalized in [16–27]. In Hoiem et al. [16], the process of detecting objects in a complex scene is enhanced by introducing the geometrical contextual information of the scene layout [28] (e.g., vertical surfaces, ground horizontal planes, etc.). More explicit reasoning on the relationship between supporting planes and objects hasbeen investigated in Hoiem et al. [29] and Hedau et al. [17,18]. Hedau et al. [17,18] introduced a flexible methodology for estimating the layout of indoor scenes by modeling the scene using a3D cube representation. Following our preliminary study [30], we too advocate the importance of geometrical contextual reasoning for object recognition and focus on demonstrating that the contextual cues provided by object location and pose can be used, in turn, to reinforce the detection and prune out false alarms (Fig. 1). Our key idea is that objects' locations and poses in the 3D space are not arbitrarily distributed but rather constrained by the fact that objects must lie on one or multiple supporting surfaces. We model such

☆ This paper has been recommended for acceptance by Sinisa Todorovic.

* Corresponding author.
E-mail addresses: yingze@umich.edu (S.Y. Bao), sunmin@umich.edu (M. Sun), silvio@eecs.umich.edu (S. Savarese).
URL: http://www.eecs.umich.edu/~yingze (S.Y. Bao).

0262-8856/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2011.08.001

supporting surfaces by hidden parameters (i.e. not explicitly observed) and seek a configuration of objects and supporting surfaces in the 3D space that best explains the observations, including the estimation of each object's location, scale and pose. To this end, we leverage on recent methods for detecting multi-category objects and estimating their poses accurately from a single image [31–36]. Unlike [16], where contextual information was partially provided by the explicit estimation of surface orientation using the geometric context operator [28], we only use the objects *themselves* for extracting contextual cues. Thus, we do not require supporting planes or other scene surfaces to be visible or detectable in order to perform the joint recognition and reconstruction. Rather, supporting planes are implicitly estimated from the observation of the object location and pose in the image. Moreover, our camera representation is general: We only hypothesize that the camera has zero skew and unit pixel ratio (but unknown focal length). Most importantly, we do not make the assumption that the camera is at fixed distance from the ground plane and has a fixed view angle. Because of these properties, our algorithm can be successfully applied in both outdoors and indoors scenarios. Notice that Hedau etal. [17,18] use cues such as vanishing lines that are complementary to ours and could be nicely integrated into our framework. Also notice that physics-based constraints such as those introduced in Gupta et al. [26] enable different ways for modeling the interaction between scene and objects wherein, in this case, objects are mostly identified as urban elements (i.e., buildings and houses). Finally, in Payet et al. [27] the analysis of textures is introduced to provide scene-specific constraints among objects.

The main contributions of our work include: 1. A novel representation that can jointly model 3D objects locations and 3D supporting surfaces (planes) from the observations in a single image. Concretely, the problem of joint scene reconstruction and object recognition is formulated as finding a set of parameters that maximize the joint probability of having a number of detected objects on K supporting planes given the observations (Section 2). 2. A relationship that connects the 2D image observation of object location and zenith angle pose with the normals of the supporting planes and with the camera focal length parameter. We prove that this relationship yields necessary conditions for estimating the camera focal length and the supporting planes' 3D orientations and locations (in the camera reference system) from the locations and zenith poses of at least 3 objects in the image. The relationship is general in that objectsdo not necessarily need to lie on the *same* supporting plane as long as their supporting planes are parallel with respect to each other and the objects are not collinear (Section 3.1). 3. A framework that uses the above relationships and a novel probabilistic formulation to jointly detect objects (so as to reduce false alarm and false negative rates) and recover (within the camera reference system) the objects' 3D locations, the 3D supporting planes, and the camera focal length parameter. All of the outcomes mentioned above are merely based on one single semi-calibrated image (Section 2). Experimental evaluation on two datasets (desk-top dataset [1] and the car and pedestrian Label-Me dataset [2]) demonstrates our theoretical claims (Section 4).

2. Modeling objects and scene layout

Given an image portraying a number of objects, our work proposes a new model for jointly recognizing objects in the scene and recovering the scene layout that best "explains" the evidence measured in the image. In this paper, the term "scene layout" indicates: i) the objects' 3D locations and poses in the camera reference system; ii) the 3D locationand orientation of their supporting planes in the camera reference system; iii) the camera focal length. In this section we will first introduce notations and assumptions and then formulate the problem.

2.1. Assumptions and notations

We assume that each object lies on a supporting plane at an up-right pose. This assumption is satisfied in most real world scenes. For example, a car is usually touching the ground by four wheels rather than only two and a wineglass is usually standing vertically rather than obliquely (Fig. 2). This is consistent with the common observation that objects rarely float in the air or appear in unstable poses. Furthermore, if multiple supporting planes co-exist in one image, we assume that these planes are all parallel to each other. This parallel relationship of planes holds for most daily-life scenes. Notice that we are *not* assuming *the camera* must be "up-right" with respect to the supporting surfaces.

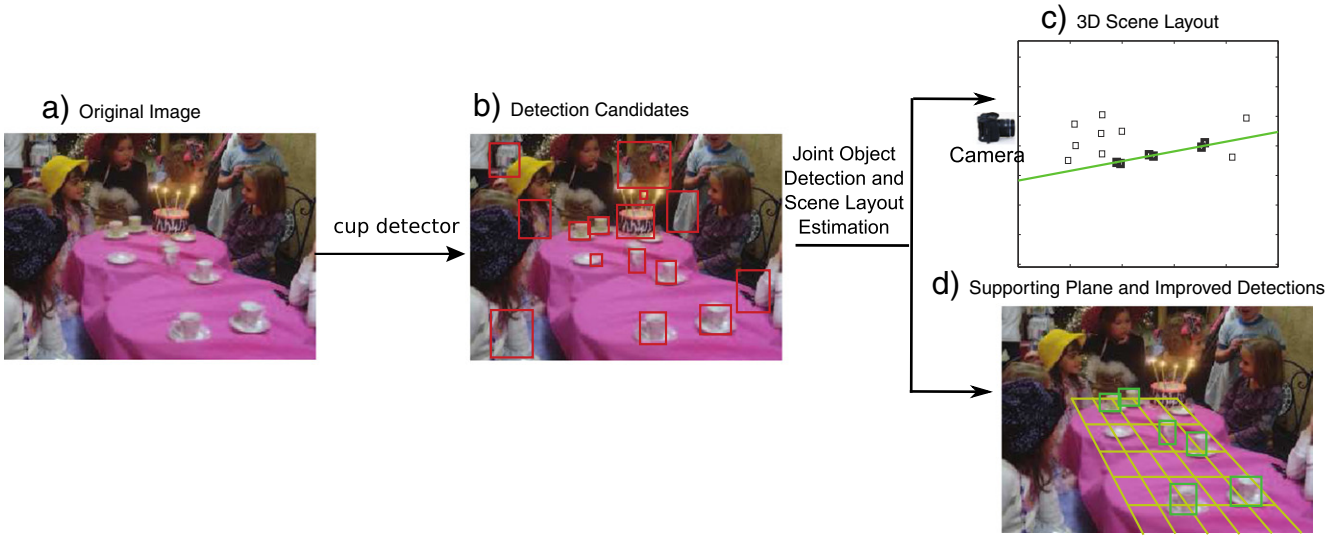

Fig. 1. A conceptual illustration of the flowchart of our algorithm. (a) Original input image with unknown camera parameters; (b) Detection candidates provided by a baseline "cup" detector; (c) The 3D layout. The figure shows the side view of the 3D reconstructed scene. The supporting plane is shown in green. Dark squares indicate the objects detected and recovered by our algorithm; light squares indicate objects detected by the baseline detector and identified as false alarms by our algorithm; (d) Our algorithm detects objects and recovers object locations and supporting plane (in gold color) orientations and locations within the 3D camera reference system from one single image. We show only a portion of the recovered supporting plane for visualization purposes.

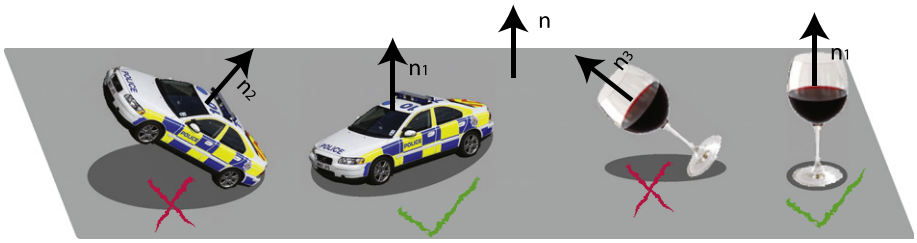

Fig. 2. If the normal of a plane is n, objects lying on this plane tend to share the same normal direction $n_1//n$. Objects whose normal is not parallel to n (e.g. n_2 and n_3) are unlikely to sit on that supporting plane.

2.1.1. Plane in 3D

A plane in 3D has three degrees of freedoms. Hence, it can be parameterized by its surface normal n (Fig. 4) and its distance h to the origin of the coordinate system (i.e. the camera).

2.1.2. Object in 3D

We define the parametrization to identify an object's location and pose in 3D coordinate system. Assuming that an object is enclosed by the tightest bounding cube lying on the supporting plane (Fig. 3(a)), the object 3D location O can be specified by knowing the centroid of the 3D bounding box. Furthermore the object's pose can be defined by the three bounding box's perpendicular surfaces' normal n, q and t (Fig. 3(a)). As discussed above, we assume all objects' up direction n should be the same as the normal of the supporting plane. Let the unit view sphere associated to an object be the collection of viewpoints equally distant from the object. In the view sphere of an object, let r be the ray that connects O and the camera center (Fig. 3(b)). Let *zenith* angle ϕ be the angle between the ray r and n (Figs. 3(b) and 4). Define *azimuth* angle θ be the angle formed by object's frontal vector q and a vector r_s. r_s is the projection of the ray r onto the plane perpendicular to n (i.e. supporting plane). We denote by ϕ the measured zenith pose from image, and by $\hat{\phi}$ the estimated zenith pose consistent with the underlying surface layout. We will explain in details how to compute $\hat{\phi}$ and measure ϕ in Section 3.1.

2.1.3. Object in 2D

An object in the image plane is uniquely identified by a bounding box *bbox* tightly enclosing the object in 2D. We define *bbox* by its center (u, v), the height h, and width w in image coordinate (Figs. 3(b) and 7).

2.1.4. Candidate detection

We assume a number of object class detectors are available and each detector returns a number of candidate detections m, where each m is defined by a bounding box *bbox* and the estimated object pose

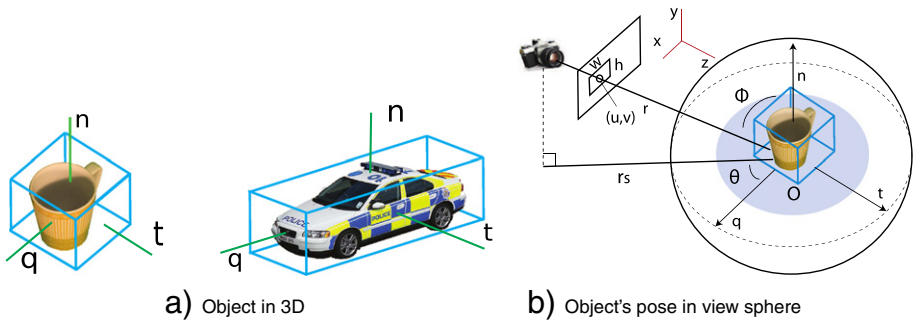

Fig. 3. (a): Three perpendicular directions characterize the pose of a rigid object in a given reference system. n is defined as the object's normal. (b): Definition of zenith angle ϕ and azimuth angle θ, given the object's pose in the camera reference coordinates.

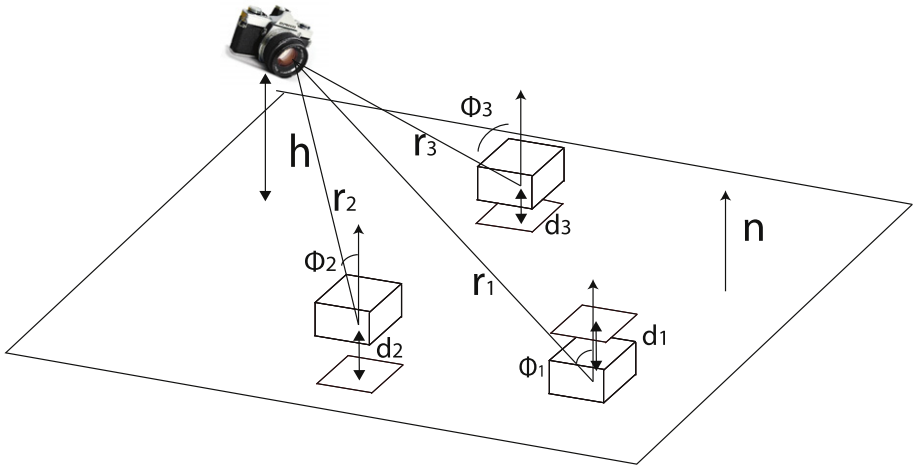

Fig. 4. Geometric relationships of ϕ, r, d, h and n.

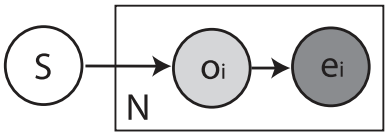

Fig. 5. Graphical model of conditional independence for supporting plane parameters and detection result, where o_i is partially observed and e_i fully observed. Details are in Section 2.2.

represented by the zenith angle ϕ and azimuth angle θ. Thus, $m = \{bbox, \phi, \theta\}$ (Figs. 3(b) and 7).

2.1.5. True-positive flag

We assign a true-positive flag t to each detection result. $t = 1$ if a candidate detection is associated to the true object category, and $t = 0$ if a candidate detection indicates the presence of an object from an incorrect category or just background. Given an image measurement (i.e. portion of the image that is used by detector to assess whether an object class has been detected and may yield a detection m or not), the detector returns a confidence score indicating how likely a detection is a true positive, i.e. $t = 1$.

2.2. Joint model of objects and supporting planes

We propose a probabilistic model which incorporates the interaction between objects and supporting planes. The key idea is that the estimation of both candidate detections and the underlying geometry is more accurate than estimating each term independently. For simplicity, we denote scene information $S = \{n, h, f\}$ where n and h are supporting plane's parameters and f is the camera focal length. We formulate the joint probability of the candidate detections $o = \{o_i\} = \{m_i, t_i\}$, image evidence $E = \{e_i\}$, and scene information S following the graphical model in Fig. 5 as

$$p(o, E, S) = p(S) \prod_{i=1}^{N} p(o_i|S) p(e_i|o_i)$$

$p(o_i|S)$ is the probability of an object given scene information. $p(o_i|S)$ can be further decomposed as $p(o_i|S) = p(t_i|m_i, S)p(m_i|S) \propto p(t_i|m_i, S)$ because the probability of a bounding box given only geometrical constraint $p(m_i|S)$is a constant. Consequently,

$$p(o, E, S) \propto p(S) \prod_{i=1}^{N} p(t_i|m_i, S) p(e_i|m_i, t_i)$$

Each term is described as follows:

$\mathbf{p(S)}$ is the scene prior which can be modeled as uniform distribution within a range of n, h and f. Details of the selection of range values for these parameters are in Section 4.

$\mathbf{p(e|t, m)}$ is related to the detection result's confidence. Assume $p(m, t)$ and $p(e)$ follow a uniform distribution, we have

$$p(e|t, m) = p(t, m|e)p(e) / p(t, m) \propto p(t, m|e)$$

where $p(t, m|e)$ is the detection's confidence returned by the detector.

$\mathbf{p(t|m, S)}$ is the probability that the detection is a true positive, given the candidate detection m and the scene information S.

One contribution of our work is to estimate $p(t|m, S)$ with the help of two geometrical relationships: 1. Relationship between focal length f, zenith angle ϕ and supporting plane normal n. 2. Relationship between the object-to-plane distance d, the object's 3D coordinates O,

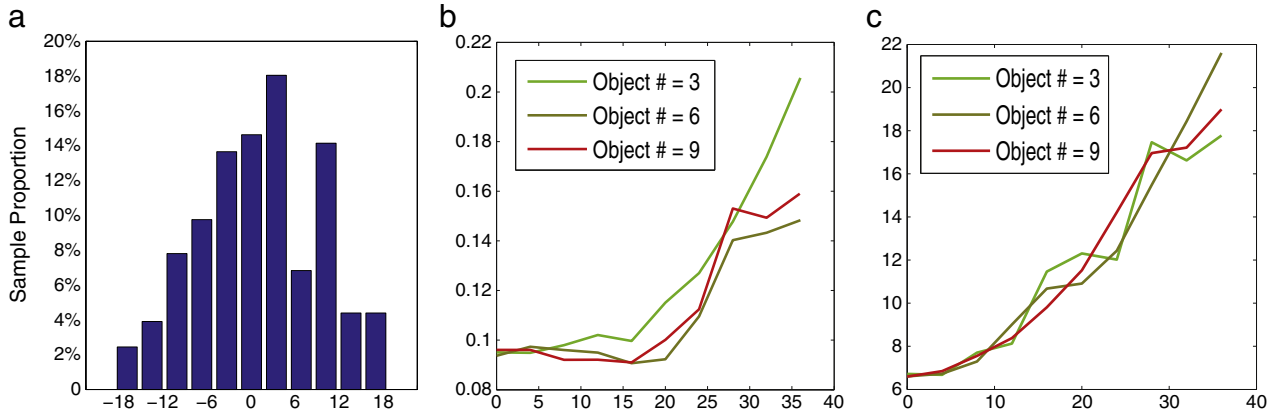

Fig. 6. (a) Histogram of the actual error of the measurement of object zenith angle φ. The Y axis is the fraction of testing samples that fall in each error bin. The X axis are error bins in degree. The standard deviation of zenith angle measurement is 8.4°. (b)(c) Error analysis of Eq. (6). X axis is the variance of Gaussian noise in degree. (b) Y axis is $e_f = \left| \left(f - \hat{f} \right) / f \right|$. (c) Y axis is $e_n = |arccos(n \cdot \hat{n})|$ in degree. This figure is best viewed in color.

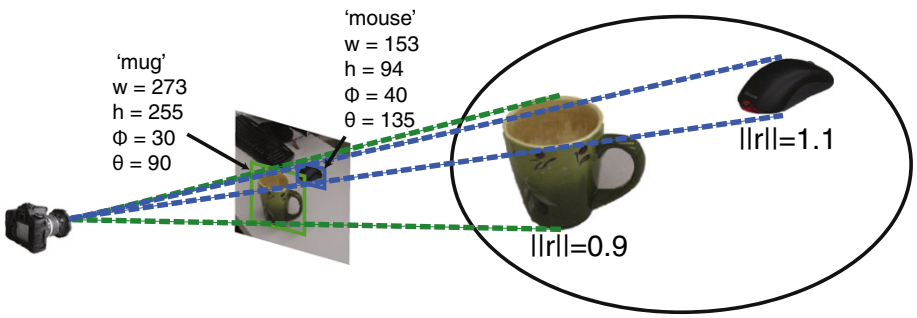

Fig. 7. An illustration of a detected object and its corresponding 3D pose. Given object's image bounding box and estimated pose, its distance to the camera can be estimated using the procedure in Section 3.2.

the plane's normal n, and the camera-to-plane distance h (Fig. 4). In Section 3 we will explain in details these relationships. Here, we formulate

$$p(t=1|m,S)\propto p(t=1|d)p\left(t=1|\phi-\hat{\phi}\right) \quad (1)$$

That is to say rather than using S directly, we use d and $\hat{\phi}$ to determine if the candidate detection m is true. We assume Gaussian distribution $p(t=1|d)=N(d;0,\sigma_d)$, and $p\left(t=1|\phi-\hat{\phi}\right)=N\left(\phi-\hat{\phi};0,\sigma_\phi\right)$, where $\hat{\phi}$ is the inferred zenith and ϕ is the measured zenith from image. Thus, $t_i=1$ is highly likely when the scale of the bounding box and the predicted pose by detector are consistent with the supporting plane.

To simultaneously estimate the scene information S, and the true-positive flag $\{t_i\}$ of each candidate detection, we want to find S and $\{t_i\}$ such that the joint probability $p(o, E, S)$ is maximized. Unknowns are $\{t_i\}$, S, and measurements are $\{m_i\}$ and $\{p(e_i|o_i)\}$ given by detector. The problem is equivalent to find S and $\{t_i\}$ by means of the following optimization problem:

$$\{S,\{t_i\}\}=\arg\max_{S,\{t_i\}} ln\,p(S)+\sum_{i=1}^{N}[ln\,p(t_i|m_i,S)+lnp(e_i|t_i,m_i)] \quad (2)$$

2.3. Solving the optimization

In this section we solve the optimization problem of Eq. (2) in Section 2.2. Define $z(S)$ as

$$\begin{aligned} z(S) &= \max_{\{t_i\}}\sum_{i=1}^{N}[ln\,p(t_i|m_i,S)+ln\,p(e_i|t_i,m_i)] \\ &= \sum_{i=1}^{N}\left\{\max_{t_i}[ln\,p(t_i|m_i,S)+ln\,p(e_i|t_i,m_i)]\right\} \end{aligned}$$

For a fixed value of S, the value of each term in the above summation can be calculated independently. Hence, the best global configuration of $\{t_i\}$ given S can be found after N comparisons of $ln\,p(t_i=1|m_i,S)+ln\,p(e_i|t_i=1,m_i)$ with $ln\,p(t_i=0|m_i,S)+ln\,p(e_i|t_i=0,m_i)$. Therefore, $\{t_i\}$ can be computed as a function of S:

$$t_i^*=\arg\max_{t_i} ln\,p(t_i|m_i,S)+ln\,p(e_i|t_i,m_i) \quad (3)$$

Having estimated t_i, to find S is equivalent to

$$S=\arg\max_{S}[ln\,p(S)+z(S)] \quad (4)$$

We propose to solve Eq. (4) by searching on a large but finite set $\mathbf{S}$ to find the optimal S. This can be computed almost in real-time by CUDA parallel programming. Let $\mathbf{F}\in R$ be the set of all the possible values of the focal length f. Let $\mathbf{N}\in R^3$ be the set of all possible values of the plane normal n. Let $\mathbf{H}$ be the set of all possible values of the plane height h. The search space is $\mathbf{S}=\mathbf{F}\times\mathbf{N}\times\mathbf{H}$. The details can be found in Algorithm (2.4).

2.4. Extension to multiple planes

The above approach estimates the single most likely supporting plane by obtaining the highest log likelihood score. This approach can be extended to handle the case of multiple supporting planes by using an iterative procedure. Denote by K the number of already estimated planes. Denote by A the set of active object detection candidates. At the beginning, $K=0$ and A is all object detection candidates. First, we employ this approach to find the best plane configuration S. Then we determine the objects that sit on plane S and remove them from A. Next, the algorithm processes the remaining detection candidates. The algorithm ceases after K is larger than a predefined threshold. Notice that, since all the supporting surfaces are assumed to have the same normals, the "at least three objects" requirement (Section 3.1) is no longer necessary for other planes except the first one. The procedure is described in Algorithm (2.4)

Algorithm 1. Estimating scene layout from images

1. Set the number of already estimated planes $K=0$. Set the active object detection set A to be the set of all object detection candidates.
2. If $K=0$, enumerate $S^j\in\mathbf{F}\times\mathbf{N}\times\mathbf{H}$; else enumerate $S^j\in f\times n\times\mathbf{H}$ where f and n are the already estimated focal length and plane normal.
3. For each enumeration S^j, estimate the flag t_i^j for all objects $o_i\in A$ by Eq. (3)
4. Given the estimated $\{t_i^j\}$ for all enumerations $\{S^j\}$, find $S^*\in\{S^j\}$ by Eq. (4).
5. Take S^* as one estimated supporting surface. Remove the objects that have true flag $t_i=1$ from A. Set $K=K+1$.
6. if K is larger than a predefined threshold, then stop. Otherwise goto 2.

3. Relating camera measurements and supporting planes

In this section we derive the relationship among the estimated zenith angle pose ϕ_i of an object in the image plane, the supporting plane normal n and the camera focal length f. We show that by measuring ϕ_i for at least three non-collinear objects, we can estimate f and n from a single image. Notice that in order for this result to be true, objects are not necessarily required to lie on a single supporting plane, but each object can have its own supporting plane as long as all the planes are parallel. This result is one of the main contributions of our paper and provides sufficient conditions for estimating $p(t_i|m_i,S)$. In Section 3.2, we will explain how to locate an object O in 3D and establish a relationship between O, h, d and n.

3.1. Relationship between focal length and supporting plane normal

We adopt homogeneous coordinates to represent objects in 3D and in the image plane coordinates. Let $(\tilde{u},\tilde{v},1)$ be the homogeneous coordinates of the object projection in the image plane. We assume that the camera is semi-calibrated. That is, we assume that the camera center (u_0, v_0) is known, the pixel ratio $\alpha=1$ and the camera has zero-skew. These are reasonable assumptions that hold in most practical cases. By translating any point in the image plane by $(u_i,v_i)=((\tilde{u}_i,\tilde{v}_i)-(u_0, v_0))^T$, we write the camera intrinsic parameter matrix as $K=diag(f,f,1)$.

Let r_i be the line of sight connecting the camera center and an object O_i, which passes through an object's location (u_i, v_i, f) in the image. Then the direction of the line of sight r_i in camera coordinates is $(u_i/f, v_i/f, 1)$. Let $n=(n_1,n_2,n_3)$ denote the normal of the supporting plane in camera coordinates. s_i and n are shown is Fig. 4. If we enforce n to have unit norm, then $n_1^2+n_2^2+n_3^2=1$. Thus:

$$(u_i,v_i,1)\begin{pmatrix}n_1\\n_2\\n_3f\end{pmatrix}=-cos\phi_i\sqrt{u_1^2+v_1^2+f^2} \quad (5)$$

Using Eq. (5), the key term $\hat{\phi}$ in Eq. (1) can be computed given n_1, n_2, n_3, and f, i.e. part of S.

3.1.1. Measuring zenith angle from single image

It is clear that our formulation relies on the measurement of the objects' zenith angles in the image plane. Recently, a number of techniques such as [33,32,31,35] have been proposed to estimate object pose from single images. We used an adapted version of [33] to

measure zenith angles ϕ from the image. Quantitative experimental analysis on our in-house dataset shows that our detector is capable of generating zenith pose classification results that are compatible with our sensitivity analysis (Section 3.1.3 and Fig. 6).

3.1.2. Estimating 3D plane orientation via object zenith angles

In this section, we show that the normal of the supporting planes and the focal length of the camera can be estimated from the objects' zenith angles phi_i and their locations from just one single image. If a total number of N measurements ϕ_i, u_i, v_i $(i=1\ldots N)$ are available, following Eq. (5) we obtain:

$$\begin{bmatrix} u_1 & v_1 & f \\ u_2 & v_2 & f \\ u_3 & v_3 & f \\ & \vdots & \\ u_N & v_N & f \end{bmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} = \begin{pmatrix} -\cos\phi_1\sqrt{u_1^2+v_1^2+f^2} \\ -\cos\phi_2\sqrt{u_2^2+v_2^2+f^2} \\ -\cos\phi_3\sqrt{u_3^2+v_3^2+f^2} \\ \vdots \\ -\cos\phi_N\sqrt{u_N^2+v_N^2+f^2} \end{pmatrix} \tag{6}$$

This equation allows us to solve $\{f, n_1, n_2, n_3\}$ from the objects' measurements ϕ_i, u_i, v_i $(i=1\ldots N)$ in just one single image. The following proposition gives the conditions for the existence of a solution of Eq. (6).

Proposition 1. *Eq.(6)admits one or at most two non-trivial solutions for $\{f, n_1, n_2, n_3\}$ if at least three non-aligned observations (u_i, v_i) (i.e. non-collinear in the image) are available. If the observations are collinear, then Eq.(6)has an infinite number of solutions.*

Proof. Suppose at least three objects are not collinear in an image, then the rank of the left matrix on the left-hand side of Eq. (6) is 3. Therefore Eq. (6) provides 3 independent constraints. The unknowns in Eq. (6) are n_1, n_2, n_3, f. With these constraints, each of n_1, n_2, n_3 can be expressed as a function of f, i.e. $n_i = n_i(f)$. Because $\|n\| = 1$, we obtain an equation about f:

$$\sum_{i=1\ldots3} n_i^2(f) = 1$$

In the above equation, f appears in the form of f^2 and f^4. Therefore, there are at most two real positive solutions for f. Given f, $\{n_1, n_2, n_3\}$ can be computed as $n_i = n_i(f)$.

If all objects are collinear in the image, then an infinite number of solutions exist for Eq. (6). If all objects are collinear, the rank of the left matrix in the left-hand side of Eq. (6) is 2. Without loss of generality, assume $(u_1, v_1) \neq 0$. In such a case, after using Gaussian elimination, Eq. (6) will be in the following form:

$$\begin{bmatrix} \alpha & \beta & f \\ \gamma & \epsilon & 0 \\ 0 & 0 & 0 \\ & \vdots & \end{bmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} = \begin{pmatrix} \zeta \\ \eta \\ 0 \\ \vdots \end{pmatrix} \tag{7}$$

If $\hat{f}, \hat{n1}, \hat{n2}, \hat{n3}$ is a solution, then $\hat{f}, \hat{n1} + km_1, \hat{n2} + km_2, \hat{n3} + km_3$ is also a solution of Eq. (7), where (m_1, m_2, m_3) is the non-trivial solution the following equation:

$$\begin{bmatrix} \alpha & \beta & f \\ \gamma & \epsilon & 0 \end{bmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = 0$$

Hence, Eq. (6) admits an infinite number of solutions. □

Eq. (6) guarantees that as long as at least 3 objects do not lie on the same line in the image, it is possible to express the focal length of the camera and the normal of the supporting planes as a function of the objects' locations and zenith pose measurements in the image. Notice that this equation does not assume that all objects are placed on one unique plane and it also does not make the assumption that the camera has no in-plane-rotation (tilt).

3.1.3. Error analysis

We use a numerical simulation to analyze the robustness of the estimation of f and n in solving Eq. (6) as a function of noise in the measurements ϕ. For a total number N of objects, first a random set of object's bounding box $\{u_i, v_i\}$, plane's normal n and focal length f are synthetically generated. Then the corresponding object's zenith angle ϕ_i is computed by Eq. (5). Next we add Gaussian noise w of variance σ to the object's zenith $\tilde{\phi}_i = \phi_i + w$. Consequently, given $\left\{\tilde{\phi}_i\right\}$ and $\{u_i, v_i\}$, we compute the normal of the plane $\hat{n}$ and the focal length $\hat{f}$, by solving Eq. (6) using the Levenberg-Marquardt method. Fig. 6(b) and (c) show the mean value of the absolute errors v.s. the number of objects and the noise level (see figure captions for details). These plots relate the accuracy in estimating n and f as a function of the error in measuring the zenith angle ϕ. Given that our detector returns ϕ with an error of about 10° (Fig. 6(a)), Fig. 6(b) and (c) show that the corresponding error in estimating n and f is reasonably low.

3.2. Locating objects in 3D

In this section, we explain the relationship between S and d and how to locate objects in the 3D camera reference system. Denote by $\|r\|$ the distance between the object location O and the camera. It is impossible to estimate $\|r\|$ without any prior knowledge about the camera or the object if only a single image is available. However, assuming that we have some prior knowledge about the real size of the 3D object, the object distance $\|r\|$ can be estimated from the object scale in the image by means of an inversely proportional relationship. Specifically, if an object's image bounding box's height and width are h and w, its category is c, and its estimated pose is θ and ϕ, we approximate its distance $\|r\|$ by the following linear combination in $\frac{1}{w}$ and $\frac{1}{h}$

$$\|r\| \simeq \left(\alpha(\theta,\phi,c)\frac{1}{w} + \beta(\theta,\phi,c)\frac{1}{h}\right)\cdot f \tag{8}$$

where α and β are function of the object's pose and class label and f is the focal length. α and β are related to the physical 3D shape of the object category. A more precise modeling of this relationship goes beyond the scope of this paper. We instead use linear regression to learn α and β for each set of θ, ϕ, c in the training set where ground truth pose and distance $\|r\|$ are available (Fig. 7). As a result, given candidate object $m = \{bbox, \theta, \phi\}$ and its category c, its 3D coordinates can be estimated in the camera coordinates as follows:

$$O \simeq \frac{\|r\|}{\sqrt{(u/f)^2 + (v/f)^2 + 1}} \begin{pmatrix} u/f \\ v/f \\ 1 \end{pmatrix}$$

This allows us to relate the 3D coordinates of candidate object O, the supporting plane parameters (n, h), and the distance d between object and the supporting plane as $d = O^T n + h$ (Fig. 4).

4. Evaluation

In this section we qualitatively demonstrate the ability of our framework to jointly estimate the scene layout (camera location, supporting plane orientation and object location in the 3D space) as well as improve the accuracy in detecting objects. We test our algorithm on a novel indoor desk-top database [1] as well as on the LabelMe [2] outdoor pedestrian and cars dataset. We use the Graphic

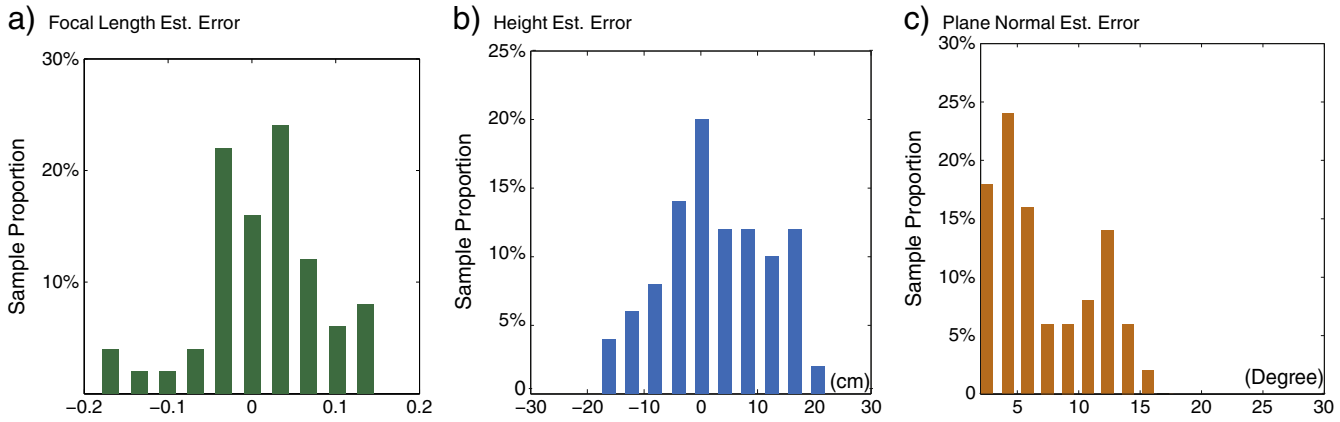

Fig. 8. Experimental results on our desk-top dataset. Y axis is the proportion of test images associated to an error interval (X axis). (a) shows the error when estimating the focal length on 50 test images: the ground-truth focal length f^i_{gt} is known and the f^i_{est} is the estimated value. The error is computed as $e^i_f = (f^i_{est} - f^i_{gt})/f^i_{gt}$. (b) is the error when estimating the camera height on 50 test images. The ground truth value of camera height h^i_{gt} ranges from 35 cm to 60 cm, and the estimated value is h^i_{est}. The error is computed as $e^i_h = h^i_{est} - h^i_{gt}$. (c) shows the error when estimating the plane normal on 50 test images. The ground truth normal is n^i_{gt} and the estimated value is n^i_{est}. The error is defined as $e^i_n = arccos(n^i_{est} n^i_{gt})$.

Processor Unit to implement the optimization procedure. In our implementation of the optimization function, the range of values for each unknown parameter is set as follows: i) plane normal has 20 discretized values for tilt direction from 15° to 17° and 5 discretized values for camera-rotation from −10° to 10°, ii) plane height has 20 discretized values from 30 cm to 80 cm for office dataset and from 1.5 m to 2 m for street dataset. iii) camera focal length has 20 discretized values from 0.8 to 1.25 fraction of the initial value of the camera focal length. The average optimization time for one 640×480 image is 0.2 seconds. Using the LabelMe dataset, we compare our algorithm with Hoiem et al. [16]. The comparison indicates that our method achieves competitive results in pruning out false positives

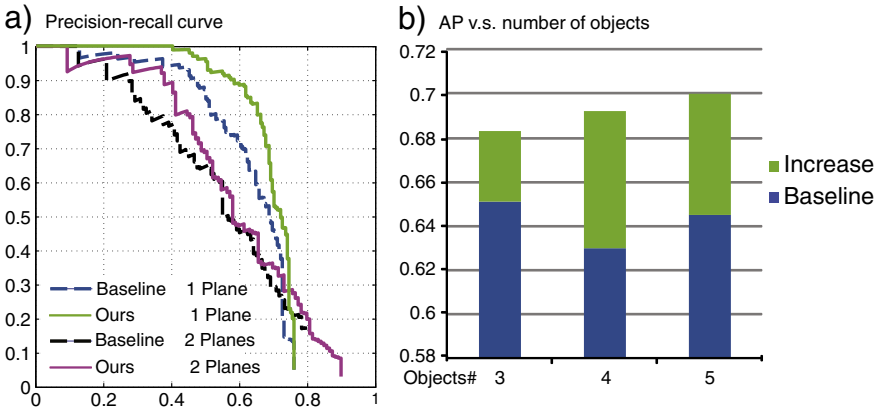

Fig. 9. Experimental results on our desk-top dataset. (a) reports precision-recall curves by the base line detector (dash) and our algorithm (solid). Precision-recall curves are shown for one and two planes separately. (b) reports the average precision as the number of objects increases.

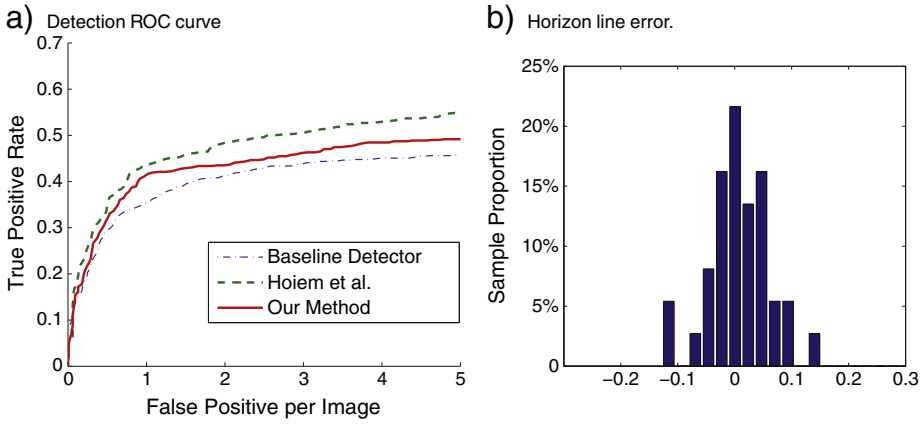

Fig. 10. Result on LabelMe dataset. (a) Car and Pedestrian detection. (b) The histogram of the horizontal vanishing line estimation error. The Y axis is the fraction of the number of testing images (samples) that have certain error.

and estimating layout properties such as the horizon line. We also show successful anecdotal results on a number of images downloaded from the web.

4.1. Desk-top scenario

We test our framework on a novel desk-top database [1] where ground truth information about the geometry of the scene is available. This dataset comprises three object categories (computer mouse, mug and stapler). Each image in the dataset portrays 3 to 5 object instances located at randomly selected positions and with random poses on one (or two) supporting plane(s) (Fig. 11). Training and testing sets contain 80 and 75 images respectively. For each image we have the available ground truth values for the camera focal length and the normal of the supporting plane in the camera reference system as well as the ground truth locations of the objects in the image. These are used for training the distance function (Eq. (8)) and for evaluating our algorithm's performance. We learn our modified version of the object detector and pose estimator in [33] on the 3-object category training set. We apply the learnt detector to the testing set and obtain a number of detected objects. This provides the baseline object detection result (e.g. "baseline" in Fig. 9(a) and (b)). For each detection we also estimate the azimuth and zenith pose of the object. Examples of detections are in Fig. 11. Among these detections we can find a number of false alarms. So

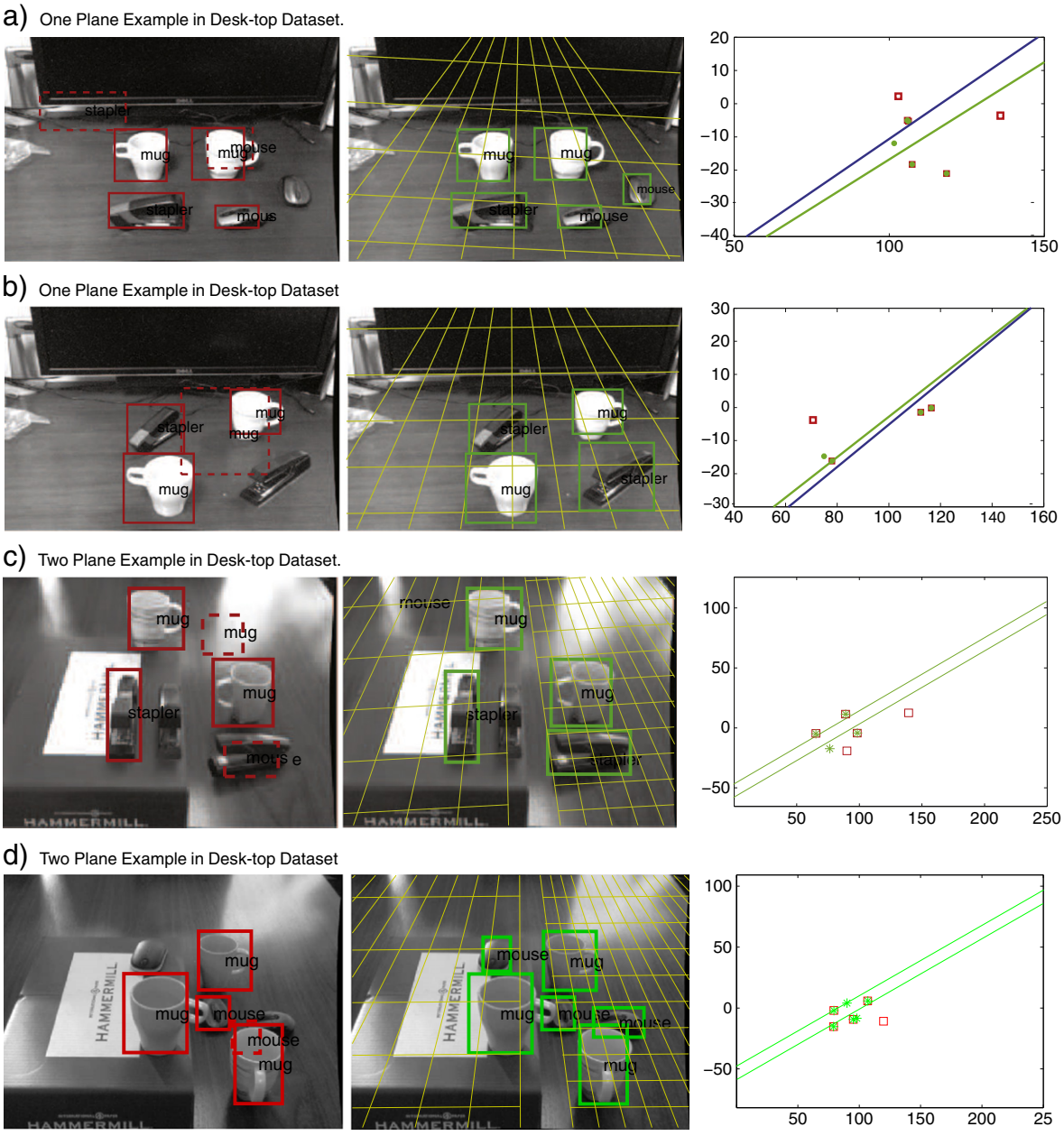

Fig. 11. Desk-top dataset: In each sub-figure we show the baseline detector results on the left; our algorithm's object detection and support plane estimation results in the middle; our algorithm's 3D scene layout reconstruction on the right. Baseline detection results are in red; dashed red boxes indicate false alarms. Our improved detection results are in green; dashed green boxes indicate false alarms. Our estimated supporting plane is superimposed in yellow. Notice that most of the supporting planes estimations are visually convincing. The 3D layout shows the side view of the 3D reconstructed scene (the camera is located at (0, 0) pointing to the right). The estimated supporting plane is in green and the ground truth supporting plane in blue. Green dots are the objects detected and recovered by our algorithm (in the 3D camera reference system); red squares are objects detected by the baseline detector. Notice that our algorithm works even when there are multiple supporting planes existing in a scene.

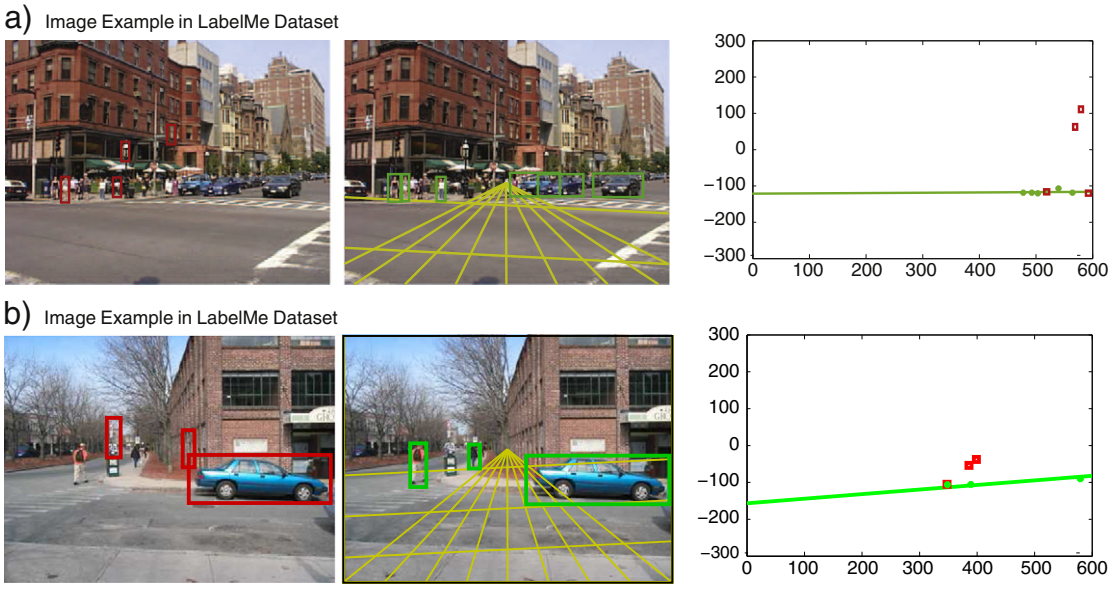

Fig. 12. LabelMe dataset: Please refer to the caption of Fig. 11 for the meaning of the figure notations.

we run our algorithm and use such detections (along with pose measurements) to jointly estimate the supporting plane normal, the camera focal length and the locations of the objects (among all detections returned by the detector) that are consistent with the estimated scene layout. Results are shown in Figs. 8 and 9. We tested our algorithm on images where one plane or two planes exist in the scene. Our testing set contains 50 images of one-plane case and 25 images of two-planes case. Fig. 9(a) shows the object detection precision-recall curve. In the one-plane case, the baseline detector average precision is 64% compared to 70% with our method. In the two-planes case, the baseline detector average precision is 56% compared to ours 61%. Furthermore, we evaluate the detection accuracy as function of the number of instances appearing in the scene per test image. We show our results in Fig. 9(b). The object detection performance improvement is obtained by using the estimated supporting plane to prune out false alarms and recover missed positives. The estimation of the supporting plane is affected by the observation noise (location and pose) associated to each object instance. As the number of observations increases, the contribution of the noise is averaged out which explains the reason the object detection performance increases with the number of instances.

4.2. Experiments on LabelMe dataset

We compare our algorithm with another state-of-the-art method that uses geometrical contextual reasoning for improving object detection rates and estimating scene geometric properties such as the horizon line [16]. We use the LabelMe database on cars and pedestrians to compare the algorithms. Since one necessary condition for our algorithm to work is that at least three objects coexist in the same image, we use a subset of the dataset provided by [16]. We remove images containing less than three instances (pedestrians or cars). We test our algorithm on 100 randomly selected images and compare our method with [16] by using the same baseline pedestrian and car detector as in [16]. Examples of detections are in Fig. 12. Fig. 10(a) compares the ROC curve for car and pedestrian detection by our algorithm to that of [16]. Fig. 10(b) shows the histogram of the relative error of our algorithm in estimating the horizontal vanishing line. Notice the median absolute error in estimating the horizontal vanishing line reported in [16] is 0.038. Detection rate and accuracy in estimating the horizon line are comparable between ours and [16]. However, notice that [16] heavily relies on: i) estimating surface geometry [28] by determining "ground", "vertical" and "sky" regions in the image; ii) assuming that the camera has a fixed distance from the ground plane (the distance is roughly the height of a person); iii) assuming that no multiple ground planes (at different heights) are present in the image. On the contrary, our algorithm: i) does not rely on estimating horizontal or vertical regions as it extracts spatial contextual information from the objects themselves (thus, our algorithm works even if the ground region is not visible at all); ii) doesnot assume fixed distance from the ground plane which can be located anywhere in the 3D space; iii) it works even if objects are supported by multiple planes located at different heights. For that reason our algorithm is particularly suitable to work in indoor settings where most of the assumptions of [16] are violated.

4.3. Anecdotal detections and reconstructions

We conclude this section by presenting a number of anecdotal examples. Fig. 13 shows joint detection and scene layout estimation on images taken from various sources including ETHZ [37] and the Internet.

5. Conclusions

We have presented a novel method that can jointly model object locations and supporting surfaces (planes) in the 3D space along with camera focal length in a single camera. We have modeled the problem of joint scene reconstruction and object recognition as the one of finding the set of parameters that maximizes the joint probability of detecting objects on several supporting planes. Experimental results have demonstrated the validity of our intuitions and assumptions. We see this work as a promising starting point for achieving coherent scene interpretation and object recognition. For instance, we believe that, by combining our approach with that of Hoeim et al. [16], the joint recognition-reconstruction paradigm may be further enhanced.

Acknowledgments

We acknowledge the support of NSF (Grant CNS 0931474) and the Gigascale Systems Research Center, one of six research centers funded under the Focus Center Research Program (FCRP), a Semiconductor Research Corporation entity.

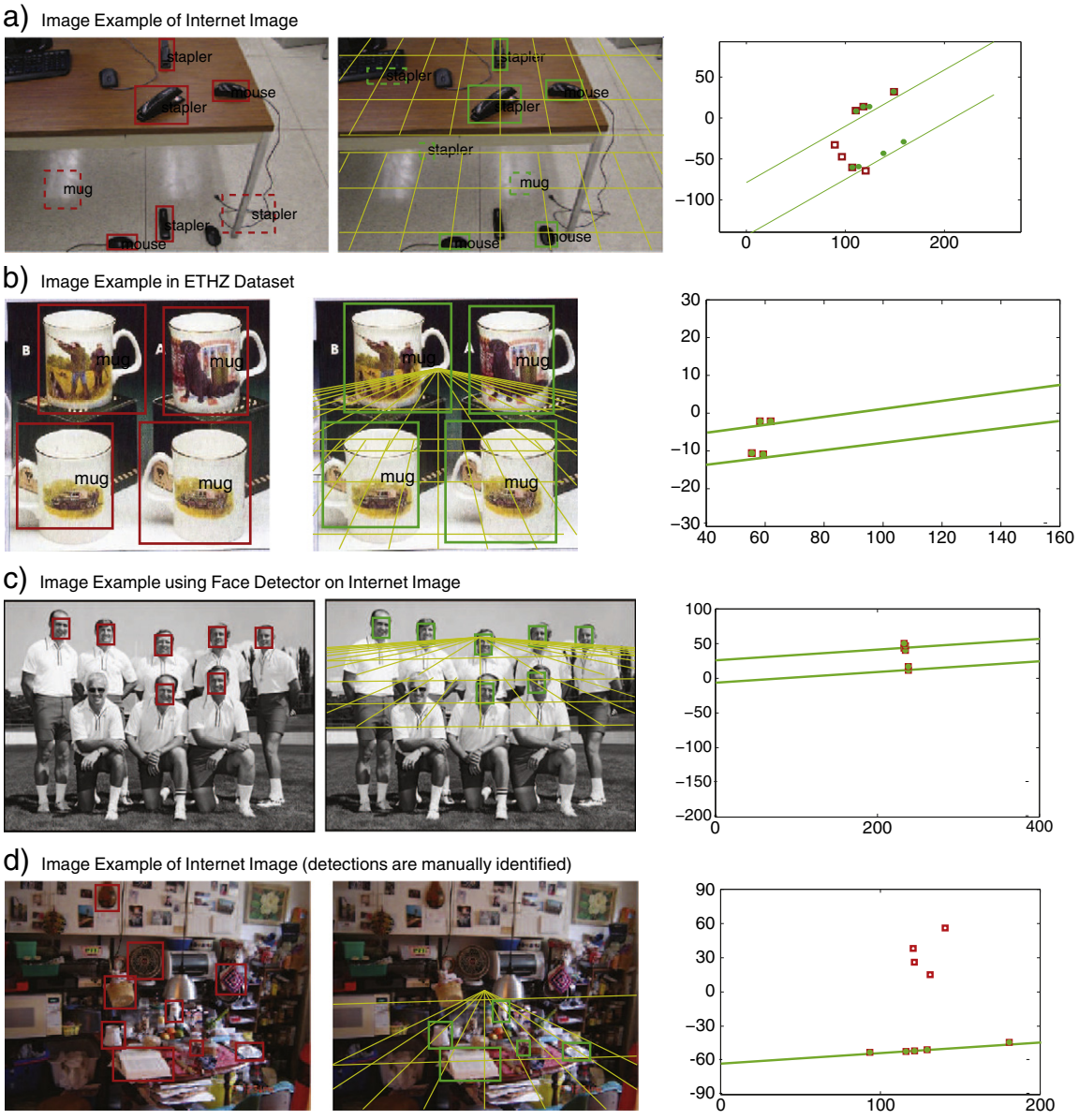

Fig. 13. Anecdotal scenarios: Please refer to the caption of Fig. 11 for the meaning of the figure notations. In (c), we use a detector to detect faces and use these (along with the fact that faces are estimated frontally) to estimate different hypothetical supporting planes. In (d), we show that our algorithm can potentially recover the supporting plane and perform contextual reasoning even when the scene is highly cluttered (here detections in red were manually identified, but successfully pruned out by our algorithm in green). This figure is best viewed in color.

References

[1] M. Sun, G. Bradski, B.-X. Xu, S. Savarese, Depth-encoded hough voting for coherent object detection, pose estimation, and shape recovery, ECCV, 2010.
[2] B.C. Russell, A. Torralba, K.P. Murphy, W.T. Freeman, Labelme: A database and web-based tool for image annotation, IJCV 77 (1–3) (2008) 157–173.
[3] S. Palmer, Vision science: photons to phenomenology, The MIT Press, 1999.
[4] I. Biederman, R. Mezzanotte, J. Rabinowitz, Scene perception: detecting and judging objects undergoing relational violations, Cognitive Psychology 14 (2) (1982) 143–177.
[5] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, CVPR, 2001.
[6] R. Fergus, P. Perona, A. Zisserman, Object class recognition by unsupervised scale-invariant learning, CVPR, 2003.
[7] P. Felzenszwalb, D. Huttenlocher, Pictorial structures for object recognition, CVPR, 2000.
[8] B. Leibe, B. Schiele, Scale invariant object categorization using a scale-adaptive mean-shift search, DAGM Annual Pattern Recognition Symposium, 2004.
[9] L. Fei-Fei, R. Fergus, A. Torralba, Recognizing and learning object categories, , 2007.
[10] Y. Ohta, Knowledge-based interpretation of outdoor natural color scenes, Pitman Publishing, Inc, 1985.
[11] H. Barrow, J. Tenenbaum, Recovering intrinsic scene characteristics from images, Computer Vision Systems, 1978.
[12] I. Biederman, On the semantics of a glance at a scene, in: M. Kubovy, J. Pomerantz (Eds.), Perceptual Organization, 1981, Ch. 8.
[13] R.A. Brooks, Model-based three dimensional interpretations of two dimensional images, IJCIA, 1981.
[14] D.A. Forsyth, J.L. Mundy, A. Zisserman, C.A. Rothwell, Using global consistency to recognise euclidean objects with an uncalibrated camera, CVPR, 1994.
[15] A.R. Hanson, E.M. Riseman, Visions: A computer system for interpreting scenes, Computer Vision Systems, 1978.
[16] D. Hoiem, A.A. Efros, M. Hebert, Putting objects in perspective, CVPR, 2006.
[17] V. Hedau, D. Hoiem, D. Forsyth, Recovering the spatial layout of cluttered rooms, ICCV, 2009.
[18] V. Hedau, D. Hoiem, D. Forsyth, Thinking inside the box: using appearance models and context based on room geometry, ECCV, 2009.
[19] S. Gould, R. Fulton, D. Koller, Decomposing a scene into geometric and semantically consistent regions, ICCV, 2009.
[20] L.-J. Li, R. Socher, L. Fei-Fei, Towards total scene understanding: classification, annotation and segmentation in an automatic framework, CVPR, 2009.
[21] G.J. Brostow, J. Shotton, J. Fauqueur, R. Cipolla, Segmentation and recognition using structure from motion point clouds, ECCV, 2008.
[22] D.C. Lee, M. Hebert, T. Kanade, Geometric reasoning for single image structure recovery, CVPR, 2009.
[23] E.B. Sudderth, A. Torralba, W.T. Freeman, A.S. Willsky, Depth from familiar objects: a hierarchical model for 3d scenes, CVPR, 2006.

[24] N. Cornelis, B. Leibe, K. Cornelis, L. Gool, 3d urban scene modeling integrating recognition and reconstruction, IJCV 78 (2–3) (2008) 121–141.
[25] A. Saxena, M. Sun, A.Y. Ng, Make3d: learning 3d scene structure from a single still image, PAMI 31 (5) (2009) 824–840.
[26] A. Gupta, A. Efros, M. Hebert, Blocks world revisited: image understanding using qualitative geometry and mechanics, ECCV, 2010.
[27] N. Payet, S. Todorovic, Scene shape from texture of objects, CVPR, 2011.
[28] D. Hoiem, A.A. Efros, M. Hebert, Geometric context from a single image, ICCV, 2005.
[29] D. Hoiem, A.A. Efros, M. Hebert, Closing the loop on scene interpretation, CVPR, 2008.
[30] S.Y. Bao, M. Sun, S. Savarese, Toward coherent object detection and scene layout understanding, CVPR, 2010.
[31] S. Savarese, L. Fei-Fei, 3d generic object categorization, localization and pose estimation, ICCV, 2007.
[32] J. Liebelt, C. Schmid, K. Schertler, Viewpoint-independent object class detection using 3d feature maps, CVPR, 2008.
[33] H. Su, M. Sun, L. Fei-Fei, S. Savarese, Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories, ICCV, 2009.
[34] M. Ozuysal, V. Lepetit, P. Fua, Pose estimation for category specific multiview object localization, CVPR, 2009.
[35] M. Arie-Nachimson, R. Basri, Constructing implicit 3d shape models for pose estimation, ICCV, 2009.
[36] A. Farhadi, M.K. Tabrizi, I. Endres, D.A. Forsyth, A latent model of discriminative aspect, ICCV, 2009.
[37] V. Ferrari, L. Fevrier, F. Jurie, C. Schmid, Groups of adjacent contour segments for object detection, PAMI 30 (1) (2008) 36–51.