Toward Coherent Object Detection And Scene Layout Understanding

Sid Ying-Ze Bao Min Sun Silvio Savarese
Dept. of Electrical and Computer Engineering, University of Michigan at Ann Arbor, USA
{yingze,sunmin,silvio}@eecs.umich.edu

Abstract

Detecting objects in complex scenes while recovering the scene layout is a critical functionality in many vision-based applications. Inspired by the work of [18], 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 the 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 a number of datasets (a novel in-house dataset and label-me[28] on car and pedestrian) 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 [27]. The recognition of objects with the estimate of their location, scale and pose 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 [33, 10, 9, 21, 8], regardless of the geometrical contextual cues. It is indeed true that objects can be in general rec-

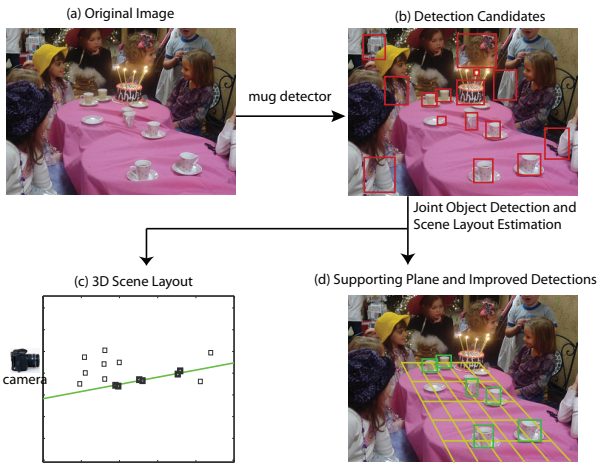


Figure 1. A conceptual illustration of the flowchart of our algorithm. (a) Original input image with unknown camera parameters; (b) Detection candidates provided by a baseline "mug" detector; (c) The 3D layout. The figure shows the side view of the 3d reconstructed scene. The supporting plane is shown 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.

ognized even if 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 [25, 2, 3, 4, 13, 15], and recently revitalized in [18, 14, 22, 5, 20, 32, 6, 30]. In [18], the process of detecting objects in a complex scene is enhanced by introducing the geometrical contextual information of the scene layout [17] (e.g., vertical surfaces, ground horizontal planes, etc). More explicit reasoning on the relationship between supporting planes and objects has been investigated in [19, 16]. In this work we too advocate the importance of geometrical contextual reasoning for object recognition and focus on demonstrating that the geometrical contextual cues provided by object location and pose can be used, in turn, to reinforce the detection and prune out false alarms. 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 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. At this end, we leverage on recent methods for detecting multi-category objects and estimating their poses accurately from a single image [29, 23, 31, 26, 1, 7]. Unlike [18], where contextual information was provided by the explicit estimation of surface orientation using the geometric context operator [17], we 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. This is conceptually similar to previous work on shape from texture [12, 24] where each texture element can be interpreted as an object. Unlike [18], we only hypothesize zero skew and unit pixel ratio camera model with unknown focal length, and do not make assumptions on the camera-to-ground distance or camera view angle. This allows our algorithm to work well on both outdoors and indoors scenes.

The main contributions of our work include: 1. A novel representation that can jointly model 3D objects location and 3D supporting surfaces (planes) from the observations from 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 (Sec.2). 2. A relationship that connects the 2D image observation of object location and zenith angle pose with the 3D supporting planes' normals and 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 objects do 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 (Sec.3.1). 3. A framework that uses the above relationships and a probabilistic formulation to jointly detect objects (so as to reduce false alarm and false negative rates) and recover (within the camera reference system) objects' 3D locations, the 3D supporting planes, and the camera focal length parameter. All of above outcomes are merely based on one single semi-calibrated image (Sec.2). Experimental evaluation on a number of datasets (a novel in-house dataset and the car and pedestrian Label-Me dataset[28]) demonstrates our theoretical claims (Sec.4).

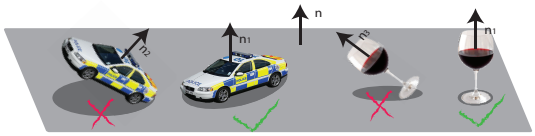


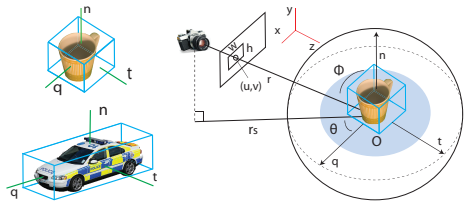
Figure 2. If the normal of a plane is n, objects lying on such plane tend to share the same normal direction $n_1//n$. Objects whose normal is not parallel to n (e.g. n_2 and n_3) are unlikely to sit on that supporting plane.

2. 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 object's 3D locations and poses in camera reference system; ii) their supporting plane's 3D orientation and location in camera reference system; iii) camera focal length parameter. 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 occupy unstable poses. Furthermore, if multiple supporting planes co-exist in one image,



(a) Object in 3D (b) Object's pose in view sphere

Figure 3. (a): Three perpendicular directions characterize the pose of an 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.

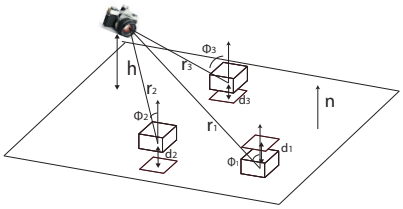


Figure 4. Geometrical relationships of ϕ, r, d, h and n

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.

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).

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-down direction n should be equal to supporting plane's normal. 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 connecting O and the camera center (Fig.3(b)). Let *zenith* angle ϕ be the angle between the ray r and n (Fig.3(b) and Fig.4). Define *azimuth* angle θ be the angle formed by object's frontal vector q and a vector r_s that 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 $\widehat{\phi}$ the estimated zenith pose consistent with the underlying surface layout. We will explain in details how to compute $\widehat{\phi}$ and measure ϕ in Sec.3.1.

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 (Fig.3(b) and Fig.7).

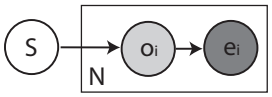


Figure 5. Graphic model of conditional independence for supporting plane parameters and detection result, where o_i is partially observed and e_i fully observed. Details in Sec.2.2.

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

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 is wrongly indicating the presence of an object from incorrect categories 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 truth 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 plane. The advantage 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 is 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

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

Since the probability of a bounding box (given only geometrical constraint), regardless of whether it is foreground or background, can be treated as a constant, we have $p(o_i|S) = p(t_i|m_i, S)p(m_i|S) \propto p(t_i|m_i, S)$. 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 prior of scene information, which may be considered to uniformly distribute within a reasonable range of n, h and f.

$\mathbf{p(e|t, m)}$ is related to the detection result's confidence. In this paper we assume $p(m, t)$ and $p(e)$ satisfy uniform distribution, therefore $p(e|t, m) = p(t, m|e)p(e)/p(t, m) \propto p(t, m|e)$, 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 true positive, given the candidate detection m and 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, object's 3D coordinates O, plane's normal n, and camera-to-plane distance h (Fig.4). In Sec.3 we will explain in details these relationships. Here, we formulate

$$p(t=1|m,S) = p(t=1|d)p(t=1|\phi-\hat{\phi}) \qquad (1)$$

In other words, 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 $p(\phi-\hat{\phi}) = N(\phi-\hat{\phi};0,\sigma_\phi)$, where $\hat{\phi}$ is the inferred zenith and ϕ is the measured zenith from image. Thus, $t_i = 1$ is possible only if the scale of the bounding box and the predicted pose of the detection are consistent with the supporting plane.

To sum up, in our problem, unknowns are $\{t_i\}, S$, and measurements are $\{m_i\}$ and $\{p(e_i|o_i)\}$ given by detector. 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. The problem is equivalent to find S and $\{t_i\}$ via the following optimization problem:

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

2.3. Solving the Optimization

In this section we solve the optimization problem of Eq.(2) in Sec.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 fix value of S, the value of each terms 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 the function of S

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

Eq.(2) is equivalent to

$$\arg\max_{S}[\ln p(S) + z(S)] \qquad (3)$$

We propose to solve Eq.(3) by searching on a large but finite set of S to find the optimal S^*. This can be computed almost in real-time by CUDA paralleling programming.

2.4. Extension to Multiple Planes

The above approach solves single plane problem by obtaining the highest log likelihood score. This approach can be extended to handle the case of multiple supporting planes by carrying it on iteratively. First, we employ this approach to find the best plane configuration S and then we remove the corresponding $\{t|t=1\}$ given S. Next, the algorithm runs again for the remaining detection candidates. If the number of planes K is known before-hand, the algorithm ceases after desired number of planes are found. If the number of planes K is unknown, we stop the algorithm after the increment of the overall log likelihood falls under a threshold. Notice that, since all the planes will have the same normals, the "at least three objects" requirement (Sec.3.1) is no longer necessary for all the other planes except the first one.

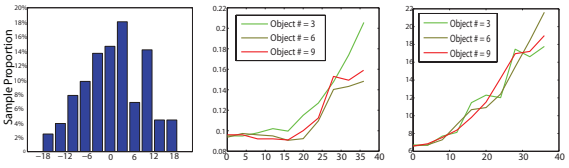
3. Relating Camera Measurements and Supporting Planes

In this section we explain in details the computation of $p(t|m,S)$ in Sec.2.2. As Eq.(1) We have decomposed $p(t=1|m,S) = p(t=1|d)p(t=1|\phi-\hat{\phi})$. In Sec.3.1 we will derive the relationship among the estimated zenith angle pose ϕ_i of an object in the image plane, the supporting plane normal n and camera focal length f. We show that by measuring ϕ_i of 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 Sec.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 $(\widetilde{u},\widetilde{v},1)$ 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 zeros-skew. These are reasonable assumptions that hold in most practical cases. By translating any point in image plane by $(u_i,v_i) = ((\widetilde{u}_i,\widetilde{v}_i) - (u_0,v_0))^T$, we write the camera intrinsic parameter matrix as $K = diag(f,f,1)$.

Let the line of sight r_i be the line 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 supporting plane's normal in camera coordinates. s_i and n are shown is Fig. 4. Notice n



(a) (b) (c)

Figure 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 have certain error. The X axis is error in degree. (b)(c) Error analysis of equation 5. X axis is the variance of Gaussian noise in degree. (b) Y axis is $e_f = |(f - \widehat{f})/f|$. (c) Y axis is $e_n = |arccos(n \cdot \widehat{n})|$ in degree. This figure is best viewed in color.

must satisfy $n_1^2 + n_2^2 + n_3^2 = 1$. Then we have

$$(u_i, v_i, 1) \begin{pmatrix} n_1 \\ n_2 \\ n_3 f \end{pmatrix} = -\cos\phi_i \sqrt{u_1^2 + v_1^2 + f^2} \quad (4)$$

Hence, using Eq.(4), the key term $\widehat{\phi}$ in Eq.(1) can be computed given n_1, n_2, n_3, and f, i.e. part of S.

Measure Zenith Angle From Image. It is clear that our formulation relies on the measurement of object zenith angles in the image plane. Recently, a number of techniques such as [31, 23, 29, 1] have been proposed to estimate object pose from single images. We used an adapted version of [31] to measure zenith angles ϕ from the image. We have found that this adapted pose detector yields satisfactory results when validated on our in-house dataset (Fig.6). The standard deviation of zenith angle measurement is 8.4°.

Estimate 3D Plane Orientation via Object Zenith Angles. In this section, we show that supporting plane's normal and camera focal length can be estimated from the object's zenith angle ϕ, location (u, v) from just one single image. If a total number of N measurements ϕ_i, u_i, v_i $(i = 1...N)$ are available, following Eq.(4) we obtain:

$$\begin{bmatrix} u_1 & v_1 & f \\ u_2 & v_2 & f \\ u_3 & v_3 & f \\ & \vdots & \\ u_N & v_N & f \end{bmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} = \begin{pmatrix} -\cos\phi_1 \sqrt{u_1^2 + v_1^2 + f^2} \\ -\cos\phi_2 \sqrt{u_2^2 + v_2^2 + f^2} \\ -\cos\phi_3 \sqrt{u_3^2 + v_3^2 + f^2} \\ \vdots \\ -\cos\phi_N \sqrt{u_N^2 + v_N^2 + f^2} \end{pmatrix} \quad (5)$$

This equation allows us to solve $\{f, n_1, n_2, n_3\}$ from objects' measurements ϕ_i, u_i, v_i (i=1...N) of just one single image. The following proposition [1] gives conditions for the existence of a solution of Eq.(5).

Proposition 1: Equation (5) admits one or at most two non-trivial solution of $\{f, n_1, n_2, n_3\}$ 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.(5) has infinite solutions.

[1] Please see authors' project webpage for the complete proof of this proposition

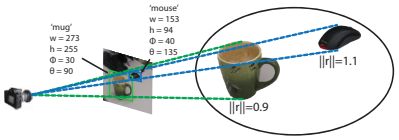


Figure 7. An illustration of a detected object and its corresponding 3D pose. Given object's image bounding box and estimated pose, its distance to the camera can be estimated using the procedure in Sec.3.2.

Equation (5) guarantees that as long as at least 3 objects do not lie on the same line in the image, it is possible to express camera focal length and supporting planes normal as a function of object locations and zenith pose measurements in the image. Notice that this equation does not assume 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).

Error Analysis. We use numerical simulation to analyze the robustness of the estimation of f and n by solving Eq.(5) as the 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 is simulated. Then the corresponding object's zenith angle ϕ_i is computed by Eq.(4). Next we add Gaussian noise w of variance σ to the object's zenith $\widetilde{\phi}_i = \phi_i + w$. Consequently, given $\{\widetilde{\phi}_i\}$ and $\{u_i, v_i\}$, we compute plane's normal $\widehat{n}$ and focal length $\widehat{f}$, by solving Eq.(5). Since Eq.(5) cannot be solved in closed form, we employ the Levenberg-Marquardt method to solve it. Fig.6(b) and Fig.6(c) shows the mean value of the absolute errors v.s. object's number and noise level: Fig.6(b) is $e_f = |(f - \widehat{f})/f|$ and Fig.6(c) is $e_n = |arccos(n \cdot \widehat{n})|$. These plots relate the accuracy in estimating n and f as function of the errors in measuring the zenith ϕ. Given that our detector returns ϕ with an error of about 10° (Fig.6(a)), Fig.6(b) and Fig.6(c) show that corresponding error in estimating n and f is still reasonable. Notice that we do not use Eq.(5) while solving the optimization in Sec.2.3.

3.2. Locating Objects in 3D

In this section, we explain the relationship between S and d, starting from locating objects in the 3D camera references 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 on the 3D objects' real size, 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 $\|r\|$ as a linear combination of $\frac{1}{w}$ and $\frac{1}{h}$

$$\|r\| \simeq (\alpha(\theta, \phi, c)\frac{1}{w} + \beta(\theta, \phi, c)\frac{1}{h}) \cdot f \quad (6)$$

where α and β are functions of object's pose and class la-

bel and f is the focal length. α and β are related to physical 3D shape of the object category. A more precise modeling of such 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 coordinate can be estimated in the camera coordinates

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

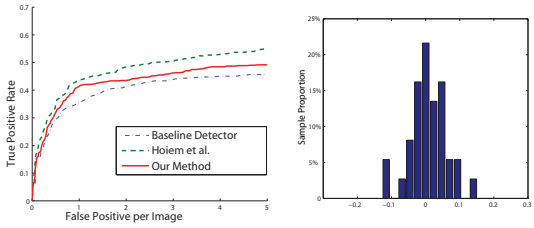
This allows us to relate the 3D coordinates of candidate object O, the supporting plane parameter (n, h), the distance d between object and the supporting plane (Fig.4) as $d = O^T n + h$.

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 as well as on the LabelMe outdoor pedestrian and cars dataset. We use Graphic Processor Unit to implement the probability optimization. In our indoor testing set, the average time to process one 640×480 image is 0.2 seconds. A benchmark comparison with [18] indicates that our method achieves competitive results in pruning out false positives 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 indoor (desktop) database where ground truth information about the geometry of the scene is available. Our dataset comprises three object categories (computer mouse, mug or stapler). Each image in the dataset portrays from $3-5$ object instances located at randomly selected positions and with random poses on one (or two) supporting plane(s) (Fig.10). Training and testing sets contain 80 and 50 images respectively. For each image we have the available ground truth values of the camera focal length and the supporting plane normal in the camera reference system as well as the ground truth locations of the objects in the image. These are used in training the distance function (Eq.(6)) and in evaluating our algorithm performance. We learn our modified version of the object detector and pose estimator in [31] on the 3-object category training set. We apply the learnt detector to the testing set and obtain a number of detected objects. For each detection we also estimate the azimuth and zenith pose of the object. Examples of detections are in Fig.10. Among these detections we can find a number of false alarms. So we run our algorithm and use such detections (along with pose measurements) to jointly estimate the supporting plane normal, camera focal length and object locations (among all detections returned by the detector) that are consistent with the estimated scene layout. Results are shown in Fig.8. Notice that our algorithm improves the average precision. In 1 plane case, the baseline detector average precision is 64% compared to ours 70%; in 2 planes case, the baseline detector average precision is 56% compared to ours 61%. Critically, our algorithm recovers the scene layout with good level of accuracy. To our best knowledge, our algorithm is the first achieving this kind of results without using external geometrical surface context detectors [17] with minimal assumptions on the camera model. Furthermore, we evaluate the detection accuracy as function of the number of instances appearing in the scene per test image. We notice that performances increase as the number of instances grow.

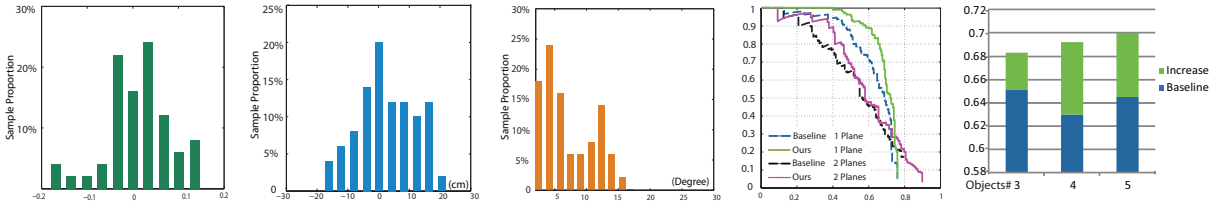


(a) Detection ROC curve (b) Horizon line error.

Figure 9. 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.

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 geometrical properties such as the horizon line [18]. 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 remove from the dataset images containing less than three instances (either pedestrians or cars). We test our algorithm on 100 randomly selected images and compare our method with [18] by using the same baseline pedestrian and car detector as in [18]. Fig.9(a) compares the ROC curve for car and pedestrian detection produced by our algorithm and [18]'s. Fig.9(b) shows the histogram of the relative error of our algorithm to estimate the horizontal vanishing line, while [18] reports their median absolute error of horizontal vanishing line estimation is 0.038. Notice that detection rate and accuracy in estimating the horizon line are comparable. Also, note that [18] heavily relies on: i) estimating surface geometry [17] 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) as-



(a) Focal Length Est. Error (b) Height Est. Error (c) Plane Normal Est. Error (d) PR Curve (e) AP v.s. Object #

Figure 8. Experimental results on our desk top dataset. For histogram (a) (b) and (c), the Y axis is the proportion of the number of testing samples (images) associated to an error interval (X axis). (a) shows focal length estimation error for 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) shows camera height estimation error for 50 test images. The ground truth value of camera height h^i_{gt} ranges from 35cm to 60cm, and the estimated value is h^i_{est}. The error is computed as $e^i_h = h^i_{est} - h^i_{gt}$. (c) shows plane normal direction estimation error for 50 test images. The ground truth normal is n^i_{gt} and the estimated value is n^i_{est}. The error of normal angle is defined as $e^i_n = arccos(n^i_{est} n^i_{gt})$. (d) plots precision recall curves by the base line detector (dash) and our algorithm (solid). We show the PR curves for one and two planes separately. (e) plots the trends of average precision with the increasing number of objects on one plane dataset.

suming 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 object themselves (thus, our algorithm works even if the ground region is not visible at all); ii) does not assume fixed distance from the ground plane which can be located everywhere 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 in [18] are violated. Notice that recent work by [16] has an interesting take on this topic.

4.3. Anecdotal Detections and Reconstructions

We conclude this section by presenting a number of anecdotal examples. The last two rows of Fig.10 show joint detection and scene layout estimation on images taken from various sources including ETHZ [11] and 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 corresponding observations 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 combining our approach with [18]'s 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.

References

[1] M. Arie-Nachimson and R. Basri. Constructing implicit 3d shape models for pose estimation. In *ICCV*, 2009.

[2] H. Barrow and J. Tenenbaum. Recovering intrinsic scene characteristics from images. *Computer Vision Systems*, 1978.

[3] I. Biederman. On the semantics of a glance at a scene. In M. Kubovy and J. Pomerantz, editors, *Perceptual Organization*, chapter 8. 1981.

[4] R. A. Brooks. Model-based three dimensional interpretations of two dimensional images. In *IJCIA*, 1981.

[5] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, 2008.

[6] N. Cornelis, B. Leibe, K. Cornelis, and L. Gool. 3d urban scene modeling integrating recognition and reconstruction. *IJCV*, 78(2-3):121–141, 2008.

[7] A. Farhadi, M. K. Tabrizi, I. Endres, and D. A. Forsyth. A latent model of discriminative aspect. In *ICCV*, 2009.

[8] L. Fei-Fei, R. Fergus, and A. Torralba. Recognizing and learning object categories. *CVPR Short Course*, 2007.

[9] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. In *CVPR*, 2000.

[10] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR*, 2003.

[11] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *PAMI*, 30(1):36–51, 2008.

[12] D. A. Forsyth. Shape from texture and integrability. In *ICCV*, 2001.

[13] D. A. Forsyth, J. L. Mundy, A. Zisserman, and C. A. Rothwell. Using global consistency to recognise euclidean objects with an uncalibrated camera. In *CVPR*, 1994.

[14] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *ICCV*, 2009.

[15] A. R. Hanson and E. M. Riseman. Visions: A computer system for interpreting scenes. *Computer Vision Systems*, 1978.

[16] V. Hedau, D. Hoiem, , and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *ICCV*, 2009.

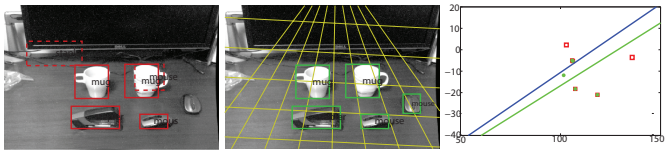
[17] D. Hoiem, A. A. Efros, and M. Hebert. Geometric context from a single image. In *ICCV*, 2005.

[18] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. In *CVPR*, 2006.

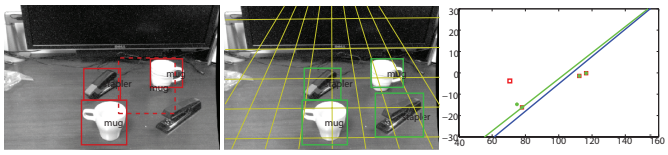
[19] D. Hoiem, A. A. Efros, and M. Hebert. Closing the loop on scene interpretation. In *CVPR*, 2008.

[20] D. C. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *CVPR*, 2009.

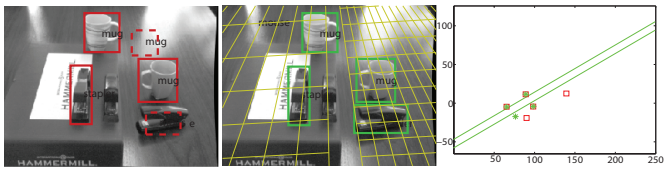
[21] B. Leibe and B. Schiele. Scale invariant object categorization using a scale-adaptive mean-shift search. In *DAGM Annual Pattern Recognition Symposium*, 2004.



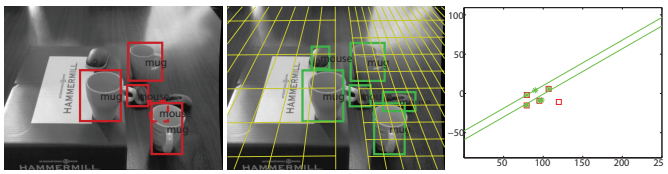
(a) One Plane Example in Our Dataset.



(b) One Plane Example in Our Dataset

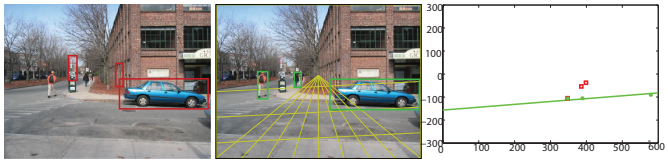


(c) Two Plane Example in Our Dataset.

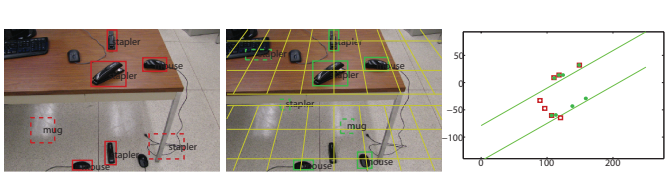


(d) Two Plane Example in Our Dataset

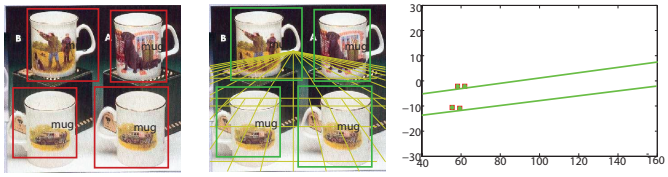
(e) Image Example in LabelMe Dataset



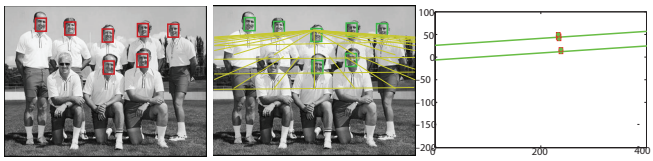
(f) Image Example in LabelMe Dataset



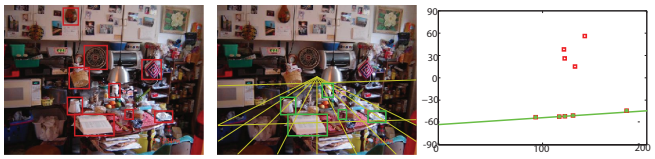
(g) Image Example of Internet Image



(h) Image Example in ETHZ Dataset



(i) Image Example using Face Detector on Internet Image



(j) Image Example of Internet Image (detections are manually identified)

Figure 10. 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 alarm. 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 on $(0, 0)$ pointing toward 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 if there are multiple supporting planes (two plane image in our dataset; ETHZ dataset [11]); The last two rows show results in anecdotal scenarios. On bottom left, 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; On bottom right, 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.

[22] L.-J. Li, R. Socher, and L. Fei-Fei. Towards total scene understanding:classification, annotation and segmentation in an automatic framework. In *CVPR*, 2009.

[23] J. Liebelt, C. Schmid, and K. Schertler. Viewpoint-independent object class detection using 3d feature maps. In *CVPR*, 2008.

[24] J. Malik and R. Rosenholtz. Computing local surface orientation and shape from texture for curved surfaces. *IJCV*, 23(2):149–168, 1997.

[25] Y. Ohta. *Knowledge-based interpretation of outdoor natural color scenes*. Pitman Publishing, Inc., 1985.

[26] M. Ozuysal, V. Lepetit, and P. Fua. Pose estimation for category specific multiview object localization. In *CVPR*, 2009.

[27] S. Palmer. *Vision Science: Photons to Phenomenology*. The MIT Press, 1999.

[28] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database and web-based tool for image annotation. 2005.

[29] S. Savarese and L. Fei-Fei. 3d generic object categorization, localization and pose estimation. In *ICCV*, 2007.

[30] A. Saxena, M. Sun, and A. Y. Ng. Make3d: Learning 3d scene structure from a single still image. *PAMI*, 31(5):824–840, 2009.

[31] H. Su, M. Sun, L. Fei-Fei, and S. Savarese. Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories. In *ICCV*, 2009.

[32] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky. Depth from familiar objects: A hierarchical model for 3d scenes. In *CVPR*, 2006.

[33] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, 2001.