Joint Semantic Segmentation and 3D Reconstruction from Monocular Video

Abhijit Kundu, Yin Li, Frank Daellert, Fuxin Li and James M. Rehg

Georgia Institute of Technology, Atlanta, USA

Abstract. We present an approach for joint inference of 3D scene structure and semantic labeling for monocular video. Starting with monocular image stream, our framework produces a 3D volumetric semantic + occupancy map, which is much more useful than a series of 2D semantic label images or a sparse point cloud produced by traditional semantic segmentation and Structure from Motion(SfM) pipelines respectively. We derive a Conditional Random Field (CRF) model defined in the 3D space, that jointly infers the semantic category and occupancy for each voxel. Such a joint inference in the 3D CRF paves the way for more informed priors and constraints, which is otherwise not possible if solved separately in their traditional frameworks. We make use of class specific semantic cues that constrain the 3D structure in areas, where multiview constraints are weak. Our model comprises of higher order factors, which helps when the depth is unobservable. We also make use of class specific semantic cues to reduce either the degree of such higher order factors, or to approximately model them with unaries if possible. We demonstrate improved 3D structure and temporally consistent semantic segmentation for difficult, large scale, forward moving monocular image sequences.

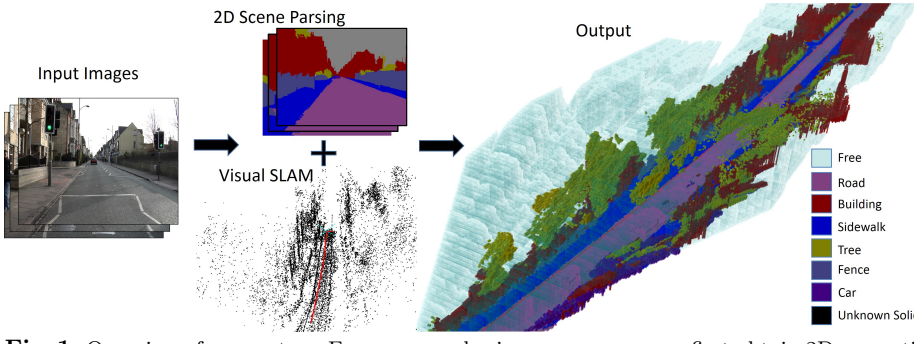

Fig. 1. Overview of our system. From monocular image sequence, we first obtain 2D semantic segmentation, sparse 3D reconstruction and camera poses. We then build a volumetric 3D map which depicts both 3D structure and semantic labels.

1 Introduction

To successfully navigate and perceive the 3D world, a robot needs to infer both its own position and information of the 3D environment. Vision-based Simultaneous Localization and Mapping (SLAM) estimates the location of the robot while incrementally building a map of the environment. However, SLAM only reveals

the structural information of the scene and the result is limited to a sparse 3D point cloud. Scene parsing, on the other hand, labels each pixel in an image or video with object categories (e.g. *Tree*, *Road*), thus providing semantic only information of the scene. But in many applications such as autonomous driving, it is important to understand both the structural and semantic information of the surroundings. In this paper, we propose a joint 3D reconstruction and scene parsing system from a fast forward-moving monocular camera.

Autonomous driving applications often involve fast forward-moving cameras. In these cases, multi-view stereo could fail due to textureless surfaces and/or low parallax, and the visual SLAM pipeline for a monocular camera only provides a very sparse set of 3D measurements. Previous work on joint reconstruction and scene parsing [9,26] require dense depth measurements and cannot accommodate to this problem.

Lifting the requirement of dense depth measurements, our input contains only sparse 3D point cloud but dense semantic labels on each pixel of each frame, the latter can be obtained through evaluating a scene parsing engine (e.g. [20]) on all the frames. We use category-specific sensor models to enhance the depth estimates, especially when no direct depth information is available. On the other hand, the knowledge of unoccupied space from successive camera positions help to reduce a lot of 3D structural ambiguities, as well as to improve structural estimates along weakly supported surfaces [12], where only vague structural information is available.

The 3D scene is represented in the form of 3D cubic subvolumes (voxel) along with per-voxel semantic labels (see Fig.1). The voxel labels include both solid semantic categories (e.g. *Car*) and *Free*, thus capturing both occupancy and semantic information in a single coherent discrete label space. We model the problem of labeling of all observable voxels with a higher order Conditional Random Field (CRF) in the 3D space. Inference of the CRF model in 3D allows for incorporating more realistic scene constraints and priors, such as 3D object support. Besides, full temporal coherency of the semantic labels is inherent to our 3D representation, because our 2D scene parsing is simply the projection of 3D semantic reconstruction to different camera positions. This representation is efficient and compact with an Octree data structure where unused voxels in the 3D map remain uninitialized and require minimal storage and computation.

Our method is applicable to popular monocular sequences like Camvid [4] which has only seen 2D segmentation results till now. Besides, our framework is flexible and can be easily extended to other sensors like laser or stereo cameras. It is quite efficient compared to standard multi-view stereo pipelines and still properly deals with noisy measurements and uncertainty. Thus, our method could find immediate use in many applications like autonomous robot navigation.

3D geometric information plays an important role in 2D semantic segmentation [2,27,19,6]. For example, Brostow et al. [2] incorporate sparse SfM features with 2D appearance features for each frame, and demonstrated its advantage over 2D appearance features alone. Ladicky et al. [19] propose a joint optimization of dense stereo and semantic segmentation for every frame. However,

temporal consistency of the segmentation is not considered in their methods. Several recent attempts [6,24,31] have addressed temporal continuity, either by pre-processing with supervoxel-based video segmentation [31], or by additional higher order potentials that enforce label consistency among projections of the same 3D point [6]. Still, most of these methods run in the 2D image space only. Our volumetric representation performs inference in 3D and achieve full temporal coherency without additional cost.

Semantic segmentation can be used to estimate 3D information [22,10,25]. For example, Liu et al. [22] guide the 3D reconstruction from a single image using semantic segmentation. Depth from semantics, though not as reliable as the SfM or multi-view stereo, has its own strengths: (1) it is complementary to the traditional geometric approaches; (2) it offers a potential denser depth measurement than SfM; (3) it is applicable for a larger range of sceneries than multi-view stereo. For a fast forward-moving monocular camera, the SfM gives very sparse point cloud and the multi-view stereo fails due to low parallax, whereas we can still rely on segmentation results.

The most relevant work are [9,26] who have independently proposed methods for simultaneous semantic segmentation and 3D reconstruction. However, both of these methods require dense depth measurements. Dense depth maps allow them to make relatively restrictive assumptions, e.g. Haene et al. [9] consider every pixel with missing depth as *Sky*. These assumptions do not hold in case of fast forward-moving monocular camera, where we only have a very sparse point cloud from SfM. Unlike [26], we propose a joint optimization scheme of both semantic segmentation and 3D reconstruction. And unlike [9], we use semantic category specific sensor models to estimate the depth as much as possible, instead of simply inserting *Free* labels for voxels with missing depth.

We explicitly model *Free* space. For applications like autonomous driving, *Free* space information is directly used in higher level tasks like path planning. Also, *Free* space provides cues to improve 3D reconstruction, especially along weakly supported surfaces [12] which is very common with forward moving cameras in urban scenes. In our framework, the *Free* space information from other cameras helps to reduce ambiguities in 3D structure.

This paper makes the following contributions:

- From a fast forward-moving monocular camera, we introduce a novel higher order CRF model for joint inference of 3D structure and semantics in a 3D volumetric model. The framework does not require dense depth measurements and efficiently utilize semantic cues and 3D priors to enhance both depth estimation and scene parsing.
- We present a data-driven category-specific process for dynamically instantiating potentials in the CRF. Our method performs tractable joint inference of 3D structure and semantic segmentation in large outdoor environments.
- We present results on challenging forward-moving monocular sequences such as CamVid and Leuven which demonstrate the value of our approach. The results have shown improved temporal continuity in scene parsing as well as improved 3D structure.

2 Problem Formulation and Notation

We are interested in the 3D map $\mathcal{M}$ comprising of several sub-volumes $m_i \in \mathcal{M}$. Where each m_i is a categorical random variable corresponding to voxel i, that can be either *Free* or one of the **solid** semantic objects like *Road*, *Building*, *Tree*, etc. For example in the Camvid [4] dataset, we used a 9 dimensional label space $\mathcal{L}_\mathcal{M} = \{$*Free*, *Road*, *Building*, *Sidewalk*, *Tree*, *Fence*, *Person*, *Car*, *UnknownSolid*$\}$. Note that this joint label space, $\mathcal{L}_\mathcal{M}$ is mutually exhaustive and is different from the label space $\mathcal{L}_\mathcal{I}$ of 2D image level semantic categories. For example there is no *Sky* in $\mathcal{L}_\mathcal{M}$, a common state used in 2D image scene parsing. Choosing this label space $\mathcal{L}_\mathcal{M}$ allows us to do the joint inference of both semantic category and and 3D structure of the scene with a single random variable per voxel.

Each pixel location $x \in \Omega$ in the images is a source of potential measurement, where $\Omega = \{1..h\}\times\{1..w\}$, with $w, h \in \mathbb{Z}^+$ being image size. We have two kinds of measurements : *with-depth* measurements denoted by z^r and *semantic-only* measurements denoted as z^s. Each measurement has an associated semantic label $l \in \mathcal{L}_\mathcal{I}$, obtained from the 2D semantic classifier output (§ 6.2) at that pixel. Each *with-depth* measurement has an additional depth $d \in \mathbb{R}$ information, which in our case is obtained from visual SLAM (§ 6.1).

The observed data is composed of all the measurements and camera poses i.e. $\mathcal{D} = \{\mathbf{z}^r_{1:P}, \mathbf{z}^s_{1:Q}, \mathbf{g}_{1:T}\}$, where $\mathbf{z}^r_{1:P}$, $\mathbf{z}^s_{1:Q}$ and $\mathbf{g}_{1:T}$ respectively denotes the set of *with-depth* measurements, *semantic-only* measurements and camera trajectory up-to time T, which in our case is simply equivalent to number of images processed. Each $g_t \in \mathrm{SE}(3)$ is a single camera pose from the camera trajectory. Since we have multiple number of *with-depth* and *semantic-only* measurements per frame, we index them using p and q respectively, where $1{\leq}p{\leq}P$ and $1{\leq}q{\leq}Q$. Also we only have very sparse depth measurements, so $P \ll Q$.

We use subscript notation to denote associated camera pose, pixel semantic label, co-ordinate and depth (if available) for a particular measurement. Thus for a *semantic-only* measurement z^s_q, $l_q \in \mathcal{L}_\mathcal{I}$ denotes 2D image semantic label at pixel coordinate x_q with camera pose g_q. Similarly for p-th *with-depth* measurement z^r_p, d_p encodes the depth of the associated 3D point X_p, measured along the ray emanating from pixel location x_p with semantic label l_p and taken from camera pose g_p. We will sometime drop the superscript in z, when the type of measurement z^r (*with-depth*) or z^s (*semantic-only*) does not matter.

A single measurement z_k only affects a subset of voxels $\mathbf{m}_k \in \mathcal{M}$. For our camera sensor, these voxels are a subset of the voxels lying along the ray emanating from camera center through the corresponding image pixel coordinate of the measurement, denoted as $R_k = \mathrm{Ray}(x_k, g_k)$. Thus the set of voxels affected by a particular measurement z^r_p (or z^s_q) is represented by $\mathbf{m}_p \in R_p$ ($\mathbf{m}_q \in R_q$).

3 Probabilistic Model

We utilize a discriminative CRF model on $P(\mathcal{M}|\mathcal{D})$ to avoid directly modeling the complex dependencies [21,28] among correlated sources of *with-depth* and

semantic-only measurements. Unlike traditional occupancy grid mapping [30] we do not assume each m_i as independent from each other. Instead, we make use of the standard *static world* conditional independence assumptions of each measurement z_k given the map $\mathcal{M}$, and independence of the map $\mathcal{M}$ w.r.t.the camera trajectory $\mathbf{g}_{1:T}$. Given these assumptions, we can factorize the posterior over map $\mathcal{M}$ given all the observation data

$$
\begin{aligned}
P(\mathcal{M}|\mathcal{D}) &\propto P(\mathcal{M}|\mathbf{g}_{1:T})P(\mathbf{z}^r_{1:P}, \mathbf{z}^s_{1:Q}|\mathcal{M}, \mathbf{g}_{1:T}) \\
&= P(\mathcal{M}) \prod_{p=1}^{P} P(z^r_p|\mathcal{M}, g_p) \prod_{q=1}^{Q} P(z^s_q|\mathcal{M}, g_q) \qquad (1) \\
&= \underbrace{P(\mathcal{M})}_{\text{prior}} \prod_{p=1}^{P} \underbrace{P(z^r_p|\mathbf{m}_p, g_p)}_{\substack{\text{forward } \textit{with-depth} \\ \text{measurement model}}} \prod_{q=1}^{Q} \underbrace{P(z^s_q|\mathbf{m}_q, g_q)}_{\substack{\text{forward } \textit{semantic-only} \\ \text{measurement model}}} \qquad (2)
\end{aligned}
$$

where the conditional independence assumptions were applied to obtain (1), and since each measurement is only dependent on a subset of voxels in $\mathcal{M}$, we can further reduce (1) to get (2). (2) uses forward sensor measurement model [30] (measurement likelihood). However, if we adopt this factorization, we would need to learn a complicated sensor model in order to parametrize the forward sensor likelihoods $P(z_k|\mathbf{m_k}, g_k)$. Reapplying Bayes rule on (2), we get the inverse sensor model version as

$$
P(\mathcal{M}|\mathcal{D}) \propto \underbrace{P(\mathcal{M})}_{\text{prior}} \prod_{p=1}^{P} \underbrace{\frac{P(\mathbf{m}_p|z^r_p, g_p)}{P(\mathbf{m}_p)}}_{\substack{\text{inverse } \textit{with-depth} \\ \text{measurement model}}} \prod_{q=1}^{Q} \underbrace{\frac{P(\mathbf{m}_q|z^s_q, g_q)}{P(\mathbf{m}_q)}}_{\substack{\text{inverse } \textit{semantic-only} \\ \text{measurement model}}} \qquad (3)
$$

which provides the hints that our factors should be similar to posterior probabilities. We can rewrite both (2) and (3) in terms of factors [16]:

$$
P(\mathcal{M}|\mathcal{D}) = \frac{1}{Z(\mathcal{D})} \underbrace{\psi_\pi(\mathcal{M})}_{\text{prior factor}} \prod_{p=1}^{P} \underbrace{\psi^p_r(\mathbf{m}_p; z^r_p, g_p)}_{\substack{\textit{with-depth} \\ \text{measurement factors}}} \prod_{q=1}^{Q} \underbrace{\psi^q_s(\mathbf{m}_q; z^s_q, g_q)}_{\substack{\textit{semantic-only} \\ \text{measurement factors}}} \qquad (4)
$$

where $Z(\mathcal{D})$ is the partition function over the observed data. We now discuss the prior factor and the measurement factors.

Priors: In the above $P(\mathcal{M})$ or the prior factor ψ_π encodes the prior distribution over the huge set of all possible $\mathcal{L}_m^{|\mathcal{M}|}$ maps. However most of these maps are highly implausible and we can enforce some constraints in form of priors to improve our solution. We enforce the following priors over the map:

- **Spatial smoothness:** Our 3D world is not completely random and exhibits some sort of spatial smoothness.
- **Label compatibility:** Certain pair of classes are more/less likely to occur adjacent to one another. For example a *Car* voxel is unlikely to be adjacent to a *Building* voxel.

- **3D Support:** For most solid semantic categories (with the exception of *Tree*), an occupied voxel increases the chance of the voxels below it to belong to the same occupied category.
- **Free space Support:** *Free* space provides cues to improve 3D reconstruction along weakly supported surfaces [12]. Highly-supported free space boundaries are more likely to occupied.

We model spatial smoothness and label compatibility using pairwise potentials (§ 4.4). 3D and Free space support constraints are implemented with unary potentials (§ 4.1). Therefore, our $\psi_\pi(\mathcal{M})$ factorizes into pairwise and unary factors.

Measurement Factors: Measurement factors $\psi_r^p(\mathbf{m}_p;z_p^r,g_p)$ and $\psi_s^q(\mathbf{m}_q;z_q^s,g_q)$ encode the constraints imposed by a particular *with-depth* and *semantic-only* measurement respectively. In general, this forms a higher order clique involving multiple voxels $\mathbf{m}_k \subset \mathcal{M}$. However for certain kind of measurements, e.g. *with-depth* measurements or *semantic-only* measurements with *Sky* label, the factor $\psi(\mathbf{m}_k|z_k,g_k)$ can be approximated by a product of unaries on each voxel in $\mathbf{m}_k$. For example when we have a *with-depth* measurement, all voxels along the ray from camera center till the observed depth are more likely to *Free*. And the voxel corresponding to the observed 3D point is likely to belong to a solid semantic category. We use category-specific measurement models (described in § 4.2 and § 4.3) which can be either unary factors or higher order factors.

CRF Model: As discussed in the above two paragraphs we model the prior factor and the measurement factors in (4) with unary, pairwise and higher order potentials. Thus, rearranging the factors in (4) in terms of their arity, we get

$$P(\mathcal{M}|\mathcal{D}) = \frac{1}{Z(\mathcal{D})} \prod_i \psi_u^i(m_i) \prod_{i,j\in\mathcal{N}} \psi_p(m_i,m_j) \prod_{R\in\mathcal{R}} \psi_h(\mathbf{m}_R) \tag{5}$$

Here $\psi_u^i(m_i)$ is the unary potential defined over each m_i, and encodes local evidence. The pairwise potential, $\psi_p(m_i,m_j)$ over two neighboring voxels falling into a neighborhood $\mathcal{N}$ enforces spatial smoothness and label compatibility among them. Higher order cliques $\psi_h(\mathbf{m}_R)$ are defined over set of voxels $\mathbf{m}_R$ along some ray emanating from a 2D image projection and helps with missing depth information. Fig.2(a) shows the corresponding factor graph $\mathcal{H}$ of the model.

A single *semantic-only* measurement z_q^s for certain classes is ill-posed for updating states of the affected voxels $\mathbf{m}_q$ since we do not know which voxel reflects back the measurement. Häne et al.[9] simply updates all $\mathbf{m}_q$ with *Free* unaries for measurements missing depth, which is clearly an improper model. In our approach, we handle such measurements without range/depth, by forming higher order factor connecting voxels along a ray. However a naive approach will lead to forming huge higher order cliques and since every pixel in every image is an potential measurement, and inference in the graphical model can become intractable very soon. To circumvent this issue, whenever applicable, we make use of semantic cues to model them with unaries or at least reduce the scope of such higher order factors.

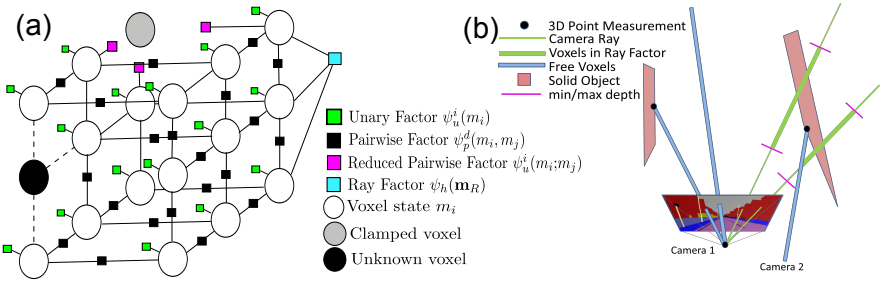

Fig. 2. (a) Factor Graph $\mathcal{H}$ of our framework. (b) Illustration of sensor models and higher order Ray factors. See text for more details.

4 Potentials

4.1 Basic Unary Potentials

We have different types of measurements, and they affect m_i differently. For example 3D depth measurement alone do not contain any semantic label information and influence all semantic label probabilities equally. Also each category of semantic observation affects the belief state of a voxel m_i, differently than others. We define the following two basic forms of unary measurement factors:

$$\psi_{\text{MISS}}(m_i) = \begin{cases} 0.6 & \text{if } m_i = \textit{Free} \\ \frac{0.4}{|\mathcal{L}_{\mathcal{M}}|-1} & \text{if } m_i \neq \textit{Free} \end{cases} \quad \text{and} \quad \psi^l_{\text{HIT}}(m_i) = \begin{cases} 0.3 & \text{if } m_i = \textit{Free} \\ 0.55 & \text{if } m_i \equiv l \\ \frac{0.15}{|\mathcal{L}_{\mathcal{M}}|-2} & \text{if } m_i \notin \{l, \textit{Free}\} \end{cases} \tag{6}$$

Fig.3 illustrates the measurement factors ψ_{MISS} and ψ^{Road}_{HIT}. Note that, we have made use use of inverse sensor model $P(m|z, g)$ for these factors. This is motivated by the fact that, it is much more easier [30] to elicit model parameters for $P(m|z, g)$ compared to the forward sensor likelihoods $P(z|m, g)$, and can be done without resorting to complicated sensor model learning. We kept the parameters same as that of laser based occupancy sensor model used in [11].

The unary potential $\psi^i_u(m_i)$ combines all the unary measurement factors that affect m_i. Thus the final unary potential over a voxel is factor product of a certain number of ψ_{MISS} and ψ^l_{HIT} factors only.

$$\psi^i_u(m_i) = [\psi_{\text{MISS}}(m_i)]^{N_M} \prod_{l \in \mathcal{L}_{\mathcal{I}} \setminus Sky} [\psi^l_{\text{HIT}}(m_i)]^{N_{Hl}} \tag{7}$$

where N_M is the total number of MISS unary factors over m_i and N_{Hl} being the number of HIT factors over m_i for semantic category l. Fig.3(c) depicts the factor graph view of this potential.

As new measurements are obtained, we keep on inserting new factors into the affected voxels. The set of voxels affected, and the kind of unary factors that gets inserted depends on the measurement type (discussed in next two subsections).

4.2 Measurements with depth

We use a projective camera sensor model, wherein the basic assumption is that each measurement is formed by reflection from a occupied voxel at some particular depth, and all voxels from the camera center to that depth are *Free*.

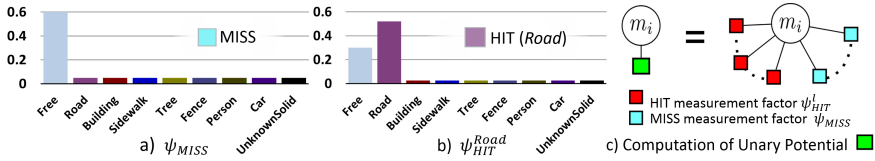

Fig. 3. a) and b) illustrates the MISS and HIT factors. c) Computation of per voxel unary potential as a product of unary contributions of several measurements affecting that voxel.

So for all voxels from camera center till the observed depth, we insert a MISS factor which increases the probability for these voxels being *Free*. And for the voxel corresponding to the observed 3D point X_p, we insert a HIT factor which makes the probability of belonging to a particular solid semantic state high. Our framework is not limited to monocular only system, the same approach can also be extended to a Laser+Vision system, where measurements from lasers affect all solid semantic category probabilities equally.

4.3 Semantic only Measurement

With sparse reconstruction most points in the image do not have direct depth measurements. However certain classes of measurements still provide a good estimation of depth. Observing *Sky* tells us that all voxels along the observed ray are more likely to be *Free*. Fig.4 LEFT shows average depth for some semantic categories across different parts of the image. We computed these statistics on the sequence seq05VD of Camvid. We first form a uniform 2D grid over the image, and then for each such grid in the image, we accumulate the depths from visual SLAM point clouds whose projection on the image lie on that grid. This gives us information about how good a *semantic-only* measurement z_q^s is in estimating the 3D depth. For each semantic class, all measurements with 2D projection x lying on the same grid gets same statistics. Two kind of statistics are computed for each such possible $(l_q, x_q) \in \mathcal{L}_\mathcal{I} \times \Omega$ measurement. The *min* depth and *max* depth for each (l_q, x_q) tells us the minimum and maximum possible depth along pixel co-ordinate x_q for 2D semantic category l_q. We then also estimate inverse sensor model $P(\mathbf{m}_p | z_q^s, g_q)$. Fig.4 shows the plots of inverse sensor model along with min/max depth for two specific *semantic-only* measurements, $(l_q = Road, x_q = [400, 700])$ and $(l_q = Building, x_q = [100, 300])$. When the statistics shows a small min-to-max bound e.g. *Road* and the inverse sensor model has a high peak, we insert unary factors according to this inverse sensor model.

However for certain classes like *Building*, depth uncertainty is too high to make it effective, since they can occur at different depths. Using unaries for these measurements introduces a lot of artifacts. So for these class of *semantic-only* measurements we construct a higher order factor involving all the voxels along the ray that lie between *min* depth and *max* depth computed for that semantic measurement. Solid *semantic-only* measurements like *Building*, *tree*, even though does not say much about the depth, confirms the fact that there is at least one occupied voxel along the ray induced by that observation. Our **Higher order Ray Potential** simply encodes this fact and can attain only two

possible values:

$$\psi_h(\mathbf{m}_R) = \begin{cases} \alpha & \text{if atleast one of } \mathbf{m}_R \text{ is } \neg \mathit{Free} \\ \beta & \text{if all of } \mathbf{m}_R \text{ is } \mathit{Free} \end{cases} \tag{8}$$

where $\mathbf{m}_R$ is set of voxels along a particular ray involved in the factor and $\alpha > \beta$. We make use of the class specific prior knowledge of the minimum depth and maximum depth of the reflecting voxel along a particular 2D back-projection. So for a ray factor $\psi_h(\mathbf{m}_R)$ caused by a measurement z_q^s, $\mathbf{m}_R = \{m_i : m_i \in R_q, min(l_q, x_q) \leq \text{depth}(m_i, g_q) \leq max(l_q, x_q)\}$. This reduces the number of voxels $|\mathbf{m}_R|$ involved in $\psi_h(\mathbf{m}_R)$, which could otherwise be very large (see Fig.2(b) for illustration). A further reduction is facilitated by strong free space measurements (see § 5.3). In contrast, the higher order factors used in [23] involve all the voxels starting from the camera. Another contrast to [23] is that our ray factor captures single view constraints which is orthogonal to multiview higher-order factors of [23] requiring costly photoconsistency computations across multiple views. Note that the higher order factor (8) is a sparse one and its of the same form as $\mathcal{P}^n$ Potts model [15] (a special case of Pattern potentials[17]) which allows us to do tractable inference (§ 6.3).

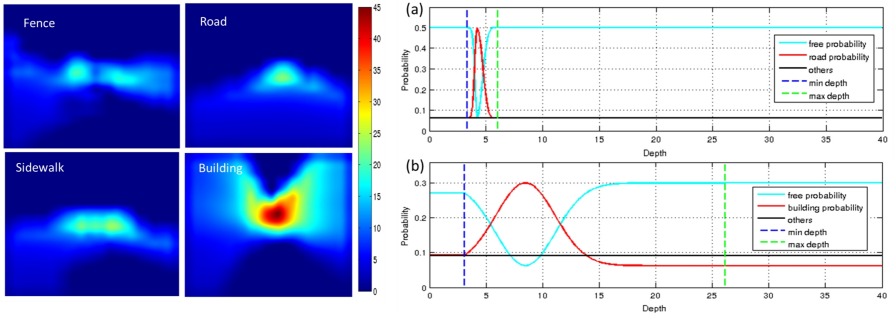

Fig. 4. LEFT: Average per category depthmap of Camvid [4] (subsequence # seq05VD) for *Fence*, *Road*, *Sidewalk* and *Building*. RIGHT: shows the inverse sensor model $P(m_i|z_q^s, g_q)$ for voxels i along the ray emanating from 2D point x_q as function of depth from camera center. (a) shows the inverse sensor model for a *Road* point measurement at 2D point co-ordinate, $x_q = [400, 700]$. (b) row shows the inverse sensor model for a *Building* observation at point $[100, 300]$. The plots also shows the min and max depth for these measurements.

4.4 Spatial smoothness and Label compatibility

The pairwise factor $\psi_p^d(m_i, m_j)$ enforces spatial smoothness and label compatibility between pairs of neighboring voxels defined by 3D neighborhood $\mathcal{N}$. Thus each voxel can have a maximum of 26 pairwise factors. The pairwise factors ψ_p^d are also dependent on relative direction d (horizontal or vertical) between the voxels. This allows us to capture properties like *Road* or *Sidewalk* voxels are more likely to be adjacent to each other in horizontal direction. So our pairwise potential is like Potts model, except that we set different weights for certain specific pairs of labels. To prevent *Free* voxels encroach other solid voxels, we set a lower cost for a $\psi_p^d(m_i = Free, m_j \neq Free)$ than other pairs in $\mathcal{L}_\mathcal{M} \times \mathcal{L}_\mathcal{M}$.

5 Data-driven Graphical Model Construction

The final graphical model is dynamically constructed and fully specified once all unary potentials has been computed.

5.1 Data Structure for Scene Representation

We use an octree based volumetric data structure which provides a compact storage of the scene. In the octree representation, when a certain subvolume observes some measurement, the corresponding node in the octree is initialized. Any *uninitialized* node in the octree represents *Unknown* areas. *Unknown* voxels are not included in the space over which we construct the graphical model and run our inference algorithm. This is different than other common approaches [23,9] of inferring over all voxels within a bounding box.

Of all factors used in our model, only the unary factor ψ_u^i is of different values for every m_i. All other factors like pairwise factors ψ_p or higher order ray factors ψ_h even though has different scopes, are fixed functions and we need to just store only **one instance** of them. Each node of the octree stores the local belief $bel(m_i)$ (as *log* probabilities) which is equal to the prior probability at time zero, and is incrementally updated to yield the final unary factor $\psi_u^i(m_i)$. Thus unlike a naive approach, we do not need to explicitly store all measurements, which is huge even for a short video sequence. Also note that all other factors apart from ψ_u^i are either precomputed, can be computed directly from voxel co-ordinates or from ψ_u^i itself without needing access to the raw measurement data.

5.2 Clamping

Even for nodes which have been initialized, if the local belief $bel(m_i)$ for a particular state $\in \mathcal{L}_\mathcal{M}$ has reached a very high probability (we used 0.98), we fix m_i to that state and treat it like evidence. This clamping of voxels which are already very confident about its label, reduces the total number of variables involved in the inference and also the scope of pairwise/higher-order factors attached to them. A pairwise factor between m_i and m_j gets *reduced* to unary factor $\psi_u^i(m_i) = \psi_p(m_i, m_j = \mathit{Free})$, when m_j gets clamped to *Free* label. In Fig.2(a), the shaded node ● represents such a clamped voxel and ■ denotes the reduced pairwise factors. Clamping of confident voxels and conservative generation of set of voxels over which we do the final inference, allows us to scale to longer sequences and not just scenes with a small fixed bounding box.

5.3 Scope Reduction of Higher Order Ray Potentials

Since the final graphical model structure $\mathcal{H}$ is computed only after all the unary potentials have been computed, it allows for further reduction of number of voxels $|\mathbf{m}_R|$ involved in higher order ray factors (8). We illustrate this with help of Fig.2(b). Suppose Camera1 receives a *semantic-only* measurement, which results

in a higher order ray factor involving voxels lying between min and max depth for that measurement. But strong free space measurements coming from other cameras (e.g. Camera2 in Fig.2(b)) helps us in further reducing the number of voxels $|\mathbf{m}_R|$ in the scope of that ray factor.

5.4 3D support and Free space support

Most solid semantic categories (with exceptions e.g. *Tree*) have a 3D support, as in an occupied voxel increases the chance of the voxels below it to belong to the same occupied category. So for voxels which have been clamped to semantic categories like *Building*, *Fence*, *Pole*, we insert a extra HIT unary factor corresponding to the same semantic category for all voxels lying directly below.

As shown by [12], highly-supported free space boundaries are more likely to be occupied. This is important for driving sequences, since most surfaces like road are very weakly supported by measurements. For voxels for which have been clamped to *Free*, we first check if there are *Unknown* voxels directly adjacent to it. If upon back-projecting these *Unknown* voxel coordinates to the images, we get a strong consensus in a solid semantic label: we initialize that voxel node and insert a single HIT unary factor corresponding to that label.

6 System Pipeline

With input monocular images, we first perform visual SLAM and an intial 2D scene parsing using standard semantic segmentation methods [20,18]. We then do a data-driven graphical model construction (§ 5.1) based on these measurements, followed by a final inference step.

6.1 Visual SLAM

Visual SLAM estimates the camera trajectory $\mathbf{g}_{1:t}$ and sparse 3D point cloud $\{X\}$ where $g_t \in \mathrm{SE}(3)$ and $X \in \mathbb{R}^3$. We do frame-to-frame matching of sparse 2D feature points, followed by RANSAC based relative pose estimation to obtain an initial estimate of the camera poses. A further improvement in feature tracking is obtained by rejecting matches across a image pair if the matched points lie on areas labeled as different semantic categories by the 2D semantic classifier. Finally we use bundle adjustment [13,1], which iteratively refines the camera poses and the sparse point cloud by minimizing a sum of all re-projection errors. Once bundle adjustment has converged, we obtain a set of sparse 3D points and corresponding camera poses from which each of these points have been observed.

6.2 Initial 2D Scene Parsing

We use the unary potentials used by Ladicky et al. [20] consisting of color, histogram of oriented gradients (HOG), pixel location features and several filter banks. We then use the dense CRF implementation of [18] to get the baseline 2D scene parsing. Since we directly work from per pixel semantic labels, any other scene parsing method can be used instead.

6.3 Inference Algorithm

For doing inference over the graphical model, we use the maximum a-posteriori (MAP) estimate $\mathcal{M}^* = \arg\max_{\mathcal{M}} P(\mathcal{M}|\mathcal{D})$ to assign a label to to each m_i. The rationale behind MAP is the big progress [14] of efficient approximate MAP inference in recent years. We use a modified message passing implementation of [14]. We use tree-reweighted (TRW) [32] messaging schedules. For computing messages to and from the higher order factors (8) we use the approach of [29]. Since our higher order factors (8) are sparse, all n outgoing messages from these higher order factors can be computed in $\mathrm{O}(n)$ ($\mathrm{O}(1)$ amortized) time.

7 Experiments and Evaluation

Since we are jointly estimating both 3D structure and semantic segmentation, it is expected that we improve upon both of them. In this section we define the evaluation criteria for measuring the above and show results to verify our claim. We demonstrate results of our method on Camvid [4] and Leuven [5,19] datasets. Both these datasets involve difficult fast forward moving cameras and has been standard dataset for semantic segmentation papers [3,19,31,24,6]. Leuven dataset contains stereo image pairs, but we demonstrate results only using monocular (left) images. To the best of our knowledge, we are not aware of any other work which has demonstrated joint 3D reconstruction and semantic segmentation on these standard monocular datasets. We additionally provide results on small subsequence of KITTI [8], again using monocular (left) images. Additional results and videos are available at the project website[1] and in supplementary material.

7.1 3D Structure Quality

We vastly improve upon the baseline 3D structure estimated through traditional SfM approach. Fig.6 shows some of our 3D reconstructions of a part of Camvid [4]. Note the improvement obtained over state of the art multi-view stereo [7] and sparse SfM in Fig.6. In the Leuven sequence, shown in Fig.5, we compare against the stereo based 2.5D method of Ladicky et al. [19] for joint segmentation and stereo. We back-project our 3D semantic map onto the cameras to obtain per frame depth/disparity image. Fig.5 qualitatively demonstrates the better quality of our 3D structure estimate, both in comparison to the stereo disparity maps and to baseline sparse SfM, even though only monocular(left) images were used compared to stereo method of [19]. In Fig.7, we compare against unary-only results with LIDAR sensor in KITTI [8].

7.2 Segmentation Quality

From our 3D joint semantic map, we can obtain 2D segmentation result by simply back-projecting it to each camera views. We evaluate segmentation quality

[1] http://www.cc.gatech.edu/~akundu7/projects/JointSegRec

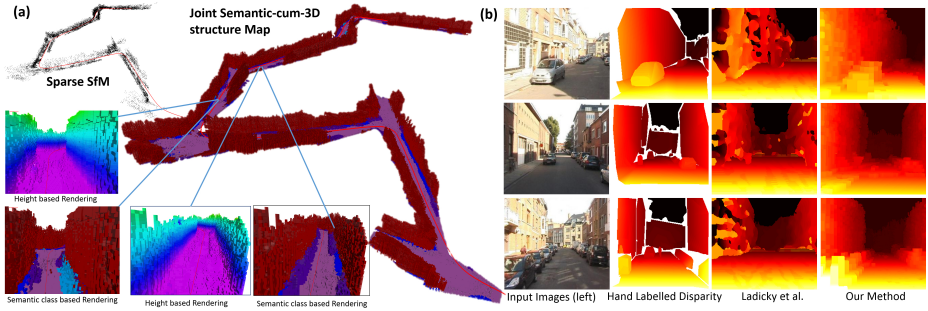

Fig. 5. Leuven [19] Results. (a): the output semantic reconstruction of the Leuven sequence, using only left (monocular) images. *Free* voxels are not shown for clarity. Note the improvement compared to initial SfM pointcloud. (b) Comparisons with the stereo method of Ladicky et al. [19], by using monocular (*left*) images *only*. We obtain 2D depth maps by back-projecting our 3D map onto the cameras. Notice the significant improvement over the depth maps of [19] when compared to the hand labeled disparity image provided by [19].

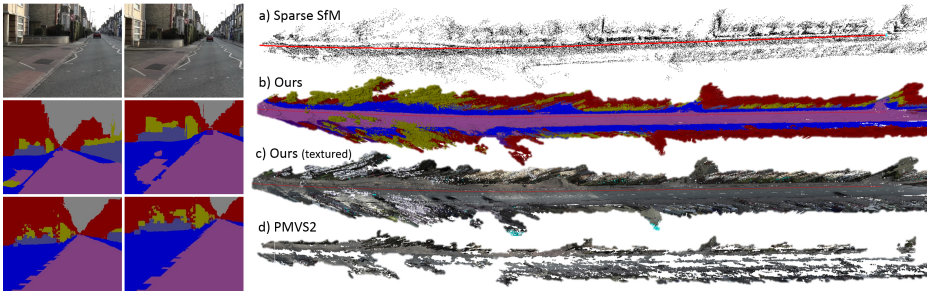

Fig. 6. CamVid [4] Results. LEFT: Top row shows two consecutive input images, middle row shows baseline 2D segmentation and bottom row shows 2D segmentation obtained by back-projecting our 3D semantic map. Note the temporal inconsistency in baseline 2D segmentation (middle row). RIGHT: a) 3D reconstruction and camera trajectory from Visual SLAM. b) Our 3D semantic + occupancy map using the same legend as in Fig.1. *Free* voxels are not shown for clarity. c) shows the same map, but textured. d)Reconstruction result by PMVS2 [7]. Note the improvement in our map (b,c) compared to sparse SfM(a)and PMVS2(d).

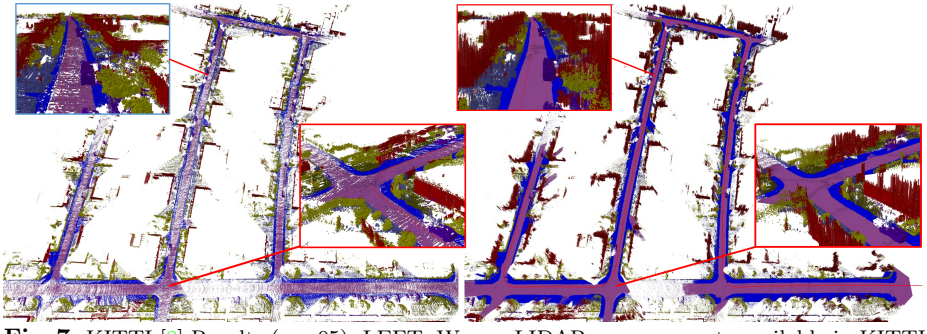

Fig. 7. KITTI [8] Results (seq 05). LEFT: We use LIDAR measurements available in KITTI using only the unary potentials described in this paper. RIGHT: Results with monocular (left) images and our full CRF model. As can be seen in the figure, even with just monocular images, we are able to achieve more complete reconstruction. For fair comparison, we only used those laser rays from the 360° LIDAR that can be seen by the left camera.

CAMVID seq05VD	*Building*		*Road*		*Car*		*Sidewalk*		*Sky*		*Tree*		*Fence*		*All*	
	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)
Ours	**0.0**	98.30	**0.0**	97.77	**0.0**	95.75	**0.0**	**98.33**	NA	99.27	**0.0**	**83.63**	**0.0**	73.74	**0.0**	**95.51**
[20]	0.114	**98.52**	0.024	95.99	0.231	89.41	0.177	96.53	NA	**99.81**	0.168	83.02	0.299	**75.59**	0.095	94.58
[24]	0.114	94.78	0.016	98.85	0.106	99.69	0.184	94.11	NA	99.21	0.173	80.34	0.249	39.06	0.084	92.41
[31]	0.025	95.01	0.004	**98.97**	0.046	**99.87**	0.062	73.17	NA	99.26	0.037	74.08	0.107	4.38	0.019	87.88

LEUVEN	*Building*		*Road*		*Car*		*Sidewalk*		*Sky*		*Bike*		*Pedestrian*		*All*	
	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)
Ours	**0.0**	**96.51**	**0.0**	**99.40**	**0.0**	**91.78**	**0.0**	66.97	NA	**95.30**	**0.0**	83.82	**0.0**	NA	**0.0**	**95.74**
[19]	0.046	95.84	0.116	98.75	0.150	91.42	0.429	**74.89**	NA	93.29	0.264	**84.68**	0.686	**61.76**	0.094	95.24

KITTI seq05	*Building*		*Road*		*Car*		*Sidewalk*		*Sky*		*Tree*		*Fence*		*All*	
	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)	H(bits)	Acc(%)
Ours	**0.0**	**98.90**	**0.0**	**98.72**	**0.0**	96.95	**0.0**	**98.35**	NA	99.37	**0.0**	96.45	**0.0**	**96.34**	**0.0**	**97.20**
[20]	0.165	97.47	0.113	87.85	0.203	**98.14**	0.158	96.00	NA	**99.75**	0.129	**97.47**	0.220	91.55	0.163	95.15

Table 1. 2D Segmentation evaluation. For evaluating temporal consistency, we give average Entropy H of SfM feature tracks (See § 7.2). Our results gives perfect zero entropy compared to non-zero entropy (indicating temporal inconsistency) for [24,31,19,20]. We also show the per pixel label accuracy. We again obtain the best results. Best scores has been **highlighted**.

in terms of both per pixel segmentation label accuracy and also temporal consistency of the segmentation in videos. We achieve significant improvement in both the measures over state of the art. To evaluate temporal consistency, we first select a set of confident SfM feature tracks which has very low re-projection errors after bundle adjustment. So these static 3D points should ideally be having same label from all the images it is visible from. So lower entropy (less changes in labels) for these SfM feature tracks is an indication of better temporal consistency. Table 1 shows the entropy scores for several state of art methods[24,19,31,20] where a higher entropy (in *bits*) indicates more temporal inconsistency. As a consequence of our model and 3D representation we achieve *perfect* consistency. We also evaluate per-pixel label accuracy and as shown in Table 1, our method achieves a noticeable gain over state of the art. The supplementary material has more discussion on these results.

8 Conclusion

We presented a method for joint inference of both semantic segmentation and 3D reconstruction, and thus provides a more holistic 3D understanding of the scene. Our framework offers several advantages : (a) Joint optimization of semantic segmentation and 3D reconstruction allows us to exploit more constraints and apply more informed regularization achieving improvement in both the tasks; (b) The 3D graphical model allows to incorporate more powerful 3D geometric cues compared to standard 2D image based spatial smoothness constraints; (c) It works for difficult forward moving monocular cameras, where sparse SfM is the only robust reconstruction method, and obtaining dense depth maps (required by [9,26]) is difficult; (d) We obtain full temporally consistent segmentations, without ad hoc constraints as in other 2D video segmentation methods [6,24,31]; (e) The output is in the form of a 3D volumetric semantic + occupancy map, which is much more useful than a series of 2D semantic label images or sparse pointcloud and it thus finds several applications like autonomous car navigation.

Acknowledgment: This work was supported by ARO-MURI award W911NF-11-1-0046.

References

1. Agarwal, S., Mierle, K., Others: Ceres solver. https://code.google.com/p/ceres-solver/ (2012)
2. Brostow, G.J., Shotton, J., Fauqueur, J., Cipolla, R.: Segmentation and recognition using structure from motion point clouds. In: ECCV (2008)
3. Brostow, G.J., Shotton, J., Fauqueur, J., Cipolla, R.: Segmentation and recognition using structure from motion point clouds. In: ECCV. pp. 44–57 (2008)
4. Brostow, G., Fauqueur, J., Cipolla, R.: Semantic object classes in video: A high-definition ground truth database. PRL 30(2), 88–97 (2009)
5. Cornelis, N., Leibe, B., Cornelis, K., Van Gool, L.: 3d urban scene modeling integrating recognition and reconstruction. IJCV 78(2-3), 121–141 (2008)
6. Floros, G., Leibe, B.: Joint 2d-3d temporally consistent segmentation of street scenes. In: CVPR (2012)
7. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. PAMI 32(8), 1362–1376 (2010)
8. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012)
9. Häne, C., Zach, C., Cohen, A., Angst, R., Pollefeys, M.: Joint 3d scene reconstruction and class segmentation. In: CVPR (2013)
10. Hoiem, D., Efros, A., Hebert, M.: Recovering surface layout from an image. IJCV 75(1), 151–172 (2007)
11. Hornung, A., Wurm, K.M., Bennewitz, M., Stachniss, C., Burgard, W.: OctoMap: An efficient probabilistic 3D mapping framework based on octrees. Autonomous Robots (2013)
12. Jancosek, M., Pajdla, T.: Multi-view reconstruction preserving weakly-supported surfaces. In: CVPR (2011)
13. Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J., Dellaert, F.: iSAM2: Incremental smoothing and mapping using the Bayes tree. IJRR 31, 217–236 (Feb 2012)
14. Kappes, J.H., Speth, M., Reinelt, G., Schnorr, C.: Towards efficient and exact map-inference for large scale discrete computer vision problems via combinatorial optimization. In: CVPR (2013)
15. Kohli, P., Ladick, L., Torr, P.: Robust higher order potentials for enforcing label consistency. IJCV 82(3), 302–324 (2009)
16. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. The MIT Press (2009)
17. Komodakis, N., Paragios, N.: Beyond pairwise energies: Efficient optimization for higher-order mrfs. In: CVPR (2009)
18. Krahenbuhl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In: NIPS (2011)
19. Ladicky, L., Sturgess, P., Russell, C., Sengupta, S., Bastanlar, Y., Clocksin, W., Torr, P.H.: Joint optimisation for object class segmentation and dense stereo reconstruction. In: BMVC (2010)
20. Ladicky, L., Russell, C., Kohli, P., Torr, P.: Associative hierarchical crfs for object class image segmentation. In: ICCV (2009)
21. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: ICML (2001)
22. Liu, B., Gould, S., Koller, D.: Single image depth estimation from predicted semantic labels. In: CVPR (2010)

23. Liu, S., Cooper, D.B.: Ray markov random fields for image-based 3d modeling: model and efficient inference. In: CVPR (2010)
24. Miksik, O., Munoz, D., Bagnell, J.A., Hebert, M.: Efficient temporal consistency for streaming video scene analysis. In: ICRA (2013)
25. Saxena, A., Chung, S., Ng, A.: 3-D Depth Reconstruction from a Single Still image. IJCV 76(1), 53–69 (2008)
26. Sengupta, S., Greveson, E., Shahrokni, A., Torr, P.H.S.: Urban 3d semantic modelling using stereo vision. In: ICRA (2013)
27. Sturgess, P., Alahari, K., Ladicky, L., Torr, P.H.S.: Combining appearance and structure from motion features for road scene understanding. In: BMVC (2009)
28. Sutton, C., McCallum, A.: An introduction to conditional random fields. PAMI 4(4), 267–373 (2012)
29. Tarlow, D., Givoni, I.E., Zemel, R.S.: Hop-map: Efficient message passing with high order potentials. In: AISTATS (2010)
30. Thrun, S., Burgard, W., Fox, D.: Probabilistic robotics. MIT Press (2005)
31. Tighe, J., Lazebnik, S.: Superparsing: Scalable nonparametric image parsing with superpixels. International Journal of Computer Vision (2012)
32. Wainwright, M.J., Jordan, M.I.: Graphical models, exponential families, and variational inference. Foundations and Trends® in Machine Learning 1(1-2), 1–305 (2008)