Blocks World Revisited: Image Understanding Using Qualitative Geometry and Mechanics

Abhinav Gupta, Alexei A. Efros, and Martial Hebert

Robotics Institute, Carnegie Mellon University

Abstract. Since most current scene understanding approaches operate either on the 2D image or using a surface-based representation, they do not allow reasoning about the physical constraints within the 3D scene. Inspired by the "Blocks World" work in the 1960's, we present a *qualitative* physical representation of an outdoor scene where objects have volume and mass, and relationships describe 3D structure and mechanical configurations. Our representation allows us to apply powerful global geometric constraints between 3D volumes as well as the laws of statics in a qualitative manner. We also present a novel iterative "interpretation-by-synthesis" approach where, starting from an empty ground plane, we progressively "build up" a physically-plausible 3D interpretation of the image. For surface layout estimation, our method demonstrates an improvement in performance over the state-of-the-art [9]. But more importantly, our approach automatically generates **3D parse graphs** which describe qualitative geometric and mechanical properties of objects and relationships between objects within an image.

1 Introduction

What does it mean to understand a visual scene? One popular answer is simply *naming* the objects present in the image – "building", "road", etc. – and possibly locating them in the image. However this level of understanding is somewhat superficial as it tells us little about the underlying structure of the scene. To really understand an image it is necessary to probe deeper – to acquire some notion of the geometric scene layout, its free space, walkable surfaces, qualitative occlusions and depth relationships, etc. Object naming has also a practical limitation: due to the heavy-tailed distribution of object instances in natural images, a large number of objects will occur too infrequently to build usable recognition models, leaving parts of the image completely unexplained. To address these shortcomings, there has been a recent push toward more geometric, rather than semantic, approaches to image understanding [8, 18]. The idea is to learn a mapping between regions in the image and planar surfaces in the scene. The resulting labeled image can often be "popped-up" into 3D by cutting and folding it at the appropriate region boundaries, much like a children's pop-up book. However this process has two limitations. First, since surface orientation labels are being estimated per region or per super-pixel, it is difficult to enforce global consistency, which often leads to scene models that are physically

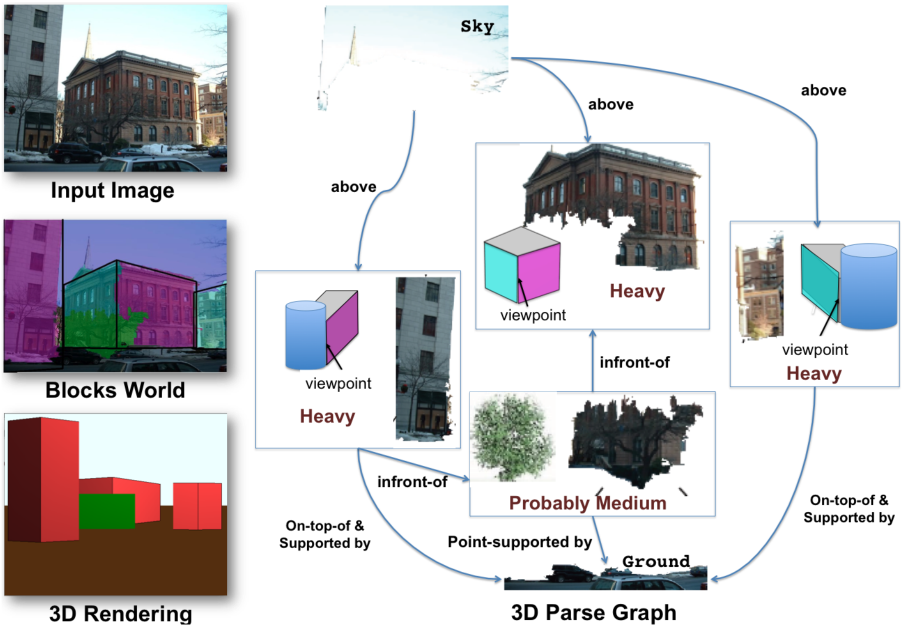

Fig. 1. Example output of our automatic scene understanding system. The 3D parse graph summarizes the inferred object properties (physical boundaries, geometric type, and mechanical properties) and relationships between objects within the scene. See more examples on project webpage.

impossible or highly unlikely. Second, even if successful, these pop-up models (also known as "billboards" in graphics) lack the physical substance of a true 3D representation. Like in a Potemkin village, there is nothing behind the pretty façades!

This paper argues that a more physical representation of the scene, where objects have volume and mass, can provide crucial high-level constraints to help construct a globally-consistent model of the scene, as well as allow for powerful ways of understanding and interpreting the underlying image. These new constraints come in the form of geometric relationships between 3D volumes as well as laws of *statics* governing the behavior of forces and torques. Our main insight is that the problem can be framed *qualitatively*, without requiring a metric reconstruction of the 3D scene structure (which is, of course, impossible from a single image). Figure 1 shows a real output from our fully-automatic system.

The paper's main contributions are: (a) a novel qualitative scene representation based on volumes (blocks) drawn from a small library; (b) the use of 3D geometry and mechanical constraints for reasoning about scene structure; (c) an iterative Interpretation-by-Synthesis framework that, starting from the empty ground plane, progressively "builds up" a consistent and coherent interpretation of the image; (d) a top-down segmentation adjustment procedure where partial scene interpretations guide the creation of new segment proposals.

Related Work: The idea that the basic physical and geometric constraints of our world (so-called *laws of nature*) play a crucial role in visual perception goes back at least to Helmholtz and his argument for "unconscious inference". In computer vision, this theme can be traced back to the very beginnings of our discipline, with Larry Roberts arguing in 1965 that *"the perception of solid objects is a process which can be based on the properties of three-dimensional transformations and the laws of nature"* [17]. Roberts' famous Blocks World was a daring early attempt at producing a complete scene understanding system for a closed artificial world of textureless polyhedral shapes by using a generic library of polyhedral block components. At the same time, researchers in robotics also realized the importance of physical stability of block assemblies since many block configurations, while geometrically possible, were not physically stable. They showed how to generate plans for the manipulation steps required to go from an initial configuration to a target configuration such that at any stage of assembly the blocks world remained stable [1]. Finally, the *MIT Copy Demo* [21] combined the two efforts, demonstrating a robot that could visually observe a blocks world configuration and then recreate it from a pile of unordered blocks (recently [2] gave a more sophisticated reinterpretation of this idea, but still in a highly constrained environment).

Unfortunately, hopes that the insights learned from the blocks world would carry over into the real world did not materialize as it became apparent that algorithms were too dependent on its very restrictive assumptions (perfect boundary detection, textureless surfaces, etc). While the idea of using 3D geometric primitives for understanding real scenes carried on into the work on generalized cylinders and resulted in some impressive demos in the 1980s (e.g., ACRONYM [16]), it eventually gave way to the currently dominant appearance-based, semantic labeling methods, e.g., [19, 5]. Of these, the most ambitious is the effort of S.C.Zhu and colleagues [23] who use a hand-crafted stochastic grammar over a highly detailed dataset of labelled objects and parts to hierarchically parse an image. While they show impressive results for a few specific scene types (e.g., kitchens, corridors) the approach is yet to be demonstrated on more general data.

Most related to our work is a recent series of methods that attempt to model geometric scene structure from a single image: inferring qualitative geometry of surfaces [8, 18], finding ground/vertical "fold" lines [3], grouping lines into surfaces [22, 13], estimating occlusion boundaries [10], and combining geometric and semantic information [9, 14, 4]. However, these approaches do not model the global interactions between the geometric entities within the scene, and attempts to incorporate them at the 2D image labeling level [15, 12] have been only partially successful. While single volumes have been used to model simple building interiors [6] and objects (such as bed) [7], these approaches do not model geometric or mechanical inter-volume relationships. And while modeling physical constraints has been used in the context of dynamic object relationships in video [20], we are not aware of any work using them to analyze static images.

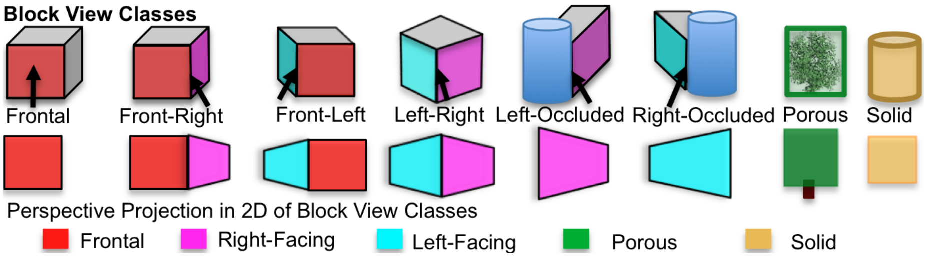

Fig. 2. Catalog of the possible block view classes and associated 2D projections. The 3D blocks are shown as cuboids although our representation imposes no such constraints on the 3D shape of the block. The arrow represents the camera viewpoint.

2 Overview

Our goal is to obtain a rich physical understanding of an outdoor scene (under camera assumptions similar to [9]) in which objects have volume and mass, and inter-object relationships are governed by the geometry and mechanics of the 3D world. Unfortunately, from a single image, it is next to impossible to estimate a precise, metric 3D representation for a generic scene. Our proposed solution is to represent 3D objects *qualitatively*, as one or more convex "blocks". We define a block as an image region represented by one of a small class of geometric primitives and qualitative density distribution(described below). The block is our basic unit of reasoning within the 3D scene. While a block is purposefully kept somewhat under-constrained to allow 3D reasoning even with a large degree of uncertainty, it contains enough information to produce a reasonable 3D rendering under some assumptions (see Figures 1 and 8).

2.1 Block Representation

Geometrically, we want to qualitatively represent the 3D space occupied by a block with respect to the camera viewpoint. Using the convexity assumption, we can restrict the projection of each block in the image to one of the eight *block-view classes* shown in Figure 2. These classes correspond to distinct aspects of a block over possible viewpoints. Figure 3(a) shows a few examples of extracted blocks in our test dataset.

We also want to represent the gravitational force acting on each block and the extent to which a block can support other blocks, which requires knowing the density of the block. Estimating density using visual cues alone is a hard problem. But it turns out that there is enough visual regularity in the world to be able to coarsely estimate a *density class* of each block: "light"(e.g. trees and bushes), "medium" (e.g. humans) and "high-density" (e.g. buildings). These three classes span the spectrum of possible densities and each class represents order-of-magnitude difference with respect to the other classes.

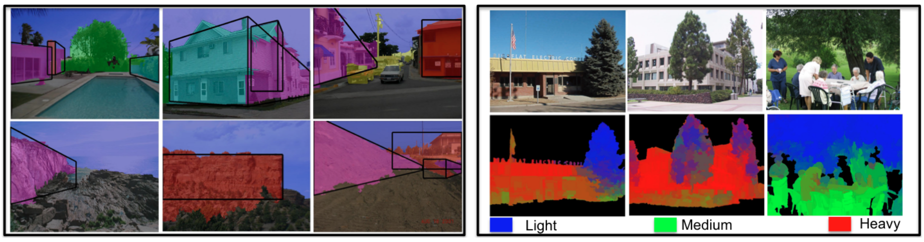

Fig. 3. (a) Examples of extracted blocks in the wide variety of images. (b) Examples of super-pixel based density estimation of our approach.

2.2 Representing Relationships

Instead of worrying about absolute depth, we only encode pairwise depth relationships between blocks, as defined by the painters' algorithm. For a given pair of blocks $\mathcal{B}_i$ and $\mathcal{B}_j$, there are three possible depth relationships: "infront of","behind" and "no-relationship". Block $\mathcal{B}_i$ is "infront of" $\mathcal{B}_j$ if $\mathcal{B}_i$ transitively occludes $\mathcal{B}_j$ in the current viewpoint and vice-versa.

To represent the mechanical configuration of the scene we use support relationships between pair of blocks. For a given pair of blocks $\mathcal{B}_i$ and $\mathcal{B}_j$, there are three possible support relationships: "supports","supported by" and "no-relationship". Figure 1 shows examples of the depth and support relationships extracted by our approach (see edges between the blocks).

2.3 Geometric and Mechanical Constraints

Having a rich physical representation (blocks with masses instead of popped-up planes) of the scene can provide additional powerful global constraints which are vital for successful image understanding. These additional constraints are:

Static Equilibrium. Under the static world assumption, the forces and torques acting on a block should cancel out (Newton's first law). We use this to derive constraints on segmentation of objects, and estimating depth and support relationships. For example, in Figure 4(c), the orange segment is rejected since it leads to the orange block which is physically unstable due to unbalanced torques.

Support Force Constraint. A supporting object should have enough strength to provide contact reactionary forces on the supported objects. We utilize density to derive constraints on the support relationships and on the relative strengths of the support*ing* and support*ed* bodies.

Volume Constraint. All the objects in the world must have finite volumes and cannot inter-penetrate each other. Figure 4(b) shows an example of how this constraint can help in rejecting bad segmentation hypotheses (red and yellow surfaces in the image cannot belong to different objects) since that would lead to 3D intersection of blocks.

Depth Ordering Constraint. The depth ordering of the objects with respect to the camera viewpoint should be consistent with the projected regions in the image as explained in Section 3.6

Block Representation

Type	Classes
View Classes	“frontal”, “front-right”, “front-left”, “left-right”, “left-occluded"," right-occluded”, “porous”, “solid-non planar”
Density Classes	“light”, “medium” and “heavy”

Relationships Representation

Type	Classes
Depth	“infront”, “behind”, “no-relationship”
Support	“Supports”, “Supported-by"," no-relationship”

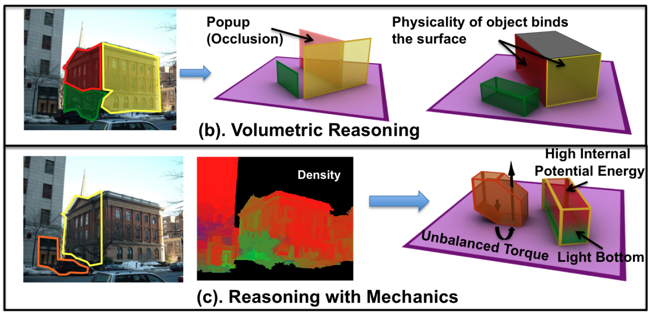

Fig. 4. (a) Our Representation (b)Role of Volume Constraint: The finite volume constraint binds the two surfaces together whereas in pop-ups the relative locations of surfaces is not constrained. (c) Role of Static Constraints: Unbalanced torque leads to rejection of the orange segment. The low internal stability (lighter bottom and heavier top) of yellow segment leads to its rejection.

3 Assembling Blocks World: Interpretation by Synthesis

We need to use the constraints described earlier to generate a physically plausible scene interpretation. While one can apply these constraints to evaluate all possible interpretations of the scene, such an approach is computationally infeasible because of the size of the hypotheses space. Similarly, probabilistic approaches like Bayesian networks where all attributes and segmentations are inferred simultaneously are also infeasible due to the large number of variables with higher-order clique constraints (physical stability has to be evaluated in terms of multiple bodies simultaneously interacting with each other). Instead, we propose an iterative “interpretation by synthesis” approach which grows the image interpretation by adding regions one by one, such that confident regions are interpreted first in order to guide the interpretation of other regions. Our approach is inspired by robotics systems which perform block assembly to reach a target configuration of blocks [11]. One nice property of our approach is that, at any given stage in the assembly, the partial interpretation of the scene satisfies all the geometrical and mechanical constraints described above.

3.1 Initialization

Before we begin assembling a blocks world from an input image, we need to generate an inventory of hypotheses that are consistent with the image. We use a multiple segmentation approach to generate a large number of regions that can correspond to blocks. We use the occlusion boundary segmenter of [10], which starts with a set of superpixels and iteratively coarsens the segmentation by combining regions selected by a local classifier. Since our goal is to generate multiple possible segmentations, we implement a randomized version of the same where the coarsening is done multiple times with different seeds and different parameters and to generate 17 possible segmentations for each image.

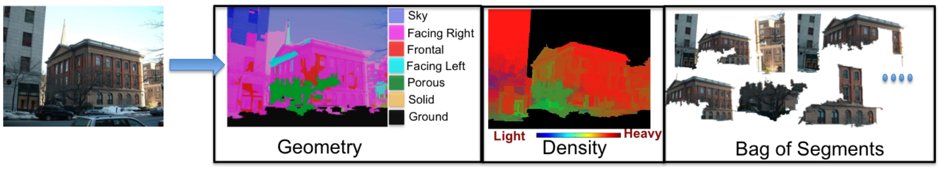

Fig. 5. Initialization: We use multiple segmentation to generate block hypothesis and a super-pixel based classifier for surface geometry and material-density.

We estimate the local surface layout labels using [8] to estimate the view class of each block hypothesis. To estimate the density of each block hypothesis, we implemented a superpixel-based local classifier which learns a mapping between image features (same as in [8]) and object densities. We learn the density classifier on a set of training images labeled with the three density classes. Figure 3(b) shows a few examples of per-superpixel density estimation for a few images. A more detailed quantitative analysis of our density classifier is provided in Section 4. Figure 5 shows the initialization of our approach. Using the "ground" and "sky" regions from the estimated surface layout, we initialize our empty blocks world consisting of ground and sky. We are now ready to start adding blocks.

3.2 Searching Block Configurations

The goal is to assemble a blocks world that is consistent with the input image. However, which block hypotheses should be used and the order in which they should be added are not known *a priori*. We employ a search strategy where for a current blocks world $\mathcal{W}_t$, we propose k possible block hypotheses. These k hypotheses are selected based on a set of local criteria: confidence in surface layout geometry, density estimation, internal physical stability (heavy-bottom and light top, c.f. Section 3.5), and support likelihood (estimated based on contact with support surfaces). Each of these k hypotheses is then provisionally "placed" into the current blocks world and scored using a cost function described in the next section. The hypothesis with the lowest cost is accepted and the corresponding block is added to reach a new configuration $\mathcal{W}_{t+1}$. The process of generating new hypothesis and cost estimation is then repeated for $\mathcal{W}_{t+1}$, until all the regions in the image have been explained. For all experiments in the paper we use $k = 4$.

3.3 Evaluating Proposals

Given a candidate block $\mathcal{B}_i$, we want to estimate its associated mechanical and geometrical properties and its relationship to the blocks already placed in the scene to minimize the following cost function:

$$\begin{aligned}\mathcal{C}(\mathcal{B}_i) = \mathcal{F}_{geometry}(\mathcal{G}_i) + \sum_{S \in ground, sky} \mathcal{F}_{contacts}(\mathcal{G}_i, S) + \mathcal{F}_{intra}(\mathcal{S}_i, \mathcal{G}_i, d) \\ + \sum_{j \in blocks} \mathcal{F}_{stability}(\mathcal{G}_i, S_{ij}, \mathcal{B}_j) + \mathcal{F}_{depth}(\mathcal{G}_i, S_{ij}, \mathcal{D}), \qquad (1)\end{aligned}$$

where $\mathcal{G}_i$ represents the estimated block-view class, $\mathcal{S}_i$ corresponds to the region associated with the block, d is the estimated density of the block, S_{ij} represent support relationships and $\mathcal{D}$ represents partial-depth ordering obtained from depth relationships. $\mathcal{F}_{geometry}$ measures the agreement of the estimated block view class with the superpixel-based surface layout estimation [8] and $\mathcal{F}_{contacts}$ measures the agreement of geometric properties with ground and sky contact points (Section 3.4). $\mathcal{F}_{intra}$ and $\mathcal{F}_{stability}$ measure physical stability within a single block and with respect to other blocks respectively (Section 3.5). Finally, $\mathcal{F}_{depth}$ measures the agreement of projection of blocks in the 2D image plane with the estimated depth ordering (Section 3.6).

Minimizing the cost function over all possible configurations is too costly. Instead, Figure 6 illustrates our iterative approach for evaluating a block hypothesis by estimating its geometric and mechanical properties such that the cost function $\mathcal{C}$ is approximately minimized. We first estimate the block-view class of the new block by minimizing $\mathcal{F}_{geometry}$ and $\mathcal{F}_{contacts}$ (Figure 6c). Using the block-view class, we compute the stability of the blocks under various support relationships by minimizing $\mathcal{F}_{intra}$ and $\mathcal{F}_{stability}$ (Figure 6d). We then use the estimated block-view class and support relationships to estimate a partial depth ordering of the blocks in the image . This minimizes the final term in our cost function, $\mathcal{F}_{depth}$ (Figure 6e). Block-geometric properties, physical stability analysis and partial depth ordering of blocks in the scene provide important cues for improving segmentation. Therefore, after computing these properties, we perform a segmentation adjustment step (Figure 6f). The final computed score is then returned to the top-level search procedure which uses it to select the best block to add. We now discuss each of the steps above in detail.

3.4 Estimating Geometry

Estimating the geometric attributes of a block involves inferring the block view class (Figure 2) and the 2D location of a convex corner, which we call "foldedge" using cues from the surface layout estimation and ground and sky contact points. Let us assume that we are adding block $\mathcal{B}_i$ with the associated segment $\mathcal{S}_i$ in the 2D image plane. We need to estimate the block-view class $\mathcal{G}_i$ and the foldedge location f_i given the surface-geometric labels for superpixels g and the evidence from ground and sky contact points in the image plane (C_i^G and C_i^S). This can be written as:

$$P(\mathcal{G}_i, f_i | g, \mathcal{S}_i, C_i^G, C_i^S) \propto P(g|\mathcal{G}_i, f_i, \mathcal{S}_i) P(C_i^G|\mathcal{G}_i, f_i) P(C_i^S|\mathcal{G}_i, f_i). \quad (2)$$

The first term indicates how well the surface layout matches the hypothesized block view class, and the second and third terms indicate the agreement between block view class and ground/sky contact points, respectively. We compute the first term as

$$P(g|\mathcal{G}_i, f_i, \mathcal{S}_i) = \frac{1}{Z} \sum_{s \in \mathcal{S}_i} P(g_s|\mathcal{G}_i, f_i, s), \quad (3)$$

where s is a super-pixel in segment $\mathcal{S}_i$. $P(g_s|\mathcal{G}_i, f_i, s)$ represents the agreement between the predicted block geometry, $(\mathcal{G}_i, f_i)$ and the result of the surface layout

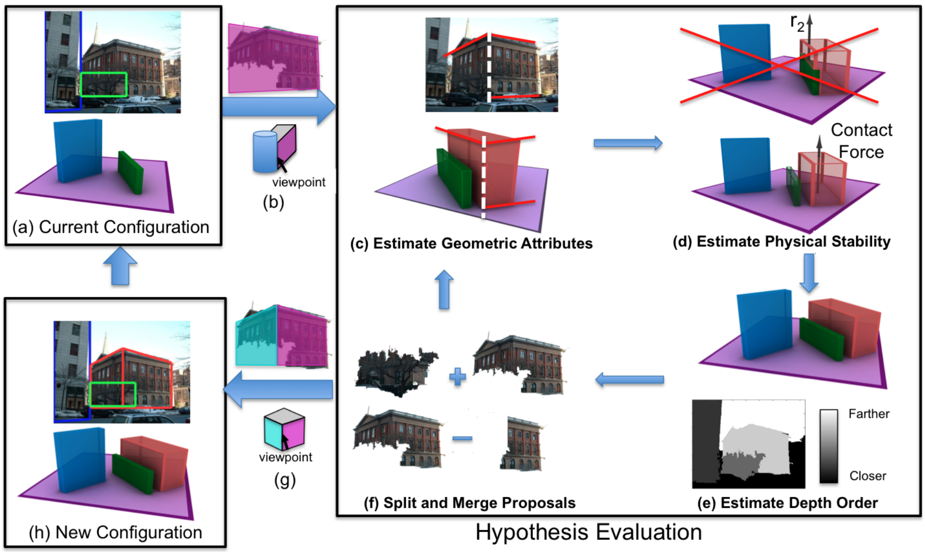
(a) Current Configuration
viewpoint
(b)
r2
Contact
Force
(c) Estimate Geometric Attributes
(d) Estimate Physical Stability
(h) New Configuration
viewpoint
(g)
(f) Split and Merge Proposals
Farther
Closer
(e) Estimate Depth Order
Hypothesis Evaluation

Fig. 6. Our approach for evaluating block hypothesis and estimating the associated cost of placing a block.

estimation algorithm for superpixel s. For example, if the block is associated with "front-right" view class and the superpixel is on the right of the folding edge, then $P(g_s|\mathcal{G}_i, f_i, s)$ would be the probability of the superpixel being labeled right-facing by the surface layout estimation algorithm.

For estimating the contact points likelihood term, we use the constraints of perspective projection. Given the block geometry and the folding edge, we fit straight lines l_g and l_s to the the ground and sky contact points, respectively, and we verify if their slopes are in agreement with the surface geometry: for a frontal surface, l_g and l_s should be horizontal, and for left- and right-facing surfaces l_g and l_s should intersect on the horizon line.

3.5 Estimating Physical Stability

Our stability measure (Figure 6d) consists of three terms. (1) **Internal Stability:** We prefer blocks with low potential energies, that is, blocks which have heavier bottom and lighter top. This is useful for rejecting segmentations which merge two segments with different densities, such as the lighter object below the heavier object shown on Figure 4(c). For computing internal stability, we rotate the block by a small angle, $\delta\theta$, (clockwise and anti-clockwise) around the center of each face; and compute the change in potential energy of the block as:

$$\Delta P_i = \sum_{c \in \{light, medium, heavy\}} \sum_{s \in \mathcal{S}_i} p(d_s = c) m_c \delta h_s, \tag{4}$$

where $p(d_s = c)$ is the probability of assigning density class c to superpixel s, δh_s is the change in height due to the rotation and m_c is a constant representing the density class. The change in potential energy is a function of three constants.

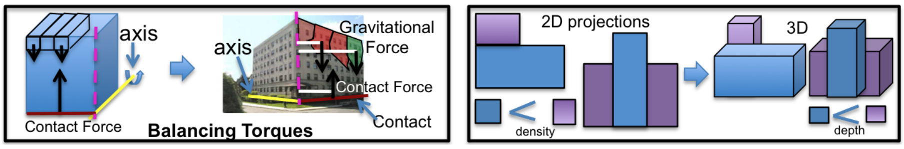

Fig. 7. (a) Computation of torques around contact lines. (b) Extracted depth constraints are based on convexity and support relationships among blocks.

Using constraints such as $\rho_{hm} = \frac{m_{heavy}}{m_{medium}} > 1$ and $\rho_{lm} = \frac{m_{light}}{m_{medium}} < 1$, we compute the expected value of $\varDelta P_i$ with respect to the ratio of densities (ρ_{hm} and ρ_{lm}). The prior on ratio of densities for the objects can be derived using density and the frequency of occurrence of different materials in our training images. (2) **Stability:** We compute the likelihood of a block being stable given the density configuration and support relations. For this, we first compute the contact points of the block with the supporting block and then compute the torque due to gravitational force exerted by each superpixel and the resultant contact force around the contact line (Figure 7a). This again leads to torque as a function of three constants and we use similar qualitative analysis to compute the stability. (3) **Constraints from Block Strength:** We also derive constraint on support attributes based on the densities of the two blocks possibly interacting with each other. If the density of the supporting block is less than density of the supported block; we then assume that the two blocks are not in physical contact and the block below occludes the contact of the block above with the ground.

3.6 Extracting Depth Constraints

The depth ordering constraints (Figure 6(e)) are used to guide the next step of refining the segmentation by splitting and merging regions. Computing depth ordering requires estimating pairwise depth constraints on blocks and then using them to form global depth ordering. The rules for inferring depth constraints are shown in Figure 7(b). These pairwise constraints are then used to generate a global partial depth ordering via a simple constraint satisfaction approach.

3.7 Creating Split and Merge Proposals

This final step involving changes to the segmentation (Figure 6f) is crucial because it avoids the pitfalls of previous systems which assumed a fixed, initial segmentation (or even multiple segmentations) and were unable to recover from incorrect or incomplete groupings. For example, no segmentation algorithm can group two regions separated by an occluding object because such a merge would require reasoning about depth ordering. It is precisely this type of reasoning that the depth ordering estimation of Section 3.6 enables. We include segmentation in the interpretation loop and use the current interpretation of the scene to generate more segments that can be utilized as blocks in the blocks world.

Using estimated depth relationships and block view classes we create **merge proposals** where two or more non-adjacent segments are combined if they share a block as neighbor which is estimated to be in front of them in the current viewpoint. In that case, the shared block is interpreted as an occluder which fragmented the background block into pieces which the merge proposal attempts to reconnect. We also create additional merge proposals by combing two or more neighboring segments. **Split proposals** divide a block into two or more blocks if the inferred properties of the block are not in agreement with confident individual cues. For example, if the surface layout algorithm estimates a surface as frontal with high-confidence and our inferred geometry is not frontal, then the block is divided to create two or more blocks that agree with the surface layout. The split and merge proposals are then evaluated by a cost function whose terms are based on the confidence in the estimated geometry and physical stability of the new block(s) compared to previous block(s). In our experiments, approximately 11% of the blocks are created using the resegmentation procedure.

4 Experimental Results

Since there has been so little done in the area of qualitative volumetric scene understanding, there are no established datasets, evaluation methodologies, or even much in terms of relevant previous work to compare against. Therefore, we will present our evaluation in two parts: 1) qualitatively, by showing a few representative scene parse results in the paper, and a wide variety of results on the project webpage [1]; 2) quantitatively, by evaluating *individual components* of our system and, when available, comparing against the relevant previous work.

Dataset: We use the dataset and methodology of Hoiem et. al [9] for comparison. This dataset consists of 300 images of outdoor scenes with ground truth surface orientation labeled for all images, but occlusion boundaries are only labelled for 100 images. The first 50 (of the 100) are used for training the surface segmentation [8] and occlusion reasoning [10] of our segmenter. The remaining 250 images are used to evaluate our blocks world approach. The surface classifiers are trained and tested using five-fold cross-validation just like in [9].

Qualitative: Figure 8 shows two examples of complete interpretation automatically generated by the system and a 3D toy blocks world generated in VRML. In the top example, the building is occluded by a tree in the image and therefore none of the previous approaches can combine the two faces of the building to produce a single building region. In a pop-up based representation, the placement of the left face is unconstrained due to the contact with ground not being visible. However, in our approach volumetric constraints aid the reasoning process and combine the two faces to produce a block occluded by the tree. The bottom example shows how statics can help in selecting the best blocks and improve block-view estimation. Reasoning about mechanical constraints rejects the segment corresponding to the whole building (due to unbalanced torque). For the selected convex block, the cues from ground and sky contact points aid in proper geometric classification of the block. Figure 9 shows a few other qualitative examples with the overlaid block and estimated surface orientations.

Quantitative: We evaluate various components of our system separately. It is not possible to quantitatively compare the performance of the entire system

[1] http://www.cs.cmu.edu/∼abhinavg/blocksworld

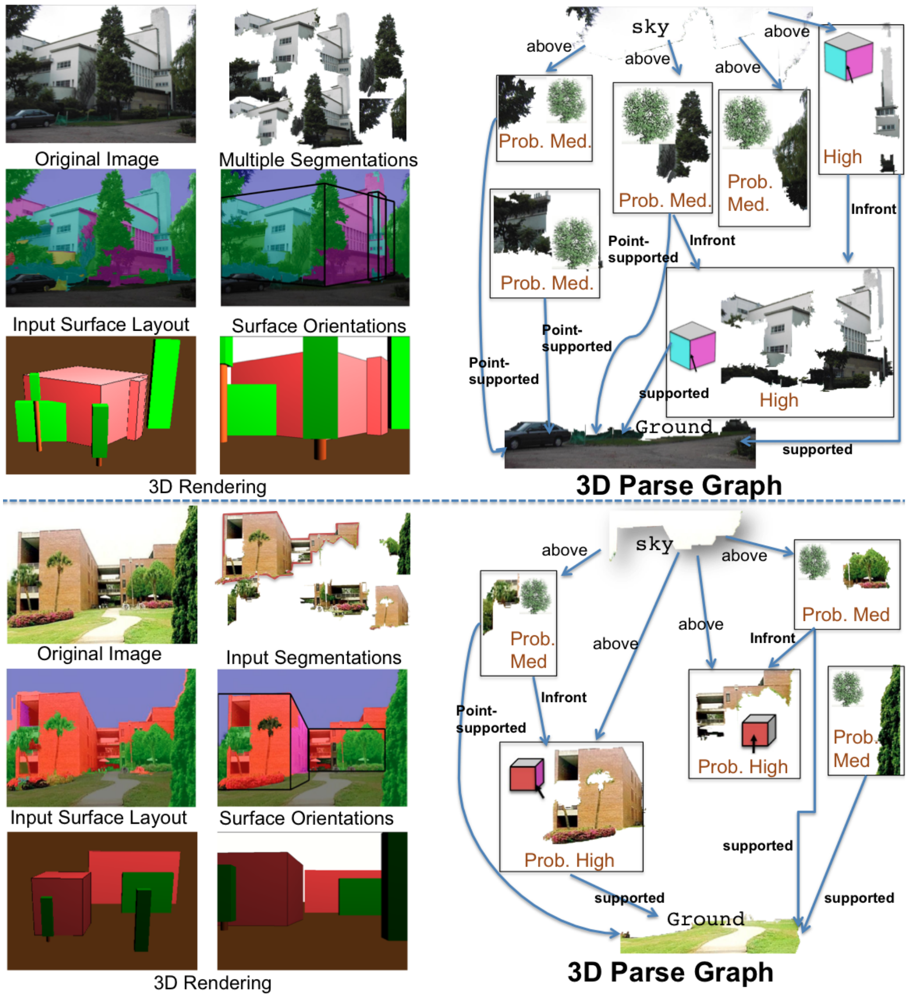

Fig. 8. Some qualitative results obtained using our approach. (Top) Our approach combines the two faces of building separated due to presence of occluder. (Bottom) Our approach correctly labels the right face by rejecting bad segments using mechanics.

because there is no baseline approach. For surface layout estimation, we compare against the state-of-the-art approach [9] which combines occlusion boundary reasoning and surface layout estimation (we removed their recognition component from the system). On the main geometric classes ("ground","vertical" and "sky"), our performance is nearly identical, so we focus on vertical sub-classes (frontal, right-facing, left-facing, porous and solids). For this comparison, we discard the super-pixels belonging to ground and sky and evaluate the performance over the vertical super-pixels. With this evaluation metric, [9] has an average performance of 68.8%. whereas our approach performs at 73.72%. Improving

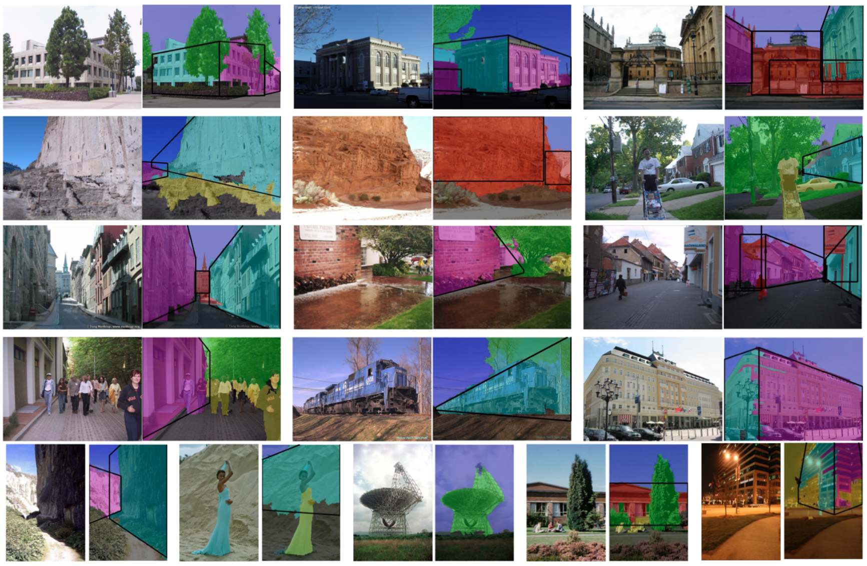

Fig. 9. Additional results to show the qualitative performance on wide variety of scenes.

vertical subclass performance on this dataset is known to be extremely hard; in fact the two recent papers on the topic [15, 12] show no improvement over [9].

	Surface Layout	Segmentation	Density Class.
Hoiem et. al (CVPR 2008)	68.8%	0.6532	-
This paper	73.72%	0.6885	69.32%

Table 1: Quantitative Evaluation

We compare the segmentation performance to [9] on 50 images whose ground truth (segmented regions and occlusion boundaries) is publicly available [10]. For comparing the block segmentation performance we use the Best Spatial Support (BSS) metric. We compute the best overlap score of each ground truth segment and then average it over all ground-truth segments to obtain the BSS score. As can be seen, our approach improves the segmentation performance of [9] by approximately 5.4%. We also evaluated the importance of different terms in the cost function. Without the ground/sky contact term, the surface layout performance falls by 1.9%. The removal of physical stability terms cause the surface layout and segmentation performance to fall by 1.5% and 1.8% respectively.

Acknowledgement: This research was supported by NSF Grant IIS-0905402 and Guggenheim Fellowship to Alexei Efros.

References

1. Blum, M., Griffith, A., Neumann, B.: A stability test for configuration of blocks. In: TR-AI Memo (1970)
2. Brand, M., Cooper, P., Birnbaum, L.: Seeing physics, or: Physics is for prediction. In: Physics-based modeling in computer vision (1995)

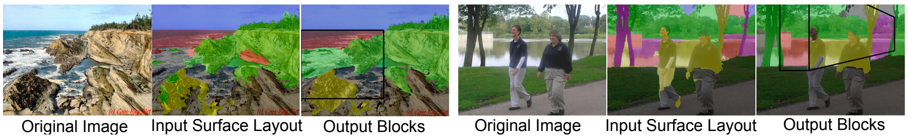

Fig. 10. Failure Cases: We fail to recover proper geometry when [8] is confident about the wrong predictions. Our approach also hallucinates blocks whenever there is a possible convex corner. For example, in the second image the wrong surface layout predicts a false corner and our approach strengthens this volumetric interpretation

3. Delage, E., Lee, H., Ng., A.Y.: A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image. In: CVPR (2006)
4. Gould, S., Fulton, R., Koller, D.: Decomposing a scene into geometric and semantically consistent regions. In: ICCV (2009)
5. Gupta, A., Davis, L.: Beyond nouns: Exploiting prepositions and comparative adjectives for learning visual classifiers. In: ECCV (2008)
6. Hedau, V., Hoiem, D., Forsyth, D.: Recovering the spatial layout of cluttered rooms. In: ICCV (2009)
7. Hedau, V., Hoiem, D., Forsyth, D.: Thinking inside the box: Using appearance models and context based on room geometry. In: ECCV (2010)
8. Hoiem, D., Efros, A.A., Hebert, M.: Recovering surface layout from an image. In: IJCV (2007)
9. Hoiem, D., Efros, A.A., Hebert, M.: Closing the loop on scene interpretation. In: CVPR (2008)
10. Hoiem, D., Stein, A., Efros, A.A., Hebert, M.: Recovering occlusion boundaries from a single image. In: ICCV (2007)
11. Ikeuchi, K., Suehiro, T.: Toward an assembly plan from observation: Task recognition with polyhedral objects. In: Robotics & Automation, (1994)
12. Lazebnik, S., Raginsky, M.: An empirical bayes approach to contextual region classification. In: CVPR (2009)
13. Lee, D., Hebert, M., Kanade., T.: Geometric reasoning for single image structure recovery. In: CVPR (2009)
14. Nedovic, V., Smeulders, A., Redert, A., Geusebroek, J.: Stages as models of scene geometry. In: PAMI (2010)
15. Ramalingam, S., Kohli, P., Alahari, K., Torr, P.: Exact inference in multi-label crfs with higher order cliques. In: CVPR (2008)
16. R.Brooks, Creiner, R., Binford, T.: The acronym model-based vision system. In: IJCAI (1979)
17. Roberts, L.: Machine perception of 3-d solids. In: PhD. Thesis (1965)
18. Saxena, A., Sun, M., Ng, A.Y.: Make3d: Learning 3d scene structure from a single still image. PAMI (2009)
19. Shotton, J., Johnson, M., Cipolla, R.: Semantic texton forests for image categorization and segmentation. In: CVPR (2008)
20. Siskind, J.: Visual event classification via force dynamics. In: AAAI (2000)
21. Winston, P.H.: The mit robot. In: Machine Intelligence (1972)
22. Yu, S., Zhang, H., Malik., J.: Inferring spatial layout from a single image via depth-ordered grouping. In: CVPR Workshop (2008)
23. Zhu, S., Mumford, D.: A stochastic grammar of images. In: Found. and Trends. in Graph. and Vision (2006)