# Single Image 3D Without a Single 3D Image

David F. Fouhey[1], Wajahat Hussain[2], Abhinav Gupta[1], Martial Hebert[1]
[1] Robotics Institute, Carnegie Mellon University, USA
[2] Aragón Institute of Engineering Research (I3A), Universidad de Zaragoza, Spain

## Abstract

*Do we really need 3D labels in order to learn how to predict 3D? In this paper, we show that one can learn a mapping from appearance to 3D properties without ever seeing a single explicit 3D label. Rather than use explicit supervision, we use the regularity of indoor scenes to learn the mapping in a completely unsupervised manner. We demonstrate this on both a standard 3D scene understanding dataset as well as Internet images for which 3D is unavailable, precluding supervised learning. Despite never seeing a 3D label, our method produces competitive results.*

## 1. Introduction

Consider the image in Fig. 1. When we see this image, we can easily recognize and compensate for the underlying 3D structure: for example, we have no trouble recognizing the orientation of the bookshelves and the floor. But how can computers do this? Traditionally, the answer is to use a supervised approach: simply collect large amounts of labeled data to learn a mapping from RGB to 3D. In theory, this is mathematically impossible, but the argument is that there is sufficient regularity to learn the mapping from data. In this paper, we take this argument one step further: we claim that there is enough regularity in indoor scenes to learn a model for 3D scene understanding without ever seeing an explicit 3D label.

At the heart of our approach is the observation that images are a product of two separate phenomena. From a graphics point of view, the image we see is a combination of (1) the coarse scene geometry or meshes in our coordinate frame and (2) the texture in some canonical representation that is put on top of these meshes. For instance, the scene in Fig. 1 is the combination of planes at particular orientations for the bookshelf and the floor, as well as the fronto-parallel rectified texture maps representing the books and the alphabet tiles. We call the coarse geometry the **3D structure** and the texture maps the **style**[1]. In the 3D world these are distinct, but when viewed as a single image, the signals for both get mixed together with no way to separate them.

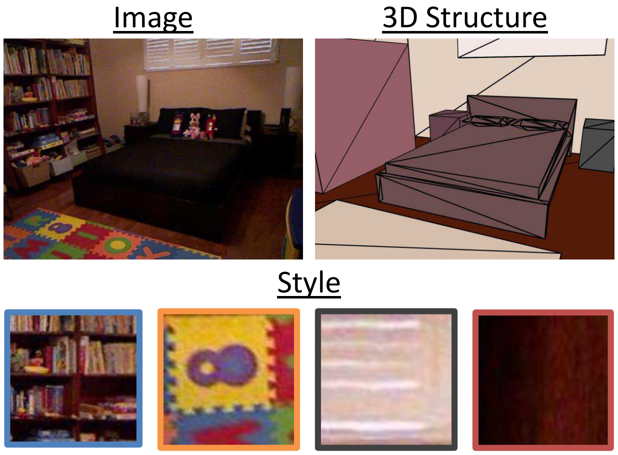

Figure 1. How can we learn to understand images in a 3D way? In this paper, we show a way to do this without using a single 3D label. Our approach treats images as a combination of a 3D model (*3D structure*) with canonical textures (*style*) applied on top. In this paper, we learn *style elements* that recognize texture (e.g., bookshelves, tile floors) rectified to a canonical view. Rather than use explicit supervision, we use the regularity of indoor scenes and a hypothesize-and-verify approach to learn these elements. We thus learn models for single image 3D without seeing a single explicit 3D label. 3D model from [18].

Based on this observation, we propose **style elements** as a basic unit of 3D inference. Style elements detect the presence of style, or texture that is correctly rectified to a canonical fronto-parallel view. They include things like cabinets, window-blinds, and tile floors. We use these style elements to recognize when a texture has been rectified to fronto-parallel correctly. This lets us recognize the orientation of the scene in a hypothesize-and-verify framework: for instance, if we warp the bookshelf in Fig. 2 to look as if it is facing right, our rectified bookshelf detector will respond strongly; if we warp it to look as if it is facing left, our rectified bookshelf detector will respond poorly.

In this paper, we show that we can learn these style elements in an unsupervised manner by leveraging the regular-

[1] Of course, the books in Fig. 1 themselves could be further represented by 3D models. However, in this paper, we ignore this fine change in far structure, and represent the books in terms of their contribution to texture.

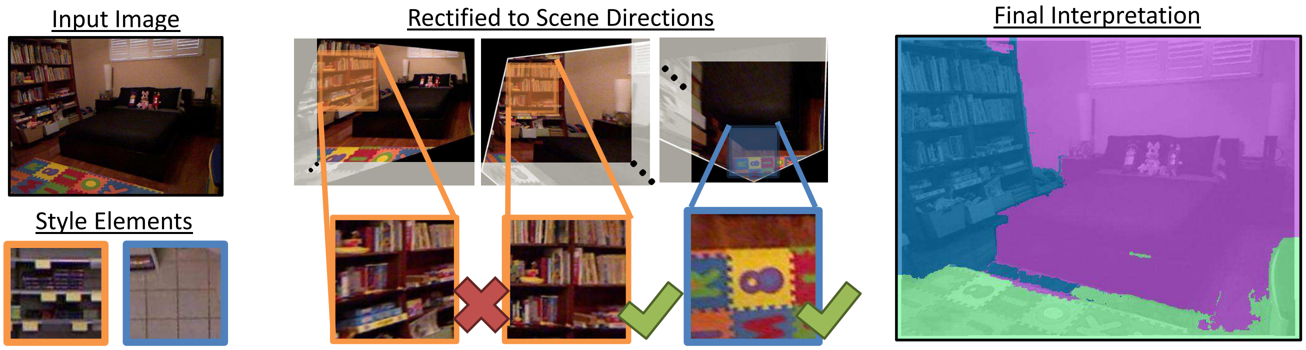

Figure 2. We infer a 3D interpretation of a new scene with style elements by detecting them in the input image rectified to the main directions of the scene. For instance, our bookshelf style-element (orange) will respond well to the bookshelf when it is rectified with the correct direction (facing leftwards) and poorly when it is not. We show how we can automatically learn these style elements, and thus a model for 3D scene understanding without any 3D supervision. Instead, the regularity of the world acts as the supervisory signal.

ity of the world's 3D structure. The key assumption of our approach is that we expect the structure of indoor scenes to resemble an inside-out-box *on average*: on the left of the image, surfaces should face right and in the middle, they should face us. We show how this prior belief can validate style elements in a hypothesize-and-verify approach: we propose a style element and check how well its detections match this belief about 3D structure over a large set of unlabeled images; if an element's detections substantially mismatch, our hypothesis was probably wrong. To the best of our knowledge, this is the first paper to propose an unsupervised learning-based approach for 3D scene understanding from a single image.

**Why unsupervised?** We wish to show that unsupervised 3D learning can be effective for predicting 3D. We do so on two datasets: NYUv2, a standard 3D dataset, and Places-205, which contains scenes not covered by Kinect datasets, such as supermarkets and airports. **Our method is unsupervised and does not use any training data or any pre-trained geometry models; nevertheless:** (1) Our method nearly matches comparable supervised approaches on NYUv2: it is within $< 3^\circ$ of 3DP [16] and better in many metrics on vertical regions. (2) When fused with 3DP, our method achieves state-of-the-art results in 4/6 metrics on NYUv2. (3) As an unsupervised approach, our method can learn from unlabeled Internet images like Places. This is fundamentally impossible for supervised methods, which must resort to pre-trained models and suffer performance loss from the domain shift. Our approach can use this data and outperforms 3DP by 3.7%.

**Why Style Elements?** Operating in this style space lets us learn about the world in a viewpoint-independent fashion. In this paper, we show how this enables us to learn unsupervised models for 3D, but we see broader advantages to this: first, we can detect particular combinations of style and structure that were not present at training time, which is impossible in many existing models; second, since our style elements are viewpoint-independent, we can share information across different viewpoints. We illustrate these advantages in Fig. 3: our method learns one element for all the orientations of the cabinets, but a standard viewer-centric approach learns one element per orientation.

## 2. Related Work

The task of predicting the 3D structure or layout from a single image is arguably as old as computer vision. Early work used extracted contours [33, 23, 7] or geometric primitives [2, 4] and rules to infer structure. However, these primitives were too difficult to detect reliably in natural images, and the community moved towards learning-based approaches. Over the past decade, one dominant paradigm has emerged: at training time, one takes a large collection of images and 3D labels and learns a mapping between the two. The argument for this paradigm is that scenes are sufficiently regular so that such a mapping can be learned from data. The mapping is often learned over segments [34, 22], discriminative patches [16], or pixels [26]. At test time, this local mapping is used on a single image to infer the 3D labels; in other works, it is again presumed that there is such regularity that one can impose even more top-down constraints, such as the Manhattan-world assumption [8], an indoor box model [20, 35], or others [27, 6, 5, 17, 1, 44]. In this work, we tackle the same class of problem, but show that there is enough regularity to even do *unsupervised* learning of models. In particular, we do not use an explicit 3D supervisory signal at any point. Additionally, our method learns across viewpoints, unlike most work on single-image 3D which learn view-dependent representations. The most related work among these methods is [25], which recognizes regions at canonical depths; in contrast, our method is unsupervised and predicts surface normals.

Our approach uses visual elements discovered from a large dataset and draws from a rich literature on discriminative patch-discovery [37, 11, 10, 24, 3]. Like Juneja et

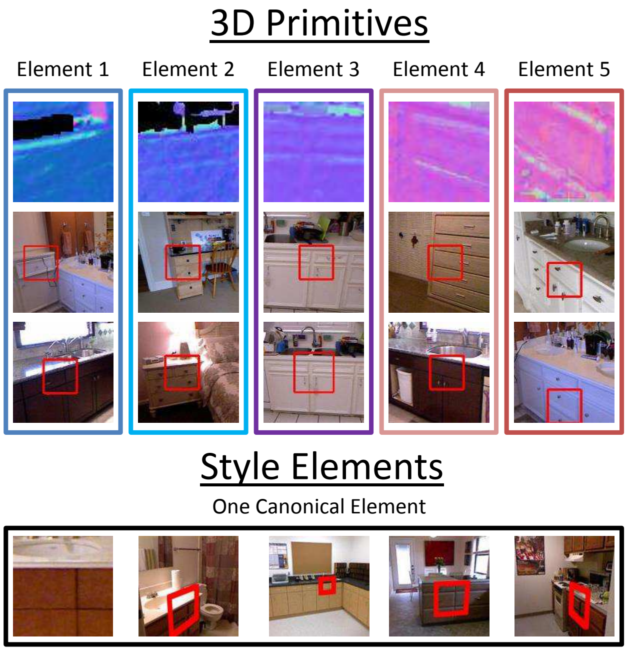

Figure 3. An illustration of sharing enabled by style elements. **Top:** elements from a 3DP model; **Bottom:** (left) a style element and (right) selections from its top 15 discovery-set detections. 3DP detects *each and every* cabinet orientation with a separate element because it is viewer-centric; our model compactly recognizes *all* cabinets orientations with one element.

al. [24], we take a hypothesize-and-verify approach which filters a large set of candidate patch hypotheses by patch detections on a dataset. Among these works, our work is most closely related to discriminative patches for 3D [16, 32] or visual-style-sensitive patches [29]. These frameworks, however, capture the Cartesian product of appearance and the label (style or 3D), meaning that for these frameworks to capture an oven-front at a particular angle, they need to see an oven-front at that particular angle. On the other hand, our approach analytically compensates for 3D structure by rectifying the image data. Thus our elements can predict labels not seen at training time (e.g., an oven at a previously unseen angle). We illustrate this in Fig. 3.

Warping images to a canonical view has been used to boost performance of local patch descriptors for tasks like location recognition [41, 38], in which 3D structure is known or estimated via pre-trained models, or in detection [21, 14], in which it is given at training time. Our work, on the other hand, is unsupervised and jointly reasons about 3D structure and style.

The idea of figuring out the 3D structure by optimizing properties of the unwarped image has been used in shape-from-texture (e.g., [15, 31]) and modern texture analysis [43] and compression [39] approaches. These works are complementary to our own: many obtain a detailed interpretation on presegmented regions or in specific domains by optimizing some criterion such as regularity within one image or a single domain. Our style elements on the other hand, are discovered automatically via the regularity in large amounts of data, and are more general than instance-level texture patterns. They can further interpret novel, generic non-presegmented scenes, although our interpretations on these cluttered scenes are more coarse in comparison.

## 3. Overview

Given a dictionary of discovered style elements, we can use this dictionary of detectors in rectified images to determine the orientation of surfaces: the elements only respond when the scene is rectified correctly. But how do we obtain this dictionary of correctly rectified style elements if we do not have 3D labels?

In Section 4.2, we show how to solve this chicken-and-egg problem with a hypothesize-and-verify approach: we hypothesize a style element, run it on the dataset, and check whether its pattern of detections is plausible. We evaluate the style element's detections by comparing it with a prior that assumes that the world is an inside-out-box. Training thus takes a collection of RGB images as input, and produces a dictionary of detectors as output. In Section 4.3, we describe how to use these style elements to interpret a new image: we run our style elements in a new image, and the detector responses vote for the underlying structure.

As this work is unsupervised, we make some assumptions. We use the Manhattan-world assumption [8] to reduce our label space to three orthogonal directions; we find these directions and rectification homographies for them using vanishing points estimated by [20]. We note, however, that there can be other directions present; we simply do not learn or detect style elements on them. We further assume that the images are upright so we can process the horizontal and vertical directions separately. Finally, our method models each style element as having a single label.

## 4. Method

Our method begins with a discovery set of images and finds style elements that will help us interpret a new image. This task entails determining the orientation of surfaces throughout the discovery set so that we can obtain fronto-parallel rectified representations of the texture.

Since we have no explicit 3D labels, this task seems hopeless: in theory, each part of each image could face any direction! We take a hypothesize-and-verify approach that lets us inject knowledge via a prior on the 3D structure of scenes. We guess a large number of style elements by rectifying the images and sampling patches. Most guesses are wrong, but some are right. We identify the correct ones by computing agreement between our prior and what each hypothesis would imply about the 3D structure of the world.

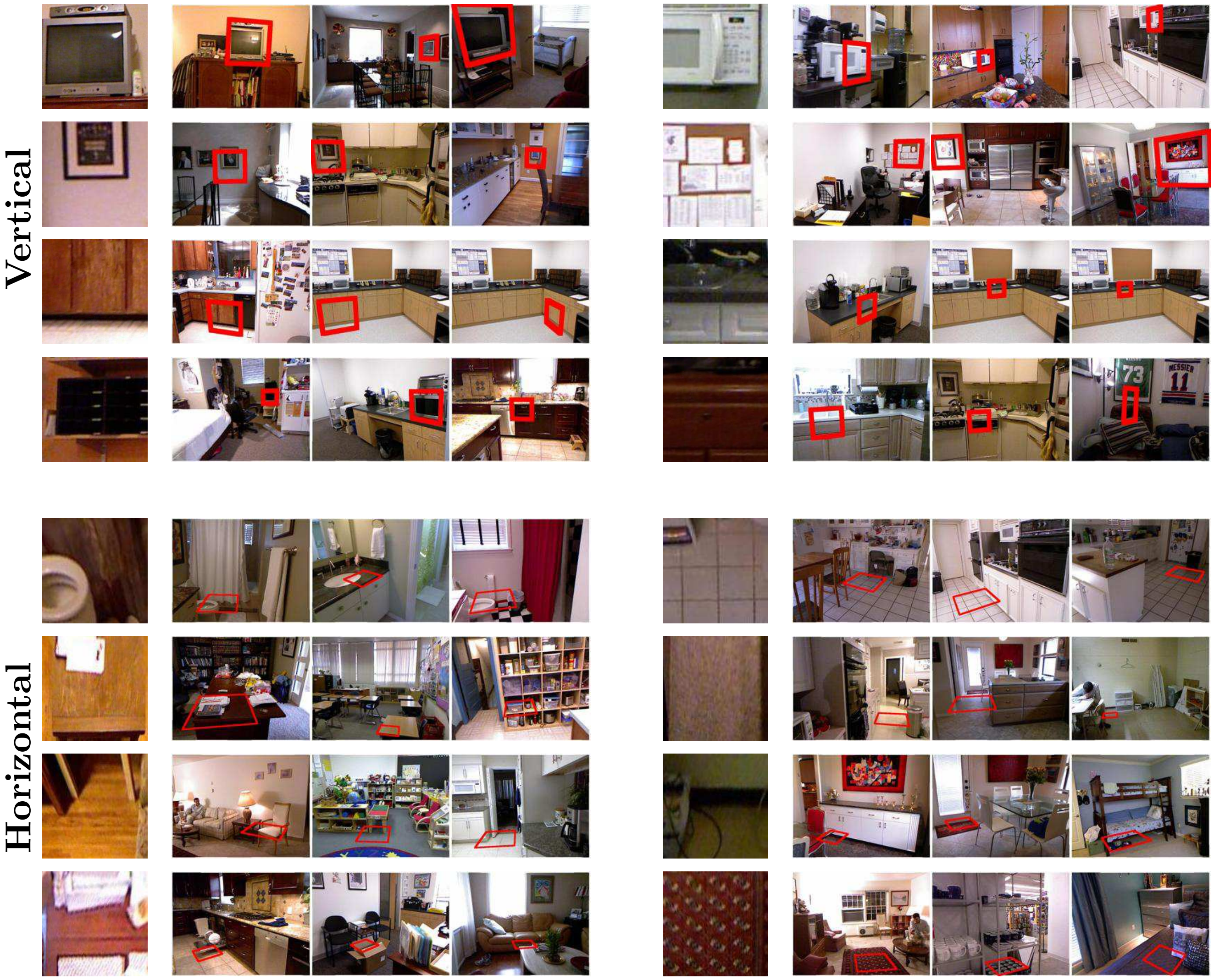

Figure 4. Selected style elements automatically discovered by our method. In each, we show the element on the left and its top detections on the discovery set; these and other detections are used to identify good and bad style elements. Notice that the top detections of most vertical style elements have a variety of orientations.

### 4.1. Prior

Consider the TV on the top-left of Fig. 4. How can we know that it is a good style element (i.e., rectified to be fronto-parallel) without knowing the underlying 3D of the image? While we do not know the 3D at that location, if we looked at the whole discovery set, we would observe a distinct pattern in terms of where TVs appear and in what direction they face: due to the regularity of human scenes, TVs on the left-hand-side of the image tend to face rightwards; on the right-hand-side, they tend to face leftwards. Thus, if we were to run our TV detector over the discovery set, we would expect to see this same distribution. On the other hand, it would be suspicious if we had a detector that only found leftwards facing TVs irrespective of where they appear in the image. We now explain how to formalize this intuition by constructing a prior that gives a probability of each orientation as a function of image location; this lets us score hypothetical style elements by their detections.

Our goal is to build a prior that evaluates the likelihood of a surface orientation as a function of pixel coordinate. Our overarching assumption is that our images are taken with an upright camera inside a box. Then, as in [22], we factor the question of orientation into two independent questions – "is the region vertical or horizontal?" and "if it is vertical, which vertical direction does it face?". We then assume the probability of vertical/horizontal depends on the $y$-coordinate in the image. For the vertical direction, we note that if we assume the world is a box, we can determine how likely each vertical direction is at each pixel as a function of its $x$ coordinate.

We formalize this prior as follows, proceeding analytically since we do not have access to data. Since we expect horizontal surfaces like floors to be more common at

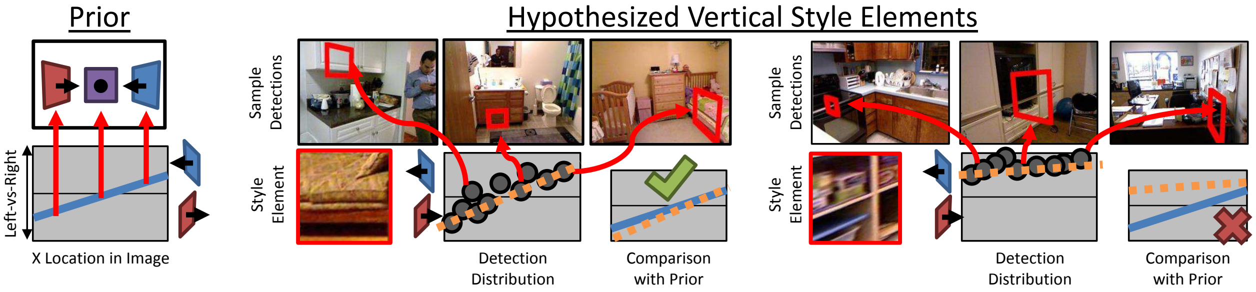

Figure 5. **Selecting hypotheses by their detections.** We compare hypothesized style elements' detections with our prior. We show a good (left) and a bad (right) style hypothesis in red squares. For each, we show a scatter plot of their detections on the discovery set, plotting the $x$-location in the image on the $x$-axis and how left-vs-right the detection is on the $y$-axis. We illustrate a few of these detections: for instance, the bedroom scene in the middle is a leftwards facing detection on the right-side of the image. On the far left, we show what our prior expects – a steady change from right-facing to left-facing. We rank elements by how well their detections match this prior: for instance, we reject the incorrect style element on the right since it predicts that everything faces left, which is unlikely under the prior.

the bottom of the image, we model the vertical/horizontal distinction with a negative exponential on the $y$-location, $\propto \exp(-y^2/\sigma^2)$. Since the camera is upright, the horizon determines the sign of the horizontal directions. For vertical directions, we assume the camera is inside a box with aspect ratio $\sim$ Uniform$[1,2]$ and all rotations equally likely. The likelihood of each direction (left-to-right) as a function of $x$ location can then be obtained in a Monte-Carlo fashion: we histogram normals at each location over renderings of 100K rooms sampled according to the assumptions.

### 4.2. Hypothesizing-and-Verifying Style Elements

Now that we have a way to verify a style element, we can use it in a hypothesize-and-verify approach. We first explain how we generate our hypotheses and then how we use the prior introduced in the the previous section to verify hypothesized style patches.

We first need a way to generate hypotheses. Unfortunately, there are an infinite number of possible directions to try at each pixel. However, if we assume the world is a box, our search space is dramatically smaller: there are only 6 possible directions and these can be obtained by estimating Manhattan-world vanishing points in the image. Once we have rectified the image to these main scene directions, we sample a large collection ($\approx 25K$ total) of patches on these rectified images. Each patch is converted to a detector via an ELDA detector [19] over HOG [9]. Most patches will be wrong because the true scene geometry disagrees with them. One wrong hypothesis appears on the right of Fig. 5 in which a shelf has been rectified to the wrong direction.

We sift through these hypotheses by comparing what their detection pattern over the discovery set implies about 3D structure with our prior. For instance, if a style patch corresponds to a correctly rectified TV monitor, then our detections should, on average, match our box assumption. If it corresponds to an incorrectly rectified monitor then it will not match. We perform this by taking the ELDA detector for each patch and looking at the location and implied orientations of the top $1K$ detections over the training set. Since our prior assumes vertical and horizontal are separate questions, we have different criteria for each. For vertical surfaces, we compute average orientation as a function of $x$ location and compare it to the average orientation under the prior, using the mean absolute difference as our criterion. For horizontal surfaces, our prior assumes that $x$ location is independent from horizontal sign (i.e., floors do not just appear on the left); we additionally do not expect floor to share many style elements with ceilings. We thus compute the correlation between $x$ and horizontal sign and the purity of up-vs-down labelings in the top firings. We illustrate this for two hypothesized vertical style elements in Fig. 5.

We use these criteria to rank a collection of hypothetical vertical and horizontal style elements. Our final model is the top 500 from each. We show some of the style elements we discover on NYU v2 in Fig. 4.

### 4.3. Inference

Given a new image and our style elements, we combine our prior and detections of the style elements to interpret the scene. We extract three directions from vanishing points to get our label space and run the style elements on the rectified images. The detections and the prior then vote for the final label. We maintain a multinomial distribution at each pixel over both whether the pixel is horizontal-vs-vertical and the vertical direction. Starting with the prior, we add a likelihood from detections: we count overlapping detections agreeing with the direction, weighted by score. We then take the maximum response, deciding whether the pixel is horizontal or vertical, and if the latter, the vertical orientation.

Our method produces good interpretations in many places, but does not handle ambiguous parts like untextured carpets well. These ambiguities are normally handled by transferring context [16] or doing some form of learned rea-

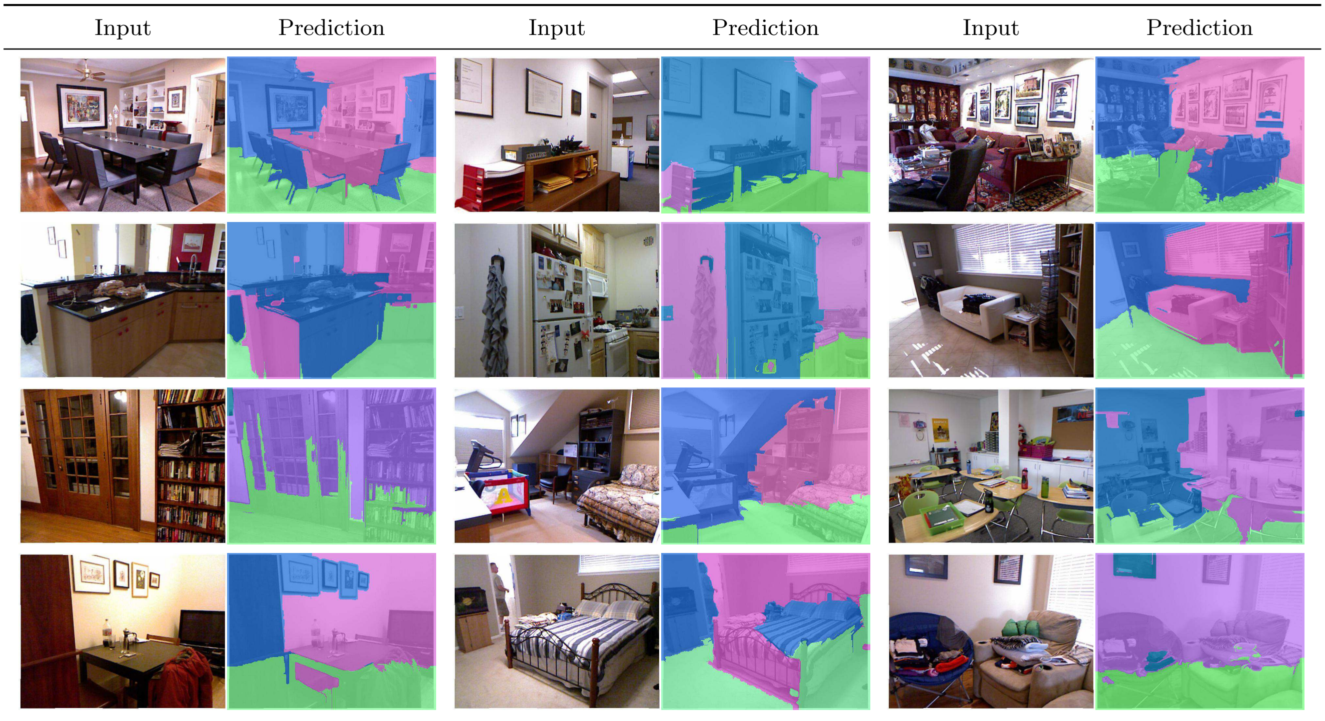

Figure 6. Sample results on NYUv2. First two rows: selected; last two row: random. In row 1, notice that the sides of objects are being labeled (discernible via conflicting colors), even though this severely violates the prior. In row 2, notice that our predictions can extend across the image or form convex corners: even though our prior is a box, we ignore in light of evidence from style elements.

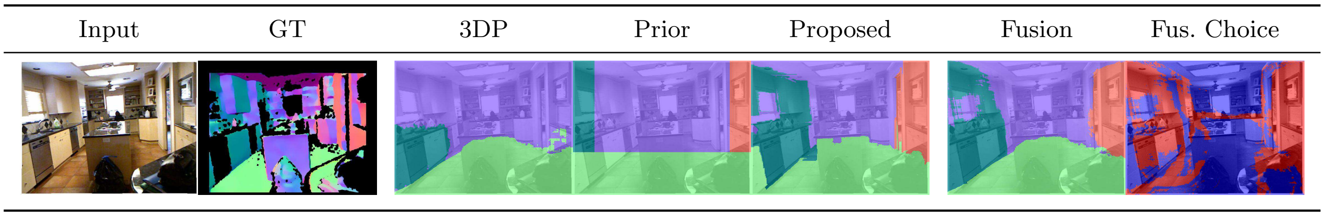

Figure 7. Comparison with the *supervised* 3DP method. The methods have complementary errors and we can fuse their results: 3DP often struggles with near-perpendicular surfaces; however these are easy to recognize once rectified by the proposed method. Our method has more trouble distinguishing floors from walls. We show the fusion result and which prediction is being used (Red: proposed; blue: 3DP).

soning [20, 35, 27, 17]. Without explicit 3D signal, we rely on unsupervised label propagation over segments: we extract multiple segmentations by varying the parameters of [13][2]. Each segment assumes the mode of its pixels, and the final label of a pixel is the mode over the segmentations.

### 4.4. Implementation Details:

We finally report a number of implementation details of our method; more details appear in the supplement. *Patch representation:* Throughout the approach, we use HOG features [9] with a $8 \times 8$ pixel cells at a canonical size of 80 pixels. *Rectification:* we obtain Manhattan-world vanishing points from [20] and rectify following [42]: after autocalibration, the remaining parameters up to a similarity transform are determined via vanishing point orthogonality; the similarity transform is handled by aligning the Manhattan directions with the image axes and by operating at multiple scales. Sample rectified images appear in Fig. 2: our detectors are discovered and tested on these images. At test time, we max-pool detector responses over multiple rectifications per vertical direction. *Initial Patch Pool:* Our hypotheses are obtained by rectifying each image in the discovery set to the scene directions and following the sampling strategy of [37] while rejecting patches whose corresponding quadrilateral has area $< 100^2$ pixels.

## 5. Experimental Validation

We now describe experiments done to validate the approach. We are guided by the following questions: (1) How well does the method work? (2) Can the approach be combined with supervised methods? and (3) Are there scenarios that only an unsupervised approach can handle?

To answer the first two questions, we use the NYUv2

[2] $(\sigma = 0.5, 1, 1.5, 2; k = 100, 200; \min = 50, 100)$

[36] dataset, which has gained wide acceptance; we find that our method does nearly as well as a comparable supervised method and that a simple learned fusion of the methods matches or surpasses the state-of-the-art in 4/6 metrics among methods not using the larger video dataset. To answer the final question, we use the Places-205 dataset [45], which has many locations not covered by Kinect datasets. Supervised approaches must resort to a model trained on existing datasets, but our method can adapt to the dataset. We find that this enables us to outperform a comparable supervised method by a large margin (3.7%).

### 5.1. Experiments on NYU v2

We first document our experimental setup. We follow standard protocols and compare against the state-of-the-art.

**Data:** We use the standard splits of [36]. We use the ground-truth normals from [26] but found similar conclusions on those from [16].

**Evaluation Criteria:** As introduced by [16], we evaluate results on a per-pixel basis over all over valid pixels. We report the mean, median, RMSE and the fraction of pixels with error below a threshold $t$, or PGP-t (percent good pixels) for $t = 11.25°, 22.5°, 30°$. Like [22], our model breaks the task into a vertical/horizontal problem and a vertical subcategory problem. We evaluate both, and evaluate the vertical task on surfaces within 30° of the y-axis.

**Baselines:** We stress that our goal as an unsupervised method is not to outperform supervised methods but instead to show that our approach is effective. We report results of all methods that could be considered state-of-the-art at the time of submission, including even those from methods using the much larger video dataset [40, 12]. The most informative comparison is with the Manhattan-world version of 3DP [16] because it keeps two sources of variance fixed, the base feature and the Manhattan-world assumption.

**Combining with supervised methods:** We learn a model, termed *3DP+Prop* that fuses our method with 3DP. Following [30], we learn random forests (100 trees, cross-validated min-children) on training data to predict whether each method's outputs are within 22.5° of the ground-truth. We train separate forests for our method before segmentation propagation, its vertical predictions only, and 3DP. We use for features the confidences and normals from all methods and the image location. At test time, we take the prediction that is most likely to be correct.

**Results:** We first report qualitative results in Fig. 6. Our method is unsupervised but obtains an accurate interpretation on most scenes. The method frequently picks up on small details, for instance, the right-wards facing chair back (1st row, left) and the side of the shelving in (1st, middle). These small details, as well as the correct inwards-pointing box of the refrigerator (2, middle), are unlikely under the box prior and demonstrate that the method is not simply regurgitating the prior. Our unsupervised propagation is sometimes too aggressive, as seen in (2nd, right, under the blinds; 3rd, left, on the floor), but helps with boundaries.

Table 1. Evaluation on all pixels on NYU v2. The most informative comparison is with 3DP and UNFOLD. Our unsupervised approach nearly matches 3DP and in combination with 3DP, obtains state-of-the-art results in 4/6 metrics. Starred methods do not use the Manhattan-world assumption.

| | Summary Stats. (°) | | | % Good Pixels | | |
|---|---|---|---|---|---|---|
| | (Lower Better) | | | (Higher Better) | | |
| | Mean | Median | RMSE | 11.25° | 22.5° | 30° |
| 3DP+Prop. | 34.6 | **17.5** | 48.7 | **40.6** | **54.7** | **59.7** |
| Ladicky* [26] | **33.5** | 23.1 | **44.6** | 27.7 | 49.0 | 58.7 |
| UNFOLD [17] | 35.2 | 17.9 | 49.6 | **40.5** | 54.1 | 58.9 |
| 3DP [16] | 36.3 | 19.2 | 50.4 | 39.2 | 52.9 | 57.8 |
| Proposed | 38.6 | 21.7 | 52.6 | 36.7 | 50.6 | 55.4 |
| Lee et al. [28] | 43.8 | 35.8 | 55.8 | 26.8 | 41.2 | 46.6 |
| (With External Data) | | | | | | |
| Wang [40] | 26.9 | 14.8 | -NR- | 42.0 | 61.2 | 68.2 |
| Eigen* [12] | 23.7 | 15.5 | -NR- | 39.2 | 62.0 | 71.1 |

Table 2. Ablative analysis

| | Mean | Median | RMSE | 11.25° | 22.5° | 30° |
|---|---|---|---|---|---|---|
| Full | 38.6 | 21.7 | 52.6 | 36.8 | 50.6 | 55.4 |
| No Segm. | 39.6 | 23.4 | 53.5 | 35.7 | 49.3 | 54.0 |
| Prior | 43.2 | 30.2 | 56.7 | 33.1 | 45.2 | 49.8 |
| 3DP on Prior | 41.7 | 27.0 | 55.6 | 34.6 | 47.1 | 51.6 |

Table 3. **Vertical** evaluation. Our method outperforms a 3DP on 3/6 metrics, but fusing the two gives a substantial boost

| | Summary Stats. (°) | | | % Good Pixels | | |
|---|---|---|---|---|---|---|
| | (Lower Better) | | | (Higher Better) | | |
| | Mean | Median | RMSE | 11.25° | 22.5° | 30° |
| Prior | 39.3 | 26.0 | 52.0 | 46.7 | 46.7 | 53.2 |
| Proposed | **33.9** | **19.7** | **46.5** | 35.1 | **53.2** | **59.7** |
| 3DP [16] | **33.9** | 19.9 | **46.5** | **36.4** | 52.6 | 58.8 |
| 3DP+Prop. | 32.1 | 18.0 | 44.6 | 37.5 | 55.2 | 61.5 |

We report quantitative results in Table 1. Our method is always within 2.5° of the most immediately comparable supervised method, 3DP, and is always within 3.8° of [17], which fuses multiple models in a complex MIQP formulation. We also substantially outperform [28], the other unsupervised method for this task. Our simple fusion of 3DP and our method obtains state-of-the-art results in 4/6 metrics among methods using only the original 795 training images. Since the Manhattan-world assumption is rewarded by some metrics and not by others, fair comparison with non-Manhattan-world methods such as [26, 12] is difficult: e.g., on PGP-11.25, due to our use of vanishing points, our method is within 2.4% of [12], which uses orders of magnitude more data.

We report ablative analysis in Table 2: we greatly outperform relying on only the prior (i.e., not using detection ev-

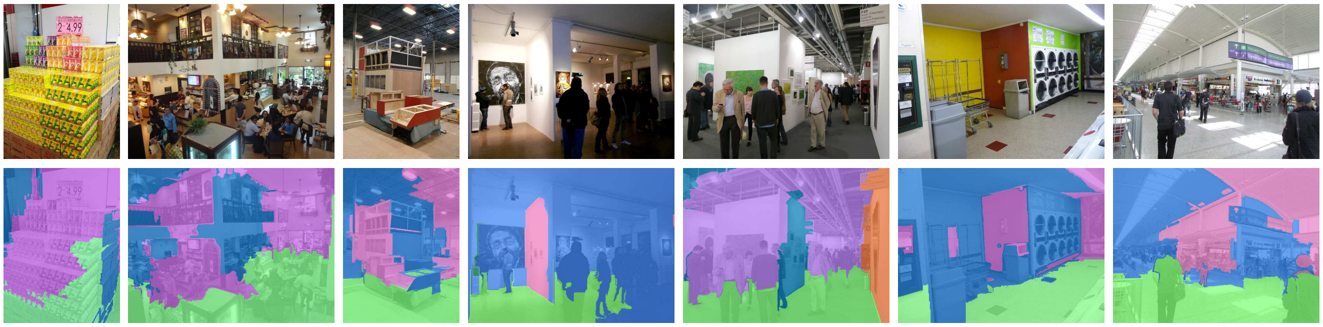

Figure 8. Results on Places-205. Note the stark contrast with NYUv2. Our method can learn from this new data despite a lack of labels.

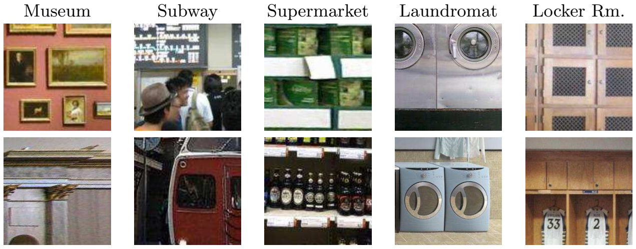

Figure 9. Example vertical elements learned on Internet images.

Table 4. Accuracy on Places-205, subdivided by category.

| | Airport | Art Gallery | Conference. | Locker Rm. | Laundromat | Museum | Restaurant | Shoe Shop | Subway | Supermarket | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3DP | 50.1 | 64.2 | 63.0 | 65.9 | 65.1 | 58.9 | 57.6 | 59.9 | **58.2** | 49.3 | 59.2 |
| FC-[40] | 51.6 | 70.3 | **71.2** | **69.4** | **71.9** | 61.2 | **63.2** | **64.0** | 56.2 | 57.7 | **63.7** |
| Prop. | **54.0** | **71.1** | 64.3 | 67.8 | 71.4 | **64.0** | 60.5 | 62.1 | 52.3 | **61.9** | 62.9 |

idence), especially on the median, and segmentation helps but does not drive performance. Training 3DP using label maps from the prior yielded worse performance. This is because 3DP relies heavily on junctions and edges in the normal map, and the the prior does not provide this detailed information. We found our method to be insensitive to parameters: changing the box prior's aspect ratio or the prior weight by a factor of two yields changes $< 0.7^\circ$ and $< 0.6\%$ across metrics; many settings produced better results than the settings used (more details in the supplement).

Our result is better in relative terms on the vertical task, as can be seen in in Table 3: it matches 3DP in 2 metrics and bests it in 3. This is because many indoor horizontal surfaces are defined by location and lack of texture, which our single-plane HOG patch approach cannot leverage; 3DP, on the other hand, learns structured patches and its vocabulary captures horizontal surfaces via edges and corners. Again, fusing our method with 3DP improves results.

## 5.2. Internet Images

We now investigate tasks that only an unsupervised method can do. Suppose one wants to interpret pictures of supermarkets, airport terminals, or another place not covered by existing RGBD datasets. The only option for a supervised approach (besides collecting new data at great expense) is to use a pre-trained model. However, with our unsupervised 3D approach, we can learn a model from images alone.

**Data:** We collected a subset of 10 categories[3] from the Places-205 dataset [45] that are not present in 3D datasets and annotated them with Manhattan-world labelings. We took at most 700 images from each category, and set aside 200 for evaluation. We sparsely annotated 10 superpixels in each image, each randomly selected to avoid bias; each could be labeled as one of the 6 possible Manhattan normals or passed if multiple directions were present. Since our label space is Manhattan world, we removed images where vanishing points could not be estimated, identified as ones where the independent estimates of [28, 20] disagree.

**Results:** We learned an unsupervised model on each category and compared its performance with models pretrained on NYU, including 3DP and a fully convolutional variant of [40] which obtains within $1.1\%$ PGP-30 on NYU. Note that standard supervised methods cannot compensate for this shift. We show qualitative results illustrating the dataset and our approach in Fig. 8: even NYU contains no washing machines, but our approach can learn elements for these from Internet data, as seen in Fig. 9. On the other hand, pretrained methods struggle to adapt to this new data. We report results in Table 4. Our approach outperforms 3DP, pretrained on NYUv2, in 9/10 categories and overall by $3.7\%$ and outperforms [40] in 4/10 categories, with gains as large as $3.9\%$. Our labels are sparse so we verify the gain over 3DP with a $95\%$ bootstrapped confidence interval; our approach consistently outperforms 3DP ($[2.7, 4.8]$).

**Acknowledgments:** This work was supported by a NDSEG fellowship to DF, Dirección General de Investigación of Spain (project DPI2012-32168) to WH, Bosch Young Faculty Fellowship to AG, ONR MURI N000141010934, NSF IIS-1320083, and a gift from Bosch Research & Technology Center. DF thanks Ishan Misra and Abhinav Shrivastava for helpful conversations.

[3] Airport, art gallery, conference room, locker room, laundromat, museum, restaurant, shoe shop, subway, supermarket

## References

[1] S. Y. Bao, A. Furlan, L. Fei-Fei, and S. Savarese. Understanding the 3D layout of a cluttered room from multiple images. In *WACV*, 2014. 2

[2] T. Binford. Visual perception by computer. In *IEEE Conference on Systems and Controls*, 1971. 2

[3] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3D human pose annotations. In *ICCV*, 2009. 2

[4] R. Brooks, R. Creiner, and T. Binford. The ACRONYM model-based vision system. In *IJCAI*, 1979. 2

[5] Y.-W. Chao, W. Choi, C. Pantofaru, and S. Savarese. Layout estimation of highly cluttered indoor scenes using geometric and semantic cues. In *ICIAP*, 2013. 2

[6] W. Choi, Y. W. Chao, C. Pantofaru, and S. Savarese. Understanding indoor scenes using 3D geometric phrases. In *CVPR*, 2013. 2

[7] M. Clowes. On seeing things. *Artificial Intelligence*, 2:79–116, 1971. 2

[8] J. Coughlan and A. Yuille. The Manhattan world assumption: Regularities in scene statistics which enable bayesian inference. In *NIPS*, 2000. 2, 3

[9] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 5, 6

[10] C. Doersch, A. Gupta, and A. A. Efros. Mid-level visual element discovery as discriminative mode seeking. In *NIPS*, 2013. 2

[11] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros. What makes Paris look like Paris? *SIGGRAPH*, 31(4), 2012. 2

[12] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. *CoRR*, abs/1411.4734, 2014. 7

[13] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2), 2004. 6

[14] S. Fidler, S. Dickinson, and R. Urtasun. 3D object detection and viewpoint estimation with a deformable 3D cuboid model. 2012. 3

[15] D. Forsyth. Shape from texture without boundaries. In *ECCV*, 2002. 3

[16] D. F. Fouhey, A. Gupta, and M. Hebert. Data-driven 3D primitives for single image understanding. In *ICCV*, 2013. 2, 3, 5, 7

[17] D. F. Fouhey, A. Gupta, and M. Hebert. Unfolding an indoor origami world. In *ECCV*, 2014. 2, 6, 7

[18] R. Guo and D. Hoiem. Support surface prediction in indoor scenes. In *ICCV*, 2013. 1

[19] B. Hariharan, J. Malik, and D. Ramanan. Discriminative decorrelation for clustering and classification. In *ECCV*, 2012. 5

[20] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *ICCV*, 2009. 2, 3, 6, 8

[21] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. In *ECCV*, 2010. 3

[22] D. Hoiem, A. Efros, and M. Hebert. Geometric context from a single image. In *ICCV*, 2005. 2, 4, 7

[23] D. Huffman. Impossible objects as nonsense sentences. *Machine Intelligence*, 8:475–492, 1971. 2

[24] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *CVPR*, 2013. 2, 3

[25] L. Ladický, J. Shi, and M. Pollefeys. Pulling things out of perspective. In *CVPR*, 2014. 2

[26] L. Ladický, B. Zeisl, and M. Pollefeys. Discriminatively trained dense surface normal estimation. In *ECCV*, 2014. 2, 7

[27] D. C. Lee, A. Gupta, M. Hebert, and T. Kanade. Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In *NIPS*, 2010. 2, 6

[28] D. C. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *CVPR*, 2009. 7, 8

[29] Y. Lee, A. Efros, and M. Hebert. Style-aware mid-level representation for discovering visual connections in space and time. In *ICCV*, 2013. 3

[30] O. Mac Aodha, A. Humayun, M. Pollefeys, and G. J. Brostow. Learning a confidence measure for optical flow. *TPAMI*, 35(5), 2013. 7

[31] J. Malik and R. Rosenholtz. Computing local surface orientation and shape from texture for curved surfaces. *IJCV*, 23, 1997. 3

[32] A. Owens, J. Xiao, A. Torralba, and W. T. Freeman. Shape anchors for data-driven multi-view reconstruction. In *ICCV*, 2014. 3

[33] L. Roberts. Machine perception of 3D solids. In *PhD Thesis*, 1965. 2

[34] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In *NIPS*, 2005. 2

[35] A. G. Schwing and R. Urtasun. Efficient Exact Inference for 3D Indoor Scene Understanding. In *ECCV*, 2012. 2, 6

[36] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from RGBD images. In *ECCV*, 2012. 7

[37] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *ECCV*, 2012. 2, 6

[38] F. Srajer, A. G. Schwing, M. Pollefeys, and T. Pajdla. MatchBox: Indoor Image Matching via Box-like Scene Estimation. In *3DV*, 2014. 3

[39] H. Wang, Y. Wexler, E. Ofek, and H. Hoppe. Factoring repeated content within and among images. *SIGGRAPH*, 27(3), 2008. 3

[40] X. Wang, D. F. Fouhey, and A. Gupta. Designing deep networks for surface normal estimation. In *CVPR*, 2015. 7, 8

[41] C. Wu, B. Clipp, X. Li, J.-M. Frahm, and M. Pollefeys. 3D model matching with viewpoint-invariant patches (VIP). 2008. 3

[42] A. Zaheer, M. Rashid, and S. Khan. Shape from angular regularity. In *ECCV*, 2012. 6

[43] Z. Zhang, A. Ganesh, X. Liang, and Y. Ma. Tilt: Transform-invariant low-rank textures. *IJCV*, 99(1), 2014. 3

[44] Y. Zhao and S. Zhu. Scene parsing by integrating function, geometry and appearance models. In *CVPR*, 2013. 2

[45] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning Deep Features for Scene Recognition using Places Database. *NIPS*, 2014. 7, 8