Adversarial Structure Matching for Structured Prediction Tasks

Jyh-Jing Hwang[1,2,*] Tsung-Wei Ke[1,*] Jianbo Shi[2] Stella X. Yu[1]
[1]UC Berkeley / ICSI
Berkeley, CA, USA
{jyh,twke,stellayu}@berkeley.edu
[2]University of Pennsylvania
Philadelphia, PA, USA
{jyh,jshi}@seas.upenn.edu

Abstract

Pixel-wise losses, e.g., cross-entropy or L2, have been widely used in structured prediction tasks as a spatial extension of generic image classification or regression. However, its i.i.d. assumption neglects the structural regularity present in natural images. Various attempts have been made to incorporate structural reasoning mostly through structure priors in a cooperative way where co-occurring patterns are encouraged.

We, on the other hand, approach this problem from an opposing angle and propose a new framework, Adversarial Structure Matching (ASM), for training such structured prediction networks via an adversarial process, in which we train a structure analyzer that provides the supervisory signals, the ASM loss. The structure analyzer is trained to maximize the ASM loss, or to emphasize recurring multi-scale hard negative structural mistakes among co-occurring patterns. On the contrary, the structured prediction network is trained to reduce those mistakes and is thus enabled to distinguish fine-grained structures. As a result, training structured prediction networks using ASM reduces contextual confusion among objects and improves boundary localization. We demonstrate that our ASM outperforms pixel-wise IID loss or structural prior GAN loss on three different structured prediction tasks: semantic segmentation, monocular depth estimation, and surface normal prediction.

1. Introduction

Pixel-wise losses, e.g. cross-entropy or L2, are widely used in structured prediction tasks such as semantic segmentation, monocular depth estimation, and surface normal prediction [21, 40, 13, 31], as a spatial extension of generic image recognition [29, 20]. However, the disadvantage of such pixel-wise losses is also obvious due to its additive nature and i.i.d. assumption of predictions: IID losses would yield the same overall error for different spatial distributions of prediction mistakes. Ideally, some mistakes such as implausible and incomplete round wheels should incur more penalty than slightly thinner wheels. Structural reasoning is thus highly desirable for structured prediction tasks.

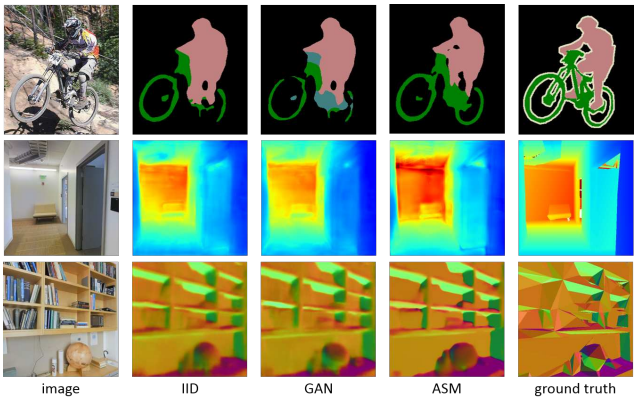

Figure 1: Experimental results on semantic segmentation on PASCAL VOC 2012 [14], monocular depth estimation, and surface normal prediction on Stanford 2D-3D-S [2]. Networks trained using IID loss and shape prior based (GAN) loss mostly fail at confusing contexts (top row) and ambiguous boundaries (bottom 2 rows) whereas our ASM approach improves upon these aspects.

Various attempts have been made to incorporate structural reasoning into structured prediction in a *cooperative* way, including two mainstreams, bottom-up Conditional Random Fields (CRFs) [28, 61] and top-down shape priors [53, 19, 24] or Generative Adversarial Networks (GANs) [17, 41, 22]: **(1)** CRF enforces label consistency between pixels and is commonly employed as a post-processing step [28, 7], or as a plug-in module inside deep neural networks [61, 39] that coordinate bottom-up information. Effective as it is, CRF is usually sensitive to input appearance changes and needs expensive iterative inference. **(2)** As an example of learning top-down shape priors, GAN emerges as an alternative to enforce structural

* Equal contributions.

regularity in the structured prediction space. Specifically, the discriminator network is trained to distinguish the predicted mask from the ground truth mask. Promising as it is, GAN suffers from inaccurate boundary localization as a consequence of generic shape modeling. More recently, Ke *et al.* [24] propose adaptive affinity fields that capture structural information with adaptive receptive fields. However, it is designed specifically for classification and cannot be extended straightforward to regression.

As a result, top-down *cooperative* approaches prefer an additional loss (together with IID) that penalizes more on the anomaly structures that are deemed undesirable. Such trained networks are thus aware of intra-category shape invariance and inter-category object co-occurrences. However, we notice that in real examples as in Fig. 1, complex and deformable shapes and confusing co-occurrences are the most common mistakes in structured prediction especially when the visual cues are ambiguous. As a result, training with shape priors sometimes deteriorates the prediction as shown in the bicycle example. We are thus inspired to tackle this problem from an *opposing* angle: top-down approaches should adapt the focus to confusing co-occurring context or ambiguous boundaries so as to make the structured prediction network learn harder.

We propose a new framework called Adversarial Structure Matching (ASM), which replaces IID losses, for training structured prediction networks via an adversarial process, in which we train a structure analyzer to provide supervisory signals, the adversarial structure matching (ASM) loss. By maximizing ASM loss, or learning to exaggerate structural mistakes from the structured prediction networks, the structure analyzer not only becomes aware of complex shapes of objects but adaptively emphasize those *multi-scale hard negative structural mistakes*. As a result, training structured prediction networks by minimizing ASM loss reduces contextual confusion among co-occurring objects and improves boundary localization. To improve the stability of training, we append a structure regularizer on the structure analyzer to compose a structure autoencoder. By training the autoencoder to reconstruct ground truth, which contains complete structures, we ensure the filters in the structure analyzer form a good structure basis. We demonstrate that structured prediction networks trained using ASM outperforms its pixel-wise counterpart and GAN on the figure-ground segmentation task on Weizmann horse dataset [5] and semantic segmentation task on PASCAL VOC 2012 dataset [14] with various base architectures, such as FCN [40], U-Net [47], DeepLab [7], and PSPNet [60]. Besides the structured classification tasks, we also verify the efficacy of ASM on structured regression tasks, such as monocular depth estimation and surface normal prediction, with U-Net [47] on Stanford 2D-3D-S dataset [2].

2. Related Works

Semantic Segmentation. The field of semantic segmentation has progressed fast in the last few years since the introduction of fully convolutional networks [40, 7]. Both deeper [60, 34] and wider [44, 47, 56] network architectures have been proposed and have dramatically boosted the performance on standard benchmarks like PASCAL VOC 2012 [14]. Notably, multi-scale context information emerges to remedy limited receptive fields, e.g., spatial pyramid pooling [60] and atrous spatial pyramid pooling [8]. Though these methods yield impressive performance w.r.t. mIoU, they fail to capture rich structure information present in natural scenes.

Structure Modeling. To overcome the aforementioned drawback, researchers have explored several ways to incorporate structure information [28, 9, 61, 39, 35, 4, 53, 19, 24]. For example, Chen et al. [7] utilized denseCRF [28] as post-processing to refine the final segmentation results. Zheng et al. [61] and Liu et al. [39] further made the CRF module differentiable within the deep neural network. Besides, low-level cues, such as affinity [49, 42, 38, 4] and contour [3, 6] have also been leveraged to encode image structures. However, these methods either are sensitive to appearance changes or require expensive iterative inference.

Monocular Depth Estimation and Normal Prediction. With large-scale RGB-D data available [50, 2, 57], data-driven approaches [23, 25, 30, 31, 33, 37, 48, 13, 15] based on deep neural networks make remarkable progress on depth estimation and surface normal prediction. Just like in semantic segmentation, some incorporate structural regularization such as CRFs into the system and demonstrate notable improvements [26, 32, 36, 51, 52, 55, 63, 10]. Recently, leveraging depth estimation and semantic segmentation for each other has emerged to be a promising direction to improve structured predictions of both tasks [11, 12, 18, 45, 50]. Others propose to jointly train a network for both tasks [43, 46, 27, 59, 54]. Our work is also along this direction as the ASM framework can train different structured prediction tasks in a consistent manner.

3. Method

We provide an overview of the framework in Fig. 2 and summarize the training procedure in Alg. 1.

3.1. Adversarial Structure Matching

We consider structured prediction tasks, in which a structured prediction network (structured predictor) $S : \mathbf{x} \mapsto \hat{\mathbf{y}}$, which usually is a deep CNN, is trained to map an input image $\mathbf{x} \in \mathbb{R}^n$ to a per-pixel label mask $\hat{\mathbf{y}} \in \mathbb{R}^n$. We propose to train such a structured predictor with another network, structure analyzer. The analyzer $A : \mathbb{R}^n \mapsto \mathbb{R}^k$

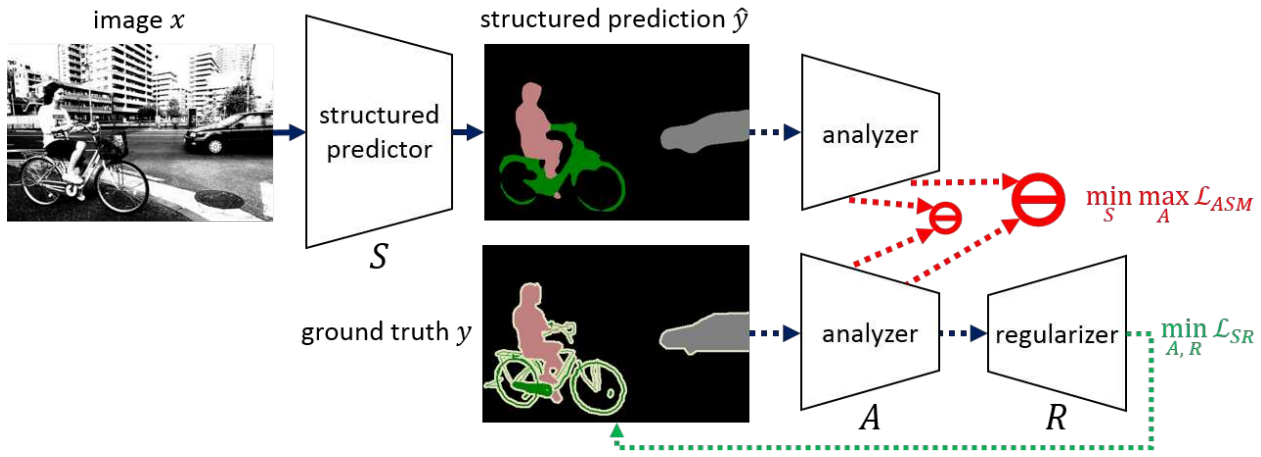

Figure 2: Framework overview: A structure analyzer extracts structure features (red arrows) from structured outputs. The analyzer is trained to maximize an adversarial structure matching (ASM) loss, or discrepancy between multi-scale structure features extracted from ground truth and from predictions of a structured predictor. The analyzer thus learns to exaggerate the hard negative structural mistakes and to distinguish fine-grained structures. The structured predictor on the contrary is trained to minimize the ASM loss. To make sure the filters in the analyzer form a good structure basis, we introduce a structure regularizer, which together with the analyzer, constitutes an autoencoder. The autoencoder is trained to reconstruct ground truth with a structure regularization (SR) loss. Dotted lines denote computations during training only.

extracts k-dimensional multi-layer structure features from either ground truth masks, denoted as $A(\mathbf{y})$, or predictions, denoted as $A(S(\mathbf{x}))$. We train the analyzer to maximize the distance between the structure features from either inputs, so that it learns to exaggerate structural mistakes, or hard negative structural examples, made by the structured predictor. On the contrary, we simultaneously train the structured predictor to minimize the same distance. In other words, structured predictor S and structure analyzer A play the following two-player minimax game with value function $V(S, A)$:

$$\min_S \max_A V(S,A) = \mathbb{E}_{\mathbf{x},\mathbf{y}} \left[\frac{1}{2} \left\| A\left(S(\mathbf{x})\right) - A(\mathbf{y}) \right\|_2^2 \right], \tag{1}$$

that is, we prefer the optimal structured predictor as the one that learns to predict true structures to satisfy the analyzer. Note that the analyzer will bias its discriminative power towards similar but subtly different structures as they occur more frequently through the course of training.

One might relate this framework to GAN [17]. A critical distinction is that GAN tries to minimize the data distributions between real and fake examples and thus accepts a set of solutions. Here, structured prediction tasks require a specific one-to-one mapping of each pixel between ground truth masks and predictions. Therefore, the discrimination of structures should take place for every patch between corresponding masks, hence the name adversarial structure matching (ASM).

It is also related to perceptual loss [16, 58] for style transfer, which uses pretrained CNN features to capture image statistics. We, on the other hand, generalize this concept by adapting the CNN and accepting any dimensional inputs.

3.2. Global Optimality of $S(\mathbf{x}) = \mathbf{y}$ and Convergence

We would like the structured predictor to converge to a good mapping of $\mathbf{y}$ given $\mathbf{x}$, if given enough capacity and training time. To simplify the dynamic of convergence, we consider both structured predictor S and analyzer A as models with infinite capacity in a non-parametric setting.

Proposition 1. *For a fixed S, if $S(\mathbf{x}) \neq \mathbf{y}$, then $\|A^*\left(S(\mathbf{x})\right) - A^*(\mathbf{y})\|_2^2$ is infinitely large for an optimal A.*

Proof. If $S(\mathbf{x}) \neq \mathbf{y}$, there exists an index i such that $S(\mathbf{x})[i] - y[i] = \epsilon$, where $\epsilon \in \mathbb{R} \setminus \{0\}$. Without loss of generality, we assume $S(\mathbf{x})[j] = y[j]$ if $j \neq i$ and let $S(\mathbf{x})[i] = c + \frac{1}{2}\epsilon$ and $y[i] = c - \frac{1}{2}\epsilon$.

We consider a special case where A_l on the i-th dimension of the input is a linear mapping, i.e., $A_l(x[i]) = w_i x[i]$. As A is with infinite capacity, we know there exists A such that

$$\left\| A\left(S(\mathbf{x})\right) - A(\mathbf{y}) \right\|_2^2 \geq \left\| A_l\left(S(\mathbf{x})\right) - A_l(\mathbf{y}) \right\|_2^2 = \left\| w_i \left(c + \frac{1}{2}\epsilon \right) - w_i \left(c - \frac{1}{2}\epsilon \right) \right\|_2^2 = |w_i \epsilon| \tag{2}$$

Note that $\|A_l\left(S(\mathbf{x})\right) - A_l(\mathbf{y})\|_2^2 \to \infty$ as $|w_i| \to \infty$. Thus $\|A^*\left(S(\mathbf{x})\right) - A^*(\mathbf{y})\|_2^2 \to \infty$. □

In practice, parameters of A are restricted within certain range under weight regularization so $\|A^*(S(\mathbf{x})) - A^*(\mathbf{y})\|_2^2$ would not go to infinity.

Corollary 1. *For an optimal A, $S(\mathbf{x}) = \mathbf{y}$ if and only if $A^*(S(\mathbf{x})) = A^*(\mathbf{y})$.*

Proof. $\Rightarrow$ If $S(\mathbf{x}) = \mathbf{y}$, $\|A(S(\mathbf{x})) - A(\mathbf{y})\|_2^2 = \|A(\mathbf{y}) - A(\mathbf{y})\|_2^2 = 0$, for any A. Hence $\|A^*(S(\mathbf{x})) - A^*(\mathbf{y})\|_2^2 = 0$.
$\Leftarrow$ If $A^*(S(\mathbf{x})) = A^*(\mathbf{y})$ or $\|A^*(S(\mathbf{x})) - A^*(\mathbf{y})\|_2^2 = 0$, $S(\mathbf{x}) \neq \mathbf{y}$ contradicts Proposition 1. Hence $S(\mathbf{x}) = \mathbf{y}$. □

Theorem 1. *If (S^*, A^*) is a Nash equilibrium of the system, then $S^*(\mathbf{x}) = \mathbf{y}$ and $V(S^*, A^*) = 0$*

Proof. From Proposition 1, we proved $V(S, A^*) \to \infty$ if $S(\mathbf{x}) \neq \mathbf{y}$. From Corollary 1, we proved $V(S, A^*) = 0$ if and only if $S(\mathbf{x}) = \mathbf{y}$. Since $V(S, A) \geq 0$ for any S and A, the Nash equilibrium only exists when $S^*(\mathbf{x}) = \mathbf{y}$, or $V(S^*, A^*) = 0$. □

From the proofs, we recognize imbalanced powers between the structured predictor and structure analyzer where the analyzer can arbitrarily enlarge the value function if the structured predictor is not optimal. In practice, we should limit the training of the analyzers or apply regularization, such as weight regularization or gradient capping to prevent gradient exploding. Therefore, we train the analyzer only once per iteration with a learning rate that is not larger than the one for structured predictor. Another trick for semantic segmentation is to binarize the predictions $S(x)$ (winner-take-all across channels for every pixel) before calculating ASM loss for analyzer. In this way, the analyzers will focus on learning to distinguish the structures instead of the confidence levels of predictions.

3.3. Reconstructing $\mathbf{y}$ as Structure Regularization

Although theoretically structure analyzers would discover any structural difference between predictions and ground truth, randomly initialized analyzers suffer from missing certain structures in the early stage. For example, if filter responses for a sharp curve are initially very low, ASM loss for the sharp curve will be as small, resulting in inefficient learning. This problem will emerge when training both structured predictors and structure analyzers from scratch. To alleviate this problem, we propose a regularization method to stabilize the learning of analyzers.

One way to ensure the filters in the analyzer form a good structure basis is through reconstructing ground truth, which contains complete structures. If filters in the analyzer fail to capture certain structures, the ground truth mask cannot be reconstructed. Hence, we append a structure regularizer on top of structure analyzer to constitute an autoencoder. we denote the structure regularizer $R : A_t(\mathbf{y}) \mapsto \mathbf{y}$, where $A_t(\cdot)$ denotes features from the structure analyzer,

Algorithm 1: Algorithm for training structured prediction networks using ASM.

for *number of training iterations* **do**

`/* Train structure analyzer */`

(optional) Binarize structured predictions $S(\mathbf{x}^{(i)})$.

Update the structure analyzer A and regularizer R by ascending its stochastic gradient:

$$\nabla_{\theta_{A,R}} \frac{1}{m} \sum_{i=1}^{m} \Big[\frac{1}{2} \Big\| A\left(S(\mathbf{x}^{(i)})\right) - A(\mathbf{y}^{(i)}) \Big\|_2^2 + \lambda IID\left(\mathbf{y}, R\left(A_t(\mathbf{y})\right)\right) \Big]$$

`/* Train structured predictor */`

Sample a minibatch with m images $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\}$ and ground truth masks $\{\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(m)}\}$.

Update the structured predictor S by descending its stochastic gradient:

$$\nabla_{\theta_S} \frac{1}{2m} \sum_{i=1}^{m} \Big\| A\left(S(\mathbf{x}^{(i)})\right) - A(\mathbf{y}^{(i)}) \Big\|_2^2$$

end

The gradient-based updates can use any standard gradient-based learning rule. Structure analyzer A should use a learning rate that is not larger than structured predictor S.

which are not necessarily the same set of features for ASM; hence the reconstruction mapping: $R(A_t(\mathbf{y})) \mapsto \mathbf{y}$. As a result, the final objective function is as follows

$$S^* = \underbrace{\operatorname*{argmin}_S \max_A \mathbb{E}_{\mathbf{x},\mathbf{y}} \left[\frac{1}{2} \|A(S(\mathbf{x})) - A(\mathbf{y})\|_2^2 \right]}_{\text{adversarial structure matching loss}} + \underbrace{\min_{A,R} \lambda \mathbb{E}_{\mathbf{y}} \left[IID\left(\mathbf{y}, R\left(A_t(\mathbf{y})\right)\right) \right]}_{\text{structure regularization loss}}, \quad (3)$$

where $IID\left(\mathbf{y}, R\left(A_t(\mathbf{y})\right)\right)$ is defined for target tasks as:

Semantic segmentation: $-\mathbf{y} \cdot \log R\left(A_t(\mathbf{y})\right)$

Depth estimation: $\|y - R\left(A_t(\mathbf{y})\right)\|_2^2$

Surface normal Prediction: $\left\| \frac{y}{\|y\|_2} - \frac{R\left(A_t(\mathbf{y})\right)}{\|R\left(A_t(\mathbf{y})\right)\|_2} \right\|_2^2$

Note that structure regularization loss is independent to S.

Figure 3: Error maps of predictions and reconstructions by the structure autoencoder on VOC PASCAL 2012. The autoencoder network is able to reconstruct the missing part of certain structures, for example, it completes the round wheel in the front. Note that neither the structure analyzer nor regularizer has access to input images.

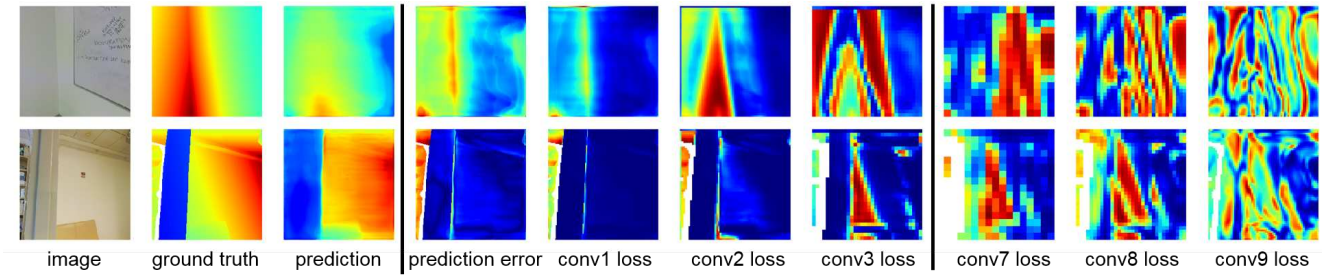

Figure 4: Visualization of loss maps from different layers in the structure analyzer. Examples are from monocular depth estimation on 2D-3D-Semantics [2]. We observe earlier layers capture lower-level structural mistakes as opposed to later layers. It is worth to note that $conv9$, with skip connection from $conv1$, provides low-level structure errors guided by global information. We observe that multi-scale structure supervision is achieved by the multi-layer structure analyzer.

4. Experiments

We demonstrate the effectiveness of our proposed methods on three structure modeling tasks: semantic segmentation, monocular depth estimation, and surface normal prediction. We conduct extensive experiments on all three tasks to compare the same structured prediction networks trained using ASM, GAN, or IID losses. (Note that GAN means training with IID loss and adversarial loss.)

We first give an overview of the datasets and implementation details in Sec. 4.1. Then we analyze what is learned in the structure analyzer to understand the mechanics of ASM in Sec. 4.2. Finally, we present our main results and analyses on segmentation in Sec. 4.3 and on depth estimation and surface normal prediction in Sec. 4.4.

4.1. Experimental Setup

Tasks and datasets. We compare our proposed ASM against GAN and IID losses on the Weizmann horse [5], PASCAL VOC 2012 [14], and Stanford 2D-3D-Semantics [2] datasets. The **Weizmann horse** [5] is a relatively small dataset for figure-ground segmentation that contains 328 side-view horse images, which are split into 192 training and 136 validation images. The **VOC 2012** [14] dataset is a well-known benchmark for generic semantic segmentation which includes 20 object classes and a 'background' class, containing $10,582$ and $1,449$ images for training and validation, respectively. We also conduct experiments on **2D-3D-Semantics** [2] dataset for monocular depth and surface normal prediction. The 2D-3D-S dataset is a large-scale indoor scene benchmark, which consists of $70,496$ images, along with depth, surface normal, instance- and semantic-level segmentation annotations. The results are reported over fold-1 data splits–Area 1,2,3,4,6 ($52,903$ images) for training and Area 5 ($17,593$ images) for testing.

Architectures. For all the structure autoencoders (i.e., analyzer and regularizer), we use U-Net [47] with either 7 conv layers for figure-ground segmentation, 5 conv layers for semantic segmentation and 9 conv layers for depth and surface normal estimation. We conduct experiments on different structured prediction architectures. On horse dataset [5], we use U-Net [47] (with 7 convolutional layers) as the base architecture. On VOC [14] dataset, we carry out experiments and thorough analyses over 3 different architectures with ResNet-101 [20] backbone, including FCN [40], DeepLab [7], and PSPNet [60], which is a highly competitive segmentation model. On 2D-3D-S [2] dataset, we use U-Net [47] (with 18 convolutional layers) as the base architecture for both depth and surface normal prediction. Aside from base architectures, neither extra parameters nor post-processing are required at inference time.

Implementation details on Weizmann horse. We use the poly learning rate policy where the current learning rate equals the base one multiplied by $(1 - \frac{\text{iter}}{\text{max\_iter}})^{0.9}$ with max iterations as 100 epochs. We set the base learning rate as

0.0005 with Adam optimizer for both S and A. Momentum and weight decay are set to 0.9 and 0.00001, respectively. We set the batch size as 1 and use only random mirroring. For ASM, We set $\lambda = 2$ for structure regularization.

Implementation details on VOC dataset. Our implementation follows the implementation details depicted in [8]. We adopt the same poly learning rate policy and set the base learning rate with SGD optimizer as 0.001 for S and 0.0005 for A. The training iterations for all experiments on all datasets are 30K. Momentum and weight decay are set to 0.9 and 0.0005, respectively. For data augmentation, we adopt random mirroring and random resizing between 0.5 and 2 for all datasets. We do not use random rotation and random Gaussian blur. We do not upscale the logits (prediction map) back to the input image resolution, instead, we follow [7]'s setting by downsampling the ground-truth labels for training ($output\_stride = 8$). The crop size is set to 336×336 and batch size is set to 8. We update BatchNorm parameters with $decay = 0.9997$ for ImageNet-pretrained layers and $decay = 0.99$ for untrained layers. For ASM, we set $\lambda = 10$ for structure regularization.

Implementation details on 2D-3D-S. We implement pixel-wise L2 loss and normalized L2 for depth and surface normal prediction as a counterpart to cross-entropy loss for segmentation. We adopt the same poly learning rate policy, and set the base learning rate to 0.01 for both S and A. While the original image resolution is 1080×1080, we down-sample image to half resolution, set "cropsize" to 512×512 for training, and keep 540×540 for testing. We use random cropping and mirroring for data augmentation. The "batchsize" is set to 8, weight decay is set to 0.0005, and models are trained for 120K iterations from scratch.

4.2. Analyses of Structure Analyzer

Before getting to the main results, we verify the structure analyzer actually captures certain structures and visualize what kind of supervisory signals it provides for training the structured predictor.

For segmentation, some objects are rigid and certain shapes are important structures, *e.g.*, rectangles for buses, circles for wheels, and parallel lines for poles. Here, we visualize the prediction, reconstruction, and their error maps in Fig. 3. We observe that that the structure autoencoder network is able to reconstruct the missing part of certain structures, for example, it completes round wheels of bikes. Note that neither the structure analyzer nor regularizer has access to input images.

We further analyze the losses from each layer in the structure analyzer for monocular depth estimation, shown in Fig. 4. We observe that in the early layers where information is local, the errors are near object boundaries. In the middle layers, *e.g.*, layer 2 and 3, the structure analyzer attends to broader context, such as the orientation of walls. In the final layer ($conv9$, with skip connection from $conv1$), low-level structure errors are guided by global information. We observe that multi-scale structure supervision is achieved by the multi-layer structure analyzer.

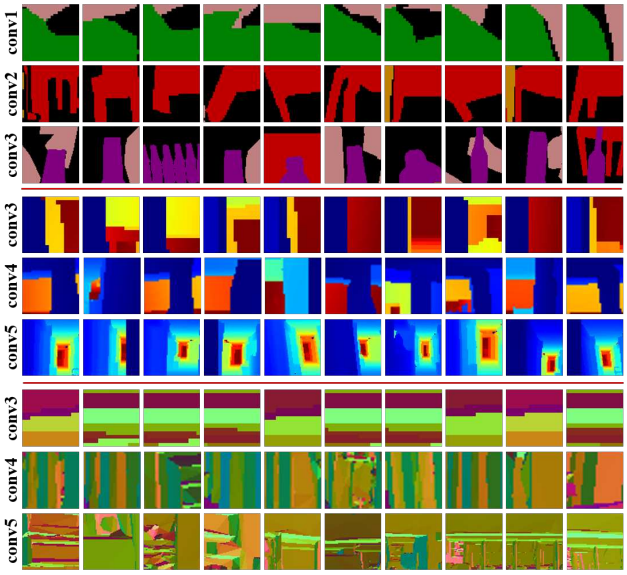

Figure 5: Top 10 input stimuli (in a row) for a filter in the analyzer with maximal activations for semantic segmentation (top 3 rows), depth estimation (middle 3 rows), and surface normal prediction (bottom 3 rows).

To understand what's learned in the analyzer, we are inspired by Zhou *et al.* [62] to visualize features in a CNN by finding the patches that maximally activate a targeted filter. We show in Fig. 5 the top 10 input stimuli for three filters in different layers for each task. We observe that each filter in the analyzer attends to a specific structure, *e.g.*, chairs, bottles, hallways, bookcases, *etc.*.

4.3. Main Results on Segmentation

We evaluate both figure-ground and semantic segmentation tasks via mean pixel-wise intersection-over-union (denoted as **mIoU**) [40]. We first conduct ablation studies on both datasets to thoroughly analyze the effectiveness of using different layers of structure features in the structure analyzer. As summarized in Table 1, using low- to mid-level features (from $conv1$ to $conv2$) of structure analyzers yields the highest performance, 79.62% and 71.60% mIoU on Weizmann horse dataset and VOC dataset, respectively). We also report mIoU on VOC dataset using different base architectures as shown in Table 2. Our proposed method achieves consistent improvements across all three base architectures, boosting mIoU by 3.23% with FCN, 0.51% with DeepLab and 1.31% with PSPNet. ASM is also 0.71% higher than GAN (incorporated with IID loss) on

Loss	Horse mIoU (%)	VOC mIoU (%)
IID	77.28	68.91
ASM (Conv1)	78.14	70.00
ASM (Conv2)	78.15	70.70
ASM (Conv1-2)	**79.62**	71.60
ASM (Conv3)	77.79	70.85
ASM (Conv1-3)	78.11	69.81
ASM w/o rec.	77.83	**72.14**
ASM w/o adv.	76.70	71.26
IID+ASM	78.34	68.49

Table 1: Ablation studies of ASM on Weizmann horse dataset with U-Net and on PASCAL VOC dataset with FCN. Generally, using low- to mid-level features (conv1 and conv2) of structure analyzers yield the best performance. It also shows that reconstruction is not always needed if base networks are pre-trained. DeepLab or PSPNet (not shown here) has the same trend as U-Net.

Base / Loss	mIoU (%)
FCN / IID	68.91
FCN / ASM	**72.14**
DeepLab / IID	77.54
DeepLab / ASM	**78.05**
PSPNet / IID	80.12
PSPNet / cGAN	80.67
PSPNet / GAN	80.74
PSPNet / ASM	**81.43**

Table 2: Experimental results on PASVAL VOC with several base models, FCN [40], DeepLab [7], and PSPNet [60]. The improvements by replacing pixel-wise losses with ASM are consistent across different base models. cGAN [22] denotes the discriminator is conditioned on the input.

VOC dataset. We show some visual comparison in Fig. 8.

Boundary Localization Improvement. We argue that our proposed method is more sensitive to complex shapes of objects. We thus evaluate boundary localization using standard contour detection metrics [1]. The contour detection metrics compute the correspondences between prediction boundaries and ground-truth boundaries, and summarize the results with precision, recall, and f-measure. We compare the results with different loss functions: IID, GAN and ASM on VOC validation set. As shown in Figure 6, ASM outperforms both IID and GAN among most categories and overall in boundary precision. (Detailed numbers including recall and f-measure can be found in the supplementary.) The boundaries of thin-structured objects, such as 'bike' and 'chair', are much better captured by ASM.

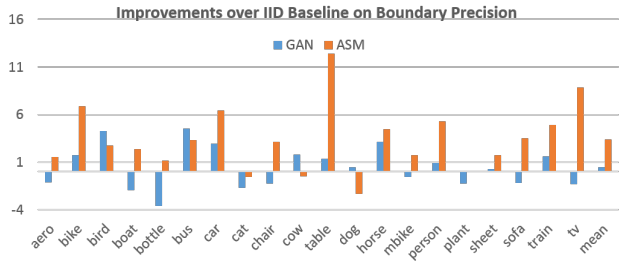

Figure 6: Improvements (%) of per-class boundary precision of PSPNet [60] on PASCAL VOC 2012 [14] validation set. ASM outperforms IID and GAN in most categories.

	rel	$\log_{10}$	rms	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
IID	0.267	0.393	1.174	0.466	0.790	0.922
cGAN	0.266	0.371	1.123	0.509	0.816	0.932
ASM	**0.252**	0.405	**1.079**	**0.540**	**0.834**	0.929

Table 3: Depth estimation measurements on 2D-3D-S [2]. Note that lower is better for the first three columns, and higher is better for the last three columns. IID is L2.

4.4. Main Results on Depth and Surface Normal

We compare ASM with pixel-wise L2 loss (IID) and cGAN [22] for monocular depth estimation. The performance is evaluated by the metrics proposed in [13], which are formulated as follows:

Abs Relative difference (rel) : $\frac{1}{|T|}\sum_{y\in T}|y-\hat{y}|/\hat{y}$

RMSE (rms) : $\sqrt{\frac{1}{|T|}\sum_{y\in T}\|y-\hat{y}\|^2}$

$\log_{10}$: $\sqrt{\frac{1}{|T|}\sum_{y\in T}\|\log y-\log\hat{y}\|^2}$

Threshold : % of y_i s.t $\max(\frac{y_i}{\hat{y}_i},\frac{\hat{y}_i}{y_i})=\delta<thr$

As summarized in Table 3, our proposed method consistently improves among most metrics, including 'rel', 'rms', and accuracy with different thresholds. Note that 'rel', '$\log_{10}$', and 'rms' metrics reflect the *mean* value of error distributions, which penalizes more on larger errors, which are usually incurred by mislabeling of the ground truth. On the other hand, accuracy with thresholds evaluates the *median* of errors, which reflects more precisely the visual quality of predictions. We notice that our method outperforms others even more significantly in *median* errors with smaller thresholds, which demonstrates the fine-grained structure discriminator power of ASM. The conclusion is consistent with our observations from visual comparisons in Fig. 8.

We compare ASM with pixel-wise normalized L2 loss (IID) and cGAN [22] for surface normal prediction. To evaluate the prediction quality, we follow [15] and report the results on several metrics, including the *mean* and *median* of error angles between the ground truth and prediction, and the percentage of error angles within $2.82°$, $5.63°$,

	Angle Distance		Within t° Deg.				
	Mean	Median	2.82°	5.63°	11.25°	22.5°	30°
IID	16.84	9.20	12.90	31.12	58.01	77.37	82.76
cGAN	17.12	9.04	12.28	31.52	58.52	77.00	82.32
ASM	16.98	**8.28**	**17.58**	**35.48**	**63.29**	**78.50**	82.47

Table 4: Surface normal estimation measurements on 2D-3D-S [2]. Note that lower is better for the first two columns, and higher is better for the last three columns. IID is normalized L2.

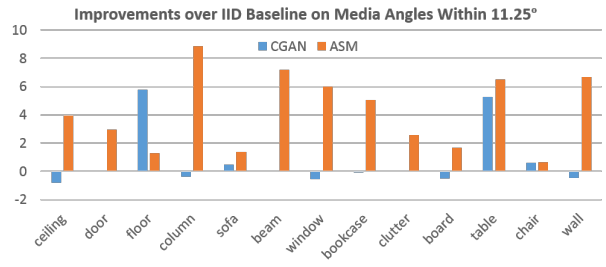

Figure 7: Improvements (%) of instance-average surface normal metric on 2D-3D-S [2]. The categories with most improvements are 'column', 'beam', 'table', and 'wall'.

11.25°, 22.5°, 30°. The results are presented in Table 4, and ASM improves metrics including *median* angles and percentage of angles within 2.82°, 5.63°, 11.25° by large margin. Similar to accuracy with thresholds for depth estimation, metrics with smaller angles are more consistent with visual quality. We conclude that ASM captures most details and outperforms baselines in most cases. We observe in Fig. 8 prominent visual improvements on thin structures, such as bookcase shelves. The surface normal prediction of larger objects, including wall and ceiling, are more uniform. Also, the contrast between adjacent surfaces are sharper with ASM.

Instance- and Semantic-level Analysis. Just as mIoU metric in semantic segmentation, the metrics proposed by [13] for depth and by [15] for surface normal are biased toward larger objects. We thus follow [24] to evaluate instance-wise metrics to attenuate the bias and fairly evaluate the performance on smaller objects. We collect semantic instance masks on 2D-3D-S dataset and formulate the instance-wise metric as $\frac{\sum_{i \in I_c} M_{i,c}}{|I_c|}$, where I_c denotes the set of instances in class c and $M_{i,c}$ is the metric for depth or surface normal of instance i in class c. (We do not use instance- or semantic-level information during training.) As shown in Fig 7, we demonstrate that ASM improves instance-wise percentage of angles within 11.25° consistently among all categories. The instance-wise analysis for depth estimation can be found in the supplementary.

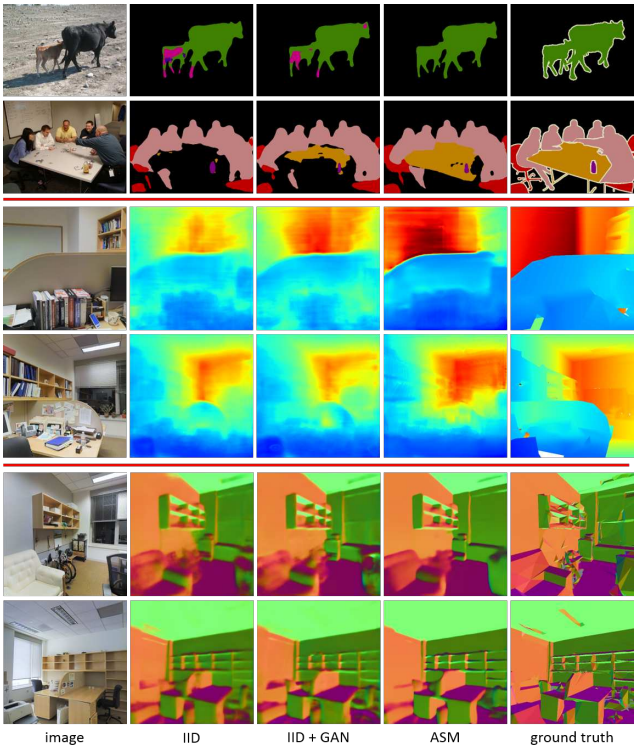

Figure 8: Visual quality comparison for semantic segmentation (top two rows) on VOC [14] validation set, monocular depth estimation (middle two rows) and surface normal prediction (bottom two rows) on [2]. Left to right: Images, predictions from training with IID, IID+(c)GAN, ASM, and ground truth masks.

5. Conclusion

We proposed a novel framework, Adversarial Structure Matching, for training structured prediction networks. We provided theoretical analyses and extensive experiments to demonstrate the efficacy of ASM. We concluded that multi-scale hard negative structural errors provide better supervision than the conventional pixel-wise IID losses (or incorporated with structure priors) in different structured prediction tasks, namely, semantic segmentation, monocular depth estimation, and surface normal prediction.

Acknowledgements. This research was supported, in part, by Berkeley Deep Drive, NSF (IIS-1651389), DARPA, and US Government fund through Etegent Technologies on Low-Shot Detection in Remote Sensing Imagery. The views, opinions and/or findings expressed should not be interpreted as representing the official views or policies of NSF, DARPA, or the U.S. Government.

References

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *TPAMI*, 2011. 7

[2] I. Armeni, A. Sax, A. R. Zamir, and S. Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017. 1, 2, 5, 7, 8

[3] G. Bertasius, J. Shi, and L. Torresani. Semantic segmentation with boundary neural fields. In *CVPR*, 2016. 2

[4] G. Bertasius, L. Torresani, S. X. Yu, and J. Shi. Convolutional random walk networks for semantic image segmentation. In *CVPR*, 2017. 2

[5] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *ECCV*, 2002. 2, 5

[6] L.-C. Chen, J. T. Barron, G. Papandreou, K. Murphy, and A. L. Yuille. Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform. In *CVPR*, 2016. 2

[7] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *arXiv preprint arXiv:1606.00915*, 2016. 1, 2, 5, 6, 7

[8] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2, 6

[9] L.-C. Chen, A. Schwing, A. Yuille, and R. Urtasun. Learning deep structured models. In *ICML*, 2015. 2

[10] X. Cheng, P. Wang, and R. Yang. Depth estimation via affinity learned with convolutional spatial propagation network. In *ECCV*, 2018. 2

[11] C. Couprie, C. Farabet, L. Najman, and Y. LeCun. Indoor semantic segmentation using depth information. *arXiv preprint arXiv:1301.3572*, 2013. 2

[12] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015. 2

[13] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NIPS*, 2014. 1, 2, 7, 8

[14] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 1, 2, 5, 7, 8

[15] D. F. Fouhey, A. Gupta, and M. Hebert. Data-driven 3d primitives for single image understanding. In *ICCV*, 2013. 2, 7, 8

[16] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015. 3

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014. 1, 3

[18] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik. Learning rich features from rgb-d images for object detection and segmentation. In *ECCV*, 2014. 2

[19] M. Gygli, M. Norouzi, and A. Angelova. Deep value networks learn to evaluate and iteratively refine structured outputs. In *ICML*, 2017. 1, 2

[20] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 5

[21] J.-J. Hwang and T.-L. Liu. Pixel-wise deep learning for contour detection. In *ICLR Workshop*, 2015. 1

[22] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017. 1, 7

[23] K. Karsch, C. Liu, and S. B. Kang. Depth transfer: Depth extraction from video using non-parametric sampling. *TPAMI*, 2014. 2

[24] T.-W. Ke, J.-J. Hwang, Z. Liu, and S. X. Yu. Adaptive affinity field for semantic segmentation. 2018. 1, 2, 8

[25] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NIPS*, 2017. 2

[26] S. Kim, K. Park, K. Sohn, and S. Lin. Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields. In *ECCV*, 2016. 2

[27] I. Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *CVPR*, 2017. 2

[28] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*, 2011. 1, 2

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1

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

[31] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016. 1, 2

[32] B. Li, C. Shen, Y. Dai, A. Van Den Hengel, and M. He. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. In *CVPR*, 2015. 2

[33] J. Li, R. Klein, and A. Yao. A two-streamed network for estimating fine-scaled depth maps from single rgb images. In *ICCV*, 2017. 2

[34] X. Li, Z. Liu, P. Luo, C. C. Loy, and X. Tang. Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade. In *CVPR*, 2017. 2

[35] G. Lin, C. Shen, A. van den Hengel, and I. Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *CVPR*, 2016. 2

[36] F. Liu, C. Shen, and G. Lin. Deep convolutional neural fields for depth estimation from a single image. In *CVPR*, 2015. 2

[37] F. Liu, C. Shen, G. Lin, and I. D. Reid. Learning depth from single monocular images using deep convolutional neural fields. *TPAMI*, 2016. 2

[38] S. Liu, S. De Mello, J. Gu, G. Zhong, M.-H. Yang, and J. Kautz. Learning affinity via spatial propagation networks. In *NIPS*, 2017. 2

[39] Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang. Semantic image segmentation via deep parsing network. In *CVPR*, 2015. 1, 2

[40] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1, 2, 5, 6, 7

[41] P. Luc, C. Couprie, S. Chintala, and J. Verbeek. Semantic segmentation using adversarial networks. *NIPS Workshop*, 2016. 1

[42] M. Maire, T. Narihira, and S. X. Yu. Affinity cnn: Learning pixel-centric pairwise relations for figure/ground embedding. In *CVPR*, 2016. 2

[43] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert. Cross-stitch networks for multi-task learning. In *CVPR*, 2016. 2

[44] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *CVPR*, 2015. 2

[45] X. Ren, L. Bo, and D. Fox. Rgb-(d) scene labeling: Features and algorithms. In *CVPR*, 2012. 2

[46] S. R. Richter, Z. Hayder, and V. Koltun. Playing for benchmarks. In *ICCV*, volume 2, 2017. 2

[47] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 2, 5

[48] A. Roy and S. Todorovic. Monocular depth estimation using neural regression forest. In *CVPR*, 2016. 2

[49] J. Shi and J. Malik. Normalized cuts and image segmentation. *TPAMI*, 2000. 2

[50] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 2

[51] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. L. Yuille. Towards unified depth and semantic prediction from a single image. In *CVPR*, 2015. 2

[52] P. Wang, X. Shen, B. Russell, S. Cohen, B. Price, and A. L. Yuille. Surge: Surface regularized geometry estimation from a single image. In *NIPS*, 2016. 2

[53] S. Xie, X. Huang, and Z. Tu. Top-down learning for structured labeling with convolutional pseudoprior. In *ECCV*, 2016. 1, 2

[54] D. Xu, W. Ouyang, X. Wang, and N. Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. *arXiv preprint arXiv:1805.04409*, 2018. 2

[55] D. Xu, E. Ricci, W. Ouyang, X. Wang, and N. Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *CVPR*, volume 1, 2017. 2

[56] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 2

[57] A. R. Zamir, A. Sax, W. Shen, L. Guibas, J. Malik, and S. Savarese. Taskonomy: Disentangling task transfer learning. In *CVPR*, 2018. 2

[58] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 3

[59] Z. Zhang, Z. Cui, C. Xu, Z. Jie, X. Li, and J. Yang. Joint task-recursive learning for semantic segmentation and depth estimation. In *ECCV*, 2018. 2

[60] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *CVPR*, 2017. 2, 5, 7

[61] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr. Conditional random fields as recurrent neural networks. In *ICCV*, 2015. 1, 2

[62] B. Zhou, D. Bau, A. Oliva, and A. Torralba. Interpreting deep visual representations via network dissection. *PAMI*, 2018. 6

[63] W. Zhuo, M. Salzmann, X. He, and M. Liu. Indoor scene structure analysis for single image depth estimation. In *CVPR*, 2015. 2