

Probabilistic Graph Attention Network with Conditional Kernels for Pixel-Wise Prediction

Dan Xu, Xavier Alameda-Pineda, *Senior Member, IEEE*, Wanli Ouyang, *Senior Member, IEEE*, Elisa Ricci, *Member, IEEE*, Xiaogang Wang, *Senior Member, IEEE*, Nicu Sebe, *Senior Member, IEEE*

Abstract—Multi-scale representations deeply learned via convolutional neural networks have shown tremendous importance for various pixel-level prediction problems. In this paper we present a novel approach that advances the state of the art on pixel-level prediction in a fundamental aspect, *i.e.* structured multi-scale features learning and fusion. In contrast to previous works directly considering multi-scale feature maps obtained from the inner layers of a primary CNN architecture, and simply fusing the features with weighted averaging or concatenation, we propose a probabilistic graph attention network structure based on a novel Attention-Gated Conditional Random Fields (AG-CRFs) model for learning and fusing multi-scale representations in a principled manner. In order to further improve the learning capacity of the network structure, we propose to exploit feature dependant conditional kernels within the deep probabilistic framework. Extensive experiments are conducted on four publicly available datasets (*i.e.* BSDS500, NYUD-V2, KITTI and Pascal-Context) and on three challenging pixel-wise prediction problems involving both discrete and continuous labels (*i.e.* monocular depth estimation, object contour prediction and semantic segmentation). Quantitative and qualitative results demonstrate the effectiveness of the proposed latent AG-CRF model and the overall probabilistic graph attention network with feature conditional kernels for structured feature learning and pixel-wise prediction.

Index Terms—Structured representation learning, attention model, conditional random fields, conditional kernels, pixel-wise prediction

1 INTRODUCTION

THE capability to effectively exploit multi-scale feature representations is considered a crucial factor for achieving accurate predictions for the pixel-level prediction in both traditional [1] and CNN-based [2], [3], [4], [5] approaches. Restricting the attention on deep learning-based solutions, existing methods [2], [4] typically derive multi-scale representations by adopting standard CNN architectures and directly considering the feature maps associated to different inner semantic layers. These maps are highly complementary: while the features from the shallow layers are responsible for predicting low-level details, the ones from the deeper layers are devoted to encode the high-level semantic structure of the objects. Traditionally, concatenation and weighted average are very popular strategies to combine multi-scale representations (see Figure 2.a). While these strategies generally lead to an increased prediction accuracy with a comparison to single-scale models, they severely simplify the complex structured relationship between multi-scale feature maps. The motivational cornerstone of this study is the following research question: is it worth modelling and exploiting complex relationships between multiple scales of deep representations for pixel-wise prediction?

Inspired by the success of recent works employing

- Dan Xu is with the Department of Engineering, University of Oxford. E-mail: {dan.xu}@robots.ox.ac.uk.
- Xavier Alameda-Pineda is with the Perception Group, INRIA. E-mail: xavier.alameda-pineda@inria.fr.
- Wanli Ouyang is with the Department of Electrical and Information Engineering, the University of Sydney. E-mail: wanli.ouyang@sydney.edu.au.
- Xiaogang Wang is with the Department of Electronic Engineering, the Chinese University of Hong Kong. E-mail: xgwang@ee.cuhk.edu.hk
- Elisa Ricci and Nicu Sebe are with the Department of Information Engineering and Computer Science, University of Trento, Italy. E-mail: {elisa.ricci, niculae.sebe}@unitn.it.

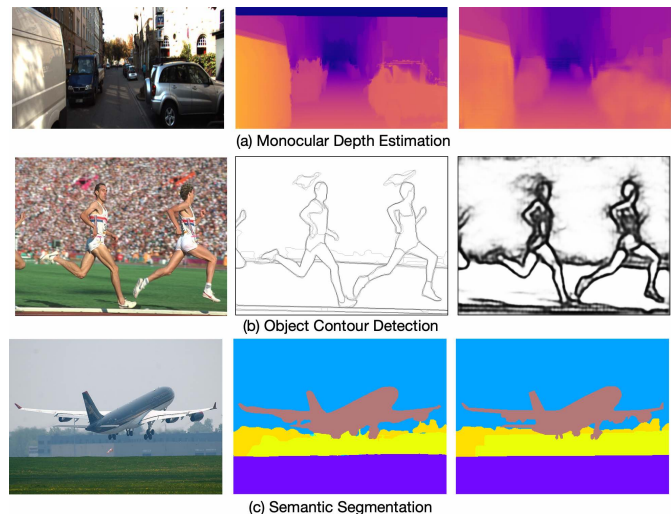


Fig. 1. The proposed model targets multi-scale structured deep representation learning and could be applied into different pixel-wise prediction problems involving both discrete and continuous prediction variables, *i.e.* (a) monocular depth estimation on KITTI, (b) object contour detection on BSDS500, and (c) semantic segmentation on Pascal-Context. The first, second and third columns are input RGB images, ground-truth and predicted results, respectively.

graphical models within deep CNN architectures [6], [7], [8] for structured prediction, we propose a probabilistic graph attention network structure base on a novel Attention-Gated Conditional Random Fields (AG-CRFs), which allows to learn effective feature representations at each scale by exploiting the information available from other scales. This is achieved by incorporating an attention mechanism [9]

seamlessly integrated into the multi-scale learning process under the form of gates [10]. Intuitively, the attention mechanism will further enhance the learning of the multi-scale representation fusion by controlling the information flow (*i.e.* messages) among the feature maps, thus improving the overall performance of the model. To further increase the learning capacity of the model, we introduce feature dependant conditional kernels for predicting the attentions and the messages, enabling them conditioned on related feature context while not shared by all the feature inputs. In contrast to previous works [6], [7], [8] aiming at structured modelling on the prediction level, our model focuses on the feature level, which leads to a much higher flexibility when applied to different pixel-wise prediction problems involving both continuous and discrete prediction variables.

We implement the proposed AG-CRF as a neural network module, and integrate it into a hierarchical multi-scale CNN framework, defining a novel probabilistic graph attention network structure, termed as PGA-Net for pixel-wise prediction. The hierarchical network is able to learn richer multi-scale features than conventional CNNs, the representational power of which is further enhanced by the proposed conditional kernel AG-CRF model. We extensively evaluate the effectiveness of the proposed model on three different continuous and discrete pixel-wise prediction tasks (see Figure 1), *i.e.* object contour prediction, monocular depth estimation and semantic segmentation, and on multiple challenging benchmarks (BSDS500 [11], NYUD-V2 [12], KITTI [13] and Pascal-Context [14]). The results demonstrate that our approach is able to learn rich and effective deep structured representations, thus showing very competitive performance to state-of-the-art methods on these tasks.

This paper extends our earlier work [15] through proposing a new feature dependant conditional kernel strategy, further re-elaborating the related works, providing more methodological details, and significantly expanding the experiments and analysis by demonstrating the effectiveness on another two popular pixel-wise prediction tasks (monocular depth estimation and semantic segmentation). Multi-scale deep features are widely demonstrated very effective. (*e.g.* [2], [16]). The importance of our work is in joint probabilistic modelling of the relationship among multi-scale deep features using the conditional random fields and the attention mechanism. To summarize, the contribution of this paper is threefold:

- First, we propose a structured probabilistic graph network for effectively learning and fusing multi-scale deep representations. We learn the multi-scale features using a probabilistic graphical attention model, which is a principled way of modelling the statistical relationship among multi-scale features.
- Second, we design an attention guided CRF graph model which models the attention as gating for controlling the message passing among features of different scales. As the passed message among the feature maps are not always useful, the attention mechanism is especially introduced to control the message passing flow among feature maps of different scales. The attention is incorporated in the probabilistic graphical model. We also introduce a conditional kernel strategy for feature dependant attention and message learning.

- Third, extensive experiments are conducted on three distinct pixel-wise prediction tasks and on four different challenging datasets, demonstrating that the proposed model and framework significantly outperform previous methods integrating multi-scale information from different semantic network layers, and show very competitive performance on all the tasks compared with the state-of-the-art methods. The proposed model is generic in multi-scale feature learning and can be flexibly employed in other continuous and discrete prediction problems.

The remainder of this paper is organized as follows. Sec. 2 introduces the related works, and then we illustrate the proposed approach in Sec. 3, and in Sec. 4 we present the details of the model implementation in deep networks. The experimental evaluation and analysis are elaborated in Sec. 5. We finally conclude the paper in Sec. 6.

2 RELATED WORK

2.1 Pixel-wise Prediction

We review previous works with deep learning networks on three important pixel-wise prediction tasks, *i.e.* contour detection, monocular depth estimation and semantic segmentation, on which we extensively demonstrate the effectiveness of the proposed approach.

Contour Detection. In the last few years several deep learning models have been proposed for detecting contours [2], [4], [17], [18], [19], [20] or crisp boundaries [21]. Among these, some works explicitly focused on devising multi-scale CNN models in order to boost performance. For instance, the Holistically-Nested Edge Detection method [2] employed multiple side outputs derived from the inner layers of a primary CNN and combine them for the final prediction. Liu *et al.* [20] introduced a framework to learn rich deep representations by concatenating features derived from all convolutional layers of VGG16. Bertasius *et al.* [18] considered skip-layer CNNs to jointly combine feature maps from multiple layers. Maninis *et al.* [4] proposed Convolutional Oriented Boundaries (COB), where features from different layers are fused to compute oriented contours and region hierarchies. However, these works combine the multi-scale representations from different layers adopting concatenation and weighted averaging schemes while not considering the dependency between the features.

Monocular Depth Estimation. There are existing recent works on monocular depth estimation based on deep CNNs [6], [22], [23], [24], [25], [26], [27], [28]. Among them, Eigen *et al.* [29] proposed a multi-scale network architecture for the task via considering two cascaded networks, performing a coarse to fine refinement of the depth prediction. They also further extend this framework to deal with multiple pixel-level predictions, such as surface normal estimation and semantic segmentation. Fu *et al.* [26] presented a novel DORN method to cast the monocular depth estimation as a deep ordinal regression problem. Lee *et al.* [28] designed a network module which utilizes the multi-scale local planar as guidance to learn more effective structure features for depth estimation. Wang *et al.* [23] introduced a CNN-CRF framework for joint depth estimation and semantic segmentation. The most related work to ours is [6], which introduced a continuous CRF

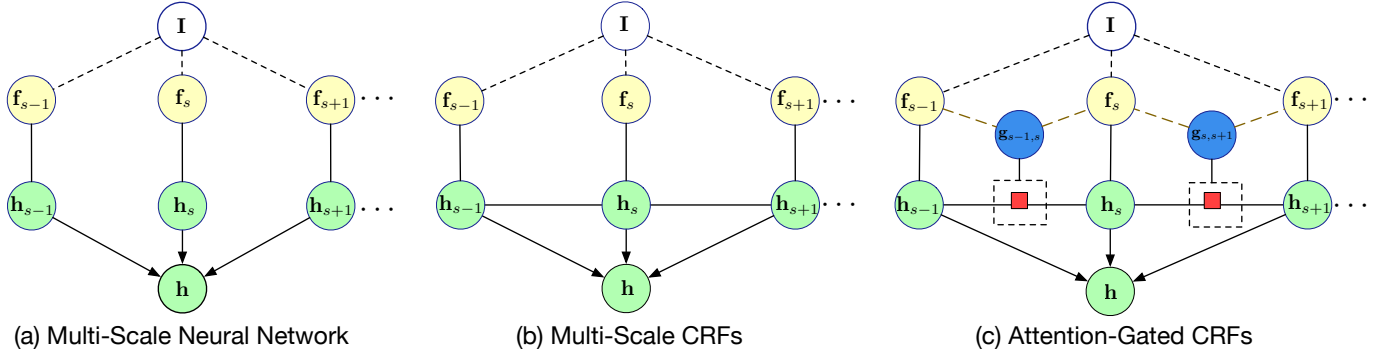


Fig. 2. An illustration of different schemes for multi-scale deep feature learning and fusion. (a) the traditional approach (e.g. concatenation, weighted averaging), (b) the proposed CRF implementing multi-scale feature fusion (c) the proposed Attention-Gated-CRF based approach.

model for end-to-end learning depth regression with a front-end CNN. Xu *et al.* [8] improved [6] by presenting a multi-scale continuous CRF model to learn the multi-scale predictions and fusion. However, these approaches purely focus on modelling the structure of the predictions, therefore answering to specific problems, and leading to task-dependent models and associated architectures. Differently, our work is focusing on statistical modeling on the structured features, thus being more flexible to be applied to different continuous or discrete prediction tasks.

Semantic Segmentation. As an important task in scene understanding, semantic segmentation has received wide attention in recent years. Long *et al.* [30] proposed a fully convolutional network for the task which significantly improved the performance and reduced the network parameters. The dilated convolution [5], [31] was devised in order to obtain bigger receptive field of the features, further boosting the segmentation performance. OCNNet [32] introduced an object-context pooling strategy based on affinity learning among pixels to capture a global context for feature refinement. Other main-stream directions mainly explored multi-scale feature learning and model ensembling [16], [33], [34], designing convolutional encoder-decoder network structures [35], [36] and performing end-to-end structure prediction with CRF models [37], [38], [39]. A more close work to ours in the literature is the GloRe approach [40] which utilizes a graph convolution model for learning generic representation for semantic segmentation. However, its modelling is only for single-scale and not in a probabilistic graph formulation.

There are also some existing works which explored joint deep learning of more than one pixel-wise prediction tasks [41], [42], [43], [44]. Specifically, Xu *et al.* [41] designed a PAD-Net architecture which learns multiple auxiliary pixel-wise tasks and presents a multi-task distillation module to combine the predictions from the different auxiliary tasks to help more important final tasks. Vandenhende and Van Gool *et al.* [44] further improved the PAD-Net by introducing a method for feature propagation of different multi-task representations. However, these works are focusing more on empirical design for learning interaction between different pixel-wise tasks, while our model targets at statistical probabilistic modeling for structured multi-scale feature fusion which could provide a theoretic explanation and thus it is beneficial for a more effective deep module design.

2.2 Deep Multi-scale Learning

The importance of combining multi-scale information has been widely revealed in various computer vision tasks [2], [45], [46]. For instance, Xie *et al.* [2] proposed a fully convolutional neural network with deep supervision for edge detection, which employs a weighted averaging strategy for the combination of multi-scale side outputs. The skipping-layer networks are also very popular for learning multi-scale representations, where the features obtained from different semantic layers of a backbone CNN are combined in an output layer to produce more robust representations. Sun and Wang *et al.* [47], [48] proposed a HRNet architecture which aims to enhance high-resolution representations via aggregating the representations with multi-scale resolutions from different network stages. To aggregate multi-scale contexts of features, the *dilation* or *à trous* convolution [49] structures are devised, which could be applied in embedded in different convolutional layers in a deep network to obtain a larger receptive field. Yang *et al.* [50] introduced DAG-CNNs to combine multi-scale features produced from different ReLU layers using element-wise addition operation. Huang *et al.* [51] recently proposed a multi-scale network architecture using densely skipping connections to pass feature flow at different scales. However, in these works, the multi-scale representations or predictions are typically combined via using simple concatenation or weighted averaging operation. We are also not aware of previous works exploring multi-scale representation learning and fusion within a probabilistic CRF graph framework. Besides, we also involve learning an attention mechanism as gating in the graph model for controlling the message passing of the feature variables.

2.3 Attention Models

Attention models [52], [53] have been successfully exploited in deep learning for various tasks such as image classification [54], speech recognition [55], image caption generation [56] and language translation [57]. Fu *et al.* [58] recently proposed a dual attention model considering a combination of both the spatial- and the channel-wise attentions for semantic segmentation. However, to our knowledge, this work is the first to introduce a structured attention model for for both discrete and continuous prediction tasks. Furthermore, we are not aware of previous studies integrating the attention mechanism into a probabilistic (CRF) framework to control the message passing between hidden variables. We

model the attention as *gates* [10], which have been used in previous deep models such as restricted Boltzman machine for unsupervised feature learning [59], LSTM for sequence learning [60], [61] and CNN for image classification [62]. However, none of these works explore the possibility of jointly learning multi-scale deep representations and an attention model within a unified probabilistic graphical model.

2.4 Structured Learning based on CRFs

The conditional random fields (CRFs) were widely used for probabilistic structured modeling in the non-deep-learning era for a wide range of problems, such as object recognition [63], information extraction [64] and pixel-wise semantic labeling [65], [66], [67], [68]. Specifically, Boykov *et al.* [68] proposed a combinational graph cut algorithm integrating cues of boundaries, regions and shapes for semantic segmentation. Krähenbühl and Koltun [67] designed a fully-connected CRF model with gaussian pair-wise potentials and accordingly proposed an efficient approximation inference solution for the model. With the rapid progress of the deep learning techniques, the CRFs are also utilized together with Convolutional Neural Network (CNN) architectures for learning structured deep predictions [6], [38], [39], [46]. Among them, Zheng *et al.* [39] first implemented the CRF inference as Recurrent Neural Networks for end-to-end learning with any backbone CNN. There are also existing works exploring learning structured deep features with CRFs, in order to more flexibly adapt to different continuous and discrete tasks, such as human pose estimation [7] and monocular depth estimation [69]. Great success has been made by these existing models; however, none of them considered simultaneously learning structured multi-scale representations and structured attention in a joint probabilistic graph formulation for pixel-wise prediction.

3 THE PROPOSED APPROACH

3.1 Problem Definition and Notation

Given an input image $\mathbf{I}$ and a generic front-end CNN model with parameters $\mathbf{W}_c$, we consider a set of S multi-scale feature maps $\mathbf{F} = \{\mathbf{f}_s\}_{s=1}^S$. Being a generic framework, these feature maps can be the output of S intermediate CNN layers or of another representation, thus s is a *virtual* scale. The feature map at scale s , $\mathbf{f}_s$ can be interpreted as a set of feature vectors, $\mathbf{f}_s = \{\mathbf{f}_s^i\}_{i=1}^N$, where N is the number of pixels. Opposite to previous works adopting simple concatenation or weighted averaging schemes [2], [16], we propose to combine the multi-scale feature maps by learning a set of latent feature maps $\mathbf{h}_s = \{\mathbf{h}_s^i\}_{i=1}^N$ with a novel *Attention-Gated CRF* model sketched in Figure 2. Intuitively, this allows a joint refinement of the features by flowing information between different scales. Moreover, since the information from one scale may or may not be relevant for the pixels at another scale, we utilise the concept of *gate*, previously introduced in the literature in the case of graphical models [70], in our CRF formulation. These gates are binary random hidden variables that permit or block the flow of information between scales at every pixel. Formally, $g_{s_e, s_r}^i \in \{0, 1\}$ is the gate at pixel i of scale s_r (receiver) from

scale s_e (emitter), and we also write $\mathbf{g}_{s_e, s_r} = \{g_{s_e, s_r}^i\}_{i=1}^N$. Precisely, when $g_{s_e, s_r}^i = 1$ then the hidden variable $\mathbf{h}_{s_r}^i$ is updated taking (also) into account the information from the s_e -th layer, *i.e.* $\mathbf{h}_{s_e}$. As shown in the following, the joint inference of the hidden features and the gates leads to estimating the optimal features as well as the corresponding attention model, hence the name *Attention-Gated CRFs*.

3.2 Attention-Gated CRFs

Given the observed multi-scale feature maps $\mathbf{F}$ of image $\mathbf{I}$, the objective is to estimate the hidden multi-scale representation $\mathbf{H} = \{\mathbf{h}_s\}_{s=1}^S$ and, accessorially the attention gate variables $\mathbf{G} = \{\mathbf{g}_{s_e, s_r}\}_{s_e, s_r=1}^S$. To do that, we formalize the problem within a conditional random field framework and write the Gibbs distribution as

$$P(\mathbf{H}, \mathbf{G} | \mathbf{I}, \Theta) = \exp(-E(\mathbf{H}, \mathbf{G}, \mathbf{I}, \Theta)) / Z(\mathbf{I}, \Theta), \quad (1)$$

where Θ is the set of parameters and E is the energy function. As usual, we exploit both unary and binary potentials to couple the hidden variables between them and to the observations. Importantly, the proposed binary potential is gated, and thus only active when the gate is open. More formally the general form¹ of the energy function writes:

$$E(\mathbf{H}, \mathbf{G}, \mathbf{I}, \Theta) = \underbrace{\sum_s \sum_i \phi_h(\mathbf{h}_s^i, \mathbf{f}_s^i)}_{\text{Unary potential}} + \underbrace{\sum_{s_e, s_r} \sum_{i, j} g_{s_e, s_r}^i \psi_h(\mathbf{h}_{s_r}^i, \mathbf{h}_{s_e}^j, \mathbf{K}_{s_e, s_r}^{i, j})}_{\text{Gated pairwise potential}} \quad (2)$$

The first term of the energy function is a classical unary term that relates the hidden features to the observed multi-scale CNN representations. The second term synthesizes the theoretical contribution of the present study because it conditions the effect of the pair-wise potential $\psi_h(\mathbf{h}_{s_e}^i, \mathbf{h}_{s_r}^j)$ upon the gate hidden variable g_{s_e, s_r}^i . Figure 2c depicts the model formulated in Equ.(2). If we remove the attention gate variables, it becomes a general multi-scale CRFs as shown in Figure 2b.

Given that formulation, and as it is typically the case in conditional random fields, we exploit the mean-field approximation in order to derive a tractable inference procedure. Under this generic form, the mean-field inference procedure writes:

$$q(\mathbf{h}_s^i) \propto \exp \left(\phi_h(\mathbf{h}_s^i, \mathbf{f}_s^i) + \sum_{s' \neq s} \sum_j \mathbb{E}_{q(g_{s', s}^i)} \{g_{s', s}^i\} \mathbb{E}_{q(\mathbf{h}_{s'}^j)} \{ \psi_h(\mathbf{h}_s^i, \mathbf{h}_{s'}^j) \} \right), \quad (3)$$

$$q(g_{s', s}^i) \propto \exp \left(g_{s', s}^i \mathbb{E}_{q(\mathbf{h}_s^i)} \left\{ \sum_j \mathbb{E}_{q(\mathbf{h}_{s'}^j)} \{ \psi_h(\mathbf{h}_s^i, \mathbf{h}_{s'}^j) \} \right\} \right), \quad (4)$$

where $\mathbb{E}_q$ stands for the expectation with respect to the distribution q .

Before deriving these formulae for our precise choice of potentials, we remark that, since the gate is a binary variable,

1. One could certainly include a unary potential for the gate variables as well. However this would imply that there is a way to set/learn the a priori distribution of opening/closing a gate. In practice we did not observe any notable difference between using or skipping the unary potential on g .

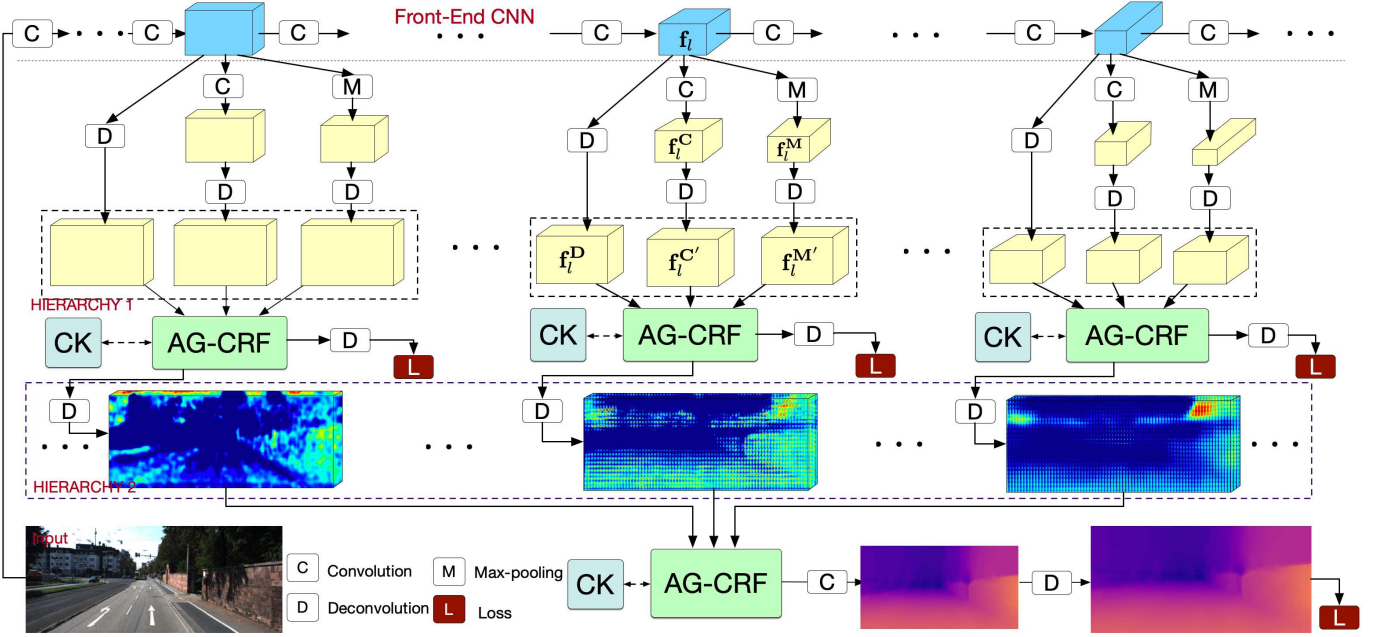


Fig. 3. An overview of the proposed Probabilistic Graph Attention Network (PGA-Net) for monocular depth detection. The symbols C, D, M and L denote the convolution, the deconvolution, the max-pooling operation and optimization loss, respectively. AG-CRF represents the proposed attention-gated CRF model with conditional kernels (CK) for structured multi-scale feature learning, which is fully differentiable and supports end-to-end training with a multi-scale CNN network. PGA-Net consists of two hierarchies. The hierarchy 1 generates rich multi-scale features which are refined by AG-CRFs, and then are passed to hierarchy 2 for final prediction.

the expectation of its value is the same as $q(g_{s',s}^i = 1)$. By defining: $\mathcal{M}_{s',s}^i = \mathbb{E}_{q(\mathbf{h}_s^i)} \left\{ \sum_j \mathbb{E}_{q(\mathbf{h}_{s'}^j)} \left\{ \psi_h(\mathbf{h}_s^i, \mathbf{h}_{s'}^j) \right\} \right\}$, the expected value of the gate writes:

$$\alpha_{s,s'}^i = \mathbb{E}_{q(\mathbf{h}_{s'}^i)} \{g_{s',s}^i\} = \frac{q(g_{s',s}^i = 1)}{q(g_{s',s}^i = 0) + q(g_{s',s}^i = 1)} \quad (5)$$

$$= \sigma(-\mathcal{M}_{s',s}^i),$$

where $\sigma()$ denotes the sigmoid function. This finding is specially relevant in the framework of CNN since many of the attention models are typically obtained after applying the sigmoid function to the features derived from a feed-forward network. Importantly, since the quantity $\mathcal{M}_{s',s}^i$ depends on the expected values of the hidden features $\mathbf{h}_s^i$, the AG-CRF framework extends the unidirectional connection from the features to the attention model, to a bidirectional connection in which the expected value of the gate allows to refine the distribution of the hidden features as well.

3.3 AG-CRF Inference

In order to construct an operative model we need to define the unary and gated potentials ϕ_h and ψ_h . In our case, the unary potential corresponds to an isotropic Gaussian:

$$\phi_h(\mathbf{h}_s^i, \mathbf{f}_s^i) = -\frac{a_s^i}{2} \|\mathbf{h}_s^i - \mathbf{f}_s^i\|^2, \quad (6)$$

where $a_s^i > 0$ is a weighting factor.

The gated binary potential is specifically designed for a two-fold objective. On the one hand, we would like to learn and further exploit the relationships between hidden vectors at the same, as well as at different scales. On the other hand, we would like to exploit previous knowledge on attention

models and include linear terms in the potential. Indeed, this would implicitly shape the gate variable to include a linear operator on the features. Therefore, we chose a bilinear potential:

$$\psi_h(\mathbf{h}_s^i, \mathbf{h}_{s'}^j) = \tilde{\mathbf{h}}_s^i \mathbf{K}_{s,s'}^{i,j} \tilde{\mathbf{h}}_{s'}^j, \quad (7)$$

where $\tilde{\mathbf{h}}_s^i = (\mathbf{h}_s^{i\top}, 1)^\top$ and $\mathbf{K}_{s,s'}^{i,j} \in \mathbb{R}^{(C_s+1) \times (C_{s'}+1)}$ being C_s the size, i.e. the number of channels, of the representation at scale s . If we write this matrix as $\mathbf{K}_{s,s'}^{i,j} = (\mathbf{L}_{s,s'}^{i,j}, \mathbf{l}_{s,s'}^{i,j}; \mathbf{l}_{s',s}^{j,i}, \mathbf{l}_{s',s}^{j,i\top}, 1)$, then $\mathbf{L}_{s,s'}^{i,j}$ exploits the relationships between hidden variables, while $\mathbf{l}_{s,s'}^{i,j}$ and $\mathbf{l}_{s',s}^{j,i}$ implement the classically used linear relationships of the attention models. In order words, ψ_h models the pair-wise relationships between features with the upper-left block of the matrix. Furthermore, ψ_h takes into account the linear relationships by completing the hidden vectors with the unity. In all, the energy function writes:

$$E(\mathbf{H}, \mathbf{G}, \mathbf{I}, \Theta) = -\sum_s \sum_i \frac{a_s^i}{2} \|\mathbf{h}_s^i - \mathbf{f}_s^i\|^2 + \sum_{s_e, s_r} \sum_{i,j} g_{s_e, s_r}^i \tilde{\mathbf{h}}_{s_r}^i \mathbf{K}_{s_r, s_e}^{i,j} \tilde{\mathbf{h}}_{s_e}^j. \quad (8)$$

Under these potentials, we can consequently update the mean-field inference equations to:

$$q(\mathbf{h}_s^i) \propto \exp \left(-\frac{a_s^i}{2} (\|\mathbf{h}_s^i\|^2 - 2\mathbf{h}_s^{i\top} \mathbf{f}_s^i) + \sum_{s' \neq s} \alpha_{s,s'}^i \mathbf{h}_s^{i\top} \sum_j (\mathbf{L}_{s,s'}^{i,j} \bar{\mathbf{h}}_{s'}^j + \mathbf{l}_{s,s'}^{i,j}) \right), \quad (9)$$

where $\bar{\mathbf{h}}_{s'}^j$ is the expected a posteriori value of $\mathbf{h}_{s'}^j$.

The previous expression implies that the a posteriori distribution for $\mathbf{h}_s^i$ is a Gaussian. The mean vector of the Gaussian and the function $\mathcal{M}$ write:

$$\begin{aligned}\bar{\mathbf{h}}_s^i &= \frac{1}{a_s^i} \left(a_s^i \mathbf{f}_s^i + \sum_{s' \neq s} \alpha_{s,s'}^i \sum_j (\mathbf{L}_{s,s'}^{i,j} \bar{\mathbf{h}}_{s'}^j + \mathbf{l}_{s,s'}^{i,j}) \right) \\ \mathcal{M}_{s',s}^i &= \sum_j \left(\bar{\mathbf{h}}_s^i \mathbf{L}_{s,s'}^{i,j} \bar{\mathbf{h}}_{s'}^j + \bar{\mathbf{h}}_s^{i\top} \mathbf{l}_{s,s'}^{i,j} + \bar{\mathbf{h}}_{s'}^{j\top} \mathbf{l}_{s',s}^{j,i} \right)\end{aligned}\quad (10)$$

which concludes the inference procedure. Furthermore, the proposed framework can be simplified to obtain the traditional attention models. In most of the previous studies, the attention variables are computed directly from the multi-scale features instead of computing them from the hidden variables. Indeed, since many of these studies do not propose a probabilistic formulation, there are no hidden variables and the attention is computed sequentially through the scales. We can emulate the same behavior within the AG-CRF framework by modifying the gated potential as follows:

$$\tilde{\psi}_h(\mathbf{h}_s^i, \mathbf{h}_{s'}^j, \mathbf{f}_s^i, \mathbf{f}_{s'}^j) = \mathbf{h}_s^i \mathbf{L}_{s,s'}^{i,j} \mathbf{h}_{s'}^j + \mathbf{f}_s^{i\top} \mathbf{l}_{s,s'}^{i,j} + \mathbf{f}_{s'}^{j\top} \mathbf{l}_{s',s}^{j,i}. \quad (11)$$

This means that we keep the pair-wise relationships between hidden variables (as in any CRF) and let the attention model be generated by a linear combination of the observed features from the CNN, as it is traditionally done. The changes in the inference procedure are straightforward. We refer to this model as partially-latent AG-CRFs (PLAG-CRFs), whereas the more general one is denoted as fully-latent AG-CRFs (FLAG-CRFs). The potential defined for the PLAG-CRF model, *i.e.* (11), has an impact on the inference of both the hidden features and the attention gates. Indeed, since the linear term does not depend on the hidden features, but on the observations, the mean of the hidden features is computed independently of this linear term:

$$\bar{\mathbf{h}}_s^i = \frac{1}{a_s^i} \left(a_s^i \mathbf{f}_s^i + \sum_{s' \neq s} \alpha_{s,s'}^i \sum_j \mathbf{L}_{s,s'}^{i,j} \bar{\mathbf{h}}_{s'}^j \right). \quad (12)$$

Likewise, the linear terms of the attention gate do not depend anymore on the hidden features but on the observations from the CNN:

$$\mathcal{M}_{s',s}^i = \sum_j \left(\bar{\mathbf{h}}_s^i \mathbf{L}_{s,s'}^{i,j} \bar{\mathbf{h}}_{s'}^j + \bar{\mathbf{f}}_s^{i\top} \mathbf{l}_{s,s'}^{i,j} + \bar{\mathbf{f}}_{s'}^{j\top} \mathbf{l}_{s',s}^{j,i} \right). \quad (13)$$

These are the two differences with respect to the inference of FLAG-CRFs. We will also introduce the implementation difference of both versions in Sec. 4.

3.4 Feature Dependant Conditional Kernels

In the model inference as shown in (10), (12) and (13), the kernels $\mathbf{L}_{s,s'}^{i,j}$, $\mathbf{l}_{s,s'}^{i,j}$ and $\mathbf{l}_{s',s}^{j,i}$ are shared for all the input features. This property restricts the learning capacity of the model: one would like those kernels to be dependant on the features so as to capture the feature correlated context, which is particularly important for pixel-wise prediction tasks. We hence propose to learn feature conditioned kernels, instead of the previous shared kernels. In practice, each kernel is predicted from the input features using a linear transformation as follows:

$$\begin{aligned}\mathbf{L}_{s,s'}^{i,j} &= \mathbf{W}_{L,s,s'}^{i,j} \text{concat}(\bar{\mathbf{h}}_s^i, \bar{\mathbf{h}}_{s'}^j) + \mathbf{b}_{L,s,s'}^{i,j}, \\ \mathbf{l}_{s,s'}^{i,j} &= \mathbf{W}_{l,s,s'}^{i,j} \bar{\mathbf{h}}_s^i + \mathbf{b}_{l,s,s'}^{i,j}, \quad \mathbf{l}_{s',s}^{j,i} = \mathbf{W}_{l,s',s}^{i,j} \bar{\mathbf{h}}_{s'}^j + \mathbf{b}_{l,s',s}^{i,j},\end{aligned}\quad (14)$$

where $\text{concat}(\cdot)$ denotes a concatenation operation function. The symbols $\{\mathbf{W}_{L,s,s'}^{i,j}, \mathbf{b}_{L,s,s'}^{i,j}\}$, $\{\mathbf{W}_{l,s,s'}^{i,j}, \mathbf{b}_{l,s,s'}^{i,j}\}$ and $\{\mathbf{W}_{l,s',s}^{i,j}, \mathbf{b}_{l,s',s}^{i,j}\}$ are the parameters of the linear transformation. By making this concise modification in the model inference, we further clearly boost the performance of the model on different pixel-wise prediction tasks, which will be elaborated in the experimental part.

4 NETWORK IMPLEMENTATION

4.1 Neural network implementation for joint learning

In order to infer the hidden variables and learn the parameters of the AG-CRFs together with those of the front-end CNN, we implement the AG-CRFs updates in neural network. A detailed computing flow is depicted in Figure 4. The implementation consists of several steps:

- Conditional kernel prediction for the kernels $\mathbf{L}_{s_e \rightarrow s_r}$, $\mathbf{l}_{s_e \rightarrow s_r}$ and $\mathbf{l}_{s_r \rightarrow s_e}$ with $\mathbf{L}_{s_e \rightarrow s_r} \leftarrow \mathbf{W}_{L,s_e \rightarrow s_r} \otimes (\mathbf{h}_{s_e} \odot \mathbf{h}_{s_r})$, $\mathbf{l}_{s_e \rightarrow s_r} \leftarrow \mathbf{W}_{l,s_e \rightarrow s_r} \otimes \mathbf{h}_{s_e}$, and $\mathbf{l}_{s_r \rightarrow s_e} \leftarrow \mathbf{W}_{l,s_r \rightarrow s_e} \otimes \mathbf{h}_{s_r}$;
- Message passing from the s_e -th scale to the current s_r -th scale is performed with $\mathbf{h}_{s_e \rightarrow s_r} \leftarrow \mathbf{L}_{s_e \rightarrow s_r} \otimes \mathbf{h}_{s_e}$, where $\otimes$ denotes the convolutional operation and $\mathbf{L}_{s_e \rightarrow s_r}$ denotes the corresponding convolution kernel;
- Attention map estimation $q(\mathbf{g}_{s_e, s_r} = 1) \leftarrow \sigma(\mathbf{h}_{s_r} \odot (\mathbf{L}_{s_e \rightarrow s_r} \otimes \mathbf{h}_{s_e}) + \mathbf{l}_{s_e \rightarrow s_r} \otimes \mathbf{h}_{s_e} + \mathbf{l}_{s_r \rightarrow s_e} \otimes \mathbf{h}_{s_r})$, where $\mathbf{L}_{s_e \rightarrow s_r}$, $\mathbf{l}_{s_e \rightarrow s_r}$ and $\mathbf{l}_{s_r \rightarrow s_e}$ are convolution kernels and $\odot$ represents element-wise product operation;
- Attention-gated message passing from other scales and adding unary term: $\bar{\mathbf{h}}_{s_r} = \mathbf{f}_{s_r} \oplus a_{s_r} \sum_{s_e \neq s_r} (q(\mathbf{g}_{s_e, s_r} = 1) \odot \mathbf{h}_{s_e \rightarrow s_r})$, where a_{s_r} encodes the effect of the $a_{s_r}^i$ for weighting the message and can be implemented as a 1×1 convolution. The symbol $\oplus$ denotes element-wise addition.

In order to simplify the overall inference procedure, and because the magnitude of the linear term of ψ_h is in practice negligible compared to the quadratic term, we discard the message associated to the linear term. When the inference is complete, the final estimate is obtained by convolving all the scales. For the inference of the partially latent model, we only need to discard the corresponding terms in the computation of the messages in the second step, and replace the latent features with observation features for the attention prediction in the third step.

4.2 Exploiting AG-CRFs with a Multi-scale Network

PGA-Net Architecture. The proposed Attention-guided Multi-scale Hierarchical Network (PGA-Net), as sketched in Figure 3, consists of a multi-scale hierarchical network (MH-Net) together with the AG-CRF model described above. The MH-Net is constructed from a front-end CNN architecture such as the widely used AlexNet [71], VGG [72] and ResNet [73]. One prominent feature of MH-Net is its ability to generate richer multi-scale representations. In order to do that, we perform distinct non-linear mappings (deconvolution $\mathbf{D}$, convolution $\mathbf{C}$ and max-pooling $\mathbf{M}$) upon $\mathbf{f}_l$, the CNN feature representation from an intermediate layer l of the front-end CNN. This leads to a three-way representation: $\mathbf{f}_l^{\mathbf{D}}$, $\mathbf{f}_l^{\mathbf{C}}$ and $\mathbf{f}_l^{\mathbf{M}}$. Remarkably, while $\mathbf{D}$ upsamples the feature map, $\mathbf{C}$ maintains its original size and $\mathbf{M}$ reduces it, and different kernel size is utilized for them to have different receptive fields, then naturally obtaining complementary

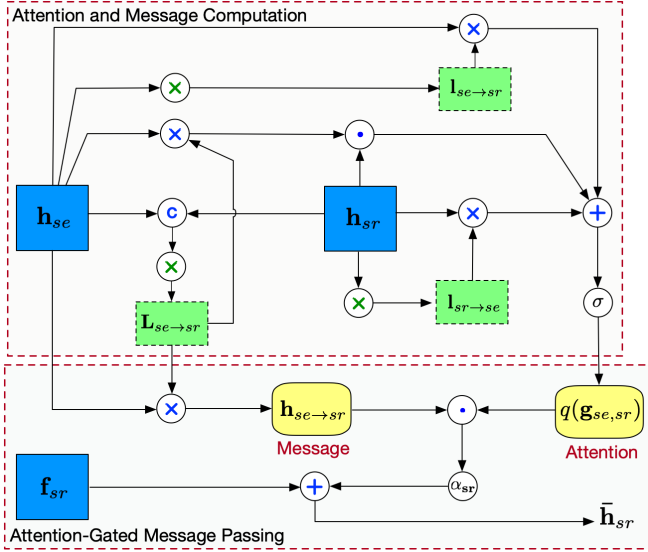


Fig. 4. The detailed computing flow of the mean-field updating of the proposed conditional kernel AG-CRF model. The symbol $\otimes$ denotes the convolutional operation. The ones with green color represent the operation for the conditional kernel prediction. The symbols $\odot$ and $\oplus$ denote element-wise multiplication and addition operation, respectively. The symbols σ and $\oplus$ represent a sigmoid and a concatenation operation, respectively.

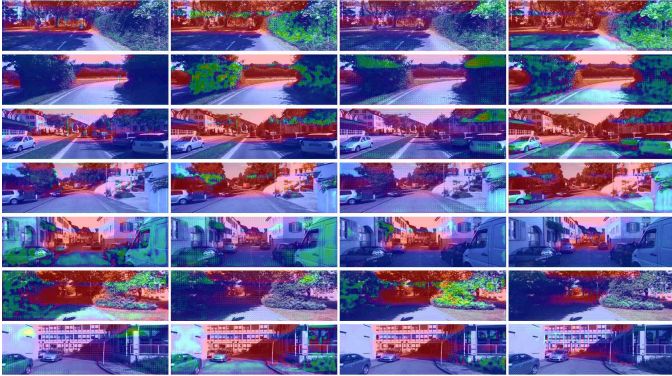


Fig. 5. The visualization of the learned attention maps in the proposed AG-CRF model. Our attention is a pixel-wise attention, *i.e.* simultaneously learning both spatial- and channel-wise attention. We visualize the attention by uniformly sampling four attention channels of the attention map. The learned attentions could capture distinct meaningful parts of the features for guiding the message passing. These attention maps are learned on the KITTI dataset for the task of monocular depth estimation.

inter- and multi-scale representations. The f_l^C and f_l^M are further aligned to the dimensions of the feature map f_l^D by the deconvolutional operation. The hierarchy is implemented in two levels. The first level uses an AG-CRF model to fuse the three representations of each layer l , thus refining the CNN features within the same scale. The second level of the hierarchy uses an AG-CRF model to fuse the information coming from multiple CNN layers. The proposed hierarchical multi-scale structure is general purpose and able to involve an arbitrary number of layers and of diverse intra-layer representations. It should be also noted that the proposed AG-CRF model is flexible to be applied into any multi-scale context in a deep learning network for structured representation refinement.

End-to-End Network Optimization. The parameters of the

model consist of the front-end CNN parameters, $\mathbf{W}_c$, the parameters to produce the richer decomposition from each layer l , $\mathbf{W}_l$, the parameters of the AG-CRFs of the first level of the hierarchy, $\{\mathbf{W}_l^1\}_{l=1}^L$, and the parameters of the AG-CRFs of the second level of the hierarchy, $\mathbf{W}^{\text{II}}$. L is the number of intermediate layers used from the front-end CNN. In order to jointly optimize all these parameters we adopt deep supervision [2] and we add an optimization loss associated to each AG-CRF module. We apply the proposed model into different pixel-wise prediction tasks, including contour detection, monocular depth estimation and semantic segmentation. For the contour detection task, as the contour pixels are significantly less than non-contour pixels, we employ the modified cross-entropy loss function of [2]. Given a training data set $\mathcal{D} = \{(\mathbf{I}_p, \mathbf{E}_p)\}_{p=1}^P$ consisting of P RGB-contour ground-truth pairs, the loss function ℓ writes:

$$\ell(\mathbf{W}) = \sum_p \beta \sum_{e_p^k \in \mathbf{E}_p^+} \log P(e_p^k = 1 | \mathbf{I}_p; \mathbf{W}) + (1 - \beta) \sum_{e_p^k \in \mathbf{E}_p^-} \log P(e_p^k = 0 | \mathbf{I}_p; \mathbf{W}), \quad (15)$$

where $\beta = |\mathbf{E}_p^+| / (|\mathbf{E}_p^+| + |\mathbf{E}_p^-|)$, $\mathbf{E}_p^+$ is the set of contour pixels of image p and $\mathbf{W}$ is the set of all parameters. For the monocular depth estimation, we utilize an L2 loss for the continuous regression as in previous works [6], [15], and for the semantic segmentation, we use a standard cross-entropy loss for multi-class classification as in [74]. The network optimization is performed via the back-propagation algorithm with stochastic gradient descent.

PGA-Net for pixel-wise prediction. After training of the whole PGA-Net, the optimized network parameters $\mathbf{W}$ are used for the pixel-wise prediction task. Given a new test image $\mathbf{I}$, the $L + 1$ classifiers produce a set of multi-scale prediction maps $\{\hat{\mathbf{E}}_l\}_{l=1}^{L+1} = \text{PGA-Net}(\mathbf{I}; \mathbf{W})$. Multi-scale predictions $\hat{\mathbf{E}}_l$ obtained from the AG-CRFs with elementary operations on the contour prediction task are shown in Fig. 6. We inspire from [2] to fuse the multiple scale predictions thus obtaining an average prediction $\hat{\mathbf{E}} = \sum_l \hat{\mathbf{E}}_l / (L + 1)$.

5 EXPERIMENTS

In this section, we demonstrate the effectiveness of the proposed approach through extensive experiments on several publicly available benchmarks, and on three different tasks involving both the continuous domain (*i.e.* monocular depth estimation) and the discrete domain (*i.e.* object contour detection and semantic segmentation). We first introduce the experimental setup and then present our results and analysis.

5.1 Experimental Setup

5.1.1 Datasets.

BSDS500 and NYUD-V2 for object contour detection. For the object contour detection task, we employ two different benchmarks: the BSDS500 and the NYUD-V2 datasets. The BSDS500 dataset is an extended dataset based on BSDS300 [11]. It consists of 200 training, 100 validation and 200 testing images. The groundtruth pixel-level labels for

TABLE 1

BSDS500 dataset: quantitative results. All the compared methods use the official training and testing data. RCF [20] utilizes extra training data for model learning. Our model outperforms Deep Crip Boundary [21] considering the same ResNet50 backbone. Res16x is an enhanced backbone based on ResNet50. 3S/5S indicates three/five feature scales are considered for the structured fusion.

Method	Backbone	ODS	OIS	AP
Human	-	.800	.800	-
Felz-Hutt [76]	-	.610	.640	.560
Mean Shift [77]	-	.640	.680	.560
Normalized Cuts [78]	-	.641	.674	.447
ISCR [79]	-	.724	.752	.783
gPb-ucm [11]	-	.726	.760	.727
Sketch Tokens [80]	-	.727	.746	.780
MCG [81]	-	.747	.779	.759
LEP [82]	-	.757	.793	.828
DeepEdge [18]	AlexNet	.753	.772	.807
DeepContour [17]	AlexNet	.756	.773	.797
HED [2]	VGG16	.788	.808	.840
CEDN [19]	VGG16	.788	.804	.834
COB [4]	ResNet50	.793	.820	.859
Deep Crip Boundary [21]	ResNet50	.803	.820	.871
Deep Crip Boundary [21]	Res16x	.810	.829	.879
RCF [20] (not comp.)	ResNet50	.811	.830	-
PGA-Net (fusion) (3S)	ResNet50	.798	.829	.869
PGA-Net (fusion) w/ CK (3S)	ResNet50	.799	.831	.872
PGA-Net (fusion) w/ CK (5S)	ResNet50	.805	.835	.878

each sample are derived considering multiple annotators. Following [2], [19], we use all the training and validation images for learning the proposed model and perform data augmentation as described in [2]. The NYUD-V2 [12] contains 1449 RGB-D images and it is split into three subsets, consisting of 381 training, 414 validation and 654 testing images. Following [2] in our experiments we employ images at full resolution (*i.e.* 560×425 pixels) both in the training and in the testing phases.

KITTI for monocular depth estimation. For the monocular depth estimation task, the KITTI dataset [13] is considered. This dataset is collected for various important computer vision tasks within a context of self-driving. It contains depth video data capture using a LiDAR sensor installed on a driving car. To have a fair comparison with existing works, we follow a standard setting of the training and testing split originally proposed by Eigen *et al.* [29]. There are in total 61 scenes selected from the raw data distribution. Specifically, we use total 22,600 frames from 32 scenes for training, and 697 frames from the rest 29 scenes for testing. The sparse ground-truth depth maps are obtained by reprojecting the 3D points captured from a velodyne laser onto the left monocular camera as in [75]. The resolution of RGB images are down-sampled to 621×188 from the original resolution of 1224×368 for training.

Pascal-Context for semantic segmentation. For the semantic segmentation task, we use the Pascal-Context dataset [14]. The Pascal-Context dataset performs the augmentation of the pixel-level segmentation annotations on the Pascal VOC 2010, and enlarges the number of semantic classes from original 20 categories to more than 400 categories. Following previous works [5], [74], we evaluate on the setting with the most frequent 59 classes, in total 60 classes plus the background class. The rest classes are masked out during training.

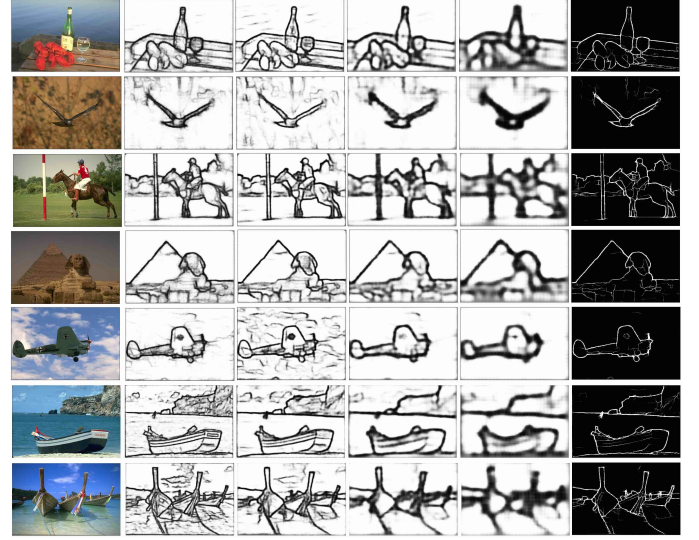


Fig. 6. Examples of predictions from different multi-scale features on BSDS500. The first column is the input test images. The 2nd to the 5th columns show the predictions from different multi-scale features. The last column shows the final contour map after standard NMS.

5.1.2 Evaluation Metrics.

Evaluation protocol on object contour detection. During the test phase standard non-maximum suppression (NMS) [83] is first applied to produce thinned contour maps. We then evaluate the detection performance of our approach according to different metrics, including the F-measure at Optimal Dataset Scale (ODS) and Optimal Image Scale (OIS) and the Average Precision (AP). The maximum tolerance allowed for correct matches of edge predictions to the ground truth is set to 0.0075 for the BSDS500 dataset, and to .011 for the NYUDv2 dataset as in previous works [2], [83], [84].

Evaluation protocol on monocular depth estimation. Following the standard evaluation protocol as in previous works [22], [23], [29], the following quantitative evaluation metrics are adopted in our experiments:

- mean relative error (rel): $\frac{1}{K} \sum_{i=1}^K \frac{|\tilde{d}_i - d_i^*|}{d_i^*}$;
- root mean squared error (rms): $\sqrt{\frac{1}{K} \sum_{i=1}^K (\tilde{d}_i - d_i^*)^2}$;
- mean log10 error (log10): $\frac{1}{K} \sum_{i=1}^K \|\log_{10}(\tilde{d}_i) - \log_{10}(d_i^*)\|$;
- scale invariant rms log error as used in [29], rms(sc-inv.);
- accuracy with threshold t : percentage (%) of d_i^* , subject to $\max(\frac{d_i^*}{\tilde{d}_i}, \frac{\tilde{d}_i}{d_i^*}) = \delta < t$ ($t \in [1.25, 1.25^2, 1.25^3]$).

Where $\tilde{d}_i$ and d_i^* is the ground-truth depth and the estimated depth at pixel i respectively; K is the total number of pixels of the test images.

Evaluation protocol on semantic segmentation. Following previous works and use the DeepLab evaluation tool, we report our quantitative results on the standard metrics of pixel accuracy (pixAcc) and mean intersection over union (mIoU) averaged over classes. Both metrics are the higher the better. The background category is all included in the evaluation as in previous works [74], [85].

5.1.3 Implementation Details.

The proposed PGA-Net is implemented under the deep learning framework *Pytorch*. The training and testing phase are

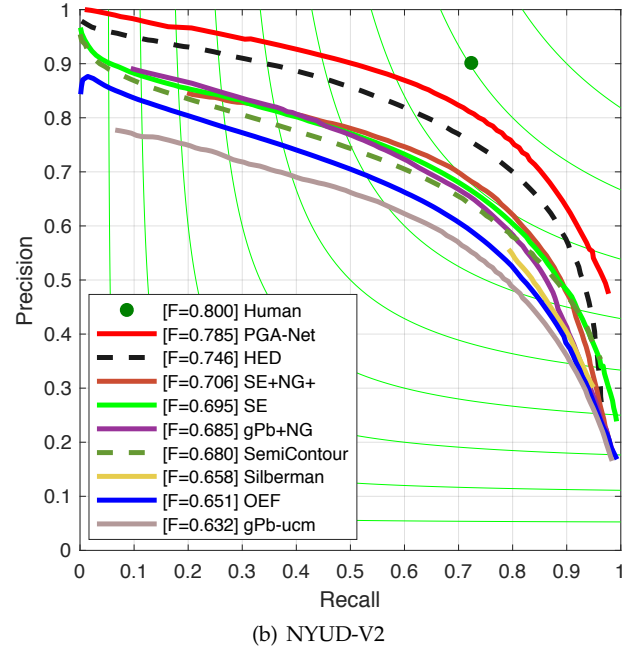
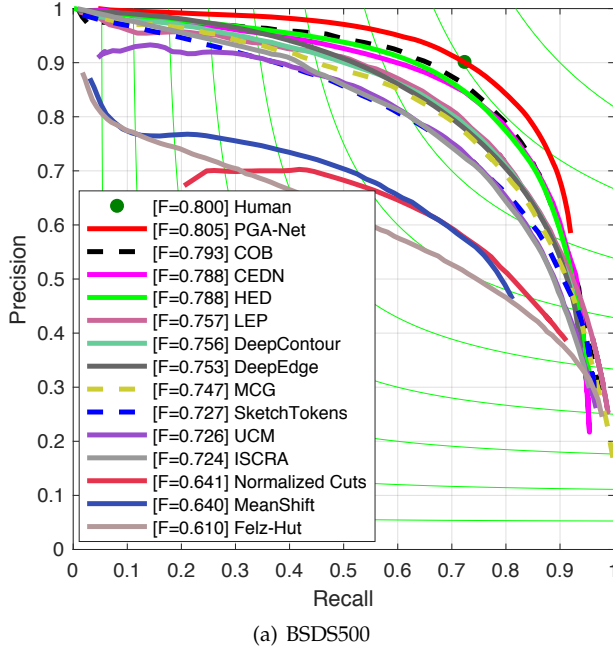


Fig. 7. Precision-Recall Curves on the BSDS500 [11] and NYUD-V2 [12] test sets. The proposed PGA-Net achieves the best performance among the competitors on the ODS metric on both datasets. The results on NYUD-V2 are all based on the RGB and HHA data.

carried out on four Nvidia Tesla P40 GPUs, each with 24GB memory. The ResNet-50 and ResNet-101 networks pretrained on ImageNet [86] are used to initialize the front-end CNN of PGA-Net for different backbone experiments. To consider the computational efficiency, our implementation only employs three scales, *i.e.* we generate multi-scale features from three different semantic layers of the backbone CNN (*i.e.* *res3d*, *res4f*, *res5c* in ResNet). In our CRF model we consider dependencies between all scales. Within the AG-CRFs, the kernel size for all convolutional operations is set to 3×3 with stride 1 and padding 1. The weighting parameters $a_{s_r}^i$ are learned automatically via using convolutional operations with a kernel size of 1×1 . For the object contour detection, the initial learning rate is set to $1e-7$ in all our experiments, and decreases 10 times after every 10k iterations. The total number of iterations for BSDS500 and NYUD v2 is 40k and 30k, respectively. The momentum and weight decay parameters are set to 0.9 and 0.0002, as in [2]. As the training images have different resolution, we need to set the batch size to 1, and for the sake of smooth convergence we updated the parameters only every 10 iterations. For the monocular depth estimation and the semantic segmentation task, following previous works [74], [87], [88] for fair comparison, the batch size is set to 8 and 16, respectively; the learning rate is set to 0.001 with a momentum of 0.9 and a weight decay of 0.0001 using a polynomial learning rate scheme as used in [5], [74]. Regarding the overall training time, it takes around 8, 13, 20 hours for the contour detection on BSDS500, depth estimation on KITTI and semantic segmentation on Pascal-Context. Our model also achieves almost real-time inference time (around 8 frames per second) for all the three tasks.

5.2 Experimental Results

In this section, we present the results of our evaluation, comparing the proposed model with several state of the

TABLE 2
Quantitative performance comparison on NYUD-V2 RGB dataset for the contour detection task with the official training/testing protocols. We achieve better performance in terms of all the metrics than RCF [20], which is the best performing method on BSDS500 with extra training data. 3S/5S indicates three/five feature scales are considered.

Method	Backbone	ODS	OIS	AP
gPb-ucm [11]	–	.632	.661	.562
OEF [89]	–	.651	.667	–
Silberman <i>et al.</i> [12]	–	.658	.661	–
SemiContour [90]	–	.680	.700	.690
SE [91]	–	.685	.699	.679
gPb+NG [92]	–	.687	.716	.629
SE+NG+ [84]	–	.710	.723	.738
HED (RGB) [2]	VGG16	.720	.734	.734
HED (HHA) [2]	VGG16	.682	.695	.702
HED (RGB + HHA) [2]	VGG16	.746	.761	.786
RCF (RGB) + HHA [20]	VGG16	.757	.771	–
RCF (RGB + HHA) [20]	ResNet50	.781	.793	–
PGA-Net (HHA) (3S)	ResNet50	.716	.729	.734
PGA-Net (RGB) (3S)	ResNet50	.744	.758	.765
PGA-Net (RGB+HHA) (3S)	ResNet50	.771	.786	.802
PGA-Net (RGB+HHA) w/ CK (3S)	ResNet50	.780	.795	.813
PGA-Net (RGB+HHA) w/ CK (5S)	ResNet50	.785	.799	.816

art methods respectively on the three different tasks. We further conduct an in-depth analysis of our model, to show the impact of different components on the performance. And Finally we present some qualitative results and analysis of the model.

5.2.1 Comparison with state of the art methods.

Comparison on BSDS500 and NYUD-V2. We first consider the BSDS500 dataset and compare the performance of our approach with several traditional contour detection methods, including Felz-Hut [76], MeanShift [77], Normalized Cuts [78], ISCRA [79], gPb-ucm [11], SketchTokens [80], MCG [81], LEP [82], and more recent CNN-based methods, including DeepEdge [18], DeepContour [17], HED [2],

TABLE 3

Quantitative performance analysis of the proposed PGA-Net on NYUD-V2 RGB dataset for the contour detection task. H2 indicates only the second hierarchy is utilized; 3S and 5S indicate three and five feature scales are considered for the structured fusion, respectively.

Method	Backbone	ODS	OIS	AP
Hypercolumn [16] (5S)	ResNet50	.720	.731	.733
HED [2] (5S)	ResNet50	.722	.737	.738
PGA-Net (baseline) (H2, 3S)	ResNet50	.711	.720	.724
PGA-Net (w/o AG-CRFs) (3S)	ResNet50	.722	.732	.739
PGA-Net (w/ CRFs) (3S)	ResNet50	.732	.742	.750
PGA-Net (w/o deep supervision) (3S)	ResNet50	.725	.738	.740
PGA-Net (w/ PLAG-CRFs) (H2, 3S)	ResNet50	.731	.742	.743
PGA-Net (w/ PLAG-CRFs) (3S)	ResNet50	.737	.749	.746
PGA-Net (w/ FLAG-CRFs) (3S)	ResNet50	.744	.758	.765
PGA-Net (w/ FLAG-CRFs + CK) (3S)	ResNet50	.751	.767	.778
PGA-Net (w/ FLAG-CRFs + CK) (5S)	ResNet50	.754	.772	.781

CEDN [19], COB [4], and Deep Crisp Boundaries [21]. We also report results of the RCF method [20], although they are not comparable because in [20] an extra dataset (*i.e.* Pascal Context) which is even larger than BSDS500, was used during RCF training to improve the results on BSDS500. In this series of experiments we consider PGA-Net with FLAG-CRFs. The results of this comparison are shown in Table 1 and Figure 7a. PGA-Net obtains an F-measure (ODS) of 0.798, thus outperforms all previous methods. The improvement over the second and third best approaches, *i.e.* COB and HED, is 0.5% and 1.0%, respectively, which is not trivial to achieve on this challenging dataset. Furthermore, when considering the OIS and AP metrics, our approach is also better, with a clear performance gap. By using the proposed strategy of conditional kernels, we further clearly boost the performance of PGA-Net on all the three metrics (*i.e.* ODS, OIS and AP), on this performance saturated dataset. Besides, comparing to Deep Crisp Boundaries, ours using 3 feature scales is comparable on the ODS metric while achieves better performance w.r.t. both the OIS and AP metrics if the same backbone architecture (*i.e.* ResNet50) is considered for both methods. While five feature scales are used for the structured fusion, ours outperforms Deep Crisp Boundaries on all the metrics with the ResNet50 backbone.

To conduct the experimental comparison on NYUDv2, following previous works [2] we also consider three different types of input representations, *i.e.* RGB, HHA [84] and RGB-HHA data. The HHA data [84] are encoded depth feature images, in which the three image channels are horizontal disparity, height above ground, and angle of the pixel's local surface normal with the inferred direction of gravity, respectively. The results corresponding to the use of both RGB and HHA data (*i.e.* RGB+HHA) are obtained by performing a weighted average of the estimates obtained from two PGA-Net models trained separately on RGB and HHA representations. As baselines we consider gPb-ucm [11], OEF [89], the method in [12], SemiContour [90], SE [91], gPb+NG [92], SE+NG+ [84], HED [2] and RCF [20]. On the NYUD-V2 dataset, our approach outperforms the RCF [20] which obtained the best performance on BSDS500 with extra training data, with a clear performance gap, where the experimental protocol for both is exactly the same on this dataset. All of them are reported in Table 2 and Figure 7b. Again, our final model (PGA-Net w/ CK) significantly

outperforms all previous comparison methods. In particular, the increased performance with respect to HED [2] and RCF [20] confirms the benefit of the proposed multi-scale feature learning and fusion scheme.

Comparison on KITTI. The state of the art comparison on the KITTI dataset for monocular depth estimation is shown in Table 4. We compare with the methods with both supervised and unsupervised settings. For the unsupervised setting, the representative works such as Zhou *et al.* [93], Garg *et al.* [75], Godard *et al.* [87], and Kuznetsov *et al.* [99] are compared. For the supervised methods, we consider the very competitive works such as Eigen *et al.* [29], Liu *et al.* [6], Kuznetsov *et al.* [99], Gan *et al.* [27], DORN [26] and Lee *et al.* [28]. Our approach also employs the supervised setting using single monocular images in training and testing. As shown in Table 4, our approach achieves top-level performance compared with both the supervised and unsupervised comparison methods. Specifically, DORN needs to assume the depth range in the training, which is not the same setting as our continuous regression, and thus not directly comparable to ours. Besides, DORN specifically works on the predictions via using an ordinal regression loss, while ours focuses on learning effective representations, therefore we are complementary to each other. By combining the proposed AG-CRF module with DORN, specifically with its multi-scale backbone and ordinary regression module, we achieve clearly better performance than DORN, which further confirms the effectiveness of the proposed AG-CRF model. Our approach also obtains the same level of performance compared with the best performing method Lee *et al.* [28] using 3 feature scales. We outperform Lee *et al.* [28] while 5 feature scales are further considered. More importantly, the proposed graph-based approach obtains significantly better results than the CRF-based methods (*i.e.* MS-CRF [88] and Liu *et al.* [6]) on the monocular depth estimation task.

Comparison on Pascal-Context. We compare the proposed PGA-Net with the most competitive methods on the Pascal-Context dataset, including ASPP [102], PSPNet [103], EncNet [74] and HRNet [47]. The experiments are conducted on both ResNet-50 and ResNet-101 backbone networks. Our PGA-Net is 2.24 and 3.05 points better on the mIoU metric than the popular method (*i.e.* EncNet, which considers a channel-wise attention for feature refinement) with ResNet-50 and ResNet-101 respectively. Compared with another multi-scale method HRNet, which utilizes the multi-scale aggregation from feature maps with different resolutions to boost the final performance, our PGA-Net obtains better performance with a clear gap. Note that the backbone structure HRNetV2-W48 used by HRNet has a bigger network capacity than the ResNet101 backbone we used. Our approach also obtains the same level performance comparing with the best performing method OCR considering the same backbone structure ResNet101. By using five feature scales for the structured fusion in the proposed model, our model outperforms OCR (55.1 vs. 54.8 in terms of mIoU using the same ResNet101 backbone). More importantly, the core idea of OCR utilizing the soft object regions is complementary to ours. We could also observe that our graph-based method significantly outperforms deep-lab-v2 [5] which also utilizes a CRF model.

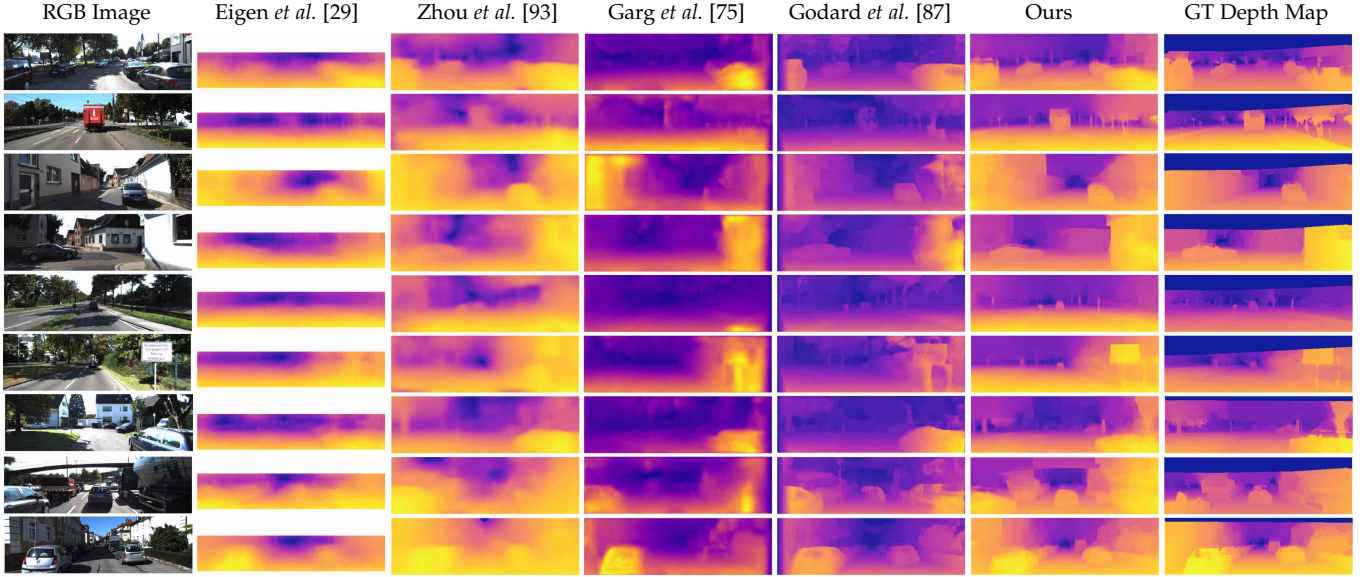


Fig. 8. Qualitative examples of monocular depth prediction on the KITTI raw dataset. The comparison with other competitive methods including Eigen *et al.* [29], Zhou *et al.* [93], Garg *et al.* [75] and Godard *et al.* [87] are presented. We perform bilinear interpolation on the sparse ground-truth depth maps for better visualization.

TABLE 4

Quantitative comparison with the state of the art methods on the KITTI raw dataset for monocular depth estimation. The proposed PGA-Net achieves top performance over all the competitors w.r.t. all the evaluation metrics. The standard training and testing sets split by Eigen *et al.* [29] are used. The 'range' means different ground-truth depth range for evaluation, and the 'sup' means the ground-truth depth is used for supervision in the training. 'CK' denotes the proposed conditional kernel strategy. The methods requiring video data are marked with *. The DORN method needs to assume the depth range in the training, which is not the same as our continuous regression setting, and thus not directly comparable, and we are highly complementary to each other since we work on learning effective representation while theirs focus on the loss level. 3S/5S indicates three/five feature scales from the front-end CNN are considered for the structured fusion by the proposed model.

Method	Setting		Error (lower is better)					Accuracy (higher is better)		
	range	sup?	rel	sq rel	rmse	rmse (log)		$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Garg <i>et al.</i> [75]	80m	No	0.177	1.169	5.285	-		0.727	0.896	0.962
Garg <i>et al.</i> [75] L12 + Aug 8x	50m	No	0.169	1.080	5.104	-		0.740	0.904	0.958
Godard <i>et al.</i> [87]	80m	No	0.148	1.344	5.927	0.247		0.803	0.922	0.964
Zhou* <i>et al.</i> [93]	80m	No	0.208	1.768	6.858	0.283		0.678	0.885	0.957
AdaDepth [94]	50m	No	0.203	1.734	6.251	0.284		0.687	0.899	0.958
Pilzer <i>et al.</i> [95]	80m	No	0.152	1.388	6.016	0.247		0.789	0.918	0.965
Wang & Lucey <i>et al.</i> [96]	80m	No	0.151	1.257	5.583	0.228		0.810	0.936	0.974
DF-Net* [97]	80m	No	0.150	1.124	5.507	0.223		0.806	0.933	0.973
Zhan* <i>et al.</i> [98]	80m	No	0.144	1.391	5.869	0.241		0.803	0.933	0.971
Kuznetsov <i>et al.</i> [99]	80m	No	-	-	4.621	-		0.852	0.960	0.986
Saxena <i>et al.</i> [100]	80m	Yes	0.280	-	8.734	0.327		0.601	0.820	0.926
Liu <i>et al.</i> [6]	80m	Yes	0.217	0.092	7.046	-		0.656	0.881	0.958
Eigen <i>et al.</i> [29]	80m	Yes	0.190	-	7.156	0.246		0.692	0.899	0.967
Mahjourian* [101]	80m	Yes	0.163	1.240	6.220	0.250		0.762	0.916	0.968
MS-CRF [88]	80m	Yes	0.125	0.899	4.685	-		0.816	0.951	0.983
Kuznetsov <i>et al.</i> [99] (supervised & stereo)	80m	Yes	-	-	4.621	0.189		0.862	0.960	0.986
Gan <i>et al.</i> [27]	80m	Yes	0.098	0.666	3.933	0.173		0.890	0.964	0.985
DORN [26]	80m	Yes	0.072	0.307	2.727	0.120		0.932	0.984	0.994
Lee <i>et al.</i> (ResNet) [28]	80m	Yes	0.061	0.261	2.834	0.099		0.954	0.992	0.998
PGA-Net (baseline)	80m	Yes	0.152	0.973	4.902	0.176		0.782	0.931	0.975
PGA-Net (w/ CRFs) (3S)	80m	Yes	0.140	0.942	4.823	0.171		0.793	0.941	0.979
PGA-Net (w/ FLAG-CRFs) (3S)	80m	Yes	0.134	0.909	4.796	0.167		0.801	0.951	0.981
PGA-Net (w/ FLAG-CRFs) (3S)	80m	Yes	0.126	0.901	4.689	0.157		0.813	0.950	0.982
PGA-Net (w/ FLAG-CRFs + CK) (3S)	80m	Yes	0.118	0.752	4.449	0.181		0.852	0.962	0.987
PGA-Net (w/ FLAG-CRFs + CK + DORN) (3S)	80m	Yes	0.063	0.267	2.634	0.101		0.952	0.992	0.998
PGA-Net (w/ FLAG-CRFs + CK + DORN) (5S)	80m	Yes	0.060	0.258	2.595	0.097		0.954	0.993	0.998

5.2.2 Model Analysis.

Baseline models. To further demonstrate the effectiveness of the proposed model and analyze the impact of the different components of PGA-Net on the countour detection task, we conduct an ablation study considering the NYUDv2 (RGB data) and the Pascal-Context dataset. We evaluated the following baseline models: (i) PGA-Net (baseline), which removes the first-level hierarchy and directly concatenates

the feature maps for prediction, (ii) PGA-Net (w/o AG-CRFs), which employs the proposed multi-scale hierarchical structure but discards the AG-CRFs, (iii) PGA-Net (w/ CRFs), which replaces our AG-CRFs with a multi-scale CRF model without attention gating, (iv) PGA-Net (w/o deep supervision) obtained by removing intermediate loss functions in PGA-Net, (v) PGA-Net with the proposed two versions of the AG-CRFs model, *i.e.* FLAG-CRFs and FLA-CRFs, and (vi)

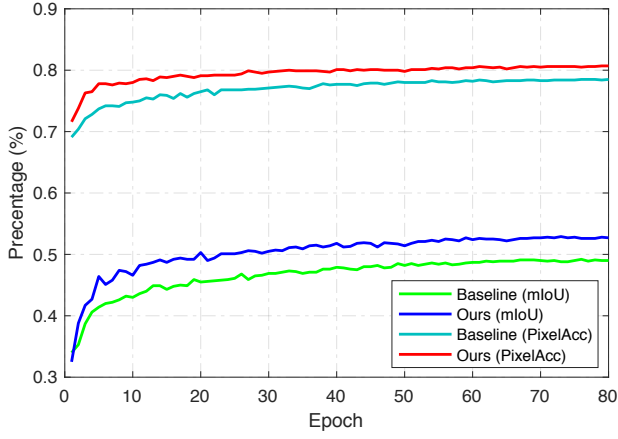


Fig. 9. Training curves of our approach and the baseline model (*i.e.* ours with the proposed AG-CRFs disabled) in terms of both the mIoU and PixAcc metrics on the Pascal-Context dataset. The number of overall training epochs is 80.

TABLE 5

Quantitative performance analysis of the proposed PGA-Net with the ResNet 50 backbone on Pascal-Context for the semantic segmentation task. ‘CK’ denotes the proposed conditional kernel strategy. H2 denotes only the second hierarchy is used. 3S, 4S and 5S indicate three, four and five feature scales from different stages are considered, respectively.

Method	Backbone	pixAcc(%)	mIoU(%)
Hypercolumn [16] (5S)	ResNet50	75.96	47.88
HED [2] (5S)	ResNet50	76.45	48.41
HRNet [47]	HRNetV2-W48	-	54.00
PGA-Net (baseline) (H2, 3S)	ResNet50	75.60	47.21
PGA-Net (w/o AG-CRFs) (3S)	ResNet50	76.70	48.73
PGA-Net (w/ CRFs) (3S)	ResNet50	77.10	49.12
PGA-Net (w/o deep supervision) (3S)	ResNet50	76.90	48.92
PGA-Net (w/ FLAG-CRFs) (H2, 3S)	ResNet50	77.41	49.22
PGA-Net (w/ FLAG-CRFs) (3S)	ResNet50	77.91	50.13
PGA-Net (w/ FLAG-CRFs) (3S)	ResNet50	78.49	51.01
PGA-Net (w/ FLAG-CRFs + CK) (3S)	ResNet50	79.62	52.15
PGA-Net (w/ FLAG-CRFs + CK) (5S)	ResNet50	80.42	53.25
PGA-Net (w/ FLAG-CRFs + CK) (4S)	HRNetV2-W48	81.31	55.18

PGA-Net w/ CK, which uses the proposed conditional kernel strategy. We also consider as reference traditional multi-scale deep learning models employing multi-scale representations, *i.e.* Hypercolumn [16] and HED [2].

Analysis. The quantitative results on different baseline models are shown in Table 3 and 5. The results clearly show the advantages of our contributions. The ODS F-measure of PGA-Net (w/o AG-CRFs) is 1.1% higher than PGA-Net (baseline), clearly demonstrating the effectiveness of the proposed hierarchical network and confirming our intuition that exploiting more richer and diverse multi-scale representations is beneficial, which could be also verified from the results on the Pascal-Context as shown in Table 5. Table 3 also shows that our AG-CRFs plays a fundamental role for accurate detection, as PGA-Net (w/ FLAG-CRFs) leads to an improvement of 1.9% over PGA-Net (w/o AG-CRFs) in terms of OSD. Besides, we could also observe that the PGA-Net (w/ FLAG-CRF) also boosts the performance over the PGA-Net (baseline) (.711 vs. .731 in terms of the ODS metric) where both of them use only the second hierarchy. Finally, PGA-Net (w/ FLAG-CRFs) is 1.2% and 1.5% better than PGA-Net (w/ CRFs) in ODS and AP metrics respectively, confirming the effectiveness of embedding an attention mechanism in the multi-scale CRF model. In Table 5,

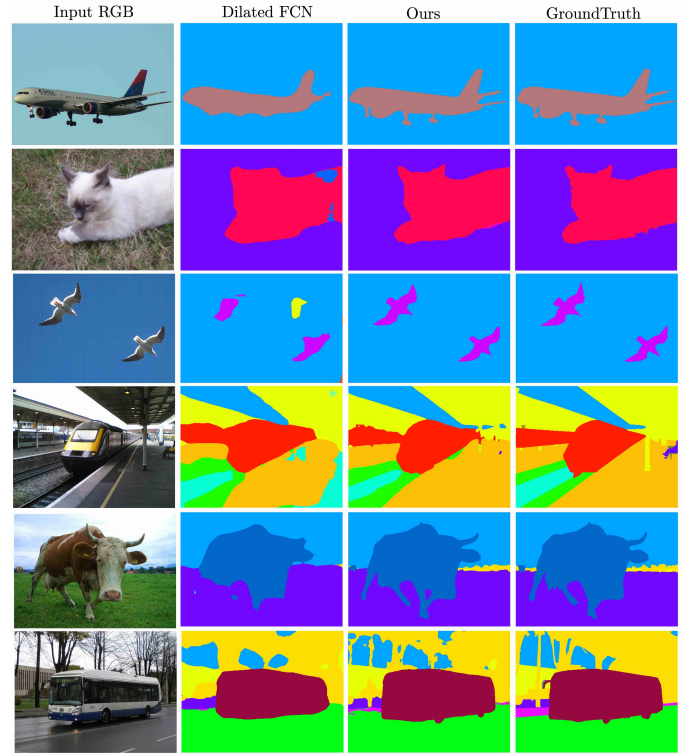


Fig. 10. Qualitative semantic segmentation results on the Pascal-Context dataset. The representative Dilated FCN method [5] is compared.

the mIoU of PGA-Net (w/ FLAG-CRFs) is 1.39 points higher than that of PGA-Net (w/ CRFs), further demonstrating the advantage of the proposed attention mechanism. PGA-Net (w/o deep supervision) decreases the overall performance of our method by 1.9% in ODS, showing the crucial importance of deep supervision for better optimization of the whole PGA-Net. Comparing the performance of the proposed two versions of the AG-CRF model, *i.e.* FLAG-CRFs and CRFs, we can see that PGA-Net (FLAG-CRFs) slightly outperforms PGA-Net (PLAG-CRFs) in both ODS and OIS, while bringing a significant improvement (around 2%) in AP. Finally, considering HED [2] and Hypercolumn [16], it is clear that our PGA-Net (FLAG-CRFs) is significantly better than these methods. Importantly, our approach utilizes only three scales while for HED [2] and Hypercolumn [16] we consider five scales. By considering five scales, our model obtains further improvement upon the results using three scales for contour detection on the NYUD-V2 (see Table 3) and for semantic segmentation on the Pascal-Context (see Table 5). We also deploy the proposed AG-CRF model into a popular multi-scale network architecture HRNet [47]. It is clear to see that our model could also boost its performance, further confirming that our model is able to be applied into different multi-scale context for effective feature fusion. From Table 3 and 5, we can also observe the effectiveness of the proposed conditional kernel strategy. PGA-Net (w/ FLAG-CRFs + CK) clearly improves over PGA-Net (w/ FLAG-CRFs) on the AP (1.3 points) for the contour detection and the mIoU (1.1 points) for the semantic segmentation. We also plot the training curves of the proposed approach and the baseline on the Pascal-Context validation in Figure 9. As shown in the

TABLE 6

Overall performance comparison with state of the art methods on the *val* set of the PASCAL-Context dataset. Spatial context post-processing is used in the pipeline of most methods except for [5], [39], [104]. Our full model achieves the best results compared with both the CRF-based or not the CRF-based approaches on the pixAcc and mIoU metrics. 3S and 5S indicate three and five feature scales are considered, respectively.

Method	Backbone	pixAcc%	mIoU%
CFM (VGG+MCG) [105]	VGG-16	-	34.4
FCN-8s [106]	VGG-16	46.5	35.1
FCN-8s [30]	VGG-16	50.7	37.8
DeepLab-v2 [5]	VGG-16	-	37.6
BoxSup [107]	VGG-16	-	40.5
ConvPP-8s [104]	VGG-16	-	41.0
PixelNet [108]	VGG-16	51.5	41.4
CRF-RNN [39]	VGG-16	-	39.3
DeepLab-v2 + CRF [5]	VGG-16	-	45.7
ASPP [102]	ResNet-50	78.3	49.2
PSPNet [103]	ResNet-50	78.6	50.6
EncNet [74]	ResNet-50	78.4	49.9
EncNet [74]	ResNet-101	79.2	51.7
HRNet [47]	HRNetV2-W48	-	54.0
OCR [109]	ResNet101	-	54.8
PGA-Net (w/ FLAG-CRFs+ CK) (3S)	ResNet-50	79.6	52.2
PGA-Net (w/ FLAG-CRFs+ CK) (3S)	ResNet-101	80.8	54.8
PGA-Net (w/ FLAG-CRFs+ CK) (5S)	ResNet-101	81.2	55.1



Fig. 11. Visualization of the learned attention maps of the proposed AG-CRF model on the Pascal-Context dataset. Our model learns pixel-wise attention in both the spatial and channel dimension. We visualize the attention as we perform on KITTI by sampling four attention channels of the attention map uniformly. The learned attentions are able to capture different semantic regions to guide the message passing among features.

figure, our approach consistently outperforms the baseline model at each training epoch, furthering demonstrating the effectiveness of the proposed AG-CRF model.

5.2.3 Qualitative Analysis.

Attention visualization. Figure 5 and Figure 11 show examples of the learned attention maps in our proposed AG-CRF model on KITTI and Pascal-Context, respectively. As our attention mechanism learns a multi-channel attention map, meaning that the attention map has the same number of channels as the feature map. We visualize four channels of

the overall 256 channels (*i.e.* every the 64-th channel). It can be observed that the learned attention map could capture the informative feature region from both the spatial and channel dimension, which we believe is an important reason for our model to effectively refine the feature maps. Taking the second row in Fig. 11 for instance, it is easy to observe that the dog, the chair and the background are activated on different channels, and for each specific channel, different spatial regions are activated. The dark blue color marks the activated parts/regions.

Prediction visualization. The multi-scale predictions and the final prediction from the PGA-Net on contour detection is shown in Figure 6. It can be observed that the multi-scale predictions are highly complementary to each other, which confirming the initial intuition of modelling the multi-scale predictions in a joint CRF model for structured prediction and fusion. Figure 8 and Figure 10 show examples of the monocular depth estimation on KITTI and the semantic segmentation on Pascal-Context respectively. Different state of the arts methods are compared in the figures. It is clearly that our approach achieves qualitatively better than these methods on both datasets.

6 CONCLUSION

We presented a novel multi-scale probabilistic graph attention networks with conditional kernels for pixel-wise prediction. The proposed model introduces two main components, *i.e.* a hierarchical architecture for generating more rich and complementary multi-scale feature representations, and an Attention-Gated CRF model using conditional kernels for robust feature refinement and fusion. We demonstrate the effectiveness of our approach through extensive experiments on three different pixel-wise prediction tasks, including continuous problems, *i.e.* monocular depth estimation, and discrete problems, *i.e.* object contour detection and semantic segmentation. Four challenging publicly available datasets, BSDS500, NYUD-V2, KITTI and Pascal-Context are considered in our experiments. The proposed model achieved very competitive performance on all the task and the datasets. The proposed approach addresses a general problem, *i.e.* how to learn rich multi-scale representations and optimally fuse them. Therefore, we believe it may be also beneficial for other continuous and discrete pixel-level prediction tasks.

ACKNOWLEDGMENTS

The authors would like to thank Nvidia Corporation for GPU donations for supporting the research.

REFERENCES

- [1] X. Ren, "Multi-scale improves boundary detection in natural images," in *ECCV*, 2008.
- [2] S. Xie and Z. Tu, "Holistically-nested edge detection," in *CVPR*, 2015.
- [3] I. Kokkinos, "Pushing the boundaries of boundary detection using deep learning," in *ICLR*, 2016.
- [4] K.-K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool, "Convolutional oriented boundaries," in *ECCV*, 2016.
- [5] 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.

- [6] F. Liu, C. Shen, and G. Lin, "Deep convolutional neural fields for depth estimation from a single image," in *CVPR*, 2015.
- [7] X. Chu, W. Ouyang, X. Wang, *et al.*, "Crf-cnn: Modeling structured information in human pose estimation," in *NIPS*, 2016.
- [8] 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*, 2017.
- [9] V. Mnih, N. Heess, A. Graves, *et al.*, "Recurrent models of visual attention," in *NIPS*, 2014.
- [10] T. Minka and J. Winn, "Gates," in *NIPS*, 2009.
- [11] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *TPAMI*, vol. 33, no. 5, pp. 898–916, 2011.
- [12] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *ECCV*, 2012.
- [13] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *IJRR*, vol. 32, no. 11, pp. 1231–1237, 2013.
- [14] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille, "The role of context for object detection and semantic segmentation in the wild," in *CVPR*, 2014.
- [15] D. Xu, W. Ouyang, E. Ricci, X. Wang, and N. Sebe, "Learning cross-modal deep representations for robust pedestrian detection," in *CVPR*, 2017.
- [16] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *CVPR*, 2015.
- [17] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang, "Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection," in *CVPR*, 2015.
- [18] G. Bertasius, J. Shi, and L. Torresani, "Deepedge: A multi-scale bifurcated deep network for top-down contour detection," in *CVPR*, 2015.
- [19] J. Yang, B. Price, S. Cohen, H. Lee, and M.-H. Yang, "Object contour detection with a fully convolutional encoder-decoder network," in *CVPR*, 2016.
- [20] Y. Liu, M.-M. Cheng, X. Hu, K. Wang, and X. Bai, "Richer convolutional features for edge detection," *arXiv preprint arXiv:1612.02103*, 2016.
- [21] Y. Wang, X. Zhao, Y. Li, and K. Huang, "Deep crisp boundaries: From boundaries to higher-level tasks," *TIP*, vol. 28, no. 3, pp. 1285–1298, 2018.
- [22] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *ICCV*, 2015.
- [23] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. Yuille, "Towards unified depth and semantic prediction from a single image," in *CVPR*, 2015.
- [24] A. Roy and S. Todorovic, "Monocular depth estimation using neural regression forest," in *CVPR*, 2016.
- [25] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," *arXiv preprint arXiv:1606.00373*, 2016.
- [26] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *CVPR*, 2018.
- [27] Y. Gan, X. Xu, W. Sun, and L. Lin, "Monocular depth estimation with affinity, vertical pooling, and label enhancement," in *ECCV*, 2018.
- [28] J. H. Lee, M.-K. Han, D. W. Ko, and I. H. Suh, "From big to small: Multi-scale local planar guidance for monocular depth estimation," *arXiv preprint arXiv:1907.10326*, 2019.
- [29] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *NIPS*, 2014.
- [30] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.
- [31] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.
- [32] Y. Yuan and J. Wang, "Ocnet: Object context network for scene parsing," *arXiv preprint arXiv:1809.00916*, 2018.
- [33] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *CVPR*, 2016.
- [34] F. Xia, P. Wang, L.-C. Chen, and A. L. Yuille, "Zoom better to see clearer: Human and object parsing with hierarchical auto-zoom net," in *ECCV*, 2016.
- [35] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *ICCV*, 2015.
- [36] V. Badrinarayanan, A. Handa, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling," *arXiv preprint arXiv:1505.07293*, 2015.
- [37] Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang, "Semantic image segmentation via deep parsing network," in *ICCV*, 2015.
- [38] A. Arnab, S. Jayasumana, S. Zheng, and P. H. Torr, "Higher order conditional random fields in deep neural networks," in *ECCV*, 2016.
- [39] 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.
- [40] Y. Chen, M. Rohrbach, Z. Yan, Y. Shuicheng, J. Feng, and Y. Kalantidis, "Graph-based global reasoning networks," in *CVPR*, 2019.
- [41] 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," in *CVPR*, 2018.
- [42] 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.
- [43] Z. Zhang, Z. Cui, C. Xu, Y. Yan, N. Sebe, and J. Yang, "Pattern-affinitive propagation across depth, surface normal and semantic segmentation," in *CVPR*, 2019.
- [44] S. Vandenhende, S. Georgoulis, and L. Van Gool, "Mti-net: Multi-scale task interaction networks for multi-task learning," in *ECCV*, 2020.
- [45] P. Buysens, A. Elmoataz, and O. Lézoray, "Multiscale convolutional neural networks for vision-based classification of cells," in *ACCV*, 2012.
- [46] D. Chen, D. Xu, H. Li, N. Sebe, and X. Wang, "Group consistent similarity learning via deep crf for person re-identification," in *CVPR*, 2018.
- [47] K. Sun, Y. Zhao, B. Jiang, T. Cheng, B. Xiao, D. Liu, Y. Mu, X. Wang, W. Liu, and J. Wang, "High-resolution representations for labeling pixels and regions," *arXiv preprint arXiv:1904.04514*, 2019.
- [48] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, *et al.*, "Deep high-resolution representation learning for visual recognition," *TPAMI*, 2020.
- [49] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," in *ICLR*, 2015.
- [50] S. Yang and D. Ramanan, "Multi-scale recognition with dag-cnns," in *ICCV*, 2015.
- [51] G. Huang and D. Chen, "Multi-scale dense networks for resource efficient image classification," in *ICLR*, 2018.
- [52] Y. Kim, C. Denton, L. Hoang, and A. M. Rush, "Structured attention networks," in *ICLR*, 2017.
- [53] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *ICLR*, 2017.
- [54] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang, "The application of two-level attention models in deep convolutional neural network for fine-grained image classification," in *CVPR*, 2015.
- [55] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *NIPS*, 2015.
- [56] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *ICML*, 2015.
- [57] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017.
- [58] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *CVPR*, 2019.
- [59] Y. Tang, "Gated boltzmann machine for recognition under occlusion," in *NIPS Workshop on Transfer Learning by Learning Rich Generative Models*, 2010.
- [60] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, "Learning precise timing with lstm recurrent networks," *JMLR*, vol. 3, no. Aug, pp. 115–143, 2002.
- [61] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.
- [62] X. Zeng, W. Ouyang, J. Yan, H. Li, T. Xiao, K. Wang, Y. Liu, Y. Zhou, B. Yang, Z. Wang, *et al.*, "Crafting gbd-net for object detection," *arXiv preprint arXiv:1610.02579*, 2016.

- [63] A. Quattoni, M. Collins, and T. Darrell, "Conditional random fields for object recognition," in *NIPS*, 2005.
- [64] S. Sarawagi and W. W. Cohen, "Semi-markov conditional random fields for information extraction," in *NIPS*, 2005.
- [65] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *ICML*, 2001.
- [66] X. He, R. S. Zemel, and M. Á. Carreira-Perpiñán, "Multiscale conditional random fields for image labeling," in *CVPR*, 2004.
- [67] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *NIPS*, 2011.
- [68] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient nd image segmentation," *IJCV*, vol. 70, no. 2, pp. 109–131, 2006.
- [69] D. Xu, W. Wang, H. Tang, H. Liu, N. Sebe, and E. Ricci, "Structured attention guided convolutional neural fields for monocular depth estimation," in *CVPR*, 2018.
- [70] J. Winn, "Causality with gates," in *AISTATS*, 2012.
- [71] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.
- [72] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [73] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.
- [74] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal, "Context encoding for semantic segmentation," in *CVPR*, 2018.
- [75] R. Garg, G. Carneiro, and I. Reid, "Unsupervised cnn for single view depth estimation: Geometry to the rescue," in *ECCV*, 2016.
- [76] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, vol. 59, no. 2, 2004.
- [77] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *TPAMI*, vol. 24, no. 5, pp. 603–619, 2002.
- [78] J. Shi and J. Malik, "Normalized cuts and image segmentation," *TPAMI*, vol. 22, no. 8, 2000.
- [79] Z. Ren and G. Shakhnarovich, "Image segmentation by cascaded region agglomeration," in *CVPR*, 2013.
- [80] J. J. Lim, C. L. Zitnick, and P. Dollár, "Sketch tokens: A learned mid-level representation for contour and object detection," in *CVPR*, 2013.
- [81] J. Pont-Tuset, P. Arbelaez, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping for image segmentation and object proposal generation," *TPAMI*, vol. 39, no. 1, pp. 128–140, 2016.
- [82] Q. Zhao, "Segmenting natural images with the least effort as humans," in *BMVC*, 2015.
- [83] P. Dollár and C. L. Zitnick, "Structured forests for fast edge detection," in *ICCV*, 2013.
- [84] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from rgb-d images for object detection and segmentation," in *ECCV*, 2014.
- [85] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ade20k dataset," in *CVPR*, 2017.
- [86] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.
- [87] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *CVPR*, 2017.
- [88] D. Xu, E. Ricci, W. Ouyang, X. Wang, N. Sebe, *et al.*, "Monocular depth estimation using multi-scale continuous crfs as sequential deep networks," *TPAMI*, vol. 41, no. 6, pp. 1426–1440, 2018.
- [89] S. Hallman and C. C. Fowlkes, "Oriented edge forests for boundary detection," in *CVPR*, 2015.
- [90] Z. Zhang, F. Xing, X. Shi, and L. Yang, "Semicontour: A semi-supervised learning approach for contour detection," in *CVPR*, 2016.
- [91] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *TPAMI*, vol. 37, no. 8, pp. 1558–1570, 2015.
- [92] S. Gupta, P. Arbelaez, and J. Malik, "Perceptual organization and recognition of indoor scenes from rgb-d images," in *CVPR*, 2013.
- [93] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *CVPR*, 2017.
- [94] J. N. Kundu, P. K. Uppala, A. Pahuja, and R. V. Babu, "Adadepth: Unsupervised content congruent adaptation for depth estimation," in *CVPR*, 2018.
- [95] A. Pilzer, D. Xu, M. Puscas, E. Ricci, and N. Sebe, "Unsupervised adversarial depth estimation using cycled generative networks," in *3DV*, 2018.
- [96] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey, "Learning depth from monocular videos using direct methods," in *CVPR*, 2018.
- [97] Y. Zou, Z. Luo, and J.-B. Huang, "Df-net: Unsupervised joint learning of depth and flow using cross-task consistency," in *ECCV*, 2018.
- [98] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. Reid, "Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction," *arXiv preprint arXiv:1803.03893*, 2018.
- [99] Y. Kuznetsov, J. Stückler, and B. Leibe, "Semi-supervised deep learning for monocular depth map prediction," in *CVPR*, 2017.
- [100] A. Saxena, M. Sun, and A. Y. Ng, "Make3d: Learning 3d scene structure from a single still image," *TPAMI*, vol. 31, no. 5, pp. 824–840, 2009.
- [101] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints," in *CVPR*, 2018.
- [102] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.
- [103] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," *arXiv preprint arXiv:1612.01105*, 2016.
- [104] S. Xie, X. Huang, and Z. Tu, "Top-down learning for structured labeling with convolutional pseudoprior," in *ECCV*, 2016.
- [105] J. Dai, K. He, and J. Sun, "Convolutional feature masking for joint object and stuff segmentation," in *CVPR*, 2015.
- [106] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.
- [107] J. Dai, K. He, and J. Sun, "Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation," in *ICCV*, 2015.
- [108] A. Bansal, X. Chen, B. Russell, A. Gupta, and D. Ramanan, "Pixelnet: Representation of the pixels, by the pixels, and for the pixels," *arXiv preprint arXiv:1702.06506*, 2017.
- [109] Y. Yuan, X. Chen, and J. Wang, "Object-contextual representations for semantic segmentation," *arXiv preprint arXiv:1909.11065*, 2019.



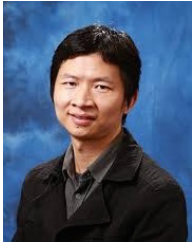
scientific paper award at ICPR 2016. He served as Area Chairs of ACM MM 2020, ICPR 2020 and WACV 2021.



computer vision, machine learning and signal processing for robotics and multimodal social behavior analysis.

Dan Xu is a Postdoctoral Research Fellow in Visual Geometry Group (VGG) at the University of Oxford. He was a Ph.D. in the Department of Information Engineering and Computer Science, and a member of Multimedia and Human Understanding Group (MHUG) at the University of Trento. He was a research assistant of Multimedia Lab in the Department of Electronic Engineering at the Chinese University of Hong Kong. His research focuses on computer vision, multimedia and machine learning. He received the best

Xavier Alameda-Pineda received M.Sc. degrees in mathematics (2008), in telecommunications (2009) and in computer science (2010) and a Ph.D. in mathematics and computer science (2013) from Université Joseph Fourier. Since 2016, he is a Research Scientist at Inria Grenoble Rhône-Alpes, with the Perception team. He served as Area Chair at ICCV'17, of ICIAP'19 and of ACM MM'19. He is the recipient of several paper awards and of the ACM SIGMM Rising Star Award in 2018. His scientific interests lie in



Wanli Ouyang received the PhD degree in the Department of Electronic Engineering, The Chinese University of Hong Kong. He is now a senior lecturer in the School of Electrical and Information Engineering at the University of Sydney, Australia. His research interests include image processing, computer vision and pattern recognition. He is a senior member of IEEE.



Elisa Ricci received the PhD degree from the University of Perugia in 2008. She is an associate professor at the University of Trento and a researcher at Fondazione Bruno Kessler. She has since been a post-doctoral researcher at Idiap, Martigny, and Fondazione Bruno Kessler, Trento. She was also a visiting researcher at the University of Bristol. Her research interests are mainly in the areas of computer vision and machine learning. She is a member of the IEEE.



Xiaogang Wang received the PhD degree in Computer Science from Massachusetts Institute of Technology. He is an associate professor in the Department of Electronic Engineering at the Chinese University of Hong Kong since August 2009. He was the Area Chairs of ICCV 2011 and 2015, ECCV 2014 and 2016, ACCV 2014 and 2016. He received the Outstanding Young Researcher in Automatic Human Behaviour Analysis Award in 2011, Hong Kong RGC Early Career Award in 2012, and CUHK Young Researcher Award 2012.



Nicu Sebe is Professor with the University of Trento, Italy, leading the research in the areas of multimedia information retrieval and human behavior understanding. He was the General Co-Chair of the IEEE FG Conference 2008 and ACM Multimedia 2013, and the Program Chair of the International Conference on Image and Video Retrieval in 2007 and 2010, ACM Multimedia 2007 and 2011. He was the Program Chair of ICCV 2017 and ECCV 2016, and a General Chair of ACM ICMR 2017. He is a fellow of the

International Association for Pattern Recognition.