This ECCV 2018 paper, provided here by the Computer Vision Foundation, is the author-created version.
The content of this paper is identical to the content of the officially published ECCV 2018
LNCS version of the paper as available on SpringerLink: https://link.springer.com/conference/eccv

Joint Task-Recursive Learning for Semantic Segmentation and Depth Estimation

Zhenyu Zhang[1], Zhen Cui[1⋆], Chunyan Xu[1], Zequn Jie[2], Xiang Li[1], Jian Yang[1⋆]

[1]PCA Lab, Key Lab of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, and Jiangsu Key Lab of Image and Video Understanding for Social Security, School of Computer Science and Engineering, Nanjing University of Science and Technology
[2]Tencent AI Lab
{zhangjesse,zhen.cui,cyx,xiang.li.implus,jyang}@njust.edu.cn
{zequn.nus}@gmail.com

Abstract. In this paper, we propose a novel joint Task-Recursive Learning (TRL) framework for the closing-loop semantic segmentation and monocular depth estimation tasks. TRL can recursively refine the results of both tasks through serialized task-level interactions. In order to mutually-boost for each other, we encapsulate the interaction into a specific Task-Attentional Module (TAM) to adaptively enhance some counterpart patterns of both tasks. Further, to make the inference more credible, we propagate previous learning experiences on both tasks into the next network evolution by explicitly concatenating previous responses. The sequence of task-level interactions are finally evolved along a coarse-to-fine scale space such that the required details may be reconstructed progressively. Extensive experiments on NYU-Depth v2 and SUN RGB-D datasets demonstrate that our method achieves state-of-the-art results for monocular depth estimation and semantic segmentation.

Keywords: Depth Estimation, Semantic Segmentation, Recursive Learning, Recurrent Neural Network, Deep Learning

1 Introduction

Semantic segmentation and depth estimation from single monocular images are two challenging tasks in computer vision, due to lack of reliable cues of a scene, large variations of scene types, cluttered backgrounds, pose changing and occlusions of objects. Recently, driven by deep learning techniques, the study on them has seen great progress and starts to benefit some potential applications such as scene understanding [1], robotics [2], autonomous driving [3] and simultaneous localization and mapping (SLAM) system [4]. Despite the successes of deep learning (especially CNNs) on monocular depth estimation [5] [6] [7] [8] [9] and semantic segmentation [10] [11] [12] [13], most of these methods emphasize to learn robust regression yet scarcely consider the interactions between them.

⋆ Corresponding authors: Zhen Cui and Jian Yang.

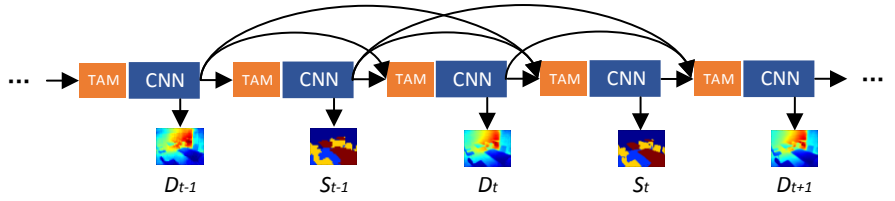

Fig. 1. Illustration of our main idea. The two tasks (i.e., depth estimation and semantic segmentation) are progressively refined to form a task alternate state sequence. At time slice t, we denote the task states as D_t and S_t respectively. Previous task-related experiences and information of the other task are adaptively propagate into the next new state (D_t) via a designed task-interactive module called Task-Attentional Module (TAM). The evolution-alternate process of the dual tasks is finally framed into the proposed task-recursive learning.

Actually, the two tasks have some common characteristics, which can be utilized for each other. For example, semantic segmentation and depth of a scene can both reveal the layout and object shapes/boundaries. The recent work in the literature [14] also indicated that leveraging the depth information from RGB-D data may facilitate the semantic segmentation. Therefore, a joint learning of both tasks should be considered to reciprocally promote for each other.

Existing joint learning of two tasks falls into the category of multi-task learning, which has been extensively studied in the past few decades [15]. It involves many cross tasks, such as detection and classification [16] [17], depth estimation and image decomposition [18], image segmentation and classification [19], and also depth estimation and semantic segmentation [20] [21] [22], etc. But such existing joint learning methods mainly belong to the shallow task-level interaction. For example, a shared deep network is utilized to extract the common features for both tasks, and bifurcates from a high-level layer to perform the two tasks individually [16] [17] [22] [19] [21] [18]. As such, in these methods, less interaction is taken due to the relative independency between tasks. However, it is well known that human learning system benefits from an iterative/looping interactive process between different tasks [23]. Taking a simplest commonsense case, alternately reading and writing can promptly improve human capability in the both aspects. Therefore, we argue whether task-alternate learning (such as cross segmentation and depth estimation) can go deeper with the breakthrough of deep learning.

To address such problem, in this paper, we propose a novel joint Task-Recursive Learning (TRL) framework to closely-loop semantic segmentation and depth estimation on indoor scenes. The interactions between both tasks are serialized as a newly-created time axis, as shown in Fig. 1. Along the time dimension, the two tasks $\{D, S\}$ are mutually collaborate to boost the performance for each other. In each interaction, the historical experiences of previous states (i.e., features of the previous time steps of the two tasks) will be selectively propagated and help to estimate the new state, as ploted by the arc and horizontal black arrows. To properly propagate the information stream, we design a Task-

Attentional Module (TAM) to correlate the two tasks, where the useful common information related to the current task will be enhanced while suppressing task-irrelevant information. Thus the learning process of the two tasks can be easily modularized into a sequence network called task-recursive learning network in this paper. Besides, considering the difficulty of high-resolution pixel-level prediction, we derive the recursive task learning on a sequence of coarse-to-fine scales, which would progressively refine the details of the estimation results. Extensive experiments demonstrate that our proposed task-recursive learning can benefit the two tasks for each other. In summary, the contributions of this paper are three folds:

- Propose a novel joint Task-Recursive Learning (TRL) framework for semantic segmentation and depth estimation. Serializing the problems as a task-alternate time sequence, TRL can progressively refine and mutually boost the two tasks through properly propagating the information stream.
- Design a Task-Attentional Module (TAM) to enclose the interaction of the two tasks, which thus can be used in those conventional networks as a general layer or module.
- Validate the effectiveness of the deeply task-alternate mechanism, and achieve some new state-of-the-art results of for the dual tasks of depth estimation and semantic segmentation on NYU Depth V2 and SUN RGBD datasets.

2 Related Work

Depth Estimation: Many works have been proposed for monocular depth estimation. Eigen *et al.* [5, 24] proposed a multi-stage CNN to resolve the monocular depth prediction. Liu *et al.* [25] and Li *et al.* [26] utilized CRF models to capture local image texture and guide the network learning process. Recently, Laina *et al.* [7] proposed a fully convolutional network with up-projection to achieve an efficient upsampling process. Xu *et al.* [6] employed multi-scale continuous CRFs as a deep sequential network. In contrast to these methods, our approach focuses on the dual-task learning, and attempts to utilize segmentation cues to promote depth prediction.

Semantic Segmentation: Most methods [10, 11, 27–29] conducted semantic segmentation from single RGB image. As the large RGBD dataset was released, some approaches [30, 31] attempted to fuse depth information for better segmentation. Recently, Cheng *et al.* [32] computed the affinity matrices from RGB images and HHA depth images for better upsampling important locations. Different from these RGBD based methods, our method does not directly use ground truth of depth, but the estimated depth for semantic segmentation, which thus essentially falls into the category of RGB image segmentation.

Multi-task Learning: The generic multi-task learning problem [15] has been studied for a long history, and numerous methods were developed in different research areas such as representation learning [33–35], transfer learning [36, 37], computer vision [38, 16, 19, 39, 17, 40]. Here the most related works are those multi-task learning methods of computer vision. For examples, the

literatures [21, 22] utilized CNN with hierarchical CRFs and multi-decoder to obtain depth estimation and semantic segmentation. In the literature [19], a cross-stitch unit was proposed to better interact two tasks. The recent proposed Ubernet [40] attempted to give a solution for various tasks on diverse datasets with limited memory. Different from these previous works, our proposed TR-L takes multi-task learning as a deep manner of task interactions. Specifically, depth estimation and semantic segmentation are mutually boosted and refined in a general recursive architecture.

3 Approach

3.1 Motivation

Here we focus on the interactive learning problem of two tasks including depth estimation and semantic segmentation from a monocular RGB image. Our motivation mainly comes from two folds: i) human learning benefits from an iterative/looping interactive process between tasks [23]; ii) Such a couple of tasks are complementary to some extent besides sharing some common information. Therefore, our aim is to make the task-level alternate interaction go deeper, so as to let the two tasks mutually boosted. The main idea is illustrated in Fig. 1. We define the task-alternate learning processes as a series of state transformation along the time axis. Formally, we denote the states of depth estimation and semantic segmentation tasks as D_p and S_p at time step p respectively, and the corresponding responses as f_D^p and f_S^p. Suppose the previous obtained experiences as $\mathcal{F}_D^{p-1:p-k} = \{f_D^{p-1}, f_D^{p-2}, \ldots, f_D^{p-k}\}$ and $\mathcal{F}_S^{p-1:p-k} = \{f_S^{p-1}, f_S^{p-2}, \ldots, f_S^{p-k}\}$, then we formulate the dual-task learning at the time clip p as

$$\begin{cases} \mathrm{D}^p = \Phi_D^p(\mathcal{T}(\mathcal{F}_D^{p-1:p-k}, \mathcal{F}_S^{p-1:p-k}), \Theta_D^p) \\ \mathrm{S}^p = \Phi_S^p(\mathcal{T}(\mathcal{F}_D^{p:p-k+1}, \ \mathcal{F}_S^{p-1:p-k}), \Theta_S^p) \end{cases}, \tag{1}$$

where $\mathcal{T}$ is the interactive function (designed as task-attentional module below), Φ_D^p and Φ_S^p are transformation functions to predict the next state with the parameters Θ_D^p and Θ_S^p to be learnt. As the time slice p, the depth estimation D_p is on the conditions of previous k-order experiences $\mathcal{F}_D^{p-1:p-k}$ and $\mathcal{F}_S^{p-1:p-k}$, and the segmentation S_t is dependent on $\mathcal{F}_D^{p:p-k+1}$ and $\mathcal{F}_S^{p-1:p-k}$. In this way, those historical experiences from both tasks will be propagated along the time sequences by using TAM. That means, the dual-task interactions will go deeper along the sequence of states. As a general idea, the framework can be adapted to other dual-task applications and even multi-task learning. We give the formulation of multi-task learning in the supplemental materials. In this paper we simply set $k = 1$ in Eqn. 1, i.e., a short-term dependency.

3.2 Network Architecture

Overview The entire network architecture is shown in Fig. 2. We use the sophisticated ResNet [41] to encode the input image. The gray cubes from Res-2

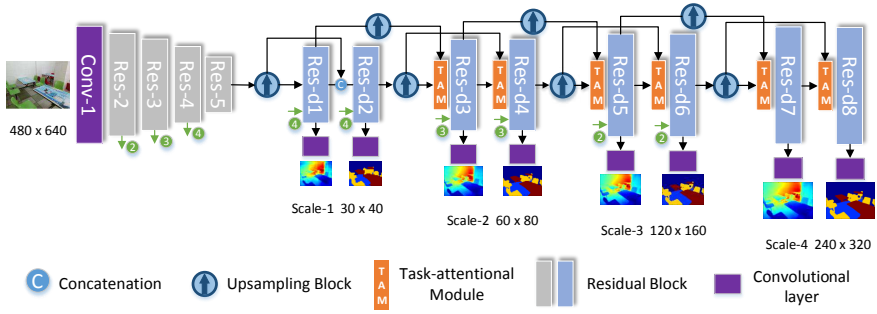

Fig. 2. The overview of our Task-Recursive Learning (TRL) network. The TRL network is an encoder-decoder architecture, which is composed of a series of residual blocks, upsampling blocks and Task-attentional Modules. The input RGB image is firstly fed into a ResNet to encode multi-level features, and then these features are fed into the task-recursive decoding process to estimate depth and semantic segmentation. In the decoder, the two tasks are alternately processed by adaptively evolving previous experiences of both tasks (i.e., the previous features of depth and segmentation), so as to boost and benefit for each other during the learning process. To estimate the current task state, the previous features of the two tasks are fed into a TAM to enhance the common information. To better refine the predicted details, we progressively execute the two tasks in a coarse-to-fine scale space.

to Res-5 are multi-scale response maps extracted from ResNet. The next decoding process is designed to solve the dual tasks based on the task-recursive idea. The decoder is composed of upsampling blocks, task-attentional modules and residual-blocks. The upsampling blocks upscale the convolutional features to required scales for pixel-level prediction. The detailed architecture will be introduced in the following subsection. For the pixel-level prediction, we introduce residual-blocks (blue cubes) to decode the previous features, which are the mirror type of the corresponding ones in the encoder but only have two bottlenecks in each residual block. The Res-d1, Res-d3, Res-d5 and Res-d7 focus on depth estimation, while the rest ones focus on semantic segmentation. The TAM is designed to perform the interaction of two tasks. During the interaction, the previous information will be selectively enhanced to adapt to the current task. For example, the TAM before Res-d5 receives inputs from two sources: one is the features upsampled from Res-d4 with segmentation information, and the other is the features upsampled from Res-d3 with depth information. During the interaction, the information of two inputs will be selectively enhanced to propagate to the next task. As the interaction times increase, the results of the two tasks are progressively refined in a mutual-boosting scheme. Another import strategy is taking a coarse-to-fine process to progressively reconstruct details and produce fine-grained predictions of high resolution. Concretely, we concatenate the different-scale features of encoder to the corresponding residual block, as indicated by the green arrows. The upsampling block and the task-attentional module will be described in the following subsections.

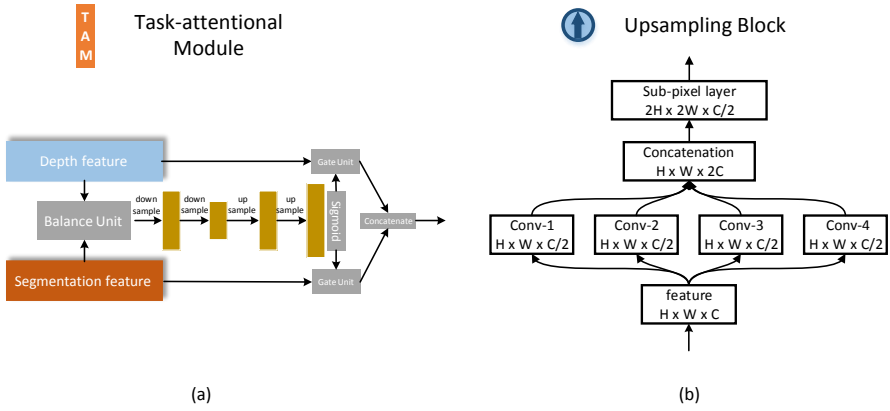

Fig. 3. The overview of our Upsampling-block and Task-attentional module.

Task-attentional module As discussed in the Section 1, semantic segmentation and depth estimation results of a scene have many common patterns, e.g., they can both reveal the object edges, boundaries or layouts. To better mine and utilize the common information, we design a task-attentional module to enhance the correlated information of the two tasks. As illustrated in Fig. 2, the TAM is used before each residual block and takes depth/segmentation features from previous residual blocks as inputs. The designed TAM are presented in Fig 3(a). The input depth/segmentation features are firstly fed into a balance unit to balance the contribution of the features of two sources. If we use f_d and $f_s \in \mathrm{R}^{H\times W\times C}$ to denote the received depth and segmentation features respectively, the balance unit can be formulated as:

$$\begin{aligned} B &= \mathrm{Sigmoid}(\Psi_1(\mathrm{concat}(f_d, f_s), \Theta_1)), \\ f_b &= \Psi_2(\mathrm{concat}(B \cdot f_d, (1-B) \cdot f_s), \Theta_2), \end{aligned} \tag{2}$$

where Ψ_1 and Ψ_2 are two convolutional layers with parameters Θ_1 and Θ_2, respectively. $B \in \mathrm{R}^{H\times W\times C}$ is the learnt balancing tensor, and $f_b \in \mathrm{R}^{H\times W\times C}$ is the balanced output of the balance unit. In this way, f_b combines the balanced information from the two sources. Next, the balanced output will be fed into a series of conv-deconvolutional layers, as illustrated by the yellow cubs in Fig 3(a). Such a mechanism is designed to get different spatial attentions by using the receptive field variation, as demonstrated in the residual attention [42]. After a Sigmoid transformation, we get an attentional map $\mathrm{M} \in \mathrm{R}^{H\times W\times C}$, which is expected to have higher responses on the common patterns. Finally, the attentional tensor M is used to generate the gated depth/segmentation features, formally,

$$\begin{aligned} f_d^g &= (1 + \mathrm{M}) \cdot f_d, \\ f_s^g &= (1 + \mathrm{M}) \cdot f_s. \end{aligned} \tag{3}$$

Thus the feature f_d and f_s may be enhanced through the learned attentional map M. The gated features f_d^g and f_s^g are further fused by concatenation followed

by one convolutional layer. The output of TAM is denoted as $f_{\text{TAM}} \in \mathrm{R}^{H\times W\times C}$. The task-attentional module can benefit our task-recursive learning method as experimentally analysed in Section 4.2.

Upsampling blocks The upsampling blocks are designed to match the scale variations during the task-recursive learning. The architecture of upsampling block is shown in Fig. 3(b). The features with size of $H \times W \times C$ are firstly fed into four parallel convolutional layers with different receptive fields (i.e., conv-1 to conv-4 in Fig. 3). These four convolutional layers are designed to capture different local structures. Then the responses produced from the four convolutional layers are concatenated to a tensor feature with size of $H \times W \times 2C$. Finally, the sub-pixel operation in [43] is applied to spatially upscale the feature. Formally, given a tensor feature T and a coordinate $[h, w, c]$, the sub-pixel operator can be defined as:

$$\mathcal{P}(T_{h,w,c}) = T_{\lfloor h/r\rfloor,\lfloor w/r\rfloor,c\cdot r\cdot \mathrm{mod}(w,r)+c\cdot \mathrm{mod}(h,r)}, \tag{4}$$

where r is the scale factor. After such sub-pixel operation, the output of one upsampling block is the feature of size $2H \times 2W \times C/2$, when we set $r = 2$. The upsampling blocks are more effective than the general deconvolution, as verified in the experiments in Section 4.2.

3.3 Training Loss

We impose the supervised loss constraint on each scale to obtain multi-scale predictions. For depth estimation, we use inverse Huber loss defined in [7] as the loss function, which can be formulated as:

$$\mathcal{L}^D(d_i) = \begin{cases} |d_i|, & |d_i| \le c, \\ \frac{d_i^2+c^2}{2c}, & |d_i| > c, \end{cases} \tag{5}$$

where d_i is the difference between prediction and ground truth at each pixel i, and c is a threshold with $c = \frac{1}{5}\max(d_i)$ as default. Such a loss function can provide more obvious gradients at the locations where the depth difference is low, and thus can help to better train the network. The loss function for semantic segmentation is a cross-entropy loss, denoted as $\mathcal{L}^S$. For a better optimization of our proposed dual-task network, we use the strategy proposed in [22] to balance the two tasks. Suppose the network predicts N pairs (w.r.t. N scales) of depth maps and semantic segmentation maps, the total loss function can be defined as:

$$\mathcal{L}(\Theta, \sigma_1, \sigma_2) = \frac{1}{\sigma_1^2}\sum_{n=1}^{N}\mathcal{L}_n^D + \frac{1}{\sigma_2^2}\sum_{n=1}^{N}\mathcal{L}_n^S + \log(\sigma_1^2) + \log(\sigma_2^2), \tag{6}$$

where Θ is the parameter of network, σ_1 and σ_2 are the balancing weights to the two tasks. Please note that the balancing weights are also optimized as

parameters during training. In practice, to avoid a potential division by zero, we redefine $\delta = \log \sigma^2$. Thus the total loss can be rewritten as:

$$\mathcal{L}(W, \delta_1, \delta_2) = \exp(-\delta_1) \sum_{n=1}^{N} \mathcal{L}_n^D + \exp(-\delta_2) \sum_{n=1}^{N} \mathcal{L}_n^S + \delta_1 + \delta_2. \tag{7}$$

4 Experiments

4.1 Experimental Settings

Dataset: We evaluate the effectiveness of our proposed method on NYU Depth V2 [1] and SUN RGBD [44] datasets. The NYU Depth v2 dataset [1] consists of RGB-D images of 464 indoor scenes. There are 1449 images with semantic labels, 795 of them are used for training and the remaning 654 images for testing. We randomly select 4k images of the raw data from official training scenes. These 4k images have the corresponding depth maps but no semantic labels. Before training our network, we first train a ResNet-50 based DeconvNet [11] for 40-class semantic segmentation using the given 795 images. Then we use the predictions of the trained DeconvNet on the 4k images as coarse semantic labels to train our network. Finally we fine-tune the network on the 795 images of standard training split. The SUN RGBD dataset [44] contains 10355 RGB-D images with semantic labels of which 5285 for training and 5050 for testing. We use the 5285 images with depth and semantic labels to train our network, and the 5050 images for evaluation. The semantic labels are divided into 37 classes. Following the settings in [7, 24, 6, 32], we use the same data augmentation strategies including cropping, scaling, flipping and rotating, to increase the diversity of data. As the largest outputs are half size of the input images, we upsample the predicted segmentation results and depth maps to the original size for comparison.

Implementation Details: We implement the proposed model using Pytorch on a single Nvidia P40 GPU. We build our network based on ResNet-18, ResNet-50 and ResNet-101, and each model is pre-trained on the ImageNet classification task [45]. ReLU activating function and Batch normalization are applied behind every convolutional layers, except for the final convolutional layers before the predictions. In the upsampling blocks, we set conv-1, conv-2, conv-3 and conv-4 with 1×1, 3×3, 5×5 and 7×7 kernel sizes, respectively. Note that we use 3×3 convolution with dilation=2 to efficiently get a 7×7 receptive field. For the parameters of training loss, we simply use initial values of $\delta_1 = \delta_2 = 0.5$ of Eqn. 7 for all scenes, and find that different initial values have no large effects on the performance. Initial learning rate is set to 10^{-5} for the pre-trained convolution layers and 0.01 for the other layers. For NYU Depth v2 dataset, we train our model on 4k unique images with coarse semantic labels and depth ground truth in 40K batch iterations, and then fine-tune the model with a learning rate of 0.001 on 795 images with depth and segmentation ground truth in 10K batch iterations. For the SUN-RGBD dataset, we train our model with 50K batch iterations on the initial learning rates, and fine-tune the non-pretrained layers for

30K batch iterations with a learning rate of 0.001. The momentum and weight decay are set to 0.9 and 0.0005 respectively, and the network is trained using SGD with batch size of 16. As there are many missing values in the depth ground truth maps, following the literatures [7, 24], we mask out the pixels that have missing depths both in the training and testing phases.

Metrics: Similar to the previous works [7, 24, 6], we evaluate our depth prediction results with the following metrics:

- average relative error (rel): $\frac{1}{n}\sum_i \frac{|\widetilde{x_i}-x_i|}{x_i}$;
- root mean squared error (rms): $\sqrt{\frac{1}{n}\sum_i(\widetilde{x_i}-x_i)^2}$;
- root mean squared error in log space (rms(log)): $\sqrt{\frac{1}{n}\sum_i(\log\widetilde{x_i}-\log x_i)^2}$;
- accuracy with threshold (δ): % of $\widetilde{x_i}$ s.t. $\max(\frac{\widetilde{x_i}}{x_i}, \frac{x_i}{\widetilde{x_i}})=\delta$ $\delta = 1.25, 1.25^2, 1.25^3$;

where $\widetilde{x_i}$ is the predicted depth value at the pixel i, n is the number of valid pixels and x_i is the ground truth.

For the evaluation of semantic segmentation results, we follow the recent works [32, 27, 46] and use the common metrics including pixel accuracy (pixel-acc), mean accuracy (mean-acc) and mean intersection over union (mean-IoU).

4.2 Ablation Study

In this section, we conduct several experiments to evaluate the effectiveness of our proposed method. The concrete ablation studies are introduced in the following.

Analysis on tasks: We first analyse the benefit of jointly predicting depth and segmentation of one image. The experiments use the same network architecture as our ResNet-18 based network and are trained on NYU Depth v2 and SUN-RGBD datasets for depth estimation and segmentation respectively. As illustrated in Table 1, our proposed TRL network obviously benefits for each other under the joint learning of depth estimation and semantic segmentation. For NYU Depth v2 dataset, compared to the gain on depth estimation, semantic segmentation has a larger gain after the dual-task learning, i.e., the improvement about 4.1% on mean class accuracy and 3.0% on IoU. One possible reason should be more data of 4k depth images than semantic labels of 795 images. In contrast, for SUN-RGBD dataset, all training samples are with depth and semantic ground truth, i.e., the training samples for both tasks are balanced. We can observe that the performance on both tasks can be promoted for each other under the framework of proposed task-recursive learning.

Architectures and baselines: We conduct experiments to analyse the effect of different network architectures. We set the baseline network with the same encoder but two parallel decoders. Each decoder corresponds to one task, which contains four residual blocks using the same type to the original TRL network decoder. To softly share the parameters and interact the two tasks, similar to the method in [19], we use the cross-stitch unit to fuse features at each scale. To evaluate the effectiveness of the task-attentional module, further, we perform

Table 1. Joint task learning v.s. single task learning on NYU Depth V2 and SUN-RGBD datasets.

	NYU-D				SUN-RGBD			
Metric	rms	rel	mean-acc	IoU	rms	rel	mean-acc	IoU
Depth only	0.547	0.172	-	-	0.517	0.163	-	-
Segmentation only	-	-	51.2	42.0	-	-	54.1	43.5
TRL-jointly	**0.510**	**0.156**	**55.3**	**45.0**	**0.468**	**0.140**	**56.3**	**46.3**

Table 2. Comparisons of different network architectures and baselines on NYU Depth v2 dataset.

Method	rms	rel	mean-acc	IoU
Baseline-I	0.545	0.171	53.5	43.2
TRL w/o TAM	0.526	0.153	54.0	43.6
TRL w/o exp-TAM	0.540	0.167	52.5	42.2
TRL w/o gate unit	0.515	0.160	55.0	44.7
TRL scale-1	0.597	0.202	50.1	40.3
TRL scale-2	0.572	0.198	51.9	41.0
TRL scale-3	0.541	0.166	53.2	43.8
TRL-ResNet18	0.510	0.156	55.3	45.0
TRL-ResNet50	0.501	0.144	56.3	46.4
TRL-ResNet101	**0.492**	**0.138**	**56.9**	**46.8**

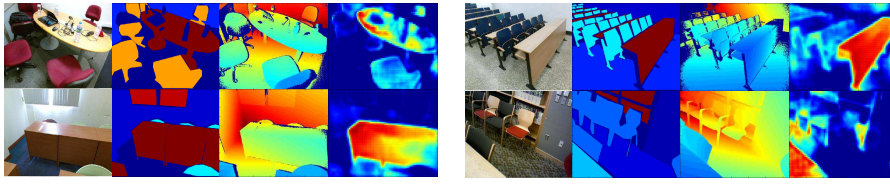

Fig. 4. Visual exhibition of the learned attentional maps. (a) input image; (b) segmentation ground truth; (c) depth ground truth; (d) learned attentional map. We can find that the attentional maps give high attention to objects, edges and boundaries which are very salient in both ground truth maps, i.e., more attention to the useful common information.

an experiment without TAMs. To verify the importance of historical experience at previous stages, we also train a TRL network without any earlier experience (i.e., not considering the TAMs and the features from previous residual blocks). Besides, we also evaluate the prediction ability of other three scales (from scale-1 to scale-3) to show the effectiveness of the coarse-to-fine mechanism. All these experimental models take ResNet-18 as infrastructure. Externally, we also train ResNet-50 and ResNet-101 based TRL networks to analyse the effect of deeper encoding networks.

As reported in Table 2, our proposed TRL network signaficantly performs better than the baseline on both tasks. Compared with the TRL network without TAMs, TRL can obtain a superior performance on both tasks. It indicates

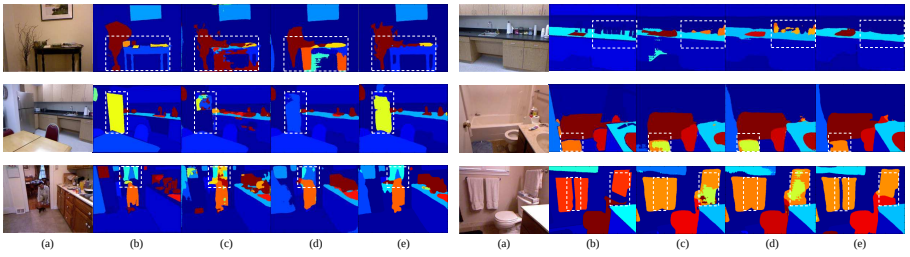

Fig. 5. Visual comparisons between TRL and baselines on NYU Depth V2 and SUN RGBD. (a) input image; (b) ground truth; (c) results of baseline; (d) results of TRL w/o TAMs; (e) results of the TRL network. It can be observed that the predictions results of our proposed TRL contain less errors and suffer less class ambiguity.

that TAMs can potentially take some common patterns of the two tasks to promote the performance. For this, we also visually exhibit the learned attentional map M from the TAMs. As observed in Fig. 4, the attentional maps have higher attention to objects, edges and boundaries, which are very obvious according to both ground truth maps. These features commonly exist in the two tasks, and thus can make TAMs capture such common information to promote both tasks. For the case without the historical experience mechanism, i.e., TRL w/o exp-TAMs, the original TRL can obtain an accumulative gain of 21.4% on the two tasks, which demonstrates that the experience mechanism is also crucial for the task-recursive learning process. In the cast that TAM has no gate unit, i.e., TRL w/o gate unit, the resulting accuracies are slightly decreased. When the scale increases, i.e., the coarse-to-fine manner, the performances are gradually improved on both tasks. An obvious reason is that details can be better reconstructed in those fine scale space. Further, when more sophisticated and deeper encoders are employed, ResNet-50 and ResNet-101, the proposed TRL network can improve the performance, which can be easily understood as the same observations in other literatures.

For a visual analysis, we show some prediction results of baselines and TRL in Fig. 5. From the figure, we can observe that the segmentation results of the two baselines suffer obvious classification error, especially as shown in the white bounding boxes. In contrast, the prediction results of TRL suffer less class ambiguity and are more reasonable visually. More ablation study and visual results can be found in our supplementary material.

4.3 Comparisons with the state-of-the-art methods

In this section we compare our method with several state-of-the-art approaches on both tasks. The experiments are conducted on NYU Depth V2 and SUN-RGBD datasets, which will be discussed below.

Depth estimation: We compare our depth estimation performance on NYU depth V2 dataset, and summarize the results in Table 3. As observed from this table, our TRL network with ResNet-50 achieves the best performance on the

Table 3. Comparisons with the state-of-the-art depth estimation approaches on NYU Depth V2 Dataset.

Method	rms	rel	rms(log)	δ_1	δ_2	δ_3
Li [26]	0.821	0.232	-	0.621	0.886	0.968
Liu [25]	0.824	0.230	-	0.614	0.883	0.971
Wang [21]	0.745	0.220	0.262	0.605	0.890	0.970
Eigen [5]	0.877	0.214	0.285	0.611	0.887	0.971
Roy [47]	0.744	0.187	-	-	-	-
Eigen [24]	0.641	0.158	0.214	0.769	0.950	0.988
Cao [48]	0.615	0.148	-	0.800	0.956	0.988
Xu-4.7k [6]	0.613	0.143	-	0.789	0.946	0.984
Xu-95k [6]	0.586	**0.121**	-	0.811	0.954	0.987
Laina [7]	0.573	0.127	0.194	0.811	0.953	0.988
TRL-ResNet18	0.510	0.156	0.187	0.804	0.951	0.990
TRL-ResNet50	**0.501**	0.144	**0.181**	**0.815**	**0.962**	**0.992**

rms, rms(log) and the δ-accuracy metrics, while this version with ResNet-18 also obtains satisfactory results. Compared with the recent method [7], our TRL is slightly inferior in the rel metric, but significantly superior in other metrics, where a total 7.67% relative gain is achieved. It is worth noting that the method in literature [7] used a larger training set which contains 12k unique image and depth pairs, but our model uses only 4k unique images (less than 12k) and still gets a better performance. Compared with the method in [6], we have the same observation that our TRL is slightly poor in rel metric but has obviously better results in all other metrics. Please note that the method in [6] attempted to use more training images (95k) to promote the performance of depth estimation. Nevertheless, if the training data is reduced to 4.7k, the accuracies have an obvious degradation for the method in [6]. In contrast, under the nearly equal size of training data, our TRL can still achieve the best performance in most metrics.

In addition, to provide a visual observation, we show some visual comparison examples in Fig. 6. The prediction results of the methods in [24, 6] usually have much noise, especially at the object boundaries, curtains, sofa and bed. On the contrary, our predictions have less noise and better match the geometry of the scenes. Therefore, these experimental results can demonstrate that our proposed approach is more effective than the state-of-the-art method by borrowing semantic segmentation information.

RGBD Semantic segmentation: We compare our TRL method with the state-of-the-art approaches on NYU Depth V2 and SUN RGBD datasets. For NYU Depth V2 dataset, as summarized in Table 4, our TRL network with ResNet-50 achieve the best pixel accuracies, but is slightly poor in mean class accuracy metric than the method in [32] and mean IoU metric than the method in [53]. It may be attributed to the imperfect depth predictions. Actually, the methods in [32, 53] used the depth ground truth as the input, and carefully designed some depth-RGB feature fusion strategies to make the segmentation prediction better benefit from the depth ground truth. In contrast, our TRL

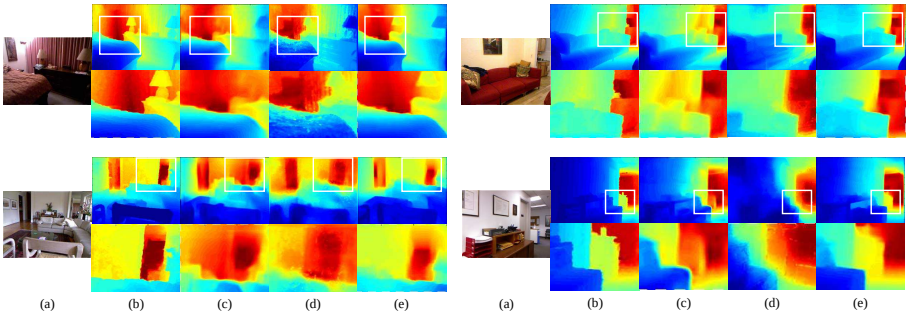

Fig. 6. Qualitative comparison with some state-of-the-art approaches on NYU depth v2 dataset. (a) input RGB image; (b) ground truth; (c) results of [24]; (d) results of [6]; (e) results of our TRL with ResNet-50. It can be easily observed that our predictions contain more details and less noise than these compared methods.

Table 4. Comparisons the state-of-the-art semantic segmentation methods on NYU Depth v2 dataset.

Method	data	pixel-acc	mean-acc	IoU
FCN [10]	RGB	60.0	49.2	29.2
Context [49]	RGB	70.0	53.6	40.6
Eigen *et al.* [24]	RGB	65.6	45.1	34.1
B-SegNet [27]	RGB	68.0	45.8	32.4
RefineNet-101 [46]	RGB	72.8	57.8	44.9
Deng *et al.* [50]	RGBD	63.8	-	31.5
He *et al.* [31]	RGBD	70.1	53.8	40.1
LSTM [51]	RGBD	-	49.4	-
Cheng *et al.* [32]	RGBD	71.9	**60.7**	45.9
3D-GNN [52]	RGBD	-	55.7	43.1
RDF-50 [53]	RGBD	74.8	60.4	**47.7**
TRL-ResNet18	RGB	74.3	55.5	45.0
TRL-ResNet50	RGB	**76.2**	56.3	46.4

method uses only RGB images as the input and conduct semantic segmentation based on estimated image depth, not depth ground truth. Although our TRL itself can obtain impressive depth estimation results, the depth estimation is still not as precise as ground truth, which usually results into more or less errors in the segmentation prediction process. Meanwhile, as the number of samples with semantic labels is limited in training for NYU Depth V2 dataset (795 images), the performance may be affected for our method.

For SUN-RGBD dataset, as reported in Table 5, our TRL network with ResNet-101 can reach the best performance in pixel-accuracy and mean IoU metrics. It is worth noting that the number of training samples with semantic labels is 5285 in SUN-RGBD, which is more than NYU Depth V2. Thus the performances on the two tasks are totally better than those on NYU Depth V2 for most methods, including our TRL network. Compared with the method in

Table 5. Comparison with the state-of-the-art semantic segmentation methods on SUN-RGBD dataset.

Method	data	pixel-acc	mean-acc	IoU
Context [49]	RGB	78.4	53.4	42.3
B-SegNet [27]	RGB	71.2	45.9	30.7
RefineNet-101 [46]	RGB	80.4	57.8	45.7
RefineNet-152 [46]	RGB	80.6	58.5	45.9
LSTM [51]	RGBD	-	48.1	-
Cheng *et al.* [32]	RGBD	-	58.0	-
CFN [54]	RGBD	-	-	48.1
3D-GNN [52]	RGBD	-	57.0	45.9
RDF-152 [53]	RGBD	81.5	**60.1**	47.7
TRL-ResNet18	RGB	81.1	56.3	46.3
TRL-ResNet50	RGB	83.6	58.2	49.6
TRL-ResNet101	RGB	**84.3**	58.9	**50.3**

[53], our TRL with ResNet-50 has a total 2.1% gain for all metrics, while the version with ResNet-101 obtains a total 4.3% gain. Note that, the method in [53] used the stronger ResNet-152 and more precise depth (i.e., ground truth) as inputs, while our TRL network uses only RGB images as the input. Overall, our TRL outperforms the current state-of-the-art methods in most evaluation metrics except the mean accuracy metric, in which ours is slightly poor but comparable.

5 Conclusions

In this paper, a novel end-to-end task-recursive learning framework had been proposed for jointly predicting depth map and semantic segmentation from one RGB image. The task-recursive learning network alternately refined the two tasks as a recursive sequence of time states. To better leverage the correlated and common patterns of depth and semantic segmentation, we also designed a task-attentional module. The module can adaptively mine the common information of the two tasks, encourage both interactive learning, and finally benefit for each other. Comprehensive benchmark evaluations demonstrated the superiority of our task-recursive network on jointly dealing with depth estimation and semantic segmentation. Meantime, we also reported some new state-of-the-art results on NYU-Depth v2 and SUN RGB-D datasets. In future, we will generalize the framework into the joint learning on more tasks.

6 Acknowledgement

The authors would like to thank the anonymous reviewers for their critical and constructive comments and suggestions. This work was supported by the National Natural Science Fund of China under Grant Nos. U1713208, 61472187, 61602244 and 61772276, the 973 Program No. 2014CB349303, the fundamental research funds for the central universities No. 30918011321, and Program for Changjiang Scholars.

References

1. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGBD images. In: ECCV. (2012) 746–760
2. Michels, J., Saxena, A., Ng, A.Y.: High speed obstacle avoidance using monocular vision and reinforcement learning. In: ICML. (2005) 593–600
3. Hadsell, R., Sermanet, P., Ben, J., Erkan, A., Scoffier, M., Kavukcuoglu, K., Muller, U., LeCun, Y.: Learning long-range vision for autonomous off-road driving. Journal of Field Robotics **26**(2) (2009) 120–144
4. Tateno, K., Tombari, F., Laina, I., Navab, N.: Cnn-slam: Real-time dense monocular slam with learned depth prediction. In: CVPR. Volume 2. (2017) 6565–6574
5. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: NIPS. (2014) 2366–2374
6. Xu, D., Ricci, E., Ouyang, W., Wang, X., Sebe, N.: Multi-scale continuous CRFs as sequential deep networks for monocular depth estimation. In: CVPR. Volume 1. (2017) 161–169
7. Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. In: 3DV. (2016) 239–248
8. Zhang, Z., Xu, C., Yang, J., Gao, J., Cui, Z.: Progressive hard-mining network for monocular depth estimation. IEEE Transactions on Image Processing **27**(8) (2018) 3691–3702
9. Zhang, Z., Xu, C., Yang, J., Tai, Y., Chen, L.: Deep hierarchical guidance and regularization learning for end-to-end depth estimation. Pattern Recognition (2018) 430–442
10. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(4) (2017) 640–651
11. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: ICCV. (2015) 1520–1528
12. Li, X., Jie, Z., Wang, W., Liu, C., Yang, J., Shen, X., Lin, Z., Chen, Q., Yan, S., Feng, J.: Foveanet: Perspective-aware urban scene parsing. ICCV (2017) 784–792
13. Wei, Y., Liang, X., Chen, Y., Jie, Z., Xiao, Y., Zhao, Y., Yan, S.: Learning to segment with image-level annotations. Pattern Recognition **59** (2016) 234–244
14. Wang, J., Wang, Z., Tao, D., See, S., Wang, G.: Learning common and specific features for rgb-d semantic segmentation with deconvolutional networks. In: ECCV. (2016) 664–679
15. Caruana, R.: Multitask learning. Machine Learning **28**(1) (1997) 41–75
16. Girshick, R.: Fast R-CNN. In: ICCV. (2015) 1440–1448
17. He, K., Gkioxari, G., Dollr, P., Girshick, R.: Mask R-CNN. IEEE International Conference on Computer Vision
18. Kim, S., Park, K., Sohn, K., Lin, S.: Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields. In: ECCV. (2016) 143–159
19. Misra, I., Shrivastava, A., Gupta, A., Hebert, M.: Cross-stitch networks for multi-task learning. In: CVPR. (2016) 3994–4003
20. Shi, J., Pollefeys, M.: Pulling things out of perspective. In: CVPR. (2014) 89–96
21. Wang, P., Shen, X., Lin, Z., Cohen, S.: Towards unified depth and semantic prediction from a single image. In: CVPR. (2015) 2800–2809
22. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. arXiv:1705.07115

23. Borst, J.P., Taatgen, N.A., Van, R.H.: The problem state: a cognitive bottleneck in multitasking. Journal of Experimental Psychology Learning Memory and Cognition **36**(2) (2010) 363
24. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: ICCV. (2015) 2650–2658
25. Liu, F., Shen, C., Lin, G., Reid, I.: Learning depth from single monocular images using deep convolutional neural fields. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**(10) (2016) 2024–2039
26. Li, B., Shen, C., Dai, Y., van den Hengel, A., He, M.: Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. In: CVPR. (2015) 1119–1127
27. Kendall, A., Badrinarayanan, V., , Cipolla, R.: Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. arXiv preprint arXiv:1511.02680 (2015)
28. Wei, Y., Xiao, H., Shi, H., Jie, Z., Feng, J., Huang, T.S.: Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In: CVPR. (2018) 7268–7277
29. Jin, X., Chen, Y., Jie, Z., Feng, J., Yan, S.: Multi-path feedback recurrent neural networks for scene parsing. In: AAAI. Volume 3. (2017) 8
30. Gupta, S., Girshick, R., Arbelez, P., Malik, J.: Learning rich features from rgb-d images for object detection and segmentation. In: ECCV. Volume 8695. (2014) 345–360
31. He, Y., Chiu, W.C., Keuper, M., Fritz, M.: Std2p: Rgbd semantic segmentation using spatio-temporal data-driven pooling. arXiv preprint arXiv:1604.02388 (2016)
32. Cheng, Y., Cai, R., Li, Z., Zhao, X., Huang, K.: Locality-sensitive deconvolution networks with gated fusion for rgb-d indoor semantic segmentation. In: CVPR. Volume 3. (2017) 1475–1483
33. Amit, Y., Fink, M., Srebro, N., Ullman, S.: Uncovering shared structures in multiclass classification. In: Machine Learning, Proceedings of the Twenty-Fourth International Conference. (2007) 17–24
34. Evgeniou, T., Pontil, M.: Regularized multi–task learning. In: Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. (2004) 109–117
35. Jalali, A., Ravikumar, P.D., Sanghavi, S., Chao, R.: A dirty model for multi-task learning. In: NIPS. (2010) 964–972
36. Razavian, A.S., Azizpour, H., Sullivan, J., Carlsson, S.: Cnn features off-the-shelf: An astounding baseline for recognition. In: CVPR Workshops. (2014) 512–519
37. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: NIPS. (2014) 3320–3328
38. Wang, X., Fouhey, D.F., Gupta, A.: Designing deep networks for surface normal estimation. In: CVPR. (2014) 539–547
39. Gebru, T., Hoffman, J., Li, F.F.: Fine-grained recognition in the wild: A multi-task domain adaptation approach. arXiv:1709.02476
40. Kokkinos, I.: Ubernet: Training a 'universal' convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In: CVPR. (2017) 5454–5463
41. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. (2016) 770–778
42. Wang, F., Jiang, M., Qian, C., Yang, S., Li, C., Zhang, H., Wang, X., Tang, X.: Residual attention network for image classification. In: CVPR. (2017) 6450–6458

43. Shi, W., Caballero, J., Huszr, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: CVPR. (2016) 1874–1883
44. Song, S., Lichtenberg, S.P., Xiao, J.: Sun RGB-D: A RGB-D scene understanding benchmark suite. In: CVPR. (2015) 567–576
45. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: CVPR. (2009) 248–255
46. Lin, G., Milan, A., Shen, C., Reid, I.: Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In: CVPR. Volume 1. (2017) 5168–5177
47. Roy, A., Todorovic, S.: Monocular depth estimation using neural regression forest. In: CVPR. (2016) 5506–5514
48. Cao, Y., Wu, Z., Shen, C.: Estimating depth from monocular images as classification using deep fully convolutional residual networks. IEEE Transactions on Circuits and Systems for Video Technology (2017)
49. Lin, G., Shen, C., Hengel, A.V.D., Reid, I.: Efficient piecewise training of deep structured models for semantic segmentation. In: CVPR. (2016) 3194–3203
50. Deng, Z., Todorovic, S., Latecki, L.J.: Semantic segmentation of rgbd images with mutex constraints. In: ICCV. (2015) 1733–1741
51. Li, Z., Gan, Y., Liang, X., Yu, Y., Cheng, H., Lin, L.: Lstm-cf: Unifying context modeling and fusion with lstms for rgb-d scene labeling. In: ECCV. (2016) 541–557
52. Xiaojuan, Q., Renjie, L., Jiaya, J., Sanya, F., Raquel, U.: 3d graph neural networks for RGBD semantic segmentation. In: ICCV. (2017) 5209–5218
53. Seong-Jin, P., Ki-Sang, H., Seungyong, L.: Rdfnet: Rgb-d multi-level residual feature fusion for indoor semantic segmentation. In: ICCV. (2017) 4990–4999
54. Di, L., Guangyong, C., Daniel, C.O., Pheng-Ann, H., Hui, H.: Cascaded feature network for semantic segmentation of rgb-d images. In: ICCV. (2017) 1320–1328