MDA-Net: Memorable Domain Adaptation Network for Monocular Depth Estimation

Jing Zhu[123]
jingzhu@nyu.edu
Yunxiao Shi[12]
yunxiao.shi@nyu.edu
Mengwei Ren[2]
mengwei.ren@nyu.edu
Yi Fang*[123]
yfang@nyu.edu

[1] NYU Multimedia and Visual Computing Lab, USA
[2] New York University, USA
[3] New York University Abu Dhabi, UAE

Abstract

Monocular depth estimation is a challenging task that aims to predict a corresponding depth map from a given single RGB image. Recent deep learning models have been proposed to predict the depth from the image by learning the alignment of deep features between the RGB image and the depth domains. In this paper, we present a novel approach, named Memorable Domain Adaptation Network (MDA-Net), to more effectively transfer domain features for monocular depth estimation by taking into account the common structure regularities (e.g., repetitive structure patterns, planar surfaces, symmetries) in domain adaptation. To this end, we introduce a new Structure-Oriented Memory (SOM) module to learn and memorize the structure-specific information between RGB image domain and the depth domain. More specifically, in the SOM module, we develop a Memorable Bank of Filters (MBF) unit to learn a set of filters that memorize the structure-aware image-depth residual pattern, and also an Attention Guided Controller (AGC) unit to control the filter selection in the MBF given image features queries. Given the query image feature, the trained SOM module is able to adaptively select the best customized filters for cross-domain feature transferring with an optimal structural disparity between image and depth. In summary, we focus on addressing this structure-specific domain adaption challenge by proposing a novel end-to-end multi-scale memorable network for monocular depth estimation. The experiments show that our MDA-Net demonstrates the superior performance compared to the existing supervised monocular depth estimation approaches on the challenging KITTI and NYU Depth V2 benchmarks.

1 Introduction

Depth estimation is an important component in many 3D computer vision tasks like shape analysis, shape generation, object detection and visual Simultaneous Localization and Mapping (visual SLAM) [3, 10, 11, 12, 13, 14, 21, 26, 31, 39, 43, 44]. Traditional approaches have made significant progress in binocular or multi-view depth estimation by taking advantage of geometry constraints of either spatial (i.e. stereo camera) or temporal (i.e. video

*:indicates corresponding author.

sequence) pairs. With the prevalence of deep convolutional neural networks, researchers have been trying to relax the constraints by tackling monocular depth estimation. Recent works ([4, 6, 17, 19, 32, 38]) have demonstrated promising results using regression-based deep learning models. Their models are trained by minimizing image-level losses with supervised signal on predicted results. Nevertheless, the cross-modality variance between the RGB image and the depth map still makes monocular depth prediction an ill-posed problem. Based on this observation, some researchers have considered solving the problem with additional feature-level structural constraints by minimizing the cross-modality residual complexity between image features and depth features. Most existing methods either consider the pixel-wise or structure-wise alignment in this regard. For instance, several architectures utilize the micro discrepancy loss as similarity measures such like sum of squared differences, correlation coefficients ([29]) and maximum mean discrepancy ([8, 27]) to align the RGB images features with depth features from pixel to pixel independently without considering the spatial dependencies. Another line of work has tried to apply the adversarial adaptation methods ([15, 18, 37]) in conjunction with task-specific losses that concentrate on macro spatial distribution similarity between the image features and depth ones. In this paper, we seek a way to address this domain adaption challenge on both pixel-wise discrepancies and the structure dependencies by extracting the structure-specific information between the two domains.

s In order to explore the pixel-wise discrepancies as well as the structure dependencies between the image features and depth features, we propose a memorable domain adaptation network, with an image-encoder-depth-decoder regression network backbone, and a specifically designed Structure-Oriented Memory (SOM) module coupled with a cross-modality residual complexity loss to minimize the gap between latent distribution of the image and depth map from both the pixel-level and structure-level. Given the observation that similar type of scenes (e.g. roadside scenes) often share common structural regularities (e.g. repetitive structure patterns, planar surfaces, symmetries), a set of filters could be trained to learn a specific structural image-depth residual patterns. Therefore, in our SOM module, we build a Memorable Bank of Filters (MBF) to store and learn the structure-aware filters, then we construct an Attention Guided Controller (AGC) to automatically select the appropriate filters (from the MBF) to capture the significant information from the given image features (generated by the image encoder) for the further depth estimation. Finally, the customized image features are fed into the depth decoder network to output the corresponding depth maps. Importantly, comparing to the direct alignment between the two domains features (e.g. direct applying L_1 loss between Z_i and Z_d), our introduced SOM module not only improves the fitting ability, but also reduces the training burden of the image encoder simultaneously. The experiments conducted on two well-known large scale benchmarks KITTI and NYU Depth V2, demonstrate that our proposed MDA-Net obtains the state-of-the-art performance on monocular depth estimation tasks. Moreover, the performance margin between model trained with SOM and the one trained with direct alignment, validate the effectiveness of our proposed SOM module. In summary, our contributions in this paper are as follows:

- We introduce memory strategies to address monocular depth estimation by designing a novel Structure-Oriented Memory (SOM) module with a Memorable Bank of Filters (MBF) and an Attention Guided Controller (AGC) for feature-level cross-modality domain adaptation.
- We propose a novel end-to-end deep learning model called MDA-Net which seamlessly integrates a front-end regression network with the SOM module that operates at

feature-level to substantially improve the depth prediction performance.

- We achieve state-of-the-art performance on two large scale benchmarks: KITTI and NYU Depth V2, which validates the effectiveness of the proposed method.

2 Related Works

Monocular depth estimation is a fundamental problem in computer vision which has widespread application in graphics, robotics and AR/VR. While previous works mainly tackle this using hand-crafted image features or probabilistic models such as Markov Random Fields (MRFs) ([34]), recent success of deep learning based methods ([4, 6, 17, 19, 32, 38]) have inspired researchers to use deep learning techniques to address the challenging depth estimation problem.

Supervised Methods A majority of works focus on supervised learning to use the learned features from CNNs to do accurate depth prediction. [5] first brought CNNs to depth regression task by integrating coarse and refined features with a two-stage network. The multi-task learning strategies were also applied in depth estimation to boost the performance. [22] utilized the semantic segmentation as objectness cues for depth estimation. Furthermore, [35] and [40] performed joint prediction of the pixel-level semantic labels as well as the depth. Surface normal information was also adopted in many recent works ([4, 30, 38, 42]). Besides, some research works also demonstrated the robustness of multi-scale feature fusion in pixel-level prediction tasks (e.g. semantic segmentation, depth estimation). [6] adopted the dilated convolution to enlarge the perceptive field without decreasing spatial resolution of the feature maps. In [2]'s work, inputs at different resolutions are utilized to build a multi-stream architecture. Instead of regression, there are also methods that discretize the depth range and transfer the regression problem to a classification problem. In the work of [6], the space-increasing discretization is proposed to reduce the over-strengthened loss for the large depth values.

Unsupervised/Semi-supervised Methods Another line of methods on monocular image depth prediction goes along the unsupervised/semi-supervised direction which mostly takes advantage of geometry constraints (e.g. epipolar geometry) on either spatial (between left-right pairs) or temporal (forward-backward) relationship. [7] proposed to estimate the depth map from a pair of stereo images by imposing the left-right consistency loss. [41] jointly learned a single view depth estimator and monocular odometry estimator using stereo video sequences, which enables the use of both spatial and temporal photometric warp constraints. Moreover, following the trend of adversarial learning, the generative adversarial networks (GANs) have been utilized in the depth estimation problem. [18] proposed an unsupervised domain adaptation strategy for adapting depth predictions from synthetic RGB-D pairs to natural scenes in the depth estimation task.

Cross-Modality Domain Adaptation In addition to the recent depth estimation methods, research works focused on the cross-modality domain adaption are also highly relevant to ours. The existence of cross modality, or domain shift, is commonly seen in real-world application, which is the consequence of data captured by different sensors (e.g. optical camera, LiDAR or stereo camera), or varying conditions (i.e. background). Most deep domain adaptation methods utilize a siamese architecture with two streams for source and target models respectively, and the network is trained with a discrepancy loss to minimize the pixel-wise shift between domains. [27] used maximum mean discrepancy together with a task-specific

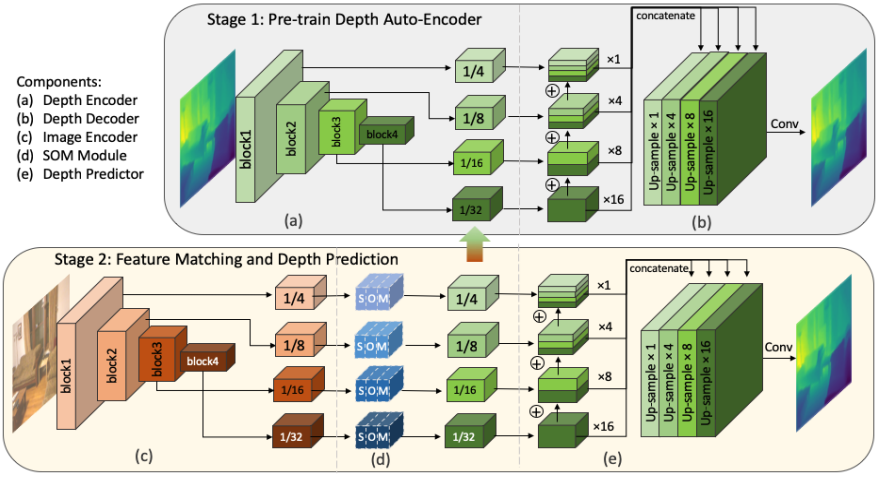

Figure 1: The network structure of Memorable Domain Adaptation Network (MDA-Net).

loss to adapt the source and target, while [36] proposed the deep correlation alignment algorithm to match the mean and covariance. [1] proposed to learn a dense representation using an auto-encoder. [28] trained the network with L_1 constrain in latent space to transfer feature from 2D to 3D in order to directly predict 3D point cloud from a single image. In our work, we aim to design a domain adaptive (SOM) module using memory mechanism, so that the image features can be automatically customized to obtain a better depth prediction.

3 Proposed Method

The monocular depth estimation problem can be defined as a nonlinear mapping $f : I \to Y$ from the RGB image I to the geometric depth map Y, which can be learned in a supervised fashion given a training set $X = \{I^t, Y^t\}_{t=1}^N$. To learn the mapping function, we propose MDA-Net as shown in Fig. 1, which is composed of a (pre-trained) depth auto-encoder, an image encoder and a depth predictor equipped with SOM module. All the components are trained into two stages. In the first stage, a series of 'target' depth features $\{Z_d^t\}_{t=1}^k \in R^k$ are learned by training a depth map auto-encoder (E_d, D_d). In the second stage, we train an image encoder E_i, SOM modules M_{id} and a depth predictor P_d to map the 2D image to the depth map in an end-to-end manner. Particularly, E_i encodes the RGB image to the 'source' image features $\{Z_i^t\}_{t=1}^k \in R^k$, which act as queries to obtain image-depth residual patterns from SOM module. The residual is then concatenated to the source feature to form a newly transferred feature set $\{Z_{id}^t\}_{t=1}^k \in R^k$ (which is expected to be aligned with the target feature $\{Z_d^t\}_{t=1}^k$ with supervision) is fed to the predictor P_d to estimate the output depth map. We will elaborate the network structures from two stages separately.

3.1 Stage 1: Depth Auto-Encoder

In order to learn a strong and robust prior over the depth map as a reference in the latent matching process, we train a depth auto-encoder (E_d, D_d) which takes a ground truth depth map $Y_d \in R^{M \times N}$ as input, and outputs a reconstructed depth map $\hat{Y}_d \in R^{M \times N}$. As shown in Figure 1 (stage 1), DenseNet-121 is utilized for constructing the depth encoder (Figure

1 (a)), in which four feature maps with cascading resolutions are extracted from different blocks (shallow to deep) for depth decoding. In order to make sure that the object contours as well as details are well preserved, we use a Feature Pyramid Network (FPN) to build the depth decoder, fusing multi-scale features in a pyramid structure. Specifically, as shown in Fig. 1 (b), four features with sizes $1/4$, $1/8$, $1/16$ and $1/32$ of the input are derived. Starting from the deepest feature, each feature map is first upsampled by a factor of 2, and element-wisely added to its following feature map. After the fusion process of the multi-scale feature maps, each of the newly generated feature maps is upsampled to size of $1/4$ the original input (or the size of the shallowest feature map), and concatenated together to form a feature volume. Finally, the output depth map is predicted via extra CNN layers on the concatenated feature volume. The FPN decoder is able to preserve details in the depth map decoding process.

3.2 Stage 2: Depth Prediction with SOM Module for Latent Space Adaptation

In the second stage, we aim to train the network in an end-to-end manner to effectively transfer the features derived from image encoder E_i from image domain to depth domain, as a strong prior over the ground truth depth, so as to better deduce the depth from the transferred prior. To this end, this stage contains three major components as shown in Fig. 1 (c), (d) and (e): the image encoder, the SOM module for latent space adaptation, and the depth predictor (E_i, M_{id}, P_d). Each component of the network will be explained below.

Image Encoder and Depth Predictor as Regression Backbone In order to make sure that the network derive both depth features and image features at the same scale, we design the encoder-decoder based backbone ((c) and (d) in Fig. 1) for stage 2 exactly the same as those of stage 1 but without weight sharing. Specifically, the structure of image encoder E_i ((c) in Fig. 1) is identical to that of depth encoder E_d ((e) in Fig. 1), and similarly for D_d ((b)) and P_d ((e)).

SOM Module for Latent Space Adaptation In the latent space, we propose an additional structure oriented memory module consisting of two collaborative units: a Memorable Bank of Filters (MBF) that stores a bank of learned filters to detect the cross-modality residual complexity between the depth feature and the image feature, and an Attention Guided Controller (AGC) which controls the interaction between the image feature with the MBF. The image feature as a specific query feature selects filters from MBF with an attention guided read controller, and the MBF is updated through a write controller that is naturally integrated into the back propagation to make the network can be trained end-to-end. The proposed SOM reading and writing process are as follows.

SOM Reading Different from reading by 'addressing' in general memory concept, the proposed SOM module is reading by 'attention', which means each memory slot is assigned with a weight, and the whole memory is merged per weights as reading output. As demonstrated in Fig. 2, given the query feature Z_i, in order to obtain weights for each memory slot, we build a LSTM-based read controller to learn the weights. Specifically, each filter from the memory slot $\{M_t\}_{t=1}^n$ is firstly convolved on the feature, and the intermediate outputs are denoted as $\{x_t\}_{t=1}^n$, where n is the memory size, and x_t is formulated as: $x_t = W_t * Z_i + b_t, M_t = (W_t, b_t)$, W_t is the kernel, b_t is the bias, and $*$ is the convolution operation. The intermediate outputs $\{x_t\}_{t=1}^n$ could be thought of as the 'unweighted/unbiased' output that takes each filter/memory slot equally. Then in order to further add weighted

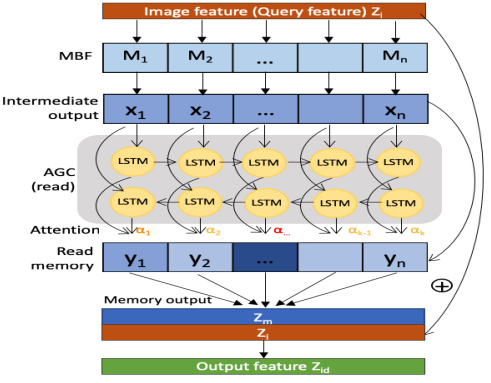

Figure 2: The SOM reading process (of a single SOM module).

attention on the result pool, a Bi-Directional Convolutional Long Short Term Memory is applied as the read controller on $\{x_t\}_{t=1}^{n}$ to explore the correlation within the pool, so as to aggregate the memory slots with strong attention. Particularly, read controller processes $\{x_t\}_{t=1}^{n}$ from two directions and computes the forward hidden sequence h_f by iterating the input from $t = 1$ to n, and the backward hidden sequence h_b by iterating the input from $t = n$ to 1. The forward/backward flow of the LSTM cell is formulated as below:

$$i_t = \sigma(W_{xi} * x_t + W_{hi} * h_{t-1} + W_{ci} \circ c_{t-1} + b_i), f_t = \sigma(W_{xf} * x_t + W_{hf} * h_{t-1} + W_{cf} \circ c_{t-1} + b_f),$$
$$c_t = f_t \circ c_{t-1} + i_t \circ \tanh(W_{xc} * x_t + W_{hc} * h_{t-1} + b_c),$$
$$o_t = \sigma(W_{xo} * x_t + W_{ho} * h_{t-1} + W_{co} \circ c_t + b_o), h_t = o_t \circ \tanh(c_t),$$

where h is the hidden sequence, σ is the logistic sigmoid function, $*$ is the convolution operator and $\circ$ denotes the Hadamard product. i_t, f_t, o_t, c_t represent input gate, forget gate, output gate, and cell activation vector respectively, and W_{hi} is the hidden-input gate matrix, while W_{xo} is the input-output gate matrix. The final attention sequence α is computed with regard to both h_f and h_b as follows: $\alpha_t = softmax(W_{h_f y} h_{f(t)} + W_{h_b y} h_{b(t)} + b_y)$, where $t = 1$ to n, and each y after softmax operation in the output sequence is associated with the weight for each memory slot (refer to α value in Fig. 2, the redder the color, the higher the attention), therefore $\sum_{i=1}^{k} \alpha_i = 1$. The memory output Z_m is a combination of the output sequence that focuses more on the slot with higher attention, while less on lower attention value: $Z_m = \sum_{t=1}^{n} y_t, y_t = \alpha_t x_t$. Finally, Z_m is concatenated with the query feature itself to reproduce a transferred feature Z_{id} that is supposed to match the distribution of the depth feature Z_d.

SOM Writing The proposed memory writer can be seamlessly integrated to network back propagation. The attention learned from the read controller will also operate in the memory writing process, and specifically, the slot with higher attention will be updated to a larger extent and vice versa. The update rule could be formulated (in a simplified form) as $W_t \leftarrow W_t + \alpha_t \eta \Delta_{W_t}$, where α_t is the attention for each slot, η is the learning rate, and Δ_{W_t} is the total gradient from both branches.

3.3 Learning objectives

We design multiple objectives to constrain the joint training of the network with details as follows.

Depth Estimation Objective The depth estimation objective poses constraints on the front-end pipeline of the single image depth estimation. In order to reduce the over-emphasized error on large depth values, we use the logarithm mean squared error ($\text{RMSE}_{\log}$)) loss to make the predictor focus more on closer objects which makes up the main portion in

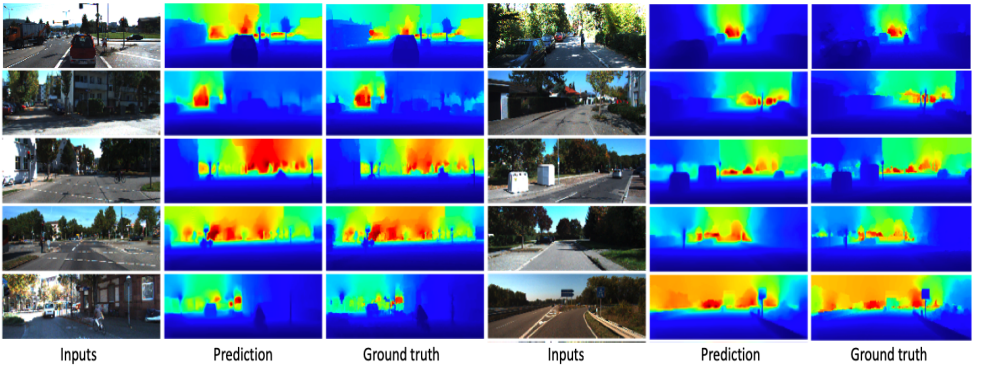

Figure 3: Results on KITTI validation set.

a depth map. The objective is formulated as $\mathcal{L}_{depth} = \sqrt{\frac{1}{N}\sum_{i\in N}||\log(d_i) - \log(d_i^*)||^2}$, where d is the ground truth depth map, while d^* is the predicted depth map.

Auto-Encoder Objective The objective for the depth auto-encoder is utilized in the first training stage. To make sure that the depth features and the image features are in the same scale with same constraints, we also applied the $\text{RMSE}_{\log}$ on the auto-encoder as $\mathcal{L}_{AE} = \sqrt{\frac{1}{N}\sum_{i\in N}||\log(d_i) - \log(\hat{d}_i)||^2}$, where d is the ground truth depth map, while $\hat{d}$ is the reconstructed depth map.

Cross-Modality Residual Complexity Objective The latent adaptation objective is applied to constrain the SOM module to minimize feature distribution discrepancies. We use L_1 loss between the 'target' depth features (pretrained from stage 1) and the SOM transferred image features. The objective is a sum of feature alignment losses at different levels as $\mathcal{L}_{CMRC} = \sum_k ||Z_{id}^k - Z_d^k||_1$, where k is the number of features involved in latent matching.

Gradient and Surface Normal Constraints To further strengthen the network by pulling out the model from local minima, we added extra constraints on the predicted depth map including the gradient loss and the surface normal loss to finetune the training. The gradient loss is defined as $\mathcal{L}_{gradient} = \frac{1}{N}\sum_{i=1}^{N}||\nabla d_i - \nabla d_i^*||_1$, and specifically, we adopt Sobel filter to calculate the gradient both vertically and horizontally; ∇d is the image gradient of the ground truth depth map, while ∇d^* is the image gradient of the predicted depth map. The surface normal loss is defined as the similarity between the surface normal of the ground truth depth map with the predicted depth map as $\mathcal{L}_{normal} = \frac{1}{N}\sum_{i=1}^{N}(1 - \frac{<\nabla d_i, \nabla d_i^*>}{||\nabla d_i||_2||\nabla d_i^*||_2})$, formulated with the corresponding gradient.

In total, the training objectives are summarized as follows: (1) In training stage 1, the total loss is: $\mathcal{L}_{S_1} = \mathcal{L}_{AE}$; (2) In training stage 2, the total loss is : $\mathcal{L}_{S_2} = \lambda_{depth}\mathcal{L}_{depth} + \lambda_{CMRC}\mathcal{L}_{CMRC} + \lambda_{gradient}\mathcal{L}_{gradient} + \lambda_{normal}\mathcal{L}_{normal}$, where λ is the weight for each objective.

4 Experiments

In this section, we present our experiments on two large-scale datasets by introducing the implementation details, benchmark performance and ablation studies validating the effectiveness of the proposed approach.

Implementation Details The proposed method is implemented using the TensorFlow and runs on a single NVIDIA TITAN X GPU with 12 GB memory. The encoder-decoder

Table 1: Performance on KITTI validation set. All scores are evaluated on Eigen split ([4]).

Method	Error (lower is better)				Accuracy (higher is better)		
	Abs Rel	Sq Rel	RMSE	$\text{RMSE}_{\log}$	$\sigma < 1.25$	$\sigma < 1.25^2$	$\sigma < 1.25^3$
Saxena [32]	0.280	3.012	8.734	0.361	0.601	0.820	0.926
Liu [23]	0.217	1.841	6.986	0.289	0.647	0.882	0.961
Zhou [40]	0.208	1.768	6.858	-	0.678	0.885	0.957
Eigen [4]	0.190	1.515	7.156	0.270	0.692	0.899	0.967
Garg [9]	0.177	1.169	5.285	-	0.727	0.896	0.962
Kundu [18]	0.167	1.257	5.578	0.237	0.771	0.922	0.971
Zhan [41]	0.135	1.132	5.585	0.229	0.820	0.933	0.971
Godard [9]	0.114	0.898	4.935	0.206	0.861	0.949	0.976
Kuznietsov [15]	0.113	0.741	4.621	0.189	0.862	0.960	0.986
Ours	**0.097**	**0.398**	**3.007**	**0.133**	**0.913**	**0.985**	**0.997**

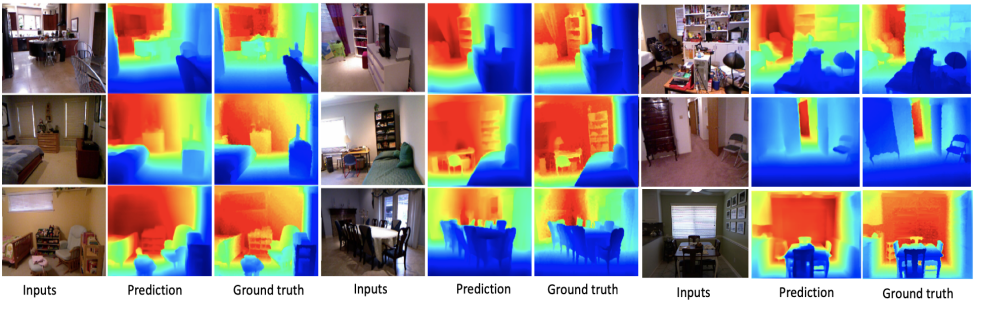

Figure 4: Examples of predicted depth maps on NYU V2 Depth dataset.

structure from both stage 1 and stage 2 are identical but without weight sharing. The depth auto-encoder is trained from scratch, while the image encoder is initialized with ImageNet ([33]) pre-trained parameters. For multi-scale feature fusion, we consider four levels of feature maps which are derived from different blocks of the DenseNet-121 backbone with the feature map sizes 1/4, 1/8, 1/16 and 1/32 of the input images. For instance, in NYU Depth V2 dataset, with the input resolution 480×640, four feature maps with cascading sizes 120×160, 60×80, 30×40, 15×20 are extracted. The network is trained with initial learning rate 0.001, and decreased every 10 epochs. The weight decay and momentum set to 10^{-6} and 0.9 respectively. We used the Adam optimizer and batch normalization during training, with normalization decay 0.97. We set the weights for each objective as $\lambda_{depth} = 1$, $\lambda_{gradient} = 1$, $\lambda_{norm} = 1$, and $\lambda_{CMRC} = 2$. The gradient loss is added after 4k steps of training, and the surface normal loss is added after 8k steps of training.

Results on KITTI Dataset (Eigen split) The KITTI dataset is a large scale dataset for autonomous driving, which contains depth images captured with LiDAR sensor mounted on a driving vehicle. In our experiment, to compare the results at the same level, we follow the experimental protocol proposed by [4], in which around 22600 images (resolution 384×1280) from 32 scenes are utilized as training data, and around 800 images from 29 scenes are used for validation. Following the previous works, the depth value of the RGB image is scaled to 0-80m. During training, the depth maps are down-scaled to resolution 192×640, and up-sampled to the original size in evaluation process. Table 1 shows the comparison with the state-of-the-art methods on KITTI dataset. We compared with state-of-the-art methods ([5, 7, 9, 18, 19, 24, 34, 41, 42]). Particularly, the methods proposed by [5, 18, 24, 34, 42] only employ monocular images in both training and testing, while approaches in [7, 9, 19, 41] are unsupervised methods that use stereo images in training and apply single image during testing. The proposed method outperforms all these methods by a large margin, and Fig. 3 displays a few visualized prediction results on examples randomly chosen from the validation dataset.

Table 2: Performance comparison on NYU Depth V2.

Method	Error			Accuracy		
	Rel	RMSE	$\log_{10}$	$\sigma < 1.25$	$\sigma < 1.25^2$	$\sigma < 1.25^3$
Saxena [34]	0.349	1.214	-	0.447	0.745	0.897
Karsch [16]	0.35	1.2	0.131	-	-	-
Liu [23]	0.335	1.06	0.127	-	-	-
Ladicky [18]	-	-	-	0.542	0.829	0.941
Zhuo [13]	0.305	1.04	-	0.525	0.838	0.962
Li [20]	0.232	0.821	0.094	0.621	0.886	0.968
Wang [38]	0.220	0.745	-	0.605	0.890	0.970
Xu [40]	0.214	0.792	0.091	0.643	0.902	0.977
Liu [25]	0.213	0.759	0.087	0.650	0.906	0.976
Roy [32]	0.187	0.744	-	-	-	-
Ours ($E_i + D_{pure}$)	0.231	0.828	0.095	0.631	0.889	0.968
Ours ($E_i + D_{FPN}$)	0.229	0.803	0.092	0.633	0.891	0.969
Ours ($E_i + D_{FPN} + align$)	0.148	0.627	0.075	0.802	0.944	0.986
Ours ($E_i + D_{FPN} + SOM$)	**0.136**	**0.604**	**0.067**	**0.814**	**0.959**	**0.990**

Results on NYU Depth V2 Dataset The NYU Depth V2 dataset contains 120K pairs of RGB-D (resolution 480×640) captured by Kinect. The dataset is manually selected and annotated into 1449 RGB-D pairs, in which 795 images are used for training, and the rest for validation. The depth value ranges from 0 to 10m. In the training process, the depth maps are down-scaled to resolution 120×160, and in testing/ evaluation, the predicted depth map is upsampled to the original resolution. Table 2 shows the comparison of the proposed method with state-of-the-art methods (official test split). We compare with both hand-crafted feature based approaches ([16, 34, 35]) and deep learning based ones ([20, 23, 25, 32, 38, 40, 45]). Fig. 4 shows examples of predicted depth maps on the NYU Depth V2 dataset.

Ablation Studies To further demonstrate the effectiveness of the proposed method, we conduct ablation studies from two aspects on NYU Depth V2 dataset. Firstly, we compare the performance of the depth estimation pipeline with different decoder structures: (1) The decoder that simply uses symmetric structure with the encoder that cascadingly upsample the feature map until the output size. (2) The decoder that takes four different feature maps from the encoder and fuses them in a pyramid fashion (as described in Section 3.1). The qualitative comparison are shown in Table 2 ($E_i + D_{pure}$ and $E_i + D_{FPN}$). As can be seen from the evaluation results, the decoder structure with pyramid multi-sacle feature fusion out-performs the one that only takes the latent feature as input by a large margin, especially in the $\delta_1 < 1.25$ metric. Therefore, it is obvious that the mixture of features from different levels are beneficial for the details compensation (i.e. contour, edges).

To validate the effectiveness of the proposed SOM module, we compare the performance of the proposed method with SOM settings against direct alignment and analyze the results. Firstly, we add the feature alignment loss for latent feature maps based on the $E_i + D_{FPN}$ structure to test the performance of direct feature alignment ($E_i + D_{FPN} + align$). The quantitative results of direct alignment rarely improved compared with the one that is trained without feature alignment loss, reflecting the limited capability of the encoder for feature adaptation. Then, we add the SOM module at feature level ($E_i + D_{FPN} + SOM$) and compare the results with the baseline structure that goes without memory. The large margin quantitative improvement in Table 2 implies that structure-specific feature alignment with memory mechanism (SOM) is superior to other approaches such as direct alignment.

5 Conclusion

In this paper, we developed a novel memory guided network named MDA-Net for monocular depth estimation, consisting of the encoder-decoder based structure, as well as the external

SOM module which is trained to learn and memorize the structure attentioned image-depth-residual pattern in cross-modality latent alignment. The proposed method achieves state-of-the-art performance on challenging large-scale benchmarks, and each component is validated to be effective in the ablation study.

6 Acknowledgement

The authors would like to highly acknowledge the research team from XMotors.ai for their insightful discussion and comments for this paper. We particularly thank Dr. Kuo-Chin Lien and Dr. Junli Gu for their valuable inputs to this research work.

References

[1] Michael Bloesch, Jan Czarnowski, Ronald Clark, Stefan Leutenegger, and Andrew J. Davison. Codeslam — learning a compact, optimisable representation for dense visual slam. In *CVPR*, June 2018.

[2] Pierre Buyssens, Abderrahim Elmoataz, and Olivier Lézoray. Multiscale convolutional neural networks for vision–based classification of cells. In *Asian Conference on Computer Vision*, pages 342–352, 2012.

[3] Guoxian Dai, Jin Xie, and Yi Fang. Siamese cnn-bilstm architecture for 3d shape representation learning. In *27th International Joint Conference on Artificial Intelligence, IJCAI 2018*, pages 670–676. International Joint Conferences on Artificial Intelligence, 2018.

[4] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *IEEE International Conference on Computer Vision*, pages 2650–2658, 2015.

[5] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *International Conference on Neural Information Processing Systems*, pages 2366–2374, 2014.

[6] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. 2018.

[7] Ravi Garg, Kumar B. G Vijay, Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. pages 740–756, 2016.

[8] Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. pages 2551–2559, 2015.

[9] Clement Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Computer Vision and Pattern Recognition*, pages 6602–6611, 2017.

[10] Zhizhong Han, Xiyang Wang, Yu-Shen Liu, and Matthias Zwicker. Multi-angle point cloud-vae: unsupervised feature learning for 3d point clouds from multiple angles by joint self-reconstruction and half-to-half prediction. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10441–10450. IEEE.

[11] Zhizhong Han, Zhenbao Liu, Chi-Man Vong, Yu-Shen Liu, Shuhui Bu, Junwei Han, and CL Philip Chen. Boscc: Bag of spatial context correlations for spatially enhanced 3d shape representation. *IEEE Transactions on Image Processing*, 26(8):3707–3720, 2017.

[12] Zhizhong Han, Xinhai Liu, Yu-Shen Liu, and Matthias Zwicker. Parts4feature: learning 3d global features from generally semantic parts in multiple views. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 766–773. AAAI Press, 2019.

[13] Zhizhong Han, Mingyang Shang, Xiyang Wang, Yu-Shen Liu, and Matthias Zwicker. Y2seq2seq: Cross-modal representation learning for 3d shape and text by joint reconstruction and prediction of view and word sequences. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 126–133, 2019.

[14] Zhizhong Han, Xiyang Wang, Chi-Man Vong, Yu-Shen Liu, Matthias Zwicker, and CL Chen. 3dviewgraph: learning global features for 3d shapes from a graph of unordered views with attention. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 758–765. AAAI Press, 2019.

[15] Judy Hoffman, Eric Tzeng, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. 30(31):4068–4076, 2015.

[16] Kevin Karsch, Ce Liu, and Sing Bing Kang. Depth extraction from video using nonparametric sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11):2144, 2012.

[17] Seungryong Kim, Kihong Park, Kwanghoon Sohn, and Stephen Lin. Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields. In *European Conference on Computer Vision*, pages 143–159, 2016.

[18] Jogendra Nath Kundu, Phani Krishna Uppala, Anuj Pahuja, and R. Venkatesh Babu. Adadepth: Unsupervised content congruent adaptation for depth estimation. 2018.

[19] Yevhen Kuznietsov, Jörg Stückler, and Bastian Leibe. Semi-supervised deep learning for monocular depth map prediction. pages 2215–2223, 2017.

[20] Bo Li, Chunhua Shen, Yuchao Dai, Anton Van Den Hengel, and Mingyi He. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1119–1127, 2015.

[21] X. Li, X. Yao, and Y. Fang. Building-a-nets: Robust building extraction from high-resolution remote sensing images with adversarial networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(10):3680–3687, 2018.

[22] Beyang Liu, Stephen Gould, and Daphne Koller. Single image depth estimation from predicted semantic labels. In *Computer Vision and Pattern Recognition*, pages 1253–1260, 2010.

[23] F. Liu, C. Shen, G. Lin, and I Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(10):2024–2039, 2016.

[24] Fayao Liu, Chunhua Shen, and Guosheng Lin. Deep convolutional neural fields for depth estimation from a single image. In *Computer Vision and Pattern Recognition*, pages 5162–5170, 2015.

[25] Miaomiao Liu, Mathieu Salzmann, and Xuming He. Discrete-continuous depth estimation from a single image. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723, 2014.

[26] Yu-Shen Liu, Yi Fang, and Karthik Ramani. Using least median of squares for structural superposition of flexible proteins. *BMC bioinformatics*, 10(1):29, 2009.

[27] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. pages 97–105, 2015.

[28] Priyanka Mandikal, Navaneet Murthy, Mayank Agarwal, and R. Venkatesh Babu. 3d-lmnet: Latent embedding matching for accurate and diverse 3d point cloud reconstruction from a single image. 2018.

[29] Andriy Myronenko and Xubo Song. Intensity-based image registration by minimizing residual complexity. *IEEE Transactions on Medical Imaging*, 29(11):1882, 2010.

[30] Xiaojuan Qi, Renjie Liao, Zhengzhe Liu, Raquel Urtasun, and Jiaya Jia. Geonet : Geometric neural network for joint depth and surface normal estimation. 2018.

[31] Mengwei Ren, Liang Niu, and Yi Fang. 3d-a-nets: 3d deep dense descriptor for volumetric shapes with adversarial networks. *arXiv preprint arXiv:1711.10108*, 2017.

[32] Anirban Roy and Sinisa Todorovic. Monocular depth estimation using neural regression forest. In *Proceedings of the IEEE CVPR*, pages 5506–5514, 2016.

[33] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, and Michael Bernstein. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[34] Ashutosh Saxena, Min Sun, and Andrew Y. Ng. *Make3D: Learning 3D Scene Structure from a Single Still Image*. IEEE Computer Society, 2009.

[35] Jianbo Shi and Marc Pollefeys. Pulling things out of perspective. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 89–96, 2014.

[36] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. pages 443–450, 2016.

[37] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. 2017.

[38] Peng Wang, Xiaohui Shen, Zhe Lin, and Scott Cohen. Towards unified depth and semantic prediction from a single image. In *Computer Vision and Pattern Recognition*, pages 2800–2809, 2015.

[39] Jin Xie, Meng Wang, and Yi Fang. Learned binary spectral shape descriptor for 3d shape correspondence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3309–3317, 2016.

[40] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. 2018.

[41] Huangying Zhan, Ravi Garg, Chamara Saroj Weerasekera, Kejie Li, Harsh Agarwal, and Ian Reid. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. 2018.

[42] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. pages 6612–6619, 2017.

[43] Jing Zhu and Yi Fang. Learning object-specific distance from a monocular image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3839–3848, 2019.

[44] Jing Zhu, John-Ross Rizzo, and Yi Fang. Learning domain-invariant feature for robust depth-image-based 3d shape retrieval. *Pattern Recognition Letters*, 119:24–33, 2019.

[45] Wei Zhuo, Mathieu Salzmann, Xuming He, and Miaomiao Liu. Indoor scene structure analysis for single image depth estimation. In *Computer Vision and Pattern Recognition*, pages 614–622, 2015.