Towards Comprehensive Monocular Depth Estimation: Multiple Heads are Better Than One
Shuwei Shao, Ran Li, Zhongcai Pei, Zhong Liu, Weihai Chen, Member, IEEE, Wentao Zhu, Xingming Wu, and Baochang Zhang

Abstract—Depth estimation attracts widespread attention in the computer vision community. However, it is still quite difficult to recover an accurate depth map using only one RGB image. We observe a phenomenon that existing methods tend to fail in different cases, caused by differences in network architecture, loss function and so on. In this work, we investigate into the phenomenon and propose to integrate the strengths of multiple weak depth predictor to build a comprehensive and accurate depth predictor, which is critical for many real-world applications, e.g., 3D reconstruction. Specifically, we construct multiple base (weak) depth predictors by utilizing different Transformer-based and convolutional neural network (CNN)-based architectures. Transformer establishes long-range correlation while CNN preserves local information ignored by Transformer due to the spatial inductive bias. Therefore, the coupling of Transformer and CNN contributes to the generation of complementary depth estimates, which are essential to achieve a comprehensive depth predictor. Then, we design mixers to learn from multiple weak predictions and adaptively fuse them into a strong depth estimate. The resultant model, which we refer to as Transformer-assisted depth ensembles (TEDepth). On the standard NYU-Depth-v2 and KITTI datasets, we thoroughly explore how the neural ensembles affect the depth estimation and demonstrate that our TEDepth achieves better results than previous state-of-the-art approaches. To validate the generalizability across cameras, we directly apply the models trained on NYU-Depth-v2 to the SUN RGB-D dataset without any fine-tuning, and the superior results emphasize its strong generalizability.

Index Terms—Monocular depth estimation, ensemble learning, deep learning, transformer, convolutional neural network.

Manuscript received 13 July 2022; accepted 21 November 2022. Date of publication 25 November 2022; date of current version 8 December 2023. This work was supported in part by the Macao Science and Technology Development Fund under Grant 0022/2019/AKP, in part by the Key Research and Development Program of Zhejiang Province under Grant 2020C01109, in part by Beijing Natural Science Foundation under Grant L223024, and in part by the National Natural Science Foundation of China under Grants 62076016, 61620106012, and 61573048. The Associate Editor coordinating the review of this manuscript and approving it for publication was Mr. Yuming Fang. (Shuwei Shao and Ran Li contributed equally to this work.) (Corresponding authors: Zhong Liu; Weihai Chen; Baochang Zhang.)

Shuwei Shao, Zhongcai Pei, and Weihai Chen are with the School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China, and also with Hangzhou Innovation Institute, Beihang University, Hangzhou 100191, China (e-mail: swshao@buaa.edu.cn; peizc@buaa.edu.cn; whchen@buaa.edu.cn).
Ran Li, Zhong Liu, Xingming Wu, and Baochang Zhang are with the School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China, and also with Hangzhou Innovation Institute, Beihang University, Hangzhou, China (e-mail: mllee1998@buaa.edu.cn; liuzhong@buaa.edu.cn; wxmbuaa@163.com; bczhang@buaa.edu.cn).
Wentao Zhu is with the Amazon, Seattle, WA 98109 USA (e-mail: wentaozl@uci.edu).

Fig. 1. Illustration of the failure cases in BTS [1] and TransDepth [2], which are highlighted by white and green boxes. By contrast, our TEDepth is capable of achieving more comprehensive and accurate depth estimates.

I. INTRODUCTION

Estimating depth map from single RGB image has been a longstanding research topic and proved to be a practical technology, with applications ranging from scene understanding [3], augmented reality [4] through to minimally invasive surgery [5]. Saxena et al. [6] proposed one of the first learning-based studies in this area, and significant advancements have been made followed by the explosion of deep learning [7], [8], [9], [10], [11], [12], [13]. Nonetheless, their depth estimation performance is still unsatisfactory.

The motivation for this work stems from the asymmetric depth error. To elaborate on it, we present several failure cases of the state-of-the-art monocular depth estimation methods on the NYU-Depth-v2 dataset [14] in Fig. 1. BTS and TransDepth fail on different regions of the same input image, as can be shown. For example, in the first column, BTS predicts the clear boundary of black box, which however fades in the estimate of TransDepth. BTS predicts the depth of chair armrest wrongly as the background depth, but TransDepth succeeds. To alleviate their failure cases, we intuitively design a depth estimator based on the strengths of each depth prediction, achieving a comprehensive and accurate depth estimation. Ensemble learning [15] is an efficacious machine learning paradigm, which combines predictions from individual models to produce...
superior results. Ensembles of neural networks, or neural ensembles for short, are now playing an essential role in ensemble learning owing to the dominance of deep learning. For instance, neural ensembles have been successfully applied to improve the classification accuracy [16], [17] and robustness of object detection [18]. Despite the steady progress in other fields of neural ensemble, the utility and impact of neural ensembles for monocular depth estimation still remains unknown and to be explored.

We delve into neural ensembles for monocular depth estimation and introduce a two-level ensemble scheme, TEDepth. An overview is demonstrated in Fig. 2, where the base (weak) predictors produce asymmetric bottom-level predictions, and the mixer integrates them into a comprehensive depth estimate at the top level. As indicated in recent studies [1], [20], the encoder is more pivotal than the decoder in an encoder-decoder based architecture for monocular depth estimation. Therefore, we focus on the encoder design and use the decoder of BTS for simplicity to construct multiple base predictors. We adopt different Transformer-based architectures e.g., [21], [22], [23] and CNN-based architectures e.g., [24], [25], [26] as encoders in parallel and prohibit the flow of information among them. It should be noted that the TEDepth is agnostic to the specific selection of network architectures and other choices are feasible. The principle of coupling Transformer and CNN lies in the fact that Transformer establishes the long-range correlation while CNN preserves the local information, which contributes to the generation of complementary depth estimates. Extensive experiments are conducted on challenging benchmarks including NYU-Depth-v2 [14], SUN RGB-D [27] and KITTI [28], which involve comparison to preceding state-of-the-art depth competitors and ensemble scheme competitors, number of base predictors, mixer types, diversity among base predictors and fusion locations. Besides, we verify its generalizability by applying the models trained on NYU-Depth-v2 to the SUN RGB-D without any finetuning.

To summarize, the contributions of this work are threefold:

- Taking inspiration from the asymmetric depth error, we introduce an efficacious framework, termed TEDepth, to achieve a comprehensive and accurate depth predictor by integrating the strengths of weak depth predictors, which provides a novel perspective in model design.
- We propose to combine CNN and Transformer to acquire asymmetric base predictors, which contribute to deliver complementary depth estimates. Besides, we perform an in-depth investigation regarding how the neural ensembles impact the monocular depth estimation.
- We conduct extensive experiments on three challenging datasets, including NYU-Depth-v2, SUN RGB-D and KITTI, demonstrating that our TEDepth outperforms previous methods by a significant margin and achieves a new state-of-the-art.

II. RELATED WORK

1) Monocular Depth Estimation: As one of the first learning-based studies, Saxena et al. [6] introduced a Markov Random Field to regress depth map from single RGB image. After that, Eigen et al. [29] proposed multi-scale networks and a scale
invariant loss. Since then, many follow-up studies have been developed to steadily improve the accuracy. Laina et al. [8] used a fully convolutional model involving four up-projection modules. Cao et al. [30] and Fu et al. [10] framed the depth estimation as a classification problem. Qi et al. [31] built a network named GeoNet to enforce geometrically consistent depth map and surface normal. Lee et al. [1] designed local planar guidance layers to recover depth map resolution. Yin et al. [32] suggested an idea of incorporating virtual surface normal into the loss calculation, which allowed exploiting geometry information. Yang et al. [2] leveraged Transformer to capture the long-distance dependencies. Vaishakh et al. [33] proposed to leverage information from coplanar pixels.

Our work significantly departs from the described methods. Instead, we draw inspiration from the asymmetric depth error and introduce Transformer-assisted depth ensembles to build a comprehensively effective depth estimator.

2) Visual Transformers: Transformer has received extensive attention as its effectiveness in natural language processing (NLP) tasks [34]. Recently, Dosovitskiy et al. [23] developed the first attempt to indicate the feasibility of Transformer architectures for image classification. Then, the visual transformers were further advanced in [21], [22], [35], for example and introduced to more generic tasks, such as semantic segmentation [36], depth estimation [2], weakly supervised object localization [37] and image generation [38]. We are inspired by the recent success of Transformer and propose to couple CNN with Transformer to acquire the diverse base predictors in TEDepth.

3) Ensemble Learning: It can be traced back to the 1990s, and is an efficacious machine learning paradigm that creates a collection of models and utilizes predictions from individual models to perform a superior estimation [15]. Bagging and Boosting [39] are one of the seminal works in the early stage of this field. The branch neural ensembles [40], particularly ensembles of CNNs [16], [17], hold an essential position in modern ensemble learning. A fundamental problem for neural ensembles is to ensure the diversity among base predictors. The sources of diversity involve using random initializations [41], different hyperparameters [42] and additional constraint loss terms [43].

In contrast to the prior ensemble methods, we explore the ensembles of visual transformers and utilize the complementarity of CNN and Transformer to enforce the diversity among base predictors. Despite the substantial research, neural ensembles for monocular depth estimation remain to be explored. The utility and impact are thoroughly evaluated in this work.

III. TRANSFORMER-ASSISTED DEPTH ENSEMBLES

A. Problem Setup

Let \( D = \{ (r_n, d_n) \}_1^N \) define a training set containing \( N \) pairs of RGB images and ground-truth depth maps. In the previous supervised learning paradigm, a DepthNet parameterized by \( \theta \) is trained to learn a mapping function

\[
\theta : r_n \rightarrow \hat{d}_n,
\]

which converts a RGB image \( r_n \) into the corresponding depth map \( \hat{d}_n \). Unlike prior works, we propose to (1) learn a collection of asymmetric mapping functions \( \{f_\theta^k\}_k^K \) through \( K \) diverse base predictors, where \( K \) denotes the number of base predictors, and (2) leverage the complementary information encoded in each predictions via the mixer.

B. Base Predictors

1) Network Architectures: To construct multiple base predictors, we leverage CNN-based architectures: ResNet [24], DenseNet [25] and ResNext [26] and Transformer-based architectures: Conformer [22], Volo [21] and R50+ViT-B/16 [23] as encoders in parallel, and adopt the decoder of BTS [1]. The Transformer-based predictors capture the complicated spatial transformation and long-range dependencies that comprise a global representation. Local feature details, on the other hand, are prone to be ignored. The CNN-based predictors are able to collect local features through convolutional operations. Besides, we find that in practice, random initialization of network parameters, random data shuffling and random augmentation, also enforce the diversity among base predictors.

2) Training Schedules: There are roughly three schedules used to train base predictors: independent training, simultaneous training and sequential training [44].

In the independent training, each base predictor is trained separately. Most ensemble models adopt the independent training schedule because of its simplicity and memory savings, in [45], [46], for example. For the simultaneous training, the parameters of all base predictors are updated together in each training iteration. The simultaneous training schedule allows involving interaction such as mutual learning [47] of base predictors in the ensemble. However, this comes at a cost of huge memory consumption. The sequential training schedule can be viewed as a compromise between the independent and simultaneous training schedules, where the base predictors are trained sequentially. In other words, the parameters of the trained base predictors are frozen before training the next base predictor, enabling unidirectional interaction.

The training schedules are primarily explored on the image classification and may not be entirely suitable for depth estimation because its network typically consists of a complicated encoder-decoder architecture. In addition, the numerous matrix multiplication operations in the Transformer determine the massive memory demand. Hence, we employ the independent training schedule. Since the base predictors do not interact with one another, they are more likely to fall into distinct local optimums. Inconsistencies in the base predictors contribute to the generation of asymmetric predictions.

To avoid biasing the estimation performance, base predictors are only trained on a subset of the whole training set. Concretely, we divide the completed training set into \( D_{train\_base} \) and \( D_{train\_mixer} \) at a 7:1 (empirical setting) ratio, and the base predictors are trained on \( D_{train\_base} \). The predictions on \( D_{train\_mixer} \) made by base predictors can then be used to train the mixer. Algorithm 1 summarizes the overall training procedure.
Algorithm 1: The TEDepth Training Procedure.

Input: Training set \( D = \{(r_n, d_n)\}_{n=1}^N \); randomly split \( D \) into \( D_{train\_base} \) and \( D_{train\_mixer} \) at a 7:1 ratio; level-0 base predictors: \( \{f_{\theta_i}\}_{i=1}^K \); level-1 mixer: \( M_\theta \).

Output: Trained base predictors \( \{f_{\theta_i}\}_{i=1}^K \) and mixer \( M_\theta \).

1) \# Train level-0 base predictors on the \( D_{train\_base} \).
2: for \( i = 1, \ldots, K \) do
3: Compute depth predictions of base predictor \( f_{\theta_i} \);
4: Calculate the loss function in 2;
5: Update the model parameters \( \{\theta_i\} \) by AdamW [48];
6: end for
7) \# Train level-1 mixer on the \( D_{train\_mixer} \).
8: Freeze network parameters of base predictors \( \{f_{\theta_i}\}_{i=1}^K \);
9: Compute feature predictions of \( \{f_{\theta_i}\}_{i=1}^K \);
10: Compute depth predictions of mixer \( M_\theta \);
11: Calculate the loss function in 2;
12: Update the model parameters \( \theta \) by AdamW.

3) Loss Function: We use a scaled scale-invariant (SSI) loss introduced by Lee et al. [1],

\[
  \mathcal{L}_{SSI} = \alpha \sqrt{\frac{1}{|\mathbf{T}|} \sum_j (g^j)^2 - \frac{\eta}{|\mathbf{T}|^2} \left( \sum_j g^j \right)^2},
\]

where \( g^j = \log \tilde{d}^j - \log \mathbf{d}^j \), \( \mathbf{d} \) is the ground-truth depth map, and \( \mathbf{T} \) stands for a set of pixels with valid ground-truth values. The \( \alpha \) and \( \eta \) are set to 10 and 0.85 based on [1].

C. Ensemble Fusion

The mixer \( M_\theta \) is leveraged for integrating the complementary predictions from base predictors. Formally,

\[
  \tilde{d} = M_\theta \left( \{f_{\theta_i}(r)\}_{i=1}^K \right).
\]

We did not fuse the final depth maps, but rather the feature maps of the penultimate layer, because multi-channel feature maps contain more exploitable information than single-channel depth maps.

The mixer contains four different types, uniformly weighted fusion, confidence-guided fusion, concatenation-based fusion and ranking-based fusion. After that, a convolutional layer can be added to generate the final depth estimate.

1) Uniformly Weighted Fusion: The predictions from base predictors are combined with uniform weight,

\[
  \mathbf{F} = \sum_{i=1}^K f_{\theta_i}(r),
\]

where \( \mathbf{F} \) denotes the fused feature map.

2) Confidence-Guided Fusion: The predictions are integrated using confidence maps,

\[
  \mathbf{F} = \sum_{i=1}^K C^i \circ f_{\theta_i}(r),
\]

with

\[
  C^i = \sigma_1 \left( \text{Conv}_{3 \times 3} \left( f_{\theta_i}(r) \right) \right),
\]

where \( C \) stands for the confidence map, \( \circ \) stands for the element-wise multiplication and \( \sigma_1(\cdot) \) denotes the sigmoid activation.

3) Concatenation-Based Fusion: The predictions are fused by the concatenation operation and convolutional layer,

\[
  \mathbf{F} = \sigma_2 \left( \text{Conv}_{3 \times 3} \left( \left\{ f_{\theta_i}(r) \right\}_1^K \right) \right),
\]

where \( [\cdot] \) stands for the concatenation operation and \( \sigma_2(\cdot) \) is the ELU activation [49].

4) Ranking-Based Fusion: The predictions are integrated via a gated activation unit based on the GRU network, with convolutions in place of fully connected layers,

\[
  z^i = \sigma_1 \left( \text{Conv}_{3 \times 3} \left( \left[ h^{i-1}, f_{\theta_i}(r_n) \right] \right) \right),
\]

\[
  s^i = \sigma_1 \left( \text{Conv}_{3 \times 3} \left( \left[ h^{i-1}, f_{\theta_i}(r_n) \right] \right) \right),
\]

\[
  \tilde{h}^i = \tanh \left( \text{Conv}_{3 \times 3} \left( \left[ s^i \circ h^{i-1}, f_{\theta_i}(r_n) \right] \right) \right),
\]

\[
  h^i = \left( 1 - z^i \right) \circ h^{i-1} + z^i \circ \tilde{h}^i,
\]

\[
  \mathbf{F} = \left[ \{h^i\}_{i=1}^K \right],
\]

where \( z \) denotes the update gate, \( s \) denotes the reset gate, and \( h \) denotes the hidden state. With the deepening of fusion, the accuracy of hidden state in ConvGRU gradually increases. At this point, feeding lower-accuracy feature maps into the ConvGRU may deteriorate the hidden state. Hence, the ConvGRU receives feature maps in the order of accuracy from low to high.

Finally, we use a layer of convolution to convert the fused feature map into the final comprehensive depth map,

\[
  \tilde{d} = \kappa \sigma_1 \left( \text{Conv}_{3 \times 3} (\mathbf{F}) \right),
\]

where \( \kappa \) denotes the scale factor, and is set to 10 for the NYU-Depth-v2 and SUN RGB-D datasets, and 80 for the KITTI dataset, following the settings in [1].

IV. EXPERIMENT

We conduct an extensive set of experiments on three standard benchmarks for indoor and outdoor scenarios, including NYU-Depth-v2, KITTI and SUN RGB-D. In the following, we first present a description of the relevant datasets, evaluation metrics and implementation details. Then, we provide quantitative and qualitative comparison results to previous state-of-the-art competitors. Finally, we demonstrate generalization and ablation studies to discuss a detailed analysis of TEDepth.
TABLE I
QUANTITATIVE DEPTH COMPARISON AGAINST PREVIOUS STATE-OF-THE-ART APPROACHES ON THE NYU-DEPTH-V2 DATASET

| Method          | Cap | Abs Rel ↓ | RMSE ↓ | log10 ↓ | δ < 1.25 ↑ | δ < 1.25² ↑ | δ < 1.25³ ↑ |
|-----------------|-----|-----------|--------|---------|------------|-------------|-------------|
| Eigen et al. [29]| 0-10m | 0.158     | 0.641  | -       | 0.769      | 0.950       | 0.988       |
| Fu et al. [10]  | 0-10m | 0.115     | 0.509  | 0.051   | 0.828      | 0.965       | 0.992       |
| Qi et al. [31]  | 0-10m | 0.128     | 0.569  | 0.057   | 0.834      | 0.960       | 0.990       |
| VNL [32]        | 0-10m | 0.108     | 0.416  | 0.048   | 0.875      | 0.976       | 0.994       |
| BTS [1]         | 0-10m | 0.113     | 0.407  | 0.049   | 0.871      | 0.977       | 0.995       |
| Chen et al. [50]| 0-10m | 0.111     | 0.514  | 0.048   | 0.878      | 0.977       | 0.994       |
| Zhang et al. [51]| 0-10m | 0.112     | 0.447  | 0.048   | 0.881      | 0.979       | 0.996       |
| DAV [52]        | 0-10m | 0.108     | 0.412  | -       | 0.882      | 0.980       | 0.996       |
| Long et al. [53]| 0-10m | 0.101     | 0.377  | 0.044   | 0.890      | 0.982       | 0.996       |
| TransDepth [2]  | 0-10m | 0.106     | 0.365  | 0.045   | 0.900      | 0.983       | 0.996       |
| P3Depth [33]    | 0-10m | 0.104     | 0.356  | 0.043   | 0.898      | 0.981       | 0.996       |
| TEDepth (ours)  | 0-10m | **0.100** | **0.349** | **0.043** | **0.907** | **0.987** | **0.998** |

"-" means not applicable. The best results are indicated in bold.

A. Datasets

NYU-Depth-v2 dataset [14] contains 120 K RGB and depth samples from 464 indoor scenes, which are captured at a resolution of 640 × 480 pixels. We adopt the official split as previous works and the dataset processed by Lee et al. [1], resulting in 24231 and 654 image-depth pairs for training and testing, respectively.

KITTI is an outdoor dataset [28] acquired from 61 scenes using equipment placed on a moving vehicle. The resolution of RGB images is around 1241 × 376 pixels. To compare with existing works, we follow the commonly used Eigen split [29], which includes 29 scenes of 697 images for the test set, and 23488 images covering 32 scenes for the training set.

SUN RGB-D is an indoor dataset [27] with roughly 10 K images captured with four sensors and a wide range of scene diversity. To validate the generalization ability of models, we randomly collect 500 images.

B. Evaluation Metrics

In line with previous works, we use the standard evaluation metrics in our experiments:

- **Abs Rel**: 1/\|T\| \sum_{d \in T} |\hat{d} - d|/d;
- **Sq Rel**: 1/\|T\| \sum_{d \in T} |\hat{d} - d|²/d;
- **RMSE**: \sqrt{1/\|T\| \sum_{d \in T} |\hat{d} - d|²};
- **log10**: 1/\|T\| \sum_{d \in T} |log10(\hat{d}) - log10(d)|²;
- δ < t: % of d satisfies (max(\hat{d}, d) - \hat{d} = δ < t) for t = 1.25, 1.25², 1.25³.

C. Implementation Details

The TEDepth is implemented in the PyTorch library [54] and trained on four NVIDIA RTX A5000 GPUs. We use the AdamW optimizer [48] where \(\beta_1 = 0.9\), \(\beta_2 = 0.999\) and \(\epsilon = 1e-6\), and a batch size of 4. The learning rate is scheduled via polynomial decay from a base value of 1e-4 with power \(p = 0.9\). The total number of epochs is set to 50.

To further increase the diversity among base predictors, we perform random horizontal flips and rotations in the ranges \([-1, 1]\) and \([-2.5, 2.5]\) for KITTI and NYU-Depth-v2 datasets, respectively, as well as the following augmentations at 50% chance: random contrast, brightness, and color adjustment with ranges of ±0.1. Based on the random crop, the resolutions of the input are 352 × 704 pixels for KITTI and 416 × 544 pixels for NYU-Depth-v2.

D. Comparison to Previous State-of-The-Art Competitors

1) Depth Estimation: Table I summarizes the comparison results on the NYU-Depth-v2 dataset, and TEDepth exceeds existing monocular depth estimation methods by a large margin. To demonstrate the competitiveness of our TEDepth in the outdoor scenario, we also present the comparison results on the KITTI dataset in Table II. Compared to several recent competing methods such as P3Depth [33] and TransDepth [2], our approach is superior.

It is worth noting that TEDepth achieves improvements on almost all metrics in both scenarios. The metrics Abs Rel, Sq Rel, RMSE, RMSE log and \(log10\) show the errors between the predicted depth and ground-truth, and the metrics \(\delta < 1.25\), \(\delta < 1.25²\) and \(\delta < 1.25³\) describe the number of estimated outliers. The consistent advances in reducing errors and the number of outliers support our standpoint that integrating the unique strengths of multiple weak predictor results in a more comprehensive and accurate depth predictor.

Figs. 3 and 4 display the qualitative depth comparisons. As presented in Fig. 3, the depth maps of TEDepth deliver finer-grained details and sharper object boundaries. In Fig. 4, the compared methods struggle with thinner structures (e.g., signs and posts) and difficult object boundaries such as trees that overlap with foliage, while ours accurately estimates the depth.
TABLE II  
QUANTITATIVE DEPTH COMPARISON ON THE KITTI DATASET

| Method            | Cap | Abs Rel ↓ | Sq Rel ↓ | RMSE ↓ | RMSE Log ↓ | δ < 1.25 ↑ | δ < 1.25² ↑ | δ < 1.25³ ↑ |
|-------------------|-----|-----------|----------|--------|------------|-------------|-------------|-------------|
| Eigen et al. [7]  | 0-80m | 0.203     | 1.548    | 6.307  | 0.282      | 0.702       | 0.898       | 0.967       |
| Fu et al. [10]    | 0-80m | 0.072     | 0.307    | 2.727  | 0.120      | 0.932       | 0.984       | 0.994       |
| VNL [32]          | 0-80m | 0.072     | -        | 3.258  | 0.117      | 0.938       | 0.990       | 0.998       |
| BTS [1]           | 0-80m | 0.061     | 0.261    | 2.384  | 0.099      | 0.954       | 0.992       | 0.998       |
| Chen et al. [50]  | 0-80m | 0.090     | 0.546    | 3.802  | 0.151      | 0.902       | 0.972       | 0.990       |
| Zhang et al. [51] | 0-80m | 0.064     | 0.265    | 3.084  | 0.106      | 0.952       | 0.993       | 0.998       |
| TransDepth [2]    | 0-80m | 0.064     | 0.252    | 2.755  | 0.098      | 0.956       | 0.994       | **0.999**   |
| P3Depth [33]      | 0-80m | 0.071     | 0.270    | 2.842  | 0.103      | 0.953       | 0.993       | 0.998       |
| TEDepth (ours)    | 0-80m | **0.056** | **0.174**| **2.223**| **0.084**  | **0.968**   | **0.996**   | **0.999**   |

| Method            | Cap | Abs Rel ↓ | Sq Rel ↓ | RMSE ↓ | RMSE Log ↓ | δ < 1.25 ↑ | δ < 1.25² ↑ | δ < 1.25³ ↑ |
|-------------------|-----|-----------|----------|--------|------------|-------------|-------------|-------------|
| Fu et al. [10]    | 0-50m | 0.071     | 0.268    | 2.271  | 0.116      | 0.936       | 0.985       | 0.995       |
| BTS [1]           | 0-50m | 0.058     | 0.183    | 1.995  | 0.090      | 0.962       | 0.994       | **0.999**   |
| Chen et al. [50]  | 0-50m | 0.087     | 0.440    | 2.907  | 0.143      | 0.913       | 0.976       | 0.991       |
| Zhang et al. [51] | 0-50m | 0.061     | 0.200    | 2.283  | 0.099      | 0.960       | 0.995       | **0.999**   |
| TransDepth [2]    | 0-50m | 0.061     | 0.185    | 1.992  | 0.091      | 0.963       | 0.995       | **0.999**   |
| TEDepth (ours)    | 0-50m | **0.054** | **0.135**| **1.664**| **0.080**  | **0.972**   | **0.997**   | **0.999**   |

The best results are indicated in bold.

![Fig. 3. Qualitative depth comparison with previous state-of-the-art approaches on the NYU-Depth-v2 dataset. The red boxes show the regions to focus on. Our TEDepth can derive depth maps with finer-grained details and more clear object boundaries.](image1)

![Fig. 4. Qualitative depth comparison on the KITTI dataset. The red boxes indicate the areas to emphasize. Our TEDepth demonstrates a better performance on thinner structures signs and posts, for example and performs better to delineate difficult object boundaries.](image2)
of these smaller details. In addition, we visualize the reconstructed 3D scenes in Figs. 5 and 6. Thanks to the comprehensive depth predictions, the point clouds of TEDeepth demonstrate few distortions and preserve prominent geometric features.

Figs. 7 and 8 show the errors and corresponding standard deviations on the important metric RMSE at different capped depth ranges. As we can see, TransDepth is more accurate than BTS, but it appears to be less stable. By contrast, our TEDeepth is able to maintain both high accuracy and stability at close, middle and distant distances.

In general, the excellent quantitative and qualitative results on NYU-Depth-v2 and KITTI datasets verify the effectiveness of the TEDeepth for monocular depth estimation.

2) Ensemble Scheme: To further demonstrate the strengths of TEDeepth, we compare it with two state-of-the-art ensemble schemes Snapshot [16] and Balaji et al. [41] in Table III. Except for differences in the ensemble scheme Table V, other configurations remain the same. We also report results of the non-ensemble method TransDepth. As can be seen, our TEDeepth is able to achieve higher accuracy with fewer parameters than the compared ensemble schemes, indicating that coupling of CNN and Transformer contributes significantly to neural ensembles.
in monocular depth estimation. Even compared to TransDepth, TEDepth has advantages not only in terms of accuracy but also in terms of quantity of parameters, e.g., in the NYU-Depth-v2 dataset.

E. Generalization on the SUN RGB-D Dataset

We also study the generalizability of TEDepth, as evidence that TEDepth does learn transferable features rather than simply memorize training data. To verify this, we evaluate the models trained by NYU-Depth-v2 on the SUN RGB-D dataset without any fine-tuning. As shown in Table IV, TransDepth achieves a significant reduction in root mean square error and the number of estimated outliers compared to BTS, but a larger error on the Abs Rel, which is susceptible to the close-range errors. In contrast, TEDepth generalizes well on all metrics across different cameras.
Fig. 9. Visualization of the penultimate layer feature maps on the NYU-Depth-v2 dataset, achieved by choosing one principle channel via PCA decomposition and then displaying the feature map as a heat map.

TABLE VIII

| Effect of Diversity Among Base Predictors |
|------------------------------------------|
| Method | Avg Rel | Sq Rel | RMSE | $f < 1.50 \%$ | $f < 1.25 \%$ |
|--------|---------|--------|------|----------------|----------------|
| TEDepth-RBF (2) | 0.038 | 0.035 | 0.360 | 0.900 | 0.900 |
| BP-ResNet101x2/SP-DenseNet161x2 | 0.055 | 0.184 | 2.363 | 0.965 | 0.995 |
| BP-Conformer/BP-Volo | 0.058 | 0.191 | 2.317 | 0.964 | 0.995 |
| TEDepth-RBF (2) | 0.056 | 0.170 | 2.257 | 0.966 | 0.996 |

All variants adopt the same mixer rbf.

F. Ablation Studies

To better understand how the different elements in TEDepth impact the overall performance, we perform a series of ablation studies, involving the number of base predictors, mixer types, diversity among base predictors and fusion locations.

1) Number of Base Predictors: Tables IX and X demonstrate the impact of number of base predictors on accuracy, parameters and FLOPs. We adopt the ranking-based fusion as mixer. It is interesting that after a certain number of base predictors fused, ensemble accuracy stops improving or even declines on some metrics. The phenomenon is consistent with the findings in [40] that many could be better than all. In addition, we notice that when the two base predictors are integrated, the accuracy improvement is the greatest, while the parameters and FLOPs of TEDepth are smallest. The increase in accuracy becomes smaller and smaller as the base predictors are fused further. Therefore, for new datasets, we advocate integrating two or three base predictors to get a good trade-off between performance and efficiency.

2) Mixer Types: In Tables VI and VII, we investigate the influence of different mixer types. It is clear that, regardless of the deployment type, TEDepth can significantly improve accuracy over base predictors. As a result, in the case of ensuring diversity among base predictors, the ensemble operation determines the approximate range of accuracy. Besides, the uniformly weighted fusion mixer performs the worst out of the four types of mixers, and the concatenation-based and ranking-based fusions perform the best on NYU-Depth-v2 and KITTI, respectively.

3) Diversity Among Base Predictors: Table VIII summarizes the effect of diversity among base predictors. We compare TEDepth-RBF (2) with the best two Transformer-based fusion model, the best two CNN-based fusion model and the best two identical base predictors fusion model. Thanks to the complementarity of CNN and visual transformer, our TEDepth-RBF (2) performs better on two datasets.

4) Fusion Locations: As shown in Table XI, we present the comparison of fusing at penultimate and final layers. Given that the multi-channel feature maps contain more exploitable information than the single-channel depth maps, the results verify the deduction.

In Figs. 9 and 10, we show the penultimate layer feature maps from base predictors to assess their contribution towards the asymmetric depth errors and the feature maps of TEDepth to highlight how fusing multiple predictors handles asymmetric depth errors. We find that the chosen base predictors indeed provide asymmetric feature maps, and the TEDepth is able to inherit the strengths of each base predictor to alleviate their catastrophic errors.

In Fig. 11, we present the performance comparison of base predictors. An interesting phenomenon is that Transformer-based predictors perform better on NYU-Depth-v2, whereas CNN-based predictors perform better on KITTI. One reason for this could be the ground-truth type. NYU-Depth-v2 has dense ground-truth depth maps. On the other hand, the ground-truth depth maps of KITTI are relatively sparse. By capturing long-range dependencies, visual transformer can improve accuracy at pixels without ground-truth supervision. It may, however, degrade the depth accuracy at pixels supervised by ground-truth. Because of the dense ground-truth depth maps in the NYU-Depth-v2 dataset, it is easier to obtain more accurate depth maps through long-range dependencies. In the case of the KITTI dataset, the opposite is true.
TABLE IX

| Method              | Abs Rel | RMSE  | log_{10} | \( \delta < 1.25 \) ↑ | \( \delta < 1.25^2 \) ↑ | Parameters | FLOPs |
|---------------------|--------|-------|----------|------------------------|------------------------|------------|-------|
| BP-ResNet101        | 0.115  | 0.401 | 0.049    | 0.872                  | 0.979                  | 69M        | 66G   |
| BP-DenseNet161      | 0.111  | 0.392 | 0.047    | 0.879                  | 0.981                  | 47M        | 61G   |
| BP-ResNext101       | 0.114  | 0.402 | 0.049    | 0.873                  | 0.977                  | 113M       | 93G   |
| BP-R50+Vit-B/16     | 0.108  | 0.375 | 0.046    | 0.890                  | 0.983                  | 152M       | 101G  |
| BP-Conformer        | 0.109  | 0.382 | 0.046    | 0.889                  | 0.981                  | 141M       | 139G  |
| BP-Volo             | 0.108  | 0.376 | 0.046    | 0.896                  | 0.983                  | 128M       | 117G  |
| TEDepth-RBF (2)     | 0.103  | 0.360 | 0.044    | 0.900                  | 0.985                  | 199M       | 170G  |
| TEDepth-RBF (3)     | 0.102  | 0.358 | 0.044    | 0.901                  | 0.986                  | 326M       | 291G  |
| TEDepth-RBF (4)     | 0.102  | 0.355 | 0.044    | 0.903                  | 0.986                  | 395M       | 362G  |
| TEDepth-RBF (5)     | 0.101  | 0.355 | 0.043    | 0.903                  | 0.986                  | 533M       | 505G  |
| TEDepth-RBF (6)     | 0.101  | 0.355 | 0.043    | 0.902                  | 0.986                  | 648M       | 602G  |

BP: Base predictor; RBF: Ranking-based fusion. The detailed explanations from TEDepth-RBF (2) to TEDepth-RBF (6) are in Table v.

TABLE X

| Method              | Abs Rel | Sq Rel | RMSE  | \( \delta < 1.25 \) ↑ | \( \delta < 1.25^2 \) ↑ | Parameters | FLOPs |
|---------------------|--------|--------|-------|------------------------|------------------------|------------|-------|
| BP-ResNet101        | 0.061  | 0.213  | 2.420 | 0.961                  | 0.994                  | 69M        | 92G   |
| BP-DenseNet161      | 0.058  | 0.201  | 2.420 | 0.962                  | 0.994                  | 47M        | 83G   |
| BP-ResNext101       | 0.058  | 0.197  | 2.336 | 0.963                  | 0.994                  | 113M       | 129G  |
| BP-R50+Vit-B/16     | 0.061  | 0.213  | 2.409 | 0.962                  | 0.995                  | 152M       | 140G  |
| BP-Conformer        | 0.063  | 0.206  | 2.366 | 0.961                  | 0.995                  | 141M       | 194G  |
| BP-Volo             | 0.069  | 0.219  | 2.378 | 0.959                  | 0.995                  | 128M       | 163G  |
| TEDepth-RBF (2)     | 0.056  | 0.178  | 2.257 | 0.966                  | 0.996                  | 253M       | 335G  |
| TEDepth-RBF (3)     | 0.056  | 0.174  | 2.223 | 0.968                  | 0.996                  | 382M       | 503G  |
| TEDepth-RBF (4)     | 0.057  | 0.185  | 2.271 | 0.967                  | 0.996                  | 429M       | 594G  |
| TEDepth-RBF (5)     | 0.056  | 0.182  | 2.265 | 0.968                  | 0.996                  | 676M       | 740G  |
| TEDepth-RBF (6)     | 0.057  | 0.183  | 2.254 | 0.968                  | 0.996                  | 745M       | 838G  |

Fig. 10. Visualization of the penultimate layer feature maps from BP-Volo, BP-Conformer, BP-ResNext101 and TEDepth on the KITTI dataset.

TABLE XI

| Dataset               | NYU-Depth-v2 | KITTI |
|-----------------------|--------------|-------|
|                       | FL-RBF (5)   | PL-RBF (5)  | FL-RBF (3)  | PL-RBF (3)  |
| Abs Rel ↓             | 0.116        | 0.101   | 0.068       | 0.056       |
| Sq Rel ↓              | 0.067        | 0.052   | 0.237       | 0.174       |
| RMSE ↓                | 0.403        | 0.355   | 2.534       | 2.223       |
| \( \log_{10} \downarrow \) | 0.050       | 0.043   | 0.030       | 0.024       |
| \( \delta < 1.25 \) ↑ | 0.868        | 0.903   | 0.956       | 0.968       |
| \( \delta < 1.25^2 \) ↑ | 0.979       | 0.986   | 0.993       | 0.996       |
| \( \delta < 1.25^3 \) ↑ | 0.996       | 0.998   | 0.993       | 0.999       |

FL: Final layer, PL: Penultimate layer.

Fig. 11. Comparison of performance with respect to the base predictors. It should be noted that the larger the shadow area, the smaller the model errors. (a) Comparison on the NYU-Depth-v2 dataset. (b) Comparison on the KITTI dataset.
V. CONCLUSION

In this work, we open up a new avenue for model design by introducing ensemble learning and thoroughly investigate its utility for monocular depth estimation. Besides, we develop an effective and easy-to-implement framework, TEDDepth, based on two levels to deliver asymmetric bottom-level predictions and adaptively merge them at the top level, enabling a comprehensive and accurate depth estimation. Extensive experiments are carried out on three challenging datasets, including NYU-Depth-v2, KITTI and SUN RGB-D datasets. The experimental results emphasize the effectiveness and strong generalizability of TEDDepth. We hope our study can encourage more works applying the neural ensembles into monocular depth estimation and enlighten the framework design of other closely related tasks, e.g., surface normal prediction.

VI. CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Shuwei Shao: Conceptualization, Methodology, Software, Validation, Writing - original draft, Writing & editing. Ran Li: Methodology, Software, Validation, Investigation, Visualization, Writing - review & editing. Zhongcai Pei: Software, Validation, Investigation, Writing - review & editing. Zhong Liu: Software, Validation, Investigation, Writing - review & editing. Weihai Chen: Resources, Funding acquisition, Writing - review & editing. Supervision. Wentao Zhu: Software, Validation, Writing - review & editing. Xuming Wu: Investigation, Writing - review & editing. Baochang Zhang: Conceptualization, Methodology, Resources, Writing - review & editing, Supervision.

REFERENCES

[1] 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,” 2019, arXiv:1907.10326.
[2] G. Yang, H. Tang, M. Ding, N. Sebe, and E. Ricci, “Transformer-based attention networks for continuous pixel-wise prediction,” in Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 16269–16279.
[3] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, “FuseNet: Incorporating depth into semantic segmentation via fusion-based CNN architecture,” in Proc. Asian Conf. Comput. Vis., 2016, pp. 213–228.
[4] W. Lee, N. Park, and W. Woo, “Depth-assisted real-time 3D object detection for augmented reality,” in Proc. 21st Int. Conf. Artif. Reality Telexistence, 2011, pp. 126–132.
[5] S. Shao et al., “Self-supervised monocular depth and ego-motion estimation in endoscopy: Appearance flow to the rescue,” Med. Image Anal., vol. 77, 2022, Art. no. 102338.
[6] A. Saxena et al., “Learning depth from single monocular images,” in Proc. Adv. Neural Inf. Process. Syst., vol. 18, pp. 1–8, 2005.
[7] D. Eigen and R. Fergus, “Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture,” in IEEE Int. Conf. Comput. Vis., 2015, pp. 2650–2658.
[8] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, “Deep depth prediction with fully convolutional residual networks,” in Proc. 4th Int. Conf. 3D Vis., 2016, pp. 239–248.
[9] J. Li, R. Klein, and A. Yao, “A two-streamed network for estimating fine-scaled depth maps from single RGB images,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 3372–3380.
[10] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, “Deep ordinal regression network for monocular depth estimation,” in Proc. IEEE Comput. Vis. Pattern Recognit., 2018, pp. 2002–2011.
[11] W. Song et al., “Contextualized CNN for scene-aware depth estimation from single RGB image,” IEEE Trans. Multimedia, vol. 22, no. 5, pp. 1220–1233, May 2020.
[12] X. Yang, Y. Gao, H. Luo, C. Liao, and K.-T. Cheng, “Bayesian denet: Monocular depth prediction and frame-wise fusion with synchronized uncertainty,” IEEE Trans. Multimedia, vol. 21, pp. 2701–2713, 2019.
[13] C. Ling, X. Zhang, and H. Chen, “Unsupervised monocular depth estimation using attention and multi-warp reconstruction,” IEEE Trans. Multimedia, vol. 24, pp. 2938–2949, 2022.
[14] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, “Indoor segmentation and support inference from RGBD images,” in Proc. Eur. Conf. Comput. Vis., 2012, pp. 746–760.
[15] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, “A. survey on ensemble learning,” Front. Comput. Sci., vol. 14, no. 2, pp. 241–258, 2020.
[16] G. Huang et al., “Snapshot ensembles: Train 1, get m for free,” in Proc. Int. Conf. Learn. Representations, 2017, pp. 1–12.
[17] Y. Yang et al., “Locally-coherent fusion in the subparameter space can be effective for ensembles of deep convolutional neural networks,” Pattern Recognit., vol. 109, 2021, Art. no. 107582.
[18] R. Soloyev, W. Wang, and T. Gaboruseva, “Weighted boxes fusion: Ensemble boxes from different object detection models,” Image Vis. Comput., vol. 107, 2021, Art. no. 104117.
[19] Y. Shen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, Apr. 2018.
[20] René Ranftl, A. Bochkovskiy, and V. Koltun, “Vision transformers for dense prediction,” in Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 12179–12188.
[21] L. Yuan, Q. Hou, Z. Jiang, J. Feng, and S. Yan, “Volo: View-lookout for visual recognition,” IEEE Trans. Pattern Anal. Mach. Intell., 2022, doi: 10.1109/TPAMI.2022.3206108.
[22] Z. Peng et al., “Conformer: Local features coupling global representations for visual recognition,” in Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 57–57.
[23] A. Dosovitskiy et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” in Proc. Int. Conf. Learn. Representations, 2021, pp. 1–12.
[24] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.
[25] G. Huang, Z. Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger, “Densely connected convolutional networks,” in Proc. IEEE Comput. Vis. Pattern Recognit., 2017, pp. 4700–4708.
[26] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1492–1500.
[27] S. Song, P. Samuelsen-Lichtenberg, and J. Xiao, “Sun RGB-D: A RGB-D scene understanding benchmark suite,” in Proc. IEEE Comput. Vis. Pattern Recognit., 2015, pp. 567–576.
[28] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets robotics: The Kitti dataset,” Int. J. Robot. Res., vol. 32, no. 11, pp. 1231–1237, 2013.
[29] D. Eigen, C. Puhrsch, and R. Fergus, “Depth map prediction from a single image using a multi-scale deep network,” in Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 2366–2374.
[30] Y. Cao, Z. Wu, and C. Shen, “Estimating depth from monocular images as classification using deep fully convolutional residual networks,” IEEE Trans. Circuits Syst. Video Technol., vol. 28, no. 11, pp. 3174–3182, 2018.
[31] X. Qi, R. Liao, Z. Liu, R. Urtasun, and J. Jia, “GeoNet: Geometric neural network for joint depth and surface normal estimation,” in Proc. IEEE Comput. Vis. Pattern Recognit., 2018, pp. 283–291.
[32] W. Yin, Y. Liu, C. Shen, and Y. Yan, “Enforcing geometric constraints of virtual normal for depth prediction,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 5684–5693.
[33] V. Patil, C. Sakaridis, A. Liniger, and L. V. Gool, “3Ddepth: Monocular depth estimation with a piecewise planarity prior,” in Proc. IEEE Comput. Vis. Pattern Recognit., 2022, pp. 1610–1621.
[34] A. Vaswani et al., “Attention is all you need,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 5998–6008.
[35] Z. Liu et al., “Swin transformer: Hierarchical vision transformer using shifted windows,” in Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 10012–10022.
[36] S. Zheng “Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 6881–6890.
[37] W. Gao et al., “TS-CAM: Token semantic coupled attention map for weakly supervised object localization,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 2886–2895.
[38] Y. Jiang, S. Chang, and Z. Wang, “TransGAN: Two transformers can make one strong GAN,” in Proc. Adv. Neural Inf. Process. Syst., 2021, pp. 14745–14758.
E. Bauer and R. Kohavi, “An empirical comparison of voting classification algorithms: Bagging, boosting, and variants,” Mach. Learn., vol. 36, no. 1, pp. 105–139, 1999.

Z.-H. Zhou, J. Wu, and W. Tang, “Ensembling neural networks: Many could be better than all,” Artif. Intell., vol. 137, no. 1/2, pp. 239–263, 2002.

B. Lakshminarayanan, A. Pritzel, and C. Blundell, “Simple and scalable predictive uncertainty estimation using deep ensembles,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 6402–6413.

F. Wenzel, J. Snoek, D. Tran, and R. Jenatton, “Hyperparameter ensembles for robustness and uncertainty quantification,” in Proc. Adv. Neural Inf. Process. Syst. 2020, pp. 6514–6527.

H. Yang et al., “DIVERGE: Diversifying vulnerabilities for enhanced robust generation of ensembles,” in Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 5505–5515.

M. MdX. Islam Yao, and K. Murase, “A constructive algorithm for training cooperative neural network ensembles,” IEEE Trans. Neural Netw., vol. 14, no. 3, pp. 820–834, Jul. 2003.

O. Russakovsky et al., “ImageNet large scale visual recognition challenge,” Int. J. Comput. Vis., vol. 115, no. 3, pp. 211–252, 2015.

F. Liao et al., “Defense against adversarial attacks using high-level representation guided denoiser,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1778–1778.

Y. Zhang, T. Xiang, T.-M. Hospedales, and H. Lu, “Deep mutual learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 4320–4328.

I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in Proc. Int. Conf. Learn. Representations, 2018, pp. 1–8.

D.-A. Clevert, T. Unterthiner, and S. Hochreiter, “Fast and accurate deep network learning by exponential linear units (ELUs),” 2015, arXiv:1511.07289.

X. Chen, X. Chen, Y. Zhang, X. Fu, and Z.-J. Zha, “Laplacian pyramid neural network for dense continuous-value regression for complex scenes,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 11, pp. 5034–5046, Nov. 2021.

J. Zhang, H. Yue, X. Wu, W. Chen, and C. Wen, “Densely connecting depth maps for monocular depth estimation,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 149–165.

L. Huynh, P. Nguyen-Ha, J. Matas, E. Rahtu, and J. Heikkilä, “Guiding deep network learning by exponential linear units (ELUs),” 2015, arXiv:1511.07289.

Zhong Liu received the B.Eng., M.Eng., and Ph.D. degrees from the Harbin Institute of Technology, Harbin, China, in 1991, 1994, and 1997, respectively. From 2000, he was an Associate Professor with the School of Automation Science and Electronic Engineering, Beihang University, Beijing, China, where he has been a Professor since 2006. He has authored or coauthored more than 50 technical papers in referred journals and conference proceedings and field more than ten patents. His research interests include bionic CPG mechanism and control, computer vision, and parallel mechanism and robotics.

Weihai Chen (Member, IEEE) received the B.Eng. degree from Zhejiang University, Hangzhou, China, in 1982, and the M.Eng. and Ph.D. degrees from Beihang University, Beijing, China, in 1988 and 1996, respectively. From 1999, he was an Associate Professor with the School of Automation Science and Electronic Engineering, Beihang University, Beijing, China, where he has been a Professor since 2007. He has authored or coauthored more than 200 technical papers in referred journals and conference proceedings and field more than 30 patents. His research interests include bio-inspired robotics, computer vision, image processing, precision mechanism, and automation and control.

Shuwei Shao received the B.Eng. degree from Xidian University, Xi’an, China, in 2019. He is currently working toward the Ph.D. degree with the School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. His research interests include image registration and depth estimation.

Zongcai Pei received the B.Eng., M.Eng. and Ph.D. degrees from the Harbin Institute of Technology, Harbin, China, in 1991, 1994, and 1997, respectively. From 2000, he was an Associate Professor with the School of Automation Science and Electronic Engineering, Beihang University, Beijing, China, where he has been a Professor since 2006. He has authored or coauthored more than 50 technical papers in referred journals and conference proceedings and field more than twenty patents. His research interests include bionic CPG mechanism and control, computer vision, and parallel mechanism and robotics.

Ran Li received the bachelor’s degree in automation science from Beihang University, Beijing, China, in 2021, where he is currently working toward the master’s degree. His research interests include computer vision and machine learning.

Ren Li received the D.Eng. degree from the Harbin Institute of Technology, Harbin, China, in 2000. He is currently working toward the Ph.D. degree with the School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. His research interests include image registration and depth estimation.

Xingming Wu received the B.Eng. and M.Eng. degrees from Zhejiang University, Hangzhou, China, in 1981 and 1985, respectively. He is currently an Associate Professor with the School of Automation Science and Electronic Engineering, Beihang University, Beijing, China. His main research interests include intelligent sensor systems, embedded systems, autonomous mobile robots, and image processing.

Baochang Zhang received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1999, 2001, and 2006, respectively. From 2006 to 2008, he was a Research Fellow with the Chinese University of Hong Kong, Hong Kong, and Griffith University, Brisbane, QLD, Australia. He is currently a Professor with the Science and Technology on Aircraft Control Laboratory, School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. He was supported by the Program for New Century Excellent Talents with the University of Ministry of Education of China. His research interests include pattern recognition, machine learning, face recognition, and wavelets.