DepthBLIP-2: Leveraging Language to Guide BLIP-2 in Understanding Depth Information

Wei Chen, Changyong Shi, Chuanxiang Ma*, Wenhao Li, and Shulei Dong

[1] School of Computer Science and Information Engineering, Hubei University
[2] Engineering Research Center of Hubei Province in Intelligent Government Affairs and Application of Artificial Intelligence (Hubei University), Hubei University
[3] Hubei Key Laboratory of Big Data Intelligent Analysis and Application (Hubei University), Hubei University

Abstract. In recent years, visual language models have made significant advancements in the fields of computer vision and natural language processing. The BLIP-2 model effectively bridges modality gaps through its lightweight Q-Former, demonstrating excellent results with low training costs and highlighting the potential development directions for visual language models. However, applying BLIP-2 to more complex quantized target tasks, such as monocular depth estimation, presents challenges. In this paper, we propose a method for monocular depth estimation using BLIP-2. Our approach draws inspiration from DepthCLIP's use of language-guided models to comprehend depth information, leveraging the Q-Former module for modality fusion. Additionally, we introduce an adaptive depth bin to enhance the model's robustness against quantized distances. We name our method DepthBLIP-2 and make our code publicly available at: `https://github.com/especiallyW/DepthBLIP-2`.

Keywords: BLIP-2 · Visual-language model · Monocular depth estimation · Few-shot transfer learning.

1 Introduction

In recent years, significant progress has been made in research on visual language models. Several works have introduced novel architectures and utilized larger datasets for pre-training, achieving state-of-the-art results across a variety of downstream tasks. However, these advances are largely driven by scaling up models and datasets rather than by major architectural innovations. This scaling trend, while beneficial, also introduces challenges such as increased training costs.

Recently, the BLIP-2 model [14] has recently addressed some of these challenges by introducing a lightweight Querying Transformer, which facilitates effective mutual learning between pre-trained visual and language models. This design bridges the gap between these modalities and enables efficient multimodal

* Corresponding author: Chuanxiang Ma. E-mail: mcx838@hubu.edu.cn.

fusion learning. Such integration highlights the potential of visual language models to serve as foundational models in a variety of applications. However, further exploration is needed to extend BLIP-2's capabilities to more complex tasks in computer vision, such as depth estimation.

Monocular depth estimation is crucial for various industrial applications, including monocular 3D object detection and point cloud reconstruction. To achieve high performance, many works [1, 2, 4, 18, 22, 25, 28] have focused on building large labeled datasets and designing sophisticated networks from scratch to extract semantic relationships and estimate depth from images.

DepthCLIP [34] was the first approach to apply visual language models to monocular depth estimation without relying on labeled datasets. It leverages CLIP [24] as a pre-trained model and reformulates depth value regression into a distance classification task. This method allows CLIP to understand object proximity in terms of semantic similarity rather than directly quantifying distance, with predefined quantized distances (depth bin) mapping the proximity for effective depth estimation.

However, DepthCLIP [34] has limitations: (1) It relies solely on a simple dot product for modality fusion, which limits the potential of language-guided depth learning. (2) The predefined quantized distances are highly sensitive to different datasets, requiring careful adjustment based on dataset and scene variations.

Given these challenges, we explored how to adapt BLIP-2 as a foundational model for depth estimation while addressing its specific challenges. Recognizing the similarities between the depth quantization challenges faced by BLIP-2 and CLIP, we hypothesized that we could adapt DepthCLIP's strategy to overcome these limitations. Specifically, we sought to transform depth regression into a distance classification task, similar to DepthCLIP, while correcting the performance issues related to predefined quantized distances.

Thus, we propose DepthBLIP-2, an innovative method that effectively transfers BLIP-2 [14] to depth estimation tasks. DepthBLIP-2 employs the lightweight Q-Former module in BLIP-2 for modality fusion, enhancing depth estimation performance while keeping the model size efficient. Additionally, we introduce learnable quantized distances, fine-tuning only a few parameters with limited data, to resolve the performance variability caused by scene changes in DepthCLIP [34].

Our experiments demonstrate the effectiveness of DepthBLIP-2 through few-shot fine-tuning and zero-shot depth estimation. We evaluate its performance on the NYU Depth V2 [23], assessing performance across different scenes and addressing the limitations observed in DepthCLIP. Our contributions are summarized as follows:

1. We introduce DepthBLIP-2, which integrates semantic language knowledge with quantized depth prediction methods, addressing performance issues caused by scene variations. This extends the potential of BLIP-2 [14] as a foundational model.
2. We validate the effectiveness of DepthBLIP-2 on the NYU Depth V2 [23] dataset, showcasing its excellent performance in unsupervised settings.

3. Our code is open-source and available for download and validation.

2 Related Work

2.1 Visual Language Models

In recent years, visual language models haves achieved significant progress, with CLIP [24] and ALIGN [8] being among the most notable examples. CLIP utilizes 400M image-text pairs, while ALIGN employs 1.8B pairs for pre-training. These models have laid a solid foundation for the application of visual language models across a variety of downstream tasks [6, 12, 17, 20, 32, 34].

Different model architectures have been proposed to handle the requirements of various downstream tasks. Broadly, these architectures fall into four categories: dual-encoder architecture [8, 24], fusion-encoder architecture [16], encoder-decoder architecture [3,31], and unified transformer architecture [15,30]. Despite their architectural differences, these models rely on large-scale image-text datasets for end-to-end pre-training. However, as the size of both models and datasets continue to grow, the associated training costs increase exponentially.

The BLIP-2 model [14] addresses this issue by focusing on bridging the gap between modalities. It introduces a Q-Former to effectively combine powerful pre-trained visual and language models. This method not only leverages the strong representation learning capabilities of individual modalities but also reduces the overall pre-training cost. By doing so, BLIP-2 expands the research landscape of visual language models and offers a promising approach for foundation models in general domains.

Despite these advantages, there are still significant challenges in adapting BLIP-2 as a general-purpose foundation model. BLIP-2 was initially designed for visual language downstream tasks, such as image captioning and visual question answering, by using a large decoder-based language model. Similar to other visual language models [8, 24], BLIP-2 [14] lacks the necessary capabilities for certain general-domain tasks. For instance, in depth estimation, BLIP-2 struggles with quantifying continuous values, which is essential for accurate depth prediction.

2.2 Monocular Depth Estimation

Monocular depth estimation aims to predict pixel-level depth values from a single image. The model extracts semantic information from the input image to infer depth of objects, thereby producing the desired depth estimation outcome.

Research on depth estimation has a history spanning more than a decade [26, 27]. Early work primarily focused on depth cues from motion and texture. However, due to the absence of dynamic characteristics, these approaches struggled to deliver high accuracy. It was not until the success of deep neural network architectures in image recognition tasks [7,11] that significant progress in monocular depth estimation was achieved. Broadly, current research methods can be divided into two categories: supervised learning and unsupervised learning.

Supervised learning approaches rely on manually annotated datasets to train models using ground-truth depth maps. The goal is to capture semantic prior knowledge and establish the relationships between semantic features. Prominent examples include Make3D [25], AdaBins [1], DORN [4], and RPSF [22]. Recently, with the introduction of transformer architectures [28] into computer vision, new methods have emerged that leverage transformers for monocular depth estimation. For instance, ASTransformer [2] and DepthFormer [18] incorporate new interaction modules to enhance the extraction of semantic features. While supervised learning approaches have demonstrated remarkable performance, they are heavily dependent on large-scale annotated datasets, resulting in significant time and computational costs.

Unsupervised learning methods, on the other hand, do not require annotated datasets. Instead, they rely on pre-text tasks to guide the model in extracting semantic information from images, thereby learning depth cues. These methods include approaches that use video sequences to impose temporal constraints on geometric information [21, 33], others that incorporate planes and lines as prior knowledge [9], and some that leverage semantic language knowledge for depth estimation [34]. Aligned with our approach of extending BLIP-2 as a foundation model for general tasks, our work focuses on using semantic language knowledge for depth estimation [34], which eliminates the need for geometric priors.

For readers interested in gaining a deeper understanding of monocular depth estimation, we highly recommend the work of [29], which provides a comprehensive review of the advancements in this field.

3 Methodology

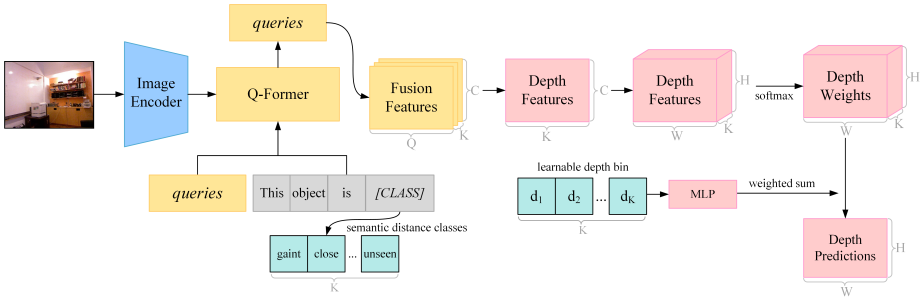

Fig. 1: Model architecture of DepthBLIP-2.

In this section, we first address the limitations of DepthCLIP and propose corresponding solutions (Sec. 3.1). We then describe the detailed process of the model in depth estimation tasks (Sec. 3.2 and Sec. 3.3).

The overall architecture of our model is illustrated in Fig. 1. Specifically, our approach is built upon the pre-trained BLIP-2 model [14]. First, we use

an image encoder to extract visual features from the input image. These visual features, along with predefined language prompts and Queries, are then fed into the Q-Former for multimodal fusion, resulting in corresponding semantic responses. Next, we apply several dimensionality transformations to these semantic responses, projecting them into adaptive depth bin. Finally, a linear combination of adaptive depth bin values is performed to produce the final depth prediction. This method fully leverages the strong feature extraction capabilities of BLIP-2 and effectively accomplishing the task of monocular depth estimation.

3.1 Existing Limitations and Solutions

Modality Fusion Previous search [10, 16, 30] has demonstrated that effective modality fusion is critical for improving the performance of visual-language models. DepthCLIP [34] employs a simple dot-product operation to achieve modality fusion, which may lead to suboptimal performance, especially in tasks such as depth estimation. This approach may not fully exploit the potential of language guidance in enhancing the performance of visual models. To address this limitation, we leverage the unique architecture of BLIP-2 [14], which utilizes Q-Former to replace the dot product operation. Q-Former not only fuses image features with language encoding but also integrates them in sophisticated manner. This fusion ensures that depth-related information is integrated effectively without significantly increasing the model's size or computational overhead.

Adaptive Depth Bin In DepthCLIP [34], the quantized depth bin is predefined and fixed. When the scene changes, the model's performance may degrade unless the quantized distances are manually adjusted for each scenario. To address this issue, we introduce a lightweight MLP module (as depicted in Fig. 1). By fine-tuning the MLP parameters with a portion of the training data, the model learns a set of adaptive depth bin that dynamically adjust based on the scene. This approach not only mitigates the negative impact of scene variations but also maintains low computational costs.

3.2 Image Encoding and Information Fusion

Image Encoding Given a monocular RGB image I, we input it into the image encoder of the BLIP-2 [14], which returns a C_1-dimensional feature map, denoted as P_{img}. The process is represented by the following equation:

$$\mathrm{P}_{img} = \mathrm{ImageEncoder}\,(I) \in \mathbb{R}^{(HW+1)\times C_1} \tag{1}$$

Our method differs from DepthCLIP [34] in that we directly extract the feature dimensions of $(HW + 1) \times C_1$. This allows the model to capture both local semantic knowledge from each patch in the feature map and global semantic knowledge across the entire images. By balancing the advantages of local and global information, this approach better supports subsequent steps where language guidance refines the image features for depth estimation.

Information Fusion After extracting the semantic knowledge from the image, we use Q-Former to allow language prompts to guide the model in understanding the relative distance of objects in the image. The first step is to define Queries, which act as soft visual prompts. Consistent with BLIP-2 model, we use 32 queries, each with a dimension of 768, as formalized below:

$$\text{Queries} \in \mathbb{R}^{Q \times C} \tag{2}$$

where Q represents the number of queries, and C is three-quarters of C_1, in accordance with the design of BLIP-2.

Next, we define language prompts to guide the model in learning depth information from the image. As in DepthCLIP [34], we use prompts such as "This object is [CLASS]". The placehoders, such as "[CLASS]", are replaced by semantic distance classes like "close", "far", "unseen" to generate K groups of semantic tokens. These tokens inform the model that it should learn the relative distance in the image. These semantic tokens, denoted as T, are then passed through Q-Former alongside the image features for deep fusion. The output is represented as follows:

$$\begin{gathered} \mathrm{P}_{fusion} = \text{Q-Former}\left(\mathrm{P}_{img}, \text{Queries}, T_i\right) \text{ for } i = 1, ..., K \\ \mathrm{P}_{fusion} \in \mathbb{R}^{Q \times C \times K} \end{gathered} \tag{3}$$

where dimensions Q and C correspond to those of Queries, and T_i represents the semantic tokens.

It is important to note that Q-Former is fully utilized for information fusion. The distance-related information extracted from the image in this manner is more precise. Since Q-Former has undergone pre-training, it can selectively extract image features that contain relevant language information while filtering out irrelevant visual details. This ensures that the fused image knowledge aligns more closely with the defined language prompts, improving the accuracy of depth estimation.

3.3 Projection and Combination of Depth Information

Guided by different language prompts, we have learned the fusion information P_{fusion} ($dim = Q \times C \times K$), which consists of K groups of image features. Each group represents the estimated depth of objects in the image, determined by language prompts that convey distance concepts. However, since the output dimension of P_{fusion} does not directly meet the requirements for depth estimation, further processing is needed. Specifically, we calculate the mean along dimension Q to obtain:

$$\mathrm{P}_f = \frac{1}{Q} \sum_{q=1}^{Q} \mathrm{P}_{fusion} \in \mathbb{R}^{C \times K} \tag{4}$$

We choose to compute the mean because each query encapsulates visual features, and averaging across Q helps preserve the overall semantic information.

After this step, we split P_f along the C dimension to obtain P_d, which now has dimension $H \times W \times K$.

Once the dimensional transformation is complete, we need to convert P_d, which currently provides a classification-based depth estimation, into continuous depth information for depth prediction. In other words, while P_d represents K levels of depth based on the guidance of semantic tokens T, the resulting depth information remains discrete. To transform this discrete information into continuous depth estimation, we quantify the distance values. For instance, the token "close" may be mapped to "1m", while "unseen" may correspond to "4m". We apply softmax function to P_d, mapping it to the range $[0, 1]$ to generate linear weights:

$$\mathrm{P}_w = \mathrm{softmax}\left(\mathrm{P}_{d_{:,:,i}}\right), \text{ for } i = 1, ..., K \tag{5}$$

Next, we perform a weighted linear combination of these weights, which represent K levels of depth, with adaptive depth bin values $\mathrm{d}(dim = K \times 1)$. Summing the K classes yields the final predicted depth value $\mathrm{P}_{\mathrm{pred}}$. The process can be formalized as:

$$\mathrm{P}_{\mathrm{pred}} = \sum_{i=1}^{K} \mathrm{P}_{w_{:,:,i}} \times \mathrm{d}_i \in \mathbb{R}^{H \times W} \tag{6}$$

where $\mathrm{P}_{\mathrm{pred}_{i,j}}$ represents the depth of the patch in i th row and j th column of the image.

It is worth noting that, unlike DepthCLIP [34], where the depth bin is predefined, our approach utilizes adaptive depth bin learned through the MLP module as described in Sec. 3.1.

4 Experiments

4.1 Datasets, Implementation Details, and Evaluation Metrics

Datasets We used the NYU Depth V2 [23] to comprehensively evaluate our method. This dataset contains indoor scenes captured by Microsoft Kinect, with paired RGB and depth images. In total, there are 1449 image pairs across 464 different scenes, with each image having a resolution of 480×640. To ensure consistency with the test datasets used in other methods for comparison, we adopted common data-splitting protocol [1]. Accordingly, our test set consists of 654 images from 215 scenes.

Implementation Details We use PyTorch to implement our ideas. Both the image encoder and Q-Former are based on the pre-trained BLIP-2 [14]. For language prompts, we followed DepthCLIP [34] by using hand-crafted templates

[1] `https://github.com/deeplearningais/curfil/wiki/Training-and-Prediction-with-the-NYU-Depth-v2-Dataset`

such as "This object is [CLASS]". The semantic distance classes used to represent object distance were defined as: ['giant', 'extremely close', 'close', 'not in distance', 'a little remote', 'far', 'unseen']. All input images were scaled to a resolution of 224×224, consistent with BLIP-2's requirements.

For the adaptive depth bin, we fine-tuned the MLP module using a subset of the data. Specifically, we used 16 images for training set and 8 images for validation (non of which overlapped with the test set). The fine-tuning was performed using the AdamW [19] optimizer, with an initial learning rate of 1e-3, a batch size of 16, and a total of 50 epochs.

Evaluation Metrics Our evaluation metrics are basically consistent with previous work. We use the following metrics to measure the effect of depth map prediction: absolute error in logarithmic space ($\log_{10}$), average absolute relative error (rel), root mean square error (rmse), threshold accuracy ($\mathrm{a_i}$). The formulas for each evaluation metrics are as follows:

$$\log_{10} = \frac{1}{n}\sum_{p}^{n} \left|\log_{10}\left(y_p\right) - \log_{10}\left(\hat{y}_p\right)\right| \tag{7}$$

$$\mathrm{rel} = \frac{1}{n}\sum_{p}^{n} \frac{|y_p - \hat{y}_p|}{y_p} \tag{8}$$

$$\mathrm{a_i} = \%\ of\ \hat{y}_p\ s.t.\ \max\left(\frac{y_p}{\hat{y}_p}, \frac{\hat{y}_p}{y_p}\right) = \mathrm{a_i} < threshold = 1.25^i \quad \text{for}\ \ i = 1, 2, 3 \tag{9}$$

$$\mathrm{rmse} = \sqrt{\frac{1}{n}\sum_{p}^{n} \left(y_p - \hat{y}_p\right)^2} \tag{10}$$

where y_p is the true value, $\hat{y}_p$ is the predicted value, and n is the number of pixels in a depth map.

4.2 Performance Comparison

Tab. 1 presents a performance comparison between our method with other monocular depth estimation methods. To facilitate a clear understanding of the differences, we categorized the results in Tab. 1 based on the pre-training dataset, fune-tuning status, and learning methods.

In general, supervised learning methods perform better than unsupervised learning methods. However, both DepthCLIP and DepthBLIP-2, which are guided by semantic language knowledge, demonstrate performance that surpasses the baseline for unsupervised methods. Our proposed DepthBLIP-2 achieves performance comparable to DepthCLIP [34], with noticeable improvements in evaluation metrics such as rel. These improvements is likely attributable to the

Table 1: Comparison of monocular depth estimation performance on NYU Depth V2.

Method	Pre-training	Fine-tune	$a_1 \uparrow$	$a_2 \uparrow$	$a_3 \uparrow$	rel $\downarrow$	$\log_{10} \downarrow$	rmse $\downarrow$
Make3D [25]	-	-	0.447	0.745	0.897	0.349	-	1.214
DORN [4]	-	-	0.828	0.965	0.992	0.115	0.051	0.509
ASTransformer [2]	-	-	0.902	0.985	0.997	0.103	0.044	0.374
DepthFormer [18]	-	-	0.921	0.989	**0.998**	0.096	0.041	0.339
RPSF [22]	-	-	**0.952**	**0.989**	0.997	**0.072**	**0.029**	**0.267**
Lower Bound	-	-	0.140	0.297	0.471	1.327	0.323	2.934
vid2depth [21]	KITTI [5]	0-shot	0.268	0.507	0.695	0.572	-	1.637
Zhang el al. [33]	KITTI [5]	0-shot	0.350	0.617	0.799	0.513	0.529	1.457
DepthCLIP [34]	CLIP [24]	0-shot	0.394	0.683	0.851	0.388	0.156	1.167
DepthBLIP-2	BLIP-2 [14]	few-shot	**0.401**	**0.707**	**0.898**	**0.363**	**0.153**	**1.132**

multimodal information fusion capabilities of Q-Former and the adaptive depth bin mechanism. Moreover, our DepthBLIP-2 has narrowed the gap with earlier supervised learning methods, such as Make3D [25]. These results further validate BLIP-2 as a robust foundation model for general-purpose domains, enhancing our confidence in its applicability.

4.3 Depth Prediction Visualization

To assess DepthBLIP-2's capability to comprehend depth information, we visualized its depth predictions and compared them with the corresponding RGB input images and ground truth data.

As illustrated in Fig. 2, we found a situation similar to DepthCLIP, where the prediction result exhibits a degree of blurriness. The model tends to focus on specific detailed objects, such as bathtubs in the first and second rows, and pots in the third row. This phenomenon is more prevalent in scenarios where semantic language knowledge guides depth estimation. As this stage, it remains unclear whether this heightened attention to certain objects is due to the limitations of semantic knowledge in capturing depth information. Another possibility, as suggested by the authors of DepthCLIP, is that the image encoder of the visual-language model, which was pre-trained on image classification tasks, tends to overemphasize detailed local features. Further investigation will be needed to determine the exact cause in future work.

4.4 Semantic Distance Classes Design

To assess the influence of different semantic distance class designs on performance, we evaluated our method across all scenes of NYU Depth V2 [23]. As shown in Tab. 2, the upper section presents various design schemes for semantic distance classes, while the lower section provides the corresponding evaluation metrics under each scheme. The results from the original class design are those reported in Tab. 1.

Table 2: Ablations of different semantic distance classes.

Class number	Prompt Design of Different Semantic Distance Class
Original class	[giant, extremely close, close, not in distance, a little remote, far, unseen]
Class 1	[extremely close, close, middle, a little far, far, quite far, unseen]
Class 2	[extremely close, very close, close, a little close, a little far, far, unseen]
Class 3	[giant, close, a little close, not in distance, a bit remote, far, unseen]

Class number	$a_1 \uparrow$	$a_2 \uparrow$	$a_3 \uparrow$	rel $\downarrow$	$\log_{10} \downarrow$	rmse $\downarrow$
Original class	**0.401**	**0.707**	**0.898**	0.363	**0.153**	**1.132**
Class 1	0.389	0.693	0.896	0.368	0.155	1.144
Class 2	0.395	0.690	0.894	0.361	**0.153**	1.154
Class 3	0.392	0.692	0.887	**0.359**	0.156	1.148

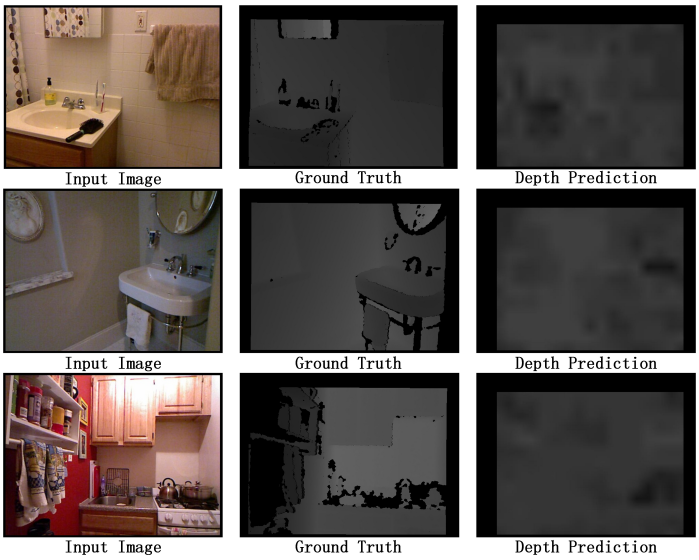

Fig. 2: Visualization of depth prediction, compared with RGB images and ground truth.

Table 3: Comparison of DepthBLIP-2 under different fine-tuning strategies.

Method	Fine-tune part	Depth bin	$a_1 \uparrow$	$a_2 \uparrow$	$a_3 \uparrow$	rel $\downarrow$	$\log_{10} \downarrow$	rmse $\downarrow$
DepthCLIP	0-shot	prefix	0.394	0.683	0.851	0.388	0.156	1.167
DepthBLIP-2	0-shot	prefix	0.384	0.683	0.856	0.368	0.156	1.172
DepthBLIP-2	Q-Former	prefix	0.388	0.687	0.857	0.377	0.158	1.162
DepthBLIP-2	MLP	auto	**0.401**	**0.707**	**0.898**	**0.363**	**0.153**	**1.132**

Our findings indicate that varying the expressions used for semantic distance classes had only a minor effect on performance. We attribute this to the model's inherent ability to recognize the semantic similarities across different expressions. For example, terms like "far" and "a little far" convey closely related meanings, resulting in only slight differences in how the model interprets them, with minimal impact on the overall performance. This suggests that it is not necessary to over-engineer the design of semantic distance classes, an appropriately selected set of expressions is sufficient to provide robustness and maintain performance. These findings offer valuable insights for our future work, where we will focus on further refining this aspect to enhance model performance.

4.5 The Necessity Of Adaptive Depth Bin

To validate the effectiveness of adaptive depth bin in depth estimation tasks, we compared the performance across various fine-tuning strategies. Traditional task adaptation methods typically require fine-tuning the entire model. In this work, we focused on selectively fine-tuning the Q-Former and the MLP module responsible for adaptive depth bin within the BLIP-2 [14] model to better suit downstream depth estimation tasks.

Tab. 3 presents a comparison of the DepthBLIP-2 model's performance under different fine-tuning strategies. Although the BLIP-2 model exhibited strong generalization capabilities in zero-shot inference, performing similarly to DepthCLIP on the NYU Depth V2 dataset, fine-tuning either the Q-Former or the MLP module further improved its performance. This indicates that targeted fine-tuning of specific components within BLIP-2, even with limited data, can significantly enhance depth estimation accuracy while maintaining an efficient model size.

The experimental data clearly show that integrating the MLP module to implement adaptive depth bin led to substantial improvements across all performance metrics. For instance, the a_1 metric increased from 0.384 to 0.401, and the rmse decreased from 1.172 to 1.132. Compared to fine-tuning the Q-Former, introducing the MLP module effectively reduces training costs. Unlike the fixed depth bin strategy used in DepthCLIP, adaptive depth bin allows the model to better handle variations in scene composition, resulting in improved generalization. This result confirms that the adaptive depth bin is crucial for improving model performance.

In conclusion, the experimental results strongly support the effectiveness of the adaptive depth bin. By fine-tuning the MLP module to create an adaptive depth bin, we can enhance depth estimation accuracy while significantly reducing training costs. This expands the potential applications of the BLIP-2 model in depth estimation tasks. Additionally, the performance similarities between DepthBLIP-2 and DepthCLIP in zero-shot inference are further analyzed in the first section of supplementary material. The second section of supplementary material presents experimental results of DepthBLIP-2 on additional datasets, further validating the broad applicability and effectiveness of the adaptive depth bin.

Table 4: Ablations of multiple depth bin in different scenes.

Depth bin	Values of depth bin (in meters)
Depth bin 1	[1.00, 1.75, 2.25, 2.50, 2.75, 3.00, 3.50]
Depth bin 2	[1.00, 1.50, 2.00, 2.25, 2.50, 2.75, 3.00]
Depth bin 3	[1.00, 1.25, 1.50, 1.75, 2.00, 2.25, 2.50]
Depth bin 4	[2.00, 2.50, 3.00, 3.25, 3.50, 3.75, 4.00]
Depth bin 5	Adaptive depth bin

Scene: bedroom	$a_1 \uparrow$	$a_2 \uparrow$	$a_3 \uparrow$	rel $\downarrow$	$\log_{10} \downarrow$	rmse $\downarrow$
Depth bin 1	0.435	0.762	0.927	0.301	0.131	0.903
Depth bin 2	0.392	0.710	0.906	**0.299**	0.143	0.967
Depth bin 3	0.272	0.550	0.793	0.342	0.188	1.165
Depth bin 4	0.415	0.713	0.876	0.443	0.147	1.003
Depth bin 5	**0.444**	**0.782**	**0.932**	0.309	**0.128**	**0.877**
Scene: kitchen	$a_1 \uparrow$	$a_2 \uparrow$	$a_3 \uparrow$	rel $\downarrow$	$\log_{10} \downarrow$	rmse $\downarrow$
Depth bin 1	0.339	0.607	0.801	0.474	0.177	1.179
Depth bin 2	0.330	0.622	0.808	0.425	0.175	1.186
Depth bin 3	0.300	0.581	0.784	**0.382**	0.190	1.286
Depth bin 4	0.264	0.504	0.687	0.750	0.221	1.405
Depth bin 5	**0.355**	**0.638**	**0.818**	0.439	**0.171**	**1.153**
Scene: bathroom	$a_1 \uparrow$	$a_2 \uparrow$	$a_3 \uparrow$	rel $\downarrow$	$\log_{10} \downarrow$	rmse $\downarrow$
Depth bin 1	0.404	0.704	0.858	0.453	0.152	0.793
Depth bin 2	0.440	0.743	0.889	0.381	0.138	0.721
Depth bin 3	0.430	0.728	0.914	0.312	0.136	0.717
Depth bin 4	0.204	0.433	0.673	0.834	0.236	1.308
Depth bin 5	**0.441**	**0.750**	**0.898**	0.361	**0.135**	**0.704**

4.6 Robustness Of Adaptive Depth Bin

Different depth bin partitions correspond to the quantized distances of different semantic distance categories. To further explore the relationship between predefined depth bin and performance, as well as to assess the robustness of the adaptive depth bin across varying scenes, we conducted detailed experiments on specific scenes from the NYU Depth V2 [23]. As presented in Tab. 4, Depth bin 1 to Depth bin 4 represent predefined depth bin, while Depth bin 5 corresponds to the adaptive depth bin obtained after fine-tuning a subset of the data. We selected three representative scenes: bedroom, kitchen, and bathroom.

When focusing on the experimental results of the predefined depth bin, we observed significant performance fluctuations across different scenes. This suggests that applying a single predefined depth bin partition uniformly across all scenes leads to performance degradation in certain scenarios. In other words, the predefined depth bin strategy is sensitive to scene-specific characteristics, a result consistent with previous findings in DepthCLIP [34].

However, when the adaptive depth bin (Depth bin 5) was employed, it consistently achieved the best or second-best performance across multiple evaluation metrics in all tested scenes. These results demonstrate that the adaptive depth bin offers strong robustness to scene variations. The proposed adaptive depth bin effectively mitigates performance loss caused by changes in scene characteristics, a crucial factor for the advancement of language-guided depth estimation research.

5 Limitations

Although our work builds upon BLIP-2 [14] and introduces several improvement strategies, there are still some limitations to address. First, while our method employs adaptive depth bin, it remains an open question whether there are more effective alternatives for learning adaptive depth bin that could further enhance performance. Second, the language prompts used in this study are handcrafted, and recent research [35] has demonstrated that the design of prompts can significantly influence model performance (Preliminary exploration of prompt learning are analyzed in the third section of supplementary material). Finally, although we use Q-Former to replace the dot-product operation to optimize the fusion of information features, the image encoder in BLIP-2 still tends to focus more on specific objects, while paying less attention to broader scene contexts such as the background.

6 Conclusion

In this paper, we propose DepthBLIP-2, which draws inspiration from DepthCLIP [34] and leverages semantic language knowledge for depth estimation. We successfully adapted this approach to the BLIP-2 [14] framework, addressing certain limitations and achieving satisfactory results.

The rise of visual-language models as foundational models for general domains is becoming increasingly evident. The architecture of BLIP-2 has the potential to serve as a foundational model across a variety of fields. We believe that guiding visual-language models with semantic language input to perform visual tasks is a promising direction for future research. It is our hope that this work will help advance research in this area, leading to higher performance and broader applications.

Acknowledgements Thanks to [13] for the open-source code that has been helpful in our research. This work is supported by the Key R & D projects in Hubei Province under Grant No.2021BAA184 and 2021BAA188. Additionally, this research is also supported by the Hubei Provincial Key R & D Project on "Key Technologies and Application Demonstration of Smart City Data Governance Based on Large Models" and the Hubei Provincial Major R & D Project on "Development of Key Technologies for Universal Interfaces for Massive Multi-Source Heterogeneous Data Fusion and Interaction."

References

1. Bhat, S.F., Alhashim, I., Wonka, P.: Adabins: Depth estimation using adaptive bins. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4009–4018 (2021)
2. Chang, W., Zhang, Y., Xiong, Z.: Transformer-based monocular depth estimation with attention supervision. In: BMVC. vol. 6, p. 7 (2021)
3. Cho, J., Lei, J., Tan, H., Bansal, M.: Unifying vision-and-language tasks via text generation. In: International Conference on Machine Learning (2021)
4. Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D.: Deep ordinal regression network for monocular depth estimation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2002–2011 (2018)
5. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The International Journal of Robotics Research **32**, 1231 – 1237 (2013)
6. Gu, X., Lin, T., Kuo, W., Cui, Y.: Zero-shot detection via vision and language knowledge distillation. CoRR **abs/2104.13921** (2021)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
8. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q.V., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: International Conference on Machine Learning (2021)
9. Jiang, H., Ding, L., Hu, J., Huang, R.: Plnet: Plane and line priors for unsupervised indoor depth estimation. In: 2021 International Conference on 3D Vision (3DV). pp. 741–750. IEEE (2021)
10. Kim, W., Son, B., Kim, I.: Vilt: Vision-and-language transformer without convolution or region supervision. In: International Conference on Machine Learning (2021), `https://api.semanticscholar.org/CorpusID:231839613`
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Communications of the ACM **60**, 84 – 90 (2012)
12. Li, B., Weinberger, K.Q., Belongie, S.J., Koltun, V., Ranftl, R.: Language-driven semantic segmentation. CoRR **abs/2201.03546** (2022)
13. Li, D., Li, J., Le, H., Wang, G., Savarese, S., Hoi, S.C.: LAVIS: A one-stop library for language-vision intelligence. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations). pp. 31–41. Association for Computational Linguistics, Toronto, Canada (Jul 2023)
14. Li, J., Li, D., Savarese, S., Hoi, S.C.H.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In: International Conference on Machine Learning (2023)
15. Li, J., Li, D., Xiong, C., Hoi, S.C.H.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: International Conference on Machine Learning (2022)
16. Li, J., Selvaraju, R.R., Gotmare, A.D., Joty, S.R., Xiong, C., Hoi, S.C.H.: Align before fuse: Vision and language representation learning with momentum distillation. In: Neural Information Processing Systems (2021)
17. Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., et al.: Grounded language-image pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10965–10975 (2022)

18. Li, Z., Chen, Z., Liu, X., Jiang, J.: Depthformer: Exploiting long-range correlation and local information for accurate monocular depth estimation. Machine Intelligence Research **20**, 837 – 854 (2022)
19. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2017), `https://api.semanticscholar.org/CorpusID:53592270`
20. Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., Li, T.: Clip4clip: An empirical study of clip for end to end video clip retrieval. Neurocomputing **508**, 293–304 (2021)
21. Mahjourian, R., Wicke, M., Angelova, A.: Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5667–5675 (2018)
22. Mel, M., Siddiqui, M.I., Zanuttigh, P.: End-to-end learning for joint depth and image reconstruction from diffracted rotation. The Visual Computer pp. 1–17 (2022)
23. Nathan Silberman, Derek Hoiem, P.K., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: ECCV (2012)
24. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
25. Saxena, A., Sun, M., Ng, A.Y.: Make3d: Depth perception from a single still image. In: Aaai. vol. 3, pp. 1571–1576 (2008)
26. Tang, C., Hou, C., Song, Z.: Depth recovery and refinement from a single image using defocus cues. Journal of Modern Optics **62**, 441 – 448 (2015)
27. Tsai, Y.M., Chang, Y.L., Chen, L.G.: Block-based vanishing line and vanishing point detection for 3d scene reconstruction. In: 2006 international symposium on intelligent signal processing and communications. pp. 586–589. IEEE (2005)
28. Vaswani, A., Shazeer, N.M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Neural Information Processing Systems (2017)
29. Vyas, P., Saxena, C., Badapanda, A., Goswami, A.: Outdoor monocular depth estimation: A research review. ArXiv **abs/2205.01399** (2022)
30. Wang, W., Bao, H., Dong, L., Bjorck, J., Peng, Z., Liu, Q., Aggarwal, K., Mohammed, O.K., Singhal, S., Som, S., Wei, F.: Image as a foreign language: Beit pretraining for all vision and vision-language tasks. ArXiv **abs/2208.10442** (2022)
31. Wang, Z., Yu, J., Yu, A.W., Dai, Z., Tsvetkov, Y., Cao, Y.: Simvlm: Simple visual language model pretraining with weak supervision. ArXiv **abs/2108.10904** (2021)
32. Xu, J., De Mello, S., Liu, S., Byeon, W., Breuel, T., Kautz, J., Wang, X.: Groupvit: Semantic segmentation emerges from text supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18134–18144 (2022)
33. Zhang, M., Ye, X., Fan, X., Zhong, W.: Unsupervised depth estimation from monocular videos with hybrid geometric-refined loss and contextual attention. Neurocomputing **379**, 250–261 (2020)
34. Zhang, R., Zeng, Z., Guo, Z.: Can language understand depth? Proceedings of the 30th ACM International Conference on Multimedia (2022)
35. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. International Journal of Computer Vision (IJCV) (2022)