Unveiling the Efficacy of Foundation Models for Depth Estimation

Pin-Ying Wu, **Zih-Hao Fu**

University of California San Diego, USA

Abstract

Monocular depth estimation is critical for 3D reasoning of a scene. While inferring from 2D to 3D is an ill-posed problem, it is challenging to recover the depth of a scene given the scarcity of depth supervisions. In this project, we aim to investigate the feasibility of using large foundation models, such as Contrastive Language-Image Pre-training (CLIP) [21] and Segment Anything (SAM) [15], for depth estimation in computer vision. Inspired by [35], we first formulate depth estimation as a distance classification task so that distance can be *inferred* from CLIP with semantic language tokens, which serves as the initial depth prediction. We then incorporate adapter networks to study whether these refinement modules can further improve the CLIP predictions, including CLIP-Adapter [7] and the proposed Multi-Scale Adapter (MSA). In addition, we investigate the efficacy of SAM for depth estimation by integrating its output into the multi-scale adapter. We conduct thorough experiments and ablation studies under both self-supervised and supervised settings with the NYU-Depth v2 Dataset [25]. Experimental results suggest that, while showing promising performance on classification and segmentation tasks, current visual foundation models still suffer from 2D-to-3D reasoning and fail to address the challenge of depth estimation in computer vision.

1 Introduction

Depth estimation from a single RGB image is a fundamental task in computer vision with applications in autonomous driving [29, 33, 32], robotics [18, 10], and augmented reality [14, 3]. Accurate depth estimation enables a range of important functionalities, such as obstacle detection [17, 31], scene understanding [2, 12], and object tracking [13]. Traditional approaches for depth estimation rely on supervised learning techniques that require large amounts of depth-labeled data, which can be expensive and time-consuming to acquire. To address this challenge, recent advancements in large-scale pre-trained models have shown promising results in various computer vision tasks. One such model is Contrastive Language-Image Pre-training (CLIP) introduced by Radford et al. [21]. CLIP is a powerful foundation model that has been pre-trained on a large corpus of text and image pairs. It learns to associate images and their textual descriptions, enabling it to understand the semantic relationship between the two modalities.

Moreover, while CLIP has been successfully applied to various vision tasks, its utilization has primarily focused on high-level visual recognition tasks. The potential of leveraging its pre-trained semantic language knowledge for quantitative vision tasks, such as depth estimation, remains largely unexplored. Monocular depth estimation plays a crucial role in various industrial applications, including monocular 3D object detection and point cloud reconstruction from images. Typically, this task requires dense depth labels to train a network to extract semantic relationships within an image and regress pixel-wise depth values. However, training networks from scratch using dense labels can be inefficient for deployment due to the high data collection and annotation cost, particularly for large-scale datasets like NYU Depth Dataset V2 [25] and KITTI [8].

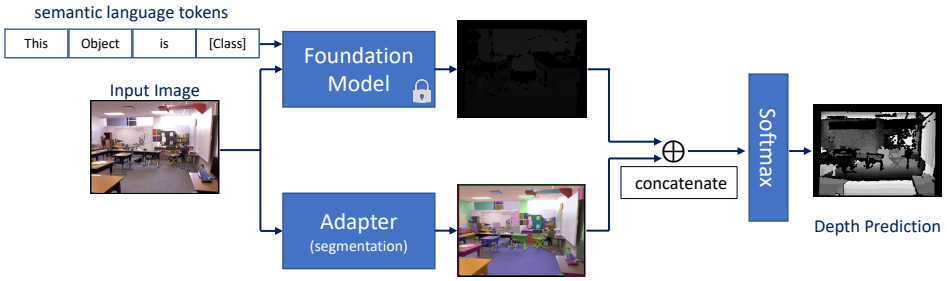

Figure 1: This project aims at investigating the efficacy of foundation models for depth estimation, where CLIP is adopted to generate an initial depth prediction, and various adapter networks are leveraged to refine the depth prediction. Segmentation from SAM is also integrated into the adapter design as a prior information.

Some existing unsupervised methods address this challenge by incorporating additional data, such as single-view videos [16] to capture time consistency or 3D priors for improved spatial modeling. However, these approaches still come with their own limitations and requirements. Therefore, we pose the question: Is it possible to prevent the costs associated with training models and collecting data by leveraging the semantic language knowledge learned by CLIP? In our research, we aim to explore whether the semantic language knowledge acquired by CLIP can be leveraged to alleviate the challenges of monocular depth estimation. By utilizing the semantic understanding encoded within CLIP, we aim to analyze whether CLIP is helpful for depth estimation and can reduce the dependence on extensive data collection and expensive model training and pave the way for a more efficient and cost-effective approach to this task.

This project investigates the feasibility of leveraging CLIP for depth estimation from monocular RGB images. The pretrained visual encoder of CLIP is designed to encode images into a feature map before applying a pooling layer, where the feature map retains valuable semantic details at each location and allows to capture local visual information. Our approach exploits this property by associating each spot on the feature map with a depth approximation based on its response to semantic language tokens. By doing so, we leverage the strengths of CLIP in understanding the semantic relationship between images and text, facilitating depth estimation without requiring a large amount of depth-labeled data. Through extensive experimentation and evaluation of benchmark datasets, we aim to analyze the effectiveness of foundation models, including CLIP and SAM, on depth estimation tasks. Our contributions lie in thorough experiments and ablation studies to conclude that CLIP and SAM are not very helpful for depth estimation tasks. This highlights the need of foundation models that can really capture 2D-to-3D information and would be insightful for the community.

2 Related Work

Deep learning-based depth estimation is a computer vision task that involves predicting the depth information of a scene from an input image or a sequence of images. This technique finds applications in diverse fields such as robotics, autonomous driving, augmented reality, and 3D reconstruction. In this section, we provide an overview of the key approaches and advancements in the field of deep learning-based depth estimation, including recent advancements related to CLIP [21].

2.1 Supervised and Unsupervised Learning Monocular Depth Estimation

Supervised learning-based methods have made remarkable strides in depth estimation since the rise of deep learning. In 2014, Eigen et al. introduced one of the pioneering deep learning models for depth estimation, utilizing a convolutional neural network (CNN) to predict depth from individual images. They proposed Depth_eigen [4], a multi-scale deep network that effectively makes a coarse global prediction based on the entire image, and another that refines this prediction locally, resulting in precise depth maps from a single image. However, supervised learning methods require a large amount of accurately labeled depth data for training. This labeling process can be time-consuming, expensive, and requires expert knowledge. In addition, supervised learning models exhibit limited generalization capability, often struggling to extend their performance to unseen or novel environments and objects.

Since they heavily rely on the patterns and information present in the training dataset. As a result, the model's performance may degrade when faced with data that significantly differs from the training distribution, leading to inaccurate depth predictions.

To overcome this challenge, researchers have explored unsupervised learning approaches for depth estimation. In 2017, Zhou et al. introduced a self-supervised learning method [37] simultaneously training depth and camera pose estimation models, which utilized a Depth CNN to generate depth maps and a Pose CNN to estimate camera poses. The two models are trained jointly by minimizing the reconstruction error obtained from the generated depth maps and the pose estimation. This self-supervised framework enables the models to learn depth cues and camera poses without requiring explicit depth annotations. Also, the models are more likely to generalize well to unseen or novel environments than supervised learning-based models.

2.2 Semantic Information and Depth Estimation

Semantic information has been considered to improve the performance of depth estimation.Segmentation can help identify object boundaries, which are important cues for depth estimation because depth discontinuities often occur at object boundaries, and understanding these boundaries can help to improve the accuracy of depth predictions. Moreover, segmentation provides priors and constraints for depth estimation. The same object category often exhibits consistent depth characteristics. For example, knowing that a certain region corresponds to a road or a sky can help constrain the depth range for that area. By incorporating such priors and constraints, depth estimation models can regularize their predictions and produce more plausible depth maps. Therefore, many researchers focus on fusing semantic information into the original depth prediction model to improve depth estimation. In 2021, Zhou et al. proposed DIFFNet [36] based on the well-developed segmentation model HRNet [26, 28, 34], using semantic information to help improve depth estimation.

2.3 CLIP and Depth Estimation

Recently, the Contrastive Language-Image Pretraining (CLIP) [21] model has gained attention in the computer vision community. Radford et al. developed CLIP as a powerful vision-and-language model that learns to understand images and their associated textual descriptions in 2021. Although CLIP is primarily designed for tasks like image classification and retrieval, its pre-training framework has shown potential for transfer learning to other vision tasks, including depth estimation. Researchers have started exploring the application of CLIP for depth estimation by leveraging its cross-modal understanding capabilities. By conditioning the CLIP model on depth-related textual prompts, it becomes possible to infer depth information from images. In 2022, Zhang et al. proposed DepthCLIP [35], which transforms the depth value regression task into a distance classification task that can be handled by CLIP. Instead of requiring precise depth values (e.g., "The object is 5 meters away"), DepthCLIP makes the depth estimation into a classification problem by asking CLIP to predict whether an object is close or far from the viewer (e.g., giant, close, far, etc.). This methodology enables us to leverage the power of the CLIP model while reducing the need for large amounts of depth-supervised data.

However, the current performance of DepthCLIP is unsatisfactory. The model's accuracy remains low when relying solely on pre-trained CLIP without any fine-tuning or additional supervised data. This raises doubts about the suitability of CLIP for the depth estimation problem. Therefore, in this project, our goal is to introduce a different adaptor that provides supplementary information, such as semantic information using SAM [15], and fuse it with CLIP features. We aim to investigate whether this approach can improve the accuracy of the model. Ultimately, we hope to determine whether CLIP can be effectively applied to depth estimation tasks.

3 Preliminaries

3.1 Large Foundation Models

Large foundation models have revolutionized the fields of computer vision and natural language processing (NLP) in recent years and enabled significant advancements in various applications. For example, Contrastive Language-Image Pre-training (CLIP) [21] leverages 400M image-text pairs to learn generalized representations, having garnered considerable attention due to its transferability to

various downstream tasks and datasets. CLIP employs an image encoder to capture visual patterns and semantic information and demonstrates a remarkable ability to perform tasks such as image classification [19], object detection [27], and visual question answering [5, 24] by combining this visual understanding with its textual knowledge. On the other hand, Segment Anything (SAM) [15] is a recent foundation model that harnesses a large segmentation dataset, SA-1B, with interactive prompts, to generate segmentation for any given images. With these foundation models, there has been multiple research to study whether we can really exploit the power of them and reduce the cost and complexity of data collection for specific downstream domains and tasks. Our project aims to investigate the efficacy of large foundation models for the depth estimation task in computer vision.

3.2 Adapter Networks

Adapters serve as small and lightweight modules that enable the integration of new capabilities into existing pretrained foundation models without requiring extensive retraining or modifying of the parameters. This provides an efficient modular extension of large foundation models. In other words, to address a specific downstream task, people can harness the power of foundation models and only train a relatively lightweight adapter with domain-specific data. With the advent of foundation models, there have been several studies about adapter networks. A pioneer work is the CLIP-adapter [7] that utilizes a two-layer MLP module to learn an adaptation for image and text embedding respectively. While showing promising improvement in image recognition tasks, there is no work studying whether this invention is applicable to the depth estimation task in computer vision. Motivated by this, we conduct thorough experiments and ablations under both self-supervised and supervised settings on adapter networks for depth estimation, where the depth generated by CLIP serves as the initial prediction and the adapters act as refinement modules. In addition to CLIP-adapter [7], we further propose a multi-scale adapter (MSA) that combines multi-scale features from different layers of CLIP to capture coarse-to-fine information from the scene and fuse local information with the high-level features for global context and semantics. We also incorporate the segmentation from the recent advent of SAM to MSA and study its efficacy.

3.3 CLIP-based Depth Estimation

CLIP [21] consists of an image encoder $VE(\cdot)$ and a text encoder $TE(\cdot)$, which extracts visual and text encodings respectively. CLIP-based depth estimation formulates a classification task and relies on these encodings.

Image Encoding. Given a monocular RGB image $\mathbf{I} \in \mathbb{R}^{H\times W\times 3}$, visual encoder $VE(\cdot)$ (without the final pooling layer) takes it as the input and extracts a C-dimensional feature map $\mathbf{F}_{img}$, i.e.,

$$\mathbf{F}^{img} = VE(\mathbf{I}) \in \mathbb{R}^{H\times W\times C}, \tag{1}$$

where H and W denote the height and width of $\mathbf{I}$, and C the number of channels (or the hidden dimension) of the features. Since the feature map $\mathbf{F}^{img}$ preserves the spatial information as in the original image, each pixel in $\mathbf{F}^{img}$ can be mapped into distance categories based on its similarity to semantic language tokens from $TE(\cdot)$. Note that each pixel in the feature map before pooling captures regional semantic information, and the pooling operation aggregates local knowledge to foster a global interpretation of the given image.

Text Encoding. The CLIP text encoder $TE(\cdot)$ projects similar semantic tokens to the neighborhood of image features due to the contrastive pre-text task. Following DepthCLIP [35], we formulate depth estimation as a distance classification task and utilize text prompts in the template of "This object is [distance class]", where [distance class] contains K classes, i.e., ["giant", "extremely close", "close", "not in distance", "a little remote", "far", "unseen"], corresponding to semantic tokens $\mathbf{T}$.

$$\mathbf{F}^{text} = TE(\mathbf{T}) \in \mathbb{R}^{K\times C}. \tag{2}$$

Depth Prediction. The depth prediction with CLIP is implemented by computing the similarity score $\mathbf{S} \in \mathbb{R}^{H\times W\times K}$, where each entry S^k_{ij} is the cosine similarity between the k-th predefined semantic tokens $\mathbf{F}^{text}_k \in \mathbb{R}^C$ and each pixel in the image features $\mathbf{F}^{img}_{ij} \in \mathbb{R}^C$, i.e.,

$$S^k_{ij} = \frac{\mathbf{F}^{text}_k \cdot \mathbf{F}^{img}_{ij}}{\|\mathbf{F}^{text}_k\|\|\mathbf{F}^{img}_{ij}\|}. \tag{3}$$

Note that each predefined distance class (or depth bin) is assigned to a physical distance d_k, e.g., "giant" and "close" corresponds to 1m and 2m, respectively. The depth estimation can then be computed as the weighted sum of these depth bins with the probability as the weights, formulated as

$$\hat{\mathbf{D}}_{ij} = \sum_{k=1}^{K} a(\mathbf{F}_{ij}^{img}, \mathbf{F}_{k}^{text}) * d_k \quad \text{where} \tag{4}$$

$$a(\mathbf{F}_{ij}^{img}, \mathbf{F}_{k}^{text}) = \frac{e^{S_{ij}^{k}/t}}{\sum_{l=1}^{K} e^{S_{ij}^{l}/t}},$$

and t is the temperature for the softmax function.

4 Exploring the Potential of Adapter Networks

In this section, we investigate whether incorporating an adapter network into CLIP can further improve depth estimation performance. To the best of our knowledge, this is the first work to study adapter networks for CLIP-based depth estimation. We first implement CLIP adapter [7] for both visual and text domains. However, as there are no convolutional blocks in the CLIP adapter, the capability of spatial reasoning may be limited. To address this, we propose a Coarse/Refined Adapter based on the Depth_Eigen architecture [4], and a U-net-like Multi-Scale Adapter (MSA) that also adopts a coarse-to-fine refinement design. Note that MSA can be further integrated with the segmentation map generated by SAM [15].

4.1 CLIP Adapter [7]

CLIP adapter [7] is implemented as a lightweight MLP, typically a two-layer fully connected layer with non-linear mapping, which provides an efficient modulation for adapting CLIP features. These additional parameters are learned with the data from the downstream domains. We employ a CLIP adapter for visual and language features respectively, as illustrated in Figure 2.

4.2 Coarse/Refined Adapter

Based on the architecture of depth_eigen [4], illustrated in Figure 3, we propose Coarse/Refined adapter by substituting the coarse encoder (i.e., the blue box) with the CLIP visual encoder. Figure 4 presents the architecture of the Coarse/Refined adapter. The coarse adapter comprises two fully-connected layers and maps the initial depth prediction generated by CLIP to coarse predictions. On the other hand, in the refining layers, we merge the coarse prediction with the CNN features extracted from the original RGB image by channel concatenation, which enables the model to recover the information that CLIP does not capture. The concatenated features are then fed to two additional convolutional layers to generate the refined prediction.

4.3 Multi-Scale Adapter (MSA)

While the Coarse/Refined Adapter is a reasonable design, the performance of the Coarse/Refined adapter underperforms the original depth_eigen model. This raises doubts about the suitability of the CLIP model for depth estimation tasks. To validate this hypothesis, we further propose a U-net-like Multi-Scale Adapter (MSA) that can be integrated with the segmentation map generated by SAM.

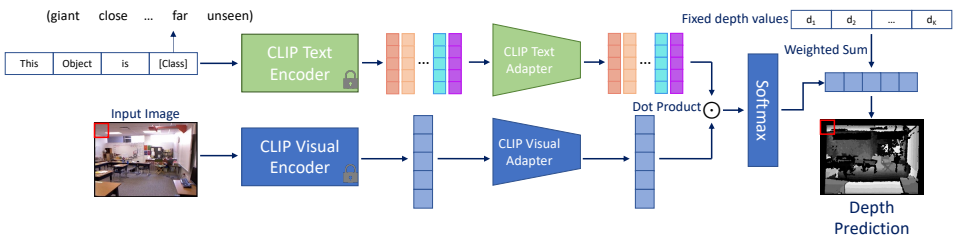

Figure 2: CLIP adapter [7].

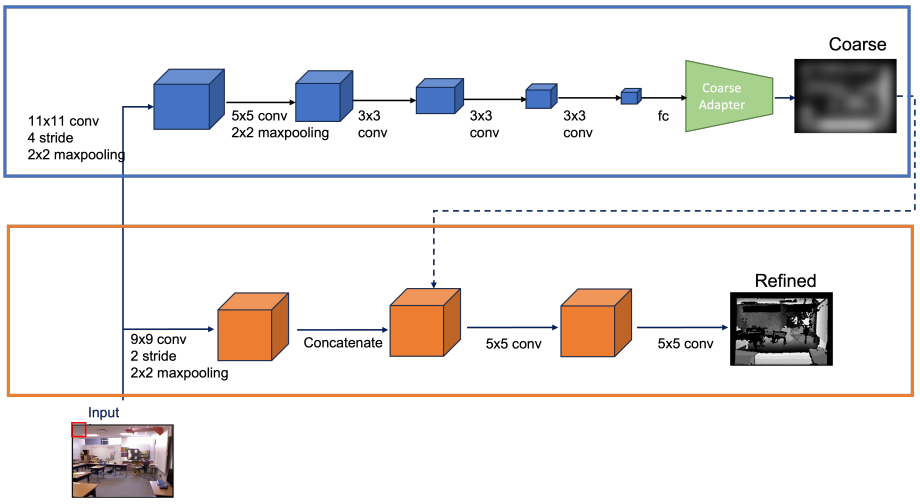

Figure 3: The structure of depth_eigen.

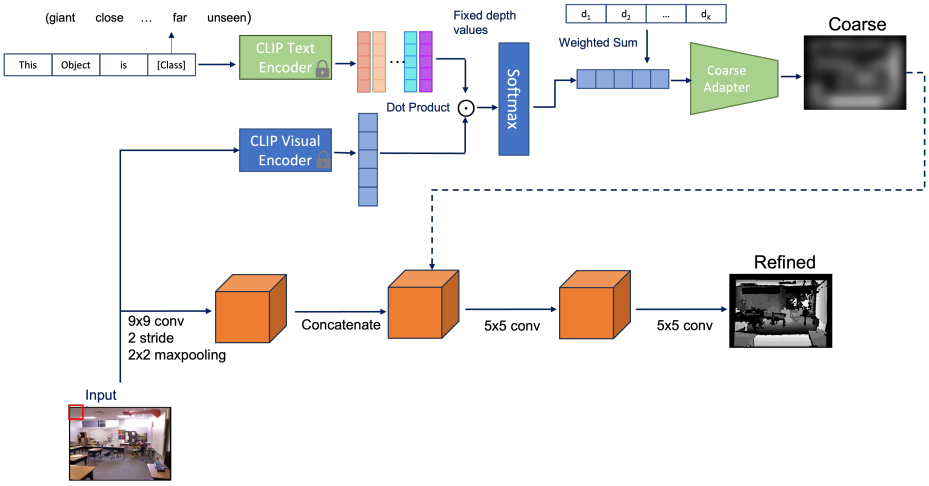

Figure 4: Coarse/Refine adapter

The architecture of the Multi-Scale Adapter is presented in Figure 5. The concept follows the idea of U-Net [22] that combines feature maps from different scales. By incorporating low-level features, the network can capture fine details and local information while leveraging high-level features for global context and semantics. We integrate multi-scale visual features extracted from different layers of the CLIP visual encoder.

In addition, motivated by the fact that the depth of the same object in the scene should have similar depth, we leverage a powerful foundation model, Segment Anything (SAM) [15] to generate a segmentation map for the original image and incorporate it to the convolutional encoders in MSA. This provides prior information of object/part segmentations that can be helpful for the depth reasoning of a scene.

4.4 Adapter Training

We study the problem under both self-supervised and supervised settings.

Self-supervised Learning (SSL). The self-supervised training follows typical techniques for self-supervised depth estimation [20, 9]. This is achieved by minimizing the photometric error between the target view $\mathbf{I}_t \in \mathbb{R}^{H\times W}$ and the recovered target view $\hat{\mathbf{I}}_t \in \mathbb{R}^{H\times W}$ reconstructed by the inverse warping of the source view $\mathbf{I}_s \in \mathbb{R}^{H\times W}$ with depth estimation $\hat{\mathbf{D}}_t \in \mathbb{R}^{H\times W}$ and the camera transformation matrix $\mathbf{T}_{t\to s} = [\mathbf{R}|\mathbf{t}]$. Based on the two-view geometry, the correspondence between

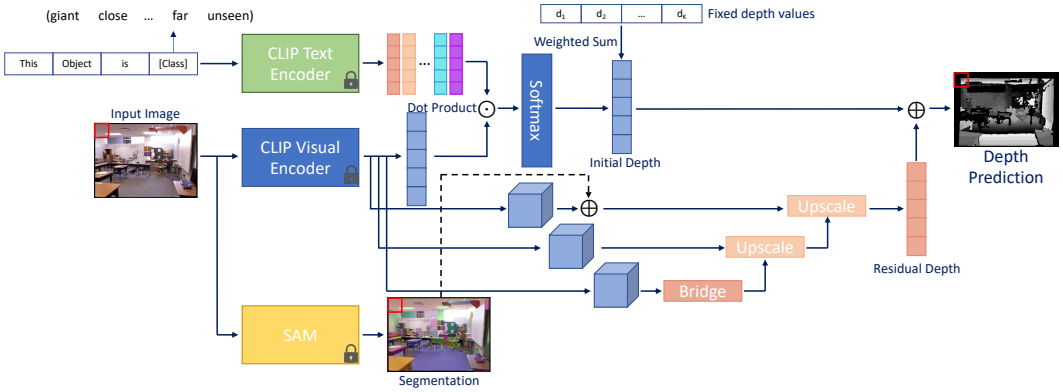

Figure 5: **Multi-Scale Adapter (MSA).** CLIP [21] visual and text encoders take a RGB image and semantic distance tokens as input respectively. The segmentation information of the image generated by Segment Anything (SAM) [15] is integrated as a prior. The multi-scale features from CLIP and the features of the segmentation map are fused with a U-net like architecture.

the source view and the target view is formulated as

$$\tilde{\mathbf{P}}_s = \mathbf{K}\mathbf{T}_{t\to s}\mathbf{D}_t(\mathbf{p}_t)\mathbf{K}^{-1}\mathbf{P}_t, \tag{5}$$

where $\mathbf{P}$ denotes the homogeneous coordinate of $\mathbf{p}$, and $\mathbf{K}$ the camera intrinsic matrix.

The photometric loss is defined as

$$L_{pe} = \alpha\frac{1 - SSIM(\mathbf{I}_t, \hat{\mathbf{I}}_t)}{2} + (1-\alpha)\|\mathbf{I}_t - \hat{\mathbf{I}}_t\| + \beta L_{smooth} \tag{6}$$

$$L_{smooth} = \frac{1}{HW}\sum_{\mathbf{P}\in\mathbf{I}_t} |\partial_x\hat{\mathbf{D}}_t(\mathbf{p})|e^{\|\partial_x I_t(\mathbf{p})\|} + |\partial_y\hat{\mathbf{D}}_t(\mathbf{p})|e^{\|\partial_y I_t(\mathbf{p})\|}, \tag{7}$$

where α and β are hyperparameters, and $SSIM(\cdot)$ denotes the structural similarity loss [30] and L_{smooth} the smoothness loss.

While self-supervised training is widely adopted for depth datasets that have stereo image pairs or consecutive frames, the NYU-Depth v2 [25] adopted in this project does not provide such information. To enable SSL using this dataset, we implement a data augmentation pipeline, including rotation and translation, to project the source frame $I_s \in \mathbb{R}^{H\times W}$ to the target frame $I_t \in \mathbb{R}^{H\times W}$, as illustrated in Figure 6.

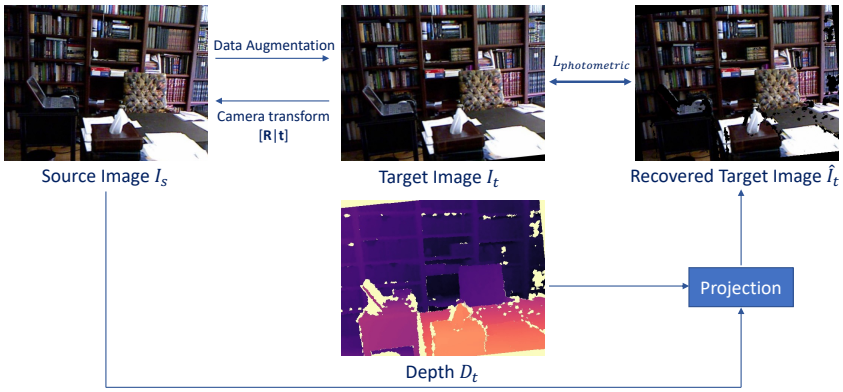

Figure 6: Pipeline of self-supervised learning on depth estimation.

Supervised Learning. We adopt the typical loss functions for supervised depth estimation tasks, including L1 loss and L2 loss for pixels defined as follows,

$$L_{L1} = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}|\hat{\mathbf{D}}_{ij} - \mathbf{D}_{ij}|, \tag{8}$$

$$L_{L2} = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}||\hat{\mathbf{D}}_{ij} - \mathbf{D}_{ij}||_2 \tag{9}$$

where $\hat{D}_{ij}$ represents the predicted depth value at pixel i, j, and D_{ij} is the corresponding ground truth depth value.

5 Experiments

5.1 Datasets

NYU Depth v2 Dataset [25] The NYU Depth v2 dataset is a popular dataset of indoor scenes. It comprises video sequences recorded by both the RGB and Depth cameras from the Microsoft Kinect. The dataset includes various indoor scenes, such as offices, bedrooms, kitchens, and living rooms, covering a wide range of objects, furniture, and room layouts. In addition to RGB color images, the dataset provides dense depth maps obtained from the Kinect camera's depth sensor. The depth maps provide per-pixel depth measurements, allowing researchers to perform depth estimation and related tasks.

The dataset follows a specific data format where RGB images are stored as jpeg files, providing the visual appearance of the scenes. Dense depth maps, representing per-pixel depth measurements, are saved as 16-bit png or tiff files. For a subset of the images, per-pixel semantic labels are available, stored as separate image files using a color-coded encoding scheme. Camera calibration parameters, which are crucial for tasks like depth map registration and camera pose estimation, are provided in accompanying text or metadata files.

5.2 Implementation Details

We implement our model with the PyTorch framework. Our image and textual encoders employ the pre-trained ResNet-50 [11] of CLIP [21]. In our study, we experimented with different hand-crafted prompts and selected "This object is [distance class]" as the prompt format. We chose a set of seven semantic distance classes: ["giant", "extremely close", "close", "not in distance", "a little remote", "far", "unseen"]. Each class corresponds to a specific depth range, which is [1.00, 1.50, 2.00, 2.25, 2.50, 2.75, 3.00]. To ensure accurate and focused predictions, we set the temperature of the final softmax function to 0.1. These choices provide a suitable framework for our main experiments, capturing indoor depth effectively while maintaining semantic understanding.

5.3 Evaluation Metrics

For the evaluation metrics, we follow previous works [1, 35, 16, 23, 6] and compare our method quantitatively with the other methods using Mean absolute relative error (rel), Root mean square error ($rmse$), Absolute error in log space ($rmse_{log}$), and Threshold accuracy (δ),

$$rel = \frac{1}{N}\sum_{p=1}^{N}\frac{|y_p - \hat{y}_p|}{\hat{y}_p} \tag{10}$$

$$rmse = \sqrt{\frac{1}{N}\sum_{p=1}^{N}(y_p - \hat{y}_p)^2} \tag{11}$$

$$rmse_{log} = \frac{1}{N}\sum_{p=1}^{N}|log_{10}(y_p) - log_{10}(\hat{y}_p)| \tag{12}$$

$$\delta = \% \text{ of } y_p \text{ s.t. } max(\frac{y_p}{\hat{y}_p}, \frac{\hat{y}_p}{y_p}) = \delta < thr \text{ for } thr = 1.25, 1.25^2, 1.25^3, \tag{13}$$

where N is the number of samples, $\hat{y}_p$ is the predicted depth, and y_p is the ground truth depth.

5.4 Quantitative Results

Model	pre-training	supervision	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$	$rel \downarrow$	$rmse_{log} \downarrow$	$rmse \downarrow$
Lower Bound	-	-	0.140	0.297	0.471	1.327	0.323	2.934
vid2depth [16]	KITTI [8]	zero-shot	0.268	0.507	0.695	0.572	-	1.637
DepthCLIP [35]	CLIP [21]	zero-shot	0.394	0.683	0.851	0.388	0.156	1.167
Make3D [23]	-	depth	0.447	0.745	0.897	0.349	-	1.214
DORN [6]	-	depth	0.828	0.965	0.992	0.115	0.051	0.509
Coarse/Refined Adapter	CLIP [21]	depth	0.394	0.695	0.873	0.354	0.418	1.121
CLIP Adapter [7]	CLIP [21]	ssl	0.369	0.667	0.848	0.353	0.161	1.205
CLIP Adapter [7]	CLIP [21]	depth	0.411	0.704	0.869	0.376	0.149	1.109
Multi-scale Adapter	CLIP [21]	ssl	0.161	0.309	0.439	0.630	0.401	1.997
Multi-scale Adapter	CLIP [21]	depth	**0.573**	**0.856**	**0.956**	**0.255**	**0.104**	**0.804**

Table 1: Performance of Monocular Depth Estimation on NYU Depth v2 [25]. We compare our methods, including CLIP adapter and Multi-scale adapter, with previous works under different settings. The supervision column indicates the used loss functions, where *ssl* indicates self-supervised learning, and *depth* indicates supervised learning.

In Table 1, we provide a comparison of our results with other monocular depth estimation methods, including supervised, self-supervised, and zero-shot learning methods pretrained on KITTI [8] dataset and CLIP [21] weights. The lower bound represents randomly generated predictions within the 0-10m depth range.

Among our proposed methods, we observe that the Multi-scale adapter using supervised learning achieves the best performance. Additionally, it is evident that methods utilizing supervised learning generally yield higher performance, while those employing self-supervised learning and zero-shot learning exhibit lower performance.

Our proposed model using supervised learning slightly outperforms DepthCLIP, but it still falls behind Make3D and DORN in terms of performance. This suggests that applying CLIP for depth estimation tasks may not be as effective as these alternative methods.

5.5 Depth Prediction Visualization

Figure 7 presents the visualization of our best-performing depth predictions, the results obtained with DepthCLIP [35], the ground truth, and the original RGB images. We can observe that our predictions are much better than DepthCLIP, where the contours and boundaries of the sink, toilet, bed, desk, chair, and table are more clear. The results are consistent with our hypothesis that adding object segmentation information helps the model learn depth prediction since the depth values of the same object are closer and change more smoothly.

5.6 Ablation Studies

CLIP as initial	loss	mode	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$	$rel \downarrow$	$rmse_{log} \downarrow$	$rmse \downarrow$
	depth (L2)	Coarse	**0.618**	**0.891**	0.969	0.228	**0.283**	**0.871**
	depth (L2)	Coarse+Refined	0.611	0.887	**0.971**	**0.215**	0.285	0.907
✓	depth (L2)	Coarse	0.217	0.431	0.612	0.444	0.986	1.601
✓	depth (L2)	Coarse+Refined	0.394	0.695	0.873	0.354	0.418	1.121

Table 2: **Ablations on Coarse/Refined adapter**.

We perform the following ablations on different adapters using different foundation model settings to understand the influence of each component.

Ablations on Coarse/Refined Adapter To evaluate the effectiveness of the CLIP model, we replaced the encoder of the depth_eigen with DepthCLIP. The results in Table Table 2 clearly demonstrate that when DepthCLIP is used as the encoder, the performance is inferior to using the

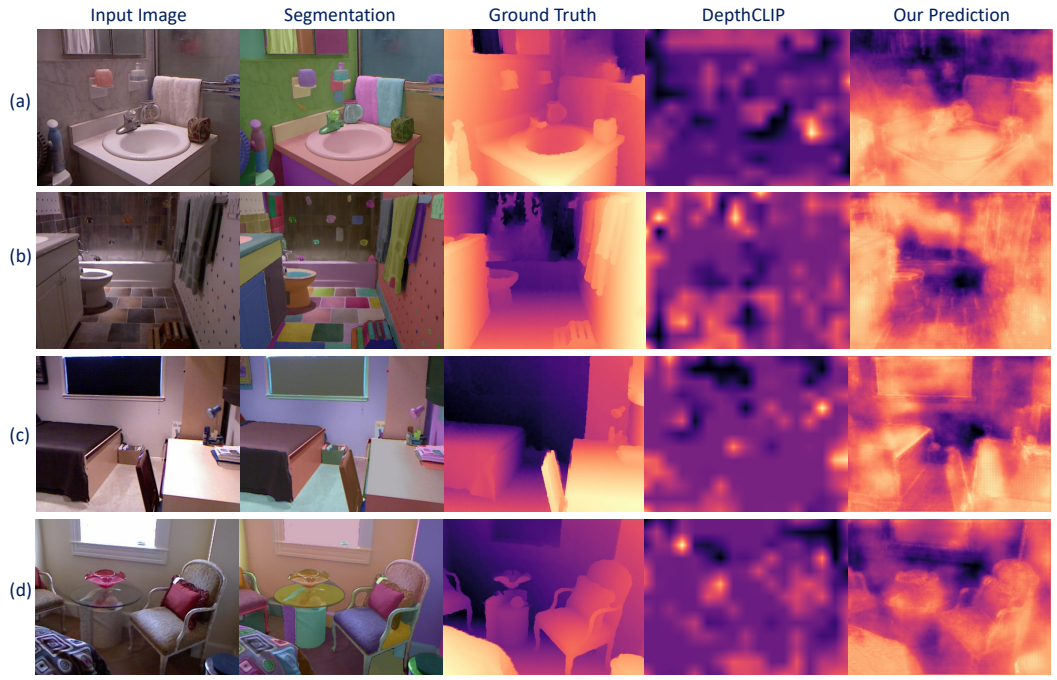

Figure 7: Visualization of our best-performing depth prediction results compared to the results of DepthCLIP [35].

CLIP	SAM	loss	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$	$rel \downarrow$	$rmse_{log} \downarrow$	$rmse \downarrow$
		ssl	0.101	0.205	0.312	0.690	0.511	2.207
	✓	ssl	0.161	0.309	0.439	0.630	0.401	1.997
✓		ssl	**0.385**	**0.678**	**0.850**	**0.386**	**0.158**	**1.177**
✓	✓	ssl	0.381	0.675	0.846	0.382	0.160	1.191
		depth (L2)	**0.456**	**0.768**	**0.920**	**0.308**	**0.130**	**0.975**
	✓	depth (L2)	0.426	0.732	0.898	0.338	0.140	1.050
✓		depth (L2)	0.385	0.674	0.846	0.398	0.159	1.184
✓	✓	depth (L2)	0.384	0.671	0.844	0.394	0.160	1.191
		depth (L1)	**0.573**	**0.856**	**0.956**	**0.255**	**0.104**	**0.804**
	✓	depth (L1)	0.576	0.856	0.955	0.254	0.104	0.808
✓		depth (L1)	0.381	0.669	0.842	0.379	0.161	1.201
✓	✓	depth (L1)	0.379	0.667	0.841	0.375	0.162	1.207

Table 3: **Ablations on CLIP and SAM** of Multi-scale adapter. The supervision column indicates the used loss functions, where *ssl* indicates self-supervised learning, and *depth* indicates supervised learning with L1 and L2 losses.

Image Adapter	Text Adapter	loss	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$	$rel \downarrow$	$rmse_{log} \downarrow$	$rmse \downarrow$
✓	✓	ssl	0.369	0.667	0.848	0.353	0.161	1.205
✓		depth (L2)	0.373	0.672	0.850	0.355	0.160	1.197
	✓	depth (L2)	0.386	0.685	0.856	0.369	0.156	1.165
✓	✓	depth (L2)	**0.411**	**0.704**	**0.869**	**0.376**	**0.149**	**1.109**

Table 4: **Ablations on image and text adapters** of CLIP adapter.

original depth_eigen. This implies that the feature maps produced by CLIP may not provide sufficient information to train a robust depth estimation model.

Ablations on CLIP and SAM To assess the efficacy of the foundation models CLIP and SAM, we also conduct experiments as shown on Table 3. We can observe that under supervised settings, the performance is better without using CLIP and SAM for both L1 and L2 loss, which concludes CLIP and SAM are not very helpful for depth estimation tasks and even harm the performance if

there are labeled data as guidance. On the other hand, the foundation models are still helpful for self-supervised settings since there is no other guidance.

Ablations on Image and Text Adapters To validate the effectiveness of image and text adapters of CLIP adapter, we conduct experiments as shown on Table 4. The results support the significance of adapters in both the visual and text domains as we can find that the best performance is achieved when utilizing both image and text adapters simultaneously as combining visual understanding with textual knowledge is crucial for depth estimation tasks.

6 Conclusion

This project explores using large foundation models like CLIP and SAM for depth estimation in computer vision. Depth estimation is crucial for 3D scene understanding, but the scarcity of depth supervisions makes it challenging. We formulate depth estimation as a distance classification task, leveraging CLIP's semantic language tokens for initial depth prediction. We incorporate adapter networks, including CLIP-Adapter, Coarse/Refined Adapter and Multi-Scale Adapter (MSA), to refine CLIP's predictions. We also investigate SAM's efficacy by integrating its output into the multi-scale adapter. Experiments using the NYU-Depth v2 Dataset reveal that current visual foundation models still struggle with 2D-to-3D reasoning and face challenges in depth estimation.

References

[1] Wenjie Chang, Yueyi Zhang, and Zhiwei Xiong. Transformer-based monocular depth estimation with attention supervision. In *BMVC*, 2021.

[2] Po-Yi Chen, Alexander H Liu, Yen-Cheng Liu, and Yu-Chiang Frank Wang. Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware representation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 2624–2632, 2019.

[3] Catherine Diaz, Michael Walker, Danielle Albers Szafir, and Daniel Szafir. Designing for depth perceptions in augmented reality. In *2017 IEEE international symposium on mixed and augmented reality (ISMAR)*, pages 111–122. IEEE, 2017.

[4] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *CoRR*, abs/1406.2283, 2014.

[5] Sedigheh Eslami, Gerard de Melo, and Christoph Meinel. Does clip benefit visual question answering in the medical domain as much as it does in the general domain? *arXiv preprint arXiv:2112.13906*, 2021.

[6] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2002–2011, 2018.

[7] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021.

[8] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.

[9] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3828–3838, 2019.

[10] Raia Hadsell, Pierre Sermanet, Jan Ben, Ayse Erkan, Marco Scoffier, Koray Kavukcuoglu, Urs Muller, and Yann LeCun. Learning long-range vision for autonomous off-road driving. *Journal of Field Robotics*, 26(2):120–144, 2009.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] Huaizu Jiang, Gustav Larsson, Michael Maire Greg Shakhnarovich, and Erik Learned-Miller. Self-supervised relative depth learning for urban scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 19–35, 2018.

[13] Longlong Jing, Ruichi Yu, Henrik Kretzschmar, Kang Li, Charles R Qi, Hang Zhao, Alper Ayvaci, Xu Chen, Dillon Cower, Yingwei Li, et al. Depth estimation matters most: improving per-object depth estimation for monocular 3d detection and tracking. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 366–373. IEEE, 2022.

[14] Megha Kalia, Nassir Navab, and Tim Salcudean. A real-time interactive augmented reality depth estimation technique for surgical robotics. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8291–8297. IEEE, 2019.

[15] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

[16] Reza Mahjourian, Martin Wicke, and Anelia Angelova. 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*, pages 5667–5675, 2018.

[17] Michele Mancini, Gabriele Costante, Paolo Valigi, and Thomas A Ciarfuglia. Fast robust monocular depth estimation for obstacle detection with fully convolutional networks. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4296–4303. IEEE, 2016.

[18] Jeff Michels, Ashutosh Saxena, and Andrew Y Ng. High speed obstacle avoidance using monocular vision and reinforcement learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 593–600, 2005.

[19] Roni Paiss, Hila Chefer, and Lior Wolf. No token left behind: Explainability-aided image classification and generation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XII*, pages 334–350. Springer, 2022.

[20] Mayank Poddar, Akash Mishra, Mohit Kewlani, and Haoyang Pei. Self-supervised learning based depth estimation from monocular images, 2023.

[21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[22] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[23] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3d: Depth perception from a single still image. In *Aaai*, volume 3, pages 1571–1576, 2008.

[24] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VIII*, pages 146–162. Springer, 2022.

[25] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. *ECCV (5)*, 7576:746–760, 2012.

[26] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019.

[27] Zhu Teng, Yani Duan, Yan Liu, Baopeng Zhang, and Jianping Fan. Global to local: Clip-lstm-based object detection from remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13, 2021.

[28] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *TPAMI*, 2019.

[29] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8445–8453, 2019.

[30] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[31] Andreas Wedel, Uwe Franke, Jens Klappstein, Thomas Brox, and Daniel Cremers. Realtime depth estimation and obstacle detection from monocular video. In *Pattern Recognition: 28th DAGM Symposium, Berlin, Germany, September 12-14, 2006. Proceedings 28*, pages 475–484. Springer, 2006.

[32] Yi Xiao, Felipe Codevilla, Akhil Gurram, Onay Urfalioglu, and Antonio M López. Multimodal end-to-end autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 23(1):537–547, 2020.

[33] Feng Xue, Guirong Zhuo, Ziyuan Huang, Wufei Fu, Zhuoyue Wu, and Marcelo H Ang. Toward hierarchical self-supervised monocular absolute depth estimation for autonomous driving applications. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2330–2337. IEEE, 2020.

[34] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. 2020.

[35] Renrui Zhang, Ziyao Zeng, Ziyu Guo, and Yafeng Li. Can language understand depth? In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6868–6874, 2022.

[36] Hang Zhou, David Greenwood, and Sarah Taylor. Self-supervised monocular depth estimation with internal feature fusion, 2021.

[37] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. *CoRR*, abs/1704.07813, 2017.