

From Perspective to Fisheye Depth Estimation and Open-Vocabulary Segmentation

Rit Gangopadhyay  and Alex Wong 

Yale Vision Laboratory, Yale University, New Haven, CT 06520, USA
{rit.gangopadhyay, alex.wong}@yale.edu

Abstract. Vision foundation models are capable of generalizing across 3-dimensional (3D) scenes with high-fidelity estimates; their empirical success can be attributed to training on large-scale datasets of perspective images. However, when transferred to wide field-of-view (FoV) images, such as those captured by fisheye cameras, they return erroneous outputs due to a covariate shift stemming from the radial distortion on the image pixels. We propose a method to generalize vision foundation models to fisheye cameras. The crux of our method lies in a set of learnable parameters, termed Distortion Extenders (DEX), that model the fisheye distortion coefficients and the distributional shift between fisheye and perspective images encoded in the latent space. By minimizing a self-supervised alignment loss, DEX transforms the latent embeddings of fisheye images to resemble those of perspective images to recover high-fidelity estimates. DEX is architecture- and task-agnostic: We demonstrate DEX on monocular depth estimation and open-vocabulary segmentation for convolution- and Transformer-based architectures, where we consistently improve over baselines across indoor and outdoor fisheye datasets. As a byproduct, the activations of DEX can also be decoded to distortion coefficients to support camera calibration. Code available at: <https://github.com/Suchisrit/DEX>.

Keywords: Adaptation · Monocular Depth Estimation · Open-Vocabulary Segmentation · Camera Calibration · Self-Supervised Learning

1 Introduction

Vision foundation models are typically trained on tens to hundreds of millions of images, enabling them to generalize across a wide range of 3-dimensional (3D) scenes and across tasks, spanning from recognition to 3D reconstruction. These models leverage the large volumes of internet images available, which are predominantly captured by standard perspective cameras of 3D scenes. The relationship of 3D points and their projection onto a 2D image plane can be described by the projective geometry of the camera, where the perspective cameras capturing the majority of the large-scale training dataset are well-approximated by a simple pinhole projection model. When training on such datasets, biases of the camera projective geometry are “baked into” these foundation models, which are

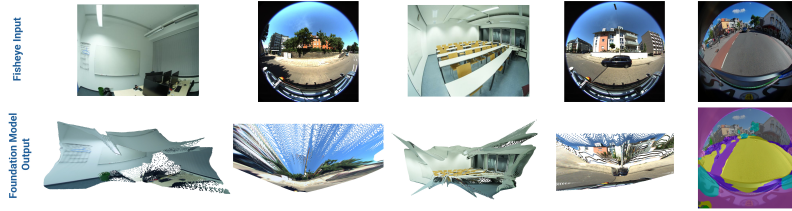


Fig. 1: 3D reconstructions and open-vocabulary segmentation of fisheye imagery produced by foundational models, UniDepthV2 [36] and LSeg [22]. The distortion in the input leads to erroneous and distorted reconstructions and segmentations.

often deployed onto mixed camera systems to enable spatial applications, from extended realities to robotic manipulation and autonomous navigation.

Yet, within these mixed camera systems are wide field-of-view (FoV) cameras necessary for environmental coverage (e.g., fisheye and other wide angle cameras). Their projective geometry differs from that of perspective cameras, in that 3D points follow a highly non-linear projection model that induces strong radial distortions. As the FoV increases, objects that are projected near the image borders become increasingly warped or distorted, deviating from the assumptions of projective geometry under which foundational models are trained. When these models are applied to fisheye images, the mismatch in projection induces a systematic distribution shift in local pixel arrangements, and in turn, the encoded features – leading to distorted 3D reconstructions and degraded recognition performance, as seen in Fig. 1.

A natural response is to “undo” this image distortion; hence, one line of work rectifies input images using known camera intrinsics, or re-projects them into a canonical representation such as equirectangular, tangent, spherical, or cube-map projections before processing [7, 13, 23, 27, 29, 64]. While this retains compatibility with models (pre)trained on perspective images, it introduces multiple resampling steps, calibration dependencies, and characteristic artifacts (e.g., stretching near poles, seams between faces, aliasing, and gaps between patches). Another approach is to train dedicated models (e.g., depth estimators or segmentation networks) for fisheye cameras [2, 21, 73], which often suffer in generalization across 3D scenes. This is partly due to the difficulty in assembling large-scale fisheye training datasets, as there exist fewer available fisheye images than perspective images. While one may finetune publicly available (foundational) models for specific cameras to improve generalization across scenes, this risks parameter drift and may degrade performance on previously learned cameras [10]. An alternative is to supplement the shortage via mixed-camera training [35], yet this trades generalizability for fidelity across cameras.

More importantly, the main school of thought aims to introduce inductive biases tailored to each camera model, which entails specialized designs and thus limits the end user’s flexibility in choosing architectures. Separately, the design of these methods is also task-specific, whether reconstruction or recognition,

leading to disparate approaches across vision tasks. Given that a pretrained model can already faithfully infer properties of the 3D scene from perspective images, erroneous estimates on fisheye images stem from a covariate shift in the values of latent embeddings due to local operations on image pixels displaced and rearranged by distortion. As such, if one can modulate their values to align with those that produced high-fidelity estimates in perspective images, then we hypothesize that one can extend the capabilities of models pretrained on images captured by perspective cameras to those of fisheye cameras. To ensure high utility, we consider an architecture- and task-agnostic approach, such that we can apply our method to both convolution-based and Transformer-based architectures for reconstruction and recognition tasks.

To do so, we begin by forgoing explicit modeling of camera projective geometry in the input and latent space of the model and build on the insight that the values of embeddings, rather than coordinates, can be adjusted to recover high-fidelity estimates. To this end, we freeze the backbone to preserve its performance on perspective images and propose a set of lightweight, learnable modules that can be inserted into each block of a frozen backbone to modulate feature embeddings of fisheye images. Rather than re-projecting pixels to a canonical image plane, these parameters implicitly model the distortion in fisheye images and operate directly in latent space to map fisheye embeddings toward the distribution of perspective image embeddings. Hence, our method bypasses the need for rectification or camera-specific inductive biases and the cost to retrain the backbone to extend perspective vision models to fisheye cameras. We term our method *Distortion Extenders*, or DEX for short.

We train DEX in a self-supervised manner using only calibrated perspective images. Given a perspective image, we synthetically distort it to obtain a fisheye counterpart. We pass the perspective image through a frozen vision model and treat its output (or intermediate features) as a reference. The fisheye image is then processed by the same backbone but augmented with DEX, and an inverse geometric transformation aligns the fisheye output back to the perspective reference frame. We minimize an alignment loss that drives the Extenders to reconcile the two estimates. This procedure requires neither ground truth nor real fisheye training data, and can be applied to both convolutional and Transformer architectures and across reconstruction (e.g., monocular depth estimation) and recognition tasks (e.g., open-vocabulary segmentation). At test time, fisheye images are fed directly into the augmented model – no calibration, re-projection, or knowledge of additional inference-time geometry is required.

We demonstrate our method on two vision tasks: monocular depth estimation and open-vocabulary segmentation. For monocular depth estimation, we further change the output parameterization from Cartesian depth z to spherical range R , leveraging the structure of fisheye projection models to more naturally express distances in high-FoV settings and to counteract the radial attenuation observed in models trained on perspective images. For open-vocabulary segmentation, we formulate our objective in the latent space to better take advantage of the affinities between an image and its associated text.

Our Contributions are as follows: (1) We introduce *Distortion Extenders* (DEX), a set of lightweight, feature-space modulators that generalizes pretrained models from perspective to fisheye imagery by transforming fisheye embeddings to align with those of perspective images. (2) We provide analysis showing that DEX indeed does model fisheye distortion coefficients and serves as latent distribution alignment. (3) DEX is architecture-agnostic and applicable to both convolution- and Transformer-based architectures. (4) DEX is also task-agnostic and enables strong zero-shot performance on real fisheye datasets for both monocular depth estimation (ScanNet++, KITTI-360) and open-vocabulary segmentation (WoodScape). As a byproduct, activations of DEX can be decoded to distortion coefficients for calibration purposes.

2 Related Works

Vision Foundation Models. Recently, foundation models (pretrained models designed to provide strong initializations for a variety of tasks) have seen significant growth in the vision domain. These models leverage large datasets through both self-supervised and supervised learning. Their flexibility enables them to adapt to and perform a wide range of tasks. CLIP [38] demonstrates broad task applicability by using contrastive learning on a large vision-language dataset and has been a core component in open-vocabulary segmentation [22,62]. DINO [3,31] uses self-supervision to extract effective features and has been used as an encoding module in depth estimators [37,65,66]. However, despite being trained on extensive datasets, they experience performance degradation when used with data captured by fisheye cameras, where errors propagate to downstream models.

Transformation-based Solutions. To enable depth estimation on unrectified images, methods typically rely on alternate representations such as equirectangular projection (ERP) [23,52], tangent projection [7,24], spherical projection [29,64] or cube mapping [27,51]. However, each of these representations has drawbacks: aside from resampling artifacts, ERP suffers from vertical stretching distortion and angular compression towards the poles, tangent projection exhibits radial stretching and nonuniform area scaling towards the edges, and cube maps have discontinuous transitions between each face of the cube. These artifacts can introduce spurious errors when processed by standard neural network operations, such as convolutions, tokenization, and patch embedding. Spherical projections further require specialized network operations, which increases implementation complexity. Additionally, these projections introduce latency during inference and require accurate camera calibration at test time, which can be sensitive to small physical changes in the system.

Depth Estimation for Specific Cameras. Hence, to avoid projecting input images to a canonical representation, another line of work seeks to train depth estimators tailored to specific camera models [8, 9, 12, 14, 21, 30, 32, 34, 41, 42, 45, 47, 49, 55–59, 67, 73], e.g., perspective, wide-angle, fisheye, omnidirectional. Yet, it is challenging to acquire training datasets for wide FoV cameras com-

pared to publicly available perspective camera datasets. As such, it is difficult to train foundation models for wide FoV cameras. Nonetheless, one can finetune existing large-scale pretrained models, e.g., foundational depth estimators, for specific cameras [1, 18, 60]. While finetuning can improve performance for a target camera, it also introduces the risk of parameter drift [5, 33], where the resulting models may lose their generalizability across 3D scenes and previously learned cameras. It also produces camera-specific models, so deployment platforms (e.g., autonomous vehicles, XR, robotic systems) with mixed camera systems must maintain multiple depth estimators, which increases operational complexity.

Generalizable Depth Estimators. To obtain models that generalize across 3D scenes and cameras, recent work has focused on developing foundational monocular depth estimators (FMDEs) [37, 40, 65]. Aside from training on large-scale datasets of mixed cameras, methods in this vein also incorporate the previously mentioned projection to some canonical space, either in the input or in the internal latent representation: DepthAnyCamera [13] applies equirectangular projection (ERP) to the input image before processing it through the network and backprojects the output to the original image space; FoVA-Depth [27] instead utilizes cube mapping of the input image; Unik3D [35] adopts a spherical internal representation and trains on 23 different datasets of mixed cameras to achieve state-of-the-art performance. However, the fidelity across all cameras is reduced. Additionally, the architectural complexity introduced by these models amounts to hundreds of millions of parameters, which imposes less flexibility on deployment platforms with specific computational requirements.

Image Segmentation has also benefited from large-scale pretrained networks. Classic convolution-based architectures like U-Net [44] and DeepLab [4] remain popular, while Transformer-based methods [63] have also shown success in capturing global context. Recent developments such as the Segment Anything Model [17] highlight the power of large-scale self-supervision and prompt-based adaptation. Semantic segmentation [4, 19, 20, 44, 54] involves recognizing the identity of segmented regions in the image, extensions of which include *Open-vocabulary Semantic Segmentation*, where the possible classes for objects in the scene are unknown until inference time. Many of such methods [15, 25, 28, 71, 72] rely on the CLIP vision-language model because of its strong alignment between image and language embeddings, which can be used to assign labels to regions of an image. Recent methods such as LSeg [22] and SED [62] encode text and language separately at input and compare their embeddings to assign regions of the image to the given set of classes. Similar to depth estimation, segmentation pipelines can suffer when encountering heavy distortions. However, most existing work to handle fisheye images can only be applied to depth estimation. We instead show how our method can be easily adapted to recognition tasks.

Differentiation from Model Extension and Adaptation Methods. The *closest* (concurrent) work to ours is Calibration Tokens [6, 10], which extend FMDEs pretrained on perspective images to fisheye images by appending trainable tokens to Transformer blocks. Unlike them, the proposed Distortion Extenders are distinct in: (1) Our method is architecture-agnostic and can be

applied to feature map-based architectures (e.g., convolutional neural networks, CNNs) as well as token-based architectures (e.g. Transformers); Calibration Tokens are limited to Transformers due to their use of tokens. (2) Calibration Tokens require manual removal of its appended tokens after each layer, which adds complexity; unlike them, our method does not require appending or discarding of any element, but automatically determines the weighting of the Extenders as a convex combination for modulation based on their affinity towards fisheye features. (3) Calibration Tokens utilize a loss specific to depth, which limits their tasks, whereas our loss is applicable to both depth estimation and segmentation. (4) A minor difference is that Calibration Tokens output depth in z-depth (Cartesian); whereas, we output in range (Euclidean) for depth estimation.

Also related are cross-attention [50] and LoRA [16]. Cross-attention determines an added bias by learning projections of different inputs to queries, keys, and values; whereas, LoRA projects the input directly to an added bias by learning a weight matrix. Unlike them, we learn Extender modulators, and a projection matrix to select amongst them, that allow the input features to directly determine weights for convex combinations of the Extenders. The result of which will modulate each encoder block within a backbone. Intuitively, one may analogize Extenders to “centers” or entries of a “codebook” that model distortion coefficients and the distributional shift between perspective and fisheye images.

3 Method

Formalization. We assume that we are given a model pretrained on perspective images, where its architecture comprises an image encoder $f(\cdot)$ and a task decoder $g(\cdot)$. We note that f and g are parameterized functions, but for ease of notation, we omit their parameters as both sets are fixed or frozen in our method. Let $I : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^3$ denote an input RGB image, where Ω denotes the image space. Inference for a task is given by $\hat{y} = g(f(I))$ for the estimate $\hat{y} \in \mathbb{R}^{H \times W \times C}$, where H and W are its spatial dimensions. In the case of regression tasks, e.g., depth estimation, $C = 1$ and $\hat{y}$ denotes a 2.5D range or depth map. In the case of classification tasks, e.g., open-vocabulary segmentation, C varies depending on the number of input classes or text labels and $\hat{y}$ are the logits corresponding to affinities between the image and text features.

To extend the pretrained model to handle fisheye images with similar fidelity as perspective images, we propose *Distortion Extenders* (DEX) to model distortions observed in the image and to correct for the covariate shift in values of the latent embeddings stemming from fisheye distortion. To this end, the crux of our method lies in a set of modulators that translate the latent embeddings of fisheye images such that they resemble those of perspective images; the result of which is a latent alignment of fisheye image embeddings to the distribution of perspective image embeddings. This naturally necessitates two steps: (1) determine the degree of distortion and their effect on the latent embeddings and (2) determine the modulation necessary to correct for their effect. To make the modeling of the broad range of distortions tractable, each Extender serves as a

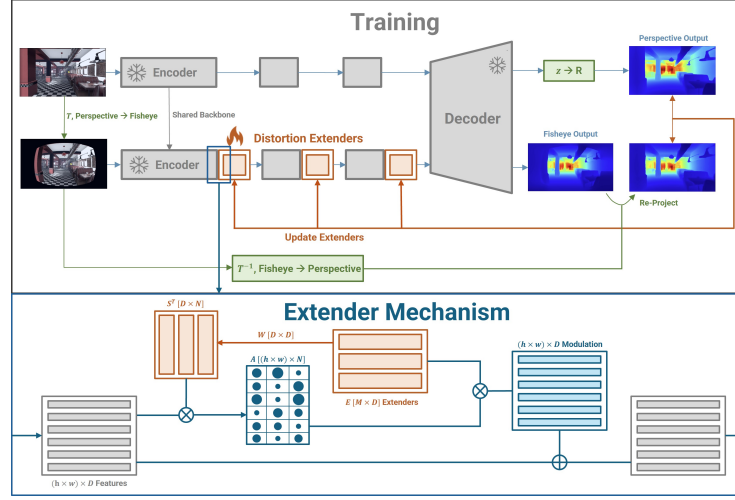


Fig. 2: Distortion Extenders. The figure illustrates the DEX training pipeline and architecture. During training, we pass perspective inputs and fisheye inputs augmented by DEX into a frozen backbone model. The fisheye features select convex combinations of Extenders for translation in the latent space. We optimize the Distortion Extenders to minimize the difference between the fisheye and perspective output.

discrete “center” in this continuous space, where convex combinations of multiple Extenders determine the translation necessary to correct for the effect of image distortion on latent embeddings of a fisheye image. As such, DEX is comprised of two primary components: (1) a “soft” selection mechanism that is parameterized by a weight matrix $\mathbf{W} \in \mathbb{R}^{D \times D}$ and (2) a set of M Extender modulation parameters $\mathbf{E} \in \mathbb{R}^{M \times D}$, where D denotes the dimensions of latent embeddings encoded by f . Thus, the full architecture of DEX is $\theta = \{\mathbf{E}^{(l)}, \mathbf{W}^{(l)}\}_{l=1}^L$ for L layers. An overview of our method is shown in Fig. 2.

Extender Mechanism. Let $\mathbf{X}^{(l)} = f^{(l)}(\mathbf{X}^{(l-1)})$ be the output of the l th layer or block of an image encoder (where $\mathbf{X}^{(0)} = I$). We learn Distortion Extenders θ , to modulate the layer’s output, $\mathbf{X}^{(l)} \in \mathbb{R}^{(h \times w) \times D}$, for all L layers. To reduce over-parametrization, we use lower rank matrices to represent $\mathbf{W} = \mathbf{W}_1^T \mathbf{W}_2$, where $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d \times D}$ with $d \ll D$. As the covariate shift stems from changes in calibration, we hypothesize a low dimensional matrix can encode the transformation. Our modulation after each transformer block reads:

$$\mathbf{S} = \mathbf{E}\mathbf{W}, \quad \mathbf{A} = \text{softmax}\left(\frac{\mathbf{X}\mathbf{S}^T}{\sqrt{D}}\right), \quad \mathbf{X} \leftarrow \mathbf{X} + \mathbf{A}\mathbf{E}, \quad (1)$$

which allows us to add a convex combination of Extenders to every latent embedding (whether token or feature vector) in $\mathbf{X}$, weighted by the affinity $\mathbf{A}$ between each latent embedding and each Extender’s selection in $\mathbf{S}$. The modulated embeddings are then used in the next layer and any skip connections:

$$\mathbf{X}^{(l+1)} = f^{(l+1)}(\mathbf{X}^{(l)} + (\mathbf{AE})^{(l)}) = f^{(l+1)}(f^{(l)}(\mathbf{X}^{(l-1)} + (\mathbf{AE})^{(l-1)}) + (\mathbf{AE})^{(l)}) \quad (2)$$

Distortion Extenders can be thought of as "centers" that model the latent calibration differences between perspective and fisheye cameras. As we take convex combinations of Extenders, they form a boundary around a range of possible modulations to correct for an image's distortion in the latent space. The intuition is that an input image's features are able to select the weights of the combination of Extenders that represents its own distortion. We study this further in Sec. 5, where we use this principle to decode Distortion Extenders into KB distortion coefficients using an image's selected weights.

Training. We aim to optimize our learnable Distortion Extenders, θ , at each of the L encoder blocks to align fisheye image latent embeddings with those of perspective images. We use a self-supervised training scheme, based on AugUndo [61]. We start with calibrated, perspective images I , and distort them with a fisheye transformation $\mathbf{T}$ to yield a synthetic fisheye image counterpart $I_f = \mathbf{T} \circ I$. We freeze a backbone model f trained on perspective images. Then, we retrieve the output of the perspective image $\hat{y} = g(f(I))$, and the output of the fisheye image augmented with DEX, e.g., $\hat{y}_f = g(f_\theta(I_f))$. We then apply the inverse fisheye transformation $\mathbf{T}^{-1}$ on $\hat{y}_f$ in the output space to facilitate comparison between the perspective output and the spatially aligned fisheye output in the perspective reference frame. We formulate the optimization as:

$$\underset{\theta}{\operatorname{argmin}} \frac{1}{N} \sum_{n=1}^N \mathcal{L}(\hat{y}^{(n)}, \mathbf{T}^{-1} \circ \hat{y}_f^{(n)}). \quad (3)$$

Distortion Extenders can be applied to both reconstruction and recognition tasks. Thus, we derive two different forms of the learning objective from Eq. 3.

Monocular Depth Estimation. In the case of depth estimation, our Distortion Extenders also learn to re-project the output from a Cartesian coordinate system to a spherical one. We treat our perspective output as the supervision signal $\hat{z} = g(f(I))$, and retrieve the output of the fisheye image with Distortion Extenders $\hat{R}_f = g(f_\theta(I_f))$. However, the baseline model's output is in depth. Thus, we convert the perspective output from z - to R -coordinates using the following process for each point on the depth map:

$$\mathbf{p}' = [x' \ y' \ 1]^\top, \quad \mathbf{P}_{\text{proj}} = \mathbf{K}^{-1} \mathbf{p}' = [X/z \ Y/z \ 1]^\top, \quad (4)$$

$$R = |z \cdot \mathbf{P}_{\text{proj}}| = |\mathbf{P}| = \sqrt{X^2 + Y^2 + z^2}. \quad (5)$$

Finally, we use the following loss to optimize the Distortion Extenders θ :

$$\mathcal{L}(\hat{R}^{(n)}, \mathbf{T} \circ \hat{R}_f^{(n)}) = \log(|\hat{R}^{(n)} - \mathbf{T} \circ \hat{R}_f^{(n)}| + 1). \quad (6)$$

Open-Vocabulary Segmentation. Most architectures for open-vocabulary segmentation involve separately encoding the image $f_i(I)$ and a set of text labels $f_l(t)$, representing the desired classes for classification. Then, some form of similarity is computed between the image embeddings and the language embeddings

in order to assign a class to each pixel of the image. Thus, instead of computing our loss on the entire network’s output, which includes a multiplication with the language modality, we compare the perspective and fisheye image embeddings:

$$\mathcal{L}(f_i^{(L)}(I^{(n)}), \mathbf{T}^{-1} \circ f_{i,\theta}^{(L)}(I_f^{(n)})) = \log(|f_i^{(L)}(I^{(n)}) - \mathbf{T}^{-1} \circ f_{i,\theta}^{(L)}(I_f^{(n)})| + 1). \quad (7)$$

Inference. At inference time, no rectification or knowledge of calibration is necessary to handle fisheye images. For depth estimation, we simply attach our Distortion Extenders to the backbone model and produce our output reconstruction $\hat{R} = f_\theta(I_f)$. The same follows for segmentation to produce our image embedding $f_{i,\theta}(I_f)$, which is later compared with language embeddings $f_l(t)$ in the baseline model’s architecture using the inner product. The additional overhead in storage and time for inference is minimal, as shown in Tab. 5.

4 Experiments

Datasets. During the training process, we synthesize fisheye images using the KB distortion model, and optimize our Distortion Extenders by comparing with the perspective image output. As such, only calibrated, perspective images are necessary for training. Ground truth or real fisheye images are not necessary. We evaluate our method on real indoor and outdoor fisheye images, where we train just one set of Distortion Extenders for both domains, showing the ability of our method to generalize across 3D scenes.

Training datasets: *NYUv2* [46] provides a wide range of indoor perspective imagery; *VOID* [57] includes perspective office, classroom, and stairwell scenes; *IRS* [53] contains synthetic, rendered perspective views of homes, restaurants, and retail environments; *Hypersim* [43] offers high-quality photorealistic indoor scenes, including both residential and commercial, captured in a perspective projection. The *Waymo* dataset [48] supplements these with a broad collection of perspective urban driving scenes. Validation datasets: *ScanNet++* [68] provides real fisheye images and ground-truth depth, collected via laser scanning and DSLR fisheye capture, enabling evaluation on authentic indoor fisheye data. *KITTI-360* [26] features suburban driving scenes recorded with a multi-sensor rig that includes fisheye cameras, along with 3D ground-truth points for assessing reconstruction accuracy. *WoodScape* [70] offers similar driving scenes with 2D semantic instance labels. We use this dataset to evaluate performance in the open-vocabulary segmentation task. See Supp. Mat. for more details on datasets.

Models. Distortion Extenders are architecture- and task-agnostic. As such, we evaluate them with state-of-the-art monocular depth estimation models using the Vision Transformer architecture: MiDaS [39], DepthAnything [65], and UniDepthV2 [36]. We also evaluate our Extenders with LSeg [22], a state-of-the-art model for open-vocabulary semantic segmentation. To show that DEX is architecture agnostic, we additionally evaluate with CNN-based models, VNL [69] for monocular depth estimation and SED [62] for open-vocabulary segmentation.

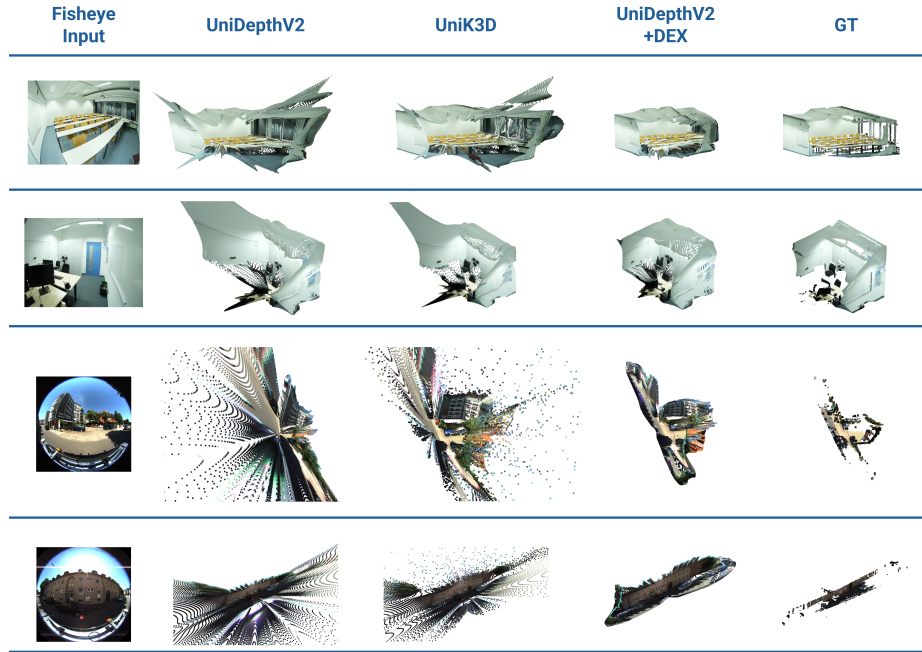


Fig. 3: MDE Qualitative Results. We show 3D reconstructions of indoor and outdoor samples from the evaluation in Tab. 1. Previous methods struggle with the outermost regions in the fisheye images, which have the most severe distortion. DEX is more robust to these distortions. In the outdoor scenes with the highest distortions, we are still able to accurately reconstruct many of the points at the edges of the images.

4.1 Evaluation on Fisheye Datasets

We evaluate our method’s performance when used with various MDEs in Tab. 1, testing on real fisheye images in both indoor and outdoor settings with the same set of Extenders. We also compare our method to UniK3D [35], a state-of-the-art method for fisheye monocular depth estimation. Additionally, we compare our method to LoRA adaptation and Calibration Tokens adapted for R . As there are currently no other models to perform segmentation specifically for fisheye images, we evaluate our method in conjunction with LSeg [22] and SED [62], which perform open-vocabulary semantic segmentation. Note: We further adapted Calibration Tokens [10] for LSeg for an additional comparison, but could not for VNL and SED as Calibration Tokens require Transformers.

We additionally compare DEX against a parameter-efficient LoRA adaptation of UniK3D to our training set. UniK3D already includes large-scale fisheye training data from ASE, aiMotive, HOI4D, and DL3DV, totaling approximately 3.263M examples, as well as ScanNet++ pinhole data. DEX achieves lower RMSE and higher δ_1 on both ScanNet++ and KITTI-360. We hypothesize that UniK3D performance has largely saturated and that further adaptation with our mix of 200K examples makes little difference.

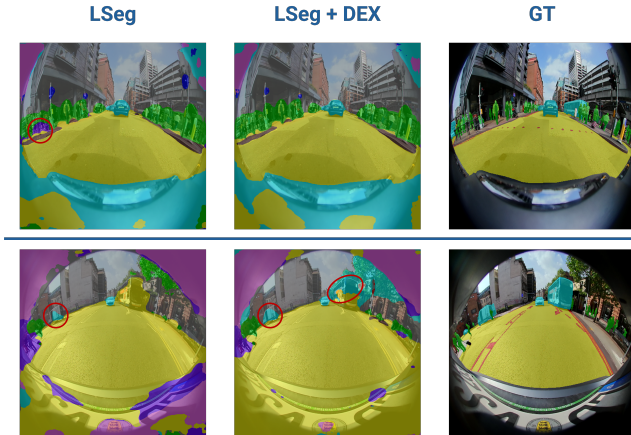


Fig. 4: Segmentation Qualitative Results. We evaluate LSeg and SED with and without DEX on the WoodScape outdoor dataset. We see improvements in the open-vocabulary segmentation, highlighting DEX’s generalizability to this recognition-based task. We evaluate on the valid regions of the images specified by the ground truth.

Indoor MDE. We evaluate the performance of MDE models with and without DEX on various scenes from ScanNet++ [68], an indoor fisheye dataset. We compare the output 3D reconstructions against the ground truth in the ground truth’s Cartesian coordinate system. Across the baseline monocular depth estimators, DEX improves by 19% in RMSE and 15% in δ_1 on average over LoRA and Calibration Tokens. Additionally, when applied to UniDepthV2, our method improves over UniK3D by 8% averaged over the metrics.

Outdoor MDE. We also evaluate on the KITTI-360 outdoor fisheye image dataset. These images feature much higher FoV and as such, much more severe fisheye distortion. Predicting in spherical coordinates is especially helpful in this case. We compare all output 3D reconstructions against the ground truth in the ground truth’s spherical coordinate system. On average, we see an 11% improvement in RMSE and 11% improvement in δ_1 over LoRA and Calibration Tokens across the baseline models. Additionally, the same set of Extenders is used for the indoor and outdoor setting, which demonstrates how DEX takes advantage of the existing backbone model’s inherent generalization.

Open-Vocabulary Segmentation. We evaluate the LSeg and SED open-vocabulary semantic segmentation models on the WoodScape segmentation dataset with and without Distortion Extenders. We query the model with the following classes during evaluation: road, lane markings, curb, human, cyclist, cars, bicycle, motorcycle, street sign, other. We find that DEX improves over Calibration Tokens by around 13% in the mIoU metric. The experiments are shown in Tab. 2.

4.2 DEX as Task-Agnostic

To further evaluate whether DEX is task-agnostic, we apply it to PanopticDepth [11], a multitask framework for depth estimation and depth-aware panoptic seg-

Table 1: MDE Performance on Indoor and Outdoor. After training a set of Distortion Extenders, we test their zero-shot performance on ScanNet++ and KITTI-360, which contain real fisheye images with varying scenes and distortion parameters.

Testset	Experiment	Model	Train Dataset	RMSE ↓	δ_1 ↑
ScanNet++ [68]	Base Model	MiDaS [39]	Mix 1.4M	0.672	0.479
	w/ LoRA [16]	MiDaS [39]	Mix 200K	0.561	0.599
	w/ Calibration Tokens [10]	MiDaS [39]	Mix 200K	0.413	0.688
	w/ Distortion Extenders	MiDaS [39]	Mix 200K	0.397	0.702
	Base Model	DepthAnything [65]	Mix 63.5M	0.705	0.454
	w/ LoRA [16]	DepthAnything [65]	Mix 200K	0.706	0.445
	w/ Calibration Tokens [10]	DepthAnything [65]	Mix 200K	0.479	0.667
	w/ Distortion Extenders	DepthAnything [65]	Mix 200K	0.414	0.708
	Base Model	UniDepthV2 [36]	Mix 16M	0.329	0.671
	w/ LoRA [16]	UniDepthV2 [36]	Mix 200K	0.235	0.854
	w/ Calibration Tokens [10]	UniDepthV2 [36]	Mix 200K	0.223	0.841
	w/ Distortion Extenders	UniDepthV2 [36]	Mix 200K	0.200	0.872
	Base Model	VNL [69]	NYUD-V2 29K	0.680	0.521
	w/ Distortion Extenders	VNL [69]	Mix 80K	0.372	0.704
	Comparison	DepthAnyCamera [13]	Indoor 670K	0.390	0.852
	Comparison	UniK3D [35]	Mix 12M	0.223	0.825
	w/ LoRA	UniK3D [35]	Mix 200K	0.218	0.835
KITTI-360 [26]	Base Model	MiDaS [39]	Mix 1.4M	6.111	0.312
	w/ LoRA [16]	MiDaS [39]	Mix 200K	2.479	0.601
	w/ Calibration Tokens [10]	MiDaS [39]	Mix 200K	2.588	0.699
	w/ Distortion Extenders	MiDaS [39]	Mix 200K	2.451	0.657
	Base Model	DepthAnything [65]	Mix 63.5M	6.484	0.318
	w/ LoRA [16]	DepthAnything [65]	Mix 200K	2.897	0.550
	w/ Calibration Tokens [10]	DepthAnything [65]	Mix 200K	2.224	0.676
	w/ Distortion Extenders	DepthAnything [65]	Mix 200K	2.022	0.745
	Base Model	UniDepthV2 [36]	Mix 16M	7.093	0.262
	w/ LoRA [16]	UniDepthV2 [36]	Mix 200K	1.916	0.771
	w/ Calibration Tokens [10]	UniDepthV2 [36]	Mix 200K	1.788	0.763
	w/ Distortion Extenders	UniDepthV2 [36]	Mix 200K	1.663	0.842
	Base Model	VNL [69]	KITTI 24K	7.719	0.245
	w/ Distortion Extenders	VNL [69]	Mix 80K	2.222	0.605
	Comparison	DepthAnyCamera [13]	Outdoor 130K	3.641	0.789
	Comparison	UniK3D [35]	Mix 12M	2.969	0.812
	w/ LoRA	UniK3D [35]	Mix 200K	2.802	0.818

Table 2: Open Vocabulary Segmentation. We evaluate the zero-shot performance of our method on the WoodScape dataset, which contains real fisheye images with varying scenes and distortion parameters.

Testset	Experiment	Model	Train Dataset	mIoU ↑	weighted IoU ↑
WoodScape [70]	Baseline	LSeg [22]	Mix 500K	0.305	0.819
	w/ Calibration Tokens	LSeg [22]	Mix 50K	0.321	0.823
	w/ Distortion Extenders	LSeg [22]	Mix 50K	0.362	0.838
	Baseline	SED [62]	Mix 120K	0.439	0.829
	w/ Distortion Extenders	SED [62]	Mix 50K	0.449	0.832

mentation. We jointly optimize DEX by summing both depth and segmentation loss terms. The depth term similarly compares DEX’s fisheye depth output to PanopticDepth’s perspective depth output. The segmentation term is computed in this case by comparing DEX’s fisheye semantic logits to PanopticDepth’s perspective semantic logits. We are able to reuse the same training set for the experiment because we use the PanopticDepth baseline outputs as supervision.

Table 3: DEX with PanopticDepth. We apply DEX to PanopticDepth and evaluate both depth estimation and segmentation on WoodScape.

Method	Depth		Segmentation	
	RMSE ↓	δ_1 ↑	mIoU ↑	weighted IoU ↑
PanopticDepth [11]	7.273	0.267	0.160	0.300
w/ Distortion Extenders	3.583	0.354	0.282	0.833

As shown in Tab. 3, DEX improves performance in both depth estimation on KITTI-360 and segmentation on WoodScape.

5 Analysis

Ablation Study. In order to characterize the contributions of various architectural decisions, we ablate them and evaluate the performance of depth estimation with the UniDepthV2 model. As expected, learning to estimate depth in the Cartesian coordinate system has an especially negative effect in the outdoor setting, which has a much wider FoV with stronger fisheye distortion. Additionally, without the low-rank decomposition, there is a slight performance degradation due to over-parametrization. See Tab. 4 for the full results.

Feature Alignment. We claim that DEX is able to make fisheye images more conducive to models trained on perspective imagery by modulating fisheye image features towards the distribution of perspective features. We verify this claim in Fig. 5, where we use a t-SNE dimensionality reduction to visualize the UniDepthV2 image features of a perspective image, and its corresponding fisheye image with and without DEX. When the model is augmented with DEX, the fisheye image features align very closely with the perspective image features.

DEX Decoding. To better understand the internal representations learned by DEX, we attempt to decode the Extenders into KB distortion coefficients for an image input. We posit that because DEX learns to align fisheye features to perspective features, Distortion Extenders implicitly encode KB distortion coefficients. Thus, we should be able to find the Extenders that encode a particular image’s distortion coefficients using the weightings of its activated Extenders.

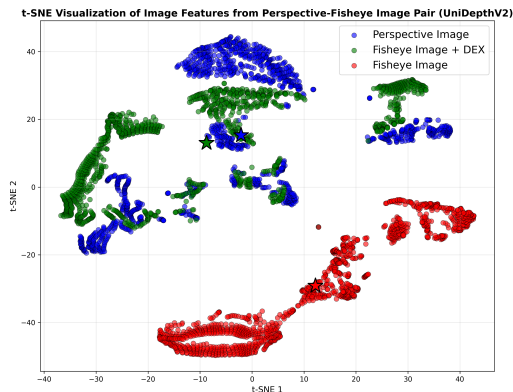
As such, we pass an image input through UniDepthV2 with our trained DEX, and extract the affinities A along with the Extenders $\mathbf{E}$ for all L layers. We then use a simple linear probe $\mathbf{W}_D \in \mathbb{R}^{D \times 4}$ to decode each Extender into four values $\left[\alpha_1^{(i)} \alpha_2^{(i)} \alpha_3^{(i)} \alpha_4^{(i)} \right]_{i=1}^M = \mathbf{E} \mathbf{W}_D$. For each layer, we take a linear combination across the α values for each Extender, using the affinities A averaged across image features. We then predict the final KB distortion coefficients $[k_1 \ k_2 \ k_3 \ k_4]$ with one more linear combination across layers, with a learnable weighting $\mathbf{W}_L \in \mathbb{R}^L$. As we also optimize DEX to extend the baseline model to spherical coordinates, this learnable weighting across layers helps us select the layers with Extenders most relevant to the fisheye distortion correction.

Table 4: Ablation Experiments. We evaluate DEX’s performance without low rank decomposition, and without extension to spherical coordinates.

Dataset	Method	RMSE	δ_1
ScanNet++	Distortion Extenders	0.200	0.872
	w/o $\hat{R}$ Extension	0.249	0.792
	w/o Low-Rank Decomposition	0.216	0.849
KITTI-360	Distortion Extenders	1.663	0.842
	w/o $\hat{R}$ Extension	7.024	0.271
	w/o Low-Rank Decomposition	1.777	0.800

Table 5: Computational Overhead. Additional memory and inference time of Distortion Extenders over the backbone models. DEX adds minimal complexity.

Model	UniDepthV2	LSeg
Base memory	0.7 GB	2.9 GB
DEX memory	2.8 MB	6.1 MB
Inference	+0.3 ms	+2.21 ms

**Fig. 5: t-SNE Plot of Fisheye and Perspective Image Features.** When using DEX, fisheye image features align much closer with corresponding perspective features.

We optimize $\mathbf{W}_D$ and $\mathbf{W}_L$ using the same training data in the DEX training pipeline, and evaluate on the same ScanNet++ evaluation set. As seen in Tab. 6, with a simple linear regressor, we are able to predict k_1 with the highest accuracy, and k_4 with lower accuracy. The degradation pattern from k_1 to k_4 aligns with the fisheye distortion effect from each parameter. Changes in k_1 will result in a much greater visual effect than changes in k_4 , e.g., moderate changes in k_4 do not yield visual differences, so the earlier coefficients dominate the discriminativeness in determining distortion. As such, Distortion Extenders encode later distortion coefficients with wider uncertainty.

We show the visual difference for perturbing the distortion coefficients in an image by the error percentages produced by our method in Fig. 6. There is very little difference caused by this shift in distortion parameters, indicating difficulty in recovering them with higher fidelity.

6 Discussion

DEX provides a simple but effective mechanism for adapting perspective-trained vision foundation models to fisheye imagery without modifying the backbone or performing explicit geometric rectification at test time. By operating directly

Table 6: KB Distortion Parameter Decoding. Distortion Extenders are able to recover KB distortion parameters for an image input.

ScanNet++	k_1	k_2	k_3	k_4	Average
MSE ↓	2.04×10^{-6}	1.53×10^{-6}	3.53×10^{-7}	8.87×10^{-9}	9.81×10^{-7}
MAE ↓	1.17×10^{-3}	9.88×10^{-4}	5.27×10^{-4}	7.05×10^{-5}	6.88×10^{-4}
MAPE (%) ↓	3.91	14.30	41.86	78.61	34.67

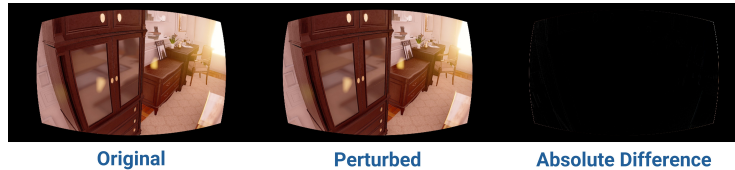


Fig. 6: Error in KB Distortion Prediction Visualized. We provide an example for the visual difference in distortion when we generate a fisheye image with a set of distortion parameters representative of ScanNet++, and when we change all of the distortion coefficients by our average error in prediction. Absolute difference between the images generated by the two different sets of distortion coefficients has little visual difference. Absolute difference is best viewed in 5×.

in feature space, our approach mitigates the projection mismatch that arises from high-FoV lenses. The improvements observed across multiple backbones and datasets (Tab. 1, 2) suggest that many of the errors in distorted views stem from a systematic feature-space shift. More broadly, this work illustrates the flexibility of our training pipeline. The same self-supervised alignment framework of using a frozen backbone, synthetic domain shift, inverse geometric alignment, and lightweight extenders can be reused across architectures and tasks.

Limitations. During training, we do not need ground truth or fisheye images. However, this means that our performance is upper-bounded by the quality of the backbone models we use as our supervision signal, which implies we must carefully select training data that the backbone model can produce high-fidelity outputs for. Luckily, FMDEs and open-vocabulary segmentation models have been trained on tens to hundreds of millions of images. This allows us to casually select training datasets. We hypothesize that one must take care in dataset selection for smaller-scale models.

Acknowledgements

This work is supported by a Google Gift Award, NSF-2112562 Athena AI Institute and the Global Industrial Technology Cooperation Center (GITCC) through a grant agreement with the Korea Institute for Advancement of Technology (KIAT), project number P0028922.

References

1. Berenguel-Baeta, B., Santos-Villafranca, M., Bermudez-Cameo, J., Pérez-Yus, A., Guerrero, J.J.: Convolution kernel adaptation to calibrated fisheye. In: 34th British Machine Vision Conference 2023, BMVC 2023, Aberdeen, UK, November 20-24, 2023. BMVA (2023)
2. Blott, G., Takami, M., Heipke, C.: Semantic segmentation of fisheye images. In: Proceedings of the European Conference on Computer Vision (ECCV) Workshops. pp. 0–0 (2018)
3. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021)
4. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K.P., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40**, 834–848 (2016)
5. Chung, Y., Park, H., Rim, P., Zhang, X., He, J., Zeng, Z., Cicek, S., Hong, B.W., Duncan, J.S., Wong, A.: Eta: Energy-based test-time adaptation for depth completion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2025)
6. Duan, R., Hong, E., Turner, E., Wong, A., Zhou, Y.: Fisheye3r: Adapting unified 3d feed-forward foundation models to fisheye lenses. In: European Conference on Computer Vision. Springer (2026)
7. Eder, M., Shvets, M., Lim, J., Frahm, J.M.: Tangent images for mitigating spherical distortion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12426–12434 (2020)
8. Ezhov, V., Park, H., Zhang, Z., Upadhyay, R., Zhang, H., Chandrappa, C.C., Kadambi, A., Ba, Y., Dorsey, J., Wong, A.: All-day depth completion. In: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE (2024)
9. Fei, X., Wong, A., Soatto, S.: Geo-supervised visual depth prediction. *IEEE Robotics and Automation Letters* **4**(2), 1661–1668 (2019)
10. Gangopadhyay, S., Kim, J.H., Chen, X., Rim, P., Park, H., Wong, A.: Extending foundational monocular depth estimators to fisheye cameras with calibration tokens. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5198–5209 (2025)
11. Gao, N., He, F., Jia, J., Shan, Y., Zhang, H., Zhao, X., Huang, K.: Panopticdepth: A unified framework for depth-aware panoptic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1632–1642 (2022)
12. Gui, X., Zhang, X.: An efficient dense depth map estimation algorithm using direct stereo matching for ultra-wide-angle images. In: Computer Graphics International Conference. pp. 117–128. Springer (2022)
13. Guo, Y., Garg, S., Miangoleh, S.M.H., Huang, X., Ren, L.: Depth any camera: Zero-shot metric depth estimation from any camera. *arXiv preprint arXiv:2501.02464* (2025)
14. Guo, Z., Seo, Y., Fu, A., Hao, M., Xia, I., Lee, J.J., Xie, X.M., Park, H., Dollar, A., Wong, A.: Vista3d: A dataset and benchmark for thin object reconstruction from vision, tactile, and 3d point clouds. In: European Conference on Computer Vision. Springer (2026)

15. He, W., Jamonnak, S., Gou, L., Ren, L.: Clip-s4: Language-guided self-supervised semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11207–11216 (2023)
16. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. ICLR **1**(2), 3 (2022)
17. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3992–4003 (2023)
18. de La Garanderie, G.P., Abarghouei, A.A., Breckon, T.P.: Eliminating the blind spot: Adapting 3d object detection and monocular depth estimation to 360 panoramic imagery. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 789–807 (2018)
19. Lao, D., Wu, Y., Liu, T.Y., Wong, A., Soatto, S.: Sub-token vit embedding via stochastic resonance transformers. In: International Conference on Machine Learning. PMLR (2024)
20. Lao, D., Yang, F., Wang, D., Park, H., Lu, S., Wong, A., Soatto, S.: On the viability of monocular depth pre-training for semantic segmentation. In: European Conference on Computer Vision. Springer (2024)
21. Lee, J., Cho, G., Park, J., Kim, K., Lee, S., Kim, J.H., Jeong, S.G., Joo, K.: Slabins: Fisheye depth estimation using slanted bins on road environments. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8765–8774 (2023)
22. Li, B., Weinberger, K.Q., Belongie, S., Koltun, V., Ranftl, R.: Language-driven semantic segmentation. arXiv preprint arXiv:2201.03546 (2022)
23. Li, M., Jin, X., Hu, X., Dai, J., Du, S., Li, Y.: Mode: Multi-view omnidirectional depth estimation with 360 cameras. In: European Conference on Computer Vision. pp. 197–213. Springer (2022)
24. Li, Y., Guo, Y., Yan, Z., Huang, X., Duan, Y., Ren, L.: Omnifusion: 360 monocular depth estimation via geometry-aware fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2801–2810 (2022)
25. Liang, F., Wu, B., Dai, X., Li, K., Zhao, Y., Zhang, H., Zhang, P., Vajda, P., Marculescu, D.: Open-vocabulary semantic segmentation with mask-adapted clip. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7061–7070 (2023)
26. Liao, Y., Xie, J., Geiger, A.: Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(3), 3292–3310 (2022)
27. Lichy, D., Su, H., Badki, A., Kautz, J., Gallo, O.: Fova-depth: Field-of-view agnostic depth estimation for cross-dataset generalization. In: 2024 International Conference on 3D Vision (3DV). pp. 1–10. IEEE (2024)
28. Lin, Y., Chen, M., Wang, W., Wu, B., Li, K., Lin, B., Liu, H., He, X.: Clip is also an efficient segmenter: A text-driven approach for weakly supervised semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15305–15314 (2023)
29. Liu, J., Xu, Y., Li, S., Li, J.: Estimating depth of monocular panoramic image with teacher-student model fusing equirectangular and spherical representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1262–1271 (2024)

30. Liu, T.Y., Agrawal, P., Chen, A., Hong, B.W., Wong, A.: Monitored distillation for positive congruent depth completion. In: *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*. pp. 35–53. Springer (2022)
31. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193* (2023)
32. Park, H., Chen, R., Rim, P., Lao, D., Wong, A.: Orcas: Unsupervised depth completion via occluded region completion as supervision. *The Fourteenth International Conference on Learning Representations* (2026)
33. Park, H., Gupta, A., Wong, A.: Test-time adaptation for depth completion. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 20519–20529 (2024)
34. Peer, P., Solina, F.: Panoramic depth imaging: Single standard camera approach. *International Journal of Computer Vision* **47**(1), 149–160 (2002)
35. Piccinelli, L., Sakaridis, C., Segu, M., Yang, Y.H., Li, S., Abbeloos, W., Van Gool, L.: Unik3d: Universal camera monocular 3d estimation. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 1028–1039 (2025)
36. Piccinelli, L., Sakaridis, C., Yang, Y.H., Segu, M., Li, S., Abbeloos, W., Van Gool, L.: Unidepthv2: Universal monocular metric depth estimation made simpler. *arXiv preprint arXiv:2502.20110* (2025)
37. Piccinelli, L., Yang, Y.H., Sakaridis, C., Segu, M., Li, S., Van Gool, L., Yu, F.: Unidepth: Universal monocular metric depth estimation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 10106–10116 (2024)
38. 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)
39. Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., Koltun, V.: Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-Shot Cross-Dataset Transfer. *IEEE Transactions on Pattern Analysis & Machine Intelligence* **44**(03), 1623–1637 (2022)
40. Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* (2021)
41. Rim, P., Park, H., Ezhov, V., Moon, J., Wong, A.: Radar-guided polynomial fitting for metric depth estimation. *arXiv preprint arXiv:2503.17182* (2025)
42. Rim, P., Park, H., Zeng, Z., Chung, Y., Wong, A.: Protodepth: Unsupervised continual depth completion with prototypes. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 6304–6316 (2025)
43. Roberts, M., Ramapuram, J., Ranjan, A., Kumar, A., Bautista, M.A., Paczan, N., Webb, R., Susskind, J.M.: Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 10912–10922 (2021)
44. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. *ArXiv abs/1505.04597* (2015)
45. Shen, Z., Lin, C., Liao, K., Nie, L., Zheng, Z., Zhao, Y.: Panoformer: panorama transformer for indoor 360 depth estimation. In: *European Conference on Computer Vision*. pp. 195–211. Springer (2022)

46. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: European Conference on Computer Vision (2012), <https://api.semanticscholar.org/CorpusID:545361>
47. Singh, A.D., Ba, Y., Sarker, A., Zhang, H., Kadambi, A., Soatto, S., Srivastava, M., Wong, A.: Depth estimation from camera image and mmwave radar point cloud. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9275–9285 (2023)
48. Sun, P., Kretschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2446–2454 (2020)
49. Upadhyay, R., Zhang, H., Ba, Y., Yang, E., Gella, B., Jiang, S., Wong, A., Kadambi, A.: Enhancing diffusion models with 3d perspective geometry constraints. *ACM Transactions on Graphics (TOG)* **42**(6), 1–15 (2023)
50. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. *Advances in neural information processing systems* **30** (2017)
51. Wang, F.E., Yeh, Y.H., Sun, M., Chiu, W.C., Tsai, Y.H.: Bifuse: Monocular 360 depth estimation via bi-projection fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 462–471 (2020)
52. Wang, N.H., Solarte, B., Tsai, Y.H., Chiu, W.C., Sun, M.: 360sd-net: 360 stereo depth estimation with learnable cost volume. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 582–588. IEEE (2020)
53. Wang, Q., Zheng, S., Yan, Q., Deng, F., Zhao, K., Chu, X.: Irs: A large naturalistic indoor robotics stereo dataset to train deep models for disparity and surface normal estimation. In: 2021 IEEE International Conference on Multimedia and Expo (ICME). pp. 1–6. IEEE (2021)
54. Wong, A., Chen, A., Wu, Y., Cicek, S., Tiard, A., Hong, B.W., Soatto, S.: Small lesion segmentation in brain mris with subpixel embedding. In: International MIC-CAI Brainlesion Workshop. pp. 75–87. Springer (2021)
55. Wong, A., Cicek, S., Soatto, S.: Learning topology from synthetic data for unsupervised depth completion. *IEEE Robotics and Automation Letters* **6**(2), 1495–1502 (2021)
56. Wong, A., Fei, X., Hong, B.W., Soatto, S.: An adaptive framework for learning unsupervised depth completion. *IEEE Robotics and Automation Letters* **6**(2), 3120–3127 (2021)
57. Wong, A., Fei, X., Tsuei, S., Soatto, S.: Unsupervised depth completion from visual inertial odometry. *IEEE Robotics and Automation Letters* **5**(2), 1899–1906 (2020)
58. Wong, A., Soatto, S.: Bilateral cyclic constraint and adaptive regularization for unsupervised monocular depth prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5644–5653 (2019)
59. Wong, A., Soatto, S.: Unsupervised depth completion with calibrated backprojection layers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12747–12756 (2021)
60. Wu, W., Zhang, Z., Tan, X., Zhang, Z., Ma, L.: Depthfisheye: Efficient fine-tuning of depth estimation models for fisheye cameras. In: International Conference on Computational Visual Media. pp. 3–18. Springer (2025)
61. Wu, Y., Liu, T.Y., Park, H., Soatto, S., Lao, D., Wong, A.: Augundo: Scaling up augmentations for monocular depth completion and estimation. In: European Conference on Computer Vision. pp. 274–293. Springer (2024)

62. Xie, B., Cao, J., Xie, J., Khan, F.S., Pang, Y.: Sed: A simple encoder-decoder for open-vocabulary semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3426–3436 (2024)
63. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. In: Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems (2021)
64. Yan, Q., Wang, Q., Zhao, K., Li, B., Chu, X., Deng, F.: Spheredepth: Panorama depth estimation from spherical domain. In: 2022 International Conference on 3D Vision (3DV). pp. 1–10. IEEE (2022)
65. Yang, L., Kang, B., Huang, Z., Xu, X., Feng, J., Zhao, H.: Depth anything: Unleashing the power of large-scale unlabeled data. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
66. Yang, L., Kang, B., Huang, Z., Zhao, Z., Xu, X., Feng, J., Zhao, H.: Depth anything v2. Advances in Neural Information Processing Systems **37**, 21875–21911 (2024)
67. Yang, Y., Wong, A., Soatto, S.: Dense depth posterior (ddp) from single image and sparse range. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3353–3362 (2019)
68. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12–22 (2023)
69. Yin, W., Liu, Y., Shen, C., Yan, Y.: Enforcing geometric constraints of virtual normal for depth prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5684–5693 (2019)
70. Yogamani, S., Hughes, C., Horgan, J., Sistu, G., Varley, P., O’Dea, D., Uricár, M., Milz, S., Simon, M., Amende, K., et al.: Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9308–9318 (2019)
71. Zhang, D., Liu, F., Tang, Q.: Correclip: Reconstructing patch correlations in clip for open-vocabulary semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 24677–24687 (2025)
72. Zhang, Y., Guo, M.H., Wang, M., Hu, S.M.: Exploring regional clues in clip for zero-shot semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3270–3280 (2024)
73. Zhao, G., Liu, Y., Qi, W., Ma, F., Liu, M., Ma, J.: Fisheyedepth: A real scale self-supervised depth estimation model for fisheye camera. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 3780–3787. IEEE (2025)