Continual Learning of Unsupervised Monocular Depth from Videos

Hemang Chawla[1,*], Arnav Varma[2,*], Elahe Arani[1,3,†], and Bahram Zonooz[1,2,†]
[1]Eindhoven University of Technology (TU/e) [2]TomTom [3]Wayve
{h.chawla, e.arani, b.zonooz}@tue.nl, arnav.varma@tomtom.com

Abstract

Spatial scene understanding, including monocular depth estimation, is an important problem in various applications such as robotics and autonomous driving. While improvements in unsupervised monocular depth estimation have potentially allowed models to be trained on diverse crowdsourced videos, this remains underexplored as most methods utilize the standard training protocol wherein the models are trained from scratch on all data after new data is collected. Instead, continual training of models on sequentially collected data would significantly reduce computational and memory costs. Nevertheless, naive continual training leads to catastrophic forgetting, where the model performance deteriorates on older domains as it learns on newer domains, highlighting the trade-off between model stability and plasticity. While several techniques have been proposed to address this issue in image classification, the high-dimensional and spatiotemporally correlated outputs of depth estimation make it a distinct challenge. To the best of our knowledge, no framework or method currently exists focusing on the problem of continual learning in depth estimation. Thus, we introduce a framework that captures the challenges of continual unsupervised depth estimation (CUDE), and define the necessary metrics to evaluate model performance. We propose a rehearsal-based dual-memory method MonoDepthCL, which utilizes spatiotemporal consistency for continual learning in depth estimation, even when the camera intrinsics are unknown.[§]

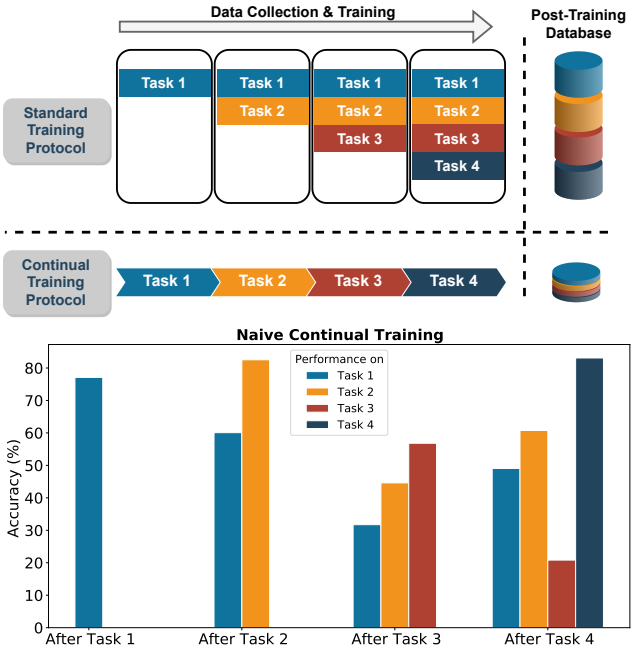

Figure 1. (Top) Standard Training Protocol involves repeatedly training the model from scratch as new data is collected. All collected data is required for future training. The Continual Training Protocol sequentially trains the model as new data is collected. Only a limited amount of data is generally stored when training continually. (Bottom) Naive continual training on the introduced CUDE framework leads to catastrophic forgetting where the depth estimation accuracy [33] on a task is highest after training on that task but is significantly reduced after training on future tasks.

1. Introduction

Vision-based systems have improved tremendously over the years with better semantic and spatial scene understanding capabilities [26]. Particularly, the capability to estimate scene depth has found applications in augmented reality, autonomous navigation, object-grasping, robot-assisted surgery, and more [22, 29, 39, 58]. With the increasing demand for cost-effective, lightweight, and flexible depth estimation solutions, monocular camera-based methods have emerged as a promising alternative to the use of LIDARs or time-of-flight sensors. Moreover, unlike traditional approaches that rely on hand-crafted features from multiple views [47], deep learning has enabled depth estimation from single images [23, 34, 41]. Among these, unsupervised methods that do not require ground-truth labels are often preferred over supervised methods given the associated data-collection costs. These unsupervised methods potentially allow the training of depth estimation models on

*Equal contribution. †Equal advisory role.
§https://github.com/NeurAI-Lab/CUDE-MonoDepthCL.git

crowdsourced videos with unknown camera intrinsics [20]. Nevertheless, it remains an underexplored problem due to the challenges associated with the standard training protocol as shown in Figure 1. Within the standard protocol, the models are retrained from scratch on all the available data after new data is collected, thus incurring high computational time and energy costs. Furthermore, to ensure that the models can be updated in the future, it is necessary to continue storing all the collected data post-training.

Instead, the continual training protocol involves incremental training of the model in sequential or stream data from different domains [35]. Continual Learning (CL) is also necessitated for practical deployments of depth estimation networks, such as in the case of a self-driving car or robot navigating through various locations, or when adapting models pre-trained on simulated data to real environments [46]. Additionally, privacy concerns may be necessary in scenarios where the robot does not have permission to store data on a server to train and update its models [7], and has limited on-board memory. However, adapting the model for the new domain is not sufficient, as it must maintain its depth estimation capability on earlier domains as well. Therefore, there is a fundamental dilemma between maintaining the stability of the old information and the plasticity to adapt to new information for CL [1, 35]. Naive CL without a strategy to handle this dilemma will result in catastrophic forgetting [38], as can be seen in Figure 1.

CL methods such as regularization [31,48,60], parameter isolation [2, 30, 59], and rehearsal [6, 8, 45] have been utilized to address this dilemma for image classification. However, depth estimation is a distinct challenge because of its high-dimensional output space. Additionally, it is typically required to make highly correlated predictions through the use of local spatial relations within images and temporal relations within videos, where the depth at one pixel may depend on the depth at other pixels. Thus, there is a need for a framework that includes a set of sequential tasks that represent these challenges, as well as suitable metrics for evaluating the performance of the models. Thereafter, methods can be designed to consider spatiotemporal relations while addressing the stability-plasticity dilemma specifically for CL in unsupervised monocular depth estimation.

Hence, we introduce a framework for continual unsupervised depth estimation (*CUDE*) to overcome the aforementioned gap. It consists of a setup of four tasks, each on a different dataset with unique characteristics representative of the challenges of domain and depth range shifts across simulated or real and indoor or outdoor scenes. Additionally, we provide the appropriate error and accuracy metrics that can be used to evaluate the performance of depth estimation trained within the continual training protocol. Finally, we propose a dual-memory rehearsal-based method *MonoDepthCL* with a spatiotemporal consistency loss for CL in depth estimation. We showcase how sequential unsupervised learning of monocular depth across multiple tasks enables the development of spatial scene understanding, even when the camera intrinsics may be unknown. Our contributions are as follows:

- We develop a framework for benchmarking of continual learning methods for unsupervised depth estimation under real-world ever-changing scenarios such as different cameras, diverse weather and lighting conditions, disparate depth ranges, and sim-to-real, indoor-to-outdoor, and outdoor-to-indoor domain shifts.
- We define metrics to evaluate the continual learning methods in the framework such that they capture various aspects of continual learning performance such as final performance, performance across the learning trajectory, and stability-plasticity trade-off.
- We propose a method - *MonoDepthCL*, for continual learning of unsupervised monocular depth estimation using multiple models to explicitly capture stability and plasticity separately. Aided by a novel spatiotemporal consistency loss, *MonoDepthCL* proves to be effective for continual learning and dealing with the stability-plasticity trade-off. *MonoDepthCL* is also shown to be effective even when the camera intrinsics are unknown.

2. Related Works

2.1. Monocular Unsupervised Depth Estimation

Monocular depth estimation is considered an important computer vision task for many applications. With the advent of deep learning, several supervised [15,44], as well as unsupervised [5, 19] approaches for depth estimation have been proposed. Further research has improved the depth estimation results in independent and identically distributed (i.i.d.) training using multiple modalities [12,23], newer architectures [24, 52], advances in feature extraction [36, 49], and 3D geometry [4, 56]. Nevertheless, estimating dense depths from monocular images for continually shifting distributions, where the previous data become unavailable as in the real world, remains an understudied problem.

2.2. Continual Learning for Dense Prediction

Most of the CL research has focused on classification tasks, with limited attention to dense prediction tasks. Early works on CL in image classification focused on regularizing important parameters for previous tasks [31, 48, 60], or complete or partial isolation of parameters relevant to each task [2, 30, 59]. However, these approaches generally undergo catastrophic forgetting without task identification at test time [16]. Rehearsal-based methods, such as experience replay (ER) [45] resolve this issue by retraining on

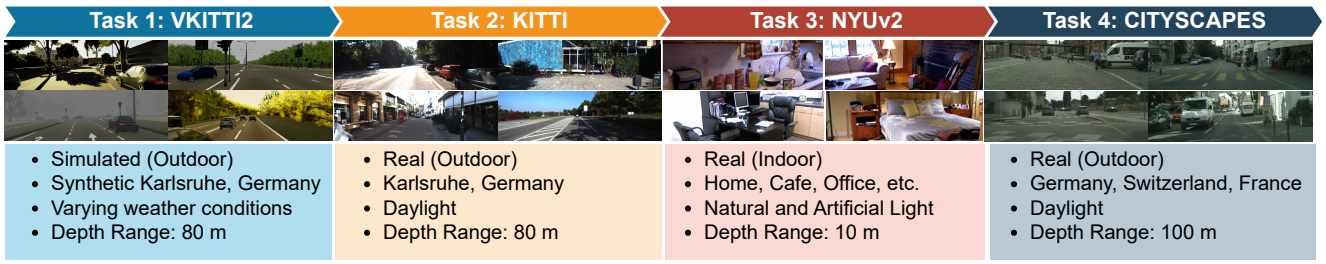

Figure 2. Continual Unsupervised Depth Estimation (CUDE) framework with four sequential tasks operating in diverse environments, weather and lighting conditions, and depth ranges.

old data stored in a small memory buffer updated with replacement [3, 6, 8, 42]. Nevertheless, it is non-trivial to extend these methods to continual learning for dense prediction tasks. Although some methods have been developed that focus on continual semantic segmentation [14, 37], they continue to address the challenge posed by the addition of newer class labels. Recently, some works have focused on related issues of domain adaptation for the spatial task of depth estimation [32, 61]. These methods rarely focus on the issue of catastrophic forgetting or are limited to a one-step transfer of learned knowledge to a single new environment. Voedisch *et al*. [54] focuses on the related task of odometry. However, it uses an infinite buffer, giving it access to all previously seen data at all times, additionally causing high memory expense and privacy concerns. To the best of our knowledge, no existing work deals with the challenges of CL in dense unsupervised depth estimation. Therefore, we introduce a framework for comprehensive evaluation of CL methods for unsupervised depth estimation, and develop a rehearsal-based method for the same.

3. Framework

Since depth estimation is a regression method with no distinct classes, CL for depth estimation is akin to a domain-incremental learning scenario, in which the input distribution changes as training progresses. This is reflected in Figure 1 where the continual training protocol is different from standard training, where the model is trained on all available datasets simultaneously. We, thus, define the CUDE framework with four tasks, each corresponding to an individual dataset as shown in Figure 2. Each task corresponds to training the depth on a unique set of videos captured in varying environments from different cameras, under diverse weather and lighting conditions, and capturing disparate depth ranges to mimic real-world ever-changing scenarios. The task sequence is VKITTI2 [10] $\rightarrow$ KITTI [17] $\rightarrow$ NYUv2 [50] $\rightarrow$ Cityscapes [13]. Here, VKITTI2 is a virtual photo-realistic dataset generated using the Unity game engine for the simulated urban setting of Karlsruhe, Germany in various imaging and weather conditions. KITTI is an outdoor scenario dataset captured in Karlsruhe, Germany, consisting of challenging scenes from both urban and highway scenarios with ground truth depth measured from a LiDAR. NYUv2 is an indoor dataset consisting of images and the corresponding ground truth depth captured from a Kinect RGBD camera. Cityscapes is another outdoor vision dataset captured in multiple locations in Germany, France, and Switzerland with ground truth depth measured using stereo vision.

This task order also spans complex domain shifts such as indoor-to-outdoor, outdoor-to-indoor, and sim-to-real. Though there could have been 24 possible task orders, we eliminate those sequences where the simulated dataset would be in the middle or end of the sequence, as they are not applicable to real-world scenarios, which typically have deployment in the real-world as the target. Additionally, 4 of the remaining 6 permutations lack either the indoor-to-outdoor or outdoor-to-indoor domain shift. Finally, from the remaining 2 sequences, the selected sequence allows us to demonstrate the impact of transitioning from sim-to-real for the same scene and camera setup, which is more realistic as robots are often trained first on the closest possible simulated version of the targeted environment before obtaining data for the real-world environment. Such a training sequence could be utilized to train perception systems for robots that operate both indoors and outdoors, such as security robots, hygiene robots, assistance robots, etc. By capturing domain shifts from simulated to real environments, outdoor to indoor environments, and vice versa, our setup aims to examine both the forgetfulness and adaptability of the CL models across different camera setups, scene distributions, and depth ranges.

Generally, multiple *error* and *accuracy* metrics are used to evaluate depth estimation performance [11]. However, these metrics are not directly suitable for quantifying CL methods for depth estimation. Instead, by measuring the error and accuracy of depth estimation on each task after training a specific task, we generate a task-wise performance matrix $A \in \mathbb{R}^{n_t \times n_t}$, where n_t is the total number of tasks. Consequently, we use the following metrics to eval-

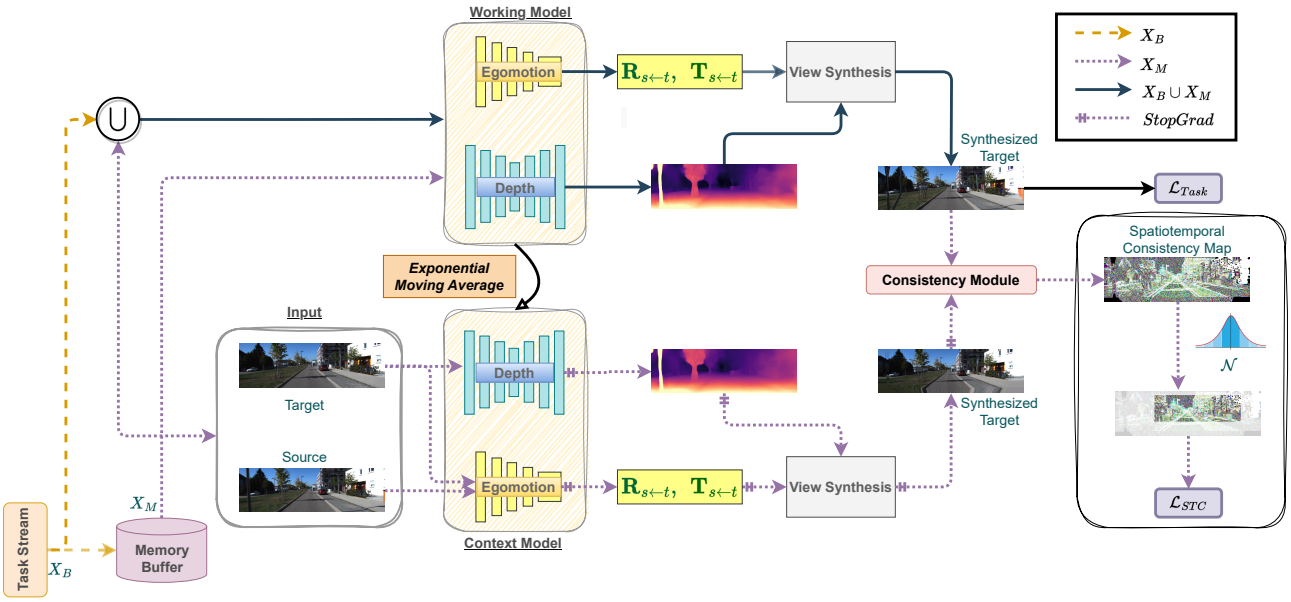

Figure 3. Schematic of our MonoDepthCL: The context model consolidates the working model using an exponential moving average. The working model learns a view synthesis (See Section 4) of the target image from the source images. A consistency module contrasts synthesized target images from both models to generate a spatiotemporal consistency map, which is cropped to a random ratio sampled from a Gaussian distribution to get a spatiotemporal consistency loss (See Algorithm 1). Note that the depth estimation and associated operations are done at 4 resolutions.

uate CL models for depth estimation. Note that depending upon whether the metric is computed for the depth errors or the depth accuracy, the lower values or the higher values indicate better performance, respectively.

Final Average (μ_{final}) measures the mean performance on all tasks after training the model on the final task. Hence, $\mu_{\text{final}} = \frac{1}{n_t}\sum_{j=1}^{n_t} A_{n_t,j}$.

Overall Average (μ_{overall}) measures the μ_{final} over all the seen tasks after training the model on each task. Hence, $\mu_{\text{overall}} = \frac{2}{n_t(n_t+1)}\sum_{i\geq j}^{n_t} A_{i,j}$. It indicates the improvements and degradations of model performance as the model is trained on different tasks.

Stability-Plasticity Trade-off (SPTO) measures how well the model tackles the dilemma between retaining performance on the previously seen tasks and the capability to learn new tasks. Hence, SPTO $= \frac{2\times A_{\mathcal{S}}\times A_{\mathcal{P}}}{A_{\mathcal{S}}+A_{\mathcal{P}}}$; where stability $A_{\mathcal{S}} = \frac{1}{n_t}\sum_{j=1}^{n_t-1} A_{n_t,j}$ is the average performance on all previously seen tasks after training the model on the final task, and plasticity $A_{\mathcal{P}} = \frac{1}{n_t}\sum_{i=1}^{n_t} A_{i,i}$ is the average performance of the tasks after the model is trained on them for the first time.

4. Method

Here, we provide an overview of unsupervised monocular depth estimation, followed by our method for continual learning in depth estimation.

4.1. Unsupervised Monocular Depth Estimation

Unsupervised monocular depth estimation is a technique used to determine the pixelwise distance of objects in a scene from a single unlabeled image. At any given training step, the input to the models is a set of temporally consecutive images, consisting of a *target* image $I_t \in \mathbb{R}^{H\times W\times 3}$, and n_s *source* images $\{I_s^j \in \mathbb{R}^{H\times W\times 3} : j = 1, 2, ...n_s\}$, where H and W are the height and width, respectively, of the images. The depth network parameterized by θ_D predicts inverse depths at four resolutions, which are then bilinearly upsampled to the input resolution to reduce texture copy artifacts [19]. Meanwhile, the ego-motion network parameterized by θ_E predicts the relative pose between each source-target image pair concatenated along the channel dimension. Combining this relative pose with the camera intrinsics matrix, each source image I_s^j is warped to the target image using the perspective projection equation [28]. This process, called *view synthesis*, is performed for each upsampled target depth prediction indexed by $i = 1, 2, 3, 4$ to obtain a synthesized target image $\hat{I}_{i,t}^j$. An appearance-based photometric error is then formed between each synthesized target image and the original target image I_t. The photometric loss connects the predictions of the depth and ego-motion networks, forming the basis for unsupervised learning of depth. Additionally, to counteract the impact of temporally stationary pixels (e.g. when there is object motion but no ego-motion), the photometric loss is only con-

Algorithm 1 Algorithm for computing spatiotemporal consistency loss $\mathcal{L}_{\text{STC}}$.

Input: Per-pixel spatiotemporal consistencies from consistency module on X_M for each source j and prediction i, $STC_i^j \in \mathcal{R}^{H \times W}$
Initialize: $\mathcal{L}_{\text{STC}}(X_M) = 0.0$
for $j \leftarrow 1$ to n_s **do**
  $STC^j \leftarrow 0.0$
  for $i \leftarrow 1$ to 4 **do**
    $r \sim \mathcal{N}(0.5, 0.1)$ ▷ Sample ratio for cropping
    $r \leftarrow \texttt{Clip}(r, \texttt{min} = 0.1, \texttt{max} = 1.0)$ ▷ Clip ratio between 0.1 and 1
    $H^{'}, W^{'} \leftarrow rH, rW$ ▷ Height and width for cropping
    $top \leftarrow \texttt{randint}(1, H^{'} - H + 1)$ ▷ Sample start row for cropping
    $left \leftarrow \texttt{randint}(1, W^{'} - W + 1)$ ▷ Sample start column for cropping
    $p_t^{'} \leftarrow \{(x, y) \; \forall \; (x, y) \in [top, top + H^{'} - 1] \times [left, left + W^{'} - 1] \cap \mathcal{Z}\}$ ▷ Set of cropped pixels
    $STC^j \leftarrow STC^j + \frac{1}{|p_t^{'}|} \sum_{p_t^{'}} STC_i^j[p_t^{'}]$ ▷ Add average of cropped losses
  $\mathcal{L}_{\text{STC}}(X_M) \leftarrow \mathcal{L}_{\text{STC}}(X_M) + STC^j/4$ ▷ Add average across predictions
$\mathcal{L}_{\text{STC}}(X_M) \leftarrow \mathcal{L}_{\text{STC}}(X_M)/n_s$ ▷ Average across source images
return $\mathcal{L}_{\text{STC}}(X_M)$

sidered at pixel locations where it is lower than the photometric loss between the unwarped source and target image at each scale. This procedure is known as automasking [19]. Finally, a per-pixel edge-aware *smoothness loss* is used to regularize the depth predictions [18]. The masked photometric loss and the smoothness loss together form the total training loss for unsupervised depth estimation, denoted by $\mathcal{L}_{depth}$ (see Supplementary Material for more details).

4.2. Continual Learning for Unsupervised Monocular Depth Estimation

Humans continually learn from new experiences without catastrophically forgetting previous experiences [25]. The complementary learning systems (CLS) theory postulates that human learning involves complex interactions between complementary learning systems that are learning at different rates [40]. This includes a fast working system adapting quickly to new experiences and a slow context system consolidating knowledge from the fast systems. One such interaction could be the replay of sequences from memory such that the fast system works in the context of consolidated representations of the slow system [21]. The fast and slow systems help model plasticity and stability, respectively. Thus, we formulate a continual learning method for unsupervised monocular depth estimation with dual-memories and replay, which we call MonoDepthCL.

Concretely, consider *working* depth and ego-motion model $\mathcal{WM}$ parameterized by $\theta^{\mathcal{WM}} = \theta_D^{\mathcal{WM}} \cup \theta_E^{\mathcal{WM}}$ and *context* depth and ego-motion model $\mathcal{CM}$ parameterized by $\theta^{\mathcal{CM}} = \theta_D^{\mathcal{CM}} \cup \theta_E^{\mathcal{CM}}$. The working model is learnable, while the context model is maintained as an exponential moving average (EMA) of the working model [3]. For replay, we employ a bounded *memory* buffer M, updated by reservoir sampling [53], which allows the buffer to approximate the distribution of samples seen by the models [27]. For an update coefficient α and update frequency $\nu \in (0, 1)$, we update the context models in the training iteration n to get,

$$\theta^{\mathcal{CM}} = \alpha_n \theta^{\mathcal{CM}} + (1 - \alpha_n)\theta^{\mathcal{WM}}, \tag{1}$$

where $\alpha_n = min(1 - 1/(n + 1), \alpha)$.

However, this only helps with addressing forgetting in the context model. Ideally, the working model should have a mechanism to retain prior knowledge, which is learning in the context of consolidated representations [21]. Furthermore, if the working model experiences catastrophic forgetting, it would also negatively affect the context model (Eq. 1), which underscores the need for such a mechanism. Therefore, we distill the knowledge of consolidated representations of memory samples from the context model back to the working model. Since depth estimation involves training via view synthesis, we ensure consistency in the synthesized targets between the context and working models. This guarantees spatial consistency in the consolidated depth maps of the target images and temporal consistency in the poses between the target and nearby source images. We refer to this as spatiotemporal consistency.

Specifically, at each training step, we sample a batch X_B from the current task stream and a batch X_M from the memory buffer. On each memory sample, we warp the source images to the target image using both the working and context models. Let $\hat{I}_{i,t}^{j,\mathcal{CM}}$ and $\hat{I}_{i,t}^{j,\mathcal{WM}}$ represent the synthesized targets for the j^{th} source image and the i^{th} depth prediction using context and working models, respectively. Then, a consistency module computes a per-pixel spatiotemporal consistency between these synthesized targets as follows:

$$\begin{aligned} STC_i^j = & \frac{\rho}{2}[1 - SSIM(\hat{I}_{i,t}^{j,\mathcal{CM}}, \hat{I}_{i,t}^{j,\mathcal{WM}})] \\ & + (1 - \rho)\left|\hat{I}_{i,t}^{j,\mathcal{CM}} - \hat{I}_{i,t}^{j,\mathcal{WM}}\right|, \end{aligned} \tag{2}$$

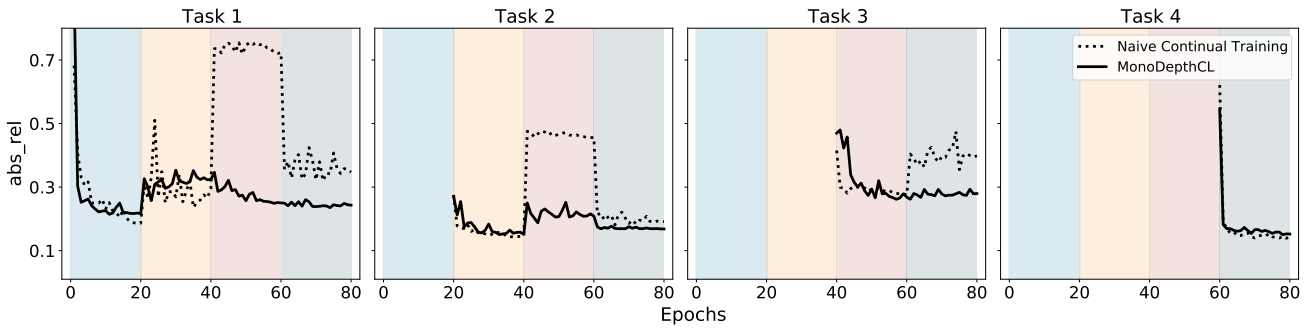

Figure 4. Performance on the tasks over epochs as models are trained with buffer size 200. Naive Continual training undergoes catastrophic forgetting, while MonoDepthCL mitigates it to a good extent.

where SSIM refers to the Structural Similarity Index between two images [57]. Each per-pixel consistency STC_i^j is an image-shaped loss of shape $H \times W$. To reduce the overfitting to buffer samples [9], improve invariance through augmentation [43], and improve efficiency, we randomly crop these per-pixel consistencies before evaluating the final spatiotemporal consistency loss $\mathcal{L}_{\text{STC}}$. We sample the ratios to be cropped for each consistency from a Gaussian distribution with 0.5 mean and 0.1 standard deviation. This would retain half the $H \times W$ consistency term on average, and retain 20-80% of the term $\sim$99.7% of the times (i.e. within 3 standard deviations). These cropped losses are then averaged across all predictions, source images, and cropped pixels to get $\mathcal{L}_{\text{STC}}$. Algorithm 1 shows the process to compute the cropped spatiotemporal consistency loss. Note that we use a warmup for the spatiotemporal consistency loss, such that it is only applied after the first task is learned. This allows the view synthesis to be learned well before it is used as a constraint between the context and working models.

Finally, the working model needs to learn the new task and is thus trained with a task loss $\mathcal{L}_{Task}$, which is the depth loss $\mathcal{L}_{depth}$ discussed earlier, on the union of current and memory batches. Putting everything together, the total loss for continual unsupervised monocular depth estimation is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{Task}(X_B \cup X_M) + \beta \mathcal{L}_{\text{STC}}(X_M). \quad (3)$$

We dub our method as ***Mono**cular **Depth** estimation with **C**ontinual **L**earning* or ***MonoDepthCL***. The complete schematic of our method can be seen in Figure 3.

5. Results

We demonstrate CL challenges in unsupervised monocular depth estimation on the CUDE framework and the effectiveness of MonoDepthCL for mitigating catastrophic forgetting. The architecture details and hyperparameters used can be found in the Supplementary Material. We compare against naive continual training (NCT), and training on the whole dataset, i.e. its joint distribution after collecting data from all tasks, dubbed *Joint*, following standard practice in the CL literature [8, 45]. NCT and Joint form the lower and upper bounds for CL, respectively. Our method has two models connected by a spatiotemporal consistency loss. Consequently, we consider two additional CL methods stemming from these models in the absence of the proposed spatiotemporal consistency loss. The fast-learning working model, if trained in isolation, would learn merely on the task loss with rehearsal, and should be capable of mitigating forgetting to some extent. This is equivalent to ER (which has already been shown to outperform most other rehearsal-based methods for classification [51]). Similarly, the context model, which maintains an exponential moving average of the working model, should also be capable of mitigating forgetting to some extent. We call this method ContextDepth.

Figure 4 shows a significant drop in the performance of NCT on each task as newer tasks are encountered, undergoing catastrophic forgetting as seen earlier in Figure 1. This is in contrast to MonoDepthCL, which improves over NCT, undergoing far less forgetting across tasks. Table 1 additionally shows that MonoDepthCL outperforms other CL methods in all buffer sizes across all metrics, demonstrating the effectiveness of the spatiotemporal consistency loss.

Increasing the buffer size leads to a general improvement across metrics for all CL methods, including MonoDepthCL. However, at low buffer size (50), ER and ContextDepth fall behind even NCT on some SPTO and $\mu_{\textbf{final}}$ metrics. Nevertheless, they outperform NCT on $\mu_{\textbf{overall}}$. This is because $\mu_{\textbf{overall}}$ measures the improvements and degradations of model performance across the task trajectory, and not just the mean accuracy after learning the final task. When rehearsal and dual model approach are combined with our spatiotemporal consistency loss, MonoDepthCL performs well on *all* metrics for the low buffer size as well. Consequently, we hypothesize that the higher performance of MonoDepthCL on the $\mu_{\textbf{overall}}$ metrics indicates its ability to learn on additional tasks. The per-

Buffer	Method	μ_{final}			μ_{overall}			SPTO		
		abs_rel↓	RMSE↓	a1↑	abs_rel↓	RMSE↓	a1↑	abs_rel↓	RMSE↓	a1↑
–	Joint	0.177	5.408	0.759		–			–	
	NCT	0.272	7.397	0.639	0.315	8.253	0.580	0.238	6.313	0.648
50	ER	0.289	7.421	0.619	0.310	7.770	0.580	0.266	6.422	0.625
	ContextDepth	0.274	7.310	0.631	0.290	7.669	0.594	0.242	6.287	0.644
	MonoDepthCL	**0.249**	**6.774**	**0.647**	**0.265**	**7.168**	**0.618**	**0.237**	**5.987**	**0.652**
200	ER	0.248	6.995	0.666	0.238	6.852	**0.679**	0.228	6.047	0.673
	ContextDepth	0.286	7.177	0.644	0.296	7.368	0.621	0.265	6.271	0.644
	MonoDepthCL	**0.228**	**6.583**	**0.673**	**0.237**	**6.753**	0.663	**0.223**	**5.832**	**0.676**
500	ER	0.261	6.879	0.680	0.236	6.586	0.691	0.242	6.033	0.678
	ContextDepth	0.252	6.695	0.677	0.239	6.585	0.677	0.242	5.932	0.672
	MonoDepthCL	**0.228**	**6.278**	**0.693**	**0.219**	**6.239**	**0.701**	**0.222**	**5.573**	**0.693**

Table 1. Performance of different methods on the CUDE framework for multiple buffer sizes. The best results for each buffer size are shown in bold.

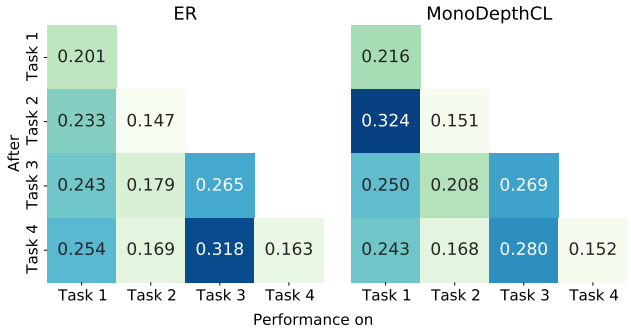

Figure 5. Taskwise errors after training on each task for buffer size 200. A more uniform distribution of color in a row indicates a lower task-task recency bias after learning the task corresponding to that row. ER shows a higher bias towards recent tasks.

formance of MonoDepthCL on the SPTO metrics further confirms that it is better equipped to handle the stability-plasticity trade-off compared to rehearsal-based ER. The stronger SPTO performance of MonoDepthCL also translates to lower task-recency bias as seen in Figure 5. The distribution of performance across previous tasks after learning each task is more uniform for our method than for ER. This is in line with recent findings in image classification [3].

Hence, benchmarking the methods on our CUDE framework demonstrates the challenge of CL for unsupervised monocular depth estimation. Our experiments additionally show that the metrics defined in CUDE capture different aspects of CL. We contend that our method with its spatiotemporal consistency loss is an effective strategy for handling the stability-plasticity trade-off in CL.

With Learned Intrinsics: Now, unsupervised monocular depth estimation requires knowledge of the camera intrinsics for the perspective projection (Section 4). This forms a roadblock for training on crowdsourced video sequences where this information is not available. While

Method	μ_{final}		
	abs_rel↓	RMSE↓	a1↑
Joint	0.180	5.455	0.752
NCT	0.264	7.280	0.640
MonoDepthCL	0.252	7.237	0.663

Table 2. Camera intrinsics K are learned during training with a buffer size 200. MonoDepthCL reduces catastrophic forgetting even without prior knowledge of camera intrinsics.

recent research has demonstrated unsupervised monocular depth estimation with learned camera intrinsics in joint training, it has not been explored in the CL setting. We train our proposed MonoDepthCL with learned intrinsics and benchmark it on the CUDE framework. The results in Table 2 demonstrate that MonoDepthCL outperforms NCT across all μ_{final} metrics. Although a considerable gap still exists between Joint and MonoDepthCL, our study represents the pioneering effort towards crowdsourced CL for unsupervised monocular depth estimation.

5.1. Ablation Study

Our spatiotemporal consistency loss has two major design components - warmup during the first task to allow the view synthesis to be learned properly before it is applied as a constraint between the working and context models; and random cropping of the spatiotemporal consistency map to introduce invariance through augmentation and improve efficiency. Table 3 details the impact of both of these design components on the continual learning metrics. It can be seen that warmup contributes to a great improvement in continual learning performance, and adding random cropping leads to a further improvement over warmup. Therefore, both the design components of our spatiotemporal consistency loss have a significant impact on the continual learning performance.

Random cropping	Warmup	μ_{final}			μ_{overall}			SPTO		
		abs_rel↓	RMSE↓	a1↑	abs_rel↓	RMSE↓	a1↑	abs_rel↓	RMSE↓	a1↑
✗	✗	0.306	7.679	0.625	0.333	8.187	0.597	0.290	6.837	0.621
✗	✓	0.237	6.623	0.669	0.239	**6.707**	**0.663**	0.231	5.869	0.669
✓	✓	**0.228**	**6.583**	**0.673**	**0.237**	6.753	**0.663**	**0.223**	**5.832**	**0.676**

Table 3. Ablation of design components of the spatiotemporal consistency loss on the CUDE framework for buffer size 200. The best results are shown in bold.

Method	μ_{final}			μ_{overall}			SPTO		
	abs_rel↓	RMSE↓	a1↑	abs_rel↓	RMSE↓	a1↑	abs_rel↓	RMSE↓	a1↑
Joint	0.182	7.413	0.755		–			–	
NCT	0.316	9.966	0.631	0.317	8.788	0.597	0.255	7.571	0.64
ER	0.289	9.618	0.637	0.264	7.782	0.649	0.255	7.388	0.637
ContextDepth	0.293	9.666	0.635	0.282	8.026	0.631	0.264	7.439	0.631
MonoDepthCL	**0.246**	**8.626**	**0.664**	**0.242**	**7.353**	**0.656**	**0.237**	**6.773**	**0.655**

Table 4. Performance of different methods on the CUDE-5 framework for buffer-size 200. The best results are shown in bold.

5.2. Longer Task Sequence (CUDE-5)

In Section 5, we noted that MonoDepthCL is expected to perform well on longer task sequences as well. Accordingly, we extend the framework CUDE to 5 tasks with an additional task from dataset DDAD [23] as shown in the Supplementary Material (Figure S1). Since DDAD contains videos from USA and Japan, and captures the domain shifts from one country to another. Additionally, the ground truth depth range for DDAD is 200m which is double than that of Cityscapes. We report the performance on CUDE-5 for buffer size 200 in Table 4. We observe that MonoDepthCL continues to mitigate forgetting and improve performance through spatiotemporal consistency.

With Learned Intrinsics: Similar to CUDE with 4 tasks, we also evaluate the performance of MonoDepthCL when the camera intrinsics of all 5 tasks are learned together with the depth in Table 5. The ability to continually learn depth estimation from additional data, without prior knowledge of camera intrinsics paves the way towards crowdsourced depth estimation.

Method	μ_{final}		
	abs_rel↓	RMSE↓	a1↑
Joint	0.185	7.699	0.751
NCT	0.315	9.868	0.614
MonoDepthCL	0.252	7.237	0.663

Table 5. μ_{final} performance on the CUDE-5 framework when the camera intrinsics K are learned during training. MonoDepthCL reduces catastrophic forgetting even without prior knowledge of camera intrinsics on the longer task sequence as well.

6. Conclusion

We introduce CUDE, a framework for benchmarking CL methods for unsupervised monocular depth estimation, along with our MonoDepthCL method. CUDE consists of four sequential tasks that span different weather and lighting conditions, depth ranges, and navigation scenarios for indoor and outdoor scenes. It also defines metrics that measure the final average performance after learning the final task, the overall average performance throughout the learning trajectory, and the stability-plasticity trade-off. Meanwhile, our proposed method follows a dual-model approach with a memory buffer for storing previously seen information. Specifically, a working model learns the task on samples from memory and data stream, while a context model is maintained as an exponential moving average of the working model. A novel spatiotemporal consistency loss efficiently enforces the view synthesis consistency between the two models. The benchmarking of MonoDepthCL on CUDE shows the effectiveness of our method and the spatiotemporal consistency loss for CL, and in dealing with the stability-plasticity trade-off. It also demonstrates the value of the framework itself, including the defined metrics that capture various aspects of CL performance. Finally, we show the applicability of MonoDepthCL for a scenario where the camera intrinsics also need to be learned.

We put forward our CUDE framework as a first step towards comprehensive benchmarking of CL methods for unsupervised monocular depth estimation. We further contend that our method, MonoDepthCL provides a promising approach towards addressing the performance gap between continual training and joint training.

Acknowledgement: The work was conducted while all the authors were affiliated with NavInfo Europe, Eindhoven, The Netherlands.

References

[1] Wickliffe C Abraham and Anthony Robins. Memory retention–the synaptic stability versus plasticity dilemma. *Trends in neurosciences*, 28(2):73–78, 2005. 2

[2] Tameem Adel, Han Zhao, and Richard E Turner. Continual learning with adaptive weights (claw). *arXiv preprint arXiv:1911.09514*, 2019. 2

[3] Elahe Arani, Fahad Sarfraz, and Bahram Zonooz. Learning fast, learning slow: A general continual learning method based on complementary learning system. In *International Conference on Learning Representations*, 2022. 3, 5, 7

[4] Jiawang Bian, Zhichao Li, Naiyan Wang, Huangying Zhan, Chunhua Shen, Ming-Ming Cheng, and Ian Reid. Unsupervised scale-consistent depth and ego-motion learning from monocular video. *Advances in neural information processing systems*, 32, 2019. 2

[5] Jia-Wang Bian, Huangying Zhan, Naiyan Wang, Tat-Jun Chin, Chunhua Shen, and Ian Reid. Auto-rectify network for unsupervised indoor depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9802–9813, 2021. 2

[6] Matteo Boschini, Lorenzo Bonicelli, Pietro Buzzega, Angelo Porrello, and Simone Calderara. Class-incremental continual learning into the extended der-verse. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2, 3

[7] Daniel J Butler, Justin Huang, Franziska Roesner, and Maya Cakmak. The privacy-utility tradeoff for remotely teleoperated robots. In *Proceedings of the tenth annual ACM/IEEE international conference on human-robot interaction*, pages 27–34, 2015. 2

[8] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020. 2, 3, 6

[9] Pietro Buzzega, Matteo Boschini, Angelo Porrello, and Simone Calderara. Rethinking experience replay: a bag of tricks for continual learning. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2180–2187. IEEE, 2021. 6

[10] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2, 2020. 3

[11] Cesar Cadena, Yasir Latif, and Ian D Reid. Measuring the performance of single image depth estimation methods. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4150–4157. IEEE, 2016. 3

[12] Hemang Chawla, Arnav Varma, Elahe Arani, and Bahram Zonooz. Multimodal scale consistency and awareness for monocular self-supervised depth estimation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5140–5146. IEEE, 2021. 2

[13] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 3

[14] Arthur Douillard, Yifu Chen, Arnaud Dapogny, and Matthieu Cord. Plop: Learning without forgetting for continual semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4040–4050, 2021. 3

[15] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014. 2

[16] Sebastian Farquhar and Yarin Gal. Towards Robust Evaluations of Continual Learning. *Lifelong Learning: A Reinforcement Learning Approach Workshop at ICML*, 2018. 2

[17] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 3

[18] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 270–279, 2017. 5

[19] 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. 2, 4, 5

[20] Ariel Gordon, Hanhan Li, Rico Jonschkowski, and Anelia Angelova. Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8977–8986, 2019. 2

[21] Anirudh Goyal and Yoshua Bengio. Inductive biases for deep learning of higher-level cognition. *Proceedings of the Royal Society A*, 478(2266):20210068, 2022. 5

[22] Vitor Guizilini, Rares Ambrus, Wolfram Burgard, and Adrien Gaidon. Sparse auxiliary networks for unified monocular depth prediction and completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11078–11088, 2021. 1

[23] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2485–2494, 2020. 1, 2, 8

[24] Vitor Guizilini, Rareș Ambruș, Dian Chen, Sergey Zakharov, and Adrien Gaidon. Multi-frame self-supervised depth with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 160–170, 2022. 2

[25] Raia Hadsell, Dushyant Rao, Andrei A Rusu, and Razvan Pascanu. Embracing change: Continual learning in deep neural networks. *Trends in cognitive sciences*, 24(12):1028–1040, 2020. 5

[26] Anthony Hu, Fergal Cotter, Nikhil Mohan, Corina Gurau, and Alex Kendall. Probabilistic future prediction for video

scene understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 767–785. Springer, 2020. 1

[27] David Isele and Akansel Cosgun. Selective experience replay for lifelong learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018. 5

[28] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015. 4

[29] 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. 1

[30] Haeyong Kang, Rusty John Lloyd Mina, Sultan Rizky Hikmawan Madjid, Jaehong Yoon, Mark Hasegawa-Johnson, Sung Ju Hwang, and Chang D Yoo. Forget-free continual learning with winning subnetworks. In *International Conference on Machine Learning*, pages 10734–10750. PMLR, 2022. 2

[31] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 2

[32] Yevhen Kuznietsov, Marc Proesmans, and Luc Van Gool. Comoda: Continuous monocular depth adaptation using past experiences. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2907–2917, 2021. 3

[33] Lubor Ladicky, Jianbo Shi, and Marc Pollefeys. Pulling things out of perspective. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 89–96, 2014. 1

[34] Sihaeng Lee, Janghyeon Lee, Byungju Kim, Eojindl Yi, and Junmo Kim. Patch-wise attention network for monocular depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1873–1881, 2021. 1

[35] Timothée Lesort, Vincenzo Lomonaco, Andrei Stoian, Davide Maltoni, David Filliat, and Natalia Díaz-Rodríguez. Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges. *Information fusion*, 58:52–68, 2020. 2

[36] Xiaoyang Lyu, Liang Liu, Mengmeng Wang, Xin Kong, Lina Liu, Yong Liu, Xinxin Chen, and Yi Yuan. Hr-depth: High resolution self-supervised monocular depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021. 2

[37] Andrea Maracani, Umberto Michieli, Marco Toldo, and Pietro Zanuttigh. Recall: Replay-based continual learning in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7026–7035, 2021. 3

[38] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 2

[39] Yue Ming, Xuyang Meng, Chunxiao Fan, and Hui Yu. Deep learning for monocular depth estimation: A review. *Neurocomputing*, 438:14–33, 2021. 1

[40] Randall C O'Reilly, Rajan Bhattacharyya, Michael D Howard, and Nicholas Ketz. Complementary learning systems. *Cognitive science*, 38(6):1229–1248, 2014. 5

[41] Vaishakh Patil, Christos Sakaridis, Alexander Liniger, and Luc Van Gool. P3depth: Monocular depth estimation with a piecewise planarity prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1610–1621, 2022. 1

[42] Quang Pham, Chenghao Liu, and Steven Hoi. Dualnet: Continual learning, fast and slow. *Advances in Neural Information Processing Systems*, 34:16131–16144, 2021. 3

[43] Senthil Purushwalkam and Abhinav Gupta. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *Advances in Neural Information Processing Systems*, 33:3407–3418, 2020. 6

[44] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12179–12188, 2021. 2

[45] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019. 2, 6

[46] Andrei A Rusu, Matej Večerík, Thomas Rothörl, Nicolas Heess, Razvan Pascanu, and Raia Hadsell. Sim-to-real robot learning from pixels with progressive nets. In *Conference on robot learning*, pages 262–270. PMLR, 2017. 2

[47] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 1

[48] Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *International Conference on Machine Learning*, pages 4528–4537. PMLR, 2018. 2

[49] Chang Shu, Kun Yu, Zhixiang Duan, and Kuiyuan Yang. Feature-metric loss for self-supervised learning of depth and egomotion. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX*, pages 572–588. Springer, 2020. 2

[50] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. *ECCV (5)*, 7576:746–760, 2012. 3

[51] Gido M van de Ven, Tinne Tuytelaars, and Andreas S Tolias. Three types of incremental learning. *Nature Machine Intelligence*, 4(12):1185–1197, 2022. 6

[52] Arnav Varma, Hemang Chawla, Bahram Zonooz, and Elahe Arani. Transformers in self-supervised monocular depth estimation with unknown camera intrinsics. In *Proceedings of the 17th International Joint Conference on Computer Vi-*

sion, Imaging and Computer Graphics Theory and Applications, VISIGRAPP 2022, Volume 4: VISAPP, pages 758–769. SCITEPRESS, 2022. 2

[53] Jeffrey S Vitter. Random sampling with a reservoir. *ACM Transactions on Mathematical Software (TOMS)*, 1985. 5

[54] Niclas Vödisch, Daniele Cattaneo, Wolfram Burgard, and Abhinav Valada. Continual slam: Beyond lifelong simultaneous localization and mapping through continual learning. In *Proceedings of the International Symposium on Robotics Research (ISRR)*, 2022. 3

[55] Chaoyang Wang, José Miguel Buenaposada, Rui Zhu, and Simon Lucey. Learning depth from monocular videos using direct methods. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2022–2030, 2018. 1

[56] Lijun Wang, Yifan Wang, Linzhao Wang, Yunlong Zhan, Ying Wang, and Huchuan Lu. Can scale-consistent monocular depth be learned in a self-supervised scale-invariant manner? In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12727–12736, October 2021. 2

[57] 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. 6

[58] Nan Yang, Lukas von Stumberg, Rui Wang, and Daniel Cremers. D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1281–1292, 2020. 1

[59] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. *arXiv preprint arXiv:1708.01547*, 2017. 2

[60] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995. PMLR, 2017. 2

[61] Zhenyu Zhang, Stephane Lathuiliere, Elisa Ricci, Nicu Sebe, Yan Yan, and Jian Yang. Online depth learning against forgetting in monocular videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4494–4503, 2020. 3