2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

Toward Robust and Efficient Online Adaptation for Deep Stereo Depth Estimation

Milo Knowles Valentin Peretroukhin W. Nicholas Greene Nicholas Roy

***Abstract*— Although deep neural networks have achieved state-of-the-art performance for stereo depth estimation, they can suffer from a significant drop in accuracy when tested on images from novel domains. Recent work has shown that self-supervised online adaptation is a promising approach for closing this performance gap. In this work, we address three unsolved challenges for online adaptation. First, we propose a method for detecting novel environments, allowing us to trigger adaptation and notify downstream systems that depth predictions are unreliable. We find that the feature similarity scores from our deep stereo network can be leveraged for *out-of-distribution* (OOD) detection, providing the necessary starting criterion for adaptation. Next, we use *online validation* to terminate adaptation when it stops improving performance, allowing us to free up computational resources. Finally, we demonstrate that existing methods for continuous adaptation cause catastrophic forgetting of the training domain. By augmenting adaptation with *experience replay*, we retain high accuracy in the training domain while rapidly improving performance in novel environments. In sum, these three contributions form the basis of a more robust and efficient deep stereo system that can recognize and adapt to new environments without forgetting.**

I. Introduction

Many tasks in robotics require dense and accurate 3D reconstructions of the scene. Depth estimation using passive stereo cameras is widely used in robotics applications because of its suitability for indoor and outdoor environments and affordable hardware cost [1]. Current state-of-the-art methods for stereo depth estimation use convolutional neural networks (CNNs) to predict depth from left-right image pairs [2]. After Mayer et al. [3] introduced the first end-to-end deep stereo network, deep learning architectures have consistently outperformed traditional algorithms.

Despite their impressive accuracy in domains with abundant labelled training data, deep stereo networks can suffer from a drop in accuracy and prediction quality in novel domains, especially when moving from synthetic training domains to the real-world [4]–[7]. Although fine-tuning using a small set of real-world images has led to strong performance on vision benchmarks such as KITTI [3], [4], [8], [9], this approach requires access to groundtruth depth labels, which are typically sparse and difficult to collect [10].

Several works have proposed *online adaptation* for deep stereo using self-supervised image reconstruction losses [4]–[6] to reduce the performance gap in novel domains. These methods can be classified as continual, "lifelong" learning, where every incoming image pair is used to perform a gradient descent update to the deep stereo network. Although the adaptive stereo systems proposed in [4]–[6] lead to rapid performance improvements in novel domains, three limitations prevent their safe and efficient real-world deployment.

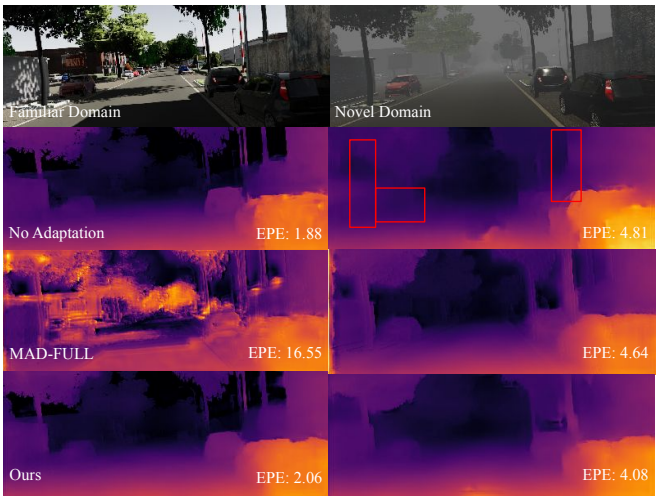


Fig. 1: The accuracy of a deep stereo network trained in a sunny, daytime environment deteriorates in the presence of fog, and the network fails to detect several trees and cars (boxed in red). Existing continuous adaptation methods, exemplified by MAD [4], improve performance in the novel environment but cause catastrophic forgetting in the training environment. Our method uses *experience replay* (ER) to preserve training accuracy during adaptation. We report the end-point-error (EPE) metric, which is the average disparity prediction error for an image (lower is better).

First, the adaptive stereo systems in [4]–[6] do not differentiate between nominal inference in familiar environments and unreliable inference in novel environments; they are always adapting. It is crucial to know when the deep stereo network is untrustworthy so that downstream planning and controls subsystems do not select dangerous actions in response to erroneous perception [11]. Second, these approaches are computationally wasteful, as they perform expensive gradient descent updates even when the model is adapted to the current domain. Even the extremely fast MADNet architecture requires 0.26s to perform inference for a single image pair on an NVIDIA Jetson TX2 [4], limiting the framerate of online adaptation to well below 4Hz. Clearly, online adaptation cannot continue indefinitely if real-time perception is required.

Third, we find that existing adaptive stereo systems, exemplified by [4], cause *catastrophic forgetting* [12], [13] of the training domain. This is a well-known problem for lifelong learning systems [14], where the agent "forgets"

Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology, Cambridge, MA 02139, USA
{milo,valentinp,wng,nickroy}@csail.mit.edu

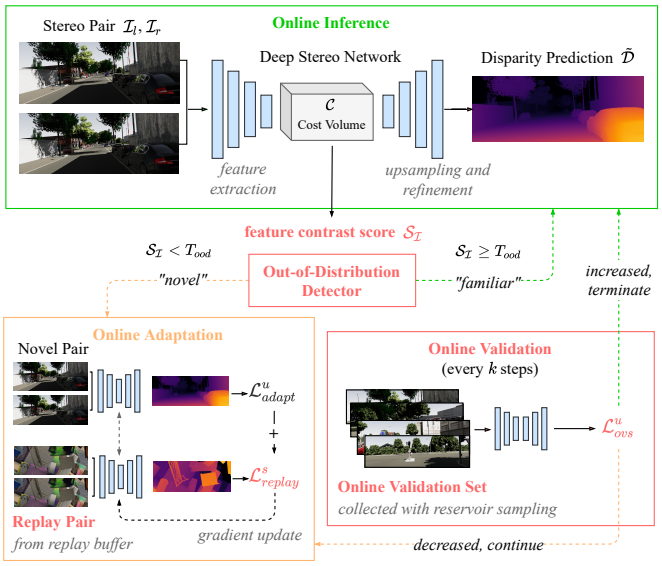


Fig. 2: An overview of our adaptive stereo system, with our contributions highlighted in red. An out-of-distribution (OOD) detector triggers adaptation when it encounters a novel image. During adaptation, we combine a self-supervised adaptation loss with a supervised experience replay loss, improving accuracy in the novel domain without forgetting the training domain. Every k steps, we evaluate the model using an online validation set (OVS) to check if adaptation has improved performance. If not, we terminate adaptation and resume fast inference without backpropagation.

a prior task after learning a new one. To the best of our knowledge, catastrophic forgetting (shown in Fig. 1) has not been addressed in the context of adaptation for deep stereo.

To address these issues, we make three contributions:

- We propose a novel *feature contrast score* for out-of-distribution (OOD) detection, allowing us to trigger adaptation when the deep stereo network has entered a novel environment.
- We show that *online validation* can terminate adaptation when it stops improving performance, allowing our system to avoid lifelong gradient descent updates.
- We demonstrate that experience replay (ER) can be used to mitigate forgetting during online adaptation.

Overall, we present a more safe and reliable adaptive stereo system that can detect unfamiliar environments, improve performance in novel domains without forgetting the training set, and avoid the long-term computational burden of adaptation.

II. Related Works

A. Deep Stereo Networks

Stereo depth estimation, like many other tasks in computer vision and robotics, has been revolutionized by deep learning [2]. After Mayer et al. [3] introduced DispNetC, new architectures have steadily led to improvements in accuracy and inference time on vision benchmarks such as KITTI [15].

For example, the authors of GC-Net [8] showed that a cost volume regularization stage allows the network to learn an appropriate feature matching cost from data, resulting in better performance than hand-crafted metrics such as cosine distance or L_2 distance. The authors of StereoNet [9] demonstrated that the spatial dimensions of the cost volume can be reduced by a factor of 16 or even 32 while achieving sub-pixel precision. In this paper, we use StereoNet for our experimental evaluations due to its low memory footprint and high accuracy on vision benchmarks such as KITTI [15].

B. Online Adaptation for Deep Stereo

Several recent works have demonstrated the effectiveness of self-supervised image reconstruction loss for online adaptation of deep stereo networks [4]–[6]. Tonioni et al. [5] and Zhang et al. [6] show that gradient-based adaptation can be accelerated using the meta-learning framework proposed by Finn et al. [16]. In addition, Tonioni et al. [4] demonstrate real-time adaptation using modular updates, where only a subset of the model's parameters are updated at a time.

An alternative approach is to normalize input data and features so that they are more similar to the training domain. For example, Zhang et al. [6] and Mancini et al. [17] propose regularization techniques based on batch normalization to compensate for shifts in the low-level image feature distributions. Song et al. [7] propose a color transfer algorithm that maps novel images into a color space that is consistent with the training set. Although these methods are not susceptible to forgetting, they are unable to compensate for high-level shifts in scene geometry and semantic content [18].

C. Catastrophic Forgetting

The problem of *catastrophic forgetting* [12], where an agent loses the ability to perform a previous task after learning a new one, is a well-known challenge for sequential learning systems [13], [14], [19], [20], but has been unaddressed for deep stereo adaptation.

One solution proposed Kirkpatrick et al. [19] is to selectively slow-down gradient updates for model parameters that are important for previous tasks. This approach is worth investigating for stereo depth estimation, but it is possible that its damping effect could counteract the adaptation process.

In this work, we build off of two methods that mitigate forgetting by revisiting prior training examples during adaptation: *experience replay* (ER) [21], and *rehearsal* [22]. In ER, which is commonly employed in reinforcement learning, a sliding-window of recent examples is used to train an agent. Similarly, rehearsal methods simultaneously train a learner on new and previously-seen examples, ensuring that short-term learning does not jeopardize performance on prior tasks.

III. Preliminaries

A. Deep Stereo Depth Estimation

In deep stereo depth estimation, a neural networks takes in a pair of left and right RGB images $\mathcal{I}_l, \mathcal{I}_r \in \mathbb{R}^{h \times w \times 3}$ with height h and width w and predicts a *disparity map* $\tilde{\mathcal{D}} \in \mathbb{R}^{h \times w}$. For rectified stereo pairs, the disparity at a pixel in the left image is the horizontal offset of the corresponding 3D scene point projected into the right image. While disparity and depth are interchangeable, disparity is a

convenient representation for algorithms that estimate depth by finding pixel correspondences between stereo images [23].

Typically, training is supervised with groundtruth disparity labels, acquired in simulation or with Lidar. However, self-supervised training (or indeed, adaptation) using only stereo image pairs is possible with an image reconstruction loss. A common choice, which we use in this work, is the self-supervised loss from Monodepth [10], which combines L1 photometric loss, structural similarity [24] and regularization terms for disparity smoothness and left-right consistency.

B. Online Adaptation Framework

In this paper, we consider the online adaptation framework used in [4]–[6], where a deep stereo network is trained offline and then adapted to a sequence of images from a new domain via gradient descent. During online adaptation, images arrive one at a time, and a *single* gradient descent update is applied to the model for each image. Groundtruth disparity labels are unavailable during adaptation, so gradient updates are performed using Monodepth self-supervised loss [10].

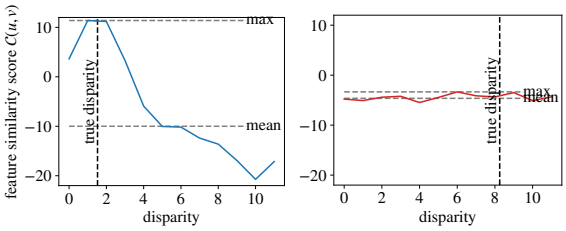
IV. Out-of-Distribution (OOD) Detection

A. Stereo Depth Estimation as Contrastive Feature Learning

Before the advent of end-to-end deep stereo networks, early efforts to integrate learning into stereo depth estimation used neural networks to learn features and distance metrics from data [25]–[29]. Many of these works can be categorized as *contrastive learning*, where a contrastive loss, such as a max-margin (hinge) loss [26] or triplet loss [25], [28], encourages a network to learn a discriminative feature representation from data. The goal is for the learned features to produce low matching costs (or high similarity scores) for corresponding image patches, and high matching costs (low similarity scores) for non-corresponding patches, enabling effective stereo matching. We will show that, although end-to-end stereo networks are not explicitly trained with a contrastive loss, they tend to learn features with similar contrastive properties.

In deep stereo architectures, a Siamese feature extraction network takes in left and right images $\mathcal{I}_l, \mathcal{I}_r$ and outputs feature maps, $F_l, F_r \in \mathbb{R}^{h_F \times w_F \times d_F}$, where h_F, w_F and d_F are the height, width, and dimensionality of the map. Each d_F-dimensional feature describes a square $p \times p$ patch in the input image, whose spatial extent p is determined by the receptive field of the neurons in the feature extractor. After *rectifying* stereo pairs as a preprocessing step, matching costs are evaluated between features along horizontal epipolar lines, and stored in a *cost volume* $\mathcal{C}$ [23]. Matching costs may be hand-crafted metrics, such as an L_2 norm, or may be a function that is learned from data [8], [9].

For notational simplicity, we model both the feature extraction and matching cost as a single function $f : \mathbb{R}^{p \times p \times 3} \times \mathbb{R}^{p \times p \times 3} \rightarrow \mathbb{R}$ that maps a pair of image patches to a scalar cost. Using f, the cost volume $\mathcal{C}$ is formed by evaluating f for a reference patch centered at each pixel in the left image, and candidate patches along a horizontal epipolar line in the right image [23]. For example, $\mathcal{C}(u, v, d)$ stores the matching cost between the reference feature $F_l(u, v)$ and the candidate feature $F_r(u, v - d)$ offset by disparity d.



(a) Well-adapted features **(b)** Poorly-adapted features

Fig. 3: In each plot, we have selected a sample image and pixel location, and plotted a "slice" of the cost volume at that location (i.e, $\mathcal{C}(u, v)$). When the model's features are well-adapted to the environment, as is the case for training images, the similarity scores in the cost volume tend to be peaked near the true disparity. In novel images, features are poorly-adapted, resulting in a less prominent peak or a multimodal distribution of similarity scores.

From the cost volume, a *soft-argmin* is applied to extract a disparity map [8]. It is important to note that, although the entries of $\mathcal{C}$ are canonically interpreted as "cost", some stereo architectures store a "similarity score" instead. In this work, we use the StereoNet architecture [9], which uses similarity scores (and therefore extracts disparity using a *soft-argmax*).

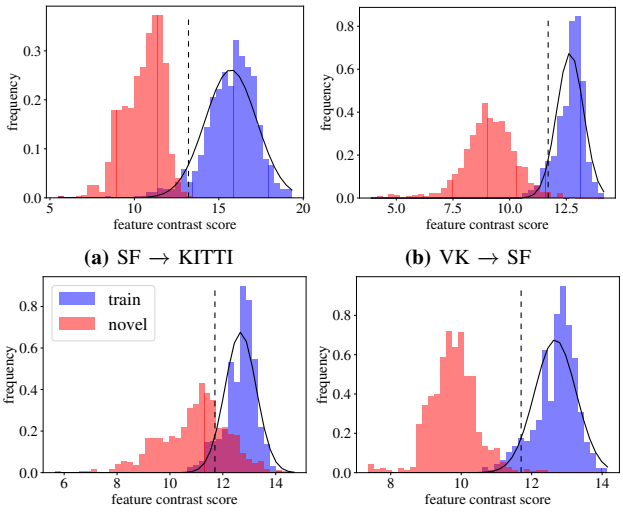
Due to occlusion, non-constant brightness, and other factors, corresponding patches in the left and right images may not have identical appearance. However, f should produce high similarity scores between corresponding patches, despite these perturbations. It is critical to note that the disparity estimate obtained from the *soft-argmax* operation is influenced by the similarity scores between a reference patch and all candidate patches along the epipolar line. Thus, disparity estimation induces an implicit contrastive feature learning objective, where the network (f) attempts to maximize the similarity score for corresponding patches, and minimize the score for all non-corresponding patches.

B. Quantifying Novelty using the Feature Contrast Score

During training, the network learns an f that maximizes the margin or "contrast" between the similarity scores of corresponding patches and the scores of non-corresponding patches. However, f is learned with respect to the distribution of training images, and may be sub-optimal for images outside of this distribution. As a result, we hypothesize that the similarity score contrast will be reduced in novel environments.

To build intuition, we plot a "slice" of the cost volume at a sample pixel in a training and novel image in Fig. 3. By construction, each slice stores the similarity scores between a reference patch and candidate patches along the epipolar line. When the network's features are well-adapted to their environment (Fig. 3a), similarity scores tend to be "peaked" at the true disparity. At a particularly bad location in the novel image (Fig. 3b), the peak is not prominent, leading to ambiguity about the true disparity value.

We quantify the *feature contrast score* $\mathcal{S}_\mathcal{I}$ at a pixel



(a) SF → KITTI **(b)** VK → SF

(c) VK → KITTI **(d)** Sun → Fog + Rain (VK)

Fig. 4: The distribution of feature contrast scores ($\mathcal{S}_\mathcal{I}$) for training and novel images. The shorthand "X → Y" indicates that the model was trained in domain X and then tested in a novel domain Y. **SF** is the simulated SceneFlow Flying dataset [3], **KITTI** is the real-world KITTI Raw dataset [15], and **VK** is the simulated Virtual KITTI dataset [34]. The network's learned features are less discriminative in the novel domain, leading to a lower $\mathcal{S}_\mathcal{I}$. Interestingly, $\mathcal{S}_\mathcal{I}$ remains high for many KITTI images after training on VK (4c), likely due to the fact that VK is designed to match the appearance of its real-world counterpart. In each plot, the vertical dashed line indicates a 5^{th} percentile OOD threshold (T_{ood}).

location (u, v) as follows:

$$\mathcal{S}_\mathcal{I}(u,v) \triangleq \mathcal{C}_{max}(u,v) - \mathcal{C}_{mean}(u,v) \tag{1}$$

where $\mathcal{C}_{max}$ and $\mathcal{C}_{mean}$ are the same quantities depicted in Fig. 3. Because we do not have access to the groundtruth disparity value at each pixel, Eq. 1 implicitly assumes that the maximum similarity score occurs at the true disparity. Although this is not true for pixels with incorrect disparity predictions (e.g 3b), we find that the difference between the maximum and mean still captures the contrast between corresponding and non-corresponding features.

For convenience, let $\mathcal{S}_\mathcal{I}$ be the average feature contrast score across all pixels in image $\mathcal{I}$. $\mathcal{S}_\mathcal{I}$ quantifies how discriminative the deep stereo network's learned features are for a particular image. As we hypothesized, the reduced effectiveness of the learned features in novel environments leads to a lower $\mathcal{S}_\mathcal{I}$, which we illustrate in Fig. 4.

An advantage of $\mathcal{S}_\mathcal{I}$ over existing autoencoder-based methods [30]–[33] is that it does not require an auxiliary network for OOD detection. $\mathcal{S}_\mathcal{I}$ reuses network computation from the cost volume, and adds minimal overhead relative to a single forward pass through the deep stereo network. For our StereoNet configuration (see Sec. VII-A), computing $\mathcal{S}_\mathcal{I}$ increases inference time by 1-2%.

C. Triggering Adaptation with Out-of-Distribution Detection

The discrepancy between training and novel $\mathcal{S}_\mathcal{I}$ distributions in Fig. 4 enables a simple yet effective threshold-based out-of-distribution (OOD) detector. Following the approach of Peretroukhin et al. [35], we compute the empirical distribution of $\mathcal{S}_\mathcal{I}$ for the training set (offline), and set a threshold T_{ood} to a low percentile of the empirical CDF. We classify an image as novel if $\mathcal{S}_\mathcal{I} < T_{ood}$. In our experiments, we chose T_{ood} to be the 5^{th} percentile, although this parameter can be tuned to balance precision and recall (see Fig. 5). [1]

V. Online Validation for Efficient Adaptation

A. Measuring Adaptation Progress

In prior works [4]–[6] we observe that most of the performance gains from adaptation are achieved within a few-hundred images, and diminish over time. Once the model has adapted to a novel domain, further adaptation may cause overfitting and cease to improve accuracy. However, existing adaptation methods such as MADNet [4] will perform gradient descent updates indefinitely, consuming substantial computational resources while bringing marginal benefits.

To avoid overfitting, and to free up resources on mobile computing platforms, we would like to stop adaptation as soon as performance plateaus. Noting that it is standard practice to use a validation loss to detect overfitting during training [36], we extend this paradigm to do *online validation* using a self-supervised loss. We maintain an *online validation set* (OVS) using images collected from the novel environment, and periodically re-compute a validation loss to check if adaptation has improved performance. If not, we terminate adaptation and return to fast inference with no gradient descent updates.

B. Sampling a Uniform Online Validation Set (OVS)

We would like our online validation set (OVS) to be a representative, uniform sample of images from the novel domain. To accomplish this, we use *reservoir sampling* [37], which is an online algorithm designed to select a uniform sample of items from a stream of unknown size. For each incoming image pair that is classified as novel, the reservoir sampling algorithm probabilistically chooses to either discard the pair or replace an item in the OVS. The probability of replacement is set such that every image pair has an equal probability of ending up in the OVS.

During online adaptation, our OOD detector (see Sec. IV-C) produces a stream of image pairs that are classified as novel. The reservoir sampler will occasionally replace an item in the OVS with a pair from the stream. To ensure that the OVS images are "held-out" during adaptation, any image pair that is added to the OVS is not used to adapt the model.

Eventually, as the deep stereo network adapts, the feature contrast score ($\mathcal{S}_\mathcal{I}$) for incoming images will increase above the threshold T_{ood}, and images will no longer be classified as novel. At this point, the OVS contains a small, uniform sample of all novel images seen so far, which we can use to validate the model's performance in the novel domain.

[1] To prevent the training images that fall below the OOD threshold T_{ood} in Fig. 4 from causing spurious adaptation, we trigger online adaptation after detecting $W = 5$ consecutive novel images. The parameter W can be increased to reduce the probability of a false start at the expense of a slight delay in adaptation.

C. Online Validation Loss

Every k steps, we compute an online validation loss $\mathcal{L}^u_{ovs}$, which is the average self-supervised loss across all of the images in the OVS. In our implementation, $\mathcal{L}^u_{ovs}$ is the same Monodepth self-supervised loss [10] that we use for adaptation. $\mathcal{L}^u_{ovs}$ allows us to track the effect of adaptation on novel domain performance.

Adaptation is terminated when (1) the OVS has not changed since the last online validation and (2) $\mathcal{L}^u_{ovs}$ has increased. These two criteria indicate that adaptation is no longer improving the model's performance on a consistent set of images. Once adaptation terminates, it can be triggered again if more "OOD" images are detected at a future time. This may indicate that the novel domain has not been sufficiently explored, and further adaptation is needed.

VI. Mitigating Forgetting via Experience Replay

To mitigate forgetting, we build off of a method from reinforcement learning (RL) called *experience replay* (ER) [21], where a learner is trained on both new and previous examples to ensure that it does not forget prior tasks. However, our ER method differs from that of most RL applications because we store a fixed sample from the training set, rather than a sliding window of recent examples.

To construct a replay buffer, we uniformly sample N_{replay} image pairs from the training set [2]. Groundtruth disparity labels are available for these images because they are from a synthetic training set. To incorporate ER into online adaptation, our model predicts disparity for one novel image pair and one sampled pair from the replay buffer. We then adapt the model using a combined loss:

$$\mathcal{L} = \mathcal{L}^u_{adapt}(\mathcal{I}_l, \mathcal{I}_r, \tilde{\mathcal{D}}) + \alpha \mathcal{L}^s_{replay}(\tilde{\mathcal{D}}_{replay}, \mathcal{D}_{replay}) \quad (2)$$

where $\tilde{\mathcal{D}}$ is the disparity prediction for the novel image pair $(\mathcal{I}_l, \mathcal{I}_r)$, and $\tilde{\mathcal{D}}_{replay}$ and $\mathcal{D}_{replay}$ are the predicted and groundtruth disparity for the experience replay image pair. $\mathcal{L}^u_{adapt}$ is the unsupervised loss from Monodepth [10], and $\mathcal{L}^s_{replay}$ is the supervised loss used to train StereoNet [9]. We found that $\alpha = 0.05$ balanced the adaptation and replay objectives well in our experiments.

We note that this approach is similar to the *rehearsal* methods proposed by Robins et al. [22], where a neural network is re-trained on sampled prior examples while it is introduced to new ones.

VII. Evaluation

A. Implementation Details

In our experiments, we use the lightweight "16X, single" StereoNet configuration [9], and perform training and adaptation using 320×960 image crops. We train the model using the robust loss function from the original StereoNet paper, and the Adam optimizer [38] with a learning rate of 10^{-4}, decaying by a factor of $\frac{1}{2}$ every 20k steps.

[2]For all experiments in this work, we set $N_{replay} = 1000$.

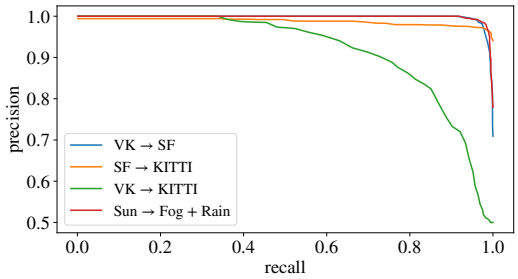


Fig. 5: Precision-recall curves for our OOD detector, which are generated by varying the classification threshold T_{ood} from Sec. IV-C. We use the same training and novel image sets from Fig. 4. The detector is effective at discriminating between Virtual KITTI (VK) and SceneFlow Flying (SF), SF and KITTI Raw (KITTI), as well as between the sunny "Clone" and the "Fog" and "Rain" sequences from VK. As in Fig. 4, the visual similarity of VK and KITTI causes difficulty distinguishing between the two.

B. Online Adaptation Methods

We consider four online adaptation methods, each of which adds one or more components of our system to a baseline adaptation procedure:

- **MAD**: The "FULL" method from MADNet [4] as a baseline for continuous online adaptation.
- **VS**: We terminate when online validation loss increases (see Sec. V). In our implementation, the OVS contains 10 images, and $\mathcal{L}^u_{ovs}$ is recomputed every $k = 100$ steps.
- **ER**: We perform gradient descent updates with respect to the combined loss function in Eq. 2, which includes an adaptation loss and a replay loss.
- **VS + ER**: Our full method depicted in Fig. 2.

All methods begin with identical network parameters, and adapt using Adam with a fixed learning rate of 5×10^{-5}.

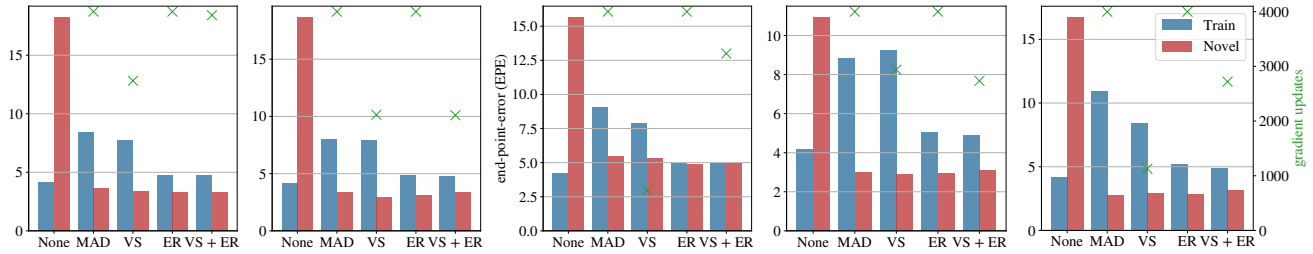
C. Training and Novel Environments

In our experiments, we pre-train StereoNet using either the SceneFlow Flying [3] or Virtual KITTI "Clone" [34] datasets, and adapt it to a novel image sequence. We include two novel sequences from Virtual KITTI (Scene 01 and Scene 20), as well as the City, Road, and Campus sequences from KITTI Raw [15]. In addition, we analyze the effect of novel weather conditions using the Virtual KITTI "Fog" and "Rain" sequences. These images are identical to their "Clone" counterparts (nominal, sunny weather), but they have simulated fog or rain effects applied.

D. Out-of-Distribution (OOD) Detection

We plot the precision-recall performance of our OOD detector in Fig. 5. The detector achieves high precision at near-perfect recall in three combinations of training and novel environments, detecting visual differences between a simulated and real dataset, two different simulated datasets, and even the same dataset with fog and rain applied.

The detector's performance is lowest when our model is trained on Virtual KITTI and tested on novel KITTI Raw images. We attribute this to the high degree of visual similarity between these two datasets. The Virtual KITTI dataset was designed to mimic the appearance of KITTI Raw,



(a) VK 01 **(b)** VK 20 **(c)** KITTI Campus **(d)** KITTI City **(e)** KITTI Road

Fig. 6: End-point-error (EPE) in the original training domain and novel adaptation domain after 4000 adaptation steps. **None** is the pre-adaptation EPE (equal for all methods). Green markers show the number of gradient updates performed by each method. All four methods reduce novel domain EPE, although **MAD** and **VS** lead to substantially higher EPE in the original training domain after adaptation. Both **ER** and **VS+ER** maintain roughly constant training domain EPE during adaptation, indicating the effectiveness of experience replay. While **VS** and **VS+ER** terminate early, **MAD** and **ER** will continue to perform gradient updates indefinitely. Our full method, **VS+ER**, performs efficient adaptation without forgetting.

leading to less separable feature contrast score distributions in Fig. 4c and thus lower precision. Still, adaptation is likely to be triggered because the majority of images from KITTI are below the classifier threshold.

E. Online Validation

In Fig. 6, we see that the methods with online validation (**VS** and **VS + ER**) terminate before the end of the adaptation experiments, switching back to fast online inference without backpropagation. Despite performing fewer gradient descent updates than **MAD**, the methods with online validation (**VS** and **VS + ER**) achieve a comparable reduction in error in the novel domain. This indicates that the additional gradient updates used by **MAD** do not improve accuracy, and are therefore wasted computation. Online validation provides a principled way to free-up computational resources when they are no longer needed for adaptation.

F. Experience Replay

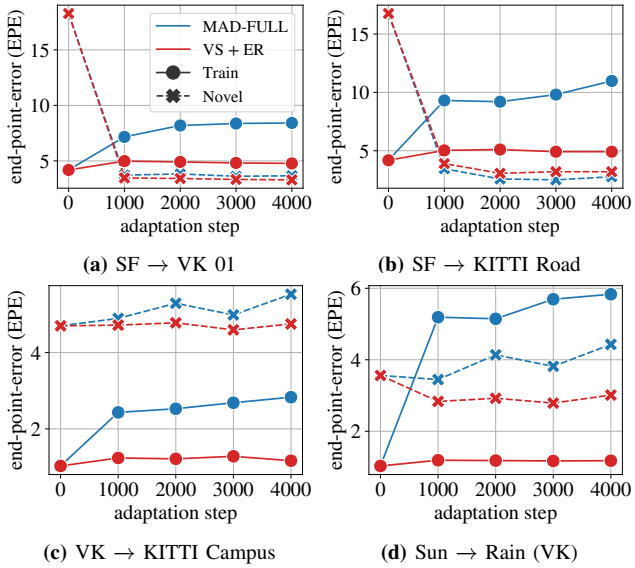
We illustrate the effect of catastrophic forgetting in Fig. 7. For **MAD**, adaptation to the novel domain causes EPE to increase dramatically in the original training domain. This poses a safety concern, because the deep stereo network may no longer be reliable in the training domain.

Our full method, **VS + ER**, improves the novel domain EPE by a comparable margin to **MAD**, while maintaining the training domain EPE at a relatively constant value. This demonstrates that it is possible to capture the benefits of adaptation without sacrificing performance in prior domains. Interestingly, in Fig. 7c and 7d, **VS + ER** achieves better adaptation than **MAD** in the novel domain. We hypothesize that including a supervised replay loss during adaptation adds robustness in domains where photometric loss is misleading, such as the VK "Rain" environment.

VIII. Conclusions and Future Work

In this work, we proposed three improvements for more safe and reliable online adaptation. We demonstrated that the *feature contrast score* can be used for out-of-distribution (OOD) detection, allowing us to trigger adaptation when the deep stereo network is unreliable. Next, we demonstrated that *online validation* provides a principled stopping criterion for adaptation, alleviating the long-term computational burden of lifelong learning. Finally, we showed that incorporating *experience replay* into adaptation can mitigate forgetting and even lead to more effective adaptation in novel environments.

A key limitation of our method is the scalability of the experience replay buffer. After adapting to a novel domain, images from that environment can then be added the replay buffer to mitigate forgetting in the future. However, maintaining a balanced replay buffer that reflects all of the domains encountered thus far is a practical challenge that we do not address in this work. In practice, while our system provides online robustness to unknown environments, we note that it can also be used to collect novel data and edge-cases that are then carefully incorporated into the offline training pipeline [39]. We leave this investigation to future work.



(a) SF → VK 01 **(b)** SF → KITTI Road

(c) VK → KITTI Campus **(d)** Sun → Rain (VK)

Fig. 7: Training and novel domain EPE during adaptation. Every 1000 steps, we show the average EPE of the model across all images in the training and novel sequences. For Fig. 7a and Fig. 7b, both the **MAD** and **VS+ER** methods lead to a rapid decrease in novel domain EPE. However, **MAD** causes training domain EPE to increase substantially. In Fig. 7c and Fig. 7d, **VS+ER** prevents forgetting and maintains a steady training EPE. We note that **MAD** causes novel domain EPE to *worsen* over time in Fig. 7d, likely due to misleading photometric loss signals from rain drops visually occluding and obscuring objects. In this case, the supervised replay loss included in the **VS+ER** method adds robustness to these effects.

REFERENCES

[1] N. Ayache, *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception*. MIT Press, 1991.

[2] L. Hamid, V. J. Laurent, B. Farid, and B. Mohammed, *A survey on deep learning techniques for stereo-based depth estimation*, 2020. arXiv: 2006.02535 [cs.CV].

[3] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016.

[4] A. Tonioni, F. Tosi, M. Poggi, S. Mattoccia, and L. D. Stefano, "Real-time self-adaptive deep stereo," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019.

[5] A. Tonioni, O. Rahnama, T. Joy, L. D. Stefano, T. Ajanthan, and P. H. Torr, "Learning to adapt for stereo," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019.

[6] Z. Zhang, S. Lathuilière, A. Pilzer, N. Sebe, E. Ricci, and J. Yang, "Online adaptation through meta-learning for stereo depth estimation," *CoRR*, vol. abs/1904.08462, 2019. arXiv: 1904.08462. [Online]. Available: http://arxiv.org/abs/1904.08462.

[7] X. Song, G. Yang, X. Zhu, H. Zhou, Z. Wang, and J. Shi, "AdaStereo: A Simple and Efficient Approach for Adaptive Stereo Matching," pp. 1–18, 2020. arXiv: 2004.04627. [Online]. Available: http://arxiv.org/abs/2004.04627.

[8] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry, "End-to-end learning of geometry and context for deep stereo regression," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2017.

[9] S. Khamis, S. Fanello, C. Rhemann, A. Kowdle, J. Valentin, and S. Izadi, "Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction," 2018, pp. 8–14.

[10] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *CVPR*, 2017.

[11] N. Sünderhauf, O. Brock, W. Scheirer, R. Hadsell, D. Fox, J. Leitner, B. Upcroft, P. Abbeel, W. Burgard, M. Milford, and P. Corke, "The limits and potentials of deep learning for robotics," *International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 405–420, 2018.

[12] M. McCloskey and N. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," *Psychology of Learning and Motivation - Advances in Research and Theory*, vol. 24, no. C, pp. 109–165, Jan. 1989.

[13] I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio, "An empirical investigation of catastrophic forgetting in gradient-based neural networks," *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, 2014.

[14] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

[15] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[16] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," D. Precup and Y. W. Teh, Eds., ser. Proceedings of Machine Learning Research, vol. 70, PMLR, Aug. 2017, pp. 1126–1135.

[17] M. Mancini, H. Karaoguz, E. Ricci, P. Jensfelt, and B. Caputo, "Kitting in the Wild through Online Domain Adaptation," *IEEE International Conference on Intelligent Robots and Systems*, pp. 1103–1109, 2018.

[18] A. Kar, A. Prakash, M.-Y. Liu, E. Cameracci, J. Yuan, M. Rusiniak, D. Acuna, A. Torralba, and S. Fidler, "Meta-sim: Learning to generate synthetic datasets," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019.

[19] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017, ISSN: 0027-8424. DOI: 10.1073/pnas.1611835114.

[20] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning.," in *International Conference on Learning Representations (ICLR)*, 2016. [Online]. Available: http://arxiv.org/abs/1509.02971.

[21] L.-J. Lin, "Self-improving reactive agents based on reinforcement learning, planning and teaching," *Mach. Learn.*, vol. 8, no. 3–4, 293–321, May 1992, ISSN: 0885-6125. DOI: 10.1007/BF00992699.

[22] A. Robins, "Catastrophic forgetting, rehearsal, and pseudorehearsal," *Connection Science: Journal of Neural Computing, Artificial Intelligence and Cognitive Research*, vol. 7, pp. 123–146, 1995.

[23] D. Scharstein, R. Szeliski, and R. Zabih, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Proceedings - IEEE Workshop on Stereo and Multi-Baseline Vision, SMBV 2001*, no. 1, pp. 131–140, 2001. DOI: 10.1109/SMBV.2001.988771.

[24] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004, ISSN: 10577149. DOI: 10.1109/TIP.2003.819861.

[25] J. Zbontar and Y. LeCun, "Computing the stereo matching cost with a convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015.

[26] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015.

[27] W. Luo, A. G. Schwing, and R. Urtasun, "Efficient deep learning for stereo matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016.

[28] V. K. B. G, G. Carneiro, and I. Reid, "Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016.

[29] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg, "Matchnet: Unifying feature and metric learning for patch-based matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015.

[30] C. Richter and N. Roy, "Safe visual navigation via deep learning and novelty detection," *Robotics: Science and Systems*, vol. 13, 2017.

[31] A. Amini, W. Schwarting, G. Rosman, B. Araki, S. Karaman, and D. Rus, "Variational autoencoder for end-to-end control of autonomous driving with novelty detection and training de-biasing," 2018.

[32] T. Amarbayasgalan, B. Jargalsaikhan, and K. Ryu, "Unsupervised novelty detection using deep autoencoders with density based clustering," *Applied Sciences*, vol. 8, p. 1468, 2018.

[33] T. Denouden, R. Salay, K. Czarnecki, V. Abdelzad, B. Phan, and S. Vernekar, "Improving reconstruction autoencoder out-of-distribution detection with mahalanobis distance," *CoRR*, vol. abs/1812.02765, 2018. arXiv: 1812.02765. [Online]. Available: http://arxiv.org/abs/1812.02765.

[34] Y. Cabon, N. Murray, and M. Humenberger, *Virtual kitti 2*, 2020. arXiv: 2001.10773 [cs.CV].

[35] V. Peretroukhin, M. Giamou, D. M. Rosen, N. W. Greene, N. Roy, and J. Kelley, "A smooth representation of belief over SO(3) for deep rotation learning with uncertainty," *Robotics: Science and Systems (RSS)*, 2020.

[36] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[37] J. S. Vitter, "Random sampling with a reservoir," *ACM Trans. Math. Softw.*, vol. 11, no. 1, 37–57, Mar. 1985, ISSN: 0098-3500. DOI: 10.1145/3147.3165. [Online]. Available: https://doi.org/10.1145/3147.3165.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[39] A. Karpathy, "Workshop on Scalability in Autonomous Driving," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.