A Strong Transfer Baseline for RGB-D Fusion in Vision Transformers

Georgios Tziafas and Hamidreza Kasaei

Abstract—The Vision Transformer (ViT) architecture has recently established its place in the computer vision literature, with multiple architectures for recognition of image data or other visual modalities. However, training ViTs for RGB-D object recognition remains an understudied topic, viewed in recent literature only through the lens of multi-task pretraining in multiple modalities. Such approaches are often computationally intensive and have not yet been applied for challenging object-level classification tasks. In this work, we propose a simple yet strong recipe for transferring pretrained ViTs in RGB-D domains for single-view 3D object recognition, focusing on fusing RGB and depth representations encoded jointly by the ViT. Compared to previous works in multimodal Transformers, the key challenge here is to use the atested flexibility of ViTs to capture cross-modal interactions at the downstream and not the pretraining stage. We explore which depth representation is better in terms of resulting accuracy and compare two methods for injecting RGB-D fusion within the ViT architecture (i.e., early vs. late fusion). Our results in the Washington RGB-D Objects dataset demonstrates that in such RGB → RGB-D scenarios, late fusion techniques work better than most popularly employed early fusion. With our transfer baseline, adapted ViTs score up to 95.1% top-1 accuracy in Washington, achieving new state-of-the-art results in this benchmark. We additionally evaluate our approach with an open-ended lifelong learning protocol, where we show that our adapted RGB-D encoder leads to features that outperform unimodal encoders, even without explicit fine-tuning. We further integrate our method with a robot framework and demonstrate how it can serve as a perception utility in an interactive robot learning scenario, both in simulation and with a real robot.

I. INTRODUCTION

Transfer learning approaches for computer vision have a long standing tradition for image classification, most popularly using Convolutional Neural Networks (CNNs). More recently, the Vision Transformer (ViT) architecture and its many variants [32, 44, 3, 22] have also shown promising transfer results, providing flexible representations that can be fine-tuned for downstream tasks, more recently also in few-shot settings [9]. This flexibility is due to the famous capability of the Transformer architecture to capture long-range dependencies in the input sequence, a trait that is missing from CNNs that are designed with inductive biases for high sensitivity in locality, through their pooling operations. This capability however comes at the cost of data inefficiency [23], as performance gains over CNNs are noticed in Transformers that are pretrained in large-scale datasets, such as ImageNet21k [40] and JFT-300M [42]. When moving from RGB-only to view-based 3D object recognition (RGB-D), a dataset of similar magnitude for pretraining is amiss, granting RGB-D representation learning a topic that has yet to grow. Recent alternative directions include transferring from models pretrained on collections of multimodal datasets [18, 31, 16, 17], although they focus on scene-level tasks, they are constrained to the use of the early fusion strategy and are often computationally intensive to fine-tune.

In this work we wish to explore such questions by revisiting the RGB-D object recognition task and study recipes for transferring an RGB-only pretrained vanilla ViT (i.e. in ImageNet1k [11]) into an RGB-D object-level dataset. We begin by exploring different representation formats for the input depth modality and design two variants that adapt ViT to fuse RGB and depth (see Fig. 1), namely: a) Early fusion, where RGB and depth are fused before the encoder and RGB-D patches are represented jointly in the sequence, and b) Late fusion, where we move the fusion operation after the encoder, leaving the patch embedders intact from their pretraining. Our hypothesis is that when fine-tuning in small (or moderate) sized datasets, the late fusion baseline is very likely to perform better, as it doesn’t change the representation of the input compared to the pretraining stage, but casts the challenge as a distribution shift in the input images (i.e. both RGB and depth are processed by the same weights and must me mapped to the same label).

Experimental results with the Washington RGB-D Objects dataset [29] positively reinforce our hypothesis, as the late fusion baseline far outperforms the early variant. More interestingly, we show that with our late fusion recipe, ViTs achieve new state-of-the-art results in this benchmark, surpassing a plethora of methods that specifically study RGB-D fusion techniques for object recognition. In our experiments we further demonstrate the representational strength of our approach by evaluating using an open-ended lifelong learning scenario, where we show that our late fusion encoder outperforms unimodal versions of same scale, even without fine-tuning. Finally, we demonstrate the applicability of our approach in the robotics domain by integrating our method with a simulated and a real robot framework and show how the robot can be taught by a human user to recognize new objects, in order to perform a table cleaning task. In summary, our contributions are threefold, namely:

- We experimentally find that late works better than early fusion in RGB → RGB-D transfer scenario
- We achieve new state-of-the-art results for RGB-D object recognition in the Washington RGB-D Objects benchmark
- We show that our method can be applied in an online lifelong learning setup, including simulation and real robot demonstrations.
Fig. 1: Two different baselines for fusing RGB-D representations in the ViT architecture. In early fusion (left), a separate projection is used for RGB and depth and the fused embeddings are fed to the encoder, providing a single \(<\text{CLS}>\) token. In late fusion (right), the same weights are used for projecting RGB and depth and the two modalities are fed separately to the encoder. The two final \(<\text{CLS}>\) tokens are fused to provide the final representation for classification.

II. RELATED WORKS

In this section we discuss previous works on RGB-D fusion with CNNs for view-based object recognition, multimodal Transformers and briefly discuss on the topic of lifelong learning, which we include as an evaluation scenario in our experiments.

A. RGB-D Fusion with CNNs

As in RGB image classification, multiple traditional CNN-based approaches have replaced conventional approaches \([4, 43]\) for extending to the RGB-D modalities. The focus of such works lies in RGB-D fusion, where deep features extracted from CNNs are fused through a multimodal fusion layer \([46]\) or custom networks \([45]\). Rahman et. al. \([14]\) propose a parallel three-stream CNN which processes two depth encodings in two streams and RGB in the last one. Cheng et. al. \([10]\) proposed to integrate Gaussian mixture models with CNNs through fisher kernel encodings. Zia et. al. \([53]\) propose mixed 2D/3D CNNs which are initialized with 2D pretrained weights and extend to 3D to also incorporate depth. Such methods study how to inject fusion in the locally-aware CNN architecture. Instead, in our work we implement fusion as a simple operation on embeddings from the different modalities and opt to gain cross-modal alignment by virtue of the long-range context modeling capabilities of the Transformer architecture.

B. Multimodal Learning with Transformers

In the absence of a large-scale RGB-D dataset for pretraining, recent works try to alleviate this bottleneck by pretraining on collections of datasets from multiple modalities \([18, 31, 16, 17]\) and rely on the flexibility of Transformers to capture cross-modal interactions. However, such methods focus on scene/action recognition or semantic segmentation tasks, leaving the RGB-D object recognition task unexplored. Furthermore, they employ an early fusion technique for converting heterogeneous modalities in the same sequence representation, leaving open questions of whether this is the best fusion technique in homogeneous modalities such as RGB-D, as well as if its the best fusion technique for directly transferring from one homogeneous modality to another, without the pretraining step. Finally, they rely heavily on model capacity and specialized Transformer architecture variants (e.g Swin \([32]\)) in order to enable multimodal pretraining to boost performance in unimodal downstream tasks. Such models set a high computational resource entry point for practitioners, casting them not widely accessible for fine-tuning in arbitrary datasets.

C. Lifelong Learning

An emerging topic in deep learning literature, most commonly referred to as Lifelong or Continual Learning, studies the scenario of a learning agent continuously incorporating new experiences from an online data stream. In the context of image classification, the challenge is stated as learning to classify images from an ever-growing set of new data and tasks, while avoiding the effect of catastrophic forgetting \([35, 8, 39, 50, 51, 41]\). Even though works for using Transformers in lifelong learning are starting to grow \([13, 47, 15]\), to the best of our knowledge, this is the first work that touches on lifelong learning with Transformers for RGB-D object recognition. We highlight however that the focus of this work is not on lifelong learning algorithms, but rather to establish a baseline in the Washington benchmark for future references.

III. APPROACH

Our goal is to have a single model that can be transferred to RGB-D downstream tasks, while being pretrained solely in RGB. Even though the two modalities are homogeneous, different depth representations might insert discrepancies in the size of the input depth image. To deal with this, we adopt the Transformer architecture, because the self-attention operation has shown a tremendous ability to model long-range dependencies of variable size inputs. Unlike standard fine-tuning strategies, we wish to enable ViT to learn from the depth modality, as well as learn how to successfully model
the correspondences between the two modalities. To that end, we explore two different RGB-D representation fusion techniques.

A. ViT Prerequisites

The ViT model handles the visual input as a sequence of image patches. The original $H \times W$ image $\mathcal{F}$ is split into patches of size $h \times w$, resulting in a total of $N = \frac{H \cdot W}{h \cdot w}$ patches. Each patch is flattened into a single vector representation $x_n \in \mathbb{R}^{3h\times w}$ and projected into an embedding space through a linear map $\mathcal{E}(x_n) = e_n$, $\mathcal{E} : \mathbb{R}^{3h\times w} \rightarrow \mathbb{R}^D$. A trainable image-level embedding $e_0$ (i.e. the $<CLS>$ token representation) is stacked with the embeddings sequence and the patch embeddings are further added with positional encodings $p_n$, either learned jointly or hand-crafted (e.g. 2D sinusoid).

The resulting sequence $[e_0, \{e_n + p_n\}_{n=1}^N]$ is passed through $L$ layers of Transformer encoder blocks, resulting in the sequence of hidden states $[h_n]_{n=0}^N$, $l = 1, \ldots, L$. For downstream classification tasks, the final hidden $<CLS>$ state $h_{cls} = h_0$ is fed into a linear layer over the number of target classes combined with a softmax loss.

B. Depth Representation

In order to make ViT compatible with the RGB-D modality, we need to express the input depth map with the same format. The original $H \times W$ depth map $d$ is split into patches of size $h \times w$, resulting in a total of $N = \frac{H \cdot W}{h \cdot w}$ patches. Each patch is flattened into a single vector representation $x_n \in \mathbb{R}^{3h\times w}$ and projected into an embedding space through a linear map $\mathcal{E}(x_n) = e_n$, $\mathcal{E} : \mathbb{R}^{3h\times w} \rightarrow \mathbb{R}^D$. A trainable image-level embedding $e_0$ is stacked with the embeddings sequence and the patch embeddings are further added with positional encodings $p_n$, either learned jointly or hand-crafted (e.g. 2D sinusoid).

The resulting sequence $[e_0, \{e_n + p_n\}_{n=1}^N]$ is passed through $L$ layers of Transformer encoder blocks, resulting in the sequence of hidden states $[h_n]_{n=0}^N$, $l = 1, \ldots, L$. For downstream classification tasks, the final hidden $<CLS>$ state $h_{cls} = h_0$ is fed into a linear layer over the number of target classes combined with a softmax loss.

a) Early Fusion: In early fusion, each modality, $\mathcal{E}_r^{rgb}$ and $\mathcal{E}_d$ and fuse the two representations before adding the position embeddings, using addition and L2 normalization:

$$e_n = \mathcal{E}_r^{rgb}(x_n^{rgb}) + \mathcal{E}_d(x_n^d) \left\| \mathcal{E}_r^{rgb}(x_n^{rgb}) + \mathcal{E}_d(x_n^d) \right\|_2$$

We call this baseline the dual-embedder, as it separates embeddings for the two modalities. In our experiments we also implement the early fusion baseline with a joint-embedder, stacking the two modalities channel-wise and using a single projection to embed them jointly. In this architecture, a single $<CLS>$ embedding is learned for the entire RGB-D pair. The key insight is that through the self-attention operation the joint RGB-D embedding $h_{cls}$ will adapt to model the inter-modal alignment between the fused representations. To make the pretrained ViT checkpoint compatible with the adapted architecture, we copy the weights of the pretrained patch embedder in both RGB and depth embedders.

b) Late Fusion: In late fusion, we pass the two images from the ViT encoder separately and aggregate their final $<CLS>$ embeddings, before passing it to the classifier. We experiment with different types of late fusion operations $f : \mathbb{R}^{D} \times \mathbb{R}^{D'} \rightarrow \mathbb{R}^{D''}$, such as max pooling ($D'' = D$), averaging ($D'' = D$), addition ($D'' = 2 \cdot D$) and concatenation ($D'' = 2 \cdot D$). In this baseline, the encoder has to learn how to classify both images of the same object view, while processing the two modalities separately. The final hidden representation used for classification is the fusion of the two hidden states for the two modalities: $h_{cls} = f(h_{cls}^{rgb}, h_{cls}^{d})$.

D. Implementation Details

In order to fine-tune the adapted ViT, we use a datasetspecific linear layer on top of the final $<CLS>$ embedding with a softmax loss over the datasets’ categorical distribution of labels. For the early fusion baseline, we replicate the implementation by [18] and develop the RGB and depth embedding layers as 2D convolution layers with $D$ feature maps and kernel size and stride of $N$. The input channels are 3 for RGB and depth separately in the dual version and 6 for the joint embedder variant. For training the late fusion baseline, we generate two batches (RGB-labels, Depth-labels) and interleave the depth batch in-between the RGB images, so that the model processes the RGB-D pair in pairs of two, even in the case of distributed parallel training with multiple GPUs and/or nodes. We experiment with the default public configurations of ViT-x, where $x = \{T,S,B,L\}$ for {tiny, small, base, large}. We develop our method using PyTorch [36] and the Hugging Face API [48].

IV. EXPERIMENTS

This section describes our evaluation setup and presents our experimental results. It is organized as follows: First (Sec. IV-A), we give the specifics of the RGB-D dataset used...
for training and the evaluation procedure. Then (Sec. [IV-B]), we perform ablation studies for different variants of the depth representation and the fusion approach. We select the best performing configuration of our ablation studies and scale it to compare with previous state-of-the-art for RGB-D object recognition in the Washington benchmark (Sec. [IV-C]). Finally (Sec. [IV-D]), we study the performance of our approach when evaluated in an online lifelong learning scenario and (Sec. [IV-E]) demonstrate how it can be integrated with a robot framework for interactive robot learning applications.

### A. Dataset and Evaluation

The Washington RGB-D object dataset [29] is a well established benchmark for object recognition tasks in RGB-D domains. It contains up to 41877 views from 300 object instances, organized into 51 categories, including common household objects (cups, bowls, mugs etc.) with variations in fine-grained attributes (e.g. color). Each view was taken from 30◦, 45◦ and 60◦ elevation angles of a Kinect sensor. For depth representations, we use the surface normals as extracted from [6] and manually perform HHA conversions. Regarding evaluation, we perform the suggested experiment as in the original paper [29]. In particular, the dataset provides 10 train/test splits, where in each split, one instance per object category is used for testing and the rest for training. For a single trial, a total of 51 category instances (∼7k RGB-D pairs) are used for validation and the remaining 249 instances (∼35k RGB-D pairs) are used for training. We use top-1 predicted accuracy as the evaluation metric and report averaged mean and standard deviation of accuracies across the 10 trials.

### B. Ablation Studies

We ablate the following aspects of our approach: a) the format of the input depth image, as described in Sec. [II-B], b) the type of RGB-D fusion used (Early vs. Late), c) the type of embedder used in the Early fusion baseline (joint-versus-dual-emb) as well as d) the type of fusion method used in the Late fusion baseline, including averaging (avg), max pooling (max) and concatenation (cat). In order to reduce computational overhead, we experiment with ViT-T using only the first trial of the Washington RGB-D evaluation setup. In order to gain better insight in each configuration’s contribution, we use three different evaluation scenarios, namely: a) k-nearest neighbour (k-NN) on top of frozen pretrained embeddings, b) training a linear head on top of frozen pretrained embeddings (Lin.Eval), and c) fine-tuning the Transformer end-to-end with a classification head (FT). We note that the early baseline does not include results with frozen representations, as the encoder cannot be used out-of-the-box for RGB-D embeddings at the input. For RGB-D methods we report results using the SurfNorm depth format, as we observe that achieves best results. For k-NN, we use k = 3 and cosine similarity as the distance metric, as we experimentally find that this the best performing setup. For Lin.Eval, we train using minibatch SGD with momentum value 0.9, batch size 128, a learning rate of 5 · 10⁻⁴ and early stopping. For fine-tuning, we use AdamW [34], batch size 512, a learning rate of 9 · 10⁻⁵ with no warmup and a linear decay over 10 epochs. We train on 2 Nvidia Titan Xp GPUs. Our results are summarized in Table I.

We observe that the SurfNorm depth representation leads to best depth-only performance in ViT, as it is found in previous works using CNNs. Regarding early fusion, we also confirm that using separate embeddings for the two modalities (dual baseline) leads to marginally better results than the joint. Regarding late fusion, the concatenation operation overperforms other studied fusion approaches, with the cost of doubling the hidden size of the classifier’s input. When comparing the two fusion baselines, we observe that the late fusion baseline far outperforms the early one. In particular, the early baseline achieves worst results than RGB-only. We believe that this result reinforces our original hypothesis, namely that in the absence of large-scale multimodal datasets for pretraining, attempting to modify the input embeddings greatly disturbs the fine-tuning process, leading to overfitting. In contrast, in the late fusion baseline, the encoder "sees" the same representations but only learns to adapt the final layers to incorporate depth features. We confirm this statement by verifying that the weights in the adapted ViT have greater absolute difference on average in the early rather than the late baseline.

### C. Offline RGB-D Object Recognition

Table II presents results for the 10 trials of the evaluation setup, compared with previous state-of-the-art methods, as reported in [6]. We use ViT-B and the best configuration from our ablation experiments (i.e. Late + cat). For reference, we include results over the 10 trials using the dual-embedder early fusion architecture. We also include another baseline, ViT-B Ensemble (Ens.), in which we have fine-tuned the ViT on RGB and depth separately and then use an ensemble of both fine-tuned models during inference. The two final representations from the fine-tuned encoders are fused and fed to the classifier, as in late fusion. We highlight that this
ViT-B and ViT-L respectively.

<table>
<thead>
<tr>
<th>Method</th>
<th>RGB</th>
<th>Depth</th>
<th>RGB-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fusion 2D/3D CNNs</td>
<td>89.0 ± 2.1</td>
<td>78.4 ± 2.4</td>
<td>91.8 ± 0.9</td>
</tr>
<tr>
<td>STEM-CalRs [1]</td>
<td>88.0 ± 2.0</td>
<td>80.8 ± 2.1</td>
<td>92.2 ± 1.3</td>
</tr>
<tr>
<td>MM-IRF-ELM [30]</td>
<td>84.3 ± 3.2</td>
<td>82.9 ± 2.5</td>
<td>89.6 ± 2.5</td>
</tr>
<tr>
<td>VGG f-RNN [3]</td>
<td>89.9 ± 1.6</td>
<td>84.0 ± 1.8</td>
<td>92.5 ± 1.2</td>
</tr>
<tr>
<td>DECO [7]</td>
<td>89.5 ± 1.6</td>
<td>84.0 ± 2.3</td>
<td>93.6 ± 0.9</td>
</tr>
<tr>
<td>MDSI-CNN [2]</td>
<td>89.9 ± 1.8</td>
<td>84.9 ± 1.7</td>
<td>92.8 ± 1.2</td>
</tr>
<tr>
<td>HP-CNN [49]</td>
<td>87.6 ± 2.2</td>
<td>85.0 ± 2.1</td>
<td>91.1 ± 1.4</td>
</tr>
<tr>
<td>RC-Fusion [53]</td>
<td>89.6 ± 2.2</td>
<td>85.9 ± 2.7</td>
<td>94.4 ± 1.4</td>
</tr>
<tr>
<td>MMFLAN [38]</td>
<td>83.9 ± 2.2</td>
<td>84.0 ± 2.6</td>
<td>93.1 ± 1.3</td>
</tr>
<tr>
<td>DenseNet12T-RNN [6]</td>
<td>91.5 ± 1.1</td>
<td>86.9 ± 2.1</td>
<td>93.5 ± 1.0</td>
</tr>
<tr>
<td>ResNet101-RNN [6]</td>
<td>92.3 ± 1.0</td>
<td>87.2 ± 2.5</td>
<td>94.1 ± 1.0</td>
</tr>
<tr>
<td>Ours (ViT-B Ens.) [5]</td>
<td>90.8 ± 1.9</td>
<td>83.7 ± 2.1</td>
<td>90.4 ± 1.5</td>
</tr>
<tr>
<td>Ours (ViT-B Early)</td>
<td>-</td>
<td>-</td>
<td>89.5 ± 1.5</td>
</tr>
<tr>
<td>Ours (ViT-B Late)</td>
<td>92.6 ± 1.1</td>
<td>83.6 ± 2.4</td>
<td>94.8 ± 1.5</td>
</tr>
<tr>
<td>Ours (ViT-L Late)</td>
<td>92.9 ± 1.3</td>
<td>83.5 ± 2.1</td>
<td>98.1 ± 1.3</td>
</tr>
</tbody>
</table>

TABLE II: Mean and standard deviation results for top-1 predicted accuracy (%) over the 10 trials of the Washington RGB-D Objects dataset. Best results are highlighted in bold, and second best are underlined. Our approach achieves new state-of-the-art results in this benchmark. The method with 1 uses one encoder per modality, thus doubling the spatial requirements.

In order to evaluate in such an open-ended scenario, we present with novel object instances throughout a lifespan. In particular, we develop a simulated user who gradually introduces new object categories to the agent by presenting an unseen view of an object category. An illustration is introduced in Fig. 2. After teaching each new category, the user examines the classification accuracy over a collected set of instances and evaluate whether the category has been learned and interference has not happened. The user can also correct mis-classified examples in order to update the category models. Training and evaluating the agent is performed until a specific protocol threshold value is met (e.g. for threshold 0.67, the accuracy rate must be at least double from the error rate), after when a new category is introduced, or the agent learned all existing categories. Random sampling is used to select new data points from each category from the evaluation dataset.

In order to measure the effect of catastrophic forgetting in the evaluation, the user tests the agent in all previous categories after each new introduction. The evaluation stops either when the agent has learned all categories without catastrophic forgetting (according to the protocol threshold) or is unable to do after attempting it for more than a specified number of Question/Correction Iterations (QCI).

**TABLE III**

| Metric               | Average Protocol Accuracy (APA): 93.1 ± 1.3 | Average number of Learned Categories: 2 | Average number of stored Instances per Category (ALC): 0.9 | Global Classification Accuracy (GCA): 93.1 ± 1.3 | Average Protocol Accuracy (APA): 93.1 ± 1.3 |

In order to assess the performance of the different methods with stricter teachers, we repeat the evaluation while setting the value of the protocol threshold to 0.7. We report results in these metrics for Washington RGB-D Objects dataset, comparing with uni-modal baselines. In this experiment we use the k-NN classifier strategy on top of pretrained embeddings, which is compatible with the Instance-Based Learning (IBL) approach of evaluation protocol. As before, we use k = 3 and cosine distance function. Results are summarized in Table III.

We observe that even without any RGB-D fine-tuning, fusing the embeddings generated by the ImageNet ViT checkpoint still provides accuracy benefits over RGB-only and Depth-only classification, in all protocol threshold settings.

E. Robot Demonstrations

We develop a simulation environment in Gazebo to evaluate the real-time performance of the proposed approach in the

Fig. 2: Cartoon illustration of the evaluation protocol implemented for an open-ended lifelong learning scenario: The simulated user teaches a new category to the robot using three randomly selected instances, and then samples new instances to evaluate the robot on all learned categories, and make sure that interference has not happened after introducing the new category. Once a certain threshold of accuracy is met, the user introduces a new category. The user is also enabled to correct mis-classifications from the model.
TABLE III: Online lifelong learning evaluation on Washington RGB-D with a simulated teacher. Higher protocol thresholds values represent a "stricter" teacher, requiring more correct answers to consider a classification trial successful. We report results for ViT-B in RGB-only, Depth-only and our RGB-D late fusion baseline. The arrow demonstrates if better results are higher or lower for each metric (refer to the text for explanation of the evaluation metrics).

<table>
<thead>
<tr>
<th>Threshold</th>
<th>Method</th>
<th>Washington RGB-D Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>QCL</td>
</tr>
<tr>
<td>0.7</td>
<td>ViT-B (RGB)</td>
<td>1325</td>
</tr>
<tr>
<td></td>
<td>ViT-B (Depth)</td>
<td>1329</td>
</tr>
<tr>
<td></td>
<td>ViT-B (RGB-D Late)</td>
<td>1325</td>
</tr>
<tr>
<td>0.8</td>
<td>ViT-B (RGB)</td>
<td>1369</td>
</tr>
<tr>
<td></td>
<td>ViT-B (Depth)</td>
<td>2029</td>
</tr>
<tr>
<td></td>
<td>ViT-B (RGB-D Late)</td>
<td>1370</td>
</tr>
<tr>
<td>0.9</td>
<td>ViT-B (RGB)</td>
<td>2368</td>
</tr>
<tr>
<td></td>
<td>ViT-B (Depth)</td>
<td>2954</td>
</tr>
<tr>
<td></td>
<td>ViT-B (RGB-D Late)</td>
<td>1695</td>
</tr>
</tbody>
</table>

Fig. 3: A sequence of snapshots capturing the experimental setup and the behaviour of the robot in Gazebo (top) and in a real-world (bottom). We randomly place objects and instruct the robot to recognize them and place them in the container.

V. CONCLUSION

In this work we propose a simple yet strong recipe for fine-tuning ViTs in RGB-D domains. We experiment with two different types of fusion (early vs. late) and demonstrate that unlike most prior arts that use early fusion, the late fusion strategy transfers better in the low-data regime. By fine-tuning a ViT with our late fusion approach, we push the state-of-the-art in the Washington RGB-D Objects benchmark by 0.9%, using non optimal configurations due to computational restraints. We further show that our approach is more robust than unimodal approaches when the training-test paradigm is replaced with an open-ended lifelong learning scenario and demonstrate how it can serve as a perception utility for interactive lifelong robot learning, both in simulation and with a real robot. We hope that our approach will lead more research on efficiently transferring ViTs for robotics-specific domains.

This work leaves us with a multitude of potential future directions, regarding the sophistication of the RGB-D fusion, the efficiency of transferring and generalization to novel domains. For the first topic, the fusion method we present in this work is a single operation between modality-specific embeddings. Other approaches that entangle fusion within the encoder can be considered in the future, such as hierarchical feature fusion. Another limitation is that our method currently fine-tunes the entire pretrained model, setting a time and compute requirement that is still considerable. There is a broad literature in using adapters for efficient parameter-light fine-tuning of Transformers in NLP [23, 37, 20]. It would be interesting to explore adapters for even more efficient transfer of ViTs in RGB-D domains. Finally, regarding generalization, our model transfers from ImageNet and its high accuracy in the Washington benchmark is guaranteed due to overlap of existing object classes (as suggested by high k-NN RGB-only scores). We expect that this is not the case when moving in-the-wild. In the future we plan to investigate transferring ViTs that are pretrained with self-supervised objectives, such as masked autoencoding, and compare their transfer performance in-the-wild with supervised methods.


