LEARNING TRANSFERABLE REWARD FOR QUERY OBJECT LOCALIZATION WITH POLICY ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose a reinforcement learning based approach to the problem of *query object localization*, where an agent is trained to localize objects of interest specified by a small exemplary set. We learn a transferable reward signal formulated using the exemplary set by ordinal metric learning. It enables test-time policy adaptation to new environments where the reward signals are not readily available, and outperforms fine-tuning approaches that are limited to annotated images. In addition, the transferable reward can allow repurposing of the trained agent from one specific class to another class. Experiments on corrupted MNIST, CU-Birds, and COCO datasets demonstrate the effectiveness of our approach.

1 INTRODUCTION

There is increasingly interest in designing models that can be adapted to unseen environments or repurposed for new tasks, with only minimal human guidance (Vinyals et al., 2016; Snell et al., 2017; Finn et al., 2017; Sun et al., 2020; Hansen et al., 2021). A small set of examples can not only serve the purpose of implicitly generalizing the trained model to the new environment during training time, but also provide the chance of updating the learning objectives with the learner during test-time.

In this paper, we focus on the reinforcement learning (RL) formulation to the problem of *query object localization*, where an agent is trained to localize the target object specified by a small set of exemplary images. The vision-based agent can be viewed as a proactive information gatherer (Guo, 2003) that actively interacts with the image environment, following a class-specific localization policy, thus will be more suitable for aerial imagery (Xia et al., 2018), robotic manipulation (Kalashnikov et al., 2018), or embodied AI (Savva et al., 2019) tasks.

During test-time, the queried object to localize may be novel, or the background environment may undergo substantial change, hindering the applicability of class-agnostic agents with fixed policy. When reward signal is available, fine-tuning methods (Julian et al., 2020) can effectively adapt agents to the new environment and yield improved performance. Different with standard RL settings, in our application the reward signal is not available during test-time, as the bounding box annotations are to be found by the localization agent on test images.

To address this problem, we propose an ordinal metric learning based framework for learning an implicitly transferable reward signal defined with a small exemplary set. An ordinal embedding network is pre-trained with data augmentation under a loss function designed to be relevant to the RL task. The reward signal allows explicit updates of the controller in the policy network with continual training during test-time. Compared to fine-tuning approaches, the agents can get exposed to the new environment more extensively with unlimited usage of test images. Informed by the exemplary set precisely, the agent is versatile to the change of localization target.

2 RELATED WORK

2.1 OBJECT LOCALIZATION

Compared to bounding-box regression approaches (Redmon et al., 2016; Ren et al., 2016), deep RL based object localization approaches (Caicedo & Lazebnik, 2015; Jie et al., 2016) have the advantage of being region-proposal free, with customized search paths to each image environment.

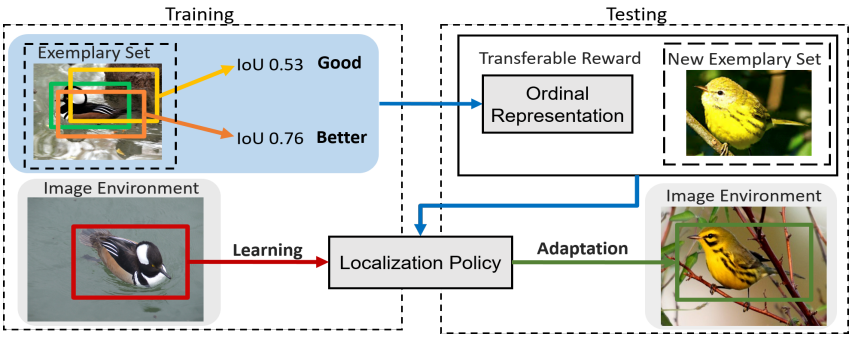

Figure 1: RL-based query object localization. Reward signal defined on an exemplary set with preference encoded in the ordinal representation. Policy trained and adapted to a new class.

The specificity of agent purely depends on the classes of bounding-boxes used in the reward. They can be made class-specific (Caicedo & Lazebnik, 2015), but the agent for each class would need to be trained separately.

Despite the rise of crowdsourcing platforms, obtaining ample amount of bounding-box annotations remains costly and error-prone. Furthermore, the quality of annotation often varies, and precise annotations for certain object class may require special expertise from annotators. The emergence of weakly supervised object localization (WSOL) (Song et al., 2014; Oquab et al., 2015; Zhou et al., 2016) methods alleviate the situation, which utilize image class labels in deriving bounding box annotations. It is known that WSOL methods have drawbacks of overly relying on the inter-class discriminative features and not being able to generalize to classes unseen during the training phase.

We argue that intra-class similarity is a more natural objective for the problem of localizing objects belonging to the target class. The name "query object localization" emphasizes more about the class-specific nature of the localization task, with the relevance defined by the similarity to the query images. A similar problem is image co-localization (Tang et al., 2014; Wei et al., 2017), where the task is to identify the common objects within a set of images. Co-localization approaches exploit the common characteristics across images to localize objects. Being unsupervised, co-localization approaches could suffer from ambiguity if there exist multiple common objects or parts, e.g., bird head and body (Jaderberg et al., 2015), which may provide the unwanted common objects as output.

2.2 Policy Adaptation

There seems to be a contradiction between the goals of training an agent with high task-specificity and better generalization performance to new situations at the same time. The key to reconcile these two goals lies in the usage of a small set of examples. There has been a paradigm shift from training static models defined with parameters to ones defined together with a support set (Vinyals et al., 2016; Snell et al., 2017), proved to be very effective in few-shot classification.

Besides the effort of meta learning implicitly adjustable models, fine-tuning on a pretrained model (Oquab et al., 2014; Finn et al., 2017; Julian et al., 2020) has also been widely used in transferring knowledge from data-abundant to data-scarce tasks. When reward signal is not available, Hansen et al. (2021) proposed a policy adaptation approach, in which the intermediate representation is fine-tuned via optimizing a self-supervised auxiliary loss while the controller is kept fixed. Our work shares the same motivation of test-time training (Sun et al., 2020), but we focus instead on the settings where the controller needs to be adapted or even repurposed for new tasks.

3 Method

3.1 Problem Formulation

In query object localization, we are given a set of images $\mathcal{I}$, and a small set of exemplary images $\mathcal{E}$. Our goal is to find the location of the bounding box containing the queried object in each test image, without candidate proposal boxes. Following the Markov Decision Process (MDP) framework for object localization (Caicedo & Lazebnik, 2015; Jie et al., 2016), we consider raw pixels in each image I_i as the ***Environment***. we define ***Action*** set with discrete actions facilitating a top-down search,

including five scaling, eight translation transformations as in (Jie et al., 2016), plus one stay action. Starting from the whole image pixels as input, the agent is trained to select the actions to transform current box at each step, by maximizing the total discounted ***Reward***. We use policy gradient with recurrent neural network (RNN) (Mnih et al., 2014) rather than DeepQNetwork with a vector of history actions and states. The agent takes pooled feature from the current box as ***State***, while it also maintains an internal state within RNN, which encodes information from history observations.

3.2 Learning Ordinal Reward Signal

Existing deep RL approaches for object localization (Caicedo & Lazebnik, 2015; Jie et al., 2016) use its ground-truth object bounding box g_i as the reward signal,

$$R = \text{sign}(\text{IoU}(b_t, g_i) - \text{IoU}(b_{t-1}, g_i)), \tag{1}$$

where $\text{IoU}(b_t, g_i)$ denotes the Intersection-over-Union (IoU) between the current window b_t and the corresponding ground-truth box g_i[1], and $\text{IoU}(b, g) = \text{area}(b \cap g)/\text{area}(b \cup g)$. Similar to the bounding box regression approaches which learns a mapping $\phi : I \mapsto g$, the image and box must be paired. However, annotated image-box pairs (I, g) may be scarce in both the training and testing phases. The reward signal in Eq. 1 is defined instance-wise, which is not transferable across images.

To address this problem, a natural idea is to define the reward signal based on learned representations between the cropped images by current window b_t, and the ground-truth window g. Given their M- dimensional representations $\mathbf{b_t}$ and $\mathbf{g}$ produced by an embedding function $f : \mathbb{R}^D \mapsto \mathbb{R}^M$ from D- dimensional image feature vectors, a distance function $d : \mathbb{R}^M \times \mathbb{R}^M \mapsto [0, +\infty)$ return the embedding distance $d(\mathbf{b_t}, \mathbf{g})$. However, $d(\mathbf{b_t}, \mathbf{g})$ may not decrease monotonically as the agent approaches to the ground-truth box g. As a result, the embedding distance based reward signal may be less effective than Eq. 1.

Ordinal metric learning. We propose to use an ordinal embedding based reward signal. For any two perturbed boxes b_j, b_k from ground truth g, embeddings $\mathbf{b_j}, \mathbf{b_k}, \mathbf{g}$ are learned, such that the relative preferences between any pair of boxes are preserved in the Euclidean space,

$$\rho_j > \rho_k \Leftrightarrow ||\mathbf{b_j} - \mathbf{g}|| < ||\mathbf{b_k} - \mathbf{g}||, \quad \forall j, k \in \mathcal{C}, \tag{2}$$

where ρ_j and ρ_k denote the preference (derived from either IoU to ground-truth box or ordinal feedbacks from user), and $\mathcal{C}$ is an ordinal constraint set constructed by sampling box pairs around g. This problem is originally posed as non-metric multidimensional scaling (Agarwal et al., 2007; Jamieson & Nowak, 2011). Although we apply a very simple pairwise-based approach, there exist other extensions such as the listwise-based (Cao et al., 2007), quadruplet-based (Terada & Luxburg, 2014) and landmark-based (Anderton & Aslam, 2019; Ghosh et al., 2019).

Data augmentation. We assume that during the training time, the exemplary set $\mathcal{E}_{\text{train}}$ contains both image I and box g. We adopt a tailored data augmentation scheme - *box perturbation*. We have found that using IoU-based partition scheme is more effective than random sampling (more details in 4.1). This can be viewed as a procedure to enhance the robustness of neural network (Szegedy et al., 2014) against box perturbations, and protect the special purpose of its usage on distinguishing reward increases or decreases. It also makes the downstream task of policy network training more efficient.

3.3 Model Architecture

In this paper, we define ρ as the IoU of box b to the ground-truth box g, i.e., $\rho = \text{IoU}(b, g)$. We choose to optimize the triplet loss (Hermans et al., 2017) for learning the desired embedding,

$$\mathcal{L}_{\text{triplet}} = \sum_{\text{anchor, ordinal pairs}} \max\Big(m + d(\mathbf{a}, \mathbf{p}) - d(\mathbf{a}, \mathbf{n}), 0\Big), \tag{3}$$

where $\mathbf{a}$ is the "anchor" embedding. $\mathbf{p}, \mathbf{n}$ are the "positive" and "negative" embeddings with larger and smaller IoUs with ground truth box g, respectively. We learn an embedding space consistent to the local ordinal constraints specified on the image pairs obtained via data augmentation.

[1]Notation: bounding box are denoted by lowercase lightface letters, e.g., g, while embedding vectors are denoted by boldface letters, e.g., $\mathbf{g} \in \mathbb{R}^d$ is a d-dimensional vector.

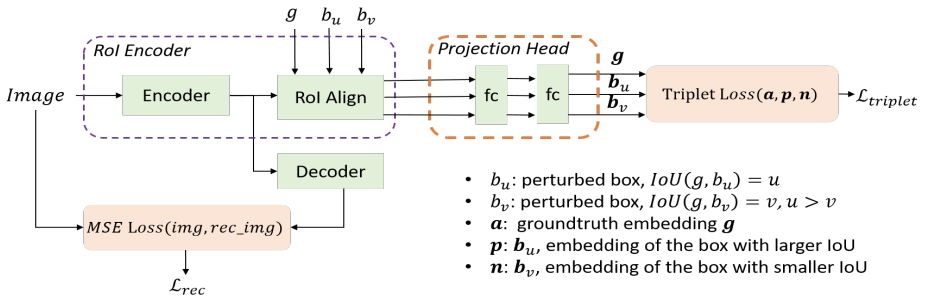

Figure 2: Learning RoI Encoder and Projection Head.

The choice of anchor. The anchor embedding $\mathbf{g}$ in Eq. 3 is not restrictive to the cropped image from the same image. For example, it could be replaced by the prototype embedding (Snell et al., 2017) of the exemplary set $\mathcal{E}$, $\mathbf{c} = 1/|\mathcal{E}| \sum_{i \in \mathcal{E}} \mathbf{g}_i$, where $\mathbf{g}_i$ is the embedding of the cropped image I_i by ground-truth box g_i. If images from multiple class are available, the prototype can be further made to be class-dependent, or clustering-based. We find that using prototype-based embedding as the anchor leads to better generalization performance than using $\mathbf{g}$ in some experiments. Moreover, even if the exemplary set during the test time only contains the cropped object, test-time policy adaptation is still feasible without image-box pairs.

State representation. Note that a good representation for defining reward may not necessarily be a good state representation at the same time - it may not contain enough information guiding the agent taking the right actions. Chen et al. (2020) suggests that adding a projection head between the representation and the contrastive loss substantially improves the quality of the learned representation. We find the use of projection head is crucial in balancing the two objectives in our task. The network architecture is shown in Fig. 2, in which an MLP projection head h is attached after an (Region of Interest) RoI encoder e. According to the given image and RoI, the RoI encoder extracts RoI feature $\mathbf{s}$ that will be used as the state representation for localization. The projection head learns ordinal embedding $\mathbf{b}$ for computing reward. The ROI alignment module handles boxes of different size. Under a joint loss function $\mathcal{L}_{embed} = \mathcal{L}_{rec} + \lambda \cdot \mathcal{L}_{trip}$, the state representation $\mathbf{s}$ can indirectly benefit from the ordinal supervision on $\mathbf{b}$, while it still must render satisfactory image reconstruction results. Beside the autoencoder scheme, the RoI encoder can use pre-trained network as well.

3.4 Training Localization Agent and Adaptation

As discussed in Section 3.2, we use the ordinal embedding rather than the bounding box coordinates to compute the improvement that agent makes, and the reward for an agent moving from state s' to s takes the following form,

$$R(s, s') = ||\mathbf{b}_t - \mathbf{c}|| - ||\mathbf{b}_{t-1} - \mathbf{c}||, \tag{4}$$

where $\mathbf{c}$ is the protoptype embedding, $\mathbf{b}_t$ is the embedding of the cropped region at step t. We expect positive reward if distance between embedded cropped image $\mathbf{b}_t$ and embedded query image $\mathbf{a}$ is decreased.For all experiments, we apply policy gradient (REINFORCE) (Williams, 1992) with entropy loss to encourage exploration. The full algorithm is outlined in Algorithm 1.

Test-time adaptation. During test-time, the agent has the option of further updating the policy network using the received reward from Eq. 4. To match test conditions, the training batch is split into two group and $\mathbf{c}$ is computed on a small subset that does not overlap with the training images to localize, while during test adaptation, $\mathbf{c}$ becomes the prototype of the test exemplary set $\mathcal{E}_{\text{test}}$. Different from $\mathcal{E}_{\text{train}}$, only cropped images are needed in $\mathcal{E}_{\text{test}}$.

Figure 3 shows an illustrative example of query object localization with task adaptation. Given two common objects coexisting in the image environment, the agent is trained to pick up one from the two during training, while quickly adapt to pick another one during testing. Different from unsupervised co-localization approaches (Wei et al., 2017) relying on low-level images clues to distinguish foreground and background, learning objectives can be specified explicitly by switching the train/test exemplary sets. With test-time adaptation, the agent can handle even partially conflict learning objectives.

Algorithm 1: Training localization agent using the proposed ordinal reward signal.

Require : initial policy π_θ, batch size N, pretrained e and h, learning rate α
Output : policy controller π_θ
for *sampled minibatch* $\{x_k\}_{k=1}^{N}$ **do**
  for *k=1,...N* **do**
    Construct an exemplary set $\mathcal{E}_{\text{batch}}$
    $\mathbf{g_j} \leftarrow h(e(x_k, g_j))$ `// compute the ordinal embedding`
    $\mathbf{c}_k \leftarrow \frac{1}{|\mathcal{E}_{\text{batch}}|} \sum_{j \in \mathcal{E}_{\text{batch}}} \mathbf{g}_j$ `// compute the prototype vector`
    $\tau_k \leftarrow \{ s_1, a_1, ...s_T, a_T \}$ by running policy π_θ `// sample trajectory`
    $\mathbf{b_t} \leftarrow$ take action a_t, and compute embedding $h(e(x_k, b_t))$ for current window b_t
    $R_k^t \leftarrow \|\mathbf{b_{t-1}} - \mathbf{c}_k\| - \|\mathbf{b_t} - \mathbf{c}_k\|$ `// compute reward`
  end
  $\theta \leftarrow \theta + \alpha \nabla_\theta \sum_{k=1}^{N} \log \pi_\theta \left(\sum_{t'=t}^{T} \gamma^{t'-t} R_k^t \right)$ `// policy update`
end

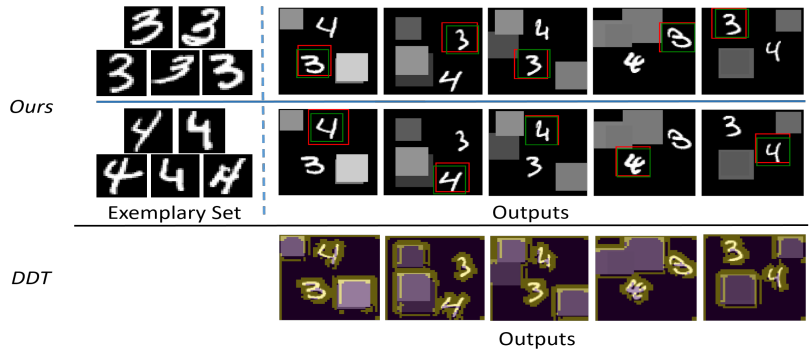

Figure 3: Selective localization vs. co-localization on two-digits data with random patch background.

3.5 Discussions

Our approach is based on the feature similarity with query images, which departs from previous bounding-box regression and RL approaches based on objectiveness. General supervised training methods are usually class-agnostic and require exposure to a large number of training classes, box-image pairs, and foreground and background variations in order to generalize well. In contrast, we allow specialized agents to be trained, with the ability of adapt to changes during the test-time.

Our approach includes three phases: ordinal representation learning, RL agent training, and test-time adaptation. Compared to general supervised training or fine-tuning methods, our approach is able to flexibly make use of various types of data in these phases (summarized in Table 1). Our agent can both take human feedback in terms of the exemplary set and perform test-time policy adaptation using unlabeled test data.

Table 1: Breaking up the requirements on data and labels in training and adaptation

	Supervised methods		Our approach		
	Training	Fine-tuning	Ordinal embedding	Agent training	Test-time adaptation
Image-box pairs	✓	✓	✓	✗	✗
Unlabeled images	✗	✗	✗	✓	✓
Exemplar images	✗	✗	✓	✓	✓

The transferability of our reward signal from training to testing crucially relies on the generalization ability of the learned ordinal representation. If the ordinal preference does not hold in the test domain, the proposed test-time policy adaptation scheme will not work. By adapting the representation with self-supervised objectives (Hansen et al., 2021), this issue might be remedied. We left it as future work. Although our approach does not directly handle the special cases of multiple queried objects or no queried object within the image environment, it can be easily modified to accomplish these tasks.

4 EXPERIMENT

In this section, we evaluate the generalization ability of the ordinal embedding as well as the performance of the trained localization agent on both source and target domain. The embedding-based reward not only improves the RL training, but also enables test-time adaptation of the learned policy. Experimental results demonstrate the effectiveness of our approach, with empirical comparisons to generic RL object localization, co-localization, and few-shot object detection baselines.

Evaluation protocol. To evaluate learned ordinal embedding, we use *OrdAcc* defined as the percentage of images within which the preference between a pair of perturbed boxes is correctly predicted. We use the Correct Localization (*CorLoc*) (Deselaers et al., 2012)) metric, which is defined as the percentage of images correctly localized according to the criterion $\text{IoU}(b_p, g) \geq 0.5$, where b_p is the predicted box and g is the ground-truth box. Mean and standard deviation $\pm$ (displayed as subscript) are reported from 10 independent runs (unless mentioned elsewise).

Implementation. We evaluate our approach on distorted versions of the MNIST handwriting dataset, the CUB-200-2011 birds dataset (Wah et al., 2011), and COCO dataset (Lin et al., 2014). For MNIST, we use three convolutional layers with ReLU activation after each layer as image encoder, while the same but mirrored structure as decoder to learn an autoencoder, and then attach RoI align layer following two fully connected (*fc*) layers as projection head for ordinal reward learning. For the CUB and the COCO datasets, we adopt layers before *conv5_3* of VGG16 pretrained on ImageNet for encoder. The projection head uses the same structure as before but with more units for each *fc* layer. All of our models were trained with Adam as optimizer (Kingma & Ba, 2015). All the models take less than one hour to finish training, implemented on PyTorch on a single NVIDIA A100 GPU.

4.1 ABLATION STUDY

We first study the effectiveness of ordinal embedding and reward as well as a few other design choices. We measure the generalization on test data *without* policy adaptation.

Setting and main results. We analyze the effectiveness of using ordinal embedding in terms of representation and reward first on synthetic datasets of corrupted MNIST digits under cluttered, random patch, Gaussian, or impulse noises (see Figure 8 in Appendix for examples). We want to find the answers to two main questions listed below:

1. Is the embedding distance based reward signal as effective as the IoU counterpart?
2. Does the ordinal triplet loss benefit the state representation learning?

Accordingly, we consider baselines in which embeddings are trained with only autoencoder or jointly with the ordinal projection head. Besides, we also compare the our embedding distance based reward against the IoU based reward used in Caicedo & Lazebnik (2015). Without loss of generality, the agent is trained on images of digit 4, and tested on images from all rest classes in testing. In particular, we are interested in evaluating the sample efficiency of different approaches. The results under different train set size are shown in Fig. 4. To distract the localization agent, cluttered noises with random 6×6 crops of digits are added to the background. All comparisons are based on the policy gradient (PG) method. With ordinal embedding presents in both the representation and reward ("AE+Ord+Embed"), the model performance is consistently better than other settings, especially when train set size is small. The benefit of triplet loss back propagation is evidenced by the comparison between "AE+Ord+IoU" and "AE+IoU". Intuitively, the similarity to the queried object is also a more natural goal than objectiveness defined by bounding boxes IoU, which is used in both generic object detection or localization and previous RL-based approaches.

Signed reward or not. Another intriguing question is that other than benefiting state representation learning, whether the embedding-based reward itself can bring any additional benefits than the IoU-based reward. From Figure 4, it appears to be a large gap between "AE+Ord+Embed" and 'AE+Ord+IoU", especially when the train set size is small. This is counter-intuitive as the ordinal reward is trained with supervision from the IoU in the embedding space, supposably it shall not bring any additional information. To further analyze this phenomenon, we take the sign operation off in

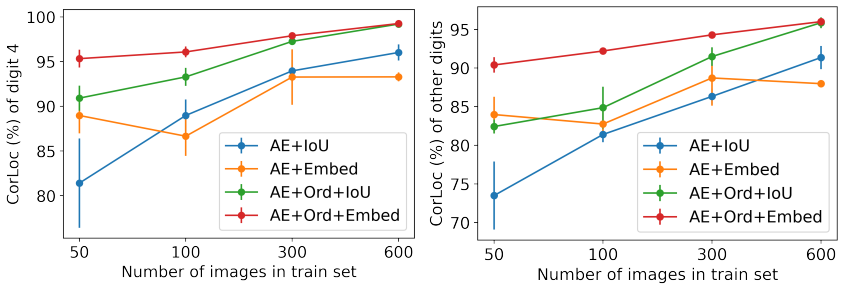

Figure 4: Comparison under different train set size. "AE+IoU": autoencoder, IoU based reward; "AE+Embed": autoencoder, embedding distance based reward; "AE+Ord+IoU": autoencoder and ordinal projection head, IoU based reward; "AE+Ord+Embed": autoencoder and ordinal projection head, embedding distance based reward. (a) Test performance on the trained digits 4, (b) Average test performance on the other nine digits.

Eq. 1 and retrain the models. In an experiment with 50 training images, with sign operation, the localization accuracy increase from 90.92% to 94.36% on test digit 4, and from 82.43% to 88.64% on other digits test set. This results show that unsigned reward works better in our case.

On policy vs. off policy. Many deep RL approaches are in favor to use deep Q-Network (DQN) to train the agent. Different from Caicedo & Lazebnik (2015) and Jie et al. (2016), we apply PG to optimize it. Besides, we adopt a top-down search strategy through RNN, while they used a vector of history actions to encode memory in these works. We evaluate these design choices with four baselines, all using "AE+Ord+Embd" for state and reward. As Table 2 shows, the agent achieves the best performance with "PG+RNN". Perhaps surprisingly, we find that empirically, with history action vectors the accuracy becomes worse when the agent is trained by DQN.

Table 2: $CorLoc(\%)$: DQN (with/without history action vector) vs. PG (with/without RNN)

Method	DQN	DQN+History	PG	PG+RNN
Digit 4	$88.80_{1.6}$	$86.54_{4.3}$	$88.98_{2.9}$	$\mathbf{94.68_{0.9}}$
Other digits	$84.21_{2.0}$	$81.75_{3.4}$	$81.91_{2.7}$	$\mathbf{89.05_{1.7}}$

Sampling scheme. We compare the efficiency of sampling schemes on generating augmented bounding box pairs, (a) *Random sampling*, where the pair of boxes are generated completely randomly, and (b) *Group-based sampling*, where dense boxes with variant scales are first generated, and then divided into 10 groups according to the IoU with ground-truth box. Each group has an IoU interval of 0.1. The sampling is first done on the group level, i.e., 2 out of 10, then sample one box from each group. Thus, the sampled boxes are likely to cover more diverse cases comparing to random sampling. Using the same number of pairs, the resulting *OrdAcc* for the two schemes is 90.12% (random sampling), 92.97% (group sampling), respectively.

4.2 Policy Adaptation during Test-Time

We further demonstrate the performance improvement with test-time policy adaptation. Through all the experiments, we assume a single source domain that contains abundant data annotations, and the target domain annotation are only available in an exemplary set of size 5. We compared our policy adaptation scheme with a standard fine-tuning scheme on the pre-trained policy network.

Results on the corrupted MNIST dataset. We use "digit 4 under random patch background noises" as the base class to train the ordinal embedding and the localization agent. During test-time, either the class of the foreground object or the background changes. The results on policy adaptation to *new digits* are shown in Table 3. Row 1 illustrates the transferability of the ordinal embedding reward, trained prototype embedding of a subgroup without the training instance, and evaluated using instances embedding from the same test image ("Ord Ins"). Rows 2 to 4 list the resulting localization accuracy after direct generalization ("Before"), fine-tuning on the exemplary set ("Finetune"), and

adaptation using all test images ("Adapt"), respectively. Our policy adaptation approach produces a substantial improvement over directly generalization, while fine-tuning approach experiences overfitting on the limited exemplary set. The localization accuracy on both the source background and *new backgrounds* environment are shown in Table 4, significant improvements are achieved using our policy adaptation scheme.

Table 3: $OrdAcc$ (%) and $CorLoc$ (%) on new digits environment.

	0	1	2	3	5	6	7	8	9	average
Ord Inst	$91.9_{0.7}$	$90.3_{2.9}$	$92.1_{1.8}$	$92.0_{0.2}$	$92.7_{0.6}$	$92.6_{0.2}$	$90.7_{0.7}$	$92.0_{0.7}$	$90.5_{0.7}$	$91.6_{0.5}$
Before	$94.2_{0.6}$	$84.1_{1.5}$	$88.7_{1.7}$	$86.5_{1.8}$	$81.2_{1.4}$	$91.9_{0.3}$	$89.5_{0.7}$	$93.0_{1.0}$	$90.8_{0.4}$	$88.9_{1.0}$
Finetune	$93.3_{2.5}$	$80.4_{3.9}$	$84.5_{3.5}$	$84.9_{1.5}$	$78.8_{2.7}$	$87.3_{2.9}$	$82.2_{5.4}$	$87.7_{4.5}$	$85.6_{5.3}$	$83.7_{4.5}$
Adapt	$99.8_{0.2}$	$95.6_{0.6}$	$98.1_{0.4}$	$97.9_{0.4}$	$88.3_{0.5}$	$99.1_{0.2}$	$98.6_{0.9}$	$99.8_{0.3}$	$99.2_{0.4}$	$97.4_{0.4}$

Table 4: $CorLoc(\%)$ when adapted to other background on corrupted MNIST dataset.

Adapt	Random patch	Clutter	Impulse noise	Gaussian noise	Mean
	$97.6_{0.4}$	$39.6_{0.5}$	$22.1_{0.7}$	$66.2_{2.0}$	56.4
✓	$\mathbf{100.0_{0.0}}$	$\mathbf{97.4_{0.3}}$	$\mathbf{99.9_{0.1}}$	$\mathbf{100.0_{0.0}}$	**99.3**

Results on the CUB dataset. We also evaluated the policy adaptation performance on the CUB dataset. The localization agent is trained on 15 species from the "Warbler" class, and adapted to different classes of "Warbler" (5 new species), "Wren", "Sparrow", "Oriole", "Kingfisher", "Vireo", and "Gull". Each test class contains a single bird species type, so we also implemented deep descriptor transforming (DDT) (Wei et al., 2017), a deep learning based co-localization approach, and added to the comparison.

Table 5: $CorLoc(\%)$ when adapted to other species/classes on CUB dataset.

	Adapt	Warbler (new)	Wren	Sparrow	Oriole	Kingfisher	Vireo	Gull	Mean
DDT		73.8	78.6	71.2	74.5	78.0	69.2	93.3	76.9
Ours		$85.5_{1.1}$	$82.9_{2.6}$	$81.3_{3.7}$	$77.9_{0.7}$	$78.9_{0.6}$	$82.2_{4.6}$	$86.3_{3.5}$	82.1
	✓	$\mathbf{89.7_{1.1}}$	$\mathbf{91.0_{1.1}}$	$\mathbf{89.3_{1.8}}$	$\mathbf{85.0_{0.8}}$	$\mathbf{85.9_{4.4}}$	$\mathbf{90.0_{0.9}}$	$\mathbf{93.9_{0.5}}$	**89.3**

4.3 TRANSFER FROM SPECIFIC TO SPECIFIC

Given a target domain one would like to deploy models to, a natural question arises on whether one should collect labeled data from an abundant number of source domains, or one specific class with potential lower data collection cost. On the COCO dataset, we selected a subset of images from five classes: *cow, cat, dog, horse, zebra*, and evaluated the transfer learning performance to the same target domain, from every other source domains. The results are shown in Fig. 5. The models perform best using embedding reward with adaptation (see "ours-adapt"). One interesting observation is that it is easy to transfer from other four classes to zebra, but not vice versa. A possible explanation might be the embedding net is biased on texture, while texture information is less adaptable than shape.

Few-shot object detection (Kang et al., 2019; Yan et al., 2019; Wang et al., 2019; 2020) methods are similar to our work in adapting models to a new test case with the help of limited annotations, but have a different requirement on the number of training classes. These methods usually require multiple classes in both stages of training and fine-tuning. On the COCO dataset, we compared the performance of our methods with TFA (Wang et al., 2020) and Meta-RCNN Yan et al. (2019), on the same *one-way 5-shot* test setting. Few-shot baselines are trained on 60 base classes and fine-tuned on all 80 classes, while our model is trained from one single class of the 5 classes randomly selected from the base classes set: *elephant, sheep, giraffe, laptop* and *suitcase*. Table 6 shows that with ordinal embedding, the agent achieves better performance even without adaptation, and performance can be further improved after adaptation.

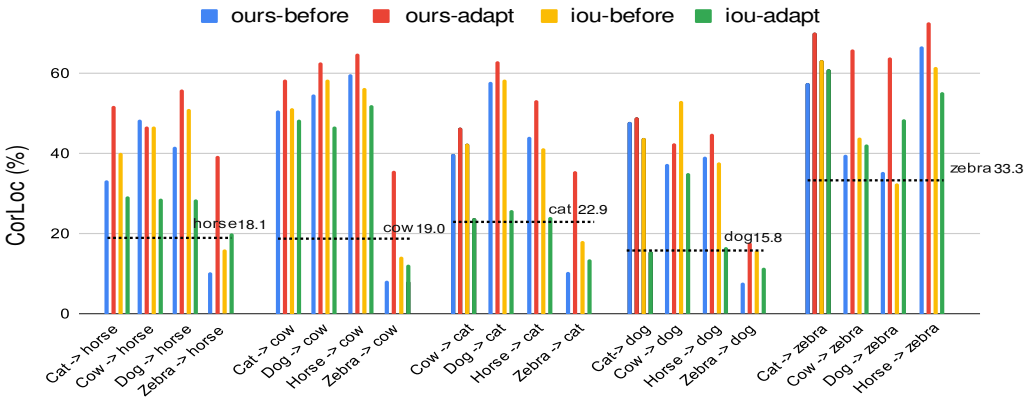

Figure 5: $CorLoc(\%)$ Comparison with IoU-based reward on COCO dataset. "ours": ordinal embedding based reward; "iou": IoU based reward, imagenet pretrained embedding. Dotted lines are the results of directly training on target class, using embedding based reward but from a ImageNet pretrained model, indicating the advantage of our approach, with ordinal embedding learned from a single source class and policy adaptation.

Table 6: $CorLoc(\%)$ Adaptation comparison with other methods on COCO dataset.

target	cat	dog	cow	horse	mean
FRCN+ft-full (Yan et al., 2019)	13.1	3.1	3.7	7.4	6.8
TFA w/ fc (Wang et al., 2020)	29.1	13.0	5.0	10.7	14.4
TFA w/ cos (Wang et al., 2020)	28.0	10.3	4.5	8.9	12.9
Ours-before	23.0	20.6	24.5	21.2	22.3
Ours-adapt	40.3	33.5	43.1	40.2	39.3

5 Conclusion

We propose an ordinal representation learning based reward, for training a localization agent to search queried object of interest. In particular, we use a small exemplary set as a guidance signal for delivering learning objectives, which can avoid potential learning ambiguity. Meanwhile, we use test image environments to inform the agent about the domain shifts, without requiring image-box pairs during the test-time.

Annotation collection plays an important role in building machine learning systems. It is one task that could benefit greatly from automation, especially in cost-sensitive applications. We aim to reduce the human labeling efforts, in terms of the number of annotate samples per class, number of annotate classes (one-class at the minimum), and the level of accuracy required, all under the same query-adaptation framework. Instead of jointly training the localization and classification model, we learn box annotations from image class labels, in a similar spirit with weakly-supervised learning. Empirically, we focus extensively on the *transfer learning* setting from one single data abundant training task to data-scarce (one-way few-shot) test tasks. Our approach is also applicable to the many-way few-shot setting where limited annotations across multiple classes are available during training, and we leave it for future study.

In making the trained RL agent generalizable to new unseen test cases, one can either expose the agent to as many cases as possible during training, or make the agent more specific but adaptable upon changed scenarios. Our approach belongs to the latter, which shows that even if only one training class is accessible, the agent with transferable reward can still adapt to various cases during the test time. As an extension work, curriculum learning with a designed sequence of targets in the exemplary set may guide the agent toward solving challenging localization tasks. We leave the robustness to bias and vulnerability to adversarial attacks for future study.

REPRODUCIBILITY STATEMENT

Code is available at:

https://anonymous.4open.science/r/Localization-by-Ordinal-Embedding-2C41

More implementation details and intermediate results can be found in Appendix A.2.

REFERENCES

Sameer Agarwal, Josh Wills, Lawrence Cayton, Gert Lanckriet, David Kriegman, and Serge Belongie. Generalized non-metric multidimensional scaling. In *Artificial Intelligence and Statistics*, pp. 11–18. PMLR, 2007.

Jesse Anderton and Javed Aslam. Scaling up ordinal embedding: A landmark approach. In *International Conference on Machine Learning*, pp. 282–290. PMLR, 2019.

Juan C Caicedo and Svetlana Lazebnik. Active object localization with deep reinforcement learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2488–2496, 2015.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *International Conference on Machine Learning*, pp. 129–136, 2007.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pp. 1597–1607. PMLR, 2020.

Thomas Deselaers, Bogdan Alexe, and Vittorio Ferrari. Weakly supervised localization and learning with generic knowledge. *International Journal of Computer Vision*, 100(3):275–293, 2012.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pp. 1126–1135. PMLR, 2017.

Nikhil Ghosh, Yuxin Chen, and Yisong Yue. Landmark ordinal embedding. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

AnYuan Guo. Decision-theoretic active sensing for autonomous agents. In *International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 1002–1003, 2003.

Nicklas Hansen, Rishabh Jangir, Yu Sun, Guillem Alenyà, Pieter Abbeel, Alexei A Efros, Lerrel Pinto, and Xiaolong Wang. Self-supervised policy adaptation during deployment. In *International Conference on Learning Representations*, 2021.

Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, pp. 2017–2025, 2015.

Kevin G Jamieson and Robert D Nowak. Low-dimensional embedding using adaptively selected ordinal data. In *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1077–1084. IEEE, 2011.

Zequn Jie, Xiaodan Liang, Jiashi Feng, Xiaojie Jin, Wen Lu, and Shuicheng Yan. Tree-structured reinforcement learning for sequential object localization. In *Advances in Neural Information Processing Systems*, pp. 127–135, 2016.

Ryan Julian, Benjamin Swanson, Gaurav S Sukhatme, Sergey Levine, Chelsea Finn, and Karol Hausman. Never stop learning: The effectiveness of fine-tuning in robotic reinforcement learning. *arXiv e-prints*, pp. arXiv–2004, 2020.

Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on Robot Learning*, pp. 651–673. PMLR, 2018.

Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8420–8429, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755, 2014.

Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. In *Proceedings of the 27th International Conference on Neural Information Processing Systems-Volume 2*, pp. 2204–2212, 2014.

Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1717–1724, 2014.

Maxime Oquab, Léon Bottou, Ivan Laptev, and Josef Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 685–694, 2015.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, 2016.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2016.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9339–9347, 2019.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 4080–4090, 2017.

Hyun Oh Song, Ross Girshick, Stefanie Jegelka, Julien Mairal, Zaid Harchaoui, and Trevor Darrell. On learning to localize objects with minimal supervision. In *International Conference on Machine Learning*, pp. 1611–1619. PMLR, 2014.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning*, pp. 9229–9248. PMLR, 2020.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations,*, 2014.

Kevin Tang, Armand Joulin, Li-Jia Li, and Li Fei-Fei. Co-localization in real-world images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1464–1471, 2014.

Yoshikazu Terada and Ulrike Luxburg. Local ordinal embedding. In *International Conference on Machine Learning*, pp. 847–855. PMLR, 2014.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in Neural Information Processing Systems*, 29:3630–3638, 2016.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset, 2011.

Xin Wang, Thomas Huang, Joseph Gonzalez, Trevor Darrell, and Fisher Yu. Frustratingly simple few-shot object detection. In *International Conference on Machine Learning*, pp. 9919–9928, 2020.

Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Meta-learning to detect rare objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9925–9934, 2019.

Xiu-Shen Wei, Chen-Lin Zhang, Yao Li, Chen-Wei Xie, Jianxin Wu, Chunhua Shen, and Zhi-Hua Zhou. Deep descriptor transforming for image co-localization. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 3048–3054, 2017.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3974–3983, 2018.

Xiaopeng Yan, Ziliang Chen, Anni Xu, Xiaoxi Wang, Xiaodan Liang, and Liang Lin. Meta r-cnn: Towards general solver for instance-level low-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9577–9586, 2019.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2921–2929, 2016.

A APPENDIX

A.1 ADDITIONAL EXPERIMENTS

A.1.1 VISUALIZATION OF DEEP DESCRIPTOR TRANSFORMING

Deep Descriptor Transforming (DDT) (Wei et al., 2017) is a co-localization method. It applies PCA on deep descriptors to generate an indicator matrix, the values of which reflect the correlations of these deep descriptors. The bounding box of the common object is then predicted by setting a threshold to filter out negatively correlated locations. However, when there are multiple common objects in the dataset, without specifying the desired target object, these co-localization methods will fail. Figure 6 shows examples when DDT is applied on corrupted MNIST with random patch background. As most of the patches are recognized as positively correlated (highlighted in yellow) to a common object, the method fails to localize target object digit 4.

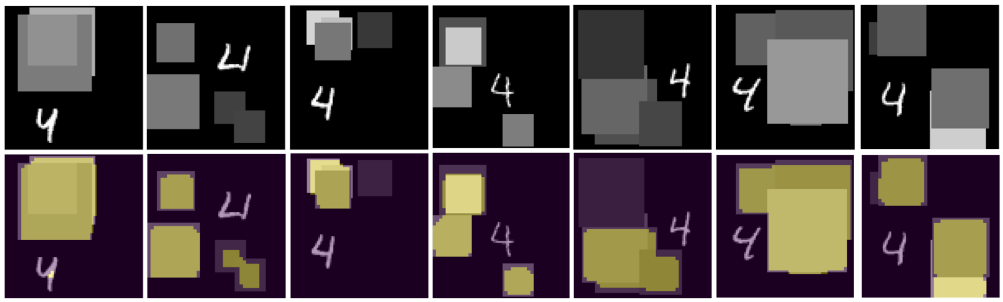

Figure 6: DDT (Wei et al., 2017) fails when there are multiple common objects in a dataset. **First row:** original image. **Second row:** patches are considered positive correlated, as they are also common objects. (Best viewed in color.)

A.1.2 ANNOTATION REFINEMENT

We explore the direction of iterative refinement of annotation quality with minimal human guidance. Given an agent trained extensively with inexact human annotations, can we adapt it to refine the annotation with a few well-annotated examples? We purposefully enlarge the ground truth bounding boxes in the corrupted MNIST and CUB datasets, and adapt the trained agent using the original ground truth box in an exemplary set of size 5.

For corrupted MNIST dataset, the agent, RoI encoder and projection head are first trained with 50 digit 3 images with loose (38×38) bounding box annotation. Then the agent is adapted to digit 1 images, where the exemplary size=5, and the ground-truth boxes are 28×28. For CUB dataset, during training, the ground-truth box in each resized 224×224 image is expanded 10 pixels each side. During adaptation we use the original ground-truth boxes. In adaptation from entire body to head experiment on CUB dataset, we train on original entire body ground-truth bounding box. During adaptation, 15 randomly sampled exemplar images are annotated with head bounding box. The agent is trained to adapt to localize head by minimizing the distance between embedding of predicted box and prototype of head embeddings. The results are shown in Figure 7 (left panel). Preliminary results with agent trained on the entire bird body and adapt to bird head are shown in Figure 7 (right panel), with a few successful cases on the first row and failure cases on the second row.

A.1.3 FEW-SHOT LOCALIZATION ON OMNIGLOT

The proposed ordinal reward signal also makes our approach amenable to few-shot training, when only a small subset of training images per class are annotated. The ordinal reward can be viewed as meta information. We evaluate our method under few-shot setting on corrupted Omniglot dataset (Lake et al., 2015) and CUB-warbler dataset. For Omniglot, We put each 28×28 character in 84×84 random patch background. The train and test set contains 25 different classes respectively, thus 500 images for each set. We randomly sample 100 iterations for training and testing. For CUB-warbler

Figure 7: Annotation refinement. **Left:** Loose to tight transfer. **Right:** entire body to head transfer.

datset, as we did in Sect. 4.2 we train on the same 15 species from the "Warbler" class, and adapted to 5 new species of "Warbler", thus 896 and 294 images respectively. We randomly sample 100 and 50 iterations for training and testing. We use 5-shot 5-way, set exemplary set size as 5, and use proto training strategy for both dataset. The results are shown in Table 7. As an implicit meta learning method, our approach achieves 99.94% and 90.52% $CorLoc$ on the two datasets. We can also leverage explicit meta learning method, such as MAML (Finn et al., 2017) to further improve the results. We will leave this part as future work. Although initial results are promising, more efforts are needed to validate whether the proposed RL approach can achieve state-of-the-art performance, but it is beyond the scope of this work.

Table 7: Evaluation under few-shot localization setting.

Dataset	$OrdAcc(\%)$	$CorLoc(\%)$
Omniglot	$95.11_{\pm 0.6}$	$99.94_{\pm 0.1}$
CUB-warbler	$91.28_{\pm 0.9}$	$90.52_{\pm 0.7}$

A.1.4 Size of Exemplary Set

We also compare the effect of different size of exemplary set during training and adaptation on CUB-warbler dataset. During training stage, we use shuffle proto training strategy, and set exemplary set size as 2, 5, 15, 25. The results without adaptation on test set are in Table 8. Both $OrdAcc$ and $CorLoc$ increase with exemplary set size. For adaptation stage, the range of exemplary set size is from 2 to 200. And the results are in Table 9. The test performance does not increase much with the exemplary set size. One possible explanation is that the data points in embedding space are compact, thus prototype doesn't change much when increasing exemplary set size. We will analyze the influence of multiple protoptypes per class in future experiments.

Table 8: Effect of exemplary set size during training stage.

Size	$OrdAcc(\%)$	$CorLoc(\%)$
2	$94.39_{\pm 1.7}$	$84.18_{\pm 6.5}$
5	$94.83_{\pm 2.0}$	$88.10_{\pm 0.2}$
15	$95.69_{\pm 1.7}$	$89.22_{\pm 1.9}$
25	$93.82_{\pm 1.0}$	$89.64_{\pm 2.3}$

Table 9: Effect of exemplary set size during adaptation.

Size	2	5	50	100	150	200
$CorLoc(\%)$	$89.12_{\pm 1.9}$	$89.67_{\pm 1.1}$	$90.15_{\pm 0.8}$	$90.36_{\pm 0.5}$	$89.63_{\pm 0.2}$	$90.14_{\pm 0.5}$

A.2 REPRODUCIBILITY

Corrupted MNIST. Examples of the four types of corrupted MNIST images are shown in Fig. 8.

Figure 8: Corrupted MNIST Datasets: 28×28 digits on four kinds of 84×84 noisy background.

Training strategies. We further evaluate the choice of anchor in the triplet loss for both the pretraining of state represtation and the ordinal reward for the training of agent. We study (i) whether ordinal embedding can be trained in reference to an anchor from a different image instances, (ii) whether it is advantageous to use the proptotype embedding of an exemplary set, rather than instance embeddings, and (iii) whether mimicking the test condition in training yields any improvement.

Table 10: Anchor choice comparison. Note that in evaluations the $OrdAcc$ is always computed using instance as anchor.

Mode	$OrdAcc(\%)$	$CorLoc(\%)$
Self	$97.2_{\pm 0.7}$	$61.0_{\pm 2.0}$
Proto	$95.2_{\pm 1.6}$	$77.9_{\pm 0.4}$
Shuffle self	$92.4_{\pm 1.4}$	$73.8_{\pm 2.5}$
Shuffle proto	$96.2_{\pm 1.5}$	$88.1_{\pm 0.2}$

We use the CUB-Warbler dataset with more foreground background variations than the corrupted MNIST dataset. The train and test set contains 15 and 5 disjoint fine-grained classes respectively, resulting 896 images for training (viewed as a single class) and 294 for testing. Table 10 shows the $OrdAcc$ and $CorLoc$ in four settings. “Self", use the embedding from images cropped by the ground-truth box from the same instance; “Shuffle self", use the ground-truth box cropped image emebedding from a different instance; Similarly, “Proto", use the prototype of a subgroup containing the training instance within the same batch; “Shuffle proto (SP)", use the prototype of a subgroup from a different batch without the training instance. Results suggest that this training strategy brings compactness to the train set, constructing an ordinal structure around the cluster. For “Shuffle proto", while the $OrdAcc$ is lower than others, its $CorLoc$ is the best with large margin. Matching the condition between training and testing indeed improves generalization to new classes on this dataset.

A.2.1 IMPLEMENT DETAILS OF ADAPTATION TO NOVEL TASKS

Selective localization. We investigate the agent's ability in localizing the object specified by the query set, when the set of images have two common objects. We use random patched MNIST, where each image has digit 3 and digit 4. First, the RoI encoder and projection head is trained with an additional contrastive loss to enlarge the distance between the two digits in embedding space,

$$loss_{embed} = loss_{rec} + \lambda_{trip} \cdot loss_{trip} + \lambda_{contr} \cdot loss_{contr}, \tag{5}$$

where $loss_{trip} = loss_{trip_3} + loss_{trip_4}$, learning two local ordinal structure around each class center in embedding space. We set the margin for both triplet losses as 10, and the margin for contrastive loss as 320 heuristically. We found that the larger the gap between the two margins, the better the performance. After learning the RoI encoder and projection head, we train the agent with the reward defined with Eq. 4 in Sect. 3.3, where $\mathbf{a}$ is the prototype embedding of the targeted digit exemplary set. Figure 5 in Sect. 4.3 shows an example where the exemplary size is 5.

A.2.2 ARCHITECTURE OF THE AGENT

The architecture of the agent is shown in Fig. 9.

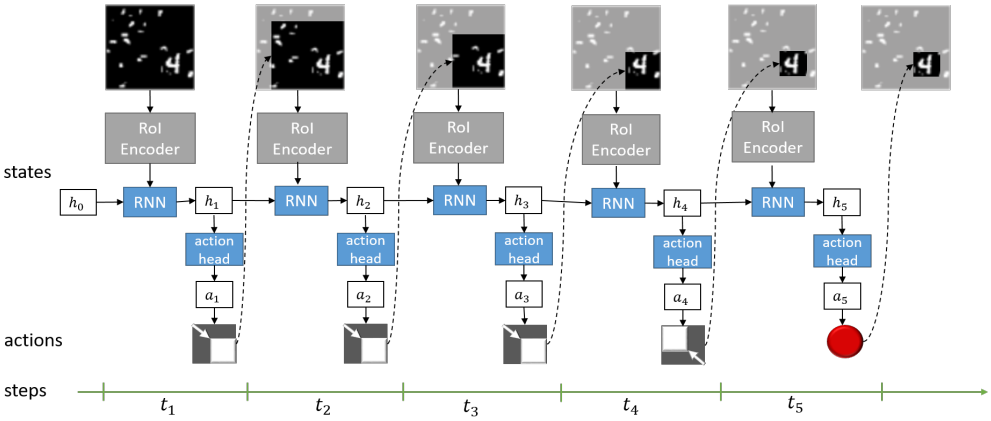

Figure 9: Agent architecture. Starting from the whole image, the agent is trained to select the actions to transform current bounding box at each step. The agent takes pooled feature from the current box as State, while it also maintains an internal state within RNN, which encodes information from history observations. The Action set is defined with discrete actions facilitating a top-down search, including five scaling, eight translation transformations as in (Jie et al., 2016), plus one stay action.

A.2.3 INTERMEDIATE RESULTS

The agent tends to predict expected actions to transform the box so that it can finally get to the target location. It can also be observed from Figure. 10 that the embedding distance between predicted boxes and ground-truth boxes is in agreement with the corresponding IoU. This further demonstrates the effectiveness of the localization agent.

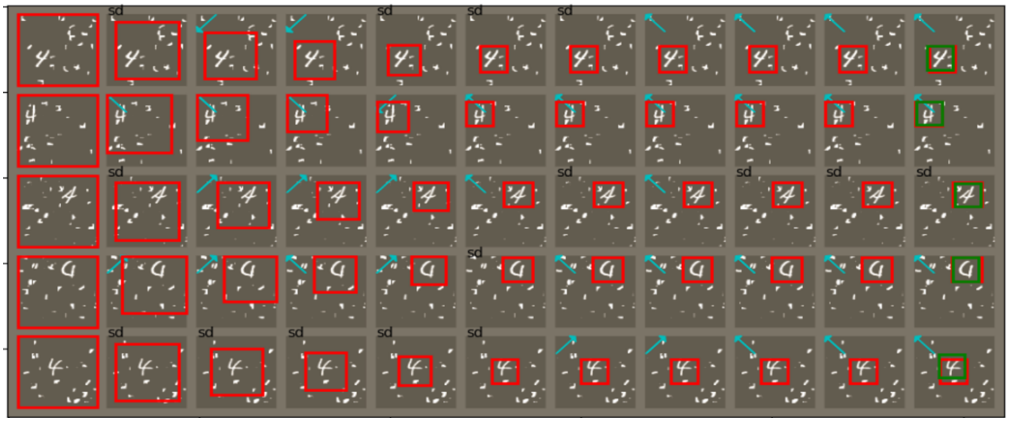

Figure 10: Starting from the whole image, the agent is able to localize digit 4 within 10 steps. Each row shows localizing one image, and each column is a time step. Green box means ground-truth bounding box.

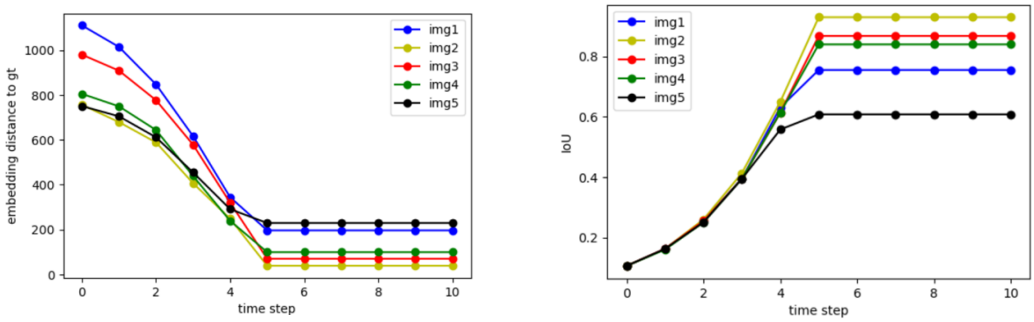

Figure 11: The IoU relation between ground-truth and predicted box is accurately represented in embedding space. **Left:** the embedding distance between ground-truth and predicted box in each step. **Right:** the IoU between ground-truth and predicted box in each step.