Adaptive Slot Attention: Object Discovery with Dynamic Slot Number

Ke Fan[1], Zechen Bai[2], Tianjun Xiao[3], Tong He[3], Max Horn[4],
Yanwei Fu[1,†], Francesco Locatello[5], Zheng Zhang[3]
[1]Fudan University [2]National University of Singapore [3]Amazon Web Services
[4]GSK.ai [5]Institute of Science and Technology Austria

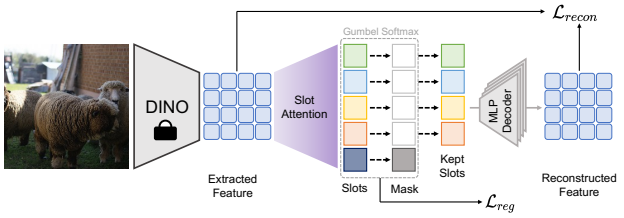

Figure 1. Illustration of our pipeline.

Object-centric learning (OCL) extracts the representation of objects with slots, offering an exceptional blend of flexibility and interpretability for abstracting low-level perceptual features. A widely adopted method within OCL is slot attention, which utilizes attention mechanisms to iteratively refine slot representations. However, a major drawback of most object-centric models, including slot attention, is their reliance on predefining the number of slots. This not only necessitates prior knowledge of the dataset but also overlooks the inherent variability in the number of objects present in each instance.

To overcome this fundamental limitation, we present a novel *complexity-aware* object auto-encoder framework as follows:

$$\begin{aligned} \min \quad & \mathbb{E}_Z \, \mathcal{L}_{recon}\left(\hat{x}, x\right) + \lambda \cdot \mathcal{L}_{reg}\left(\pi\right) \\ \text{where} \quad & S_1, \cdots, S_{K_{max}} = g_{slot}\left(f_{enc}(x)\right), \\ & Z \sim \pi\left(z\right), \; \hat{x} = f_{dec}\left(S, Z\right) \end{aligned} \tag{1}$$

where g_{slot} is the slot attention bottleneck, $\pi(z) = \pi(z_1, \cdots, z_{K_{max}})$ is a learnable sampling strategy, and complexity regularization $\mathcal{L}_{reg}$ is defined as expectation number of kept slots. Compared to ordinary slot attention, before decoding, our **AdaSlot** (Adaptive Slot Attention) first drops the unimportant slot, then reconstructs the images/features.

Naturally, without any regularization, AdaSlot tends to greedily keep all the slots, as more slots generally lead to better reconstruction quality. In contrast, our complexity regularization compels the model to achieve the reconstruction objective while utilizing as few slots as possible.

To learn the sampling strategy, we use a light-weight neural network $h_\theta : \mathbb{R}^D \to \mathbb{R}^2$ to predict the slot sampling strategy:

$$\pi = \text{Softmax}(h_\theta(S)) \in \mathbb{R}^{K \times 2}. \tag{2}$$

To keep the gradient propagation, we apply the Gumbel-Softmax with Straight-Through Estimation. The masked slot decoder $f_{dec}\left(S, Z\right)$ suppresses the information of dropped slots according to sampling result Z. The whole pipeline is displayed in Fig. 1.

We compared AdaSlot with fixed-slot models with different numbers on synthetic datasets CLEVR10 and MOVi-C/E, and real-world dataset COCO. Extensive studies demonstrate the effectiveness of our AdaSlot in two folds. First, our AdaSlot achieves comparable or superior performance to those best-performing fixed-slot models on several datasets. Second, to some extent, our AdaSlot is capable of selecting an appropriate slot number based on the complexity of the specific image as shown in Fig. 2.

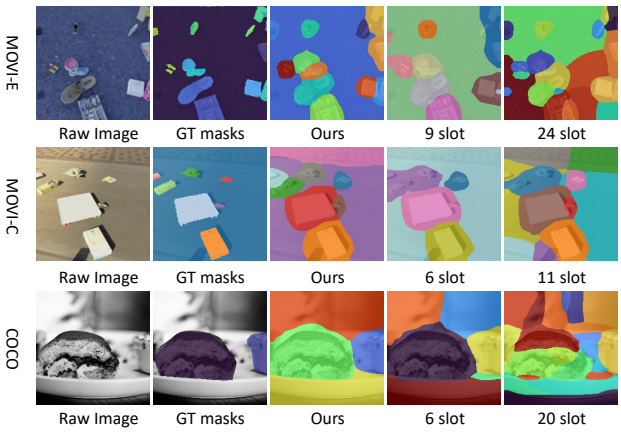

Figure 2. Visualization of instance-level adaptive slot number selection. We compare our AdaSlot and two fixed-slot models on three datasets.

Max and Francesco did the work at Amazon; † corresponding authors; This paper will be presented on CVPR 2024 main conference.