σ-Adaptive Decoupled Prototype for Few-Shot Object Detection (Supplementary Material)

In the supplementary material, we present additional details which are not included in the main paper due to space limitations, to provide further insights into our method.

Specifically, we include:

- Theoretical proof (§A and §B).
- Effect of η on SigmE Power Normalization (§C).
- Implementation details of K=1 (§D).
- Pipelines of per-class prototype *vs*. per-sample prototype (§E).
- Applying σ-ADP to transformer-based FCT (§F)
- Visualization results (attention maps) of the entangled *vs*. disentangled prototypes, and detection boxes of σ-ADP (§G).

A. Approximating the optimal prototype is equivalent to minimizing the variance

For meta-learning-based detectors, the prototypes of L classes ($\bar{\mathbf{\Phi}}_{\boldsymbol{L}} \equiv \{\bar{\mathbf{\Phi}}_{\boldsymbol{l}}\}_{l \in \mathcal{I}_L}$) should have the maximum similarity to K support samples ($\mathbf{\Phi}_{\boldsymbol{k}}$), per class. This is indicated by the maximum expectation of cosine similarity within the same class and across L classes, as follows:

$$\max \mathbb{E}_{\bar{\mathbf{\Phi}}_{\boldsymbol{l}}}[\mathbb{E}_{\mathbf{\Phi}_{\boldsymbol{k}}}[Cos(\bar{\mathbf{\Phi}}_{\boldsymbol{l}}, \mathbf{\Phi}_{\boldsymbol{k}})]],$$

$$\mathbb{E}_{\bar{\mathbf{\Phi}}_{\boldsymbol{l}}, \mathbf{\Phi}_{\boldsymbol{k}}}\Big[\frac{\bar{\mathbf{\Phi}}_{\boldsymbol{l}} \cdot \mathbf{\Phi}_{\boldsymbol{k}}}{\|\bar{\mathbf{\Phi}}_{\boldsymbol{l}}\|_2 \cdot \|\mathbf{\Phi}_{\boldsymbol{k}}\|_2}\Big] = \mathbb{E}\Big[\frac{\bar{\mathbf{\Phi}}_{\boldsymbol{l}}}{\|\bar{\mathbf{\Phi}}_{\boldsymbol{l}}\|_2}\Big] \cdot \mathbb{E}\Big[\frac{\mathbf{\Phi}_{\boldsymbol{k}}}{\|\mathbf{\Phi}_{\boldsymbol{k}}\|_2}\Big]$$

Demonstrated by [31] (Refer to Propositional 5.), the expectation of ratio is a closed-form approximation to the ratio of expectation in computer vision application, then:

$$\mathbb{E}\Big[\frac{\bar{\mathbf{\Phi}}_{\boldsymbol{l}}}{\|\bar{\mathbf{\Phi}}_{\boldsymbol{l}}\|_2}\Big] \cdot \mathbb{E}\Big[\frac{\mathbf{\Phi}_{\boldsymbol{k}}}{\|\mathbf{\Phi}_{\boldsymbol{k}}\|_2}\Big] \approx \frac{\mathbb{E}[\bar{\mathbf{\Phi}}_{\boldsymbol{l}}] \cdot \mathbb{E}[\mathbf{\Phi}_{\boldsymbol{k}}]}{\mathbb{E}[\|\bar{\mathbf{\Phi}}_{\boldsymbol{l}}\|_2] \cdot \mathbb{E}[\|\mathbf{\Phi}_{\boldsymbol{k}}\|_2]}$$

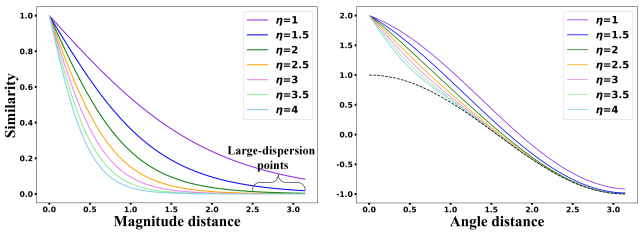
According to the relation of expectation and variance, we have:

$$\mathbb{E}[\|\mathbf{\Phi}_{\boldsymbol{k}}\|_2^2] = \mathbb{D}[\|\mathbf{\Phi}_{\boldsymbol{k}}\|_2] + \mathbb{E}[\|\mathbf{\Phi}_{\boldsymbol{k}}\|_2]^2,$$
$$\sqrt{\mathbb{E}[\|\mathbf{\Phi}_{\boldsymbol{k}}\|_2^2]} \geq \mathbb{E}[\|\mathbf{\Phi}_{\boldsymbol{k}}\|_2]$$

The vectors ϕ_k and $\bar{\phi}_l$ are both C-dimensional, and we assume each dimension of a vector is independent.

$$\mathbb{E}[\|\mathbf{\Phi}_k\|_2^2] = \mathbb{E}[\sum_{i=1}^{C}(\phi_i^k)^2] = \sum_{i=1}^{C}[\mathbb{D}[\phi_i^k] + \mathbb{E}[\phi_i^k]^2],$$
$$\mathbb{E}[\|\bar{\mathbf{\Phi}}_l\|_2^2] = \mathbb{E}[\sum_{i=1}^{C}(\bar{\phi}_i^l)^2] = \sum_{i=1}^{C}[\frac{1}{K}\mathbb{D}[\phi_i^k] + \mathbb{E}[\phi_i^k]^2]$$

Figure 5: Effect of η on SigmE Power Normalization for filtering out the large dispersion features (5a), and adjusting the angle similarity via an element-wise addition (5b). The black dotted line represents the angle similarity (cosine distribution) without any adaption.



(a) (b)

Finally, we integrate the above formulations and obtain:

$$\mathbb{E}_{\bar{\mathbf{\Phi}}_{\boldsymbol{l}}}[\mathbb{E}_{\mathbf{\Phi}_{\boldsymbol{k}}}[Cos(\bar{\mathbf{\Phi}}, \mathbf{\Phi}_{\boldsymbol{k}})]] \geq \frac{\mathbb{E}[\mathbf{\Phi}_{\boldsymbol{k}}] \cdot \mathbb{E}[\bar{\mathbf{\Phi}}_{\boldsymbol{l}}]}{\sqrt{\mathbb{E}[\|\mathbf{\Phi}_{\boldsymbol{k}}\|_2^2]} \cdot \sqrt{\mathbb{E}[\|\bar{\mathbf{\Phi}}_{\boldsymbol{l}}\|_2^2]}}$$
$$\geq \frac{\sum_{i=1}^{C}\mathbb{E}[\boldsymbol{\phi}_{\boldsymbol{i}}^{\boldsymbol{k}}]^2}{\sum_{i=1}^{C}\mathbb{D}[\phi_i^k] + \sum_{i=1}^{C}\mathbb{E}[\phi_i^k]^2}$$

B. 'Refine once' and 'Refine twice' perform similarly

We define the prototype generated by the 'Refine once' strategy as $\boldsymbol{p}_{1+2}$ and that generated by the 'Refine twice' strategy as $\boldsymbol{p}_2$ ($\boldsymbol{p}_1$ is the first refined prototype). The $\bar{\mathbf{\Phi}}$ represents the class-level representation which is K-average pooled over K support features $\mathbf{\Phi}_{\boldsymbol{k}}$. Formulations are defined as follows:

$$\boldsymbol{p}_1 = Cos(\mathbf{\Phi}_{\boldsymbol{k}}, \bar{\mathbf{\Phi}}) \cdot \mathbf{\Phi}_{\boldsymbol{k}}$$
$$\boldsymbol{p}_2 = \mathbb{D}(\mathbf{\Phi}_{\boldsymbol{k}}, p_1)^{-1} \cdot \mathbf{\Phi}_{\boldsymbol{k}}$$
$$\boldsymbol{p}_{1+2} = (Cos(\mathbf{\Phi}_{\boldsymbol{k}}, \bar{\mathbf{\Phi}}) + \mathbb{D}(\mathbf{\Phi}_{\boldsymbol{k}}, \bar{\mathbf{\Phi}})^{-1}) \cdot \mathbf{\Phi}_{\boldsymbol{k}}$$

Here, $\boldsymbol{p}_{1+2}$ and $\boldsymbol{p}_2$ are normalized prototypes, and the ratio of the their expectations is:

$$\frac{\mathbb{E}[\boldsymbol{p}_{1+2}]}{\mathbb{E}[\boldsymbol{p}_2]} = \frac{\mathbb{E}[(Cos(\mathbf{\Phi}_{\boldsymbol{k}}, \bar{\mathbf{\Phi}}) + \mathbb{D}(\mathbf{\Phi}_{\boldsymbol{k}}, \bar{\mathbf{\Phi}})^{-1}) \cdot \mathbf{\Phi}_{\boldsymbol{k}}]}{\mathbb{E}[\mathbb{D}(\mathbf{\Phi}_{\boldsymbol{k}}, \boldsymbol{p}_1)^{-1} \cdot \mathbf{\Phi}_{\boldsymbol{k}}]}$$
$$= \frac{Cos(\mathbf{\Phi}_{\boldsymbol{k}}, \bar{\mathbf{\Phi}}) + \mathbb{D}(\mathbf{\Phi}_{\boldsymbol{k}}, \bar{\mathbf{\Phi}})^{-1}}{\mathbb{D}(\mathbf{\Phi}_{\boldsymbol{k}}, \boldsymbol{p}_1)^{-1}}$$
$$= \frac{1 - \frac{1}{2}(\mathbf{\Phi}_{\boldsymbol{k}} - \bar{\mathbf{\Phi}})^2 + \frac{C}{(\mathbf{\Phi}_{\boldsymbol{k}} - \bar{\mathbf{\Phi}})^2}}{\frac{C}{(\mathbf{\Phi}_{\boldsymbol{k}} - \boldsymbol{p}_1)^2}}$$
$$= \frac{1 - \frac{1}{2}(\mathbf{\Phi}_{\boldsymbol{k}} - \bar{\mathbf{\Phi}})^2 + \frac{C}{(\mathbf{\Phi}_{\boldsymbol{k}} - \bar{\mathbf{\Phi}})^2}}{\frac{4C}{(\mathbf{\Phi}_{\boldsymbol{k}} - \bar{\mathbf{\Phi}})^4}}$$

We define $(\mathbf{\Phi_k} - \bar{\mathbf{\Phi}})^2 = x$, where $x \in [0, 4]$. We can then derive it as follows:

$$
\begin{aligned}
f(x) &= \frac{(1 - \frac{1}{2}x)x^2 + Cx}{4C} \\
&= \frac{x^2 - \frac{1}{2}x^3 + Cx}{4C} \\
\because \quad & f'(x) > 0, \quad x \in [0, 4] \\
\therefore \quad & 0 \leqslant f(x) \leqslant 1 + \frac{4}{C}
\end{aligned}
$$

Together with above formulas, we obtain:

$$
\frac{\mathbb{E}[\boldsymbol{p}_{1+2}]}{\mathbb{E}[\boldsymbol{p}_2]} \leqslant 1 + \frac{4}{C}
$$

Usually C is large enough ($C = 1024 > 10^3$) so that $\boldsymbol{p}_{1+2}$ is close to $\boldsymbol{p}_2$. As a result, 'Refine once' performs similar to 'Refine twice'.

C. The effect of η on SigmE Power Normalization

Features obtained by adapting a single similarity metric are inherently limited in their ability to capture all the intrinsic characteristics within a given class. This is due to the fact that a single metric can only be discriminative in one feature space, and therefore may not generalize well to other feature spaces. This can result in similarity bias, which significantly lowers the generalization ability of the detector, particularly when training data is limited. To address this issue, it is important to consider multiple similarity metrics. By doing so, the detector can map samples more compactly into a smaller feature space, resulting in more discriminative features that can improve the detector's overall performance.

We consider both angle distance (cosine similarity) and magnitude distance, measured by the spread of data points around the mean using the metric σ. This σ metric captures the frequency of feature occurrence and filters out rare co-occurring features, highlighting common areas where they do appear. To eliminate nuisance variability in visual features caused by intra-class variations such as scale, pose, and texture, we use Power Normalization (PN). This allows us to up-weight features that deviate less from the mean, with a hyperparameter η controlling the sharpness of the output distribution.

Figure 5a shows that increasing η results in a narrower/concentrated distribution, which helps to avoid the prototype representations from being affected by trivial variations. The adaptation effect can be seen in Figure 5b, where a larger value of η yields greater emphasis on descriptors with a large angle similarity. By re-evaluating the support features according to σ-adaptive similarity, the detector is able to prioritize the intrinsic representations that have both high angle and magnitude similarity with support features during prototype learning.

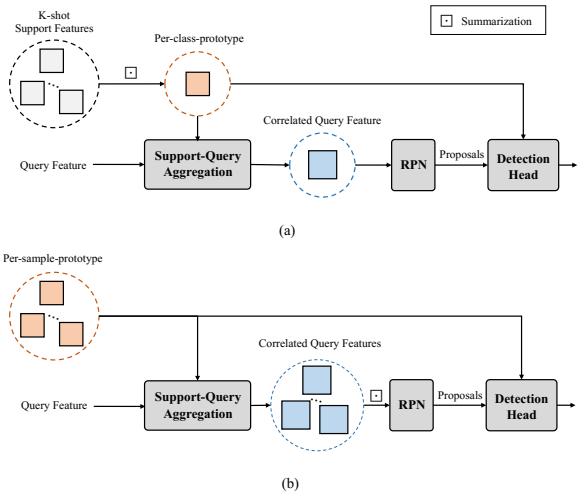


Figure 6: Per-class prototype (a) and per-sample prototype (b) explained.

D. Implementation details of K=1

When K=1, the σ-ADP performs self-refinement on the sample itself. For example, for the spatial-based relation maps, the mean point $\bar{\mathbf{\Phi}}$ of $\mathbf{\Phi_k} \in \mathbb{R}^{N \times 1 \times C}$ has the shape of $1 \times 1 \times C$. When computed with $\mathbf{\Phi_k}$ in Eq. 1 and Eq. 2, the resulting output has the size of $N \times 1 \times 1$. Subsequently, in Eq. 5, $\mathbf{\Phi_k}$ undergoes spatial re-weighting, resulting in a size of $N \times 1 \times C$ output.

E. Per-class and Per-sample Prototype-based Pipelines

Few-shot object detection involves two learning regimes: 1) a single prototype per category, and 2) an individual prototype per support sample (following strategies used in methods [24, 55]). In the first regime, a set of support features extracted from encoding network (EN) are summarized via σ-ADP to create a class-level prototype. This prototype is then cross-correlated against the query feature (extracted from EN) in the branch of Support-Query Aggregation to obtain a single correlated query feature, which is fed into RPN and the followed detection head. In the second setting, each support sample is treated as an individual prototype and aggregated separately with the query feature. The resulting set of correlated query features is averaged (channel-wisely concatenated, then decreased by the stack of FC layers) and passed to RPN. Figure 6 (a) and (b) illustrate the two pipelines. Comparison results for the 5-shot protocol on the FSOD testset for novel classes are shown below, which demonstrates that 1) neither class-level proto-

type nor support-query interactions benefit from intra-class variations, and 2) a robust class-level prototype boots the FSOD performance over per-sample prototype.

Prototype	Novel(5-shot)		
	mAP	AP_{50}	AP_{75}
Per-sample	26.7	29.7	24.7
Per-class	**29.9**	**32.7**	**27.3**

F. Applying σ-ADP to transformer-based FCT

We use FCT [16] as the baseline and incorporate σ-ADP after stage3 of the backbone network. We decouple the output of the support branch in stage3 to compute task-specific and σ-adaptive prototypes. These prototypes serve for the proposal generator and stage4&Pairwise matching, respectively. The results presented below are for split1, novel classes on PASCAL VOC dataset.

Method	Venue	1-shot	2-shot	3-shot	5-shot	10-shot
FCT	CVPR 2022	49.9	57.1	57.9	63.2	67.1
Ours+FCT		51.7	59.0	60.6	65.5	69.8

G. Visualization

G.1. Qualitative Results of Attention Maps

Visualizing the attention maps on the support images is an effective way to help understand how decoupled task-specific prototypes benefit the few-shot object detection (FSOD) task. In Figure 7, we show three types of attention maps. The first type, denoted by $\bar{\mathbf{\Phi}}'$, represents an entangled task-agnostic prototype that correlates with each support image. However, this prototype needs to balance the mismatched goals of both RPN and DH tasks, leading to activations in inaccurate regions and suboptimal solutions. The second type of attention maps, labeled as $\bar{\mathbf{\Phi}}^{\ddagger}$, correspond to spatial-wise prototypes that address the task of 'where to look' in RPN. The third type, marked as $\bar{\mathbf{\Phi}}^{\dagger}$, represents channel-wise prototypes used for the 'what to look for' task in the Detection Head (DH). Decoupling these prototypes allows them to focus on individual tasks by attending to spatial patterns and semantic clues separately, which provides more precise information for the query images. These results demonstrate that decoupling task-specific prototypes can improve the FSOD performance by providing task-specific attention maps, and that entangled prototypes are not as effective due to their need to balance the mismatched goals of multiple tasks.

G.2. Results of Detected Boxes

We visualize the detection results achieved by the proposed σ-ADP in Figure 8. It is evident that the model can accurately detect objects belonging to the novel categories, without the need for fine-tuning.

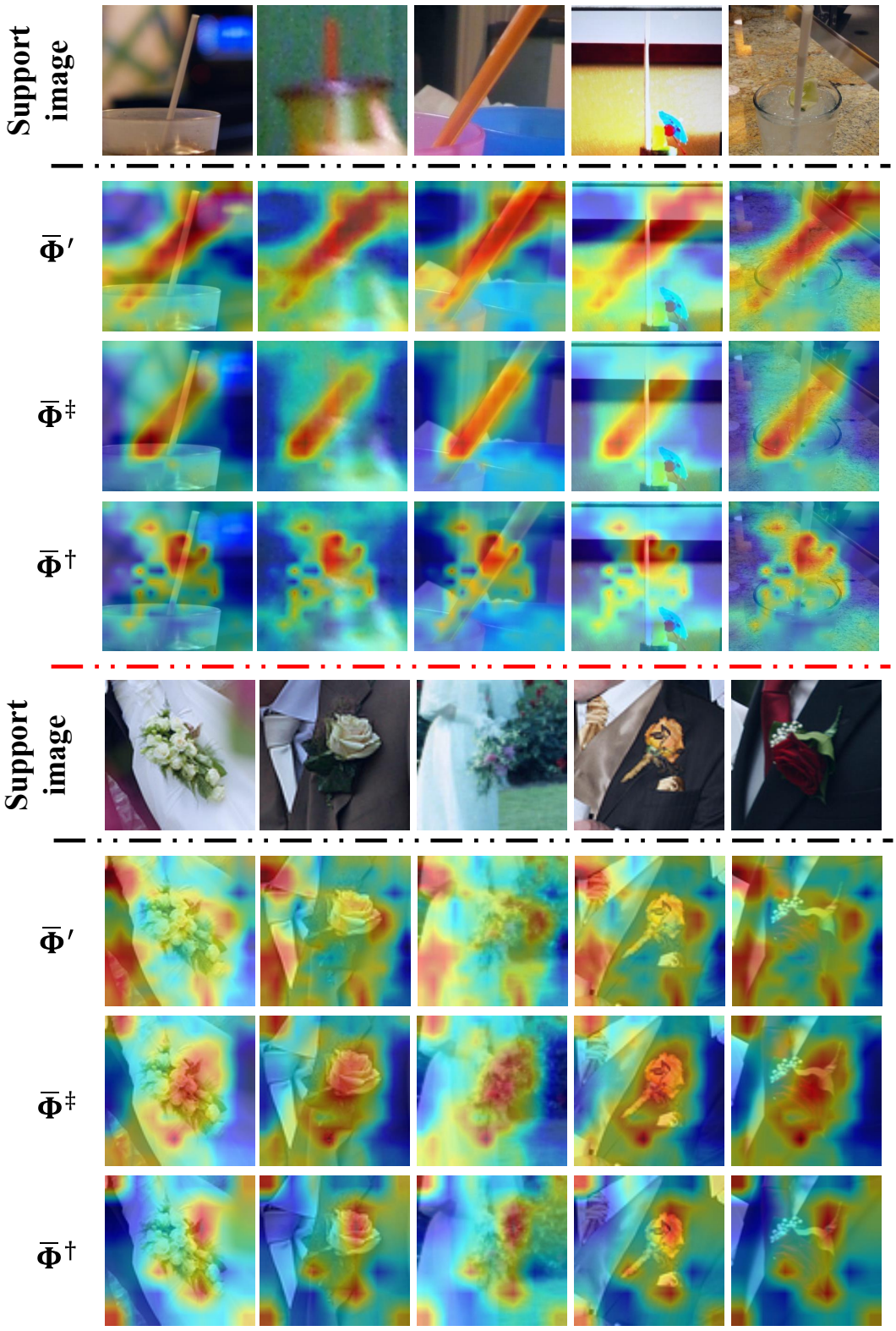


Figure 7: Attention maps on support images w.r.t. the entangled task-agnostic prototype ($\bar{\Phi}'$) and decoupled task-specific prototypes, spatial-wise $\bar{\Phi}^{\ddagger}$ for the 'where to look' task and channel-wise $\bar{\Phi}^{\dagger}$ for the 'what to look for' task. Refer to §G.1 for detailed descriptions. Zoom in to view the details.

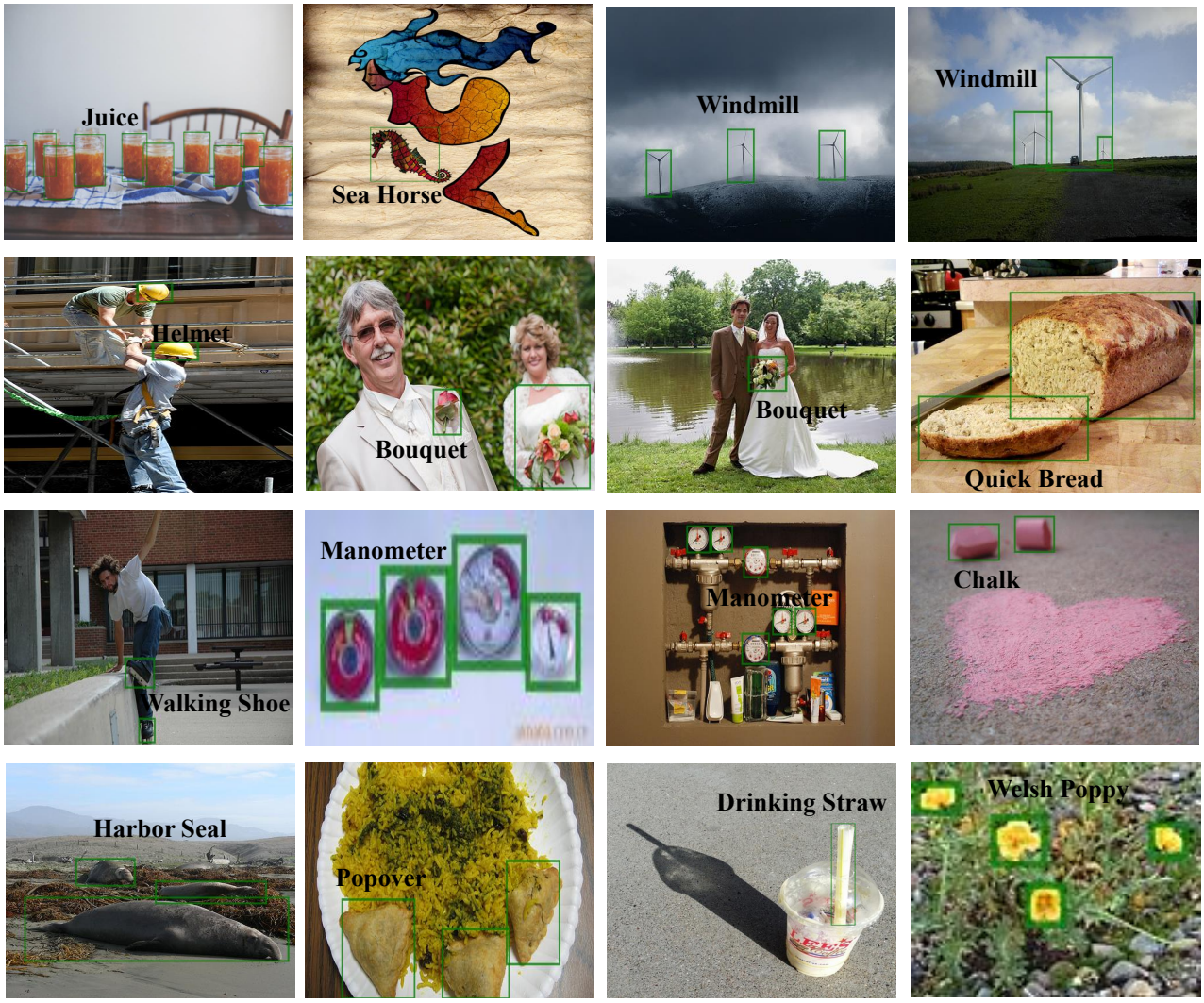


Figure 8: Detection results of σ-ADP for novel categories in the query images on the FSOD testset (Note we do not use meta fine-tuning). The green boxes indicate the detected objects and the corresponding categories are displayed as text.