Learning What Not to Segment: A New Perspective on Few-Shot Segmentation

—— Chunbo Lang, Gong Cheng, Binfei Tu, CVPR 2022 oral
Few-shot Segmentation

- **Semantic Segmentation**
  - Segment the targets of semantic categories (seen)
  - Required a large amount of labeled data
  - Can not handle the unseen categories

- **Few-shot Segmentation:**
  - Segment the targets of a specific semantic category (unseen)
  - Leveraging few labeled data

![Diagram showing labeled data, raw data, and prediction](image-url)
Few-shot Segmentation

- Train set $D_{\text{train}}$ with categories $C_{\text{base}}$, test set $D_{\text{test}}$ with categories $C_{\text{novel}}$
  - $C_{\text{base}} \cap C_{\text{novel}} = \emptyset$
- Input construction: episode $= \{S, Q\}^N$
  - Support set $S = \{(x_i^s, m_i^s)\}_{i=1}^K$
  - Query set $Q = \{(x_i^q, m_i^q)\}$
  - The categories of $S$ and $Q$ are the same
- Prediction $= f(Q \mid S)$
Motivation

(a) Conventional approaches to train the FSS model introduce a bias towards the seen classes rather than being ideally class-agnostic.
- sensitive to the quality of support images

(b) Proposed BAM
- Base learner identify confusible regions in the query image
- Base learner provide highly reliable segmentation results
**Method - Base learner**

- **Base learner**: PSPNet trained on $D_{\text{base}}$

  \[
  f_b^q = \mathcal{F}_{\text{conv}}(\mathcal{E}(x^q)) \in \mathbb{R}^{c \times h \times w},
  \]

  \[
  p_b = \text{softmax} \left( \mathcal{D}_b(f_b^q) \right) \in \mathbb{R}^{(1+N_b) \times H \times W}
  \]

  \[
  \mathcal{L}_{\text{base}} = \frac{1}{n_{\text{bs}}} \sum_{i=1}^{n_{\text{bs}}} \text{CE} \left( p_{b,i}, m_{b,i}^q \right),
  \]
Method - Meta learner

- **Meta learner**: segment the object in query image under the guidance of support images

\[ v_s = \mathcal{F}_{pool}(f_m \odot I(m^s)) \in \mathbb{R}^c \]

\[ p_m = \text{softmax} \left( \mathcal{D}_m \left( \mathcal{F}_{guidance}(v_s, f_m^q) \right) \right) \in \mathbb{R}^{2 \times H \times W} \]

- \( \mathcal{D}_m \): ASPP
**Method - Ensemble**

- Integrate the prediction of base learner $p_b^f = \sum_{i=1}^{N_b} p_b^i$
- Obtain Gram matrices $G^{s/q}$

\[
A_s = F_{\text{reshape}}(f_{\text{low}}^s) \in \mathbb{R}^{C_1 \times N}
\]

\[
G^s = A_s A_s^T \in \mathbb{R}^{C_1 \times C_1}
\]

\[
\psi = \|G^s - G^q\|_F
\]
Method - K-shot

- **Conventional method**: average the support feature vectors
  - Suboptimal: low quality support images contains less guidance

- **Weighted fusion**: a smaller value of $\psi_i$ indicates a greater contribution
  - $\psi_i = \|G_i^s - G^q\|_F$
  - Sort $\psi_k$, $\psi_k = \text{concate}(\psi_i)$
  - $\eta = \text{softmax} \left( w_2^T \text{ReLU} (w_1^T \psi_t) \right) \in \mathbb{R}^K$
  - $\eta$ reverts to the original order
  - Fusing support feature vectors and $\psi$ with $\eta$
### 实验

<table>
<thead>
<tr>
<th>Backbone</th>
<th>Method</th>
<th>1-shot</th>
<th>5-shot</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fold-0</td>
<td>Fold-1</td>
<td>Fold-2</td>
</tr>
<tr>
<td>SG-One</td>
<td>40.20</td>
<td>58.40</td>
<td>48.40</td>
</tr>
<tr>
<td>PANet</td>
<td>42.30</td>
<td>58.00</td>
<td>51.10</td>
</tr>
<tr>
<td>F WB</td>
<td>47.00</td>
<td>59.60</td>
<td>52.60</td>
</tr>
<tr>
<td>CRNet</td>
<td>56.90</td>
<td>68.20</td>
<td>54.40</td>
</tr>
<tr>
<td>PFPNet</td>
<td>59.60</td>
<td>65.70</td>
<td>59.60</td>
</tr>
<tr>
<td>Baseline</td>
<td>63.18</td>
<td>70.77</td>
<td>66.14</td>
</tr>
<tr>
<td>BAM (ours)</td>
<td>65.68</td>
<td>71.41</td>
<td>65.56</td>
</tr>
<tr>
<td>CA Net</td>
<td>52.50</td>
<td>65.90</td>
<td>51.30</td>
</tr>
<tr>
<td>PGNnet</td>
<td>56.00</td>
<td>66.90</td>
<td>50.60</td>
</tr>
<tr>
<td>CRNet</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>PPFPnnet</td>
<td>48.58</td>
<td>60.58</td>
<td>55.71</td>
</tr>
<tr>
<td>Baseline</td>
<td>64.30</td>
<td>70.70</td>
<td>60.30</td>
</tr>
<tr>
<td>BAM (ours)</td>
<td>68.97</td>
<td>73.59</td>
<td>67.55</td>
</tr>
</tbody>
</table>

### MS COCO

![MS COCO Diagram](attachment:ms_coco_diagram.png)
消融实验

<table>
<thead>
<tr>
<th>PT</th>
<th>$\mathcal{L}_{\text{meta}}$</th>
<th>Init.</th>
<th>$\psi$</th>
<th>mIoU</th>
<th>FB-IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>√</td>
<td>√</td>
<td>√</td>
<td>57.61</td>
<td>59.12</td>
<td>70.75</td>
</tr>
<tr>
<td>√</td>
<td>√</td>
<td>√</td>
<td>59.76</td>
<td>71.94</td>
<td>72.79</td>
</tr>
<tr>
<td>√</td>
<td>√</td>
<td>√</td>
<td>62.49</td>
<td>75.43</td>
<td>77.26</td>
</tr>
<tr>
<td>√</td>
<td>√</td>
<td>√</td>
<td>64.41</td>
<td>77.06</td>
<td>77.26</td>
</tr>
</tbody>
</table>

$$p_f^0 = \mathcal{F}_{\text{ensemble}}(\mathcal{F}_\psi(p_m^0), p_b^0)$$

$$\mathcal{F}(p_m^0 \oplus \psi)$$

nn.Parameter(torch.tensor([[1.0],[0.0]]))