**Δ-encoder: An effective sample synthesis method for few-shot object recognition**

Eli Schwartz*1, Leonid Karlinsky*4, Joseph Shtok5, Sivan Harary1, Mattias Marder1, Abhishek Kumar2, Rogerio Feris1, Raja Giryes5, Alex M. Bronstein3

1 -IBM Research AI; 2 - School of Electrical Engineering, Tel-Aviv University, Tel-Aviv, Israel; 3 - Department of Computer Science, Technion, Haifa, Israel

---

**Few-shot classification problem:** presented with a few samples of $K$ novel categories, produce a classifier for the $K$ categories.  

**Generative approach:** synthesize new samples from the few available examples to improve the classifier training.

Our main idea

1. Learn to recover the relative transformation (the delta) between pairs of samples from the same category.  
2. Sample from a distribution of such transformations within the set of known categories.  
3. Learn to re-apply the transformations to examples of novel categories to produce additional samples thereof.

---

Training data – 1 sample per category

Query image

Classifier

Recovered target

Data synthesis – thousands of samples

Classifer training

Training

Synthesis

Target

Reference

Encoder

Delta $\Delta$

$Z$

Decoder

$\Delta$

Target

Reference

Encoder

Sampled target

Sampled reference

$Z$

Decoder

Sampled data

Synthesized new class example

New class reference

---

**A variant of an auto-encoder operating in feature space.**  
Network learns to encode the $\Delta$ between reference and target images.  
This $\Delta$ is used to recover the target image as a (non-linear) combination of the reference and $\Delta$.

**At test time encoded $\Delta$s are sampled from random training image pairs belonging to the same category.**  
Sampled $\Delta$s are used to create samples for the new categories by combining them with the new category reference examples.

---

Visualization of two-way one-shot classification trained on synthesized examples. Correctly classified images are framed in magenta (golden retriever) and yellow (African wild dog). The only two images seen at training time and used for sample synthesis are framed in blue. Note the non-trivial relative arrangement of examples belonging to different classes handled successfully by our approach. Plotted using t-SNE applied to image features.

---

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Previous SOA 1-shot/5-shot</th>
<th>Ours 1-shot/5-shot</th>
</tr>
</thead>
<tbody>
<tr>
<td>minImageNet</td>
<td>58.5 / 76.9</td>
<td>59.9 / 69.7</td>
</tr>
<tr>
<td>CIFAR-100</td>
<td>63.4 / 78.4</td>
<td>66.7 / 79.8</td>
</tr>
<tr>
<td>Caltech-256</td>
<td>63.8 / 80.5</td>
<td>73.2 / 83.6</td>
</tr>
<tr>
<td>CUB</td>
<td>69.6 / 84.1</td>
<td>69.8 / 82.6</td>
</tr>
</tbody>
</table>

Generated samples for 12-way one-shot.  
The red crosses mark the original 12 examples (one per class). The generated points are colored according to their class.

---

Synthesized samples visualization. The single example image is framed in blue. All other images represent the synthesized samples visualized using their nearest "real image" neighbors in the feature space. The two-dimensional embedding was produced by t-SNE.

---

1-shot/5-shot 5-way accuracy using features externally pre-trained on a non-overlapping set of ImageNet categories.

<table>
<thead>
<tr>
<th>Method</th>
<th>AWA2</th>
<th>APY</th>
<th>SUN</th>
<th>CUB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nearest neighbor (baseline)</td>
<td>65.9</td>
<td>75.9</td>
<td>72.7</td>
<td>58.7</td>
</tr>
<tr>
<td>Prototypical Networks</td>
<td>80.8</td>
<td>95.3</td>
<td>90.1</td>
<td>74.7</td>
</tr>
<tr>
<td>Δ-encoder</td>
<td>90.5</td>
<td>96.4</td>
<td>93.4</td>
<td>93.0</td>
</tr>
</tbody>
</table>

---

IBM Research AI

AI

TEL AVIV UNIVERSITY

© 2018 IBM Corporation. All rights reserved.