Disentangling 3D Prototypical Networks for Few-Shot Concept Learning

Mihir Prabhudesai*, **Shamit Lal***, **Darshan Patil***†, **Hsiao-Yu Tung**,
Adam W Harley, Katerina Fragkiadaki
Carnegie Mellon University
{mprabhud, shamitl, dapatil, htung, aharley, katef}@cs.cmu.edu

1 Introduction

Humans can learn new concepts from just one or a few samples. Consider the example in Figure 1. Assuming there is a person who has no prior knowledge about *blue* and *carrot*, by showing this person an image of a blue carrot and telling him "this is a *carrot* with *blue* color", the person can easily generalize from this example to (1) recognizing the color *blue* on different objects, (2) using the newly learned concepts to answer questions regarding the visual scene. Motivated by this, we explore computational models that can achieve these generalizations for visual concept learning.

We propose disentangling 3D prototypical networks (D3DP-Nets), a model that learns to disentangle RGB-D images into objects, their 3D locations, sizes, 3D shapes and styles, as shown in Figure 2. Our model disentangles objects into different attributes through a self-supervised view prediction task. Specifically, D3DP-Nets uses differentiable unprojection and rendering operations to go back and forth between the input RGB-D (2.5D) image and a 3D scene feature map. From the scene feature map, our model learns to detect objects and disentangles each object into a 3D shape code and an 1D style code through a shape/style disentangling autoencoder. We use adaptive instance normalization layers (Huang & Belongie, 2017) to encourage shape/style disentanglement within each object. Our key intuition is to represent objects and their shapes in terms of **3D feature representations disentangled from style variability** so that the model can correspond objects with similar shape by explicitly rotating and scaling their 3D shape representations during matching.

With the disentangled representations, D3DP-Nets can recognize new concepts regarding object shapes, styles and spatial arrangements from a few human-supplied labels by training concept classifiers only on the relevant feature subspace and ignore irrelevant visual features. This also allows D3DP-Nets to recognize novel attribute compositions not present in the training data.

We test D3DP-Nets in few-shot concept learning and visual question answering (VQA). We show that shape/style classifiers trained on our disentangled representation outperform classifiers operating on other (Eslami et al., 2018; Huang et al., 2018) entangled/disentangled representations, which use 2D/2.5D representations. We also show that the VQA modular network that incorporates our concept classifiers shows improved generalization over the state-of-the-art (Mao et al., 2019) with dramatically fewer examples. The main contribution of this paper is to identify the importance of using disentangled 3D feature representations for few-shot concept learning. Due to lack of space please find our related work section in the supplementary.

2 Disentangling 3D Prototypical Networks (D3DP-Nets)

The architecture of D3DP-Nets is illustrated in Figure 2. D3DP-Nets consists of two main components: (a) an image-to-scene encoder-decoder, and (b) an object shape/style disentanglement encoder-decoder. Next, we describe these components in detail.

Project page: https://mihirp1998.github.io/project_pages/d3dp/

*Equal contribution

†Work done while in Carnegie Mellon University

NeurIPS 2020 Workshop on Object Representations for Learning and Reasoning, Vancouver, Canada.

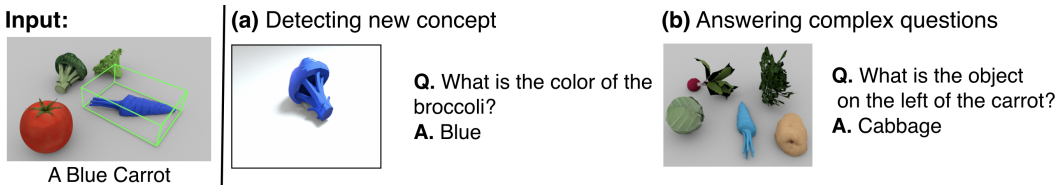

Figure 1: Given a single image-language example regarding new concepts (e.g., blue and carrot). On the right, we show tasks the proposed model can achieve using this grounding.

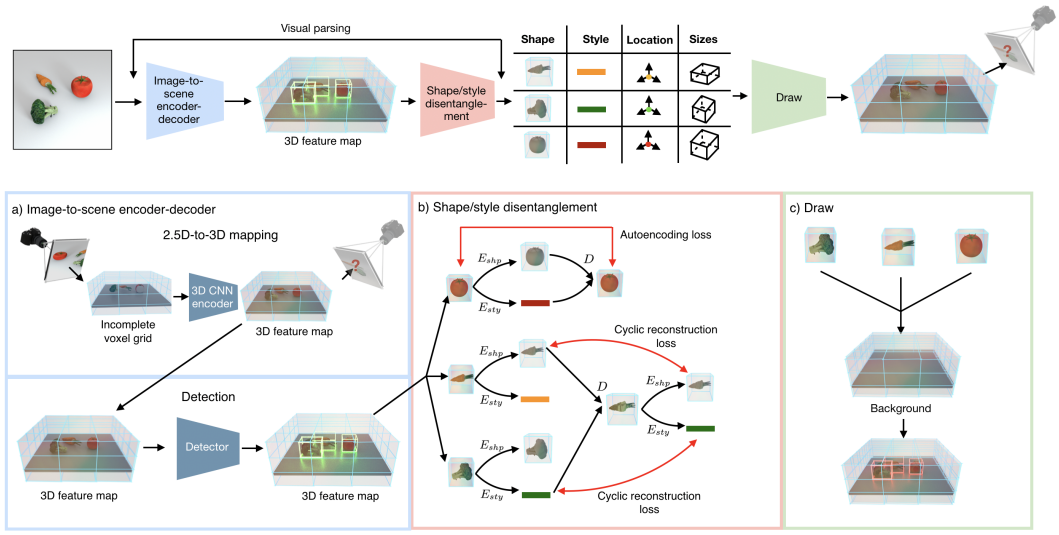

Figure 2: **Architecture for disentangling 3D prototypical networks (D3DP-Nets). (a)** Given multi-view posed RGB-D images of scenes as input during training, our model learns to map a single RGB-D image to a completed scene 3D feature map at test time, by training for view prediction. From the completed 3D scene feature map, our model learns to detect objects from the scene. **(b)** In each 3D object box, we apply a shape-style disentanglement autoencoder that disentangles the object-centric feature map to a 3D (feature) shape code and a 1D style code. **(c)** Our model can compose the disentangled representations to generate a novel scene 3D feature map.

Image-to-scene encoder-decoder: A 2D-to-3D scene differentiable encoder E^{sc} maps an input RGB-D image to a 3D feature map $\mathbf{M} \in \mathbb{R}^{w \times h \times d \times c}$, of the scene, where w, h, d, c denote width, height, depth and number of channels, respectively. Every (x, y, z) grid location in $\mathbf{M}$ describes the semantic and geometric properties of a corresponding 3D physical location in the 3D world scene. When the input to D3DP-Nets is a sequence of images as opposed to a single image, each image I_t in the sequence is encoded to a corresponding 3D per frame map $\mathbf{M}_t$, warped to the same coordinate frame as $\mathbf{M}_0$ and then averaged with the map built thus far. D3DP-Nets are self-supervised by view prediction, predicting RGB images and occupancy grids for query viewpoints.

$$\mathcal{L}^{view-pred} = \|\mathrm{D}^{\mathrm{sc}}\left(\mathrm{rotate}(\mathbf{M}, v_q)\right) - I_q\|_1 + \log(1 + \exp(-O_q \cdot \mathrm{D}^{\mathrm{occ}}((\mathrm{rotate}(\mathbf{M}, v_q)), v_q))), \quad (1)$$

where I_q and O_q are the ground truth RGB image and occupancy map respectively, v_q is the query view, and $\mathrm{rotate}(\mathbf{M}, v_q)$ is a trilinear resampling operation that rotates the content of a 3D feature map $\mathbf{M}$ to viewpoint v_q. Occupancy labels are computed through raycasting, similar to Harley et al. (2020). We provide more details on the architecture of our model in the supplementary material.

We train a 3D object detector that takes as input the output of the scene feature map $\mathbf{M}$ and predicts 3D axis-aligned bounding boxes, similar to Harley et al. (2020). This is supervised from ground-truth 3D bounding boxes without class labels.

Object shape/style disentanglement Given a set of 3D object boxes $\{b^o | o = 1 \cdots |\mathcal{O}|\}$ where $\mathcal{O}$ is the set of objects in the scene, D3DP-Nets obtain corresponding object feature maps $\mathbf{M}^o = \mathrm{crop}(\mathbf{M}, b^o)$ using the 3D bounding box coordinates b^o. Each object feature map is resized to a fixed

resolution of $16 \times 16 \times 16$, and fed to an object-centric autoencoder whose encoding modules predict a 4D shape code $z^o_{\text{shp}} = \text{E}_{shp}(\mathbf{M}^o) \in \mathbb{R}^{w \times h \times d \times c}$ and a 1D style code $z^o_{\text{sty}} = \text{E}_{sty}(\mathbf{M}^o) \in \mathbb{R}^c$. A decoder D composes the two using adaptive instance normalization (AIN) layers (Huang & Belongie, 2017): $AIN(z, \gamma, \beta) = \gamma \left(\frac{z - \mu(z)}{\sigma(z)} \right) + \beta$, where z is obtained by a 3D convolution on z_{shp}, μ and σ are the channel-wise mean and standard deviation of z, and β and γ are extracted using single-layer perceptrons from z_{sty}. The object encoders and decoders are trained with an autoencoding objective and a cycle-consistency objective which ensure that the shape and style code remain consistent after composing, decoding and encoding again (see Figure 2 (b)):

$$\mathcal{L}^{dis} = \frac{1}{|\mathcal{O}|} \sum_{o=1}^{|\mathcal{O}|} \left(\underbrace{\|\mathbf{M}^o - \text{D}(\text{E}_{shp}(\mathbf{M}^o), \text{E}_{sty}(\mathbf{M}^o))\|_2}_{\text{autoencoding loss}} + \underbrace{\sum\nolimits_{i \in \mathcal{O} \setminus o} \mathcal{L}^{c-shp}(\mathbf{M}^o, \mathbf{M}^i) + \mathcal{L}^{c-sty}(\mathbf{M}^o, \mathbf{M}^i)}_{\text{cycle-consistency loss}} \right), \tag{2}$$

where $\mathcal{L}^{c-shp}(\mathbf{M}^o, \mathbf{M}^i) = \|\text{E}_{shp}(\mathbf{M}^o) - \text{E}_{shp}(\text{D}(\text{E}_{shp}(\mathbf{M}^o), \text{E}_{sty}(\mathbf{M}^i)))\|_2$ is the shape consistency loss and $\mathcal{L}^{c-sty}(\mathbf{M}^o, \mathbf{M}^i) = \|\text{E}_{sty}(\mathbf{M}^o) - \text{E}_{sty}(\text{D}(\text{E}_{shp}(\mathbf{M}^i), \text{E}_{sty}(\mathbf{M}^o)))\|_2$ is the style consistency loss.

We further include a view prediction loss on the synthesized scene feature map $\bar{\mathbf{M}}$, which is composed by replacing each object feature map $\mathbf{M}^o$ with its re-synthesized version $\text{D}(z^o_{\text{shp}}, z^o_{\text{sty}})$, resized to the original object size, as shown in Figure 2(c). The view prediction reads: $\mathcal{L}^{view-pred-synth} = \|\text{D}^{\text{sc}}\left(\text{rotate}(\bar{\mathbf{M}}, v^{t+1})\right) - I_{t+1}\|_1$. The total unsupervised optimization loss for D3DP-Nets reads:

$$\mathcal{L}^{uns} = \mathcal{L}^{view-pred} + \mathcal{L}^{view-pred-synth} + \mathcal{L}^{dis}. \tag{3}$$

3D disentangled prototype learning Given a set of human annotations in the form of labels for object attributes (shape, color, material, size), our model computes prototypes for each concept (e.g. "red" or "sphere") in an attribute, using only the relevant feature embeddings. For example, object category prototypes are learned on top of shape codes, and material and color prototypes are learned on top of style codes. In order to classify a new object example, we compute the nearest neighbors between the inferred shape and style embeddings from the D3DP-Nets with the prototypes in the prototype dictionary. This non-parametric classification method allows us to detect objects even from a single example, and also improves when more labels are provided by co-training the underlying feature representation space as in Snell et al. (2017). To compute the distance between an embedding x and a prototype y, we define the following rotation- aware distance metric: $\langle x, y \rangle_R = \max_{r \in \mathcal{R}} \langle \text{Rotate}(x, r), y \rangle$ if x, y are 4D. $\langle x, y \rangle_R = \langle x, y \rangle$, if x and y are 1D, where $\langle x, y \rangle$ means the cosine similarity between x and y and $\text{Rotate}(x, r)$ explicitly rotates the content in 3D feature map x with angle r through trilinear interpolation. We exhaustively search across rotations $\mathcal{R}$, in a parallel manner, considering increments of 10° along the vertical axis. Our model initializes the concept prototypes by averaging the feature codes of the labelled instances. When annotations for concepts are provided, we can jointly finetune our prototypes and neural modules (as well as D3DP-Net weights) using a binary cross entropy loss, whose logits are inner products between neural embeddings and prototypes.

3 Experiments

Few-shot object shape and style category learning We evaluate D3DP-Nets in its ability to classify shape and style concepts from few annotated examples on CLEVR (Johnson et al., 2017) datasets: We train D3DP-Nets self-supervisedly on posed multiview images in the dataset and learn the prototypes for each concept category. During training, we consider a single labeled instance for each shape and style category present in the dataset. During testing, we consider a pool of 1000 object instances.

In this experiment, we use ground-truth bounding boxes. We compare D3DP-Nets with 2D, 2.5D and 3D versions of Prototypical Networks (Snell et al., 2017) that similarly classify object image crops by comparing object feature embeddings to prototype embeddings. Specifically, we learn prototypical embeddings over the visual representations produced by the following baselines: (i) *2D MUNIT* (Huang et al., 2018) which disentangles shape and style within each object-centric 2D image RGB patch using the 2D equivalent of the shape-style disentanglement architecture of our model, which learns using an autoencoding objective (ii) *2.5D MUNIT* an extension of 2D MUNIT which uses concatenated RGB and depth as input. (iii) *3DP-Nets*, a version of D3DP-Nets where object shape-style disentanglement is omitted (iv) Generative Query Network *GQN* of Eslami et al. (2016) which encodes multiview images of a scene and camera poses into a 2D feature map and is trained using cross-view prediction, similar to our model. All baselines are trained with the same unlabeled multiview image set as our method. All

	Shape	Style
D3DP-Net	**0.70**	**0.61**
3DP-Net	0.57	0.09
2D MUNIT	0.47	0.41
2.5D MUNIT	0.55	0.46
GQN	0.45	0.11

Table 1: 1-shot classification accuracy on shape/style codes

models classify each test image into a shape, and style category. Few-shot concept classification results are shown in Table 1. D3DP-Nets outperforms all four baselines.

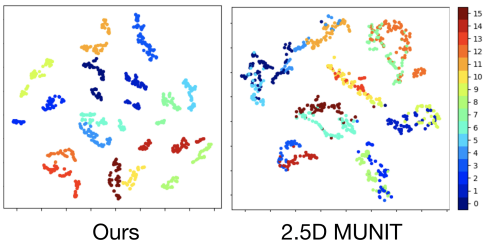

Figure 3: t-SNE visualization on styles codes.

Few-shot visual question answering We integrate concept detectors built on the D3DP-Nets representation into modular neural networks for visual question answering, in which a question about an image is mapped to a computational graph over a small number of reusable neural modules including object category detectors, style detectors and spatial expression detectors. Specifically, we build upon the recent Neuro-Symbolic Concept Learner (NSCL) (Mao et al., 2019). For example, in the question *"How many yellow objects are there?"*, the model first uses the color classifier to predict for all objects the probability that they are yellow, and then uses the resulting probability map to give an answer. NSCL learns 1D prototypes for object shape, color and material categories and classifies objects to labels using nearest neighbors to these prototypes. In our D3DP-Nets-VQA architecture, we have 3D instead of 2D object proposals, and disentangled 3D shape and 1D color/material and spatial relationship prototypes instead.

We compare D3DP-VQA against the following models: i) *NSCL-2D* the state of the art model of Mao et al. (2019) that uses a ResNet-34 pretrained on ImageNet as input feature representations ii) *NSCL-2.5D*, in which the object visual representations for shape/color/material are computed over RGB and depth concatenated object patches as opposed to RGB alone. This model is pretrained with a autoencoding loss iii) *NSCL-2.5D-disentangle* that uses disentangled object representations generated by our 2.5D MUNIT disentangling model, iv) *D3DP without 3D shape prototypes*, a version of D3DP-Nets that replaces the 3-dimensional shape codes with 1D ones obtained by spatial pooling v) *D3DP without disentanglement*, that learns prototypes for shape, color and material on top of entangled 3D tensors. We consider the same supervision for our model and baselines in the form of densely annotated scenes with object attributes and 3D object boxes. We use ground-truth neural programs so as to not confound the results with the performance of a learned parser. More details on the VQA experimental setup and additional ablative experiments are included in the supplementary file.

VQA performance results are shown in Table 2. We evaluate by varying the number of training scenes from 10 to 250. To test our model's one shot generalization ability on questions about object categories it had not seen in the original training set, we introduce a new test set consisting of only novel objects. We generate a test set of 500 scenes in the CLEVR environment with three new objects: "cheese", "garlic", and "pepper" and introduce them to our model and baselines using only one example image of each, associated with its shape category label. The results described in Table 2 indicate that our model is able to maintain its ability to answer questions even when seeing completely novel objects and with very few training examples. The SOTA 2D model outperforms our model on the in domain test set because it is able to exploit **pretraining on ImageNet**, which our models are unable to do. However, our model is able to adapt much better than both the 2D and 2.5D baselines when operating at the extremely low data regime or the one shot generalization setting.

VQA Model	In domain test set					One shot test set				
	Number of Training Examples					Number of Training Examples				
	10	25	50	100	250	10	25	50	100	250
D3DP	**0.809**	0.872	0.902	0.923	0.939	**0.775**	**0.836**	**0.834**	**0.828**	**0.845**
D3DP without 3D shape prototypes	0.798	0.858	0.538	0.905	0.932	0.410	0.410	0.517	0.745	0.771
D3DP without shape/style disentanglement	0.458	0.407	0.616	0.806	0.788	0.457	0.402	0.616	0.807	0.792
NSCL-2D (Mao et al., 2019)	0.733	**0.927**	**0.959**	**0.978**	**0.990**	0.594	0.708	0.703	0.789	0.743
NSCL-2D Mao et al. (2019) without ImageNet pretraining	0.514	0.624	0.682	0.844	0.931	0.467	0.502	0.553	0.624	0.679
NSCL-2.5D (Mao et al., 2019)	0.594	0.737	0.828	0.881	0.925	0.528	0.633	0.651	0.633	0.633
NSCL-2.5D-disentangled (Mao et al., 2019)	0.436	0.486	0.640	0.735	0.842	0.430	0.462	0.517	0.561	0.564

Table 2: VQA performance of our model and baselines in CLEVR (Johnson et al., 2017) under a varying number of annotated training scenes.

4 Conclusion and Future Directions

We presented D3DP-Nets, a model that learns disentangled 3D representations of objects and distills them into 3D and 1D prototypes of shapes and styles using multiview RGB-D videos of static scenes. We trained classifiers of prototypical object categories, object styles, and spatial relationships, on disentangled relevant features. We showed that they generalize better than 2D representations or 2D disentangled representations, with less training data. We hope our work will stimulate interest in self-supervising 3D feature representation for 3D visual recognition and question answering in domains with few human labels.

References

S. M. Ali Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, Koray Kavukcuoglu, and Geoffrey E. Hinton. Attend, infer, repeat: Fast scene understanding with generative models. *CoRR*, abs/1603.08575, 2016. URL http://arxiv.org/abs/1603.08575.

S. M. Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S. Morcos, Marta Garnelo, Avraham Ruderman, Andrei A. Rusu, Ivo Danihelka, Karol Gregor, David P. Reichert, Lars Buesing, Theophane Weber, Oriol Vinyals, Dan Rosenbaum, Neil Rabinowitz, Helen King, Chloe Hillier, Matt Botvinick, Daan Wierstra, Koray Kavukcuoglu, and Demis Hassabis. Neural scene representation and rendering. *Science*, 360(6394): 1204–1210, 2018. ISSN 0036-8075. doi: 10.1126/science.aar6170.

Adam W Harley, Fangyu Li, Shrinidhi K Lakshmikanth, Xian Zhou, Hsiao-Yu Fish Tung, and Katerina Fragkiadaki. Learning from unlabelled videos with contrastive predictive neural 3d mapping. In *ICLR*, 2020.

Xun Huang and Serge J. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. *CoRR*, abs/1703.06868, 2017. URL http://arxiv.org/abs/1703.06868.

Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2901–2910, 2017.

Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, and Jiajun Wu. The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences From Natural Supervision. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJgMlhRctm.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 4077–4087. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/6996-prototypical-networks-for-few-shot-learning.pdf.