

What Matters For Meta-Learning Vision Regression Tasks?

Ning Gao^{1,2} Hanna Ziesche¹ Ngo Anh Vien¹ Michael Volpp² Gerhard Neumann²
¹Bosch Center for Artificial Intelligence ²Autonomous Learning Robots, KIT
 {ning.gao, hanna.ziesche}@de.bosch.com anhvien.ngo@bosch.com
 {michael.volpp, gerhard.neumann}@kit.edu

Abstract

Meta-learning is widely used in few-shot classification and function regression due to its ability to quickly adapt to unseen tasks. However, it has not yet been well explored on regression tasks with high dimensional inputs such as images. This paper makes two main contributions that help understand this barely explored area. First, we design two new types of cross-category level vision regression tasks, namely object discovery and pose estimation of unprecedented complexity in the meta-learning domain for computer vision. To this end, we (i) exhaustively evaluate common meta-learning techniques on these tasks, and (ii) quantitatively analyze the effect of various deep learning techniques commonly used in recent meta-learning algorithms in order to strengthen the generalization capability: data augmentation, domain randomization, task augmentation and meta-regularization. Finally, we (iii) provide some insights and practical recommendations for training meta-learning algorithms on vision regression tasks. Second, we propose the addition of functional contrastive learning (FCL) over the task representations in Conditional Neural Processes (CNPs) and train in an end-to-end fashion. The experimental results show that the results of prior work are misleading as a consequence of a poor choice of the loss function as well as too small meta-training sets. Specifically, we find that CNPs outperform MAML on most tasks without fine-tuning. Furthermore, we observe that naive task augmentation without a tailored design results in underfitting.

1. Introduction

Humans are able to rapidly learn the fundamentals of new tasks within minutes of experience based on prior knowledge. For instance, humans can classify novel objects by capturing the distinguishable properties (e.g., textures, shapes and scales) from only a few examples. Meta-learning is proposed to learn relevant knowledge from various tasks and generalize to unseen tasks with only a

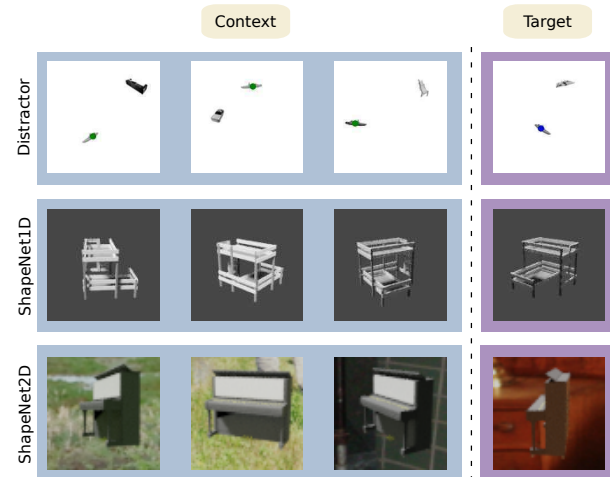


Figure 1. Meta-learning vision regression tasks are designed to i) identify the queried object from context and predict its position for target images (Distractor), ii) identify the object’s canonical pose from context and predict the 1D rotation relative to the canonical pose for target images (ShapeNet1D), iii) predict the 2D rotation w.r.t. the canonical pose with random background (ShapeNet2D). Predictions are performed on unseen objects.

few samples. Of the various meta-learning algorithms, MAML-based models [7, 8, 29, 47] and Neural Processes (NPs) [10, 11, 16, 20] are two variants which are receiving increasing attention in the recent years. Both algorithms try to learn good prior knowledge from related tasks without expanding the learned parameters or sacrificing efficiency at inference. While these methods have shown promising results in many domains, such as few-shot classification [29, 33, 35, 36, 38] and hyperparameter optimization [9, 40, 43], an extensive study on meta-learning vision regression tasks has not yet been conducted. This is in particular true for NPs which have mostly been investigated on tasks with low-dimensional input such as function regression or pixel-wise completion [22, 27, 30, 42].

In this paper, we make two major contributions to the largely unexplored area of meta-learning on high-

dimensional input tasks. On the algorithmic level, inspired by SimCLR [3], we propose an improvement to NPs by employing contrastive learning at the functional space (FCL) and still train the model in an end-to-end fashion. On the experimental side, we propose two application datasets, object discovery and pose estimation, which are based on high-dimensional inputs and require the meta-learning models to learn and reason at an image level.

For the first application we create a regression task called ‘‘Distractor’’ (see Fig. 1), where each image contains two objects, the queried object and a distractor object, placed at random positions. The goal of this task is to identify the queried object and predict its position in the image plane. Unlike previous tasks such as image completion, where each pixel is considered as an independent input, our task requires the model to learn a high-level representation from the entire image. The second application (i.e., pose estimation) is inspired by prior work [28, 31, 45, 46] on the Pascal1D dataset. As this dataset shows limited object variations and features only 1D rotation around the azimuth axis, we generate two new datasets with increasing task diversity, e.g., by introducing random background, cross-categorical object variations and 2D rotation. Since the background is generated from real-world images instead of blank as in prior work, our datasets significantly increase the task difficulty and allow us to perform a thorough investigation of the performance for the considered meta-learning approaches. Examples of our datasets are shown in Fig. 1 where i) ShapeNet1D contains 1D rotations as in Pascal1D, however with larger object variations and ii) ShapeNet2D features 2D rotation and random background.

For both applications, we evaluate the performance on novel objects at both **intra-category** (IC) and **cross-category** (CC) levels. The results on Distractor show that our proposed algorithmic improvements significantly increase the performance, indicating our methods can enhance the task expressivity. The results on pose estimation demonstrate that meta-learning can successfully be applied to predict poses of unknown objects, which has a huge potential in robotic grasping and virtual/augmented reality (VR/AR).

Prior work [28, 31, 45, 46] on Pascal1D also demonstrates that meta-learning algorithms suffer from overfitting, especially with limited training data. Our work analyzes the effect of different techniques commonly adopted in recent meta-learning methods (i.e., data augmentation, task augmentation, regularization and domain randomization) on aforementioned datasets. We empirically find that the meta-learning algorithms employed in our work ultimately lead to overfitting regardless of dataset size for both applications. Moreover, our work shows that the results in prior work [31, 46], where MAML typically performs best for such tasks, are misleading. In particular, we find Condi-

tional Neural Processes (CNPs) [10] are more flexible and efficient than MAML in the investigated pose regression tasks. Additionally, we find that MAML [7] suffers from underfitting especially on large-scale datasets and depends heavily on hyperparameter tuning.

The primary contributions of this work can be summarized as follows: (1) We investigate meta-learning algorithms on vision regression tasks and demonstrate their ability to tackle structured problems. (2) We propose functional contrastive learning on the task representation of CNPs and thereby improve its expressivity. (3) We quantitatively analyze various deep learning techniques to alleviate meta overfitting. Our results rectify misleading conceptions from prior work, e.g., that MAML performs best for such tasks. We also present insights and practical recommendations on designing and implementing meta-learning algorithms on vision regression tasks.

2. Related Work

Meta-Learning. In meta-learning, also known as *learning to learn*, a learning agent gains meta knowledge from previous learning episodes or different domains and then uses this acquired knowledge to improve the learning on future tasks [18]. MAML is an optimization-based meta-learning method and represents the meta knowledge as the model parameters, where learning good initial parameters can enable quick adaptation to new tasks with only few update steps on a small number of samples [7]. Different from MAML, Neural Processes (NPs) constitute a class of neural latent variable models and interpret meta-learning as conditional few-shot function regression [11]. Similar to Gaussian Processes, NPs model distributions over functions conditioned on contexts [11, 15, 20]. Meta-learning algorithms have been applied successfully in low-dimensional function regression [10, 11, 20, 42], image completion [16, 22, 27], few-shot classification [29, 33, 35, 36, 38, 41], reinforcement learning [13, 17, 32, 47, 48], and neural architecture search (NAS) [5, 23, 26, 49]. Recent works [8, 31, 45, 46] go one step further and apply meta-learning to pose estimation using gray-scale images. However, in these studies, the prediction is restricted to 1D rotation and the employed loss function is ill-posed as it does not take the periodicity of rotation into consideration. Moreover, [14] proposes to improve meta-learning by adding contrastive representation learning from disjoint context sets. A follow-up work [21] further extends this idea to time series data by combining contrastive learning with ConvNP [16]. However, in contrast to these two methods which need to learn a representation in a self-supervised way and fine-tune on downstream tasks subsequently, we use functional contrastive learning (FCL) between context and target sets and train in an end-to-end fashion.

Meta Overfitting. It is well-known that meta-learning algorithms suffer from two notorious types of overfitting: i) **Memorization overfitting** occurs when the model only conditions on the input to predict the output instead of relying on the context set [46]; ii) **Learner overfitting** happens when the prediction model and meta-learner overfit only to the training tasks and cannot generalize to novel tasks even though the prediction can condition on the context set [31]. Recently, different methods have been proposed to mitigate those overfitting issues, e.g., adding a regularization term on weights to restrict the memorization [46]. However, tuning a regularization term between underfitting and overfitting is challenging [24]. Subsequently, a related work [31] applied task augmentation which helps both memorization and learner overfitting. Meanwhile, [45] proposed MetaMix and Channel Shuffle to linearly combine features of context and target sets and replace channels with samples from different tasks. Furthermore, Ni et al. [28] empirically showed that data augmentation can also alleviate meta overfitting. Moreover, they find that employing data augmentation on target set achieves better performance. However, extensive comparisons on how these methods perform individually or combinedly are missing. In this work, we separate these techniques into data augmentation (DA), task augmentation (TA), meta-regularization (MR) and domain randomization (DR), and quantitatively compare them in different combinations on the two aforementioned applications in order to arrive at a better understanding and consistent comparisons.

3. Study Design

We now briefly describe both MAML and CNP in a unified way. We assume that all tasks are sampled from the same distribution $p(\mathcal{T})$, each task \mathcal{T}_i includes a context set $\mathcal{D}_C^i = \{(x_{C,1}, y_{C,1}), \dots, (x_{C,K}, y_{C,K})\}_i$ and a target set $\mathcal{D}_T^i = \{(x_{T,1}, y_{T,1}), \dots, (x_{T,M}, y_{T,M})\}_i$ where K and M are the number of samples in each set which could be different for each task. The entire training dataset is denoted as $\mathcal{D} = \{\mathcal{D}_C^i, \mathcal{D}_T^i\}_{i=1}^N$ where N is the number of tasks sampled for training. During inference, the model is tested on a new task $\mathcal{T}^* \sim p(\mathcal{T})$ given a small context set, from which it has to infer a new function $f^* : (\mathcal{D}_C^*, x_T^*) \rightarrow \hat{y}_T^*$. In meta-learning, there are two types of learned parameters, the first is the meta-parameters θ , which are learned during a meta-training phase using \mathcal{D} . The second is task-specific parameters ϕ^* which are updated based on samples from a new task \mathcal{D}_C^* conditioned on the learned meta-parameters θ . Predictions can be constructed as $\hat{y}_T^* = f_{\theta, \phi^*}(x_T^*)$, where f is the meta-model parameterized by θ and ϕ^* .

MAML considers both θ and ϕ^* as weights of neural networks while CNP considers only θ as neural weights. Different from MAML, which updates ϕ^* by gradient optimization on the new task samples, CNP takes ϕ^* as task

representation and predicts it from the context set as $\phi^* = \bigoplus_{i=1}^K h_{\theta}(x_{C,i}^*, y_{C,i}^*)$. Here \bigoplus is a permutation invariant operator, h is an encoder parameterized by θ . Subsequently, a decoder g_{θ} will take ϕ^* as an additional input and output $\hat{y}_T^* = g_{\theta}(x_T^*, \phi^*)$. Note that meta-parameters θ are fixed after meta-training phase, therefore CNPs don't require any fine-tuning as MAML.

3.1. Problem Setting

In this paper, we consider two types of image-based regression tasks, namely object discovery and pose estimation. First, we propose a non-trivial object discovery task called Distractor, which is only used for evaluating CNP variants. In contrast to existing object detection tasks [6, 12, 25, 34] that are designed to specify all object instances from an input image, our task aims to i) distinguish the queried object from other distractors and additionally ii) predict its 2D location in the image plane. Therefore, it is essential to learn a distinctive embedding ϕ^* that can represent various queried objects given their associated context images $\{x_{C,i}^*\}_{i=1}^K$ and corresponding positions $\{y_{C,i}^*\}_{i=1}^K$. Note that the distractors are sampled randomly from all categories and in many cases their appearances closely resemble the queried object. Hence, it is expected that aggregating multiple context pairs helps extracting expressive information to disambiguate the tasks and thus improve the performance.

The second task, pose estimation, is evaluated on three datasets, namely Pascal1D, ShapeNet1D and ShapeNet2D with incremental difficulty, caused e.g., by extending inference to unseen cross-category objects, adding random backgrounds and extending 1D rotations to 2D rotations. Note that in this task, each object has a random canonical pose, which has to be learned from a context set \mathcal{D}_C^* where $\{y_{C,i}^*\}_{i=1}^K$ are the ground-truth rotations of context images $\{x_{C,i}^*\}_{i=1}^K$.

We use these tasks for an exhaustive evaluation of meta-learning algorithms: i) We evaluate the performance of CNPs using different aggregation operators, i.e., mean [10], max, bayesian aggregation (BA) [39] and cross-attention (CA) [20]. ii) We evaluate MAML on Pascal1D and ShapeNet1D following [31, 46] and compare it with different CNP variants. iii) Furthermore, we investigate meta overfitting with respect to different choices, e.g., augmentations, regularization, aggregation operators and task properties. iv) Moreover, we combine functional contrastive learning (FCL) with CNPs and compare it with original CNPs.

3.2. Datasets

We generate **Distractor** that contains 12 object categories from ShapeNetCoreV2 [2], where each category includes 1000 randomly sampled objects. For each object we create 36 128×128 gray-scale images, containing two ob-

jects with random azimuth rotation and 2D position (see Fig. 1). The data generation is based on an extended version of a prior open-source pipeline [15]. We choose 10 categories for training, where we reserve 20% of the data for intra-category (IC) evaluation. The remaining 2 categories are only used for cross-category (CC) evaluation. The second dataset, **Pascal1D** [46], contains 65 objects from 10 categories. We randomly select 50 objects for training and the other 15 objects for testing. 100 128×128 gray-scale images are rendered for each object with a random rotation in azimuth angle normalized between $[0, 10]$. Since the performance is limited due to the size of the dataset, we generate a larger dataset, **ShapeNet1D** which includes 30 categories. 27 of these are used during training and IC evaluation, the other 3 categories are used for CC evaluation. For each training category, we randomly sample 50 objects for training and 10 for IC evaluation while CC evaluation is performed on 20 objects for each unseen category. To further increase the task difficulty, we create **ShapeNet2D** which includes 2D rotations. We restrict the azimuth angles to the range $[0^\circ, 180^\circ]$ in order to reduce the effect of symmetric ambiguity while elevations are restricted to $[0^\circ, 30^\circ]$. Furthermore, we use RGB images and employ randomly sampled real-world images from SUN2012 [44] as background instead of static background.

3.3. Data Augmentation, Domain Randomization, Task Augmentation and Meta Regularization

Data Augmentation (DA). We use standard image augmentation techniques in our work, i.e., *Dropout* and *Affine* for all tasks, and an additional *CropAndPad* for all pose regression tasks. Furthermore, we employ *Contrast*, *Brightness* and *Blur* for ShapeNet2D. Details are presented in Appendix A.4.

Domain Randomization (DR). For ShapeNet2D, we additionally employ DR [37] by regenerating background images for all training data after every 2k training iterations while the data used for evaluation remain the same.

Task Augmentation (TA). Task augmentation adds randomness to each task in order to encourage the meta-learner to learn non-trivial solutions instead of simply memorizing the training tasks. Following [31], we sample random noise $\epsilon^{(t)}$ from a discrete set for each task and create new tasks by adding the noise to the regression targets: $D_C^{(t)} = \{x_{C,i}^{(t)}, y_{C,i}^{(t)} + \epsilon^{(t)}\}_{i=1}^K$ and $D_T^{(t)} = \{x_{T,i}^{(t)}, y_{T,i}^{(t)} + \epsilon^{(t)}\}_{i=1}^M$. Specifically, we sample 2D position noise from a discrete set $\epsilon \in \{0, 1, 2, \dots, 16\}^2$ for Distractor. For Pascal1D, we use the same noise set $\{0., 0.25, 0.5, 0.75\}$ as proposed in [31, 46] while $\{0., 0.125, 0.25, \dots, 2\}$ for ShapeNet1D. In ShapeNet2D, we first only add random noise in the azimuth angle from the discrete set $\{-10^\circ, -9^\circ, \dots, 20^\circ\}$ and in a second step add additional elevation noise from the set $\{-5^\circ, -4^\circ, \dots, 10^\circ\}$ for further comparison.

Meta Regularization (MR). Following Yin et al. [46], we employ MR on the weights θ of the neural networks. Furthermore, we find that it is crucial to fine-tune the coefficient β which modulates the regularizer and task information stored in the meta-parameters θ . In our experiments, we use $\beta = 1e^{-4}$ for Pascal1D, $1e^{-7}$ for ShapeNet1D and ShapeNet2D. More details about MR are presented in Appendix A.5.

3.4. Functional Contrastive Learning (FCL)

The representations learned by CNP are invariant under permutation of the elements within a given context set. This property is achieved by a permutation invariant aggregation mechanism, e.g., max aggregation. However, another desirable property of the representation is invariance across context sets of the same task. In particular, the representations of different context sets belonging to the same task should be close to each other in the embedding space, while representations of different tasks should be farther apart. To achieve this, we add an additional contrastive loss at the functional space and train the model in an end-to-end fashion. The contrastive cross-entropy loss is defined as follows [3]:

$$\mathcal{L}_{\text{FCL}} = -\frac{2}{N} \sum_{t=1}^N \log \frac{\exp(\text{sim}(\phi_C^{(t)} \cdot \phi_T^{(t)})/\tau)}{D(\phi_C^{(t)})D(\phi_T^{(t)})}, \quad (1)$$

where N denotes the number of tasks per batch. $(\phi_C^{(t)}, \phi_T^{(t)})$ denotes a positive pair of latent representations of a given task obtained from context and target set respectively. More specifically, the pairs are obtained via max aggregation $\phi_C^{(t)} = \max(r_{C,1}^{(t)}, \dots, r_{C,K}^{(t)})$ and $\phi_T^{(t)} = \max(r_{T,1}^{(t)}, \dots, r_{T,M}^{(t)})$, where K denotes the number of context pairs per task and M the number of target pairs per task. \max returns the element-wise maximum over the latent variables $r_i = h_\theta(x_i, y_i)$ which are output by the encoder network h_θ for each context pair (x_i, y_i) . τ is a temperature parameter, which is crucial for learning good representations (details are presented in Appendix A.1). $\text{sim}(\cdot)$ is the cosine similarity and $D(\phi_i^t)$ sums the similarity of all positive and negative pairs for ϕ_i^t :

$$D(\phi_i^t) = \sum_{k=1}^N \sum_{j \in \{C,T\}} \mathbb{1}_{\{[k \neq t] \vee [j \neq i]\}} \exp\left(\frac{\text{sim}(\phi_i^t \cdot \phi_j^k)}{\tau}\right), \quad (2)$$

where $\mathbb{1}_{\{[k \neq t] \vee [j \neq i]\}} \in \{0, 1\}$ is an indicator evaluating to 1 only if the representations are sampled from different tasks or different sets. The log-value in Eq. (1) can be interpreted as the weighted importance of the positive pair. Therefore, this loss function encourages the model to obtain large similarity for positive pairs and small for negative pairs.

3.5. Objective Functions and Evaluation Metrics

Pascal1D. Following prior work [31, 45, 46] we conduct experiments using the MSE score between predicted and ground-truth azimuth rotation for both training and evaluation. However, this loss function does not take the ambiguity of coterminal angles into account. Hence, it can hamper the training process, e.g., predicting 359° for a ground-truth angle of 0° incurs a higher loss than predicting 180°. Nevertheless, we follow the same setup to obtain a fair comparison to prior works.

ShapeNet1D. Instead of using MSE score, we use the “cosine-sine-loss” for training and a prediction error defined in terms of the angular degree for evaluation. The loss of a single sample is defined as:

$$\mathcal{L} = |\cos(y) - \cos(y^*)|^2 + |\sin(y) - \sin(y^*)|^2, \quad (3)$$

where y^* is the ground-truth rotation and y the predicted rotation. The prediction error used for evaluation is defined as follows:

$$\mathcal{E} = \min\{\mathcal{E}_{y^+,y^*}, \mathcal{E}_{y^-,y^*}, \mathcal{E}_{y,y^*}\}, \quad (4)$$

where

$$\mathcal{E}_{y^\pm,y^*} = |y \pm 360 - y^*|, \mathcal{E}_{y,y^*} = |y - y^*|.$$

ShapeNet2D. We represent the 2D rotation as quaternion in both training and evaluation. The loss of a single sample is accordingly defined as follows:

$$\mathcal{L} = \min\left\{\left|q^* - \frac{q}{\|q\|}\right|, \left|q^* - \frac{-q}{\|q\|}\right|\right\}, \quad (5)$$

where q^* denotes the ground-truth unit quaternion and q denotes the predicted quaternion. We empirically find that using this objective function achieves a better performance than constraining the scalar part of q to be positive. We hypothesize that enforcing the scalar constraint breaks the continuity of the rotation representation and therefore hampers training.

4. Experiments

In this section we present experimental results¹, perform a thorough analysis and provide insights and recommendations. Instead of presenting the results following the task sequence, we structure this section by different algorithmic choices and perform a systematic comparison over all tasks by raising different questions. Appendix A.6 provides visualization examples of different tasks.

¹Codes and data are available at <https://github.com/boschresearch/what-matters-for-meta-learning>

Methods	Mean	Max	BA	CA	Max _{FCL}
No Aug	6.02	5.11	4.63	5.13	3.70
	6.89	6.17	5.91	6.39	4.61
DA	2.67	2.45	2.44	2.65	2.00
	4.10	3.75	3.97	4.08	3.05
TA	6.29	6.18	6.33	6.32	5.45
	7.19	7.04	7.02	7.02	6.66
TA+DA	3.20	3.09	2.65	3.05	2.60
	6.07	5.14	4.67	4.98	3.90

Table 1. Prediction error (pixel) on euclidean distance in the 2D image plane for Distractor. Different aggregation methods and augmentations are employed. The first row shows results for intra-category (IC) evaluations, the second row for cross-category (CC).

Methods	MAML	CNP (Mean)	CNP (CA)
No Aug	1.69 (0.22)	5.28 (0.51)	4.66 (0.74)
MR	1.90 (0.27)	2.96 (0.21)	3.33 (0.27)
TA	1.02 (0.06)	1.98 (0.22)	1.36 (0.25)
DA	2.10 (0.09)	3.69 (0.13)	2.90 (0.03)
TA+DA	1.31 (0.14)	2.29 (0.19)	1.77 (0.33)

Table 2. Pascal1D pose estimation error. MSE and standard deviations are calculated with 5 random seeds.

Methods	MAML	CNP (Max)	CNP (CA)
No Aug	25.27	14.97 (0.37)	8.19 (0.30)
	21.63	18.09 (0.21)	9.13 (0.18)
MR	13.23	12.71 (0.26)	8.87 (0.36)
	16.55	14.77 (0.35)	8.43 (0.39)
TA	23.01	10.89 (0.27)	7.92 (0.25)
	20.59	14.43 (0.55)	9.18 (0.50)
DA	14.69	8.64 (0.21)	6.24 (0.15)
	16.02	9.87 (0.35)	6.54 (0.19)
TA+DA	17.96	7.66 (0.18)	5.81 (0.23)
	18.79	8.66 (0.19)	6.23 (0.12)
TA+DA+FCL	—	7.82 (0.08)	6.44 (0.36)
	—	8.84 (0.04)	6.74 (0.20)
TA+DA+MR	13.45	10.54 (0.37)	8.28 (0.17)
	14.44	10.76 (0.30)	8.04 (0.10)

Table 3. ShapeNet1D pose estimation error(°). Results are calculated with 5 random seeds except for MAML. The first row presents results for IC and the second row for CC.

MAML or CNPs? We compare MAML and CNPs on two pose estimation datasets, Pascal1D and ShapeNet1D. We obtain similar results as [31, 46] on Pascal1D, where MAML performs better than CNPs and the latter shows more severe overfitting (see Tab. 2). However, Tab. 3 illustrates that both CNP variants outperform MAML with

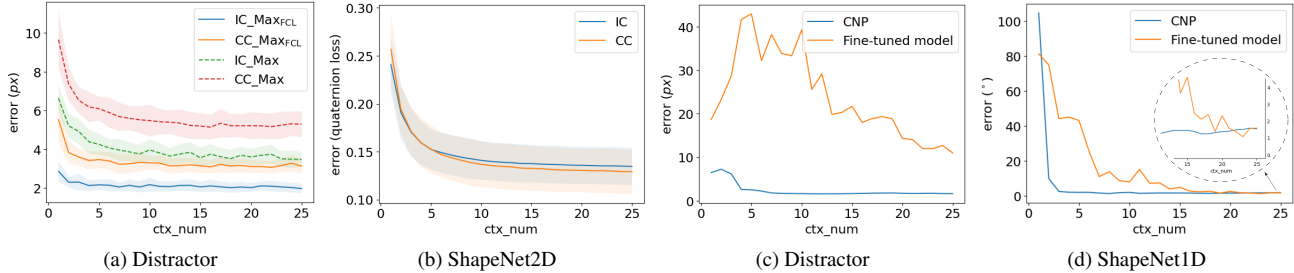


Figure 2. (a) CNP Prediction error (pixel) vs context number for the Distractor task using Max aggregation and Max + FCL (Max_{FCL}). Results are evaluated on novel objects from both intra-category (IC) and unseen cross-category (CC) levels. (b) CNP (CA) Prediction error vs context number for ShapeNet2D using DA + TA. (c) We compare a classical object detection method and CNP (Max) using same dataset for training on Distractor. The classical model is further fine-tuned on each new task. The results are shown in dependence of the number of images used for fine-tuning or as context set. (d) Prediction error between the fine-tuned model and CNP (CA) on ShapeNet1D.

a large margin on ShapeNet1D. It is good to note that, the prediction errors of all methods in Tab. 2 after denormalizing are larger than 30° , indicating the experiments of prior work on Pascal1D simply used too little meta-data to make informative conclusions about the quality of different algorithms. Our interpretation is that MAML tries to learn a good initial prior (global optimum) which needs to be optimized on each specific task (fine-tuned optimum) within few samples and updates. On small datasets, MAML can easily find a global optimum that satisfies all the training tasks. At the same time MAML also overfits less, since the fine-tuning from global to fine-tuned optimum happens during inference time. However, finding a global optimum is getting difficult for large-scale datasets due to the increasing task diversity. Consequently, more samples and updates are necessary to fine-tune the task-specific parameters ϕ (see Sec. 3), which also explains why MAML is sensitive to hyperparameter tuning [1]. Furthermore, MAML shows much longer training times than CNPs, which limits us to conduct exhaustive comparisons on more complicated tasks such as Distractor or ShapeNet2D. In contrast, CNPs use the local parameterization ϕ as a fixed dimensional output of the encoder, which forces the model to learn an informative low-rank representation from the contexts. Meanwhile, increasing data and task diversity will encourage the model to extract more expressive and mutual-exclusive task representations.

DA, DR, TA or MR? From the results of different experiments presented in Tab. 1, Tab. 2, Tab. 3 and Tab. 4, it is obvious that DA improves the performance across all tasks and methods. Tab. 4 also shows the importance of DR on ShapeNet2D which cannot be simply compensated by DA. TA hinders the performance on Distractor but benefits all pose regression tasks. The reason for this is that, for Distractor, TA increases task complexity by shifting the origin of the image plain by the sampled noise, thus creating N^2

Methods	IC ($1e^{-2}$)	CC ($1e^{-2}$)
None	38.33 (0.33)	39.81 (0.31)
DR	18.67 (0.13)	20.05 (0.12)
DR+MR	27.89 (0.61)	28.99 (0.46)
DR+TA _{azi}	16.94 (0.13)	18.42 (0.26)
DR+TA _{azi+ele}	16.62 (0.12)	17.76 (0.35)
DA	19.32 (0.09)	17.98 (0.09)
DR+DA	14.26 (0.09)	13.91 (0.14)
DR+DA+TA _{azi+ele}	14.12 (0.14)	13.59 (0.10)
DR+DA+TA _{azi+ele} + FCL	14.01 (0.09)	13.32 (0.18)

Table 4. Comparison of different augmentation techniques on ShapeNet2D. Results are calculated with 3 random seeds using CNP (CA) as baseline.

copies of the original task, where $N = 16$ is the number of non-zero elements in the noise set. However, since these task copies live in independent coordinate frames, the increased task diversity is irrelevant to the original task. For pose regression tasks, by contrast, TA augments the canonical poses of the existing data, which coherently benefits the original task as the augmented canonical poses remain in the coordinate frame of the original task. Therefore, even though TA increases the cross-entropy $\mathcal{H}(Y|X)$ for both cases as demanded in [31], only the pose regression tasks gain additional benefits. MR results in underfitting as combining MR with augmentations leads to worse performance than using the same augmentations alone for both ShapeNet1D and ShapeNet2D (see Tab. 3 and Tab. 4). Furthermore, MR requires extensive fine-tuning on the regularization parameter β (see Sec. 3.3) to modulate between underfitting and overfitting.

Effect of the context set size in CNPs. We compare the prediction error w.r.t. the size of the context set for Distractor (see Fig. 2a) and ShapeNet2D (see Fig. 2b).

Methods	CA _S	CA _M	CA _L	Max _S	Max _M	Max _L
No Aug	18.60 (0.78)	12.08 (0.44)	8.19 (0.30)	30.44 (0.82)	18.86 (0.34)	14.97 (0.37)
	19.95 (1.08)	12.62 (0.87)	9.13 (0.18)	30.59 (1.14)	21.78 (0.47)	18.09 (0.21)
TA	18.69 (0.87)	10.70 (0.98)	7.92 (0.25)	21.67 (0.66)	13.69 (0.27)	10.89 (0.27)
	19.24 (0.79)	12.05 (0.73)	9.18 (0.50)	23.60 (0.88)	16.76 (0.62)	14.43 (0.55)
TA+DA	7.86 (0.21)	6.32 (0.11)	5.81 (0.23)	11.00 (0.16)	8.23 (0.34)	7.66 (0.18)
	7.49 (0.35)	6.48 (0.41)	6.23 (0.12)	12.98 (0.48)	9.65 (0.40)	8.66 (0.19)

Table 5. Performance on ShapeNet1D using small (S), medium (M) and large (L) training dataset sizes for CNP with cross-attention (CA) and Max aggregation. The first row presents results for intra-category (IC) and the second row for cross-category (CC) evaluation. MSE and standard deviations are calculated with 5 random seeds.

Both figures show that increasing the context set size benefits the performance, indicating that both Max and CA aggregations can merge useful information from different context pairs and thereby reduce the task ambiguity. In addition, we find that the model can further improve the performance given the size of context set surpasses the maximum number used for training (15 for both tasks). In particular, there is a small performance gap between intra- and cross-category evaluation for Distractor which is however absent for ShapeNet2D. We believe this indicates that Distractor has more task ambiguity than pose estimation and thus explains why Distractor gains more benefits from FCL than ShapeNet2D (see Tab. 1 and Tab. 4).

CNPs vs pretrained models. It is a common practice in vision task to pretrain a model on a large-scale dataset (e.g., ImageNet [4]) in order to obtain good prior features and reduce training time. To conduct a fair comparison of this approach to our model regarding data efficiency, we first pretrain a classical object detection model jointly over all tasks using the same training data as for CNPs. After training has finished, we fine-tune the pretrained model further on each specific new task using different numbers of images. Results are shown in Fig. 2c for Distractor and Fig. 2d for ShapeNet1D, where the horizontal axis denotes the number of images used for fine-tuning or as contexts for CNPs, respectively. Both figures show that CNPs outperform the pretrained model especially for small numbers of contexts. In the Distractor task, CNP (Max) outperforms the fine-tuned model with a large margin after 25 context images are given. Note that CNPs are capable of transferring to various tasks simultaneously. In contrast, the pretrained model requires separate tuning on each given task, which results in a decreased performance on prior learning tasks.

Which aggregation methods should I use? Cross-attention (CA) performs better than mean aggregation on Pascal1D (see Tab. 2) and Max on ShapeNet1D (see Tab. 3), while it achieves a similar performance to Max aggregation and BA on ShapeNet2D (see Tab. 6). In contrast, mean ag-

Methods	IC ($1e^{-2}$)	CC ($1e^{-2}$)
CNP+Mean	15.04 (0.08)	15.45 (0.13)
CNP+Max	14.20 (0.06)	13.56 (0.28)
CNP+BA	14.16 (0.08)	13.56 (0.18)
CNP+CA	14.12 (0.14)	13.59 (0.10)

Table 6. Comparison of aggregation methods on ShapeNet2D using DR+DA+TA. Results are calculated with 3 random seeds.

gregation used in the original CNP performs the worst on both Pascal1D and ShapeNet2D. Our interpretation is that Mean assigns the same importance to each context while the other aggregation operators can allocate different weights. Max assigns a weight of one to a context and zero to all others for each dimension of the representation while BA assigns the weights predicted by another neural network. Meanwhile, CA assigns importance by comparing the similarity between context inputs $\{x_C^i\}_{i=1}^K$ and target input x_T at the feature space.

Furthermore, we find that CA achieves competitive results on all pose estimation tasks but performs slightly worse than BA and Max on Distractor (see Tab. 1) though still better than mean aggregation. This indicates that CA helps in learning representations for object-centric images. Distractor, however, contains objects with random locations, requiring the model to disregard positional information. Methods like CA, which compare similarity between contexts and target over feature space, face inherent difficulties on Distractor. This is due to the fact that CNNs, owing to their translational equivariant nature, are prone to encode some positional information into the extracted image features. Consequently, CA, which compares the similarity directly on this feature space, inevitably forces the model to focus on positional similarity, which leads to a suboptimal allocation of importance.

How much meta-data is essential? We split the training data of ShapeNet1D into subsets of three different sizes, with 10 objects per category for the small dataset (S), 30

objects per category for the medium dataset (M) and 50 for the large dataset (L). Afterwards, we test the performance of CNP with Max aggregation and CA on each of them. The results in Tab. 5 show that Max overfits on the small dataset by simply memorizing all training tasks while CA works much better. Moreover, CA trained on small dataset achieves a comparable performance with Max on large dataset after using TA and DA, and even outperforms Max on the cross-category level. Thus, we conclude that using CA in combination with augmentation techniques can drastically alleviate the overfitting problem and therefore requires less meta-data on object-centric vision tasks than Max. In contrast, MAML performs much worse on ShapeNet1D (L) (see Tab. 3) than CNPs and thus hardly profits from an increased dataset.

Data augmentation. Tab. 7 shows the effect of each individual data augmentation technique (see Sec. 3.3) on ShapeNet2D. The first row contains results obtained with all techniques applied jointly. In the other rows, one of the techniques is removed respectively. We find that removing *Affine* leads to the worst performance which indicates that object-centric pose regression tasks are more sensitive to scale. On the other hand, omitting *CropAndPad* even leads to an performance increase.

Methods	Val	Test
All	0.1417	0.1410
w/o CropAndPad	0.1412	0.1368
w/o Affine	0.1623	0.1743
w/o Dropout	0.1452	0.1445
w/o Contrast	0.1482	0.1406
w/o Brightness	0.1454	0.1380
w/o Blur	0.1426	0.1422

Table 7. Comparison of different data augmentation techniques on ShapeNet2D using CNP (CA) + DR as baseline.

Methods	IC	CC
Same Ctx	2.30 (0.04)	3.46 (0.06)
Diff Ctx	2.16 (0.05)	3.25 (0.05)
Ctx & Target	2.00 (0.02)	3.05 (0.08)

Table 8. Analysis of FCL + CNP on different choices of positive pairs using: i) the same context set with different augmentations (Same Ctx), ii) different context sets from the same task (Diff Ctx), iii) context and target sets (Ctx & Target). Prediction error (pixel) is calculated with 3 random seeds.

Does FCL improve CNPs? Tab. 1 shows the evaluation on Distractor using different aggregation methods where

Max_{FCL} denotes Max aggregation with FCL. Modulating task representation by functional contrastive learning (FCL) alleviates meta overfitting across all augmentation levels and thus achieves a significant improvement in performance. Fig. 2a further compares the performance of Max and Max_{FCL} for different context set sizes, showing that our methods can differentiate the queried object and distractors well, even for very small context sets. Furthermore, we investigate the influence of FCL on the predicted task representations over all 12 categories using different clustering metrics, where the results show that FCL leads to a more dispersed latent distribution compared to the original CNPs, which can improve generalization capability to unseen tasks. T-SNE visualizations of the task representations along with the results of cluster metrics are provided in Appendix A.1.

FCL on different sets. We compare FCL on three choices of positive pairs: i) We use the same context set but with different data augmentations. ii) We use different context sets sampled from the same task. iii) We use context and target sets from the same task. We test the performance on Distractor using Max aggregation and DA. For each choice, we run three experiments with different seeds and present the average performance in Tab. 8. Compared to Tab. 1, all three choices consistently outperform CNP (Max) while using FCL on context and target sets achieves the best performance.

Limitations. Since meta-learning algorithms are data-driven methods, generalization depends on the diversity of training tasks. However, the augmentation methods used in our work are limited in their capability of creating new and diverse training tasks. Therefore, using a generative model to enrich training data could be one possibility to achieve higher diversity. Furthermore, concerning the class of NPs, we restricted ourselves to the deterministic CNPs in the experiments. We leave an in-depth exploration of stochastic NPs on vision tasks to future work.

5. Conclusion

In this paper, we investigate MAML and CNPs on several image-level regression tasks and analyze the importance of different choices in mitigating meta overfitting. Furthermore, we provide insights and practical recommendations of different algorithmic choices for CNPs with respect to various task settings. In addition, we combine CNPs with functional contrastive learning in task space and train in an end-to-end manner, which significantly improves the task expressivity of CNPs. We believe that our work can lay the basis for future work on designing and implementing meta-learning algorithms in image-based regression tasks.

References

- [1] Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your MAML. In *International Conference on Learning Representations*, 2019. [6](#)
- [2] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, L. Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *ArXiv*, abs/1512.03012, 2015. [3](#)
- [3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020. [2](#), [4](#)
- [4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. [7](#)
- [5] Thomas Elsken, Benedikt Staffler, Jan Hendrik Metzen, and Frank Hutter. Meta-learning of neural architectures for few-shot learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. [2](#)
- [6] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, 2010. [3](#)
- [7] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 06–11 Aug 2017. [1](#), [2](#)
- [8] Chelsea Finn, Aravind Rajeswaran, Sham Kakade, and Sergey Levine. Online meta-learning. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1920–1930. PMLR, 09–15 Jun 2019. [1](#), [2](#)
- [9] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1568–1577. PMLR, 10–15 Jul 2018. [1](#)
- [10] Marta Garnelo, Dan Rosenbaum, Christopher Maddison, Tiago Ramalho, David Saxton, Murray Shannahan, Yee Whye Teh, Danilo Rezende, and S. M. Ali Eslami. Conditional neural processes. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1704–1713. PMLR, 10–15 Jul 2018. [1](#), [2](#), [3](#)
- [11] Marta Garnelo, Jonathan Schwarz, Dan Rosenbaum, Fabio Viola, Danilo J. Rezende, S. M. Ali Eslami, and Yee Whye Teh. Neural processes. In *ICML Workshop on Theoretical Foundations and Applications of Deep Generative Models*, 2018. [1](#), [2](#)
- [12] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361, 2012. [3](#)
- [13] Muhammad Waleed Gondal, Shruti Joshi, Nasim Rahaman, Stefan Bauer, Manuel Wuthrich, and Bernhard Schölkopf. Function contrastive learning of transferable meta-representations. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 3755–3765. PMLR, 18–24 Jul 2021. [2](#)
- [14] Muhammad Waleed Gondal, Shruti Joshi, Nasim Rahaman, Stefan Bauer, Manuel Wuthrich, and Bernhard Schölkopf. Function contrastive learning of transferable meta-representations. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 3755–3765. PMLR, 18–24 Jul 2021. [2](#)
- [15] Jonathan Gordon, John Bronskill, Matthias Bauer, Sebastian Nowozin, and Richard Turner. Meta-learning probabilistic inference for prediction. In *International Conference on Learning Representations*, 2019. [2](#), [4](#)
- [16] Jonathan Gordon, Wessel P. Bruinsma, Andrew Y. K. Foong, James Requeima, Yann Dubois, and Richard E. Turner. Convolutional conditional neural processes. In *International Conference on Learning Representations*, 2020. [1](#), [2](#)
- [17] Abhishek Gupta, Russell Mendonca, YuXuan Liu, Pieter Abbeel, and Sergey Levine. Meta-reinforcement learning of structured exploration strategies. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. [2](#)
- [18] Timothy M Hospedales, Antreas Antoniou, Paul Mi-

- caelli, and Amos J. Storkey. Meta-learning in neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. [2](#)
- [19] Alexander B. Jung, Kentaro Wada, Jon Crall, Satoshi Tanaka, Jake Graving, Christoph Reinders, Sarthak Yadav, Joy Banerjee, Gábor Vecsei, Adam Kraft, Zheng Rui, Jirka Borovec, Christian Vallentin, Semen Zhydenko, Kilian Pfeiffer, Ben Cook, Ismael Fernández, François-Michel De Rainville, Chi-Hung Weng, Abner Ayala-Acevedo, Raphael Meudec, Matias Laporte, et al. `imgaug`. <https://github.com/aleju/imgaug>, 2020. Online; accessed 01-Feb-2020. [13](#)
- [20] Hyunjik Kim, Andriy Mnih, Jonathan Schwarz, Marta Garnelo, Ali Eslami, Dan Rosenbaum, Oriol Vinyals, and Yee Whye Teh. Attentive neural processes. In *International Conference on Learning Representations*, 2019. [1](#), [2](#), [3](#)
- [21] Kozuka Kazuki Ohama Iku Luca Rigazio Konstantinos Kallidromitis, Denis Gudovskiy. Contrastive neural processes for self-supervised learning. In *Asian Conference on Machine Learning*, 2021. [2](#)
- [22] Byung-Jun Lee, Seunghoon Hong, and Kee-Eung Kim. Residual neural processes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):4545–4552, Apr. 2020. [1](#), [2](#)
- [23] Hayeon Lee, Eunyoung Hyung, and Sung Ju Hwang. Rapid neural architecture search by learning to generate graphs from datasets. In *International Conference on Learning Representations*, 2021. [2](#)
- [24] Hao Li, Pratik Chaudhari, Hao Yang, Michael Lam, Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. Rethinking the hyperparameters for fine-tuning. In *International Conference on Learning Representations*, 2020. [3](#)
- [25] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 740–755, 2014. [3](#)
- [26] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. In *International Conference on Learning Representations*, 2019. [2](#)
- [27] Christos Louizos, Xiahan Shi, Klamer Schutte, and Max Welling. The functional neural process. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. [1](#), [2](#)
- [28] Renkun Ni, Micah Goldblum, Amr Sharaf, Kezhi Kong, and Tom Goldstein. Data augmentation for meta-learning. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8152–8161. PMLR, 18–24 Jul 2021. [2](#), [3](#)
- [29] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *ArXiv*, abs/1803.02999, 2018. [1](#), [2](#)
- [30] Alexander Norcliffe, Cristian Bodnar, Ben Day, Jacob Moss, and Pietro Liò. Neural {ode} processes. In *International Conference on Learning Representations*, 2021. [1](#)
- [31] Janarthanan Rajendran, Alexander Irpan, and Eric Jang. Meta-learning requires meta-augmentation. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 5705–5715. Curran Associates, Inc., 2020. [2](#), [3](#), [4](#), [5](#), [6](#)
- [32] Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, and Deirdre Quillen. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5331–5340. PMLR, 09–15 Jun 2019. [2](#)
- [33] Mengye Ren, Sachin Ravi, Eleni Triantafillou, Jake Snell, Kevin Swersky, Josh B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. Meta-learning for semi-supervised few-shot classification. In *International Conference on Learning Representations*, 2018. [1](#), [2](#)
- [34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2015. [3](#)
- [35] 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, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. [1](#), [2](#)
- [36] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H.S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1199–1208, 2018. [1](#), [2](#)
- [37] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ In-*

- ternational Conference on Intelligent Robots and Systems (IROS)*, pages 23–30, 2017. [4](#)
- [38] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. [1](#), [2](#)
- [39] Michael Volpp, Fabian Flürenbrock, Lukas Grossberger, Christian Daniel, and Gerhard Neumann. Bayesian context aggregation for neural processes. In *International Conference on Learning Representations*, 2021. [3](#)
- [40] Michael Volpp, Lukas P. Fröhlich, Kirsten Fischer, Andreas Doerr, Stefan Falkner, Frank Hutter, and Christian Daniel. Meta-learning acquisition functions for transfer learning in bayesian optimization. In *International Conference on Learning Representations*, 2020. [1](#)
- [41] Risto Vuorio, Shao-Hua Sun, Hexiang Hu, and Joseph J Lim. Multimodal model-agnostic meta-learning via task-aware modulation. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. [2](#), [12](#), [13](#)
- [42] Qi Wang and Herke Van Hoof. Doubly stochastic variational inference for neural processes with hierarchical latent variables. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 10018–10028. PMLR, 13–18 Jul 2020. [1](#), [2](#)
- [43] Ying Wei, Peilin Zhao, and Junzhou Huang. Meta-learning hyperparameter performance prediction with neural processes. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 11058–11067. PMLR, 18–24 Jul 2021. [1](#)
- [44] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3485–3492, 2010. [4](#)
- [45] Huaxiu Yao, Long-Kai Huang, Linjun Zhang, Ying Wei, Li Tian, James Zou, Junzhou Huang, and Zhenhui Li. Improving generalization in meta-learning via task augmentation. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 11887–11897. PMLR, 18–24 Jul 2021. [2](#), [3](#), [5](#)
- [46] Mingzhang Yin, George Tucker, Mingyuan Zhou, Sergey Levine, and Chelsea Finn. Meta-learning without memorization. In *International Conference on Learning Representations*, 2020. [2](#), [3](#), [4](#), [5](#), [13](#)
- [47] Jaesik Yoon, Taesup Kim, Ousmane Dia, Sungwoong Kim, Yoshua Bengio, and Sungjin Ahn. Bayesian model-agnostic meta-learning. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. [1](#), [2](#)
- [48] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. *CoRR*, abs/1910.10897, 2019. [2](#)
- [49] Barret Zoph and Quoc V. Le. Neural architecture search with reinforcement learning. In *International Conference on Learning Representations*, 2017. [2](#)

A. Appendix

A.1. Functional Contrastive Learning on CNPs

τ	1.0	0.5	0.2	0.07	0.007
IC	8.5550	8.9810	8.8551	7.8196	8.1409
CC	10.4660	10.5135	10.5604	8.8420	9.3846

Table 1. Results of the evaluation on ShapeNet1D using different temperature values in FCL.

τ	1.0	0.5	0.2	0.07	0.007
IC	0.1564	0.174	0.1962,	0.1441	0.1401
CC	0.1594	0.1758	0.2089	0.1390	0.1332

Table 2. Results of the evaluation on ShapeNet2D using different temperature values in FCL.

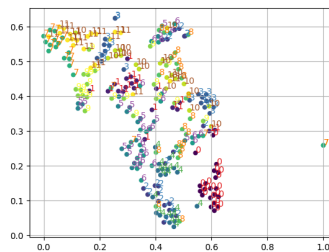
Methods	Max	Max _{FCL}
ARI \uparrow	0.21	0.20
MI \uparrow	1.13	1.03
SS \uparrow	0.31	0.15
CHI \uparrow	118.73	18.90
DBI \downarrow	1.00	1.65

Table 3. Analysis of latent task representation on Distractor between Max and Max_{FCL} using various clustering metrics.

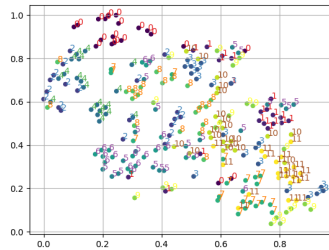
A grid search on hyperparameter τ is very expensive especially on vision tasks. Therefore, we search only on a discrete set $\{0.007, 0.7, 0.2, 0.5, 1.0\}$ and find that $\tau = 0.07$ shows the best performance on ShapeNet1D and $\tau = 0.007$ on ShapeNet2D. The results are shown in Tab. 1 and Tab. 2.

For the Distractor task, we visualize the task representation obtained for novel objects in Fig. 1 where each color or number denotes one category and each point denotes the representation of each novel object. $\{10, 11\}$ are the novel categories $\{sofa, watercraft\}$. Note that each object is considered as a single task and all tasks are learned in a category-agnostic manner. This figure indicates that Max_{FCL} can better shrink the distance between similar objects and repel the different ones implicitly. For instance, without a contrastive loss there is one outlier in Fig. 1a that is far away in representation space from the other objects. In particular, some samples are not well clustered based on categories, which is due to the high object variations within the same category.

Furthermore, we investigate the influence of FCL on the predicted task representations over all 12 categories using five clustering metrics, namely Adjusted Rand Index (ARI),



(a) Max



(b) Max_{FCL}

Figure 1. Visualization of latent variables on (a) max aggregation (b) max aggregation + functional contrastive learning (Max_{FCL}).

Mutual Information (MI), Silhouette Score (SS), Calinski-Harabasz Index (CHI) and Davies-Bouldin Index (DBI). Results are shown in Tab. 3. FCL leads to a more dispersed latent distribution compared to the original CNP, which reduces the vacancy in the latent space and thus improve the generalization ability to unseen tasks.

A.2. Training Details

For all tasks, we use 500k training iterations for CNPs and 70k for MAML. Furthermore, the best model on the intra- and cross-category dataset is saved during training. This leads to better models than early stopping with manually defined intervals. All experiments are conducted on a single NVIDIA V100-32GB GPU. Distractor and ShapeNet2D need around 3 – 5 days for training, depending on different choices of augmentations, Pascal1D needs 8 hours and ShapeNet1D around 12 hours.

Additional Results. We have evaluated MMAML [41], a conditional variant of MAML, on ShapeNet1D based on reviewer’s recommendation in Tab. 4. The results is worse than MAML, indicating that the designed task-aware modulation in MMAML doesn’t benefit our tasks.

A.3. Task Augmentation

The angular orientation of Pascal1D is normalized to $[0, 10]$ whereas ShapeNet1D uses radians with range $[0, 2\pi]$. For ShapeNet2D, the azimuth angles are restricted to the range $[0^\circ, 180^\circ]$ in order to reduce the effect of symmetric

MMAML	No Aug	DA	TA	DA+TA
IC	19.6900	26.3624	19.0705	27.4973
CC	20.6123	26.4090	19.4285	27.3120

Table 4. Performance of MMAML [41] on ShapeNet1D.

ambiguity while elevations are restricted to $[0^\circ, 30^\circ]$. we add random noise to both azimuth and elevation angles and then convert the rotation to quaternions for training.

A.4. Data Augmentation

Affine scales images between 80% – 120% of their size along x and y axis and translate the images between $-10\% - 10\%$ relative to the image height and width, and fills random value for the newly created pixels. *Dropout* either drops random 1%-10% of all pixels or random image patches with 2% – 25% of the original image size. *CropAndPad* pads each side of the images less than 5% of the image size using random value or the closest edge value. For ShapeNet2D, we furthermore add *GammaContrast* with a range $[0.5, 2.0]$, *AddToBrightness* with a range $[-30, 30]$ and *AverageBlur* using a window of $k \times k$ neighbouring pixels where $k \in [1, 3]$. We use the open-source package [19] for all data augmentations.

A.5. Meta Regularization

Yin et al. [46] employ regularization on weights, the loss function is defined as:

$$\mathcal{L} = \mathcal{L}_O + \beta D_{\text{KL}}(q(\theta; \theta_\mu, \theta_\sigma) || r(\theta)) \quad (6)$$

where L_O denotes the original loss function defined individually in Distractor and pose estimation. meta-parameters θ denote the parameters which are not used to adapt to the task training data. Function $r(\theta)$ is a variational approximation to the marginal which is set to $\mathcal{N}(\theta; 0, I)$ in Yin et al. [46]. We follow the same setup in our experiments.

A.6. Examples of Inference Results

We visualize examples of evaluation on novel categories in Fig. 2 for Distractor, Fig. 3 for ShapeNet1D and Fig. 4 for ShapeNet2D.

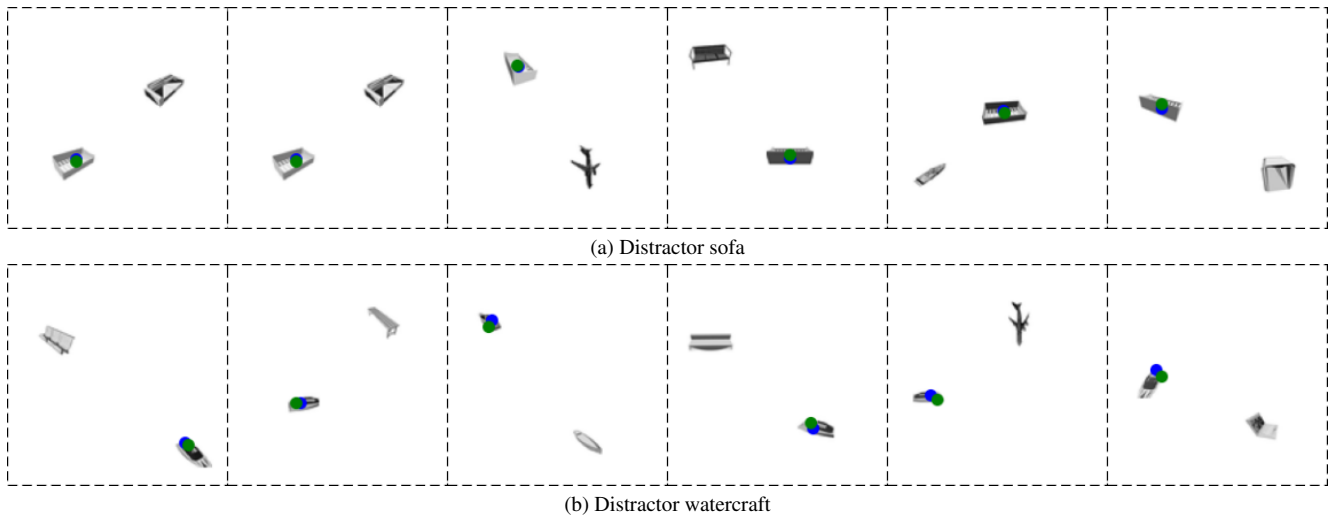


Figure 2. Examples of Distractor on novel categories (sofa and watercraft) where green dots are ground-truth and blue dots are predicted positions.

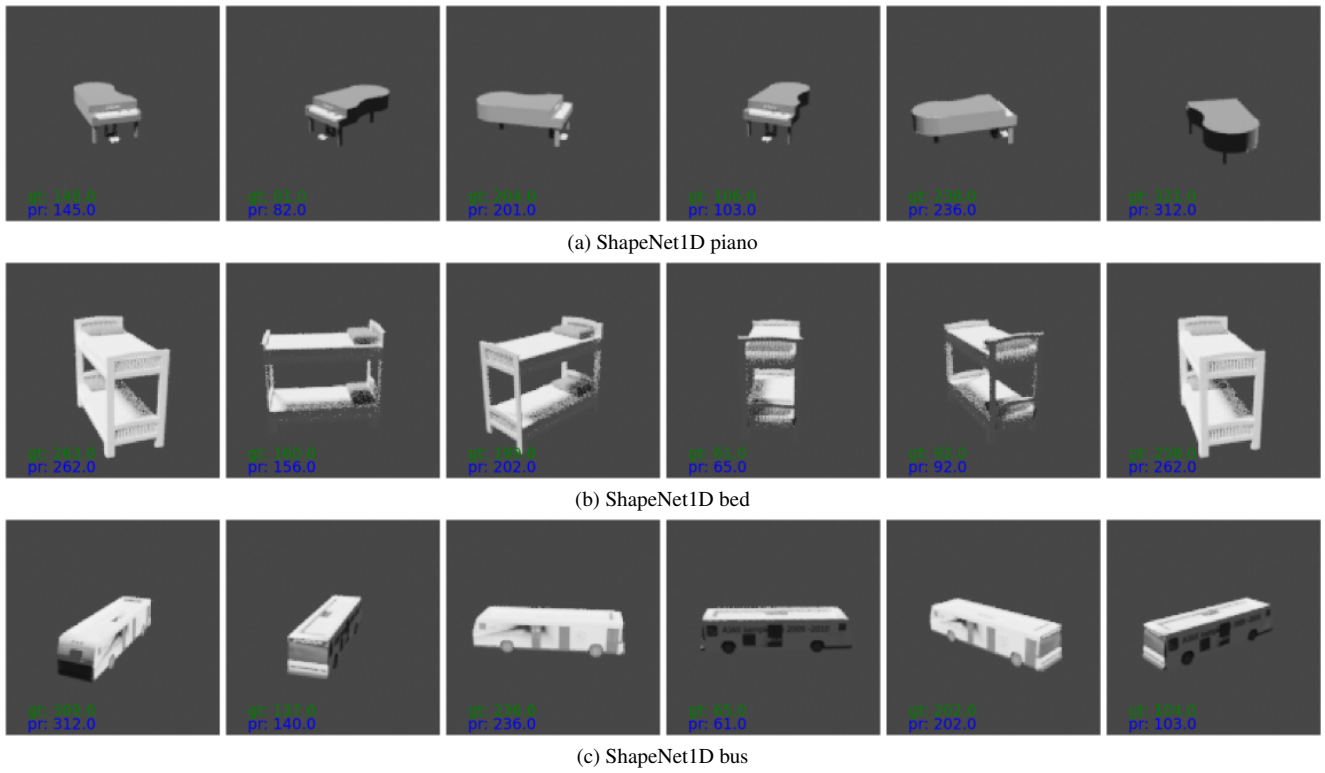


Figure 3. Examples of ShapeNet1D on novel categories (piano, bed, bus).



Figure 4. Examples of ShapeNet2D on novel categories (piano, bed, bus). Predictions are converted to (azimuth, elevation) angles.