In-Context Symmetries: Self-Supervised Learning through Contextual World Models

Sharut Gupta[*], **Chenyu Wang**[*], **Yifei Wang**[*], **Tommi Jaakkola**
MIT CSAIL
{sharut, wangchy, yifei_w, jaakkola}@mit.edu

Stefanie Jegelka
TU Munich, MIT CSAIL
stefje@mit.edu

Abstract

Self-supervised learning for vision typically focuses on learning invariant or equivariant representations through data transformations, but this approach introduces priors and biases that weaken performance in tasks not aligned with these symmetries. Inspired by world models, we propose Contextual Self-Supervised Learning (CONTEXTSSL), which learns a general representation adaptable to different transformations by leveraging context—a memory module that tracks task-specific states, actions (transformations), and future states. Instead of enforcing invariance, CONTEXTSSL learns equivariance to all transformations, enabling the model to encode general features while adapting to task-specific symmetries with a few examples. Empirically, we demonstrate significant performance gains over existing methods on equivariance-related tasks. Code is available at https://github.com/Sharut/In-Context-Symmetries.

1 Introduction

Recent advances in self-supervised learning (SSL) of image representations have achieved competitive performance to it's supervised counterparts across various tasks, including image classification [9, 6, 43, 29, 17, 3, 18, 19, 35, 31, 10, 25, 11, 42, 36, 44]. Most SSL approaches rely on joint-embedding architectures that bring semantically similar (positive) pairs closer together and push dissimilar (negative) pairs apart. Positive pairs are typically generated through data augmentations like color changes, cropping, or orientation shifts. These methods often enforce either invariance [9, 6, 11, 25, 42, 19] or equivariance [21, 13, 12, 15, 1, 16] to augmentations, introducing strong inductive biases that may not generalize well across different downstream tasks. For instance, invariance to image flipping is useful for classification but hinders segmentation, leading to brittle representations that may require task-specific retraining with tailored augmentations. This motivates our central question

Can incorporating context into self-supervised vision algorithms eliminate augmentation-based inductive priors and enable dynamic adaptation to varying task symmetries?

This work suggests a positive answer to this question by proposing to enhance the current joint embedding architecture with a finite context — an abstract representation of a task, containing a few demonstrations that inform about task-specific symmetries, as shown in Figure 3(c). Based on this idea, we propose **Context**ual **S**elf-**S**upervised **L**earning (CONTEXTSSL), a contrastive learning framework that uses a transformer module to adapt to selective invariance or equivariance to transformations by paying attention to context representing a task. Unlike previous approaches with built-in symmetries, the ability of CONTEXTSSL to adapt to varying data symmetries—all without undergoing any parameter updates—enables it to learn a general representation across tasks, devoid of specific inductive priors. We demonstrate that in the absence of context, CONTEXTSSL

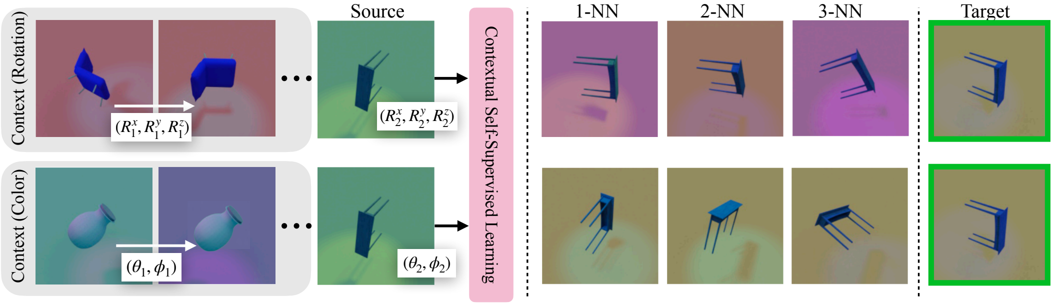


Figure 1: We apply a transformation (rotation or color) on a source image in latent space and retrieve the nearest neighbor (NN) of the predicted representation when the context contains pairs of data transformed by (top row) 3D rotation (R^x, R^y, R^z); (bottom row) color transformation (θ, ϕ). In the top row, we see that CONTEXTSSL learns equivariance to rotation and invariance to color as the NN representations match the target's angle but not its color. In the bottom row, it adapts to the color context and enforces the reverse, be equivariant to color and invariant to rotation.

learns a general representation by encoding all relevant features and data transformations. As the context increases, the model tailors its symmetries to a task, encouraging equivariance to a subset of transformations and invariance to the rest (as shown in Figure 1). We validate our approach on real world a physiological dataset MIMIC III [28], a fairness benchmark UCI-Adult [2], 3DIEBench and CIFAR10. We also extend CONTEXTSSL to supervised learning, demonstrating its ability to identify task-defining features through context.

2 Beyond Built-in Symmetry: Contextual Self-Supervised Learning

Recognizing the limitations of existing augmentation-specific SSL methods, we propose a new paradigm: **Context**ual **S**elf-**S**upervised **L**earning (CONTEXTSSL). Unlike traditional methods, this approach learns a single model that adapts to be either invariant or equivariant based on context-specific augmentations, tailored to the needs of the task or data at hand. Instead of enforcing a fixed set of symmetries, CONTEXTSSL learns these symmetries from contextual cues, thus capturing the unique set of features of downstream tasks. This adaptability allows it to serve as a general-purpose SSL framework, capable of learning from a diverse array of pretraining tasks with varying symmetry priors and seamlessly adapting to different downstream tasks.

2.1 Contextual World Models

Symmetries as Context. Given a set of groups of input transformations $\{\mathcal{G}_1, \ldots, \mathcal{G}_M\}$, the goal of CONTEXTSSL is to build a general representation that is adaptive to a set of multiple symmetries corresponding to these different groups. Each group $\mathcal{G}_c$ can be represented through the joint distribution $P(x, a, y|\mathcal{G}_c)$, where x is the input sample (sampled from an unlabeled dataset), a represents the parameters of the transformation drawn from $\mathcal{G}_c$ and applied to x, and y is the transformed input. We approximate this probability distribution by drawing K samples from the joint distribution and form a context $C(\mathcal{G}_c) = [(x_1, a_1, y_1), \ldots, (x_K, a_K, y_K)]$, where $x_i, a_i, y_i \sim P(x, a, y|\mathcal{G}_c), i \in [K]$.

Contextual World Models. To implement this broad goal, we propose to adaptively learn the symmetries represented by $\mathcal{G}_c$ by training the model:

$$y_i \approx h((x_i, a_i); (x_1, a_1, y_1), \ldots, (x_{i-1}, a_{i-1}, y_{i-1})). \tag{1}$$

While the requested prediction y_i concerns only the inputs x_i and a_i, the model can now pay attention to the experience so far, enforcing relevant symmetries for the augmentation group $\mathcal{G}_c$. The predictor h is updated by minimizing the loss at each context length $\sum_{i=1}^{K} \ell(h((x, a_i); C_{i-1}), y_i)$ where $C_i = \{(x_1, a_1, y_1), \ldots, (x_{i-1}, a_{i-1}, y_{i-1})\}$ represents the context before index i.

A natural approach to enable context-based training is through attention mechanisms in transformer-based autoregressive models. Large language models excel at in-context learning, generalizing to new tasks by focusing on a few examples. Inspired by this, we train a decoder-only transformer model in-context, conditioning on relevant context $C(\mathcal{G}_c)$ that represents the transformation group $\mathcal{G}_c$.

2.2 Contextual Self-Supervised Learning (CONTEXTSSL)

Motivated by these ideas, we construct pairs of points $\{(x_i, y_i)\}_{i=1}^{K}$ by either 1) sampling a transformation group $(\mathcal{G})$ and applying an augmentation from (gG) to x_i to obtain y_i; or 2) if available, sampling a meta-latent and its transformation parameters based on the difference between their latent parameters. Pairs can also be transformed by augmentations from other transformation groups, but the context $C(\mathcal{G})$ only uses parameters from $\mathcal{G}$.

Each input pair $\{(x_i, y_i)\}_{i=1}^{K}$ is independently transformed by the encoder into latent representations. The representation of x_i is then concatenated with its transformation action a_i. This concatenated vector (x_i, a_i) and the transformed input y_i form the context for the symmetry $\mathcal{G}$. The output embeddings are aligned using the InfoNCE loss, minimized at each context length. If a_i is set to zero for all tokens, the model enforces invariance to $\mathcal{G}$ by aligning x_i and y_i without considering transformation parameters. The overall loss is optimized as follows:

$$\mathcal{L}_{\text{CONTEXTSSL}}(h) = \mathbb{E}_{\mathcal{G}\sim\{\mathcal{G}_1,\ldots\mathcal{G}_M\}}\mathbb{E}_{C(\mathcal{G})}\sum_{i=1}^{K}\left[-\log\frac{\exp h((x_i,a_i)|C_i(\mathcal{G}))^\top h(y_i|C_i(\mathcal{G}))/\tau}{\sum_{j=1}^{K}\exp h((x_i,a_i)|C_i(\mathcal{G}))^\top h(y_j|C_j(\mathcal{G}))/\tau}\right]$$

where transformed data tokens y_j $(j \neq i)$ form the negatives. We use a similar symmetric loss term using y_i as the anchor, (x_i, a_i) and (x_j, a_j) $(j \neq i)$ as the positive and negatives respectively.

At inference, we adapt representation extraction to the specific needs of the downstream task, whether it requires equivariance or invariance to a transformation group $\mathcal{G}$. For tasks benefiting from equivariance, we use the maximum context length K from training, constructing $\{(x_i, a_i, y_i)\}_{i=1}^{K}$ where a_i belongs to $\mathcal{G}$ and transforms test data x_i into y_i. For tasks needing invariance, we use $\{(x_i, 0, y_i)\}_{i=1}^{K}$ as the context. Specifically, including the augmentation parameters for transformations in a group $\mathcal{G}$ in the context enforces equivariance, while excluding them enforces invariance. In both cases, the data are still transformed using augmentations, regardless of the type of symmetry desired. However, this implementation bears two key challenges, as detailed below.

Context Masking. A challenge in minimizing alignment loss is the model's tendency toward shortcut learning, where it treats the embeddings of (x_i, a_i) as identical to y_i due to access to x_i. This leads to constant representations for each pair (x_i, y_i), perfectly minimizing the loss but undermining the model's effectiveness. To counter this, we mask the input token (x_i, a_i) for each y_i in the context, ensuring that when encoding y_i, the transformer only has access to the past context $C_i = \{(x_1, a_1, y_1), \ldots, (x_{i-1}, a_{i-1}, y_{i-1})\}$, excluding its corresponding pair.

However, as shown in Figure 4, for $p = 0$, a residual challenge of shortcut learning persists when distinguishing the positives from the negatives. Since the context corresponding to each negative is different from that of the anchor and the positive, the model could employ trivial solutions, such as using the mean of the context vector to differentiate between positives and negatives. To mitigate this issue, we introduce an additional layer of randomness to our masking strategy. Specifically, for each token in the context, we implement random masking with a probability p for tokens preceding it.

Avoiding collapse to Invariance. A trivial but undesirable solution that minimizes our optimization objective is invariance to the input transformations. As illustrated in Figure 5, naively training CONTEXTSSL leads to poor equivariance with respect to the transformations. Previous works [15] have also identified this concern. For our setting, we introduce a rather simple approach that involves jointly training an auxiliary predictor. This predictor is designed to predict the latent transformations of the target sample y_i from the concatenated input vector (x_i, a_i).

3 Experimental Results

3.1 Quantitative Assessment of Adaptation to Task-Specific Symmetries

We use the 3D Invariant Equivariant Benchmark (3DIEBench) [15] and CIFAR10 to test our approach. We compare CONTEXTSSL with 1) VICReg [6] and SimCLR [9] among the invariant self-supervised approaches; 2) EquiMOD [13], SEN [32] and SIE [15] amongst the equivariant baselines. We report the test performance on context lengths 0, 2, 14, 30, and 126. To assess the quality of the invariant representations, we employ linear classification over frozen features. For the equivariant counterpart, we report R^2 on the task of predicting the corresponding transformation. More details about pretraining algorithms and training setup are provided in Appendix C.

$\mathcal{G}$	Method	Rotation prediction (R^2)	Color prediction (R^2)	Classification (top-1)
	Invariant			
	SimCLR	0.506	0.148	**85.3**
	VICReg	0.371	0.023	76.3
Rotation	*Equivariant*	*Higher is better*	*Lower is better*	
	EquiMOD	0.512	0.097	**82.4**
	SIE	0.671	**0.011**	77.3
	SEN	0.633	0.055	81.5
	CONTEXTSSL, rot. context	**0.744**	0.023	80.4
Color		*Lower is better*	*Higher is better*	
	EquiMOD	0.429	0.859	**82.1**
	SIE	**0.304**	0.975	70.3
	SEN	0.386	0.949	77.6
	CONTEXTSSL, color context	0.344	**0.986**	80.4

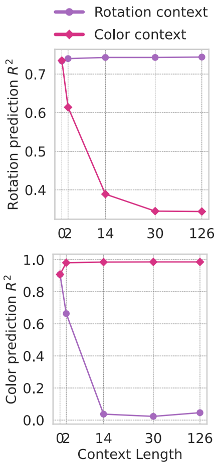


Figure 2: (Left) Quantitative evaluation on invariant (classification) and equivariant (rotation prediction, color prediction) tasks; (Right) Performance of CONTEXTSSL on equivariant (top) rotation prediction; (bottom) color prediction tasks with varying contexts and lengths. The algorithm increasingly demonstrates equivariance to rotation (color) as the rotation (color) context length increases while simultaneously becoming more invariant to color (rotation).

Invariant Classification and Equivariant transformation prediction task. As shown in Section 3.1, invariant self-supervised learning methods such as SimCLR and VICReg achieve high downstream classification accuracies but underperform in equivariant augmentation prediction tasks. Among the equivariant baselines, EquiMOD persistently maintains its downstream classification accuracy but exhibits improvements in augmentation prediction tasks only when trained to be equivariant to color. In contrast, CONTEXTSSL exhibits equivariance to both rotation and color in the absence of context. As seen from the two rows corresponding to CONTEXTSSL in Section 3.1, when the context corresponds to pairs of data with transformations sampled from the rotation (color) group, the model adaptively learns to be invariant to color (rotation) while improving equivariance to rotation (color). Appendix D.12 shows that CONTEXTSSL learns equivariance or invariance to the same transformation based on the context.

Table 1: Quantitative evaluation of learned predictors equivariant to only rotation based on Mean Reciprocal Rank (MRR) and Hit Rate H@k on the validation dataset. CONTEXTSSL learns to be more equivariant to rotation with context.

Method	MRR (↑)					H@1 (↑)					H@5 (↑)				
	0	2	14	30	126	0	2	14	30	126	0	2	14	30	126
EquiMOD			0.16					0.05					0.22		
SEN			0.17					0.05					0.22		
CONTEXTSSL	0.240	0.270	0.373	0.396	**0.402**	0.108	0.129	0.223	0.245	**0.292**	0.366	0.412	0.541	0.561	**0.568**

Equivariant Measures Based on Nearest Neighbours Retrieval. Table 1 illustrates the performance of CONTEXTSSL on MRR and H@k compared to baseline methods with trained equivariance to rotation. CONTEXTSSL outperforms the baseline models, and its performance on all the metrics consistently improves with increasing context length, showing adaptation to rotation-specific features.

Additional results. Additional results to understand the role of context mask, auxiliary predictor are shown in Figure 4 and Figure 5 respectively. Further, we demonstrate that CONTEXTSSL extends to naturally occurring symmetries and sensitive features in fairness and physiological datasets such as the MIMIC III [28] and UCI Adult [2]. Unlike 3DIEBench where meta-latents for each data are available, we manually construct positives by applying augmentations like crop and blur on CIFAR10. The results for the combinations of crop and blur are reported in Table 2. Results on additional transformation pairs are provided in Appendix D.9. Moreover, we explore broader applications of our algorithm, specifically for supervised learning and present these results in Table 5 and Appendix D.10. Related works and additional discussion about the algorithmic implications can be found in Appendix B.

Acknowledgement

This research was supported in part by Office of Naval Research grant N00014-20-1-2023 (MURI ML-SCOPE), NSF Award CCF-2112665 (TILOS AI Institute), and the Alexander von Humboldt Foundation. CW and TJ acknowledge support from NSF Expeditions grant (award 1918839: Collaborative Research: Understanding the World Through Code) and Machine Learning for Pharmaceutical Discovery and Synthesis (MLPDS) consortium. We acknowledge MIT SuperCloud and Lincoln Laboratory Supercomputing Center for providing HPC resources that have contributed to this work.

References

[1] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15619–15629, 2023. URL https://arxiv.org/abs/2301.08243.

[2] Arthur Asuncion, David Newman, et al. Uci machine learning repository, 2007.

[3] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32, 2019.

[4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *International Conference on Learning Representations*, 2022.

[5] Amir Bar, Yossi Gandelsman, Trevor Darrell, Amir Globerson, and Alexei Efros. Visual prompting via image inpainting. *Advances in Neural Information Processing Systems*, 35: 25005–25017, 2022.

[6] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2105.04906.

[7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 2020.

[8] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. URL https://arxiv.org/abs/2104.14294.

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. URL http://proceedings.mlr.press/v119/chen20j/chen20j.pdf.

[10] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020.

[11] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15750–15758, 2021.

[12] Rumen Dangovski, Li Jing, Charlotte Loh, Seungwook Han, Akash Srivastava, Brian Cheung, Pulkit Agrawal, and Marin Soljačić. Equivariant contrastive learning. In *International Conference on Learning Representations*, 2022.

[13] Alexandre Devillers and Mathieu Lefort. Equimod: An equivariance module to improve self-supervised learning. In *International Conference on Learning Representations*, 2023.

[14] Alaaeldin El-Nouby, Michal Klein, Shuangfei Zhai, Miguel Angel Bautista, Alexander Toshev, Vaishaal Shankar, Joshua M Susskind, and Armand Joulin. Scalable pre-training of large autoregressive image models. *arXiv preprint arXiv:2401.08541*, 2024.

[15] Quentin Garrido, Laurent Najman, and Yann Lecun. Self-supervised learning of split invariant equivariant representations. *International Conference on Machine Learning*, 2023. URL https://arxiv.org/pdf/2302.10283.pdf.

[16] Quentin Garrido, Mahmoud Assran, Nicolas Ballas, Adrien Bardes, Laurent Najman, and Yann LeCun. Learning and leveraging world models in visual representation learning. *arXiv preprint arXiv:2403.00504*, 2024.

[17] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *International Conference on Learning Representations*, 2018.

[18] Spyros Gidaris, Andrei Bursuc, Gilles Puy, Nikos Komodakis, Matthieu Cord, and Patrick Pérez. Obow: Online bag-of-visual-words generation for self-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6830–6840, 2021.

[19] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[20] Sharut Gupta, Stefanie Jegelka, David Lopez-Paz, and Kartik Ahuja. Context is environment. In *The Twelfth International Conference on Learning Representations*, 2023.

[21] Sharut Gupta, Joshua Robinson, Derek Lim, Soledad Villar, and Stefanie Jegelka. Structuring representation geometry with rotationally equivariant contrastive learning. *International Conference on Learning Representations*, 2023.

[22] David Ha and Jürgen Schmidhuber. World models. *Advances in Neural Information Processing Systems*, 2018.

[23] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *International Conference on Learning Representations*, 2020.

[24] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

[25] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[26] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[27] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

[28] A Johnson, T Pollard, and R Mark III. Mimic-iii clinical database (version 1.4). physionet. 2016, 2016.

[29] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 577–593. Springer, 2016.

[30] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *Annual Meeting of the Association for Computational Linguistics*, 2021.

[31] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6707–6717, 2020.

[32] Jung Yeon Park, Ondrej Biza, Linfeng Zhao, Jan Willem van de Meent, and Robin Walters. Learning symmetric embeddings for equivariant world models. *International Conference on Machine Learning*, 2022.

[33] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[34] Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *International conference on machine learning*, pages 8583–8592. PMLR, 2020.

[35] Ravid Shwartz-Ziv, Micah Goldblum, Hossein Souri, Sanyam Kapoor, Chen Zhu, Yann LeCun, and Andrew G Wilson. Pre-train your loss: Easy bayesian transfer learning with informative priors. *Advances in Neural Information Processing Systems*, 35:27706–27715, 2022.

[36] Nenad Tomasev, Ioana Bica, Brian McWilliams, Lars Buesing, Razvan Pascanu, Charles Blundell, and Jovana Mitrovic. Pushing the limits of self-supervised resnets: Can we outperform supervised learning without labels on imagenet? *arXiv preprint arXiv:2201.05119*, 2022.

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[38] Xinlong Wang, Wen Wang, Yue Cao, Chunhua Shen, and Tiejun Huang. Images speak in images: A generalist painter for in-context visual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6830–6839, 2023.

[39] Tete Xiao, Xiaolong Wang, Alexei A Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. In *International Conference on Learning Representations*, 2021.

[40] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9653–9663, 2022.

[41] Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. *International Conference on Learning Representations*, 2024.

[42] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR, 2021.

[43] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2111.07832.

[44] Pan Zhou, Yichen Zhou, Chenyang Si, Weihao Yu, Teck Khim Ng, and Shuicheng Yan. Mugs: A multi-granular self-supervised learning framework. *arXiv preprint arXiv:2203.14415*, 2022.

Appendix

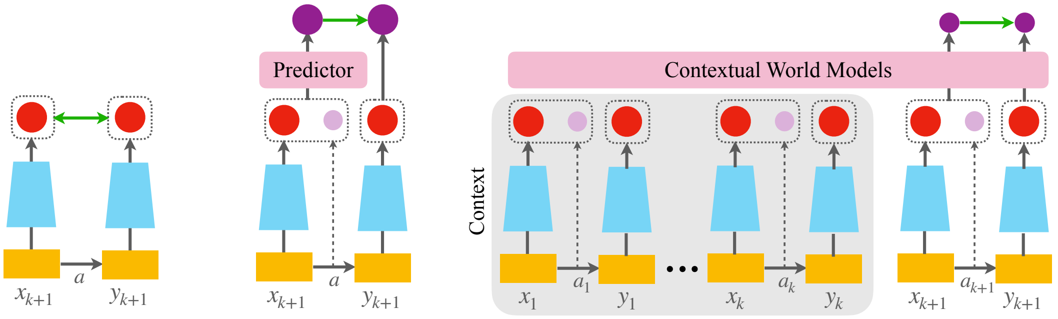
A Related Work

Self-Supervised Learning. Existing SSL methods generally belong to two categories: invariant learning [9, 6, 11, 25, 42, 19] and equivariant learning. The representative method for invariant learning is contrastive learning, which draws the representations of positive samples together in the latent space such that the representations are invariant to data augmentation. Contrastive learning can learn highly discriminative features at the cost of losing certain image information due to the invariance constraint [39]. Motivated by this limitation, recent works explore merging contrastive learning with equivariant learning tasks by separate embedding [39, 15], augmentation-conditioned predictor [13, 16], and explicit equivariant transformation [21]. However, existing works still inherit the limitations of contrastive learning: its symmetry prior is built on a given set of manual augmentations and is not adaptive to downstream tasks. In contrast, our method enables the contextual world model to adapt its symmetry to the contextual data, which is more flexible and generalizable to various tasks.

World Models. World modeling has achieved notable success in reinforcement learning (RL) for model-based planning [22, 34, 23] and vision [24, 27, 41], where it involves predicting future states based on current observations and actions. This concept, however, has not yet been fully leveraged in visual representation learning. Nevertheless, Garrido et al. [16] shows that several families of self-supervised learning approaches can be reformulated through the lens of world modeling. Equivariant self-supervised learning methods. Specifically, Masked Image Modeling approaches [26, 4, 14, 40] consider masked pixels and target pixel reconstruction as their action and next state. Other equivariant learning approaches [13, 32, 15] consider data transformations and representation of the target image as their action and next state pair. However, unlike true world modeling, these approaches do not track past experiences, a component critical for generalization. Our method instead leverages context to track past experiences in terms of state, action, and next-state triplets, enabling it to adapt and generalize to varying environments.

In-context Learning. Our work is inspired by and extends the concept of in-context learning (ICL) [7] to training. Initially studied in the context of language, in-context learning has recently been adapted for vision tasks [20, 38, 5, 30], allowing models to infer environmental features or tasks directly from input prompts without predefined notions. For example, Visual Prompting [38, 5] uses a task input/output example pair and a query image at test time, and uses inpainting to generate the desired output. Gupta et al. [20] propose using unlabeled data as context at training to extract environment-specific signals and address domain generalization. ICL has been extensively explored in various domains, including vision, language, and multimodal tasks. However, our work is the first to apply ICL to vision self-supervised representation learning.

B Additional Discussion and Future Perspectives



(a)

(b)

(c)

Figure 3: Family of approaches in self-supervised learning (a) **Joint Embedding** methods [9, 6, 8] encode invariances to input transformations a by aligning representations across views of the same image; (b) **Image World Models** [16, 1] train a world model in the latent space and encode equivariance to input transformations; (c) **Contextual World Models** (ours) selectively enforce equivariance or invariance to a subset of input transformations based on context $\{(x_i, a_i, y_i)\}_{i=1}^{k}$

The field of language modeling has witnessed a significant paradigm shift over the past decade, moving towards foundation models that generalize across a variety of tasks either directly or through distillation. However, this shift toward generalization has been conspicuously absent in the vision domain. This is largely because self-supervised approaches for vision still heavily rely on inductive priors strongly introduced by enforcing either invariance or equivariance to data augmentations. This renders representations brittle in downstream tasks that do not conform to these priors and necessitates retraining the representation separately for each task. This work forgoes any notion of pre-defined symmetries and instead trains a model to infer the task-relevant symmetries directly from the context through what we term Contextual Self-Supervised Learning (CONTEXTSSL). The ability of our model to learn selective equivariances and invariances based on mere context opens up new avenues for effectively handling a broader range of tasks, particularly in dynamic environments where the relevance of specific features may change over time. However, we limit our scope of symmetries to hand-crafted transformations in the data and do not explore naturally occurring symmetries. Nonetheless, CONTEXTSSL lays the groundwork for models that can potentially discern and adapt to the underlying patterns of tasks, recognize shortcuts, and more effectively generalize across unseen scenarios. Through this work, we hope to contribute to a broader understanding of how machines can learn more like humans — contextually, adaptively, and with an eye toward the infinite variability of the real world.

C Supplementary experimental details and assets disclosure

To evaluate the efficacy of our proposed algorithm CONTEXTSSL, our experiments are designed to address the following questions:

i) How does CONTEXTSSL fare against competitive invariant and equivariant self-supervised learning approaches in terms of performance across varying context sizes and different sets of data transformations?

ii) How effectively can CONTEXTSSL identify task-specific symmetries, both within the scope of self-supervised learning and beyond?

iii) What roles do specific components such as selective masking and the auxiliary latent transformation predictor play in facilitating the learning of general and context-adaptable representations?

C.1 Assets

We do not introduce new data in the course of this work. Instead, we use publicly available widely used image datasets for the purposes of benchmarking and comparison.

C.2 Hardware and setup

Each experiment was conducted on 1 NVIDIA Tesla V100 GPUs, each with 32GB of accelerator RAM. The CPUs used were Intel Xeon E5-2698 v4 processors with 20 cores and 384GB of RAM. All experiments were implemented using the PyTorch deep learning framework.

C.3 Datasets

3D Invariant Equivariant Benchmark (3DIEBench). To test equivariance and invariance to multiple data transformations, we use the 3D Invariant Equivariant Benchmark (3DIEBench) [15] which has been specifically designed to address the limitations of existing datasets in evaluating invariant and equivariant representations. It contains images of 3D objects along with their latent parameters such as object rotation, lighting color, and floor color. Since we have access available to individual meta latent parameters, transformation parameters between two views of an object are calculated as the difference between their individual latents. We test our approach on 3DIEBench under two settings 1) Considering two transformation groups: rotation and color with the aim of learning invariance to one and equivariance to another after conditioning on context; 2) Considering one transformation group, say rotation and learning to enforce invariance or equivariance to rotation with context. As previously mentioned, all methods are trained for 1000 epochs using a batch size of 512 on 128×128 resolution images. We use the standard training, validation and test splits, made publicly available by the authors [15].

CIFAR10. 3DIEBench dataset is limited to only rotations and color as transformation groups. We extend our approach to include more common self-supervised benchmarks, such as CIFAR-10, incorporating transformations like blurring, color jitter, and cropping. Unlike 3DIEBench, we manually construct positive pairs by applying compositions of these handcrafted augmentations. We consider three transformation groups: crop, blur and color. Similar to 3DIEBench, we consider combinations of two groups for each training run. We use the standard training, validation and test splits.

C.4 Baseline Algorithms

Among the invariant self-supervised approached, we compare our approach to VICReg [6] and and SimCLR [9]. For each method, comparisons are drawn using their originally proposed architectures. For the equivariant baselines, we consider EquiMOD [13], SIE [15] and SEN [32]. Similar to Garrido et al. [15], For SEN, we use the InfoNCE loss instead the original triplet loss. To discard the performance gains potentially arising from CONTEXTSSL's transformer architecture, for each approach, we consider an additional baseline that replaces the original projection heads or predictor with our transformer model. Given an algorithm name $\mathcal{N}$, we refer to this baseline as $\mathcal{N}^+$. Amongst these, we report the best performing variant in our results. For $\mathcal{N}^+$, we conduct analysis in two distinct settings: 1) a 'no context' or $c = 0$ invariant condition, and 2) a fully contextualized setting with a context length of 126.

C.5 Training Protocol

To ensure a fair comparison across different algorithms for each dataset, we use a standardized neural network backbone. Precisely, for our encoder, we use a ResNet-18 backbone pre-trained on ImageNet. For CONTEXTSSL, output features from the encoder are transformed into the context sequence, which is then processed by the decoder-only Transformer [37] from the GPT-2 Transformer family [33]. Our model configuration includes 3 layers, 4 attention heads, and a 2048-dimensional embedding space, consistently applied across all datasets. Linear layers are utilized to convert the input sequence into the transformer's latent embedding of dimension 2048 and to map the predicted output vectors to the output space of dimension 512.

We fix the maximum training context length to 128. Since for every y, the corresponding token (x_i, a_i) is masked out, context length L corresponds to effective context length $L-2$. Thus, we report CONTEXTSSL's performance over varying test context length of 0, 2, 14, 30 and 126. On all datasets, we train CONTEXTSSL with the Adam optimizer with a learning rate of $5e^{-5}$ and weight decay $1e{-}3$. For baseline self-supervised approaches, in their original architecture, we use a learning rate of $1e^{-3}$ with no weight decay. However, when tested using the transformer architecture, we choose one of the above two optimizer hyperpameters. Consequently, performance of the best performing model is reported among the two baselines. Similar to Garrido et al. [15], we report hyper-parameters and architectures specific to each method:

- **SimCLR [9]** We train using a 2048-2048-2048 dimensional multi-layered perceptron (MLP) based projection head with a temperature of 0.5.
- **VICReg [6]** We train using a 2048-2048-2048 MLP for the projection head and use weight of 10 for both the invariance loss and variance loss and 1 for covariance loss.
- **SEN [32]** Similar to other approaches we use a projection head of dimension 2048-2048-2048 and temperature 0.1.
- **EquiMod [13]** We use the standatd projection head of dimensions 1024-1024-128 and use equal weighing of the invariance and the equivariance loss.
- **SIE** [15] We use two 1024-1024-1024 projection heads, one for invariant latent space and other for equivariant. When trained to learn equivariance to only rotation or only color, we use weight of 10 for both the invariance loss and variance loss, 1 for the covariance loss and 4.5 for the equivariant loss. However, when trained to be equivariant to both rotation and color jointly, we use 10 as the equivariant weight.

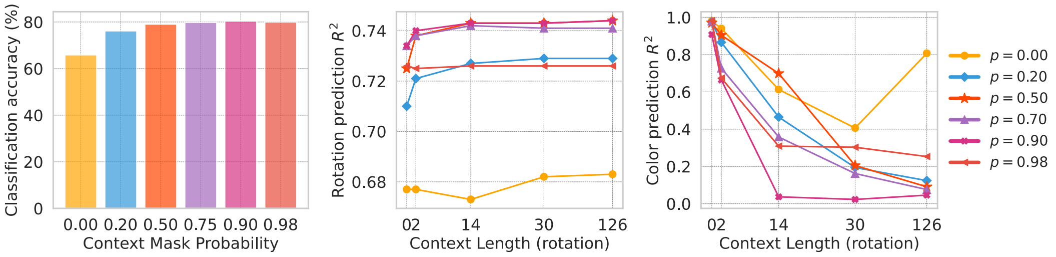
C.6 Evaluation metrics

In line with established self-supervised learning methodologies, we begin by assessing the quality of the learned representations through downstream tasks. For evaluating invariant representations, we employ linear classification over frozen features. To evaluate equivariant representations, we predict the corresponding data transformation. This prediction takes representations from two differently transformed views of the same object and regresses on the applied transformation between them. Further, we use Mean Reciprocal Rank (MRR) and Hit Rate at k (H@k) to evaluate the performance for our context predictor. Given the source data and the transformation action, we identify the k nearest neighbors in the embedding space. MRR is calculated as the average reciprocal rank of the target embedding within these nearest neighbors. Hit rate-k (H@k) assigns a score of 1 if the target embedding is within the k-nearest neighbors of the predicted embedding and 0 otherwise. Similar to Garrido et al. [15], we restrict the search for nearest neighbors to different views of the same object, thus ensuring that the predictor is not penalized for retrieving an incorrect object in a pose similar to the correct one.

D Additional Experiments

D.1 Role of Context Mask and Auxiliary Predictor

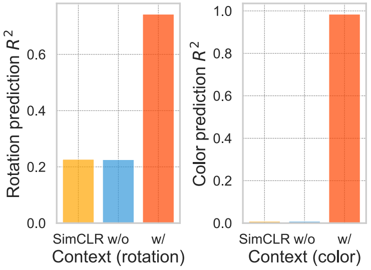
Role of Context Mask. To illustrate how context masking effectively eliminates shortcuts, we conduct an ablation study with varying masking probabilities, detailed in Figure 4. We observed that as masking probability increases, performance on both classification and prediction tasks initially improves but later declines, reaching optimal performance at a masking probability of 90%.



(a) (b) (c)

Figure 4: Role of context mask to avoid context based shortcuts in CONTEXTSSL

Role of Auxiliary Predictor. We demonstrate that the auxiliary predictor is crucial for the model to achieve equivariance. In its absence, as depicted in Figure 5, while the model retains its performance on the invariant classification task, it fails to learn equivariance, and cannot effectively adapt to different contexts.



(a) (b)

Figure 5: Role of auxiliary predictor to avoid the trivial solution of invariance.

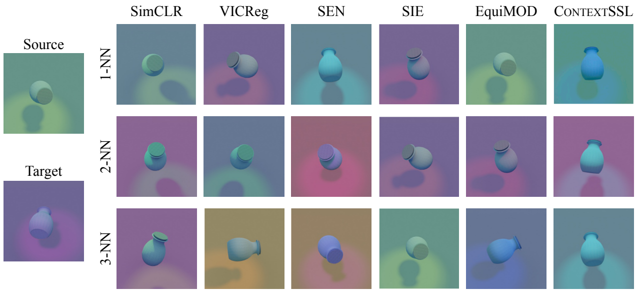


Figure 6: Nearest neighbors of different methods taking as input the source image and rotation angle. CONTEXTSSL aligns best with the rotation angle of the target image.

D.2 Qualitative Assessment of Adaptation to Task-Specific Symmetries

We conduct a qualitative assessment of model performance by taking the nearest neighbors of the predictor output when inputting a source image and a transformation variable, as shown in Figure 6. The nearest neighbors of invariance models (SimCLR and VICReg) have random rotation angles. Equivariance baselines (SEN, SIE, EquiMOD) correctly generate the target rotation angle for some of the 3-nearest neighbors but fail in others. CONTEXTSSL outperforms by successfully identifying the correct angle in all 3-nearest neighbors while remaining invariant to color variations. Additional qualitative assessments for CONTEXTSSL with varying context are provided in Appendix D.8.

D.3 Expanding to Diverse Data Transformations

Unlike 3DIEBench where meta-latents for each data are available, we manually construct positives by applying augmentations like crop and blur on CIFAR10. The results for the combinations of crop and blur are reported in Table 15. Consistent with our previous results, while almost retaining the classification performance as SimCLR, ContextSSL learns to adaptively enforce equivariance to crop (blur) and invariance to blur (crop) depending upon the context. Note that the invariance performance initially improves with increasing context length but then diminishes. This occurs due to the 90% random masking ratio during training, which necessitates out-of-distribution generalization when the context length is large. Results on additional transformation pairs are provided in Appendix D.9.

Table 2: Performance of ContextSSL on invariant (classification) and equivariant (crop prediction, blur prediction) tasks in CIFAR-10 under the environment of crop, i.e. ContextSSL (crop), and blur, i.e. ContextSSL (blur).

Method	Crop prediction (R^2)					Blur prediction (R^2)					Classification (top-1)
	0	2	14	30	126	0	2	14	30	126	Representation
SimCLR			0.459					0.371			89.1
SimCLR$^+$ (c=0)			0.448					0.361			88.9
SimCLR$^+$			0.362					0.444			59.9
ContextSSL (crop)	0.608	0.607	0.607	0.608	0.608	0.920	0.854	0.624	0.667	0.694	88.5
ContextSSL (blur)	0.609	0.482	0.434	0.417	0.465	0.920	0.923	0.925	0.925	0.925	88.5

D.4 Context World Models Beyond Self-Supervised Learning

While our analysis has primarily focused on self-supervised learning, the concept of context is versatile and extends beyond representation learning. In principle, irrespective of the task at hand, paying attention to context can learn and identify features defined by it. To validate this and explore broader applications of our algorithm, we consider a supervised learning task where our transformer model is trained to directly predict the labels corresponding to an input image. We further corrupt the labels to be directly influenced by the augmentation group transforming the data. Specifically, for 3DIEBench dataset, we add a constant value of 10 to each label if the context corresponds to the rotation group and leave it unchanged otherwise. We report classification performance along with rotation and color prediction equivariant measures. As shown in Table 5, ContextSSL's classification accuracy improves with context, demonstrating its ability to better identify the underlying symmetry group with increase in context. Additional results are provided in Appendix D.10. Further, ContextSSL serves as a general framework that can adapt to different training regimes such as supervised learning.

D.5 ContextSSL on Naturally Occurring Symmetries

We show that ContextSSL extends to naturally occurring symmetries and sensitive features in fairness and physiological datasets such as the MIMIC III [28] and UCI Adult [2]. To demonstrate this, we train ContextSSL to be selectively equivariant or invariant to gender by merely attending to different contexts. This is crucial; for instance, equivariance is needed for gender-specific medical diagnoses where different medicine dosages are required, while invariance is essential for fairness in tasks such as predicting hospital stay duration or medical cost. We present these results in Table 3 and Table 4, with details in the caption. From Table 3 , we can observe that ContextSSL learns equivariance to gender in one context, improving gender and medical diagnosis prediction for MIMIC-III. In another context, ContextSSL achieves higher invariance to gender, resulting in superior performance on fairness metrics like equalized odds (EO) and equality of opportunity (EOPP) for hospital stay (LOS) prediction. We observe similar results for fairness of income prediction in the UCI Adult dataset, as shown in Table 4 of the attached document.

D.6 Quantitative Assessment of Adaptation to Task-Specific Symmetries

In this section, we present additional results on the quantitative assessment of model performance on 3DIEBench, including the evaluation of learned representations on equivariant tasks (rotation and color prediction) to predict individual latent values. In contrast, the results in Section 3.1 focus on predicting relative latent values between pairs of image embeddings as inputs.

Table 3: Performance of ContextSSL on the MIMIC III dataset [28]. For each data point x_i, we create the transformed data y_i by flipping the value of gender. For this experiment, equivariance is needed for gender-specific medical diagnoses where different medicine dosages are required, while invariance is essential for fairness in tasks such as predicting hospital stay duration or medical cost. We observe that when the environment is equivariant to gender, both gender prediction and medical treatment prediction improve with context. When the environment is invariant, embedding fairness of hospital stay (LOS) prediction as measured by equalized odds (EO) and equality of opportunity (EOPP), improves with context.

$\mathcal{G}$	Gender prediction Acc ↑		LOS prediction Acc ↑		Equalized odds ↓		Equality of opportunity ↓		Treatment prediction Acc ↑	
Context Length	0	126	0	126	0	126	0	126	0	126
Equivariant	0.969	0.991	0.942	0.944	0.028	0.035	0.023	0.031	0.333	0.344
Invariant	0.969	0.626	0.942	0.943	0.028	0.023	0.023	0.004	0.333	0.316

Table 4: Performance of ContextSSL on the UCI Adult [2] dataset. For each data point x_i, we create the transformed data y_i by flipping the value of gender. When the environment is equivariant to gender, both gender prediction and income prediction improve with context. When the environment is invariant, embedding fairness of income prediction measured by equalized odds (EO) and equality of opportunity (EOPP), improves with context.

$\mathcal{G}$	Gender prediction Acc ↑		Income prediction AUC ↑		Equalized odds ↓		Equality of opportunity ↓	
Context Length	0	126	0	126	0	126	0	126
Equivariant	0.985	0.999	0.900	0.900	0.114	0.130	0.061	0.101
Invariant	0.985	0.605	0.900	0.899	0.114	0.066	0.061	0.047

Table 5: Performance of ContextSSL on equivariant tasks (including classificaion) for context-dependent labels. ContextSSL adapts to context-dependent labels with varying context.

Method	Rotation prediction (R^2)					Color prediction (R^2)					Classification (top-1)				
	0	2	14	30	126	0	2	14	30	126	0	2	14	30	126
SimCLR (color)			0.537					0.056					72.0		
SimCLR (rotation)			0.537					0.056					14.2		
SimCLR$^+$ (c=0) (color)			0.427					-0.007					80.4		
SimCLR$^+$ (c=0) (rotation)			0.427					-0.007					5.2		
SimCLR$^+$ (color)			0.424					0.243			16.8	15.1	15.6	14.8	14.0
SimCLR$^+$ (rotation)			0.424					0.243			56.1	58.2	58.4	58.4	59.1
ContextSSL (color)	0.556	0.542	0.538	0.540	0.539	0.913	0.973	0.981	0.982	0.982	8.9	82.4	82.7	82.8	83.0
ContextSSL (rotation)	0.556	0.624	0.661	0.665	0.666	0.913	0.379	0.111	0.095	0.093	73.5	82.7	82.6	82.6	83.0

D.6.1 Invariant Classification and Equivariant transformation prediction task

As shown in Table 6, invariant self-supervised learning methods such as SimCLR and VICReg underperform in equivariant augmentation prediction tasks. The equivariant baselines, EquiMOD, SIE, and SEN, exhibit improvements compared to the invariant baselines in some of the augmentation prediction tasks. However, their degree of equivariance is much worse compared to ContextSSL. Besides, aligning them with different targeted symmetry groups requires retraining the entire model. In contrast, ContextSSL employs a single model capable of learning equivariance to rotation and invariance to color (or vice versa) based on the given context. As seen from the two rows corresponding to ContextSSL Section 3.1, when the context corresponds to pairs of data with transformations sampled from the rotation (color) group, the model adaptively learns to be invariant to color (rotation) while retaining equivariance to rotation (color).

Results in Section 3.1 are the average value over three random seeds. We provide the standard deviation for rotation and color prediction of ContextSSL in Table 7 and Table 8.

D.6.2 Equivariant Measures Based on Nearest Neighbours Retrieval

Similar to Table 1, we provide the performance of ContextSSL on MRR and H@k compared to baseline methods with trained equivariance to rotation. While Table 1 uses the validation set data as the retrieval library, Table 9 provides the results using the training set data. ContextSSL outperforms the baseline models, and its performance on all the metrics consistently improves with increasing context length, showing adaptation to rotation-specific features.

Table 6: Quantitative evaluation of learned representations on equivariant (rotation prediction, color prediction) tasks to predict individual latent values.

$\mathcal{G}$	Method	Rotation prediction (R^2)					Color prediction (R^2)				
		0	2	14	30	126	0	2	14	30	126
	Invariant										
	SimCLR			0.791					0.137		
	SimCLR$^+$(c=0)			0.773					0.061		
	SimCLR$^+$			0.544					0.498		
	VICReg			0.660					0.011		
	VICReg$^+$(c=0)			0.615					0.061		
Rotation + Color	*Equivariant*										
	EquiMOD			0.712					0.221		
	SIE			**0.760**					**0.972**		
	SEN			0.617					0.888		
Rotation	EquiMOD			0.707					0.033		
	SIE			0.790					**0.001**		
	SEN			0.723					0.437		
	ContextSSL[1]	0.838	0.839	0.840	0.840	**0.840**	0.895	0.620	0.021	0.014	0.021
Color	EquiMOD			0.660					0.855		
	SIE			**0.560**					0.974		
	SEN			0.713					0.876		
	ContextSSL[2]	0.838	0.800	0.699	0.666	0.685	0.895	0.981	0.985	0.985	**0.986**

Table 7: Performance of ContextSSL in 3DIEBench in rotation prediction under the environment of rotation, i.e. ContextSSL (rotation), and color, i.e. ContextSSL (color), with standard deviations over three random seeds.

Method	Rotation prediction (R^2)				
	0	2	14	30	126
ContextSSL (rotation)	0.734 ± 0.002	0.740 ± 0.004	0.743 ± 0.001	0.743 ± 0.001	0.744 ± 0.001
ContextSSL (color)	0.735 ± 0.001	0.614 ± 0.108	0.389 ± 0.054	0.345 ± 0.040	0.344 ± 0.003

Table 8: Performance of ContextSSL in 3DIEBench in color prediction under the environment of rotation, i.e. ContextSSL (rotation), and color, i.e. ContextSSL (color), with standard deviations over three random seeds.

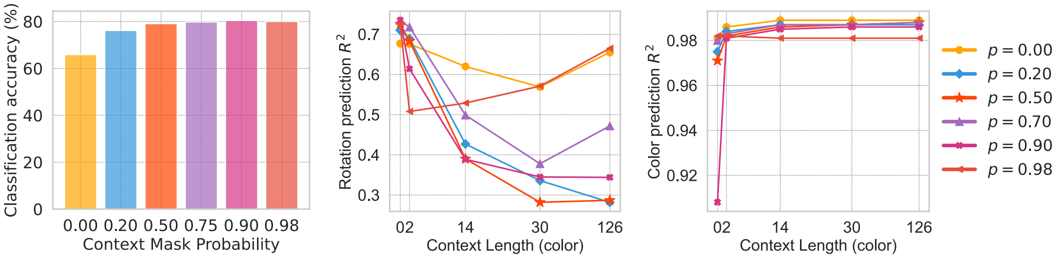
Method	Color prediction (R^2)				
	0	2	14	30	126
ContextSSL (rotation)	0.908 ± 0.002	0.664 ± 0.166	0.037 ± 0.010	0.023 ± 0.001	0.046 ± 0.007
ContextSSL (color)	0.908 ± 0.002	0.981 ± 0.002	0.985 ± 0.001	0.986 ± 0.001	0.986 ± 0.001

Table 9: Quantitative evaluation of learned predictors equivariant to only rotation based on Mean Reciprocal Rank (MRR) and Hit Rate H@k on training dataset. ContextSSL learns to be more equivariant to rotation with context.

Method	MRR (↑)					H@1 (↑)					H@5 (↑)				
	0	2	14	30	126	0	2	14	30	126	0	2	14	30	126
EquiMOD			0.17					0.06					0.24		
SEN			0.17					0.06					0.24		
ContextSSL	0.282	0.321	0.470	0.498	**0.531**	0.132	0.263	0.375	0.398	**0.402**	0.436	0.495	0.650	0.669	**0.680**

D.7 Role of Context Mask and Auxiliary Predictor

In this section, we provide additional results for the role of context mask and auxiliary predictor.



(a) (b) (c)

Figure 7: Role of context mask to avoid context based shortcuts in CONTEXTSSL under color context

D.7.1 Role of Context Mask

In addition to Figure 4, we provide the performance of the rotation and color prediction tasks with varying masking probabilities under the environment of color in Figure 7. We observed that as masking probability increases, performance on both classification and prediction tasks initially improves but later declines, reaching optimal performance at a masking probability of 90%.

Results in Figure 4 and Figure 7 are the average value over three random seeds. We provide the standard deviation for rotation and color prediction of CONTEXTSSL in Table 10 and Table 11.

Table 10: Performance of CONTEXTSSL rotation prediction tasks in 3DIEBench under different random masking probabilities, with standard deviations over three random seeds.

Context	Probability	Rotation prediction (R^2)				
		0	2	14	30	126
Rotation	0.00	0.677 ± 0.004	0.677 ± 0.002	0.673 ± 0.009	0.682 ± 0.003	0.683 ± 0.003
	0.20	0.710 ± 0.002	0.721 ± 0.006	0.727 ± 0.002	0.729 ± 0.001	0.729 ± 0.001
	0.50	0.725 ± 0.001	0.738 ± 0.005	**0.743** ± **0.001**	**0.743** ± **0.001**	**0.744** ± **0.001**
	0.75	**0.734** ± **0.002**	0.738 ± 0.006	0.742 ± 0.004	0.741 ± 0.004	0.741 ± 0.002
	0.90	**0.734** ± **0.002**	**0.740** ± **0.004**	**0.743** ± **0.001**	**0.743** ± **0.001**	**0.744** ± **0.001**
	0.98	0.726 ± 0.002	0.725 ± 0.003	0.726 ± 0.002	0.726 ± 0.003	0.726 ± 0.003
Color	0.00	**0.677** ± **0.004**	0.676 ± 0.005	0.620 ± 0.019	0.569 ± 0.019	0.655 ± 0.010
	0.20	0.710 ± 0.002	0.689 ± 0.013	0.427 ± 0.031	0.336 ± 0.007	**0.282** ± **0.022**
	0.50	0.725 ± 0.001	0.683 ± 0.006	0.390 ± 0.031	**0.282** ± **0.013**	0.287 ± 0.002
	0.75	0.734 ± 0.002	0.718 ± 0.002	0.499 ± 0.035	0.378 ± 0.054	0.472 ± 0.015
	0.90	0.735 ± 0.001	0.614 ± 0.108	**0.389** ± **0.054**	0.345 ± 0.040	0.344 ± 0.003
	0.98	0.726 ± 0.002	**0.508** ± **0.127**	0.529 ± 0.141	0.571 ± 0.125	0.665 ± 0.023

Table 11: Performance of CONTEXTSSL color prediction tasks in 3DIEBench under different random masking probabilities, with standard deviations over three random seeds.

Context	Probability	Color prediction (R^2)				
		0	2	14	30	126
Rotation	0.00	0.981 ± 0.002	0.940 ± 0.033	0.613 ± 0.123	0.406 ± 0.125	0.807 ± 0.080
	0.20	0.975 ± 0.001	0.866 ± 0.171	0.465 ± 0.113	0.194 ± 0.057	0.124 ± 0.027
	0.50	0.971 ± 0.002	0.904 ± 0.086	0.699 ± 0.028	0.205 ± 0.054	0.091 ± 0.016
	0.75	0.980 ± 0.001	0.727 ± 0.351	0.358 ± 0.233	0.162 ± 0.021	0.076 ± 0.009
	0.90	**0.908** ± **0.002**	**0.664** ± **0.166**	**0.037** ± **0.010**	**0.023** ± **0.001**	**0.046** ± **0.007**
	0.98	0.982 ± 0.001	0.674 ± 0.368	0.309 ± 0.139	0.303 ± 0.118	0.253 ± 0.033
Color	0.00	0.981 ± 0.002	**0.986** ± **0.002**	**0.989** ± **0.001**	**0.989** ± **0.001**	**0.989** ± **0.001**
	0.20	0.975 ± 0.001	0.984 ± 0.002	0.987 ± 0.001	0.987 ± 0.001	0.987 ± 0.001
	0.50	0.971 ± 0.002	0.982 ± 0.002	0.986 ± 0.002	0.987 ± 0.002	0.988 ± 0.001
	0.75	0.980 ± 0.001	0.983 ± 0.001	0.987 ± 0.001	0.987 ± 0.001	0.988 ± 0.001
	0.90	0.908 ± 0.002	0.981 ± 0.002	0.985 ± 0.001	0.986 ± 0.001	0.986 ± 0.001
	0.98	**0.982** ± **0.001**	0.982 ± 0.001	0.981 ± 0.001	0.981 ± 0.001	0.981 ± 0.001

D.7.2 Role of Auxiliary Predictor

We provide the complete results corresponding to Figure 5 in Table 12 to demonstrate that the auxiliary predictor is crucial for the model to achieve equivariance. In its absence, while the model retains its performance on the invariant classification task, it fails to learn equivariance, performs similarly to the invariant models, and cannot effectively adapt to different contexts.

Table 12: Performance of CONTEXTSSL on classification, rotation and color prediction tasks in 3DIEBench with and without the auxiliary predictor

Method	Rotation prediction (R^2)					Color prediction (R^2)					Classification (top-1)
	0	2	14	30	126	0	2	14	30	126	Representation
SimCLR			0.227					-0.004			85.3
SimCLR$^+$ (c=0)			0.230					-0.004			83.4
SimCLR$^+$			0.245					0.028			42.3
CONTEXTSSL (w/o) (rotation)	0.227	0.227	0.226	0.226	0.227	-0.003	-0.003	-0.003	-0.004	-0.004	80.8
CONTEXTSSL (w/o) (color)	0.227	0.227	0.226	0.226	0.227	-0.003	-0.003	-0.003	-0.004	-0.004	80.8
CONTEXTSSL (rotation)	0.734	0.740	0.743	0.743	0.744	0.908	0.664	0.037	0.023	0.046	80.4
CONTEXTSSL (color)	0.735	0.614	0.389	0.345	0.344	0.908	0.981	0.985	0.986	0.986	80.4

D.8 Qualitative Assessment of Adaptation to Task-Specific Symmetries

D.8.1 Comparison with Baseline Approaches

We provide additional results to the qualitative assessment comparing with different models in Figure 8. The nearest neighbors of invariance models (SimCLR and VICReg) have random rotation angles. Equivariance baselines (SEN, SIE, EquiMOD) correctly generate the target rotation angle for some of the 3-nearest neighbors but fail in others. CONTEXTSSL outperforms by successfully identifying the correct angle in all 3-nearest neighbors while remaining invariant to color variations.

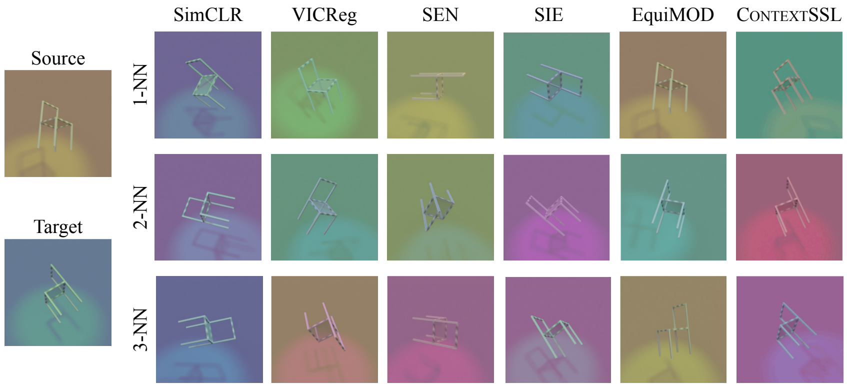


Figure 8: Nearest neighbors of different methods taking as input the source image and rotation angle. CONTEXTSSL aligns best with the rotation angle of the target image.

D.8.2 Nearest Neighbour Retrieval with Varying Context

In this section, we conduct a qualitative assessment of model performance by taking the nearest neighbors of the predictor output when inputting a source image and a transformation variable, and show the change in retrieving quality in Figure 9, Figure 10, and Figure 11. We observe that the nearest neighbors have a closer rotation angle (color) to the target image under rotation (color) context as context length increases, indicating CONTEXTSSL's ability to adapt to the given context as context length increases.

D.9 Expanding to Diverse Data Transformations

Unlike 3DIEBench where meta-latents for each data are available, we manually construct positives by applying augmentations like crop and blur on CIFAR10. The results for the combinations of crop and blur are reported in Table 15. We additionally provide the results for the combinations of crop and

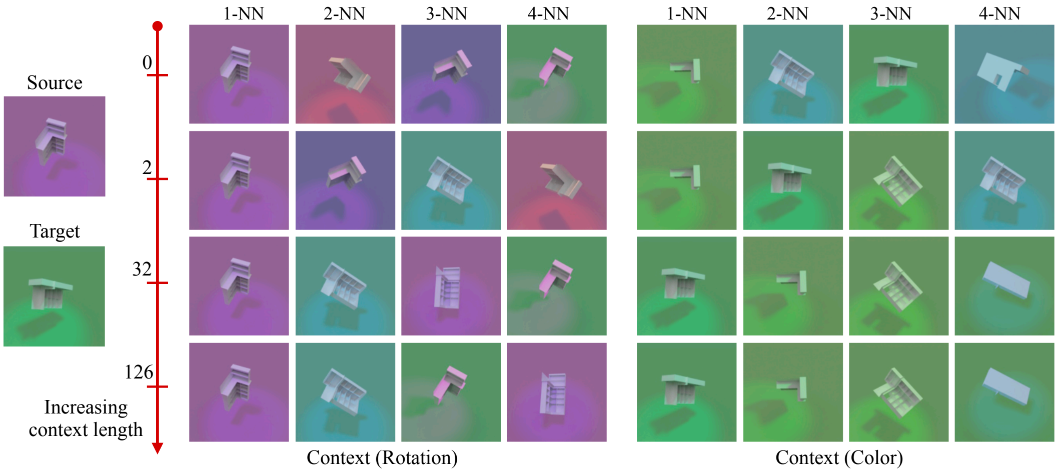


Figure 9: Nearest neighbors of CONTEXTSSL taking as input the source image and rotation angle at different context lengths. As context increases, CONTEXTSSL aligns better with the rotation angle (color) of the target image when the context is based on rotation (color).

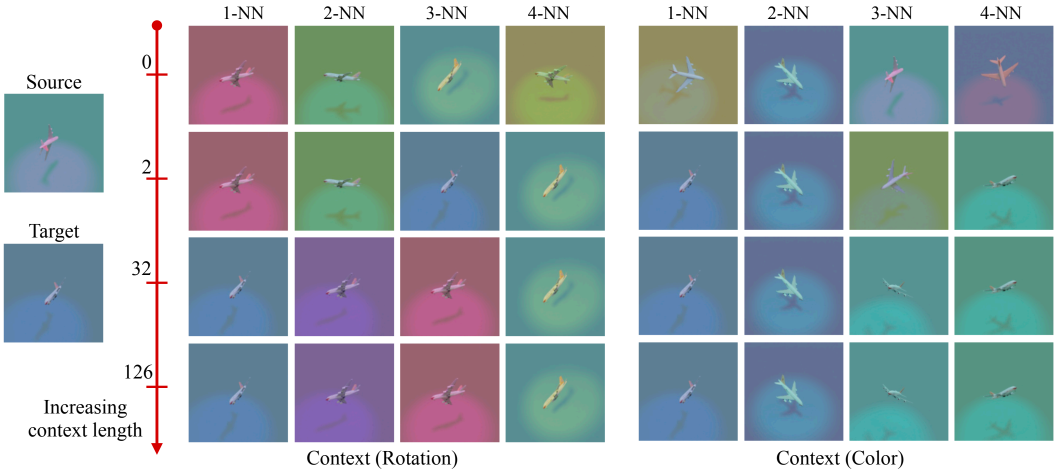


Figure 10: Nearest neighbors of CONTEXTSSL taking as input the source image and rotation angle at different context lengths. As context increases, CONTEXTSSL aligns better with the rotation angle (color) of the target image when the context is based on rotation (color).

color in Table 14 and crop and blur in Table 15. Consistent with our previous results, while almost retaining the classification performance as SimCLR, CONTEXTSSL learns to adaptively enforce equivariance and invariance to different environments depending upon the context.

Table 13: **CIFAR-10 Color-Blur.** Performance of CONTEXTSSL on invariant (classification) and equivariant (color prediction, blur prediction) tasks in CIFAR-10 under the environment of color, i.e. CONTEXTSSL (color), and blur, i.e. CONTEXTSSL (blur).

Method	Color prediction (R^2)					Blur prediction (R^2)					Classification (top-1)
	0	2	14	30	126	0	2	14	30	126	Representation
SimCLR			0.154					0.371			89.1
SimCLR$^+$ (c=0)			0.054					0.361			88.9
SimCLR$^+$			0.318					0.444			59.9
CONTEXTSSL (color)	0.518	0.519	0.519	0.519	0.519	0.916	0.793	0.699	0.735	0.823	88.9
CONTEXTSSL (blur)	0.518	0.353	0.241	0.259	0.333	0.916	0.916	0.916	0.916	0.917	88.8

In addition to the results for predicting relative latent values between pairs of image embeddings as input in Table 15, Table 14, and Table 13, we provide the evaluation of learned representations on

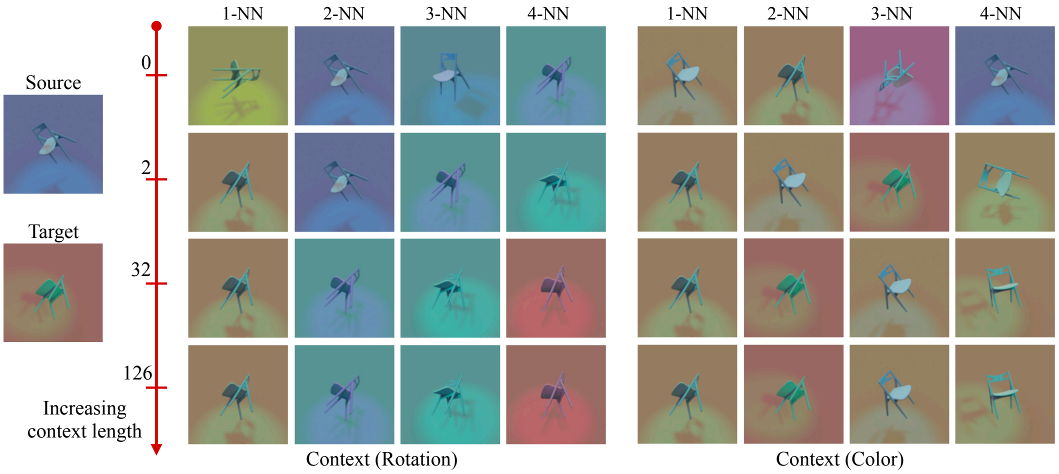


Figure 11: Nearest neighbors of ContextSSL taking as input the source image and rotation angle at different context lengths. As context increases, ContextSSL aligns better with the rotation angle (color) of the target image when the context is based on rotation (color).

Table 14: **CIFAR-10 Crop-Color.** Performance of ContextSSL on invariant (classification) and equivariant (crop prediction, color prediction) tasks in CIFAR-10 under the environment of crop, i.e. ContextSSL (crop), and color, i.e. ContextSSL (color).

Method	Crop prediction (R^2)					Color prediction (R^2)					Classification (top-1)
	0	2	14	30	126	0	2	14	30	126	Representation
SimCLR			0.459					0.154			89.1
SimCLR$^+$ (c=0)			0.448					0.054			88.9
SimCLR$^+$			0.362					0.318			59.9
ContextSSL (crop)	0.606	0.606	0.607	0.607	0.607	0.522	0.378	0.253	0.264	0.301	87.5
ContextSSL (color)	0.605	0.467	0.387	0.466	0.511	0.523	0.525	0.527	0.527	0.527	87.5

equivariant tasks (rotation and color prediction) to predict individual latent values, as shown in **??**, Table 17, and Table 16 respectively. Both results lead to the same conclusion, that ContextSSL is able to adaptively enforce equivariance and invariance to different environments depending upon the context.

Table 15: **CIFAR-10 Crop-Blur.** Performance of ContextSSL on equivariant (crop prediction, blur prediction) tasks in CIFAR-10 under the environment of crop, i.e. ContextSSL (crop), and blur, i.e. ContextSSL (blur), to predict individual latent values.

Method	Crop prediction (R^2)					Blur prediction (R^2)				
	0	2	14	30	126	0	2	14	30	126
SimCLR			0.382					0.122		
SimCLR$^+$ (c=0)			0.375					0.111		
SimCLR$^+$			0.202					0.322		
ContextSSL (crop)	0.576	0.575	0.576	0.576	0.576	0.835	0.795	0.630	0.644	0.663
ContextSSL (blur)	0.575	0.504	0.463	0.443	0.474	0.835	0.835	0.836	0.837	0.837

D.10 Context World Models Beyond Self-Supervised Learning

We report classification performance along with rotation and color prediction equivariant measures. The results for predicting relative values are shown in Table 5 and the results for predicting individual latent values are shown in Table 18. The equivariance (invariance) performance of ContextSSL improves with increased context.

Table 16: **CIFAR-10 Color-Blur.** Performance of ContextSSL on equivariant (color prediction, blur prediction) tasks in CIFAR-10 under the environment of color, i.e. ContextSSL (color), and blur, i.e. ContextSSL (blur), to predict individual latent values.

Method	Color prediction (R^2)					Blur prediction (R^2)				
	0	2	14	30	126	0	2	14	30	126
SimCLR			0.121					0.122		
SimCLR$^+$ (c=0)			0.039					0.111		
SimCLR$^+$			0.242					0.322		
ContextSSL (color)	0.488	0.488	0.488	0.488	0.488	0.837	0.711	0.628	0.672	0.730
ContextSSL (blur)	0.488	0.376	0.286	0.309	0.362	0.837	0.838	0.838	0.838	0.837

Table 17: **CIFAR-10 Crop-Blur.** Performance of ContextSSL on equivariant (crop prediction, color prediction) tasks in CIFAR-10 under the environment of crop, i.e. ContextSSL (crop), and color, i.e. ContextSSL (color), to predict individual latent values.

Method	Crop prediction (R^2)					Color prediction (R^2)				
	0	2	14	30	126	0	2	14	30	126
SimCLR			0.382					0.121		
SimCLR$^+$ (c=0)			0.375					0.039		
SimCLR$^+$			0.202					0.242		
ContextSSL (crop)	0.570	0.572	0.572	0.572	0.572	0.495	0.417	0.342	0.356	0.373
ContextSSL (color)	0.570	0.490	0.447	0.492	0.515	0.495	0.496	0.497	0.497	0.497

Table 18: **Context-Dependent Labels Classification Task.** Performance of ContextSSL on equivariant (rotation prediction, color prediction) tasks for context-dependent labels to predict individual latent values. As context length increases, ContextSSL becomes more equivariant to color (or rotation) and more invariant to rotation (or color) within the respective environment.

Method	Rotation prediction (R^2)					Color prediction (R^2)				
	0	2	14	30	126	0	2	14	30	126
SimCLR			0.781					0.058		
SimCLR$^+$ (c=0)			0.478					-0.003		
SimCLR$^+$			0.695					0.267		
ContextSSL (color)	0.751	0.751	0.750	0.750	0.749	0.915	0.973	0.980	0.981	0.981
ContextSSL (rotation)	0.750	0.778	0.797	0.795	0.795	0.915	0.375	0.104	0.091	0.090

D.11 Performance on Encoder Representations and Predictor Embedding

We analyze the difference between the performance on representation and the performance on predictor embedding for both the invariance (classification) task and equivariance (rotation prediction) task in Table 19 and Table 20. ContextSSL maintains almost the same performance for rotation prediction using either representations or embeddings, while the performance of all other baselines drops significantly when using the embeddings. Similar conclusions apply to the classification case, except for SimCLR$^+$, for which the classification accuracy for both representations and embeddings is low.

D.12 Enforcing Invariance or Equivariance to the Same Transformation Using Context

Apart from adaptively learning equivariance to a subset of transformation groups and invariance to the rest as shown in Section 3.1, we extend ContextSSL to operate within environments characterized by a single transformation. Motivated by this, we ask the question: *Can* ContextSSL *adapt to learn equivariance or invariance to the same transformation depending on the context?*. At training, we randomly sample one of these environments. If the environment corresponds to enforcing equivariance, we construct our context in the same way as before i.e. pairs of positives transformed using augmentations sampled from the transformation group. However, if the environment corresponds to enforcing invariance, we maximize alignment between positives transformed by augmentation

Table 19: Model performance in rotation prediction task, within the rotation-equivariant environment. The R^2 values are calculated for both the representations and the embeddings (output of projection head for invariant models (VICReg, SimCLR) or predictor for equivariant models (SEN, EquiMod, SIE, CONTEXTSSL). Unlike other models, which experience a significant performance drop between representations and embeddings, CONTEXTSSL maintains consistent performance.

Method	Rotation prediction (R^2)		
	Representations	Embeddings	Change
VICReg	0.37	0.23	-0.14
SimCLR	0.51	0.23	-0.28
SEN	0.63	0.39	-0.24
EquiMod	0.51	0.39	-0.12
SIE	0.67	0.60	-0.07
CONTEXTSSL (rotation)	**0.74**	**0.74**	**-0.00**

Table 20: Performance of CONTEXTSSL on accuracy of predictor embeddings for context-dependent labels.

Method	Classification (top-1)						
	0	2	14	30	126	Representation	Change
SimCLR			52.7			85.3	-32.6
SimCLR$^+$ (c=0)			72.4			83.4	-11.0
SimCLR$^+$			41.8			42.3	-0.5
CONTEXTSSL (rotation)	76.6	76.9	75.6	76.9	77.5	80.4	-2.9
CONTEXTSSL (color)	76.6	75.3	71.7	72.6	76.5	80.4	-3.9

sampled from the transformation group without conditioning on that augmentation. Take rotation in 3DIEBench as an example. As shown in Table 21, similar to our results in two transformation setting (rotation and color) in Section 3.1, CONTEXTSSL effectively adapts to enforce invariance and equivarance to rotation depending on the context. Results for predicting individual latents are provided in Table 22.

Table 21: **Single Transformation Setting.** Performance of CONTEXTSSL in 3DIEBench under the equivariant environment, i.e. CONTEXTSSL (rotation), and the invariant environment, i.e. CONTEXTSSL (none), with respect to rotation.

Method	Rotation prediction (R^2)					Classification (top-1)
	0	2	14	30	126	Representation
SimCLR			0.506			85.3
SimCLR$^+$ (c=0)			0.478			83.4
SimCLR$^+$			0.247			42.3
CONTEXTSSL (rotation)	0.737	0.737	0.736	0.737	0.738	80.6
CONTEXTSSL (none)	0.737	0.717	0.477	0.377	0.473	80.6

Table 22: **Single Transformation Setting.** Performance of CONTEXTSSL in 3DIEBench under the equivariant environment, i.e. CONTEXTSSL (rotation), and the invariant environment, i.e. CONTEXTSSL (none), with respect to rotation, to predict the individual latent values.

Method	Rotation prediction (R^2)				
	0	2	14	30	126
SimCLR			0.791		
SimCLR$^+$ (c=0)			0.773		
SimCLR$^+$			0.544		
CONTEXTSSL (rotation)	0.778	0.777	0.767	0.768	0.777
CONTEXTSSL (none)	0.839	0.829	0.721	0.667	0.698