No Free Lunch in Self Supervised Representation Learning

Ihab Bendidi
IBENS, Ecole Normale Supérieure
Minos Biosciences
Paris, France

Adrien Bardes
INRIA, Ecole Normale Supérieure
FAIR, Meta
Paris, France

Ethan Cohen
IBENS, Ecole Normale Supérieure
Synsight
Paris, France

Alexis Lamiable
IBENS, Ecole Normale Supérieure
Paris, France

Guillaume Bollot
Synsight
Evry, France

Auguste Genovesio
IBENS, Ecole Normale Supérieure
Paris, France

Abstract

Self-supervised representation learning in computer vision heavily relies on hand-crafted image transformations to derive meaningful, invariant features. Yet, the literature has limited explorations on the impact of transformation design. This work delves into this relationship, particularly its effect on domains beyond natural images. We posit that transformation design acts as beneficial supervision. We establish that transformations influence representation features and clustering relevance, and further probe transformation design's effect on microscopy images, where class differences are subtler than in natural images, leading to more pronounced impacts on encoded features. Conclusively, we showcase that careful transformation selection, based on desired features, enhances performance by refining the resulting representation.

1 Introduction

In Self-Supervised Representation Learning (SSRL), models are often trained to learn consistent representations from two transformed versions of the same image. The goal of SSRL is to utilize large unannotated datasets to acquire representations beneficial for downstream tasks with limited annotated data, making it stand as a cornerstone of *Deep Learning* in computer vision Bardes et al. [2022a], Caron et al. [2021, 2018], Chen et al. [2020a,b], Grill et al. [2020], Zbontar et al. [2021], rivaling or even surpassing supervised learning in certain downstream tasks.

SSRL leans heavily on varied image transformations, intended to retain semantic content amidst distortion. Though SSRL achieves impressive classification accuracies on natural images, optimizing transformation parameters has markedly boosted model performances Chen et al. [2020a]. Yet, the broader implications of these augmentation choices remain sporadically explored Grill et al. [2020], Wagner et al. [2022], especially across tasks Zhang and Ma [2022] and domains Xiao et al. [2021]. The depth of impact these choices have on pretraining, feature extraction, and performance in varied domains remains ambiguous.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Table 1: **Metrics for clustering, linear evaluation, and LPIPS** Zhang et al. [2018] for VGG11 on MNIST LeCun et al. [1998] using MoCov2 Chen et al. [2020b] with distinct transformation sets. Bold: First Set emphasizes digits, Second Set focuses on handwriting style. Top1 Accuracy from Linear Evaluation; LPIPS with AlexNet Krizhevsky et al. [2012] indicates perceptual similarity; lower is better. The Silhouette score Rousseeuw [1987] shows good cluster quality for the second set, contrasting its AMI score's failure to reflect digit clusters.

Transformation sets	Silhouette	AMI	Top1 Acc	LPIPS
Rotation+Crop	0.74	0.79	98.4	0.22
Rotation+Crop+Padding	0.78	0.81	99.3	0.25
Rotation+Crop+Padding +ColorInversion (First Set)	**0.87**	**0.83**	**99.6**	**0.33**
Rotation+Crop+Flips	0.71	0.66	96.2	0.32
Rotation+Crop+Flips+RandomErasing (Second Set)	**0.66**	**0.37**	**62.1**	**0.51**

Some important questions remain unanswered. Are the image features encoded into the latent representations being affected by the choice of transformations ? And what is the amplitude of these issues in domains other than natural images? In this paper, we report and analyze the outcomes of our experimentation to shed light on this subject. Using convolution based approaches on small to medium scale datasets, our contributions can be succinctly summarized as follows:

- Through Self-Supervised training analysis, we show that selecting specific transformation combinations optimizes models for distinct feature encoding. This strategic choice can result in the inclusion or omission of certain features, enabling models to be fine-tuned for diverse tasks.
- We explore the impact of transformation choice in Self-Supervised Learning within the biological realm, where class distinctions are subtle. Our results highlight the amplified significance of transformation selection in this domain, indicating that a focused feature definition enhances results. Furthermore, our data reveals the advantage of an informed transformation choice in SSRL over transfer learning in small datasets with domain variations.

2 The choice of transformations is a subtle layer of weak supervision

2.1 Transformation choice impacts clustering and representation information

Investigating the effects of transformation variations is vital for grasping their impact on representation quality. We focus on the unsupervised clustering task, exploring how transformation choices dictate the encoded information. We assess representations achieved by encoders of varying architectures (VGG11, ResNet18, ConvNeXt-Tiny) using two major SSRL techniques (MoCov2 Chen et al. [2020b], BYOL Grill et al. [2020]) on the MNIST dataset LeCun et al. [1998]. Our research scrutinizes the impact of transformation compositions on embedded information and its task relevance. We use two transformation sets. The first, including padding, color inversion, mild random rotation, and random cropping, targets maintaining the digit's integrity. The second, comprising vertical flips, strong random rotation, and random erasing, examines representations when disrupting the digit. Additional configurations from these sets were tested in subsequent training rounds. Each iteration, executed five times

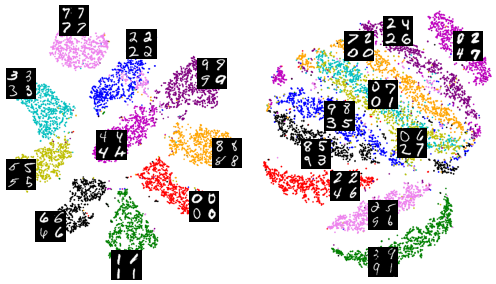


Figure 1: **Transformation choices guide the features learned, tailoring a model for varied tasks.** A t-SNE projection of MNIST dataset's ten-class clustering LeCun et al. [1998] showcases two representations from MoCo V2 trainings Chen et al. [2020b]. One retains digit class data *(left)* via padding, color inversion, and cropping; the other emphasizes handwriting classes *(right)* through flips, rotations and erasing.

Table 2: **Comparison of the mutual information score Vinh et al. [2010] using transformations and SSRL methods**, averaged over five runs and benchmarked against ImageNet models. Transformations include rotations, affine shifts, color adjustments, and flips. One set adds cropping, the other uses rotations. The weighted set uses our dual SSRL losses, aiming for superior phenotypic extraction, especially on smaller datasets.

Transformations	SSRL approach	Backbone	Nocodazole	Cytochalasin B	Taxol
First Set	MoCo v2	VGG13	0.19	0.27	0.16
		ResNet18	0.17	0.25	0.15
	Byol	VGG13	0.21	0.28	0.19
		ResNet18	0.2	0.25	0.17
	VICReg	VGG13	0.19	0.26	0.2
		ResNet18	0.16	0.25	0.21
Second Set	MoCo v2	VGG13	0.37	0.45	0.38
		ResNet18	0.33	0.42	0.31
	Byol	VGG13	0.38	0.48	0.41
		ResNet18	0.35	0.44	0.34
	VICReg	VGG13	0.38	0.44	0.36
		ResNet18	0.34	0.43	0.3
Combination of Sets	MoCo v2	VGG13	0.51	0.66	0.52
		ResNet18	0.46	0.63	0.47
	Byol	VGG13	0.51	0.64	**0.54**
		ResNet18	0.47	0.61	0.48
	VICReg	VGG13	**0.55**	**0.67**	0.51
		ResNet18	0.5	0.63	0.45
Pretrained models with Imagenet		VGG16	0.34	0.55	0.36
		ResNet101	0.39	0.57	0.43

with unique seeds, had its average score calculated. For a thorough assessment of our results and to determine transformation impacts on perceptual image similarities, we utilize the Learned Perceptual Image Patch Similarity (LPIPS) metric Zhang et al. [2018]. Post-training, K-Means clustering Lloyd [1982] with ten clusters is executed, followed by a linear evaluation using digit labels. We gauge clustering efficiency using the Silhouette score Rousseeuw [1987]. We also apply the Adjusted Mutual Information score (AMI) Vinh et al. [2010] for a robust clustering evaluation. Detailed score descriptions are in Supplementary materials.

From Table 1, a decline in the AMI and top1 accuracy scores emerges transitioning from basic transformations (rotation, crop) to the second set. This is accompanied by increased perceptual dissimilarity in the second set, expected given the destructive nature of random erasing versus flips. This leads to a diminished digit-associated representation, evident in the reduced digit classification accuracy, unaffected by supervised training during linear evaluation. However, while the AMI score drops more sharply, the silhouette score shows a mild decrease. Indicating, the clusters from the second set, although different in feature focus, remain well-defined. As seen in Figure 1, clusters from both sets are distinct. The latter set's clusters emphasize handwriting traits like line thickness. More results for different architectures and SSRL approaches are in the Supplementary Materials, showing consistent trends. Thus, selecting specific transformations during training can intentionally guide the encoding of certain image features, optimizing some specific task performance.

2.2 The effect of transformations correlates with the subtlety of a domain

To study the impact of transformations on domains with subtle class image differences, we examine microscopy images from BBBC021v1 Caie et al. [2010], which presents cells under two conditions (untreated and treated with a compound). These cells exhibit subtle within-condition variability and differences between conditions. Such nuances present a significant challenge for self-supervised representation learning. We preprocess the images by extracting a 196x196 pixels region around each cell nucleus, focusing on compounds Nocodazole, Cytochalasin B, and Taxol. Using VGG13 Simonyan and Zisserman [2015] and ResNet18 with MoCov2 Chen et al. [2020b], BYOL Grill et al. [2020], and VICReg Bardes et al. [2022a], we run two distinct trainings for each compound, with different transformation compositions, repeated five times. K-Means Lloyd [1982] clustering (k=2) is

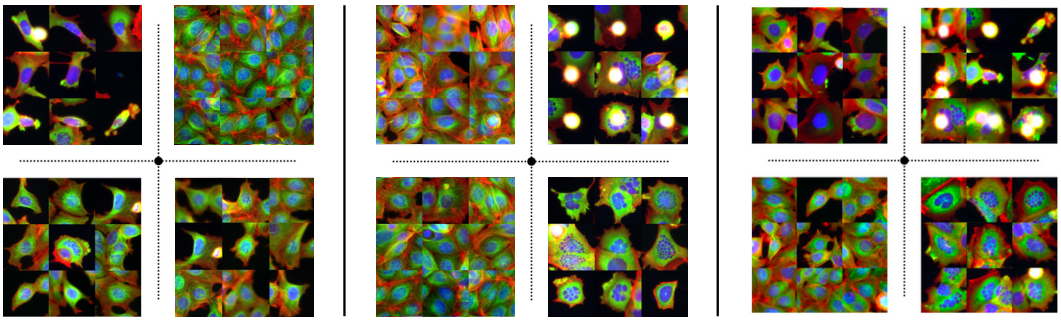


Figure 2: **An illustration of the various K-Means (k=4) clustering results on the data subset of Nocodazole** , utilizing different combinations of augmentations with a VGG13 as backbone and MoCov2 as SSRL approach, the objective being to separate the distinct morphological reactions of the cells into different clusters.

conducted on test embeddings, and the Adjusted Mutual Information score (AMI) Vinh et al. [2010] is computed based on the compound's ground truth labels. We provide more details on the dataset, data processing and the experiments in the Supplementary Materials.

Table 2 presents the AMI scores from two transformation sets in training, against a pretrained ImageNet model. The first set, comprised of color jitter, flips, rotation, affine, and random crops, emphasizes inter-cellular interactions, while the second, comprised of color jitter, flips, affine, and center crops targets single cell and nucleus features. By substituting slight random cropping with strong random rotations, we achieve a higher mean AMI score. This indicates better separation of untreated from treated cells, capturing even subtle differences. Random cropping, however, omits relevant cellular features, proving less effective. We hypothesize that understanding the compound's effect on cellular morphology depends on intracellular changes and inter-cellular interactions. Inspired by this, we train using a weighted sum of two SSRL losses, each from the aforementioned transformation sets. Table 2 shows our biology-informed approach surpasses Transfer Learning on a smaller dataset. We perform an ablation study on the effect of each transformation in the Supplementary Materials. Contrasting with datasets having clear image differences, the significant impact here implies transformations play a pivotal role in discerning subtler class distinctions. Optimizing transformation selection for specialized datasets can yield performances widely surpassing supervised pretrained models, even with limited data.

Beyond clustering into two conditions, we wonder what combination of transformations could lead to a proper clustering of cell phenotypes (or morphology). We test varied transformation compositions using a VGG13 Simonyan and Zisserman [2015] and MoCov2 Chen et al. [2020b]. Applying K-Means Lloyd [1982] clustering (k=4) on test set representations highlights diverse outcomes, much like in Section 2.1. The Nocodazole treatment, combined with color jitter, flips, rotation, affine, and random crops, clusters by cell number and size, not morphology (*Figure 2 left*). Replacing affine and random crops with a center crop distinguishing 50% of the image around the central cell gives clusters with two discernible phenotypes but also bifurcates untreated cells (*Figure 2 center*). Yet, our dual SSRL losses using both transformation sets yields perfect phenotype separation (*Figure 2 right*). This parallels findings in Section 2.1. The nuanced differences in this dataset heighten transformation selection sensitivity, leading to various possible representations. In sum, transformation combinations in this scenario can be a subtle supervision influencing outcomes, or a potent tool for specific tasks.

3 Conclusion

In this work, we explore transformation choices on convolution-based methods, emphasizing small to medium-scale datasets. Our experiments highlight the pivotal role of transformation selection, and combination in the efficacy of self-supervised representations. Through strategic transformation choices optimizing feature encoding, we boost task-specific outcomes. Our results indicate heightened implications of transformation choices in domains with nuanced class differences, often surpassing pretrained models on smaller datasets. Essentially, a good representation often depends on the

intended task. While SSRL originally aimed to bypass such limitations, our context-specific findings illuminate the advantages of well-selected transformations based on key features.

This study recognizes its focus on medium-scale datasets and convolution methods. Future work might delve into larger datasets and transformer architecture integration. Extending the analysis to more diverse downstream tasks could further elucidate transformation impact. An exciting avenue is leveraging informed transformations to refine base models for specific tasks without labeled data. Overall, this study contributes valuable insights and suggests promising avenues for future investigations in the field of transformation learning within deep learning frameworks.

4 Acknowledgments

This work was supported by ANR–10–LABX–54 MEMOLIFE and ANR–10 IDEX 0001 –02 PSL* Université Paris and was granted access to the HPC resources of IDRIS under the allocation 2020-AD011011495 made by GENCI.

References

Samira Abnar, Mostafa Dehghani, Behnam Neyshabur, and Hanie Sedghi. Exploring the limits of large scale pre-training. In *ICLR*, 2022.

Randall Balestriero, Leon Bottou, and Yann LeCun. The effects of regularization and data augmentation are class dependent. In *NeurIPS*, 2022.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022a.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicregl: Self-supervised-learning of local visual features. In *NeurIPS*, 2022b.

Anis Bourou, K. Daupin, V. Dubreuil, A. De Thonel, V. Lallemand-Mezger, and A. Genovesio. Unpaired image-to-image translation with limited data to reveal subtle phenotypes. In *ISBI*, 2023.

Peter D. Caie, Rebecca E. Walls, Alexandra Ingleston-Orme, Sandeep Daya, Tom Houslay, Rob Eagle, Mark E. Roberts, and Neil O. Carragher. High-Content Phenotypic Profiling of Drug Response Signatures across Distinct Cancer Cells. *Molecular Cancer Therapeutics*, 9:1913–1926, 2010.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020a.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning, 2020b.

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, 2021.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *The International Conference on Artificial Intelligence and Statistics*, 2011.

Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation policies from data. In *CVPR*, 2019.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. ICCV, 2021.

Aleksandr Ermolov, Aliaksandr Siarohin, Enver Sangineto, and Nicu Sebe. Whitening for self-supervised representation learning. In *ICML*, 2021.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, nov 2020. doi: 10.1038/s42256-020-00257-z.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In *NeurIPS*, 2020.

Weiran Huang, Mingyang Yi, and Xuyang Zhao. Towards the generalization of contrastive self-supervised learning, 2021.

Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. In *NeurIPS*, 2020.

Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

Alexis Lamiable, Tiphaine Champetier, Francesco Leonardi, Ethan Cohen, Peter Sommer, David Hardy, Nicolas Argy, Achille Massougbodji, Elaine Del Nery, Gilles Cottrell, Yong-Jun Kwon, and Auguste Genovesio. Revealing invisible cell phenotypes with conditional generative modeling. 2022.

Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. The mnist database of handwritten digits, 1998.

Daesoo Lee and Erlend Aune. Computer vision self-supervised learning methods on time series, 2021.

Yonggang Li, Guosheng Hu, Yongtao Wang, Timothy Hospedales, Neil M. Robertson, and Yongxin Yang. Dada: Differentiable automatic data augmentation. In *ECCV*, 2020.

Zengyi Li, Yubei Chen, Yann LeCun, and Friedrich T. Sommer. Neural manifold clustering and embedding, 2022.

Aoming Liu, Zehao Huang, Zhiwu Huang, and Naiyan Wang. Direct differentiable augmentation search. In *ICCV*, 2021.

Vebjorn Ljosa, Katherine L. Sokolnicki, and Anne E. Carpenter. Annotated high-throughput microscopy image sets for validation. *Nature Methods*, 9:637–637, 2012.

S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28:129–137, 1982.

Umar Masud, Ethan Cohen, Ihab Bendidi, Guillaume Bollot, and Auguste Genovesio. Comparison of semi-supervised learning methods for high content screening quality control. In *ECCV 2022 BIM Workshop*, 2022.

Dipan K. Pal, Sreena Nallamothu, and Marios Savvides. Towards a hypothesis on visual transformation based self-supervision. In *British Machine Vision Conference*, 2020.

Mandela Patrick, Yuki M. Asano, Polina Kuznetsova, Ruth Fong, João F. Henriques, Geoffrey Zweig, and Andrea Vedaldi. On compositions of transformations in contrastive self-supervised learning. In *ICCV*, 2021.

Alexis Perakis, Ali Gorji, Samriddhi Jain, Krishna Chaitanya, Simone Rizza, and Ender Konukoglu. Contrastive learning of single-cell phenotypic representations for treatment classification. In *Machine Learning in Medical Imaging*, pages 565–575. 2021.

Colorado J Reed, Sean Metzger, Aravind Srinivas, Trevor Darrell, and Kurt Keutzer. Selfaugment: Automatic augmentation policies for self-supervised learning. In *CVPR*, 2021.

Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

Camille Ruppli, Pietro Gori, Roberto Ardon, and Isabelle Bloch. Optimizing transformations for contrastive learning in a differentiable framework. In *MICCAI MILLanD workshop*, 2022.

Nikunj Saunshi, Orestis Plevrakis, Sanjeev Arora, Mikhail Khodak, and Hrishikesh Khandeparkar. A theoretical analysis of contrastive unsupervised representation learning. In *ICML*, 2019.

Nikunj Saunshi, Jordan T. Ash, Surbhi Goel, Dipendra Misra, Cyril Zhang, Sanjeev Arora, Sham M. Kakade, and Akshay Krishnamurthy. Understanding contrastive learning requires incorporating inductive biases. *CoRR*, 2022.

Anshul Shah, Aniket Roy, Ketul Shah, Shlok Kumar Mishra, David Jacobs, Anoop Cherian, and Rama Chellappa. Halp: Hallucinating latent positives for skeleton-based self-supervised learning of actions. In *CVPR*, 2023.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 11:2837–2854, 2010.

Julius von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. In *NeurIPS*, 2021.

Diane Wagner, Fabio Ferreira, Danny Stoll, Robin Tibor Schirrmeister, Samuel Müller, and Frank Hutter. On the importance of hyperparameters and data augmentation for self-supervised learning. *CoRR*, 2022.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, 2020.

Xiao Wang and Guo-Jun Qi. Contrastive learning with stronger augmentations, 2021.

Zixin Wen and Yuanzhi Li. Toward understanding the feature learning process of self-supervised contrastive learning. In *ICML*, 2021.

Tete Xiao, Xiaolong Wang, Alexei A. Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. In *ICLR*, 2021.

Seunghan Yang, Debasmit Das, Simyung Chang, Sungrack Yun, and Fatih Porikli. Distribution estimation to automate transformation policies for self-supervision. In *NeurIPS Workshop: Self-Supervised Learning - Theory and Practice*, 2021.

Chun-Hsiao Yeh, Cheng-Yao Hong, Yen-Chi Hsu, Tyng-Luh Liu, Yubei Chen, and Yann LeCun. Decoupled contrastive learning. In *ECCV*, 2022.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *ICML*, 2021.

Junbo Zhang and Kaisheng Ma. Rethinking the augmentation module in contrastive learning: Learning hierarchical augmentation invariance with expanded views. CVPR, 2022.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

Mingkai Zheng, Shan You, Fei Wang, Chen Qian, Changshui Zhang, Xiaogang Wang, and Chang Xu. Ressl: Relational self-supervised learning with weak augmentation. NeurIPS, 2021.

A Related work

Self supervised representation learning (SSRL). Contrastive learning approaches Chen et al. [2020a, 2021, 2020b], Yeh et al. [2022] have shown great success in avoiding trivial solutions in which all representations collapse into a point, by pushing the original image representation further away from representations of negative examples. These approaches follow the assumption that the augmentation distribution for each image has minimal inter-class overlap and significant intra-class overlap Abnar et al. [2022], Saunshi et al. [2019]. This dependence on contrastive examples has since been bypassed by non contrastive methods. The latter either have specially designed architectures Caron et al. [2021], Chen and He [2021], Grill et al. [2020] or use regularization methods to constrain the representation in order to avoid the usage of negative examples Bardes et al. [2022a], Ermolov et al. [2021], Lee and Aune [2021], Li et al. [2022], Zbontar et al. [2021], Bardes et al. [2022b]. Another line of work Kalantidis et al. [2020], Shah et al. [2023] focuses on obtaining positive and negative examples in the feature space, bypassing the need of augmenting the input images with transformations.

Impact of image transformations on SSRL. Compared to the supervised learning field, the choice and amplitude of transformations has not received much attention in the SSRL field Balestriero et al. [2022], Cubuk et al. [2019], Li et al. [2020], Liu et al. [2021]. Studies such as Wen and Li [2021] and Wang and Isola [2020] analyzed in a more formal setting the manner in which augmentations decouple spurious features from dense noise in SSRL. Some works Chen et al. [2020a], Geirhos et al. [2020], Grill et al. [2020], Perakis et al. [2021] explored the effects of removing transformations on the overall accuracy. Other works explored the effects of transformations by capturing information across each possible individual augmentation, and then merging the resulting latent spaces Xiao et al. [2021], while some others suggested predicting intensities of individual augmentations in a semi-supervised context Ruppli et al. [2022]. However the latter approach is limited in practice as individual transformations taken alone were shown to be far less efficient than compositions Chen et al. [2020a]. An attempt was made to explore the underlying effect of the choice of transformation in the work of Pal et al. [2020], one of the first works to discuss how certain transformations are better adapted to some pretext task in self supervised learning. This study suggests that the best choice of transformations is a composition that distorts images enough so that they are different from all other images in the dataset. However favoring transformations that learn features specific to each image in the dataset should also degrade information shared by several images in a class, thus damaging model performance. Altogether, it seems that a good transformation distribution should maximize the intra-class variance, while minimizing inter-class overlap Abnar et al. [2022], Saunshi et al. [2019]. Other works proposed a formalization to generalize the composition of transformations Patrick et al. [2021], which, while not flexible, provided initial guidance to improve results in some contexts. This was followed by more recent works on the theoretical aspects of transformations, von Kügelgen et al. [2021] that studied how SSRL with data augmentations identifies the invariant content partition of the representation, Huang et al. [2021] that seeks to understand how image transformations improve the generalization aspect of SSRL methods, and Zhang and Ma [2022] that proposes new hierarchical methods aiming to mitigate a few of the biases induced by the choice of transformations.

Learning transformations for SSRL. A few studies showed that optimizing the transformation parameters can lead to a slight improvement in the overall performance in a low data annotation regime Ruppli et al. [2022], Reed et al. [2021]. However, the demonstration is made for a specific downstream task that was known at SSRL training time, and optimal transformation parameters selected this way were shown not to be robust to slight change in architecture or task Saunshi et al. [2022]. Other works proposed optimizing the random sampling of augmentations by representing them as discrete groups, disregarding their amplitude Wagner et al. [2022], or through the retrieval of strongly augmented queries from a pool of instances Wang and Qi [2021]. Further research aimed to train a generative network to learn the distribution of transformation in the dataset through image-to-image translation, in order to then avoid these transformations at self supervised training time Yang et al. [2021]. However, this type of optimization may easily collapse into trivial transformations.

Performance of SSRL on various domains and tasks. Evaluation of SSRL works relies almost exclusively on the accuracy of classification of natural images found in widely used datasets such as Cifar Krizhevsky [2009], Imagenet Deng et al. [2009] or STL Coates et al. [2011]. This choice is largely motivated by the relative ease of interpretation and understanding of the results, as natural images can often be easily classified by eye. This, however, made these approaches hold potential biases concerning the type of data and tasks for which they could be efficiently used. It probably also

has an impact on the choice and complexity of the selected transformations aiming at invariance: some transformations could manually be selected in natural images but this selection can be very challenging in domains where differences between classes are invisible. The latter was intuitively mentioned in some of the previously cited studies. Furthermore, the effect of the choice of transformation may be stronger on domains and task where the representation is more thoroughly challenged. This is probably the case in botany and ornithology Xiao et al. [2021] but also in the medical domain Ruppli et al. [2022] or research in biology Bourou et al. [2023], Lamiable et al. [2022], Masud et al. [2022].

B Datasets

We perform the experiments mentioned in Section 2.2 on microscopy images available from BBBC021v1 Caie et al. [2010], a dataset from the Broad Bioimage Benchmark Collection Ljosa et al. [2012]. This dataset is composed of breast cancer cells treated for 24 hours with 113 small molecules at eight concentrations, with the top concentration being different for many of the compounds through a selection from the literature. Throughout the quality control process, images containing artifacts or with out of focus cells were deleted, and the final dataset totalled into 13,200 fields of view, imaged in three channels, each field composed of thousands of cells. In the cells making up the dataset, twelve different primary morphological reactions from the compound-concentrations were identified, with only six identified visually, while the remainder were defined based on the literature, as the differences between some morphological reactions were very subtle. We perform a simple cell detection in each field of view, in order to crop cells centered in (196x196px) images. We then filter the images by compound-concentration, and keep images treated by Nocodazole at its 4 highest concentrations. These compound-concentrations result in 4 morphological cell reactions, for which we don't have individual labels for each cell image. We sample the same number of images from views of untreated cells, totalling into a final data subset of 3500 images, which we split into 70% training data, 10% validation data and 20% test data. We repeat the same process for the compounds Taxol and Cytochalasin B, each containing 4 and 2 morphological reactions respectively, to result in data subsets of sizes 1900 and 2300 images, respectively.

C Additional Analysis : Transformation choices induce inter-class bias

In order to understand the ramifications of transformations on the performance of a model, we delve into the examination of the behavior of models that are trained with widely adopted SSRL techniques on the benchmark datasets Cifar10, Cifar100 Krizhevsky [2009] and Imagenet100 Deng et al. [2009], while altering the magnitude and likelihood of the transformations. With a Resnet18, a Resnet50 and a ConvNeXt-Tiny architectures as backbones, we employ a fixed set of transformations, comprised of randomized cropping, chromatic perturbations, and randomized horizontal inversions. Subsequently, we uniformly sample a set of amplitude and probability values for each transformation, in order to create a diverse range of test conditions. Each training is repeated a number of times (three for Imagenet and five for Cifar), with distinct seed values, and the mean and standard deviation of accuracy, measured through linear evaluation over frozen weights, are computed over these five trainings for each method and each transformation value.

As depicted in Figure 3, we observe minimal fluctuation in the overall accuracy of each model as we slightly alter any one of the transformations. This stands in stark contrast to the class-level accuracies observed, in which we discern significant variation in the accuracy value for many classes, as we vary the parameters of transformations, hinting at a greater impact of variations in transformation parameters on the class-level. Through the same figure, it becomes apparent that a number of classes exhibit distinct, and at times, entirely antithetical behaviors to each other within certain ranges of a transformation parameter. In the context of the datasets under scrutiny, this engenders a bias in the conventional training process of models, which either randomly samples transformation parameters or relies on hyperparameter optimization on overall accuracy to determine optimal parameters. This bias manifests itself in the manner in which choosing specific transformation parameters would impose a penalty on certain classes while favoring others. This is demonstrated in Figure 3 by the variation in accuracy of the Caterpillar and Crocodile classes for a model trained using VICReg Bardes et al. [2022a], as the crop size is varied (bottom left plot). The reported accuracies uncover that smaller crop sizes prove advantageous for the Caterpillar class, stimulating the model to recognize repetitive patterns and features consistent across the length of the caterpillar's body. However, the Crocodile

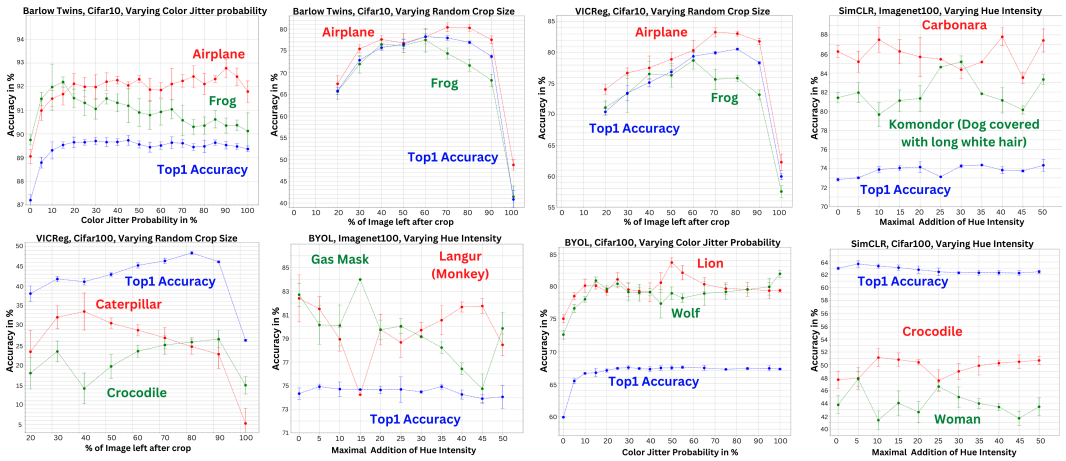
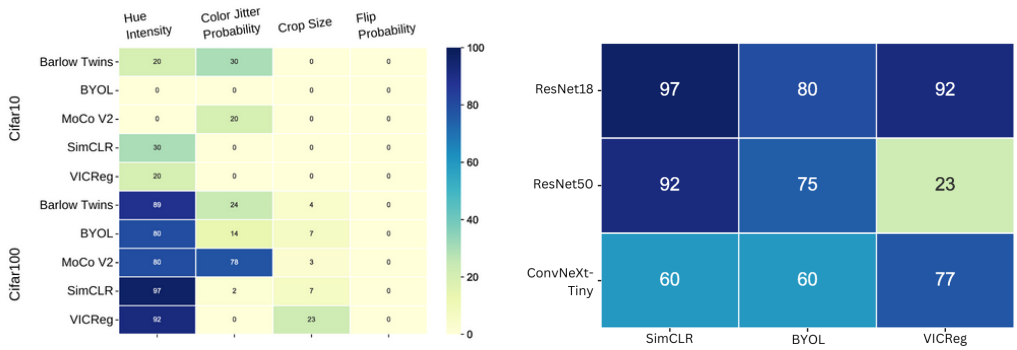


Figure 3: **Different transformation parameter choices induce an inter-class bias.** Inter-class accuracy results versus variation of a transformation parameter, for Resnet18 architectures trained with various SSRL methods on the benchmark datasets Cifar10, Cifar100 and Imagenet100. Each dot and associated error bar reflects the mean and standard deviation of three runs for Imagenet100 and five runs for Cifar with different random seeds. While overall accuracy remains relatively consistent across a range of transformation parameters, these transformations can have a subtle but significant impact on individual class performance, either favoring or penalizing specific classes.

class doesn't fare as well under similar conditions. This can be explained by considering the differing morphologies of the two subjects. The Caterpillar class benefits from smaller crops as the caterpillars exhibit uniformity across their body parts. Conversely, for the Crocodile class, a small crop size could potentially capture a segment like the tail, which could be misattributed to other classes, such as snakes, due to its isolated resemblance. Therefore, the choice of transformation probability or intensity directly affects class-level accuracies, an impact that may not be immediately apparent when only considering the overall accuracy.

In order to gain deeper understanding of the inter-class bias observed in our previous analysis, we aim to further investigate the extent to which this phenomenon impacts the performance of models trained with self-supervised learning techniques. By quantitatively assessing the correlation scores between class-level accuracies obtained under different transformation parameters, we aim to measure the prevalence of this bias in self-supervised learning methods. More specifically, a negative correlation score between the accuracy of two classes in response to varying a given transformation would indicate opposing reactions for those classes to the transformation parameter variations. Despite its limitations, such as the inability to quantify the extent of bias and the potential for bias to manifest in specific ranges while remaining positively correlated in others (See Lion/Wolf pair in Figure 3), making it difficult to detect, this measure can still provide a preliminary understanding of the degree of inter-class bias. To this end, we conduct a series of experiments utilizing a ResNet18 encoder on the benchmark datasets of Cifar10 and Cifar100 Krizhevsky [2009]. We employ a diverse set of state-of-the-art self-supervised approaches: Barlow Twins Zbontar et al. [2021], MoCov2 Chen et al. [2020b], BYOL Grill et al. [2020], SimCLR Chen et al. [2020a], and VICReg Bardes et al. [2022a], and use the same fixed set of transformations as in our previous analysis depicted in Figure 3. We vary the intensity of the hue, the probability of color jitter, the size of the random crop, and the probability of horizontal inversion through 20 uniformly sampled values for each, and repeat each training five times with distinct seed values. We compute the Pearson, Kendall and Spearman correlation coefficients for each pair of classes with respect to a given transformation parameter, as well as their respective p-values, and define class pairs with opposite behaviors as those with at least one negative correlation score of the three measured correlations lower than -0.3 and a p-value lower than 0.05. We then measure the ratio of classes with at least one opposite behavior to another class, compared to the total number of classes, in order to understand the extent of inter-class bias for a given transformation, method, and dataset.



(a) (b)

Figure 4: **A comprehensive analysis to quantify the occurrences of negative correlations among individual classes accuracies in the Cifar10 and Cifar100 datasets.** This analysis is conducted using different backbones and SSRL approaches, along with various transformations, through examination of the ratio of classes exhibiting negative correlations with each other in relation to the total number of classes within Cifar10 and Cifar100. (a) Notably, within Cifar100 with ResNet18, a higher proportion of negatively correlated classes is observed, which can be attributed to the increased number of classes that may overlap when increasing color jitter. (b) We employ different backbones (ResNet18, ResNet50, and ConvNeXt-Tiny) and self-supervised approaches (SimCLR, BYOL, and VICReg) on Cifar100 while manipulating the hue intensity. We compute the ratio of classes exhibiting negative correlations with each other in relation to the total number of classes within Cifar100. The results obtained from these various configurations remain in similar ranges of each other while varying both the SSRL approach and the backbone used. This provides an evidence that the observed patterns in (a) are independant of the SSL method and encoder architectures.

Table 3: **Correlation values between class properties and the effect of transformations on classes.** We focus on class properties such as Intrinsic Dimension, Texture Analysis, Fourier Transform, and Spectrum of Feature Covariance, and transformations such as Hue Intensity, Color Jitter Probability, Crop Size, applied on Cifar100. Values significantly larger than 1 indicate a notable difference between behavior groups with respect to the varying transformation. Asterisks (*) denote p-values > 0.05, indicating less significant correlations

Class properties	Hue Intensity	Color Jitter Probability	Crop Size
Intrinsic Dimension	16.64	19.15	0.21*
Texture Analysis	16.39	4.89	0.18*
Fourrier Transform	0.71*	7	5.19
Spectrum of Feature Covariance	1.27	0.56*	0.97*

Our findings, as represented in Figure 4a, indicate that the extent of inter-class bias for the self-supervised learning methods of interest varies among different transformations. This variability is primarily due to the fact that while these transformations aim to preserve the features that define a class across the original image and its transformed versions, they can also inadvertently compromise information specific to a particular class, while favoring the information of another class. Notably, within Cifar100, a dataset encompassing a diverse range of natural image classes, we observe a significant presence of inter-class bias when manipulating hue intensity. This outcome can be attributed to the optimization of specific features through each transformation choice, which may not be optimal for certain classes. To substantiate the generality of these findings across convolution-based networks, we conduct a comparative analysis on Cifar100 using ResNet18, ResNet50, and ConvNext-Tiny as encoders, along with BYOL, SimCLR, and VICReg as the self-supervised learning approaches. By varying hue intensity, the results, as presented in Table 4b, reaffirm the consistent trend.

To investigate the potential relationship between abstract class properties and their preferred transformations, we conduct a thorough analysis of each class's response to varying transformation parameters. We explore class accuracy behavior under distinct transformations, namely, Hue Intensity,

Color Jitter Probability, and Crop Size. By computing the slope of the linear regression line that best fits the accuracy-transformation data for each class and each model, we categorize the behavior of each class accuracy as ascending, descending, or random. Simultaneously, we compute a texture analysis measure, and a Fourier transform measure for each class in the Cifar100 dataset, as well as the spectrum of feature covariance and the intrinsic dimension using the features resulting from a ResNet model pretrained on ImageNet in a supervised manner. Using Anova and Manova correlation metrics, we then compute the correlation between these class properties and the class behaviors when varying a specific transformation.

Our results, summarized in Table 3, provide insights into the relationship between abstract image class properties and the effect of variations of transformation parameters. For instance, the Intrinsic Dimension and Texture Analysis of image classes exhibit substantial correlation with variation of Hue Intensity, implying that the intrinsic complexity and texture attributes of classes could significantly influence their response to changes in this transformation. A similar pattern is noticed with the Color Jitter Probability, albeit with a somewhat weaker correlation. Interestingly, the Spectrum of Feature Correlation shows minimal correlation with all transformations, suggesting that the covariance of class features might not significantly affect the class response to transformations. The Fourier Transform property showed mixed results, with a weak correlation with Hue Intensity but a stronger one with Crop Size, as the crop transformation can induce a varying degree of loss of signal in the image.

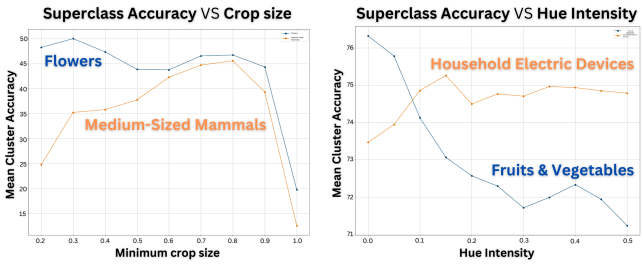


Figure 5: **Specific transformation choices control superclass performance.** A comparative analysis of the mean accuracy for superclasses in Cifar100, considering trainings conducted using BYOL, SimCLR, and VICReg as SSRL approaches, with variations in crop size or hue intensity. Noteworthy observations reveal consistent patterns across all models, illustrating how each transformation parameter can exert a distinct influence on different superclasses. Consequently, each superclass exhibits unique optimal parameters, emphasizing the potential for transformation parameter selection to effectively modulate the performance of specific superclasses.

These results imply that the choice of transformations not only introduces a inter-class bias that can subtly impact performance in real-world scenarios, but it also presents an opportunity to harness this bias to achieve a desired balance in class performance or optimize specific class accuracies for specific use cases. We focus in the following analysis on the coarse grained labels of Cifar100, commonly called superclasses in the literature. As demonstrated in Figure 5, we observe that certain superclasses in the Cifar100 dataset exhibit improved recognition when specific transformation parameters are applied, when other don't. This highlights the potential of consciously selecting and studying transformations in our training process to enhance the performance of specific class clusters or achieve a balanced performance across classes. Therefore, the careful tailoring of specific transformations and their parameters becomes crucial in preserving desired information within classes, presenting a potential avenue for improvement in training.

D Model training

D.1 Study of inter-class bias in self supervised classification

For results in Section C and Table 4, we run several trainings of 12 SSRL methods Bardes et al. [2022a], Caron et al. [2018, 2020], Chen et al. [2020a], Chen and He [2021], Chen et al. [2020b], Dwibedi et al. [2021], Grill et al. [2020], Lee and Aune [2021], Zbontar et al. [2021], Zheng et al. [2021] on Cifar10 and Cifar100 Krizhevsky [2009] with a Resnet18 architecture, with a pretraining of 1000 epochs without labels. We use the same transformations with similar parameters on all approaches, namely a 0.4 maximal brightness intensity, 0.4 maximum contrast intensity, 0.2 maximum saturation intensity, all with a fixed probability of 80%. With a maximal hue intensity of 0.5, we

vary the hue probability of application between 0% and 100% by uniformly sampling 20 probability values in this range. We use stochastic gradient descent as optimization strategy for all approaches, and through a line search hyperparameter optimization, we use a batch size of 512 for all approaches except Dino Caron et al. [2021] and Vicreg Bardes et al. [2022a] for which we use a batch size of 256. Following the hyperparameters used in the literature of each approach, we use a projector with a 256 output dimension for most methods, except for Barlow Twins Zbontar et al. [2021], Simsiam Chen and He [2021], Vicreg Bardes et al. [2022a] and Vibcreg Lee and Aune [2021], for which we use a projector with a 2048 output dimension, and DeepclusterV2 Caron et al. [2018] and Swav Caron et al. [2020] for which we use a projector with a 128 output dimension. For some momentum based methods (Byol Grill et al. [2020], MocoV2+ Chen et al. [2020b], NNbyol Dwibedi et al. [2021] and Ressl Zheng et al. [2021]), we use a base Tau momentum of 0.99 and use a base Tau momentum of 0.9995 for Dino Caron et al. [2021]. For Mocov2+ Chen et al. [2020b], NNCLR Dwibedi et al. [2021] and SimCLR Chen et al. [2020a], we use a temperature of 0.2.

We train the Barlow Twins Zbontar et al. [2021] based model with a learning rate of 0.3 and a weight decay of 10^{-4}, and Byol Grill et al. [2020] as well as NNByol Dwibedi et al. [2021] with a learning rate of 1.0 and a weight decay of 10^{-5}. We train DeepclusterV2 Caron et al. [2018] with a learning rate of 0.6, 11 warmup epochs, a weight decay of 10^{-6}, and 3000 prototypes. We train Dino with a learning rate of 0.3, a weight decay of 10^{-4}, and 4096 prototypes, while we train MocoV2+ Chen et al. [2020b] with a learning rate of 0.3, a weight decay of 10^{-4} and a queue size of 32768. For NNclr Dwibedi et al. [2021], we use a learning rate of 0.4, a weight decay of 10^{-5}, and a queue size of 65536. We train ReSSL Zheng et al. [2021] with a learning rate of 0.05, and a weight decay of 10^{-4}, while we train SimCLR Chen et al. [2020a] with a learning rate of 0.4, and a weight decay of 10^{-5}. For Simsiam Caron et al. [2020], we use a learning rate of 0.5, and a weight decay of 10^{-5}, and use for Swav Caron et al. [2020] a learning rate of 0.6, a weight decay of 10^{-6}, a queue size of 3840, and 3000 prototypes. We use for Vicreg Bardes et al. [2022a] and Vibcreg Lee and Aune [2021] a learning rate of 0.3, a weight decay of 10^{-4}, an invariance loss coefficient of 25, and a variance loss coefficient of 25. We use a covariance loss coefficient of 1.0 for Vicreg Bardes et al. [2022a] and a covariance loss coefficient of 200 for Vibcreg Lee and Aune [2021]. We perform linear evaluation after each pretraining for all methods, through freezing the weights of the encoder and training a classifier for 100 epochs. We use 5 different global seeds (5, 6, 7, 8, 9) for each hue intensity value, and compute the mean top1 accuracy resulting from the linear evaluation, using each of the 5 different experiences. Each training run was made on a single V100 GPU. On ImageNet100 Deng et al. [2009], we train a Resnet18 encoder with BYOL Grill et al. [2020], MoCo V2 Chen et al. [2020b], VICReg Bardes et al. [2022a] and SimCLR Chen et al. [2020a], using a batch size of 128 for 400 epochs. We use a learning rate of 0.3 and a weight decay of 10^{-4} for MoCo V2 and VICReg, and a weight decay of 10^{-5} and learning rates of 0.4 and 0.45 for SimCLR and BYOL respectively. We repeat each experience three times with three global seeds (5,6 and 7) and compute its mean and standard deviation. We repeat the same experiment with the same parameters on ResNet50 and ConvNeXt-Tiny.

For the results in Figures 3 and 4a, we reuse the same training hyperparameters for Barlow Twins Zbontar et al. [2021], Moco V2 Chen et al. [2020b], BYOL Grill et al. [2020], SimCLR Chen et al. [2020a] and Vicreg Bardes et al. [2022a], and uniformly sample 10 values in the range of [0;0.5] for the maximal hue intensity, with a fixed 80% probability. We run different experiments for the 5 global seeds for each hue intensity, and compute their mean and standard deviation. We perform the same process while fixing maximal hue intensity to 0.1, and varying its probability by uniformly sampling 20 probability values in the range [0;100]. We repeat a similar process for the random cropping and horizontal flips, by sampling 8 values uniformly in the range of [20;100] of the size ratio to keep of the image, and sampling 20 values uniformly in the range of [0;100] for the probability of application of horizontal flips.

D.2 MNIST Clustering

For the displayed clustering results in Figure 1 of the main paper, we use a VGG11 Simonyan and Zisserman [2015] architecture, with a projector of 128 output dimension, trained using a MocoV2+ Chen et al. [2020b] loss function on Mnist LeCun et al. [1998] for 250 epochs. We use an Adam optimizer, a queue size of 1024, and a batch size of 32. We set temperature at 0.07, learning rate at 0.001, and weight decay at 0.0001. We run two trainings with two separate sets of compositions of transformations, each run on a single V100 GPU, and perform a Kmeans (K=10) clustering on the resulting

representations of the test set. For the digit clustering, we use a composition of transformations composed of a padding of 10% to 40% of the image size, color inversion, rotation with a maximal angle of 25°, and random crop with a scale in the range of [0.5;0.9] of the image, and then a resizing of the image to 32x32 pixels. For the handwriting flow clustering, we use a composition of transformations composed of horizontal & vertical flips with an application probability of 50% each, rotations with a maximal angle of 180°, random crop with a scale in the range of [0.9;1.1], and random erasing of patches of the image, with a scale in the range of [0.02;0.3] of the image and a probability of 50%. We perform linear evaluation by training classifiers on the frozen representations of the trained models, in order to predict the digit class, and evaluate using the top1 accuracy score. For results on Table 7 and Figure 9, we use ResNet18 and ConvNeXt-Tiny architectures with BYOL and MoCov2 ass SSRL approaches. We use for BYOL a learning rate of 0.01 and a weight decay of 10^{-5}, with a projector with a 256 output dimension. We then use the same parameters and augmentations as previous trainings.

D.3 Clustering evaluation with the AMI Score

We use the adjusted mutual information (AMI) Vinh et al. [2010] in Section 2 of the main paper to evaluate clustering quality, and to measure the similarity between two clusterings. It is a value that ranges from 0 to 1, where a higher value indicates a higher degree of similarity between the two clusterings. This score holds an advantage over clustering accuracy Kuhn [1955] in one main aspect, being that the clustering accuracy only measures how well the clusters match the labels of the true clusters, and does not take into account the structure within the clusters, such as heterogeneity of some of the clusters. This is unlike the AMI score, which takes into account both the structure between the clusters and the structure within the clusters, by measuring the "agreement" between the groupings of a predicted cluster and the groupings of the true cluster. If both clusterings agree on most of the groupings, then the AMI score will be high, and inversely low if they do not.

The AMI score can be computed with the formula :

$$AMI(X,Y) = \frac{MI(X,Y) - E(MI(X,Y))}{max(H(X),H(Y)) - E(MI(X,Y))}$$

Where $MI(X,Y)$ is the mutual information between the two clusterings, $E(MI(X,Y))$ is the expected mutual information between the two clusterings, $H(X)$ is the entropy of the clustering X, and $H(Y)$ is the entropy of the clustering Y. Mutual information (MI) is a measure of the amount of information that one variable contains about another variable. In the context of AMI, the two variables are the clusterings X and Y. $MI(X,Y)$ is a measure of to what extent the two clusterings are related to each other. Entropy is a measure of the amount of uncertainty in a random variable. In the context of AMI, the entropy of a clustering ($H(X)$ or $H(Y)$) is a measure of how much uncertainty exists within the clustering. Expected mutual information ($E(MI(X,Y))$) is the average mutual information between the two clusterings, assuming that the two clusterings are independent.

The adjusted mutual information (AMI) is calculated by first subtracting the expected mutual information ($E(MI(X,Y))$) from the actual mutual information ($MI(X,Y)$). This results in a measure of of the extent of the relationship between the two clusterings beyond what would be expected by chance. This value is then divided by the difference between the maximum possible entropy ($max(H(X),H(Y))$) and the expected mutual information ($E(MI(X,Y))$). Normalization of the result is achieved through this process, ensuring that it is always between 0 and 1. Figure 6 shows the results of a clustering achieved on Nocodazole vs untreated cells, with the AMI score computed after randomisation of the ground truth labels, in contrast to clustering results achieved without randomizing the labels.

D.4 Cellular Clustering

For the results in Section 2.2 of the main paper, we use a VGG13 Simonyan and Zisserman [2015] architecture, trained using a MocoV2+ Chen et al. [2020b] loss function on the data subsets of the microscopy images available from BBBC021v1 Caie et al. [2010], presented in Section B, with a batch size of 128, for 400 epoch. We use an Adam optimizer, a queue size of 1024, and set temperature at 0.07, learning rate at 0.001, and weight decay at 0.0001. Each training is made on a single V100 GPU. We perform a Kmeans (K=2) on the resulting representations of the test set of

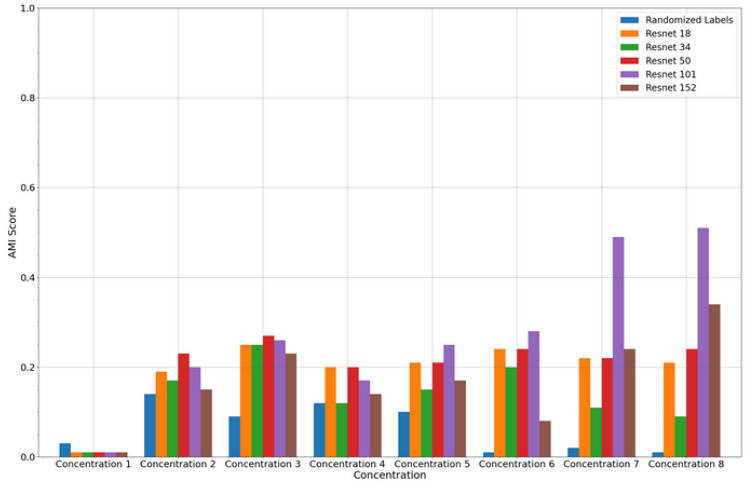


Figure 6: We perform a Kmeans clustering (K=2) on the Nocodazole and untreated images using Resnet models of various sizes, and evaluate them using the AMI score Vinh et al. [2010] on a similar number of randomly sampled images of each concentration of Nocodazole and a similar number of untreated cell images. We observe that Resnet101 outperforms the other Resnet model sizes. We also perform an experiment to better interpret the AMI scores achieved, by randomizing the labels and performing a clustering with Resnet101, and reporting the AMI score of the clustering compared to the randomized labels.

the data subsets of Nocodazole, Cytochalasin B and Taxol, and evaluate the quality of the achieved clusters compared to the ground truth using the AMI score Vinh et al. [2010].

In Table 2 of the main paper, we achieve the first AMI result through usage of an affine transformation composed of a rotation with an angle of 20°, a translation of 0.1 and a shear with a 10° angle, coupled with color jitter with a brightness, contrast and saturation intensity of 0.4, and a hue intensity of 0.125, with a 100% probability, as well as random cropping of the image with a scale in the range of [0.9;1.1] and resizing to original image size of 196x196 pixel. For the second row result, we use an affine transformation composed of a rotation with an angle of 20°, a translation of 0.1 and a shear with a 10° angle, coupled with color jitter with a brightness, contrast and saturation intensity of 0.4, and a hue intensity of 0.125, with a 100% probability, and a random rotation with a maximal angle of 360°. The results achieved through a Resnet101, are achieved by performing a Kmeans (K=2) on the representations achieved on the compound subsets using the Resnet101, and evaluating the cluster assignment quality compared to ground truth using the AMI score Vinh et al. [2010]. The choice of Resnet101 over other Resnet sizes is motivated through testing the performance of different sizes of Resnets on the different concentrations of Nocadozole/untreated cells, on which Resnet101 consistently shows the highest performance on the 4 highest concentrations, as shown in Figure 6.

For the clusterings in Figure 2 of the main paper, we train the same architecture with the same hyperparameters on different compositions of transformations, and perform a Kmeans (K=4) on the resulting representations of the test set. For all the clusters, the images displayed are the images closest to the centroïd of each cluster using an euclidean distance. The clustering in Figure 2 *left* is achieved by using a composition of color jitter with a brightness, contrast and saturation intensity of 0.4, and a hue intensity of 0.125, with a 100% probability, and horizontal and vertical flips, each with 50% probability of application, as well as random rotations with a maximal angle of 360°, an affine transformation composed of a rotation with an angle of 20°, a translation of 0.1 and a shear with a 10° angle, and a random crop with a scale sampled in the range [0.9;1.1], followed by a resizing of the image to the original size. The clustering in Figure 2 *right* is achieved by color jitter with a brightness, contrast and saturation intensity of 0.4, and a hue intensity of 0.125, with a 100% probability, horizontal and vertical flips, each with 50% probability of application, random rotations with a maximal angle of 360°, and a center crop with a scale of 0.5, followed by a resizing of the image to the original size. For the clustering in Figure 10, we trained the model with a sum of the losses (and corresponding transformations) of the models used in Figure 2. Through a gridsearch

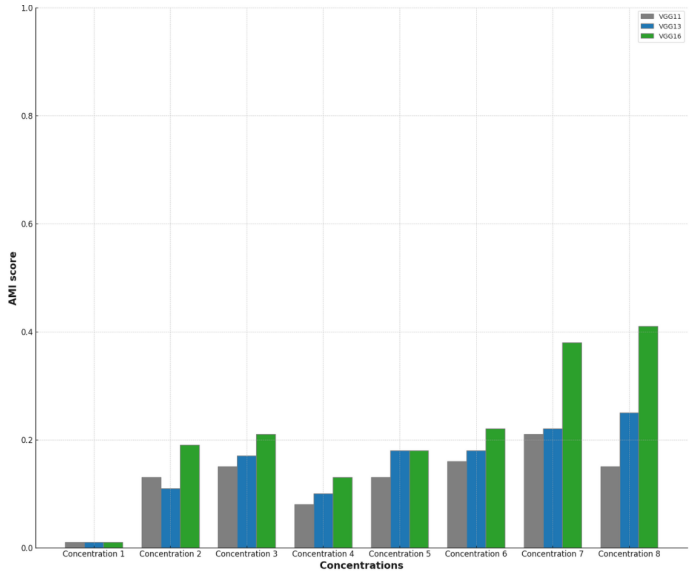


Figure 7: We perform a Kmeans clustering (K=2) on the Nocodazole and untreated images using VGG models of various sizes, and evaluate them using the AMI score Vinh et al. [2010] on a similar number of randomly sampled images of each concentration of Nocodazole and a similar number of untreated cell images. We observe that VGG16 outperforms the other VGG model sizes.

hyperparameter optimization, with the goal of optimizing the AMI score of a Kmeans clustering (K=2), we attributed a coefficient of 0.4 to the loss of the model used in Figure 2 *left*, and a coefficient of 1.0 to the loss of the model used in Figure 2 *right*.

E Additional results

Table 4: The mean and standard deviation of the top1 linear evaluation accuracy, obtained through the training of a Resnet18 architecture using 12 self-supervised approaches on the Cifar10 dataset, are presented. The approaches include VicReg Bardes et al. [2022a], DeepCluster v2 Caron et al. [2018], SWAV Caron et al. [2020], SimCLR Chen et al. [2020a], SimSiam Chen and He [2021], MoCo Chen et al. [2020b], NNCLR Dwibedi et al. [2021], BYOL Grill et al. [2020], VIBCReg Lee and Aune [2021], Barlow Twins Zbontar et al. [2021], and ResSL Zheng et al. [2021]. The experiment involves the uniform sampling of 20 values for the hue transformation probability, while maintaining a fixed maximal intensity of 0.5, and all other transformation parameters are kept constant. The results indicate that despite the variation in the transformation probability, the overall accuracy of each method remains relatively consistent, with a minimal standard deviation value.

	Barlow Twins	Byol	Deep Cluster v2	MoCo V2+	nnByol	nnclr	Ressl	SimCLR	SimSiam	SwaV	Vibcreg	Vicreg
Mean	89.59	92.09	86.9	**92.37**	91.3	89.76	90.21	90.16	89.6	86.96	82.47	89.82
Std	0.73	**0.37**	1.9	0.44	0.57	0.74	0.85	0.87	1.01	1.2	0.89	0.94

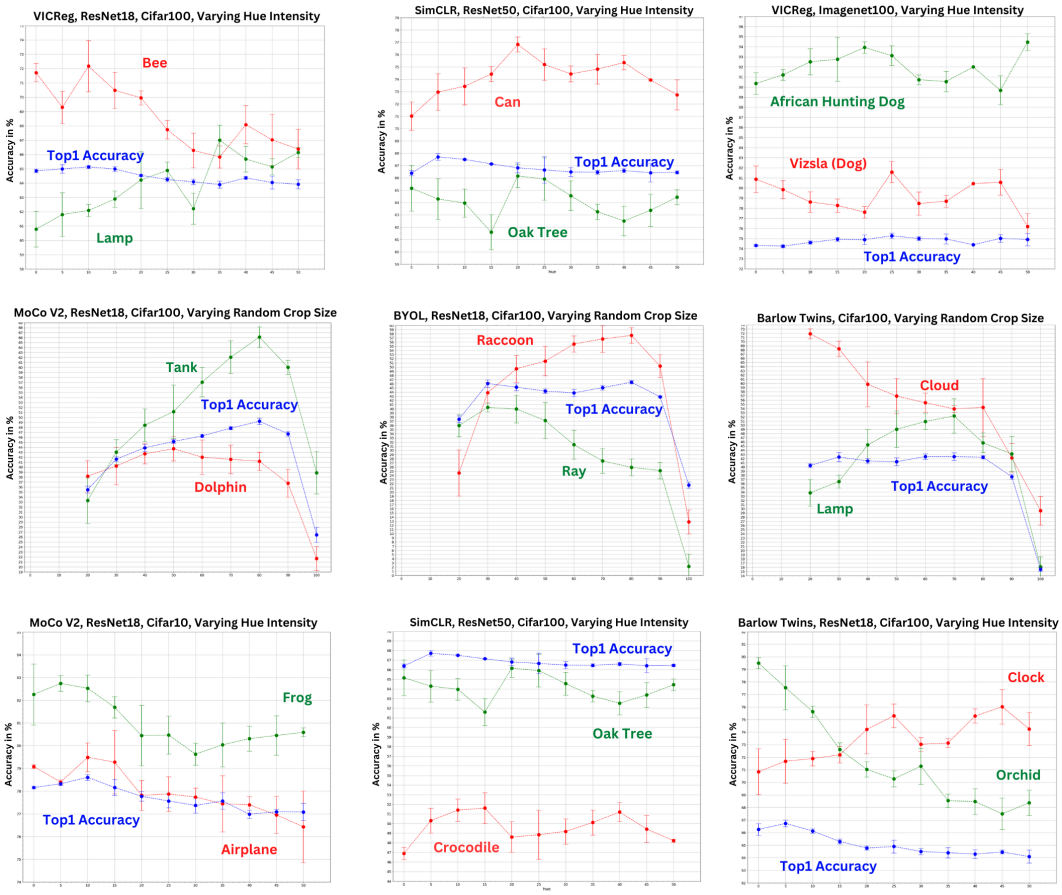


Figure 8: Inter-class accuracy results for Resnet architectures trained with various SSRL methods on the benchmark datasets Cifar10, Cifar100 and Imagenet100, as the parameters of different image transformations are varied. Each dot and associated error bar reflects the mean and standard deviation of five runs for Cifar, each with a different random seed. The results demonstrate that while overall accuracy remains relatively consistent across a range of transformation parameters, these transformations can have a subtle but significant impact on individual class performance, either favoring or penalizing specific classes.

Table 5: Shared Classes with Inter-class Bias Across Different Self-Supervised Learning (SSL) Approaches. This table showcases the number of shared classes demonstrating inter-class bias within varying SSL methods, considering three key transformation parameters: Hue Intensity, Color Jitter Probability, and Crop Size. The SSL methods included in this analysis are Barlow Twins, BYOL, MoCo v2, SimCLR, and VICReg, all trained using a ResNet18 on Cifar100. The results highlight the degree of shared inter-class bias and the influence of different transformations, reinforcing the notion that class-level biases can be transformation and SSL-method dependent.

Number of shared classes with inter-class bias	Hue Intensity	Color Jitter Probability	Crop size
In all 5 SSL approaches	51	0	0
In a minimum of 3 SSL approaches	97	3	4
In a minimum of 2 SSL approaches	99	27	8

Table 6: Comparison of classes with significant negative correlations under variations of Hue Intensity for Linear Evaluation and Fine-tuning phases. The table displays the number of classes with statistically significant negative correlations (p-value < 0.05) for both Linear Evaluation and Fine-tuning under different SSL methodologies, SimCLR, BYOL, and VicReg, all with ResNet18 as the backbone. The last row represents the percentage of shared classes between Linear Evaluation and Fine-tuning that exhibited the same behavior trend (ascending, descending, or random).

Methodology	Simclr	BYOL	VicReg
Resnet18 + Linear Evaluation	97	80	92
Resnet18 + Finetuning	96	100	92
Class Behavior match	45%	52%	53%

Table 7: The outcomes of linear evaluation for various architectures (VGG11, ResNet18, ConvNeXt-Tiny) trained with different self-supervised representation learning (SSRL) methods (MoCov2 Chen et al. [2020b], BYOL Grill et al. [2020]) on the MNIST dataset LeCun et al. [1998]. The models were trained using a set of transformations consisting of random rotations, crops, flips, and random erasing. Notably, the results exhibit consistent patterns across the different self-supervised approaches and backbone architectures, demonstrating the robustness of the observed outcomes.

METHOD	VGG11	RESNET18	CONVNEXT-TINY
BYOL	61.3	62.5	51.6
MOCOV2	62.1	63.8	58.7

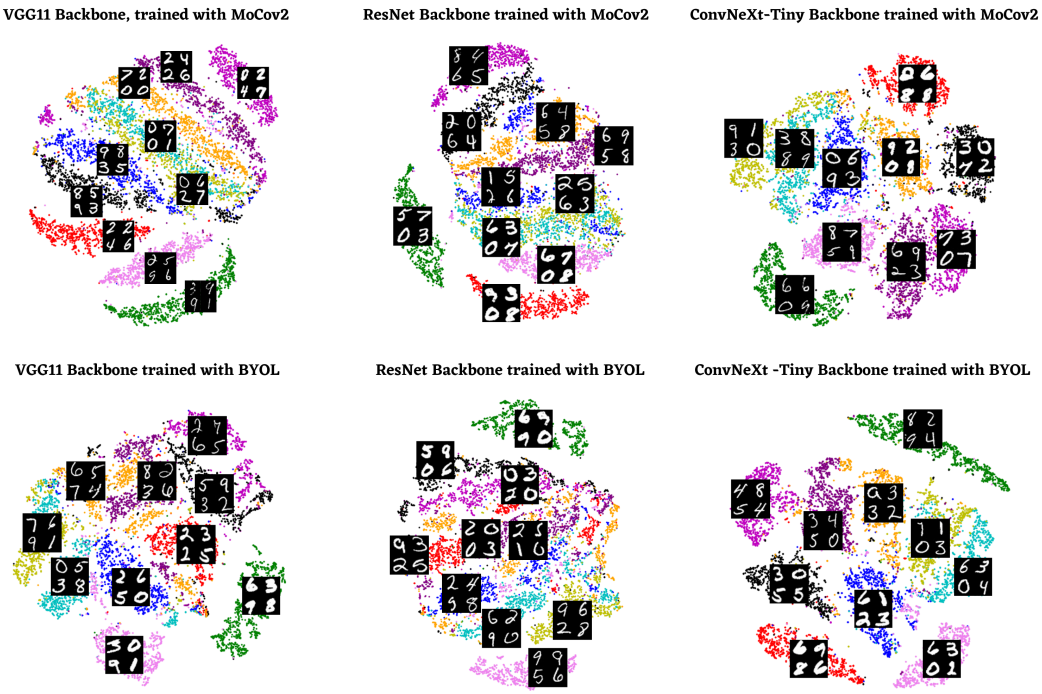


Figure 9: Clustering results of MNIST dataset using various backbones trained with BYOL and MoCov2 as SSRL approaches, using specific image transformations that preserve the handwriting style and line thickness.

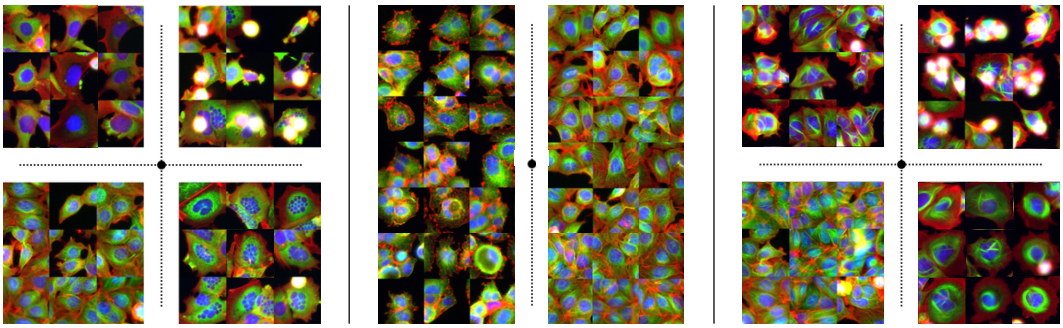


Figure 10: **The clustering results achieved through the utilization of two MoCo v2 losses Chen et al. [2020b] with a VGG13 backbone, each with a distinct set of transformations**, on the Nocodazole *(left)*, Cytochalasin B *(middle)*, and Taxol *(right)* image treatment subsets. One loss employs color jitter, flips, rotation, affine transformation, and random cropping, while the other uses rotations, center cropping, color jitter, and flips. The clustering results demonstrate that the phenotypes of each subset are clearly separated and represented in each cluster, as evidenced by the images closest to its centroïd.