Random Field Augmentations for Self-Supervised Representation Learning

Philip Andrew Mansfield[*] MEMES@GOOGLE.COM
Google Research

Arash Afkanpour[*] ARASH.AFKANPOUR@VECTORINSTITUTE.AI
Vector Institute

Warren Richard Morningstar WMORNING@GOOGLE.COM
Google Research

Karan Singhal KARANSINGHAL@GOOGLE.COM
Google Research

Editors: Sophia Sanborn, Christian Shewmake, Simone Azeglio, Nina Miolane

Abstract

Self-supervised representation learning is heavily dependent on data augmentations to specify the invariances encoded in representations. Previous work has shown that applying diverse data augmentations is crucial to downstream performance, but augmentation techniques remain under-explored. In this work, we propose a new family of local transformations based on Gaussian random fields to generate image augmentations for self-supervised representation learning. These transformations generalize the well-established affine and color transformations (translation, rotation, color jitter, etc.) and greatly increase the space of augmentations by allowing transformation parameter values to vary from pixel to pixel. The parameters are treated as continuous functions of spatial coordinates, and modeled as independent Gaussian random fields. Empirical results show the effectiveness of the new transformations for self-supervised representation learning. Specifically, we achieve a 1.7% top-1 accuracy improvement over baseline on ImageNet downstream classification, and a 3.6% improvement on out-of-distribution iNaturalist downstream classification. However, due to the flexibility of the new transformations, learned representations are sensitive to hyperparameters. While mild transformations improve representations, we observe that strong transformations can degrade the structure of an image, indicating that balancing the diversity and strength of augmentations is important for improving generalization of learned representations.

Keywords: Self-supervised learning, Representation learning, Gaussian random fields, Local symmetry

1. Introduction

Data augmentations play a crucial role in joint embedding self-supervised representation learning methods. They specify the transformations under which the representations must remain invariant. In the absence of any prior knowledge, most self-supervised learning methods assume that each data point is semantically different from other examples in the data set. Data augmentations, on the other hand, relate each example to its transformed

[*] Equal contribution

versions via a soft positive label. While some previous work studied the impact of augmentations on representations (Chen et al., 2020a; Caron et al., 2020), for the most part this remains an under-explored area in self-supervised learning. Since these transformations specify what representations learn, a natural question is whether additional diverse transformations improve generalizability and robustness of representations.

In this work we introduce and study a family of visual local transformations based on *Gaussian random fields*. In particular we define local spatial and color transformations to modify the position and color of pixels using Gaussian random fields. The new transformations are a generalization of the standard affine (rotation, translation, etc.) and color transformations used in many methods (Chen et al., 2020a; Grill et al., 2020; Chen and He, 2021) and operate at the pixel level. Our empirical results in both in-distribution and out-of-distribution tasks demonstrate the effectiveness of these transformations for representation learning.

2. Related work

2.1. Joint Embedding Methods

Joint embedding self-supervised learning methods use a variety of objective functions to create invariance of representations across multiple views of the same images. These views are usually generated by applying several transformations that do not change the semantics of an image. Based on the objective function, these methods can be divided into several categories. For example, contrastive methods such as CPC (Oord et al., 2018), SimCLR (Chen et al., 2020a) and MoCo (He et al., 2020) use InfoNCE contrastive loss to pull representations of different augmentations of an image together, while pushing representations of different images apart. Clustering methods, e.g., DeepCluster (Caron et al., 2018) and SwAV (Caron et al., 2020) use a combination of clustering and constrastive loss to learn a similar representation for different views of an image. Canonical correlation analysis methods, such as Barlow Twins (Zbontar et al., 2021) and VICReg (Bordes et al., 2023) rely on correlation analysis of features in the representation space. Their objective is defined to maximize correlation of the same feature across multiple views, while decorrelating different features. Self-distillation methods such as BYOL (Grill et al., 2020) and SimSiam (Chen and He, 2021) use a dual encoder architecture where one encoder is a slightly different version of the other (e.g., an exponential moving average encoder in BYOL). The model is trained by maximizing similarity between representations of the encoders fed with two views of the same image.

In contrast to joint embedding methods, representation learning based on masked image modeling does not rely on data augmentations. Similar to the masked token prediction task in BERT pretraining (Devlin et al., 2018) the general principle is to mask parts of an image and minimize a loss to reconstruct them given the remaining parts. Most notably He et al. (2022) takes advantage of vision transformers (Dosovitskiy et al., 2020) to learn representations with this approach.

2.2. Image Augmentations

SimCLR (Chen et al., 2020a) studied the effectiveness of several augmentations including random crop, cutout, color jitter, Sobel filter, Gaussian blur, Gaussian noise, and global rotation. They examined individual and pairs of augmentations for representation learning. They observed that random crop and color jitter are the most effective augmentations when these representations are used for ImageNet classification. Most subsequent work in self-supervised representation learning, e.g. Chen et al. (2020b); Zbontar et al. (2021); Bardes et al. (2021), use the same set of augmentations. One exception is multi-crop proposed by Caron et al. (2020) where multiple small crops are taken as additional views of a source image. In this case the model is trained to produce the same representation for small crops and views generated by the composition of other augmentations.

Bordes et al. (2023) studied the impact of different combinations of augmentations. They showed the combination of random crop and a grayscale transformation is quite competitive, measured by classification accuracy on ImageNet, to the full augmentation set.

One of the shortcomings of the current augmentations is that they are selected to achieve the best performance on ImageNet classification. It is possible that representations learned via these augmentations do not perform well on other downstream tasks. Ericsson et al. (2021) investigated how the learned invariances affect the performance across a diverse set of downstream tasks. They showed that in some tasks a subset of data augmentations outperforms the default combination of SimCLR augmentations.

Image augmentations remain under-explored given their importance to representation learning, especially for out-of-distribution downstream tasks, motivating our work.

3. Random Field Transformations

3.1. Gaussian Random Fields

A local transformation is characterized by one or more parameter fields where each (pixel) position has its own transformation parameter(s). A random parameter field ensures diversity of transformations. At the same time, complete independence of parameters results in distortions that make the final image unrecognizable. Therefore parameters must be relatively slowly varying continuous functions of spatial coordinates, and nearby values of the random field must be suitably correlated with each other. Gaussian random fields offer a convenient mathematical tool for this purpose. Here we provide a brief description of Gaussian random fields. There are numerous resources on this topic, cf. Adler et al. (2007).

A random field is a stochastic process with a structured parameter space. Let $\mathcal{X}$ denote a parameter space, such as the Euclidean space. Given $\mathcal{X}$, a random field ϕ is a collection of random variables

$$\{\phi(x) : x \in \mathcal{X}\}.$$

In a Gaussian random field any finite number of variables constitute a multivariate Gaussian distribution. Therefore, a Gaussian random field is fully characterized by its mean (μ) and covariance (Σ) functions:

$$\begin{aligned} \mu(x) &= \mathbb{E}[\phi(x)], \\ \Sigma(x, y) &= \mathbb{E}[(\phi(x) - \mu(x))(\phi(y) - \mu(y))]. \end{aligned}$$

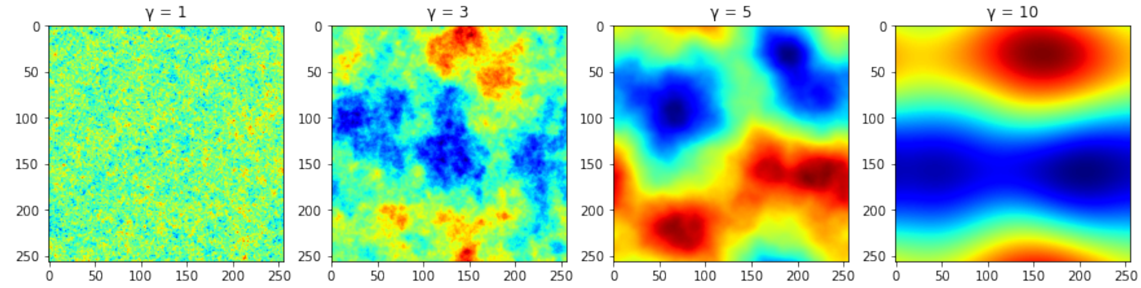

Figure 1: Gaussian random fields with different values of the power law exponent.

If the mean of a random field is constant across $\mathcal{X}$ and the covariance is a function of the difference $(x - y)$ only, then the random field is *homogeneous*. Additionally, if the covariance is a function of the Euclidean distance $|x - y|$ then ϕ is also *isotropic*. With some abuse of notation an isotropic random field is usually written as $\Sigma(x, y) = \Sigma(|x - y|)$. A homogenous and isotropic Gaussian random field is particularly interesting because it is fully characterized by its covariance (equivalently correlation) function, and this function only depends on the distance between points in the parameter space.

Generating a random field in the spatial domain is computationally expensive. However, it can be easily calculated in the frequency domain. The power spectrum, which is the Fourier transform of the correlation function, characterizes a Gaussian random field in the frequency domain. In our experiments we specified the power spectrum as *power law*: $P(k) \propto k^{-\gamma}$, where γ controls the correlation of points in the spatial domain: larger values result in higher correlation among distant points. Figure 1 shows examples of random fields with different γ values.

3.2. Image Transformations with Gaussian Random Fields

Spatial affine transformations such as rotation, translation, scaling, etc. are usually parameterized by a few parameters that specify the magnitude of transformation globally. Consider the translation transformation. It requires two parameters, t_X and t_Y which determine the amount of translation across X and Y axes respectively. One way to generalize this transformation is to use pixel-specific translation values, i.e. $t_X(x, y)$ and $t_Y(x, y)$, where t_X and t_Y are Gaussian random fields. To ensure images remain recognizable, transformations are set up such that local changes are small. This is primarily controlled by γ, the exponent of the power law used as the spectrum function. We loosely use the term *kernel width* to refer to this parameter. A large value for kernel width indicates a strong correlation between pixels even if they are far apart, resulting in a smoother random field. In addition, we limit the magnitude of the random field by a parameter α such that $-\alpha \leq \theta(x, y) \leq \alpha$, where θ denotes the random field. Eq. 1 shows the general form of a local affine transformation applied to a 2-dimensional source point (x^s, y^s).

$$\begin{bmatrix} x^s \\ y^s \end{bmatrix} = \begin{bmatrix} \theta_{11}(x,y) & \theta_{12}(x,y) & \theta_{13}(x,y) \\ \theta_{21}(x,y) & \theta_{22}(x,y) & \theta_{23}(x,y) \end{bmatrix} \begin{bmatrix} x^t \\ y^t \\ 1 \end{bmatrix} \tag{1}$$

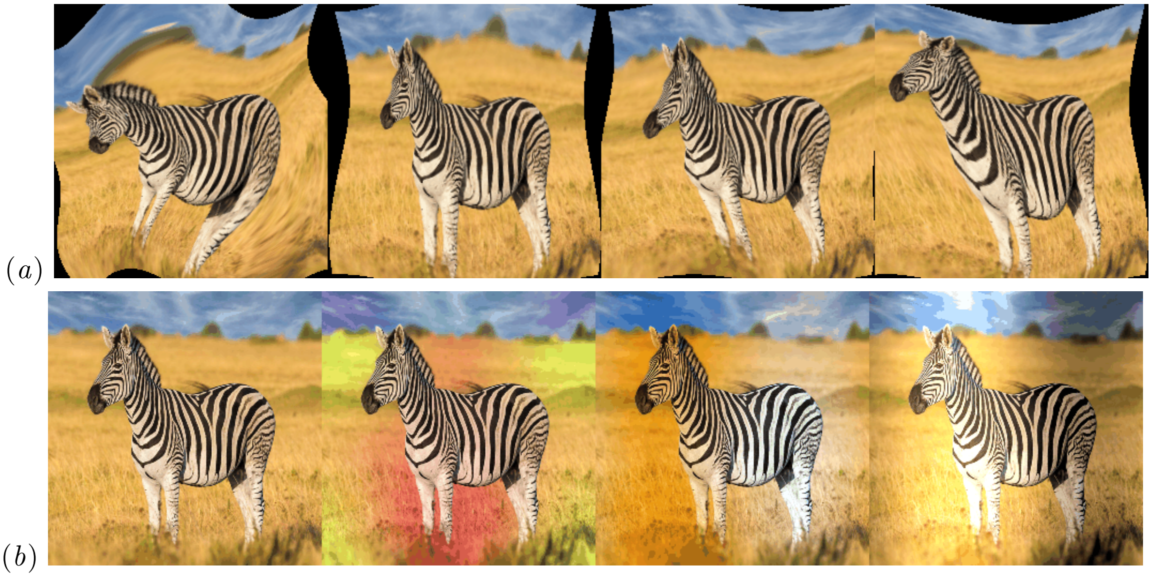

Figure 2: Local transformations with Gaussian random fields. Top left to top right: rotation, scale, shear, translation. Bottom left to bottom right: original, hue, saturation, brightness.

As is common in Computer Graphics, we multiply the transformation matrix by the target coordinates, (x^t, y^t), to fully cover the target space. Multiplication by the source coordinates on the other hand could result in undefined values for some target coordinates. See more details in Foley et al. (1994).

In our experiments we focused on four common affine transformations: rotation, scaling, shearing, and translation. For example, the local scale transformation matrix is given by,

$$\theta_{scale}(x, y; \gamma_x, \gamma_y, \alpha_x, \alpha_y) = \begin{bmatrix} 1 + g_x(x, y; \gamma_x, \alpha_x) & 0 & 0 \\ 0 & 1 + g_y(x, y; \gamma_y, \alpha_y) & 0 \end{bmatrix} \tag{2}$$

where g_x and g_y are independent Gaussian random fields, parameterized by smoothness parameters γ and scale factors α such that $-\alpha \leq g(x, y; \gamma, \alpha) \leq \alpha$. In Eq. 2 we use different random fields for the X and Y axes. The matrices of other affine transformations are available in Appendix A.

We apply local color transformations to hue, saturation, and value channels separately. For each channel, a Gaussian random field is added to the channel values to obtain the new values. Figure 2 shows examples of local affine and color transformations.

4. Empirical Results

In all experiments we use SimCLR (Chen et al., 2020a) as the self-supervised representation learning method. Pretraining of the encoder is performed on the ImageNet training split with 1.2 million images. Following the linear probing protocol of previous papers, we evaluated each setting by training a linear classifier on the output representations of the frozen

	ImageNet	**iNaturalist**
	Top-1 / Top-5	**Top-1 / Top-5**
Baseline (SimCLR augmentations)	0.7056 / 0.9022	0.3873 / 0.5983
Local color jitter	0.7045 / 0.9013	0.3964 / 0.6071
Local rotate	0.7007 / 0.8945	0.4159 / 0.6171
Local scale	0.7102 / 0.8964	0.4174 / 0.6245
Local shear	0.7219 / 0.9031	0.4102 / 0.6228
Local translate	**0.7223 / 0.9015**	**0.4231 / 0.6267**

Table 1: Effect of atomic random field augmentations (in addition to SimCLR augmentations) on learned representations measured by downstream classification accuracy. Bold numbers indicate the highest Top-1 accuracy.

encoder network. Our downstream tasks are image classification on two datasets: ImageNet (in-distribution) and iNaturalist 2018 (out-of-distribution). In each case a linear classifier was trained on the training split of the dataset and then evaluated on the validation split.

In all experiments we apply local transformations in addition to the standard SimCLR augmentations (Chen et al., 2020a). Random field augmentations are applied before the SimCLR augmentations, with the exception of crop and resize, which we apply as the first augmentation to resize images to 224×224 in order to reduce the computational cost of local transformations.

4.1. Atomic Local Transformations

In this experiment we evaluate five local transformations: color jitter, rotation, scaling, shearing, and translation. We choose each parameter range so that the resulting transformation does not make the images unrecognizable. For each image, γ (the random field smoothness parameter) is sampled uniformly from $[7, 10]$. The random field scale factor (α) is uniformly sampled from $[0, 1/3]$. A local transformation is applied to each of the two views of SimCLR with probability 0.8. Table 1 shows Top-1 and Top-5 classification accuracy on the ImageNet and iNaturalist 2018 downstream tasks.

In the ImageNet task (in-distribution) local scale, shear, and translate outperform the baseline. Local color jitter and rotation, on the other hand degrade accuracy. In the iNaturalist task (out-of-distribution) all local transformations outperform the baseline. In both cases local color jitter slightly underperforms local affine transformations, which could indicate that the classification task is more sensitive to local color changes than local spatial changes. Another observation is that local rotation performance is generally slightly worse than other local affine transformations. This could be due to larger structural changes made to an image by local rotation compared to other local affine transformations (see Figure 2 for an example).

ImageNet	$\alpha \in [0, 1/3]$	$\alpha \in [0, 2/3]$	$\alpha \in [0, 1]$
$\gamma \in [3, 7]$	0.6981 / 0.8896	0.6879 / 0.8788	0.6595 / 0.865
$\gamma \in [7, 10]$	**0.7223 / 0.9015**	0.6939 / 0.8873	0.6917 / 0.8879
$\gamma \in [3, 10]$	0.7045 / 0.8939	0.6937 / 0.8808	0.6723 / 0.8697
iNaturalist	$\alpha \in [0, 1/3]$	$\alpha \in [0, 2/3]$	$\alpha \in [0, 1]$
$\gamma \in [3, 7]$	0.4046 / 0.6055	0.4043 / 0.6021	0.3964 / 0.5895
$\gamma \in [7, 10]$	**0.4231 / 0.6267**	0.4080 / 0.6169	0.4180 / 0.6245
$\gamma \in [3, 10]$	0.4183 / 0.6255	0.4136 / 0.6092	0.4043 / 0.6003

Table 2: Top-1 / Top-5 classification accuracy of representations trained with local translation over different ranges of γ and α. Top: ImageNet, bottom: iNaturalist 2018.

4.2. Effect of Random Field Parameters

We performed a grid search on the two parameters of the random fields, γ and α. For each parameter we specified different intervals for uniform sampling. The γ intervals include $[3, 7]$, $[7, 10]$, and $[3, 10]$. Usually $\gamma < 3$ yields strong local distortions that destroy the global structure, making an image unrecognizable. On the other end $\gamma > 10$ yields almost no difference to augmentations with $\gamma = 10$. The α intervals include $[0, 1/3]$, $[0, 2/3]$, and $[0, 1]$. In this experiment we focus on local translate and apply it to both views of SimCLR, each with probability 0.8. Similar to the previous experiment, we follow the standard linear probing protocol by training a linear classifier on the output of a frozen encoder. Table 2 shows top-1 and top-5 accuracy numbers on each validation set.

Among these combinations $\gamma \in [7, 10], \alpha \in [0, 1/3]$ leads to the best classification accuracy on both downstream tasks. Strong distortions, achieved by smaller γ or larger α could lead to transformations that change the spatial structure of images too drastically, leading to worse performance of representations in downstream tasks.

With the best combination of parameters, i.e. $\gamma \in [7, 10], \alpha \in [0, 1/3]$, we performed a sweep over the probability parameter that determines how often a random field transformation is applied to the image. Table 3 shows the results. Broadly, as the probability value increases, downstream classification accuracy increases too. This trend continues until reaching maximum accuracy at $p = 0.8$. Pushing the probability value further to 1.0, however, leads to a decline in accuracy, similar to other work that has observed benefit to applying augmentations stochastically.

4.3. Composite Transformations

We study composite affine transformations in this section. For simplicity we only consider the composition of two atomic local transformations. For each atomic transformation γ and α are sampled uniformly from $[7, 10]$, and $[0, 1/3]$ respectively. In order to ensure that the combination of transformations remain within the same bounds as individual transformations the scale factor of each transformation is multiplied by $1/\sqrt{2}$ before application.[1] A

[1] To combine N transformations, this coefficient should be $1/\sqrt{N}$.

Probability	**ImageNet** **Top-1 / Top-5**	**iNaturalist** **Top-1 / Top-5**
0.0	0.7056 / 0.9022	0.3873 / 0.5983
0.2	0.7134 / 0.8971	0.3906 / 0.6030
0.4	0.7207 / 0.9034	0.3954 / 0.6060
0.6	0.7140 / 0.9036	0.3903 / 0.6060
0.8	0.7223 / 0.9015	0.4231 / 0.6267
1.0	0.6929 / 0.8893	0.3937 / 0.6031

Table 3: Top-1 / Top-5 classification accuracy of downstream classification for different values of the probability parameter.

ImageNet	Rotate	Scale	Shear	Translate
Rotate	0.7007	0.7092	0.7155	0.716
Scale	-	0.7102	0.7235	0.7088
Shear	-	-	0.7219	0.7119
Translate	-	-	-	0.7223

iNaturalist	Rotate	Scale	Shear	Translate
Rotate	0.4159	0.4026	0.4006	0.4044
Scale	-	0.4174	0.3848	0.3953
Shear	-	-	0.4102	0.3966
Translate	-	-	-	0.4231

Table 4: Top-1 downstream classification accuracy of composite transformations on ImageNet (top) and iNaturalist (bottom) data sets. Diagonal elements show the accuracy of atomic transformations.

composite transformation is then formed by multiplying the matrices of individual transformations in random order. Similar to the previous experiments each composite transformation is applied with probability 0.8. Table 4 shows the results. While in the ImageNet task (in-distribution) combining transformations generally improves performance, in the iNaturalist task (out-of-distribution) performance degrades by combining local transformations. Since combining local transformations can generally be interpreted as stronger distortions in the local structure of an image, these results indicate that too strong distortions could have a negative effect on representations. This observation is also supported by the results in Section 4.2.

5. Conclusion

Image augmentations play a crucial role in joint embedding self-supervised learning methods. Yet different augmentation methods have been studied minimally in this context. This motivates work exploring whether additional diverse augmentations could result in more robust and generalizable representations. In this paper we introduced random field augmentations as a generalization of some of the previous forms of augmentations, in particular crop-and-resize (equivalently scale and translate) and color jitter, which according to Chen et al. (2020a) are the two most effective augmentations for SimCLR. Our new transformations vastly increase the space of augmentations by enabling coordinate-based transformations where transformation parameters are selected according to Gaussian random fields.

We performed multiple empirical studies for in-distribution and out-of-distribution cases. These studies include a comparison of different types of transformations, measuring the effect of transformation parameters on the quality of representations and a comparison of composite transformations. The results showed effectiveness of the new transformations when applied in addition to the standard transformations of SimCLR. Due to the flexibility of the new transformations, careful hyperparameter tuning must be performed on the random field parameters. While mild transformations generally improve representations, we showed that strong transformations, which could significantly change the structure of an image, led to performance degradation.

Future work can apply random field augmentations to different self-supervised representation learning methods with different model architectures and downstream tasks, studying the effect of these flexible transformations on generalization in different contexts.

References

Robert J Adler, Jonathan E Taylor, et al. *Random fields and geometry*, volume 80. Springer, 2007.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

Florian Bordes, Randall Balestriero, and Pascal Vincent. Towards democratizing joint-embedding self-supervised learning. *arXiv preprint arXiv:2303.01986*, 2023.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149, 2018.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

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, 2020a.

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, 2020b.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Linus Ericsson, Henry Gouk, and Timothy M Hospedales. Why do self-supervised models transfer? investigating the impact of invariance on downstream tasks. *arXiv preprint arXiv:2111.11398*, 2021.

James D Foley, Andries Van Dam, Steven K Feiner, John F Hughes, and Richard L Phillips. *Introduction to computer graphics*, volume 55. Addison-Wesley Reading, 1994.

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.

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.

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.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

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.

Appendix A. Local Transformation Matrices

In all transformations the center of the coordinate system is the center of the image. Let $g(x, y; \gamma, \alpha)$, $g_x(x, y; \gamma_x, \alpha_x)$ and $g_y(x, y; \gamma_y, \alpha_y)$ be Gaussian random fields. The atomic local affine transformations are defined as follows:

Local Rotate

$$\theta_{rotate}(g) = \begin{bmatrix} \cos \pi g & -\sin \pi g & 0 \\ \sin \pi g & \cos \pi g & 0 \end{bmatrix}$$

Local Scale

$$\theta_{scale}(g_x, g_y) = \begin{bmatrix} 1 + g_x & 0 & 0 \\ 0 & 1 + g_y & 0 \end{bmatrix}$$

Local Shear

$$\theta_{shear}(g_x, g_y) = \begin{bmatrix} 1 & g_x & 0 \\ g_y & 1 & 0 \end{bmatrix}$$

Local Translate

$$\theta_{translate}(g_x, g_y) = \begin{bmatrix} 1 & 0 & g_x \\ 0 & 1 & g_y \end{bmatrix}$$