Detecting Manual Alterations in Biological Image Data Using Contrastive Learning and Pairwise Image Comparison

Georgii Nekhoroshkov
MIPT
Moscow, Russia
nekhoroshkov.gs@phystech.edu

Daniil Dorin
MIPT
Moscow, Russia
dorin.dd.contact@gmail.com

Andrii Hraboviy
MIPT
Moscow, Russia
grabovoy.av@phystech.edu

Abstract

In this paper, we address the problem of detecting manipulations in biological images. Ensuring the integrity of biological image data is essential for reliable scientific research. The study focuses on developing a model for pairwise image comparison using contrastive learning, demonstrating high pairwise comparison metrics to detect manual modifications or more subtle alterations. The proposed method outperforms state-of-the-art models, including SimCLR and Barlow Twins, in the task of biological image comparison on complex cell datasets. This work contributes to automated fraud detection and data validation in biological research.

Keywords: Machine Learning, Pairwise Image Comparison, Self-Supervised Learning, Fine-Tuning, Automated Fraud Detection, Detecting Data Alterations

1 Introduction

Our work aims to develop a machine learning solution for the problem of reusing existing biological and medical snapshots to demonstrate results in newly published biological articles. Fake images negatively impact on medicine by providing false or fabricated results and undermining the credibility of new scientific work in these fields. Advances in self-supervised learning for images ([1], [2], [3], [4], [5]). shows that it is possible to achieve the same results or even outperform supervised representations. Existing state-of-the-art self-supervised learning approaches demonstrate remarkable results in pairwise image comparison tasks (`SimCLR` [6], `CLIP` [7], Barlow Twins [8]). However, their performance significantly worsens when applied to complex biological data. It requires developing model that is more sensitive to subtle changes in the image content while remaining resistant to various manual alterations, such as color jittering, flipping, rotation and noise application. At present, the problem of matching biological and medical images remains unsolved due to the complexity of distinguishing snapshots of similar objects, where differences can only be identified by experts in the field.

We propose a solution, based on `Barlow Twins` [8], trained and fine-tuned specifically for complex biological scans. The model belongs to the family of self-supervised learning (SSL) methods, which have been proven to be competitive with supervised representations ([9]). We experiment with different loss and training specifics in order to achieve best accuracy metrics.

Preprint. Under review.

By leveraging a pretrained `ResNet50` backbone model from `Barlow Twins` [8], it does not require large snapshot datasets to achieve state-of-the-art accuracy on the dataset, formed of different cell images from `Cell Image Library` and `Kaggle` datasets. This solution can be widely used by biological articles proofreaders to verify the authenticity of provided images and detect borrowings from known datasets.

Outline. The remainder of this paper is organized as follows. First, we formally define the problem we aim to solve and outline the key challenges. Next, we present our proposed solution in detail, discussing its architecture and specific design and training pipeline choices. We then conduct a series of computational experiments to evaluate our model, comparing its performance with that of existing state-of-the-art approaches, specifically `Barlow Twins` [8]. Finally, we conclude the paper by summarizing our findings and discussing the broader implications and potential impact of our approach on the problem domain.

2 Problem

Given dataset $\mathcal{D}$, consisting of N biological snapshots:

$$\mathcal{D} = \{d_i \in \mathcal{S}, i \in [0, N)\}, \, \mathcal{S} \subseteq {\mathbb{R}_+}^{H \times W \times C}$$

where $\mathcal{S}$ is the image space.

For simplicity, we will refer to a pair of images with the same content before alterations as a *similar* pair; otherwise, it will be called *dissimilar*.

Our goal is to construct a model $\mathcal{M}$ using self-supervised contrastive learning (SSCL) that can effectively distinguish between dissimilar pairs of images while accurately identifying similar ones. This approach enables the model to learn meaningful representations without the need for extensive labeled data, making it particularly valuable in domains where annotations are scarce or expensive to obtain.

Let x and y be two input images such that $x, y \in \mathcal{S}$, where $\mathcal{S}$ is the set of all images. The model $\mathcal{M}$ is composed of two primary components:

$$\mathcal{M}(x, y) = h(f(x), f(y))$$

Here, f is an encoder network that maps each image into a d-dimensional representation space:

$$f(x) = v_x \in \mathbb{R}^d, \quad f(y) = v_y \in \mathbb{R}^d$$

These embeddings v_x and v_y are then passed to a similarity function or linear classifier h, which produces a scalar output:

$$h(v_x, v_y) = s \in [0, 1]$$

The output value s indicates the model's prediction: $s \approx 1$ for similar image pairs, and $s \approx 0$ for dissimilar pairs. A common implementation for h is a not complex neural network $\mathcal{N}$ applied to the concatenation of v_x and v_y:

$$h(v_x, v_y) = \sigma(\mathcal{N}(v_x, v_y))$$

where σ is the sigmoid activation function.

To evaluate the model's effectiveness, we rely on several classification performance metrics. These include accuracy, precision, recall, F1-score and ROC AUC. These metrics provide a clear way to evaluate our model and directly compare it to leading methods like `Barlow Twins` and `SimCLR`.

3 Method

The challenge of detecting reused biological and medical images lies in the difficulty of distinguishing between visually similar images while maintaining invariance to various transformations. Our dataset $\mathcal{D}$ consists of biological snapshots $d_i \in [0, 256)^{224\times224\times3}$, obtained from publicly available sources. We define a *solution* as any method intended to address the stated problem, and we refer to the *model* as the SSCL solution, proposed in our work.

The structure of our model is inspired by the `Barlow Twins` framework [8], which we adapt and refer to as `Barlow Twins Adaptation (BTA)`. The model comprises three deep neural network components: an encoder f, a projector p, and a linear classifier s that outputs a single scalar value in the range $[0, 1]$, representing the predicted similarity between input pairs.

The encoder is based on a pretrained `ResNet50`, obtained from the original `Barlow Twins` repository [8], and is used to extract high-level feature representations from input images. The projector follows the architecture proposed in the original work, with a slight modification: the first layer has an input size of 2048, while the subsequent three layers each have an input size of 4096. This component maps the encoder's output into a space suitable for computing the Barlow Twins loss.

The classifier is a lightweight, trainable neural network that takes the projected embeddings as input and outputs a similarity score through a sigmoid activation function. This component is trained using binary cross-entropy loss to distinguish between positive and negative pairs.

Following training approach is considered classical. To train the projector, we begin with a batch of images X. Each image x_i is augmented in two different ways to produce two modified versions, x_i^A and x_i^B. These two batches of augmented images, X^A and X^B, are then processed by the embedding function f to yield embeddings Y^A and Y^B. The embeddings are subsequently passed through the projector function p, resulting in the projected embeddings Z^A and Z^B. These projected embeddings are used to compute the loss function $\mathcal{L}$ as proposed in `Barlow Twins` [8]:

$$\mathcal{L}_{BT} = \sum_i (1 - \mathcal{C}_{ii})^2 + \lambda_{BT} \sum_i \sum_{j \neq i} \mathcal{C}_{ij}^2,$$

where λ is a positive constant, and $\mathcal{C}_{ij}$ is the cross-correlation matrix between the outputs of the two networks along the batch dimension, defined as:

$$\mathcal{C}_{ij} = \frac{\sum_b z_{b,i}^A z_{b,j}^B}{\sqrt{\sum_b (z_{b,i}^A)^2} \sqrt{\sum_b (z_{b,j}^B)^2}}.$$

For accuracy evaluation, we train the similarity network function s while keeping the model's weights frozen. Given a batch of images X, we generate two batches, X^A and X^B, where X^A comprises images from X with random modifications, and X^B is a shuffled version of X with different modifications applied. Consequently, for each image x_i, there is a corresponding modified image x_i^A and a randomly transformed image x_i^B, with a probability of 0.4 that they form a *similar* pair. These batches are processed sequentially through the embedding function f, returning Y^A and Y^B, and the projector p, returning resulting embeddings Z^A and Z^B, which then being fed into the similarity function s. The function s produces a vector P with values in the range $[0, 1]$, where each element P_i represents the estimated likelihood that the pair (y_i^A, y_i^B) is similar. The similarity network is trained using Binary Cross-Entropy (BCE) loss.

We compare this training approach with different one. Freezing backbone's weights, we train projector and similarity net together with a combined loss:

$$\mathcal{L} = \mathcal{L}_{BCE} + \lambda \cdot \mathcal{L}_{BT}$$

where λ is a positive trade-off constant. This loss function enables joint training of the linear classifier and the projector by minimizing a combined loss. The combined loss consists of the previously described loss for projector training, which encourages the embeddings to be invariant and decorrelated, and the binary cross-entropy (BCE) loss, which guides the linear classifier based on labeled pairs. By integrating both objectives, the model can possibly learn more informative representations.

Algorithm 1 BTA

```
Require: Augmented batches $X^A$ and $X^B$, backbone $f$, projector $p$, weight $\lambda_{BT}$
    Projector training:
    for $i \in \{A, B\}$ do:
        $Y_i = f(\mathcal{X}_i)$
        $Z_i = p(Y_i)$
        $Z_i = Normalize(Z_i)$
    end for
    $\mathcal{C} = \frac{1}{N} Z_A^T Z_B$
    $\mathcal{L}_{inv} = \sum_i (1 - \mathcal{C}_{ii})^2$
    $\mathcal{L}_{red} = \sum_{i \neq j} \mathcal{C}_{ij}^2$
    $\mathcal{L}_{BT} = \mathcal{L}_{inv} + \lambda_{BT} \cdot \mathcal{L}_{red}$
    return $\mathcal{L}_{BT}$

Require: Augmented batches $X^A$ and $X^B$, target $target$, backbone $f$, projector $p$, linear classifier $s$.
    Linear classifier training:
    for $i \in \{A, B\}$ do:
        $Y_i = f(\mathcal{X}_i)$
        $Z_i = p(Y_i)$
        $Z_i = Normalize(Z_i)$
    end for
    $preds = s(Z_A, Z_B)$
    $\mathcal{L}_{BCE} = BCELoss(preds, target)$
    return $\mathcal{L}_{BCE}$
```

Algorithm 2 BTA-combined

```
Require: Augmented batches $X^A$ and $X^B$, backbone $f$, projector $p$, linear classifier $s$, weight $\lambda_{BT}$, loss-factor $\lambda$
    Combined training:
    for $i \in \{A, B\}$ do:
        $Y_i = f(\mathcal{X}_i)$
        $Z_i = p(Y_i)$
        $Z_i = Normalize(Z_i)$
    end for
    $\mathcal{C} = \frac{1}{N} Z_A^T Z_B$
    $\mathcal{L}_{inv} = \sum_i (1 - \mathcal{C}_{ii})^2$
    $\mathcal{L}_{red} = \sum_{i \neq j} \mathcal{C}_{ij}^2$
    $\mathcal{L}_{BT} = \mathcal{L}_{inv} + \lambda_{BT} \cdot \mathcal{L}_{red}$
    $Z^A = RepeatInterleave(Z^A, dim = 0)$
    $Z^B = Repeat(Z^B, dim = 1)$
    $\mathcal{L}_{BCE} = s(Z^A, Z^B)$
    $\mathcal{L}_{BTA} = \mathcal{L}_{BCE} + \lambda \cdot \mathcal{L}_{BT}$
    return $\mathcal{L}_{BTA}$
```

Due to the combined training, the algorithm computes the embeddings Z^A and Z^B only once to calculate the combined loss $\mathcal{L}_{BTA}$. The BCE loss is evaluated for every pair (Z_i^A, Z_j^B), unlike in the linear classifier training used in BTA.

To our model trained this way we will refer as `BTA-combined`. Accuracy metrics comparisons of these and state-of-the-art methods are demonstrated in the following section.

4 Computational Experiment

In this section, we present experiments comparing our model—trained using two distinct strategies—with the `Barlow Twins` method on a specially curated cell dataset. To ensure a fair and meaningful comparison, we adopt the same linear classifier architecture as the model head in both our approaches and the adapted `Barlow Twins`. Our goal is to evaluate whether our solution can outperform the baseline in terms of several accuracy-related metrics, including accuracy, F1-score, precision, recall, and AUC ROC.

Our dataset consists of 700 microscopic images representing human, animal, and plant cells. These images were sourced from publicly available repositories, including `Kaggle` and the `Cell Image Library`. The dataset was divided into training (80%) and validation (20%) subsets, with all images being distinct to promote diversity and prevent data leakage during model evaluation.

We introduce and compare two variants of our proposed architecture:

- **BTA (Barlow Twins Adaptation)**: In this setup, we first train the projector for 200 epochs, followed by training the linear classifier for an additional 100 epochs.
- **BTA-Combined**: This variant involves joint training of the projector and the linear classifier over 200 epochs.

To maintain consistency, the linear classifier in the original `Barlow Twins` implementation was also trained for 100 epochs under the same settings. All models were optimized using the `AdamW` optimizer with a cosine annealing learning rate schedule defined as follows:

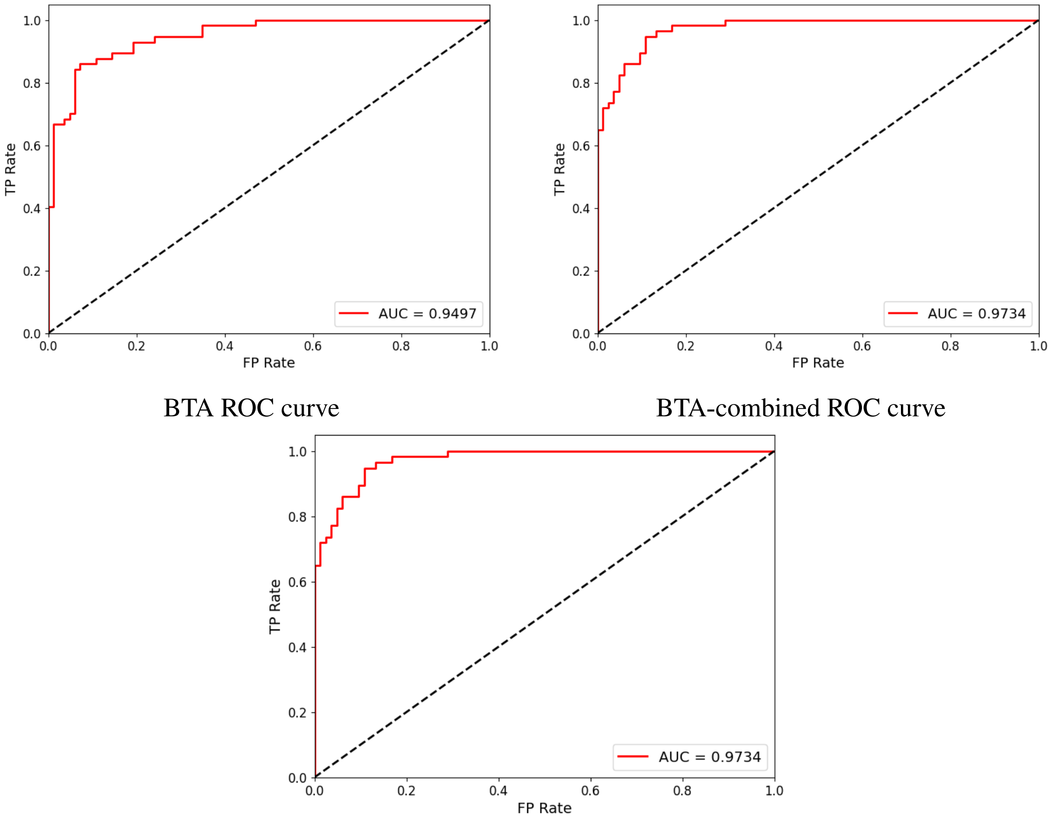
$$q_k = \frac{1}{2} \cdot (1 + \cos(\pi \cdot \frac{k}{K}))$$

$$\gamma_k = \gamma_{start} \cdot q_k + \gamma_{end} \cdot (1 - q_k)$$

where k is the current epoch, K is the total number of epochs, and the learning rates are set as $\gamma_{start} = 3 \cdot 10^{-3}$ and $\gamma_{end} = 5 \cdot 10^{-4}$. This schedule helps the models converge more smoothly by gradually decreasing the learning rate during training.

Original `Barlow Twins` model shows the least promising results, which can be described by complexity of given biological images, while BTA and BTA-combined each outperform state-of-the-art model in any accuracy metrics.

Metric	BTA	BTA-combined	Barlow Twins
Accuracy	**0.89**	0.87	0.68
Precision	0.87	**0.92**	0.54
Recall	**0.84**	0.71	0.43
F1-score	**0.85**	0.80	0.48
AUC ROC	0.94	**0.97**	0.69



BTA ROC curve

BTA-combined ROC curve

Barlow Twins ROC curve

Figure 1: Comparison of AUC ROC scores for every method

In this experiment, we set $\lambda_{BT} = 4 \cdot 10^{-3}$ for projector training loss function and $\lambda = 2 \cdot 10^{-5}$ for BTA-combined training. The chosen λ_{BT} value is similar to the value proposed in the [8] article, while impact of λ value we will explore in the following section.

5 Hyperparameter values

In this section, we conduct experiments to estimate the best hyperparameter values over accuracy metrics (accuracy and F1-score). While we set $\lambda_{BT} = 4 \cdot 10^{-3}$, which is nearly the same as in the main article's λ_{BT} value ([8]), we want to estimate λ for BTA-combined training.

To determine the best value for λ, we tested multiple values during training of the BTA-combined model, running each experiment for 100 epochs. If λ is set "too low", the projector loss barely affects training. Without this component, the model fails to learn useful features from the projector, leading to worse accuracy. However, if λ is "too high", the BCE loss (responsible for training the classifier) gets overshadowed. This imbalance prevents the classifier from learning to make reliable predictions, even if the projector works well. After testing different values, we found that $\lambda = 2 \cdot 10^{-5}$ strikes the right balance: it allows both the projector and classifier to contribute effectively during training. At this value, the model achieves its highest accuracy compared to other tested options, as shown in Figure 2.

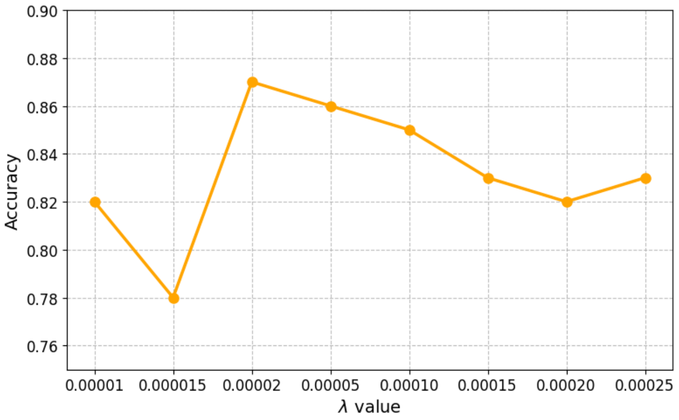


Figure 2: Model accuracy varying λ value

The F1-score curve (Figure 3) further validates the optimal choice of $\lambda = 2 \cdot 10^{-5}$, mirroring the accuracy trends. Unlike accuracy, which measures overall correctness, the F1-score balances precision and recall, making it particularly sensitive to binary classification imbalance since we set the probability of the pair of images being the same in accuracy validation to 0.4. As was mentioned before, the sharp decline in F1-score at higher λ values ($> 10^{-4}$) suggests that an overemphasis on the projector loss destabilizes the classifier's ability to reconcile false positives and negatives. Conversely, at lower λ ($< 2 \cdot 10^{-5}$), the F1-score somehow drops as the accuracy does indicating that residual projector regularization—even when minimal—still aids in extracting features that harmonize precision and recall. The symmetry between the accuracy and F1-score maxima underscores that $\lambda = 2 \cdot 10^{-5}$ maximizes overall performance on linear classification metrics.

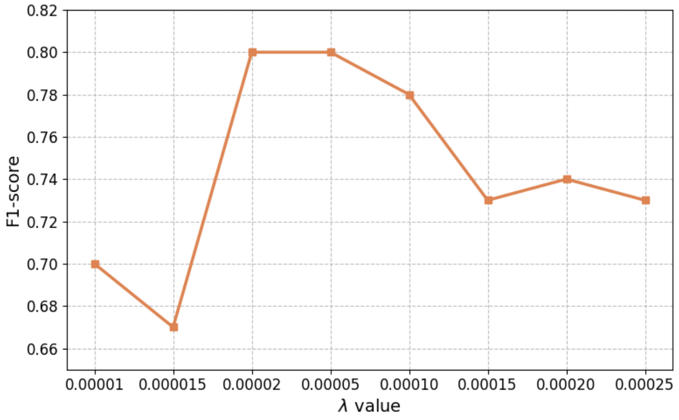


Figure 3: Model F1-score varying λ value

6 Conclusion

Our results show that our model significantly outperforms the state-of-the-art `Barlow Twins` framework [8] on pairwise image comparison. The main innovation is our new training method, which uses a combined loss to jointly train a parallel projector and linear classifier. This approach performs as well as the traditional two-stage training setup. We believe future work in pairwise image comparison, especially in complex visual tasks, can benefit from our unified strategy, potentially setting a new standard for handling visually ambiguous image pairs.

References

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

[2] Chen Xinlei and He Kaiming. Exploring simple siamese representation learning. In *arXiv preprint arXiv:2011.10566*, 2020.

[3] Chen Xinlei, Fan Haoqi, Girshick Ross, and He Kaiming. Improved baselines with momentum contrastive learning. In *arXiv preprint arXiv:2003.04297*, 2020.

[4] Chen Ting, Kornblith Simon, Norouzi Mohammad, and Hinton Geoffrey. A simple framework for contrastive learning of visual representations. In *arXiv preprint arXiv:2002.05709*, 2020.

[5] Grill Jean-Bastien, Strub Florian, Altché Florent, Tallec Corentin, H. Richemond Pierre, Buchatskaya Elena, Doersch Carl, Avila Pires Bernardo, Daniel Guo Zhaohan, Gheshlaghi Azar Mohammad, Piot Bilal, Kavukcuoglu Koray, Munos Rémi, and Valko Michal. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (PMLR)*, 2020.

[7] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021.

[8] 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 (ICML)*, 2021.

[9] Iaroslav Melekhov, Juho Kannala, and Esa Rahtu. Siamese network features for image matching. In *23rd International Conference on Pattern Recognition (ICPR)*, 2016.

[10] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. *arXiv preprint arXiv:physics/0004057*, 2000.

[11] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. *arXiv preprint arXiv:1503.02406*, 2015.

[12] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[13] Johannes L. Schönberger, Alexander C. Berg, and Jan-Michael Frahm. Paige: Pairwise image geometry encoding for improved efficiency in structure-from-motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[14] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[15] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

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

[17] Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. *arXiv preprint arXiv:1904.13132*, 2019.

[18] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A. Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 5050–5060, 2019.