Optical Wavelength Guided Self-Supervised Feature Learning For Galaxy Cluster Richness Estimate

Gongbo Liang[1,2] **Yuanyuan Su**[1] **Sheng-Chieh Lin**[1] **Yu Zhang**[1]
Yuanyuan Zhang[3] **Nathan Jacobs**[1]

[1] University of Kentucky, Lexington, KY, USA
[2] Eastern Kentucky University, Richmond, KY, USA
[3] Fermi National Accelerator Laboratory, Batavia, IL, USA

Project Page: www.gb-liang.com/owg

Abstract

Most galaxies in the nearby Universe are gravitationally bound to a cluster or group of galaxies. Their optical contents, such as optical richness, are crucial for understanding the co-evolution of galaxies and large-scale structures in modern astronomy and cosmology. The determination of optical richness can be challenging. We propose a self-supervised approach for estimating optical richness from multi-band optical images. The method uses the data properties of the multi-band optical images for pre-training, which enables learning feature representations from a large but unlabeled dataset. We apply the proposed method to the Sloan Digital Sky Survey. The result shows our estimate of optical richness lowers the mean absolute error and intrinsic scatter by 11.84% and 20.78%, respectively, while reducing the need for labeled training data by up to 60%. We believe the proposed method will benefit astronomy and cosmology, where a large number of unlabeled multi-band images are available, but acquiring image labels is costly.

1 Introduction

Most of the galaxies in the Universe, including our own galaxy, reside in clusters or groups of galaxies [16]. The optical richness of a galaxy cluster, λ, is a measure of the number of galaxies physically bounded to the system. It can be a tracer of the underlying dark matter halo that can be utilized to constrain dark energy parameters [3]. λ is also crucial to our understanding of the galaxy evolution and the growth of large structures in the Universe [9]. However, the measurement of λ is confronted by the presence of foreground and background objects, leading to the uncertainty of the location of a given galaxy along the line of sight. The membership of cluster galaxies can be determined with their spectroscopic redshifts. But spectroscopic surveys are expensive, and it is yet to be practical to perform spectroscopic follow-up for a large fraction of the sky [6].

Photometric surveys in multiple wavelength ranges, i.e. multi-band optical imaging, are much more achievable and have become an advanced imaging technique that is widely used for astronomical and cosmological studies [12]. Each multi-band optical image is a single channel image that is acquired using a specific optical wavelength band. The commonly used wavelength bands include *u*, *g*, *r*, *i*, and *z* (see Figure S1 for an example). The imaging modality is efficient for detecting galaxy clusters, but the richness of a galaxy cluster cannot be directly estimated.

Convolutional neural networks (CNNs) have been rapidly adopted in the astronomy and cosmology domains [11, 17, 20]. However, training a CNN typically requires a large number of labeled examples [10, 7, 15], limiting their applicability for many real-world problems [8, 19, 18]. To overcome this limitation, we propose a novel self-supervised training method for λ estimation. The

Third Workshop on Machine Learning and the Physical Sciences (NeurIPS 2020), Vancouver, Canada.

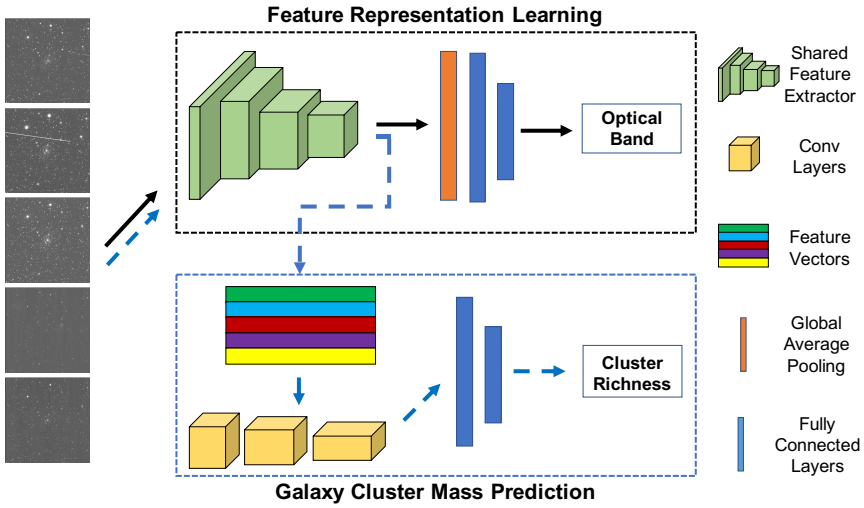


Figure 1: Two branches self-supervised learning architecture: 1) feature extractor via optical band classifications (solid black line); 2) galaxy cluster richness estimation (dashed blue line).

proposed method utilizes the data properties of the multi-band optical images and enables model training with a large but unlabeled dataset.

We believe rich information about λ is embedded in different optical bands. Thus, we exploit latent connections between optical bands and λ by learning the feature representations through optical wavelength band classifications. More specifically, we first learn image feature through band classification tasks. No manual annotations are required for the feature learning since wavelength band labels are known. Next, a downstream λ estimation network is built using the pre-trained feature extractor and trained on a small labeled dataset. We apply the proposed method to images taken from the Sloan Digital Sky Survey (SDSS) [2]. The result shows that our method significantly improves the λ estimate while reducing the need for labeled training data by up to 60%.

2 Method

The proposed method contains two branches (Figure 1): 1) an optical wavelength-guided feature learning branch and 2) a galaxy cluster richness estimation branch. A CNN feature extractor is shared between the two branches. The two branches can be optimized simultaneously or trained in separate phases. For simplicity, we present the two branches in a two-phase training setup.

2.1 Feature Learning

The feature learning branch is a classification network with multiple convolutional (Conv) layers followed by a global average pooling (GAP) layer and two fully connected (FC) layers (Figure 1 solid black line). The network treats each of the multi-band images independently and predicts the corresponding optical band label for each image. The Conv layers are used as a feature extractor, which learns meaningful features from the images. The FC layers output the logits for a softmax. Cross-entropy loss is used for the feature learning branch training. Since the band label is automatically assigned to each image when the image was acquired, no manually annotated label is needed when training the feature learning branch. Besides, by treating each optical image independently, the feature learning branch naturally increases the training set size by a factor of five.

2.2 Galaxy Cluster Richness Estimation

The richness estimation branch is another CNN network (Figure 1 dashed blue line), which shares the feature extractor with the feature learning branch. The network takes multi-band optical images as input and estimates the galaxy cluster richness (λ). Five multi-band optical images of the same

galaxy cluster are passed through the feature extractor. The five feature maps are then concatenated and passed through a sequence of Conv layers and FC layers. The Conv layers aim to 1) convert the band classification trained features to the regression task, and 2) learn meaningful representations across the five different bands. The FC layers are used to perform the regression prediction. Table S1 shows the pseudocode of training the richness estimation branch.

The feature extractor can be used fixed or jointly optimized during the richness estimation branch training. The mean squared error (MSE) loss with the intrinsic scatter between the ground-truth richness and predicted richness (a custom regularization term) is used in the training. The loss can be written as:

$$\text{loss} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2 + \alpha \text{std}(|Y - \hat{Y}|), \tag{1}$$

where the first term is the MSE loss and the second term is the intrinsic scatter between the ground-truth richness and predicted richness, in which α is a weight scalar, std$(\cdot)$ denotes the standard deviation, $|\cdot|$ denotes the absolute value, Y and $\hat{Y}$ are the sets of ground-truth and the predicted λ with the magnitude of n, y_i and $\hat{y_i}$ indicate the ground-truth and predicted λ for the i^{th} sample in one training batch.

2.3 Implementation

We implement this work in PyTorch [13]. The ResNet-18 [4] model is used as the backbone of the feature learning branch. A 1×1 Conv layer followed with BatchNorm [5] and a rectified linear unit (ReLU) is added before the GAP layer of the ResNet-18 model. An FC layer with 512 neurons is added after the GAP layer. The SGD optimizer with a learning rate of 10^{-4} and a momentum of 0.9 is used in training. Cross-entropy is used as the loss function in feature learning branch training.

The richness estimation branch contains the shared feature extractor, three Conv layers, and three FC layers. GAP is applied to each of the feature maps with an output shape of 512×1. The five multi-band optical images of a galaxy cluster are passed through the feature extractor. Then, the five feature maps are concatenated and passed through the rest of the network. Table S2 shows the detailed architecture of the richness estimation branch. The SGD optimizer with a learning rate of 5×10^{-6} and a momentum of 0.9 is used in training. Equation 1 is used as the loss function for the richness estimation branch training.

3 Experiments

3.1 Experimental Setup

We use the SDSS Release 12 dataset [1] in this study. SDSS is an optical wide-field imaging survey covering one-third of the sky ($\sim 14000\text{deg}^2$) and provides five broad-band data (u, g, r, i, z) with a typical depth of ~ 21 magnitude in the i-band. We select the images according to the cluster catalog generated by the red-sequence Matched-filter Probabilistic Percolation (redMaPPer) cluster finding algorithm [14]. We set the image size to be the physical size of 1 Mpc ($= 3.09 \times 10^{22}$m) around each cluster center. The labeled values of λ are taken from the results given by the redMaPPer algorithm.

In total, $24,596$ optical observations are used in this study. In the feature learning stage, we randomly partition the dataset into training and testing sets, with a $4:1$ ratio. We pre-train the feature learning branch for 100 epochs. In the galaxy cluster richness estimation training stage, we randomly partition the dataset into ten sets of roughly equal size. Ten-fold cross-validation is used to evaluate model performance. For both partitioning steps, we ensure that all images of a given galaxy cluster are in the same partition

3.2 Evaluation Method

We evaluate the proposed method based on the λ estimation and the degree of need for labeled instances for training. We compare the proposed method (denoted as *Ours*) against a baseline model (denoted as *Base*). Both *Ours* and *Base* have the same architecture, but the feature extractor of *Ours* is pre-trained via the optical band classification task. We train each model multiple times using different percentages of the training data (between 1% and 100%). The data are randomly selected from each

Table 1: Detailed Performance of *Base* and *Ours*

Metric	Model	Percentage of Training Data											
		1%	5%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
MAE	Base	2.3923	0.4792	0.2943	0.2397	0.2456	0.2072	0.1980	0.1921	0.1903	0.1856	0.1965	0.1832
	Ours	**0.4188**	**0.2566**	**0.2554**	**0.2204**	**0.2118**	**0.1895**	**0.1824**	**0.1813**	**0.1802**	**0.1714**	**0.1675**	**0.1615**
Sigma	Base	1.3304	0.5810	0.4549	0.3708	0.3118	0.3447	0.2819	0.3106	0.2973	0.2659	0.2585	0.2565
	Ours	**0.5166**	**0.3318**	**0.3117**	**0.2918**	**0.2724**	**0.2555**	**0.2312**	**0.2321**	**0.2308**	**0.2198**	**0.2205**	**0.2032**

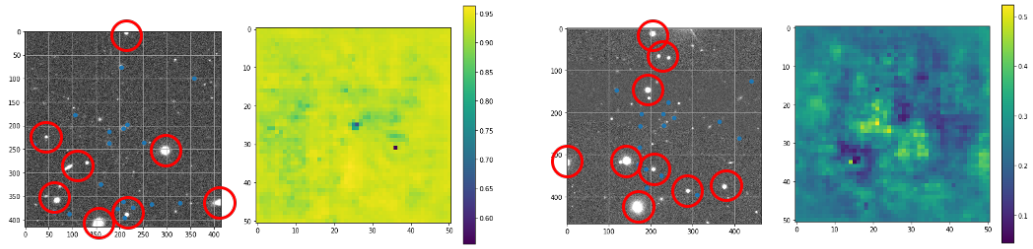


Figure 2: Two randomly selected occlusion testing results show that the optical band classification network may consider the member galaxies and foreground stars when making the decision. For each example, left: an optical image, right: occlusion map, blue dot: member galaxy, red circle: foreground star.

data fold. For the models that are trained with the same amount of training data, we ensure to use the same subsets in the training of both *Ours* and *Base*. We use the mean absolute error (MAE) and the intrinsic scatter between the ground-truth richness and predicted richness (Sigma) as the evaluation metrics. For both metrics, a smaller value indicates a better performance.

3.3 Richness Estimate

Table 1 and Figure S2 show the results of *Ours* and *Base* using different amounts of training data. Each model was run three times. The mean value of the three trials are shown in Table 1. The mean value with the standard deviation are shown in Figure S2. The result reveals that our approach is superior to the *Base* model in all setting, with better gains when only a few labeled images are available. For instance, when using 1% of the labeled data ($\approx$ 245 instances), the *Base* model has an MAE of 2.3923 and a Sigma of 1.3304, while the *Ours* has 0.4188 and 0.5166, respective. The proposed method reduces the MAE by 82.49% and reduces the Sigma by 61.70%.

The best performance of *Base* is 0.1832 MAE and 0.2565 Sigma, when it is trained with 100% of the labeled data. *Ours* surpasses the best performance on MAE of *Base* with only 50% of training data (0.1824 MAE) and the best performance on Sigma of *Base* with only 40% of training data (0.2555 Sigma). Thus, the need for manual labels is reduced by 50% or 60% when using the proposed method. One particular observation is that for every fraction of the full training dataset considered, the proposed method is superior across all measures. The best performance of *Ours* is 0.1659 MAE and 0.2080 Sigma, which are 11.84% and 20.78% improvement compared with *Base*.

3.4 Pre-Trained Feature

CNN feature extractors encode images into a high dimensional space. For instance, the ResNet-18 feature extractor used in this work encodes an input image of $224 \times 224 \times 3$ pixels to a $7 \times 7 \times 512$ feature space [4]. Evaluation or analysis of the learned features is usually non-trivial due to the high dimensionality. We evaluate the feature extractor by applying an occlusion test on the pre-trained optical band classification network (solid black line in Figure 1).

Figure 2 shows the occlusion testing results for two randomly selected samples. For each example, the input optical image is displayed on the left, and the occlusion map is displayed on the right. Blue dots indicate the member galaxies of the cluster taken from [14]. Red circles indicate foreground

stars. To conduct this experiment, we use a small patch to occlude part of the input image and use the occluded image to test the model. We repeat this process by occluding every possible location of the input image. The pixel value of the occlusion map is the probability of being the correct prediction when the corresponding part is occluded. The brighter color indicates higher probability and darker color indicates lower probability. For instance, in Figure 2 Left, the occlusion map shows that when occluding the center of the image, the predicted probability changed dramatically, indicating cluster central regions may be more important than other areas to the decision-making process.

By comparing the occlusion map and the input image, we noticed that a significant performance change often happens when occluding foreground stars or member galaxies. This systematic phenomenon may indicate the foreground stars and member galaxies are critical to the classification decision. From this end, we believe the pre-trained features may represent both foreground stars and member galaxies well.

Conceptually speaking, galaxy richness estimation is a process of separating member galaxies from other objects in the same image. The ability to represent member galaxies and foreground stars is an important criterion that leads to the success of this task. Thus, we believe the feature learned by the proposed method is highly relevant to richness estimations.

4 Concluding Remarks

We proposed a novel self-supervised learning method for galaxy cluster richness estimation using multi-band optical images. The method utilizes the data properties of the multi-band optical images for pre-training and enables learning feature representations using a large but unlabeled dataset. We believe the proposed self-supervised feature learning method is not limited to galaxy cluster richness estimation. It is potentially useful for any task where multi-band images are available but acquiring manual labels is expensive and time-consuming.

Broader Impact

A robust deep neural network usually requires a large labeled data set for training, which often does not exist in the real world. The astronomy and cosmology domains are not exceptions. Limited labeled data prevents the adoption of the latest advanced neural network techniques in the domains. We consider our contributions to this work as the following:

- A novel self-supervised approach on galaxy cluster richness estimation, which improves the galaxy cluster richness estimate while reducing the need for labeled training data.
- The concept of using color bands as a guidance for pre-training is not limited to the astronomy and cosmology domains. It also works in the natural imaging domain (see Section S1).
- To our best knowledge, this is the first work that applies a self-supervised training strategy in galaxy cluster richness estimation.

Acknowledgments and Disclosure of Funding

This work was sponsored by Grant No. IIS-1553116 from the U.S. National Science Foundation.

Funding for SDSS-III has been provided by the Alfred P. Sloan Foundation, the Participating Institutions, the National Science Foundation, and the U.S. Department of Energy Office of Science. The SDSS-III web site is http://www.sdss3.org/. SDSS-III is managed by the Astrophysical Research Consortium for the Participating Institutions of the SDSS-III Collaboration including the University of Arizona, the Brazilian Participation Group, Brookhaven National Laboratory, Carnegie Mellon University, University of Florida, the French Participation Group, the German Participation Group, Harvard University, the Instituto de Astrofisica de Canarias, the Michigan State/Notre Dame/JINA Participation Group, Johns Hopkins University, Lawrence Berkeley National Laboratory, Max Planck Institute for Astrophysics, Max Planck Institute for Extraterrestrial Physics, New Mexico State University, New York University, Ohio State University, Pennsylvania State University, University of Portsmouth, Princeton University, the Spanish Participation Group, University of Tokyo, University of Utah, Vanderbilt University, University of Virginia, University of Washington, and Yale University.

References

[1] Shadab Alam, Franco D Albareti, Carlos Allende Prieto, Friedrich Anders, Scott F Anderson, Timothy Anderton, Brett H Andrews, Eric Armengaud, Éric Aubourg, Stephen Bailey, et al. The eleventh and twelfth data releases of the sloan digital sky survey: final data from sdss-iii. *The Astrophysical Journal Supplement Series*, 219(1):12, 2015.

[2] Michael R Blanton et al. Sloan digital sky survey iv: Mapping the milky way, nearby galaxies, and the distant universe. *The Astronomical Journal*, 154(1):28, 2017.

[3] Zoltan Haiman, Joseph J Mohr, and Gilbert P Holder. Constraints on cosmological parameters from future galaxy cluster surveys. *The Astrophysical Journal*, 553(2):545, 2001.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016.

[5] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proc. ICML*, 2015.

[6] Michael E Levi et al. The dark energy spectroscopic instrument (desi). *arXiv:1907.10688*, 2019.

[7] Gongbo Liang, Yu Zhang, Xiaoqin Wang, and Nathan Jacobs. Improved trainable calibration method for neural networks on medical imaging classification. In *British Machine Vision Conference*, 2020.

[8] Geert Litjens et al. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, 2017.

[9] Heliana Estefanía Luparello, M Lares, D Paz, Claudia Yamila Yaryura, DG Lambas, and Nelson Padilla. Brightest group galaxies and the large-scale environment. *Monthly Notices of the Royal Astronomical Society*, 448(2):1483–1493, 2015.

[10] Radu Paul Mihail, Gongbo Liang, and Nathan Jacobs. Automatic hand skeletal shape estimation from radiographs. *IEEE transactions on nanobioscience*, 18(3):296–305, 2019.

[11] Michelle Ntampaka et al. A deep learning approach to galaxy cluster x-ray masses. *The Astrophysical Journal*, 876(1):82, 2019.

[12] Kristina Nyland et al. An application of multi-band forced photometry to one square degree of servs: Accurate photometric redshifts and implications for future science. *The Astrophysical Journal Supplement Series*, 230(1):9, 2017.

[13] Adam Paszkeand et al. Pytorch: An imperative style, high-performance deep learning library. In *Proc. NeurISP.* 2019.

[14] ES Rykoff et al. redmapper. i. algorithm and sdss dr8 catalog. *The Astrophysical Journal*, 785(2):104, 2014.

[15] Tawfiq Salem, Scott Workman, and Nathan Jacobs. Learning a dynamic map of visual appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[16] Volker Springel and Lars Hernquist. The history of star formation in a λ cold dark matter universe. *Monthly Notices of the Royal Astronomical Society*, 339(2):312–334, 2003.

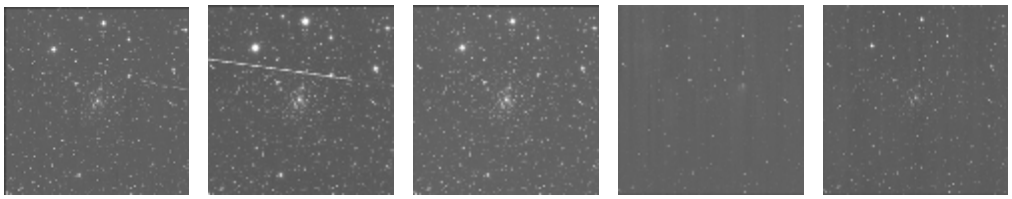
[17] Yuanuan Su et al. A deep learning view of the census of galaxy clusters in illustristng. *Monthly Notices of the Royal Astronomical Society*, 2020.

[18] Xiaoqin Wang et al. Inconsistent performance of deep learning models on mammogram classification. *Journal of the American College of Radiology*, 17(6):796–803, 2020.

[19] Ying Yu, Min Li, Liangliang Liu, Yaohang Li, and Jianxin Wang. Clinical big data and deep learning: Applications, challenges, and future outlooks. *Big Data Mining and Analytics*, 2(4):288–305, 2019.

[20] Yu Zhang, Gongbo Liang, Yuanyuan Su, and Nathan Jacobs. Multi-branch attention networks for classifying galaxy clusters. In *Proceedings of International Conference on Pattern Recognition*, 2021.

Supplementary Materials



(a) *g*-band (b) *i*-band (c) *r*-band (d) *u*-band (e) *z*-band

Figure S1: A galaxy cluster shows in multi-band images.

Table S2: Detailed Architecture for the Galaxy Cluster Richness Estimate Branch

Layer	**Kernel Shape**	**Out Shape**
Shared Feature Extractor	–	$512 \times 5 \times 1$
Conv1	1×1	$512 \times 5 \times 1$
BatchNorm	–	–
ReLU	–	–
Conv2	3×3	$510 \times 5 \times 16$
BatchNorm	–	–
ReLU	–	–
Conv3	3×3	$508 \times 5 \times 64$
BatchNorm	–	–
ReLU	–	–
FC1	–	1024
FC2	–	512
FC3	–	1

Table S1: Pseudocode for Galaxy Cluster Richness Estimation Branch Training

Pseudocode

```
// let CNN_ft be the shared feature extractork
// let CNN_reg be the regression network

outputs ← []        // an empty array
for every galaxy cluster C{
    // let [I_h, I_i, I_r, I_u, I_z] be multi-band images of C
    ft ← []         // an empty array
    for(i=0; i<5, i++){
        ft[i].concatenate(CNN_ft(C[i]))
    }
    outputs.append(CNN_res(ft))
}

loss = loss_function(outputs, ground_truth)
```

S1 Performance on Natural Imaging Data

We believe that the proposed method is not only limited to the astronomy and cosmology domains. We show the evaluation result of the proposed method in the natural domain using the CIFAR-10 and UCB200 datasets in this section. The self-training strategy in this section is slightly different from

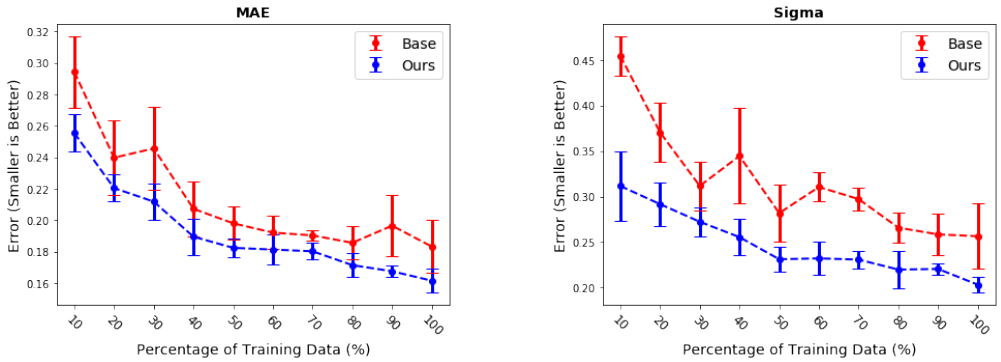


Figure S2: The performance (mean and standard deviation) of Base and Ours using 10% to 100% of training data. Left: The mean absolute error (MAE). Right: The intrinsic scatter between the ground-truth richness and predicted richness(Sigma).

the one for galaxy cluster richness estimation, but they all follow the same principle of using color band/channel information for pre-training.

S1.1 Pre-Trian Feature Extractor via Channel-Ordering

In the natural imaging domain, images usually have three color channels: *R*-channel, *G*-channel, and *B*-channel. For the pre-training task, instead of predicting the channel label of a single channel that is described in Section 2.1, we want to predict the color channel orders of a given input. More specifically, we randomly shuffle the color channels of an input image during the training time, such as from RGB to GBR. Then, we let the network predict the channel ordering of the given image. After the feature extractor is trained through the channel-ordering task, we can fine-tune it on the downstream classification tasks.

S1.2 Classification Result

Figure S3 shows the comparing result of the proposed method (*Ours*) and the baseline model (*Base*) on the CIFAR-10 and UBC-200 datasets. Both *Base* and *Ours* have the same ResNet-18 architecture. The only difference is that the feature extractor of *Ours* is pre-trained on the channel-ordering task. The figure reveals that the *Ours* improves the classification performance on CIFAR-10 and UCB200 by 5.38% and 10.43%, respectively. The *Base* achieved its best performance on both datasets uses 100% of the training data. The *Ours* can achieve a similar performance using only 60% or 80% of the training data, respectively.

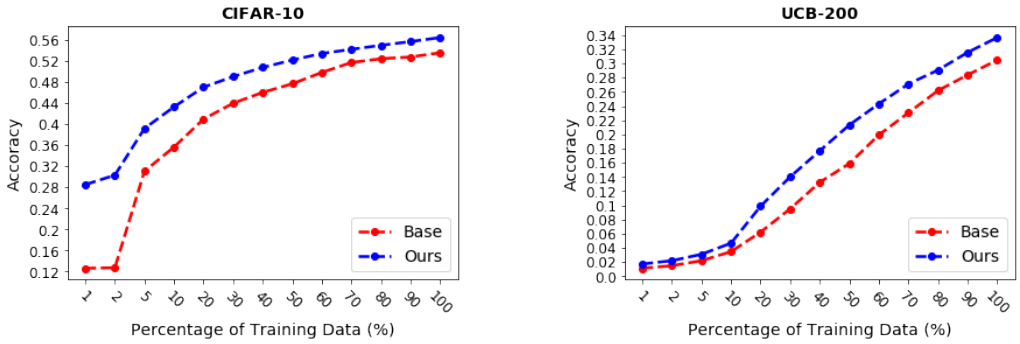


Figure S3: Channel-ordering pre-trained model performance on CIFAR10 and UCB200.