Knowledge Distillation with Relative Representations for Image Representation Learning

Patrick Ramos[1 [0009−0000−2035−8729]], Raphael Alampay[1 [0000−0001−7498−8830]], and Patricia Abu[1 [0000−0002−8848−6644]]

ALIVE, Ateneo de Manila University, Quezon City, Philippines
patrick.john.ramos@obf.ateneo.edu

Abstract. Relative representations allow the alignment of latent spaces which embed data in extrinsically different manners but with similar relative distances between data points. This ability to compare different latent spaces for the same input lends itself to knowledge distillation techniques. We explore the applicability of relative representations to knowledge distillation by training a student model such that the relative representations of its outputs match the relative representations of the outputs of a teacher model. We test our Relative Representation Knowledge Distillation (RRKD) scheme on supervised and self-supervised image representation learning with MNIST and show that an encoder can be compressed to 47.71% of its original size while maintaining 91.92% of its full performance. We demonstrate that RRKD is competitive with or outperforms other relation-based distillation schemes in traditional distillation setups (CIFAR-10, CIFAR-100, SVHN) and in a transfer learning setting (Stanford Cars, Oxford-IIIT Pets, Oxford Flowers-102). Our results indicate that relative representations are an effective signal for knowledge distillation. Code will be made available at https://github.com/Ramos-Ramos/rrkd.

Keywords: Knowledge distillation · Relative representations · Latent space.

1 Introduction

Although the latent spaces learned by neural networks are expected to be solely reliant on their data and optimization constraints, stochastic factors such as random weight initialization cause models with similar constraints to learn different latent spaces. However, these spaces are actually intrinsically similar and differ by a quasi-isometric transformation [15]. Relative representations [12] leverage this phenomenon and re-express latents of the same input but in different spaces as vectors of similarities to other "anchor" latents in their respective vector spaces. This method of representing latents is invariant to isometry and allows the alignment of latent spaces for zero-shot communication between them.

The idea of having separate models trained on the same task with the same data draws similarities with knowledge distillation [6], where a smaller student model must generalize to data in a manner similar to a larger teacher one. This is usually done by using one or more outputs of the teacher model (e.g. final class logits, intermediate features) as a supervisory signal for training the student.

There are two factors driving the intuition that relative representations can be used for knowledge distillation. Firstly, [12] report that in a collection of models optimized with the same objective across different hyperparameters, model performance is correlated with latent space similarity to a reference gold model. Secondly, computing relative representations is fully differentiable, allowing their similarity to be used as a knowledge distillation objective.

We conduct a preliminary but extensive investigation exploring the applicability of relative representations to knowledge distillation in an image representation learning setting. Our Relative Representation Knowledge Distillation (RRKD) method consists of converting student and teacher outputs to relative representations and optimizing a matching objective between them as opposed to the original representations. Our contributions are as follows:

- We design a knowledge distillation scheme, called RRKD, centered around matching the relative representations of a student to those of a teacher.
- In both supervised and self-supervised image representation learning experiments with MNIST, we demonstrate that relative representations are capable of distilling knowledge in a teacher-student framework.
- We show that RRKD can outperform similar relation-based distillation methods across a variety of benchmarks, with results extending to transfer learning.
- Through an ablation study, we show that online selection of anchors by using in-batch references is an effective anchor selection strategy.

2 Related Work

Knowledge distillation [6] is a model compression method that trains a small student network to mimic the behavior of a large reference teacher model. Rather than rely solely on supervision from the training data, students learn from targets created by the teacher. The classical form of knowledge distillation is seen in the classification setting, where the student's logits are are matched to "soft target" logits generated by the teacher alongside the ground truth hard labels from the training data.

FitNets [20] improve on classical distillation by introducing feature representations from intermediate layers as targets. When using intermediate representations during distillation, the outputs of "guided" layers in the student are matched to those of "hint" layers in the teacher. Intermediate feature-based distillation methods may also distill transformations of the intermediate features such as attention maps [24] or neuron selectivities [7] computed from CNN feature maps.

While knowledge distillation methods typically distill the final or intermediate encodings of inputs, there exist several relation-based distillation methods that instead distill the relationships between data points, like our proposed method. Such methods distill the relationships between triplets or pairs of data. Prior works have formulated these relationships as distances in embedding space [23, 16], instance relation graphs [11], joint probability distributions [18], and correlations [19]. Most similar to our work is similarity-preserving knowledge distillation [22], which distills the l_2-normalized outer product of a batch of model activations with itself. We also use the outer product of model activations as a distillation signal, but perform l_2-normalization before the outer product to create a cosine similarity map of instances within a batch, as discussed in Section 4.

3 Background: Relative Representations

Given training data X and an encoder $\phi : X \to \mathbb{R}^d$, the standard d-dimensional embedding $\phi(x) \in \mathbb{R}^d$ for $x \in X$ is referred to as the absolute representation of x. Relative representations instead express x in terms of its similarity to, or *relative* to, other points in X. Specifically, x is represented as an m-dimensional vector of similarities of its absolute representation to the absolute representations of a set of m anchor points.

To create a relative representation, we select a set of m anchor points from the training data, denoted as $A \subseteq X$. The relative representation of $x \in X$ is then computed as

$$r(x) = \langle s(\phi(x), \phi(A_0)), \ldots, s(\phi(x), \phi(A_{m-1}))\rangle \in \mathbb{R}^m, \tag{1}$$

where $s(\cdot, \cdot)$ is a similarity function producing a scalar similarity score for two latents. This effectively projects x to an m-dimensional space where each dimension is its similarity to one of the m anchors.

4 Relative Representation Distillation

Fig. 1 provides an overview of RRKD. Given a batch of n inputs, we extract teacher and student absolute representations $Z_t, Z_s \in \mathbb{R}^{n \times d}$ using a frozen teacher encoder ϕ_t and a learnable student encoder ϕ_s, respectively. The teacher and student absolute representations are then converted to relative representations $V_t, V_s \in \mathbb{R}^{n \times n}$. Computing relative representations for zero-shot communication between latent spaces typically involves choosing a set of anchors beforehand to use in computing all relative representations. However, instead of selecting a fixed set of anchor points, we use all embeddings in the batch as anchors akin to the intra-batch comparisons performed in contrastive learning [3]. This anchor selection method is compared to other strategies in an ablation study in Section 5.4. We follow [12] and choose cosine similarity as the similarity function s, resulting in V_t and V_s being $n \times n$ cosine similarity maps of

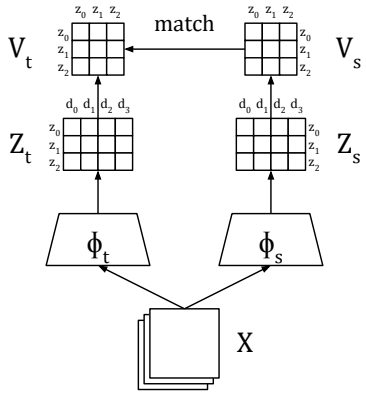

Fig. 1. Relative representation knowledge distillation overview. Teacher ϕ_t and student ϕ_s respectively encode inputs X into latents Z_t and Z_s. The latents are converted to relative representations V_t and V_s using themselves as anchors. A cosine similarity-based loss is used to encourage V_s to match V_t. ϕ_t is frozen.

embeddings to each other. The relative representations can be formulated as

$$V_m = \widetilde{Z_m}\widetilde{Z_m}^{\intercal}, \tag{2}$$

where $\widetilde{Z_m}$ is a row-wise l_2-normalized Z_m and $m \in \{t, s\}$ refers to the teacher (t) and student (s) models. This computes the cosine similarity map by taking the outer product of the l_2-normalized absolute representations.

The distillation loss $\mathcal{L}_D$ then encourages the student's relative representations to match those of the teacher's by taking the negative average logarithm of their pairwise cosine similarities, which are rescaled between 0 and 1. We add a small constant $\epsilon = 1\text{e}{-8}$ to the rescaled cosine similarity before the logarithm for numerical stability. The use of cosine similarity is based on experiments by [12] using the similarity of a network's relative representations with those of a reference model as a performance proxy.

$$\mathcal{L}_D = -\frac{1}{n}\sum_{i=0}^{n-1}\log\left(\frac{V_{t,i}\cdot V_{s,i}+1}{2\|V_{t,i}\|_2\|V_{s,i}\|_2}+\epsilon\right) \tag{3}$$

$\mathcal{L}_D$ assumes that the networks ϕ_t and ϕ_s are encoders that project to a latent space, such as models trained with self-supervised learning, rather than supervised classifiers that produce class logits. While the logit space can be considered a latent space where the dimensions correspond to each class, permutation symmetries in such a space may result in student relative representations that are similar to a teacher's but assign the highest scores to the wrong class. Therefore it is not recommended to use the logits as the latent space for computing relative representations.

Nevertheless, it is still possible to distill a teacher classifier into a student encoder using relative representations. Simply removing the teacher's classification head results in an encoder. The distillation then happens in the latent space prior to the projection to class logits. Furthermore, class labels can be used in distillation if available. When distilling a model with labels Y, we add a classification head on top of ϕ_s which projects Z_s into class label predictions P and add the classical cross entropy classification loss $\mathcal{L}_{CE}$ to the total loss. The loss for distilling models with labels is then a linear combination of the distillation and cross entropy losses:

$$\mathcal{L}_L = \lambda_D \mathcal{L}_D + \lambda_{CE} \mathcal{L}_{CE}(P, Y), \tag{4}$$

where λ_D and λ_{CE} are weights for the distillation and cross entropy losses respectively. In practice, we set $\lambda_D = 1$ and use $\lambda_{CE} = 1$ when distilling with class labels and $\lambda_{CE} = 0$ when distilling without. Setting $\lambda_{CE} = 0$ makes the loss function equivalent to Equation 3.

Note that the use of relative representations always results in latents of the same size between the teacher and student, which frees us from the usual restriction in knowledge distillation of needing the same dimensionality for teacher and student representations.[1] This means not only can student models be "shorter" (less layers) than the teacher, but they can also be "thinner" (smaller dimension size).

5 Experiments

We evaluate RRKD in an image representation learning setting. We perform distillation experiments for image encoders trained with supervision and self-supervision, compare RRKD to other relation-based distillation schemes, and perform an ablation study on the method of selecting anchors. Note that in all experiments, the aim is not to maximize raw performance on the target dataset but to maximize the preservation of a teacher's performance.

5.1 Distillation Compared to Training from Scratch

We distill both self-supervised and supervised MLP encoders trained on MNIST using RRKD. We evaluate all models with linear probe evaluation using a logistic regression classifier. This includes models distilled with class labels, as our primary goal is to use RRKD to train student encoders rather than classifiers, and the standard protocol for evaluating encoder representations is linear probe evaluation [3, 1]. Furthermore, we find it more effective to discard the classification head after distillation and attach a new linear head.

[1] Some works try to address this by projecting the student outputs with a learnable linear layer to have the same dimensionality as the teacher [20, 8]. Our work is similar in that regard as computing the relative representations can also be considered a linear projection, but our method does not require learning the projection weights.

Table 1. Distillation results for self-supervised MNIST models.

Model	Parameter Count	Linear
Teacher (AE-64)	109K	87.99
Baseline (AE-32)	52K	74.14
Student (AE-32)	52K	80.88

Table 2. Distillation results for supervised MNIST models.

Model	Parameter Count	Linear
Teacher (MLP-1200)	2M	95.64
Baseline (MLP-32)	26K	90.87
Student (MLP-32)	26K	92.94

To investigate the value of distillation compared to simply training the smaller model from scratch, students are compared to baselines which follow the same architecture but are trained in the same manner as the teacher. Following a hyperparameter sweep across learning rates $\{1e{-}1, 1e{-}2, 1e{-}3\}$ (extended until $1e{-}8$ for the self-supervised baseline), we train all models for 20 epochs using SGD with learning rate $1e{-}1$ and batch size 128, except for the self-supervised baseline, which was trained with learning rate $1e{-}8$. The models in the self-supervised MNIST encoder distillation experiment use 0.9 momentum to accelerate optimization.

Self-supervised MNIST We train an MLP auto-encoder with hidden layer sizes {128, 64, 128} on MNIST with an MSE pixel reconstruction loss. We use the encoder of the auto-encoder as the teacher and distill it into a student MLP with layer sizes {64, 32}. This student can be viewed as the encoder of an auto-encoder with hidden layer sizes {64, 32, 64}, which serves as the baseline. All models use ReLU activation and 0.5 dropout. Distillation is performed using the standard relative representation loss described in Equation 3.

We report linear probe evaluation accuracies averaged across three trials and model parameter counts in Table 1. With only 47.71% of the number of parameters of the teacher, the student is able to retain 91.92% of the teacher's performance. The student also achieves a higher accuracy than a baseline trained from scratch, outperforming it by 6.74%.

Supervised MNIST We train an 3-layer MLP with hidden layer sizes {1200, 1200} on MNIST classification. Stripping the classification layer leaves a 2-layer MLP teacher encoder projecting to a 1200-dimensional latent space. The student is a 2-layer MLP that has one hidden layer of size 32 and projects to 32 dimensions. ReLU activation and dropout of 0.5 are also used. Relative representation distillation is performed using class labels with the loss described in Equation 4.

The average accuracies over three trials and model sizes are presented in Table 2. The student preserves 97.18% of the teacher's accuracy while being

Table 3. Comparison to other relation-based distillation methods. Best non-teacher results are indicated in **bold**.

Method/Model	Parameter Count	CIFAR-10 [10]	CIFAR-100 [10]	SVHN [13]
Teacher	11M	88.13	60.72	94.66
SPKD [22]	1M	85.85	56.35	**94.56**
LPKD [2]	1M	85.56	57.04	94.49
RRKD (ours)	1M	**86.21**	**57.48**	94.55

merely 1.3% of its size. Furthermore, the student outperforms the baseline by 2.07%.

These MNIST experiments demonstrate that transferring teacher knowledge with RRKD is capable of producing compressed encoders that are more performant than ones that were trained from scratch.

5.2 Comparison to Other Relation-based Distillation Methods

We compare our relative representation distillation scheme to SPKD [22] and LPKD [2], two other relation-based distillation methods. SPKD distills a similarity map computed by the row-wise l_2-normalized outer product of latents from a teacher hint layer to that of a student guided layer, and can be combined with a cross entropy loss with class labels. In our experiments, we use $\gamma = 1$ and follow the SPKD authors by only using the last hidden layers as hint and guided layers. Meanwhile, LPKD distills a squared error distance map and was proposed to be combined with a cross entropy loss with class labels and a soft cross entropy loss with teacher logits. We use $\gamma = 1.5$, $\lambda = 2$, and $\tau = 0.5$. For both methods, please refer to [22] and [2] for an explanation of these hyperparameters.

Across CIFAR-10 [10], CIFAR-100 [10], and SVHN [13], we train a supervised ResNet-18[5] teacher with a final hidden layer of 512. The teacher is then distilled using the distillation losses and class labels into a ResNet-9 encoder with a 256 embedding size, which we refer to as ResNet-9×0.5. Images are kept at their original 32×32 resolution and are augmented only with random horizontal flipping. We train and distill for 50 epochs on CIFAR-10 and CIFAR-100 and for 20 epochs on SVHN. Models are optimized with SGD with learning rate 1e−1 (determined after a hyperparameter sweep across learning rates $\{1e{-}1, 1e{-}2, 1e{-}3\}$) and batch size 128. Momentum of 0.9 is used during distillation to be consistent with the use of momentum across benchmark distillation methods. These hyperparemters are also used in subsequent experiments.

Test accuracies on each dataset is reported in Table 3. RRKD outperforms baseline methods on CIFAR-10 (+0.36% from next best method) and CIFAR-100 (+0.44%), and has only a small deficit (−0.01%) compared to the best performing method on SVHN, all while being 11.05% the size of the teacher on average. These results indicate that RRKD can perform on par with other relation-based distillation methods if not better.

Table 4. Comparison to other relation-based distillation methods on transfer learning with distillation.

Method/Model	Parameter Count	Stanford Cars [9]	Oxford-IIIT Pets [17]	Oxford Flowers-102 [14]
Teacher	21M	78.60	90.52	85.76
SPKD [22]	11M	72.63	77.19	80.50
LPKD [2]	11M	70.47	73.81	77.27
RRKD (ours)	11M	**76.79**	**81.22**	**84.62**

Table 5. Anchor selection ablation study results.

Anchor Selection Method	Anchors	MNIST Test Accuracy
In-batch	128	**81.42**
Random	128	80.21
Per class	130	80.59
Best per class	130	80.52

5.3 Transfer Learning

We also conduct distillation experiments in the transfer learning setting, where a pretrained student is fine-tuned while being distilled from a pretrained teacher already fine-tuned on the target dataset. Teachers are ResNet-34s pretrained on ImageNet [4] and fine-tuned on Stanford Cars [9], Oxford-IIIT Pets [17], and Oxford Flowers-102 [14]. Students are ImageNet-pretrained ResNet-18s. Images are augmented with Inception-style random cropping [21] to 224×224, horizontal flipping, and color jitter. We train and distill models for 20 epochs on Stanford Cars and Oxford-IIIT Pets and for 50 epochs on Oxford Flowers-102.

We report the accuracies of the distilled models in Table 4. RRKD outperforms other methods across all datasets. Averaged across the three datasets, the distilled student is capable of retaining 95.36% of the teacher's original accuracy with 52.64% the parameter count of a full teacher, showing that distilling relative representations can also be extended to transfer learning.

5.4 Anchor Selection

We perform an ablation study on the anchor selection method. We evaluate the linear probe evaluation of encoders distilled from the self-supervised MNIST MLPs described in Section 5.1 using four different anchor selection strategies:

- **In-batch.** This is the main method used in this study, described in Section 4. We use all embeddings in the batch as anchors, with a batch size of 128.
- **Random.** We select a fixed set 128 of random embeddings to be used as anchors throughout training.
- **Per class.** We randomly sample 13 examples per class to create a fixed set of 130 anchors. We choose 13 examples in order to have the smallest number of anchors greater than or equal to the number of in-batch anchors while giving each class equal representation.

– **Best per class.** This is similar to the previous method of choosing 13 examples per class but we instead choose the 13 instances per class that the teacher model is most confident in. For each class, we identify the 13 correctly classified examples with the largest probability for that particular class according to the teacher.

We show the linear probe evaluation results averaged across three trials in Table 5. Using in-batch anchors provides the highest linear probe evaluation, even when other anchor selection methods (per class, best per class) have more anchors. We hypothesize this may be because in-batch anchors expose the student to an larger number of reference points during training as using in-batch anchors means that the anchors change at each step and that all data points in the dataset are treated as anchors at some point.

6 Conclusion

We devise a knowledge distillation method based on relative representations, called RRKD. In image representation learning, student image encoders trained with RRKD are more performant than similar models trained without distillation. On a variety of image classification datasets, students trained with RRKD are competitive with if not better than other relation-based distillation methods.

Future works can further this study by exploring the scoring function used in generating relative representations alongside the loss function for comparing these representations. These experiments can also be extended to other architectures such as Transformer-based models and to other domains such as language and graph representation learning.

References

1. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021)
2. Chen, H., Wang, Y., Xu, C., Xu, C., Tao, D.: Learning student networks via feature embedding. IEEE Transactions on Neural Networks and Learning Systems **32**(1), 25–35 (2020)
3. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
4. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
6. Hinton, G., Vinyals, O., Dean, J., et al.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 **2**(7) (2015)

7. Huang, Z., Wang, N.: Like what you like: Knowledge distill via neuron selectivity transfer. arXiv preprint arXiv:1707.01219 (2017)
8. Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., Liu, Q.: Tinybert: Distilling bert for natural language understanding. arXiv preprint arXiv:1909.10351 (2019)
9. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: Proceedings of the IEEE international conference on computer vision workshops. pp. 554–561 (2013)
10. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
11. Liu, Y., Cao, J., Li, B., Yuan, C., Hu, W., Li, Y., Duan, Y.: Knowledge distillation via instance relationship graph. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7096–7104 (2019)
12. Moschella, L., Maiorca, V., Fumero, M., Norelli, A., Locatello, F., Rodolà, E.: Relative representations enable zero-shot latent space communication. arXiv preprint arXiv:2209.15430 (2022)
13. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning (2011)
14. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing. pp. 722–729. IEEE (2008)
15. Olah, C.: Visualizing representations: Deep learning and human beings (Jan 2015), `http://colah.github.io/posts/2015-01-Visualizing-Representations/`
16. Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3967–3976 (2019)
17. Parkhi, O.M., Vedaldi, A., Zisserman, A., Jawahar, C.: Cats and dogs. In: 2012 IEEE conference on computer vision and pattern recognition. pp. 3498–3505. IEEE (2012)
18. Passalis, N., Tefas, A.: Learning deep representations with probabilistic knowledge transfer. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 268–284 (2018)
19. Peng, B., Jin, X., Liu, J., Li, D., Wu, Y., Liu, Y., Zhou, S., Zhang, Z.: Correlation congruence for knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5007–5016 (2019)
20. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550 (2014)
21. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9 (2015)
22. Tung, F., Mori, G.: Similarity-preserving knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1365–1374 (2019)
23. You, S., Xu, C., Xu, C., Tao, D.: Learning from multiple teacher networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1285–1294 (2017)
24. Zagoruyko, S., Komodakis, N.: Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. arXiv preprint arXiv:1612.03928 (2016)