In this paper, we propose the $K$-Shot Contrastive Learning (KSCL) of visual features by applying multiple augmentations to investigate the sample variations within individual instances. It aims to combine the advantages of inter-instance discrimination by learning discriminative features to distinguish between different instances, as well as intra-instance variations by matching queries against the variants of augmented samples over instances. Particularly, for each instance, it constructs an instance subspace to model the configuration of how the significant factors of variations in $K$-shot augmentations can be combined to form the variants of augmentations. Given a query, the most relevant variant of instances is then retrieved by projecting the query onto their subspaces to predict the positive instance class. This generalizes the existing contrastive learning that can be viewed as a special one-shot case. An eigenvalue decomposition is performed to configure instance subspaces, and the embedding network can be trained end-to-end through the differentiable subspace configuration. Experiment results demonstrate the proposed $K$-shot contrastive learning achieves superior performances to the state-of-the-art unsupervised methods.

**Abstract**—In this paper, we propose the $K$-Shot Contrastive Learning (KSCL) of visual features by applying multiple augmentations to investigate the sample variations within individual instances. It aims to combine the advantages of inter-instance discrimination by learning discriminative features to distinguish between different instances, as well as intra-instance variations by matching queries against the variants of augmented samples over instances. Particularly, for each instance, it constructs an instance subspace to model the configuration of how the significant factors of variations in $K$-shot augmentations can be combined to form the variants of augmentations. Given a query, the most relevant variant of instances is then retrieved by projecting the query onto their subspaces to predict the positive instance class. This generalizes the existing contrastive learning that can be viewed as a special one-shot case. An eigenvalue decomposition is performed to configure instance subspaces, and the embedding network can be trained end-to-end through the differentiable subspace configuration. Experiment results demonstrate the proposed $K$-shot contrastive learning achieves superior performances to the state-of-the-art unsupervised methods.

**Index Terms**—Unsupervised learning, self-supervised learning, contrastive learning

**1 INTRODUCTION**

Unsupervised learning of visual features has attracted wide attentions as it provides an alternative way to efficiently train very deep networks without labeled data that are often expensive to collect [1], [2]. Recent breakthroughs in this direction focus on two categories of methods: contrastive learning [3], [4], [5] and transformation prediction [6], [7], [8], [9], among many alternative unsupervised methods such as generative adversarial networks [10], [11], [12], [13], and auto-encoders [14], [15].

The former category [3], [4], [5], [16], [17], [18], [19] trains a network based on a self-training task by distinguishing between different instance classes each containing the samples augmented from the same instance. Such a contrastive learning problem seeks to explore the inter-instance discrimination to perform unsupervised learning. On the contrary, the other category of transformation prediction methods [6], [7], [8], [9] train a deep network by predicting the transformations used to augment input instances. It attempts to explore the intra-instance variations under multiple augmentations to learn the feature embedding.

A good visual representation ought to combine both advantages of the inter-instance discrimination and the intra-instance variations [8], [7], [20], [21]. In particular, the feature embedding should not only capture the significant intra-instance variations among augmented samples from each instance, as well as discern the distinction between instances by considering their potential variations to enable the inter-instance discrimination. In other words, the inter-instance discrimination should be performed by matching a query against all potential variants of an instance. To this end, we propose a novel $K$-shot contrastive learning as a first attempt to combine their strengths, and we will show that most of existing contrastive learning methods are a special one-shot case.

In particular, we apply multiple augmentations to transform each instance, resulting in an instance subspace spanned by the augmented samples. Each instance subspace learns significant factors of variations from the augmented samples, which configures how these factors can be linearly combined to form the variants of the instance. Then given a query, the most relevant sample of variant for each instance is retrieved by projecting the query onto the associated subspace [1], [2]. After that, the inter-instance discrimination is conducted by assigning the query to the instance class with the shortest projection distance. An eigenvalue decomposition is performed to configure each instance subspace with the orthonormal eigenvectors as its basis. This configuration of instance subspaces is non-parametric and differentiable, allowing an end-to-end training of the embedding network [23] through back-propagation.

Experiment results demonstrate that the proposed $K$-Shot Contrastive Learning (KSCL) can consistently improve the state-of-the-art performance on unsupervised learning. Particularly, with the ResNet50 backbone, it improves the top-1 accuracy of the SimCLR and the MoCo v2 to 68.8% on ImageNet over 200 epochs. It also reaches a higher top-1 accuracy of 71.4% over 800 epochs than the baseline SimCLR and the rerun MoCo v2. For the sake of fair comparison, all these improvements are achieved with the same experiment settings such as network architecture, data augmentation, training strategy and the version of deep learning framework and libraries. The consistently improved performances with the same model settings suggest the proposed KSCL can serve as a generic plugin to further increase the accuracy of contrastive learning methods on downstream tasks.

The remainder of the paper is organized as follows. We will review the related works in Section 2 and present...
the proposed $K$-shot contrastive learning in Section 3. Implementation details will be depicted in Section 4. We will demonstrate the experiment results in Section 5 and conclude the paper in Section 6.

2 RELATED WORKS

In this section, we review the related works to the proposed $K$-Short Contrastive Learning (KSCL) in the following four areas. A more comprehensive review of related methods on unsupervised models can be found in [20].

2.1 Contrastive Learning

Contrastive learning [3] was first proposed to learn unsupervised representations by maximizing the mutual information between the learned representation and a particular context. It usually focused on the context of the same instance to learn features by discriminating between one example from the other in an embedding space [4], [5], [24]. For example, the instance discrimination has been used as a pretext task by distinguishing augmented samples from each other in a minibatch [4], over a memory bank [24], or a dynamic dictionary with a queue [5]. The comparison between the augmented samples of individual instances was usually performed on a pairwise basis. The state-of-the-art performances on contrastive learning have relied on a composite of carefully designed augmentations [4] to prevent the unsupervised training from utilizing side information to accomplish the pretext task. This has been shown necessary to reach competitive results on downstream tasks.

2.2 Transformation Prediction

Transformation prediction [6], [7] also constitutes a category of unsupervised methods in learning visual embeddings. In contrast to contrastive learning that focuses on inter-instance discrimination, it aims to learn the representations that equivary against various transformations [8], [25], [26]. These transformations are used to augment images and the learned representation is trained to capture the visual structures from which these transformations can be recognized. It focuses on modeling the intra-instance variations from which variants of an instance can be leveraged on downstream tasks such as classification [6], [7], [25], object detection [6], [25], semantic segmentations on images [6], [27] and 3D cloud points [26]. This category of methods provide an orthogonal perspective to contrastive learning based on inter-instance discrimination.

2.3 Few-Shot Learning

From an alternative perspective, contrastive learning based inter-instance discrimination can be viewed as a special case of few-shot learning [28], [29], [30], [31], [32], where each instance is a class and it has several examples augmented from the instance. The difference lies that the examples for each class can be much abundant since one can apply many augmentations to generate an arbitrary number of examples. Of course, these examples are not statistically independent as they share the same instance. Based on this point of view, the non-parametric instance discrimination [24] and thus several perspective works [4], [5] can be viewed as an extension of the weight imprinting [33] by initializing the weights of each instance class with the embedded feature vector of an augmented sample, resulting in the inner product and cosine similarity used in these algorithms [4], [5], [24]. Such a surprising connection between the non-parametric instance discrimination and the few-shot learning may open a new way to train the contrastive prediction model. In this sense, the proposed $K$-shot contrastive learning generalizes the few-shot learning by imprinting the orthonormal basis of an instance subspace with the embeddings of augmented samples from the instance.

2.4 Capsule Nets

The length of a vectorized feature representation has been used in capsule nets pioneered by Hinton et al. [34], [35]. In capsule nets, a group of neurons form a capsule (vector) of which the direction represents different instantiation that equivaries against transformations and the length accounts for the confidence that a particular class of object is detected. From this point of view, the projected vector of a query example to an instance subspace in this paper also carries an analogy to a capsule. Its direction represents the instantiated configuration of how $K$-shot augmentations from the instance are linearly combined to form the query, while its length gives rise to the likelihood of the query belonging to this instance class, since a longer projection means a shorter distance to the subspace. This idea of using projections onto several capsule subspaces each corresponding to a class has shown promising results by effectively training deep networks [35].

3 THE APPROACH

In this section, we define a $K$-shot contrastive learning as the pretext task for training unsupervised feature embedding with Multiple Instance Augmentations (MIAs).

3.1 Preliminaries on Contrastive Learning

Suppose we are given a set of $N$ unlabeled instances $\mathcal{X} \triangleq \{x_n\}$ in a minibatch (e.g., in the SimCLR [4]) or from a dictionary (e.g., the memory bank in non-parametric instance discrimination [24] and the dynamic queue in the MoCo [5]). Then the contrastive learning can be formulated as classifying a query example $x$ into one of $N$ instance classes corresponding to an instance $x_n$.

The goal is to learn a deep network embedding each instance $x_n$ and the query $x$ to a feature vector $v_n$ and $v$. Then the probability of the embedded query $v$ belonging to an instance class $n$ is defined as

$$p(n|v) = \frac{\exp(\text{sim}(v_n, v)/\tau)}{\sum_{i=1}^{N} \exp(\text{sim}(v_i, v)/\tau)}$$

(1)

where a similarity measure $\text{sim}(\cdot, \cdot)$ (e.g., cosine similarity) is defined between two embeddings, and $\tau$ is a positive temperature hyperparameter. When the query $v$ is the embedding of an augmented sample from $x_n$, $p(n|v)$ gives rise to the probability of a relevant embedding $v_n$ being successfully retrieved from the instance class $n$. One can minimize the contrastive loss called InfoNCE in [3] resulting from the
negative log-likelihood of the above probability over a
dictionary to train the embedding network.

The idea underlying the contrastive learning approach
is a good representation ought to help retrieve the relevant
samples from a set of instances \( \mathcal{X} \) given a query \( \mathbf{x} \). For
example, the SimCLR \([4]\) has achieved the state-of-the-art
performance by applying two separate augmentations to
each instance in a mini-batch. Then, given a query example,
it views the sample augmented from the same instance as the
positive example, while treating those augmented from the
other instances as negative ones. Alternatively, the MoCo
\([5]\) seeks to retrieve relevant samples from a dynamic queue
spanned by the augmented samples of each instances. Given
the intra-instance variations through a linear subspace
factors of variations. Therefore, we are motivated to explore
intra-instance variations, since the most relevant sample of
each augmented sample individually fails to leverage such

Then, the projection of the query against the variants of each instance in the associated
instance subspace is

\[
\mathbf{x}_n\text{-shot augmentations.}
\]

Let \( \theta_{\mathbf{v}_n} \) be the acute angle between the query \( \mathbf{v} \) and the
instance subspace \( \mathcal{S}_n \). Then we have \( \| \Pi_n(\mathbf{v}) \|_2 = \cos(\theta_{\mathbf{v}_n}) \),
i.e., the projection length can be viewed as the cosine
similarity between the query and the whole instance subspace.

Compared with the cosine similarity between individual
embeddings of instances used in literature \([4], [5], [36]\),
it aims to learn a better representation by discriminating
different instance subspaces containing the variations of
sample augmentations.

Now we can define the probability of \( \mathbf{v} \) belonging to an
instance class \( n \)

\[
p(n|\mathbf{v}) = \frac{\exp(\|\Pi_n(\mathbf{v})\|_2/\tau)}{\sum_{n=1}^{N} \exp(\|\Pi_n(\mathbf{v})\|_2/\tau)}
\]

Then the KSCL seeks to train the embedding network by
maximizing the loglikelhood of the above probability over
mini-batches to match a query against the correct instance.
Particularly, given a query \( \mathbf{v} \) of a unit norm, its projection
length achieves its maximum if \( \mathbf{v} \) belongs to \( \mathcal{S}_n \), i.e., it is a
linear combination of \( K \)-shot augmentations \( \{ \mathbf{v}_n^k \} \). In other
words, it matches the query against all linear combinations of
the augmented samples from each instance \( \mathbf{x}_n \), and retrieves
the most similar one by projecting the query onto the instance
subspace with the shortest distance.

4 IMPLEMENTATIONS

In this section, we discuss the details to implement the
proposed \( K \)-Shot Contrastive Learning (KSCL) model.

4.1 Projection onto Instance Subspace via Eigenvalue
Decomposition

Mathematically, there is a close-form solution to the projection
\( \Pi_n(\mathbf{v} - \mathbf{m}_n) \) onto the instance subspace \( \mathcal{S}_n \) spanned by
\( K \)-shot augmentations \( \{ \mathbf{v}_n^k \} \)'s. Suppose there exists an
orthonormal basis for \( \mathcal{S}_n \) denoted by the columns of a matrix
\( \mathbf{W}_n \), the projection \( \Pi_n(\mathbf{v}) \) of a feature vector \( \mathbf{v} \) can be written as
\( \mathbf{v}_n^\top \mathbf{W}_n \mathbf{v} \).

Since we have \( \Pi_n(\mathbf{v}_n^k) = \mathbf{v}_n^k \) with \( \{ \mathbf{v}_n^k \} \) spanning \( \mathcal{S}_n \), the
problem of finding \( \mathbf{W}_n \) can be formulated by minimizing the
following projection residual

\[
\min_{\mathbf{W}_n} \sum_{n=1}^{K} \| \mathbf{v}_n^k - \Pi_n(\mathbf{v}_n^k) \| = \text{tr}(\mathbf{W}_n^\top \Sigma_n \mathbf{W}_n + \Sigma_n) \tag{4}
\]

where \( \Sigma_n = \mathbf{V}_n \mathbf{V}_n^\top \), with \( \mathbf{V}_n \triangleq [\mathbf{v}_n^1, \ldots, \mathbf{v}_n^K] \) containing the
embeddings of the \( K \) augmented samples in its columns.

After conducting an eigenvalue decomposition on the
positive-definite matrix \( \Sigma_n \), the eigenvectors corresponding
to the largest \( K \) eigenvalues give rise to an orthonormal basis
\( \mathbf{W}_n \) of the associated instance subspace, which minimizes
\( [4] \).

Since the eigenvalue decomposition is differentiable, the
embedding network can be trained end-to-end through the
error back-propagation. However, like the other contrastive
learning methods \([3], [36]\), the errors will only be back-
propagated through the embedding network of queries to
save the computing cost.
we will choose \( L \) this allows a distinct number of eigenvectors per instance to a preset percentage of total eigenvalues. The more percentage projection matrix \( \tilde{W} \) methods.

In this section, we perform experiments to compare the KSCL with the other state-of-the-art unsupervised learning

In practice, rather than setting \( L \) to a prefixed number, we will choose \( L \) such as the largest \( L \) eigenvalues cover a preset percentage of total eigenvalues. The more percentage of total eigenvalues are preserved, the smaller the projection residual is in Eq. (3); when \( L \geq K \), the residual vanishes. This allows a distinct number of eigenvectors per instance to flexibly model various degrees of variations among \( K \)-shot augmentations.

4.3 One-Shot Contrastive Learning when \( K = 1 \)

It is not hard to see that the cosine similarity used in SimCLR and MoCo is a special case when \( K = 1 \), i.e., they are one-shot contrastive learning of visual embeddings. When \( K = 1 \), there is a single augmented sample \( v_n \) per instance. Its instance subspace \( S_n \) collapses to a vector \( v_n \). Since \( v_n \) is \( \ell_2 \)-normalized to have a unit length in the SimCLR and the MoCo, the projection length of a query \( v \) to this single vector becomes \(|v_n^Tv|\). This is the cosine similarity between two vectors used in existing contrastive learning methods [4], [5], [24] up to an absolute value.

5 Experiments

In this section, we perform experiments to compare the KSCL with the other state-of-the-art unsupervised learning methods.
Table 1 shows that, after unsupervised pretraining of the KSCL with 200 epochs and a batch size of 256, the KSCL achieves a top-1 accuracy of 68.8% with $K = 5$ augmentations and $\rho = 40\%$ of preserved eigenvalues. It is worth noting that a larger batch size is often required to sufficiently train the SimCLR while the other models such as KSCL and MoCo maintain a long dynamic queue as the dictionary. By viewing the SimCLR with a larger batch size of 8, 192 as a baseline, the KSCL makes a much larger improvement of 2.1% than the MoCo v2 (0.9%) on the SimCLR baseline under 200 epochs. The KSCL also improves the top-1 accuracy to 71.4% on the ImageNet over 800 epochs of pretraining. Although a better result may be obtained by finetuning the hyperparameter and the data augmentation, we stick to the same experimental setting in the previous methods [4], [38] for a direct comparison.

We also visualize the learned basis images in Figure 2. The last column presents the basis images spanning the underlying instance subspace for a “cat” image. The weight beneath each image is the inner product between the decomposed eigenvector and the embedding of the corresponding augmentation, and each base is a weighted combination of the augmented images in the row. The results show that two bases suffice to capture the major variations among the five image augmentations, while the remaining three only model the minor ones that can be discarded as noises.

### 5.3 Impacts of $K$ and $\rho$ on Performance

We also study the impact of different $K$’s and $\rho$’s on the model performance. Table 2 shows the top-1 accuracy under various $K$’s and $\rho$’s. When $K = 1$, it reduces to one-shot contrastive learning which is similar to the MoCo v2. The difference 67.2% vs. 67.5% between the KSCL ($K = 1$) and the MoCo v2 is probably because we did not fine-tune the temperature $\tau$ for the projection length to optimize the KSCL.

The accuracy increases with a larger number of $K$ augmentations per instance and a smaller value $\rho$ of perceived eigenvalues. This implies that eliminating the minor noisy variations (as illustrated in Figure 2) with a smaller $\rho$ could improve the performance. Further growing $K$ only marginally improves the performance. This is probably because the data augmentation adopted in experiments is limited to those used in the compared methods for a direct comparison. Applying more types of augmentations (e.g., jigsaw and rotations) may inject more intra-instance variations that encourage to use a larger $K$. However, studying the role of more types of augmentations in contrastive learning is beyond the scope of this paper, and we leave it to future research.

### 5.4 Results on VOC Object Detection

Finally, we evaluate the unsupervised representations on the VOC object detection task [39]. The ResNet-50 backbone pretrained on the ImageNet dataset is fine-tuned with a Faster RCNN detector [40] end-to-end on the VOC 2007+2012 trainval set, and is evaluated on the VOC 2007 test set. Table 3 compares the results with both the MoCo models. Under the same setting, the proposed KSCL outperforms the compared MoCo v1 and MoCo v2 models. The SimCLR model does not report on the VOC object detection task in [4].

### 6 Conclusion

In this paper, we present a novel $K$-shot contrastive learning to learn unsupervised visual features. It randomly draws $K$-shot augmentations and applies them separately to each instance. This results in the instance subspace modeling how the significant factors of variances learned from the augmented samples can be linearly combined to form the variants of an associated instance. Given a query, the most relevant samples are then retrieved by projecting the query onto individual instance subspaces, and the query is assigned to the instance subspace with the shortest projection distance. The proposed $K$-shot contrastive learning combines the advantages of both the inter-instance discrimination and the intra-instance variations to discriminate the distinctions between different instances. The experiment results demonstrate its superior performances to the state-of-the-art contrastive learning methods based on the same experimental setting.

### References

[1] X.-S. Hua and G.-J. Qi, “Online multi-label active annotation: towards large-scale content-based video search,” in *Proceedings of the 16th ACM international conference on Multimedia*, 2008, pp. 141–150.
Fig. 2: The learned basis in an instance subspace. Each of the first five columns is an augmented image from an instance, and the last column is the basis images each of which is synthesized as a linear combination of the five augmented images weighted by the inner product with the corresponding eigenvector in the embedding space.

[2] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, T. Mei, M. Wang, and H.-J. Zhang, “Correlative multilabel video annotation with temporal kernels,” ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), vol. 5, no. 1, pp. 1–27, 2008.

[3] A. v. d. Oord, Y. Li, and O. Vinyals, “Representation learning with contrastive predictive coding,” arXiv preprint arXiv:1807.03748, 2018.

[4] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” arXiv preprint arXiv:2002.05709, 2020.

[5] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, “Momentum contrast for unsupervised visual representation learning,” arXiv preprint arXiv:1911.05722, 2019.

[6] S. Gidaris, P. Singh, and N. Komodakis, “Unsupervised representation learning by predicting image rotations,” arXiv preprint arXiv:1803.07728, 2018.

[7] L. Zhang, G.-J. Qi, L. Wang, and J. Luo, “Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data,” arXiv preprint arXiv:1901.04596, 2019.

[8] G.-J. Qi, L. Zhang, C. W. Chen, and Q. Tian, “Avt: Unsupervised learning of transformation equivariant representations by auto-encoding variational transformations,” in Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 8130–8139.

[9] M. Norouzi and P. Favaro, “Unsupervised learning of visual representations by solving jigsaw puzzles,” in European Conference on Computer Vision. Springer, 2016, pp. 69–84.

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Advances in neural information processing systems, 2014, pp. 2672–2680.

[11] G.-J. Qi, “Loss-sensitive generative adversarial networks on lipschitz densities,” International Journal of Computer Vision, vol. 128, no. 5, pp. 1118–1140, 2020.

[12] G.-J. Qi, L. Zhang, H. Hu, M. Edraki, J. Wang, and X.-S. Hua, “Global versus localized generative adversarial nets,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1517–1525.

[13] Y. Zhao, Z. Jin, G.-j. Qi, H. Lu, and X.-s. Hua, “An adversarial approach to hard triplet generation,” in Proceedings of the European conference on computer vision (ECCV), 2018, pp. 501–517.

[14] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, “Stacked convolutional auto-encoders for hierarchical feature extraction,” in International conference on artificial neural networks. Springer, 2011, pp. 52–59.

[15] G. E. Hinton, A. Krizhevsky, and S. D. Wang, “Transforming auto-encoders,” in International conference on artificial neural networks. Springer, 2011, pp. 44–51.

[16] P. Bachman, R. D. Hjelm, and W. Buchwalter, “Learning representations by maximizing mutual information across views,” in Advances in Neural Information Processing Systems, 2019, pp. 15509–15519.

[17] O. J. Hénaff, A. Srinivas, J. De Fauw, A. Razavi, C. Doersch, S. Eslami, and A. v. d. Oord, “Data-efficient image recognition
TABLE 2: The top-1 accuracy of the proposed KSCL with varying $K$’s and $\rho$’s under 200 epochs of pretraining on ImageNet. The ResNet-50 backbone was pretrained with two-layer MLP head by applying the same combination of enhanced data augmentations used in SimCLR including stronger color distortion and blurring for a fair comparison. The top-1 accuracy is obtained by training a single-layer linear classifier upon the pretrained features. We also compare the computing time used to train the KSCL per epoch in eight V100. Note when $K = 1$, $\rho$ need not be set as it becomes a trivial case of an one-shot contrastive learning.

| $K$ | $\rho$ | epochs | batch size | top-1 accuracy | time/epoch (min.) |
|-----|--------|---------|------------|----------------|-------------------|
| 1   | –      | 200     | 256        | 67.2           | 16                |
| 3   | 40%    | 200     | 256        | 68.5           | 26                |
| 5   | 40%    | 200     | 256        | 68.8           | 37                |
| 5   | 90%    | 200     | 256        | 68.4           | 37                |

TABLE 3: The comparison between the proposed KSCL ($K = 5$ and $\rho = 40\%$) and the MoCo models. The pretrained ResNet-50 backbone was transferred to train on VOC 2007+2012 trainval set with a Faster R-CNN detector end-to-end, and evaluated on the VOC 2007 test set. The COCO metrics were adopted to evaluate the performance.

| Model | epochs | batch size | AP$_{50}$ | AP | AP$_{75}$ |
|-------|--------|------------|-----------|----|-----------|
| MoCo v1 | 200     | 256        | 81.5      | 55.9 | 62.6         |
| MoCo v2 | 200     | 256        | 82.4      | 57.0 | 63.6         |
| MoCo v2 | 800     | 256        | 82.5      | 57.4 | 64.0         |
| KSCL  | 200     | 256        | 82.4      | 57.1 | 63.9         |
| KSCL  | 800     | 256        | 82.7      | 57.5 | 64.2         |

with contrastive predictive coding,” arXiv preprint arXiv:1905.09272, 2019.

[18] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, “Learning deep representations by mutual information estimation and maximization,” arXiv preprint arXiv:1808.06670, 2018.

[19] Y. Tian, D. Krishnan, and P. Isola, “Contrastive multiview coding,” arXiv preprint arXiv:1906.05849, 2019.

[20] G.-J. Qi and J. Luo, “Small data challenges in big data era: A survey of recent progress on unsupervised and semi-supervised methods,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020.

[21] J. Wang, Z. Zhao, J. Zhou, H. Wang, B. Cui, and G. Qi, “Recommending flickr groups with social topic model,” Information retrieval, vol. 15, no. 3–4, pp. 278–295, 2012.

[22] X. Shu, J. Tang, G.-J. Qi, Z. Li, Y.-C. Jiang, and S. Yan, “Image classification with tailored fine-grained dictionaries,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 28, no. 2, pp. 454–467, 2016.

[23] S. Chang, G.-J. Qi, C. C. Aggarwal, J. Zhou, M. Wang, and T. S. Huang, “Factorized similarity learning in networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2020, pp. 7163–7172.

[24] X. Gao, W. Hu, and G.-J. Qi, “Graphter: Unsupervised learning of graph equivariant representations via auto-encoding node-wise transformations,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 7163–7172.

[25] G.-J. Qi, “Hierarchically gated deep networks for semantic segmentation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2267–2275.

[26] Z. Peng, Z. Li, J. Zhang, Y. Li, G.-J. Qi, and J. Tang, “Few-shot image recognition with knowledge transfer,” in Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 441–449.

[27] G.-J. Qi, “Hierarchically gated deep networks for semantic segmentation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 11719–11727.

[28] H. Xu, H. Xiong, and G. Qi, “Flat: Few-shot learning via autoencoding transformation regularizers,” arXiv preprint arXiv:1912.12674, 2019.