

A Game of Finding Delicate Data Augmentations: Reinforcement Contrastive Learning of Visual Representations

Zehua Zang^{1,2}[0000-0001-8969-3218], Rui Wang²[0000-0001-5369-9116], Lixiang Liu²[0000-0002-3890-1758], Fuchun Sun^{2,3}[0000-0003-3546-6305]

¹ University of Chinese Academy of Sciences, Beijing, 101408, China

² Institute of Software Chinese Academy of Sciences, Beijing, 100190, China

³ Tsinghua University, Beijing, 100084, China
wangrui@iscas.ac.cn

Abstract. Self-supervised contrastive learning aims to extract discriminative information from data by conducting instance-level contrastive analysis. However, this approach is limited by stochastic data augmentation techniques, which may result in either trivial augmentations that fail to enhance the contrastive model or excessive augmentations that impede model convergence. To address these, recent studies have explored the use of Reinforcement Learning (RL) to develop dataset-specific data augmentation strategies by leveraging pre-trained models in a supervised manner. Nevertheless, these methods are heavily dependent on the quality of the pre-trained models and are not well-suited for unsupervised learning scenarios. Data augmentations generated by these methods are dataset-specific and remain static during training, which leads the model to converge to a local optimum. Thus, we propose a novel approach that frames the task of identifying effective data augmentations for contrastive learning as a game, thereby transforming the parameter tuning problem into a policy search task that aligns with RL. Building on this, we introduce an online self-supervised framework designed to generate challenging yet convergent self-supervised contrastive tasks. We evaluate the proposed method on multiple datasets for visual self-supervised representation learning and empirically demonstrate that our approach achieves significant performance improvements at most 1.2%.

Keywords: Contrastive Learning, Self-Supervised Learning, Reinforcement Learning.

1 Introduction

Due to their demonstrated empirical efficacy, contrastive learning approaches have gained significant traction as a prominent methodology for acquiring visual representations in a self-supervised framework [1, 2, 3, 4, 5]. Within the domain of computer vision, self-supervised contrastive learning techniques have been shown to surpass the performance of established supervised benchmarks in key tasks such as object detection and image classification [6, 7].

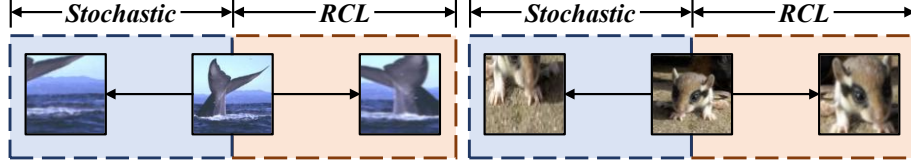


Fig. 1. Comparison between stochastic augmentations and RCL data augmentations.

Contrastive learning fundamentally hinges on two critical components [8]: the selection of similar positive pairs (x^*, x^+) and dissimilar negative pairs (x^*, x^-) , where x^* represents the anchor sample, while x^+ and x^- denote the positive and negative samples, respectively. The training objective, typically formulated as noise-contrastive estimation (NCE) [9, 10], trains the feature extractor $f(\cdot)$ to cluster positives and repel negatives in latent space. However, learning efficiency depends critically on pair selection: overly similar positives or inherently distant negatives provide weak training signals, limiting discriminative power [4, 11]. To address this challenge, recent methodologies have sought to impose stronger constraints on the contrastive learning paradigm. DebiasedCL [12] and HardCL [13] improves contrastive learning via weighted sampling of informative pairs, inspired by positive-unlabeled learning [14, 15]. However, their reliance on stochastic data augmentations fundamentally limits gains. While MetAug [4] and RandAugment [16] introduce feature-level or constrained augmentations, they still suffer from a key weakness: unreliable augmentation quality. Nevertheless, these approaches still depend on stochastic data augmentations, which inherently suffer from a critical limitation: the informativeness of the augmented inputs generated through stochastic methods cannot be guaranteed. This is because such approaches produce trivial augmentations that fail to provide meaningful learning signals, thereby limiting the effectiveness of the model. The examples in **Fig. 1** substantiate this assertion.

To tackle the intrinsic challenges present in current methods for enhancing data augmentations, we propose an online approach specifically designed for the self-supervised representation learning paradigm, termed **Reinforcement Contrastive Learning (RCL)**. By framing the task of identifying optimal data augmentations as a game, we transform the parameter tuning challenge into a policy search problem, aligning naturally with the principles of RL. RCL employs a **Reinforced Data Augmentation Generator (RAG)** to produce augmented images in a self-supervised manner. Central to this framework is the conceptual definition of delicate data augmentations for contrastive learning, which are designed to introduce meaningful challenges into the training process, thereby enhancing the robustness and discriminative power of self-supervised visual representations. To achieve this, RAG employs a hierarchical agent structure with a highly organized action space. The main agent, functioning as the high-level agent, oversees the selection of data augmentation types, while each sub-game agent, acting as the low-level agent, generates the parameters for the chosen augmentation type. During the augmentation process, the main agent first selects a data augmentation type, and the corresponding sub-game agent then produces the parameters, resulting in the final augmented images. Recognizing that features better represent the state of images and the current contrastive model compared to raw images, we utilize features

extracted by the backbone network as the state for each agent rather than the original images. This approach not only aligns with the intuitive understanding of feature representations but also reduces computational costs. Additionally, we introduce a task-free reward function based on InfoNCE loss [1, 17], enabling our method to operate without requiring pre-trained models or annotated data. It is important to note that RAG and the backbone network are trained iteratively until convergence, ensuring that the data augmentations generated by RAG evolve dynamically alongside the training of the backbone network. Since RAG operates on the features of individual images, the resulting augmentations are instance-oriented rather than dataset-oriented. Our key contributions are as follows:

1. We propose a self-supervised reinforcement contrastive learning approach RCL to address the challenges existing in data augmentation improvement methods.
2. We propose the reinforced data augmentation generator RAG to generate instance-level data augmentations, which are dynamically updated, aligning with the training of the backbone network.
3. Empirical results demonstrate that RCL generates nontrivial augmented samples (at most 1.2%), which contributes to the challenging while convergent contrastive task.

2 Related Works

2.1 Self-Supervised Contrastive Learning

Contrastive learning [39, 41, 43] aims to learn a representation space where anchor points are close to positives and distant from negatives. Noise Contrastive Estimation (NCE) [17] treats each instance as a unique class to address computational challenges. SimCLR [6, 8] uses augmented views of the same image as positives and other images as negatives, benefiting from larger batch sizes. MoCo [3, 18] and CMC [1] accumulate negatives in a memory bank for efficient contrastive learning. BYOL [2] and SimSiam [7] forego negatives, instead relying on dual networks or stop-gradient techniques. Unlike these methods, our proposed learnable data augmentation policy mitigates the instability caused by stochastic augmentations.

2.2 Data Augmentation with RL

RL [38, 40, 42] enables agents to make sequential decisions to maximize cumulative rewards. Based on optimization methods, RL techniques are generally categorized into value-based methods [45, 46] and policy-gradient methods [19, 20].

Several studies [21, 22, 23, 24, 25, 26] leverage RL to automate data augmentation. Smart Augmentation [21] combines same-class samples to boost generalization. AutoAugment [22] uses an RNN to find optimal policies, while PBA [23] evolves them during training. Fast AutoAugment [24] discovers strategies post-training using unseen data. Adversarial AutoAugment [25] speeds up training with dataset augmentations, and Deep AutoAugment [26] replaces defaults with fully automated searches.

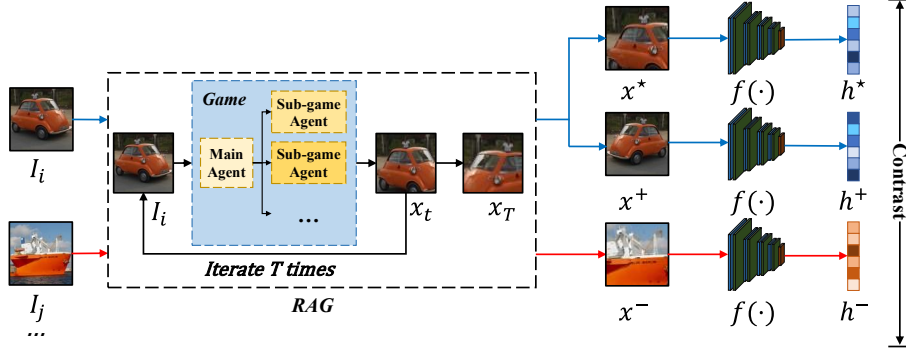


Fig. 2. The framework of RCL. An image I_i is fed into RAG to generate the anchor sample x^* and the positive sample x^+ . Other images $\{I_j, \dots\}$ in the same minibatch are processed by RAG to generate negative samples $\{x^-\}$. Then, we extract features $\{h\}$ of the augmented data $\{x\}$. The contrastive model is trained by the contrastive loss. Given a feature h_i , the main agent selects a data augmentation type according to the feature h_i . The corresponding sub-game agent generates the parameters of specific data augmentation type, which derives a temporary augmented image x_t in the procedure. Such process is iterated in T times to get the final augmented image x_T .

3 Method

3.1 A Game of Finding Delicate Data Augmentation

In this section, we introduce a RL framework designed to identify effective data augmentation strategies for contrastive learning. Given an input image I , our objective is to generate an augmented sample that introduces an appropriate level of challenge to the training process. The generation process can be formally defined as follows:

$$x_T = A(I); \quad A = A_1 \circ \dots \circ A_n, \quad (1)$$

where $\circ$ denotes function composition, and A_i represents a specific data augmentation process. Unlike existing methods [1, 2, 7, 8, 18] that rely on stochastic augmentations, our approach employs multiple delicate data augmentations, which are learned and optimized through a RL framework. This ensures that the augmentations are tailored to enhance the contrastive learning process effectively.

3.2 Reinforcement Contrastive Learning

The overall framework of RCL is illustrated in **Fig. 2**. Given a set of input images $I_i, I_j, \dots$, the RAG module produces refined augmented variants of each image. Both the main agent and the sub-game agents leverage features extracted from these images to inform their data augmentation strategies over a sequence of T steps.

State Space: The primary challenge is constructing a meaningful state space from high-dimensional raw images. Thus, we introduce a parameter-sharing encoder to extract image features, which is related to the state space. The state at time-step t is formulated as follows:

$$s_t = h_t = f(x_t), \quad (2)$$

where $f(\cdot)$ denotes the encoder of the contrastive model, x_t represents the input image at time step t , and h_t corresponds to the feature representation extracted from x_t . At each time step, the feature $h_t = f(x_t)$ is computed and subsequently used as input by both the main agent and the sub-game agents. This feature-based representation not only preserves task-relevant information but also significantly reduces computational complexity due to its lower dimensionality compared to raw pixel inputs.

Table 1. The action space of each agent.

Agent	Agent Type	Action Type	Range
Main	Main	Discrete	0-ResizedCrop, 1-Flipping
ResizedCrop	Sub-game	Continuous	[Top_pos, Bottom_pos, Left_pos, Right_pos]
Flipping	Sub-game	Discrete	0-Flip, 1-None

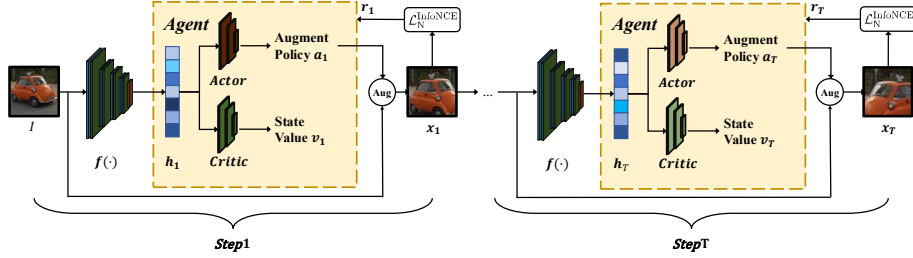


Fig. 3. Agent training framework. At step t , the feature h_t of the image x_{t-1} is extracted by the contrastive model $f(\cdot)$, i.e., the feature extractor. Then, the feature h_t is fed into the actor and critic network. The actor network output the augmentation policy a_t , and the critic network estimates the value of a_t . The augmented image x_t is generated by the augmentation policy a_t . In addition, the reward r_t is computed based on the InfoNCE loss.

Action Space: We employ a suite of data augmentation operations, each characterized by parameters of varying types and magnitudes. The resulting action space is highly structured and poses substantial challenges for exploration. To mitigate this complexity, we adopt a hierarchical RL framework that decomposes the raw action space into more tractable subspaces. Specifically, this framework comprises a main agent responsible for selecting the augmentation type and a set of sub-agents generates the parameters, respectively. This hierarchical decision process is illustrated in **Fig. 3**, with the corresponding action space formally detailed in **Table 1**.

Reward Function: A central challenge is designing an effective reward signal that accurately reflects the quality of augmentations and facilitates efficient policy learning. In this work, we investigate the role of prior knowledge in the data augmentation task and introduce a reward function that is independent of pre-trained models. Our proposed reward function can be broadly applicable across different datasets, thereby enhancing generalizability and encouraging richer exploration during augmentation.

The reward function encourages the agent to explore diverse augmentation strategies. It guides the agent to generate nuanced positives that differ significantly from the anchor image, making the contrastive task more challenging, while negatives are crafted to appear visually like the anchor at first glance, increasing task difficulty and

promoting better convergence. To this end, we introduce a novel reward function grounded in a bounded variant of the InfoNCE objective, as followed.

$$r_t = \frac{1}{|\beta \mathcal{L}_N^{\text{InfoNCE}}(h) - \beta \mathcal{L}_N^{\text{InfoNCE}}(h^A)|}, \quad (3)$$

where r_t is the reward function, h is the feature extracted by encoder f_θ from the original image I , h^A is the feature extracted by encoder f_θ from the augmented image x^T , β is a hyperparameter, and $\mathcal{L}_N^{\text{InfoNCE}}$ is the InfoNCE loss function:

$$\mathcal{L}_N^{\text{InfoNCE}} = -\log \frac{\exp(\text{sim}(\mathbf{h}^*, \mathbf{h}^+))}{\sum_{k=1}^N \exp(\text{sim}(\mathbf{h}^*, \mathbf{h}^k))} \quad (4)$$

The formulation of **Equation 3** and the selection of the hyperparameter β play a pivotal role in constraining the training objective of the RAG module, thereby ensuring stable convergence during learning. In summary, the proposed reward function provides two key advantages: (1) it operates in a fully self-supervised manner, eliminating the dependence on task-specific labels or supervised learning signals, and thus enables agent training without additional annotation overhead; and (2) it effectively guides the model toward globally optimal solutions by promoting informative and discriminative data augmentations within the contrastive learning framework.

4 Experiments

In this section, we evaluate the performance of our proposed RCL approach against established baseline methods and conduct a series of ablation experiments.

4.1 Main Experiments

To demonstrate the performance of RCL, we conduct experiments evaluating Top-1 accuracy on the image classification task across four datasets: STL-10 [29], CIFAR-10 [30], CIFAR-100 [30], and Tiny-ImageNet [32]. The baselines for comparison include state-of-the-art methods from recent years, such as CMC [1], SimCLR [8], MoCo-v2 [18], Barlow Twins [36], BYOL [37], SwAV [33], DINO [34], ESSL [35].

The experimental results are summarized in **Table 2**. We integrate RCL with four representative SOTA—CMC, SwAV, DINO, and ESSL. Across all configurations, the incorporation of RCL consistently yields competitive or superior performance. Notably, RCL+ESSL achieves the highest accuracy on CIFAR-10 (84.8%), CIFAR-100 (51.7%), and Tiny-ImageNet (34.3%), while RCL+DINO attains the best result on STL-10 (87.4%). Overall, the results highlight the potential of RCL as a versatile augmentation framework that can be seamlessly integrated with existing methods to achieve improved performance.

Table 2. Classification performance on four image datasets.

Methods	Epochs	STL-10	CIFAR-10	CIFAR-100	Tiny-ImageNet
CMC	500	81.1	73.9	46.9	18.3
SimCLR	500	72.9	69.5	31.4	24.2
MoCo-v2	500	80.1	65.7	43.2	21.3

BarlowTwins	500	85.5	83.0	51.1	32.7
BYOL	500	83.7	50.6	26.2	32.3
SwAV	500	85.1	79.6	49.4	32.9
DINO	500	85.8	80.3	50.2	33.5
ESSL	500	86.1	84.3	51.1	32.0
RCL + CMC	500	81.7	74.3	47.3	20.7
RCL + SwAV	500	86.5	80.3	51.0	34.0
RCL + DINO	500	87.4	80.4	50.6	33.6
RCL + ESSL	500	87.3	84.8	51.7	34.3

4.2 Effectiveness of RAG and Time Complexity Analysis

To validate the hypothesis that a slightly elevated InfoNCE loss can benefit the training of contrastive models, we design a controlled toy experiment. Specifically, given an input image I , we first compute its current InfoNCE loss with the original view, denoted by $\mathcal{L}^{\text{InfoNCE}}(I)$. We then apply stochastic data augmentation $N = 20$ times to generate a set of augmented images $\{I_i^A\}_{i=1}^N$. To guide the selection of a challenging but informative augmentation, we introduce a hyperparameter $\beta = 1.5$. The selected image Ix^A for contrastive learning is determined as:

$$I^A = \arg \min_{I_i^A} \left| \beta \mathcal{L}^{\text{InfoNCE}}(f(I_i)) - \mathcal{L}^{\text{InfoNCE}}(f(I_i^A)) \right| \quad (5)$$

Training with slightly higher InfoNCE loss samples helps the contrastive model escape poor local minima and learn stronger representations. **Table 3** compares three settings: (1) standard training, (2) training with Slightly Larger Loss Samples (SLLS), and (3) training with our RCL framework. SLLS boosts performance by selecting harder augmentations but causes instability and high cost. In contrast, RCL maintains both efficiency and robustness. **Fig. 4** shows RAG's reward curve convergence, with shaded areas representing standard deviation.

For complexity perspective, RCL exhibits a time complexity of $O(n)$ over M episodes with T steps per episode. Nevertheless, the use of light agent models and low-dimensional feature representations ensures the added overhead remains manageable.

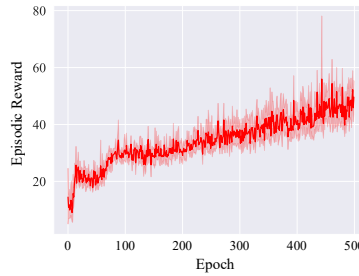


Fig. 4. Training reward curves of RAG module.

Table 3. Time complexity and Top1 Acc of CMC, CMC+SLLS and RCL on STL-10.

Methods	Time Cost	Top1 Acc
---------	-----------	----------

CMC	1.000	81.07
CMC + SLLS	14.408	81.45
CMC + RAG	3.166	81.67

4.3 Comparison to Other Types of Data Augmentation

RCL uses RL to dynamically optimize augmentation strategies for each image, unlike AugMix [44] and AutoAugment, which apply fixed or stochastic policies. RCL tailors augmentations based on each image's characteristics and training context, enabling more context-aware transformations that preserve semantic consistency and improve contrastive learning. This adaptability enhances robustness and generalizability compared to static policies. As shown in **Table 4**, RCL outperforms AutoAugment and AugMix, achieving the highest performance with CMC as the base architecture.

Table 4. Top1 Acc of RCL, AutoAugment and AugMix on Tiny-ImageNet.

Methods	RCL	AutoAugment	AugMix
CIFAR10	20.7	18.9	20.1

4.4 Ablation Study on The Game Training Epochs

To examine the trade-offs between performance, cost, and training scale, we vary training epochs (game size) across [50, 75, 100, 125]. As shown in **Fig. 5**, computational time increases linearly with epochs (Fig. 5a), while top-1 accuracy shows diminishing returns, rising from 80.5% to 82.5% with a flattening trend (Fig. 5b). Error bars indicate moderate variance across runs.

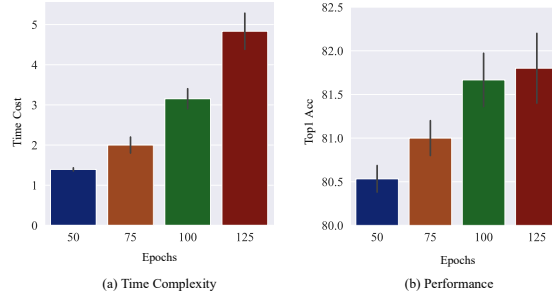


Fig. 5. Time complexity from a range of different game training epochs (a) and the performance from a range of different game training epochs (b).

Given the trade-off between rising computational cost and slowing performance gains, we choose 100 training epochs as the optimal setting. This strikes a balance between efficiency and performance, avoiding the steep resource cost of minimal improvements beyond this point—a choice aligned with typical cost-benefit considerations in resource-limited settings.

4.5 Visualizations of Data Augmentation

To illustrate RCL's advantages, **Fig. 6** compares images from stochastic augmentations and RAG. Stochastic methods often retain irrelevant backgrounds and sometimes

remove key foreground features. In contrast, RAG gradually removes background clutter while preserving essential structures—for instance, keeping the full car shape or the cat’s face intact. By preserving discriminative features and filtering out irrelevant content, RCL produces more effective augmentations for contrastive learning, leading to better performance.

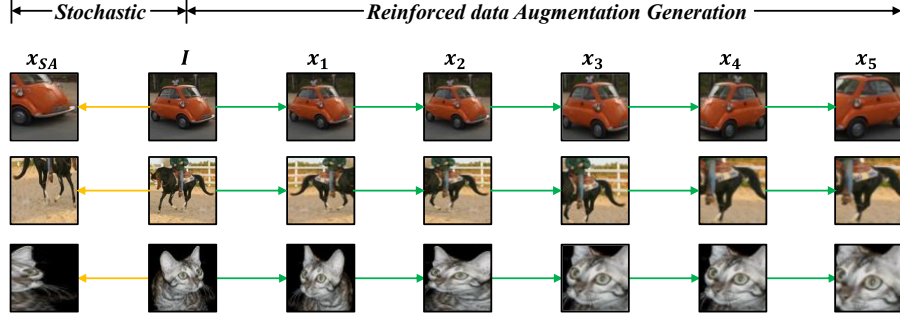


Fig. 6. Data Augmentation Visualization.

5 Conclusions

To address key limitations in current self-supervised contrastive learning, we frame augmentation selection as a game, turning parameter tuning into a policy search problem. This leads to an online approach that seeks challenging yet convergent contrastive tasks. Experiments across multiple datasets show that RCL significantly outperforms state-of-the-art methods in visual representation learning.

References

1. Tian, Y. et al.: Contrastive Multiview Coding. In: Vedaldi, A. et al. (eds.) Computer Vision - ECCV 2020, Part XI. Lecture Notes in Computer Science, vol. 12356, pp. 776–794. Springer, Cham (2020).
2. Grill, J.-B. et al.: Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning. In: Larochelle, H. et al. (eds.) Advances in Neural Information Processing Systems 33 (NeurIPS 2020), pp. 1–12. Springer, Cham (2020).
3. Chen, X. et al.: Improved Baselines with Momentum Contrastive Learning. CoRR, abs/2003.04297 (2020).
4. Li, J. et al.: MetAug: Contrastive Learning via Meta Feature Augmentation. In: Chaudhuri, K. et al. (eds.) International Conference on Machine Learning (ICML 2022). Proceedings of Machine Learning Research, vol. 162, pp. 12964–12978. PMLR (2022).
5. Qiang, W. et al.: Interventional Contrastive Learning with Meta Semantic Regularizer. In: Chaudhuri, K. et al. (eds.) International Conference on Machine Learning (ICML 2022). Proceedings of Machine Learning Research, vol. 162, pp. 18018–18030. PMLR (2022).
6. Chen, T. et al.: Big Self-Supervised Models are Strong Semi-Supervised Learners. In: Larochelle, H. et al. (eds.) Advances in Neural Information Processing Systems 33 (NeurIPS 2020), pp. 1–12. Springer, Cham (2020).

7. Chen, X. et al.: Exploring Simple Siamese Representation Learning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2021), pp. 15750–15758. Computer Vision Foundation / IEEE (2021).
8. Chen, T. et al.: A Simple Framework for Contrastive Learning of Visual Representations. In: Proceedings of the 37th International Conference on Machine Learning (ICML 2020). Proceedings of Machine Learning Research, vol. 119, pp. 1597–1607. PMLR (2020).
9. van den Oord, A. et al.: Representation Learning with Contrastive Predictive Coding. CoRR, abs/1807.03748 (2018).
10. Gutmann, M. et al.: Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: Teh, Y.W. et al. (eds.) Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2010). JMLR Proceedings, vol. 9, pp. 297–304. JMLR.org (2010).
11. Wang, Y. et al.: Chaos is a Ladder: A New Theoretical Understanding of Contrastive Learning via Augmentation Overlap. In: The Tenth International Conference on Learning Representations (ICLR 2022). OpenReview.net (2022).
12. Chuang, C.-Y. et al.: Debaised Contrastive Learning. In: Larochelle, H. et al. (eds.) Advances in Neural Information Processing Systems 33 (NeurIPS 2020), pp. 1–12. Springer, Cham (2020).
13. Robinson, J.D. et al.: Contrastive Learning with Hard Negative Samples. In: 9th International Conference on Learning Representations (ICLR 2021). OpenReview.net (2021).
14. Elkan, C. et al.: Learning classifiers from only positive and unlabeled data. In: Li, Y. et al. (eds.) Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 213–220. ACM (2008).
15. du Plessis, M.C. et al.: Analysis of Learning from Positive and Unlabeled Data. In: Ghahramani, Z. et al. (eds.) Advances in Neural Information Processing Systems 27 (NeurIPS 2014), pp. 703–711. Springer, Cham (2014).
16. Cubuk, E.D. et al.: RandAugment: Practical Automated Data Augmentation with a Reduced Search Space. In: Larochelle, H. et al. (eds.) Advances in Neural Information Processing Systems 33 (NeurIPS 2020), pp. 1–12. Springer, Cham (2020).
17. Wu, Z. et al.: Unsupervised Feature Learning via Non-Parametric Instance Discrimination. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018), pp. 3733–3742. Computer Vision Foundation / IEEE Computer Society (2018).
18. He, K. et al.: Momentum Contrast for Unsupervised Visual Representation Learning. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020), pp. 9726–9735. Computer Vision Foundation / IEEE (2020).
19. Schulman, J. et al.: Proximal Policy Optimization Algorithms. CoRR, abs/1707.06347 (2017).
20. Haarnoja, T. et al.: Soft Actor-Critic: Off-Policy Maximum Entropy Deep RL with a Stochastic Actor. In: Dy, J.G. et al. (eds.) Proceedings of the 35th International Conference on Machine Learning (ICML 2018). Proceedings of Machine Learning Research, vol. 80, pp. 1856–1865. PMLR (2018).
21. Lemley, J. et al.: Smart Augmentation Learning an Optimal Data Augmentation Strategy. IEEE Access, vol. 5, pp. 5858–5869 (2017).
22. Cubuk, E.D. et al.: AutoAugment: Learning Augmentation Policies from Data. CoRR, abs/1805.09501 (2018).
23. Ho, D. et al.: Population Based Augmentation: Efficient Learning of Augmentation Policy Schedules. In: Chaudhuri, K. et al. (eds.) Proceedings of the 36th International Conference on Machine Learning (ICML 2019). Proceedings of Machine Learning Research, vol. 97, pp. 2731–2741. PMLR (2019).

24. Lim, S. et al.: Fast AutoAugment. In: Wallach, H.M. et al. (eds.) *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, pp. 6662–6672. Springer, Cham (2019).
25. Zhang, X. et al.: Adversarial AutoAugment. In: *8th International Conference on Learning Representations (ICLR 2020)*. OpenReview.net (2020).
26. Zheng, Y. et al.: Deep AutoAugment. In: *The Tenth International Conference on Learning Representations (ICLR 2022)*. OpenReview.net (2022).
27. Schulman, J. et al.: High-Dimensional Continuous Control Using Generalized Advantage Estimation. In: Bengio, Y. et al. (eds.) *4th International Conference on Learning Representations (ICLR 2016)*. Conference Track Proceedings (2016).
28. Sutton, R.S. et al.: *RL - An Introduction*. Adaptive Computation and Machine Learning. MIT Press (1998).
29. Coates, A. et al.: An Analysis of Single-Layer Networks in Unsupervised Feature Learning. In: Gordon, G.J. et al. (eds.) *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2011)*. JMLR Proceedings, vol. 15, pp. 215–223. JMLR.org (2011).
30. Krizhevsky, A. et al.: Learning Multiple Layers of Features from Tiny Images. Technical Report, University of Toronto (2009).
31. Szegedy, C. et al.: Rethinking the Inception Architecture for Computer Vision. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, pp. 2818–2826. IEEE Computer Society (2016).
32. Deng, J. et al.: ImageNet: A Large-Scale Hierarchical Image Database. In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pp. 248–255. IEEE Computer Society (2009).
33. Caron, M. et al.: Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In: Larochelle, H. et al. (eds.) *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*. Springer, Cham (2020).
34. Zhang, H. et al.: DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection. In: *The Eleventh International Conference on Learning Representations (ICLR 2023)*. OpenReview.net (2023).
35. Li, J. et al.: Rethinking Generalizability and Discriminability of Self-Supervised Learning from Evolutionary Game Theory Perspective. CoRR, abs/2412.00542 (2024).
36. Zbontar, J. et al.: Barlow Twins: Self-Supervised Learning via Redundancy Reduction. In: Meila, M. et al. (eds.) *Proceedings of the 38th International Conference on Machine Learning (ICML 2021)*. Proceedings of Machine Learning Research, vol. 139, pp. 12310–12320. PMLR (2021).
37. Grill, J.-B. et al.: Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning. In: Larochelle, H. et al. (eds.) *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*. Springer, Cham (2020).
38. Sun, C. et al.: T2MAC: Targeted and Trusted Multi-Agent Communication through Selective Engagement and Evidence-Driven Integration. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2024)*, vol. 38, no. 13, pp. 15154–15163. Springer, Cham (2024).
39. Li, J. et al.: Information Theory-Guided Heuristic Progressive Multi-View Coding. *Neural Networks*, vol. 167, pp. 415–432 (2023).
40. Sun, C. et al.: Intrinsic Motivated Multi-Agent Communication. In: *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2021)*, pp. 1668–1670. Springer, Cham (2021).

41. Li, J. et al.: Modeling Multiple Views via Implicitly Preserving Global Consistency and Local Complementarity. *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 7, pp. 7220–7238 (2023).
42. Sun, C. et al.: M2I2: Learning Efficient Multi-Agent Communication via Masked State Modeling and Intention Inference. *arXiv preprint arXiv:2501.00312* (2024).
43. Li, J. et al.: Disentangle and Remerge: Interventional Knowledge Distillation for Few-Shot Object Detection from a Conditional Causal Perspective. In: Williams, B. et al. (eds.) *Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI 2023)*. AAAI Press, pp. 1323–1333 (2023).
44. Hendrycks, D. et al.: AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty. In: *8th International Conference on Learning Representations (ICLR)*. OpenReview.net (2020).
45. Sun, C., Qiang, W., Li, J.: Demo:scdr1: Scalable and customized distributed reinforcement learning system. In: *44th IEEE International Conference on Distributed Computing Systems, ICDCS 2024*, pp. 1404–1407. IEEE, Jersey City, NJ, USA (2024a)
46. Sun, C. et al.: Revisiting Communication Efficiency in Multi-Agent Reinforcement Learning from the Dimensional Analysis Perspective. *CoRR abs/2501.02888* (2025).