Crafting Better Contrastive Views for Siamese Representation Learning

Xiangyu Peng[1*] Kai Wang[1*] Zheng Zhu[2] Mang Wang[3] Yang You[1†]
[1]National University of Singapore [2]Tsinghua University [3]Alibaba Group
{xiangyupeng, kai.wang, youy}@comp.nus.edu.sg
zhengzhu@ieee.org wangmang.wm@alibaba-inc.com
Code: https://github.com/xyupeng/ContrastiveCrop

近年来，基于自监督对比学习的方法因采用孪生结构（Siamese Structure）而显著获益，该结构旨在最小化正样本对之间的距离。为了实现高性能的孪生表征学习，关键在于设计优质的对比样本对。然而，大多数现有工作仅通过随机采样生成同一图像的不同裁剪，这忽略了语义信息，从而可能降低视图的质量。为此，我们提出了一种名为 *ContrastiveCrop* 的新方法，可以有效地生成更优的裁剪以提升孪生表征学习性能。

首先，我们在完全无监督的训练过程中提出了一种语义感知的目标定位策略，指导生成能够避免伪阳性（例如对象与背景）的对比视图。此外，我们通过实验发现，外观相似的视图对于孪生模型的训练影响有限。因此，我们进一步设计了一种中心抑制采样机制，以增大裁剪的多样性。值得注意的是，该方法在对比学习中对正样本对进行了深入考虑，同时几乎不增加额外的训练开销。作为一种即插即用且与框架无关的模块，*ContrastiveCrop* 在 CIFAR-10、CIFAR-100、Tiny ImageNet 和 STL-10 数据集上的分类精度提升了 0.4

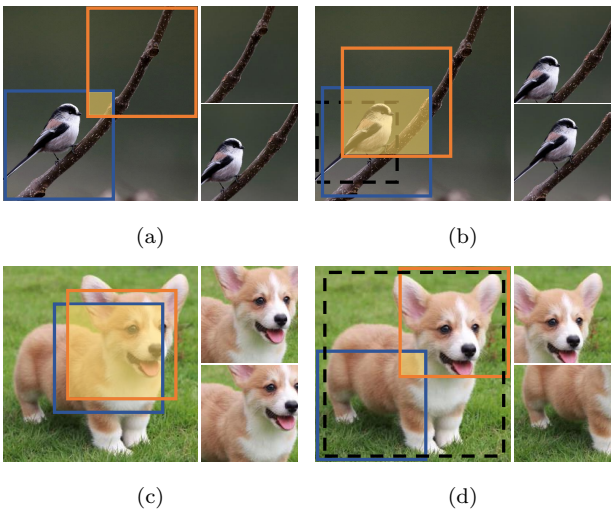


图 1. 我们提出 *ContrastiveCrop* 方法的动机。(a) 和 (c) 是通过典型的 *RandomCrop* 生成的，而 (b) 和 (d) 则是我们方法生成的裁剪视图。(a) 中展示了目标与背景的伪阳性问题，我们通过目标定位并将裁剪中心限制在黑色虚线框的边界内（如 (b) 所示）来解决。此外，通过在 (d) 中使裁剪远离中心，增大了裁剪的差异性，避免了 (c) 中外观相近的问题。

1. 引言

自监督学习（SSL）因其在利用大规模未标注数据方面的潜力，在计算机视觉领域引起了广泛关注。作为 SSL 的主流方法之一，对比学习在多个下游任务（如目标检测、分割和姿态估计 [16,18,21,27,32,50]）上表现出超过有监督方法的优越性能。这些令人鼓舞的结果在很大程度上得益于孪生结构（Siamese Structure）的使用。这种结构被广泛应用于当前最先进的无监督方法中，包括 SimCLR [5]、MoCo V1 V2 [7,20]、BYOL [17] 和 SimSiam [8]。通常情况下，孪生结构以同一图像的两个增强视图作为输入，并最小化它们在嵌入空间中的距离。在选择适当视图的情况下，孪生网络表现出强大的泛化视觉特征学习能力 [37]。

*Equal contribution.
†Corresponding author.

对比学习的核心问题之一是如何设计正样本的选择。一些研究通过强数据增强（如颜色失真和拼图变换 [4, 37]）生成不同的正样本视图。另有研究 [34] 采用无监督混合方法 [48, 49] 来生成包含多个样本的正样本对。此外，[51] 从特征层面进行变换，创建难以区分的正样本。尽管采用了不同的技术，这些方法通常通过 *RandomCrop* 对图像采样生成多样化的视图。

作为一种基础采样方法，*RandomCrop* 对所有裁剪区域均匀采样。然而，它忽略了配对视图的语义信息，而语义信息对于更高效和准确地学习表征至关重要。如图 1a所示，在没有目标（如尺度和位置）先验的情况下，随机裁剪容易错过目标。优化目标与背景在嵌入空间中的距离可能会误导表征学习。此外，图 1c表明随机裁剪无法总是提供目标的充分变化。这些高相似性的视图对判别模型的学习贡献有限。

为解决上述问题，我们提出了 *ContrastiveCrop*，旨在为孪生表征学习生成更优的对比样本对。伪阳性问题表明一种更优的对比学习采样策略需要考虑图像的内容信息。因此，我们提出了一种语义感知的定位方案，用于指导裁剪选择，避免大多数伪阳性，如图 1b所示。此外，我们提出了一种中心抑制采样策略，减少高相似性的正样本对，如图 1d中展示了更具多样性的裁剪。语义感知定位和中心抑制采样策略能够优雅地结合，为对比学习生成更优的裁剪视图。

ContrastiveCrop 在配对时同时考虑了语义信息和保持较大变化性。作为一种即插即用的方法，它可以轻松地应用于孪生结构。更重要的是，无论是否使用负样本，该方法对对比学习框架具有通用性。该策略在训练开销几乎可以忽略的情况下，持续提升了 SimCLR、MoCo、BYOL 和 SimSiam 在 CIFAR-10、CIFAR-100、Tiny ImageNet 和 STL-10 数据集上的分类准确率提升 0.4

本文的主要贡献总结如下：

- 据我们所知，这是首次研究对比学习中常用的 *RandomCrop* 方法的问题。我们提出了 *ContrastiveCrop*，专门为此任务生成更优的视图。
- 在 *ContrastiveCrop* 中，我们采用语义感知定位以避免大多数伪阳性，并设计了中心抑制采样策略以减少高相似性的正样本对。
- 在多种数据集上，*ContrastiveCrop* 在多种流行对比学习方法中均显著优于 *RandomCrop*，显示了其在孪生表征学习中的有效性和通用性。

2. 相关工作

在本节中，我们介绍与本文相关的对比学习和正样本选择的研究工作。

2.1. 对比学习

对比学习的核心思想是在嵌入空间中拉近正样本对的距离，同时拉远负样本对的距离。这种方法在无标注的视觉表征学习中展现出巨大的潜力 [2, 23, 29, 30, 36, 43, 47]。最近，基于孪生结构的对比学习方法在下游任务中取得了显著性能提升 [5, 7, 8, 15, 17, 20, 40, 45, 46]，其中一些方法甚至超越了监督模型。

里程碑式的成果——SimCLR [5] 提出了一个简单的对比视觉表征学习框架。通过引入非线性变换头，该方法显著提升了表征学习的质量。另一项著名的工作是 MoCo [20]，其核心创新在于引入内存库机制，用动量平滑更新大量负样本以增强一致性。此外，也有不依赖负样本的表征学习方法被提出，例如 BYOL [17] 通过训练在线网络预测目标网络的输出，其中目标网络通过动量缓慢更新。作者假设，在线网络附加的投影头以及动量编码器在避免无负样本情况下的模型坍缩中起到了重要作用。SimSiam [8] 进一步探索了简单的孪生网络，表明无需负样本对、大批量训练和动量编码器，网络也可以学习有意义的表征，并强调了停止梯度在防止坍缩中的作用。除此之外，还有理论分析和实证研究被提出，以更好地理解对比学习的行为和特性 [1, 3, 6, 9, 24, 31, 35, 39, 39, 41, 44, 53]。

2.2. 正样本选择

对比学习中的关键问题之一是正样本选择的设计。生成正样本对的一种直观方法是通过数据增强对样本进行不同视图的生成。大多数自监督学习的工作采用直接从监督学习中移植的增强管道 [12, 13, 19, 26, 48, 49]。Chen 等人 [5] 对一系列数据变换的效果进行了全面研究，发现随机裁剪和随机颜色扰动的组合可以带来更好的性能提升。Tian 等人 [37] 提出了 *InfoMin 原则*，通过捕获视图之间的互信息甜点来生成正样本对，并基于此设计了 *InfoMin Augmentation*。

与本篇最接近的成果是 [33]，该工作同样利用无监

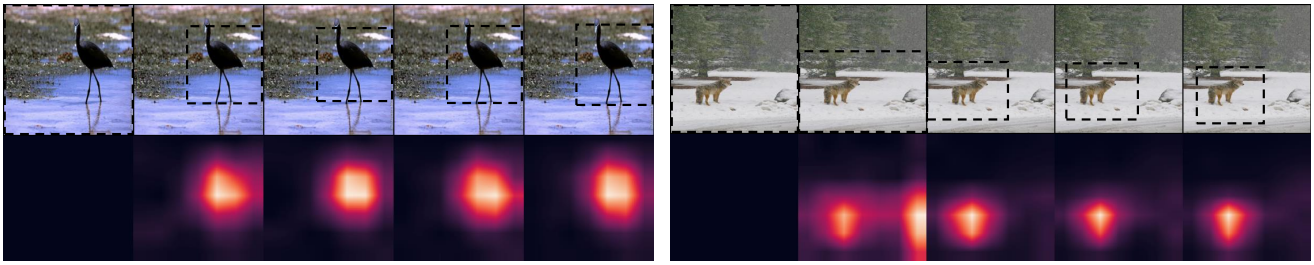


图 2. 每个子图从左到右展示了定位的训练动态。我们将定位框初始化为整个图像，并以固定间隔使用最新的热图更新。请注意，我们的目标不是获得精确的定位，而是通过找到感兴趣的目标来指导裁剪的生成。

督显著性图作为裁剪约束，但裁剪仍是随机采样的。以上所有工作通常将 *RandomCrop* 用作生成输入视图的基本采样方法，但我们发现它可能不是对比学习的最优解。另有研究 [28] 在裁剪时考虑了目标-场景关系，但需要额外的目标提案算法。本研究提出了 *ContrastiveCrop*，一种专门为对比学习设计的正样本生成方法，无需额外的外部算法支持即可生成更优的正样本视图

3. 方法

在本节中，我们介绍了用于孪生表征学习的 *ContrastiveCrop* 方法。首先，我们简要回顾了 *RandomCrop* 作为基础知识。然后，我们描述了语义感知定位和中心抑制采样，作为我们 *ContrastiveCrop* 方法的两个子模块。最后，我们进一步讨论了该方法的有利特性，以便更好地理解其优势。

3.1. 基础知识

RandomCrop 是一种高效的数据增强方法，广泛应用于监督学习和自监督学习（SSL）。在此简要复现该技术，并以 Pytorch[1] 中的 API 为例进行说明。给定图像 I，我们首先从预定义的范围内（例如，$s \in [0.2, 1.0]$ 和 $r \in [3/4, 4/3]$）确定裁剪的比例 s 和长宽比 r。然后，根据 s 和 r 可以得到裁剪区域的高度和宽度。接下来，裁剪的位置会在图像平面上随机选择，条件是裁剪区域完全位于图像内。*RandomCrop* 的过程可以表示为：

$$(x, y, h, w) = \mathbb{R}crop(s, r, I), \tag{1}$$

其中，$\mathbb{R}crop(\cdot, \cdot, \cdot)$ 是一个随机采样函数，返回表示裁剪区域的四元组 (x, y, h, w)。将 I 表示为输入图像，(x, y) 表示裁剪中心的坐标，(h, w) 表示裁剪区域的高度和宽度。通常，裁剪的比例 s 和长宽比 r 是灵活设置的，从而能够生成不同尺寸的裁剪区域。

原则上，*RandomCrop* 可以选择所有个体裁剪区域，因此能够提供样本的多样视图。然而，它是等概率采样的（即每个单独的视图被采样的概率相同），这忽略了图像的语义信息。如图 1a所示，*RandomCrop* 在物体尺度较小时容易生成假阳性样本。对于对比学习中具有不同尺度的物体，*RandomCrop* 由于忽略了语义信息，必然会生成假阳性样本。因此，优化图 3中的假阳性样本可能会误导表征学习。因此，为裁剪设计一种语义感知的采样策略，对于孪生表征学习至关重要。

3.2. 语义感知定位

为了解决 *RandomCrop* 在内容理解上的不足，我们设计了一个语义感知定位模块，能够有效地减少假阳性样本情况的发生，并且以无监督的方式进行训练。为了更好地研究孪生网络中特征学习的过程，我们在图 2中可视化了在不同训练阶段（例如，第 0、20、40、60、80 轮）生成的热力图。值得注意的是，通过对最后一个卷积层特征在通道维度上求和并将其归一化到 [0, 1] 来生成热力图。可视化表征出几个要点：1）孪生表征学习框架能够捕捉物体的位置，这可以用来引导生成更好的裁剪区域；2）热力图可以大致指示物体的位置，但在早期阶段可能需要一些预训练。

基于上述分析，我们提出在训练过程中使用热力图中的信息来定位物体。具体来说，*RandomCrop* 在训练的早期阶段用于收集整个图像的语义信息。然后，我们应用一个指示函数从热力图中获得物体的边界框 B，

[1] https://pytorch.org/vision/stable/transforms.html

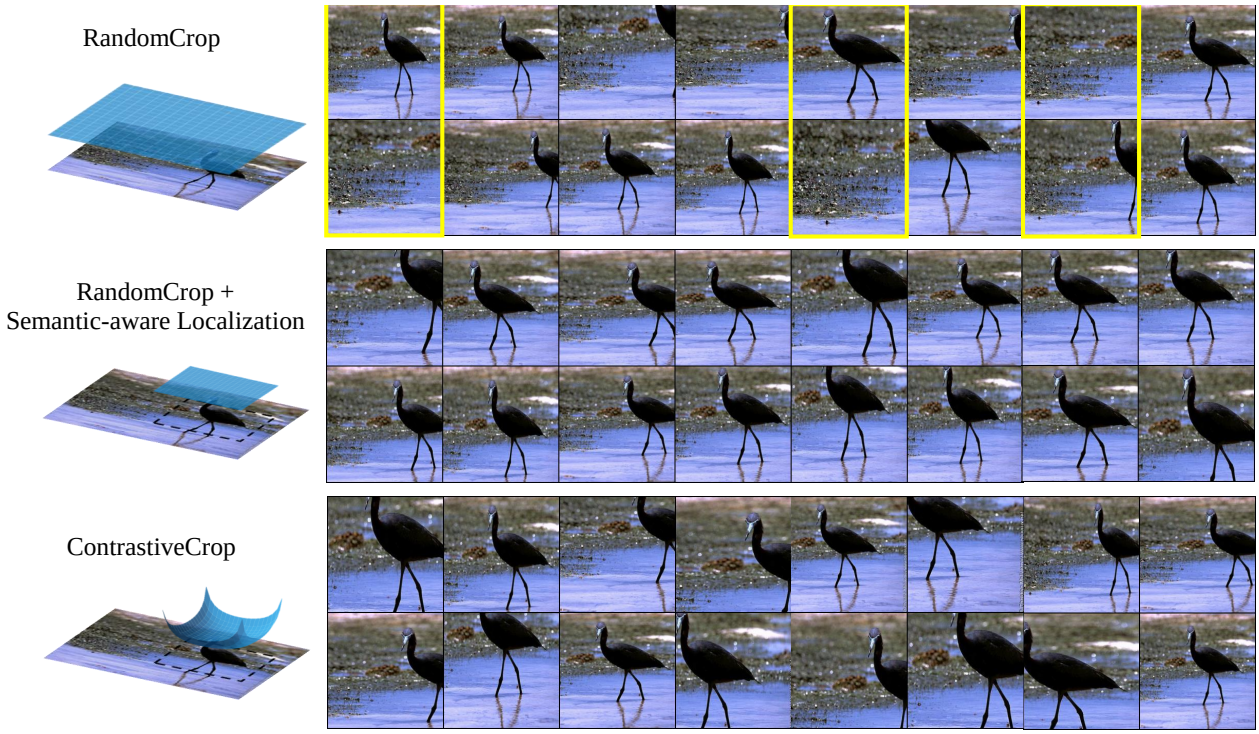


图 3. *RandomCrop*、*RandomCrop* + 语义感知定位和我们的 *ContrastiveCrop* 的可视化。我们在左侧展示了三种设置的采样分布和可操作区域，并在右侧展示了相应的采样对。使用 *RandomCrop* 生成的配对包括一些完全错过目标的假阳性（用黄色框标出）。使用 *RandomCrop* 配合 语义感知定位可以减少假阳性对，但会引入容易的正样本对，这些对之间相似性较大。最后，我们的 *ContrastiveCrop* 方法既能减少假阳性配对，又能同时增加样本的多样性。

其表达式为：

$$B = L(\mathbb{1}[M > k]), \tag{2}$$

其中，M 表示热力图，$k \in [0, 1]$ 是激活的阈值，$\mathbb{1}$ 是指示函数，L 计算激活位置的矩形闭包。获得边界框 B 后，可以生成语义裁剪区域，表达式如下：

$$(\dot{x}, \dot{y}, \dot{h}, \dot{w}) = \mathbb{R}crop(s, r, B), \tag{3}$$

其中，$\dot{x}$、$\dot{y}$、$\dot{h}$、$\dot{w}$、s、r 和 $\mathbb{R}crop$ 的定义与式 1 相似。考虑到可能存在较为粗糙的定位结果，我们通过仅限制裁剪中心位于 B 内来扩大可操作区域。这也减少了训练和推理阶段分辨率不一致产生的潜在负面影响 [38]。

在训练阶段，边界框会定期更新，从而利用模型最新学习到的特征。值得注意的是，我们的目标不会得到精确的定位，而是通过找到较相关的物体来引导裁剪区域的生成。边界框的尺度由阈值参数 $k \in [0, 1]$ 控制。一般来说，较大的 k 会导致较小的边界框，从而限制生成裁剪区域的多样性。然而，较小的 k 可能仍然包含许多不相关的背景纹理，无法有效找到物体。我们在第 4.4节研究了不同阈值 k 的影响。我们通过实验发现，所提出的定位模块对该参数不敏感，并且在较广泛的 k 范围内能够显著提升性能。

最后，我们展示了语义感知定位的采样效果（见图 3）。与 *RandomCrop* 相比，可以发现应用了上述模块后，假阳性配对大幅减少。这表明，未标注数据训练的自监督神经网络能够识别感兴趣的物体及其位置。因此，生成视图时不再需要额外的区域提示或真实边界框 [10, 52]。

3.3. 中心抑制采样

语义感知定位方案为减少假阳性配对提供了有效的指导，但由于可操作区域较小，增加了相似图像对出现的概率。为了解决这个问题，本小节介绍了中心抑制采样方法。

其主要思路是通过分散采样位置，减少样本聚集在中心区域的概率。具体而言，我们采用参数相同的贝塔分布 $\beta(\alpha,\alpha)$，该分布呈对称性。通过调整参数 α，我们可以灵活控制分布的形状。由于目标是增大采样区域的方差，我们设置 $\alpha<1$，从而得到一个 U 形分布（即靠近中心的概率较低，远离中心的位置概率较高）。这样，采样的区域更可能分布到可操作区域的边缘，避免了过多重叠的情况。

将中心抑制采样与语义感知定位结合后，我们可以最终将 *ContrastiveCrop* 表示为：

$$(\dot{x},\dot{y},\dot{h},\dot{w}) = \mathbb{C}crop(s,r,B), \tag{4}$$

其中 $\mathbb{C}crop$ 表示应用中心抑制分布的采样函数，B 是与公式 3 中相同的边界框。需要注意的是，贝塔分布的形状由参数 α 决定，并影响采样区域的方差。在第 4.4节中，我们研究了不同 α 值的影响，包括 $\alpha>1$ 时形成的倒 U 形分布。

Algorithm 1 *ContrastiveCrop* 用于 Siamese 表示学习

输入： 图像 I，裁剪尺度 s，裁剪比例 r，激活阈值 k，贝塔分布参数 α。

$h = \sqrt{s \cdot r}$ ▷ 裁剪区域的高度

$w = \sqrt{s/r}$ ▷ 裁剪区域的宽度

$F = \text{Forward}(I)$ ▷ 最后一层的特征

$M = \text{Normalize}(F)$ ▷ 标准化后的热图

$B = L(\mathbb{1}[M > k])$ ▷ 根据公式 2计算边界框

$x = B_{x0} + (B_{x1} - B_{x0}) \cdot u, u \sim \beta(\alpha,\alpha)$

$y = B_{y0} + (B_{y1} - B_{y0}) \cdot v, v \sim \beta(\alpha,\alpha)$

▷ 从 β 分布中采样裁剪区域的中心 x 和 y

输出： 裁剪区域 $C = (x,y,h,w)$

我们在图 3中可视化了 *ContrastiveCrop* 的效果。与 *RandomCrop* 相比，我们的方法通过语义感知定位显著减少了假阳性对。同时，通过应用中心抑制分布，它增加了正样本对之间的方差。算法 1展示了 *ContrastiveCrop* 的管道。整个模块与其他变换无关，并且可以轻松集成到通用的对比学习框架中。

3.4. 讨论

为了更好地理解 *ContrastiveCrop* 的行为，我们讨论了几个可能增强其有效性的属性。我们首先研究了语义信息与正样本相似度之间的关系。我们将裁剪区域的类别分数作为表征分类语义信息丰富性的指标。正样本对的相似度则在隐层空间中计算，作为正样本表征之间的余弦相似度。类别分数和相似度都是通过对标准 ResNet-50 [22] 模型进行大量裁剪实验得出的平均结果，且该模型是基于 ImageNet [14] 标签进行训练的。它们之间的关系如图 4所示。从中可以发现，

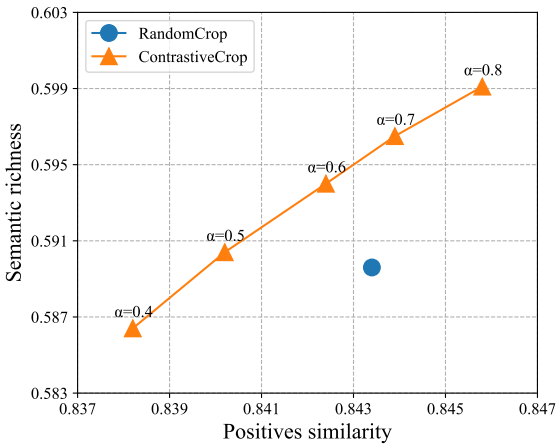


图 4. 语义丰富度与正样本相似度的关系。图中的点通过变化 α（固定 $k=0.1$，见第 4.2节）得到，每个点的得分是通过大量裁剪试验的平均结果计算得出的。与 *RandomCrop* 相比，我们的 *ContrastiveCrop* 在相同相似度水平下（纵向）传递了更多的语义信息，并且在相同语义信息下（横向）产生了更少相似的正样本。

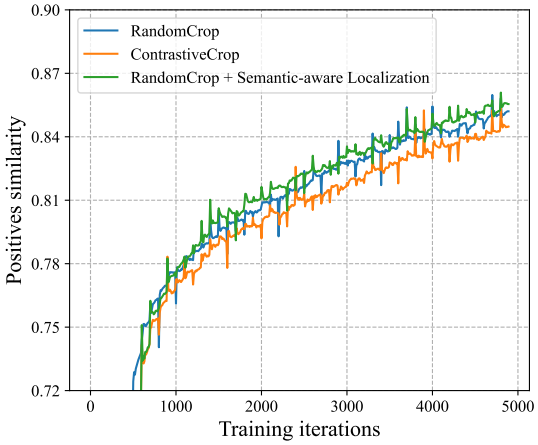


图 5. 训练中的正样本相似度。较小的正样本相似度表示更困难的正样本，这有助于增强表示学习 [51]。与 *RandomCrop* 基线相比，仅加入定位会导致相似度略有增大。我们的 *ContrastiveCrop* 结合了语义感知定位和中心抑制采样，有效降低了正样本的相似度。

方法	CIFAR-10		CIFAR-100		Tiny ImageNet		STL-10	
	R-Crop	*C-Crop*	*R-Crop*	*C-Crop*	*R-Crop*	*C-Crop*	*R-Crop*	*C-Crop*
SimCLR [5]	89.63	**90.08**	60.30	**61.91**	45.19	**46.21**	88.95	**89.53**
MoCo [20]	86.73	**88.78**	56.10	**57.65**	47.09	**47.98**	89.17	**89.81**
BYOL [17]	91.96	**92.54**	63.75	**64.62**	46.08	**47.23**	91.84	**92.42**
SimSiam [8]	90.96	**91.48**	64.79	**65.82**	43.03	**44.54**	89.39	**89.83**

表 1. 不同对比方法和数据集的线性分类结果。*R-Crop* 和 *C-Crop* 分别表示 *RandomCrop* 和 *ContrastiveCrop*。我们采用 ResNet-18 作为基础模型，并使用统一的训练设置复现所有方法，具体设置参见第4.2节。

ContrastiveCrop 在相同方差水平下，比 *RandomCrop* 传递了更多的语义信息，这显示了语义感知定位的有效性。此外，在相等的语义信息下，*ContrastiveCrop* 比 *RandomCrop* 具有更大的方差，这得益于中心抑制采样。

我们进一步在图 5中可视化了训练过程中正样本对的相似度。如图所示，单独向 *RandomCrop* 添加语义感知定位会略微增加相似度，因为定位限制了裁剪区域在较小的操作区域内。而我们的 *ContrastiveCrop* 进一步结合了中心抑制采样，显示出比其他两者更小的正样本相似度。这表明 *ContrastiveCrop* 所采样的正样本对更为复杂，这有助于学习更具视图不变性的特征，正如 FT [51] 中所建议的那样。然而，与 FT 通过减少特征空间中的正样本相似度不同，我们是直接从原始数据中采样更困难的裁剪区域，同时仔细考虑了语义信息。

4. 实验

在本节中，我们使用主流对比学习方法对多个数据集进行广泛实验，以展示我们方法的有效性和通用性。Sec. 4.1 介绍了数据集和对比方法，Sec. 4.2 描述了实现细节。在 Sec. 4.3 中，我们通过 CLEP 评估方法性能。Sec. 4.4 展示了消融实验的结果，最后在 Sec. 4.5 中展示了下游目标检测与分割任务的迁移性能。

4.1. 数据集与基线方法

我们在多个数据集上评估了当前最先进的无监督对比学习方法。这些数据集包括 **CIFAR-10/CIFAR-100** [25]、**Tiny ImageNet**、**STL-10** [11] 和 **ImageNet** [14]。这些数据集通常用于目标识别，图像中包含对象的典型视图。基线对比学习方法包括 SimCLR [5]、MoCo V1 & V2 [7, 20]、BYOL [17] 和 SimSiam [8]。

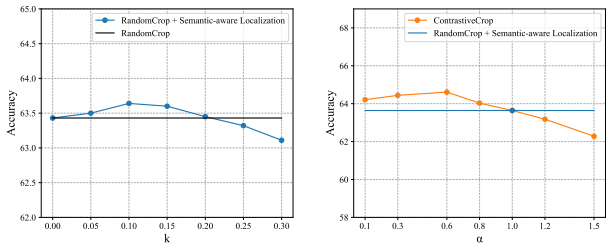
4.2. 实现细节

我们的 *ContrastiveCrop* 方法旨在为对比学习生成更好的视图，对自监督学习框架及其相关训练组件（如骨干网络、损失函数和优化器等）无特定依赖。因此，我们在对比实验中保持相同的训练设置。在进一步调整超参数的情况下，预计可以获得更大收益，但此并非本篇重点。

对于小型数据集（例如 CIFAR-10/100、Tiny ImageNet 和 STL-10），我们在 所有实验中使用相同的训练设置。在预训练阶段，我们使用 ResNet-18 [22] 以批量大小 512 和余弦退火学习率 0.5 训练 500 个 epoch。线性分类器以初始学习率 10.0 训练 100 个 epoch，并在第 60 和 80 个 epoch 时衰减为原来的 0.1 倍。

在 ImageNet 实验中，我们采用 ResNet-50 作为基础模型。MoCo V1、MoCo V2 和 SimSiam 的预训练设置完全遵循其原始论文。我们重新实现了 SimCLR，使用批量大小 512 和余弦退火学习率 0.05。所有基线方法的线性分类器训练设置与 [20] 相同。

对于我们的方法，激活阈值 k 设置为 0.1，采样参数 α 设置为 0.6。定位框以 20% 的频率更新（即，总共更新 4 次，不包括最后一个 epoch），对训练开销的影响可以忽略不计；如 Sec. 3.2 所述，在首次更新前使用 *RandomCrop* 收集全局信息。所有实验均在一个 8-GPU 服务器上进行。我们使用 SGD 优化器，动量为 0.9，权重衰减在预训练和线性评估阶段分别为 10^{-4} 和 0。



(a) 准确率 *v.s.* k　　(b) 准确率 *v.s.* α

图 6. 关于 k 和 α 的 IN-200 消融实验结果。子图 (a) 比较了 *RandomCrop* 基线（黑线）与 *RandomCrop* + 语义感知定位（蓝线）。在子图 (b) 中，我们固定 $k = 0.1$（蓝线），对比研究 *ContrastiveCrop* 在不同 α 下的表现。

4.3. 线性分类

在本节中，我们使用常见的流程，通过线性分类验证了我们的方法。我们冻结编码器的预训练权重，并在其基础上训练一个监督线性分类器。实验结果报告了验证集上的 Top-1 分类准确率。

CIFAR-10/100、Tiny ImageNet 和 STL-10 上的结果。 这些小型数据集上的实验结果如表 1 所示。在所有实验中使用相同的训练设置，*ContrastiveCrop* 始终至少提升基线方法 0.4%。结果表明，该方法具有通用性，并且不需要繁重的参数调优。定位框在训练过程中以 20% 的频率更新（即，总共更新 4 次，不包括最后一个 epoch），训练开销可以忽略不计。

ImageNet 上的结果。 ImageNet 的实验结果分为两部分：1）标准的 ImageNet-1K（IN-1K），用于预训练；2）IN-200，包含 IN-1K 中随机挑选的 200 个类别，用于消融实验。如表 2 所示，使用 SimCLR、MoCo V1、MoCo V2 和 SimSiam 时，我们的方法在 IN-1K 上分别比 *RandomCrop* 提升了 0.25%、1.09%、0.49% 和 0.33%。在 IN-200 上可以看到更大的提升。基线方法的一致性提升表明 *ContrastiveCrop* 对于对比学习方法的有效性和通用性。

4.4. 消融实验

在消融实验中，我们分别研究了语义感知定位模块和中心抑制采样的作用。同时，我们也研究了 *ContrastiveCrop* 与不同变换组合时的效果。实验使用 MoCo V2 和 ResNet-50，并报告了 IN-200 上的线性分类结果。

方法	网络结构	训练周期	IN-200 Top-1	IN-1K Top-1
SimCLR	R50	100	62.14	61.60
SimCLR+CC	R50	100	**63.08**	**61.85**
MoCo V1	R50	100	64.52	57.25
MoCo V1+CC	R50	100	**65.80**	**58.34**
MoCo V2	R50	100	63.43	64.40
MoCo V2+CC	R50	100	**64.61**	**64.89**
SimSiam	R50	100	62.89	65.62
SimSiam+CC	R50	100	**63.54**	**65.95**

表 2. 比较 *RandomCrop* 和我们提出的 *ContrastiveCrop* 方法 (表中为 CC) 在 IN-200 和 IN-1K 上的线性分类结果。所有模型在相同的训练设置下进行预训练 100 批次，以确保公平比较。

语义感知定位。 在我们的方法中，无监督语义感知定位作为生成裁剪的指导。我们研究了 k 对定位框大小的影响，其中较大的 k 会导致较小的框。同时，我们与未使用定位框的 *RandomCrop* 进行了比较（即 $k = 0$）。实验结果如图 6a 所示。从图中可以发现，在 k 的范围为 0.05 到 0.2 时，使用定位框比 *RandomCrop* 基线（黑线）表现更好。这表明在很大程度上去除假阳性是有效的。然而，当 k 超过 0.25 时，性能开始快速下降。我们推测原因是较小的边界框大幅减少了视图的多样性，从而使得学习判别性特征变得困难。

更新频率	0%	10%	20%	30%	50%
准确率 (%)	63.43	64.40	64.61	64.40	64.11

表 3. 线性分类准确率与不同定位框更新频率的关系。0% 表示没有更新（即 *RandomCrop* 基线），50% 表示在训练的中期进行一次更新。*RandomCrop* 在第一次更新之前应用。

我们还研究了定位框更新频率的影响，如表 3 所示。结果表明，即使在训练中间仅更新一次（即 50%），也可以超过 *RandomCrop* 基线（即 0%）。在 10% ∼ 30% 的范围内，随着更新次数的增加，性能提升更显著。这些结果表明我们的方法在不同更新频率下均表现良好。

预训练模型	IN-1K	VOC 检测			COCO 实例分割			COCO 检测		
	Top-1	AP	AP_{50}	AP_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
随机初始化	-	33.8	60.2	33.1	29.3	46.9	30.8	26.4	44.0	27.8
监督学习	76.1	53.5	81.3	58.8	33.3	54.7	35.2	38.2	58.2	41.2
InfoMin [37]	70.1	57.6	82.7	64.6	34.1	55.2	36.3	39.0	58.5	42.0
MoCoV1 [20]	60.6	55.9	81.5	62.6	33.6	54.8	35.6	38.5	58.3	41.6
MoCoV1 + *ContrastiveCrop*	**63.0**	**56.1**	**81.7**	**63.0**	**33.9**	**55.2**	**36.1**	**38.8**	**58.5**	**41.9**
MoCoV2 [7]	67.5	57.0	82.4	63.6	34.2	55.4	36.2	39.0	58.6	41.9
MoCoV2 + *ContrastiveCrop*	**67.8**	**57.3**	**82.5**	**63.8**	**34.5**	**55.5**	**36.4**	**39.2**	**58.8**	**42.2**

表 4. 在 PASCAL VOC 检测和 COCO 检测与实例分割任务上的微调结果。所有模型在 ImageNet-1K 上预训练 200 个 epoch。在 VOC 上，训练和评估集分别为 `trainval2007+2012` 和 `test2007`，在 COCO 上分别为 `train2017` 和 `val2017`。所有模型在 VOC 上微调 24K 次迭代，在 COCO 上微调 90K 次。

中心抑制采样 在本工作中，我们使用 β 分布进行中心抑制采样，从而通过不同的 α 控制其方差。我们通过迭代多个 α 研究不同方差的影响。结果如图 6b 所示，其中 $k = 0.1$ 用于定位。可以发现，当 $\alpha < 1$ 时，我们的 *ContrastiveCrop* 始终优于用于定位的 *RandomCrop*，表明中心抑制采样的有效性。我们还研究了 $\alpha > 1$ 的情况，其方差小于均匀分布（即 $\alpha = 1$）。当 $\alpha > 1$ 时，准确率出现下降。这表明更大的裁剪方差有助于更好的对比。

翻转	色调抖动 + 灰度化	模糊	*R-Crop*	*C-Crop*
✓	✓	✓	63.4	**64.6**
✓			50.4	**50.9**
	✓		60.6	**61.4**
		✓	44.9	**45.2**
			45.5	**45.9**

表 5. MoCo V2 中使用的其他变换的消融实验。我们将 *ColorJitter* 和 *Grayscale* 作为单一的颜色变换。*R-Crop* 和 *C-Crop* 分别表示 *RandomCrop* 和 *ContrastiveCrop*。这些结果基于在 IN-200 上训练 100 轮的 ResNet-50。

***ContrastiveCrop* 与其他变换的组合。** 为了进一步比较 *ContrastiveCrop* 和 *RandomCrop* 的效果，我们研究了它们与其他图像变换的组合。这里，我们选择了 MoCo V2 [7] 中使用的变换，包括 *Flip*、*ColorJitter*、*Grayscale* 和 *Blur*。消融实验结果如表 5 所示。在移除所有其他变换的情况下，*ContrastiveCrop* 比 *RandomCrop* 高出 0.4%，这直接证明了其优越性。此外，仅添加一种额外的变换时，*ContrastiveCrop* 也比 *RandomCrop* 提升了 0.3% ∼ 0.8%。当包含所有变换时，取得了最大的 1.2% 提升，这表明通过进一步的颜色变换可以更大程度地挖掘 *ContrastiveCrop* 的潜力。此外，这些结果表明 *ContrastiveCrop* 具有与其他变换兼容且正交的特性。

4.5. 下游任务

在本节中，我们通过目标检测和实例分割任务衡量方法的可迁移性。参考先前的工作 [20, 51]，我们在 IN-1K 上预训练 ResNet-50 200 个 epoch。对于下游任务，我们使用 PASCAL VOC [16] 和 COCO [27] 作为基准，并采用 MoCo 的 detectron2 代码库中的相同设置 [42]。预训练模型的所有层在目标数据集上端到端微调。

PASCAL VOC 目标检测。 参考 [20]，我们使用带有 R50-C4 [21] 主干的 Faster R-CNN [32] 作为检测器。在 `trainval2007+2012` 集上微调模型，并在 VOC `test2007` 上进行评估。结果如表 4 所示。与 MoCo V1 基线相比，我们的方法在 AP、AP_{50} 和 AP_{75} 上分别

实现了 +0.2、+0.2 和 +0.4 的提升。

COCO 目标检测/实例分割。 对于目标检测和实例分割，我们使用带有 R50-C4 主干的 Mask R-CNN [21] 作为模型。在 `train2017` 集上训练 90K 次迭代，并在 `val2017` 集上进行评估。如表 4 所示，所提出的 *ContrastiveCrop* 在所有指标上均表现出色。

5. 讨论与结论

在本文中，我们提出了 *ContrastiveCrop* 方法，该方法专为改进对比式表示学习中的视图生成而设计。*ContrastiveCrop* 采用语义感知的定位方式，避免了大多数错误的正样本，并引入中心抑制采样以减少冗余的正样本对。我们在对样本进行变换时创新性地引入了语义信息，并全面研究了适合对比学习的变异范围。通过与 SimCLR、MoCo、BYOL 和 SimSiam 等多种最先进的对比学习方法的广泛实验，我们证明了该方法的有效性和通用性。最后，由于正样本设计在对比学习中的重要作用，我们希望本研究能为未来正样本设计的研究提供启发。

6. 致谢

本研究得到了新加坡国家研究基金会 AI Singapore 项目的支持（AISG 奖项编号：AISG2-PhD-2021-08-008）。我们感谢 Google TFRC 提供云 TPU 的访问支持；感谢瑞士国家超级计算中心（CSCS）提供对 Piz Daint 超级计算机的访问支持；感谢德州高级计算中心（TACC）提供对 Longhorn 和 Frontera 超级计算机的访问支持；感谢卢森堡国家超级计算组织 LuxProvide 提供对 MeluXina 超级计算机的访问支持。

参考文献

[1] Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019. 2

[2] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NeurIPS*, 2019. 2

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021. 2

[4] Pengguang Chen, Shu Liu, and Jiaya Jia. Jigsaw clustering for unsupervised visual representation learning. In *CVPR*, 2021. 2

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 1, 2, 6

[6] Ting Chen, Calvin Luo, and Lala Li. Intriguing properties of contrastive losses. *arXiv preprint arXiv:2011.02803*, 2020. 2

[7] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 1, 2, 6, 8

[8] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *CVPR*, 2021. 1, 2, 6

[9] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*, 2021. 2

[10] Ming-Ming Cheng, Ziming Zhang, Wen-Yan Lin, and Philip Torr. Bing: Binarized normed gradients for objectness estimation at 300fps. In *CVPR*, 2014. 4

[11] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *AISTATS*, 2011. 6

[12] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018. 2

[13] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPR*, 2020. 2

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5, 6

[15] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. *arXiv preprint arXiv:2104.14548*, 2021. 2

[16] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 1, 8

[17] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 1, 2, 6

[18] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018. 1

[19] Ryuichiro Hataya, Jan Zdenek, Kazuki Yoshizoe, and Hideki Nakayama. Faster autoaugment: Learning augmentation strategies using backpropagation. In *ECCV*, 2020. 2

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 1, 2, 6, 8

[21] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 1, 8, 9

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5, 6

[23] Olivier J Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*, 2019. 2

[24] Tianyu Hua, Wenxiao Wang, Zihui Xue, Sucheng Ren, Yue Wang, and Hang Zhao. On feature decorrelation in self-supervised learning. In *ICCV*, 2021. 2

[25] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 6

[26] Sungbin Lim, Ildoo Kim, Taesup Kim, Chiheon Kim, and Sungwoong Kim. Fast autoaugment. *NeurIPS*, 2019. 2

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1, 8

[28] Shlok Mishra, Anshul Shah, Ankan Bansal, Abhyuday Jagannatha, Abhishek Sharma, David Jacobs, and Dilip Krishnan. Object-aware cropping for self-supervised learning. *arXiv preprint arXiv:2112.00319*, 2021. 3

[29] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020. 2

[30] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2

[31] Senthil Purushwalkam and Abhinav Gupta. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *arXiv preprint arXiv:2007.13916*, 2020. 2

[32] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 1, 8

[33] Ramprasaath R. Selvaraju, Karan Desai, Justin Johnson, and Nikhil Naik. Casting your model: Learning to localize improves self-supervised representations. In *CVPR*, pages 11058–11067, June 2021. 2

[34] Zhiqiang Shen, Zechun Liu, Zhuang Liu, Marios Savvides, Trevor Darrell, and Eric Xing. Un-mix: Rethinking image mixtures for unsupervised visual representation learning. *arXiv preprint arXiv:2003.05438*, 2020. 2

[35] Yuandong Tian, Xinlei Chen, and Surya Ganguli. Understanding self-supervised learning dynamics without contrastive pairs. *arXiv preprint arXiv:2102.06810*, 2021. 2

[36] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *ECCV*, 2019. 2

[37] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *arXiv preprint arXiv:2005.10243*, 2020. 1, 2, 8

[38] Hugo Touvron, Andrea Vedaldi, Matthijs Douze, and Hervé Jégou. Fixing the train-test resolution discrepancy. *arXiv preprint arXiv:1906.06423*, 2019. 4

[39] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, 2020. 2

[40] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. *CVPR*, 2021. 2

[41] Mike Wu, Chengxu Zhuang, Milan Mosse, Daniel Yamins, and Noah Goodman. On mutual information in contrastive learning for visual representations. *arXiv preprint arXiv:2005.13149*, 2020. 2

[42] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 8

[43] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018. 2

[44] Tete Xiao, Xiaolong Wang, Alexei A Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. *ICLR*, 2021. 2

[45] Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Zhenguo Li, and Ping Luo. Detco: Unsupervised contrastive learning for object detection. *arXiv preprint arXiv:2102.04803*, 2021. 2

[46] Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. *CVPR*, 2021. 2

[47] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, 2019. 2

[48] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019. 2

[49] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *ICLR*, 2018. 2

[50] Junhao Zhang, Yali Wang, Zhipeng Zhou, Tianyu Luan, Zhe Wang, and Yu Qiao. Learning dynamical human-joint affinity for 3d pose estimation in videos. *IEEE Transactions on Image Processing*, 30:7914–7925, 2021. 1

[51] Rui Zhu, Bingchen Zhao, Jingen Liu, Zhenglong Sun, and Chang Wen Chen. Improving contrastive learning by visualizing feature transformation. *arXiv preprint arXiv:2108.02982*, 2021. 2, 5, 6, 8

[52] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014. 4

[53] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D Cubuk, and Quoc V Le. Rethinking pre-training and self-training. *arXiv preprint arXiv:2006.06882*, 2020. 2