Neural Networks 176 (2024) 106350

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Full Length Article

Any region can be perceived equally and effectively on rotation pretext task using full rotation and weighted-region mixture

Wei Dai [a], Tianyi Wu [a], Rui Liu [a], Min Wang [a], Jianqin Yin [b], Jun Liu [a,*]

[a] Centre for Robotics and Automation, City University of Hong Kong, Hong Kong, China
[b] School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, China

ARTICLE INFO

Keywords:
Self-supervised learning
Full rotation
Data mixing
Vision impairment

ABSTRACT

In recent years, self-supervised learning has emerged as a powerful approach to learning visual representations without requiring extensive manual annotation. One popular technique involves using rotation transformations of images, which provide a clear visual signal for learning semantic representation. However, in this work, we revisit the pretext task of predicting image rotation in self-supervised learning and discover that it tends to marginalise the perception of features located near the centre of an image. To address this limitation, we propose a new self-supervised learning method, namely FullRot, which spotlights underrated regions by resizing the randomly selected and cropped regions of images. Moreover, FullRot increases the complexity of the rotation pretext task by applying the degree-free rotation to the region cropped into a circle. To encourage models to learn from different general parts of an image, we introduce a new data mixture technique called WRMix, which merges two random intra-image patches. By combining these innovative crop and rotation methods with the data mixture scheme, our approach, FullRot + WRMix, surpasses the state-of-the-art self-supervision methods in classification, segmentation, and object detection tasks on ten benchmark datasets with an improvement of up to +13.98% accuracy on STL-10, +8.56% accuracy on CIFAR-10, +10.20% accuracy on Sports-100, +15.86% accuracy on Mammals-45, +15.15% accuracy on PAD-UFES-20, +32.44% mIoU on VOC 2012, +7.62% mIoU on ISIC 2018, +9.70% mIoU on FloodArea, +25.16% AP_{50} on VOC 2007, and +58.69% AP_{50} on UTDAC 2020. The code is available at https://github.com/anthonyweidai/FullRot_WRMix.

1. Introduction

Unsupervised learning for visual representations, unlike its supervised counterpart, eliminates the need for manual labelling, resulting in substantial savings in time and resources. Self-supervised learning, a subset of unsupervised learning, has demonstrated remarkable achievements across various computer vision tasks such as classification (Bao, Dong, Piao, & Wei, 2021; Chen, Fan, Girshick, & He, 2020; Chen & He, 2021; Chen, Kornblith, Norouzi, & Hinton, 2020; Doersch, Gupta, & Efros, 2015; Gidaris, Singh, & Komodakis, 2018; He, Fan, Wu, Xie, & Girshick, 2020; Mazumder, Singh, & Namboodiri, 2021; Misra & Maaten, 2020; Noroozi & Favaro, 2016; Noroozi, Vinjimoor, Favaro, & Pirsiavash, 2018; Peng, Dong, Bao, Ye, & Wei, 2022; Zbontar, Jing, Misra, LeCun, & Deny, 2021), segmentation (Bao et al., 2021; Chen & He, 2021; Gidaris et al., 2018; He et al., 2020; Noroozi et al., 2018; Peng et al., 2022; Wang et al., 2022; Zbontar et al., 2021), object localisation (Chen, Fan, et al., 2020; Chen & He, 2021; Doersch et al., 2015; Gidaris et al., 2018; He et al., 2020; Misra & Maaten, 2020; Noroozi et al., 2018; Zbontar et al., 2021), depth estimation (Zhang et al., 2022), and counting (Chen, Zhou, Li, Wei, & Xiao, 2022). Self-supervision approaches can be categorised by whether they employ contrastive learning. Contrastive methods require careful configuration of the objective function (Chen, Kornblith, et al., 2020; Zbontar et al., 2021) and data augmentation (Chen, Fan, et al., 2020; Chen & He, 2021; Chen, Kornblith, et al., 2020). In contrast, non-contrastive methods can integrate a cross-entropy objective function and straightforward data augmentation. One non-contrastive technique in self-supervised learning is predicting an image's rotation degree (Gidaris et al., 2018), which is relatively simple and intuitive. However, when using a fixed transformation centre, the rotation angle can primarily be inferred from the features near the image's centre or border. Consequently, the visual information near the image's centre may be neglected and underutilised.

In this paper, we aim to re-evaluate the pretext task of image rotation and attempt to answer the fundamental question: can we

* Corresponding author.
E-mail addresses: wei.dai@my.cityu.edu.hk (W. Dai), wu.tianyi@my.cityu.edu.hk (T. Wu), rui.liu@my.cityu.edu.hk (R. Liu), min.wang@my.cityu.edu.hk (M. Wang), jqyin@bupt.edu.cn (J. Yin), jun.liu@cityu.edu.hk (J. Liu).

https://doi.org/10.1016/j.neunet.2024.106350
Received 8 June 2023; Received in revised form 15 January 2024; Accepted 28 April 2024
Available online 30 April 2024

systematically incorporate image features from both border and central regions into the pretext tasks? Two commonly used pretext tasks, context prediction (Doersch et al., 2015) and jigsaw puzzles (Noroozi & Favaro, 2016; Noroozi et al., 2018), involve slicing the selected regions and deducing the relative locations of image patches. These tasks address the issue of neglecting partial features by considering different regions of an image. In addition, mixed sample data augmentation (MSDA) methods such as Mixup (Zhang, Cisse, Dauphin, & Lopez-Paz, 2018), CutMix (Yun et al., 2019), HMix (Park, Yun, & Chun, 2022), and GMix (Park et al., 2022), have been proposed to prevent deep learning model from disregarding non-discriminative parts of images. The MSDA methods blend regions from two different images. Taking inspiration from these spatial signal processing pretext tasks (Doersch et al., 2015; Noroozi & Favaro, 2016) and MSDA techniques (Park et al., 2022; Yun et al., 2019; Zhang et al., 2018), we introduce a novel pretext task and data mixture method. This new approach involves learning the rotation degree by observing the geometric transformations of a region combined with two random circular patches.

The contributions of this work can be summarised as follows:

- We have identified a significant limitation in the conventional rotation pretext task, which often overlooks features near the centre of images, akin to visual impairment. To address this, we propose a novel pretext method called FullRot to provide a more comprehensive understanding of semantic features.
- The FullRot method enhances the recognition of features in different image regions that were previously overlooked by the conventional rotation pretext task. This approach improves vision correction by randomly selecting, cropping, rotating, and resizing the partial region of images.
- We introduce an augmentation strategy called WRMix, which enhances models' feature generalisation and localisation capabilities. WRMix combines two intra-image patches using a weighted mask, allowing for blending with different magnitudes in different regions.
- We evaluate the FullRot with WRMix on diverse datasets, including STL-10, CIFAR-10, CIFAR-100, Sports-100, Mammals-45, PAD-UFES-20, ISIC 2018, FloodArea, TikTokDances, PASCAL VOC 2007, PASCAL VOC 2012, and UTDAC 2020, covering tasks of classification, semantic segmentation, and object detection. The experimental results demonstrate that the novel method achieves the best performance, outperforming other self-supervised learning methods, with the exception of securing the third position on the CIFAR-100 classification task.

In the subsequent sections, we review relevant works on self-supervised learning and data mixture in Section 2. We then present the FullRot and WRMix methods in Section 3, discuss the experimental results in Section 4, and summarise the discoveries in Section 6.

2. Related works

Self-supervised learning utilises either contrastive or non-contrastive learning strategies. Contrastive methods, such as SimCLR (Chen, Kornblith, et al., 2020), MoCo (Chen, Fan, et al., 2020; He et al., 2020), SimSiam (Chen & He, 2021), and Barlow (Zbontar et al., 2021) aim to maximise the similarities between two augmented images from the same source. SimCLR (Chen, Kornblith, et al., 2020), a simple framework for contrastive learning, employs stochastic data augmentation to transform samples, generating positive pairs and identifying the source sample of a pair using contrastive loss. Additionally, MoCo (momentum contrast) (He et al., 2020) updates the primary and asymmetric momentum encoders separately. MoCo v2 (Chen, Fan, et al., 2020) further enhances MoCo by utilising a projection head and more complex data augmentation. SimSiam (simple siamese) (Chen & He, 2021) also enhances the asymmetric degree using a stop-gradient operation. Barlow (Zbontar et al., 2021), named after neuroscientist H. Barlow and based on SimCLR, applies a distortion operation to process the input sample and embeds the projection output to prevent the model from collapsing into a constant. These methods require careful design of the objective function (Chen, Kornblith, et al., 2020; Zbontar et al., 2021) and heavily depend on the configurations of data augmentation (Chen, Fan, et al., 2020; Chen & He, 2021; Chen, Kornblith, et al., 2020).

In comparison, non-contrastive methods propose pretext tasks to derive input images and output labels from unlabelled data. These pretext tasks can directly utilise objective functions, such as cross-entropy loss, from supervised learning (Doersch et al., 2015; Gidaris et al., 2018; Noroozi & Favaro, 2016; Noroozi, Pirsiavash, & Favaro, 2017). Recently, Bao et al. (2021) and Peng et al. (2022) introduced a token-based approach known as "bidirectional encoder representation from image transformer" (BEiT) while He et al. (2022) proposed a pixel-based method called "masked autoencoders" (MAE) for the restoration of masked images. However, BEiT and MAE are unsuitable for convolutional neural networks (CNNs) due to their dependency on patch embedding and vision transformer (ViT) encoders. ViT architectures, compared to CNNs, have insufficient inductive biases (Dosovitskiy et al., 2021). Therefore, ViT is more sensitive to perturbations and less able to generalise well to inadequate training data. Although Huang, Xu, Wang, Wang, and Zhang (2022) introduced a masked image recovery pretext task in convolutional networks, this task relies on a specifically designed decoder and may bear the expensive computational cost of recovering each pixel of covered regions.

Rotation is a commonly used data augmentation method in machine learning and computer vision tasks (Mehta & Rastegari, 2021). It has been employed as a pretext task in self-supervised learning by Gidaris et al. (2018). In this approach, a source image is used to generate four additional images by rotating 0, 90, 180, and 270 degrees separately. To improve image synthesis, the rotation has also been integrated with GANs as a self-supervised method called SS-GAN (Chen, Zhai, Ritter, Lucic, & Houlsby, 2019). Another related work is the rotation-based open set (ROS) proposed by Bucci, Loghmani, and Tommasi (2020), which applies image rotation to create common relations between source and target domains and align their distributions. Furthermore, using the distance penalty theory, Feng, Xu, and Tao (2019) introduced semantic feature decoupling to disassociate individual image discrimination from rotation features. Qing, Zeng, Cao, and Huang (2021) applied rotation transformations to both labelled and unlabelled data as an auxiliary task to enhance the regularisation of supervising signals. For better transferable representations, Liu, Li, Lei, and Shi (2022) utilised rotation self-supervision for knowledge distillation, incorporating complementary labels (CL) in an additional task, known as SELF-CL. Moreover, Lim, Lim, Lee, and Tan (2023) introduced the Self-supervised Contrastive Learning method (SCL), which combines rotation degree prediction and contrastive learning techniques to minimise the distance between each training image, specifically for few-shot image classification. However, methods such as SS-GAN (Chen et al., 2019), ROS (Bucci et al., 2020), semantic feature decoupling (Feng et al., 2019), auxiliary rotation self-supervision (Qing et al., 2021), SELF-CL (Liu, Li, Lei & Shi, 2022), and SCL (Lim et al., 2023) utilise multiple self-supervised classification heads, which presumably consume significant computational resources.

Although the rotation pretext task has inspired various representation learning research, several problems are still associated with this self-supervision method. When images have apparent but partial characteristics near their boundaries (see Fig. 1a), the visual cue after rotation could cause a lack of awareness of information close to the boundaries or centres (see Fig. 1b–d). Moreover, the use of rotation degrees that are not multiples of 90, such as 45, 135, 225, or 315, leads to a significant drop in the evaluation performance, as Gidaris et al. (2018) reported. The problem may be caused by conspicuous visual corners left by rotation transformations without multiples of 90° (see Fig. 1ef). This problem will be further discussed in Section 3.1.

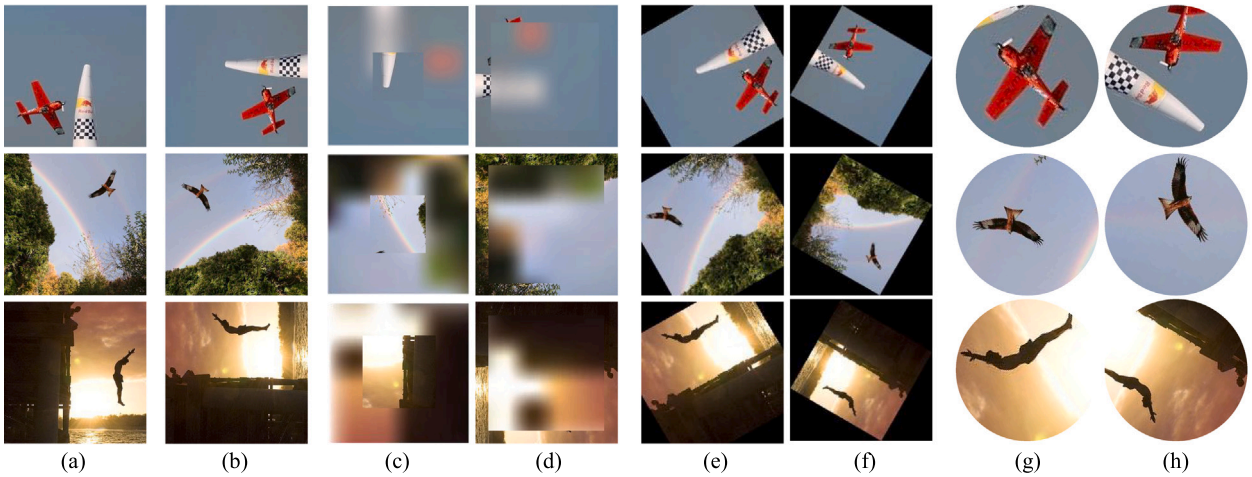

Fig. 1. Rotated image samples generated from (a) original images by (b) vanilla rotation, (c) rotation with the blurred region, (e–f) rotation without multiples of 90°, and (g–h) full rotation (FullRot). When comparing (a) and (b), individuals familiar with the objects in the image can likely determine the rotation angle. Even if the image's boundary (c) or centre (d) is blurred, estimating the angle becomes more intricate but still feasible. To incorporate features from all regions of an image on the pretext task, FullRot randomly crops regions from the image. Furthermore, by removing four corners and retaining a circular shape, the image can be rotated by any degree. Otherwise, the corners of images become apparent features (e-f), shifting the pretext task from predicting angles to searching for corners.

Relative position encompasses two common pretext tasks, context (Doersch et al., 2015) and Jigsaw puzzles (Noroozi & Favaro, 2016). These tasks involve slicing selected regions from an image to develop a visuospatial representation of patches. Jigsaw++ (Noroozi et al., 2018) further rearranges tiles extracted from two different images into two new patch grids to hinder the pretext task and improve the performance of self-supervised learning. Besides, Kim, Cho, Yoo, and Kweon (2018) integrated inpainting and colourisation with jigsaw puzzles to create the "completing damaged jigsaw puzzles" task, improving the representation of learning performance. Compared to the rotation pretext, the relative position pretext utilises spatial signals from different patches of an input image. Mazumder et al. (2021) bridged rotation and relative position by slicing images and rotating the patches. Although this strengthens the consciousness of image centres during self-supervised training, the tile gridding process increases the number of pretext classes and the computational cost.

Mixed sample data augmentation (MSDA) is used in computer vision to expand the dataset and smooth the decision boundary by synthesising multiple samples to create a new sample for training a potent and reliable deep learning model (Kim, Choo, & Song, 2020; Liu, Li, Wang et al., 2022; Park et al., 2022; Verma et al., 2019; Yun et al., 2019; Zhang et al., 2018). These augmentation techniques can be applied at either the image (Kim, Choo, & Song, 2020; Park et al., 2022; Yun et al., 2019; Zhang et al., 2018) or feature (Verma et al., 2019) level. At the image level, samples are combined using techniques such as linear interpolation (Zhang et al., 2018), patch removal (DeVries & Taylor, 2017), or cut-and-paste method (Kim, Choo, & Song, 2020; Park et al., 2022; Yun et al., 2019). Liu, Li, Wu et al. (2022) proposed Mixup and CutMix by utilising the class activation maps to guide the generation of the mixed data. Instead of blending ground truths as supervision signals, interpolated losses (Vu et al., 2023) were developed for mingled data in object detection.

Most recently, MSDA has shown improved performance on contrastive self-supervised learning across various computer vision applications, including image classification (Kalantidis, Sariyildiz, Pion, Weinzaepfel, & Larlus, 2020; Kim, Lee, Bae, & Yun, 2020; Shen et al., 2022), object segmentation (Kalantidis et al., 2020), object detection (Shen et al., 2022), video representation learning (Wang et al., 2021), and domain adaptation (Lee et al., 2021). To learn more robust features in contrastive learning, negative pairs were introduced by hard negative mixing (Kalantidis et al., 2020) or image-level mingling within a single branch (Kim, Lee, et al., 2020). Shen et al. (2022) deployed intra-image synthesis to increase the number of pairs with a memory bank and computed a hybrid loss for mitigating the overfitting problem. However, the requirement of an additional memory bank inevitably leads to expensive computational costs. More importantly, the impact of these data-mixing methods on non-contrastive self-supervised learning is still under rapid development and requires further investigation.

3. Methodology

The central concept of this study is to incorporate the awareness of different regions within an image into the rotation pretext task. The objective is to estimate the rotation transformation parameters that were applied to the input image and improve the performance and effectiveness of the rotation pretext task.

3.1. Full rotation framework

This section presents the methodology and overall framework of the FullRot method, as depicted in Fig. 2.

Random Centre of Rotation. Rotating the entire image may overlook important features near the image borders, as the rotation transformation can primarily be inferred from information near the centre or boundary of the image (see Fig. 1cd). To address this issue and mitigate visual impairment near the boundaries, we adopted a random centre of rotation. In each training epoch, an arbitrary rotation centre is selected to guarantee that every region is treated as necessary as the "centre" in the vanilla rotation method (see Fig. 1gh).

Random Crop Ratio. The crop ratio is defined as the diameter of the cropped region divided by the length of the short edge of an image (*i.e.*, width or height, whichever is minimum). A smaller crop ratio provides more choices for the rotation centres. However, it also reduces the richness of features in the cropped image. To strike a balance and ensure sufficient information for the pretext task, the crop ratio is randomly selected from a range of m to 1. m is the minimum crop ratio and set to 0.6.

Full Rotation Degree. A digital image typically has a rectangular shape with a horizontal orientation, which restricts the output shape of rotated images to rectangles. When the rotation degree is not multiples of 90° (*e.g.*, 120° and 240°), the four corners of the rotated images become visually salient and may simplify the pretext task (see Fig. 1ef). To

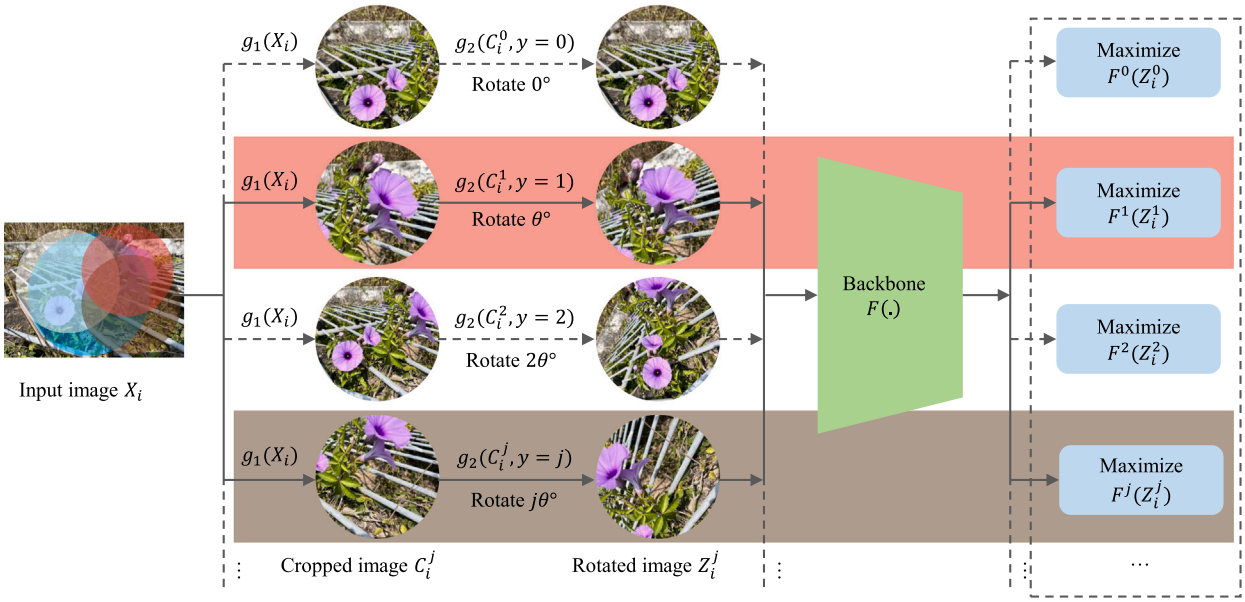

Fig. 2. Illustration of FullRot self-supervised framework. It randomly selects and crops the region from the input image X_i using a random crop ratio, rotates the cropped image C_i^j using specific degrees with a constant gap, and then feeds the rotated image Z_i^j into the backbone $F(.)$ for maximising prediction probability of rotation degrees $F^y(Z_i^j)$. During training, only two random classes of rotated images are utilised in each training epoch, as indicated with red or brown colour in the figure.

mitigate the negative effect of corners, we introduce circular cropping. By cropping the image into a circular shape, it can be rotated by any degree. Specifically, if there are d rotation classes, the rotation degree gap between neighbouring classes is $360°/d$.

The Number of Feeding Images. In the pretext task that relies on the mutual relationship of geometrically transformed outputs (Doersch et al., 2015; Gidaris et al., 2018; Noroozi & Favaro, 2016; Noroozi et al., 2018), all transformed images from the same source are typically used in every training epoch. However, not all rotation classes have sufficient representative features for practical self-supervised training. Inspired by contrastive learning approaches (Chen & He, 2021; Chen, Kornblith, et al., 2020) that utilise only two augmented samples from the same image, two rotated images generated from the same image are randomly chosen for training in each epoch. This approach increases the intricacy of the pretext task and improves the performance of self-supervised learning (see Table 5). Moreover, maintaining a consistent number of rotated input images across different rotation classes d simplifies the training process by ensuring an equal number of training epochs.

Loss Function. Assuming that the geometrically transformed cropped image is $C_i = g_1(X_i)$, where $X_i \in \mathbb{R}^{W\times H\times C}$ is the i th original input image and g_1 represents the cropping operator with a random region and varying ratio. Then, the rotated image with label y can be defined as $Z_i^y = g_2(C_i|y)$, where g_2 represents the rotation operator. The probability of the rotation transformation is given by:

$$F(Z_i^{y^*}|\omega) = \{F^y(Z_i^{y^*}|\omega)\}^T \tag{1}$$

where, the feature backbone $F(.)$ takes a cropped image $Z_i^{y^*}$ with an unknown label y^*, $F^y(Z_i^{y^*}|\omega)$ is the predicted probability of rotation transformation, T represents the number of rotation classes, and ω denotes the learnable parameters of backbone $F(.)$.

Given a set of N training images, the total number of training images is denoted as $M = \{X_i\}_{i=0}^N$. Thus, the objective of self-supervised training is to minimise the crop-entropy loss:

$$-\frac{1}{NT}\sum_{i=0}^{N}\sum_{y=0}^{T-1}\log(F^y(g_2(g_1(X_i)|y)|\omega)) \tag{2}$$

3.2. Weighted-region mixture

In this section, we introduce the weighted-region mixture (WRMix) technique, which draws inspiration from the concepts of Mixup (Zhang et al., 2018) and CutMix (Yun et al., 2019) The idea behind WRMix is to leverage regional information of an image to guide the FullRot method in learning the underrepresented parts of objects. Given that the cropped regions in FullRot have random locations and side lengths (illustrated in Section 3.1), these regions can be combined to form a region that contains rich visual information. WRMix focuses on amalgamating two regions from the same image, called the intra-image mixture, as opposed to the inter-image mixture. Moreover, the intra-image mixture is more advantageous for FullRot, as highlighted in Table 7. Therefore, we specifically discuss intra-image mixture as an example of WRMix in this section.

To simplify the pretext task, only the training images are regenerated, while the training labels remain the same (using the same rotation angle). Two randomly cropped regions Z_{ia}^j and Z_{ib}^j from the same image with the same rotation angle $j\theta°$ are blended to create a new training sample $\tilde{Z}_i^j$. The blending operation is defined as follows:

$$\tilde{Z}_i^j = \mathbf{M}\odot Z_{ia}^j + (\mathbf{1}-\mathbf{M})\odot Z_{ib}^j \tag{3}$$

where $\mathbf{M} \in (0,1)^{W\times H}$ denotes a real-valued mask that determines where the two images are mingled using a combination wight λ_1 following beta distribution $\sim$ beta(α,α). The symbol $\odot$ represents element-wise matrix multiplication.

To create the mask $\mathbf{M}$, the bounding box $(l_{x1}, l_{y1}, l_{x2}, l_{y2})$ of the weighted region should be sampled. The mask $\mathbf{M}$ is first separated from the central region and then moved by a random distance according to:

$$\begin{aligned}\Delta_x &\sim \mathcal{U}_{[-(1-\lambda')W/2,(1-\lambda')W/2]}\\ \Delta_y &\sim \mathcal{U}_{[-(1-\lambda')H/2,(1-\lambda')H/2]}\end{aligned} \tag{4}$$

and

$$\begin{aligned}l_{x1} &= (1-\lambda')W/2 + \Delta_x\\ l_{y1} &= (1-\lambda')H/2 + \Delta_y\\ l_{x2} &= (1+\lambda')W/2 + \Delta_x\\ l_{y2} &= (1+\lambda')H/2 + \Delta_y\end{aligned} \tag{5}$$

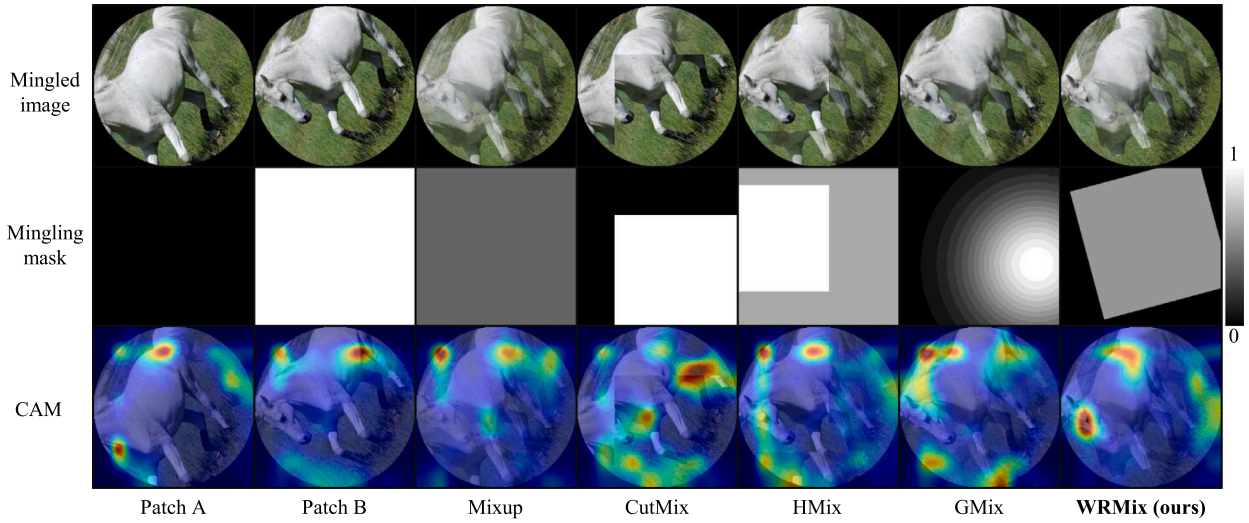

Fig. 3. Examples generated by different data blending methods.

where W and H are the width and height of the image, the ratio of the cropped mask area is $\frac{(l_{x2}-l_{x1})(l_{y2}-l_{y1})}{HW} / = \lambda'^2$, and λ' is the proportion of cropping length to the original image size, defined as

$$\lambda' = \mathrm{c} + \sqrt[4]{1-\lambda_2}(1-\mathrm{c}) \tag{6}$$

where $\lambda_2 \sim \mathrm{beta}(\alpha, \alpha)$, and c is a constant set to 0.5, meaning the mask' minimum length is half the original image's side length.

To establish a relationship between the weight and the area of the masking region, λ_1 and λ_2 have the following relationship:

$$\lambda_1 = \mathrm{beta}(\alpha, \alpha) + \mathrm{b}(1-\lambda'^2) \tag{7}$$

where b is a constant set to 0.2. This formula ensures that as the amalgamated area increases, the combination weight λ_1 is more likely to follow the standard beta distribution.

Examples of augmented images and corresponding class activation map (CAM) are shown schematically in Fig. 3. WRMix combines patches from two different regions of an image and highlights understated features by repositioning spatial features, distinguishing itself from Mixup (Zhang et al., 2018) and CutMix (Yun et al., 2019). In the third column of Fig. 3, the CAM of different data mixture methods indicate that CutMix (Yun et al., 2019) and HMix (Park et al., 2022) tend to direct the network towards the mask boundary. However, the WRMix mask is rotated randomly to help alleviate the negative effect of irrelevant information on the mask boundary. Thus, WRMix captures more general features than CutMix and HMix, even when using a cropped mask.

Algorithm 1 describes the implementation of WRMix in the FullRot pretext task. Based on the ablation results of α in Fig. 5, α is set to 1 in all experiments unless otherwise specified. Therefore, λ_1 and λ_2 are sampled from the continuous uniform distribution $\mathcal{U}_{[0,1]}$.

Algorithm 1 Pseudocode of WRMix in a python-like style.

```
for x in Loader:
        # create patches and labels by FullRot
        x1, x2, Labels = fullRotation(x)
        W, H = x1.size

        # sampled from the beta distribution
        Lam2 = beta(Alpha, Alpha)
        # restricted within (c, 1)
        Lam2 = c + (1 - Lam2) ** (0.25) * (1 - c)
        # weighted random region generation
        BBox, BBoxArea = randWRBBox([W, H], Lam2)
        # the proportion of region that is not blended
        NonMix = 1 - BBoxArea / (W * H)
        # follows the beta distribution with bias
        Lam1 = min(beta(Alpha, Alpha) + b * NonMix, 1)

        Mask = zeros([H, W])
        # weighted bounding location
        Mask[BoundingBox] = 1 - Lam1
        # randomly rotation
        Mask = rotate(Mask, uniform(0, 360))
        # mix patches 1 and 2 with a mask
        MixImage = Mask * x1 + (1 - Mask) * x2

        Output = ResNet(MixImage)
        Loss = CrossEntropyLoss(Output, Labels)
```

3.3. Evaluation protocol

Based on the empirical studies, our framework has several specific design choices to achieve the optimised output. We evaluated the proposed algorithms with the following conditions.

Dataset. To statistically validate the robustness of our proposed method, we test our method and the control group in diverse scenarios across three independent tasks, including classification, semantic segmentation, and object detection.

The dataset of classification task used in this paper includes CIFAR-10/100 (Krizhevsky, 2009), STL-10 (Coates, Ng, & Lee, 2011), Sports-100 (Piosenka, 2022), Mammals-45 (Asaniczka, 2023), and PAD-UFES-20 (Pacheco et al., 2020). CIFAR-10 and CIFAR-100 consist of 50,000 training and 10,000 test 32×32 images of common objects with 10 and 100 classes, respectively. With a larger input size, 96×96, STL-10 consists of 5,000 training and 8,000 test common object images with 10 classes, and 100,000 unlabelled images. Sports-100 is a collection of sports images with 100 sport types, including 13,493 training, 500 validation and 500 test 224×224 images. Besides, the Mammals-45 dataset comprises 13,751 images and 45 mammal classes sourced from Google Images. Meanwhile, PAD-UFES-20 is a smartphone-collected skin lesions dataset with 2,298 images and 6 classes. Without the division of training and test data, five-fold cross-validation is applied to Mammals-45 and PAD-UFES-20. For balanced comparison, Mammals-45 and PAD-UFES-20 images are resized to 224×224. Additionally, due to computational resource constraints, 100,000 STL-10 unlabelled

images are applied to pretrain deep neural networks in a self-supervised manner.

The performance of our method is also verified on the semantic segmentation task using PASCAL VOC 2012 (Everingham et al., 2015), ISIC 2018 (Codella et al., 2019; Tschandl, Rosendahl, & Kittler, 2018), FloodArea (Karim, Sharma, & Barman, 2022), and TikTokDances (Roman, 2019), while object detection task using PASCAL VOC 2007+2012 (Everingham et al., 2015; Everingham, Van Gool, Williams, Winn, & Zisserman, 2010) and UTDAC 2020 (Song, Li, Dai, Wang, & Chen, 2023). PASCAL VOC 2007 has 9,963 images with 20 classes and 24,640 ROI-annotated objects. PASCAL VOC 2012 includes 11,530 images with 20 classes, 27,450 ROI-labelled objects, and 6,929 segmentation masks. Both datasets follow the 1:1 data division for training/validation and test sets. For the object detection task, the training set of PASCAL VOC 2007 + 2012 is utilised to finetune networks, and the PASCAL VOC 2007 test set is used for evaluation (the labels of the test set of PASCAL VOC 2012 are inaccessible). In addition, ISIC 2018 has 2,594 training images and 1,000 testing images with corresponding masks for unhealthy areas. FloodArea is a dataset of flood surveys containing 290 images of flood-affected areas, each accompanied by a corresponding mask for the water region. Besides, the TikTokDances dataset comprises 2,615 images, each featuring a masked dancing human picture. The FloodArea and TikTokDances datasets adhere to a 4:1 ratio for the division of training and test sets. In addition, the UTDAC 2020 dataset is dedicated to underwater object detection, encompassing 5,168 training and 1,293 test images with 4 classes: echinus, holothurian, starfish, and scallop.

Default Setting. The networks for classification, segmentation, and detection tasks are trained on an RTX3090 GPU and an Intel Xeon Platinum 8375C CPU. We choose ResNet50 (He, Zhang, Ren, & Sun, 2016) and ViT-B/16 (Dosovitskiy et al., 2021) as the basic CNN and ViT encoder network, respectively. The learning rate is adjusted using a cosine schedule (Loshchilov & Hutter, 2017), and the initial 10% of the training epochs allocated for learning rate warm-up and freezing the weights of the transferred backbones. The code for the three vision tasks is inspired by Mehta, Abdolhosseini, and Rastegari (2022) and implemented using PyTorch (Paszke et al., 2019) and Detectron2 (Wu, Kirillov, Massa, Lo, & Girshick, 2019) package, excluding the MAE (He et al., 2022) and BEiT v2 (Peng et al., 2022) models, which are trained and validated using their official implementations. Unless otherwise stated, the number of FullRot classes d is set to 12, in accordance with the studies referenced in Section 4.3.

4. Experimental results

In this section, we present the experimental results of FullRot and WRMix on various tasks. The performance of FullRot + WRMix is first evaluated on the STL-10, CIFAR-10/100, and PAD-UFES-20 datasets for classification tasks in Section 4.1. The classification results demonstrate that FullRot + WRMix delivers the best performance among all tested self-supervised learning (SSL) methods on STL-10, CIFAR-10, Sports-100, Mammals-45, and PAD-UFES-20, as well as non-contrastive SSL methods on CIFAR-100.

In Section 4.2, we investigate the versatility of FullRot and WRMix by evaluating their performance on ISIC 2018/FloodArea/TikTokDances segmentation task, PASCAL VOC (*abbr.*VOC) segmentation and detection tasks, and the UTDAC 2020 detection task. The results reveal that the combination of FullRot and WRMix outperforms the state-of-the-art (SOTA) SSL methods in these tasks, except for securing a commendable second place in the TikTokDances segmentation task.

Additionally, the ablation studies for FullRot and WRMix are presented in Section 4.3 to analyse the impact of different components and parameters. Finally, we provide visualisation of t-SNE plots of features produced by different SSL methods and class activation maps (CAM) for vanilla rotation and FullRot in Section 4.4, offering insights into the learned representations.

4.1. Image classification results

Implementation Details. During pretraining, FullRot and other SOTA methods are trained for 200 epochs on the STL-10 unlabelled set. The pretrained models are then finetuned for 100 epochs on the respective training set of STL-10, CIFAR-10/100, Sports-100, Mammals-45, and PAD-UFES-20. The batch size used is 1024, 256, and 128 images for STL-10, CIFAR-10/100, and Sports-100/Mammals-45/PAD-UFES-20, respectively. The learning rate starts at 0.0002 and is increased to 0.002 for the first 10% epochs before annealing to 0.0002 with Adam optimiser (Loshchilov & Hutter, 2018). Fundamental data augmentation techniques such as image resizing are applied, and cross-entropy loss with label smoothing (value 0.1) and class sensitivity are used while training all non-contrastive SSL methods. For contrastive SSL methods, the settings reported in Chen, Fan, et al. (2020), Chen and He (2021), Chen, Kornblith, et al. (2020) and Zbontar et al. (2021) are strictly followed. The performance of SOTA methods is evaluated by averaging the results from three experiments on the STL-10, CIFAR-10/100, and Sports-100 test sets, while for Mammals-45 and PAD-UFES-20, five-fold cross-validations are performed. The evaluation metrics used are top-1 linear classification accuracy and 5-nearest neighbours (5-nn) classification accuracy.

Comparison with Non-pretrained Baseline. It is evident from Table 1 that the CNN-based SSL methods exceptionally surpass the non-pretrained CNN baseline with 12.64% $\sim$ 19.34% linear and 10.72% $\sim$ 18.45% 5-nn classification accuracy rises on the STL-10 test set. On the CIFAR-10 test set for the CNN-based methods, the accuracy gaps are smaller, ranging from 2.34% to 5.08% linear and 1.03% to 3.56% 5-nn, excluding Barlow, which has the lowest linear (84.39%) and 5-nn (83.48%) accuracy. The gaps become larger on the CIFAR-100 test set for the CNN-based methods, which ranges from 2.14% to 11.90% linear and 3.70% to 11.92% 5-nn accuracy, except for Barlow, which has the lowest 5-nn accuracy at 50.46%. On the Sports-100 and Mammals-45 test sets, the CNN-based SSL methods perform better than the non-pretrained baseline with no less than 0.5% linear and 5-nn accuracy, except that Barlow exhibits the lowest 5-nn accuracy falling $>$ 3% lower than the baseline on both datasets and Jigsaw++ records the second-lowest 5-nn accuracy of 81.35% on the Mammals-45 test set.

Furthermore, the ViT-based SSL methods (*i.e.*, MAE and BEiT v2) attain a significant improvement over non-pretrained ViT baseline with a 24.03% $\sim$ 33.56% linear and 29.53% $\sim$ 40.19% 5-nn accuracy rises on the STL-10 test set. These ViT-based SSL methods also obtain a minimum of 35% increases in linear and 5-nn accuracy over the non-pretrained ViT baseline on the CIFAR-10/100, Sports-100, and Mammals-45 test sets.

Even with a larger domain difference between the STL-10 unlabelled set and PAD-UFES-20 dataset, all tested SSL methods still show considerable accuracy growths (*e.g.*, +0.87% $\sim$ +11.52% linear accuracy) on the PAD-UFES-20 test set compared to the one without pretraining.

These results demonstrate that self-supervised pretraining using FullRot or other SOTA methods can significantly enhance representation learning, except for Barlow, which may lead to overfitting and worse performance on CIFAR-10/100, Sports-100, and Mammals-45 classification tasks. Notably, the FullRot + WRMix method exceeds the non-pretrained group by $>$ 10% linear and 5-nn accuracy on the STL-10 and PAD-UFES-20 datasets, and by $>$5% on the CIFAR-10/100 and Mammals-45 datasets, except for $>$3% 5-nn accuracy on the CIFAR-10 and Sports-100 datasets.

Comparison with the SOTA SSL Methods. According to Table 1, FullRot with WRMix achieves the highest classification accuracy, 88.11% & 89.79% & 95.53% & 91.24% & 72.54% linear and 87.80% & 89.77% & 96.07% & 88.87% & 73.19% 5-nn, among the tested SSL methods on STL-10, CIFAR-10, Sports-100, Mammals-45, and PAD-UFES-20, respectively. FullRot + WRMix outperforms the second-place methods, MoCo v2, by 1.21% 5-nn accuracy on the PAD-UFES-20

Table 1
Linear and 5-nearest neighbours classification results for different methods and datasets.The best results of non-contrastive and contrastive learning methods are emphasised in bold and underlined, respectively.

Self-supervision	Method	STL-10		CIFAR-10		CIFAR-100		Sports-100		Mammals-45		PAD-UFES-20	
		linear ⇑	5-nn	linear	5-nn	linear	5-nn	linear	5-nn	linear	5-nn	linear	5-nn
Non-pretrained Baseline	ViT	50.10	45.46	38.87	38.27	16.42	17.65	22.53	26.87	32.18	28.27	55.94	51.96
	CNN	68.77	69.35	84.71	86.21	52.34	51.26	91.07	90.60	85.05	81.82	62.49	63.75
Contrastive Learning	SimSiam (Chen & He, 2021)	82.85	83.25	89.74	89.76	64.24	62.54	93.93	93.07	87.42	85.07	63.36	63.58
	MoCo v2 (Chen, Fan, et al., 2020)	84.35	84.22	88.77	88.90	62.34	60.60	94.07	94.40	89.72	87.02	70.58	71.98
	SimCLR (Chen, Kornblith, et al., 2020)	86.75	87.22	89.60	89.72	64.22	63.18	95.00	95.07	90.80	88.40	66.84	66.15
	Barlow (Zbontar et al., 2021)	87.61	83.88	84.39	83.48	54.48	50.46	93.20	87.07	87.85	74.51	69.19	68.54
Non-contrastive Learning	MAE[a] (He et al., 2022)	74.13	74.99	81.23	81.34	55.08	53.24	87.93	85.87	75.38	75.89	63.55	58.04
	Rotation (Gidaris et al., 2018)	81.41	80.07	87.37	87.24	56.84	54.96	94.13	93.27	89.94	87.10	69.63	70.02
	BEiT v2[a] (Peng et al., 2022)	83.66	85.65	81.92	81.57	56.04	56.56	91.80	92.93	80.86	80.24	67.46	68.70
	Jigsaw++ (Noroozi et al., 2018)	85.51	82.92	87.76	88.27	59.68	55.95	93.67	90.93	86.07	81.35	66.49	67.19
	Jigsaw (Noroozi & Favaro, 2016)	85.26	84.27	87.05	87.59	58.26	56.24	92.20	91.33	86.07	83.39	64.71	66.10
	Context (Doersch et al., 2015)	85.71	85.56	88.34	88.45	59.52	56.68	92.47	91.47	87.10	82.59	66.71	68.76
	FullRot (Ours)	**87.46**	**87.29**	**89.62**	**89.64**	**62.23**	**60.59**	**95.13**	**95.13**	**90.82**	**88.59**	**72.11**	**72.76**
	FullRot + WRMix (Ours)	**88.11**	**87.80**	**89.79**	**89.77**	**62.51**	**60.71**	**95.53**	**96.07**	**91.24**	**88.87**	**72.54**	**73.19**

[a] Employ ViT as the backbone.

test set. Moreover, FullRot achieves top-tier linear (62.51%) and 5-nn (60.71%) accuracy among non-contrastive SSL methods on the CIFAR-100 test set.

Furthermore, FullRot without WRMix surpasses the performance of traditional rotation on all six classification datasets with 5-nn accuracy improvement of +7.22% on STL-10, +2.40% on CIFAR-10, +5.63% on CIFAR-100, +1.86% on Sports-100, +1.49% on Mammals-45, and +2.74% on PAD-UFES-20. Besides, the inclusion of WRMix significantly enhances the performance of FullRot on all tested datasets (*e.g.*, 0.94% 5-nn accuracy on Sports-100, 0.65% linear accuracy on STL-10, 0.43% 5-nn accuracy on PAD-UFES-20, 0.42% linear accuracy on Mammals-45, 0.28% linear accuracy on CIFAR-100, and 0.17% linear accuracy on CIFAR-10).

4.2. Segmentation and detection results

To evaluate the general-purpose characteristics of FullRot and WRMix, we benchmark FullRot with WRMix on two broadly investigated tasks: semantic segmentation in Section 4.2.1 and object detection in Section 4.2.2.

4.2.1. Semantic segmentation results

Implementation Details. We consolidate the pretrained ResNet50 backbone with DeepLabv3+ (Chen, Zhu, Papandreou, Schroff, & Adam, 2018). The segmentation network is finetuned on the VOC 2012, ISIC 2018, FloodArea, or TikTokDances training set with an input size 512 × 512. The training process consists of 100 epochs, a batch size of 32 images, and the Adam optimiser with cross-entropy loss. The learning rate is ceased from 5×10^{-5} to 5×10^{-4} for the first 10% epochs and then annealed to 1×10^{-6}. The performance is evaluated on the corresponding validation set using mean intersection over union (mIoU), and the results are averaged in three experimental runs.

Results. Semantic segmentation results are revealed in Table 2, which demonstrates that SSL methods outperform the non-pretrained baseline with 1.51% ∼ 18.14%, 0.68% ∼ 8.08%, 0.10% ∼ 2.34%, and 0.38% ∼ 11.46% increased mIoU on VOC 2012, ISIC 2018, FloodArea, and TikTokDances test sets, respectively, except for BEiT v2, which has the lowest mIoU at 74.96% on FloodArea. Among tested SSL methods, FullRot without WRMix achieves the second-highest mIoU of 87.31% on ISIC 2018 and the third-highest mIoU of 41.72% and 95.14% on VOC 2012 and TikTokDances, respectively. When combined with WRMix, FullRot attains the highest mIoU of 43.02% on VOC 2012, 87.46% on ISIC 2018, and 84.66% on FloodArea. Additionally, FullRot with WRMix secures the second-highest mIoU of 95.18% on TikTokDances. These results suggest that WRMix can effectively assist FullRot in learning more critical representations. Besides, FullRot + WRMix overtakes the conventional rotation method with a mIoU increase of 14.61% on VOC 2012, 1.79% on ISIC 2018, 0.76% on FloodArea, and 0.14% on TikTokDances, demonstrating the statistical significance of our refined rotation pretext task.

4.2.2. Object detection results

Implementation Details. We deploy the backbone with the faster region-based convolutional network (Faster R-CNN) detector (Ren, He, Girshick, & Sun, 2017). For ViT backbone, a feature pyramid is used to bridge the backbone and Faster R-CNN as suggested by Li, Mao, Girshick, and He (2022). The training is performed with input dimensions ranging from 480 to 800 pixels, 24,000 iterations, and a batch size of 4 images on the VOC 2007 + 2012 or UTDAC 2020 training set. The other settings of the Faster R-CNN detector follow the default configurations of the VOC detection task in Detectron2 (Wu et al., 2019). The performance of object detection is evaluated on the VOC 2007 and UTDAC 2020 test sets by averaging three experiments' average precision at the IoU of 0.50, AP_{50}, and the two more strict metrics of COCO-style AP (at IoU of 0.50:0.05:0.95) & AP_{75} (at IoU of 0.75).

Results. The detection results on the VOC 2007 and UTDAC 2020 test sets presented in Table 3 illustrate that SSL methods deliver >5% AP_{50} and >2% AP & AP_{75} increment compared to non-pretrained baseline. Among all tested SSL methods, FullRot shows the top-tier AP_{50} of 48.77% & 75.54 (with WRMIx) and 47.36% & 74.82 (without WRMIx), surpassing the non-pretrained baseline, 17.94% & 53.97% AP_{50}, on the VOC 2007 and UTDAC 2020 test sets, respectively. Moreover, FullRot with WRMIx also achieves over 6% increased AP_{50}, AP, and AP_{75} on both datasets compared to the traditional rotation method. These results highlight the superior performance of FullRot and WRMix in representation learning for object detection, further validating the effectiveness of our refined rotation pretext task.

4.3. Ablation studies

Implementation Details. Considering the limited computational resources, the self-supervised training is conducted using 100 epochs.

Table 2
Segmentation results for different methods and datasets. The best results of non-contrastive and contrastive learning methods are emphasised in bold and underlined, respectively.

Self-supervision Method		VOC 2012	ISIC 2018	FloodArea	TikTokDances
		mIoU ⇑	mIoU	mIoU	mIoU
Non-pretrained Baseline	ViT	9.07	78.51	75.55	74.39
	CNN	24.88	84.65	82.32	93.86
Contrastive Learning	SimSiam (Chen & He, 2021)	37.22	86.88	84.27	95.10
	Barlow (Zbontar et al., 2021)	38.42	85.72	83.02	94.35
	MoCo v2 (Chen, Fan, et al., 2020)	38.65	<u>87.18</u>	84.43	95.08
	SimCLR (Chen, Kornblith, et al., 2020)	<u>42.43</u>	87.01	<u>84.63</u>	<u>95.23</u>
Non-contrastive Learning	BEiT v2[a] (Peng et al., 2022)	10.58	79.84	74.96	79.04
	MAE[a] (He et al., 2022)	26.82	86.59	76.57	85.85
	Rotation (Gidaris et al., 2018)	28.41	85.67	83.90	95.04
	Jigsaw++ (Noroozi et al., 2018)	32.05	85.73	82.64	94.33
	Context (Doersch et al., 2015)	32.23	85.85	82.44	94.35
	Jigsaw (Noroozi & Favaro, 2016)	32.28	85.33	82.84	94.24
	FullRot (Ours)	**41.72**	**87.31**	**84.19**	**95.14**
	FullRot + WRMix (Ours)	**43.02**	**87.46**	**84.66**	**95.18**

[a] Employ ViT as the backbone.

Table 3
Detection results for different methods and datasets. The best results of non-contrastive and contrastive learning methods are emphasised in bold and underlined, respectively.

Self-supervision Method		VOC 2007			UTDAC 2020		
		AP_{50} ⇑	AP	AP_{75}	AP_{50}	AP	AP_{75}
Non-pretrained Baseline	CNN	17.94	8.00	5.74	53.97	24.91	18.44
	ViT	19.50	8.31	6.03	56.16	25.80	19.02
Contrastive Learning	SimSiam (Chen & He, 2021)	34.00	16.90	14.62	50.44	22.89	16.29
	Barlow (Zbontar et al., 2021)	43.51	19.69	14.43	50.19	22.77	16.15
	MoCo v2 (Chen, Fan, et al., 2020)	45.66	23.37	20.60	55.24	25.75	19.36
	SimCLR (Chen, Kornblith, et al., 2020)	<u>46.40</u>	<u>24.38</u>	<u>22.15</u>	<u>71.25</u>	<u>35.52</u>	<u>30.61</u>
Non-contrastive Learning	Rotation (Gidaris et al., 2018)	23.61	10.80	8.31	66.47	31.60	25.11
	Jigsaw (Noroozi & Favaro, 2016)	26.84	12.01	8.73	16.85	5.70	2.02
	Context (Doersch et al., 2015)	27.11	12.46	9.80	41.20	16.88	9.75
	Jigsaw++ (Noroozi et al., 2018)	28.53	13.24	10.86	18.46	6.41	2.58
	BEiT v2[a] (Peng et al., 2022)	38.06	18.15	14.65	67.73	33.39	28.46
	MAE[a] (He et al., 2022)	42.41	20.28	15.79	70.66	34.62	28.97
	FullRot (Ours)	**47.36**	**23.40**	**19.64**	**74.82**	**38.10**	**33.73**
	FullRot + WRMix (Ours)	**48.77**	**24.42**	**20.74**	**75.54**	**38.59**	**34.58**

[a] Employ ViT as the backbone.

The evaluation metric is linear classification accuracy on the STL-10 test set. Moreover, the default setting for the self-supervised method is FullRot + WRMix unless specified otherwise. The other training and evaluating settings for classification follow those described in Section 4.1.

Impact of Crop Centre and Ratio. To investigate whether the random cropping centre and the random crop ratio can affect the pretext task, we also train and evaluate FullRot using the original centre and a fixed crop ratio of 0.8, respectively. The results illustrated in Table 4 show a 0.64% increase in accuracy by using a random centre and a 0.62% increment in accuracy by using a random crop ratio.

Impact of Crop Shape. Examining the effect of cropping an image into a circle versus a rectangle with multiples of 90°, it is found that maintaining a circular shape for the cropped region statistically improves FullRot's performance. As shown in Table 4, there is a 0.42% increase in accuracy on the STL-10 classification task. The improved results suggest that the rectangular shape introduces less complex and discriminative features near the cropped borderline, particularly at the four corners.

Table 4
The impact of four different crop schemes (✗: cancel setting, ✓: use setting).

Crop scheme		Accuracy
Random Centre	✗	86.60
	✓	**87.24** $^{\uparrow 0.64}$
Random Crop Ratio	✗	86.62
	✓	**87.24** $^{\uparrow 0.62}$
Cropped Into a Circle	✗	86.82
	✓	**87.24** $^{\uparrow 0.42}$
Background Removal	✗	85.10
	✓	**87.24** $^{\uparrow 2.14}$

Impact of Cropped Background. We also train and evaluate FullRot by retaining the background of the cropped region during self-supervised training, and the results can be found in the last column

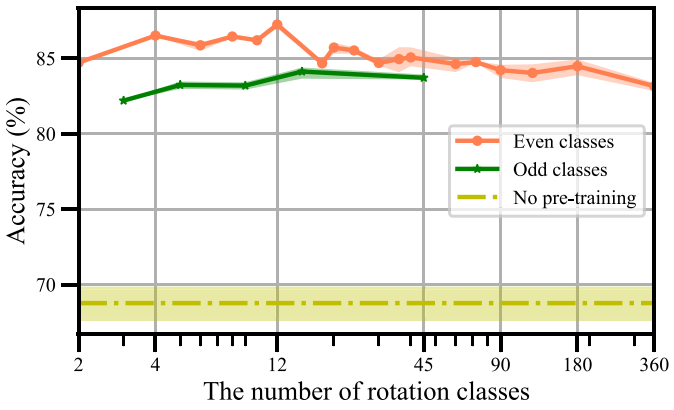

Fig. 4. The impact of the different number of rotation classes.

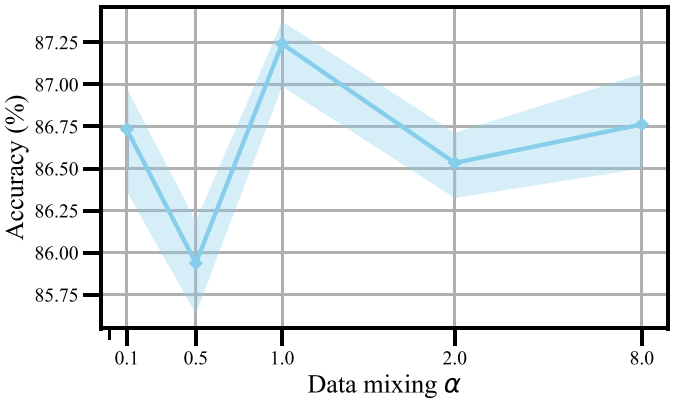

Fig. 5. The impact of α for image blending.

of Table 4. The presence of the background introduces dominant and irrelevant features of objects, which strongly correlates with features near the crop borderline, resulting in a 2.14% drop in accuracy.

The Number of FullRot Classes. To discover the most suitable number of FullRot classes, we train and evaluate the ResNet50 backbone with all possible rotation classes whose angles have integer values. The STL-10 classification results, shown in Fig. 4, illustrate that the performance of FullRot with even rotation classes is increasing from 84.72% (2 classes) to 87.24% (12 classes) and then dropping to 83.15% (360 classes). Similarly, the performance of FullRot with odd rotation classes increases from 82.20% (3 classes) to 84.12% (9 classes) and then decreases to 83.71% (45 classes). Besides, there is a strong positive correlation (correlation coefficient of 0.6392) between the number of even/odd rotation classes and prediction accuracy. Although the line of odd classes is under the even classes, FullRot, with any number of classes, still exceeds the non-pretrained baseline by >13% accuracy. More importantly, the majority of even classes between 4 to 24 show peak performance (>85.5% accuracy). Thus, the study selects 12 rotation classes as the default setting, reaching the best results of 87.24% accuracy.

The Number of Feeding Classes. To figure out the effect of the number of rotation classes feeding, we train and evaluate FullRot by randomly feeding 1, 2 (default), 4(33.33%), 6 (half), and 12 (all) rotation classes of each image on every training epoch. To keep the close number of input rotated images, the corresponding training epochs are 200, 100, 50, 34, and 17. The results are shown in Table 5. Although FullRot + WRMix secure higher accuracy than none-pretraining baseline (66.77% in Table 1) regardless of feeding classes, two feeding classes outperform all other options, achieving more than 0.40% higher accuracy on STL-10. Moreover, feeding only two classes also simplifies

Table 5
The impact of the number of images feeding on the experimental results.

# Feeding classes ⇑	Epochs	Feeding times	Accuracy
1	200	200	$85.87^{+0.20}_{-0.17}$
2	100	200	$\mathbf{87.24}^{+0.13}_{-0.25}$
4 (33.3%)	50	200	$85.75^{+0.36}_{-0.25}$
6 (50%)	34	204	$86.48^{+0.14}_{-0.20}$
12 (100%)	17	204	$86.84^{+0.08}_{-0.13}$

Table 6
The impact of different data mixture methods.

Data-mixing method	Accuracy ⇑
–	86.59
GMix (Park et al., 2022)	85.90 $^{\downarrow 0.69}$
Mixup (Zhang et al., 2018)	86.35 $^{\downarrow 0.24}$
CutMix (Yun et al., 2019)	86.47 $^{\downarrow 0.12}$
HMix (Park et al., 2022)	86.85 $^{\uparrow 0.26}$
WRMix (Ours)	**87.24** $^{\uparrow \mathbf{0.65}}$

Table 7
The impact of inter-image and intra-image blend.

Methods	κ ⇑	Accuracy
Inter-image Mixture	0	$86.95^{+0.31}_{-0.20}$
Hybrid Mixture	0.5	$86.76^{+0.19}_{-0.18}$
Intra-image Mixture	1	$\mathbf{87.24}^{+0.13}_{-0.25}$

the choice of the maximum training epoch regardless of the total number of rotation classes d. Thus, it is recommended to load two rotation classes on every epoch.

The Method of Data Mixture. To demonstrate the efficacy of WRMix, four data mixture methods, including Mixup (Zhang et al., 2018), CutMix (Yun et al., 2019), HMix (Park et al., 2022), and GMix (Park et al., 2022), are compared in this section. As indicated in Table 6, Mixup, CutMix, and GMix lead to lower classification accuracy (<86.59%) than pretraining without mingling data. More importantly, WRMix delivers +0.39% ∼ +1.34% classification accuracy on STL-10 compared to SOTA methods. Such results illustrate that there is a significant performance improvement in representation learning because of WRMix.

Inter-image vs. Intra-image Mixture. Setting κ as 0, 0.5, 1 represents the possibility of intra-image mixing ($1 - \kappa$ for inter-image synthesising). As shown in Table 7, intra-image blending with $\kappa = 1$ outperforms other κ values, showing a > 0.25% accuracy improvement.

Beta Distribution Parameter α. α is the parameter that affects the combination weight λ_1 and size of cropping mask (see Section 3.2). When varying the values of α (0.5, 1, 2, and 8), the results, represented in Fig. 5, reveal that $\alpha = 1$ secures the highest classification accuracy of 87.24% on STL-10 test set, outperforming other α values by more than 0.48%. This indicates that the uniform distribution $\mathcal{U}_{[0,1]}$ is favourable.

4.4. Visualisation

Implementation Details. To gain a deeper insight into the effect of the different SSL methods, we visualise the learned features using t-distributed stochastic neighbour embedding (t-SNE) (Van der Maaten & Hinton, 2008). To figure out the regions of interest for vanilla rotation and FullRot (either with WRMix or without WRMix) and the performance gap between the odd and even number of rotation classes, we extract the CAM from the final layer of the backbone by using LayerCAM (Jiang, Zhang, Hou, Cheng, & Wei, 2021) on the TorchCAM tool (Fernandez, 2020).

Feature Maps. It can be observed from the t-SNE results shown in Fig. 6 that all SSL methods exhibit a more diverse distribution of

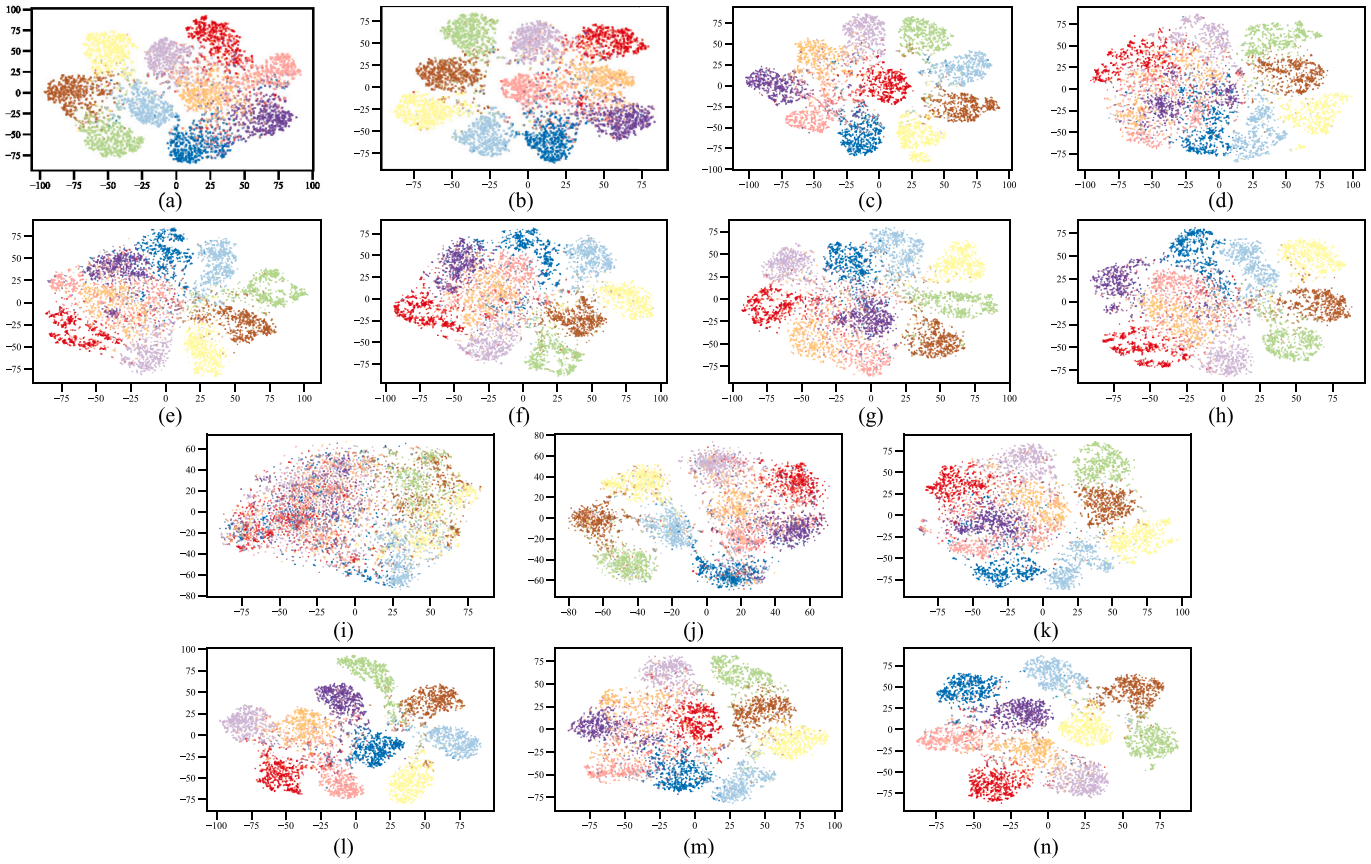

Fig. 6. The comparison of t-SNE plots for STL feature generated by (a) SimSiam, (b) MoCo v2, (c) SimCLR, (d) Barlow, (e) Non-pretrained CNN baseline, (f) Jiasaw++, (g) Jigsaw, (h) Context, (i) Non-pretrained ViT baseline, (j) MAE, (k) BEiT v2, (l) Vanilla rotation, (m) FullRot (ours), and (n) FullRot + WRMix (ours).

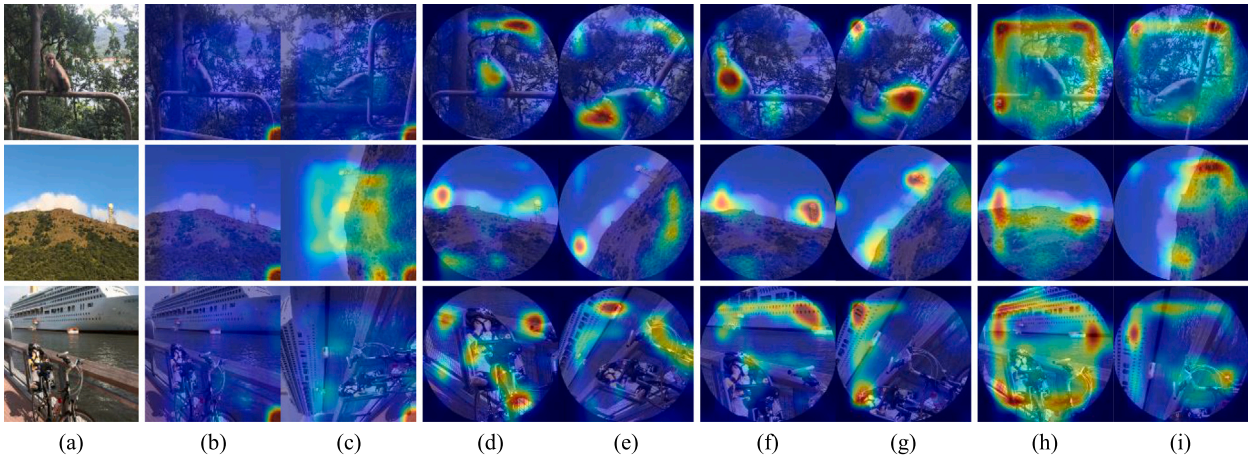

Fig. 7. The comparison of CAM for (a) input images from methods (b-c) vanilla rotation, FullRot (d-e), and FullRot + WRMix with (f-g) the even number of rotation classes and (h-i) the odd number of rotation classes. The rotation degree for (b) & (d) & (f) & (h) is 0°, (c) is 90°, (e) & (g) is 60°, and (i) is 80°.

features compared to the non-pretrained baseline. Notably, FullRot without (Fig. 6m) or with (Fig. 6n) WRMix reveals a closer distribution of intra-class features among tested SSL methods, which suggests that FullRot and WRMix encouraged the network to treat the underestimated regions equally crucial as other regions, leading to more balanced and robust feature representations.

Attention Maps. The examples of CAM presented in Fig. 7. For vanilla rotation (Fig. 7bc), it can be observed that the focus is mainly confined to a small region in the lower-left corner of the image. In contrast, FullRot (Fig. 7de) exhibits a more concentrated region closely related to the objects in the images, indicating that FullRot can capture more relevant and informative regions for representation learning than the conventional method. Moreover, FullRot with WRMix (Fig. 7fg) highlights the objects (*e.g.*, the monkey in the first-row image) in the images more prominently than FullRot without WRMix, while minimising the emphasis on the background.

Fig. 7hi illustrates the CAM of FullRot + WRMix with odd classes, covering a more extensive region than even classes, which may be the main reason for the significant performance gap between the odd and even number of rotation classes. The absence of a point reflection in odd classes simplifies the pretext task, leading to a different distribution of attention in the learned representations.

5. Discussion

The narrow margin of 1.73% linear accuracy (SimSiam) and 2.47% 5-nn accuracy (SimCLR) (see Table 1) on the CIFAR-100 test set and 0.05% mIoU (SimCLR) (see Table 2) on the TikTokDances test set between FullRot + WRMix and the leading methods suggests that FullRot + WRMix is a competitive approach. The difference in performance could be attributed to CIFAR-100 and TikTokDances having ten times and one-tenth times classes as STL-10 unlabelled set, respectively, resulting in a significant domain shift. To address this, it may be beneficial to supplement the self-supervised pretraining dataset with additional data or explore ways to reduce the domain difference between the CIFAR-100/TikTokDances and STL-10 datasets.

On the other hand, FullRot + WRMix seizes the top-tier performance on skin lesions classification on PAD-UFES-20 (72.54% linear and 73.19% 5-nn accuracy in Table 1) and segmentation on ISIC 2018 (87.46% mIoU in Table 2). This indicates its potential for practical application in label-efficient medical image analysis tasks.

Moreover, the training times of FullRot with and without WRMix are similar, with a negligible difference of only 0.5% in total times (5.73 h versus 5.76 h), as WRMix does not increase the number of feeding images. Additionally, as a mixed sample data augmentation method, WRMix holds enormous potential to be developed as a general data augmentation technique for enhancing image representation learning.

Further investigation is warranted to explore the effectiveness of inter-image and intra-image mingling in different circumstances and datasets due to only a minor margin among different circumstances (<0.5% linear accuracy in Table 7).

Furthermore, our concept of full rotation could be extended to sphere rotation for three-dimensional data, such as point clouds. The experimental designs and discussions presented in this work could serve as a reference for future development of cubic rotation with multiple 90° angles and sphere rotation. These could potentially evolve into fundamental data augmentation methods for 3D data in the future. Additionally, WRMix also possesses the potential to be integrated with weighted and rotated regions of 3D data.

6. Conclusion

In this paper, we have conducted a comprehensive study on self-supervised representation learning using rotation transformations as a pretext task. We have identified that the conventional rotation pretext task lacks effectiveness in capturing features near the centre of images. To overcome this limitation, we have proposed FullRot, a novel approach that involves random cropping of an arbitrary region with a random crop ratio. Moreover, we have introduced circular cropping and rotation of the cropped region to any degree to enhance the complexity of the pretext task. Besides, we have explored different mixed sample data augmentation methods and introduced WRMix, a novel approach for blending two intra-image patches. Equipped with these innovative techniques, FullRot + WRMix has achieved state-of-the-art performance on ten benchmark datasets. Notably, FullRot + WRMix has demonstrated superior performance compared to the conventional rotation pretext task on ten benchmark datasets and three vision tasks with significant improvements (*e.g.*, +6.70% linear accuracy on STL-10, +2.53% 5-nn accuracy on CIFAR-10, +5.67% linear accuracy on CIFAR-100, +2.80% 5-nn accuracy on Sports-100, +1.77% 5-nn accuracy on Mammals-45, +3.17% 5-nn accuracy on PAD-UFES-20, +14.61% mIoU on VOC 2012, 1.79% mIoU on ISIC 2018, +0.76% mIoU on FloodArea, +25.16% AP_{50} on VOC 2007, and +9.07% AP_{50} on UTDAC 2020). Overall, the findings of this study highlight the effectiveness of FullRot and WRMix and their potential for improving self-supervised feature learning.

Table 8
Training time cost of self-supervised training for the tested methods on the STL-10 unlabelled set.

Self-supervision Method		Time cost/h
Contrastive Learning	Barlow (Zbontar et al., 2021)	6.00
	SimCLR (Chen, Kornblith, et al., 2020)	6.02
	SimSiam (Chen & He, 2021)	5.59
	MoCo v2 (Chen, Fan, et al., 2020)	3.79
Non-contrastive Learning	MAE (He et al., 2022)	4.16
	Jigsaw (Noroozi & Favaro, 2016)	4.07
	Context (Doersch et al., 2015)	3.58
	BEiT v2 (Peng et al., 2022)	4.93
	Rotation (Gidaris et al., 2018)	11.21
	Jigsaw++ (Noroozi et al., 2018)	13.67
	FullRot (Ours)	5.73
	FullRot + WRMix (Ours)	5.76

CRediT authorship contribution statement

Wei Dai: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Validation, Visualization, Writing – original draft, Writing – review & editing. **Tianyi Wu:** Conceptualization, Writing – review & editing. **Rui Liu:** Data curation. **Min Wang:** Validation. **Jianqin Yin:** Conceptualization, Supervision, Writing – original draft, Writing – review & editing. **Jun Liu:** Conceptualization, Supervision, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Jun Liu reports financial support was provided by Research Grant Council (RGC) of Hong Kong. Jun Liu reports financial support was provided by Guangdong Province Basic and Applied Basic. Jun Liu reports financial support was provided by Science and Technology Innovation Committee of Shenzhen.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported by the Research Grant Council (RGC) of Hong Kong under Grant 11217922, 11212321 and Grant ECS-21212720, Guangdong Province Basic and Applied Basic Research Fund Project 2019A1515110175, and the Science and Technology Innovation Committee of Shenzhen under Grant Type-C SGDX20210823104001011.

Appendix

Training time cost

In real-world applications, computational resources are often limited. To assess the computational cost of tested models, we measure the time during self-supervised training. The results, presented in Table 8, indicate that 75% of the tested methods require less than 6 h during training. FullRot with or without WRMix integration also exhibits less than 6 h, 5.76 or 5.73 h, respectively, with only a marginal difference of 0.03 h. Moreover, by only utilising two rotation classes per epoch, FullRot significantly reduces training time, saving over 50% compared to vanilla rotation.

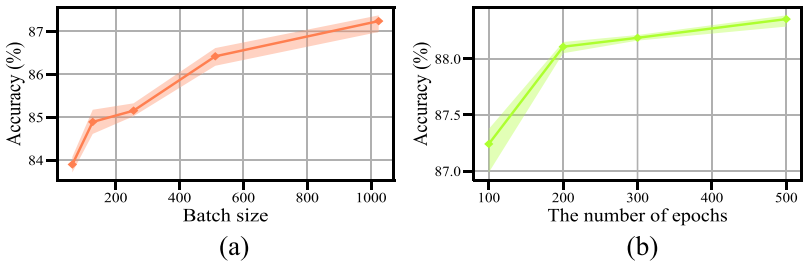

Fig. 8. The impact of (a) batch size and (b) the number of epochs of self-supervised pretraining on the performance.

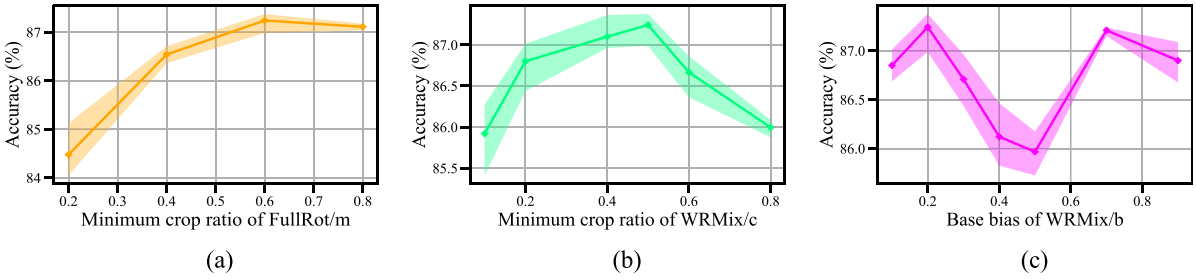

Fig. 9. The impact of (a) minimum crop ratio m of FullRot, (b) minimum crop ratio c of WRMix, and (c) base bias b of WRMix.

Table 9
The impact of batch shuffle and mask rotation (✗: cancel setting, ✓: use setting).

Ablation Item		Accuracy
Batch Shuffle	✗	86.59
	✓	**87.24** $^{\uparrow 0.65}$
Mask Rotation	✗	86.76
	✓	**87.24** $^{\uparrow 0.48}$

Extracurricular ablation

Implementation Details. For the implementation of self-supervised training and validation, we followed the settings described in Section 4.3, unless specified otherwise.

Batch Size. The choice of batch size has a significant impact on the performance of self-supervised learning methods (Chen & He, 2021; Chen, Kornblith, et al., 2020). To understand the effect of different batch sizes on the FullRot + WRMix approach, we conducted experiments. Due to the memory restriction of 24 GB on the RTX3090, the maximum batch size we could use was 1024 images per iteration. As shown in Fig. 8a, the experimental results indicate a gradual improvement in performance as the batch size increases from 64 to 1024 images, with accuracy rising from 83.90% to 87.24%. Such results suggest that a larger batch size provides more general information about image objects compared to a relatively small batch size, thereby enhancing the effectiveness of FullRot + WRMix.

Pretraining Epochs. The number of epochs used for self-supervised learning is an essential factor consideration in terms of computational cost. We conducted experiments to analyse the impact of the number of pretraining epochs on performance. The experimental results are presented in Fig. 8b and indicate a significant increase in performance, with a gain of over 0.8% classification accuracy, as the number of pretraining epochs increases from 100 to 200. Subsequently, the accuracy shows a slower growth, with an increase from 88.11% to 88.35% when the number of pretraining epochs reaches 500. To balance computational resources and maintain a competitive performance, we chose 200 epochs for self-supervised pretraining.

Batch Shuffle. To mitigate the impact of loading the rotated images from the same source in adjacent locations within a batch, we implemented a batch shuffle mechanism. Batch shuffling ensures that all rotated images within a batch are randomly rearranged. As shown in Table 9, when the batch shuffle is disabled, we observed a linear accuracy decrease of 0.65%. Therefore, the order of rotated images is irrelevant information and can negatively affect the representation learning performance of FullRot + WRMix.

Mask Rotation of WRMix. In conventional methods such as Mixup, CutMix, and HMix, the orientation of the masked region for image synthesis remains constant. However, by introducing random mask rotation, a more diverse combination of image patches can be achieved. The experimental results are demonstrated in the last column in Fig. 3, where the mingling mask of two patches is rotated. By applying rotation to the mask, it can be reshaped as an irregular polygon, further enhancing the diversity of the composite image. The results presented in Table 9 indicate that mask rotation boosts +0.48% linear accuracy increment for FullRot + WRMix.

Crop Ratio of FullRot. The minimum crop ratio, denoted as m, for FullRot is discussed in Section 3.1. The impact of varying m on the STL-10 classification accuracy can be observed in Fig. 9a. As m increases from 0.2 to 0.6, the accuracy improves from 84.48% to 87.24%, but then drops to 87.11% when m is reduced to 0.8. Based on these results, we have chosen the default value of m as 0.6.

Crop Ratio of WRMix. The minimum crop ratio of WRMix, denoted as c, for WRMix is described in Eq. (6). Fig. 9b presents the results of the STL-10 classification accuracy as c is varied. It shows that as c increases from 0.1 to 0.5, the accuracy improves from 85.92% to 87.24%, but declines to 86.00% when c reaches 0.8. The inverted U shape of Fig. 9b indicates that neither a small (c = 0.8) nor extensive (c = 0.2) range of the mask region of WRMix yields satisfactory results. To ensure an appropriate diversity of mask side lengths, we recommend maintaining c between 0.2 and 0.6. Among the tested values, we have selected 0.5 as the optimal value for c, as it corresponds to the highest performance.

Base Bias of WRMix. The base bias, denoted as b, for WRMix is defined in Eq. (7). The impact of varying b on the performance of FullRot + WRMix can be observed in Fig. 9c. The line graph in Fig. 9c exhibits an 'm' shape, indicating that the most satisfactory performance, with classification accuracies of 87.24% and 87.21%, can be achieved by using either a b value of 0.2 or 0.7. These results suggest that

the influence of amalgamated area on λ_1 should be maintained within specific regions. Specifically, either a small (b = 0.2) or large (b = 0.7) value for the base bias can yield the best performance. We have chosen 0.2 as the optimal one from all tested values for b.

References

Asaniczka (2023). *Mammals Image Classification Dataset (45 Animals)*. Kaggle, http://dx.doi.org/10.34740/KAGGLE/DS/3999173, URL https://www.kaggle.com/ds/3999173.

Bao, H., Dong, L., Piao, S., & Wei, F. (2021). BEiT: BERT pre-training of image transformers. In *International conference on learning representations*.

Bucci, S., Loghmani, M. R., & Tommasi, T. (2020). On the effectiveness of image rotation for open set domain adaptation. In *Proceedings of the European conference on computer vision* (pp. 422–438).

Chen, X., Fan, H., Girshick, R., & He, K. (2020). Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297.

Chen, X., & He, K. (2021). Exploring simple Siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 15750–15758).

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR.

Chen, T., Zhai, X., Ritter, M., Lucic, M., & Houlsby, N. (2019). Self-supervised GANs via auxiliary rotation loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 12154–12163).

Chen, H., Zhou, Y., Li, J., Wei, X., & Xiao, L. (2022). Self-supervised multi-category counting networks for automatic check-cut. *IEEE Transactions on Image Processing*, *31*, 3004–3016.

Chen, L., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision* (pp. 801–818).

Coates, A., Ng, A., & Lee, H. (2011). An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 215–223). JMLR Workshop and Conference Proceedings.

Codella, N., Rotemberg, V., Tschandl, P., Celebi, M. E., Dusza, S., Gutman, D., et al. (2019). Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (ISIC). arXiv preprint arXiv:1902.03368.

DeVries, T., & Taylor, G. W. (2017). Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552.

Doersch, C., Gupta, A., & Efros, A. A. (2015). Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision* (pp. 1422–1430).

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In *International conference on learning representations*.

Everingham, M., Eslami, S., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2015). The PASCAL visual object classes challenge: A retrospective. *International Journal of Computer Vision*, *111*(1), 98–136.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision*, *88*, 303–338.

Feng, Z., Xu, C., & Tao, D. (2019). Self-supervised representation learning by rotation feature decoupling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10364–10374).

Fernandez, F. G. (2020). TorchCAM: Class activation explorer. https://github.com/frgfm/torch-cam.

Gidaris, S., Singh, P., & Komodakis, N. (2018). Unsupervised representation learning by predicting image rotations. In *International conference on learning representations*.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., & Girshick, R. (2022). Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 16000–16009).

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9729–9738).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Huang, C., Xu, Q., Wang, Y., Wang, Y., & Zhang, Y. (2022). Self-supervised masking for unsupervised anomaly detection and localization. *IEEE Transactions on Multimedia*.

Jiang, P., Zhang, C., Hou, Q., Cheng, M., & Wei, Y. (2021). LayerCAM: Exploring hierarchical class activation maps for localization. *IEEE Transactions on Image Processing*, *30*, 5875–5888.

Kalantidis, Y., Sariyildiz, M. B., Pion, N., Weinzaepfel, P., & Larlus, D. (2020). Hard negative mixing for contrastive learning. In *Advances in Neural Information Processing Systems (NIPS): vol. 33*, (pp. 21798–21809).

Karim, F., Sharma, K., & Barman, N. R. (2022). *Flood area segmentation*. Kaggle, URL https://www.kaggle.com/datasets/faizalkarim/flood-area-segmentation.

Kim, D., Cho, D., Yoo, D., & Kweon, I. S. (2018). Learning image representations by completing damaged jigsaw puzzles. In *IEEE winter conference on applications of computer vision* (pp. 793–802). IEEE.

Kim, J.-H., Choo, W., & Song, H. O. (2020). Puzzle mix: Exploiting saliency and local statistics for optimal mixup. In *International conference on machine learning* (pp. 5275–5285). PMLR.

Kim, S., Lee, G., Bae, S., & Yun, S.-Y. (2020). MixCo: Mix-up contrastive learning for visual representation. arXiv preprint arXiv:2010.06300.

Krizhevsky, A. (2009). *Learning multiple layers of features from tiny images* (Ph.D. thesis).

Lee, K., Zhu, Y., Sohn, K., Li, C.-L., Shin, J., & Lee, H. (2021). *i*-Mix: A domain-agnostic strategy for contrastive representation learning. In *International conference on learning representations*.

Li, Y., Mao, H., Girshick, R., & He, K. (2022). Exploring plain vision transformer backbones for object detection. In *European conference on computer vision* (pp. 280–296). Springer.

Lim, J. Y., Lim, K. M., Lee, C. P., & Tan, Y. X. (2023). SCL: Self-supervised contrastive learning for few-shot image classification. *Neural Networks*.

Liu, J., Li, B., Lei, M., & Shi, Y. (2022). Self-supervised knowledge distillation for complementary label learning. *Neural Networks*, *155*, 318–327.

Liu, Z., Li, S., Wang, G., Tan, C., Wu, L., & Li, S. Z. (2022). Decoupled mixup for data-efficient learning. arXiv preprint arXiv:2203.10761.

Liu, Z., Li, S., Wu, D., Liu, Z., Chen, Z., Wu, L., et al. (2022). AutoMix: Unveiling the power of mixup for stronger classifiers. In *Proceedings of the European conference on computer vision* (pp. 441–458). Springer.

Loshchilov, I., & Hutter, F. (2017). SGDR: Stochastic gradient descent with warm restarts. In *International conference on learning representations*.

Loshchilov, I., & Hutter, F. (2018). Decoupled weight decay regularization. In *International conference on learning representations*.

Mazumder, P., Singh, P., & Namboodiri, V. P. (2021). Improving few-shot learning using composite rotation based auxiliary task. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 2654–2663).

Mehta, S., Abdolhosseini, F., & Rastegari, M. (2022). CVNets: High performance library for computer vision. arXiv preprint arXiv:2206.02002.

Mehta, S., & Rastegari, M. (2021). MobileViT: Light-weight, general-purpose, and mobile-friendly vision transformer. In *International conference on learning representations*.

Misra, I., & Maaten, L. v. d. (2020). Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 6707–6717).

Noroozi, M., & Favaro, P. (2016). Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proceedings of the European conference on computer vision* (pp. 69–84).

Noroozi, M., Pirsiavash, H., & Favaro, P. (2017). Representation learning by learning to count. In *Proceedings of the IEEE international conference on computer vision* (pp. 5898–5906).

Noroozi, M., Vinjimoor, A., Favaro, P., & Pirsiavash, H. (2018). Boosting self-supervised learning via knowledge transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 9359–9367).

Pacheco, A. G., Lima, G. R., Salomao, A. S., Krohling, B., Biral, I. P., de Angelo, G. G., et al. (2020). PAD-UFES-20: A skin lesion dataset composed of patient data and clinical images collected from smartphones. *Data in Brief*, *32*, Article 106221.

Park, C., Yun, S., & Chun, S. (2022). A unified analysis of mixed sample data augmentation: A loss function perspective. In *Advances in neural information processing systems*.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, *32*, 1–12.

Peng, Z., Dong, L., Bao, H., Ye, Q., & Wei, F. (2022). Beit v2: Masked image modeling with vector-quantized visual tokenizers. arXiv preprint arXiv:2208.06366.

Piosenka, G. (2022). *100 Sports image classification*. Kaggle, URL https://www.kaggle.com/datasets/gpiosenka/sports-classification.

Qing, Y., Zeng, Y., Cao, Q., & Huang, G.-B. (2021). End-to-end novel visual categories learning via auxiliary self-supervision. *Neural Networks*, *139*, 24–32.

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(06), 1137–1149.

Roman, K. (2019). *Human segmentation dataset - TikTok dances*. Kaggle, URL https://www.kaggle.com/datasets/tapakah68/segmentation-full-body-tiktok-dancing-dataset.

Shen, Z., Liu, Z., Liu, Z., Savvides, M., Darrell, T., & Xing, E. (2022). Un-mix: Rethinking image mixtures for unsupervised visual representation learning. *vol. 36*, In *Proceedings of the AAAI conference on artificial intelligence* (2), (pp. 2216–2224).

Song, P., Li, P., Dai, L., Wang, T., & Chen, Z. (2023). Boosting R-CNN: Reweighting R-CNN samples by RPN's error for underwater object detection. *Neurocomputing*, *530*, 150–164.

Tschandl, P., Rosendahl, C., & Kittler, H. (2018). The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data*, *5*(1), 1–9.

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(11).

Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D., et al. (2019). Manifold mixup: Better representations by interpolating hidden states. In *International conference on machine learning* (pp. 6438–6447). PMLR.

Vu, T., Sun, B., Yuan, B., Ngai, A., Li, Y., & Frahm, J.-M. (2023). LossMix: Simplify and generalize mixup for object detection and beyond. arXiv preprint arXiv:2303.10343.

Wang, J., Gao, Y., Li, K., Lin, Y., Ma, A. J., Cheng, H., et al. (2021). Removing the background by adding the background: Towards background robust self-supervised video representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11804–11813).

Wang, Y., Zhuo, W., Li, Y., Wang, Z., Ju, Q., & Zhu, W. (2022). Fully self-supervised learning for semantic segmentation. arXiv preprint arXiv:2202.11981.

Wu, Y., Kirillov, A., Massa, F., Lo, W., & Girshick, R. (2019). Detectron2. https://github.com/facebookresearch/detectron2.

Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., & Yoo, Y. (2019). CutMix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 6023–6032).

Zbontar, J., Jing, L., Misra, I., LeCun, Y., & Deny, S. (2021). Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning* (pp. 12310–12320). PMLR.

Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. (2018). Mixup: Beyond empirical risk minimization. In *International conference on learning representations*.

Zhang, Y., Gong, M., Li, J., Zhang, M., Jiang, F., & Zhao, H. (2022). Self-supervised monocular depth estimation with multiscale perception. *IEEE Transactions on Image Processing*, *31*, 3251–3266.