Pattern Recognition Letters 166 (2023) 53–60

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Jigsaw-ViT: Learning jigsaw puzzles in vision transformer

Yingyi Chen [a,*], Xi Shen [b,*], Yahui Liu [c], Qinghua Tao [a], Johan A.K. Suykens [a]

[a] ESAT-STADIUS, KU Leuven, Belgium
[b] Tencent AI Lab, Shenzhen, China
[c] University of Trento, Italy

ARTICLE INFO

Article history:
Received 23 August 2022
Revised 29 November 2022
Accepted 26 December 2022
Available online 28 December 2022

Edited by Jiwen Lu

Keywords:
Vision transformer
Jigsaw puzzle
Image classification
Label noise
Adversarial examples

ABSTRACT

The success of Vision Transformer (ViT) in various computer vision tasks has promoted the ever-increasing prevalence of this convolution-free network. The fact that ViT works on image patches makes it potentially relevant to the problem of jigsaw puzzle solving, which is a classical self-supervised task aiming at reordering shuffled sequential image patches back to their original form. Solving jigsaw puzzle has been demonstrated to be helpful for diverse tasks using Convolutional Neural Networks (CNNs), such as feature representation learning, domain generalization and fine-grained classification. In this paper, we explore solving jigsaw puzzle as a self-supervised auxiliary loss in ViT for image classification, named Jigsaw-ViT. We show two modifications that can make Jigsaw-ViT superior to standard ViT: discarding positional embeddings and masking patches randomly. Yet simple, we find that the proposed Jigsaw-ViT is able to improve on both generalization and robustness over the standard ViT, which is usually rather a trade-off. Numerical experiments verify that adding the jigsaw puzzle branch provides better generalization to ViT on large-scale image classification on ImageNet. Moreover, such auxiliary loss also improves robustness against noisy labels on Animal-10N, Food-101N, and Clothing1M, as well as adversarial examples. Our implementation is available at https://yingyichen-cyy.github.io/Jigsaw-ViT.

1. Introduction

Vision Transformer (ViT) [1] is an architecture inherited from Natural Language Processing [2] while applied to image classification with taking raw image patches as inputs. Different from classical Convolutional Neural Networks (CNNs), the architectures of ViTs are based on self-attention modules [2], which aim at modeling global interactions of all pixels in feature maps. More precisely, ViTs take sequential image patches as inputs, and the attention mechanism enables interaction and aggregation directly among patch information. Therefore, compared to CNNs where image features are progressively learnt from local to global context via reducing spatial resolution, ViT enjoys obtaining global information from the very beginning. Up till now, such convolution-free networks have been achieving great success on various computer vision tasks, including image classification [3–8], object detection [6,9,10], semantic segmentation [9–11] and image generation [12], etc.

The fact that ViTs work on image patches makes it potentially relevant to one classical image patch-based learning task, that is, jigsaw puzzle solving. Solving jigsaw puzzle aims at reordering shuffled sequential image patches back to their original form. In practice, the problem is interesting for cultural heritage and archaeology to search the correct configuration given numerous fragments of an art masterpiece [13]. However, in the Computer Vision community, the most interesting aspect could be that it provides *off-the-shelf* annotations for free considering a given image. Despite its simplicity, it has shown effectiveness in diverse Computer Vision tasks based on CNNs such as: self-supervised feature representation learning [14], domain generalization [15] and fine-grained classification [16]. Motivated by the fact that both jigsaw puzzle solving and ViT share the same basis of learning from image patches, we consider incorporating solving jigsaw puzzle to ViT for image classification tasks.

In this paper, we explore leveraging the jigsaw puzzle solving problem as a self-supervised auxiliary loss of a standard ViT, named Jigsaw-ViT. Precisely, as shown in Fig. 1, in addition to the standard classification flow in the end-to-end training, we add a jigsaw flow whose goal is to predict the absolute positions of the input patches by solving a classification problem. Notably, we make

* Corresponding author.
E-mail addresses: yingyi.chen@esat.kuleuven.be (Y. Chen), tisonshen@tencent.com (X. Shen).

https://doi.org/10.1016/j.patrec.2022.12.023

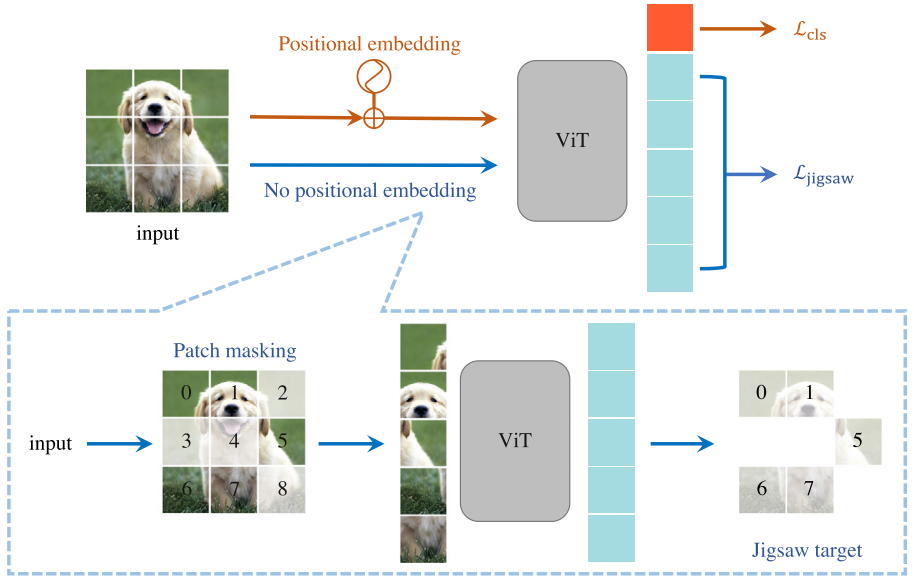

Fig. 1. Overview framework of our Jigsaw-ViT. (Top) We incorporate jigsaw puzzle solving (in blue flow) into the standard ViT for image classification (in red). During the training, we jointly learn the two tasks. (Bottom) The details of our jigsaw puzzle flow. We mask some patches, i.e., patch masking, and remove positional embeddings when feeding patches to ViT. For each unmasked patch, the model should predict the class corresponding to the patch position.

two important modifications compared to the naive jigsaw puzzle when feeding input patches to ViTs: *(i)* we get rid of the positional embeddings in the jigsaw flow, by which we prevent the model from cheating from explicit clues in the positional embeddings; *(ii)* we randomly mask some input patches, i.e., *patch masking*, and then aim at predicting only the positions of those unmasked patches, hence making the prediction rely on global context rather than several particular patches.

Despite its simplicity, we find that our Jigsaw-ViT is able to improve on both generalization and robustness over the standard ViT, which is usually rather a trade-off [17]. To be specific, in terms of generalization, we observe a steady increase in classification accuracy on ImageNet-1K [18] that our jigsaw flow brings to the ViTs. As for robustness, we first show that the proposed jigsaw flow provides consistent improvement against noisy labels on three important real-world benchmarks, i.e., Animal-10N [19], Food-101N [20] and Clothing1M [21]. Then, we show that our proposed Jigsaw-ViTs can effectively enhance the robustness of ViTs against adversarial attacks in both black-box and white-box attack settings.

To summarize, our contributions are as follows: *First*, we propose to introduce the jigsaw puzzle solving task into ViT-based models, namely Jigsaw-ViT, with two techniques: removing positional embeddings, and randomly masking patches. *Second*, empirical results suggest that our jigsaw flow not only improves the generalization ability of ViTs on large-scale image classification, but also the robustness against label noise and adversarial examples. Our implementation is available at https://yingyichen-cyy.github.io/Jigsaw-ViT.

2. Related work

Solving jigsaw puzzle in CNNs Solving jigsaw puzzles aims at recovering an original image from its shuffled patches, which is a classical pattern recognition problem dating back to [22]. Rather than setting the jigsaw puzzle solving as the ultimate goal [23], nowaday works treat solving jigsaw puzzles as a pre-text task to other visual recognition tasks [15,16]. These methods assume that rich feature representations could be learnt in a self-supervised manner, which would be useful for fine-tuning with task-specific data. For example, [15] solves classification and jigsaw together to improve semantic understanding for domain generalization tasks, and [16] combines jigsaw puzzles and the progressive training for fine-grained classification. These methods use full sequential image patches and are built in the context of CNNs, while here we randomly mask image patches and introduce jigsaw naturally in the context of ViTs.

Vision Transformers Transformer proposed in [2] originally designed for natural language processing has shown promising performance for Computer Vision tasks [1,4–6]. Vision Transformers (ViTs) [1] directly inherit from transformer with image patch sequences as inputs, and have achieved superior performance than their counterpart CNNs for various tasks [9–11,24]. The success of ViTs has also encouraged the emergence of a wide variety of ViT variants [3–5,25]. One of the most representative works is DeiT [3], which introduces a distillation token and a teacher-student strategy specific to transformers, leading to competitive performance on ImageNet [18]. Although ViTs are convolution-free, recent works [6,26,27] also build stronger ViT variants by resembling some merits from CNNs. Notably, our work is different from [24] since [24] does not include jigsaw puzzle solving in the optimization goal, where they only shuffle patches in the triplet loss branch.

Learning with noisy labels The task aims at learning models achieving good clean test accuracy while being trained on noisy annotated data. Mainstream solutions include: label correction which corrects possibly wrong labels with more consistent substitutes [28,29], semi-supervised learning which trains networks in a semi-supervised manner with only the clean labels used [30], and sample reweighting which assigns more weights to samples possibly clean [31]. In particular, for sample reweighting, Co-teaching [31] is a classical method that cross-updates its two base networks on the small-loss samples selected by its peer. Based on this, Nested Co-teaching [32] improves performance by including compression regularization during the training.

Adversarial examples Deep neural networks are fragile to adversarial examples [33] where human-imperceptible perturbations on clean images can cheat the network to give wrong predictions. These adversarial examples can be generated in the white-box settings where attacker has full access to information inside the target model. Mainstream attacks include Fast Gradient Sign Method (FGSM) [34], projected gradient descent (PGD) [35], the

ensemble auto-attack (AA) [36], etc. In real-world scenarios, adversarial attacks are commonly done in black-box settings since the much information, e.g., gradients, of the target models are hard to obtain. Existing black-box attacks are mostly conducted either in query-based [37] or transfer-based ways [38]. The former relies on querying the outputs of the target models, while the latter uses surrogate models to generate adversarial examples. Recent work [39] also studies the adversarial robustness of ViTs where an ensemble of ViTs and CNNs can achieve good robustness.

3. Method

In this section, we present details of the proposed Jigsaw-ViT. A brief introduction to ViT [1] is firstly given in Section 3.1. Then, we present our Jigsaw-ViT in Section 3.2.

3.1. Vision transformer

Given an image $\mathbf{I} \in \mathbb{R}^{H\times W\times C}$, where H and W are spatial dimensions and C denotes the number of channels, we first divide the image into a sequence of non-overlapped 2D patches $\mathbf{I}_p \in \mathbb{R}^{L\times(P\times P\times C)}$ where the patch resolution is $P\times P$ and $L = HW/P^2$ is the sequence length (the number of patches). In ViT [1], these patches are linearly projected to D-dimensional features used as the patch embeddings $[\mathbf{z}_0^1,\ \mathbf{z}_0^2\ ...\ \mathbf{z}_0^L]^T \in \mathbb{R}^{L\times D}$. A learnable class token denoted by *CLS*, *i.e.*, $\mathbf{z}_0^{\text{cls}} \in \mathbb{R}^D$ is prepended to the sequence of the patch embeddings $\mathbf{z}_0$ leading to $\mathbf{z}_0 = [\mathbf{z}_0^{\text{cls}},\ \mathbf{z}_0^1,\ \mathbf{z}_0^2\ ...\ \mathbf{z}_0^L]^T \in \mathbb{R}^{(L+1)\times D}$. Usually, positional embeddings $\mathbf{p} = [\mathbf{p}^{\text{cls}},\ \mathbf{p}^1,\ \mathbf{p}^2\ ...\ \mathbf{p}^L]^T \in \mathbb{R}^{(L+1)\times D}$ are added to the sequential patch embeddings, thus $\mathbf{v}_0 = \mathbf{z}_0 + \mathbf{p}$ serves as the input to the Transformer encoder.

The Transformer encoder [2] consists of alternating layers of layer normalization, multi-head self-attention and multi-layer perceptron blocks, denoted as $LN(\cdot)$, $MSA(\cdot)$ and $MLP(\cdot)$, respectively. For an encoder with N layers, the final class prediction $\mathbf{y}_{\text{pred}}$ is the final embedding associated to the class token $\mathbf{v}_N^{\text{cls}}$, such that

$$\begin{aligned} \mathbf{v}_i' &= MSA(LN(\mathbf{v}_{i-1})) + \mathbf{v}_{i-1}, \quad i \in [1,2,...,N] \\ \mathbf{v}_i &= MLP(LN(\mathbf{v}_i')) + \mathbf{v}_i', \quad i \in [1,2,...,N] \\ \mathbf{y}_{\text{pred}} &= MLP(LN(\mathbf{v}_N^{\text{cls}})). \end{aligned} \tag{1}$$

3.2. Jigsaw-ViT

Solving jigsaw puzzles aims at reordering shuffled sequential patches back to their original format. Previous CNN-based methods have proved that various computer vision tasks are benefited from learning jigsaw puzzles [14–16]. In this section, we show how to incorporate a jigsaw puzzle flow into the regular end-to-end training of standard ViTs.

The overview of our approach is illustrated in Fig. 1 where image classification problem using ViT is the focus in this paper. More precisely, our goal is to train a ViT model that jointly considers solving the standard classification and jigsaw puzzles in its optimization objective. Accordingly, the total loss $\mathcal{L}_{\text{total}}$ simultaneously involves two cross-entropy losses (*CE*s), i.e., the class prediction loss on the class token $\mathcal{L}_{\text{cls}}$ and the position prediction loss on the patch tokens $\mathcal{L}_{\text{jigsaw}}$:

$$\mathcal{L}_{\text{total}} = \underbrace{CE(\mathbf{y}_{\text{pred}}, \mathbf{y})}_{\mathcal{L}_{\text{cls}}} + \eta \underbrace{CE(\tilde{\mathbf{y}}_{\text{pred}}, \tilde{\mathbf{y}})}_{\mathcal{L}_{\text{jigsaw}}} \tag{2}$$

where $\tilde{\mathbf{y}}_{\text{pred}}$ and $\tilde{\mathbf{y}}$ denote the position prediction and the corresponding real position, respectively, and η is a hyper-parameter balancing the two losses.

As detailed in the bottom part of Fig. 1, our injected jigsaw puzzle flow is different from naive jigsaw puzzle implementation in twofold: *(i)* we get rid of positional embeddings in the jigsaw puzzle flow; *(ii)* we randomly mask $\lfloor \gamma L \rfloor$ patches in the input where $\gamma \in [0, 1)$ is a hyper-parameter denoting the mask ratio. The former prevents models from being cheated from explicit clues in the positional embeddings, and the latter encourages the prediction to rely on global context rather than several particular patches. Their helpfulness for the main classification task is shown in Section 4. Since the proposed jigsaw puzzle solving is a self-supervised task, it can be easily plugged into existing ViTs without modifying the original architecture.

Architecture We employ DeiT-Small/16 [3] without distillation token as the backbone in our experiments if not specified, which is the same architecture as ViT-Small/16 in [1]. DeiT-Small/16 has the embedding dimension of $D = 384$ with 6 heads, $N = 12$ layers and the total number of parameters is approximately 22M. The training image resolution is 224×224 leading to 14×14 patches as inputs. The prediction head of the image classification flow is a single fully connected layer mapping from the encoder embedding dimension to the number of classes. For our jigsaw flow, we adopt a 3-layer *MLP* head after the encoder where the dimensions of the first two layers are equal to the encoder embedding dimension, and the output dimension of the last layer equals to the total number of image patches ($L = 14 \times 14$).

4. Experiments

In this section, we demonstrate the effectiveness of our approach on three tasks: generalization on large-scale classification on ImageNet [18], and robustness to noisy labels and adversarial examples. In Section 4.1, we evaluate our approach on the classification task with ImageNet dataset. In Section 4.2, we extensively validate our approach on three real-world noisy label datasets: Animal-10N [19], Food-101N [20] and Clothing1M [21]. In Section 4.3, we show our jigsaw flow improves ViTs' robustness against adversarial examples. Ablation study is given in Section 4.4. Further discussions on setting jigsaw puzzle solving as a pretext task and exploring the benefits of Jigsaw-ViT to downstream tasks are additionally provided in Section 4.5. More experimental details including ablations can be found in the supplementary material.

4.1. Generalization on large-scale image classification

ImageNet [18] is a standard dataset for large-scale image classification, and we use ILSVRC-2012 ImageNet-1K dataset containing 1,000 classes and approximately 1.3M images for the evaluation of our Jigsaw-ViTs in improving standard ViTs on large-scale image classification task.

Training details We train both DeiT [3] and our Jigsaw-ViT from scratch following the same training protocols in [3], where AdamW is taken as optimizer with the base learning rate of 5e-4, a weight decay of 0.05, the overall batch size as 1,024, and 300 training epochs. If not specified, for all experiments, we follow the data augmentation strategies in [3], e.g., Rand-Augment, MixUp and CutMix. We set the balancing hyper-parameter in (2) $\eta = 0.1$ and mask ratio $\gamma = 0.5$ here.

Results Table 1 shows the performances of Jigsaw-ViTs with different backbones on ImageNet-1K validation set and ImageNet V2 [40] which is a distinct test set suitable for measuring the overfitting level of models. The backbones include DeiT [3] with different capacities from tiny to base. DeiT-Tiny/16 is similar to DeiT-Small/16 but with fewer parameters: embedding dimension of 192 with 3 heads, 12 layers and the total number of parameters is approximately 5M. DeiT-Base/16 has an embedding dimension of 768, 12 heads, 12 layers and an approximate total number of parameters of 86M. For all architectures on both datasets, adding jigsaw branch consistently improves upon the baselines by training from

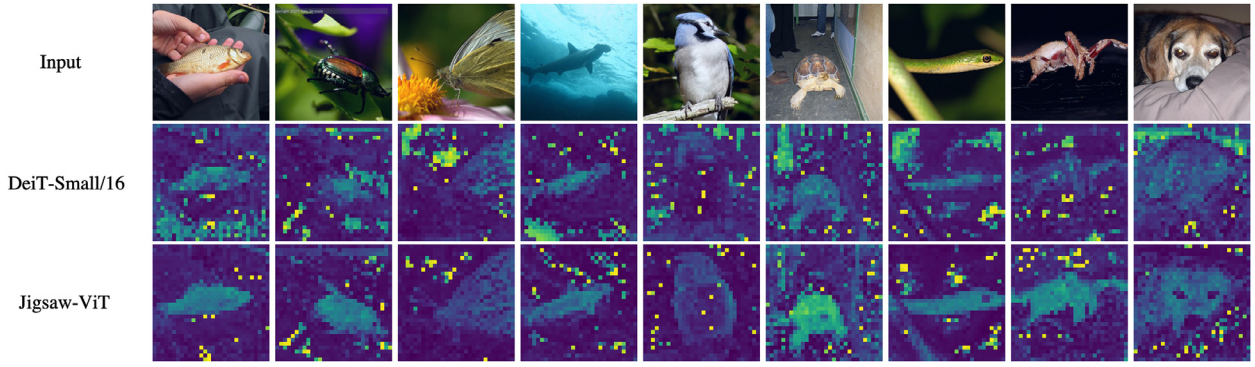

Fig. 2. Attention map associated to the class token of the last layer. We show the attention map for DeiT-Small/16 [3] and Jigsaw-ViT trained on ImageNet-1K [18]. Jigsaw-ViT learns clearer salient-object attentions over the listed instances. More examples can be found in the supplementary material.

Table 1
Image classification on ImageNet-1K [18] validation set and ImageNet V2 [40]. We compare to DeiT [3] with different capacities and report top-1 accuracy (%). We also show how much each Jigsaw-ViT model is above the baseline with ↑.

Backbone	#params	ImageNet-1K		ImageNet V2	
		Baseline	Jigsaw-ViT	Baseline	Jigsaw-ViT
DeiT-Tiny/16	5M	72.2	**74.1** ↑ 1.9	60.2	**61.4** ↑ 1.2
DeiT-Small/16	22M	79.8	**80.5** ↑ 0.7	68.5	**69.3** ↑ 0.8
DeiT-Base/16	86M	81.8	**82.1** ↑ 0.3	**71.0**	**71.0**

scratch or fine-tuning, verifying the effectiveness of our Jigsaw-ViT in attaining better generalization performances on large-scale image classification.

To further investigate the impact of our injected jigsaw flow in ViTs, we visualize the self-attention maps of the baseline ViT and Jigsaw-ViT trained on ImageNet-1K with DeiT-Small/16 in Fig. 2 following the visualization protocols in [51], which concatenates features of different heads associated to the class token. As in Fig. 2, Jigsaw-ViT is able to learn more distinctive salient-object attentions than DeiT-Small/16. The reason of this difference can be that solving jigsaw puzzle in ViT requires to understand the whole image so as to predict the correct spatial relationship between different image patches with a randomly shuffled order. Note that there are noisy hightlights on the background for both DeiT-Small/16 and Jigsaw-ViT. This is consistent with the statement in [51] that supervised ViTs attend less well to objects in both qualitatively and quantitatively than pure self-supervised ViTs. However, even though there are noises in attention maps, the proposed Jigsaw-ViT still manages to obtain better attention maps than its counterpart DeiT-Small/16. We refer to supplementary material for more visual results.

4.2. Robustness to label noise

A more challenging classification problem is conducted to evaluate our Jigsaw-ViTs, that is, the image classification with noisy labels. Three popular real-world noisy label datasets are extensively evaluated: Animal-10N [19], Food-101N [20] and Clothing1M [21], where Animal-10N [19] and Food-101N [20] are with noisy labels at a relatively low ratio, Clothing1M [21] contains noisy labels at a high ratio.

Training details Animal-10N [19] consists of 10 classes of animal images crawled online with manually annotated labels. The dataset consists of 50,000 training images with label noise ratio ∼8% and 5,000 clean testing images. Food-101N [20] contains 310,009 training images of food recipes collected online and are classified 101 classes with noise ratio ∼20%. Following [20], the learnt models should be evaluated on the test set of Food-101 of 25,250 clean labeled images. Clothing1M [21] is a large-scale dataset containing 1 million images of clothing crawled online. The dataset is categorized into 14 classes, containing 1,000,000 training images with noise ratio ∼38% and 10,526 test images. We follow the preprocessing in [29,32] for this dataset.

We train all the models from scratch, unlike most methods requiring extra data and learning from ImageNet-1K pretrained models [29,30,43,50], e.g., methods in Tables 2 (b) and 3 use ImageNet-1K pretrained ResNet-50 [49]. We use AdamW with a weight decay of 0.05 and train for 400K iterations with batch size 128. The training starts with a linear learning rate warm-up for 20K steps and cosine learning rate decay with a maximum learning rate of 1e-3 and a minimum of 1e-6.

Results We report top-1 accuracy on datasets with low noise rate, i.e., Animal-10N [19] and Food-101N [20] in Table 2. Interestingly, the baseline DeiT-Small/16 [3] achieves promising results on both datasets and already outperforms some competitive approaches, which demonstrates the powerful capabilities of ViTs on this task. Note that, in terms of model complexity, this ViT is comparable to ResNet-50 and much lighter than VGG architectures which are commonly used in the community. Our Jigsaw-ViT trained under $\mathcal{L}_{\text{total}}$ consistently outperforms DeiT-Small/16 trained under a single $\mathcal{L}_{\text{cls}}$ with substantial improvements. Moreover, our methods also achieve the best performances on both datasets in Table 2 among all compared state-of-the-art methods specifically designed for this noisy label task. These results indicate that our deployed auxiliary loss can implicitly serve as a practical regularization for learning with noisy labels. Similarly in Table 3, we observe the same tendency on the experiments with Clothing1M [21] dataset. Despite the high noisy ratio on Clothing1M, our Jigsaw-ViTs maintain to achieve promising results.

Additionally, it is worth mentioning that our method can serve as complementary strategies to other state-of-the-art methods to further boost their performances. In particular, we incorporate our Jigsaw-ViT to NCT [41] (Jigsaw-ViT+NCT). NCT is a two-stage method designed for combating label noise. Notably, by implementing NCT with our Jigsaw-ViT, the top-1 accuracy of NCT is improved by 0.4%, achieving the state-of-the-art result of 75.4%. These experiments not only verify the superiority of Jigsaw-ViTs as a stand-alone method, but also the effectiveness as a promising complementary tool to existing methods in combating label noise with boosted performances.

4.3. Robustness to adversarial examples

In this section, we investigate the robustness of Jigsaw-ViT against adversarial examples with perturbations on input images under both black-box and white-box attaches.

Training details Adversarial examples are crafted images by adding visually imperceptible perturbations to the clean images, which deteriorates the model predictions. In accordance to

Table 2
Image classification on datasets with low noise rate. We compare to state-of-the-art approaches and report test top-1 accuracy (%) on Animal-10N [19] (noise ratio ∼ 8%, in Table (a)) Food-101N [20] (noise ratio ∼ 20%, in Table (b)). We also show how much Jigsaw-ViT model is above DeiT-Small/16 [3] with ↑.

Method	CE [29]	Dropout [41,42]	SELFIE [19]	PLC [29]	NCT [32,41]	S3 [43]	Ours	
							DeiT-Small/16 [3]	Jigsaw-ViT
Acc. (%)	79.4	81.3	81.8	83.4	84.1	88.5	87.2	**89.0** ↑ 1.8
Backbone	VGG-19bn [44], #params: 143.7M, FLOPS: 19.7G						DeiT-Small/16, #params: 22M, FLOPS: 4.6G	

(a) Animal-10N [19]

Method	CE [29]	CleanNet [20]	MWNet [45,46]	SMP [47]	NRank [48]	PLC [29]	WarPI [46]	DeiT-Small/16 [3]	Jigsaw-ViT
Acc. (%)	81.7	83.5	84.7	85.1	85.2	85.3	85.9	84.2	**86.7** ↑ 2.5
Backbone	ResNet-50 [49], #params: 25.6M, FLOPS: 4G							DeiT-Small/16, #params: 22M, FLOPS: 4.6G	

(b) Food-101N [20]

Table 3
Image classification on datasets with high noise rate. We compare to state-of-the-art approaches and report test top-1 accuracy (%) on Clothing1M [21] (noise ratio ∼ 38%). We also show how much Jigsaw-ViT + NCT is above NCT with ↑.

Method	JO [28]	PLC [29]	ELR+ [50]	DivideMix [30]	S3 [43]	NCT [32,41]	Ours		
							DeiT-Small/16 [3]	Jigsaw-ViT	Jigsaw-ViT + NCT
Acc. (%)	72.2	74.0	74.8	74.8	74.9	75.0	71.6	72.4	**75.4** (Comp. NCT ↑ 0.4)
Backbone	ResNet-50 [49], #params: 25.6M, FLOPS: 4G						DeiT-Small/16, #params: 22M, FLOPS: 4.6G		

Section 4.1, we conduct the experiments on the models trained with ImageNet-1K [18]. Following [56], the validation set with 50,000 images are used as clean samples to generate adversarial examples. In the experiments, both black-box and white-box settings are considered for adversarial attacks.

In black-box settings, we first consider the transfer-based attacks where adversarial examples are generated by attacking surrogate models and then fed into the target models (e.g., our Jigsaw-ViTs) to evaluate the robustness performance. We then consider query-based attacks, which only require multiple queries of the outputs of the target models to perform the attacks. Following the settings in [56,57], methods used for generating the adversarial examples in transfer-based attacks are: FGSM [34], basic iterative method (BIM) [52], PGD [35], momentum iterative boosting (MI) [53] and the ensemble AA [36] which include white-box attacks. These attacks are crafted under maximum L_∞-norm perturbation $\epsilon = 16$ with respect to pixel values in [0,255], step size $\alpha = 2$, and number of steps 10 if it is a multi-step attack such as BIM, PGD and MI [57]. We adopt the pretrained ViT-Small/16 in [1] and pretrained ResNet-152 [49] on ImageNet-1K [18] as the surrogate models which are white-box with gradient available. The target victim models are DeiT-Small/16 and our Jigsaw-ViT whose gradients are inaccessible. For query-based attacks, we consider the popular square attack (Square) [54] with L_∞-norm perturbation $\epsilon = 16$ and different querying numbers $\{50, 100, 200, 500\}$.

In white-box settings, the adversarial examples are generated by directly attacking the accessible target victim models (DeiT-Small/16 and our Jigsaw-ViT). We consider both gradient-based attacks including FGSM and PGD, and one typical non-gradient-based attack named CW [55]. As in [57], we consider FGSM and PGD with the commonly-used L_∞-norm bounds of perturbations as $\epsilon \in [4, 8, 16]$. For CW, the L_2-norm perturbations are used with box-constraint parameter $c = 1$ and step size varying in $\{10, 20\}$.

Results Tables 4 and 5 report the top-1 accuracy (%) of the victim DeiT-Small/16 and our Jigsaw-ViT against black-box and white-box attacks, respectively. For adversarial examples generated by various attacks on different surrogate models in Table 4, the performances of target models all degrade drastically compared to their clean counterparts in Table 1, demonstrating the challenge of this task. In contrast, our Jigsaw-ViTs provides distinctively higher accuracy than standard ViTs under both transfer-based attacks and query-based ones. Such robustness of Jigsaw-ViT over DeiT-Small/16 becomes even more significant as the square attack utilizes more queries for stronger crafts in Table 4 (b). Specifically, Jigsaw-ViT exceeds DeiT-Small/16 by 9.7% with maximal 500 queries, compared the 2.8% improvement with maximal 50 queries.

In the white-box attacking results in Table 5, in addition to outperforming DeiT-Small/16 in all cases, our Jigsaw-ViT has non-zero accuracy in cases where DeiT-Small/16 is completely crafted by the PGD attack (steps $= 7$, $\epsilon \in \{8, 16\}$) in Table 5 (a). Results in Table 5 (b) relating to non-gradient-based CW attacks further demonstrate the effectiveness of the proposed Jigsaw-ViT, together verifying that injecting the proposed jigsaw puzzle flow to ViTs successfully provides improvements on the robustness against adversarial examples under various settings.

4.4. Ablation study

In this section, we first study the effect of positional embeddings in the jigsaw branch, and then investigate the impacts of the two hyper-parameters in Jigsaw-ViT: the loss balancing coefficient η in (2), the mask ratio γ of jigsaw image patches.

Effect of positional embedding in the jigsaw branch We conduct experiments on ImageNet-1K [18] and noisy label datasets including Animal-10N [19], Food-101N [20] and Clothing1M [21]. Results are given in Table 6, showing that the removal of positional embeddings in the jigsaw branch helps provide consistent improvement over all tested datasets, which validates our effective design of the jigsaw branch in ViTs.

Impact of η and γ The experimental investigations on the two hyper-parameter involved in the proposed Jigsaw-ViTs are conducted on datasets Animal-10N [19], Food-101N [20] and Clothing1M [21]. The results are illustrated in Table 7. First, Jigsaw-ViT provides consistent improvement on all three datasets compared to its non-jigsaw counterpart i.e., DeiT-Small/16 ($\eta = 0$). Moreover, the improvement is quite robust to the choices of both η and γ. Second, injecting the jigsaw puzzle solving with non-zero mask ratio to ViT indeed brings consistent performance boost over standard ViT, as we still observe that non-zero mask ratio γ shows better performances than the case of $\gamma = 0$ on different datasets and

Table 4
Robustness to adversarial examples in black-box settings. We report top-1 accuracy (%) after attacks on ImageNet-1K [18] validation set (higher numbers indicate better model robustness). (a) Transfer-based attacks where adversarial examples are generated by attacking a surrogate model. (b) Query-based attacks where adversarial examples are generated by querying the target classifier for multiple times. We also show how much our Jigsaw-ViT model is above DeiT-Small/16 [3] with ↑.

Surrogate	Target	FGSM [34]	BIM [52]	PGD [35]	MI [53]	AA [36]
ViT-Small/16	DeiT-Small/16	32.8	40.2	44.7	37.6	59.8
	Jigsaw-ViT	**34.8** ↑ 2.0	**43.0** ↑ 2.8	**47.7** ↑ 3.0	**40.2** ↑ 2.6	**62.5** ↑ 2.7
ResNet-152	DeiT-Small/16	59.0	65.9	67.9	65.3	70.6
	Jigsaw-ViT	**60.5** ↑ 1.5	**68.0** ↑ 2.1	**69.8** ↑ 1.9	**66.8** ↑ 1.5	**72.2** ↑ 1.6

(a) Acc. (%) under transfer-based attacks.

Attack	Num. queries	DeiT-Small/16	Jigsaw-ViT
Square [54]	50	49.3	**52.1** ↑ 2.8
	100	36.6	**41.4** ↑ 4.8
	200	22.8	**30.8** ↑ 8.0
	500	7.2	**16.9** ↑ 9.7

(b) Acc. (%) under query-based attacks.

Table 5
Robustness to adversarial examples in white-box settings. We report top-1 accuracy (%) after attacks on ImageNet-1K [18] validation set (higher numbers indicate better model robustness). (a) White-box attacks with L_∞-norm perturbation. (b) White-box attacks with L_2-norm perturbation. We also show how much our Jigsaw-ViT model is above DeiT-Small/16 [3] with ↑.

Attack	Model	$\epsilon = 4$	$\epsilon = 8$	$\epsilon = 16$
FGSM [34]	DeiT-Small/16	29.0	25.4	21.1
	Jigsaw-ViT	**34.1** ↑ 5.1	**30.2** ↑ 4.8	**24.9** ↑ 3.8
PGD [35], Steps=5	DeiT-Small/16	1.5	0.3	0.1
	Jigsaw-ViT	**5.3** ↑ 3.8	**2.3** ↑ 2.0	**1.3** ↑ 1.2
PGD [35], Steps=7	DeiT-Small/16	0.5	0.0	0.0
	Jigsaw-ViT	**2.9** ↑ 2.4	**0.6** ↑ 0.6	**0.2** ↑ 0.2

(a) Acc. (%) under attacks with L_∞-norm perturbation.

Attack	Steps	DeiT-Small/16	Jigsaw-ViT
CW [55]	10	10.0	**14.9** ↑ 4.9
	20	1.6	**4.7** ↑ 3.1

(b) Acc. (%) under attacks with L_2-norm perturbation.

Table 6
Comparisons between Jigsaw-ViT w/ and w/o pos. emb. We report test top-1 accuracy (%) on ImageNet-1K [18] and noisy label datasets including Animal-10N [19], Food-101N [20] and Clothing1M [21]. We also show how much Jigsaw-ViT w/o pos. emb. is above Jigsaw-ViT w/ pos. emb. with ↑.

	ImageNet-1K	Animal-10N Noise ∼ 8%	Food-101N Noise ∼ 20%	Clothing1M Noise ∼ 38%
w/ pos. emb.	80.3	87.3	84.7	71.1
w/o pos. emb.	**80.5** ↑ 0.2	**88.7** ↑ 1.4	**86.5** ↑ 1.8	**72.4** ↑ 1.3

lower mask ratios can already lead to good improvements. Hence, these evidences all demonstrate the effectiveness of the proposed approach.

4.5. Further discussions: pretext and downstream tasks

To further explore potentials of Jigsaw-ViT, we investigate the settings of both pretext and downstream tasks. First, we set the jigsaw puzzle solving as a pretext task so as to testify the necessity of building it as an auxiliary loss term. Second, we consider whether Jigsaw-ViT can benefit downstream tasks, such as semantic segmentation.

Jigsaw puzzle solving as a pretext task We set jigsaw puzzle solving, which is a self-supervised learning problem, as a pretext task for learning with noisy labels. During the pretext training, we adopt AdamW with a weight decay of 0.05 and train for 200K iterations with batch size 128. We set a linear learning rate warm-up for 20K steps and cosine learning rate decay with a maximum learning rate of 1e-3 and a minimum of 1e-6. Then, the backbone

Table 7
Ablation study on real-world noisy label datasets. We report test top-1 accuracy (%) on the following datasets: Animal-10N [19] (noise ratio ∼ 8%), Food-101N [20] (noise ratio ∼ 20%) and Clothing1M [21] (noise ratio ∼ 38%). Note that $\eta = 0$ corresponds to DeiT-Small/16 [3].

Method	η	MaskRatio	Animal-10N Noise ∼ 8%	Food-101N Noise ∼ 20%	Clothing1M Noise ∼ 38%
DeiT-Small/16	0	–	87.2	84.2	71.6
Jigsaw-ViT	0.5	0	88.6	86.1	71.9
		0.1	88.7	85.8	71.9
		0.2	88.7	86.1	71.8
		0.5	88.1	86.2	71.8
	1	0	88.6	86.1	72.3
		0.1	88.6	86.4	72.2
		0.2	88.7	86.3	72.1
		0.5	88.3	**86.7**	71.5
	2	0	88.6	86.4	71.7
		0.1	88.7	86.4	**72.4**
		0.2	**89.0**	86.5	71.9
		0.5	87.8	86.1	71.6

Table 8
Jigsaw puzzle solving as a pretext task. We report test top-1 accuracy (%) of models trained with jigsaw puzzle solving being the pretext task (entry "Jigsaw") and without it (entries "✗") on noisy label datasets. We also show how much DeiT with jigsaw puzzle solving as pretext task or as auxiliary loss improves over DeiT with ↑.

Method	Pretext task	Animal-10N Noise ∼ 8%	Food-101N Noise ∼ 20%	Clothing1M Noise ∼ 38%
DeiT-Small/16	✗	87.2	84.2	71.6
	Jigsaw	88.0 ↑ 0.8	**87.3** ↑ 3.1	71.9 ↑ 0.3
Jigsaw-ViT	✗	**89.0** ↑ 1.8	86.7 ↑ 2.5	**72.4** ↑ 0.8

Table 9
Semantic segmentation as a downstream task. Performance of Segmenters [11] on ADE20K [58] validation set with pretrained weights provided by DeiT [3] and Jigsaw-ViT trained on ImageNet-1K [18]. We also show how much Segmenter with Jigsaw-ViT pretrained weight improves over its DeiT pretrained counterpart with ↑.

Method	Pretrained	Pixel Acc.	mIoU (SS)
Seg-Small/16	DeiT-Small/16	80.2	42.3
	Jigsaw-ViT-Small	**80.6** ↑ 0.4	**42.9** ↑ 0.6
Seg-Base/16	DeiT-Base/16	81.1	45.1
	Jigsaw-ViT-Base	**81.3** ↑ 0.2	**45.7** ↑ 0.6

is fine-tuned on the same noisy label dataset with only $\mathcal{L}_{\text{cls}}$ used. Table 8 gives the comparisons of models trained with (entry "Jigsaw") and without (entry "✗") the jigsaw pretext task. Using jigsaw puzzle solving as a pretext task successfully outperforms training DeiTs from scratch without it. It shows that building jigsaw puzzle solving as an auxiliary loss term, i.e., Jigsaw-ViT, commonly leads to better performances than setting it as a pretext task. Rather than utilizing jigsaw puzzle solving as a pretext task, building it as an auxiliary loss term in the end-to-end training can in general be encouraged in usage to enhance model's robustness against noisy labels.

Semantic segmentation as a downstream task Semantic segmentation aims to label each image pixel with a corresponding category of the underlying object. Segmenter [11] builds upon ViTs and extends to semantic segmentation with extra semantic class tokens and a mask transformer decoder. Since this SOTA should be implemented on a pretrained ViT, we set the weights of DeiT and Jigsaw-ViT in Table 1 as pretrained weights for training on the challenging ADE20K dataset [58]. Table 9 shows that Segmenters with Jigsaw-ViT pretrained weights outperform their DeiT pretrained counterparts with respect to both mIoU with single scale (SS) inference, and pixel accuracy. These experiments verify that Jigsaw-ViT is also promising to benefit downstream task such as semantic segmentation, further showing the potentials of our proposed method.

5. Conclusion

In this paper, we inject jigsaw flow into the standard ViT by solving classical jigsaw puzzle as a self-supervised auxiliary loss during optimization, namely Jigsaw-ViT. To better utilize jigsaw puzzle solving, we propose to discard the positional embeddings in the jigsaw flow to avoid being directly hinted about the explicit position clues. Meanwhile, we observe that dropping some patches, i.e., patch masking, also plays a positive role in this procedure. The introduced auxiliary loss by injecting the jigsaw flow with the aforementioned two strategies not only enhances ViTs' performances for large-scale classification on ImageNet, but also attains stronger resistance to label noise and adversarial examples. These show that Jigsaw-ViTs improve standard ViTs in both generalization and robustness, even though such two aspects are usually considered as a trade-off. Our proposed method is an easy-to-reproduce, and yet a very effective tool to boost the performance of ViTs and even a plug-in complementary tool to enhance other methods.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was jointly supported by the European Research Council under the European Unions Horizon 2020 research and innovation program/ERC Advanced Grant E-DUALITY (787960), Research Council KU Leuven: Optimization framework for deep kernel machines C14/18/068, The Research Foundation Flanders (FWO) projects: GOA4917N (Deep Restricted kernel Machines: Methods and Foundations), Ph.D./Postdoctoral grant, the Flemish Government (AI Research Program), EU H2020 ICT-48 Network TAILOR (Foundations of Trustworthy AI-Integrating Reasoning, Learning and Optimization), Leuven.AI Institute.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.patrec.2022.12.023

References

[1] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, in: Proceedings of the ICLR, 2021.
[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of the NeurIPS, 30, 2017.
[3] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, H. Jégou, Training data-efficient image transformers & distillation through attention, in: Proceedings of the ICML, 2021, pp. 10347–10357.
[4] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, L. Zhang, CVT: Introducing convolutions to vision transformers, in: Proceedings of the ICCV, 2021, pp. 22–31.
[5] C.-F.R. Chen, Q. Fan, R. Panda, Crossvit: cross-attention multi-scale vision transformer for image classification, in: Proceedings of the ICCV, 2021, pp. 357–366.
[6] Y. Li, T. Yao, Y. Pan, T. Mei, Contextual transformer networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. (2022).
[7] X. Mao, G. Qi, Y. Chen, X. Li, R. Duan, S. Ye, Y. He, H. Xue, Towards robust vision transformer, in: Proceedings of the CVPR, 2022, pp. 12042–12051.
[8] T. Yao, Y. Pan, Y. Li, C.-W. Ngo, T. Mei, Wave-vit: unifying wavelet and transformers for visual representation learning, in: Proceedings of the ECCV, 2022, pp. 328–345.
[9] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: hierarchical vision transformer using shifted windows, in: Proceedings of the ICCV, 2021, pp. 10012–10022.
[10] T. Yao, Y. Li, Y. Pan, Y. Wang, X.-P. Zhang, T. Mei, Dual vision transformer, arXiv preprint arXiv:2207.04976(2022).
[11] R. Strudel, R. Garcia, I. Laptev, C. Schmid, Segmenter: transformer for semantic segmentation, in: Proceedings of the ICCV, 2021, pp. 7262–7272.
[12] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, Z. Liu, S. Ma, C. Xu, C. Xu, W. Gao, Pre-trained image processing transformer, in: Proceedings of the CVPR, 2021, pp. 12299–12310.
[13] M.-M. Paumard, D. Picard, H. Tabia, Image reassembly combining deep learning and shortest path problem, in: Proceedings of the ECCV, 2018, pp. 153–167.
[14] M. Noroozi, P. Favaro, Unsupervised learning of visual representations by solving jigsaw puzzles, in: Proceedings of the ECCV, 2016, pp. 69–84.
[15] F.M. Carlucci, A. D'Innocente, S. Bucci, B. Caputo, T. Tommasi, Domain generalization by solving Jigsaw puzzles, in: Proceedings of the CVPR, 2019, pp. 2229–2238.
[16] R. Du, D. Chang, A.K. Bhunia, J. Xie, Z. Ma, Y.-Z. Song, J. Guo, Fine-grained visual classification via progressive multi-granularity training of jigsaw patches, in: Proceedings of the ECCV, 2020, pp. 153–168.
[17] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, M. Jordan, Theoretically principled trade-off between robustness and accuracy, in: Proceedings of the ICML, 2019, pp. 7472–7482.

[18] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: Proceedings of the CVPR, 2009, pp. 248–255.
[19] H. Song, M. Kim, J.-G. Lee, Selfie: refurbishing unclean samples for robust deep learning, in: Proceedings of the ICML, 2019, pp. 5907–5915.
[20] K.-H. Lee, X. He, L. Zhang, L. Yang, Cleannet: transfer learning for scalable image classifier training with label noise, in: Proceedings of the CVPR, 2018, pp. 5447–5456.
[21] T. Xiao, T. Xia, Y. Yang, C. Huang, X. Wang, Learning from massive noisy labeled data for image classification, in: Proceedings of the CVPR, 2015, pp. 2691–2699.
[22] H. Freeman, L. Garder, Apictorial jigsaw puzzles: the computer solution of a problem in pattern recognition, IEEE Trans. Electron. Comput. (2) (1964) 118–127.
[23] A.C. Gallagher, Jigsaw puzzles with pieces of unknown orientation, in: Proceedings of the CVPR, 2012, pp. 382–389.
[24] S. He, H. Luo, P. Wang, F. Wang, H. Li, W. Jiang, Transreid: transformer-based object re-identification, in: Proceedings of the ICCV, 2021, pp. 15013–15022.
[25] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, R. Girshick, Masked autoencoders are scalable vision learners, arXiv (2021b).
[26] A. Srinivas, T.-Y. Lin, N. Parmar, J. Shlens, P. Abbeel, A. Vaswani, Bottleneck transformers for visual recognition, in: Proceedings of the CVPR, 2021, pp. 16519–16529.
[27] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, L. Shao, Pyramid vision transformer: a versatile backbone for dense prediction without convolutions, in: Proceedings of the ICCV, 2021, pp. 568–578.
[28] D. Tanaka, D. Ikami, T. Yamasaki, K. Aizawa, Joint optimization framework for learning with noisy labels, in: Proceedings of the CVPR, 2018, pp. 5552–5560.
[29] Y. Zhang, S. Zheng, P. Wu, M. Goswami, C. Chen, Learning with feature-dependent label noise: a progressive approach, in: Proceedings of the ICLR, 2021.
[30] J. Li, R. Socher, S.C. Hoi, Dividemix: learning with noisy labels as semi-supervised learning, in: Proceedings of the ICLR, 2020.
[31] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, M. Sugiyama, Co-teaching: robust training of deep neural networks with extremely noisy labels, in: Proceedings of the NeurIPS, 31, 2018.
[32] Y. Chen, S.X. Hu, X. Shen, C. Ai, J.A.K. Suykens, Compressing features for learning with noisy labels, IEEE Trans. Neural Netw. Learn. Syst. (2022) 1–15.
[33] N. Dalvi, P. Domingos, S. Sanghai, D. Verma, Adversarial classification, in: Proceedings of the SIGKDD, 2004, pp. 99–108.
[34] I.J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, in: Proceedings of the ICLR, 2015.
[35] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, in: Proceedings of the ICLR, 2018.
[36] F. Croce, M. Hein, Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks, in: Proceedings of the ICML, 2020, pp. 2206–2216.
[37] J. Yang, Y. Jiang, X. Huang, B. Ni, C. Zhao, Learning black-box attackers with transferable priors and query feedback, NeurIPS 33 (2020) 12288–12299.
[38] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z.B. Celik, A. Swami, Practical black-box attacks against machine learning, in: Proceedings of the ACM ASIACCS, 2017, pp. 506–519.
[39] K. Mahmood, R. Mahmood, M. Van Dijk, On the robustness of vision transformers to adversarial examples, in: Proceedings of the ICCV, 2021, pp. 7838–7847.
[40] B. Recht, R. Roelofs, L. Schmidt, V. Shankar, Do imagenet classifiers generalize to imagenet? in: Proceedings of the ICML, 2019, pp. 5389–5400.
[41] Y. Chen, X. Shen, S.X. Hu, J.A. Suykens, Boosting co-teaching with compression regularization for label noise, in: Proceedings of the CVPR Workshop, 2021, pp. 2688–2692.
[42] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.
[43] C. Feng, G. Tzimiropoulos, I. Patras, S3: Supervised self-supervised learning under label noise, arXiv preprint arXiv:2111.11288(2021).
[44] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proceedings of the ICLR, 2015.
[45] J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, D. Meng, Meta-weight-net: learning an explicit mapping for sample weighting, in: Proceedings of the NeurIPS, 32, 2019.
[46] H. Sun, C. Guo, Q. Wei, Z. Han, Y. Yin, Learning to rectify for robust learning with noisy labels, 124, 2022, p. 108467.
[47] J. Han, P. Luo, X. Wang, Deep self-learning from noisy labels, in: Proceedings of the ICCV, 2019, pp. 5138–5147.
[48] K. Sharma, P. Donmez, E. Luo, Y. Liu, I.Z. Yalniz, Noiserank: unsupervised label noise reduction with dependence models, in: Proceedings of the ECCV, 2020, pp. 737–753.
[49] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the CVPR, 2016, pp. 770–778.
[50] S. Liu, J. Niles-Weed, N. Razavian, C. Fernandez-Granda, Early-learning regularization prevents memorization of noisy labels, NeurIPS 33 (2020) 20331–20342.
[51] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, A. Joulin, Emerging properties in self-supervised vision transformers, in: Proceedings of the ICCV, 2021, pp. 9650–9660.
[52] A. Kurakin, I.J. Goodfellow, S. Bengio, Adversarial examples in the physical world, in: Proceedings of the Artificial intelligence Safety and Security, 2018, pp. 99–112.
[53] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, J. Li, Boosting adversarial attacks with momentum, in: Proceedings of the CVPR, 2018, pp. 9185–9193.
[54] M. Andriushchenko, F. Croce, N. Flammarion, M. Hein, Square attack: a query–efficient black-box adversarial attack via random search, in: Proceedings of the ECCV, 2020, pp. 484–501.
[55] N. Carlini, D. Wagner, Towards evaluating the robustness of neural networks, in: Proceedings of the IEEE Symposium on Security and Privacy, 2017, pp. 39–57.
[56] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, A.L. Yuille, Improving transferability of adversarial examples with input diversity, in: Proceedings of the CVPR, 2019, pp. 2730–2739.
[57] D. Wu, Y. Wang, S.-T. Xia, J. Bailey, X. Ma, Skip connections matter: On the transferability of adversarial examples generated with resnets, ICLR, 2020.
[58] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, A. Torralba, Semantic understanding of scenes through the ADE20K dataset, Int. J. Comput. Vis. 127 (3) (2019) 302–321.