Data-free Knowledge Distillation for Object Detection

Akshay Chawla, Hongxu Yin, Pavlo Molchanov and Jose Alvarez
NVIDIA
akshaych@alumni.cmu.edu, {dannyy, pmolchanov, josea}@nvidia.com

Abstract

We present DeepInversion for Object Detection (DIODE) to enable data-free knowledge distillation for neural networks trained on the object detection task. From a data-free perspective, DIODE synthesizes images given only an off-the-shelf pre-trained detection network and without any prior domain knowledge, generator network, or pre-computed activations. DIODE relies on two key components—first, an extensive set of differentiable augmentations to improve image fidelity and distillation effectiveness. Second, a novel automated bounding box and category sampling scheme for image synthesis enabling generating a large number of images with a diverse set of spatial and category objects. The resulting images enable data-free knowledge distillation from a teacher to a student detector, initialized from scratch.

In an extensive set of experiments, we demonstrate that DIODE's ability to match the original training distribution consistently enables more effective knowledge distillation than out-of-distribution proxy datasets, which unavoidably occur in a data-free setup given the absence of the original domain knowledge.

1. Introduction

Object Detection is a fundamental problem in computer vision where the aim is to accurately localize instances of objects in an image, out of a pre-defined set of classes. The combination of advancements in convolution neural networks [18, 39, 13] and availability of large and diverse datasets [36, 20] has led to a steady improvement in the accuracy of object detectors.

Accuracy improvement has come at the cost of ever increasing model complexity, computation and latency requirements. One way to decrease complexity without sacrificing performance is knowledge distillation (KD) [14]. Knowledge distillation allows us to train a compact model, known as student network, from one-or-more large pre-trained models, also known as teacher networks. Knowledge distillation accomplishes this by guiding the student using the teacher's predictions, which contain rich inter-class and object location information. The distilled student attains its best accuracy when it has access to the teacher's original training data. However, this data may not be available due to its size or because its release poses privacy or safety concerns.

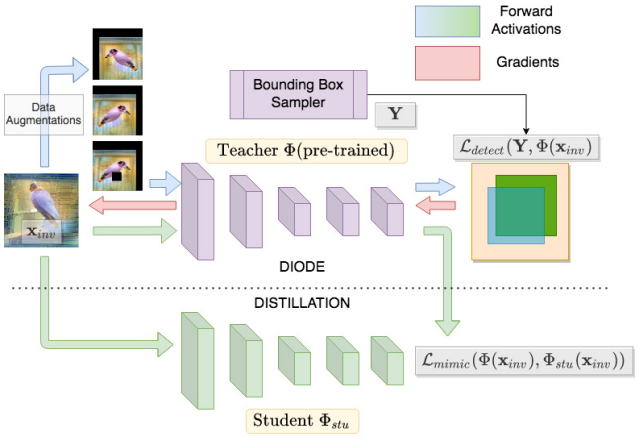

Figure 1: Data-free knowledge distillation framework for object detection. We first introduce DIODE that optimizes noise into images $\mathbf{x}_{inv}$ by inverting only a pre-trained object detection model Φ, equipped with differential data augmentation and a novel box sampling strategy to generate targets $\mathbf{Y}$. The synthesized dataset can then enable data-free knowledge distillation from teacher Φ to a new student detector Φ_{stu}.

DeepInversion [44] has recently emerged as an approach for synthesizing data from neural networks to solve for the absence of data. Contrary to generative approaches, DeepInversion does not require a generator or adversarial training to synthesize images conforming to a data distribution. Instead, it optimizes a batch of images, starting from noise, by matching the statistics of deep feature distributions to those stored in the batch-normalization (BN) layers of the network. This allows for generating images which have visual characteristics similar to the training dataset. However, DeepInversion is limited to synthesizing images only from deep classification networks.

In this paper, we propose a method for data-free knowledge distillation of deep object detection networks that consists of two main steps: a) image synthesis from a pre-trained model via a model inversion process we term DIODE, and b) an object detection task-specific knowledge distillation method on the synthesized images.

For the first step, we develop DIODE, a set of improvements on DeepInversion that allow us to synthesize high quality images with localized and category conditioned objects from an off-the-shelf pre-trained object detector. The key difference between DeepInversion [44] and DIODE is replacing the classification loss with an object detection loss, a set of extensive data augmentation strategies, and a box sampling method to enable the generation of a large number of images. We also introduce a tiling strategy and a false positive aggregation strategy to expand the density of objects in generated images. As a result, we are able to synthesize a new dataset of images containing objects from all predictable object categories in various locations at multiple scales.

In the second step, we use a large dataset of synthetically generated images for knowledge distillation between models. To this end, we formulate data-free knowledge distillation for object detection. Our formulation enables us to distill the knowledge from an off-the-shelf detector into another by only accessing images from a proxy dataset and without requiring its labels. This is the only work that performs data-free knowledge distillation tailored to object detection networks to the best of our knowledge.

Through extensive experiments, we show that DIODE improves the quality and generalizability of inverted images allowing them to effectively capture the model's training data distribution. These images also yield distillation efficacy on par with same-domain datasets, and consistently outperform out-of-domain datasets by large margins. DIODE alleviates the need of any prior domain knowledge for distillation, where conventional proxy datasets may suffer a performance drop due to a potential domain gap incurred from lack of apriori domain knowledge.

In summary, the contributions of this paper are twofold. First, we propose DIODE, a deep inversion algorithm for object detection. Our approach includes a new method to generate bounding boxes and their category labels, and differential data augmentation to improve image quality and generalizability. We also develop tiling and false positive aggregation techniques to increase object density. Second, we show how to transfer the knowledge from an off-the-shelf pre-trained object detector to a student network, without accessing its dataset. Extensive experiments show that our generated dataset outperforms (0.450 mAP) out-of-domain proxy datasets (0.313 mAP) by a significant margin improvement for the task of distillation.

The rest of the paper is structured as follows. We first summarize related works in Section 2. Then, in Section 3, we describe our approach to invert images for object detection. In Section 4, we introduce our data-free mimic learning approach to distill the knowledge of a pre-trained teacher detector to a student network. Finally, in Section 5, we present our experimental setup and discuss results.

2. Related work

Object detection. Modern object detectors consist of a backbone that is borrowed from state-of-the-art image classification CNNs e.g VGG-16[39], ResNets[13], EfficientNet[40] etc, and adding additional layers that predict boxes and labels from backbone features. Popular methods can be broadly categorized as: (a) Two-stage detectors which include an object proposal step to first extract category independent features and then make box and category predictions from them like RCNN [12], Fast-RCNN [11] and Faster-RCNN [33]. (b) One-stage detectors which combine object proposal and detection into a unified prediction model such as SSD [21] and Yolo [30] and their variants [31, 32]. The most popular detectors are one-stage detectors since they are designed with efficiency and inference latency in mind. Specifically, Yolo-V3 [32] is popular because it is faster than SSD and has accuracy that matches two-stage detectors.

Knowledge distillation. Knowledge distillation is a method to transfer knowledge from one-or-more pre-trained teacher models to a single student model. Introduced by [1, 14], the authors discover that a large model learns better representations of the data and its outputs provide rich inter-class information. They show that augmenting the training of student with a distillation loss that matches the predictions between teacher and student, improves its final accuracy. Ba and Caruana [1] coin this method as mimic-learning, however, Hinton *et al*. [14] refer to it as knowledge distillation.

Knowledge distillation has also been applied to object detection. Li *et al*. [19] mimic the ROI-pooled feature responses between student and teacher to learn an efficient detector. Shmelkov *et al*. [37] mimic the logit responses from roi-pooled features between student and teacher to combat catastrophic forgetting during incremental learning. Chen *et al*. [4] use mimic learning between the CNN backbones of a teacher and student Faster-RCNN. Wang *et al*. [41] apply mimic-learning to imitate the responses of a teacher on regions near the ground truth boxes. Mehta and Ozturk [24] develop the approach *objectness scaled distillation* which weights the loss incurred by each teacher predicted object, by its confidence score. Unlike other methods, Mehta and Ozturk [24] perform distillation on one-shot detectors. All these methods rely on images from the teacher's dataset, which can become difficult to access.

Data generation. In recent years, generative adversarial

networks (GANs) have become a popular paradigm to generate data [29, 46, 16, 17, 3]. They model data distribution using a generator which maps random noise to images and a discriminator which distinguishes between real and generated images. Recent works such as BigGan[3] and StyleGan[17] enable generation of highly realistic images. However, training the generator requires access to real data. An alternative line of work generates images through inverting pre-trained models. DeepDream [26, 8, 23, 38] back-propagates gradients into inputs towards generating features of the target classes. Through image prior regularization such as total variation, DeepDream allows for noise-to-image generation of smooth, category specific images. Along the same lines, DeepInversion [44] introduces a feature map regularizer based on BN information and enables synthesis of high fidelity images for deep networks trained on ImageNet. Both these methods only work for the classification task.

Data-Free KD. Aforementioned methods have inspired the recent emerging trend towards data-free knowledge distillation. Bhardwaj *et al*. [2] use DeepDream with activation vectors from 10% of original data to generate images. Lopes *et al*. [22] use a variant of the model inversion attack [10] that maximizes the similarity between response of original dataset and pre-computed activation statistics to generate a proxy dataset. Nguyen *et al*. [28] use a pre-trained GAN generator as a prior for performing a model inversion attack. Alternatively, Chen *et al*. [5] and Micaelli and Storkey [25] reformulate the classification network as a discriminator and train an external generator network to synthesize images that maximize the discriminator's response. All these existing methods use either meta-data, pre-computed activations or GAN generators to build up the proxy dataset. In contrast, recent work has started exploring knowledge distillation without any prior knowledge of the original dataset. Nayak *et al*. [27] generate a proxy dataset by only modelling the inter-class information from softmax layer. Yin *et al*. introduce DeepInversion [44] and adaptive DeepInversion towards generating proxy dataset for data-free KD with the latter having an additional competition regularization term that encourages teacher-student disagreement. All the aforementioned methods work only for the classification task.

3. DIODE: DeepInversion for Object Detection

In this section, we introduce **D**eep**I**nversion for **O**bject **DE**tection (**DIODE**), a novel image synthesis method to generate synthetic data similar to the distribution used to train a deep object detector. Our method only requires a pre-trained model and does not rely on auxiliary information (*e.g*. meta-data, feature activation) or additional network (*e.g*. pre-trained generative networks).

Given a batch of N input images $\mathbf{x}_{inv} \in \mathbb{R}^{N\times 3\times H\times W}$, and a pre-trained detection network $\Phi(\mathbf{x})$, we formulate DIODE as a regularized minimization problem that starts with every pixel initialized from random noise $\mathbf{x}_{i,c,u,v} \sim \mathcal{N}(0,1)$ and optimizes:

$$\mathbf{x}_{inv} = \min_{\mathbf{x}} \mathcal{L}_{detect}(\Phi(\mathbf{x}), \mathbf{Y}) + \mathcal{R}_{DI}(\mathbf{x}), \tag{1}$$

where $\mathcal{R}_{DI}$ is a regularization term added to steer away from adversarial examples and towards the distribution of images presented while training the detector, and $\mathcal{L}_{detect}$ is a loss function between pre-trained detector's predictions and desired targets $\mathbf{Y} \in \mathbb{R}^{K\times 6}$. This loss function is the same as the one used to train the object detector and it is responsible for synthesizing category and location conditioned objects in $\mathbf{x}_{inv}$. This is usually achieved by combining a box category loss $\mathcal{L}_{category}$, a box dimension loss $\mathcal{L}_{box}$ and a grid location loss $\mathcal{L}_{conf}$. Formulations of $\mathcal{L}_{conf}, \mathcal{L}_{box}, \mathcal{L}_{category}$ vary across the detector architectures. Common choices include binary cross entropy for $\mathcal{L}_{conf}$, cross entropy for $\mathcal{L}_{category}$ and either L1 or generalized IoU (GoI) [34] for $\mathcal{L}_{box}$. The targets $\mathbf{Y}$ consist of K boxes, where every k$^{\text{th}}$ box is defined by six parameters: batch index ($\mathbf{Y}_{k,1}$), a bounding box category ($\mathbf{Y}_{k,2}$) out of C categories and bounding box coordinates ($\mathbf{Y}_{k,3:6}$) *x,y,w,h*.

We use the regularizer $\mathcal{R}_{DI}$ to govern on image fidelity. $\mathcal{R}_{DI}$ consists of two parts: A prior term $\mathcal{R}_{prior}$ as in DeepDream [23] that acts on image priors, and the BN regularization term $\mathcal{R}_{BN}$ as in DeepInversion [44] that regularizes feature map distributions:

$$\mathcal{R}_{DI}(\mathbf{x}) = \mathcal{R}_{prior}(\mathbf{x}) + \mathcal{R}_{BN}(\mathbf{x}), \tag{2}$$

where $\mathcal{R}_{prior}$ checks on total variation, $\mathcal{R}_{TV}$, and L2 norm of the input:

$$\mathcal{R}_{prior}(\mathbf{x}) = \alpha_{TV}\mathcal{R}_{TV} + \alpha_{l_2}||\mathbf{x}||_2^2. \tag{3}$$

Total variation encourages adjacent pixels to have the same intensity by minimizing their L1 distances:

$$\mathcal{R}_{TV} = \sum_{i=1}^{N}\sum_{c=1}^{3}\sum_{u=1}^{W-1}\sum_{v=1}^{H-1} |\mathbf{x}_{i,c,u,v} - \mathbf{x}_{i,c,u+1,v}| + |\mathbf{x}_{i,c,u,v} - \mathbf{x}_{i,c,u,v+1}|. \tag{4}$$

This makes synthetic image conform to the fact that natural images are "smooth", effectively acting as a prior for natural images that has been widely shown to improve fidelity [23, 26, 44]. The second term, $||\mathbf{x}||_2^2$ prevents the generated images from saturating during the optimization process.

Akin to DeepInversion [44], DIODE utilizes the regularization $\mathcal{R}_{BN}$ to take advantage of the average feature statistics of training data that are cached in the BN layers of the detector. This pushes to valid feature distribution from

Variable	Sampling distribution	Variable description
$\mathbf{Y}_{k,1}$	-	batch index k, fixed
$\mathbf{Y}_{k,2}$	Cat.$(C, (1/C...1/C))$	object category
$\mathbf{Y}_{k,3}$	$\mathcal{U}[0, W)$	box x-center
$\mathbf{Y}_{k,4}$	$\mathcal{U}[0, H)$	box y-center
$\mathbf{Y}_{k,5}$	$\mathcal{U}[W_{min}, W_{max})$	box width
$\mathbf{Y}_{k,6}$	$\mathcal{U}[H_{min}, H_{max})$	box height

Table 1: Bounding box sampling space for DIODE image synthesis. We sample one object $\mathbf{Y}_k$ out of C categories for every image $\mathbf{x}_k$. The min. and max. box dimensions are set to $W_{min}/H_{min} = 0.1W$ and $W_{max}/H_{max} = 0.75W$. Cat. - categorical distribution; $\mathcal{U}$ - uniform distribution.

low- to high-levels of network embedding for the synthetic data. To this end, $\mathcal{R}_{BN}$ matches the feature statistics, i.e., channel-wise mean $\mu_l(\mathbf{x})$ and variance $\sigma_l^2(\mathbf{x})$ of the current batch, to those stored in BN layer $\mu_l^{BN}/\sigma_l^{2\,BN}, l = 1...L$, with L being the total number of BN layers:

$$\mathcal{R}_{BN}(\mathbf{x}) = \alpha_{BN} \sum_{l=1}^{L} ||\mu_l(\mathbf{x}) - \mu_l^{BN}||_2 + ||\sigma_l^2(\mathbf{x}) - \sigma_l^{2\,BN}||_2 \quad (5)$$

A combination of prior terms $\mathcal{R}_{prior}$ and BN regularization $\mathcal{R}_{BN}$ pushes the generated images closer to the teacher's training distribution. The weights α_{BN}, α_{TV}, and α_{l2} control their relative importance.

3.1. Bounding box sampling

In this section we propose a bounding box sampling strategy to automatically sample the targets $\mathbf{Y}$ required for generating images. These targets could be potentially provided manually, however, it becomes infeasible to repeatedly query the user for generating a large dataset.

To make this process data-free, we propose an alternate sampling strategy that samples one object $\mathbf{Y}_k \in \mathbb{R}^6$ for each image in the batch $\mathbf{x}_k \in \mathbb{R}^{3\times H\times W}$. This allows us to effectively and efficiently sample a large set of bounding boxes and category labels to guide the generation of images with a high degree of diversity. Table 1 summarizes the details of the sampling process.

This box sampler generates one object per image. To increase the object density, we propose two techniques: (1) tiling strategy and (2) false positive prediction sampling ($\mathbf{Y}_{FP}$ sampling). The tiling strategy grids multiple one-label generated images to create a multi-object image. Alternatively, $\mathbf{Y}_{FP}$ sampling is developed as a by-product of our observation that during DIODE, $\mathcal{R}_{BN}(\mathbf{x})$ causes the emergence of context relevant objects in addition to the initialized targets $\mathbf{Y}$. These false positive objects $\mathbf{Y}_{FP}$ are eventually suppressed to minimize the task loss in eq. 1. However, we can aggregate $\mathbf{Y}_{FP}$ that appear with high confidence to build complex targets that are semantically consistent with the label space of teacher's dataset.

As a result of our sampling strategy, DIODE is completely independent of detection labels from available datasets. Augmented with either tiling or $\mathbf{Y}_{FP}$ sampling, as we will show in our experiments in section 5, DIODE can yield objects of varying dimensions, counts, and categories in a single image to facilitate downstream tasks, *e.g.* distillation.

3.2. Differentiable augmentations for DIODE

Given the need to simultaneously satisfy bounding boxes and category labels during inversion, we observe that though eq. 1 improves image quality and imposes strong feature constraints on inputs, it converges quickly in the optimization process, and hence, leads to early saturation of image fidelity and generalizability.

To challenge the optimization process and generate images that are robust against label preserving transformations, we augment DIODE with a varied set of data augmentations that have been widely shown to be beneficial for training object detectors. This forces the semantic content of inverted images to be invariant to augmentations and thus conforming to natural images.

One key requirement for data augmentation in an inversion setup, however, is differentiability. The transfer function must be differentiable to enable the propagation of gradients from the final loss function to the input images. We consider the following augmentation strategies that satisfy this constraint: (1) random horizontal flips, (2) x-y translation jitter, (3) random brightness (4) random contrast and (5) cutout [7]. Note that DeepInversion [44] adopts the first two strategies, i.e., x-y jitter and horizontal flips, for inverting classification networks. We find that they are not enough for the challenging task of object detector inversion, and the enriched set of transformations is crucial to improve the quality of generated images. As we will show in our experiments, using all these augmentations yields a significant improvement in the visual fidelity and generalizability of $\mathbf{x}_{inv}$.

4. Data-Free Knowledge Distillation for Object Detection

In this section, we propose a method to use a large dataset of synthetically generated images for distilling a student detector from a pre-trained off-the-shelf teacher detector. Unlike existing approaches for distillation of object detectors [19, 37, 42], our approach does not require access to images **or** labels from the teacher's training data. Additionally, unlike previous data-free distillation approaches that only work on image classification models [2, 22, 28], we distill deep object detector networks.

For distilling deep object detectors, we make use of the mimic learning knowledge distillation paradigm which matches the predictions between student and teacher neural networks in addition to training with ground truth labels [1]. However, we consider *only* the component which guides the optimization of student detector using teacher's predictions on inputs $\mathbf{x}$. These predictions encode rich inter-category information and soft object proposals that can be transferred to the student. More precisely, we formulate the distillation loss as $\mathcal{L}_{mimic}$ which minimizes the L2 distance between teacher's and student's predictions on a collection of input images $\mathcal{X}$:

$$\Phi^*_{stu} = \min_{\Phi} \sum_{\mathbf{x} \in \mathcal{X}} \mathcal{L}_{mimic}(\Phi(\mathbf{x}), \Phi_{stu}(\mathbf{x})) \quad (6)$$

$$\mathcal{L}_{mimic}(\Phi(\mathbf{x}), \Phi_{stu}(\mathbf{x})) = ||\Phi(\mathbf{x}) - \Phi_{stu}(\mathbf{x})||_2^2. \quad (7)$$

The collection of images $\mathcal{X}$ can either be our synthesized images $\mathcal{X}_{DIODE}$ or belong to a proxy dataset $\mathcal{X}_{proxy}$. Since our approach is data-free, we make use of our synthesized dataset from DIODE and its variants for minimizing objective (eq. 7) and optimizing the student. As we will show later, this outperforms proxy datasets that suffer the domain gap problem.

5. Experiments

We next demonstrate the ability of DIODE for image synthesis from off-the-shelf deep object detection networks, and then perform data-free knowledge distillation between two deep object detection networks. We first show a wide range of synthetic images, where we provide in depth ablation studies to quantify the contribution of each individual proposed techniques. Then, we compare synthesized dataset with DIODE to other proxy datasets for transferring the knowledge between two networks.

5.1. Implementation details

We use a Yolo-V3 one-stage object detection network [32] as the teacher and student networks in our experiments. Both networks have same architecture since our goal is to extract knowledge from pre-trained model and not network compression. The teacher (Φ) is pre-trained on the MS-COCO dataset and yields an accuracy of 0.608 mAP. For the loss function $\mathcal{L}_{detect}$, we use the formulation by [15] that uses binary cross entropy for $\mathcal{L}_{conf}$ and $\mathcal{L}_{category}$ and generalized IoU [34] for $\mathcal{L}_{box}$. We use average precision (AP@IoU=0.5) as our measure of detector accuracy.

We generate $\mathbf{x}_{inv}$ by optimizing the cost function (eq. 1) for 5000 total iterations. We utilize a multi-resolution image generation approach as in [44] to enable large batchsize during the inversion process. This enables a faster generation of a large quantity of high resolution images. Check supplementary material for DIODE hyper-parameter values.

Verifier detection network ($\mathbf{\Phi}_{verif}$). We use a separate Yolo-V3-Tiny pre-trained on the teacher's dataset as a verifier network, following the paradigm of [44], to check on the generalizability of the inverted images $\mathbf{x}_{inv}$. A batch of inverted images is said to have generalized well if they make highly accurate verifier predictions. This implies that $\mathbf{x}_{inv}$ have not overfitted to the pre-trained model Φ being inverted, since they contain visual characteristics that can be independently recognized by a separate detection network. Note that the verifier serves only as an experimental tool for image generalizability analysis - it is *not required* during image synthesis in DIODE's loss computation (eq. 1).

Image tiling. Our bounding box strategy allows DIODE to synthesize one object per image. However, real images often contain multiple objects based on context. To increase the object density of our inverted dataset, we utilize a tiling strategy which merges (up to 25) synthetic images by tiling them into a single multi-object image.

$\mathbf{Y}_{FP}$ sampling. An alternative way to increase object density is false positive prediction sampling ($\mathbf{Y}_{FP}$ sampling) as described in section 3.1. In our experiments, we found that due to constantly evolving targets $\mathbf{Y}$, the quality of generated image suffers. While we can choose to synthesize images twice, once for generating targets and then for generating images with fixed targets, this takes considerable time and resources. To lessen this cost, we generate labels with $\mathbf{Y}_{FP}$ sampling only once for the lowest resolution (160), and then use them as fixed targets for multiresolution DIODE.

5.2. Image synthesis using DIODE

In this section, we first provide qualitative results of the generated images and then, ablation studies analyzing different components of DIODE.

Figure 2 shows representative examples of images generated using DIODE. As we can see, we can generate diverse, high quality and generalizable images. In a closer look at these generated images, we can observe that even though it does not explicitly optimize for context, DIODE generates context around targets. For example, the train is synthesized on top of a track and the ship is placed in a reflective surface of water. Importantly, these images have been generated without requiring any access to the teacher's training data, pre-computed activations or GAN generator.

Next, in Figure 3, we provide an example of $\mathbf{Y}_{FP}$ sampling in DIODE. As shown, the image starts with a single target (microwave) and during the DIODE iterations, we encounter context relevant false positive predictions such as cup and bowl, and retain them as targets. As a result, the generated image has a complex label space with semantically consistent and overlapping objects.

We now focus on quantifying the benefits of differentiable data augmentation. Table 2 shows the individual and

Figure 2: Images generated by DIODE on a Yolo-V3 off-the-shelf detector pre-trained on MS-COCO. The conditioning targets $\mathbf{Y}$ for each image are represented by a red box and category label. Note that the DIODE depicts target objects in contextually correct backgrounds with realistic details.

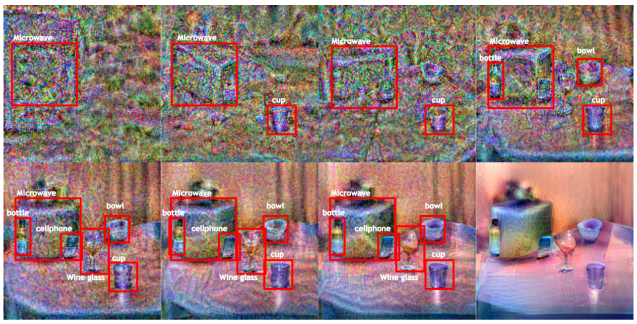

Figure 3: $\mathbf{Y}_{FP}$ sampling in DIODE. Image with its current targets shown at DIODE optimization iterations: (top-left $\rightarrow$ bottom-right) 800, 1200, 1600, 2000, 2400, 2800, 3200 and 4000.

cumulative impact of each strategy on the verifier (Φ_{verif}) accuracy. As shown, every individual data augmentation improves over the baseline of no augmentation. Importantly, even though cutout by itself is a strong augmenter, combining all augmentation methods yields significant improvements over any individual augmentations in terms of accuracy and robustness against initialization (lower variance). This is because data augmentation challenges the optimization process and results in images whose semantic content is invariant to augmentation, just like real data. Qualitative comparisons for this experiment are shown in Fig. 4. In the figure, we can observe clearer boundaries on the wine-glass and the emergence of class specific distinctive patterns such as the ribs on an umbrella, wheels on the car and layers of the burger. From these results, we can conclude that adding differentiable data augmentation improves the quality and the visual fidelity of the generated images.

DIODE	Data Augmentation: flip	jitter	brightness	contrast	cutout	Φ_{verif} (mAP)
w/o aug.	-	-	-	-	-	0.49 ± 0.063
w/ aug.	✓					0.52 ± 0.026
		✓				0.51 ± 0.017
			✓			0.53 ± 0.034
				✓		0.49 ± 0.025
					✓	0.70 ± 0.024
Ours	✓	✓	✓	✓	✓	$\mathbf{0.71 \pm 0.014}$

Table 2: Ablation study of the data augmentation based on Φ_{verif} accuracy. We report mean and std of 5 runs with different seeds, same targets.

DIODE is not limited to Yolo-V3 detector but can be adapted to other object detectors. Figure 5 shows qualitative examples of DIODE applied to SSD300 [21] object detector. Compared to Figure 2, we observe better color distribution but worse object boundaries.

With DIODE, we are now able to very effectively generate a large set of synthetic images: we repeat the process and generate 2500 batches via DIODE, each batch sampled from a different random seed and set of targets, of batch size 48, resolution at 416×416, 120k images in total. In parallel, we also generate a version of this dataset with $\mathbf{Y}_{FP}$ sampling. Then, in the next section, we use these new synthetic images for data-free knowledge distillation.

Figure 4: Qualitative effect of data augmentation on DIODE synthesized images from Yolo-V3 pre-trained on MS-COCO with (bottom) and without (top) data augmentation methods. left to right: wine glass, bench, umbrella, car, microwave oven, burger.

Figure 5: Images generated by DIODE on SSD300 detector pre-trained with MS-COCO 2017.

5.3. Data-free Knowledge Distillation for Object Detection

We focus now on evaluating the synthetic images generated by DIODE in the context of data-free knowledge distillation (KD) for object detection. To this end, we consider a pre-trained teacher model Φ and distill its information to a student detector Φ_{stu}. We first quantify the distillation impact of using images and labels from the teacher's dataset. Then, in a second experiment, we evaluate the effect of our tiling approach on generalization, and, finally, we evaluate our distillation approach compared to distilling using other proxy datasets. For these experiments, we use mimic learning as distillation method. We provide details of hyperparameters for this experiment in the supplemental material.

First, we quantify the distillation impact of using images and labels from the teacher's dataset for inversion. In Table 3 we first show the best case performance of mimic learning, where it has access to images from MS-COCO.

Input images	original image	original label	mAP
MS-COCO (original)	✓	✓	0.524
DIODE	X	✓	0.462
DIODE	X	X	0.418 (**data-free**)

Table 3: Ablation study towards data-free knowledge distillation from a pre-trained Yolo-V3 network (0.608 mAP) on MS-COCO dataset as teacher, to a new Yolo-V3 network initialized from scratch. mAP measured on MS-COCO validation set.

#objects per image	distribution	mAP
1	–	0.418
1-25	random	0.433
1-25	uniform	0.426
1-25	MS-COCO like	**0.435**
1-25	VOC like	0.429

Table 4: Ablation study comparing the performance of student distilled on tiled images generated by different objects per image distributions. We tile between 1 - 25 synthetic single-object images, depending on the target distribution. mAP is measured on the MS-COCO validation set.

The difference in the accuracy of student (0.524 mAP) and teacher (0.608 mAP) informs us that we are limited by the current method of knowledge distillation. Then, we show the best possible performance of DIODE, by using synthetic images conditioned on MS-COCO labels. Next, we show our data-free approach which uses neither images, nor labels from training data. The difference between last two rows reveals that there is significant information present in the label space of MS-COCO, and our $\mathbf{Y}_{FP}$ sampling and tiling strategies are an attempt to counter this difference.

Next, we perform a study to choose the objects per image distribution for tiling synthesized images in Table 4. Real datasets contain a variable number of objects per image, and we generate four datasets of 120000 tile images, where the objects per image distribution is chosen to be random, uniform or similar to an existing dataset like MS-COCO [20] or VOC [9]. In the same table the first row shows the accuracy of a student distilled on synthetic images without tiling. Results indicate the following: (1) tiling improves performance over single object DIODE images (2) the best results are achieved by using a distribution similar to teacher's dataset and (3) random tiling works as a competitive alternative when no information is available about teacher's dataset.

Finally, we compare distillation using DIODE generated images to other proxy datasets. For comparison we consider two types: *in-distribution* and *out-of-distribution* proxies. *In-distribution datasets* contain objects from categories similar to those of the teacher's dataset (MS-COCO). They represent the scenario where we are aware of the

Dataset	# required images	dataset distribution	performance (mAP)
original teacher			
MS-COCO	117k	original training set	0.608
distillation to student (reinitialized)			
ImageNet [6]	120k	same-domain	0.466
VOC [9]	22k	same-domain	0.443
BDD100k [45]	160k	out-of-domain	0.313
GTA5 [35]	50k	out-of-domain	0.285
Data-free w/ DIODE (ours)	**0***	–	**0.418**
Data-free w/ DIODE+tiles (ours)	**0***	–	**0.435**
Data-free w/ DIODE+$\mathbf{Y}_{FP}$ sampling (ours)	**0***	–	**0.450**

Table 5: Knowledge distillation results from a pre-trained Yolo-V3 object detector to a new Yolo-V3 initialized from scratch. *: for a fair comparison, we report results based on 120k DIODE synthetic images and 120k DIODE tiled images (from Table 4). Note that this requires no access to any external image nor labels.

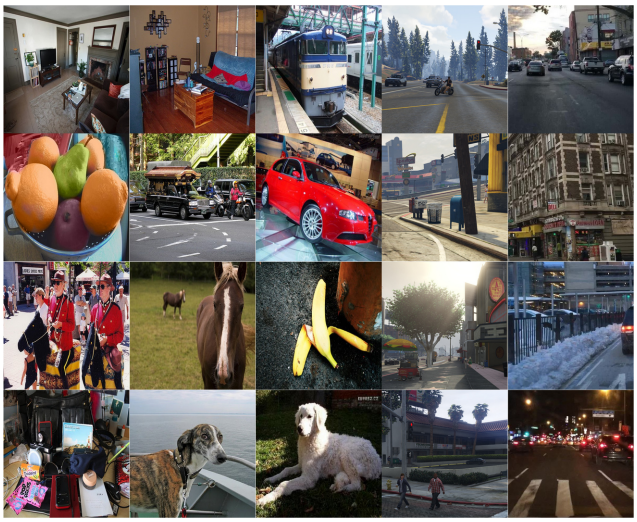

Figure 6: Proxy datasets for knowledge distillation, from left to right by columns: (1) MS-COCO (2) Pascal-VOC (3) ImageNet (4) GTA5 (5) BDD100k.

data used to train the the teacher and we select proxies that are as close as possible. On the other hand, *out-of-distribution datasets* have a minimum category overlap with the teacher's training data. In this case, they represent the scenario where we are unaware of the original training data and, therefore, we select proxy datasets randomly. In particular, in this experiment, we use VOC2007+2012 [9] and ImageNet [6] that represent images of common objects as in-distribution, and BDD100k [45] and GTA5 [35] that represent data for autonomous car scenario as out-of-distribution proxy datasets. Note that we adjust the proxy datasets to match the number of samples in teacher's dataset.

Table 5 shows the accuracy comparison between student's distilled from DIODE generated images and the proxy datasets. As shown, distilling using the images generated by our proposed approach with $\mathbf{Y}_{FP}$ sampling (0.450 mAP) outperforms the best out-of-domain proxy dataset (0.313 mAP) and is competitive with distilling on same domain datasets (0.466 mAP). These results are a consequence of the amount of similarity between the original training data and the proxy data. Fig. 6 shows that same domain datasets are extremely similar to MS-COCO. They contain objects from equivalent categories and thus are able to achieve better distillation accuracy. In comparison, out-of-domain proxy datasets such as GTA5 or BDD100k contain objects from few of the training data categories and are also contextualized differently, leading to worse results. We also note that increasing object density by $\mathbf{Y}_{FP}$ sampling leads to better generalization than tiling due to contextually relevant targets. However, as noted in section 5.1, $\mathbf{Y}_{FP}$ sampling requires more resources so tiling may be preferred when compute is limited.

6. Conclusion

In this paper, we have proposed a method for data-free knowledge distillation of deep object detection networks. Our approach consists of two main components: DIODE, a framework to synthesize images from a pre-trained detection model via model inversion, and a data-free mimic learning approach to distill the knowledge from a teacher to a student on the synthesized images for object detection. Our qualitative and quantitative experiments demonstrate the quality and and generalizability of the synthesized images. Moreover, data-free distillation for object detection using these synthesized images yields a significant improvement (0.450 mAP) compared to out-of-domain proxy datasets (0.313 mAP) and are competitive with same-domain proxy datasets (0.466 mAP). Both DeepInversion [44] and DIODE are limited to synthesizing images from networks with batch-normalization layers. Further exploration is required to extend these methods to other normalization layers such as group-normalization [43].

References

[1] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Advances in neural information processing systems*, pages 2654–2662, 2014.

[2] Kartikeya Bhardwaj, Naveen Suda, and Radu Marculescu. Dream distillation: A data-independent model compression framework. *arXiv preprint arXiv:1905.07072*, 2019.

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[4] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. In *Advances in Neural Information Processing Systems*, pages 742–751, 2017.

[5] Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. Data-free learning of student networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3514–3522, 2019.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[7] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[8] Alexey Dosovitskiy and Thomas Brox. Inverting visual representations with convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4829–4837, 2016.

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[10] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333, 2015.

[11] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[12] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[14] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[15] Glenn Jocher, Yonghye Kwon, guigarfr, Josh Veitch-Michaelis, perry0418, Ttayu, Marc, Gabriel Bianconi, Fatih Baltacı, Daniel Suess, idow09, WannaSeaU, Wang Xinyu, Timothy M. Shead, Thomas Havlik, Piotr Skalski, NirZarrabi, LukeAI, LinCoce, Jeremy Hu, IlyaOvodov, GoogleWiki, Francisco Reveriano, Falak, and Dustin Kendall. ultralytics/yolov3: 43.1map@0.5:0.95 on coco2014, May 2020.

[16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[17] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, page 1097–1105, Red Hook, NY, USA, 2012. Curran Associates Inc.

[19] Quanquan Li, Shengying Jin, and Junjie Yan. Mimicking very efficient network for object detection. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 6356–6364, 2017.

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014.

[21] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[22] Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. Data-free knowledge distillation for deep neural networks. *arXiv preprint arXiv:1710.07535*, 2017.

[23] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015.

[24] Rakesh Mehta and Cemalettin Ozturk. Object detection at 200 frames per second. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[25] Paul Micaelli and Amos J Storkey. Zero-shot knowledge transfer via adversarial belief matching. In *Advances in Neural Information Processing Systems*, pages 9551–9561, 2019.

[26] A Mordvintsev, C Olah, and M Tyka. Inceptionism: Going deeper into neural networks, 2015.

[27] Gaurav Kumar Nayak, Konda Reddy Mopuri, Vaisakh Shaj, R Venkatesh Babu, and Anirban Chakraborty. Zero-shot knowledge distillation in deep networks. *arXiv preprint arXiv:1905.08114*, 2019.

[28] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in neural information processing systems*, pages 3387–3395, 2016.

[29] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[30] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[31] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[32] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *ArXiv*, abs/1804.02767, 2018.

[33] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[34] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 658–666, 2019.

[35] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *European conference on computer vision*, pages 102–118. Springer, 2016.

[36] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[37] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. Incremental learning of object detectors without catastrophic forgetting. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3400–3409, 2017.

[38] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[39] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[40] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2019.

[41] Tao Wang, Li Yuan, Xiaopeng Zhang, and Jiashi Feng. Distilling object detectors with fine-grained feature imitation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4933–4942, 2019.

[42] Yi Wei, Xinyu Pan, Hongwei Qin, Wanli Ouyang, and Junjie Yan. Quantization mimic: Towards very tiny cnn for object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 267–283, 2018.

[43] Yuxin Wu and Kaiming He. Group normalization. In *European conference on computer vision*. Springer, 2018.

[44] Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8715–8724, 2020.

[45] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2(5):6, 2018.

[46] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.