Augmenting Object Detection Supervised Training with Pixel-level Contrastive Learning

Yasser Abdelaziz Dahou Djilali[1,2]
yasser.djilali@tii.ae
Kebin Wu[2]
kebin.wu@tii.ae
Kevin McGuinness[1]
kevin.mcguinness@dcu.ie
Ebtessam Almazrouei[2]
ebtesam.almazrouei@tii.ae
Merouane Debbah[2]
merouane.debbah@tii.ae
Noel O'Connor[1]
Noel.OConnor@dcu.ie

[1] Dublin City University (DCU), Dublin, Ireland.
[2] The Technology Innovation Institute, Abu Dhabi, UAE

Abstract

Despite the success of deep learning based object detectors, evaluating intermediate representations using explicit objectives is less common, and the structure of the latent space is usually ignored, thereby compromising the expressive power of the detectors. In this paper, we propose a pixel-level contrastive training for object detection in a fully supervised setting. The core hypothesis is to enforce priors on the latent space with desirable properties along with the supervised objective, that favor better generalization. The main intuition is to push spatially close pixel representations to be more similar than further away ones. This captures spatial smoothness for better class prediction, and spatial discrimination around edge areas, to provide more accurate bounding boxes. Our training scheme can be integrated into existing Detection Transformer like frameworks without any inference overhead. Training the recently introduced Detection Transformer (i.e., DETR, Deformable-DETR, DN-DETR) with our setting improves performance on COCO across all metrics, but also on the cars driving detection dataset BDD100K, which is more challenging than COCO (i.e, various weather conditions, higher number of objects per scene, etc). We hope this work will influence reconsideration of the common supervised object detection training paradigm.

1 Introduction and related works

Object detection is a fundamental task in computer vision, the goal of which is to identify and locate objects of interest within an image. Over the last decade, significant progress has been achieved, driven by the availability of large scale datasets (e.g., COCO [19]), and the emergence and rapid progress of deep learning techniques. Some common detectors

include: FasterRCNN [29], RetinaNet [21], YOLO [28], etc. Despite remarkable advances in the field of object detection, these algorithms augment the architecture with hand-crafted components specific for the detection task, such as proposal generation, anchor design, and non-maximum suppression (NMS) post-processing. In contrast, DETR [4] shifted the object detection paradigm, casting the problem as a set-based prediction one, eliminating the hand-designed components, and maintaining comparable performance against the well-established FasterRCNN [29]. DETR [4] combines several techniques such as bipartite matching loss, transformer encoder-decoder with parallel decoding to design DEtection Transformer (DETR). The approach formulates the object detection task as an image-to-set problem. The model outputs a fixed-length unordered set of classes and bounding boxes of all possible objects present in the image. The bipartite matching forces unique one-to-one predictions. Intuitively, the decoder queries can be interpreted as humans saccading at various spatial locations of an image; each human hence observes others before making its prediction. This potentially allows for reasoning to emerge as the transformer decoder can associate objects that it encounters, compare their correlations, and make analogies over the recurring patterns, as the cross attention acts as a relative filtering testing inter-channels of the same latent representation. However, DETR is difficult to optimize, and suffers from slow convergence, i.e., more than 300 epochs are needed to obtain comparable performance to FasterRCNN [29].

Several hypotheses have been proposed to account for this. First, the attention weights are uniformly assigned to all pixels in the feature maps at initialization, hence attending to meaningless locations that do not contribute to the feature propagation mechanism. Second, the discrete bipartite matching is unstable under stochastic optimization, as the same query is matched with different objects across epochs. Last, the decoder cross attention is under optimized in the early training, resulting in noisy contextual information for the queries.

Inspired by the deformable convolution [6], [41] adds a translation term into the formula of the transformer attention, allowing a sparse spatial sampling by attending to a smaller set of locations (reference points). Consequently, this gating mechanism approximates the full self-attention via the locality inductive bias excluding potential long-term dependencies from the calculation. [30] leverages a backbone with FPN [20] to produce multi-stage features, then a binary classifier is trained using the FCOS ground-truth assignment [32] rule to select a sub-set of the features to be fed to the transformer encoder. Hence a decoder-free model discarding the cross-attention modules potentially behind the slow convergence.

UP-DETR[7] tackle the issue of random queries initialization, the authors designed an unsupervised pretext task named random query patch detection as a pre-training step. For a given image, a random set of patches are cropped, and the transformer is trained to predict bounding boxes of these query patches. Similarly, Efficient-DETR [38] proposed a dense prediction to select the top-k proposals, their corresponding vectors representation are fed as object queries, and their 4-d proposals as used as reference points in the deformable attention. Anchor-DETR [35] induced the anchor points corresponding to a spatial location via the receptive field as learned object queries, to goal being to alleviate the ambiguous optimization of 'one region, multiple objects'. DAB-DETR [22] extended the work of [35] by explicitly learning anchor boxes as queries, where the 4-d points are the label's boxes, pushing the queries to attend to locations where there is an object, while also being scale adaptive as height and width result in non-isotropic priors.

On solving the bipartite matching instability, DN-DETR [22] sets an auxiliary denoising task, where the goal is to reconstruct noisy queries. This loss is not part of the bipartite matching, making the optimization more stable as it is easier to solve. Intuitively, if the transformer decoder is able to denoise a bounding box, it has to adjust the attention weights

to be more precise in localization, which is in turn part of the Hungarian loss. DINO [40] augmented DN-DETR by leveraging a contrastive denoising training. For a given ground truth box, two noise ratios are added, the smaller is marked as positive, the remaining one as negative. The contrastive denoising forces the model not to produce duplicate bounding boxes.

These approaches project images to a non-linear latent space as feature maps, then decode them to a learned set of logits depending on the architecture being used (i.e. proposals for FasterRCNN, and queries for DETR-like approaches). However, these methods ignore the latent space topology, and assume that the loss function supervises the feature maps implicitly. Although achieving good detection metrics on the test set, this may not guarantee an adequate latent space topology in deep learning. However, what constitutes an ideal detection latent space is a core question that has not been adequately addressed to date. Preferably, it should favor: i) Spatial smoothness of pixels within the same bounding box of spatial-locations sharing the same semantic class. ii) Effectively discriminate sudden transitions (object boundaries) by identifying close pixels in the input space with different class semantics.

Hence we argue that, notwithstanding the remarkable results of existing algorithms on common benchmarks, learning a better regularized pixel representation space with these desired properties can potentially favor effective generalization. The recent success of contrastive similarity learning motivates this work to induce priors for a better latent space topology. Contrastive learning [17] refers to learning by maximizing the mutual information (MI) between the anchor and the set of positives, while minimizing the MI with negatives pairs (i.e., pushing away the l_2 normalized features of negatives by a distance on the hypersphere). This is achieved based on variations of Noise Contrastive Estimation [9, 10]. Refer to [10, 31] for further details on the derivation of the NCE loss. Clearly, however, these methods enforce alignment and uniformity of the latent space [33], thus, may be sub-optimal for tasks requiring granular dense pixel predictions such as detection. Recently, [37] considered each pixel as a separate class, where nearby pixels are positives, and further ones from a pre-defined threshold are negatives.Then, the noise contrastive estimation (NCE) loss [9, 26] is calculated channel-wise using the positive/negative mask encouraging spatial sensitivity.

Authors of [34] proposed a fully-supervised pixel-wise contrastive algorithm for semantic segmentation. The mask is generated from the segmentation maps: for a pixel i belonging to a class $\bar{c}$, the set of positives share the same class $\bar{c}$, whereas the negatives are from different classes $\mathcal{C}/\bar{c}$. The negative pairs are sampled from a memory bank, hence, exploring the rich semantics across pixels located in different images. The contrastive loss brings improvements over existing segmentation models (e.g., DeepLabV3, HRNet, and OCR). In this work, we leverage a simple yet effective pixel-level constrastive learning algorithm seeking better granular representations in a fully supervised way. The aim is to avoid overlapped redundant representation exploration in the image space, and over adjacent layers. As shown in Figure 1, the algorithm attempts to encourage consistency over the spatially close pixels across the two views processed by the regular supervised branch and the momentum branch, respectively. Additionally, to push further exploration of latent space for distinctive representation learning, we adopt multiple layers of pixel-level consistency regularization. Hence, this setting achieves higher gain outlining the effectiveness of the proposed objective function. Moreover, integrating this module does not bring additional overhead at inference time, nor any modification to the base model. Our contributions are as follows:

- We propose a plug-and-play (multiple-layer) pixel-level contrastive learning module to improve the performance of DETR and its variants. It leads to better representations by

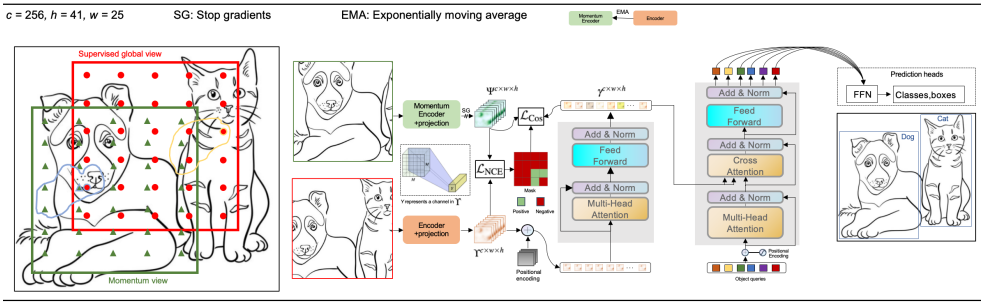

Figure 1: Complete pipeline for training. The top branch (momentum encoder+projection head) is composed of an encoder and a projection head. The bottom branch is the DETR-like functions, consisting of an encoder, followed by the transformer encoder-decoder. The momentum encoder and projection head are kept fixed (no gradient flow). The positive/negative mask is created using the crop coordinates of the global and momentum views.

enabling a balance between spatial smoothness and spatial sensitivity.

- We find that the expressive power of the transformer encoder can eliminate the use of negatives pairs necessary for the contrasive loss, and still enforce properties of interest.
- Comprehensive experiments using different DETR variants are conducted on COCO, and the challenging driving dataset BDD100K [39].We observe a constant gain over the common object detection metrics.

2 Method

The common supervised training implicitly configures the latent space through empirical risk minimization. However, it is unclear whether better loss functions improving accuracy on a test set are not violating the speculated topology of a good representation space [2, 15], through some less understood neural network dynamics. Our algorithm breaks the object detection learning task into its first principles. The proposed method explicitly forces the properties of local smoothness of input and representation, and spatial coherence across a set of observations. The aim is to learn discriminative pixel-to-pixel representations using the contrastive loss $\mathcal{L}_{\text{NCE}}$. Additionally, the cosine loss $\mathcal{L}_{\text{Cos}}$ captures the pixel-to-region consistency. Ideally, $\mathcal{L}_{\text{NCE}}$ and $\mathcal{L}_{\text{Cos}}$ regularize the latent space so the backbone encoder $\mathbf{f}_\theta$, and the transformer encoder Ω_ω is invariant to small and local changes (i.e., data augmentations).

2.1 Overview of the approach

For an image dataset $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{|\mathcal{D}|}\}$ where $\mathbf{x}_i \in \mathcal{R}^{3\times H\times W}$, we define a set of transformations $\mathcal{T}$ and $\mathcal{P}$, with empirical probability distribution $p(\mathbf{X})$ sampling a given operation. The set $\mathcal{T}$ contains standard operations as in [4] (i.e., horizontal flip, global cropping, random resizing). The set $\mathcal{P}$ contains standard transformations used in contrastive learning such as random crops, random jitter in color space, random conversion to gray-scale, random horizontal flips, and solarization [5, 8]. We denote $\mathbf{x_e} \sim \mathcal{T}(x)$ and $\mathbf{x_m} \sim \mathcal{P}(x)$. Different from

classification, object detection is a dense prediction task, in which an image-wise contrastive learning may lead to sub-optimal solutions. Instead, we propose to use the pixel-level contrastive learning to regularize the representation space. The aim is to learn representations capturing discriminative pixel-to-pixel features (i.e., spatial sensitivity) with $\mathcal{L}_{\text{NCE}}$. However, objects usually occupy blobs, hence, pixel-to-region contrast is enforced with $\mathcal{L}_{\text{Cos}}$. The two losses are partially adversarial, thus, optimizing to the Nash equilibrium ends in a network that is effective in detecting continuous object in the space, but also in the detection of sudden transitions to a different class. There are important differences that distinguish our approach from previous works [34, 37]. First, the transformer encoder Ω_ω highly varies the representations fed by the convolutional encoder, thus, no need for a memory bank to avoid $\mathcal{L}_{\text{NCE}}$ collapsing to a trivial solution. Second, we leverage a dynamic threshold for creating the positives/negatives mask instead of using the ground truth bounding boxes.

2.2 Supervised contrastive object detection

DETR Loss. After using the Hungarian algorithm to compute the optimal matching over a fixed set of N predictions. Denote $y_i = (c_i, b_i)$ as a sample from the ground truth set (c_i: class label, $b_i \in [0,1]^4$: bounding box), and corresponding prediction as $\bar{y}_i = (\bar{c}_i, \bar{b}_i)$, the loss function $\mathcal{L}_{\text{sup}}(y_i, \bar{y}_i)$ is defined as: $\mathcal{L}_i^{CE}(c_i, \bar{c}_i) + \mathcal{L}_i^{box}(b_i, \bar{b}_i)$, where:

$$\mathcal{L}_i^{CE} = -\mathbb{1}_{c_i}^T \log(\text{softmax}(\bar{c}_i)) \tag{1}$$

We can observe that the softmax optimizes only for the logits without any access to the learned representations [27]. Moreover, the softmax cross-entropy loss function highly affects the penultimate layers [15], but potentially lacks in structuring the representation space. These issues have been rarely addressed in training object detectors, including DETR-based ones.

Encoder ($\mathbf{f}_e$). The encoder is a network $\mathbf{f_e} : \mathbf{x_e} \mapsto \Gamma$ parameterised by θ_e. $\mathbf{f}_e$ is implemented as a backbone ResNet50 [11], followed by a 2-layer 1×1 convolutional projection head with batch normalization and ReLU activation, that reduces the channels dimension from 2048 to 256.

Momentum encoder ($\mathbf{f}_m$). Following [8, 13], we adopt the momentum update rule for $\mathbf{f_m} : \mathbf{x_m} \mapsto \Psi$, parameterised by θ_m. $\mathbf{f}_m$ architecture is identical to $\mathbf{f}_e$, and is updated as follows:

$$\theta_m \leftarrow \beta\theta_m + (1-\beta)\theta_e, \tag{2}$$

where β is a momentum coefficient (set to 0.99). We place a stop-gradient on θ_m, and only the parameters θ_e are updated by back-propagation. Intuitively, the momentum encoder ($\mathbf{f}_m$) can be seen as a past smoothed version of ($\mathbf{f}_e$).

Transformer encoder. Ω_ω maps $\Gamma^{c\times h\times w}$ to $\gamma^{c\times h\times w}$. Γ is first wrapped to a sequence of size $c \times hw$, then augmented with 2D positional encodings [1]. The multi-head self-attention layers perform message parsing across Γ channels, in order to capture the contextual information. It acts as a smoothing prior, hence, pixels sharing the same semantic class repulsively attend irrespective of their position in the image, dropping all structural information. Furthermore, γ is the smoothed transform of Γ, holding more coherence both spatially in the neighborhood of a given pixel i and semantically for further pixels j sharing the same class. Objects can be fairly queried from such a representation.

Table 1: Results fo DETR-like models on COCO. Superscript $^+$ refers to the two-level cosine loss applied at two levels of patterns.

Model	Epochs	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Deformable DETR [41]	50	43.8	62.6	47.7	26.4	47.1	58.0
Deformable DETR + **ours** $\alpha = 1, \beta = 1$	50	43.8 (+0.0)	62.6 (+0.0)	48.0 (+0.3)	26.6 (+0.2)	47.5 (+0.4)	57.1 (-0.9)
Deformable DETR + **ours** $\alpha = 1, \beta = 0$	50	44.1 (+0.3)	63.2 (+0.8)	48.2 (+0.5)	27.1 (+0.7)	47.5 (+0.4)	57.9 (-0.1)
Deformable DETR + **ours** $\alpha = 0.2, \beta = 1$	50	43.8 (+0.0)	62.8 (+0.2)	47.9 (+0.2)	26.5 (-0.1)	47.0 (-0.1)	58.4 (+0.4)
Deformable DETR + **ours** $\alpha = 0, \beta = 1$	50	44.4 (+0.6)	63.5 (+0.9)	48.6 (+1.1)	27.0 (+0.6)	47.6 (+0.5)	59.2 (+1.2)
DN DETR [18]	50	44.1	64.4	46.7	22.9	48.0	63.4
DN DETR + **ours** $\alpha = 0, \beta = 1$	50	44.3 (+0.2)	64.3 (-0.1)	46.9 (+0.2)	22.9 (+0.0)	47.9 (-0.1)	63.9 (+0.5)
DN Deformable DETR [18]	50	48.6	67.4	52.7	31.0	52.0	63.7
DN Deformable DETR + **ours** $\alpha = 0, \beta = 1$	50	48.9 (+0.3)	67.0 (-0.4)	53.0 (+0.3)	30.4 (-0.7)	51.9 (-0.1)	65.1 (+1.4)
DN Deformable DETR [18]	12	43.4	61.9	47.2	24.8	46.8	59.4
DN Deformable DETR + **ours**$^+$ $\alpha = 0, \beta = 1$	12	43.9 (+0.5)	61.8 (-0.1)	47.3 (+0.1)	26.2 (+1.4)	46.9 (+0.1)	59.2 (-0.2)

2.3 Contrastive loss

Given $\Gamma^{c\times h\times w}$ and $\Psi^{c\times h\times w}$ computed by $\mathbf{f}_e$ and $\mathbf{f}_m$ respectively, the pretext task is to contrast nearby pixels in the input space, but non-aligned in after performing the crops on x, to generate x_e and x_m. The core idea is to preserve locality by maximizing the mutual information between pairs of spatially close pixels via the pixel-level contrastive loss. A key component is to define positives/negatives sets, in such a way that $\mathcal{L}_{\text{NCE}}$ and $\mathcal{L}_{\text{Sup}}$ follow converging landscapes.

Positives/Negatives mask. Using the relative crop coordinates from both views, we simply calculate the spatial distance between pairs of channel locations in the feature maps space [37]. Then, the positives and negatives pairs are obtained based on a threshold 'th':

$$\mathcal{M}(i,j) = \begin{cases} 1 & d(i,j) \leq th, \\ 0 & \text{otherwise}, \end{cases} \tag{3}$$

where i and j are channel locations from the two feature maps, $d(i,j)$ is the normalized spatial distance. Nevertheless, the random resizing in $\mathcal{T}$ to large shapes suitable for detection requires adaptive threshold, thus, we formulate th as $th = \frac{1}{S|C|}\sum_c^{|C|} d(i,j)_c$, ($S = 9$). Using $\mathcal{M}$ for all channels, we compute the pixel-level contrastive loss:

$$\mathcal{L}^i_{\text{NCE}} = -\frac{1}{|\mathcal{P}_i|}\sum_{i^+\in\mathcal{P}_i} \log \frac{\exp\left(\Gamma_i \cdot \Psi^+_i/\tau\right)}{\exp\left(\Gamma_i \cdot \Psi^+_i/\tau\right) + \sum_{j^-\in\mathcal{N}_j}\exp\left(\Gamma_i \cdot \Psi^-_i/\tau\right)}, \tag{4}$$

where the anchor pixel i is located in both views wrapped in the feature maps space (i.e., the mask is calculated using normalized crops coordinate in the feature maps size). $\mathcal{P}_i$ and $\mathcal{N}_i$

Table 2: Results on the BDD100K dataset. Superscript * indicates ResNet-50 Dilated Convolutions (DC). Superscript + refers to the two-level cosine loss.

Model	Epochs	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
DETR [4]	150	20.9	44.1	17.0	7.2	24.4	39.9
Deformable DETR [41]	50	31.1	57.7	28.5	14.8	36.6	52.4
Deformable DETR + **ours**	50	31.5	58.0	29.1	15.1	36.7	52.6
$\alpha = 0, \beta = 1$		(+0.4)	(+0.3)	(0.3)	(+0.3)	(+0.1)	(+0.2)
DN-Deformable DETR*[18]	50	35.2	62.4	33.0	17.3	41.1	58.3
DN-Deformable DETR* + **ours**$^+$	50	36.1	63.3	34.3	18.2	41.2	58.4
$\alpha = 0, \beta = 1$		(+0.9)	(+0.9)	(+1.3)	(+0.9)	(+0.1)	(+0.1)

are vector representations of positives and negatives in the second view assigned for pixel i using $\mathcal{M}$. τ is a temperature scalar sharpening the distribution, set to 0.3. all representations are l_2 normalized, $\cdot$ denotes the inner (dot) product. The final loss is the average $\mathcal{L}^i_{\text{NCE}}$ of all anchors i lying in the intersection of x_e and x_m as shown in Figure 1. The supervised loss $\mathcal{L}_{\text{sup}}$ optimizes better features for classification and localization; $\mathcal{L}_{\text{NCE}}$ constraints the latent space to learn improved spatial sensitivity across discriminative pixels, leading to improved results.

2.4 Non-contrastive loss

The threshold value impacts various object sizes, as negative samples may fall within the same bounding box, resulting in ambiguous optimization. Thus, we attempt to discard the negative pairs [8], thus, optimizing only for spatial smoothness. The transformer encoder enforces strict asymmetry by effectively updating Γ to produce γ, acting as a variational prior on CNN-based features to better suit transformer decoding. The task is to measure features consistency among γ, the loss is calculated against Ψ, which holds discriminative pixel features to some extent [37]. We rely on the transformer encoder expressive power breaking the symmetry to avoid collapse without the need of negative samples. We formulate the loss function as:

$$\mathcal{L}^i_{\text{Cos}} = \frac{1}{|\mathcal{P}_i|} \sum_{i^+ \in \mathcal{P}_i} -\gamma_i \Psi^+_i. \tag{5}$$

The mask in Equation (3) is used to retrieve the set of positive pairs $\mathcal{P}_i$ for an anchor pixel i. The final loss function is given by:

$$\mathcal{L}_T = \mathcal{L}_{\text{Sup}} + \alpha \mathcal{L}_{\text{NCE}} + \beta \mathcal{L}_{\text{Cos}}, \tag{6}$$

where α and β are hyper-parameters controlling the contribution of $\mathcal{L}_{\text{NCE}}$ and $\mathcal{L}_{\text{Cos}}$.

3 Experimental results

We evaluate the proposed algorithm by measuring the common object detections metrics first on COCO 2017 [19], then on the cars driving object detection dataset BDD100K [39] that contains 70000/10000 *train*/*val* images, covering 10 classes. We train DETR-like models (i.e. Deformable DETR [41], DN-DETR [18] with the proposed method for further validation of the hypoethsis, the choice is motivated by the fast convergence within 50 epochs as compared

Table 3: Results for transferring COCO trained weights on the BDD100K dataset. Superscript $^{w/o}$ indicates training without our method, whereas superscript ^{w} is with our method.

Model	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Linear probes						
DN-Deformable-DETR$^{w/o}$	27.9	51.3	25.8	11.8	32.4	53.9
DN-Deformable-DETR^{w}	28.3	51.6	26.3	11.9	32.5	54.0
Fine tuning						
DN-Deformable-DETR$^{w/o}$	33.0	59.3	31.0	14.8	36.9	58.5
DN-Deformable-DETR^{w}	33.4	59.9	31.9	15.5	38.0	57.8

to DETR [4] (500 epochs). To match their training settings, we initialize the backbones using ImageNet [16] pretrained weights, but randomly initialize the transformer layers. We adopt the set of augmentations used in BYOL [8] for x_m. The proposed method could be applied to other non-transformer based detector with careful design to avoid collapse.

Technical details. We add the PyTorch implementation of our approach to each of the respective models. Training parameters are kept the same as the original approaches. Experiments are conducted on four Nvidia A 100 GPUs with 80 GB memory per card. The batch size is set to 12 for most experiments, unless mentioned otherwise.

3.1 Main results

Table 1 highlights the main results on the COCO 2017 validation set. We augment the different approaches training procedure with the proposed method. We first evaluate the core hypothesis on Deformable-DETR [41] deriving four variations using different values of α and β in Equation (6). Jointly training both $\mathcal{L}_{\text{NCE}}$ and $\mathcal{L}_{\text{Cos}}$ (i.e. $\alpha = 1, \beta = 1$) does not improve the performance, precisely, AP_L drops by (-0.9), this may be explained by the negative terms in $\mathcal{L}_{\text{NCE}}$, where representations per channel of the same large object are split between negatives and positives due to the threshold in Equation (3). Lowering the contribution of $\mathcal{L}_{\text{NCE}}$ in the gradients ($\alpha = 0.2$ and $\beta = 1$) solves the issue of AP_L degradation **(+0.4)**. Clearly, however, it does not improve the global performance among all metrics. This may also suggests that $\mathcal{L}_{\text{NCE}}$ and $\mathcal{L}_{\text{Cos}}$ both optimize for spatial smoothness/sensitivity through diverging landscapes, and can not be jointly optimized. Optimizing for $\mathcal{L}_{\text{NCE}}$ results in a slight overall improvement **(+0.3)** in AP. Finally, the best scores across all metrics are achieved by setting $\alpha = 0$. AP_{50} and AP_{75} gain **(+0.9)** and **(+1.1)** respectively, meaning $\mathcal{L}_{\text{Cos}}$ encourages more accurate bounding boxes. Larger improvements **(+1.2)** are observed on AP_L. This can be linked to better spatial smoothing, resulting in compact representations across the channels of larger objects. We train the remaining approaches with this configuration (i.e. $\alpha = 0, \beta = 1$).

Lastly, we first train DN-DETR Li et al. [18] using full self-attention as in DETR [4]. With one pattern encoding, a small gain is observed, questioning the effectiveness of $\mathcal{L}_{\text{Cos}}$ on a single pattern scheme. Then, DN-Deformable DETR exploits the deformable attention, hence, being able to use three pattern encodings. We observe a slight improvement in AP **(+0.3)**, a surprising degradation in AP_S and AP_M, and a large improvement in AP_L **(+1.4)**. A potential hypothesis can be related to $\mathcal{L}_{\text{Cos}}$ operating on the last feature maps of size ($c \times wh$) with the smallest spatial resolution (i.e., save memory usage as mask size in Equation (3) is ($c \times wh \times wh$)) among the 3 patterns. Thus, over-smoothing this feature maps misrepresented the small and medium size objects, which usually lie within higher resolution maps.

Object detection for cars driving on BDD100K. To further validate the effectiveness of the proposed method, we conducted experiments on a challenging driving dataset BDD100K [39]. It holds 100K driving videos collected under diverse scene types including city streets, residential areas, and highways, during shifting weather conditions at different times of the day. Each video is 40-seconds long and with a FPS of 30. Thus, there are more than 100 million frames in total. Among these frames, only parts of them are labeled for detection (1 frame in each video), with 70K, 10K, and 20K labeled images for $train/val/test$, respectively. Altogether, there are 1.8 million labeled objects representing 10 classes, including bus, light, sign, person, bike, truck, motor, car, train, and ride. Comparing with the commonly used COCO 2017, BDD100K is more challenging in two aspects: (1) over 55% of objects are of small size, which requires high resolution in the feature map; and (2) the images are captured in both daytime and nighttime, posing diverse illumination conditions. We first trained the existing approaches on the 70k training set of BDD100K as baselines (see Supplementary for full results). Table 2 shows that all metrics drop by a large margin compared to COCO (i.e. AP for Deformable DETR drops by **(12.7)**), highlighting the complexity of this task. The motivation behind this choice is to measure the model's capacity on a more complex distribution. Table 2 depicts that our method slightly improve the metrics over the baselines. DN-Deformable DETR* + **ours**$^+$ optimizes $\mathcal{L}_{\text{Cos}}$ on two distinctive feature maps of the multiple patterns training using different thresholds provide for better features for all sized objects. This setting exhibits the top score for all metrics (i.e. AP **(+0.9)**).

How does pre-taining on COCO transfer to BDD100K? We study transferring pre-trained COCO features on BDD100K using the 12 epochs training setting. Using DN-Deformable DETR, we first freeze the network and train only the classification head. Then, we proceed with fine-tuning the whole architecture. It can be noticed that initializing with weights trained leveraging our method surpasses vanilla training across all metrics. This demonstrates the effectiveness of our method on out of distribution generalization.

Instance segmentation Following [4, 17, 29], we train a segmentation head on top of the frozen weights using some configurations mentioned in Table 1. This measures the generalization capacity of each hypothesis space on a disjoint downstream task. We argue that a regularized space with the aforementioned prior properties should be suitable for instance segmentation. The segmentation head consists of 5-blocks of (2D Convolution, Group Normalization [36], ReLU), augmented with three FPN layers [20], it outputs a binary mask for each detected object. We first initialized with vanilla Deformable-DETR weights and trained the segmentation head for 15 epochs using the DICE/F-1 loss [25]. This setting achieves the following scores: [AP: 21,5, AP_S: 17.8, AP_M:23.6, AP_L:26.6]. Then, Deformable-DETR+**ours** weights lead to a notable gain of **(1.5)**, (i.e. [AP: 21,5 $\rightarrow$ 23.0, AP_S: 17.8 $\rightarrow$ 20.2, AP_M:23.6 $\rightarrow$ 24.9, AP_L:26.6] $\rightarrow$ 26.9). Results on instance segmentation validate the hypothesis that enforcing spatial smoothness on the latent representation favors better generalization on a dense prediction task unlike object detection.

4 Conclusion and future works

We introduced a novel training framework for Detection Transformer based models leveraging an objective function at the latent space level to enforce prior knowledge, which improves the generalization and expressive power of the model. The objective function explores pixels semantic consistency. Hence, spatially close pixel features are encouraged to maximally share information. We outline two important directions to be addressed:

Robustness against adversarial attacks. Object detectors are intrinsically vulnerable to adversarial attacks [24] such as DPATCH [23], or latent space noise perturbations [3]. Studying the effect of these attacks on a regularized latent space is interesting, and can potentially bring new insights to the community in terms of model interpretability.

Intermediate objectives impact on the transformer-encoder layers. [15] measured the similarity between hidden representations of networks using the centered kernel alignment metric [14], and noticed that differences among loss functions are apparent only in the last layers of the network, and do not impact the latent space. Exploring the effect of the pixel-level consistency loss on the transformer-encoder layers, and addressing the question of whether this objective forces the layer to be semantically discriminative is a promising line of research.

Furthermore, the proposed method leads to encouraging results and demonstrates excellent transfer-ability to other dense image prediction tasks such as instance segmentation. However, this training paradigm raises new challenges such as balancing the different loss functions, and carefully creating the negative/positive pairs to handle varying sized objects. We believe this is an important step towards explicitly learning a good representation space for the object detection task.

References

[1] Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V Le. Attention augmented convolutional networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3286–3295, 2019.

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[3] Igor Buzhinsky, Arseny Nerinovsky, and Stavros Tripakis. Metrics and methods for robustness evaluation of neural networks with generative models. *Machine Learning*, pages 1–36, 2021.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[6] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

[7] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1601–1610, 2021.

[8] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[9] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

[10] Michael U Gutmann and Aapo Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, 13(2), 2012.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[14] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pages 3519–3529. PMLR, 2019.

[15] Simon Kornblith, Ting Chen, Honglak Lee, and Mohammad Norouzi. Why do better loss functions lead to less transferable features? *Advances in Neural Information Processing Systems*, 34:28648–28662, 2021.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[17] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934, 2020.

[18] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13619–13627, 2022.

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[20] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[21] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[22] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. Dab-detr: Dynamic anchor boxes are better queries for detr. *arXiv preprint arXiv:2201.12329*, 2022.

[23] Xin Liu, Huanrui Yang, Ziwei Liu, Linghao Song, Hai Li, and Yiran Chen. Dpatch: An adversarial patch attack on object detectors. *arXiv preprint arXiv:1806.02299*, 2018.

[24] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[25] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE, 2016.

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

[27] Tianyu Pang, Kun Xu, Yinpeng Dong, Chao Du, Ning Chen, and Jun Zhu. Rethinking softmax cross-entropy loss for adversarial robustness. *arXiv preprint arXiv:1905.10626*, 2019.

[28] 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.

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[30] Zhiqing Sun, Shengcao Cao, Yiming Yang, and Kris M Kitani. Rethinking transformer-based set prediction for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3611–3620, 2021.

[31] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *CoRR*, abs/1906.05849, 2019. URL http://arxiv.org/abs/1906.05849.

[32] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019.

[33] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020.

[34] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7303–7313, 2021.

[35] Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. Anchor detr: Query design for transformer-based object detection. *arXiv preprint arXiv:2109.07107*, 2021.

[36] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[37] Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16684–16693, 2021.

[38] Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang. Efficient detr: improving end-to-end object detector with dense prior. *arXiv preprint arXiv:2104.01318*, 2021.

[39] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020.

[40] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

[41] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.