Time-rEversed diffusioN tEnsor Transformer: A new TENET of Few-Shot Object Detection

Shan Zhang[⋆,†], Naila Murray[♣], Lei Wang[♦], and Piotr Koniusz[⋆,§,†]

[†]Australian National University [♣]Meta AI
[♦]University of Wollongong [§]Data61/CSIRO
[†]firstname.lastname@anu.edu.au, [♦]leiw@uow.edu.au, [♣]murrayn@fb.com

Abstract. In this paper, we tackle the challenging problem of Few-shot Object Detection. Existing FSOD pipelines (i) use average-pooled representations that result in information loss; and/or (ii) discard position information that can help detect object instances. Consequently, such pipelines are sensitive to large intra-class appearance and geometric variations between support and query images. To address these drawbacks, we propose a Time-rEversed diffusioN tEnsor Transformer (TENET), which i) forms high-order tensor representations that capture multi-way feature occurrences that are highly discriminative, and ii) uses a transformer that dynamically extracts correlations between the query image and the entire support set, instead of a single average-pooled support embedding. We also propose a Transformer Relation Head (TRH), equipped with higher-order representations, which encodes correlations between query regions and the entire support set, while being sensitive to the positional variability of object instances. Our model achieves state-of-the-art results on PASCAL VOC, FSOD, and COCO.

Keywords: Few-Shot Object Detection; Transformer; Multiple order pooling; Heat Diffusion Process

1 Introduction

Object detectors based on deep learning, usually addressed by supervised models, achieve impressive performance [32,33,34,8,27,13] but they rely on a large number of images with human-annotated class labels/object bounding boxes. Moreover, object detectors cannot be easily extended to new class concepts not seen during training. Such a restriction limits supervised object detectors to predefined scenarios. In contrast, humans excel at rapidly adapting to new scenarios by *"storing knowledge gained while solving one problem and applying it to a different but related problem"* [43], also called a "*transfer of practice*" [44].

Few-shot Object Detection (FSOD) [4,11,12,48,7,51,52] methods mimic this ability, and enable detection of test classes that are disjoint from training classes.

[⋆] SZ was mainly in charge of the pipeline/developing the transformer. PK (corresponding author) was mainly in charge of mathematical design of TENET & TSO. Code: `https://github.com/ZS123-lang/TENET`.

They perform this adaptation using a few “support” images from test classes. Successful FSOD models must (i) find promising candidate regions-of-Interest (RoIs) in query images; and (ii) accurately regress bounding box locations and predict RoI classes, under large intra-class geometric and photometric variations.

To address the first requirement, approaches [48,7,51,52] use the region proposal network [34]. For example, FSOD-ARPN [7], PNSD [51] and KFSOD [52] cross-correlate query feature maps with a class prototype formed from average-pooled (*ie*., first-order) features, second-order pooled representations and kernel-pooled representations, respectively. These methods use a single class prototype which limits their ability to leverage diverse information from different class samples. Inspired by Transformers [41], approach [23] uses average pooling over support feature maps to generate a vector descriptor per map. Attention mechanism is then used to modulate query image features using such descriptors.

The above methods rely on first- and second-order pooling, while so-called higher-order pooling is more discriminative [17,18,15]. Thus, we propose a non-trivial Time-rEversed diffusioN tEnsor Transformer (TENET). With TENET, higher-order tensors undergo a time-reversed heat diffusion to condense signal on super-diagonals of tensors, after which coefficients of these super-diagonals are passed to a Multi-Head Attention (MHA) transformer block. TENET performs second-, third- and fourth-order pooling. However, higher-order pooling suffers from several issues, *ie*., (i) high computational complexity of computing tensors with three/more modes, (ii) non-robust tensor estimates due to the limited number of vectors being aggregated, and (iii) tensor burstiness[17].

To this end, we propose a Tensor Shrinkage Operator (TSO) which generalizes spectral power normalization (SPN) operators [18], such as the Fast Spectral MaxExp operator (MaxExp(F)) [18], to higher-order tensors. As such, it can be used to reduce tensor burstiness. Moreover, by building on the linear algebra of the heat diffusion process (HDP) [36] and recent generalisation of HDP to SPN operators [18], we also argue that such operators can reverse the diffusion of signal in autocorrelation or covariance matrices, and high-order tensors, instead of just reducing the burstiness. Using a parametrization which lets us control the reversal of diffusion, TSO condenses signal captured by a tensor toward its super-diagonal, preserving information along it. This super-diagonal serves as our final representation, reducing the feature size from d^r to d, making our representation computationally tractable. Finally, shrinkage operators are known for their ability to estimate covariances well when only a small number of samples are available [22]. To the best of our knowledge, we are the first to show that MaxExp(F) is a shrinkage operator, and to propose TSO for orders $r \geq 2$.

To address the second requirement, FSOD-ARPN introduces a multi-relation head that captures global, local and patch relations between support and query objects, while PNSD passes second-order autocorrelation matrices to a similarity network. However, FSOD-ARPN and PNSD do not model spatial relations [9]. The QSAM [23] uses attention to highlight the query RoI vectors that are similar to the set of support vectors (obtained using only first-order spatial average pooling). Thus, we introduce a Transformer Relation Head (TRH) to improve modeling of spatial relations. TRH computes self-attention between spatially-aware features and global spatially invariant first-, second- and higher-order TENET representations of support and/or query RoI features. The second attention mechanism of TRH performs cross-attention between Z support embeddings (for Z-shot if $Z \geq 2$), and a set of global representations of query RoIs. This attention encodes similarities between query RoIs and support samples.

The TENET RPN and the TRH blocks of our FSOD pipeline are equipped with discriminative TENET representations, improving both RoI proposal generation, and the encoding of relations between query and support features.

Below are our contributions:

i. We propose a Time-rEversed diffusiON tEnsor Transformer unit, called TENET, which captures high-order patterns (including multi-way feature occurrences) and decorrelates them/reduces tensor burstiness. To this end, we generalize the MaxExp(F) operator [18] for autocorrelation and/or covariance matrices to higher-order tensors by introducing the so-called Tensor Shrinkage Operator (TSO).

ii. We propose a Transformer Relation Head (TRH) that is sensitive both to the variability between the Z support samples provided in a Z-shot scenario, and to positional variability between support and query objects.

iii. In our Supplementary Material (§A), we demonstrate that TSO emerges from the MLE-style minimization over the Kullback-Leibler (KL) divergence between the input and output spectrum, with the latter being regularized by the Tsallis entropy [1]. Thus, we show that TSO meets the definition of shrinkage estimator whose target is the identity matrix (tensor).

Our proposed method outperforms the state of the art on novel classes by 4.0%, 4.7% and 6.1% mAP on PASCAL VOC 2007, FSOD, and COCO respectively.

2 Related Works

Below, we review popular FSOD methods and vision transformers, followed by a short discussion on feature grouping, tensor descriptors and spectral power normalization.

Few-shot Object Detection. A Low-Shot Transfer Detector (LSTD) [4] leverages rich source domain to construct a target domain detector with few training samples but needs to be fine-tuned to novel categories. Meta-learning-based approach [48] reweights RoI features in the detection head without fine-tuning. Similarly, MPSR [46] deals with scale invariance by ensuring the detector is trained over multiple scales of positive samples. NP-RepMet [49] introduces a negative- and positive-representative learning framework via triplet losses that bootstrap the classifier. FSOD-ARPN [7] is a general FSOD network equipped with a channel-wise attention mechanism and multi-relation detector that scores pair-wise object similarity in both the RPN and the detection head, inspired by Faster R-CNN. PNSD [51], inspired by FSOD-ARPN [7], uses contraction of second-order autocorrelation matrix against query feature maps to produce attention maps. Single-prototype (per class) methods suffer information loss. Per-sample Prototype FSOD [23] uses the entire support set to form prototypes of a class but it ignores spatial information within regions. Thus, we employ TENET RPN and TRH to capture spatial and high-order patterns, and extract correlations between the query image and the Z-shot support samples for a class.

Transformers in Vision. Transformers, popular in natural language processing [41], have also become popular in computer vision. Pioneering works such as ViT [9] show that transformers can achieve the state of the art in image recognition. DETR [3] is an end-to-end object detection framework with a transformer encoder-decoder used on top of backbone. Its deformable variant [53] improves the performance/training efficiency. SOFT [30], the softmax-free transformer approximates the self-attention kernel by replacing the softmax function with Radial Basis Function (RBF), achieving linear complexity. In contrast, our TENET is concerned with reversing the diffusion of signal in high-order tensors via the shrinkage operation, with the goal of modeling spatially invariant high-order statistics of regions. Our attention unit, so-called Spatial-HOP in TRH, also uses RBF to capture correlations of between spatial and high-order descriptors.

Multi-path and Groups of Feature Maps. GoogleNet [38] has shown that multi-path representations (several network branches) lead to classification improvements. ResNeXt [47] adopts group convolution [20] in the ResNet bottleneck block. SK-Net [25], based on SE-Net [10], uses feature map attention across two network branches. However, these approaches do not model feature statistics. Somewhat closer to our idea is bilinear pooling [35], which correlates two groups of feature channels from two regions, whereas ReDRO [31] samples groups of features to apply the matrix square root over submatrices to improve the computational speed. In contrast, for TENET, we form fixed groups of features to form second-, third- and fourth-order tensors, and we show that simply using groups of features to form second-order matrices is not effective.

Second-order Pooling (SOP). Region Covariance Descriptors for texture [39,40] and object category recognition [17] use SOP. SOP [19] uses spectral pooling for fine-grained image classification, whereas SoSN [50] leverages SOP and element-wise Power Normalization (PN) [17] for end-to-end few-shot learning. In contrast, we develop a few-shot detector that tackles multi-object localization and classification. Similarly to SoSN, PNSD [51] uses SOP with PN as representations which are passed to the detection head. So-HoT [14] that uses high-order tensors for domain adaptation is also somewhat related to TENET but So-HoT uses multiple polynomial kernel matrices, whereas we apply TSO to achieve decorrelation and shrinkage. TENET without TSO reduces to polynomial feature maps and performs poorly.

Power Normalization. PN [16] limits the so-called burstiness of first- and second-order statistics, which is 'the property that a given visual element appears more times in an image than a statistically independent model would predict', due to the binomial PMF-based feature detection factoring out feature counts [16,17,19]. Element-wise MaxExp pooling [16] gives likelihood of 'at least one particular visual word being present in an image', whereas SigmE pooling [19] is its practical approximation. Noteworthy are the recent Fast Spectral MaxExp operator, MaxExp(F) [18], which reverses the heat diffusion on the underlying loopy graph of second-order matrix to some desired past state [36], and Tensor Power-Euclidean (TPE) metric [15]. TPE alas uses the Higher Order Singular Value Decomposition [21], which makes TPE intractable for hundreds of region proposals per image, and hundreds of thousands of images per epoch. Thus, we develop TENET and TSO, which reverses diffusion on high-order tensors by shrinking them towards the tensor's super-diagonal.

3 Background

Below, we detail notations/tensor algebra pre-requisites, and demonstrate how to calculate multiple higher-order statistics and Power Normalization functions, followed by revisiting of Transformer block.

Notations. Let $x \in \mathbb{R}^d$ be a d-dimensional feature vector. $\mathcal{I}_N$ stands for the index set $\{1, 2, ..., N\}$. We define a vector of all-ones as $\mathbf{1} = [1, ..., 1]^T$. Let $\boldsymbol{\mathcal{X}} = \uparrow\!\otimes_r x$ denote a tensor of order r generated by the r-th order outer-product of x, $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{d_1 \times d_2 \ldots \times d_r}$. Typically, capitalised boldface symbols such as $\boldsymbol{\varPhi}$ denote matrices, lowercase boldface symbols such as $\boldsymbol{\phi}$ denote vectors and regular case such as $\varPhi_{i,j}$, ϕ_i, n or Z denote scalars *e.g.*, $\varPhi_{i,j}$ is the (i, j)-th coefficient of $\boldsymbol{\phi}$.

High-order Tensor Descriptors (HoTD). Below we formalize the notion of higher-order descriptors [14].

Proposition 1. *Let $\boldsymbol{\varPhi} \equiv \{\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_N \in \mathbb{R}^d\}$ and $\boldsymbol{\varPhi}' \equiv \{\boldsymbol{\phi}'_1, ..., \boldsymbol{\phi}'_M \in \mathbb{R}^d\}$ be feature vectors extracted from some two image regions. Let $\boldsymbol{w} \in \mathbb{R}^N_+$, $\boldsymbol{w}' \in \mathbb{R}^M_+$ be some non-negative weights and $\boldsymbol{\mu}, \boldsymbol{\mu}' \in \mathbb{R}^d$ be the mean vectors of $\boldsymbol{\varPhi}$ and $\boldsymbol{\varPhi}'$, respectively. A linearization of the sum of polynomial kernels of degree r,*

$$\begin{aligned}&\langle \boldsymbol{\mathcal{M}}(\boldsymbol{\varPhi}; \boldsymbol{w}, \boldsymbol{\mu}),\, \boldsymbol{\mathcal{M}}(\boldsymbol{\varPhi}'; \boldsymbol{w}', \boldsymbol{\mu}') \rangle = \\ &\frac{1}{NM} \sum_{n=1}^{N} \sum_{m=1}^{M} w_n^r w_m'^r \, \langle \boldsymbol{\phi}_n - \boldsymbol{\mu}, \boldsymbol{\phi}'_m - \boldsymbol{\mu}' \rangle^r,\end{aligned} \tag{1}$$

yields the tensor feature map

$$\boldsymbol{\mathcal{M}}(\boldsymbol{\varPhi}; \boldsymbol{w}, \boldsymbol{\mu}) = \frac{1}{N} \sum_{n=1}^{N} w_n^r \uparrow\!\otimes_r (\boldsymbol{\phi}_n - \boldsymbol{\mu}) \in \mathbb{R}^{d \times d \ldots \times d}. \tag{2}$$

For brevity, in our paper we set $\boldsymbol{w} = \boldsymbol{w}' = 1$ and $\boldsymbol{\mu} = \boldsymbol{\mu}' = 0$, whereas orders $r = 2, 3, 4$. Specifically, we formulate the second/third/fourth-order feature map as: $\boldsymbol{\mathcal{M}}{\times_2}(\boldsymbol{\varPhi}) = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\phi}_n \boldsymbol{\phi}_n^T \in \mathbb{R}^{d \times d}$; $\boldsymbol{\mathcal{M}}{\times_3}(\boldsymbol{\varPhi}) = \frac{1}{N} \sum_{n=1}^{N} (\boldsymbol{\phi}_n \boldsymbol{\phi}_n^T) \boldsymbol{\phi}_n^T \in \mathbb{R}^{d \times d \times d}$; $\boldsymbol{\mathcal{M}}{\times_4}(\boldsymbol{\varPhi}) = \frac{1}{N} \sum_{n=1}^{N} (\boldsymbol{\phi}_n \boldsymbol{\phi}_n^T)(\boldsymbol{\phi}_n \boldsymbol{\phi}_n^T) \in \mathbb{R}^{d \times d \times d \times d}$

(Eigenvalue) Power Normalization ((E)PN). For second-order matrices, MaxExp(F), a state-of-the-art EPN [18], is defined as

$$g(\lambda;\eta)=1-(1-\lambda)^{\eta} \tag{3}$$

on the ℓ_1-norm normalized spectrum from SVD ($\lambda_i := \lambda_i/(\sum_{i'} \lambda_{i'} + \varepsilon)$), and on symmetric positive semi-definite matrices as

$$\widehat{\mathcal{G}}_{\text{MaxExp}}(\boldsymbol{M};\eta)=\mathbb{I}-(\mathbb{I}-\boldsymbol{M})^{\eta}\,, \tag{4}$$

where $\boldsymbol{M}$ is a trace-normalized matrix, that is, $\boldsymbol{M} := \boldsymbol{M}/(\text{Tr}(\boldsymbol{M})+\varepsilon)$ for some small $\varepsilon \approx 1e-6$, and $\text{Tr}(\cdot)$ defines the trace of tensor. The time-reversed heat diffusion process is adjusted by integer $\eta \geq 1$. The larger the value of η is, the more prominent the time reversal is. $\widehat{\mathcal{G}}_{\text{MaxExp}}$ is followed by the element-wise PN, called SigmE [18]:

$$\mathcal{G}_{\text{SigmE}}(p;\eta') = 2/(1+e^{-\eta' p})-1, \tag{5}$$

where p takes each output entry of Eq. (4), $\eta' \geq 1$ controls detecting feature occurrence *vs.* feature counting trade-off.

Transformers The transformer [41] is a network architecture based on blocks of alternating attention and MLP layers. Each attention layer takes as input a set of query ($\boldsymbol{Q}$), key ($\boldsymbol{K}$) and value ($\boldsymbol{V}$) matrices. Let $\boldsymbol{Q} \equiv \{\mathbf{q}_1, ..., \mathbf{q}_N \in \mathbb{R}^d\}$, $\boldsymbol{K} \equiv \{\mathbf{k}_1, ..., \mathbf{k}_N \in \mathbb{R}^d\}$, and $\boldsymbol{V} \equiv \{\mathbf{v}_1, ..., \mathbf{v}_N \in \mathbb{R}^d\}$, where N is the number of input feature vectors, also called tokens, and d is the channel dimension. A generic attention layer can then be formulated as:

$$\text{Atten}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \alpha(\boldsymbol{Q},\boldsymbol{K})\boldsymbol{V}^T. \tag{6}$$

The key self-attention function α is composed of a nonlinear function β and a relation function γ. A dominant instantiation of α is the scaled dot-product based softmax function[41], defined as:

$$\beta = \text{softmax}(\cdot), \quad \gamma(\boldsymbol{Q},\boldsymbol{K}) = \frac{\boldsymbol{Q}^T\boldsymbol{K}}{\sqrt{d}}. \tag{7}$$

In addition, LayerNorm (LN) and residual connections are added at the end of each block.

To facilitate the design of linear self-attention, [30] introduces a softmax-free self-attention function with the dot-product replaced by a Gaussian kernel as:

$$\beta = \exp(\cdot), \quad \gamma(\mathbf{q}_i,\mathbf{k}_j) = \frac{-\|\mathbf{q}_i-\mathbf{k}_j\|_2^2}{2\sigma^2}, \tag{8}$$

where $(i,j) \in \mathcal{I}_N$, σ^2 is the kernel variance set by cross-validation, and $\mathbf{q}_i$ and $\mathbf{k}_j$ are l_2-normalized.

The multi-head attention layer is an enhancement of the attention layer, where T attention units are applied and their outputs are then concatenated together. Concretely, this operation splits input matrices $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ along their channel dimension d into T groups and performs attention on each group:

$$\text{MHA}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \text{Concat}(head_1,\ldots,head_T), \tag{9}$$

where $head_m$=Atten$(\boldsymbol{Q}_m,\boldsymbol{K}_m,\boldsymbol{V}_m)$, Concat concatenates along the channel dimension, and the inputs $\boldsymbol{Q}_m \in \mathbb{R}^{d/T\times N}$, $\boldsymbol{K}_m \in \mathbb{R}^{d/T\times N}$, and $\boldsymbol{V}_m \in \mathbb{R}^{d/T\times N}$ form the m^{th} group.

4 Proposed Approach

Our approach follows the paradigm of learning a matching function between representations of query RoIs and representations of support RoIs supplied during testing episodes. Our approach is illustrated in Fig. 1 for a set of Z support RoIs $\{\boldsymbol{X}\}$ and a query image $\boldsymbol{X}^*$. It comprises three main modules:

i. an **encoding network** (EN) which is used to extract features from both query and support images. We use ResNet-50 as our EN. Given an input image, EN outputs a feature map $\boldsymbol{\Phi} \in \mathbb{R}^{d\times N}$, where $N = H \times W$ and H and W denote the spatial dimensions of the feature map;
ii. an **embedding and RoI extraction module** which generates RoIs from the query image and computes discriminative embeddings for both the resultant query RoIs as well as the support RoIs; and
iii. a **transformer relation** head which encodes relations between query and support features and embedddings using self- and cross-attention. This head outputs a set of representations, one per RoI, that is then fed into a classifier and bounding-box refinement regressor.

Next, we describe the embedding and RoI extraction module, followed by the transformer relation head.

4.1 Extracting representations for support and query RoIs

Our module for generating embeddings and query RoIs is shown in Fig. 1. Central to this module is our HOP unit for generating discriminative embeddings by aggregating over higher-order tensor descriptors (HoTDs). In this unit, features are split along the channel mode to form multiple feature map tensors, from which second-, third- and fourth- order tensors are computed. The aggregation operator is a generalization of the MaxExp(F) to higher-order tensors, called the tensor shrinkage operator (TSO). Using the TSO, derived later, we extract representations $\hat{\boldsymbol{\psi}}_r$ from HoTDs $\boldsymbol{\mathcal{M}}^{(r)}$ as follows:

$$\hat{\boldsymbol{\psi}}_r = \text{Diag}\left(\widehat{\boldsymbol{\mathcal{G}}}_{\text{TSO}}\left(\boldsymbol{\mathcal{M}}^{(r)};\eta_r\right)\right), \tag{10}$$

$$\boldsymbol{\psi}_r = \boldsymbol{\mathcal{G}}_{\text{SigmE}}\left(\hat{\boldsymbol{\psi}}_r;\eta'\right), \tag{11}$$

where $\widehat{\boldsymbol{\mathcal{G}}}_{\text{TSO}}(\cdot)$ is the TSO, and Diag is the super-diagonal extraction operation. We now derive the TSO as a generalization of MaxExp(F).

Tensor Shrinkage Operator. Ledoit and Wolf [22] define autocorrelation / covariance matrix estimation as a compromise between the sample matrix $\boldsymbol{M}$ and a highly structured operator $\boldsymbol{F}$, which achieved by computing a convex linear combination $(1-\delta)\boldsymbol{M}+\delta\boldsymbol{F}$. For symmetric positive semi-definite matrices and other representations which rely on some spectrum, one can estimate in a similar manner, by minimizing some divergence $d(\boldsymbol{\lambda},\boldsymbol{\lambda}')$ between the source and target spectra, where $\boldsymbol{\lambda}'$ is regularized by $\Omega(\boldsymbol{\lambda}')$:

$$\boldsymbol{\lambda}^* = \underset{\boldsymbol{\lambda}'\geq 0}{\arg\min}\, d(\boldsymbol{\lambda},\boldsymbol{\lambda}') + \delta\Omega(\boldsymbol{\lambda}'). \tag{12}$$

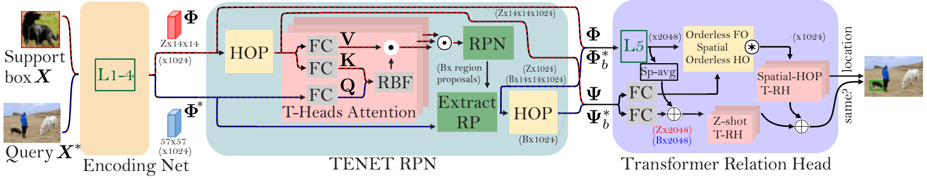


Fig. 1: Our pipeline. We input ground truth support crops and the query image to the encoding network (EN). The resulting convolutional feature maps ($\boldsymbol{\Phi}$ for support, $\boldsymbol{\Phi}^*$ for query) are input to the TENET-RPN module to produce feature sets for query image RoIs ($\boldsymbol{\Phi}_b^*$), and high-order pooled representations for both support crops ($\boldsymbol{\Psi}$) and query image RoIs ($\boldsymbol{\Psi}_b^*$). TENET contains HOP units which compute high-order tensor descriptors and then apply a novel tensor shrinkage operator to them, yielding more discriminative representations ($\boldsymbol{\Psi}$ and $\boldsymbol{\Psi}_b^*$) with identical dimensionality to the input features. These more discriminative representations are then passed to the Transformer relation head (TRH), along with the convolutional features $\boldsymbol{\Phi}$ and $\boldsymbol{\Phi}_b^*$. The TRH consists of Z-shot and Spatial-HOP attention units for measuring similarities across support regions and query proposals, and for refining the localization of target objects. The HOP embeddings of support images and query proposals are $\boldsymbol{\Psi}$ (Z vector) and $\boldsymbol{\Psi}_b^*$ ($b \in \boldsymbol{\mathcal{B}}$, a set of query proposals). $\oplus$ denotes element-wise addition and $\odot$ means matrix multiplication.

Let λ be the ell_1-norm normalized spectrum, as in Eq. (3). Then, $g(\lambda;\eta) = 1-(1-\lambda)^\eta$ is in fact a shrinkage operator, a solution to the problem in Eq. (12) with $\delta = 1$, and the Kullback-Leibler divergence and the Tsallis entropy substituting the divergence d and the regularization term $\varOmega$, respectively. Please refer to §A of supplementary material for the proof, where we also discuss the highly structured operator $\boldsymbol{F}$, *ie*. the target of the shrinkage operator, is the identity matrix.

Based on these observations, we can readily extend MaxExp(F) to general high-order tensors with TSO as follows:

$$\widehat{\boldsymbol{\mathcal{G}}}_{\text{TSO}}\left(\boldsymbol{\mathcal{M}}^{(r)};\eta\right) = \boldsymbol{\mathcal{I}}_r - \left(\boldsymbol{\mathcal{I}}_r - \boldsymbol{\mathcal{M}}^{(r)}\right)^\eta \tag{13}$$

where the identity tensor of order r is defined as $\boldsymbol{\mathcal{I}}_r$, where all elements $\mathcal{I}_{1,\ldots,1} = \mathcal{I}_{2,\ldots,2} \ldots = \mathcal{I}_{d,\ldots,d} = 1$ and $\mathcal{I}_{i_1,\ldots,i_r} = 0$ if $i_j \neq i_k$ for $j \neq k$, $j,k \in \mathcal{I}_r$.

For integers $\eta \geq 2$ and even orders $r \geq 2$, computing $\eta - 1$ tensor-tensor multiplications $\left(\boldsymbol{\mathcal{I}}_r - \boldsymbol{\mathcal{M}}^{(r)}\right)^\eta$ has the complexity $\mathcal{O}\left(d^{\frac{3}{2}r}\eta\right)$. For odd orders $r \geq 3$, due to alternations between multiplications in $\left\lfloor \frac{r}{2} \right\rfloor$ and $\left\lceil \frac{r}{2} \right\rceil$ modes, we have the complexity $\mathcal{O}\left(d^{\left\lfloor \frac{r}{2} \right\rfloor} d^{2\left\lceil \frac{r}{2} \right\rceil}\eta\right) \approx \mathcal{O}\left(d^{\frac{3}{2}r}\eta\right)$. Thus, the complexity of Eq. (13) w.r.t. integer $\eta \geq 2$ scales linearly. However, for even orders r, one can readily replace $\left(\boldsymbol{\mathcal{I}}_r - \boldsymbol{\mathcal{M}}^{(r)}\right)^\eta$ with exponentiation by squaring [2], whose cost is $\log(\eta)$. This readily yields the sublinear complexity $\mathcal{O}\left(d^{\frac{3}{2}r}\log(\eta)\right)$ w.r.t. η.

Algorithm 1 Tensor Shrinkage Operator with Exponentiation by Squaring, left part for even orders and right part for odd orders r.

```
Input: $\boldsymbol{\mathcal{M}}$ for a forward pass, $\eta \geq 1$, $r=2,4,...$
$\boldsymbol{\mathcal{M}}^*_1=\boldsymbol{\mathcal{I}}_r-\boldsymbol{\mathcal{M}}$, $n=\text{int}(\eta)$, $t=1$, $q=1$
while $n\neq 0$:
    if $n\&1$:
        if $t>1$: $\boldsymbol{\mathcal{G}}_{t+1}=\boldsymbol{\mathcal{G}}_t \times_{1,\dots,r/2} \boldsymbol{\mathcal{M}}^*_q$, else: $\boldsymbol{\mathcal{G}}_{t+1}=\boldsymbol{\mathcal{M}}^*_q$
        $n\leftarrow n-1$, $t\leftarrow t+1$
    $n\leftarrow \text{int}(n/2)$
    if $n>0$:
        $\boldsymbol{\mathcal{M}}^*_{q+1}=\boldsymbol{\mathcal{M}}^*_q \times_{1,\dots,r/2} \boldsymbol{\mathcal{M}}^*_q$
        $q\leftarrow q+1$
Output: $\widehat{\boldsymbol{\mathcal{G}}}_{\text{TSO}}(\boldsymbol{\mathcal{M}})=\boldsymbol{\mathcal{I}}_r-\boldsymbol{\mathcal{G}}_t$
```

```
Input: $\boldsymbol{\mathcal{M}}$ for a forward pass, $\eta=3^0,3^1,3^2,...,$ $r=3,5,...$
$\boldsymbol{\mathcal{M}}^*_1=\boldsymbol{\mathcal{I}}_r-\boldsymbol{\mathcal{M}}$, $n=\text{int}(\eta)$, $q=1$
while $n\neq 0$:
    $n\leftarrow \text{int}(n/3)$
    if $n>0$:
        $\boldsymbol{\mathcal{M}}^*_{q+1}=\boldsymbol{\mathcal{M}}^*_q \times_{1,\dots,\lfloor r/2\rfloor} \boldsymbol{\mathcal{M}}^*_q \times_{1,\dots,\lceil r/2\rceil} \boldsymbol{\mathcal{M}}^*_q$
        $q\leftarrow q+1$
Output: $\widehat{\boldsymbol{\mathcal{G}}}_{\text{TSO}}(\boldsymbol{\mathcal{M}})=\boldsymbol{\mathcal{I}}_r-\boldsymbol{\mathcal{M}}^*_q$
```

Algorithms 1 shows how TSO can be evaluated very efficiently for even and odd orders r, respectively. We restrict the latter variant to orders $r=3^0,3^1,3^2,...$ for brevity but derivations of a more complete recurrent formula for $r=3,5,7,...$ is straightforward. Finally, we note that matrix-matrix and tensor multiplications with cuBLAS are highly parallelizable so the $d^{\frac{3}{2}r}$ part of complexity can be reduced in theory even to $\log(d)$.

As we have $\boldsymbol{\mathcal{I}}_r-(\boldsymbol{\mathcal{I}}_r-\boldsymbol{\mathcal{M}})^\eta\rightarrow\boldsymbol{\mathcal{I}}_r$ as $\eta\rightarrow\infty$. For this reason, for sufficiently large η, the heat reverses to super-diagonals, which carry the majority of the signal as long as $1\ll\eta\ll\infty$. For this reason, we limit the number of coefficients of feature representations by extracting the super-diagonals from the TSO-processed $\boldsymbol{\mathcal{M}}^{(r)}$ as in Eq. 10, where r indicates the order of HoTD $\boldsymbol{\mathcal{M}}$. In our experiments, we use $r=2,3,4$.

As super-diagonals contain heat (information) obtained by diffusing the complements $1-\lambda_i$ in η steps in the spectral domain, which is simply realized by $\eta-1$ tensor products, the actual TSO is a form of aggregation along the tensor product mode(s). For this reason, We pass $\hat{\boldsymbol{\psi}}_r$ via the element-wise SigmE from Eq. (5), as in Eq. (11) to detect the presence of at least one feature being detected in $\hat{\boldsymbol{\psi}}_r$ after several aggregation steps. In what follows, we drop the r subscript from $\boldsymbol{\psi}_r$ for ease of notation. In section 5.2 we compare performance for different values of r.

TENET-RPN: Extracting query RoIs and their representations. To extract query RoIs we first generate a set $\boldsymbol{\mathcal{Z}}$ of TSO representations $\{\boldsymbol{\psi}\}_{z\in\boldsymbol{\mathcal{Z}}}$ from the support images, with $Z=|\boldsymbol{\mathcal{Z}}|$, and let $\boldsymbol{\Psi}\equiv\{\boldsymbol{\psi}\}_{z\in\boldsymbol{\mathcal{Z}}}$. We then perform cross-attention between $\boldsymbol{\Psi}\in\mathbb{R}^{d\times Z}$ and the N features $\boldsymbol{\Phi}^*\in\mathbb{R}^{d\times N}$ extracted from the query image. The attention input $\boldsymbol{Q}$ is then generated from $\boldsymbol{\Phi}^*$, while $\boldsymbol{K}$ and $\boldsymbol{V}$ are both generated from $\boldsymbol{\Psi}$. The output of the transformer block in Eq. 6 is fed into an RPN layer to output a set $\boldsymbol{\mathcal{B}}$ of $B=|\boldsymbol{\mathcal{B}}|$ query RoIs.

Each of these query RoIs is then represented using TSO representations to produce the set $\{\boldsymbol{\psi}^*\}_{b\in\boldsymbol{\mathcal{B}}}$. Both sets of representations, $\{\boldsymbol{\psi}^*\}_{b\in\boldsymbol{\mathcal{B}}}$ and $\{\boldsymbol{\psi}\}_{z\in\boldsymbol{\mathcal{Z}}}$, are passed to the next module, the transformer relation head, described next.

4.2 Transformer relation head

As mentioned previously, the goal of our transformer relation head , illustrated in Fig. 1, is to enhance features and query RoI embeddings from query and support images that are similar. It takes, as input, the set of TSO representations $\{\boldsymbol{\psi}_b^* \in \mathbb{R}^d\}_{b\in\boldsymbol{\mathcal{B}}}$ generated for each query RoI in $\boldsymbol{\mathcal{B}}$, and the set of TSO representations $\{\boldsymbol{\psi}_z \in \mathbb{R}^d\}_{z\in\boldsymbol{\mathcal{Z}}}$ for support images. TSO representations are derived from features extracted from layer 4 of ResNet-50, leading to a channel dimension of size $d = 1024$. TRH also takes as input query RoI features $\{\boldsymbol{\Phi}_b^* \in \mathbb{R}^{2d\times N}\}_{b\in\boldsymbol{\mathcal{B}}}$ and, lastly, support features $\{\boldsymbol{\Phi}_z \in \mathbb{R}^{2d\times N}\}_{z\in\boldsymbol{\mathcal{Z}}}$. These are both extracted from layer 5 of ResNet-50, leading to a channel dimension of size $2d$, *ie*. 2048.

They are then fed into 2 different transformers: (i) a **Z-shot** transformer head, which performs cross-attention between globally-pooled representations of the query images and support images; and (ii) a **Spatial-HOP** transformer head, which performs self-attention between the set of global representations and spatial feature vectors for a given image, be it a query or support image. We describe both in more detail next.

Z-shot transformer head. This transformer consists of a cross-attention layer formed with:

$$Q \equiv \{\mathbf{q}_1, ..., \mathbf{q}_B, |\mathbf{q}_b = \boldsymbol{W}_q \left[\bar{\boldsymbol{\phi}}_b \oplus \boldsymbol{W}_p\boldsymbol{\psi}_b^*\right]\}, \tag{14}$$

$$K \equiv \{\mathbf{k}_1, ..., \mathbf{k}_Z, |\mathbf{k}_z = \boldsymbol{W}_k \left[\bar{\boldsymbol{\phi}}_z \oplus \boldsymbol{W}_p\boldsymbol{\psi}_z\right]\}, \tag{15}$$

$$V \equiv \{\mathbf{v}_1, ..., \mathbf{v}_Z, |\mathbf{v}_z = \boldsymbol{W}_v \left[\bar{\boldsymbol{\phi}}_z \oplus \boldsymbol{W}_p\boldsymbol{\psi}_z\right]\}, \tag{16}$$

where $\oplus$ denotes element-wise addition, and $\bar{\boldsymbol{\phi}}$ denotes average-pooled features $(1/N)\boldsymbol{\Phi}^*\mathbf{1}$. The matrices $\boldsymbol{W}_q \in \mathbb{R}^{2d\times 2d}$, $\boldsymbol{W}_k \in \mathbb{R}^{2d\times 2d}$, $\boldsymbol{W}_v \in \mathbb{R}^{2d\times 2d}$, and $\boldsymbol{W}_p \in \mathbb{R}^{2d\times d}$ are learned weights. The $\boldsymbol{W}_p$ weights are shared by query RoI and support TSO representations and project such representations into a $2d$ space. In this way, each attention query vector $\mathbf{q}_b$ combines the extracted features for a query image RoI and the TSO representations for that RoI. Analogously, each key vector $\mathbf{k}_z$ and value vector $\mathbf{v}_z$ combines the extracted features and TSO representation for a support image. The layer performs cross-attention, enhancing representations of the RoIs and support images with similar information.

Spatial-HOP transformer head. This transformer consists of a layer that performs self-attention on sets of representations, both global and local, extracted either from Z support images, or B query RoIs. We describe its operation for a set $\boldsymbol{\mathcal{Z}}$ of support images. For the set $\boldsymbol{\mathcal{Z}}$ we compute $\boldsymbol{\Phi}_{\boldsymbol{\mathcal{Z}}}^{\dagger} \in \mathbb{R}^{2d\times N}$, where $\boldsymbol{\Phi}_{\boldsymbol{\mathcal{Z}}}^{\dagger} = (1/Z)\sum_{z\in\boldsymbol{\mathcal{Z}}}\boldsymbol{\Phi}_z$, and $\boldsymbol{\psi}_{\boldsymbol{\mathcal{Z}}}^{\dagger} \in \mathbb{R}^d$, where $\boldsymbol{\psi}_{\boldsymbol{\mathcal{Z}}}^{\dagger} = (1/Z)\sum_{z\in\boldsymbol{\mathcal{Z}}}\boldsymbol{\psi}_z$. We split $\boldsymbol{\Phi}_{\boldsymbol{\mathcal{Z}}}^{\dagger}$ along the channel dimension of size $2d$ to create two new matrices $\boldsymbol{\Phi}_{\boldsymbol{\mathcal{Z}}}^{\dagger u} \in \mathbb{R}^{d\times N}$ and $\boldsymbol{\Phi}_{\boldsymbol{\mathcal{Z}}}^{\dagger l} \in \mathbb{R}^{d\times N}$. We let $\boldsymbol{\Phi}_{\boldsymbol{\mathcal{Z}}}^{\dagger l} \equiv \{\boldsymbol{\phi}_{\boldsymbol{\mathcal{Z}},1}^{\dagger l}, ..., \boldsymbol{\phi}_{\boldsymbol{\mathcal{Z}},N}^{\dagger l} \in \mathbb{R}^d\}$. Self-attention is then performed over the following set $\boldsymbol{\mathcal{T}}_{\boldsymbol{\mathcal{Z}}}$ of token vectors:

$$\boldsymbol{\mathcal{T}}_{\boldsymbol{\mathcal{Z}}} \equiv \{\boldsymbol{\phi}_{\boldsymbol{\mathcal{Z}},1}^{\dagger l}, ..., \boldsymbol{\phi}_{\boldsymbol{\mathcal{Z}},N}^{\dagger l}; \bar{\boldsymbol{\phi}}_{\boldsymbol{\mathcal{Z}}}^{\dagger u}; \boldsymbol{W}_g\boldsymbol{\psi}_{\boldsymbol{\mathcal{Z}}}^{\dagger}\}. \tag{17}$$

Table 1: Comparison of different methods in terms of mAP (%) on three splits of the VOC 2007 testing set.

Method/Shot		Split 1				Split 2				Split 3				Mean±std			
		1	3	5	10	1	3	5	10	1	3	5	10	1	3	5	10
FRCN	ICCV12	11.9	29.0	36.9	36.9	5.9	23.4	29.1	28.8	5.0	18.1	30.8	43.4	7.6±3.1	23.5±4.5	32.3±3.3	36.4±6.0
FR	ICCV19	14.8	26.7	33.9	47.2	15.7	22.7	30.1	39.2	19.2	25.7	40.6	41.3	16.6±1.9	25.0±1.7	34.9±4.3	42.6±3.4
Meta	ICCV19	19.9	35.0	45.7	51.5	10.4	29.6	34.8	45.4	14.3	27.5	41.2	48.1	14.9±3.9	30.7±3.2	40.6±4.5	48.3±2.5
FSOD	CVPR20	29.8	36.3	48.4	53.6	22.2	25.2	31.2	39.7	24.3	34.4	47.1	50.4	25.4±3.2	32.0±4.8	42.2±4.2	47.9±3.9
NP-RepMet	NeurIPS20	37.8	41.7	47.3	49.4	**41.6**	43.4	47.4	49.1	33.3	39.8	41.5	44.8	37.6±3.4	41.6±1.5	45.4±2.8	47.8±2.1
PNSD	ACCV20	32.4	39.6	50.2	55.1	30.2	30.3	36.4	42.3	30.8	38.6	46.9	52.4	31.3±4.4	36.2±4.2	44.5±3.8	49.9±5.4
MPSR	ECCV20	41.7	51.4	55.2	61.8	24.4	39.2	39.9	47.8	**35.6**	42.3	48.0	49.7	33.9±7.2	44.3±5.2	47.7±6.2	53.1±6.2
TFA	ICML20	39.8	44.7	55.7	56.0	23.5	34.1	35.1	39.1	30.8	42.8	49.5	49.8	31.4±6.7	40.5±4.6	46.8±8.6	48.3±7.0
FSCE	CVPR21	44.2	51.4	61.9	63.4	27.3	43.5	44.2	50.2	22.6	39.5	47.3	54.0	31.4±9.3	44.8±4.9	51.1±7.7	55.9±5.6
CGDP+FRCN	CVPR21	40.7	46.5	57.4	62.4	27.3	40.8	42.7	46.3	31.2	43.7	50.1	55.6	33.1±5.6	43.67±2.3	50.0±6.0	54.8±6.6
TIP	CVPR21	27.7	43.3	50.2	56.6	22.7	33.8	40.9	46.9	21.7	38.1	44.5	50.9	24.0±2.6	38.4±4.0	45.2±4.3	52.47±5.3
FSOD$^{\mathrm{up}}$	ICCV21	43.8	50.3	55.4	61.7	31.2	41.2	44.2	48.3	35.5	43.9	50.6	53.5	36.8±5.2	45.1±3.8	50.1±4.6	54.5±5.5
QSAM	WACV22	31.1	39.2	50.7	59.4	22.9	32.1	35.4	42.7	24.3	35.0	50.0	53.6	26.1±3.5	35.4±2.9	45.4±3.6	51.9±3.8
TENET	(Ours)	**46.7**	**55.4**	**62.3**	**66.9**	40.3	**44.7**	**49.3**	**52.1**	35.5	**46.0**	**54.4**	**54.6**	**40.8±3.6**	**48.7±4.7**	**55.3±3.1**	**57.9±5.8**

where $\boldsymbol{W}_g \in \mathbb{R}^{d\times d}$ denotes learned weights that are shared between support and query representations. The transformed tokens encode relations among global and local representations that have been pooled across all Z shots. We compute similar transformed tokens using analogous representations for each query RoI.

The outputs of the Z-shot and Spatial-HOP transformer heads are aggregated to form representations for each query RoI, which are then fed into a classifier and bounding-box regressor (see §C of **Suppl. Material** for details).

5 Experiments

Datasets and Settings. For PASCAL VOC 2007/12 [6], we adopt the 15/5 base/novel category split setting and use training/validation sets from PASCAL VOC 2007 and 2012 for training, and the testing set from PASCAL VOC 2007 for testing, following [11,7,51,23]. For MS COCO [28], we follow [48], and adopt the 20 categories that overlap with PASCAL VOC as the novel categories for testing, whereas the remaining 60 categories are used for training. For the FSOD dataset [7], we split its 1000 categories into 800/200 for training/testing.

Implementation Details. TENET uses ResNet-50 pre-trained on ImageNet [5] and MS COCO [28]. We fine-tune the network with a learning rate of 0.002 for the first 56000 iterations and 0.0002 for another 4000 iterations. Images are resized to 600 pixels (shorter edge) and the longer edge is capped at 1000 pixels. Each support image is cropped based on ground-truth boxes, bilinearly interpolated and padded to 320×320 pixels. We set, via cross-validation) SigmE parameter $\eta'=200$ and TSO parameters $\eta_2 = \eta_3 = \eta_4 = 7$. We report standard metrics for FSOD, namely mAP, AP, AP_{50} and AP_{75}.

5.1 Comparisons with the State of the Art

PASCAL VOC 2007/12. We compare our method to QSAM[23], FSOD$^{\mathrm{up}}$ [45], CGDP+FRCN [26], TIP [24], FSCE [37], TFA [42], Feature Reweighting (FR) [11], LSTD [4], FRCN [34], NP-RepMet [49], MPSR[46], PSND [51] and FSOD [7]. Table 1 shows that our TENET method outperforms FSOD by a 7.1–15.4% margin. For the 1- and 10-shot regime, we outperform QSAM [23] by ∼14.7%.

MS COCO. Table 2a compares TENET with QSAM[23], FSOD$^{\text{up}}$ [45], CGDP+ FRCN [26], TIP [24], FSCE [37], TFA [42], FR [11], Meta R-CNN [48], FSOD [7] and PNSD[51] on the MS COCO minival set (20 novel categories, 10-shot protocol). Although MS COCO is more challenging in terms of complexity and the dataset size, TENET boosts performance to 19.1%, 27.4% and 19.6%, surpassing the current SOTA method QSAM by a large margin, 6.1%, 2.7% and 7.5% on AP, AP_{50} and AP_{75}.

FSOD. In Table 2b we compare TENET (5-shot protocol) with PNSD [51], FSOD [7], LSTD [4] and LSTD (FRN [34]). We re-implement BD&TK, modules of LSTD, based on Faster-RCNN for a fair comparison. TENET gives SOTA results of 35.4% AP_{50} and 31.6% AP_{75}.

Table 2: Comparison with SOTA on the MS COCO minival set (2a) and FSOD testset (2b).

Shot	Method		AP	AP_{50}	AP_{75}
10	LSTD	AAAI18	3.2	8.1	2.1
	FR	ICCV12	5.6	12.3	4.6
	Meta	ICCV19	8.7	19.18	6.6
	MPSR	ECCV20	9.8	17.9	9.7
	FSOD	CVPR20	11.1	20.4	10.6
	PNSD	ACCV20	12.3	21.7	11.7
	TFA	ICML20	9.6	10.0	9.3
	FSCE	CVPR21	10.7	11.9	10.5
	CGDP+FRCN	CVPR21	11.3	20.3	11.5
	FSOD$^{\text{up}}$	ICCV21	11.6	23.9	9.8
	QSAM	WACV22	13.0	24.7	12.1
	TENET	(Ours)	**19.1**	**27.4**	**19.6**

(a)

Shot	Method		AP_{50}	AP_{75}
5	LSTD (FRN)	AAAI18	23.0	12.9
	LSTD	AAAI18	24.2	13.5
	FSOD	CVPR20	27.5	19.4
	PNSD	ACCV20	29.8	22.6
	QSAM	WACV22	30.7	25.9
	TENET	(Ours)	**35.4**	**31.6**

(b)

5.2 Hyper-parameter and ablation analysis

TENET. Table 3a shows that among orders $r=2$, $r=3$ and $r=4$, the second-order variant is, unsurprisingly, the most informative one. We next consider pairs of orders, and the triplet $r=2,3,4$. As the number of tensor coefficients grows quickly w.r.t. r, we split the 1024 channels into groups *e.g.*, $r= 2,3$. A 3:1 split means that second- and third-order tensors are built from 768 and 256 channels $(768+256=1024)$. We report only the best splits. For pairs of orders, variant $r=2,3$ outperforms other combinations. Triplet $r=2,3,4$, the best performer, outperforms $r = 2$ by 5.8% and 2.7% in novel classes (5- and 10-shot), and 2.1% and 2.4% in base classes. As all representations are 1024-dimensional, we conclude that multi-order variants are the most informative.

TSO. Based on the best channel-wise splits in Table 3a, we study the impact of η_r (shrinkage/decorrelation) of TSO to verify its effectiveness. Figure 2a shows mAP w.r.t. the individual η_2, η_3 and η_4 for $r = 2$, $r = 3$ and $r = 4$. We then investigate the impact of η_r on pairwise representations, where we set the same η_r for pairwise variants, *e.g.*, $\eta_2 = \eta_3$. Again, the same value of η_r is used for triplet $r=2,3,4$. Note that for $\eta_r=1$, TSO is switched off and all representations reduce to the polynomial feature maps in So-HoT [14]. As shown in Figure 2a, TSO is very beneficial ($\sim$ 5% gain for triplet $r= 2,3,4$ over not using TSO).

TRH. Our detector is designed to have a heightened discriminative ability to distinguish between different classes. Its Z-Shot and Spatial-HOP Transformer

Table 3: Results on VOC2007 testset for applying TENET in RPN or TRH (3a, top panel of 3b). TRH ablation shown in bottom panel of 3b.

r			dim. split	Shot(Novel)		Shot(Base)		Speed
2	3	4		5	10	5	10	(img/ms)
✓				56.5	64.2	71.7	75.5	32
	✓			55.7	63.2	67.0	72.1	69
		✓		51.4	58.9	68.7	74.8	78
✓	✓		**3:1**	58.3	63.2	69.3	75.1	42
✓		✓	3:1	56.1	62.4	70.8	75.4	68
	✓	✓	2:2	51.8	61.7	68.1	73.6	71
✓	✓	✓	6:1:1	53.6	62.7	69.4	72.8	
			5:2:1	**62.3**	**66.9**	**73.8**	**77.9**	59
			5:1:2	53.9	63.1	69.7	73.3	
			4:2:2	61.4	65.0	70.4	74.9	
			4:3:1	59.1	63.6	71.8	75.2	
			4:1:3	61.0	64.1	68.9	72.5	

(a)

	RPN	TRH	Shot(Novel)		Shot(Base)	
	r		5	10	5	10
a	1	1	53.4	61.8	64.9	72.1
b	2,3,4	1	57.2	63.7	68.8	76.6
c	2,3,4	2,3,4	61.0	65.4	71.3	77.3
d	2,3,4	1,2,3,4	**62.3**	**66.9**	**73.8**	**78.2**

TRH		Shot(Novel)		Shot(Base)	
Z-shot	Spatial-HOP	5	10	5	10
✓		58.5	63.2	69.3	75.1
	✓	61.0	65.8	71.7	76.5
✓	✓	**62.3**	**66.9**	**73.8**	**78.2**

(b)

Table 4: Effect of varying (a) group within MHA in Tab. 4a and (b) TENET block in Tab. 4b on PASCAL VOC 2007 (5/10-shot, novel classes). When varying TENET block, group number is fixed to 4 (best value).

TA		1	2	4	8	16	32	64
Shot (Novel)	5	58.3	59.1	**61.8**	60.5	58.4	58.5	56.0
	10	61.2	62.8	**65.8**	64.2	61.2	61.7	60.4

(a)

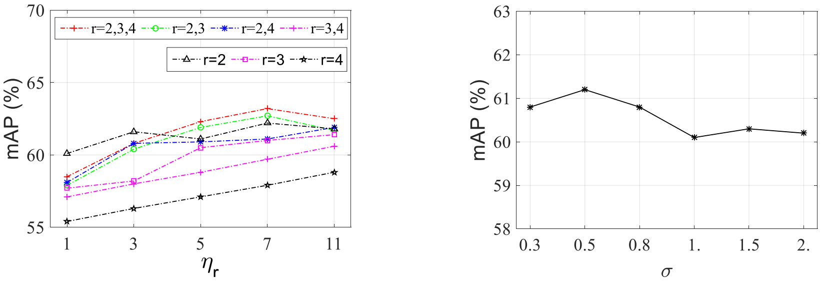
TB		1	2	3	4	5
Shot (Novel)	5	61.8	**62.3**	62.1	61.8	61.9
	10	65.8	**66.9**	66.4	66.1	66.4

(b)

Relation Heads encode similarities between support and query images. We quantify the empirical impact of each in Table 3b (bottom), which shows that both heads complement each other to produce higher performance.

Other hyperparameters. We first examine the influence of Gaussian kernel parameter σ. We vary σ from 0.3 to 3 and show in Fig 2b that $\sigma = 0.5$ gives the best performance. We now fix σ and investigate the impact of varying the number of heads used in T-Heads Attention (TA). Table 4a shows best performance with $TA = 4$. Lastly, we vary the number of TENET blocks (TB) and show in Table 4b that our model performance is stable across different choices, particularly for $TB \geq 2$. Unless otherwise noted, $TA = 2$ and $TB = 4$, respectively, on VOC dataset. In our suppl. material, we present more results and discussions on FSOD and MS COCO dataset.

Impact of TENET on RPN and TRH. Table 3b shows ablations w.r.t. TENET variants: 1) either in RPN or in TRH, or 2) in both RPN and TRH. Comparing results for settings a,b, and c confirms that using second-, third- and fourth-orders simultaneously is beneficial for both RPN and TRH, achieving 3.8%/1.9% as well as 3.8%/1.8% improvement on novel classes over the first-order-only variant. In addition, results for settings c and d show that TRH can better encode the information carried within support regions by leveraging both first-order and higher-order representations.



(a) (b)

Fig. 2: mAP (VOC2007 dataset, novel classes, 10-shot) w.r.t. varying η_r in TSO (Fig. 2a) and the σ of Gaussian Kernel in self-attention function (Fig. 2b) .

6 Conclusion

We have proposed TENET, which uses higher-order tensor descriptors, in combination with a novel tensor shrinkage operator, to generate highly-discriminative representations with tractable dimensionality. We use these representations in our proposed transformer relation heads to dynamically extract correlations between query image regions and the entire support set for a class. TENET has heightened robustness to large intra-class variations, leading to SOTA performance on popular benchmarks.

References

1. Exponentiation by squaring. Wikipedia, https://en.wikipedia.org/wiki/Tsallis_entropy, accessed: 12-03-2021 3
2. Exponentiation by squaring. Wikipedia, https://en.wikipedia.org/wiki/Exponentiation_by_squaring, accessed: 12-03-2021 8
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV 2020. Lecture Notes in Computer Science, vol. 12346, pp. 213–229. Springer (2020) 4
4. Chen, H., Wang, Y., Wang, G., Qiao, Y.: LSTD: A low-shot transfer detector for object detection. In: McIlraith, S.A., Weinberger, K.Q. (eds.) Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018. pp. 2836–2843. AAAI Press (2018), https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16778 1, 3, 11, 12
5. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Li, F.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA. pp. 248–255. IEEE Computer Society (2009). https://doi.org/10.1109/CVPR.2009.5206848, https://doi.org/10.1109/CVPR.2009.5206848 11, 22
6. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J.M., Zisserman, A.: The pascal visual object classes (VOC) challenge. Int. J. Comput. Vis. **88**(2), 303–338 (2010). https://doi.org/10.1007/s11263-009-0275-4, https://doi.org/10.1007/s11263-009-0275-4 11
7. Fan, Q., Zhuo, W., Tai, Y.: Few-shot object detection with attention-rpn and multi-relation detector. CoRR **abs/1908.01998** (2019), http://arxiv.org/abs/1908.01998 1, 2, 3, 11, 12
8. Girshick, R.B.: Fast R-CNN. In: 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015. pp. 1440–1448. IEEE Computer Society (2015). https://doi.org/10.1109/ICCV.2015.169, https://doi.org/10.1109/ICCV.2015.169 1
9. Hu, H., Zhang, Z., Xie, Z., Lin, S.: Local relation networks for image recognition. In: ICCV 2019. pp. 3463–3472. IEEE (2019). https://doi.org/10.1109/ICCV.2019.00356, https://doi.org/10.1109/ICCV.2019.00356 2, 4
10. Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E.: Squeeze-and-excitation networks. IEEE Trans. Pattern Anal. Mach. Intell. **42**(8), 2011–2023 (2020). https://doi.org/10.1109/TPAMI.2019.2913372, https://doi.org/10.1109/TPAMI.2019.2913372 4
11. Kang, B., Liu, Z., Wang, X., Yu, F., Feng, J., Darrell, T.: Few-shot object detection via feature reweighting. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019. pp. 8419–8428. IEEE (2019). https://doi.org/10.1109/ICCV.2019.00851, https://doi.org/10.1109/ICCV.2019.00851 1, 11, 12
12. Karlinsky, L., Shtok, J., Harary, S., Schwartz, E., Aides, A., Feris, R.S., Giryes, R., Bronstein, A.M.: Repmet: Representative-based metric learning for

classification and few-shot object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. pp. 5197–5206. Computer Vision Foundation / IEEE (2019). https://doi.org/10.1109/CVPR.2019.00534 1

13. Kong, T., Yao, A., Chen, Y., Sun, F.: Hypernet: Towards accurate region proposal generation and joint object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 845–853. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.98, https://doi.org/10.1109/CVPR.2016.98 1
14. Koniusz, P., Tas, Y., Porikli, F.: Domain adaptation by mixture of alignments of second- or higher-order scatter tensors. CoRR **abs/1409.1556** (2016) 4, 5, 12
15. Koniusz, P., Wang, L., Cherian, A.: Tensor representations for action recognition. TPAMI (2020) 2, 5
16. Koniusz, P., Yan, F., Gosselin, P.H., Mikolajczyk, K.: Higher-order occurrence pooling on mid-and low-level features: Visual concept detection. Tech. Report (2013) 4
17. Koniusz, P., Yan, F., Gosselin, P., Mikolajczyk, K.: Higher-order occurrence pooling for bags-of-words: Visual concept detection. IEEE Trans. Pattern Anal. Mach. Intell. **39**(2), 313–326 (2017). https://doi.org/10.1109/TPAMI.2016.2545667, https://doi.org/10.1109/TPAMI.2016.2545667 2, 4
18. Koniusz, P., Zhang, H.: Power normalizations in fine-grained image, few-shot image and graph classification. TPAMI (2020) 2, 3, 5, 6
19. Koniusz, P., Zhang, H., Porikli, F.: A deeper look at power normalizations. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 5774–5783. IEEE Computer Society (2018). https://doi.org/10.1109/CVPR.2018.00605 4, 5
20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Commun. ACM **60**(6), 84–90 (2017). https://doi.org/10.1145/3065386, http://doi.acm.org/10.1145/3065386 4
21. Lathauwer, L.D., Moor, B.D., Vandewalle, J.: A multilinear singular value decomposition. SIAM J. Matrix Analysis and Applications **21**, 1253–1278 (2000) 5
22. Ledoit, O., Wolf, M.: Honey, i shrunk the sample covariance matrix. The Journal of Portfolio Management **30**(4), 110–119 (2004). https://doi.org/10.3905/jpm.2004.110, https://jpm.pm-research.com/content/30/4/110 2, 7
23. Lee, H., Lee, M., Kwak, N.: Few-shot object detection by attending to per-sample-prototype. In: WACV, 2022, Waikoloa, HI, USA, January 3-8, 2022. pp. 1101–1110. IEEE (2022). https://doi.org/10.1109/WACV51458.2022.00117, https://doi.org/10.1109/WACV51458.2022.00117 2, 4, 11, 12
24. Li, A., Li, Z.: Transformation invariant few-shot object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3094–3102 (2021) 11, 12
25. Li, X., Wang, W., Hu, X., Yang, J.: Selective kernel networks. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. pp. 510–519. Computer Vision Foundation / IEEE (2019). https://doi.org/10.1109/CVPR.2019.00060 4
26. Li, Y., Zhu, H., Cheng, Y., Wang, W., Teo, C.S., Xiang, C., Vadakkepat, P., Lee, T.H.: Few-shot object detection via classification refinement and distractor retreatment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15395–15403 (2021) 11, 12

27. Lin, T., Dollár, P., Girshick, R.B., He, K., Hariharan, B., Belongie, S.J.: Feature pyramid networks for object detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 936–944. IEEE Computer Society (2017). https://doi.org/10.1109/CVPR.2017.106, https://doi.org/10.1109/CVPR.2017.106 1
28. Lin, T., Maire, M., Belongie, S.J., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. In: Fleet, D.J., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V. Lecture Notes in Computer Science, vol. 8693, pp. 740–755. Springer (2014). https://doi.org/10.1007/978-3-319-10602-1_48, https://doi.org/10.1007/978-3-319-10602-1_48 11
29. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. CoRR **abs/2103.14030** (2021), https://arxiv.org/abs/2103.14030 22
30. Lu, J., Yao, J., Zhang, J., Zhu, X., Xu, H., Gao, W., Xu, C., Xiang, T., Zhang, L.: SOFT: softmax-free transformer with linear complexity. CoRR **abs/2110.11945** (2021), https://arxiv.org/abs/2110.11945 4, 6
31. Rahman, S., Wang, L., Sun, C., Zhou, L.: Redro: Efficiently learning large-sized spd visual representation. In: European Conference on Computer Vision (2020) 4
32. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 6517–6525. IEEE Computer Society (2017). https://doi.org/10.1109/CVPR.2017.690, https://doi.org/10.1109/CVPR.2017.690 1
33. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. CoRR **abs/1804.02767** (2018), http://arxiv.org/abs/1804.02767 1
34. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada. pp. 91–99 (2015) 1, 2, 11, 12
35. Shih, Y., Yeh, Y., Lin, Y., Weng, M., Lu, Y., Chuang, Y.: Deep co-occurrence feature learning for visual object recognition. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 7302–7311. IEEE Computer Society (2017). https://doi.org/10.1109/CVPR.2017.772, https://doi.org/10.1109/CVPR.2017.772 4
36. Smola, A.J., Kondor, R.: Kernels and regularization on graphs. Learning Theory and Kernel Machines pp. 144–158 (2003) 2, 5
37. Sun, B., Li, B., Cai, S., Yuan, Y., Zhang, C.: FSCE: few-shot object detection via contrastive proposal encoding. CoRR **abs/2103.05950** (2021), https://arxiv.org/abs/2103.05950 11, 12
38. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015. pp. 1–9. IEEE Computer Society (2015). https://doi.org/10.1109/CVPR.2015.7298594, https://doi.org/10.1109/CVPR.2015.7298594 4

39. y Terán, A.R.M., Gouiffès, M., Lacassagne, L.: Enhanced local binary covariance matrices (ELBCM) for texture analysis and object tracking. In: Eisert, P., Gagalowicz, A. (eds.) 6th International Conference on Computer Vision / Computer Graphics Collaboration Techniques and Applications, MIRAGE '13, Berlin, Germany - June 06 - 07, 2013. pp. 10:1–10:8. ACM (2013). https://doi.org/10.1145/2466715.2466733, https://doi.org/10.1145/2466715.2466733 4
40. Tuzel, O., Porikli, F., Meer, P.: Region covariance: A fast descriptor for detection and classification. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) Computer Vision - ECCV 2006, 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006, Proceedings, Part II. Lecture Notes in Computer Science, vol. 3952, pp. 589–600. Springer (2006). https://doi.org/10.1007/11744047_45, https://doi.org/10.1007/11744047_45 4
41. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems 30,2017. pp. 5998–6008 (2017), https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html 2, 4, 6
42. Wang, X., Huang, T.E., Gonzalez, J., Darrell, T., Yu, F.: Frustratingly simple few-shot object detection. In: ICML 2020. Proceedings of Machine Learning Research, vol. 119, pp. 9919–9928. PMLR (2020), http://proceedings.mlr.press/v119/wang20j.html 11, 12
43. West, J., Venture, D., Warnick, S.: Spring research presentation: A theoretical foundation for inductive transfer. Brigham Young University, College of Physical and Mathematical Sciences (2007) 1
44. Woodworth, R.S., Thorndike, E.L.: The influence of improvement in one mental function upon the efficiency of other functions. Psychological Review (I) **8**(3), 247–261 (1901). https://doi.org/10.1037/h0074898 1
45. Wu, A., Han, Y., Zhu, L., Yang, Y.: Universal-prototype enhancing for few-shot object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9567–9576 (October 2021) 11, 12
46. Wu, J., Liu, S., Huang, D., Wang, Y.: Multi-scale positive sample refinement for few-shot object detection. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J. (eds.) Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XVI. Lecture Notes in Computer Science, vol. 12361, pp. 456–472. Springer (2020) 3, 11
47. Xie, S., Girshick, R.B., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 5987–5995. IEEE Computer Society (2017). https://doi.org/10.1109/CVPR.2017.634, https://doi.org/10.1109/CVPR.2017.634 4
48. Yan, X., Chen, Z., Xu, A., Wang, X., Liang, X., Lin, L.: Meta R-CNN: towards general solver for instance-level low-shot learning. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019. pp. 9576–9585. IEEE (2019). https://doi.org/10.1109/ICCV.2019.00967, https://doi.org/10.1109/ICCV.2019.00967 1, 2, 3, 11, 12
49. Yang, Y., Wei, F., Shi, M., Li, G.: Restoring negative information in few-shot object detection. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems 33: Annual Confer-

ence on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual (2020), https://proceedings.neurips.cc/paper/2020/hash/240ac9371ec2671ae99847c3ae2e6384-Abstract.html 3, 11

50. Zhang, H., Koniusz, P.: Power normalizing second-order similarity network for few-shot learning. In: IEEE Winter Conference on Applications of Computer Vision, WACV 2019, Waikoloa Village, HI, USA, January 7-11, 2019. pp. 1185–1193. IEEE (2019). https://doi.org/10.1109/WACV.2019.00131, https://doi.org/10.1109/WACV.2019.00131 4
51. Zhang, S., Luo, D., Wang, L., Koniusz, P.: Few-shot object detection by second-order pooling. In: Proceedings of the Asian Conference on Computer Vision (2020) 1, 2, 3, 4, 11, 12
52. Zhang, S., Wang, L., Murray, N., Koniusz, P.: Kernelized few-shot object detection with efficient integral aggregation. In: IEEE Conference on Computer Vision and Pattern Recognition (2022) 1, 2
53. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable DETR: deformable transformers for end-to-end object detection. In: ICLR 2021. OpenReview.net (2021), https://openreview.net/forum?id=gZ9hCDWe6ke 4

Time-rEversed diffusioN tEnsor Transformer: A new TENET of Few-Shot Object Detection (Supplementary Material)

Shan Zhang[⋆,†], Naila Murray[♣], Lei Wang[♦], and Piotr Koniusz[⋆,§,†]

[†]Australian National University [♣]Meta AI
[♦]University of Wollongong [§]Data61/CSIRO
[†]firstname.lastname@anu.edu.au, [♦]leiw@uow.edu.au, [♣]murrayn@fb.com

Below are additional derivations, evaluations and illustrations of our method.

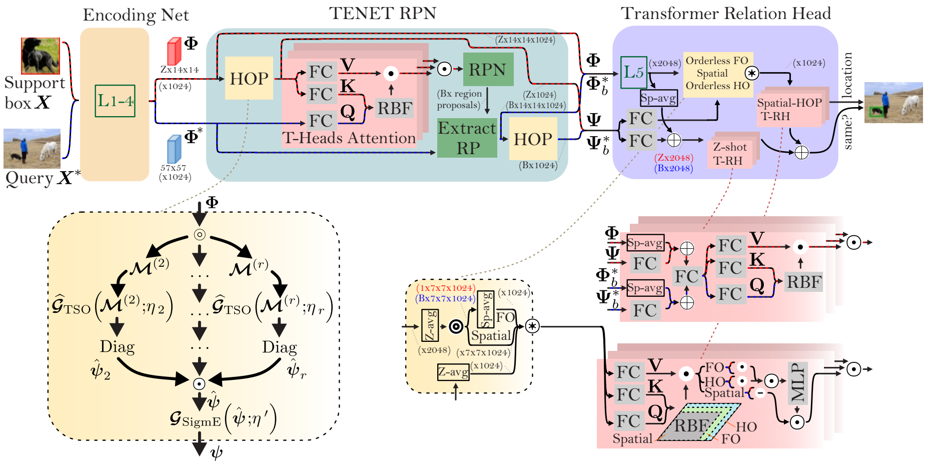


Fig. 3: Detailed illustration of our pipeline. We follow the architecture from Fig. 1 and expand blocks such as HOP, 'Orderless FO, Spatial, Orderless HO', Z-shot T-RH and Spatial-HOP T-RH. See the text for detailed descriptions.

In Figure 3, we have:

i. The HOP unit uses operator $\circledcirc$ to split the channel mode into groups ($e.g.$, 2:1:1 split means two parts of the channel dimension are used to form $\boldsymbol{\mathcal{M}}^{(2)}$, one part to form $\boldsymbol{\mathcal{M}}^{(3)}$, and one part to form $\boldsymbol{\mathcal{M}}^{(4)}$). Subsequently, TSO with parameters $\eta_2 = ... = \eta_r = \eta$ are applied for orders $r = 2, ..., r$, and diagonal entries are extracted from each tensor and concatenated by $\odot$. Finally, element-wise SigmE pooling with parameter η' is applied.

[⋆]SZ was mainly in charge of the pipeline/developing the transformer. PK (corresponding author) was mainly in charge of mathematical design of TENET & TSO. Code: `https://github.com/ZS123-lang/TENET`.

ii. The ‘Orderless FO, Spatial, Orderless HO’ block combines the first-order (FO), spatial and high-order (HO) representations. Operator ‘Z-avg’ performs average pooling along Z-way mode, operator ‘Sp-avg’ performs average pooling along the spatial modes of feature maps, operator $\circledcirc$ simply splits the channel mode into two equally sized groups (each is half of the channel dimension), and operator $\circledast$ performs concatenation of FO, spatial and HO representations along the spatial mode of feature maps *e.g.*, we obtain $N+2$ fibers times 1024 channels.

iii. The Z-shot T-RH is a transformer which performs attention on individual Z-shots. The spatial representation $\boldsymbol{\Phi}$ is average pooled along spatial dimensions by ‘Sp-avg’ and combined with high-order $\boldsymbol{\Psi}$ (passed by a FC layer) via addition $\oplus$. Another FC layer follows and subsequently the value, key and query matrices are computed, with an RBF attention between the key and query, and operator $\bullet$ multiplying the value matrix with the RBF attention matrix. The whole head may be repeated T times and outputs concatenated by $\odot$ to form the output of this transformer block.

iv. The Spatial-HOP T-RH simply takes the inputs from the ‘Orderless FO, Spatial, Orderless HO’ block, and computes the value, the key and the query matrices per support and query RoIs. The attention obtained with the RBF kernel has $(N+2)\times(N+2)$ size. Thus, the attention kernel is composed of spatial attention, FO-spatial and HO-spatial attention. After multiplying the kernel with the value matrix, we extract the spatial, FO and HO representations, respectively. Support and query first-order representations (FO) are element-wisely multiplied by $\bullet$ (we call it the multiplicative relationship operator). Support and query high-order representations (HO) also use the multiplicative relationship operator. Finally, support and query spatial representations use the subtraction operator $\ominus$. After the concatenation of FO and HO relational representations by $\odot$, passing via an MLP (FC+ReLU+FC), and concatenation with the spatial relational representations, we have an output of the single attention block, which can be repeated T times.

A TSO acts as the shrinkage estimator with the target of identity matrix.

Theorem 1. *Let λ be the ell$_1$-norm normalized spectrum, as in Eq. (3). Then, $g(\lambda;\eta)=1-(1-\lambda)^\eta$ is in fact a shrinkage operator, a solution to the problem in Eq. (12) with $\delta=1$, and the Kullback-Leibler divergence and the Tsallis entropy substituting the distance/divergence[1] d and the regularization term $\varOmega$, respectively.*

Proof. Let $d(\boldsymbol{\lambda},\boldsymbol{\lambda}')=D_{\mathrm{KL}}\boldsymbol{\lambda}^\circ\boldsymbol{\lambda}^{\circ\prime}=-\big(\sum\limits_{i\in\mathcal{I}_d}\lambda_i^\circ\log\lambda_i^{\circ\prime}\big)+\big(\sum\limits_{i\in\mathcal{I}_d}\lambda_i^\circ\log\lambda_i^\circ\big)$, where $\boldsymbol{\lambda}^\circ=1–\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}^{\circ\prime}=1–\boldsymbol{\lambda}'$ are complements of $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$. Let $\varOmega(\boldsymbol{\lambda}';\alpha)=\frac{1}{\alpha}(1–\sum\limits_{i\in\mathcal{I}_d}\boldsymbol{\lambda}^{\circ\prime\alpha})=$

[1] The KL divergence is not a distance, as it is not a metric measure due to its non-symmetric nature.

$\frac{1}{\alpha}(1-\sum\limits_{i\in\mathcal{I}_d}(1-\boldsymbol{\lambda}')^{\alpha})$. We define $f(\boldsymbol{\lambda},\boldsymbol{\lambda}')=-(\sum\limits_{i\in\mathcal{I}_d}\lambda_i^{\circ}\log\lambda_i^{\circ\prime})+(\sum\limits_{i\in\mathcal{I}_d}\lambda_i^{\circ}\log\lambda_i^{\circ})+\frac{1}{\alpha}(1-\sum\limits_{i\in\mathcal{I}_d}\boldsymbol{\lambda}^{\circ\prime\alpha})$ which we minimize w.r.t. $\boldsymbol{\lambda}'$ by computing $\frac{\partial f}{\partial\lambda_i'}=0$, that is,

$$\frac{\partial f}{\partial\lambda_i'}=\frac{1-\lambda_i}{1-\lambda_i'}-(1-\lambda_i')^{\alpha-1}=0\Leftrightarrow\lambda_i'=1-(1-\lambda_i)^{\frac{1}{\alpha}}. \tag{18}$$

Let $\frac{1}{\alpha}=\eta$ in Eq. (18), which completes the proof.

Moreover, from the above proof, it is easy to conclude that TSO is not a mere naive linear interpolation between λ and the target 1.

Theorem 2. *The highly structured operator* $\boldsymbol{F}$*, that is, the target of the shrinkage operator is the identity matrix.*

Proof. This is easily seen because $\lim\limits_{\eta\rightarrow\infty}1-(1-\lambda_i)^{\eta}=1$ if $\boldsymbol{\lambda}\neq 0$ is the ℓ_1-norm normalized spectrum from SVD *e.g.*, $\boldsymbol{U}\boldsymbol{\lambda}\boldsymbol{U}^T=\boldsymbol{M}\succcurlyeq 0$, that is, $\lambda_i:=\lambda_i/(\sum_{i'}\lambda_{i'}+\varepsilon)$, $\varepsilon>0$, and thus we have $\boldsymbol{U}\boldsymbol{U}^T=\mathbb{I}$.

B Ablation Study on Encoding Network

Below we perform ablations of the backbone (Encoding Network, termed as EN in main paper). We use ConvNet (ResNet-50) and Transformer network [29] (Swin-B^7/ Swin-B^{12} pre-trained on ImageNet-22K [5] with window size of 7/12), as shown in Table 5c. The comparisons are conducted by changing the backbone, whereas other settings remain unchanged. When ResNet-50 is replaced by Swin-B^7, we gain an improvement of 0.3% and 0.5% in the 5/10-shot setting (novel classes).

C Details of Transformer Relation Head (TRH) with Z-shot and Spatial-HOP blocks.

As Z-shot T-RH is described in Eq. (14)–(16) of the main paper, below we focus on describing Spatial-HOP T-RH.

This head first forms a so-called self-attention on a set $\boldsymbol{\mathcal{Z}}$ of support regions and $\boldsymbol{\mathcal{B}}$ query RoIs, respectively. We formulate its operation for B query RoIs (refer §4.2 of main paper for support regions). Concretely, it takes, as input, RoI features $\boldsymbol{\Phi}^*_{\boldsymbol{\mathcal{B}}}\in\mathbb{R}^{2d\times NB}$ ($2d$ because layer 5 of ResNet-50 maps d-dimensional features to $2d$-dimensional features), where $\{\boldsymbol{\Phi}^*_b\in\mathbb{R}^{2d\times N}\}_{b\in\boldsymbol{\mathcal{B}}}$, and $\boldsymbol{\psi}^*_{\boldsymbol{\mathcal{B}}}\in\mathbb{R}^{d\times B}$, where $\{\boldsymbol{\psi}^*_b\in\mathbb{R}^d\}_{b\in\boldsymbol{\mathcal{B}}}$. We split $\boldsymbol{\Phi}^*_{\boldsymbol{\mathcal{B}}}$ along the channel dimension of size $2d$ to create two new matrices $\boldsymbol{\Phi}^{*u}_{\boldsymbol{\mathcal{B}}}\in\mathbb{R}^{d\times NB}$ and $\boldsymbol{\Phi}^{*l}_{\boldsymbol{\mathcal{B}}}\in\mathbb{R}^{d\times NB}$. We let $\boldsymbol{\Phi}^{*l}_b\equiv\{\boldsymbol{\phi}^{*l}_{b,1},...,\boldsymbol{\phi}^{*l}_{b,N}\in\mathbb{R}^d\}$. Self-attention is then performed over the following set $\boldsymbol{\mathcal{T}}_b$ of token vectors, in parallel across B RoIs of $\boldsymbol{\mathcal{T}}_{\boldsymbol{\mathcal{B}}}$:

$$\boldsymbol{\mathcal{T}}_b\equiv\{\boldsymbol{\phi}^{*l}_{b,1},...,\boldsymbol{\phi}^{*l}_{b,N};\bar{\boldsymbol{\phi}}^{*u}_b;\boldsymbol{W}_g\boldsymbol{\psi}^*_b\}. \tag{19}$$

where $\bar{\boldsymbol{\phi}}$ denotes average-pooled features and $\boldsymbol{W}_g \in \mathbb{R}^{d\times d}$ denotes learning weights that are shared between query and support representations.

Based on these representations between support regions (with subscript $\boldsymbol{\mathcal{Z}}$) and query RoIs, we then perform cross-attention $\mathcal{R}$, as follows:

$$\mathcal{R}_l = \left[\boldsymbol{\Phi}^{\dagger l}_{\boldsymbol{\mathcal{Z}},i} - \boldsymbol{\Phi}^{*l}_{\boldsymbol{\mathcal{B}},i}\right] \in \mathbb{R}^{d\times B},\;\; i \in \mathcal{I}_N, \tag{20}$$

$$\mathcal{R}_u = \begin{bmatrix} \bar{\boldsymbol{\phi}}^{\dagger u}_{\boldsymbol{\mathcal{Z}}} \cdot \bar{\boldsymbol{\phi}}^{*u}_{\boldsymbol{\mathcal{B}}} \\ \boldsymbol{\psi}^{\dagger}_{\boldsymbol{\mathcal{Z}}} \cdot \boldsymbol{\psi}^{*}_{\boldsymbol{\mathcal{B}}} \end{bmatrix} \in \mathbb{R}^{2d\times B}, \qquad \mathcal{R} = \begin{bmatrix} \mathcal{R}_l \\ \boldsymbol{W}_u \mathcal{R}_u \end{bmatrix} \in \mathbb{R}^{2d\times B}, \tag{21}$$

where the learnable weight $\boldsymbol{W}_u \in \mathbb{R}^{d\times 2d}$ projects the channel-wise concatenated matrix to a d dimension; operators (-) and $(\cdot)$ indicate broadcast element-wise subtraction and multiplication, respectively, where support feature is replicated B times to match with the query RoIs.

Finally, the outputs of the Z-shot and Spatial-HOP transformer heads are individually fed into a classifier, which are aggregated to form a classification score for each query RoI. Bounding-box regressor takes the output of Spatial-HOP transformer head as input for localization. The above process is shown in Fig. 3.

D Ablation Study on Transformer Relation Head (TRH) with Z-shot and Spatial-HOP blocks.

As the supplementary setting for the top panel of Tab. 3b (in the main paper), we utilize $r=1$ in RPN and $r=2,3,4$ in TRH, achieving 2.7%/2.4% improvement on novel/base classes, 5-shot protocol, over the variant applied $r=1$ in both RPN and TRH.

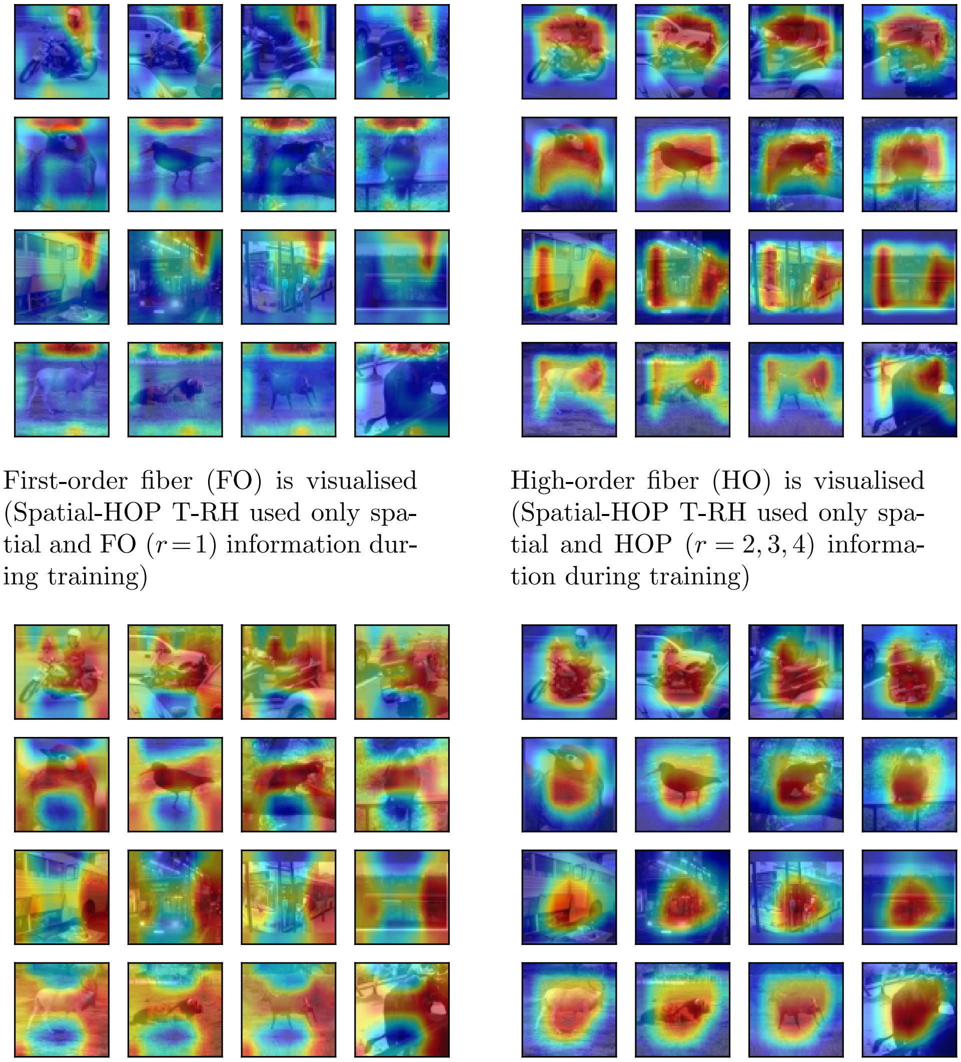
We then conduct more ablation studies on Spatial-HOP transformer head to analyze the impact brought by each component (5/10-shot setting on novel classes, VOC 2007). The results are shown on Table 5a. Specifically, we mainly ablate three variants: spatial maps of assorted size (as in the table) with either orderless HOP representation of order $r=1$ or $r=2,3,4$, or both $r=1,2,3,4$.

Furthermore, to investigate the impact of spatial attention, we use bilinearly subsampled maps, ranging from 1×1 to 7×7 in spatial size. Not surprisingly, the Spatial-HOP head performs best when utilizing larger spatial maps, together with the orderless high-order and first-order tensor descriptors.

E Visualization of Attention Maps of the Spatial-HOP block.

To explain why our model benefits from the combination of spatial attention, and orderless first-order and high-order representations, we provide qualitative results based on displaying attention maps.

Firstly, we performed training where Spatial-HOP T-RH used only spatial and first-order information (FO) during training. To obtain the picture, we



First-order fiber (FO) is visualised (Spatial-HOP T-RH used only spatial and FO ($r=1$) information during training)

High-order fiber (HO) is visualised (Spatial-HOP T-RH used only spatial and HOP ($r=2,3,4$) information during training)

Spatial fibers are max-pooled and then visualised (Spatial-HOP T-RH used spatial, FO and HOP information ($r=1,2,3,4$) during training)

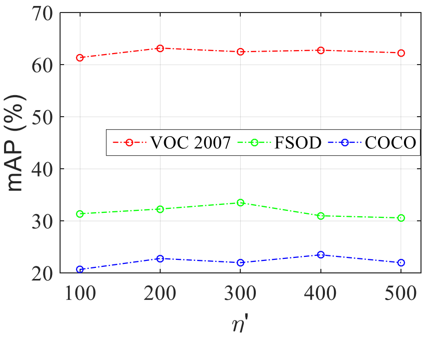
First-order fiber (FO) and High-order fiber (HO) are averaged and then visualised (Spatial-HOP T-RH used spatial, FO and HOP information ($r=1,2,3,4$) during training)

Fig. 4: Visualization of attention maps of self-attention w.r.t. support regions. The results are produced on VOC2007 test set, novel classes (motorbike, bird, bus and cow). See text for detailed descriptions.

Table 5: Experimental results of different variants of Transformer Relation Head (TRH), by varying Z-shot and Spatial-HOP blocks, are in Tab. 5a. Digits $1, ..., 4$ indicate different orders included or excluded from each experiment. 'Spatial' is the size of spatial map (downsampled by the bilinear interpolation). Next, Tab. 5c is an ablation of different variants of Encoding Network (5/10-shot setting on VOC2007 testing set was used in Tab. 5a and 5c). Finally, Fig. 5b shows mAP w.r.t. η' in SigmE (10-shot protocol on VOC2007 and COCO testing dataset, 5-shot setting on FSOD testing dataset).

Z−shot (1,2,3,4)	Spatial	1	2,3,4	5-shot	10-shot
✓	7×7	✓		57.9	64.2
			✓	61.3	65.8
		✓	✓	**62.3**	**66.9**
	5×5	✓		58.7	63.7
			✓	60.3	64.3
		✓	✓	61.1	65.2
	3×3	✓		54.8	57.9
			✓	56.0	59.2
		✓	✓	56.6	60.1
	1×1	✓		45.1	49.3
			✓	46.8	51.9
		✓	✓	47.4	52.4
	7×7	✓		55.2	60.7
			✓	59.4	64.6
		✓	✓	61.0	65.8
	5×5	✓		57.6	61.5
			✓	58.6	63.0
		✓	✓	60.3	63.4
	3×3	✓		52.4	54.8
			✓	54.1	57.8
		✓	✓	54.5	58.3
	1×1	✓		44.0	47.1
			✓	45.2	48.4
		✓	✓	46.3	50.1

(a)



(b)

EN	5-shot	10-shot
ResNet-50	62.3	66.9
Swin-B7	**62.6**	**67.4**
Swin-B12	62.0	66.7

(c)

picked $\bar{\boldsymbol{\phi}}^{\dagger u}_{\boldsymbol{\mathcal{Z}}}$ from Eq. (17) and we looked how it correlates with the N spatial representations $\boldsymbol{\phi}^{\dagger l}_{\boldsymbol{\mathcal{Z}},1}, ..., \boldsymbol{\phi}^{\dagger l}_{\boldsymbol{\mathcal{Z}},N}$. To that end, we passed these 'spatial fibers' and FO representation via the RBF kernel of Eq. (8), and we then reshaped N into the spatial map (7×7 size).

Figure 4 (top left) shows how the first-order representation (FO) correlates with each spatial fiber in the attention of transformer. As Spatial-HOP T-RH block uses information averaged over K images of the same class in an episode (K-way images), each column shows one of these support images. Each row shows a different class image from Z-shot support images in the episode.

Subsequently, we performed training where Spatial-HOP T-RH used only spatial and high-order information (HO) during training. Thus, we picked the high-order representation $\boldsymbol{W}_g\boldsymbol{\psi}^{\dagger}_{\boldsymbol{\mathcal{Z}}}$ from Eq. (17) and we looked how it correlates with the N spatial representations $\boldsymbol{\phi}^{\dagger l}_{\boldsymbol{\mathcal{Z}},1}, ..., \boldsymbol{\phi}^{\dagger l}_{\boldsymbol{\mathcal{Z}},N}$. To that end, we passed these 'spatial fibers' and HO representation via the RBF kernel of Eq. (8), and we then reshaped N into the spatial map (7×7 size).

Figure 4 (top right) shows how the high-order representation (HO) correlates with each spatial fiber in the attention of transformer. As before, we visualise $K\times Z$ images from an episode given the K-way Z-shot problem.

Comparing FO an HO representations, HO is by far more focused on the foreground objects that correlate in the semantic sense with the object class. This explains why HO representations help our model obtain better results compared to traditional attention mechanisms that focus only on capturing spatial correlations of a region.

Figure 4 (bottom left) shows how the spatial fibers from the attention matrix that is max-pooled along columns (we of course removed FO and HO before pooling along columns). We follow the same procedure as above, however, this time the Spatial-HOP T-RH block was utilizing the spatial, FO and HO information during training. Clearly, spatial attention can focus on complex spatial patterns in contrast to the focus of FO and HO.

Figure 4 (bottom right) shows how the first-order representation (FO), averaged with the high-order representation (HO), correlate with each spatial fiber in the attention of transformer. We follow the same procedure as above, and still use the spatial, FO and HO information in the Spatial-HOP T-RH block during training. Clearly, utilizing $r=1,2,3,4$ compares favourably with utilizing either $r=1$ or $r=2,3,4$ during training.

F Impact of η' of SigmE.

According to Section 4, TSO benefits from element-wise PN, realized by the SigmE operator in Eq. (5), which depends on the parameter η'. Figure 5b shows that $\eta'=200$ is a good choice on VOC dataset but $\eta'=300/400$ helps obtain the best results on FSOD/COCO dataset. Overall, our approach is not overly sensitive to this parameter, and setting $\eta'=200$ on all datasets if a good choice.

G Hyperparameters on the FSOD and COCO datasets.

Tables 6a and 6b present the impact of the number of head used in T-Heads Attention (TA) and TENET block (TB) on results. We fix the $\sigma=0.5$ (the best value of standard deviation of the RBF kernel of transformers, selected by cross-validation on FSOD and COCO dataset) and then we investigate TA and TB (the number of attention units per block, and the number of blocks, respectively). Two heads together with two blocks are the best on the FSOD dataset, while eight heads aligned with three blocks yield the best results on the COCO dataset. Table 6c shows results on FSOD and COCO w.r.t. the dimension split along the feature channel (*e.g.*, if $r=2,3$, ratio 3:1 means that three parts of channel dimension are taken to form the second-order representation, and one part of channel dimension is taken to form the third-order representation). The table also shows the impact of η_r of TSO on results, where η_r are individual parameters for each order r. Overall, using all three orders, as denoted by $r=2,3,4$, outperforms a second-order representation, indicated by $r=2$. Importantly, TSO

Table 6: Ablation studies on the FSOD and COCO datasets (5/10-shot, novel classes), w.r.t. the effect of varying (a) the number of heads used in T-Heads Attention, as shown in Tab. 6a, and (b) the number of TENET blocks as shown in Tab. 6b. mAP of variants of High-order Tensor Descriptors (HoTD) with TSO ($\eta_r > 1$) and without TSO ($\eta_r = 1$) is in Tab. 6c.

TA	FSOD	COCO
	5-shot	10-shot
1	30.5	20.1
2	**31.7**	22.3
4	31.2	22.6
8	30.8	**23.5**
16	30.0	23.0
32	29.4	21.8
64	29.5	21.5

(a)

TB	FSOD	COCO
	5-shot	10-shot
1	31.7	23.5
2	**33.5**	24.2
3	32.6	**25.1**
4	31.0	24.8
5	31.2	23.1

(b)

r			dim. split	η_r (FSOD)	5-shot		η_r (COCO)	10-shot	
2	3	4			AP_{50}	AP_{75}		AP_{50}	AP_{75}
✓				7	33.1	29.6	10	25.7	17.5
✓	✓		3:1	7,7	33.7	30.4	10,10	26.0	18.2
✓	✓	✓	5:2:1	7,7,7	**35.4**	**31.6**	10,10,10	**27.4**	**19.6**
✓	✓	✓	5:2:1	1,1,1	30.8	28.4	1,1,1	22.1	14.3

(c)

is used when $\eta_r > 1$. Without TSO ($\eta_r = 1$), results drop by a large margin, which highlights the practical importance of TSO on results.