Supplement to TDAM: Top-Down Attention Module for Contextually Guided Feature Selection in CNNs

Shantanu Jaiswal[1], Basura Fernando[1,3], and Cheston Tan[1,2,3]

[1] Institute of High Performance Computing, A*STAR, Singapore
jaiswals@ihpc.a-star.edu.sg
[2] Institute for Infocomm Research, A*STAR, Singapore
[3] Centre for Frontier AI Research, A*STAR, Singapore

1 Experiment and model implementation details

In this section of the supplement, we provide more details on model implementation and experimental frameworks. As mentioned in the paper, we utilize the Pytorch framework [10] to implement all our models and carry out experiments. We use three prominent CNN model classes – ResNet [3], MobileNetV3 [4] and ConvNeXt [9] – in our experiments. For ResNet and MobileNet, we adopt the base code for respective models from the "timm" library [17] and Pytorch's torchvision package. For ConvNeXt, we utilize official implementation code [9]. For Squeeze-Excitation (SE) [5], Convolutional Block Attention Module (CBAM) [18], Efficient Channel Attention (ECA) [15] and Frequency Channel Attention (FCA) [11] we utilize respective official implementations in PyTorch and add to all blocks of the original respective models (with the exception of MobileNetV3 models for which we use the official design with original placement of SE modules in both large and small models). We similarly implement our top-down attention module (TD) in PyTorch and add to all blocks of layers 3 and 4 for ResNet variants (with the exception of ResNet101 as stated in main text), and last 3 layers for MobileNetV3 large. Similarly, for ConvNeXt, we apply at blocks of stages 3 and 4. As mentioned in the main paper, our motivation to apply at final layers/stages is that these layers are noted to capture a higher degree of semantically relevant features [20]. We provide empirical results for applying at earlier layers in supplemental section 2.

For random initialization of weights we follow the official PyTorch implementation settings for all our models, with He initialization used for convolution layers [2] and linear layers, while batch normalization weights and biases are initialized with 1's and 0's respectively. Further details can be found in code at `https://github.com/shantanuj/TDAM_Top_down_attention_module`.

1.1 Large-scale object classification (ImageNet-1k)

Datasets. As mentioned in the paper, we utilize the official ImageNet ILSVRC-12 [1] training set comprising 1.2 million images with 1,000 object classes to

train all our models. As stated in paper, for evaluation of our models, we consider two different validation sets – the original ILSVRC-12 validation set comprising 50,000 images [1] and the "matched-frequency" split of the more recent ImageNet V2 comprising 10,000 new images [12]. We hereafter refer to these as ImageNet-V1 and ImageNet-V2 respectively. Further, for the other 2 splits of the ImageNet V2 dataset, performance increments were found to be similar and can be tested through provided code. We assess models based on their top1 and top5 validation classification accuracy for these subsets on single central crop inputs. Additionally, for models with our TD module, which output localized object predictions at each computation step (as shown in fig. 1), we only consider the most confident prediction during both training and evaluation with the exception of a minority of images that comprise multiple objects, for which we consider only predictions with unique localization maps (having an IOU of $<$ 0.5).

Model training and evaluation details. For experiments on ImageNet-1k, we utilize the "timm" library for training ResNet and MobileNet-V3 variants, while for ConvNeXt, we use the official provided source code (also built on "timm"). For ResNet and MobileNet, we use a learning rate of 0.1 for a batch-size of 128, and models are trained with cosine learning rate decay for 160 epochs (for ResNet50 and ResNet101) and 110 epochs (for ResNet18 and ResNet34) including 5 warmup epochs. We utilize label smoothing, and data augmentation of input data normalization, random horizontal flips with probability 0.5 and cutmix [19] and mixup [21] of 0.5 (only applied for larger models starting from ResNet50). We use the standard Stochastic Gradient Descent (SGD) optimizer with momentum of 0.9 and weight decay 0.0001. For validation, we use data normalization and single center crop of 224 x 224. For ConvNeXt-Tiny, we utilize official source code and settings for training, and train the model with exponential moving average weight decay of 0.9999. We train for 300 epochs with a slighty reduced batch-size of 122 per GPU (instead of original 128), 4 nodes and update frequency of 8 to make models fit in memory during training. All additional analysis including gradCAM [14] maps and feature correlation is performed on the ImageNet-1k validation set. All models were trained across 4 Nvidia Tesla V100 DGXS each with 16GB.

To operate our TD model at different computational steps, we modify the time steps of all blocks in layer 4 accordingly (from t=1 to original number of time steps). Note that during time step modification, models were not fine-tuned, and time-step modification is only utilized during validation.

For fair analysis of computational and parameter requirements, we clearly indicate in table 1 the computational, parameter complexity, computational speed (in terms of frames per second (FPS)) and computational GPU memory consumed during training and inference. The ptflops library was utilized to calculate model computational and parameter complexity. TD based models have lesser memory consumption and faster training and inference speed as model depth increases than other high-performing attention modules including CBAM [18], FCA-TS [11] and SE [5]. Additionally, we believe that utilization of top-down

feedback can be a useful constraint for neural architecture search techniques for network design in addition to the current focus on network depth, width, residual connections and cardinality.

1.2 Weakly-supervised object localization

For our next experiment of weakly-supervised object, we utilize the ImageNet-V1 object bounding boxes annotations. We do not perform any re-training or finetuning of our model, and simply utilize generated Grad-CAM maps at the final layer of our model to generate predicted bounding boxes. To do so, for fair comparison for all models, we follow the same strategy as utilized in the Grad-CAM map paper [14], wherein a Grad-CAM map is first generated for each predicted class, then binarized with a threshold of 15% of the maximum intensity to generate a corresponding heatmap, (resulting in connected segments of pixels), and finally, a bounding box is drawn around the single largest connected segment of pixels. As per the original ILSVRC-12 localization challenge, a predicted bounding box is only counted as correct if firstly, the predicted class is correct and secondly, the IOU of the predicted bounding box and ground truth bounding box is more than 0.5.

As before for ImageNet-1k classification, we report top-1 and top-5 accuracy. For models with our TD module, which output localized object predictions at each computation step (as shown in fig. 1), we only consider the most confident prediction with the exception of a minority of images that comprise multiple objects, for which we consider only predictions with unique localization maps (having an IOU of < 0.5).

1.3 Fine-grained image classification and multi-label image classification

For fine-grained image classification experiments we use the datasets Caltech Birds (CUB-200) [16] and Stanford Dogs [7]. CUB-200 comprises of 200 fine-grained bird categories, with 5,994 training images and 5,794 validation images. Stanford Dogs comprises of 120 breeds of dogs with 12,000 training images and 8,580 validation images. We adopt the 'Weakly Supervised Data Augmentation Network' (WSDAN) framework [6] and directly replace the backbone model with respective base models pre-trained on ImageNet in the previous experiment. We use a batch size of 12 with a learning rate of 1e-3 decayed every 2 epochs by a factor of 0.9 for a total of 160 epochs and image size of 448 x 448. Specific to the WSDAN framework, we make use of 32 attention maps and 'beta' as 5e-2 as done in the original method.

For multi-label image classification experiments, we use the MS-COCO [8] dataset comprising 80 object categories, and use the COCO-14 training set comprising 82,081 training images and 40,137 validation images. We utilize the 'Asymmetric loss' (ASL) implementation [13] by directly replacing the backbone with pretrained ImageNet models. We use a batch size of 64 with image size of 448 x 448 for 40 epochs and a single cycle learning rate schedule with maximum

learning rate of 1e-4 with a single cycle scheduler and percentage of total epochs before rise (pct) set to 0.2.

2 Additional experimental results

Further backbones for large scale object classification. In table 1, we provide results for all backbones and models we considered for our experiments on ImageNet-1k classification. As stated in main text, FCA-TS [11] is to our knowledge the most recent existing state-of-the-art attention module. In addition to experiments reported in main paper, we find that for ConvNeXt-Tiny, adding TD results in a 0.46% top1 improvement for ImageNet-V1 and 0.41% for ImageNet-V2. In contrast, adding FCA-TS results in comparatively minor improvements of 0.12% and 0.06% respectively.

In table 2, we provide ablation experiment results, for analysis of the choice of attention ('joint' vs 'top'), contributions of channel and spatial attention technique, feedback distance ('m') and computation steps ('t'). We also report results for adding the TD module to additional lower layers (such as layer 2), where we find performance gain is lesser than when the module is added to only higher layers (perhaps due to lack of semantically-rich features in lower layers). Further, for SE, adding the module at all layers works better than only higher layers.

Further examples of "shifting attention". We provide more examples of gradCAM map analysis of input images in fig. 1 to qualitatively analyze model processing over computational steps.

Analysis of model performance at different testing resolutions. We provide further evaluation of models at resolutions from 112x112 to 448x448 for ImageNet-V1 in fig. 2. As shown, at very low resolutions (below 168x168), model performance drops drastically for both attention modules and the original ResNet model. TD-based models remain relatively robust to alternate attention modules at these resolutions too.

3 Additional model analysis for feature selectivity

In this section of the supplement, we provide more details of model analysis and further analysis including an approach to quantify selective co-activation of input and output channels and selectivity in output layer channels. We do this by analyzing pairwise correlation between input and output channels of respective feature maps and entropy of both inter-channel pairwise correlation and block outputs in the final layer of the model.

3.1 Approach

First, we analyze selective co-activation between channels of input feature map and output feature map by assessing two factors – mean correlation between input and output feature maps and entropy of the softmaxed distribution of

Method	BB.	Param.	FLOPs	Mem.(Gb)		FPS/gpu		ImageNet-V2		ImageNet-V1	
-	-	-	-	Trn	Val	Trn	Val	Top1	Top5	Top1	Top5
ResNet [3]	Resnet18	11.69 M	1.82 G	11.9	5.1	538	1683	57.53	80.54	70.51	89.66
SE [5]		11.78 M	1.82 G	12.2	5.3	520	1547	58.11	80.83	70.84	89.97
CBAM [18]		11.78 M	1.82 G	13.4	5.4	452	1037	58.56	81.30	70.99	90.13
ECA [15]		11.69 M	1.82 G	12.2	5.1	512	1491	58.06	80.88	70.64	89.78
FCA-TS [11]		11.78 M	1.82 G	12.3	5.4	504	1382	58.49	81.22	70.93	90.01
TDtop (t=2, m=2)		11.77 M	2.63 G	12.0	5.3	459	1233	59.02	**81.60**	**71.59**	**90.44**
TDjoint (t=2, m=1)		11.86 M	2.29 G	11.8	5.3	493	1404	**59.31**	81.53	71.55	90.35
ResNet [3]	Resnet34	21.80 M	3.68 G	14.9	6.0	372	954	62.39	83.05	73.63	91.34
SE [5]		21.95 M	3.68 G	15.5	6.1	340	904	62.75	83.39	73.97	91.79
CBAM [18]		21.96 M	3.68 G	16.9	6.4	278	692	62.89	84.01	74.21	**92.05**
ECA [15]		21.80 M	3.68 G	15.4	6.1	341	921	62.71	83.31	73.86	91.53
FCA-TS [11]		21.95 M	3.68 G	15.7	6.2	325	880	62.94	83.91	74.20	91.91
TDtop (t=2, m=2)		21.95 M	5.64 G	16.0	6.4	270	760	**63.22**	**84.26**	**74.55**	92.00
TDjoint (t=2, m=1)		22.10 M	4.72 G	15.1	6.3	297	816	63.08	83.97	74.43	91.91
ResNet [3]	Resnet50	25.56 M	4.12 G	29.5	16.1	704	2143	66.39	86.59	77.51	93.64
SE [5]		28.07 M	4.13 G	32.4	16.0	615	1911	66.92	86.88	78.03	93.88
CBAM [18]		28.07 M	4.14 G	37.6	20.7	420	1442	67.28	87.04	78.59	93.95
ECA [15]		25.56 M	4.13 G	31.5	16.1	652	1989	66.72	86.95	78.11	93.85
FCA-TS [11]		28.07 M	4.13 G	32.4	16.3	590	1876	67.19	87.02	78.70	94.01
TDjoint (t=2, m=1)		27.65 M	4.59 G	31.9	16.2	601	1890	67.66	87.02	**78.96**	94.19
TDtop (t=2, m=1)		27.06 M	4.59 G	31.8	16.0	612	1905	67.21	86.98	78.82	93.98
TDtop (t=2, m=3)		27.66 M	5.98 G	35.3	16.3	498	1539	**67.70**	**87.08**	78.90	**94.23**
ResNet [3]	Resnet101	44.55 M	7.85 G	39.2	16.6	460	1376	69.64	89.09	80.36	95.31
SE [5]		49.29 M	7.86 G	45.5	16.9	368	1201	69.88	89.17	80.84	95.42
CBAM [18]		49.29 M	7.88 G	53.3	21.4	269	862	70.03	89.35	81.20	95.64
FCA-TS [11]		49.29 M	7.86 G	47.0	17.1	312	1164	70.12	89.42	81.15	95.59
TDjoint (t=2, m=1)		46.75 M	8.37 G	41.0	16.8	396	1237	**70.56**	**89.44**	**81.62**	**95.76**
TDjoint (t=2, m=1, L4)		45.94 M	8.01 G	40.3	16.8	413	1258	70.28	89.39	81.12	95.49
ConvNeXt-Tiny [9]	eXt	28.19 M	4.46 G	54.4	19.1	259	2001	70.83	**89.63**	81.31	95.75
FCA-TS [11]		28.90 M	4.46 G	58.2	19.6	237	1965	70.89	89.41	81.42	95.69
TDjoint (t=2, m=1)		30.28 M	5.50 G	62.0	19.2	220	1942	**71.24**	89.59	**81.77**	**95.79**
TDjoint (t=2, m=2)		29.05 M	6.54 G	61.6	19.2	228	1842	71.10	89.52	81.71	95.72

Table 1. Top1 & Top5 single-crop classification accuracy (%) of models integrated with our TD module in comparison to baselines on original ImageNet-V1 [1] and recent ImageNet-V2 [12] validation sets.

each output channel feature map's correlation vector with all feature maps of input channels. A low entropy value in this case indicates an output channel highly correlating (and thereby co-activating) with a select few input channels. Second, we analyze precision in output channel activations by measuring the entropy of the softmaxed distribution of corresponding max-pooled values. In higher semantically-richer layers, a low entropy value indicates high activation of a select few output channels.

Formally, we compute correlation $\rho(\mathbf{X}, \mathbf{Y})$ between an input feature map $\mathbf{X} = \mathbf{X_t^0}$ with channels $i \in \{1..C_X\}$ and output feature map $\mathbf{Y} = \mathbf{X_t^N}$ with channels $j \in \{1..C_Y\}$, and corresponding entropy $\mathbf{H}(\mathbf{X}, \mathbf{Y^j})$ of each output channel

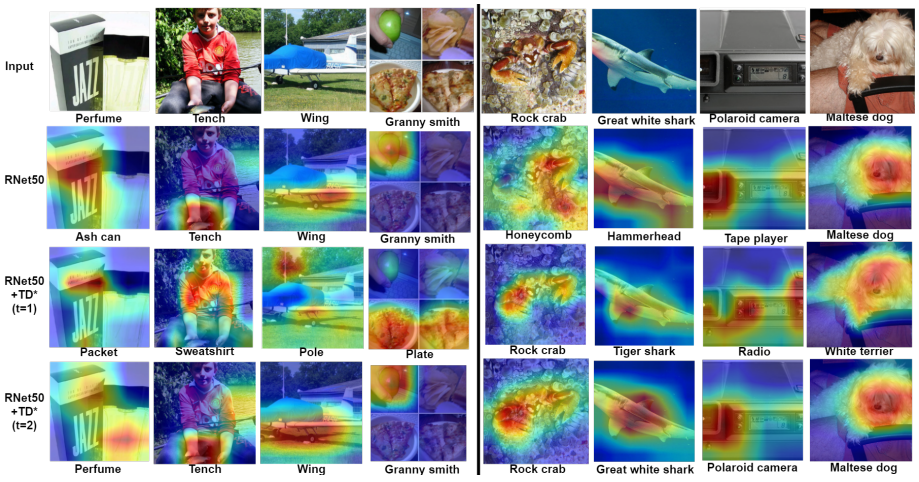

Fig. 1. Further representative examples of "attention shifting" over computational steps of our model based on Grad-CAM analysis. In the first 4 examples, the TD model iteratively attends to distinct objects and also has a more selective and complete feature activation at each computation step compared to original ResNet50. In the further 4 examples, it iteratively attends to relevant features for better discrimination of finer classes.

j's correlation vector with input channels as follows:

$$\rho(\mathbf{X^i}, \mathbf{Y^j}) = \sum_{\mathbf{m=1}}^{\mathbf{H_i}} \sum_{\mathbf{n=1}}^{\mathbf{W_i}} \mathbf{X'^i} \otimes \mathbf{Y'^j} \tag{1}$$

$$\rho(\mathbf{X}, \mathbf{Y}) = \frac{\sum_{\mathbf{i=1}}^{\mathbf{C_X}} \sum_{\mathbf{j=1}}^{\mathbf{C_Y}} \rho(\mathbf{X^i}, \mathbf{Y^j})}{\mathbf{C_X C_Y}} \tag{2}$$

$$\mathbf{H}(\mathbf{X}, \mathbf{Y^j}) = \mathbf{H}(\sigma(\rho(\mathbf{X}, \mathbf{Y^j}))) \tag{3}$$

where $\mathbf{X'^i}$ and $\mathbf{Y'^j}$ correspond to normalized feature map of channel i and channel j of input $\mathbf{X}$ and output $\mathbf{Y}$ respectively. $\mathbf{H_i}$ and $\mathbf{W_i}$ refer to spatial dimensions of both $\mathbf{X}$ and $\mathbf{Y}$ (necessarily equal spatial dimensions), $\otimes$ denotes element-wise multiplication, σ denotes the softmax function and $\mathbf{H}$ denotes entropy. We make use of min-max spatial normalization of each feature map to obtain spatial activations between 0 and 1. For entropy calculation, no spatial normalization is used.

Further, we calculate entropy $\mathbf{H}(\mathbf{Y})$ of the output feature map $\mathbf{Y} = \mathbf{X_t^N}$ as follows:

$$\mathbf{H}(\mathbf{Y}) = \mathbf{H}(\sigma(\mathbf{MaxPool}(\mathbf{Y}))) \tag{4}$$

where MaxPool is performed to squeeze spatial dimensions.

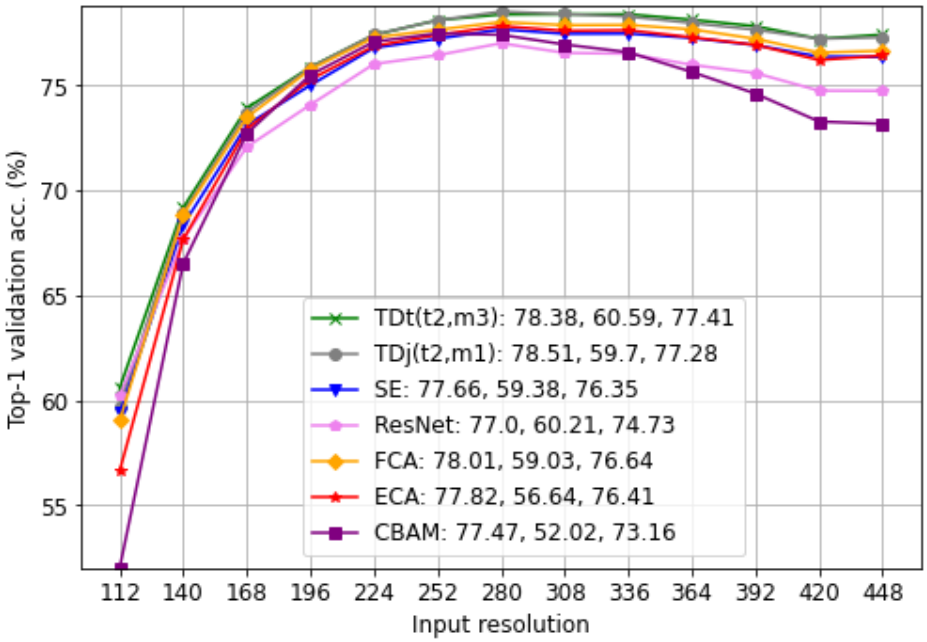

Fig. 2. Performance of models (ResNet50 backbone) on ImageNet-V1 (ILSVRC-12 [1]) at different test resolutions with best accuracy, accuracy at 112x112 and accuracy at 448x448 reported in plot legend. TD models obtain better results at both lower and higher resolutions than alternate attention modules.

3.2 Analysis results

We perform analysis on ResNet50 models trained on ImageNet-1k classification and use feature activations for the entire ImageNet-V1. In the supplement, we report analysis results for layer 4 blocks of the model. Since the input to the first block of both layers does not have the same spatial dimensions as its output, we do not consider the first block of either layer in analysis.

Selective co-activation of input and output channels. As shown in tables 3, we first find that the mean correlation $\rho(\mathbf{X}, \mathbf{Y})$ between input and output feature maps for models with our top-down (TD) module is higher in the final layer convolutional blocks than the original ResNet50 model. Second, as shown in histogram plots in fig. 3, a significantly higher number of output channels in layer4 have decreased $\mathbf{H}(\mathbf{X}, \mathbf{Y^j})$ for our optimal TD configuration (TDtop(t=2,m=3)) in comparison to the original ResNet50 model (we also find a similar result in fig. 4 for layer3 blocks). Taking both these factors into account indicates that for models with our TD module, a greater number of output channels co-activate with a select set of input channels (based on histogram of $\mathbf{H}(\mathbf{X}, \mathbf{Y^j})$) and with greater intensity (based on increased $\rho(\mathbf{X}, \mathbf{Y})$). Additionally,

as shown in table 3, models with TD have decreased $\mathbf{H}(\mathbf{Y})$ indicating more selective channel activations in output feature map.

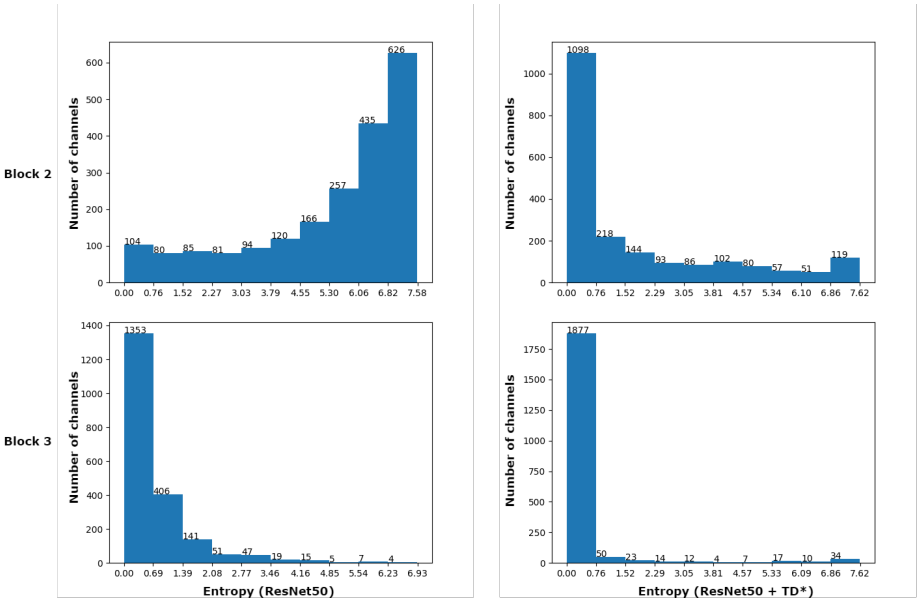

Fig. 3. Histogram of mean entropy of pairwise correlation ($\mathbf{H}(\mathbf{X}, \mathbf{Y}^{\mathbf{j}})$) b/w input and output feature maps for output channels of ResNet50 layer 4 blocks 2 and 3. Decreased entropy indicates an output channel co-activates with a select few input channels. A notably higher number of output channels have decreased entropy in TD-based model.

Model	Backbone	Param.	FLOPs	ImageNet-V1	
-	-	-	-	Top1	Top5
ResNet	ResNet50	25.56 M	4.12 G	76.02	92.95
TDt (m=0)		27.92 M	4.13 G	76.06	92.93
TDj (m=1)		27.65 M	4.59 G	77.41	93.62
TDt (m=1)		27.06 M	4.59 G	77.36	93.59
TDj (m=2)		27.65 M	5.63 G	76.98	93.48
TDt (m=2)		27.06 M	5.63 G	77.33	93.52
TDj (m=3)		29.82 M	5.98 G	**77.44**	**93.71**
TDt (m=3)		27.66 M	5.98 G	77.40	93.63
TDj (L 3,4)	Resnet50	27.65 M	4.59 G	**77.41**	**93.62**
SE		28.07 M	4.13 G	76.77	93.49
SE (L 3,4)		27.92 M	4.13 G	76.64	93.35
TDj (L 4)		26.95 M	4.28 G	76.95	93.47
TDj (L 2,3,4)		27.76 M	4.80 G	77.10	93.52
ResNet	ResNet18	11.37 M	1.82 G	74.41	92.09
TDt (t=2)		11.37 M	2.63 G	75.06	92.13
TDt (t=3)		11.37 M	3.45 G	**75.34**	**92.27**
TDt (t=4)		11.38 M	4.25 G	75.18	92.19
TDt (t=5)		11.38 M	5.06 G	75.14	92.18
MbNet	MobileNet-V3(Large)	4.45 M	0.23 G	75.14	92.24
TDj (t=2)		3.71 M	0.26 G	75.43	92.31
TDj (t=3)		3.72 M	0.29 G	**75.57**	**92.39**
TDj (t=4)		3.72 M	0.32 G	75.45	92.35
TDj (t=5)		3.73 M	0.35 G	75.28	92.29
Chn→ Sp	ResNet50	27.65 M	4.59 G	**78.47**	**94.59**
Chn||Sp		27.65 M	4.59 G	77.61	93.98
Chn only		27.65 M	4.59 G	76.98	93.35
Sp only		27.65 M	4.59 G	76.02	92.01
Sp→ Chn		27.65 M	4.59 G	76.10	92.05
Rec_Conv		29.65 M	4.95 G	77.43	93.82
Identity		25.58 M	5.56 G	76.21	92.13

Table 2. Ablative analysis results for our TD module. First row set: choice of attention operation (TDjoint or TDtop) and feedback distance 'm' (with ResNet50 on ImageNet-1k). Second row set: adding TDjoint (with m=1) and SE to alternative layers. Third row set: feedback computation steps 't' for ResNet18 and MobileNetV3 (Large) on a hierarchically reduced subset of ImageNet-1k with 200 classes. Fourth row set: choice of feedback attention technique.

Blk.	-	Original	TDt(2,3)	TDj(2,1)
Blk.2	ρ	1.1 ± 0.1	**1.5 ± 0.2**	1.3 ± 0.2
	H	5.67	4.86	5.09
Blk.3	ρ	1.7 ± 0.1	**2.0 ± 0.18**	1.9 ± 0.2
	H	2.52	1.85	2.14

Table 3. ResNet50 Layer 4 analysis results: Mean pairwise-correlation (ρ(X,Y)) b/w input and output feature maps denoted by ρ and output activation entropy (H(Y)) denoted by H. TDt(2,3) is TDtop(t=2, m=3) and TDj(2,1) is TDjoint(t=2, m=1).

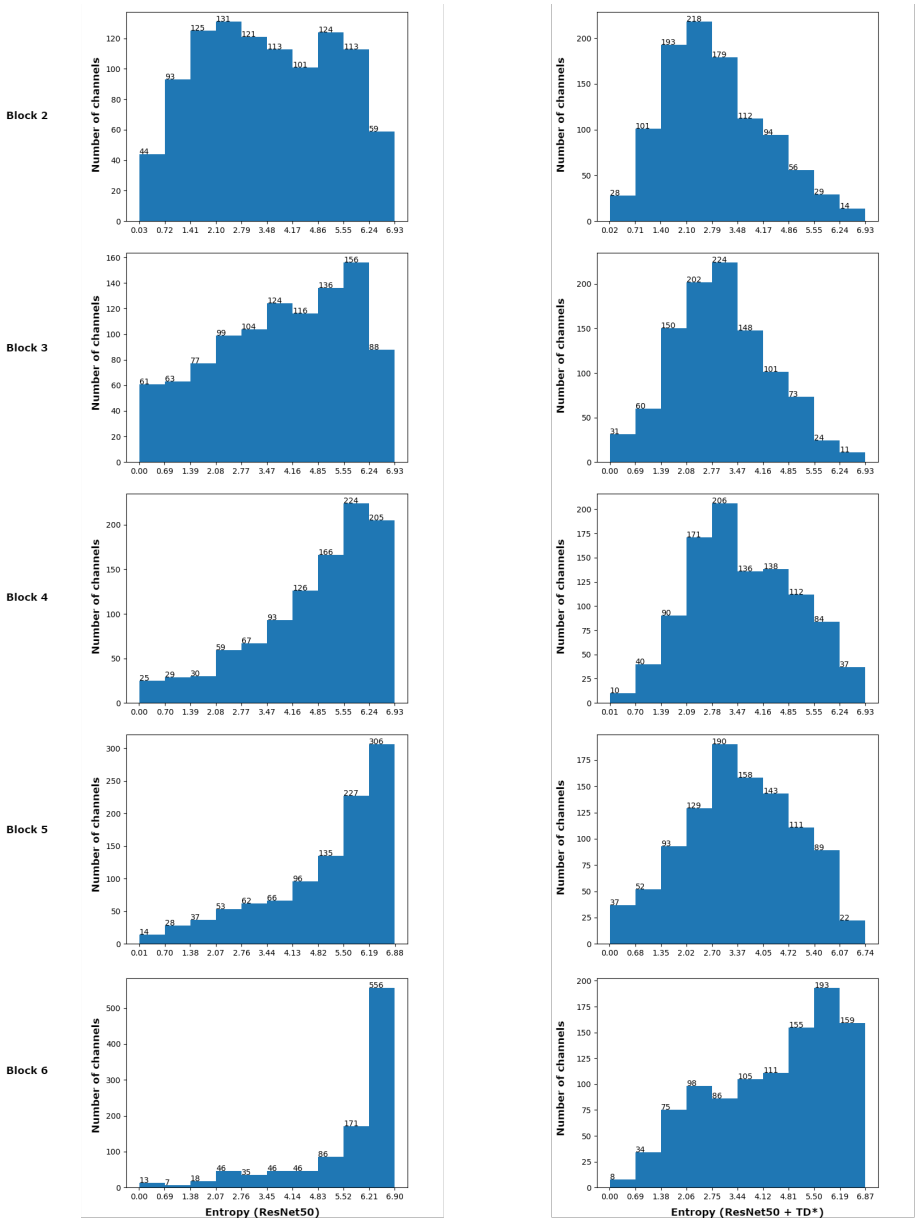

Fig. 4. Histogram of mean entropy of pairwise correlation ($H(X, Y^j)$) b/w input and output feature maps for ResNet50 layer3 blocks 2-6.

References

1. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
2. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
3. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
4. Howard, A., Sandler, M., Chu, G., Chen, L.C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., et al.: Searching for mobilenetv3. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1314–1324 (2019)
5. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7132–7141 (2018)
6. Hu, T., Qi, H., Huang, Q., Lu, Y.: See better before looking closer: Weakly supervised data augmentation network for fine-grained visual classification. arXiv preprint arXiv:1901.09891 (2019)
7. Khosla, A., Jayadevaprakash, N., Yao, B., Fei-Fei, L.: Novel dataset for fine-grained image categorization. In: First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition. Colorado Springs, CO (June 2011)
8. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
9. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. arXiv preprint arXiv:2201.03545 (2022)
10. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 32. Curran Associates, Inc. (2019), `https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf`
11. Qin, Z., Zhang, P., Wu, F., Li, X.: Fcanet: Frequency channel attention networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 783–792 (2021)
12. Recht, B., Roelofs, R., Schmidt, L., Shankar, V.: Do imagenet classifiers generalize to imagenet? In: International Conference on Machine Learning. pp. 5389–5400. PMLR (2019)
13. Ridnik, T., Ben-Baruch, E., Zamir, N., Noy, A., Friedman, I., Protter, M., Zelnik-Manor, L.: Asymmetric loss for multi-label classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 82–91 (2021)
14. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)

15. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: Eca-net: efficient channel attention for deep convolutional neural networks, 2020 ieee. In: CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (2020)
16. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-UCSD Birds 200. Tech. Rep. CNS-TR-2010-001, California Institute of Technology (2010)
17. Wightman, R.: Pytorch image models. `https://github.com/rwightman/pytorch-image-models` (2019). https://doi.org/10.5281/zenodo.4414861
18. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: Proceedings of the European conference on computer vision (ECCV). pp. 3–19 (2018)
19. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6023–6032 (2019)
20. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: European conference on computer vision. pp. 818–833. Springer (2014)
21. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 (2017)