Rotate to Attend: Convolutional Triplet Attention Module

Diganta Misra *
Landskape
mishradiganta91@gmail.com

Trikay Nalamada *
Indian Institute of Technology, Guwahati
nalamada.trikay@gmail.com

Ajay Uppili Arasanipalai *
University of Illinois, Urbana Champaign
aua2@illinois.edu

Qibin Hou
National University of Singapore
andrewhoux@gmail.com

1. More Experiments

In this section, we provide results for additional experiments that we ran to evaluate the performance of triplet attention on other vision tasks adjacent to the main focus on image classification and object detection in the paper.

In particular, we expand our Mask RCNN model to use a keypoint detection head, as specified in [5], and evaluate the existing Mask-RCNN model on the COCO instance segmentation task. We also observe the effect of kernel size k in the convolution operations within the triplet attention module added to different standard architectures.

In addition, we provide more GradCAM [10] and GradCAM++ [2] visualizations, and observe some interesting patterns in the resulting heatmaps, which we discuss further in Sec. 3.

2. Effect of kernel size k

Architecture	Dataset	k	Param.	FLOPs	Top-1 (%)
ResNet-20 [6]	CIFAR-10	3	**0.270M**	**2.011G**	92.66
		5	0.271M	2.019G	92.71
		7	0.272M	2.032G	**92.91**
VGG-16 + BN [11]	CIFAR-10	3	**15.254M**	**0.316G**	91.73
		5	15.255M	0.317G	92.05
		7	15.256M	0.32G	**92.33**
ResNet-18 [6]	ImageNet	3	**11.69M**	**1.823G**	70.33
		7	11.69M	1.825G	**71.09**
ResNet-50 [6]	ImageNet	3	**25.558M**	**4.131G**	76.12
		7	25.562M	4.169G	**77.48**
MobileNetV2 [9]	ImageNet	3	**3.506M**	**0.322G**	**72.62**
		7	3.51M	0.327G	71.99

Table 1. Effect of kernel size k for triplet attention in standard CNN architectures on CIFAR-10 [7] and ImageNet [3]. We observe a general trend of improvement in performance with increasing kernel size aside from MobileNetV2.

*Equal Contribution

We do baseline experiments to compare the effect of using different kernel sizes k in triplet attention and show our results in Tab. 1. We conduct experiments on both CIFAR-10 and ImageNet with different network architectures to demonstrate the flexibility of the proposed triplet attention. From Tab. 1, we observe a general trend of improvement in performance with increasing kernel size. When deployed in lighter-weight models, like MobileNetV2 [9], we observed a smaller kernel to outperform its larger kernel counterpart and thus overall have less complexity overhead.

3. GradCAM

In addition to the GradCAM results presented in the paper, we observed many more instances of triplet attention generating heatmaps that are consistently tighter or wider when required and more meaningful. We use the same method that we followed in the paper to obtain GradCAM [10] and GradCAM++ [2] heatmap visualizations for the ImageNet [3] test set images that we illustrate in Fig. 1.

The most interesting visualization is in the first example (left image on the first row). The image shows two devices - one that resembles a cassette player and an iPod. While this image could potentially benefit from multiple labels and bounding boxes, the class prescribed by the ImageNet dataset is ”TapePlayer” (predicted correctly by triplet attention) and not ”iPod” (the top class prediction from both CBAM and the vanilla ResNet50). We speculate that the attention maps in triplet attention help the model develop an accurate estimate of global, long-range dependencies in the image. Since the iPod is smaller, its distinct circular control pad coupled with the locality of the discrete convolution operator employed by the ResNet architecture could potentially mislead the network toward predicting the smaller, more recognizable object.

The second example (right image on the first row) also demonstrates an incorrect class prediction that can be attributed to an inability to capture global features. All mod-

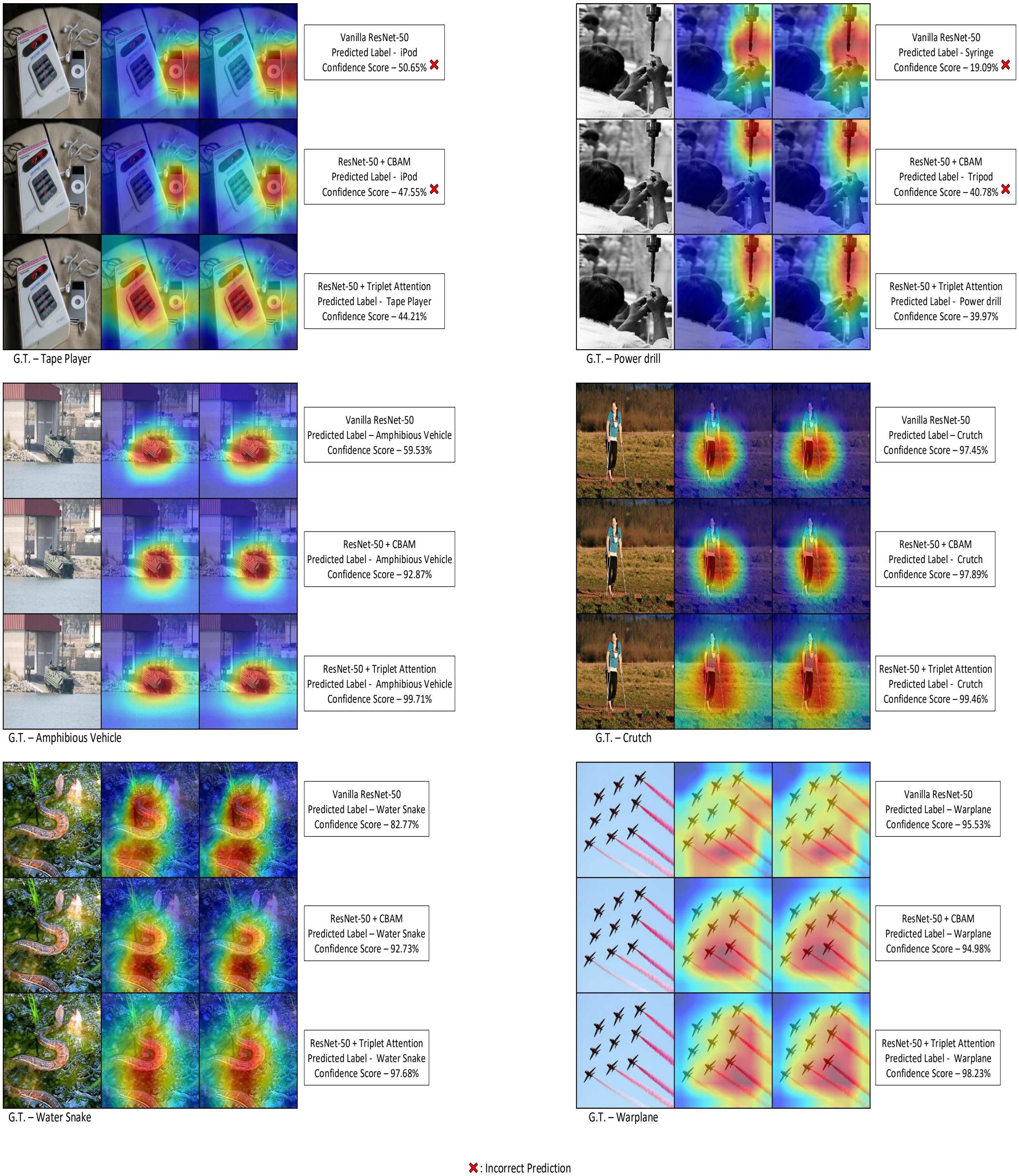

Figure 1. **Visualization of GradCAM and GradCAM++ results.** The results were obtained for six random samples from the ImageNet validation set and were compared for a baseline ResNet-50, CBAM integrated ResNet-50 and a triplet attention integrated ResNet-50 architecture. Ground truth (G.T) labels for the images are provided below the original samples and the networks prediction and confidence scores are provided in the corresponding boxes.

Backbone	Detectors	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
ResNet-50 [6]	Mask RCNN [5]	34.2	55.9	36.2	**18.2**	37.5	46.3
ResNet-50 + 1 NL block [12]		34.7	56.7	36.6	-	-	-
GCNet [4]		35.7	**58.4**	37.6	-	-	-
ResNet-50 + Triplet Attention (Ours)		**35.8**	57.8	**38.1**	18.0	**38.1**	**50.7**

Table 2. **Instance Segmentation mAP (%) on MS-COCO** : Triplet Attention results in higher performance gain with minimal computational overhead

Backbone	Detectors	AP	AP_{50}	AP_{75}	AP_M	AP_L
ResNet-50 [6]	Keypoint RCNN	63.9	86.4	69.3	59.4	72.4
ResNet-50 + CBAM [13]		**64.8**	85.5	**70.9**	**60.3**	72.8
ResNet-50 + Triplet Attention (Ours)		64.7	**85.9**	70.4	**60.3**	**73.1**

Table 3. **Person Keypoints Detection baselines**: Triplet Attention provides improvement over vanilla architecture and competitive results as compared to the more complex CBAM incorporated model.

Backbone	Detectors	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
ResNet-50 [6]	Keypoint RCNN	53.6	82.2	58.1	36	61.4	70.8
ResNet-50 + CBAM [13]		54.3	82.2	59.3	37.1	**61.9**	71.4
ResNet-50 + Triplet Attention (Ours)		**54.8**	**83.1**	**59.9**	**37.4**	**61.9**	**72.1**

Table 4. **Object detection mAP(%) on the MS COCO validation set using the Keypoint RCNN**. Triplet Attention results in consistent higher performance gains across all the metrics.

els focus on a similar region of the image, but CBAM and vanilla ResNet predict the wrong class with reasonably high accuracy. Predicting *power drill* correctly for this image likely requires a representation of the global context since there seem to be few local features that can be associated with that class label. The other heatmaps continue to suggest that triplet attention produces tighter and more discriminative bounds when appropriate, across a variety of image classes.

4. COCO Instance Segmentation

The Mask RCNN architecture introduced in [5] produces segmentation masks in addition to bounding boxes. We use the Mask RCNN model augmented with our triplet attention layer, trained on the COCO 2017 dataset (as described in section 4.3 of the main paper) to perform instance segmentation, using the detectron2 code base [14]. We provide our results of various AP scores in Tab. 2 along with results from other models that used similar training schemes. On instance segmentation, triplet attention continues to provide a substantial improvement (nearly a 6% increase across AP scores at negligible computational overhead) over the baseline ResNet50 model and also outperforms other newer, larger models like GCNet [1].

5. COCO Keypoint Detection

In addition to the other COCO segmentation and object detection tasks, we further train the Mask RCNN model on the COCO human keypoint detection task. The training configuration is similar to that we used for our Mask RCNN model on the instance segmentation and object detection tasks - we use the same 1x training schedule with identical values for batch size, learning rate, et cetera. as we did for our Mask RCNN model as well as the baseline [5]. For the keypoint detection head, the model generates 1500 proposals per image using the region proposal network implemented in Faster RCNN [8], which is implemented as the default configuration in detectron2 [14].

We provide a table of results comparing our Mask RCNN based keypoint detector to the baseline implementation as well as CBAM [13], another method that computationally much more expensive yet obtains similar results. Tab. 3 provides the resulting AP scores for the keypoint annotations on the COCO 2017 validation set. Tab. 4 provides the AP scores for the bounding box annotations, which we generate while training on the keypoint annotations.

References

[1] Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *Proceedings of the IEEE Inter-*

national Conference on Computer Vision Workshops, pages 0–0, 2019.

[2] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847. IEEE, 2018.

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

[4] Zilin Gao, Jiangtao Xie, Qilong Wang, and Peihua Li. Global second-order pooling convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3024–3033, 2019.

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

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

[7] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

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

[9] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[10] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[11] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[12] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.

[13] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[14] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.