Appendices of "Inner-Imaging Networks: Put Lenses into Convolutional Structure"

Yang Hu, *Student Member, IEEE,* Guihua Wen, Mingnan Luo, Dan Dai, Wenming Cao, Zhiwen Yu, *Senior Member, IEEE,* and Wendy Hall

APPENDIX A
SUPPLEMENTATION ON ANALYSIS OF DIFFERENT INNER-IMAGING TYPES

TABLE I: Comparison results of InI-models with or without dilated G-filter over 5 runs.

Model	Add dilated G-filter	CIFAR-10
InI-ResNet-164-square-3	no	4.24 ± 0.10
InI-ResNet-164-square-3-d	yes	**4.15 ± 0.08**
InI-ResNet-164-mix-5	no	4.15 ± 0.09
InI-ResNet-164-mix-5-d	yes	**4.11 ± 0.13**
InI-WRN-16-8-square-3	no	3.90 ± 0.09
InI-WRN-16-8-square-3-d	yes	**3.85 ± 0.13**
InI-WRN-16-8-mix-5	no	3.84 ± 0.07
InI-WRN-16-8-mix-5-d	yes	**3.80 ± 0.11**
InI-WRN-28-10-square-3	no	3.28 ± 0.10
InI-WRN-28-10-square-3-d	yes	**3.24 ± 0.04**
InI-WRN-28-10-mix-5	no	3.21 ± 0.09
InI-WRN-28-10-mix-5-d	yes	**3.19 ± 0.10**

The comparison results of the InI models with or without dilated G-filter are presented in Table I. It can be seen that the dilated G-filter helps the InI-models improve further on the original bases.

About effects of the shape of inner-imaged map. We compare the different shapes of the folded inner-imaged maps, which causes little fluctuations in results, as Fig. 1 shows, the inner-imaged map closer to the square performs slightly better.

APPENDIX B
COMPUTATIONAL EFFICIENCY COMPARISON OF EXPERIMENTAL MODELS

In this part, we give the floating-point computation statistics (FLOPs) and the training time record of each round of Epoch (Epoch Time) of the proposed Inner-Imaging (InI) model and all comparison models, to analyze the rationality of the proposed approach in terms of computational consumption.

Yang Hu, Guihua Wen, MingnanLuo, Dan Dai, Zhiwen Yu are with the School of Computer Science and Engineering, South China University of Technology, China. Yang Hu is also with University of Southampton, UK. Email: superhy199148@hotmail.com, crghwen@scut.edu.cn, Phone no: +86-18998384808.

Wenming Cao is with the Department of Computer Science, City University of Hong Kong, Hong Kong. Email:wenmincao2-c@my.cityu.edu.hk.

Wendy Hall is with the Web Science Institute, University of Southampton, UK. Email:wh@ecs.soton.ac.uk, Phone no: +44(0)2380592388.

Guihua Wen is the corresponding author.

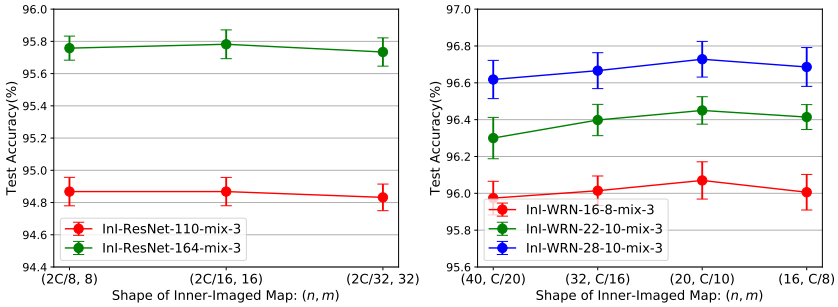

Fig. 1: Test accuracy curves by InI-ResNet (left) and InI-WRN on CIFAR-10, concerning the shape of inner-imaged map.

TABLE II: Various types of G-filter sets.

Type	Name	Set of G-filters
Square	square-1	$\{(3 \times 3)\}$
	square-2	$\{(1 \times 1), (3 \times 3)\}$
	square-3	$\{(1 \times 1), (3 \times 3), (5 \times 5)\}$
	square-4	$\{(1 \times 1), (2 \times 2), (3 \times 3), (5 \times 5)\}$
	square-5	$\{(1 \times 1), (2 \times 2), (3 \times 3), (4 \times 4), (5 \times 5)\}$
Mix	mix-1	$\{(3 \times 3)\}$
	mix-2	$\{(1 \times 5), (3 \times 3)\}$
	mix-3	$\{(1 \times 5), (3 \times 3), (5 \times 1)\}$
	mix-4	$\{(1 \times 1), (1 \times 5), (3 \times 3), (5 \times 1)\}$
	mix-5	$\{(1 \times 1), (1 \times 5), (3 \times 3), (5 \times 1), (5 \times 5)\}$
Horizontal	horizon-n	$\{(1 \times 1), \ldots, (1 \times n)\}$
Vertical	vertical-n	$\{(1 \times 1), \ldots, (n \times 1)\}$
Simplified[a]	simple-1	$\{(2 \times 1)\}$
	simple-3	$\{(1 \times 1), (2 \times 1), (2 \times 2)\}$
Dilated[b]	d	$\{\ldots, (5 \times 5, s = 2)\}$

[a]Used only for simplified version of InI module in ResNets.
[b]Used only in conjunction with other types of G-filters, not separately.

The configuration of the experimental equipment is as follows: GPU: GTX 1080Ti * 4, CPU: Intel Xeon E5-2620 * 2, Memory: 64G. On the CIFAR and SVHN datasets, we only use one GPU, on the ImageNet dataset, we use 4 GPUs in parallel. All experiments are implemented using the deep learning framework MXNet[1].

Tables III and IV list the FLOPs and Epoch Time of the various modes of the InI model and the compared models. These two tables are corresponding with the ablation studies in the main content of this paper. Table V lists the FLOPs and Epoch Time of the comparison attention-based models, and computational consumption information of experimental models on the ImageNet dataset are listed in Table VI.

Compared with the same backbone SE-Net, any models using the InI mechanism only increases the calculation of FLOPs by less than 0.3%, and the extra training time is less than 10 seconds per epoch. Even if compared with the backbone model, the increase in FLOPs of InI models is not more than 0.5%. On the ImageNet dataset, the InI model has achieved the performance improvement described in the main text while only costs a tiny amount of computational increase, compared with SE-Net and the corresponding

[1]https://mxnet.apache.org/

TABLE III: Computing consumption(FLOPs and Averaged wall-clock time per training epoch) of All-CNN, ResNet, SE-ResNet and multiple modes of InI-models on CIFAR-100.

Model	Joint	Aggregation	Fold	Dilated	FLOPs	Epoch Time
SE-All-CNN [1]	–	–	–	–	156.4M	17 sec
InI-All-CNN-square-1 (ours)	–	–	√	–	156.5M	19 sec
InI-All-CNN-square-3 (ours)	–	√	√	–	156.7M	20 sec
ResNet-110 [2]	–	–	–	–	253.2M	37 sec
ResNet-164 [2]	–	–	–	–	380.6M	65 sec
SE-ResNet-110 [1]	–	–	–	–	253.9M	53 sec
SE-ResNet-164 [1]	–	–	–	–	381.6M	82 sec
InI-ResNet-110-square-1* (ours)	–	–	√	–	254.0M	54 sec
InI-ResNet-110-simple-1 (ours)	√	–	–	–	254.1M	55 sec
InI-ResNet-110-simple-3 (ours)	√	√	–	–	254.3M	57 sec
InI-ResNet-110-square-1 (ours)	√	–	√	–	254.2M	56 sec
InI-ResNet-110-square-3 (ours)	√	√	√	–	254.5M	59 sec
InI-ResNet-110-square-3-d (ours)	√	√	√	√	254.6M	60 sec
InI-ResNet-164-square-1* (ours)	–	–	√	–	381.6M	82 sec
InI-ResNet-164-simple-1 (ours)	√	–	–	–	381.7M	84 sec
InI-ResNet-164-simple-3 (ours)	√	√	–	–	381.8M	85 sec
InI-ResNet-164-square-1 (ours)	√	–	√	–	381.7M	84 sec
InI-ResNet-164-square-3 (ours)	√	√	√	–	381.9M	87 sec
InI-ResNet-164-square-3-d (ours)	√	√	√	√	381.9M	88 sec
InI-ResNet-164-mix-5 (ours)	√	√	√	–	382.1M	90 sec
InI-ResNet-164-mix-5-d (ours)	√	√	√	√	382.2M	91 sec

∗Without joint modeling of residual and identity mappings in ResNets.

backbone, the FLOPs calculation amount of the InI model is increased by about 0.2% and 0.3%.

APPENDIX C
TRAINING/TESTING LOGS OF THE REPRESENTATIVE EXPERIMENTAL MODELS

In this section, We illustrate the training log curves of some representative models to facilitate the subsequent researches and analysis of the training optimization process of the proposed Inner-Imaging network.

Since the image recognition task is essential in the field of computer vision. For deep computer vision models, their proposals need to take the state-of-the-art performance on image classification to prove their ability. So, on well-known image classification data sets, such as CIFAR, further performance improvement has been much more challenging than in the first few years. Nevertheless, the proposed InI model can achieve good performance improvement on the basis of the ordinary channel-wise attentional SE-Net.

Figure 2 shows the training log curves of SE-Net and InI-mix-5-Net on both CIFAR-10 and CIFAR-100 datasets, from Figure 2, we can notice that although on the training error rate curve, InI-mix-5-Net has no big gap with SE-Net, in the test error rate curve, InI-mix-5-Net can achieve better results than SE-Net, especially in the last stage of training. The above phenomenon occurs in the backbone of both ResNet and WRN, which shows that the proposed InI-Net has better generalization ability than SE-Net.

On the ImageNet dataset, further improvement of results is also challenging. Figure 3 shows the training log curves of SE-Net and InI-mix-5-Net on ImageNet dataset, with backbones of ResNet-50 and ResNet-101 [7]. It can be seen that compared to SE-Net, the proposed InI-Net has a faster convergence speed on ImageNet, and our InI-Net ultimately can maintain the better accuracy performance than the SE-Net. Another point that can be seen is that SE-Net's fitting capacity on ImageNet is not as good as the InI-Net, which is more evident on smaller backbone networks.

TABLE IV: Computing consumption(FLOPs and Averaged wall-clock time per training epoch) of WRN, SE-WRN and multiple modes of InI-models over 5 runs on CIFAR-100.

Model	Joint	Aggregation	Fold	Dilated	FLOPs	Epoch Time
WRN-22-10 [3]	–	–	–	–	3.828G	64 sec
WRN-28-10 [3]	–	–	–	–	5.243G	112 sec
SE-WRN-16-8 [1]	–	–	–	–	1.548G	28 sec
SE-WRN-22-10 [1]	–	–	–	–	3.839G	76 sec
SE-WRN-28-10 [1]	–	–	–	–	5.257G	125 sec
InI-WRN-16-8-simple-3 (ours)	√	√	–	–	1.549G	29 sec
InI-WRN-16-8-square-1 (ours)	√	–	√	–	1.549G	28 sec
InI-WRN-16-8-square-3 (ours)	√	√	√	–	1.551G	30 sec
InI-WRN-16-8-square-3-d (ours)	√	√	√	√	1.552G	31 sec
InI-WRN-16-8-mix-5 (ours)	√	√	√	–	1.554G	32 sec
InI-WRN-16-8-mix-5-d (ours)	√	√	√	√	1.555G	33 sec
InI-WRN-22-10-simple-3 (ours)	√	√	–	–	3.841G	77 sec
InI-WRN-22-10-square-1 (ours)	√	–	√	–	3.841G	76 sec
InI-WRN-22-10-square-3 (ours)	√	√	√	–	3.843G	78 sec
InI-WRN-22-10-square-3-d (ours)	√	√	√	√	3.844G	79 sec
InI-WRN-22-10-mix-5 (ours)	√	√	√	–	3.846G	81 sec
InI-WRN-22-10-mix-5-d (ours)	√	√	√	√	3.847G	82 sec
InI-WRN-28-10-square-1* (ours)	–	–	√	–	5.257G	125 sec
InI-WRN-28-10-simple-1 (ours)	√	–	–	–	5.258G	125 sec
InI-WRN-28-10-simple-3 (ours)	√	√	–	–	5.259G	126 sec
InI-WRN-28-10-square-1 (ours)	√	–	√	–	5.258G	126 sec
InI-WRN-28-10-square-3 (ours)	√	√	√	–	5.262G	128 sec
InI-WRN-28-10-square-3-d (ours)	√	√	√	√	5.263G	129 sec
InI-WRN-28-10-mix-5 (ours)	√	√	√	–	5.266G	131 sec
InI-WRN-28-10-mix-5-d (ours)	√	√	√	√	5.268G	132 sec

∗Without joint modeling of residual and identity mappings in ResNets.

TABLE V: Computing consumption(FLOPs and Averaged wall-clock time per training epoch) of comparative attentional models on CIFAR-100.

Model	FLOPs	Epoch Time
Two-level Attention [4]	16M	5 sec
Attention-ResNet-452 [5]	1.360G	–
CBAM-ResNet-110 [6]	254.1M	62 sec
CBAM-ResNet-164 [6]	381.8M	95 sec
InI-ResNet-110-square-3-d + spa (ours)	254.7M	61 sec
InI-ResNet-110-square-3-d + spa×4 (ours)	255.1M	62 sec
InI-ResNet-164-mix-5-d + spa (ours)	382.6M	92 sec
InI-ResNet-164-mix-5-d + spa×4 (ours)	383.0M	93 sec

APPENDIX D
COMPARISON RESULT WITH OTHER ATTENTIONAL MODELS

This section supplements the experimental results of the comparison between the proposed InI model and other attention-based models. In the main text, we mainly compared the ordinary channel-wise attention [1], and the combined model of spatial attention and channel-wise attention CBAM [6]. To make the comparison between the InI model and the attention-based method of similar type more adequate, we list their comparison results on the CIFAR dataset in Table VII. We also add pure spatial attention methods Two-level Attention [4], Attention-ResNet [5], and multi-attention combination methods A^2-ResNet [9] and GE-ResNet [8] joined the competition with the InI model as benchmarks. Table VIII lists the experimental results of attention-based models on ImageNet. For datasets CIFAR and ImageNet, all experimental settings

TABLE VI: Computing consumption(FLOPs and Averaged wall-clock time per training epoch) of comparative models on ImageNet.

Model	FLOPs	Epoch Time
Two-level Attention [4]	762M	65 sec
Attention-ResNet-56 [5]	6.300G	–
ResNet-50 [7]	3.858G	1135 sec
ResNet-101 [7]	7.570G	2071 sec
ResNet-152 [7]	11.300G	3254 sec
SE-ResNet-50 [1]	3.860G	1201 sec
SE-ResNet-101 [1]	7.575G	2161 sec
SE-ResNet-152 [1]	11.320G	3403 sec
InI-ResNet-50-mix-5-d (ours)	3.865G	1235 sec
InI-ResNet-101-mix-5-d (ours)	7.581G	2224 sec
CBAM-ResNet-50 [6]	3.864G	–
CBAM-ResNet-101 [6]	7.581G	–
GE-ResNet-50 [8]	3.870G	–
GE-ResNet-101 [8]	7.590G	–
InI-ResNet-50-mix-5-d + spa (ours)	3.873G	1282 sec
InI-ResNet-50-mix-5-d + spa$\times$4 (ours)	3.882G	1335 sec
InI-ResNet-101-mix-5-d + spa (ours)	7.588G	2272 sec
InI-ResNet-101-mix-5-d + spa$\times$4 (ours)	7.593G	2316 sec

TABLE VII: Test error((mean $\pm$ std) %) of InI-models over 5 runs, compare with other attentional models on CIFAR.

Model	Spatial	Channel-wise	CIFAR-10	CIFAR-100
Two-level Attention [4]	$\surd$	–	10.66 $\pm$ 0.28	39.2 $\pm$ 0.31
Attention-ResNet-452 [5]	$\surd$	–	3.90	20.45
SE-ResNet-110 [1]	–	$\surd$	5.65 $\pm$ 0.16	25.79 $\pm$ 0.26
SE-ResNet-164 [1]	–	$\surd$	4.79 $\pm$ 0.15	22.47 $\pm$ 0.19
InI-ResNet-110-square-3-d (ours)	–	$\surd$	5.10 $\pm$ 0.12	24.83 $\pm$ 0.11
InI-ResNet-164-mix-5-d (ours)	–	$\surd$	**4.09 $\pm$ 0.15**	**21.34 $\pm$ 0.10**
CBAM-ResNet-110 [6]	$\surd$	$\surd$	5.52 $\pm$ 0.12	25.49 $\pm$ 0.30
CBAM-ResNet-164 [6]	$\surd$	$\surd$	4.60 $\pm$ 0.18	21.19 $\pm$ 0.13
GE-ResNet-110 [8]	$\surd$	$\surd$	4.93	23.36
GE-ResNet-164 [8]	$\surd$	$\surd$	4.07	20.85
InI-ResNet-110-square-3-d + spa (ours)	$\surd$	$\surd$	4.53 $\pm$ 0.13	22.98 $\pm$ 0.16
InI-ResNet-110-square-3-d + spa$\times$4 (ours)	$\surd$	$\surd$	4.43 $\pm$ 0.18	22.76 $\pm$ 0.20
InI-ResNet-164-mix-5-d + spa (ours)	$\surd$	$\surd$	3.97 $\pm$ 0.17	20.43 $\pm$ 0.15
InI-ResNet-164-mix-5-d + spa$\times$4 (ours)	$\surd$	$\surd$	**3.89 $\pm$ 0.18**	**20.29 $\pm$ 0.21**

and devices information is consistent with the describing in Appendix B and the main text.

In Tables VII and VIII, values "Spatial" and "Channel-wise" indicate the attention-based mechanisms used in the experimental models. It can be seen that no matter using the single attention strategy or the combination of multi-attention on spatial and channel-wise, the proposed InI model can achieve the best performance on both CIFAR-10, CIFAR-100, and ImageNet datasets.

APPENDIX E
COMPARISON RESULT WITH STATE-OF-THE-ART VARIETIES OF THE SQUEEZE-AND-EXCITATION NETWORKS

The proposed method represents a novel strategy of channel relationship modeling. As an improved version of squeeze and exception networks (SE-Net) [1], the proposed method can be widely applicable to almost CNN networks, including other improved models for SE-Net.

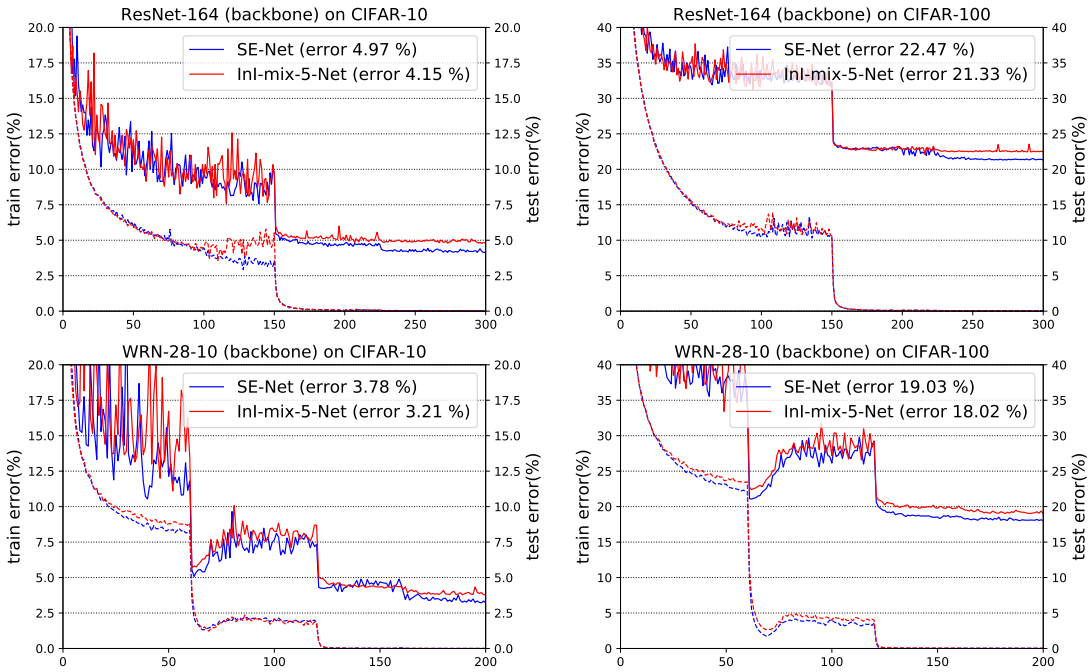

Fig. 2: Training log curves for SE-Network and the proposed InI-mix-5-Network on CIFAR-10 and CIFAR-100 dataset, with backbones of ResNet-164 [7] and WRN-28-10 [3]. Solid lines denote test error (y-axis on the right), dashed lines denote training loss (y-axis on the left).

We further apply Inner-Imaging method to the structure of selective kernel networks (SK-Net) [10]. Specifically, we replace the channel relation coding operation of "Fuse" stage in SK-Net with Inner-Imaging operation. Similarly, we use the same softmax method as SK-Net to select convolution channels with different convolution kernel sizes, after modeling grouped relationship with G-filter.

In addition to directly applying inner imaging module to SK-Net, we also try to choose more convolution kernel sizes, to give full play to the group relationship modeling ability of the InI model for convolution kernels of various sizes. We add 1×1 or 7×7 options to the original 3×3, 5×5 convolution kernels, but keep the total number of convolution channels in each layer unchanged, the number of convolution kernels of various sizes is evenly distributed. In terms of nomenclature, '(3,5)' denotes the original version in SK-Net, '(1,3,5)' represents the version with 1×1 convolution kernel added, '(1,3,5,7)' represents the version with 1×1 and 7×7 convolution kernel added.

Table IX shows the compared results on ImageNet. In addition to SK-Net, we also compared with the ResNeSt [11] which is published recently. It can be seen that the InI method further improve the performance of SK-Net, and the better results are obtained on more diverse convolution kernel combinations, like (1,3,5) and (1,3,5,7). The InI-SK-Net-50 achieves the better results than ResNeSt-50-fast. Although the InI-SK-Net-101 is slightly worse than ResNeSt-101 and ResNeSt-101-fast, it also achieves very competitive performance. Our InI-SK-Net has less floating-point computation than ResNeSt-101, and ResNeSt-101 conducts more training epoches (270 epoches). Also, we can notice that the performance of the InI-SK-Net-101(1,3,5,7) is slightly worse than InI-SK-Net-101(1,3,5), the reason is: in the deeper

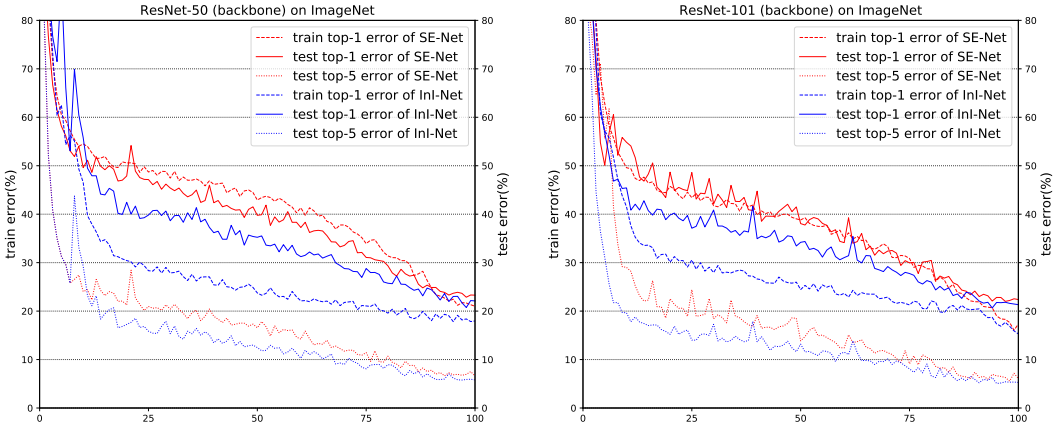

Fig. 3: Training log curves for SE-Network and the proposed InI-mix-5-Network on ImageNet dataset, with backbones of ResNet-50 and ResNet-101 [7]. Solid lines denote test top1 error (y-axis on the right), point-liked dashed lines denote test top5 error (y-axis on the right), line-liked dashed lines denote training loss (y-axis on the left).

TABLE VIII: Single crop error rates (%) of attentional models on ImageNet.

Model	Spatial	Channel-wise	top-1	top-5
Two-level Attention [4]	√	–	40.96	19.11
Attention-ResNet-56 [5]	√	–	21.76	5.90
SE-ResNet-50 [1]	–	√	23.29	6.62
SE-ResNet-101 [1]	–	√	22.38	6.07
InI-ResNet-50-mix-5-d (ours)	–	√	22.10	5.79
InI-ResNet-101-mix-5-d (ours)	–	√	**21.33**	**5.28**
CBAM-ResNet-50 [6]	√	√	22.66	6.31
CBAM-ResNet-101 [6]	√	√	21.51	5.69
A^2-ResNet-50 [9]	√	√	23.00	6.50
GE-ResNet-50 [8]	√	√	21.88	5.80
GE-ResNet-101 [8]	√	√	20.74	5.29
InI-ResNet-50-mix-5-d + spa (ours)	√	√	21.44	5.57
InI-ResNet-50-mix-5-d + spa×4 (ours)	√	√	21.16	5.42
InI-ResNet-101-mix-5-d + spa (ours)	√	√	20.81	5.17
InI-ResNet-101-mix-5-d + spa×4 (ours)	√	√	**20.52**	**5.06**

network, due to the increase of convolution rate, large size convolution kernel is redundant.

APPENDIX F
EXPERIMENTAL RESULTS FOR OBJECT DETECTION TASK ON PASCAL VOC DATASET

In this session, we perform the InI network as a backbone in object detection tasks, to more fully verify the broad application advantages of the proposed InI model.

The setting information of the object detection experiment is as follows:

Dataset: PASCAL-VOC (VOC) [12] dataset is applied to conduct the object detection part of our experiments. We perform evaluation on the VOC 2007 test set and train all the experimental models on the union of VOC 2007 [12] train/val and VOC 2012 [13] train/val ("07 + 12"). In VOC 2007 dataset, there are 9963 images containing 24640 annotated objects, which belong 20 classes, the images in VOC

TABLE IX: Single crop error rates (%) of channel-wise attention variant models on ImageNet.

Model	Params.	FLOPs	top-1
SK-Net-50	27.5M	4.47G	20.79
ResNeSt-50-fast	27.5M	4.34G	19.36
ResNeSt-50	27.5M	5.39G	**18.87**
InI-SK-Net-50(3,5)-mix-5-d	27.8M	4.48G	19.50
InI-SK-Net-50(1,3,5)-mix-5-d	27.9M	4.48G	19.36
InI-SK-Net-50(1,3,5,7)-mix-5-d	27.9M	4.49G	19.33
SK-Net-101	48.9M	8.46G	20.19
ResNeSt-101-fast	48.2M	8.07G	18.03
ResNeSt-101	48.3M	10.2G	**17.73**
InI-SK-Net-101(3,5)-mix-5-d	48.6M	8.47G	18.53
InI-SK-Net-101(1,3,5)-mix-5-d	48.7M	8.48G	18.48
InI-SK-Net-101(1,3,5,7)-mix-5-d	48.7M	8.48G	18.67

2007 has been split into 50% for training/validation and 50% for testing. In VOC 2012 train/val dataset, there are 11530 images containing 27450 annotated objects of 20 classes.

Training: We use experimental models as the backbones to encode the image features and apply single shot multi-box detector (SSD) [14] and Faster-RNN [15] as detectors, which are two popular detection frameworks. The total number of training epochs is 250. We use a weight decay of 0.0005 and a momentum of 0.9. In all the experiments, the size of the input image is fixed to 300 for the simplicity.

Evaluation: We use mAP@0.5 as the evaluation metric of object detection task, variable 0.5 means that a prediction is positive if intersection over union (IoU) ≥ 0.5.

Equipment: The equipment for performing the object detection experiment is the same as described in Appendix B, there are four GTX 1080Ti GPU, two Intel Xeon E5-2620 CPU, and 64G memory.

TABLE X: Object detection mAP(%) with backbone of the proposed InI models and other comparison models on PASCAL VOC 2007 test set.

Detector	Backbone	mAP@.5
SSD [14]	VGG-16 [16]	77.6
	SE-VGG-16 [1]	79.1
	CBAM-VGG-16 [6]	79.3
	InI-VGG-16-mix-5-d (ours)	80.1
	InI-VGG-16-mix-5-d + spa×4 (ours)	**80.6**
	ResNet-50 [7]	78.8
	SE-ResNet-50 [1]	80.1
	CBAM-ResNet-50 [6]	80.2
	InI-ResNet-50-mix-5-d (ours)	80.8
	InI-ResNet-50-mix-5-d + spa×4 (ours)	**81.5**
Faster-RCNN [15]	ResNet-50 [7]	78.1
	SE-ResNet-50 [1]	79.7
	CBAM-ResNet-50 [6]	79.9
	InI-ResNet-50-mix-5-d + spa×4 (ours)	**81.0**

Table X lists the results of the proposed InI model and other compared networks on PASCAL-VOC object detection task. The squeeze and excitation framework and CBAM framework are applied on deep convolutional network VGG-16 [16], ResNet-50 [7], to provide various backbone networks for object detection.

From Table X, it can be seen that the proposed InI model can achieve the best performance when using both SSD and Faster-RCNN as detector. Compare the benchmark models we used, although the outcome of SE-Net is much better than original networks, CBAM network didn't get significant improvement than

SE-Net. This shows that the improvement of modeling efficiency of channels is more critical than spatial attention, and the proposed InI model has achieved better results than the CBAM network when the spatial attention function is not used. Besides, when our InI model adds spatial attention, the object detection performance is further improved, indicating that our InI model can stimulate the potential of the spatial attention mechanism more than direct application.

Figure 2 shows some test examples of the proposed InI model compared with the baseline models, on PASCAL-VOC 2007 object detection task. These examples illustrate the correction of some false detections, missed detections, and misclassifications situations.

REFERENCES

[1] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," *computer vision and pattern recognition*, 2018.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," *european conference on computer vision*, pp. 630–645, 2016.

[3] S. Zagoruyko and N. Komodakis, "Wide residual networks," *British Machine Vision Conference*, pp. 87.1–87.12, 2016.

[4] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang, "The application of two-level attention models in deep convolutional neural network for fine-grained image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 842–850.

[5] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," *computer vision and pattern recognition*, pp. 6450–6458, 2017.

[6] S. Woo, J. Park, J. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," *european conference on computer vision*, pp. 3–19, 2018.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *computer vision and pattern recognition*, pp. 770–778, 2016.

[8] J. Hu, L. Shen, S. Albanie, G. Sun, and A. Vedaldi, "Gather-excite: Exploiting feature context in convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2018, pp. 9401–9411.

[9] Y. Chen, Y. Kalantidis, J. Li, S. Yan, and J. Feng, "Aˆ 2-nets: Double attention networks," in *Advances in Neural Information Processing Systems*, 2018, pp. 352–361.

[10] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019, pp. 510–519.

[11] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, Z. Zhang, H. Lin, Y. Sun, T. He, J. Mueller, R. Manmatha *et al.*, "Resnest: Split-attention networks," *arXiv preprint arXiv:2004.08955*, 2020.

[12] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results," http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[13] M. Everingham, L. Van. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results," http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[14] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *international conference on learning representations*, 2015.

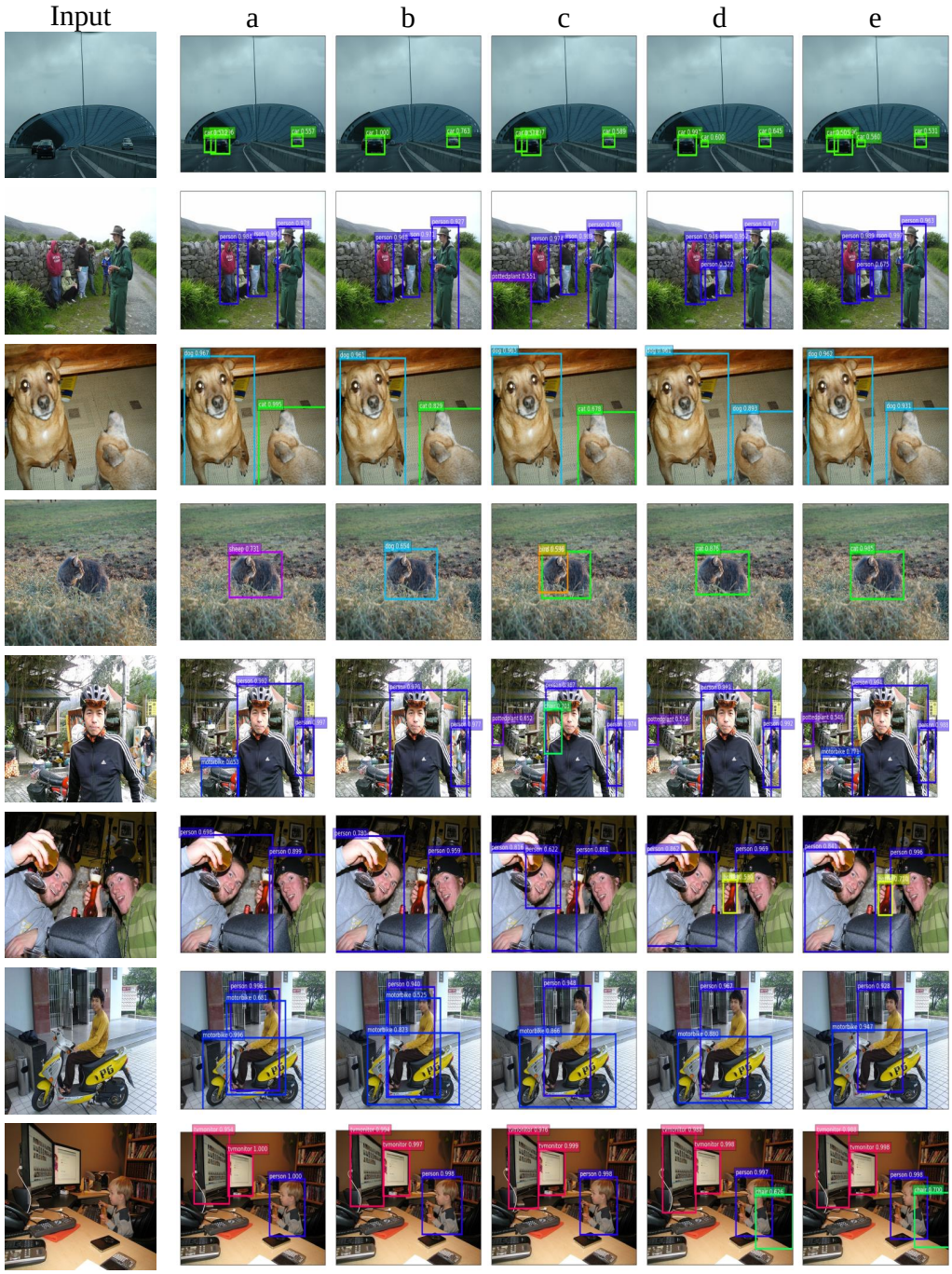

Fig. 4: Examples of VOC-2007 object detection test results of SSD detector with backbones of InI model and comparison models, The leftmost column is the input image, and the remaining columns indicate using different backbones, **a:** ResNet-50 [7]; **b:** SE-ResNet-50 [1]; **c:** CBAM-ResNet-50 [6]; **d:** InI-ResNet-50-mix-5-d (ours); **e:** InI-ResNet-50-mix-5-d + spa×4 (ours).