Deep Knowledge Distillation using Trainable Dense Attention

Bharat Bhusan Sau*[1]
sau.bharatbhusan@gmail.com
Soumya Roy*[2]
meetsoumyaroy@gmail.com
Vinay P. Namboodiri[3]
vpn22@bath.ac.uk
Raghu Sesha Iyengar[1]
bm15resch11003@iith.ac.in

[1] Indian Institute of Technology Hyderabad, India
[2] Indian Institute of Technology Kanpur, India
[3] University of Bath, England

Abstract

Knowledge distillation based deep model compression has been actively pursued in order to obtain improved performance on specified student architectures by distilling knowledge from deeper networks. Among various methods, attention based knowledge distillation has shown great promise on large datasets. However, this approach is limited by hand-designed attention functions such as absolute sum. We address this shortcoming by proposing trainable attention methods that can be used to obtain improved performance while distilling knowledge from teacher to student. We also show that, using dense connections efficiently between attention modules, we can further improve the student's performance. Our approach, when applied to ResNet50(teacher)-MobileNetv1(student) pair on ImageNet dataset, has a reduction of 9.6% in Top-1 error rate over the previous state-of-the-art method.

1 Introduction

A fundamental challenge in deep learning is to make models faster without compromising their accuracy. Neural networks, which obtain state-of-the-art results on real-world datasets, are deep, have billions of parameters and incur significant computational cost [7]. On the other hand, shallow networks are fast but achieve significantly lower accuracy. To reduce this trade-off, several approaches like knowledge distillation [2], model pruning [27] and model quantization [6] have been proposed. In this work, we focus on knowledge distillation because of it's efficacy to increase accuracy of smaller networks. In knowledge distillation, information like soft probabilities [10] or feature maps [24] are extracted from a deep network (or *teacher*) and this knowledge is used to train a shallower network (or *student*).

Attention Transfer [36] is the first knowledge distillation method which has shown promising results on ImageNet. They compute attention maps ($R^{H\times W}$) from different convolution layers ($R^{C\times H\times W}$) of a teacher network to train the student. However, it has a major drawback: it relies on using hand-designed functions (like squared-sum) to extract attention maps. These attention maps are unable to capture the richness of multi-channel feature maps.

*Bharat Bhusan Sau and Soumya Roy have contributed equally.

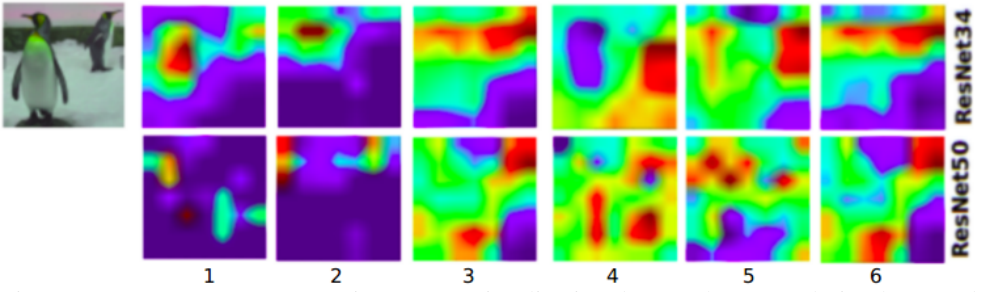

Figure 1: Feature Maps vs Attention Maps: Visualization done on last convolution layer and its corresponding attention encoder of ResNet34 and ResNet50. Feature maps (Column 1-2) are more sparse than dense attention maps (Col. 4-5). However, squared-map [36] (AT) of the feature maps (Col. 3) are almost same as that of dense attention maps (Col. 6).

To address this drawback, we propose to distill trainable multi-channel attention maps, instead of 1-channel hand-crafted attention maps, from teacher. Multi-channel ($R^{K\times H\times W}$) spatial attention maps are learnt from selected convolution layers ($R^{C\times H\times W}$) where $K <= C$. Moreover, using DenseNet [12] as our inspiration, we improve the learnability of attention maps by creating inter-block dense connections between the attention modules (hence the name *dense* attention). It should be noted that our work is not about attention or increasing accuracy of teacher network, but to show that a teacher network could be equipped with simple yet effective attention modules to learn what are the most valuable things to teach a student.

Classical knowledge distillation methods like KD [10] have another important shortcoming on large-scale datasets like ImageNet - student networks are unable to mimic very deep teachers [4, 18]. However, in our approach, we find that, a smaller network like ResNet18 is able to mimic dense spatial attention from a very deep network like ResNet152. This implies that knowledge, learnt in the form of dense spatial attention, from a very deep teacher is useful to train a much smaller student network.

Our method is also unaffected by the *scale* of the dataset, i.e., it performs equally well on both small and large-scale datasets. We also observe that for really small student networks like ShuffleNetv2 [17], our method significantly outperforms existing methods.

To summarize, our contributions are as follows:

1. We propose a way to distill trainable spatial attention maps for knowledge transfer. To the best of our knowledge, this is the first work in the model compression space which transfers trainable attention.

2. We show that, efficiently creating dense connections between attention modules helps to learn more effective and transferable attention maps and this improves accuracy of a student network significantly.

3. Through our experiments on large-scale datasets like ImageNet [25], Places2 [37] and small-scale datasets like CUB [30], 10% of ImageNet, CIFAR100 [15], we demonstrate that, while addition of dense multi-channel attention blocks do not make the teacher network significantly more accurate, the knowledge extracted from them significantly improves the performance of the student network. Additionally, we show that this method can also be used to mimic very deep teacher networks.

2 Related Works

Our work is closely related to knowledge distillation. It is also related to visual attention, as we used spatial attention module for extracting knowledge from teacher.

2.1 Visual Attention

Visual attention helps models to naturally filter information according to their importance. Spatial attention is a type of visual attention which takes feature maps as input and generates masks for each position. Residual-Attention [31] is one of the pioneer works to use attention on CNN, it uses complex trunk-and-mask attention mechanism. SqueezeNet [11] (channel attention), CBAM [32] (channel attention + spatial attention), Harmonious-Attention [16] (spatial attention + channel attention + hard regional attention) simplified it further and increased efficiency of deep network. In our work, we have used only spatial attention module, to extract transferable attentional information.

2.2 Knowledge Distillation

Knowledge distillation is first proposed by Bucilua et al. [3]. Ba and Caruana [2] has shown that squared error between logit values works better than softened probabilities. Fitnets [24] has used intermediate hidden layer outputs along with softened probabilities. Noisy-Teacher [26] has perturbed logit outputs with small amounts of Gaussian noise during teacher-student training, thereby creating the effect of multiple teachers.

However, these methods do not work well for large and challenging datasets like ImageNet. To overcome this drawback, AT [36] has transferred 1-channel hand-designed attention maps (like channel-wise sum of squared activations) using L_2 loss. However, this attention map fails to capture the richness of the underlying multi-channel convolution map. ESKD [4] has observed that a student network trained using KD [10] on ImageNet has lower accuracy than a corresponding network trained from scratch. In order to circumvent this problem, they have stopped knowledge distillation early in the training process and perform gradient descent using only cross-entropy loss for the remaining epochs. [5] has proposed to transform the final layer feature map output by using an autoencoder, which can be trained in an unsupervised fashion. Factor Transfer (FT) [14] has used a paraphrase network to encode teacher knowledge and a translator network to help the student learn from this encoded knowledge. Slimmable neural network [34, 35] proposed to train the slim networks with inplace-distillation method using soft-probabilities as target. [18] performs multi-step knowledge distillation by employing an intermediate-sized network to bridge the gap between a deep teacher network and a student network. SSKD [33] uses self-supervision loss between teacher and student to train the student network. Margin-ReLU [8] has used pre-ReLU feature maps and has filtered them using their margin-ReLU concept. CRD [28] has used contrastive based objective for knowledge transfer from one or more teacher networks to a student network. We have chosen Margin-ReLU [8] as our primary baseline as it outperforms other methods like FT/CRD [14, 28] and it's source code is publicly available.

We address the issues of AT [36] by extracting trainable multi-channel spatial attention (Section 3.1) maps from the teacher network. We also improve these maps by introducing dense connections between attention modules (Section 3.2).

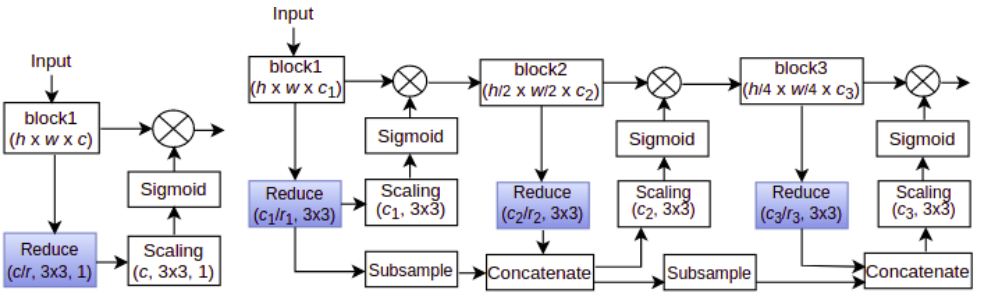

Figure 2: **Left:** Proposed multi-channel spatial attention module. Output of the colored box is multi-channel spatial attention. $\frac{c}{r}$ is number of channels in the attention block.
Right: Dense Multi-channel Spatial Attention: Simplified dense connection between multi-channel spatial attention blocks. Output of colored box is dense spatial attention. r_1, r_2, r_3 are reduction rate of block1, block2 and block3 respectively. This is a 3-block dense connection.

3 Methods

Section 3.1 describes multi-channel spatial attention module. Section 3.2 describes how learning of these spatial attention maps can be improved through dense connections.

3.1 Multi-channel Spatial Attention

Multi-channel spatial attention module is a simplified spatial attention module with multi-channel encoder. Teacher network is modified with this attention module.

Spatial attention module used in the mask branch in Residual-Attention [31] is computationally expensive to learn and hence it is not suitable for our task. Harmonious-Attention [16] has used a more simplified approach but the resulting spatial attention map has just 1 channel. We argue that, in order to encode rich positional information from multi-channel feature maps, it is necessary to learn multi-channel attention maps. This is because the same spatial position can have different importance for different feature maps. Hence we want to construct an attention module which can be easily learned and can produce multi-channel attention encoder.

Our multi-channel attention module is shown in Fig. 2 (Left). At first, 3×3 convolution operation is applied on the input feature map (say, x_i) followed by scaling and sigmoid operations. We can summarize our approach as follows:

$$x_i^{mc} = \text{BatchNorm}(\text{Conv}_{3\times3}(x_i)) \tag{1}$$

Eqn. 1 is the encoder of the attention module. x_i^{mc} is the multi-channel spatial attention map and this map is then scaled and applied on the input feature map x_i using Eqn. 2 and 3.

$$y_i = \text{Sigmoid}(\text{BatchNorm}(\text{Conv}_{3\times3}(\text{Activation}(x_i^{mc})))) \tag{2}$$

Eqn. 2 is the decoder of the attention module and *Activation* is any activation function.

$$x_i = x_i \odot y_i \tag{3}$$

Number of output channels in the encoder depends on the *reduction rate*. For example, if we use $\frac{1}{4}$ as the reduction rate for a given layer which has 256 channels, then the attention encoder will have $\frac{256}{4} = 64$ channels for resulting attention map.

In Section 3.2, we improve on multi-channel spatial attention by incorporating simplified dense connections between attention blocks.

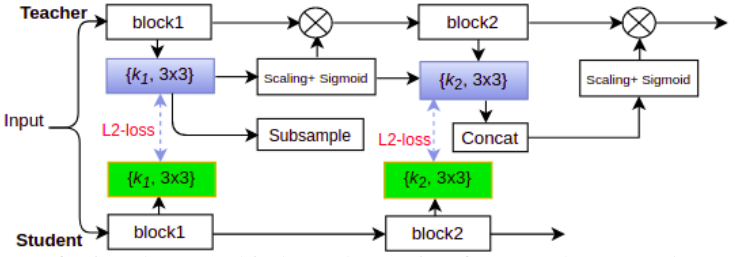

Figure 3: Transferring dense multi-channel attention from teacher to student. Green boxes denote the auxiliary $\text{Conv}_{3\times3}$ layers (encoder part) added to student. k_1 and k_2 are the channels of the multi-channel attention map. The auxiliary layers are used only during training, not during inference.

3.2 Dense Multi-channel Spatial Attention

Attention modules placed on intermediate layers of a deep network suffer from vanishing-gradient problem for the following reasons:

1. Attention module contains sigmoid activation function (see Eqn. 2).
2. Distance from final output layer to intermediate attention modules is large.

While it is necessary to use sigmoid function for normalizing decoder outputs, we can reduce effects of vanishing-gradient by using dense connections [12] between the attention modules, reducing their distance from final output layer.

In DenseNet [12], a given layer in a network is directly connected to all its previous layers. We can apply a similar principle in our case as well. Multi-channel attention maps from previous layers (i.e., encoder outputs of previous attention blocks) are subsampled (in order to have same spatial resolution for concatenation) using a 3×3 convolution as follows:

$$\text{Sub}(x_i^{mc}) = \text{BatchNorm}(\text{Conv}_{3\times3}(\text{Activation}(x_i^{mc})) \tag{4}$$

If $x_1^{mc},...,x_{i-1}^{mc}$ are the multi-channel spatial attention maps from previous blocks (obtained using Eqn. 1), then dense connection for the i^{th} block is created by computing x_i^{datn} as follows:

$$x_i^{datn} = \text{concat}[\text{Sub}(x_1^{mc}),\ \text{Sub}(x_2^{mc}),\ ..,\ x_i^{mc}] \tag{5}$$

To reduce memory consumption, instead of using all previous attention encoder outputs for concatenation, we simplify Eqn. 5 by concatenating x_{i-1}^{datn} and x_i^{mc} as shown in Fig 2 (Right). In this case, x_i^{datn} is computed as follows:

$$x_i^{datn} = \text{concat}[\text{Sub}(x_{i-1}^{datn}),\ x_i^{mc}] \tag{6}$$

Obviously $x_1^{datn} = x_1^{mc}$ as it has no previous input. For all other cases, x_i^{datn} contains information from all previous attention layers. Like Section 3.1, x_i^{datn} is subsequently scaled and applied on the input feature map x_i using Eqn. 2 and 3. This simplified dense connections (Eqn. 6) makes it efficient to train the teacher as well as transferring knowledge to student.

In DenseNet [12], connections between different convolution blocks improves learning of feature maps (or *trunk* branch). Similarly, connections between different multi-channel spatial attention modules (and hence the name *dense* attention) improves learning of attention maps (or *mask* branch) by shortening distance between output layer and attention layer .

3.3 Attention Transfer to Student

Teacher network is modified with dense attention modules, and trained or finetuned accordingly (see supplementary material for details). However, student network is modified with only auxiliary Conv3x3 layers (i.e., the encoder part, see Eqn. 1) during training to mimic the target dense attention maps, as shown in Fig. 3. During inference, these layers are discarded, hence, the time complexity of the student network during inference remains unchanged.

Attention maps from teacher and student must be normalized before passing it to loss function. Two-step normalization is used: mean-subtraction (by mean calculated across the channels), and then with L_2 norm. Standard cross-entropy loss with ground-truth labels and L_2 loss between attention maps of teacher and student are used as the loss function. Let T_i and S_i are normalized attention output vector from i^{th} block of teacher and student respectively. Then the loss function can be expressed as:

$$\text{Cross-entropy loss} + \beta * \sum_i ||T_i - S_i||^2 \tag{7}$$

β, in the above equation, is decayed during training in order to avoid overfitting. See supplementary material for details.

3.4 Summary of trainable Attention Transfer method

1. Decide the mapping between teacher and student intermediate layers.
2. Add dense attention modules to teacher (see Section 3.2).
3. Train teacher with standard hyperparameters.
4. Add auxiliary encoder layers to student network as shown in Fig. 3.
5. Normalize encoder outputs(attention maps) from teacher and student.
6. Train student network using Eq. 7.

4 Experimental Evaluation

We thoroughly evaluate our method (abbreviated as *Dense-ATN*) on ImageNet [25], Places365 [37], 10% subset of ImageNet [25] and CUB [30] which have different grains (fine-grained and regular) and scales (small and large-scale). It should be noted that all these datasets have high resolution images (224×224 or higher) and they represent more real-world scenarios. We also evaluate our method on CIFAR100 [15] dataset which consists tiny images of resolution 32×32.

4.1 Experimental Setup

For baseline methods, the teacher network is not modified. For Multi-channel and Dense-ATN methods, the teacher network is modified with multi-channel spatial attention and dense multi-channel spatial attention modules respectively. Training hyperparameters for teacher and student are given in the supplementary material.

	ResNet34 → ResNet18			
	ResNet34		ResNet18	
Method	Top-1	Top-5	Top-1	Top-5
Baseline(no transfer)	-	-	29.70	10.56
AT[36]	26.69	8.58	29.30	10.04
FT[14]	26.69	8.58	28.57	9.71
ESKD[1]	26.69	8.58	29.16	-
CRD[23]	26.69	8.58	28.83	9.87
Margin-ReLU[8]	26.69	8.58	28.86	9.96
Multi-channel	25.69	8.16	28.43	9.48
Dense-ATN	25.55	8.12	**27.96**	**9.10**

Table 1: Results for ResNet34 (Teacher) → ResNet18 (Student) on ImageNet: Our methods outperform the baselines by at least 2.1% in Top-1 and 6.3% in Top-5 error rates.

	ResNet50 → MobileNet			
	ResNet50		MobileNet	
Method	Top-1	Top-5	Top-1	Top-5
Baseline(no transfer)	-	-	30.78	11.08
AT[36]	23.84	7.14	30.44	10.67
FT[14]	23.84	7.14	30.12	10.50
Margin-ReLU[8]	23.84	7.14	28.75	9.66
CRD[23]	23.84	7.14	29.6	-
Multi-channel	22.77	6.45	26.54	8.45
Dense-ATN	22.73	6.40	**25.93**	**8.14**

Table 2: Results for ResNet50 (Teacher) → MobileNet (Student) on ImageNet: Our methods outperform these baselines by at least 9.8% in Top-1 and 15.7% in Top-5 error rates.

4.2 ImageNet

ImageNet 2012 [25] has 1000 classes and over 1.2 million training images each of size 224×224. The models are evaluated on a validation set of 50,000 images.

Results are shown in Table 1 and Table 2. Using multi-channel attention transfer, we see significant improvements over the baseline methods. With dense multi-channel attention, we obtain even better results and it beats the current state-of-the-art methods by a significant margin. This indicates that dense connections between multi-channel spatial attention modules help the model learn more transferable attentional features. The accuracy of the teacher network with dense attention is slightly higher than the vanilla pre-trained version. However, this difference in accuracy has little contribution towards the significant improvement of our student model over the baseline methods. This is explained in more details in the supplementary material.

4.3 Places365

This is a large-scale dataset on scene-recognition. Places365-Standard [37] has 365 classes and over 1.8 million training images each of size 224×224. Validation set consists of 36,500 samples. In both Places365 and CUB, we consider AT [36] and Margin-ReLU [8] as baselines because of the availability of their source codes. Results are presented in Table 3. Here, the student network trained using our method achieves the same accuracy as the teacher. Had we used a more accurate teacher, the student network could have achieved an even higher accuracy.

	ResNet50 → MobileNet	
	ResNet50	MobileNet
Method	Top-1	Top-1
Baseline(no transfer)	-	46.93
AT[36]	44.40	45.42
Margin-ReLU[8]	44.40	45.09
Multi-channel	44.46	44.81
Dense-ATN	44.43	**44.38**

Table 3: Results on Places365

	ResNet50 → ResNet18	
	ResNet50	ResNet18
Method	Top-1	Top-1
Baseline(no transfer)	-	27.7
AT[36]	23.24	26.35
Margin-ReLU[8]	23.24	25.56
Multi-channel	23.16	24.39
Dense-ATN	23.07	**23.63**

Table 4: Results on CUB

Network	Baseline	AT[36]	CRD[28]	Margin-ReLU[8]	**Dense-ATN**
ResNet18	50.89	48.21	47.23	46.16	**44.56**
MobileNet	47.60	45.42	46.1	44.67	**42.78**
ShuffleNetv2_x1.0	51.59	48.96	49.48	48.23	**46.27**
ShuffleNetv2_x0.5	57.79	56.82	58.47	57.33	**54.07**

Table 5: Results on ImageNet 10% dataset: Top-1 error rate of different student networks. ResNet152 and ResNet152-DATN as teacher has error-rate of 42.92% and 42.10% respectively.

4.4 10% subset of ImageNet

Here, we create a small-scale dataset by taking 10% of training samples per class from ImageNet 2012 [25] dataset, i.e., 128 samples/class are randomly selected for training purpose. Validation set remains as it is. We use ResNet152 as teacher and try out four different students: ResNet18, MobileNet, ShuffleNetv2_x1.0 [17], ShuffleNetv2_x0.5 [17].

Results are shown in Table 5. Our method gives an average 4% relative improvement over the nearest baseline. We also observe that, in case of a tiny network like ShuffleNetv2_x0.5, our method gives a healthy absolute improvement of 3.72 over the base model (student base accuracy), whereas other methods fail to give any significant improvements.

4.5 CUB

CUB-200-2011 [30] is another small-scale dataset, popularly used for fine-grained image classification. It has 200 classes and over 11000 training images, each of size 224×224. We follow the standard practice of initializing the network with ImageNet pretrained weights and then fine-tuning on CUB. Results are presented in Table 4.

4.6 CIFAR-100

CIFAR100 [15] dataset contains 50k training images with 500 samples per class, and 10k testing images with 100 samples per class. The image resolution is 32×32. Results are shown in Table 6. Our method gives significant improvement over state-of-the-art methods, thus showing it's versatile nature. For a fair comparison, we re-run the publicly available code for SSKD [33] without using the KD loss [10]. Hence our results for SSKD are lower than those reported in the original paper.

Teacher Student	wrn40-2 wrn40-1	resnet32×4 resnet8×4	ResNet50 vgg8	resnet32×4 ShuffleNetV1	resnet32×4 ShuffleNetV2	wrn40-2 ShuffleNetV1
Teacher	75.61	79.42	79.34	79.42	79.42	75.61
Teacher-DATN	75.84	79.42	79.76	79.42	79.42	75.84
Student	71.98	72.50	70.36	70.5	71.82	70.5
KD [10]	73.54	73.33	73.81	74.07	74.45	74.83
FitNet [24]	72.24	73.5	70.69	73.59	73.54	73.73
AT [36]	72.77	73.44	71.84	71.73	72.73	73.32
SP [29]	72.43	72.94	73.34	73.48	74.56	74.52
CC [27]	72.21	72.97	70.25	71.14	71.29	71.38
VID [1]	73.3	73.09	70.3	73.38	73.4	73.61
RKD [20]	72.22	71.9	71.5	72.28	73.21	72.21
PKT [21]	73.45	73.64	73.01	74.1	74.69	73.89
AB [9]	72.38	73.17	70.65	73.55	74.31	73.34
FT [14]	71.59	72.86	70.29	71.75	72.5	72.03
NST [13]	72.24	73.3	71.28	74.12	74.68	74.89
SSKD [33]-KD	71.42	69.2	71.2	75.34	76.31	74.6
CRD [28]	74.14	75.51	74.3	75.11	75.65	76.05
Dense-ATN	**74.37**	**76.12**	**75.14**	**77.43**	**77.09**	**76.86**

Table 6: Top-1 accuracy (%) on CIFAR100.

5 Analysis

All our experiments in this section are conducted on the ImageNet dataset. However, the teacher-student combination is different for different observations. More analysis is provided in the supplementary material.

5.1 Can we mimic very deep teachers?

ESKD [4], Teacher-Assistant [18] have shown that if the capacity difference between teacher and student networks is too large, then the student network finds it difficult to mimic the teacher. Both these works have used soft probabilities like KD [10] to train a smaller student. We show that this hypothesis does not hold if we transfer dense attention maps. We used ResNet18 as student, which has very low capacity compared to teachers like ResNet50 (Top-1 error: 23.84) and ResNet152 (Top-1 error: 21.69). From Table 7, we observe that both Top-1 and Top-5 accuracy of ResNet18 (student) keeps increasing as the teacher become more and more deep.

From this result, we hypothesize that, it is easier for a student model to learn positional information from very deep teacher. Dense attention encodes relative importance of various spatial positions, hence, it is not as sparse as feature maps. We can visualize this in Fig 1.

5.2 Effect of kernel size on attention transfer:

Even at its simplest form, dense spatial attention learns very effective and transferable attention maps. In this section, we show that, we can improve this just by increasing complexity of the attention encoder. When we increase the kernel size of the encoder, gradually from 1×1 to 5×5, accuracy of the student also keeps on increasing (see Table 8). Hence, it is possible to learn more transferable attention features by properly increasing complexity of encoder and decoder.

	Deep Teacher → ResNet18(Student)					
	ResNet34		ResNet50		ResNet152	
Method	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
Baseline	29.70	10.56	29.70	10.56	29.70	10.56
AT[36]	29.30	10.04	29.26	10.03	29.54	10.08
Margin-ReLU[3]	28.86	9.96	28.08	9.28	27.95	9.06
Dense-ATN	27.96	9.10	27.06	8.66	**26.94**	**8.59**

Table 7: Deep teacher (ResNet34 / ResNet50 / ResNet152) to ResNet18: Dense-attention is transferable from very deep teacher like ResNet152 to a smaller network like ResNet18.

	ResNet50 → MobileNet			
	ResNet50		MobileNet	
Kernel Size	Top-1	Top-5	Top-1	Top-5
1×1	22.80	6.48	26.33	8.27
3×3	22.73	6.40	25.93	8.14
5×5	22.55	6.12	**25.84**	**8.03**

Table 8: Effect of increasing kernel size in encoder

	ResNet50 → ResNet18			
	ResNet50		ResNet18	
Activation	Top-1	Top-5	Top-1	Top-5
ReLU	22.36	6.21	27.34	8.74
Sigmoid	23.08	6.58	27.17	8.76
Tanh	22.80	6.50	27.10	8.72
Swish	22.73	6.40	**27.06**	**8.66**

Table 9: Effect of Activation function

5.3 Effect of Activation function on attention transfer:

Non-linear activation function is used in the attention block. We have applied it before providing it to the decoder as well as in subsampling (see Eqn. 2 and 4). Here, we analyse the effects of various activation functions in the attention block on its transferability to student. Results are shown in Table 9.

From these experiments, we observe that Swish [23] activation is best suited for learning transferable attention maps. Due to its smooth gating functionality, Swish helps to learn smoother attention maps, which are more transferable. ReLU [19] is less effective here than other non-linear activations. The tanh function is also very effective and quite close to Swish. Hence, we can either use Swish or tanh activation in the attention block. However, due to the universal use of Swish activation, we prefer to use it in our dense attention blocks for all our experiments.

6 Conclusion

In this work, we have proposed to use dense attention mechanism, which is trainable, efficient yet effective. Dense attention can transfer superior knowledge to student via knowledge distillation method and gives state-of-the-art results on real-world datasets. Though we have used our method on image classification tasks, it is actually quite generic in nature. It's use can be extended to other important areas in computer vision like object detection, semantic segmentation, semi-supervised and unsupervised learning.

References

[1] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D. Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[2] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Advances in neural information processing systems*, pages 2654–2662, 2014.

[3] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541. ACM, 2006.

[4] Jang Hyun Cho and Bharath Hariharan. On the efficacy of knowledge distillation, 2019.

[5] Yoojin Choi, Mostafa El-Khamy, and Jungwon Lee. Universal deep neural network compression. *arXiv preprint arXiv:1802.02271*, 2018.

[6] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

[7] 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.

[8] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. *arXiv preprint arXiv:1904.01866*, 2019.

[9] Byeongho Heo, Minsik Lee, Sangdoo Yun, and Jin Young Choi. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *AAAI*, pages 3779–3787, 2019.

[10] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[11] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[12] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[13] Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. *CoRR*, abs/1707.01219, 2017. URL http://arxiv.org/abs/1707.01219.

[14] Jangho Kim, Seonguk Park, and Nojun Kwak. Paraphrasing complex network: Network compression via factor transfer. *CoRR*, abs/1802.04977, 2018. URL http://arxiv.org/abs/1802.04977.

[15] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009.

[16] Wei Li, Xiatian Zhu, and Shaogang Gong. Harmonious attention network for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2285–2294, 2018.

[17] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, pages 116–131, 2018.

[18] Seyed-Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant: Bridging the gap between student and teacher. *CoRR*, abs/1902.03393, 2019. URL http://arxiv.org/abs/1902.03393.

[19] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

[20] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[21] Nikolaos Passalis and Anastasios Tefas. Learning deep representations with probabilistic knowledge transfer. In *Eur. Conf. Comput. Vis.*, 2018.

[22] Baoyun Peng, Xiao Jin, Jiaheng Liu, Dongsheng Li, Yichao Wu, Yu Liu, Shunfeng Zhou, and Zhaoning Zhang. Correlation congruence for knowledge distillation. In *Int. Conf. Comput. Vis.*, 2019.

[23] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[24] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[25] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[26] Bharat Bhusan Sau and Vineeth N Balasubramanian. Deep model compression: Distilling knowledge from noisy teachers. *arXiv preprint arXiv:1610.09650*, 2016.

[27] Suraj Srinivas and R Venkatesh Babu. Data-free parameter pruning for deep neural networks. *arXiv preprint arXiv:1507.06149*, 2015.

[28] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SkgpBJrtvS.

[29] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *Int. Conf. Comput. Vis.*, 2019.

[30] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[31] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2017.

[32] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018.

[33] Guodong Xu, Ziwei Liu, Xiaoxiao Li, and Chen Change Loy. Knowledge distillation meets self-supervision. In *European Conference on Computer Vision (ECCV)*, 2020.

[34] Jiahui Yu and Thomas S Huang. Universally slimmable networks and improved training techniques. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1803–1811, 2019.

[35] Jiahui Yu, Linjie Yang, Ning Xu, Jianchao Yang, and Thomas Huang. Slimmable neural networks. *arXiv preprint arXiv:1812.08928*, 2018.

[36] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

[37] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.