KTAN: Knowledge Transfer Adversarial Network

Peiye Liu
Beijing University of Posts and Telecommunications
Beijing, China
liupeiye@bupt.edu.cn

Wu Liu
JD AI Research
Beijing, China
liuwu1@jd.com

Huadong Ma
Beijing University of Posts and Telecommunications
Beijing, China
mhd@bupt.edu.cn

Zhewei Jiang
Columbia University
New York, US
zj2139@columbia.edu

Mingoo Seok
Columbia University
New York, US
ms4415@columbia.edu

***Abstract*—Knowledge distillation was pioneered to transfer the generalization ability of a large *teacher* deep network to a light-weight *student* network. The student network can retain the high quality of the teacher network, yet exhibiting low computational complexity and storage requirement, which is attractive for deploying a deep convolution neural network on a resource-constrained mobile device. However, most of the existing methods focus on transferring the probability distribution of a softmax layer in a teacher network and neglect the intermediate representations. However, we find that the intermediate representation is critical for a student network to better understand the transferred generalization as compared to the probability distribution only. In this paper, therefore, we propose such a knowledge transfer adversarial network method which holistically considers both intermediate representations and probability distributions of a teacher network. To transfer the knowledge of intermediate representations, we set high-level teacher feature maps as a target, toward which the method trains student feature maps. Furthermore, to support various structures of a student network, we arrange a novel teacher-to-student layer. Finally, the proposed method employs an adversarial learning process. Specifically, it includes a discriminator network to fully exploit the spatial correlation of feature maps during the training process of a student network. The experimental results demonstrate that the proposed method can significantly improve the performance of a student network on two important vision tasks, image classification and object detection.**

***Index Terms*—Knowledge transfer, distillation, adversarial learning**

I. Introduction

The AlexNet [20] and various other deep convolution neural network (CNN) models have demonstrated the state-of-the-art performance in computer vision tasks [1], [5]–[8], [10], [13], [22], [24], [26], [33], [34]. However, the top-performance CNN models generally rely on wide and deep architecture consisting of large number of synapses and neurons [28]. Training and deploying such complex CNN models indeed incur large computation and storage cost, which limits the implementation of a CNN on a resource-limited device.

To tackle the challenges, researchers have attempted techniques to scale the complexity of CNN models. One such technique is network quantization, which tries to convert a full-precision CNN model into a low-precision one [2], [3], [32]. Another technique is network pruning, which attempts to remove the redundant and insignificant connections (weights) [12], [21]. Some practical downsides of these approaches include the memory requirement of storing a large number of indices, and the reliance on special hardware/software accelerators [18]. By contrast, knowledge distillation (KD) aims to train a light-weight model with the knowledge transferred from a trained large model [15], [23] without changing its original architecture or extra storage for indices.

Among the KD techniques, the seminal work by Hinton *et al.* [15] collects the output probability distribution of the softmax layer of a teacher network and use them as target objectives in training a student network. It demonstrates promising results in the classification task. However, as it considers the probability distribution of the output layer as the only knowledge to transfer, it could miss the rich information residing in the other layers and its application can be limited to the classification tasks using the softmax loss function.

Recent studies [27], [30] proposed to exploit intermediate representations as knowledge distillation targets. Specifically, they use the outputs in the convolution layers of a teacher network. The knowledge in feature maps contains not only the feature values and their spatial correlations, which are requisite in various deep CNN models. For transferring this knowledge, they directly align the values of intermediate representations of a teacher and a student network. However, such direct aligning cannot effectively transfer the latent spatial correlation. Given the importance of such information in computer vision tasks, the direct aligning remains a limitation.

In this paper, we propose a new framework titled knowledge transfer adversarial network (KTAN) which works as a general class of the existing KD methods. KTAN has two main operations: 1) knowledge extraction and 2) knowledge learning. In *knowledge extraction*, since the deeper convolution layer extracts more complicated and high dimensional features, we choose the feature maps of teacher's last convolution layer as

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

the shared knowledge. We consider both pixel values as well as region values named spatial information. Since most, if not all, of CNNs contain convolution layers, our framework can be applicable to those CNNs that do not have a softmax layer.

In *knowledge learning*, we adopt the concept of the generative adversarial networks (GAN) and propose to employ three networks in the knowledge transfer framework: 1) a teacher generative network (TGN); 2) a student generative network (SGN); and 3) a discriminator network (DN). The framework operation is illustrated in Figure 1. The TGN first observes a large network model and generates the teacher feature map (TFM) as the knowledge to transfer. In this framework, the TGN includes a teacher-to-student layer to accommodate the topological differences between TFM and student feature map (SFM). Then, for transferring the pixel-to-pixel value, we train the SGN to produce SFM similar to TFM using the MSE loss function. After this, for transferring the region-to-region value, we perform adversarial training by using convolution discriminator network understand those spatial information. The goal of SGN is to maximize the probability of being classified as SFM by the discriminator, while the DN's optimization target is to identity an output as coming from TGN rather than SGN. Therefore, the region-to-region value hidden in the shared knowledge can be delivered from teacher to student network. In this way, the proposed framework is suitable for various computer vision tasks, such as classification and detection, and is evaluated on image classification and object detection benchmarks, demonstrating performance improvements.

To summarize, the contributions of this work are as follows:

- We propose a knowledge transfer adversarial convolution network to provide the light-weight student network with greater intermediate representation knowledge from a deeper/wider teacher model;
- We introduce Teacher-to-Student layer such that intermediate spatial information knowledge can be delivered in an adversarial learning manner to a student network with arbitrary topology;
- We conduct extensive experiments on image classification and object detection tasks, demonstrating the merit of the proposed knowledge transfer adversarial network (KTAN)

II. Related Works

A student model with its knowledge transferred from a well trained teacher model typically achieve better inference results than a small model trained in the conventional way. There are two components in KT: how to extract the shared knowledge from a large teacher model; and how to transfer it to a simpler student model.

Among the early works on KT, [15] concludes the softmax output from a large teacher model contains information on how the model distinguish between classes. However, the softmax output can only perform the classification task; for tasks optimizing different objective function, for example bounding boxes for detection problem, the softmax output cannot be applied due to the low number of classes.

Researchers also attempt to extract intermediate representation from a teacher model. [27] extended the idea of KD and introduced FitNet to compress a network from wide and relatively shallow to thin and deep. In order to learn the generalization of a teacher network, FitNet adopted a squared difference objective function to make the student models mimic the middle layer output of the teacher network. Although the middle layer output in CNN models offer knowledge on a teacher network's generalization, the directly matching learning solution ignored the correlation between models, thus can hardly learn how the teacher model generalize. Later, [30] proposed an idea of Attention Transfer (AT). This method encodes the spatial areas of a teacher network most focused on attention maps, and then transfers the attention maps as shared knowledge to a student network. However, this method's objective function is also a directly matching learning function. Although the attention maps may contain useful spatial information generalized by the teacher network, the directly matching learning process could not sufficiently transfer the spatial generalization of a teacher network to a student network.

III. Knowledge Transfer Adversarial Network

CNN typologies have increasingly become deeper and wider for better inference accuracy performance [13]. On the other hand, to reduce inference time and requirement of computation and storage, cutting down the scale of the model is often needed [16]. Considering this trade-off between resource and accuracy performance, we propose a method to transfer the knowledge from a large teacher network to a small student network. This section is organized into four subsections to introduce our teacher-student knowledge transfer framework. Sec 3.1 introduces the process of extracting the shared knowledge from a large teacher model. Considering the different structure between two models, we design a teacher-to-student regressor layer for matching the size of TFM and SFM. Sec 3.2 presents a normal method for transferring the shared knowledge from a teacher network to student network. In Sec 3.3, in order to make up the deficiency of the directly aligning method, we propose an convolution adversarial network for understanding the spatial information in shared knowledge.

A. Knowledge Extraction

In the field of computer vision, deep CNNs have achieved great success on tasks such as classification [9], localization [29], detection [25] and segmentation [17].

In a CNN model, convolution layers extract features from input or previous layer activations, the features become more complex as the network becomes deeper. Unlike fully connected (FC) layers, each convolution layer are comprised of multiple linear image filters capable of capturing more complex visual features with spacial information. The filter $F \in \mathbb{R}^{k_w \times k_h \times c}$ is convolved with the multiple channel of input images or feature maps $I \in \mathbb{R}^{w \times h \times c}$ from last layer to produce a new image I'.

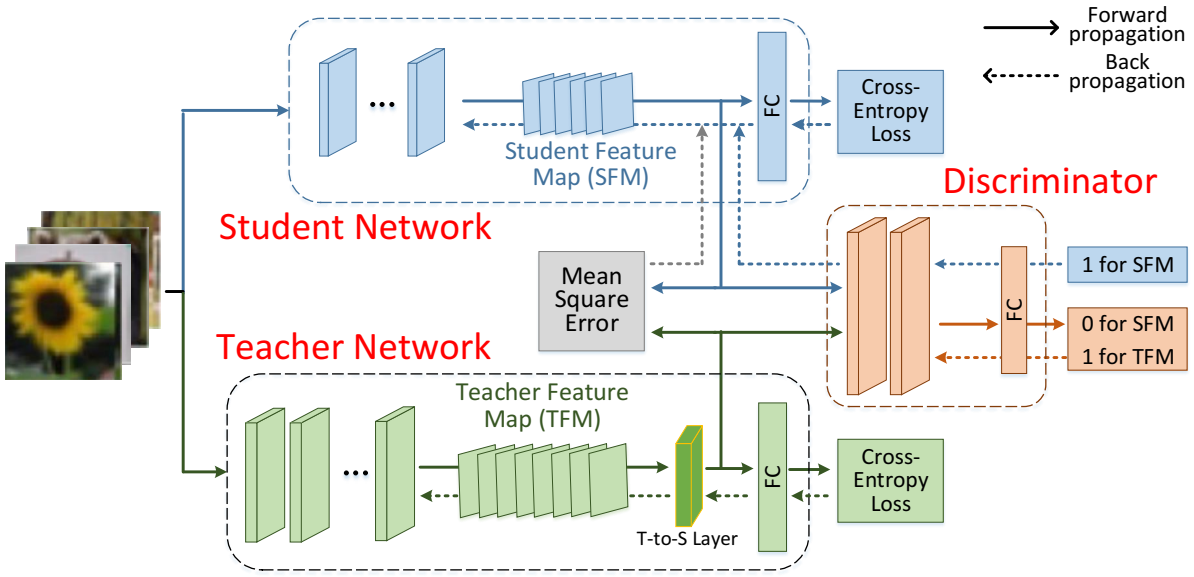


Fig. 1: The architecture of the Knowledge Transfer Adversarial Network on classification task. The green lines represent Feedfoward (FW) and Back propagation (BP) of the teacher network, the blue lines represent FW and BP of the student network, and the orange lines represent FW and BP of the discriminator network. (Best seen in color)

$$I'(x,y) = \sum_{i_x=1-\lceil \frac{k_w}{2} \rceil}^{\lfloor \frac{k_w}{2} \rfloor} \sum_{i_x=1-\lceil \frac{k_h}{2} \rceil}^{\lfloor \frac{k_h}{2} \rfloor} \sum_{i_c=1}^{c} I(x+i_x, y+i_y, i_c) \cdot F(i_x, i_y, i_c) \tag{1}$$

where k_w and k_h represent the kernel width and height of F, w and h represent the size of input images or feature maps, c represents the number of channel of input images.

As shown in [31], a trained shallow convolution layer corresponds to low-level features like edge, angle, and curve. A deeper convolution layer corresponds to more complicated features like circle and rectangle. As the network topology deepens, the model can extract more complex and higher dimensional features. Deep CNN features also better represent the generalizability of the network than shallow ones. Hence, we utilize the feature maps (FM) of the last convolution layer of teacher network as the shared knowledge.

Student Feature Map (SFM) and Teacher Feature Maps (TFM) having non-matching dimensions is an obstacle to the knowledge learning process. To handle the transfer, we add a teacher-to-student regressor to the end of the last convolution layer in the teacher network, whose output matches the size of the SFM. To lower the memory constraint and to retain spatial information in the shared knowledge, we define a convolution regressor layer for resizing the high dimensional feature maps. Let $M_{t,w} \times M_{t,h}$ and C_t be the TFM's spatial size and number of channels. Correspondingly, let $M_{l,w} \times M_{l,h}$ and C_l be the SFM's spatial dimension and number of channels. Given a shared knowledge of size $(C_t, M_{t,w} \times M_{t,h})$, the teacher-to-student regressor sets the output channel as C_l of learning layer and adopts its kernel size of $(M_{t,i}+2\times P-K_i)/S_i+1 = M_{l,i}$, where $i \in \{h, w\}$. The detailed training process of Teacher-to-Student layer is shown in Algorithm 1. We obtain a regressed shared knowledge with the same size of student network.

B. Knowledge Pixel-to-Pixel Learning

After we obtain the shared knowledge from the teacher network, effective transfer to student network is required. An obvious way to transfer the pixel values of shared knowledge is encouraging a student network to simulate the output of a teacher network. In this method, a Mean Square Error (MSE) is adopted as extra objective function to train the student network. Considering the FM of the student network as $m_s \in \mathbb{R}^{w \times h \times c}$ and the FM of the teacher network as $m_t \in \mathbb{R}^{w' \times h' \times c'}$, an extra loss function can be calculated by:

$$L_{MSE} = \frac{1}{r^2 cwh} \sum_{n=1}^{c} \sum_{x=1}^{rw} \sum_{y=1}^{rh} (m_t(x,y,n) - m_s(x,y,n))^2 \tag{2}$$

Where r is the scale ratio, w and h are the width and height of m_s.

As shown in Equation 2, the MSE objective function aims to train a student network by aligning the pixe-to-pixel value of SFM and TFM. However, the shared knowledge also contains correlation spatial information between pixels and channels, which are ignored in this transfer method.

Algorithm 1 Training process of the Teacher-to-Student layer.

Input: The Teacher network model T, divided to convolution parts G and fully connected parts C;

Output: The weight of Teacher-to-Student layer, w_r;

1: Load pre-trained G, C, and initial the Teacher-to-Student layer w_r randomly;
2: **for** number of k step training iterations **do**
3: Input N training samples $\{I^1, ..., I^N\}$ with label $\{y^1, ..., y^N\}$ to the teacher network.
4: Update the w_r by cross-entropy loss $\mathcal{H} = (\mathbf{y}, C(R(G(i))))$
5: **end for**

C. Knowledge Region-to-Region Learning

Knowledge directly learning method transfers the pixel-to-pixel value by aligning the value of shared knowledge from a teacher and a student network. As the shared knowledge is consisted of multiple feature maps, the directly learning method ignores the spatial information in the regional feature. Hence, we attempt to use another convolution network to extract those spatial information in shared knowledge and employ a region-to-region knowledge transferring approach from a teacher to a student network.

Inspired by the Generative Adversarial Network (GAN) [11], we propose the Knowledge Transfer Adversarial network (KTAN) to transfer the region values of shared knowledge from a large teacher network to a small student network, as shown in Figure 1. In our KTAN, in addition to the teacher and student networks, we include a discriminator. The discriminator network receives the shared intermediate representation from a teacher network as well as the corresponding features of a student network. The discriminator uses two convolution layers to process the regional feature of the received feature maps. Similar to the discriminator network in GAN, our discriminator convolution network aims to distinguish whether one piece of knowledge originate from a teacher network or a student network. The student network is trained with the goal to deceive the discriminator convolution network by generating feature maps similar to those from a teacher network. Therefore, due to the end-to-end training process, the region-to-region value hidden in the shared knowledge can be delivered from a teacher to a student network.

In the adversarial training process, to obtain the teacher network's TFM distribution m_t over image data i, we represent data space mapping as $T(i; w_t)$. T refers to the teacher network's final convolution layer. w_t represents its weight. Then, a Teacher-to-Student layer is defined by R with weights w_r. This layer represents a mapping to feature map space $R(m_t; w_r)$, which is a regression result between the teacher and student networks. To transfer the spatial information in TFM, we feed m_s and $R(m_t)$ to discriminator model D to maximize the probability of correctly labeling feature maps as from Teacher-to-Student layer or the student network S. The student network is simultaneously trained to minimize the distinction between $D(m_s)$ and $D(R(m_t))$ through $log(1 - D(S(i)))$ (original objective function) and MSE difference between m_s and $R(m_t)$. In the training stage of the framework, we first pre-train the layer R with image data i to obtain w_r, as presented in Algorithm 1. Then, we devise k steps of optimizing S with the original task's loss function and MSE object function before adversarial optimization. After that, we simultaneously train the S and D playing the following two-player min-max game and a detailed example training process on classification task is presented in Algorithm 2.

Algorithm 2 Training process of KTAN on classification task.

Input: The Student network model, composed of generator S and classifier C; The trained Teacher-to-Student layer R; and Teacher network T;

Output: The improved student model, S and C;

1: Load pre-trained S, C, T, and initialize the discriminator network D randomly; Pre-train the Teacher-to-Student layer R;
2: **for** k iterations of pre-training **do**
3: Input n training samples $I^1, ..., I^n$ with label $y^1, ..., y^n$ to student networks.
4: Update S using cross-entropy loss LCE and the Mean square error loss between m_s and $R(m_t)$

$$L_{CE}(y, C(m_s)) + \beta L_{MSE}(R(m_t), m_s)$$

5: **end for**
6: **for** iterations of adversarial training **do**
7: Input same N training samples $I^1, ..., I^N$ to G and T.
8: Sample N student feature maps m_s from S.
9: Sample N teacher feature maps $R(m_t)$ from R.
10: **Update** network D by L_D, i.e., probability of m_s being labeled a teacher network's generalization:

$$L_D = \sum_{n-1}^{N} -logD(m_s)$$

11: **Update** network S by:

$$\mathcal{H}(\mathbf{y}, C(m_s)) + \alpha L_D(y_{R(m_t)}, m_s) + \beta L_{MSE}(R(m_t)), m_s)$$

12: **Update** network C by:

$$\mathcal{H}(\mathbf{y}, C(m_s))$$

13: **end for**

IV. Experiment

In this section, we perform two computer vision tasks to verify our knowledge transfer adversarial network model: image classification and object detection. Classification is the most popular challenge in computer vision, suitable for verification and comparison; while object detection can demonstrate the generalizability of our model.

A. Image classification

For classification, we evaluate our model on two standard datasets, CIFAR-10 and CIFAR-100. The CIFAR is a popular

image classification benchmark. It contains 50k training images and 10K testing images with 10 and 100 classes, where instances are 32 x 32 color images involving airplanes, cats, human and so on. In the experiments, we utilize random horizontal flips and random crops for data argumentation. For general training, SGD method is used with a mini-batch size of 32 and learning rate of 0.2. For adversarial training, we initialize the learning rate at 10^{-2} and the weight decay at 10^{-4}.

TABLE I: Knowledge transfer results on CIFAR

Method	CIFAR-10(%)	CIFAR-100(%)
Student	93.6	73.1
KD [Hinton et al.]	94.7	76.0
FitNet [Romero et al.]	94.4	75.2
AT [Zagoruyko et al.]	94.5	74.5
DLN	94.4	75.3
KTAN	94.7	77.1
KD+KTAN	**95.0**	**77.6**
Teacher	95.1	78.0

On the CIFAR datasets, we use a very deep residual network Resnet-1001 [14] as the teacher network and a shallow version of Inception network [19] as the student network. We compare our model with several state-of-the-art knowledge transfer methods, including KD [15], FitNet [27] and AT [30].

(1) **Teacher**. A large CNN model (Resnet-1001) trained by the true label objective, which contains 1001 layers.
(2) **Student**. A small CNN model (Inception) trained by the true label objective.
(3) **KD** [15]. This method utilizes the softmax output of a teacher network as shared knowledge. We raise the softmax temperature for teacher network to 4, and set the weight given to the teacher cross-entropy to 0.9, following [15].
(4) **FitNet** [27]. This method utilizes an intermediate representation as shared knowledge and applying a direct knowledge learning process. We transfer the last convolution layer's output to a student network since the simpler student network needs less regularization from the teacher network than a thin and deep one. The weight given to the transfer loss follows [27].
(5) **AT** [30]. This method utilizes only the attention maps as shared knowledge and applies a direct knowledge learning process. Due to the different structure of Teacher and Student, we can only align the attention maps of the last convolution layer in the two networks. The weight given to the transfer loss is 0.05, following the explanation in [30].
(6) **Directly learning network (DLN)**. Our KTAN network without the adversarial learning process.
(7) **KTAN**. This method utilizes the FM of a deep convolution layer and applies an adversarial knowledge learning process. We set the α to 0.6 and β to 0.5.
(8) **KTAN + KD**. We combine our KTAN and KD to transfer both the FM and softmax output to **student**. The adversarial learning process is only applied on FM shared knowledge.

As shown in Table 1, our KTAN model indeed improves the performance of the original student network, which indicates the effectiveness of the intermediate representation based adversarial learning process. Comparing with other methods, our KTAN model is also competitive. In the CIFAR-10 dataset, KTAN archives the best performance among the methods. DLN method slightly outperforms FitNet. The reason is that the regressor layer used in FitNet contain some shared knowledge. For AT method, because of the topological difference between the teacher and student networks, it is hard to map attention maps of all convolution layers from a teacher to a student network. In many cases, only the last convolution layer's attention map is suitable for transferring to a student network, which contains few spatial information than a high-level generalization of a deep convolution layer. Our KTAN method shows better performance than DLN, which indicates the adversarial training process in the KTAN model indeed improves spatial information retention in a student network. In the CIFAR-100 dataset, the large number of classes provide more shared knowledge on the teacher network's probability distribution, the softened softmax output achieves better results on CIFAR-100 than on CIFAR-10. For the same reason, our KTAN model also achieves the best performance on CIFAR-100.

TABLE II: mAP result on Pascal VOC 2007 dataset

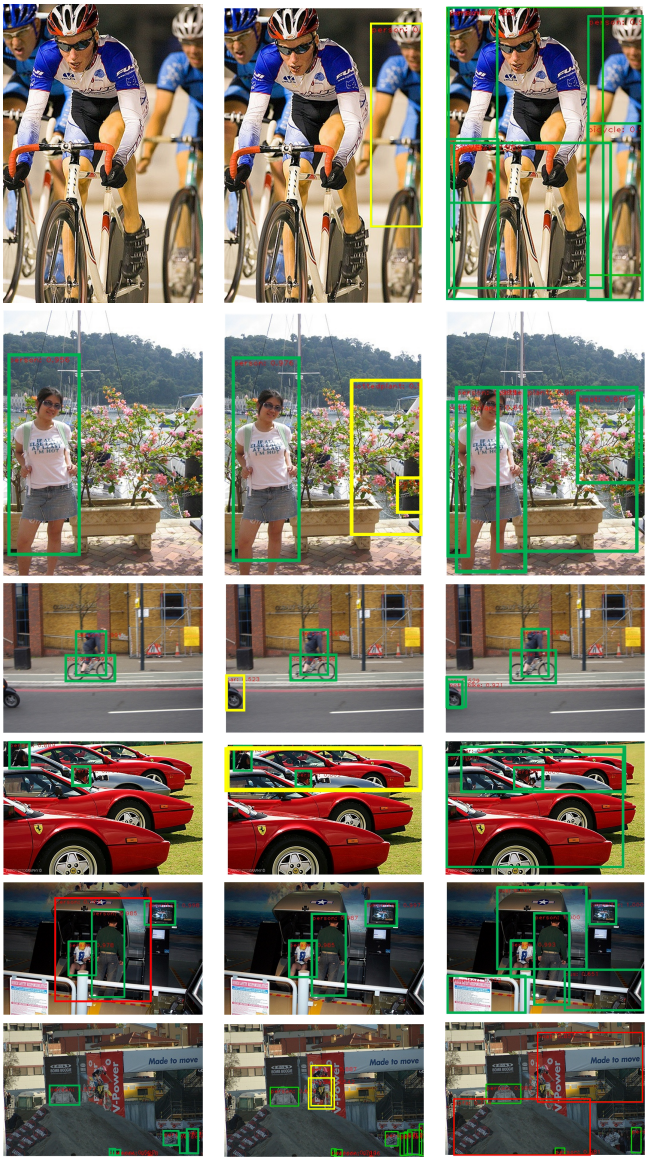
Method	Architecture	mAP
Student	Faster-RCNN (Res50)	70.4
KD	Faster-RCNN (Res50)	70.9
FitNet	Faster-RCNN (Res50)	71.4
DLN	Faster-RCNN (Res50)	71.5
KTAN	Faster-RCNN (Res50)	**73.0**
KTAN+KD	Faster-RCNN (Res50)	**73.4**
Teacher	Faster-RCNN (Res152)	75.4

B. Object detection

Although several works have successfully improved small networks' image classification capability, few explored the performance of their methods on other computer vision tasks, like object detection.

In this section, we perform experiments to compare several methods in object detection, including KD [15], FitNet [27], DLN and our KTAN. We evaluate them on PASCAL VOC 2007 dataset [4]. On this dataset, we select the Faster-RCNN network as the object detection architecture, then use ResNet152 [13] as teacher model and ResNet-50 [13] as student model. Following the settings in [10], we train models on VOC 2007 trainval, and evaluate them on the test set with the mean Average Precision (mAP).

As shown in Table 2, our KTAN model achieves the best performance on PASCAL VOC 2017 dataset. Unlike typical classification tasks, object detection problems contain two optimizing targets, including predicting bounding boxes and classification results for all instances. In the Faster-RCNN method, it applies a region proposal network (RPN) to generate candidate bounding boxes from the output feature map of the last convolution layer. Then, it maps candidate bounding boxes into feature map and classifies each bounding boxes. Therefore, a softmax output of a large model only contains the probability



(a) (b) (c)

Fig. 2: The detection results of a) original student network, b) the KTAN network, and c) teacher network. The yellow bounding boxes in the middle column represent the improved detection results after knowledge transfer and the red bounding box in left column represents the false detection in the original student network. (Best seen in color)

distribution of candidate bounding boxes. Alternatively, our KTAN model extracts the teacher model's last convolution layer's feature maps as shared knowledge, which include more high dimensional information about the dataset. Furthermore, through the adversarial training process, our KTAN method can transfer more spatial information from a large model to a small one than the DLN method.

We also provide some detection results of KTAN on PASCAL VOC 2007 test set. As shown in Figure 2, the first line represents the detection results of original student network, the last line represents teacher networks' results, and the middle line shows the prediction of our KTAN network. The yellow bounding boxes in the middle line represent the improved detection results after the knowledge transfer. Through our KTAN method, a simple Faster-RCNN model can generate better feature maps with the shared knowledge of a large Faster-RCNN model than the original one, allowing it to correctly detect more bounding boxes than the original model. With a better feature map, the improved model can also remove the false positives, as shown in the fourth column. Furthermore, since our KTAN model only learns an intermediate representation from a teacher network, it can eliminate false positives originated from the teacher network's final FC layers.

V. Conclusion

In this paper, we propose a deep feature maps knowledge based adversarial knowledge transfer framework for various computer vision tasks, which is implemented in two sequential processes. For the knowledge extraction operation, we propose to transfer the intermediate representation of teacher to student network. A Teacher-to-Student layer is designed to bridge the structural difference between the teacher and student neural networks. Unlike prior directly matching knowledge learning solution, we devise an adversarial training framework to teach the student network the spatial information in the shared knowledge, since the most valuable information in the shared knowledge is the spatial correlation between feature maps. Our method has experimentally shown improvement in student network's knowledge of the teacher network's generalization.

In the future work, we aim to explore more powerful adversarial frameworks and pursue more applications of our KTAN methods, specifically on computer vision tasks, like video caption, video semantic understanding, etc.

Acknowledgement

This work is partially supported by the Funds for International Cooperation and Exchange of the National Natural Science Foundation of China (No. 61720106007), the Funds for Creative Research Groups of China (No.61921003), the 111 Project (B18008), the Semiconductor Research Corporation (SRC) under task 2810.034, and National Science Foundation (NSF) under No.1919147.

References

[1] Han Cai, Tianyao Chen, Weinan Zhang, Yong Yu, and Jun Wang. Efficient architecture search by network transformation. In *AAAI Conference on Artificial Intelligence*, 2018.

[2] Yu Cheng, Duo Wang, Pan Zhou, and Tao Zhang. A survey of model compression and acceleration for deep neural networks. In *arXiv preprint arXiv:1710.09282*, 2017.

[3] Matthieu Courbariaux, Itay Hubara, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1. In *arXiv preprint arXiv:1602.02830*, 2016.

[4] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. In *International Journal of Computer Vision*, 2010.

[5] Chuang Gan, Naiyan Wang, Yi Yang, Dit-Yan Yeung, and Alex G Hauptmann. Devnet: A deep event network for multimedia event detection and evidence recounting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[6] Chuang Gan, Tianbao Yang, and Boqing Gong. Learning attributes equals multi-source domain generalization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[7] Chuang Gan, Yi Yang, Linchao Zhu, Deli Zhao, and Yueting Zhuang. Recognizing an action using its name: A knowledge-based approach. *International Journal of Computer Vision*, 2016.

[8] Chuang Gan, Ting Yao, Kuiyuan Yang, Yi Yang, and Tao Mei. You lead, we exceed: Labor-free video concept learning by jointly exploiting web videos and images. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[9] Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, 2017.

[10] Ross Girshick. Fast r-cnn. In *IEEE International Conference on Computer Vision*, 2015.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.

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

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, 2016.

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

[16] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. In *arXiv preprint arXiv:1704.04861*, 2017.

[17] Yuan-Ting Hu, Jia-Bin Huang, and Alexander Schwing. Maskrnn: Instance level video object segmentation. In *Advances in Neural Information Processing Systems*, 2017.

[18] Gao Huang, Shichen Liu, Laurens Van der Maaten, and Kilian Q Weinberger. Condensenet: An efficient densenet using learned group convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[19] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, 2015.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012.

[21] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *International Conference on Learning Representations*, 2017.

[22] Peiye Liu, Wu Liu, and Huadong Ma. Weighted sequence loss based spatial-temporal deep learning framework for human body orientation estimation. In *IEEE International Conference on Multimedia and Expo*, 2017.

[23] Ping Luo, Zhenyao Zhu, Ziwei Liu, Xiaogang Wang, Xiaoou Tang, et al. Face model compression by distilling knowledge from neurons. In *AAAI Conference on Artificial Intelligence*, 2016.

[24] Risto Miikkulainen, Jason Liang, Elliot Meyerson, Aditya Rawal, Daniel Fink, Olivier Francon, Bala Raju, Hormoz Shahrzad, Arshak Navruzyan, Nigel Duffy, et al. Evolving deep neural networks. In *Artificial Intelligence in the Age of Neural Networks and Brain Computing*, 2019.

[25] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems*, 2017.

[26] Esteban Real, Sherry Moore, Andrew Selle, Saurabh Saxena, Yutaka Leon Suematsu, Jie Tan, Quoc V Le, and Alexey Kurakin. Large-scale evolution of image classifiers. In *International Conference on Machine Learning-Volume*, 2017.

[27] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *International Conference on Learning Representations*, 2014.

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

[29] Xiu-Shen Wei, Chen-Lin Zhang, Yao Li, Chen-Wei Xie, Jianxin Wu, Chunhua Shen, and Zhi-Hua Zhou. Deep descriptor transforming for image co-localization. In *International Joint Conference on Artificial Intelligence*, 2017.

[30] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *International Conference on Learning Representations*, 2017.

[31] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, 2014.

[32] Aojun Zhou, Anbang Yao, Yiwen Guo, Lin Xu, and Yurong Chen. Incremental network quantization: Towards lossless cnns with low-precision weights. In *International Conference on Learning Representations*, 2017.

[33] Barret Zoph and Quoc V Le. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*, 2016.

[34] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.