CAM-KD: A Class Activation Map Based Approach Towards Knowledge Distillation

Sadman Sakib Alif, Shahriyar Zaman Ridoy, Jannat Sultana

***Abstract*—In recent years, convolution neural network (CNN) based models have seen significant accuracy improvements, but their large number of parameters hinder their widespread adoption. Knowledge Distillation is a popular method of model compression where the knowledge of a teacher model is distilled to a student model using output logits. The baseline approach only captures the high-level knowledge and lacks explainability regarding the exact nature of the knowledge. In our work, we propose a novel approach, CAM-based Knowledge Distillation, where we add a dimension of explainability to the existing distillation framework. We use class activation maps (CAM) as an extra source of knowledge, which captures the label and the structural information during the distillation process. We conduct comprehensive experiments on the CIFAR10 dataset and observe a boost in performance as compared to the baseline distillation method. As far as we are aware, our suggested approach is the first to compute the overall distillation loss by combining the CAM and KD losses.**

***Index Terms*—Knowledge Distillation,Class Activation Maps,Explainable-Knowledge Distillation.**

I. Introduction

DEEP convolution neural networks (CNNs) have demonstrated progressively higher performance in tasks such as image classification [[7], [18], [19]], object detection [[13], [16], [3]] and semantic segmentation [23]. However, in practice, the inference speed of such networks is limited, particularly in the case of edge computing scenarios, due to high computational cost. So, in real-world applications, simpler and compressed models are used. Knowledge Distillation (KD) is a popular method of model compression.

As proposed in [5], a large model is used to train a smaller, more efficient student model without compromising the simpler model's performance. This is achieved by using the teacher's predictions as a source of knowledge and matching the student's predictions. This method has many variations, and they can be broadly categorized into two classes: label-based methods [12] and structure-based methods [11].

In the label-based approach, the softmax output of the teacher serves as an additional source of knowledge for training the student model. However, The output logits are a high-level abstraction of the teacher's knowledge and fail to capture the inner complexities of the deep neural architecture. On the other hand, the structure-based approach uses some intermediate feature maps as a source of knowledge for the student model [19]. Here, the student tries to learn the intermediate features of the teacher. However, this approach has limitations due to the different architectures of the teacher and the student model.

In our work, we propose a new approach, CAM-based Knowledge Distillation.where we use the teacher's class activation maps (CAM) [22] as an additional source to train our student network. Class Activation Map is a heatmap that gives us information regarding which specific region the model looks at while making a decision. Using CAM during training can distill both high and low-level knowledge into the student model. In our training method, we pass a random image from each batch and produce a corresponding class activation map of the teacher and the student. Then, we calculate the mean absolute error between the teacher CAM and the student CAM. The additional loss is integrated with the distillation & student loss in an additive manner.

Our contributions can be summarized as follows:

- We propose a novel approach of transferring knowledge from the teacher using class activation maps (CAM). In our work, we generate CAMs of teacher and student and use it as an additional source of knowledge and combine it with the baseline distillation loss framework.
- Our approach adds an extra dimension of explainability to the knowledge distillation process and helps provide insights into the inner workings of the machine learning models.
- We conduct comprehensive experiments on the CIFAR10 dataset to evaluate our approach and the results show a boost in the performance of the student network trained using our approach

II. Related Work

A. Knowledge Distillation

Devices with low resources, including mobile phones and embedded devices, find it challenging to implement

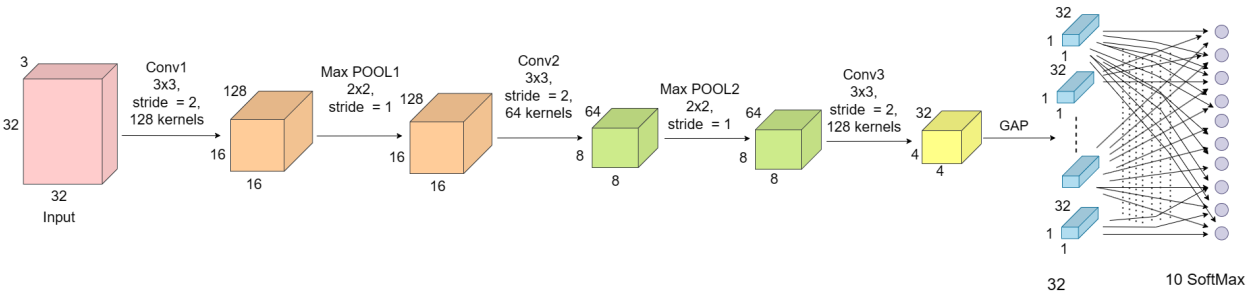

Fig. 1: This figure shows a detailed breakdown of our custom student model, which we have named CAM-Net.

deep learning-based models due to their size. Using knowledge from a larger model (the teacher model) to train a smaller model (the student model) is one potential option. [1]

Knowledge distillation was first conceptualized by Hinton et al. [5], who introduced the idea of moving knowledge from a complex teacher model to a more straightforward student model. The authors showed that by imitating the teacher's soft goal probability in addition to the final predictions, the student model may perform on par with the teacher model.

The knowledge distillation learning methods can be categorized into three primary groups based on whether the teacher model is updated concurrently with the student model: Distillation techniques include offline, online, and self-distillation. [4]

Knowledge distillation was expanded upon by Zagoruyko and Komodakis [8], who introduced the concept of dark knowledge. They underlined how crucial it is to include in training logits, or the unnormalized class scores, from the teacher's model. This method improved the knowledge imparted to the student, which enhanced performance.

In order to transfer intermediate representations from the teacher to the student, Yim et al. [21] devised feature distillation. The student model was able to achieve improved generalization by aligning the internal feature maps, which allowed it to recognize intricate patterns and correlations seen in the teacher's representations.

Ensemble distillation, which involves transferring knowledge from an ensemble of models to a single student model, was proposed by Lopez-Paz et al. [14]. Subsequent studies have expanded on this work, which showed that combining information from several sources enhanced the distilled student model's applicability and robustness.

The idea of self-distillation was first presented by Furlanello et al. [2] This technique involves a student model extracting knowledge from its own predictions. This new method showed how a student model could use the knowledge it had learned during training to compress the model and increase performance.

B. *Class Activation Map(CAM)*

Class Activation Maps (CAMs) are a technique for visualizing and interpreting convolutional neural networks (CNNs) that were first introduced by Zhou et al. [22]. Their work established the use of spatial attention maps to comprehend CNN decision-making processes by highlighting the discriminative regions in an image that contribute most to a given class prediction.

Grad-CAM, an extension of CAM that makes use of the gradient information coming into a CNN's last convolutional layer, was first presented by Selvaraju et al. [17]. Grad-CAM further enhances interpretability by emphasizing the significance of each pixel in the decision-making process and offering more detailed visual explanations.

Researchers have been investigating CAMs as a tool for model interpretation as the need for model explainability has grown. A CAM-based pooling layer was presented by Wang et al. [20] to improve weakly supervised object localization. In addition to increasing the model's accuracy, the method increased its transparency by providing clear visual explanations.

III. Method

In this section, we first analyze the baseline knowledge distillation loss. Then, we discuss how we generate the class activation maps and calculate the cam loss. Finally, we combine the cam loss with the KD loss function.

A. *General Knowledge Distillation*

Given the output logits of the student model (s_i) and the one hot encoded label, cross-entropy loss L_{ce} is used to calculate the student loss. For the distillation part, given pre-softmax outputs of the teacher (t_i) and the student model (s_i), the logits are divided by a temperature (T). Then, Kullback-Leibler divergence [6] is used to calculate the distillation loss. The loss function:

$$L_{distil.} = L_{ce} + \sum_{i=1}^{n} KL(softmax(\frac{t_i}{\tau}), softmax(\frac{s_i}{\tau}))$$

B. *Generating the Class Activation Maps*

In our work, we generate class activation maps (CAM) to enhance the training process of the student network. There are many variations in the process of developing class activation maps [[17], [22], [15]]; for our method, we utilize the original approach mentioned in [22]. In the process suggested in [22], a global average pooling is done on the last convolution layer feature maps and then passed to a fully connected layer, which is then used to produce the outputs. By projecting the output layer weights back onto the convolution feature maps, this straightforward architecture can effectively highlight the significant areas within an image. We chose this method because it only needs one forward pass to generate the class activation maps. In contrast, gradient-based methods require a C number of backpropagation operations to produce class activation maps of C distinct classes. The formula for creating a class activation map, M_c:

$$M_c = \sum_k w_k^c F_k$$

Where, $w_k c$ is the weight corresponding to the weight in the last fully connected layer for class c ; F_k represents globally average pooled activation of unit k at spatial location (x,y). The weights help to emphasize the importance of F_k. A weighted channel wise linear summation is essentially calculated to highlight the significance of a particular unit at spatial location (x,y)

C. *CAM-based Distillation*

The class activation maps function as an additional source of information in our suggested method, enhancing the baseline distillation loss function. In our approach, the last convolution feature maps get upsampled to match the dimensionality of the training image; this ensures both the teacher and student CAMs have matching dimensions. These get flattened and produce vectors of L dimensions, where L = (image height x image width). The CAM loss is calculated as follows:

$$L_{CAM} = \frac{1}{n} \sum_{i=1}^{n} |M_i^t - M_i^s|$$

Where M^t and M^s represent the class activation maps of the teacher and student respectively.

The total loss is calculated by adding the CAM loss to the baseline distillation loss function:

$$L_{total} = L_{distillation} + L_{cam}$$

Algorithm 1 Pseudo code to calculate the CAM loss

```
tw ← last layer weights of teacher
sw ← last layer weights of student
t_conv,t_pred ← last conv. layer output,index of pred.
upsample(t_conv)
p_w ← tw[:,t_pred]
t_h ← dot(t_conv,p_w)
s_conv,s_pred ← last conv. layer output,index of pred.
upsample(s_conv)
p_w ← tw[:,s_pred]
s_h ← dot(s_conv,p_w)
cam_loss ← mae(th,ts)
return cam_loss
```

Model	Train. Param.	Temperature	Accuracy
ResNet50	27.8 M	-	94.8%
CAM-Net	0.096 M	2	66.8%
		3	**67.2%**
		4	64.9%
		5	65.3%
		6	66.6%
		7	66.3%
		8	65.3%
		9	65.4%

TABLE I: Performance of CAM-Net (student network) for different temperature values

IV. Experimental Setup

In this section,we provide information regarding the experimental settings.

A. *Dataset.*

We ran our experiments on the CIFAR10 [10] dataset, a very popular dataset commonly used for benchmarking in computer vision and machine learning. The dataset consists of 60,000 colored images, each with a 32x32 dimensionality corresponding to 10 classes. The dataset is divided into a training set and a testing set with 50,000 and 10,000 photos, respectively.

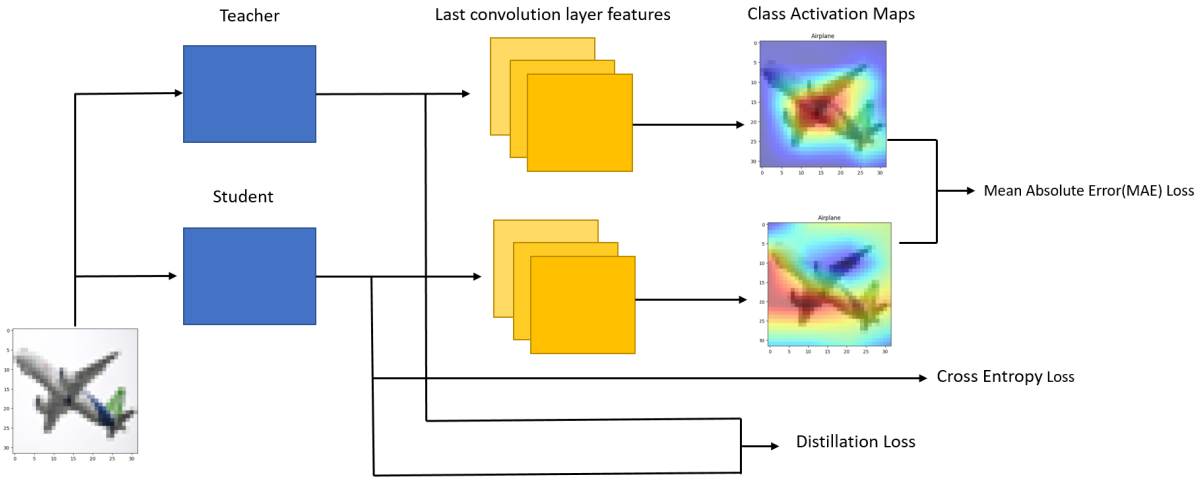

Fig. 2: An overview of our proposed approach

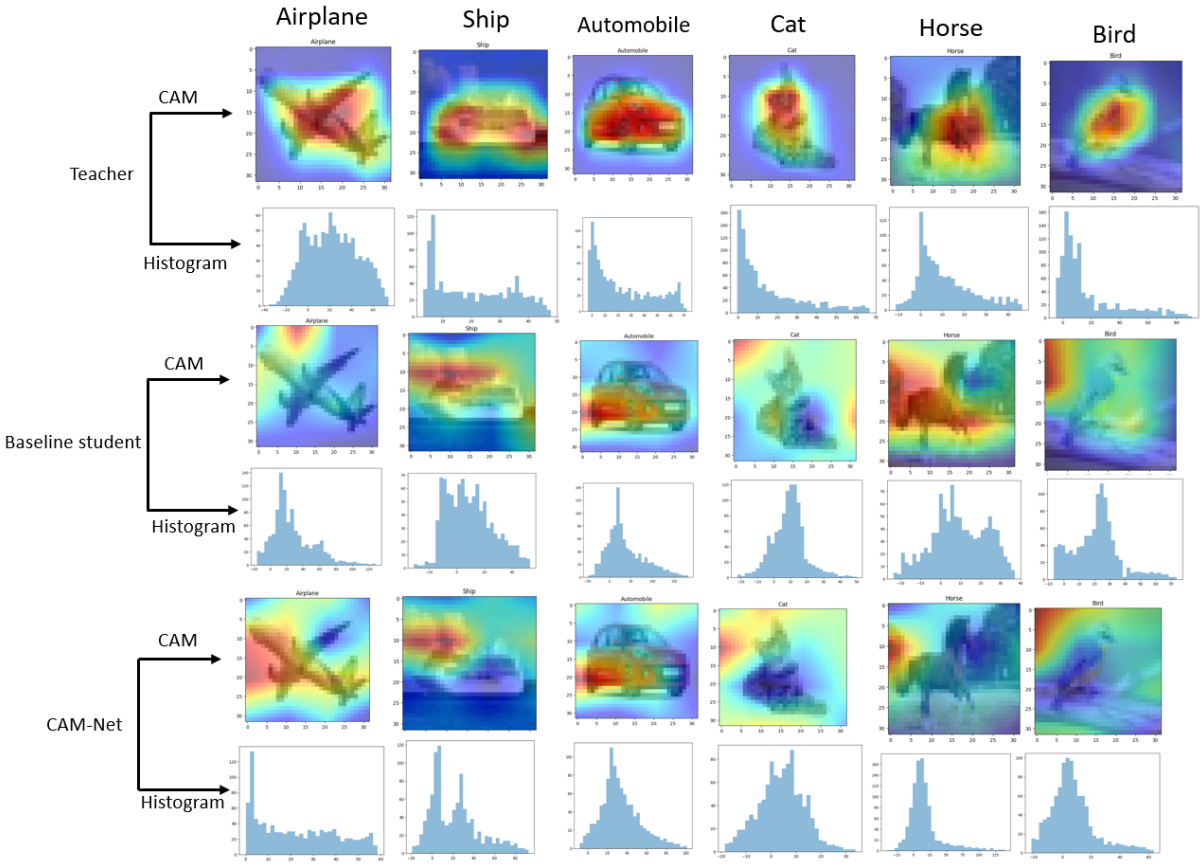

Fig. 3: Heatmaps and histograms produced by the student trained using baseline distillation and the student trained using our modified approach

Teacher	Student	Distillation Loss	CAM Loss	Accuracy
ResNet50	Cam-Net	Baseline Distillation	-	67.2%
		Baseline Distillation + CAM Loss	Mean Absolute Error	**68.3%**
		Only CAM Loss	Mean Absolute Error	64.5%
		Baseline Distillation + CAM Loss	Mean Absolute Error + Softmax	64.1%
		Baseline Distillation + CAM Loss	Mean Absolute Error + Sigmoid	65.7%
		Baseline Distillation + CAM Loss	Mean Absolute Error + Random 5 samples	65.2%

TABLE II: Student model performance results on the CIFAR10 dataset based on variations in the loss function

B. Student-teacher Architectures.

We used the Resnet50 [9] minus the top as the feature extractor for our teacher model. We add a custom top layer to the base teacher model consisting of a convolution layer, a global averaging pooling layer, and a classification layer. The total number of parameters for the teacher model is 27.8 million. For the student, we have designed a custom model, CAM-Net, which consists of 0.096 million parameters. A detailed breakdown of the student architecture is provided in Fig 1.

C. Implementation Details.

We conducted all of our experiments on Kaggle. We used Kaggle's Nvidia Tesla P100 GPU and 16 GB of RAM to train our models. We have used the adam optimizer with a learning rate of 0.001 for all distillation processes and set the temperature value as 3. The student loss function is calculated using cross-entropy loss. In each of the experiments, the student model ran for ten epochs. The experiments are carried out using the TensorFlow framework.

First, we train our student model using the Adam Optimizer and a learning rate 0.001. Next, we conduct knowledge distillation using ResNet-50 as the teacher and our custom student (CAM-Net) as the student.

Our novel distillation method employs variations in how the cam loss gets incorporated into the distillation process. In calculating the loss between the teacher CAM and student CAM, we experiment with different methods like mean absolute error, mean squared error, normalizing the CAM vectors using a softmax process and then calculating the mean absolute error, normalizing the CAM vectors using a sigmoid function and calculating the mean absolute error.

V. Results Analysis

For the evaluation of our training method, we have conducted extensive experiments : (1) Finding the optimum temperature for distillation, (2) Integrating the CAM-based loss with the existing distillation loss, (3) Utilizing different loss functions to evaluate the CAM-based loss (4) Varying the number of samples used to calculate the Class Activation Maps. (5) Comparison of the generated Class Activation Maps

Optimum Temperature. We conducted a temperature search as shown in I, to find the most optimum temperature to extract the maximum knowledge from the existing distillation framework. We concluded a temperature value of 3 exhibited the best results.

Classification with CAM. Table II summarizes the results obtained for image classification using the CIFAR10 dataset. Firstly, we train the student model on the CIFAR10 dataset without knowledge distillation and get an accuracy of **62.6%**. Next, we use baseline knowledge distillation to teach the student model; in this case, the student model accuracy reaches 66.6%, significantly higher than the no KD approach. To experiment with our novel method, we first replace the distillation loss with cam loss and obtain an accuracy of 64.5%, less than the baseline distillation performance. Next, we use the softmax process to normalize the CAM vectors of the teacher and the student and then use mean absolute error to calculate the CAM loss and obtain an accuracy of 64.1%. Next, we normalize the cam vectors using a sigmoid function and get an accuracy of 65.7% , higher than the softmax-based normalization approach. Finally, we use a mean absolute error to calculate the CAM loss, add it with the student and distillation loss, and obtain an accuracy of **68.3%**, higher than the baseline distillation and all other attempted approaches. As this particular approach has shown the most promise,

we have used it to generate the class activation maps for comparison.

Class Activation Maps Comparison. In Fig 3. we can observe the class activation maps of the teacher, the student trained using baseline distillation, and the student trained using cam-based distillation, respectively. The heatmaps provide visual proof that the student trained using our proposed approach can focus (red region) on the object of interest better than the baseline distillation approach.

VI. Conclusion

Our study proposed a novel approach to distilling knowledge using class activation maps. For our experiments, we used ResNet-50 as the teacher and our custom model CAM-Net as the student model, which has 0.096 million trainable parameters. We conducted extensive experiments on the CIFAR10 dataset. In our experiments, we generated class activation maps for students and teachers. We calculated the loss between the corresponding CAMs using techniques such as mean absolute error, mean squared error, softmax normalization, sigmoid normalization, and varying samples. From the results, we found that by calculating the CAM loss using mean absolute error and adding it to the total loss function, our custom model CAM-Net reached an accuracy of **68.3%**, which is better than the baseline and all other attempted approaches. Our proposed method is the first instance where CAM loss and KD loss are combined to calculate the total distillation loss. By incorporating CAMs in the distillation process, we have enhanced the performance of the student and have also added an extra feature of explainability to the distillation process.

In the future, we plan to use this method on datasets of medical images. Furthermore, while our research is limited to the combination of CAM loss and KD loss, we aim to investigate novel KD techniques that improve both interpretability and explainability.

References

[1] Abdolmaged Alkhulaifi, Fahad Alsahli, and Irfan Ahmad. Knowledge distillation in deep learning and its applications. *PeerJ Computer Science*, 7:e474, 2021.

[2] Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *International Conference on Machine Learning*, pages 1607–1616. PMLR, 2018.

[3] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2918–2928, 2021.

[4] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819, 2021.

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

[6] Shuyi Ji, Zizhao Zhang, Shihui Ying, Liejun Wang, Xibin Zhao, and Yue Gao. Kullback–leibler divergence metric learning. *IEEE transactions on cybernetics*, 52(4):2047–2058, 2020.

[7] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 491–507. Springer, 2020.

[8] Nikos Komodakis and Sergey Zagoruyko. Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer. In *ICLR*, 2017.

[9] Brett Koonce and Brett Koonce. Resnet 50. *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pages 63–72, 2021.

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

[11] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2604–2613, 2019.

[12] Yongcheng Liu, Lu Sheng, Jing Shao, Junjie Yan, Shiming Xiang, and Chunhong Pan. Multi-label image classification via knowledge distillation from weakly-supervised detection. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 700–708, 2018.

[13] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[14] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*, 2015.

[15] Mohammed Bany Muhammad and Mohammed Yeasin. Eigen-cam: Class activation map using principal components. In *2020 international joint conference on neural networks (IJCNN)*, pages 1–7. IEEE, 2020.

[16] Rui Qian, Xin Lai, and Xirong Li. 3d object detection for autonomous driving: A survey. *Pattern Recognition*, 130:108796, 2022.

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

[18] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems*, 34:24261–24272, 2021.

[19] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698, 2020.

[20] Jilan Xu, Junlin Hou, Yuejie Zhang, Rui Feng, Rui-Wei Zhao, Tao Zhang, Xuequan Lu, and Shang Gao. Cream: Weakly supervised object localization via class re-activation mapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9437–9446, 2022.

[21] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4133–4141, 2017.

[22] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[23] Hao Zhou, Lu Qi, Hai Huang, Xu Yang, Zhaoliang Wan, and Xianglong Wen. Canet: Co-attention network for rgb-d semantic segmentation. *Pattern Recognition*, 124:108468, 2022.