Enhanced Channel-Wise Attention Based CNN

Aditi Bagora

CS21MTECH14007

Deekshitula Sreeteja

CS21MTECH14009

Harini T K

CS21MTECH14002

Nandini Choudhary

CS21MTECH11009

Samina Haque

CS21MTECH14006

Abstract

With the increase in popularity of attention based mechanism for various computer vision tasks like Image Captioning, Object detection, Segmentation etc, improving the representation power of a CNN is crucial. In traditional attention based mechanisms the input to the attention module is the adjacent convolutional layer only. Our work will be on enhancing the quality of feature maps extracted from a CNN using a channel-wise attention based mechanism that incorporates features from previous convolution layers using skip connections. We aim at comparing the generated feature maps with state-of-art models and report our observations.

1. Introduction

The central building block of convolution neural networks(CNN) is convolution operation which outputs a feature map. A feature map captures representations from different parts of the image. The feature maps can then be used for different purposes. Increasing the quality of feature maps increases the representation power of the network. Various attention based mechanisms are used to enhance the quality of these maps.

Attention is arguably one of the most powerful concepts in the deep learning field nowadays. It is based on a common-sensical intuition that we “attend to” a certain part when processing a large amount of information.

The attention mechanism is used to enhance the assignment of the most informative feature representations while suppressing the less useful ones, thus allowing the model to adaptively focus on the important regions in the context.

Types of attention mechanisms:

1. **Hard attention:** The attention mechanism when applied to some patches or sequences of the data, it can be considered as the Local/Hard attention mechanism.
2. **Soft attention:** The attention when applied in the network is to learn, every patch or sequence of the data can be called a Soft/global attention mechanism.
3. **The Selective Kernel (SK) Attention:** It enhances the expressiveness of the model bypassing the feature

map through two convolution layers of different kernel sizes, followed by the extraction of channel attention.

4. **Channel wise attention:** A Channel Attention Module is a module for channel-based attention in convolutional neural networks. We produce a channel attention map by exploiting the inter-channel relationship of features. As each channel of a feature map is considered as a feature detector, channel attention focuses on 'what' is meaningful given an input image. To compute the channel attention efficiently, we squeeze the spatial dimension of the input feature map.

5. **The Squeeze-and-Excitation Net(SENet) attention :** It explicitly model inter-dependencies in the channel. It can significantly improve the performance of the model by adding only a small number of parameters and computational cost through global average pooling and full connection layers.

6. **Spatial Attention:** A Spatial Attention Module is a module for spatial attention in convolutional neural networks. It generates a spatial attention map by utilizing the inter-spatial relationship of features. Different from the channel attention, the spatial attention focuses on where is an informative part, which is complementary to the channel attention. To compute the spatial attention, we first apply average-pooling and max-pooling operations along the channel axis and concatenate them to generate an efficient feature descriptor.

The attention based methods usually incorporates features only from the last convolution layer. The idea of involving feature maps from previous convolution layers brought in the concept of skip connections.

The skip connections in deep architecture bypass some of the neural network layers and feed the output of one layer as the input to the following levels. It is a standard module and provides an alternative path for the gradient with backpropagation. Skip connections can be performed using summation and concatenation.Skip connections pass information that may be lost due to the depth of layers. In a fully connected network, small object detection is difficult in deep networks so using skip connection would bring clarity in edges, texture, shapes etc. Skip connections also ensure feature reusability.

In CNNS, the different filters will first find spatial features by sliding the kernel across the input image. Hence each channel of the output feature map will comprise of abstract spatial information. A typical Convolutional neural network will not discriminate between these different channels. Since each channel captures low-level feature representation of parts of the image, it is essential to weigh the channels differently. Channel-wise attention incorporates this idea by explicitly modelling inter-dependencies between the channels, effectively re-calibrating channel-wise feature responses.

2. Literature Review

Attention based mechanisms has gained popularity in increasing the performance and quality of CNN features. The paper [2] contains a survey of existing attention based architectures for deep learning models that includes Single channel, Multi-channel model feeding on multi-scale data, Skip-layer model, Bottom-up/ top-down model, Skip-layer model with multi-scale saliency single network. It also contains survey of attention based architecture based on machine vision broadly categorized into attention-based CNNs, CNN transformer pipelines and hybrid transformers. The survey tells that hybrid architectures outperforms the state of art model but are cost effective and resource hungry. The paper also points out that industry footprint of CNN based model is larger. Thus, improvements in attention based models can be beneficial.

A typical CNN architecture would weigh each of its channels equally while creating the output feature maps. The basic idea of this paper [3] is to overcome the inability of previous architectures to adequately model channel-wise feature dependencies. SENets are designed to improve the representational power of a CNN network by explicitly modelling inter-dependencies between the channels of its convolutional features, effectively re-calibrating channel-wise feature responses. Firstly, each feature map is squeezed by aggregating feature maps across their spatial dimensions (H×W) into a vector of size n (channel descriptor) to have global understanding of each channel, where n is the number of convolutional channels. The aggregation is followed by an excitation operation where the channel descriptor is fed through a two-layer neural network outputting a vector of same size. These n values of the vector are used as collection of per-channel modulation weights and are applied on original feature map by scaling each channel based on its importance. SENets performs different roles at different depths throughout the network. At the initial layers of the network, it strengthens the low-level feature representations by exciting the channel-wise features in a class-agnostic fashion. In later layers, squeeze and excitation blocks respond to different inputs in a highly class-specific fashion, effectively strengthening abstract feature representation of the network. Besides the huge performance boost given by the squeeze and excitation blocks, they can be easily added to the existing architectures. SENets have achieved state-of-the-art performance through a wide range of experiments across multiple datasets and tasks. [4] is an enhancement over squeeze-excitation method they proposed a method that balances the feature maps and trunk using controllable parameters.

Convolutional Block Attention Module (CBAM) [6] is a simple effective attention add-on module which infers attention maps along two separate dimensions: channel and spatial. CBAM contains two sequential sub-modules called the Channel Attention Module (CAM) attending the channels (C) and the Spatial Attention Module (SAM) attending the height and width (HxW). Given an image, the channel and spatial attention modules of CBAM focus on 'what' and 'where' respectively. Spatial Attention Module (SAM): Firstly, average-pooling and max-pooling operations are applied across the channels and the output from each of them are concatenated to generate a feature descriptor. This feature descriptor is convolved by a standard convolution layer producing a 2D spatial attention map which encodes where to emphasize or suppress. Channel Attention Module (CAM): The spatial dimension of the input feature map is squeezed using Average Pooling and Max Pooling generating two different spatial context descriptors. Both descriptors are then forwarded to multi-layer perceptron with one hidden layer to produce the channel attention map. The resulting attention maps are then multiplied element-wise with the input feature map for adaptive feature refinement. CBAM empirically confirmed that the design choice of exploiting both channel and spatial features is better than using each independently. CBAM consistently showed great improvements in representation power of networks through extensive experiments with various state-of-the-art models, outperforming all the baselines on different benchmark datasets.

Skip connections also known as Shortcut Connections, were introduced to solve different problems in different architectures. Skip Connections are extra connections between nodes in different layers of a neural network that skip one or more layers of non-linear processing. Basically it feeds the output of one layer to another layer instead of the next one. In Deep CNN we can use Skip Connections for two purposes. First, the information that we had in primary layers can be fed explicitly to later layers using Skip Connections. And second to solve the degradation problem i.e to handle vanishing gradient problem. Deeper a CNN network is, the model faces more problems while updating parameters. This is because the value of error gradient becomes less than one and when considering the product of such two numbers, it tends to 0. So on going in the backward direction there is no update for early layers. Also to handle this problem, we prefer to use skip connections. Skip connections can be used in 2 fundamental ways -

1. **Addition (As in residual architecture ResNet)** Output from the previous layer is added to the layer ahead.

2. **Concatenation (As in densely connected architecture DenseNet)** Concates the output feature maps of previous layers with the next layer.

[1] the paper involves methods to apply fusion of attention from layer of different resolution.
We use this idea to connect different layers for attention extraction through a bridge network. As our model is similar to ResNet and DenseNet, we are trying to take advantage of Skip Connections to combine previous CNN's output with last CNN output and to ease the difficulty of Backpropagation faced in deep CNN models. Our module components are average pooling, concatenation, full connection, gradient back-propagation, which becomes less complicated because of skip connections.

In BA-NET [7] the main idea is to bridge the outputs of the previous convolution layers through skip connections for the generation of channel weights. Bridge attention adds new perspective in traditional neural network architecture which gives potential improvement in performance of other existing channel attention mechanism. BA-Net uses the idea of skip connection to connect different layers for attention extraction through bridge.

The attention mechanism has proven to be very effective in various computer vision and natural language procesing tasks. It enhances the global context by establishing the channel relationship at the global level. A Discriminative Channel Diversification Network [5] for Image Classification gives more attention to the spatially distinguishable channels while taking account of the channel activation.

3. Methodology

3.1. Dataset

For now, we are using CIFAR-10 dataset (Canadian Institute For Advanced Research) for training and testing our model. The CIFAR-10 dataset contains 60,000 32x32 color images in 10 different classes. The 10 different classes represent airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks. There are 6,000 images of each class. We got all results using CIFAR-10[1] dataset.

3.2. Baselines

3.2.1 ResNet 50

The main motivation of the ResNet 50[2] original work was to address the degradation problem in a deep network. Adding more layers to a sufficiently deep neural network would cause saturation in accuracy and then the accuracy degrades.

ResNet consists of residual blocks which adopted the residual learning for every stacked layer. In case of residual learning formulation ,when identity mappings are optimal

[1] https://github.com/Jyouhou/SENet-cifar10

[2] https://github.com/pravinkr/resnet50-cifar10-keras

i.e. g(x)=x, the optimization will drive the weights towards zero of the residual function. The final output can be written as y = f(x,W) + x where W is the weight parameters which are learn during training the model. The operation f + x is performed by skip connection(shortcut path) and element-wise addition. This is a simple block where no additional parameter are added in the skip connection.

ResNet uses skip connection to add the output from an earlier layer to a later layer. This helps in addressing the vanishing gradient problem.

In ResNet-50 the stacked layers in the residual block will always have 1×1, 3×3, and 1×1 convolution layers. The 1×1 convolution first reduces the dimension and then the features are calculated in bottleneck 3×3 layer and then the dimension is again increased in the next 1×1 layer.

The Implementation of ResNet50 using Keras on CIFAR-10 Dataset with 25 epoch on CIFAR-10 dataset gives a test accuracy of 75%.

3.2.2 SE-Net

Since the idea of our implementation revolves around channel-wise attention, Squeeze and excitation network (popular work on channel-wise attention) is chosen to be trained on the CIFAR-10 dataset.

The goal of SE-Nets (Squeeze and Excitation Networks)[3] is to improve the representations produced by a convolution neural network by explicitly modelling the inter-dependencies between the channels of the convolution feature maps. The structure of squeeze and excitation block is depicted in the above figure 4. The input volume X is mapped to the feature map U using any given transformation. The feature map U is squeezed for the aggregation of the spatial information across the spatial dimensions (H×W). On squeezing, we obtain a vector of size n, called the channel descriptor. This channel descriptor provides global understanding of each channel. The aggregation is then followed by the excitation operation where the channel descriptor is fed through a two-layer neural network outputting a vector of same size as that of channel descriptor. These n values of the vector are used as collection of per-channel modulation weights and are applied on original feature map by scaling each channel based on its importance. An implementation of the paper Squeeze-and-Excitation Networks on CIFAR-10 dataset is provided in https://github.com/Jyouhou/SENet-cifar10 and the same is being rendered by the running the code.

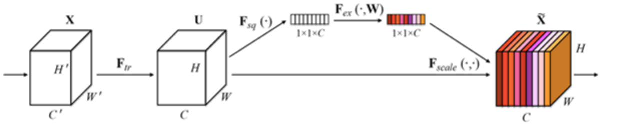


Figure 1. SE-Net

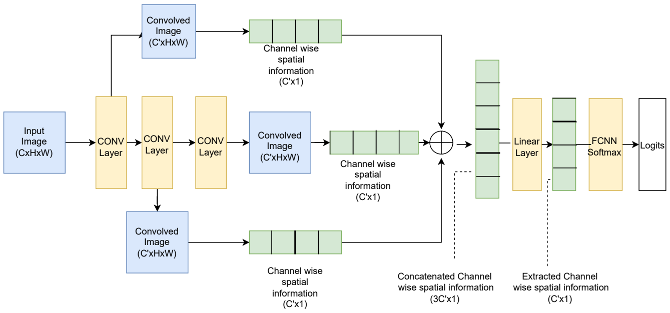
Training Details: Some data augmentation techniques used are padding, cropping of the input image and horizontal flipping. Parameter updates during training are done through Stochastic Gradient descent with learning rate being set to 0.1, momentum set to 0.9 and batch size being set to 128 images. The Squeeze and excitation network is trained over CIFAR-10 dataset for 150 epochs. For each epoch, the train accuracy, test/validation accuracy and loss values are printed. At the end of 150 epochs, the model saves the list of test accuracies obtained in each epoch in a text file. From this the average test accuracy is found to be 87.42%. On comparing SE-Net to the resnet-50 model trained and tested over CIFAR-10 dataset in terms of average test accuracy, we can observe that the accuracy obtained from SE-Net is considerably high. This outperformance is simply because of the ability of SE-Net to capture inter-dependencies between the channels of its convolutional features, effectively re-calibrating channel-wise feature responses.

Baselines	Accuracy
ResNet50	0.75
SENet	0.87

Table 1. Literature baseline results

3.3. Proposed model

We propose an architecture[4] as depicted in Figure 2. The proposed model is an integration of different architectures such as CNNs, skip-connections, channel wise attention mechanism, BA-Net [7].



Input Image (CxHxW)
CONV Layer
CONV Layer
CONV Layer
Convolved Image (C'xHxW)
Convolved Image (C'xHxW)
Convolved Image (C'xHxW)
Channel wise spatial information (C'x1)
Channel wise spatial information (C'x1)
Channel wise spatial information (C'x1)
Linear Layer
FCNN Softmax
Logits
Concatenated Channel wise spatial information (3C'x1)
Extracted Channel wise spatial information (C'x1)

Figure 2. Proposed enhanced channel wise attention based CNN

The input image is passed through various convolutional layers and the convolved image is obtained. Channel-wise

[3] https://github.com/Jyouhou/SENet-cifar10

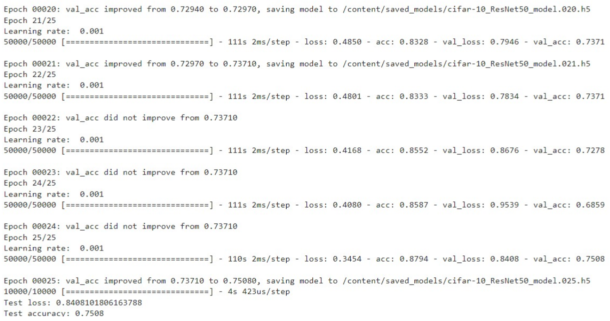
[4] https://colab.research.google.com/drive/1gdCHeRRvPz7BlH-EF5nTbyB4XnDBBYXR?usp=sharing

spatial information is extracted using global average pooling per channel for each convolved image. These channel wise information is concatenated or combined to form a vector that involves all the channel across the entire network. We propose to introduce an additional learnable linear layer that naturally abstracts most important channel wise information. The channel wise spatial information is extracted from each convolution layer($c \times 1$). This information is concatenated ($n \times c \times 1$) and passed through a linear layer to extract information ($c \times 1$).

The extracted information can then be passed to a fully connected network to find the class probability. The loss is calculated and the model paramaters are updated using back propogation.

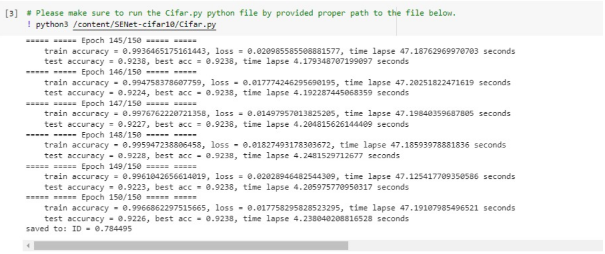
We have currently created an architecture. We will train the model with proper hyper paramaters and test it on CIFAR-10 dataset. We hypothesize that the proposed model will outperform the baselines as it uses information from previous layers and also extracts most important channel wise information that corresponds to better feature extraction and thus giving better classification results.

4. Literature baseline implementation screenshots



```
Epoch 00020: val_acc improved from 0.72940 to 0.72970, saving model to /content/saved_models/cifar-10_ResNet50_model.020.h5
Epoch 21/25
Learning rate:  0.001
50000/50000 [==============================] - 111s 2ms/step - loss: 0.4850 - acc: 0.8328 - val_loss: 0.7946 - val_acc: 0.7371

Epoch 00021: val_acc improved from 0.72970 to 0.73710, saving model to /content/saved_models/cifar-10_ResNet50_model.021.h5
Epoch 22/25
Learning rate:  0.001
50000/50000 [==============================] - 111s 2ms/step - loss: 0.4801 - acc: 0.8333 - val_loss: 0.7834 - val_acc: 0.7371

Epoch 00022: val_acc did not improve from 0.73710
Epoch 23/25
Learning rate:  0.001
50000/50000 [==============================] - 111s 2ms/step - loss: 0.4168 - acc: 0.8552 - val_loss: 0.8676 - val_acc: 0.7278

Epoch 00023: val_acc did not improve from 0.73710
Epoch 24/25
Learning rate:  0.001
50000/50000 [==============================] - 111s 2ms/step - loss: 0.4080 - acc: 0.8587 - val_loss: 0.9539 - val_acc: 0.6859

Epoch 00024: val_acc did not improve from 0.73710
Epoch 25/25
Learning rate:  0.001
50000/50000 [==============================] - 110s 2ms/step - loss: 0.3454 - acc: 0.8794 - val_loss: 0.8408 - val_acc: 0.7508

Epoch 00025: val_acc improved from 0.73710 to 0.75080, saving model to /content/saved_models/cifar-10_ResNet50_model.025.h5
10000/10000 [==============================] - 4s 423us/step
Test loss: 0.8408101806163788
Test accuracy: 0.7508
```

Figure 3. Resnet results on CIFAR10



```
[3] # Please make sure to run the Cifar.py python file by provided proper path to the file below.
    ! python3 /content/SENet-cifar10/Cifar.py
    ===== ===== Epoch 145/150 ===== =====
        train accuracy = 0.9936465175161443, loss = 0.020985585508881577, time lapse 47.18762969970703 seconds
        test accuracy = 0.9238, best acc = 0.9238, time lapse 4.179348707199097 seconds
    ===== ===== Epoch 146/150 ===== =====
        train accuracy = 0.994758378607759, loss = 0.017774246295690195, time lapse 47.20251822471619 seconds
        test accuracy = 0.9224, best acc = 0.9238, time lapse 4.192287445068359 seconds
    ===== ===== Epoch 147/150 ===== =====
        train accuracy = 0.9976762220721358, loss = 0.01497957013825205, time lapse 47.19840359687805 seconds
        test accuracy = 0.9227, best acc = 0.9238, time lapse 4.204815626144409 seconds
    ===== ===== Epoch 148/150 ===== =====
        train accuracy = 0.995947238806458, loss = 0.01827493178303672, time lapse 47.18593978881836 seconds
        test accuracy = 0.9228, best acc = 0.9238, time lapse 4.2481529712677 seconds
    ===== ===== Epoch 149/150 ===== =====
        train accuracy = 0.9961042656614019, loss = 0.02028946482544309, time lapse 47.125417709350586 seconds
        test accuracy = 0.9223, best acc = 0.9238, time lapse 4.205975770950317 seconds
    ===== ===== Epoch 150/150 ===== =====
        train accuracy = 0.9966862297515665, loss = 0.017758295828523295, time lapse 47.19107985496521 seconds
        test accuracy = 0.9226, best acc = 0.9238, time lapse 4.238040208816528 seconds
    saved to: ID = 0.784495
```

Figure 4. SE-Net results on CIFAR10

References

[1] Yimian Dai, Fabian Gieseke, Stefan Oehmcke, Yiquan Wu, and Kobus Barnard. Attentional feature fusion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3560–3569, 2021. 3

[2] Abdul Mueed Hafiz, Shabir Ahmad Parah, and Rouf Ul Alam Bhat. Attention mechanisms and deep learning for machine vision: A survey of the state of the art. *arXiv preprint arXiv:2106.07550*, 2021. 2

[3] 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. 2

[4] Chunjie Luo, Jianfeng Zhan, Tianshu Hao, Lei Wang, and Wanling Gao. Shift-and-balance attention. *arXiv preprint arXiv:2103.13080*, 2021. 2

[5] Krushi Patel and Guanghui Wang. A discriminative channel diversification network for image classification. *Pattern Recognition Letters*, 153:176–182, 2022. 3

[6] 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. 3

[7] Yue Zhao, Junzhou Chen, Zirui Zhang, and Ronghui Zhang. Ba-net: Bridge attention for deep convolutional neural networks. *arXiv preprint arXiv:2112.04150*, 2021. 3, 4