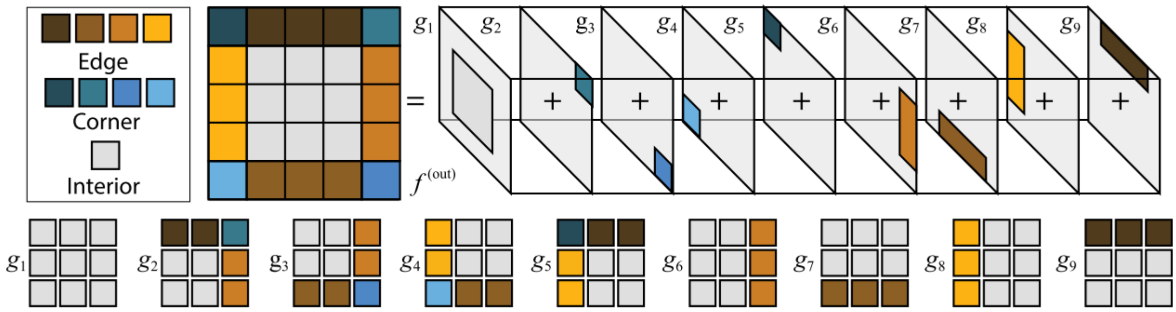
Convolution on Edges in Deep Learning



Background

In computer vision, Convolutional Neural Networks (CNNs) are vital for tasks like image segmentation and disparity estimation. Convolution operation includes different types of padding, including zero padding, same padding, and reflex padding, which are critical for extracting the feature information on the boundary. The accuracy of boundary and edge information in images, particularly at the top, bottom, left, and rightmost columns, holds paramount importance in various applications within computer vision and image processing. Correct boundary information is essential for accurately identifying upcoming objects in different applications (i.e., Video Object Detection, Image Segmentation, etc.). Errors at the boundaries can lead to missed or false detections, impacting the system's reliability.

Problem Specification

When employing small convolution kernels, such as 2x2 or 3x3, in neural networks, the need for extensive padding along the image borders is minimized. This is advantageous as it helps preserve the original spatial dimensions of the feature maps and retains more of the core information within the image. However, the scenario shifts when larger convolution kernels (such as 5x5, 7x7, etc.) come into play. Using large kernels can necessitate significant padding, which can introduce certain challenges. Excessive padding can lead to increased computational costs during both training and inference. Padding can also cause a loss of locality in the convolution operation, making preserving local relationships between features in the input data is challenging. This can hinder the network's ability to capture local patterns effectively.

Suggested Method

A literature study on boundary convolution and padding types will be conducted. Based on analysis, a novel padding type or convolutional filter will be proposed. The first option is that we can train a deep learning architecture to apply padding. Although it will increase the computational complexity, model performance/accuracy is our primary focus. The second option is to propose an adaptive convolution kernel that does not require padding in the feature extraction while preserving the spatial resolution. This will also help reduce the computational complexity of the deep learning architectures.

Difficulty Level

This project has a <u>HIGH</u> difficulty level. The student will run the publicly available Python source code of any chosen state-of-the-art architecture and analyze/visualize the effect of different padding on the features map. The student will also analyze the computational cost of the different convolutional kernels combined with padding techniques. At the end of this project, the student will be able to demonstrate the importance of boundary convolution and the trade-off between computational cost (i.e., # parameters and model file size) and model performance (i.e., output accuracy).

Pre-requisite

The student should have basic knowledge of deep Learning and understand Python coding with TensorFlow libraries.

Relevant Articles

[1] Innamorati, Carlo, et al. "Learning on the edge: Investigating boundary filters in cnns." *International Journal of Computer Vision* 128.4 (2020): 773-782.

[2] Alsallakh, Bilal, et al. "Mind the Pad--CNNs can Develop Blind Spots." *arXiv preprint arXiv:2010.02178* (2020).

[3] Islam, Md Amirul, et al. "Position, padding and predictions: A deeper look at position information in cnns." *arXiv preprint arXiv:2101.12322* (2021).

[4] Alguacil, Antonio, et al. "Effects of boundary conditions in fully convolutional networks for learning spatio-temporal dynamics." *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, Cham, 2021.

[5] Islam, Md Amirul, et al. "Boundary Effects in CNNs: Feature or Bug?." (2020).

Useful Tools

i. Google CoLaboratory for running Python code and training deep learning model: https://colab.research.google.com/

ii. Recent Image segmentation architectures: https://paperswithcode.com/task/semantic-segmentation

iii. Image segmentation databases: https://paperswithcode.com/datasets?task=semantic-segmentation

Project proposed by Ali Hassan (ali.hassan@miun.se)