A Generalized Multimodal Convolutional Neural Network for Improved Input Attribution in Classification Tasks

VDK CNNet: A Variable-Density Convolutional Kernel for Improved Input Attribution in Classification Tasks

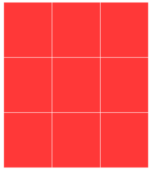
Presented by:
M Shifat Hossain
AI and Emerging Computing Lab
University of Central Florida

Problem Statement

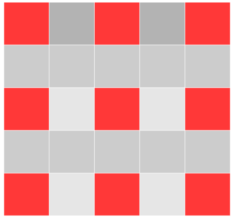
- Kernel size in CNNs play a huge role in generalizing objects in an image [1].
- Larger features are often hard for a small kernel of convolutional layer.
  - Later stage conv layers rely on previous stage conv layers.
  - This requires the model to learn the required features at different depths of a CNN model.
- Usage of smaller kernel requires deeper networks to adequately detect larger spatial features.
- Usage of larger kernels can solve these problems.
- Using larger kernels in conv layers require higher memory and computing resources.

Proposed Method

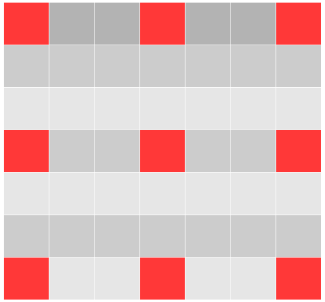
- Variable density conv layer has the structure as shown.
- The red cells have non-zero (trainable) values.
- The gray cells have zero (non-trainable) values.
- The non-zero density of the proposed conv kernels starts from fully dense to sparse conv kernels.



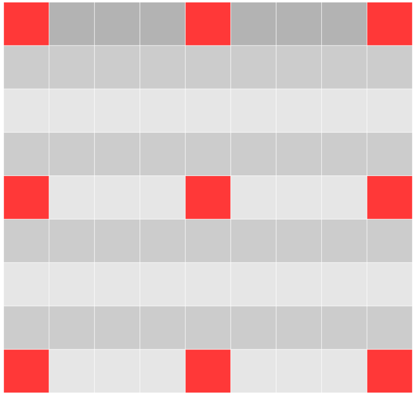
Conventional 3x3 conv kernel



Low-density 5x5 conv kernel



Low-density 7x7 conv kernel



Sparse 9x9 conv kernel

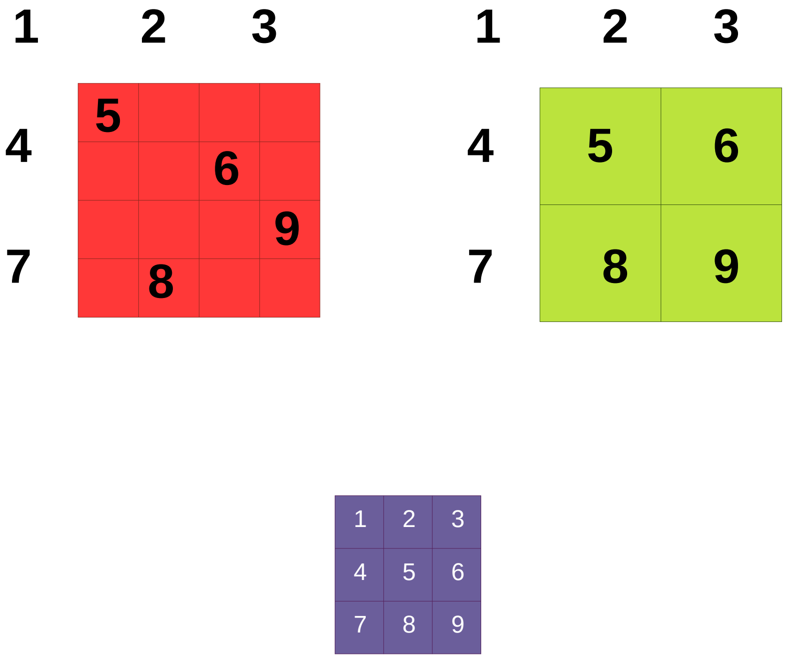
But, if there are **large kernels** in conv layers, wouldn’t that take the **same memory and computational resources?**

Yes…
But, there is a **solution**...

Proposed Method

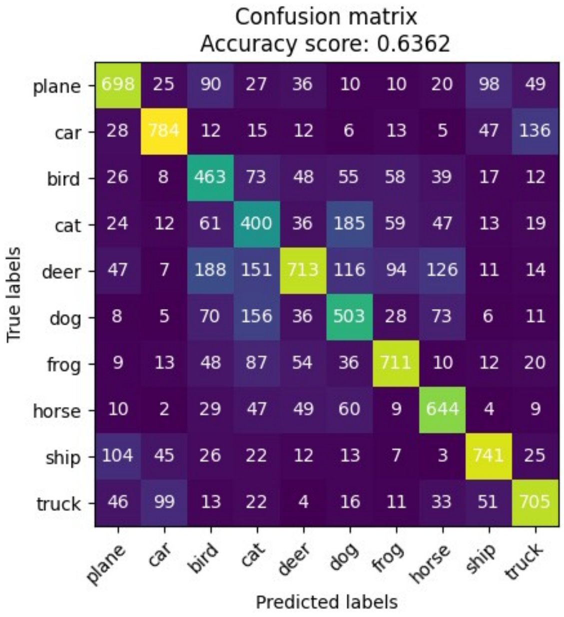
Solution

- The trick is to downscale the input image keeping the kernel size small.
- If an input image is downsampled using a pooling function, the conv operation on that image is equivalent to perform conv operation with low-density kernel on the full scale image.
  - Let, the green colored pixels are the downsampled image using pool operator.
  - Red image has the full resolution (4x4).
  - Let’s run a 3x3 conv kernel (purple) on the downsampled (green) image. The numbers on the image pixels represent kernel cells.
- In this study, we implement a three low-density conv operation layers to evaluate the model.
- Also compared to a baseline 3 3x3 conv layer CNN model.
- Dataset used: CIFAR10

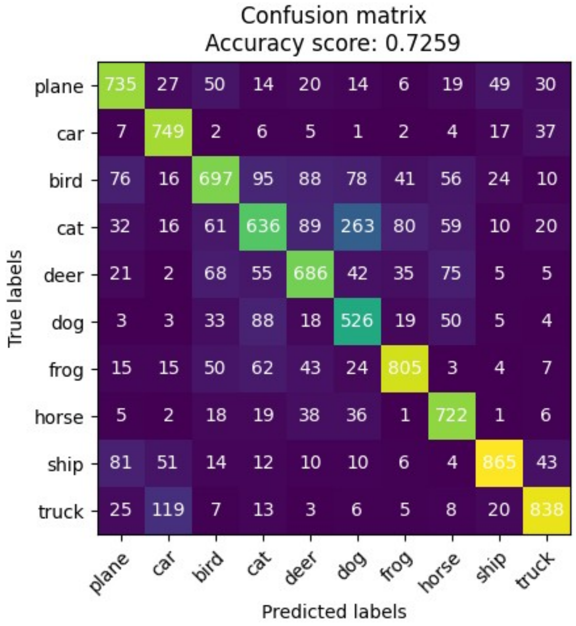


Results

Classification Accuracy - CIFAR10



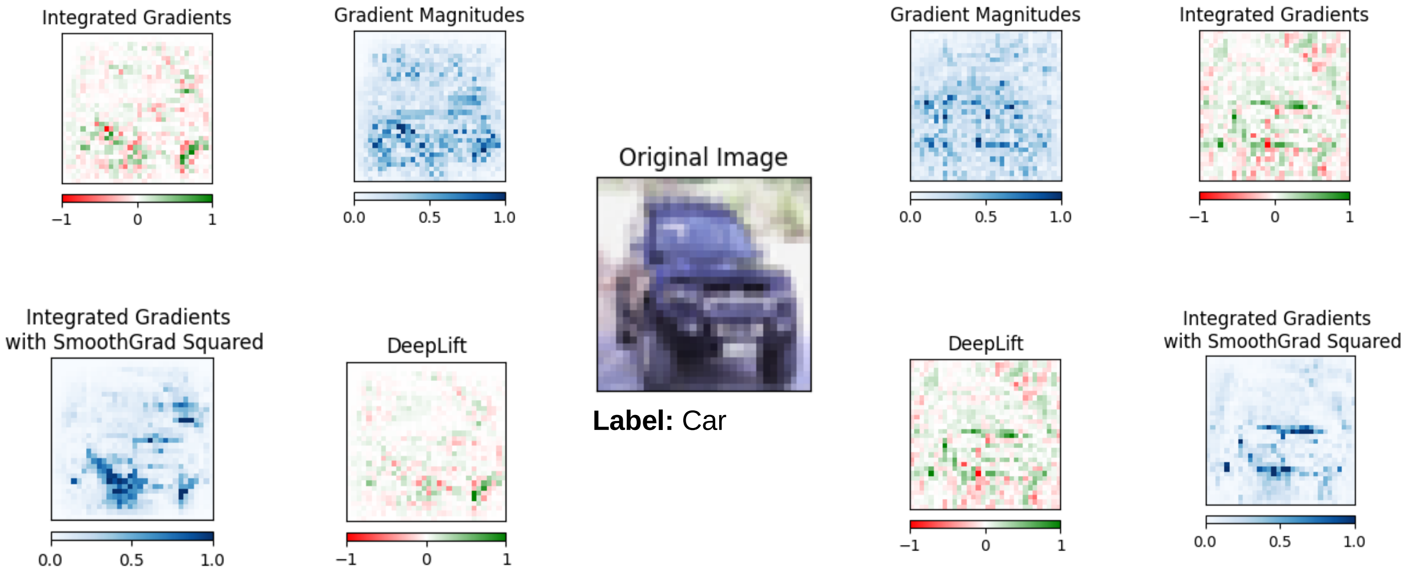
Baseline CNN Model



VDK CNNet Model

Results

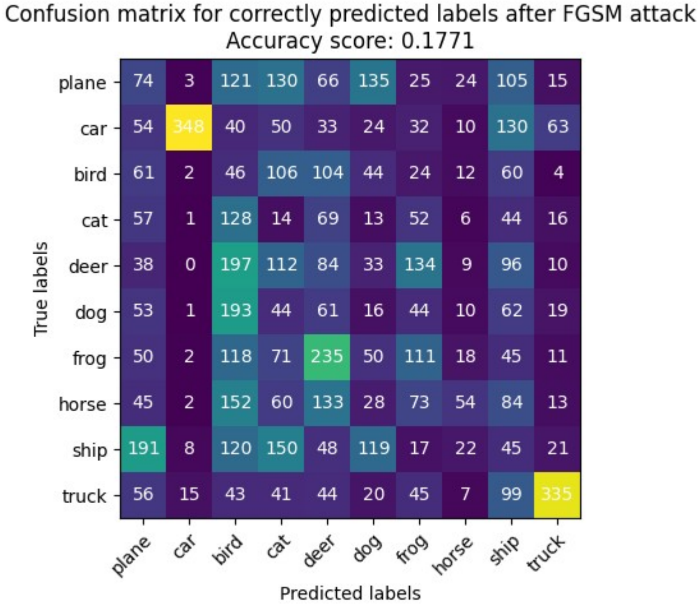
Attribution Analysis



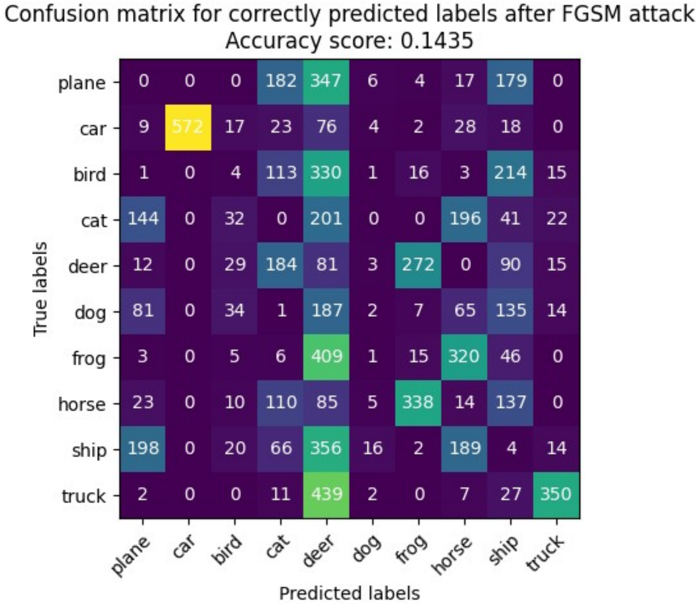
Baseline CNN Model **VDK CNNet Model**

Results

Robustness Analysis: Fast Gradient Signed Method (FGSM) Attack



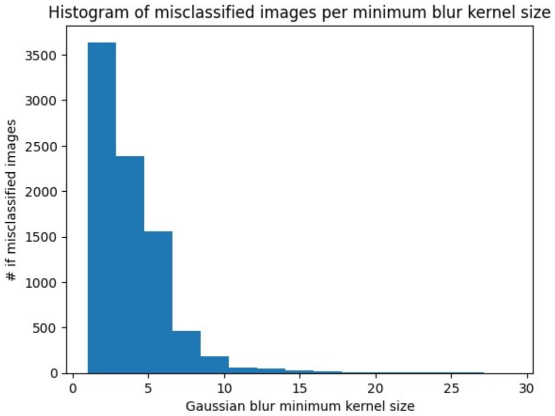
Baseline CNN Model



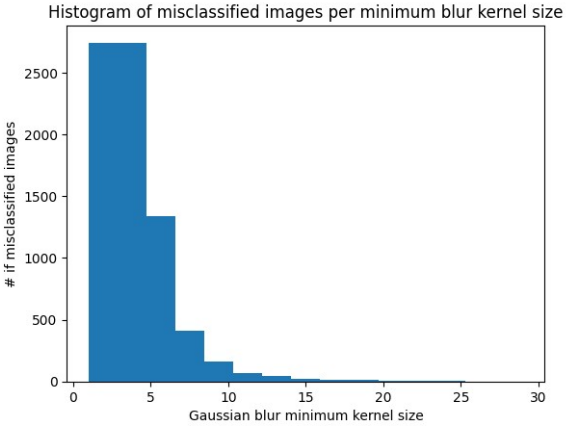
VDK CNNet Model

Results

Robustness Analysis: Gaussian Blur Attack



Immune samples: 15.91%
Affected samples: 84.09%



Immune samples: 24.29%
Affected samples: 75.71%

Baseline CNN Model

VDK CNNet Model

Thank You

Questions?

VDK CNNet Model Architecture