Debugging the Internals of Convolutional Networks

eXplainable AI paper presentation

Jakub Krajewski

Outline

1. Introduction and main goal
2. Feature-Map artifacts
3. Asymmetries in the learned weights
4. Limitations and conclusions

Introduction and main goal

Artifacts in convolutional networks

- Visualizing CNNs mostly focuses on high-level understanding of features and weights to analyze predictions

Artifacts in convolutional networks

- Visualizing CNNs mostly focuses on high-level understanding of features and weights to analyze predictions
- More subtle effects caused by convolution arithmetic can also have a significant impact on model behaviour

Artifacts in convolutional networks

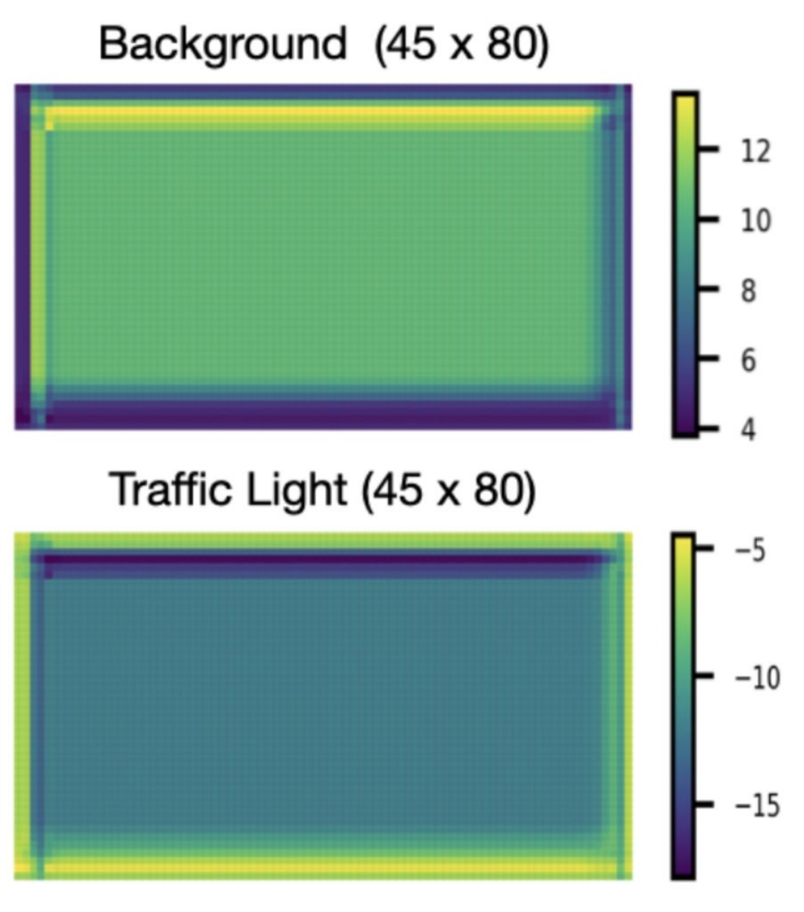
- Visualizing CNNs mostly focuses on high-level understanding of features and weights to analyze predictions
- More subtle effects caused by convolution arithmetic can also have a significant impact on model behaviour
- In feature maps, these can be seen as artifacts

Artifacts in convolutional networks

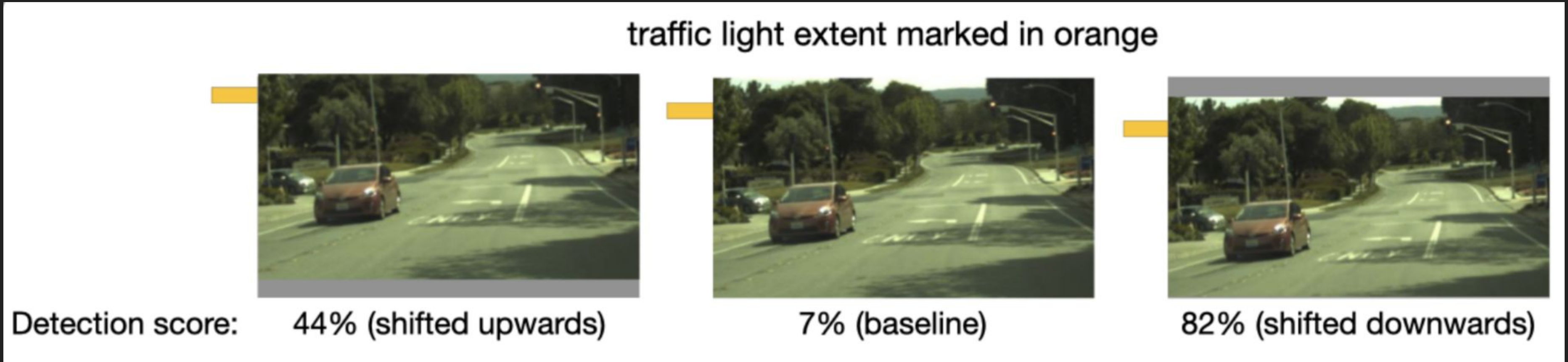
- Visualizing CNNs mostly focuses on high-level understanding of features and weights to analyze predictions
- More subtle effects caused by convolution arithmetic can also have a significant impact on model behaviour
- In feature maps, these can be seen as artifacts
- The effects can further manifest in the form of asymmetries in the learned weights

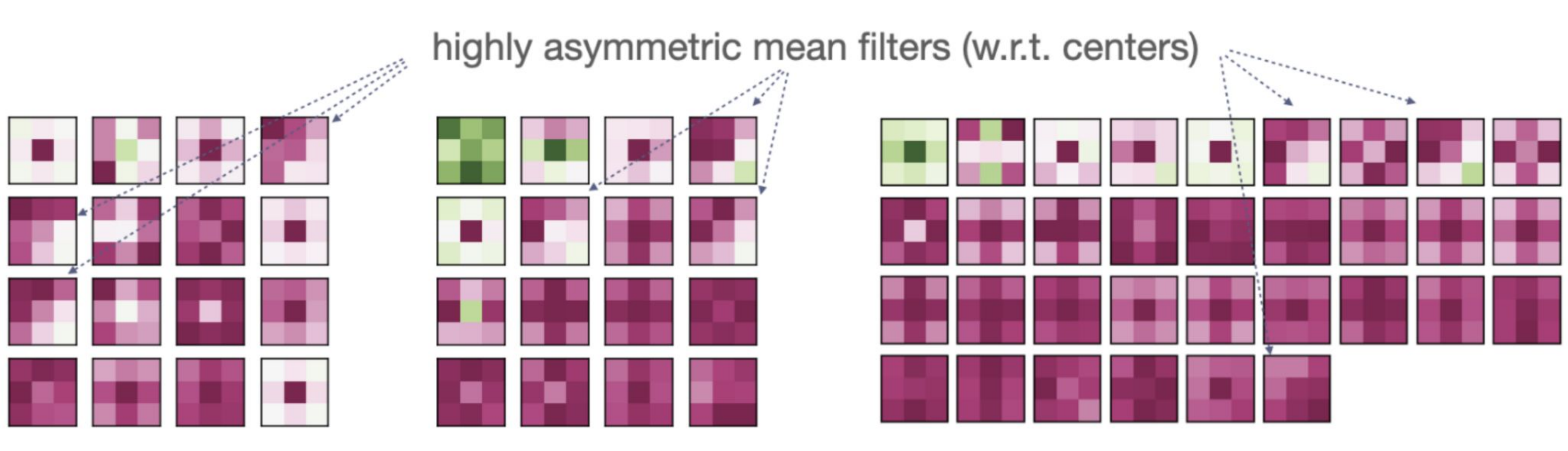
Artifacts in convolutional networks

- Visualizing CNNs mostly focuses on high-level understanding of features and weights to analyze predictions
- More subtle effects caused by convolution arithmetic can also have a significant impact on model behaviour
- In feature maps, these can be seen as artifacts
- The effects can further manifest in the form of asymmetries in the learned weights
- Later, we will discuss the details of why such problems emerge in ConvNets



Background (45 x 80)
12
10
8
6
4
Traffic Light (45 x 80)
−5
−10
−15


traffic light extent marked in orange
Detection score:
44% (shifted upwards)
7% (baseline)
82% (shifted downwards)


highly asymmetric mean filters (w.r.t. centers)

Main goal

Main goal

Detecting, visualizing and explaining biases in CNN models (artifacts, asymmetries) caused by specifics of the convolutional arithmetic

Feature-Map artifacts

How to inspect feature maps?

- The naive way would be to generate them for a number of images and try to analyze “manually”

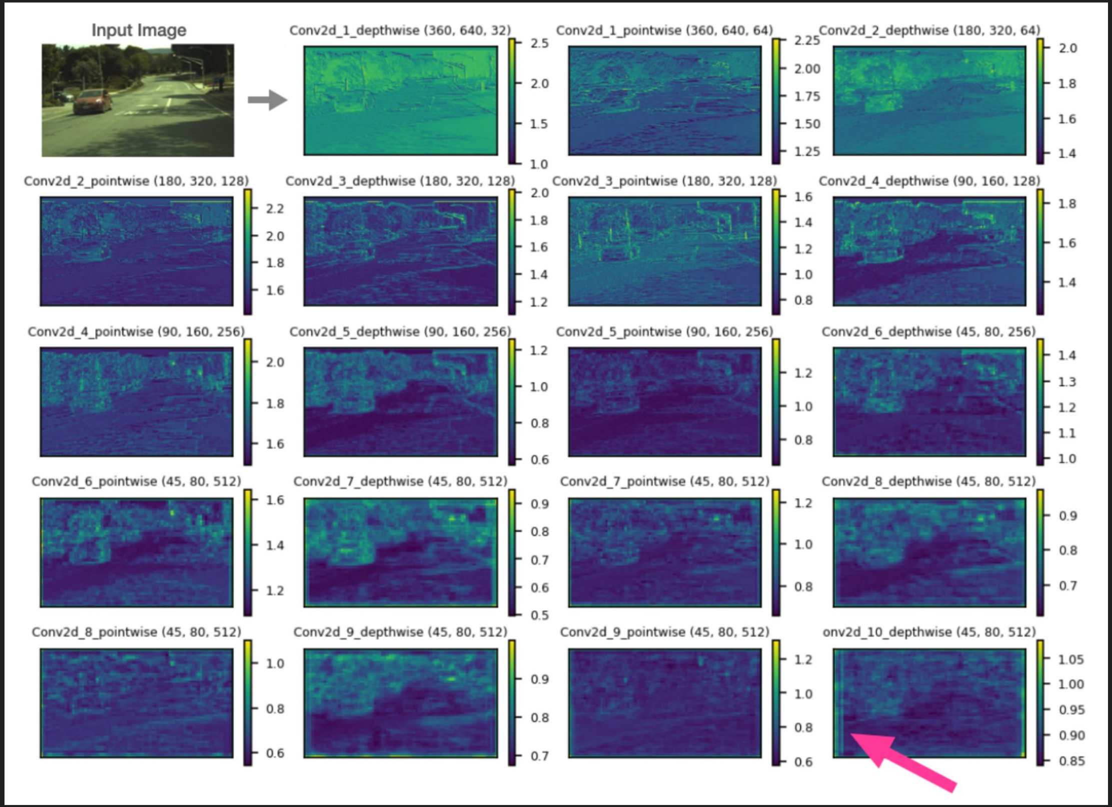
How to inspect feature maps?

- The naive way would be to generate them for a number of images and try to analyze “manually”
- This poses challenges regarding scalability and ability to spot the artifacts

How to inspect feature maps?

- The naive way would be to generate them for a number of images and try to analyze “manually”
- This poses challenges regarding scalability and ability to spot the artifacts
- To enable scalable inspection, we can aggregate the feature maps per layer and provide an overview

How to inspect feature maps?

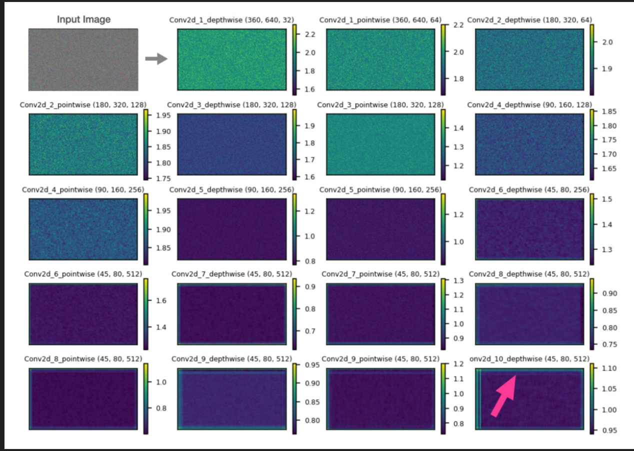


How to make artifacts stand out?

- Random input

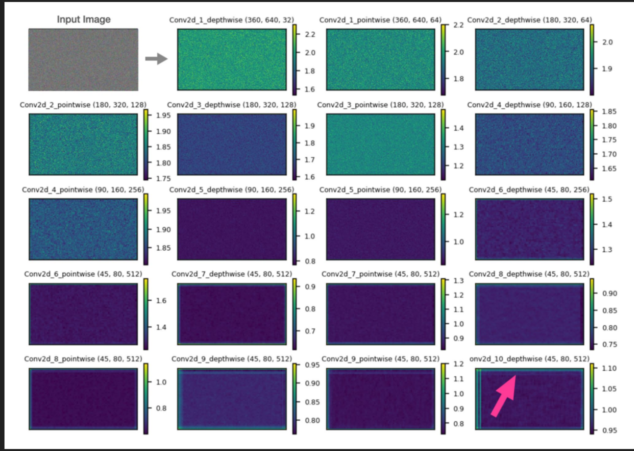
How to make artifacts stand out?

- Random input



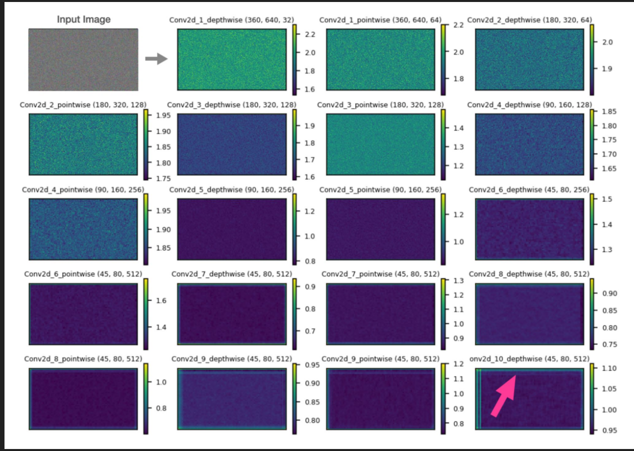
How to make artifacts stand out?

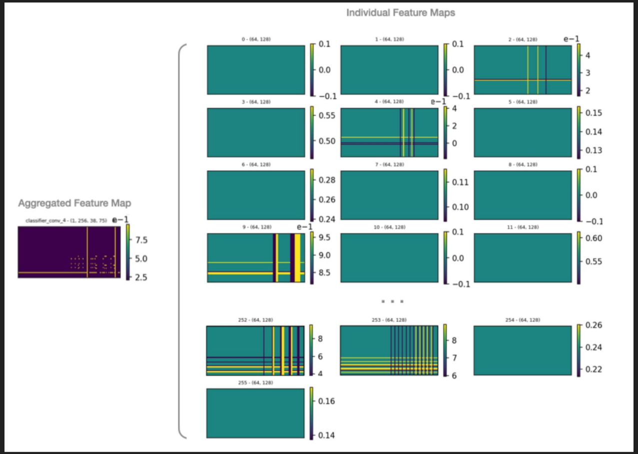
- Random input
- Constant input



How to make artifacts stand out?

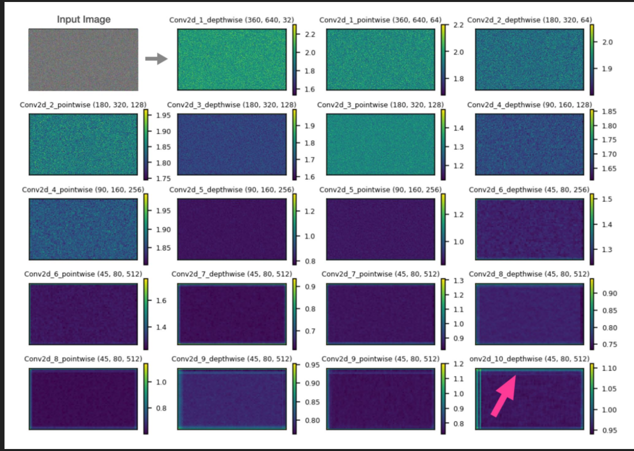
- Random input
- Constant input

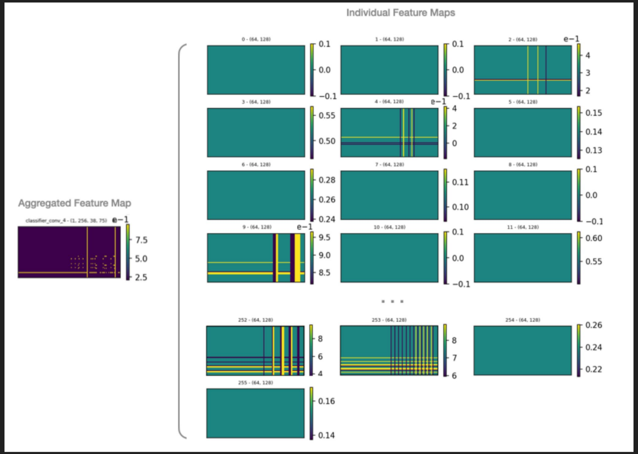




How to make artifacts stand out?

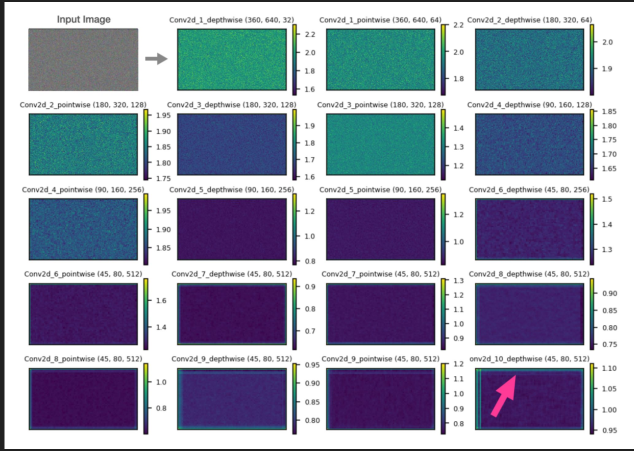
- Random input
- Constant input
- Simplified input

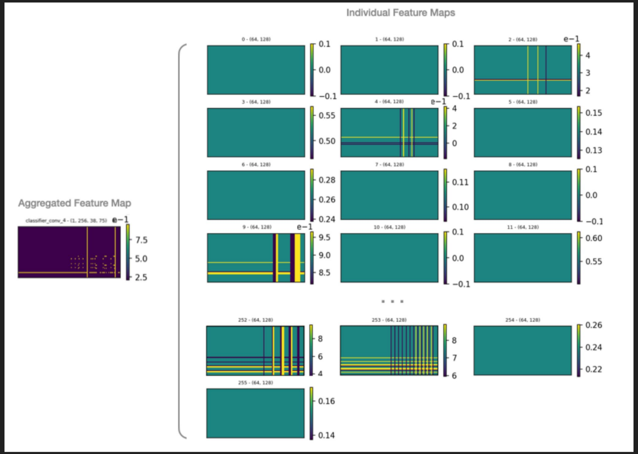


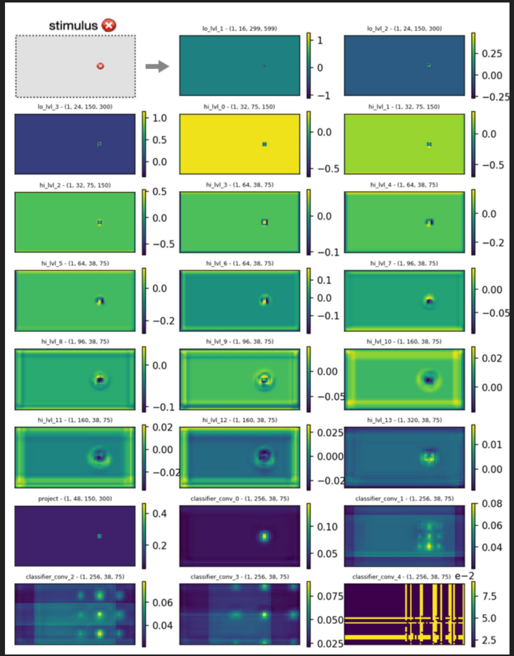


How to make artifacts stand out?

- Random input
- Constant input
- Simplified input







What causes the artifacts?

- Padding introduces an artificial boundary around the feature maps, especially under 0-padding

What causes the artifacts?

- Padding introduces an artificial boundary around the feature maps, especially under 0-padding
- Over multiple layers, this boundary interacts with neighboring values

What causes the artifacts?

- Padding introduces an artificial boundary around the feature maps, especially under 0-padding
- Over multiple layers, this boundary interacts with neighboring values
- The artificial motive gradually interferes with the activation patterns inside the feature maps

What causes the artifacts?

- Padding introduces an artificial boundary around the feature maps, especially under 0-padding
- Over multiple layers, this boundary interacts with neighboring values
- The artificial motive gradually interferes with the activation patterns inside the feature maps
- This can have a significant impact on small object detection

Asymmetries in the learned weights

The problem of asymmetries

- Weights learned by CNNs during training can adapt to the presence of feature artifacts during training

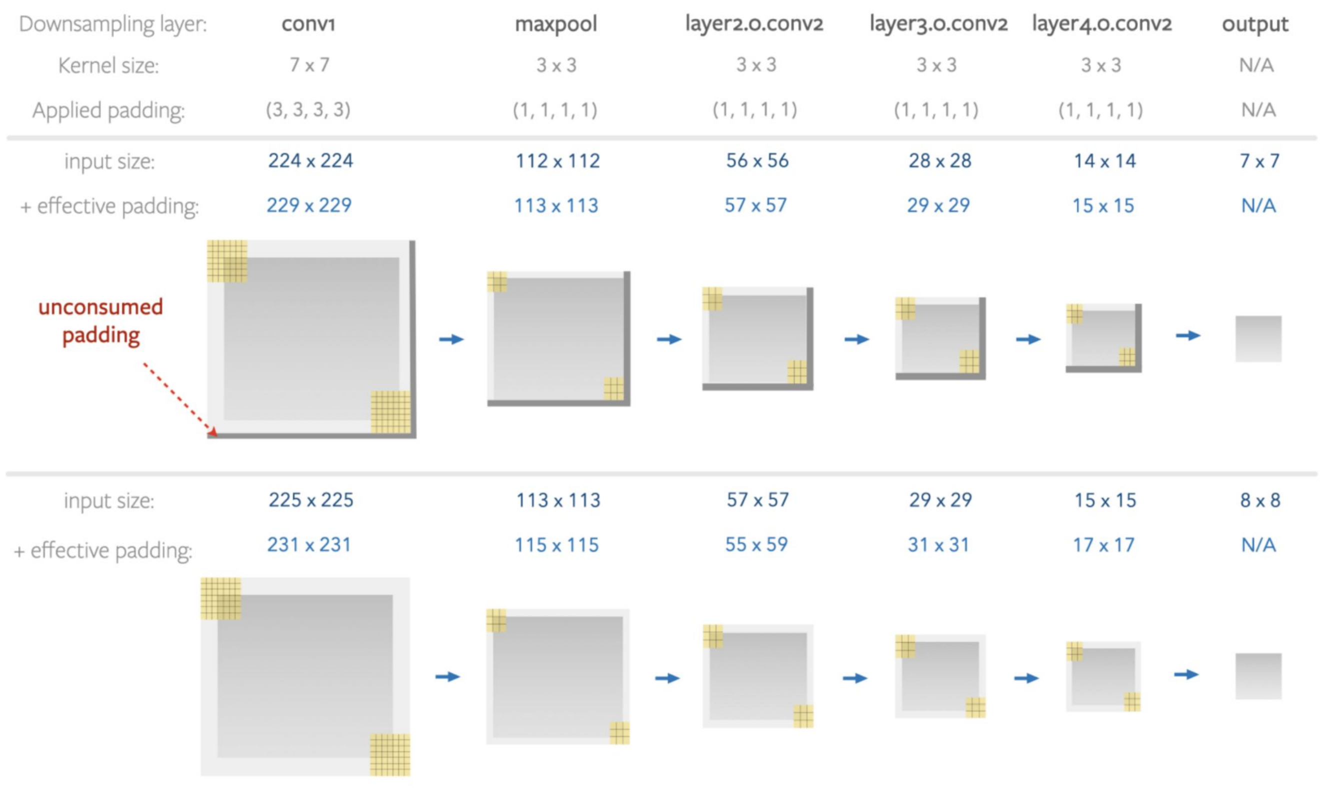
The problem of asymmetries

- Weights learned by CNNs during training can adapt to the presence of feature artifacts during training
- Layers ignore parts of the padding applied when their input size is an even number

The problem of asymmetries

- Weights learned by CNNs during training can adapt to the presence of feature artifacts during training
- Layers ignore parts of the padding applied when their input size is an even number
- This leads to patterns in kernels caused by the model architecture rather than data

The problem of asymmetries



Examining asymmetries

- To inspect the problem, we can compute the mean $k \times k$ patch in the input of each convolution layer

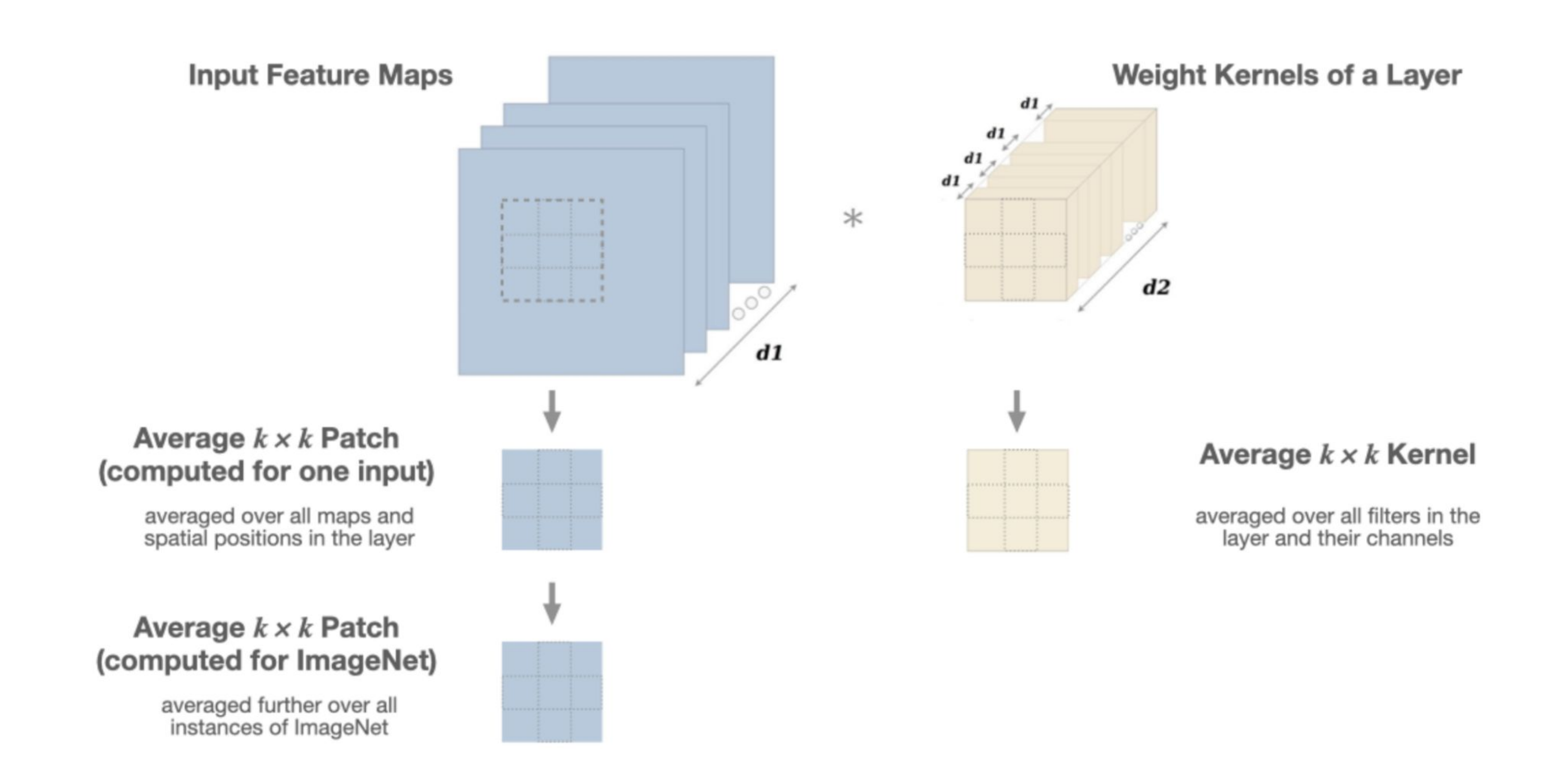
Examining asymmetries

- To inspect the problem, we can compute the mean $k \times k$ patch in the input of each convolution layer
- We can then average the input over the spatial positions, channels, and instances of the training set

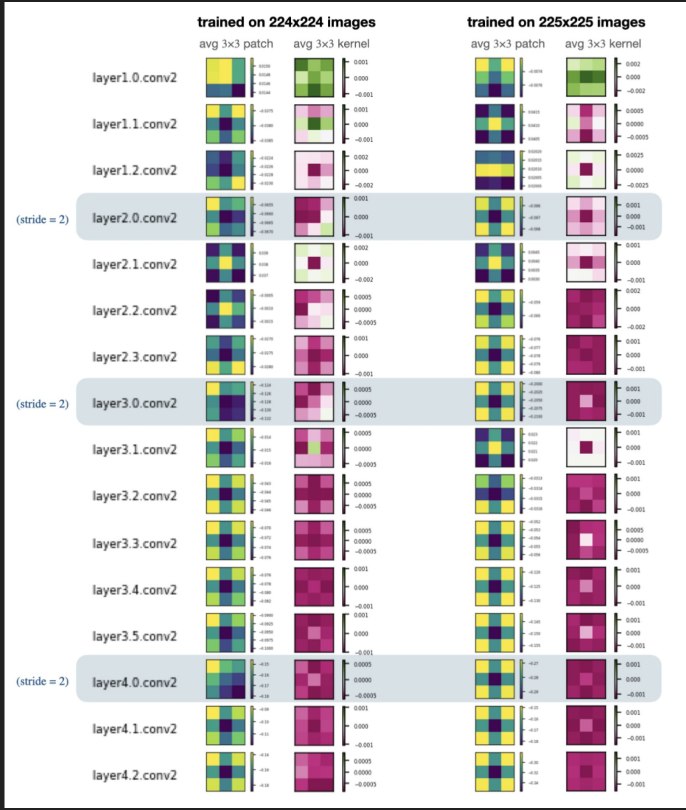
Examining asymmetries

- To inspect the problem, we can compute the mean k × k patch in the input of each convolution layer
- We can then average the input over the spatial positions, channels, and instances of the training set
- In addition, we can also compute the mean k × k kernel in the weights of the corresponding layer

Examining asymmetries



Examining asymmetries



Visualization

- Padding introduces an artificial boundary around the feature maps, especially under 0-padding

Visualization

- Padding introduces an artificial boundary around the feature maps, especially under 0-padding
- The asymmetries appear when the model is trained on 224 × 224 images

Visualization

- Padding introduces an artificial boundary around the feature maps, especially under 0-padding
- The asymmetries appear when the model is trained on 224 × 224 images
- Mean kernels of stride-convolution downsampling layers exhibit similar asymmetrie

Limitations and conclusions

Limitations

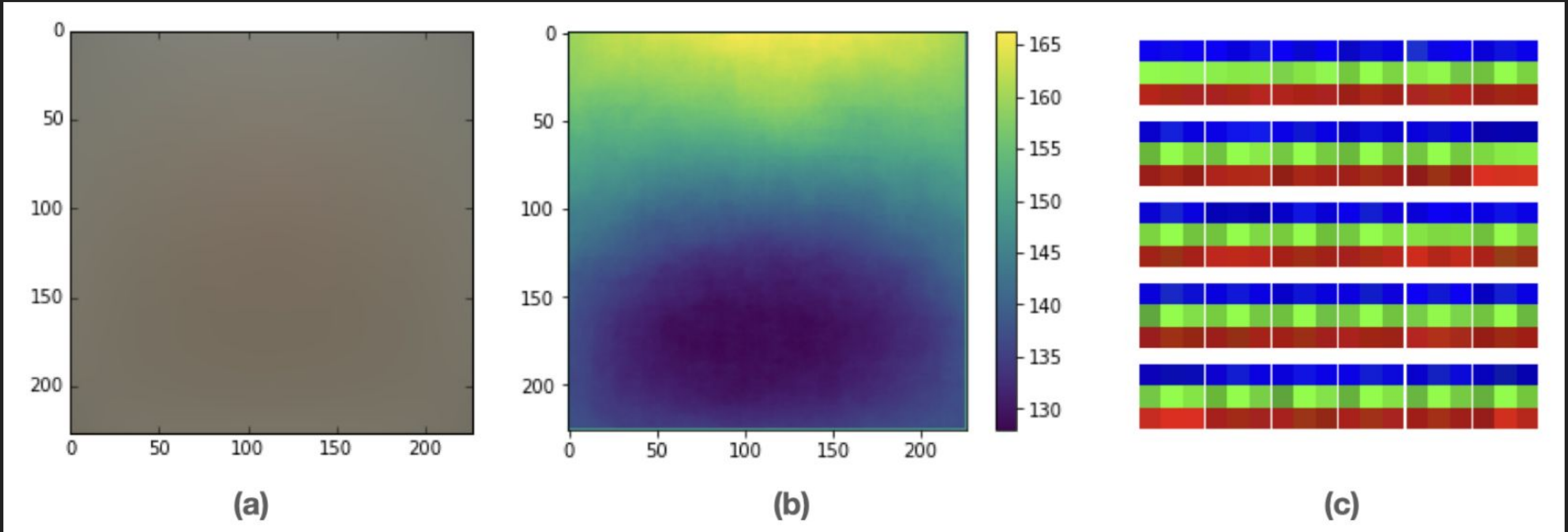
- Exposing asymmetries in the learned weights becomes challenging with kernels larger than 3×3 in size

Limitations

- Exposing asymmetries in the learned weights becomes challenging with kernels larger than 3×3 in size
- Moreover, the asymmetries might not be an artifact of CNN arithmetic, but rather reflect properties of the dataset

Limitations

- Exposing asymmetries in the learned weights becomes challenging with kernels larger than 3 × 3 in size
- Moreover, the asymmetries might not be an artifact of CNN arithmetic, but rather reflect properties of the dataset
- Analyzing the dataset features and examining the weight kernels under different hyperparameters can help determine possible reasons



0
50
100
150
200
0
50
100
150
200
(a)
0
50
100
150
200
0
50
100
150
200
165
160
155
150
145
140
135
130
(b)
(c)

Conclusions

- Feeding baseline inputs helps in exposing artifacts in feature maps

Conclusions

- Feeding baseline inputs helps in exposing artifacts in feature maps
- Computing the mean feature map and kernel in a layer is crucial to expose spatial bias and asymmetries

Conclusions

- Feeding baseline inputs helps in exposing artifacts in feature maps
- Computing the mean feature map and kernel in a layer is crucial to expose spatial bias and asymmetries
- Considering multiple visual representations of the model internals is helpful to iteratively reveal patterns

Conclusions

- Feeding baseline inputs helps in exposing artifacts in feature maps
- Computing the mean feature map and kernel in a layer is crucial to expose spatial bias and asymmetries
- Considering multiple visual representations of the model internals is helpful to iteratively reveal patterns
- Aspects of convolutional arithmetic cause variations in model representation. Mitigating them is important to improve robustness and accuracy

Thank you for your attention!