A Study on Group Equivariant CNNs

Geometric Data Analysis, MVA

Polina Barabanshchikova   Anshuman Sinha

Télécom Paris
Institut Polytechnique de Paris

December 19, 2023

Contents

Motivation: Comparision with Regular CNNs

CNNs

- are inherently equivariant to translations
- are not equivariant to other transformations: rotations, reflections, etc.
- use data augmentation to tackle such problems

Group Equivariant CNNs [Cohen, 2016]

- are designed to be equivariant to symmetry groups
- use specific filters to guarantee equivariance to group elements
- have enhanced capacity for group specific representation learning

Study & Contributions

1. Theoretical study of base paper's approaches
2. Testing and benchmarking on:
   - classification task: MNIST, MNIST-Rot
   - segmentation task on dermascopic images

Symmetry Groups

A group is a non-empty set G together with a binary operation "$\cdot$" $: G \times G \to G$ such that

- $\forall a, b, c \in G \, : \, a \cdot (b \cdot c) = (a \cdot b) \cdot c$
- $\exists e \in G \; \forall a \in G \, : \, a \cdot e = e \cdot a = a$
- $\forall a \in G \; \exists a^{-1} \in G \, : \, a \cdot a^{-1} = a^{-1} \cdot a = e$

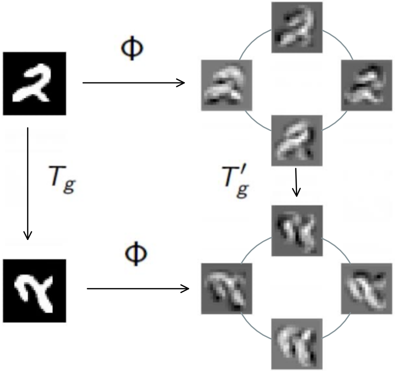
Examples

- $\mathbb{Z}^2$ – 2D integer translations
- $p4$ (pn) – compositions of translations and rotations by $\pi/2$ ($2\pi/n$)
- $p4m$ – compositions of translations, mirror reflections, and rotations by $\pi/2$

Equivariance

An operator $\Phi : X \to Y$ is G-equivariant if it commutes with the group action:

$$\Phi(T_g\, x) \;=\; T'_g\, \Phi(x)$$



Equivariance

An operator $\Phi : X \to Y$ is G-equivariant if it commutes with the group action:

$$\Phi(T_g\, x) \;=\; T'_g\, \Phi(x)$$

Properties

- invariance is a special case of equivariance with $T'_g \Phi = \Phi$
- composition preserves equivariance $\implies$ deep neural networks
- sum preserves equivariance $\implies$ skip-connections and residual blocks

G-equivariant convolutions

Regular convolution (correlation) transforms a stack of feature maps $f : \mathbb{Z}^2 \to \mathbb{R}^K$ by

$$[f \star \psi](x) = \sum_{y \in \mathbb{Z}^2} \sum_{k} f_k(y)\psi_k(x - y)$$

G-equivariant convolution transforms a stack of feature maps $f : H \to \mathbb{R}^K$ by

$$[f \star \psi](g) = \sum_{h \in H} \sum_{k} f_k(h)\psi_k(g^{-1}h) = \sum_{h \in H} \sum_{k} f_k(h) T_g[\psi_k](h)$$

Moreover, if g is a composition of a translation t and a transformation s, then

$$[f \star \psi](g) = \sum_{h \in H} \sum_{k} f_k(h) T_t[T_s \psi_k](h)$$

It can be implemented as a regular convolution with filters $T_s\psi_k$! But $\psi_k : G \to \mathbb{R}$.

Classification task: MNIST

CNN Model Architecture:

- 5 layers of 3×3 convolutions
- 8, 16, 32, 64 and 128 channels respectively
- ReLU activation, batch normalization and 3D max-pooling

Channel sizes of models to preserve the number of net trainable parameters:

- p4CNN: 8, 16, 32, 64, 128
- p6CNN: 8, 16, 32, 64, 72
- p8CNN: 4, 16, 16, 64, 64
- p4mCNN: 4, 16, 16, 64, 64

MNIST: Experiments & Analysis

Model	MNIST test accuracy	MNIST (+transforms) test accuracy
CNN	98.2%	34.1%
p4CNN	95.8%	62.9%
p6CNN	96.0%	42.0%
p8CNN	95.9%	63.5%
p4mCNN	94.2%	79.6%

Table: Accuracy of models trained on MNIST and tested on MNIST and on randomly rotated and flipped MNIST images

Key Points:

- CNN model (original accuracy: 98.2%) is highly unsuccessful at classifying rotated numbers
- Poor performance of p6CNN might be due to the lack of right angles among it's group elements - interpolation of pixel values affects accuracy

Rotated MNIST: Experiments & Analysis

Model	Test accuracy
CNN	92.43%
p4CNN	96.17%
p6CNN	94.82%
p8CNN	96.01%
p4mCNN	93.95%

Table: Accuracy of models trained and tested on Rotated MNIST

Key Points:

- p8CNN performs poorer than p4CNN; probably due to lower capacity of the model (reduced channels)
- p4mCNN lacks promising results as test set has no reflected images; mirroring parameters are unused.

Segmentation Models

SegNet Model Architecture: [Badrinarayanan, 2017]

- 4 encoder blocks followed by max-pooling
- 4 decoder blocks preceded by upsampling (or transposed convolution)
- ReLU activation and batch normalization
- last layer: max-pooling over stabilizer dimension

Analysis

- max-pooling in spatial dimension is H-equivariant $\iff$ subsample on a subgroup $H \in G$
- last max-pooling is coset pooling $\implies$ it is G-equivariant
- transposed G-convolutions with stride are G-equivariant $\implies$ upsampling is G-equivariant

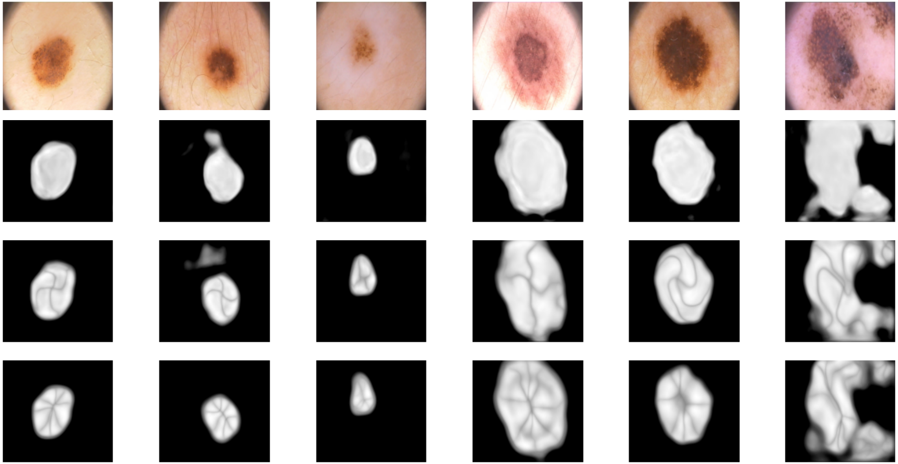
Predictions

Input

SegNet

p4SegNet

p4mSegNet



Segmentation Scores

Model	train IoU	test IoU	rotated test IoU
SegNet	0.885	0.718	0.646
p4SegNet	0.87	**0.786**	**0.786**
p4mSegNet	0.848	0.745	0.745
SegNet+	0.859	0.767	0.771
p4SegNet+	0.894	**0.788**	**0.788**

Table: Performance of segmentation models

Overview of G-equivariant CNNs

Advantages

- Equivariance guaranties
- Weight sharing
- Efficient implementation for *split* groups
- Better generalization properties

Limitations

- Only discrete groups are supported
- Matrix operations cannot perfectly model actions of some discrete groups (pn, $n > 4$)
- Computational time depends on the size of the group
- Number of weights per channel grows with the size of the group

Steerable CNNs

- Convolution kernels parameterised as steerable functions [Cohen, 2017, Weiler, 2018]
- Feature maps encode directional information by convolving with basis functions
- Fourier basis functions produce a vector field feature map of Fourier coefficients [Fageot & Uhlmann, 2021]

Advantage: These feature maps can recover continuous transformed signals as opposed to distinct discrete elements sampled explicitly.

Limitation: The number of basis functions remains a hyperparameter. This imposes an approximation on other information encoded by higher order frequencies.

References

Taco Cohen and Max Welling (2016)
Group Equivariant Convolutional Networks
International Conference on Machine Learning 48, 2990 – 2999.

Taco Cohen and Max Welling (2017)
Steerable CNNs
International Conference on Learning Representations.

Maurice Weiler, Mario Geiger Max Welling, Wouter Boomsma and Taco Cohen (2018)
3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data
Conference on Neural Information Processing Systems 32.

Vijay Badrinarayanan, Alex Kendall and Roberto Cipolla (2017)
SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation
IEEE Transactions on Pattern Analysis and Machine Intelligence 39(12).

Julien Fageot, Virginie Uhlmann, Zsuzsanna Puspoki, Benjamin Beck, Michael Unser and Adrien Depeursinge (2021)
Principled Design and Implementation of Steerable Detectors
IEEE Transactions on Image Processing

Robustness

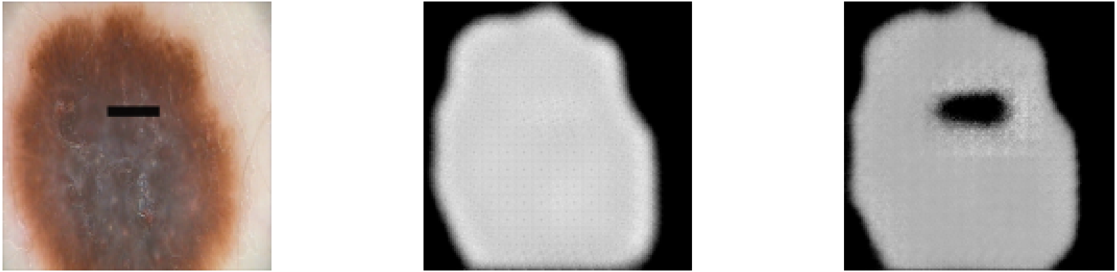


Figure: Left: Corrupted input. Middle: *p4SegNet*$_T$ prediction. Right: *SegNet*$_T$ prediction.

Feature Maps

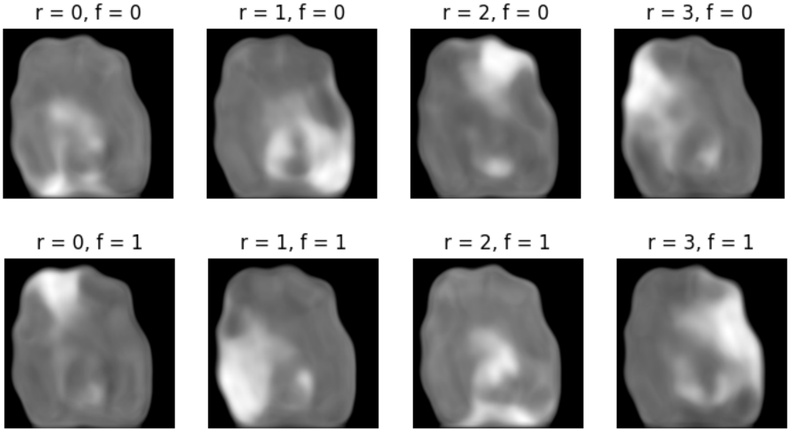
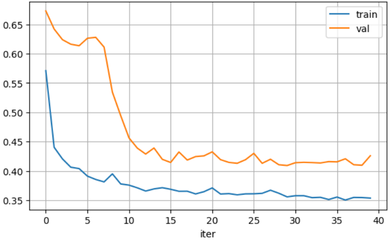


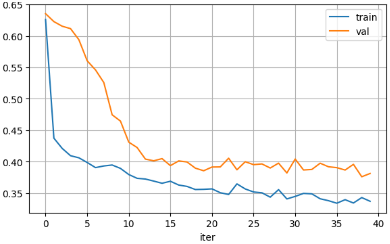
Figure: Feature maps taken from the last layer of p4mSegNet before pooling across the stabilizer dimension

Training plots

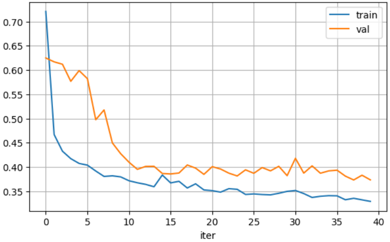
SegNet



p4SegNet



p4mSegNet



The End