LEARNING ROTATION INVARIANCE IN DEEP HIERARCHIES USING CIRCULAR SYMMETRIC FILTERS

Dhruv Kohli, Biplab Ch Das, Viswanath Gopalakrishnan, Kiran Nanjunda Iyer

Samsung R&D Institute, Bangalore, India

PROBLEM

- **Rotation invariant classification of images without data augmentation or input transformation.**

SOLUTION

- **CNN with circular symmetric filters and global pooling (max/avg).**

Circular Symmetric Kernels

A continuous kernel $f : \mathbb{R}^2 \to \mathbb{R}$ is *circular symmetric* if the corresponding polar representation of the function satisfies the following property,

$$f(r\cos\alpha, r\sin\alpha) = f(r\cos\beta, r\sin\beta) \quad \forall r \in \mathbb{R}^+ \wedge \alpha, \beta \in [0, 2\pi] \qquad (1)$$

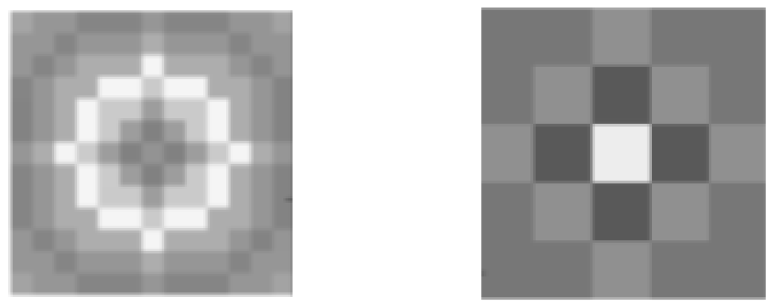

Figure 1: Circular symmetric kernels of size 13×13 and 5×5.

Rotation Equivariant Kernels

Continuous convolution in polar coordinates with h as input image, f as kernel and g as output, may be expressed as,

$$g_{(r_o,\theta_o)} = f * h_{(r_o,\theta_o)}$$

The convolution equation over the rotated image $h^{(rot)}$ with rotation angle ϕ with respect to origin, is similarly written as,

$$g^{(rot)}_{(r_o,\theta_o)} = f * h^{(rot)}_{(r_o,\theta_o)}$$

The convolving kernel f is said to be *rotation equivariant (covariant)* if,

$$g_{(r_o,\theta_o)} = g^{(rot)}_{(r_o,\theta_o+\phi)}$$

Loss function

$$\Theta = \underset{\Theta}{\operatorname{argmin}} \left[\frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(x_i, \Theta) + \lambda \sum_{l=1}^{L} \sum_{n=1}^{N_l \times d_l} \sum_{r=1}^{R_{l,n}} \sum_{\theta_i} \sum_{\theta_j}_{\theta_i,\theta_j \in S_{l,n,r}} (\theta_i - \theta_j)^2 \right]$$

The loss function consist of two terms:

- Mean prediction error on the training dataset.
- Penalty due to circular symmetry constraint - sum of squared euclidean distance between each pair of parameters lying on a circular ring of radius r in the n_{th} constrained kernel of the l_{th} convolution layer, summed over circular rings of all possible radii in each constrained kernel in each convolution layer.

Rotation Invariant CNN Architecture

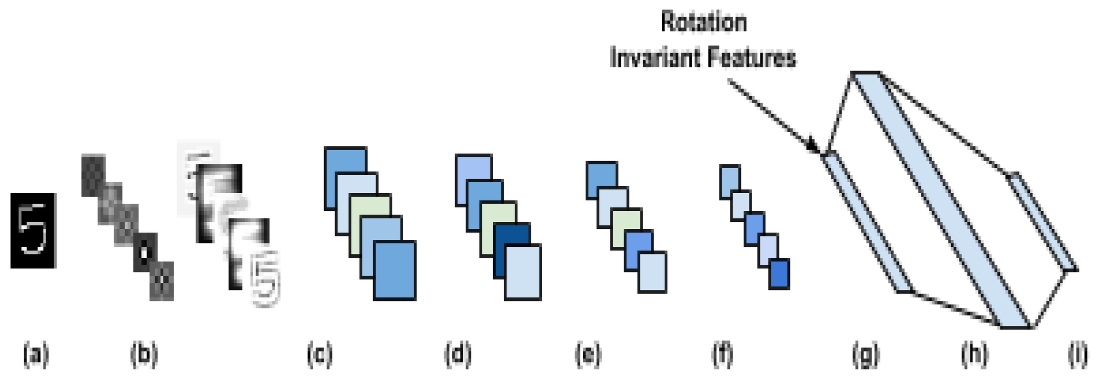

Figure 2: Network topology description. Input image x (a) is passed through a sequence of convolution layers containing **circular symmetric kernels** (b,f) and a **global maxpooling layer** (g) until the vector of scalars is not achieved. This vector serves as an input to a fully connected layer (h) possibly with dropout and further propagates to the network output (i).

Circular Symmetric Kernel ⇒ Rotation Equivariant Kernel

Let f represents a circular symmetric kernel. **Notation used:** $p_{(a,\xi)} = p(a\cos\xi, a\sin\xi)$.

$$g_{(r_o,\theta_o)} = \int_0^{2\pi}\int_0^{\infty} f_{(r,\theta)}\, h(r_o\cos\theta_o - r\cos\theta, r_o\sin\theta_o - r\sin\theta)\, r\,dr\,d\theta \qquad (2)$$

$$h^{(rot)} \xleftarrow[\text{with respect to origin}]{\text{Anti-clockwise rotation of } \phi} h$$

$$\begin{bmatrix}\cos\phi & -\sin\phi \\ \sin\phi & \cos\phi\end{bmatrix} \begin{bmatrix} r_o\cos\theta_o - r\cos\theta \\ r_o\sin\theta_o - r\sin\theta \end{bmatrix} = \begin{bmatrix} r_o\cos(\theta_o+\phi) - r\cos(\theta+\phi) \\ r_o\sin(\theta_o+\phi) - r\sin(\theta+\phi) \end{bmatrix} \qquad (3)$$

$$h^{rot}(r_o\cos(\theta_o+\phi) - r\cos(\theta+\phi), r_o\sin(\theta_o+\phi) - r\sin(\theta+\phi)) = h(r_o\cos\theta_o - r\cos\theta, r_o\sin\theta_o - r\sin\theta) \qquad (4)$$

$$\text{eq. } 2 \xleftarrow{\text{replacing}} \text{eq. } 4 \wedge \text{using eq. } 1$$

$$g_{(r_o,\theta_o)} = \int_0^{2\pi}\int_0^{\infty} f_{(r,\theta+\phi)}\, h^{rot}(r_o\cos(\theta_o+\phi) - r\cos(\theta+\phi), r_o\sin(\theta_o+\phi) - r\sin(\theta+\phi))\, r\,dr\,d\theta \qquad (5)$$

$$\theta \leftarrow \theta + \phi$$

$$\begin{aligned} g_{(r_o,\theta_o)} &= \int_{-\phi}^{2\pi-\phi}\int_0^{\infty} f_{(r,\theta)}\, h^{rot}(r_o\cos(\theta_o+\phi) - r\cos\theta, r_o\sin(\theta_o+\phi) - r\sin\theta)\, r\,dr\,d\theta \\ &= f * h^{(rot)}_{(r_o,\theta_o+\phi)} \\ &= g^{(rot)}_{(r_o,\theta_o+\phi)} \end{aligned} \qquad (6)$$

Experiments and Results

Layer	Parameters and output channel size
input	size: 32×32, channel: 1
convolution	kernel: 5×5, channel: 40
relu	
convolution	kernel: 5×5, channel: 40
relu	
convolution	kernel: 5×5, channel: 40
relu	
convolution	kernel: 5×5, channel: 80
relu	
convolution	kernel: 5×5, channel: 80
relu	
global max pooling	
linear	channel: 5120
relu	
dropout	rate: 0.5
linear	channel: 10
softmax	

Table 1: The topology of the network used in our experiments.

Trained on	Architecture	Test Accuracy, %	
		MNIST-ROT	MNIST-ORIG
MNIST-ROT	Without Circular Symmetry	95.00	50.87
	With Circular Symmetry ($\lambda = 1$)	94.31	94.38
	With Circular Symmetry ($\lambda = 3$)	94.08	94.22
MNIST-ORIG	Without Circular Symmetry	33.46	99.42
	With Circular Symmetry ($\lambda = 1$)	50.66	99.43
	With Circular Symmetry ($\lambda = 3$)	62.41	99.08

Table 2: Accuracies obtained by the proposed CNN architecture with and without circular symmetric kernels over the test set of MNIST-ORIG and MNIST-ROT (mnist-rot-12k) datasets.

Method	Error, % MNIST-ROT	Parameters, M	Flops, M
TI-Pooling	2.2	3.47	248
Circular Symmetric Kernel (ours)	5.69	0.78	84

Table 3: Comparison of our model with circular symmetry constraint with TI-Pooling method. Note that the flop computation for TI-Pooling excludes the flops used for computing 24 rotations of the input image.

Advantages

- High generalization across similar datasets ⇒ Robust architecture.
- Less number of parameters ⇒ Low computation.
- No data augmentation and transformation.