Adding geometry, invariance & interpretability to filters in deep learning

Guillermo SAPIRO

Assume you got all images of the same class. Best representation: those are all the same vectors, perpendicular to the other class vector.
Replace soft-max loss by orthogonal low-rank loss.

Regularizing with filter structures

Grab a working network, add an extra geometry filter for better performances. CNN, etc are just learning filters. What if we decompose those? Go down on the # of parameters by truncating, more stable. [DCFNet]
Almost no performance loss but gain about an order of magnitude in the # of params.
Basically: we represent our signal in a sparse fashion.

Invariant / Equivarient representation

Data augmentation (translation, rotation, scaling) adds no information. Regular NNs are invariant by translation but not rotation/scale invariant.
If we replace our filter on-basis, they become rotational invariant. [Rot DCF paper].
Forces the system to be invariant. We also add truncation to reduce the # of parameters.

Scale-invariance is a bit trickier but is the same concept.

Amélie Chatelain