Warped Convolutions: Efficient Invariance to Spatial Transformations

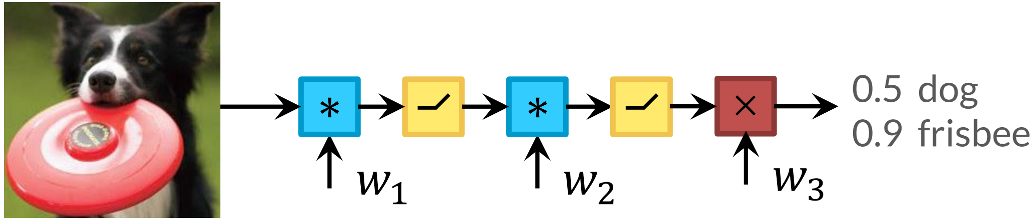
João F. Henriques   Andrea Vedaldi

UNIVERSITY OF OXFORD

Translation-equivariance and CNNs

Convolutional Neural Networks (CNN) are ubiquitous in Computer Vision

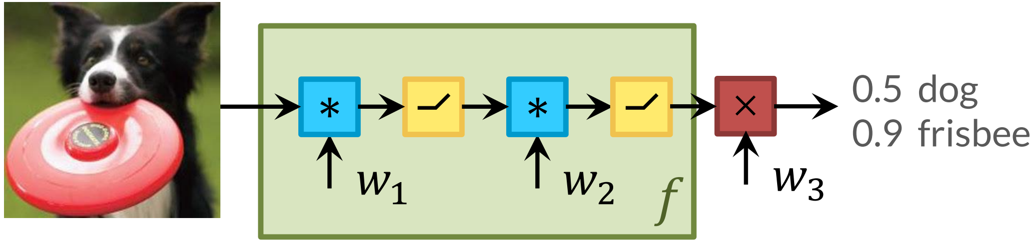
- The first stage in most image recognition pipelines is a CNN.



Translation-equivariance and CNNs

Convolutional Neural Networks (CNN) are ubiquitous in Computer Vision

- A large part of its success is due to **translation-equivariance**.
- Translating the input image also translates the predictions.



$$\underbrace{f}_{\text{CNN}}(\overbrace{\tau_t}^{\text{Translation by } t}(\underbrace{x}_{\text{Input}})) = \tau_t(f(x))$$

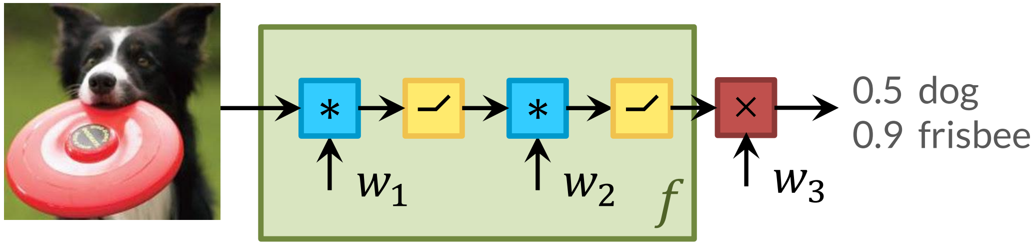
Translation-equivariance and CNNs

Why translation-equivariance?

- Vastly **fewer parameters** to learn in linear layers.

  For example, ResNet's 1st layer:

  - 9,408 convolutional parameters.
  - $\sim 1.2 \times 10^{11}$ if simple linear layer (FC)!
- Less computation (limited filter support).
- Local memory access (faster).
- Reflects statistics of natural images.

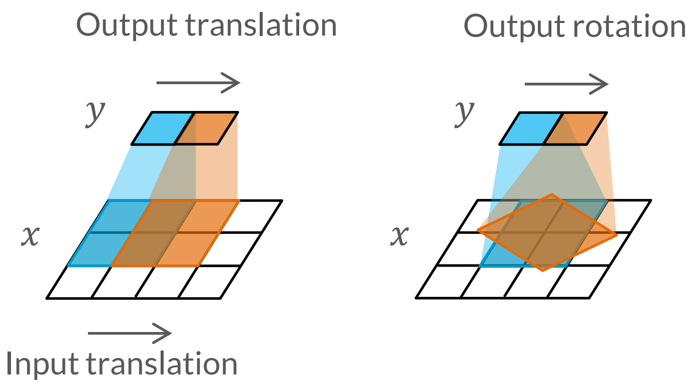


$$\underbrace{f}_{\text{CNN}}(\overbrace{\tau_t}^{\text{Translation by } t}(\underbrace{x}_{\text{Input}})) = \tau_t(f(x))$$

___________-equivariance and CNNs?

Image statistics are largely invariant to other transformations (scale, rotation, etc).

$\Rightarrow$ Can we get the same benefits in those cases?

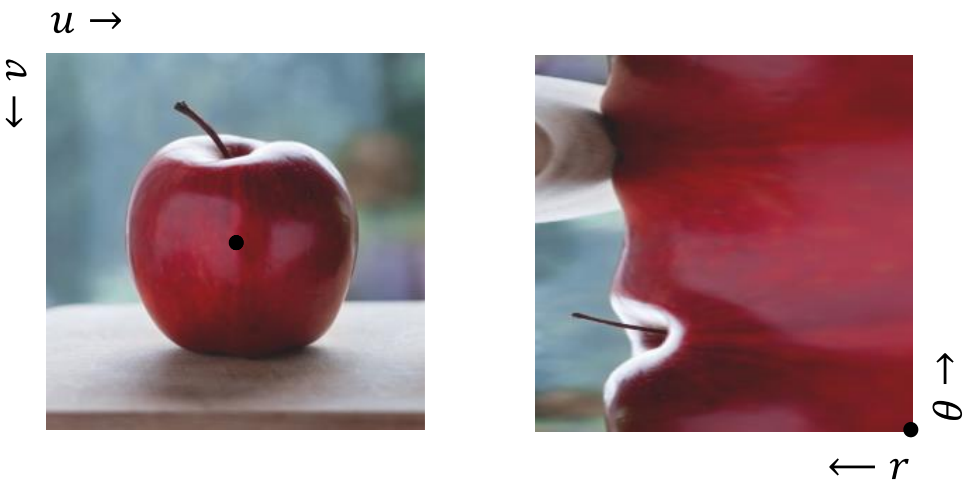


Problems:

- Spatially-varying filter, requires computing transformation at every step.
- Loses access to modern fast convolution algorithms (Winograd, FFT).

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**



- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

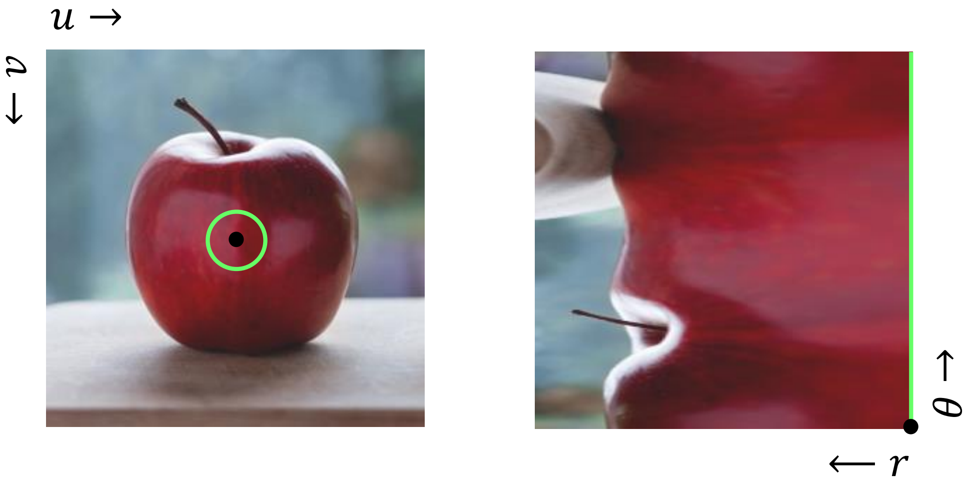
$$r = \log \sqrt{u^2 + v^2}$$

$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**



- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

$$r = \log \sqrt{u^2 + v^2}$$

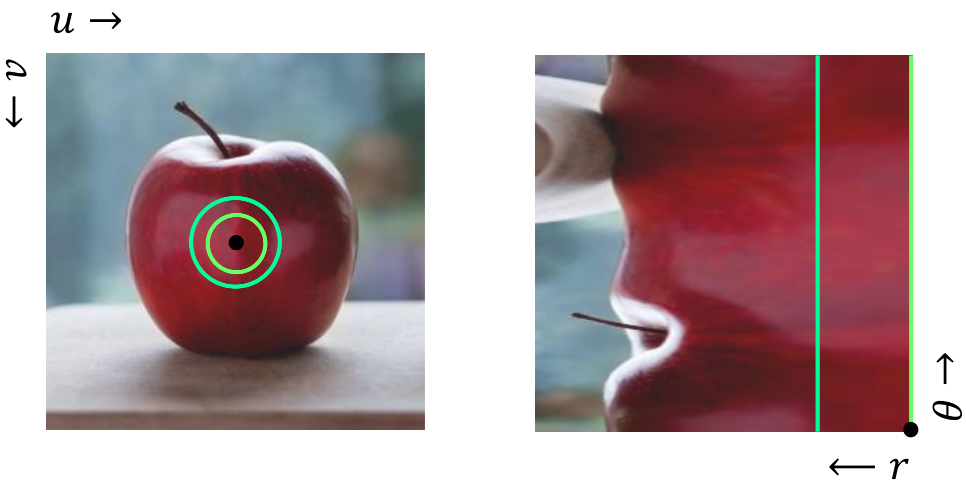
$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

UNIVERSITY OF OXFORD

Inspiration: **Log-Polar Transform**



- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

$$r = \log \sqrt{u^2 + v^2}$$

$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

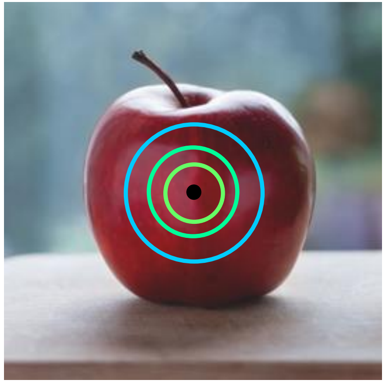
The mind-bending Log-Polar Transform

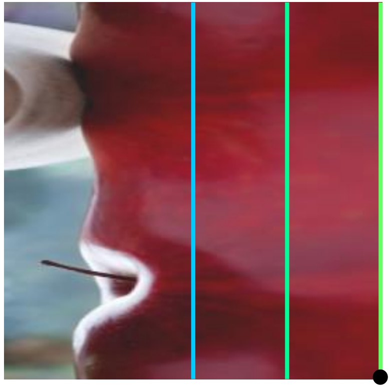
UNIVERSITY OF OXFORD

Inspiration: **Log-Polar Transform**

$u \rightarrow$

$v \downarrow$





$\leftarrow r$

$\theta \uparrow$

- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

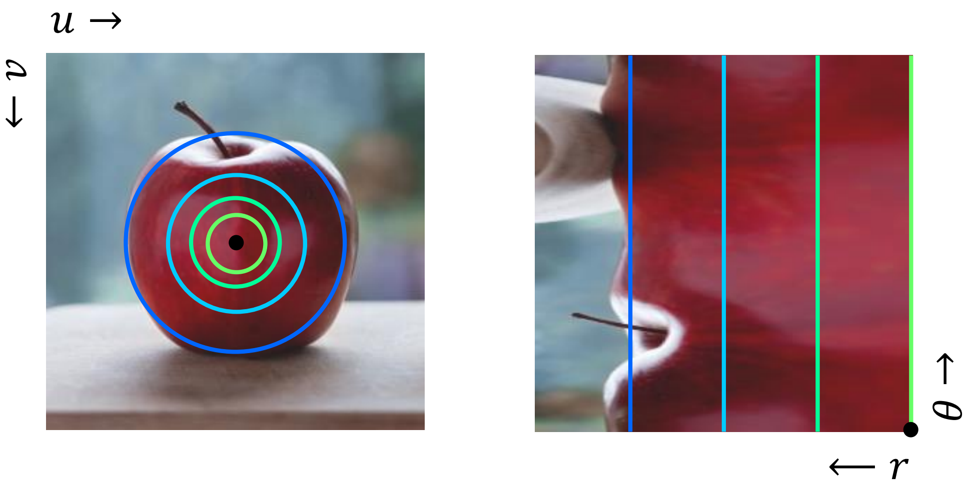
$$r = \log \sqrt{u^2 + v^2}$$

$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**



- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

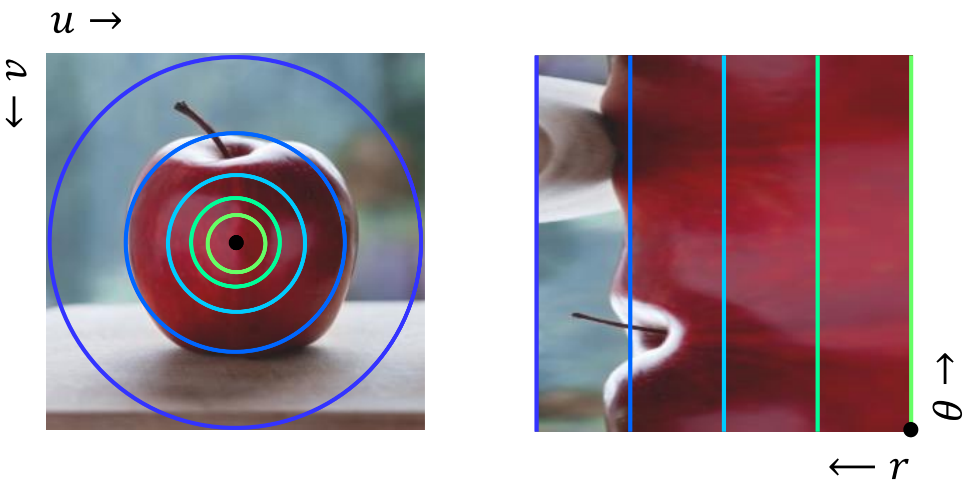
$$r = \log \sqrt{u^2 + v^2}$$

$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**



- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

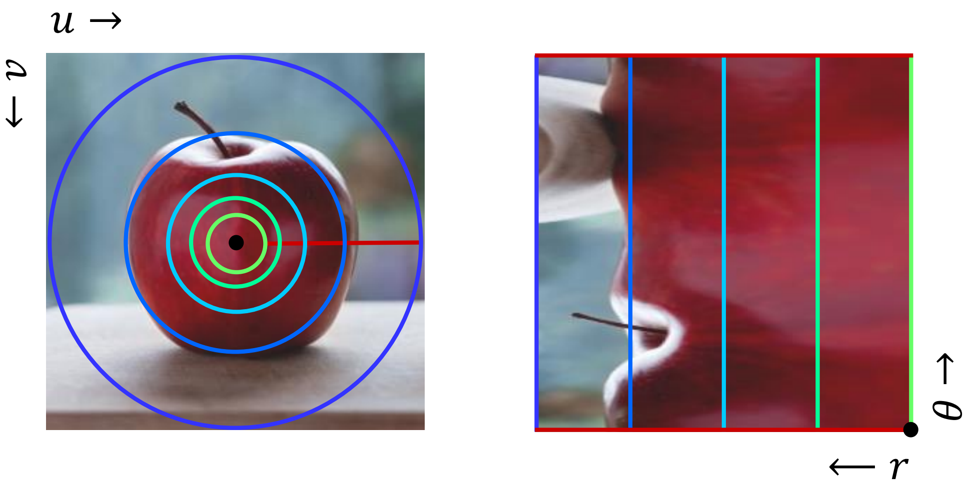
$$r = \log \sqrt{u^2 + v^2}$$

$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**



- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

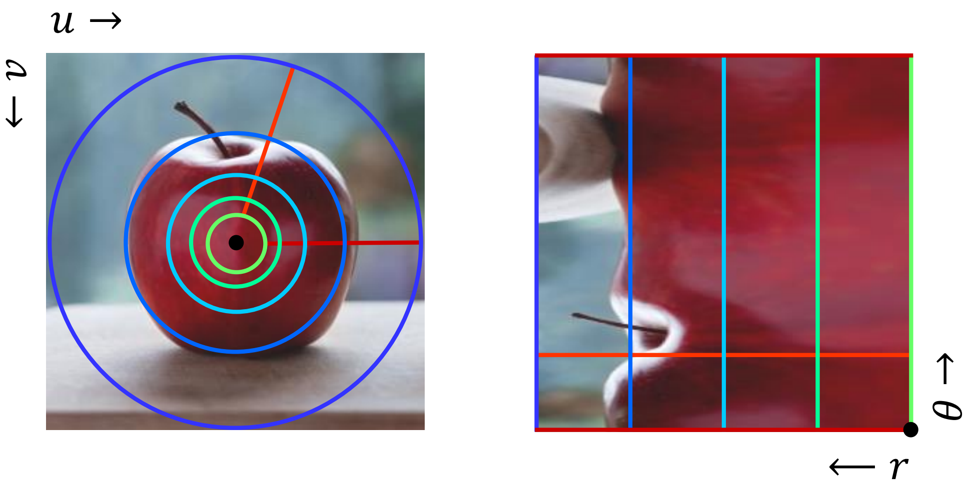
$$r = \log \sqrt{u^2 + v^2}$$

$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**



- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

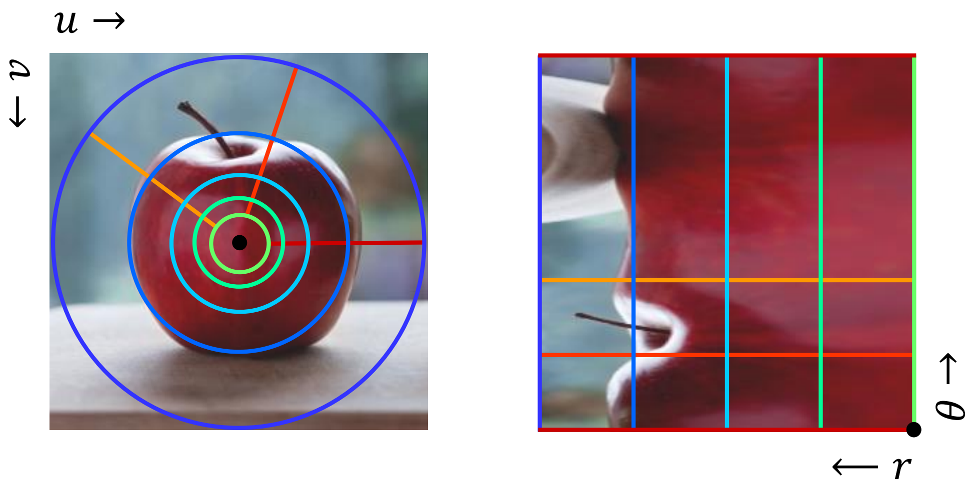
$$r = \log \sqrt{u^2 + v^2}$$

$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**



- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

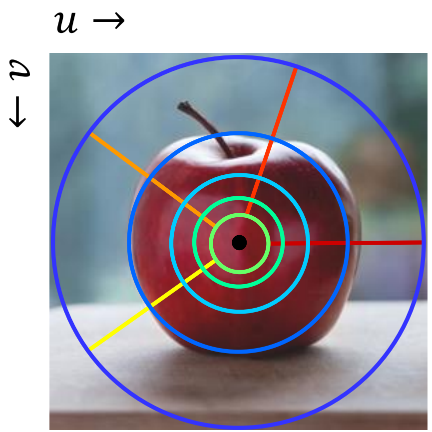
$$r = \log \sqrt{u^2 + v^2}$$

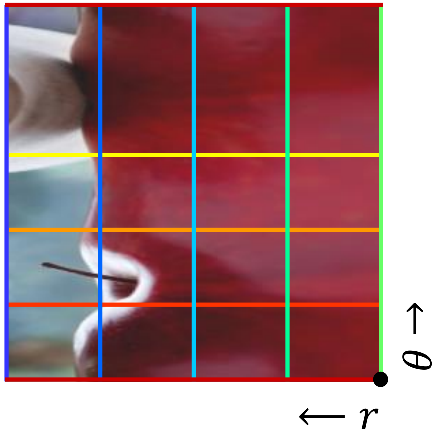
$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**





- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

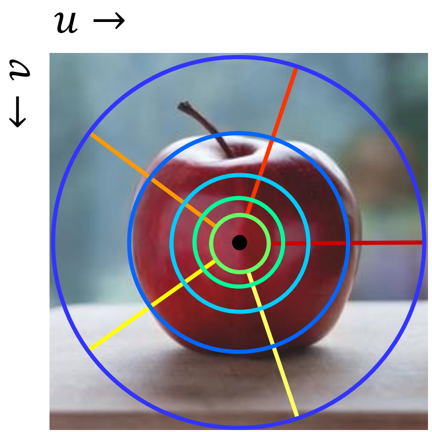
$$r = \log \sqrt{u^2 + v^2}$$

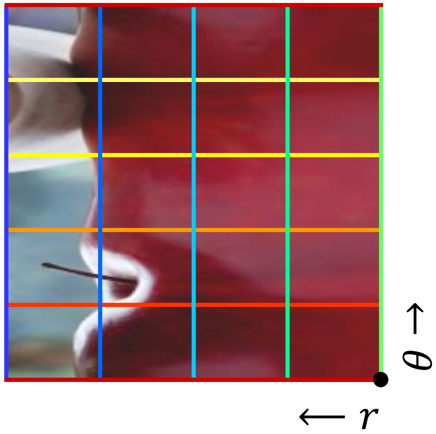
$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**





- A well-known trick from signal processing.
- Remaps (warps) space according to:

$$(u, v) \mapsto (r, \theta)$$

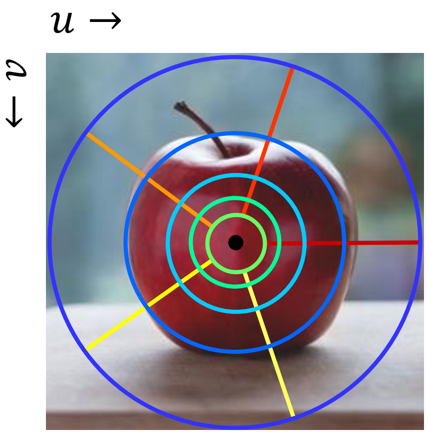
$$r = \log \sqrt{u^2 + v^2}$$

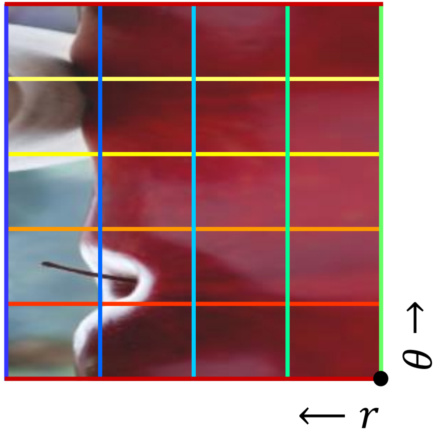
$$\theta = \tan^{-1}\left(\frac{v}{u}\right)$$

(E.g. by bilinear interpolation.)

The mind-bending Log-Polar Transform

Inspiration: **Log-Polar Transform**





Scale/rotation
in the **original** space (u, v)

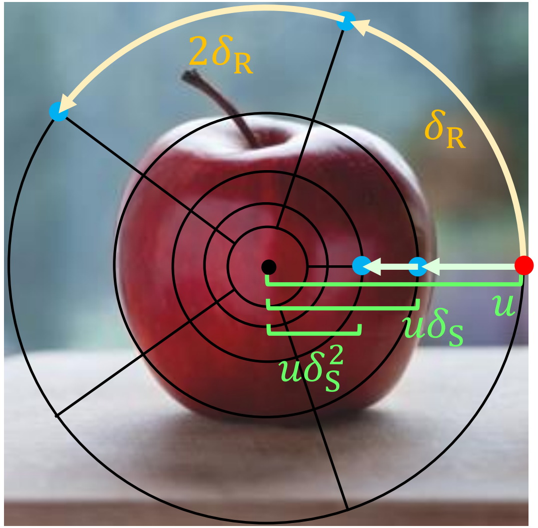
$\Longleftrightarrow$

Horizontal/vertical **translation**
in the **warped** space (r, θ)

A CNN in this warped space will implicitly work with scales/rotations.

Generalizing

Observation



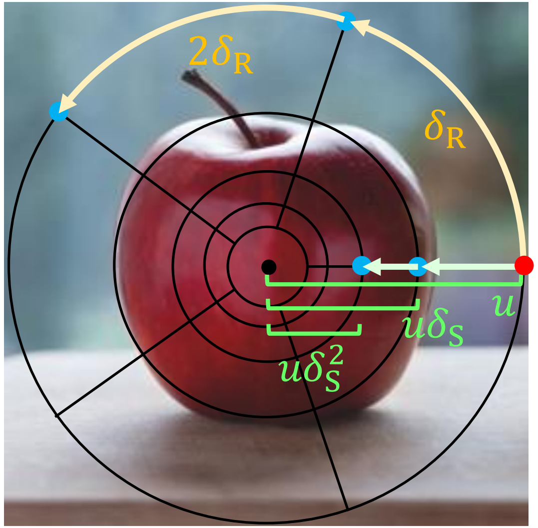
The log-polar interpolation (warp) grid can be generated as follows:

- Take an arbitrary pivot point x_0.
- Consider an elementary scale δ_S and an elementary rotation δ_R (w.r.t. the origin •).
- Repeatedly apply elementary scales/rotations to x_0 to obtain the grid points.

(n scales, m rotations → $n \times m$ grid)

Generalizing

⇒ **Does this process generalize to other transformations?**



- Answer: **Yes!** (Proof in the paper.)

Requirements:

- 2D parameter-space (e.g. scale + rotation).
- Transformation group $\mathcal{G}$ must be Abelian (i.e., composition of transformations does not depend on their order, $gh = hg$, for $g, h \in \mathcal{G}$).

Transformation examples

- We can create *analogues* of the log-polar warp for many other spatial transformations.
- They guarantee equivariance to aspect ratio, smooth deformations, and some 3D operations.

Scale / rotation

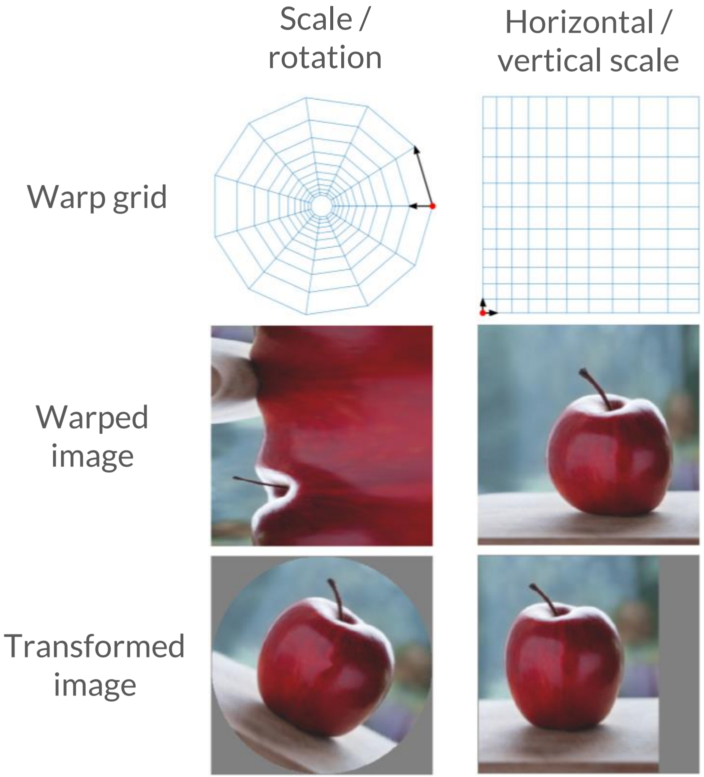
Warp grid

Warped image

Transformed image

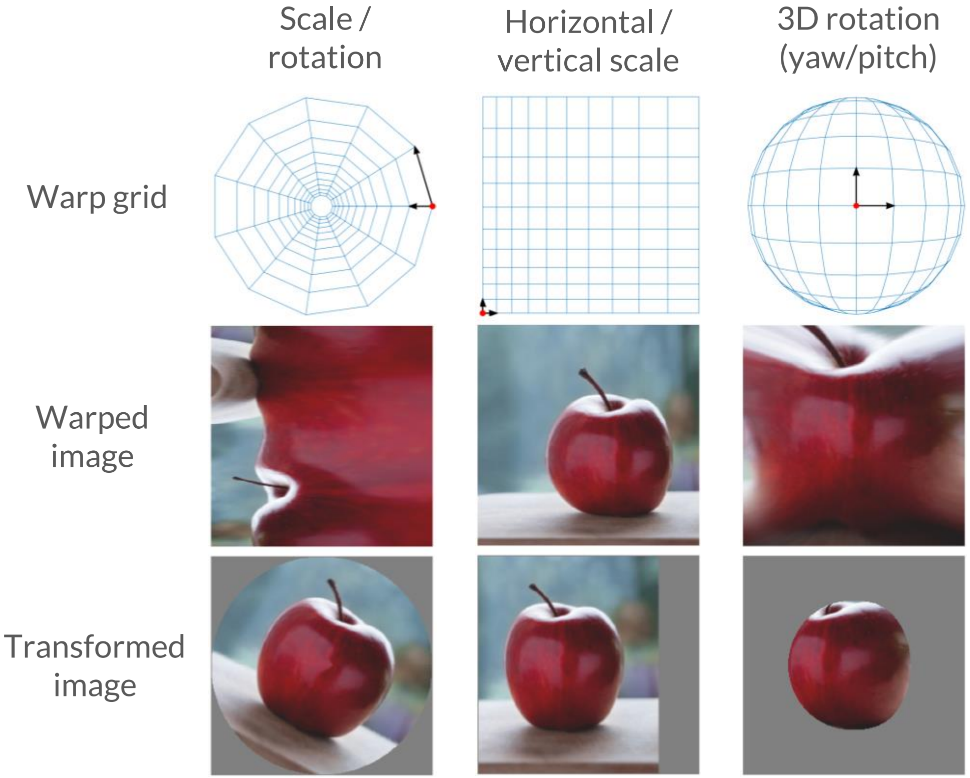
Transformation examples

- We can create *analogues* of the log-polar warp for many other spatial transformations.
- They guarantee equivariance to aspect ratio, smooth deformations, and some 3D operations.



Transformation examples

- We can create *analogues* of the log-polar warp for many other spatial transformations.
- They guarantee equivariance to aspect ratio, smooth deformations, and some 3D operations.



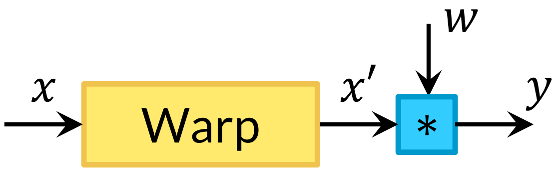
Transformation-equivariant CNNs

UNIVERSITY OF OXFORD

A recipe for transformation-equivariant CNNs

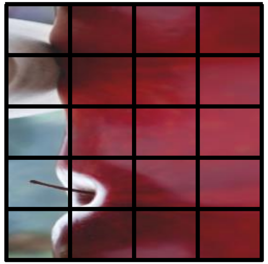
1. (Offline.) Generate warp grid from elementary transformations.
2. Apply warp to input (bilinear interpolation).
3. Apply standard convolutional operators.

The result can be shown to be equivariant to the chosen transformation.



Input image x

Warped image x'



Experiments

Google Earth dataset
Vehicle pose estimation

(Scale/rotation equivariance)

	Rot. err.	Scale err.
CNN+FC	22.54	5.04
CNN+softargmax	9.36	4.87
Warped CNN	8.29	4.79
(Dieleman et al., 2015)	31.11	4.29

AFLW dataset
Head pose estimation

(3D rotation equivariance)

	Yaw err.	Pitch err.
CNN+FC	12.56	6.59
STN (Jaderberg et al., 2015)	13.65	7.22
Warped CNN	7.07	5.28

Conclusions

- Convolutional operators can be generalized to a broad class of spatial transformations.
- We present a construction based on a single warp (negligible overhead) and standard convolutions.
- This allows us to train fast CNNs that are equivariant to other useful transformations.
- *Future work*: mixing filter banks of different transformations inside a CNN.