Group Equivariant Convolutional Networks

Seminar: Selected Topics in DL: Equivariance and Dynamics

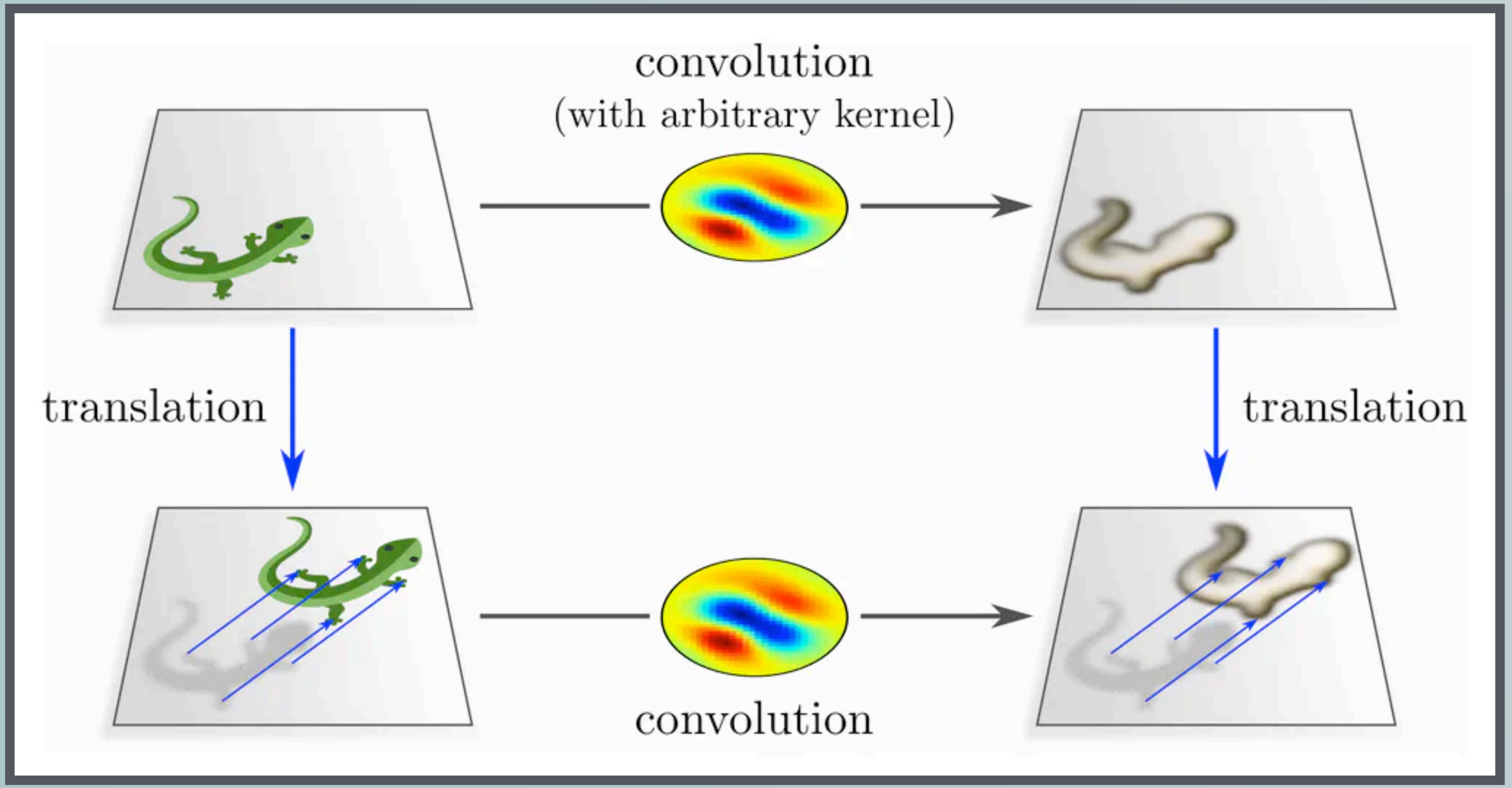
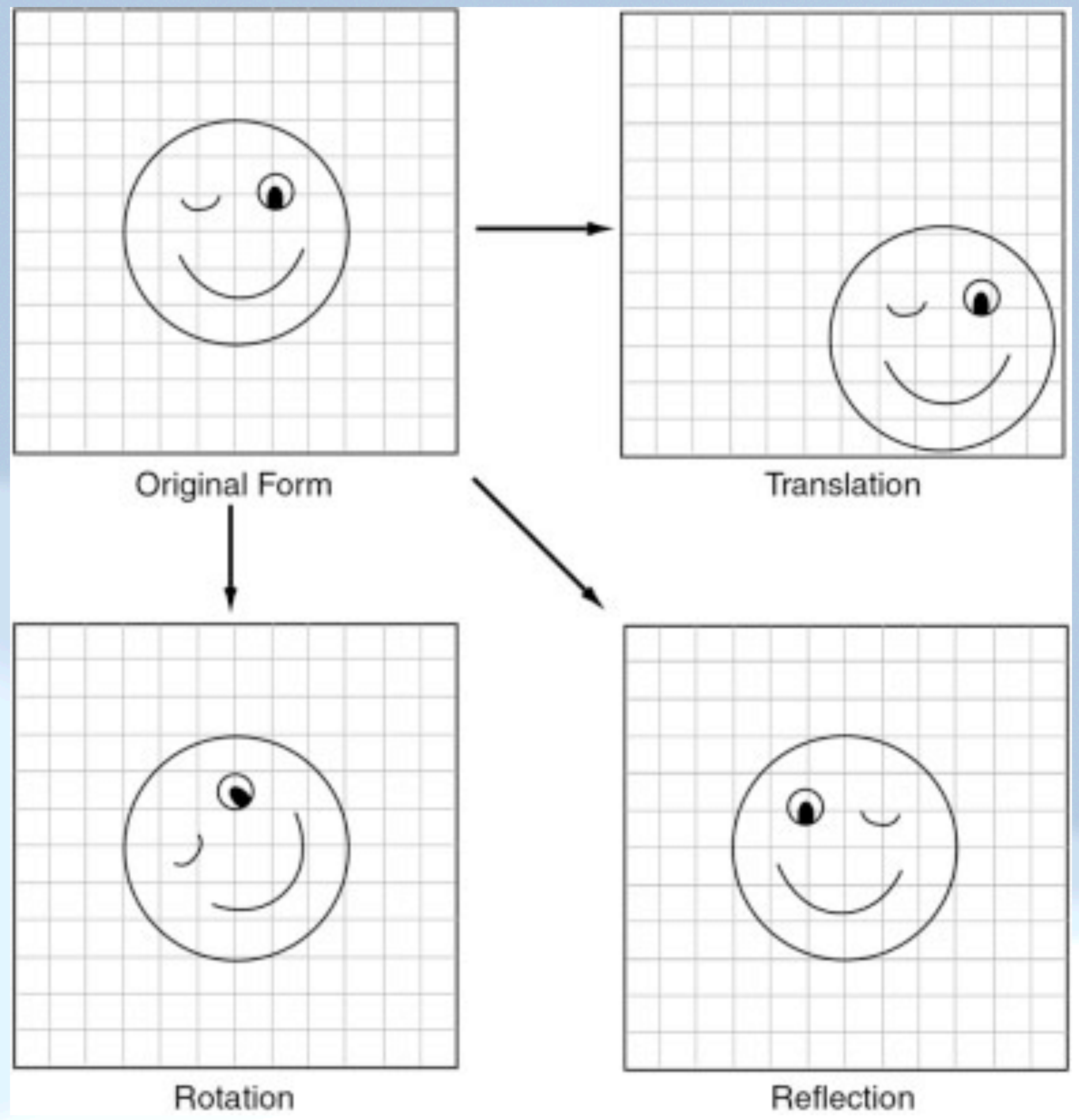


Illustration of translation equivariance. Image credit: Maurice Weiler

20.05.2025 | Iñaky Ordiales

Based on T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

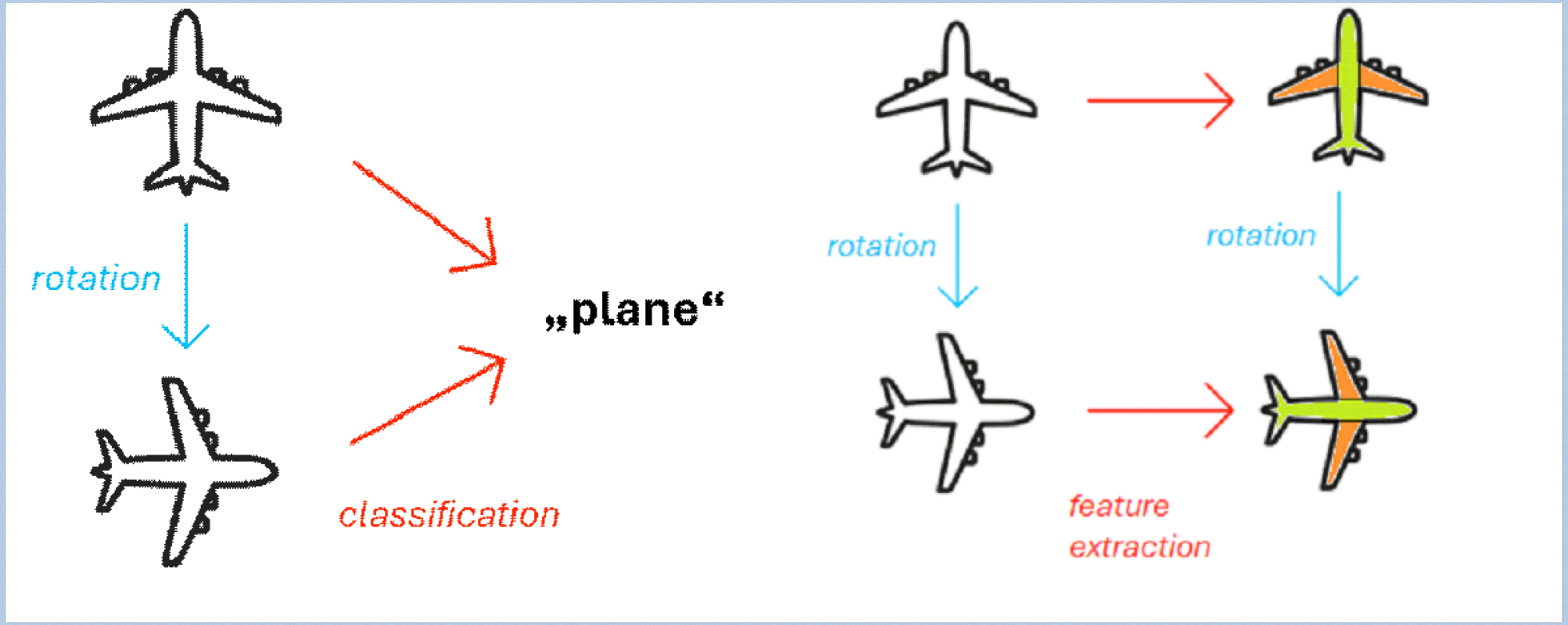
- Published in the Proceedings of the 33rd International Conference on Machine Learning (ICML), New York, USA, 2016.
- **Cited by ~2500** (Google Scholar).
- CNNs:
  Pros: Weight sharing (translation equivariance).
  Limitations: only exploit translation.
- G-CNNs:
  Pros: exploit larger symmetry groups (***translation, rotation, reflection***).
- G-CNNs benefits:
  - reduces sample complexity
  - increases expressive capacity,
  - no parameter increase.



Joan T. Richtsmeier

Source: T. S. Cohen and M. Welling, “Group equivariant convolutional networks”, 2016.

Equivariance vs Invariance



Tin Barisin

- Equivariance:
  Symmetry preserved through layers.

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

  Equivariance maintains feature relationships.
  $T(gh) = T(g)T(h)$

  Good **inductive bias** for deep networks , aids **generalization.**

- Invariance:
  Special case of equivariance where output is unchanged by transformation.

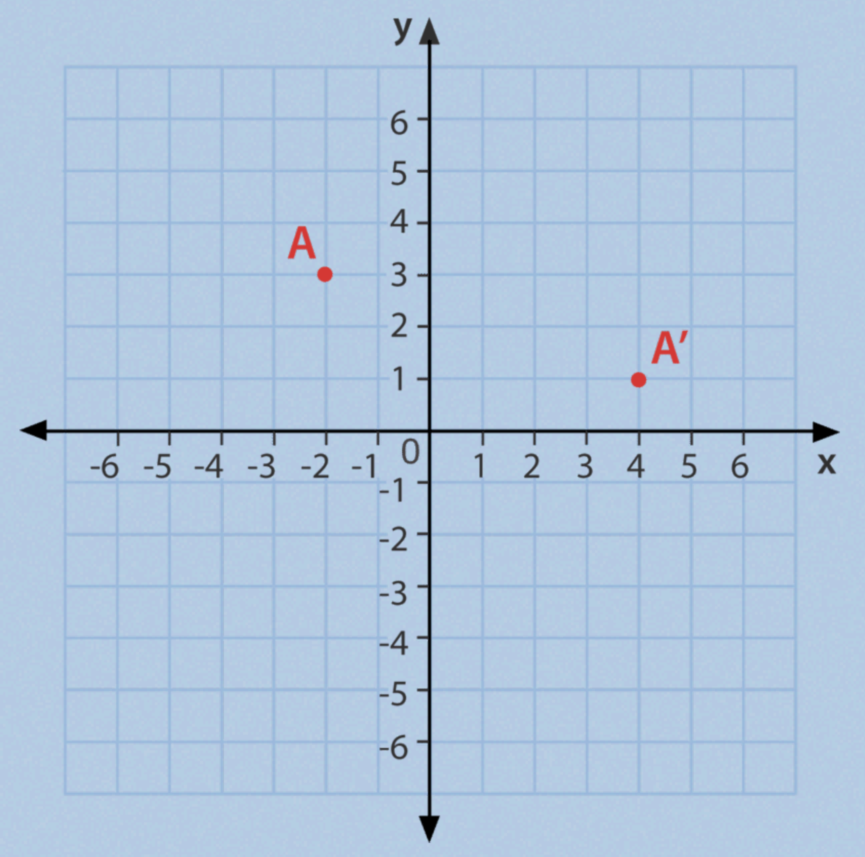
  $$\Phi(T_g x) = \Phi(x)$$

  Loses spatial configuration.

Source: T. S. Cohen and M. Welling, “Group equivariant convolutional networks”, 2016.

Mathematical Framework: Symmetry Groups

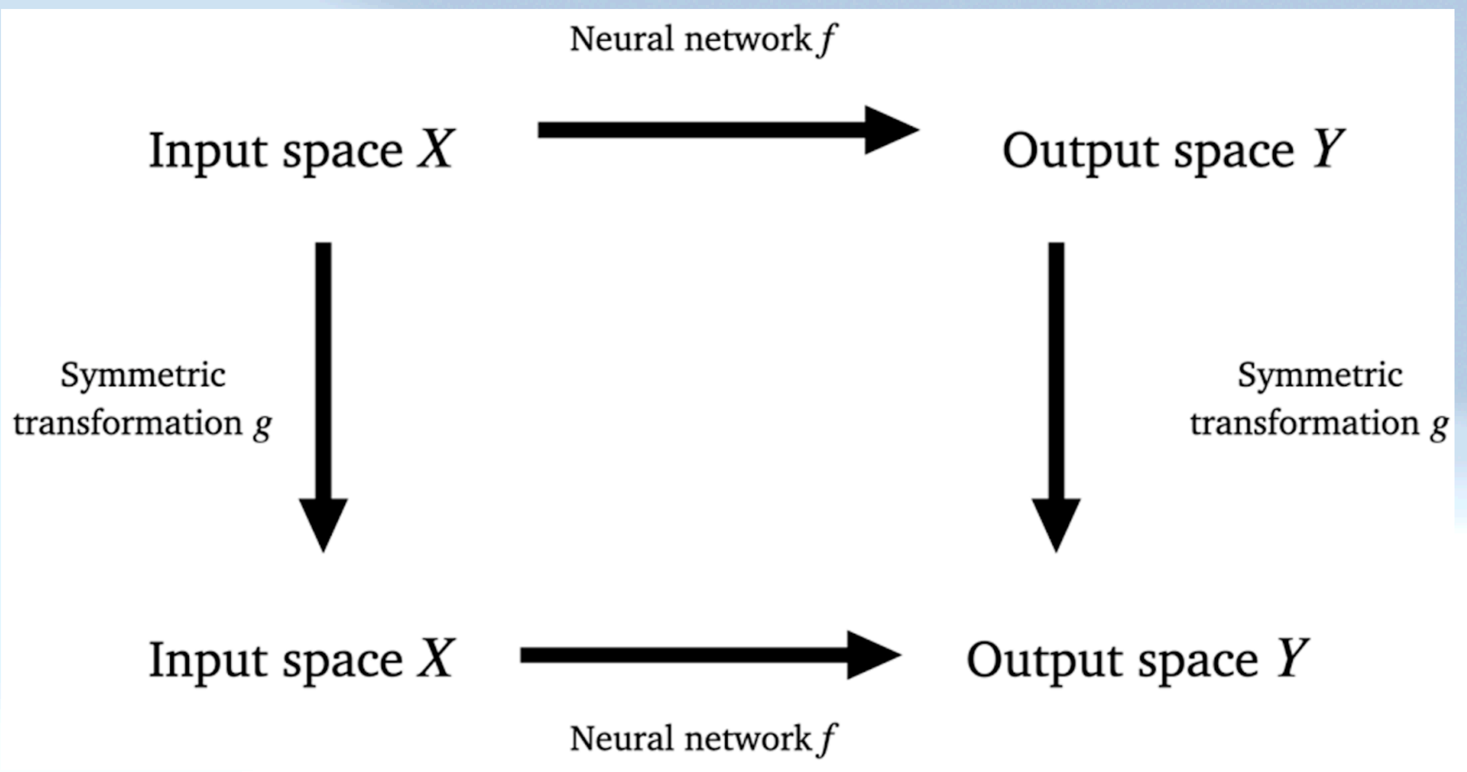
- Symmetry Group: **Transformations** leaving an object **invariant** (closure, inverse, identity).



- Example: 2D Integer Translations $\mathbb{Z}^2$
  Group operation: addition.
  Basic example of group structure for convolution properties.

- Feature Map Transformation:
  $[L_g f](x) = [f \circ g^{-1}](x) = f(g^{-1}x)$

  How a transformation g acts on a feature map f.
  $L_g L_h = L_{gh}$



Henry Kvinge

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Group p4

- $p4$: translations + 90° rotations
  Parameterized by three integers: (r, u, v)
  $x(u', v')$ points being acted upon.

$$g(r,u,v)\cdot x \simeq \begin{bmatrix}\cos(r\pi/2) & -\sin(r\pi/2) & u\\ \sin(r\pi/2) & \cos(r\pi/2) & v\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}u'\\ v'\\ 1\end{bmatrix}$$

where $0 \le r < 4$ and $(u, v) \in \mathbb{Z}^2$.

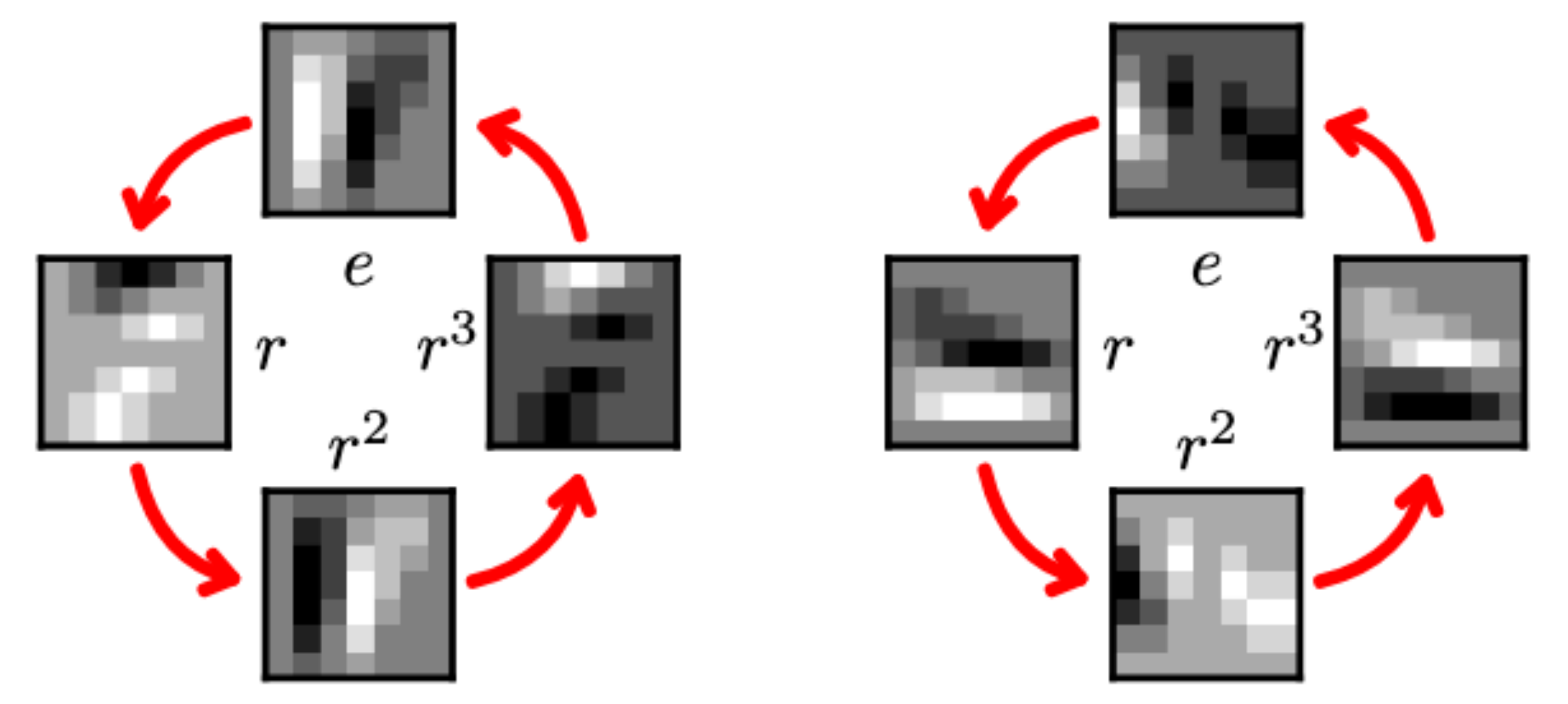


Figure 1. A p4 feature map and its rotation by r.

T.S. Cohen and M. Welling.

- Action on pixel coordinates and group operation: **matrix multiplication.**

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Group p4m

- $p4m$: translations + reflections + 90° rotations.
  Parameterized by 4 integers: (m, r, u, v)
  $x(u', v')$ points being acted upon.

$$g(m, r, u, v) \cdot x \simeq \begin{bmatrix} (-1)^m \cos(\frac{r\pi}{2}) & -(-1)^m \sin(\frac{r\pi}{2}) & u \\ \sin(\frac{r\pi}{2}) & \cos(\frac{r\pi}{2}) & v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix}$$

where $m \in \{0,1\}$, $0 \leq r < 4$ and $(u, v) \in \mathbb{Z}^2$.

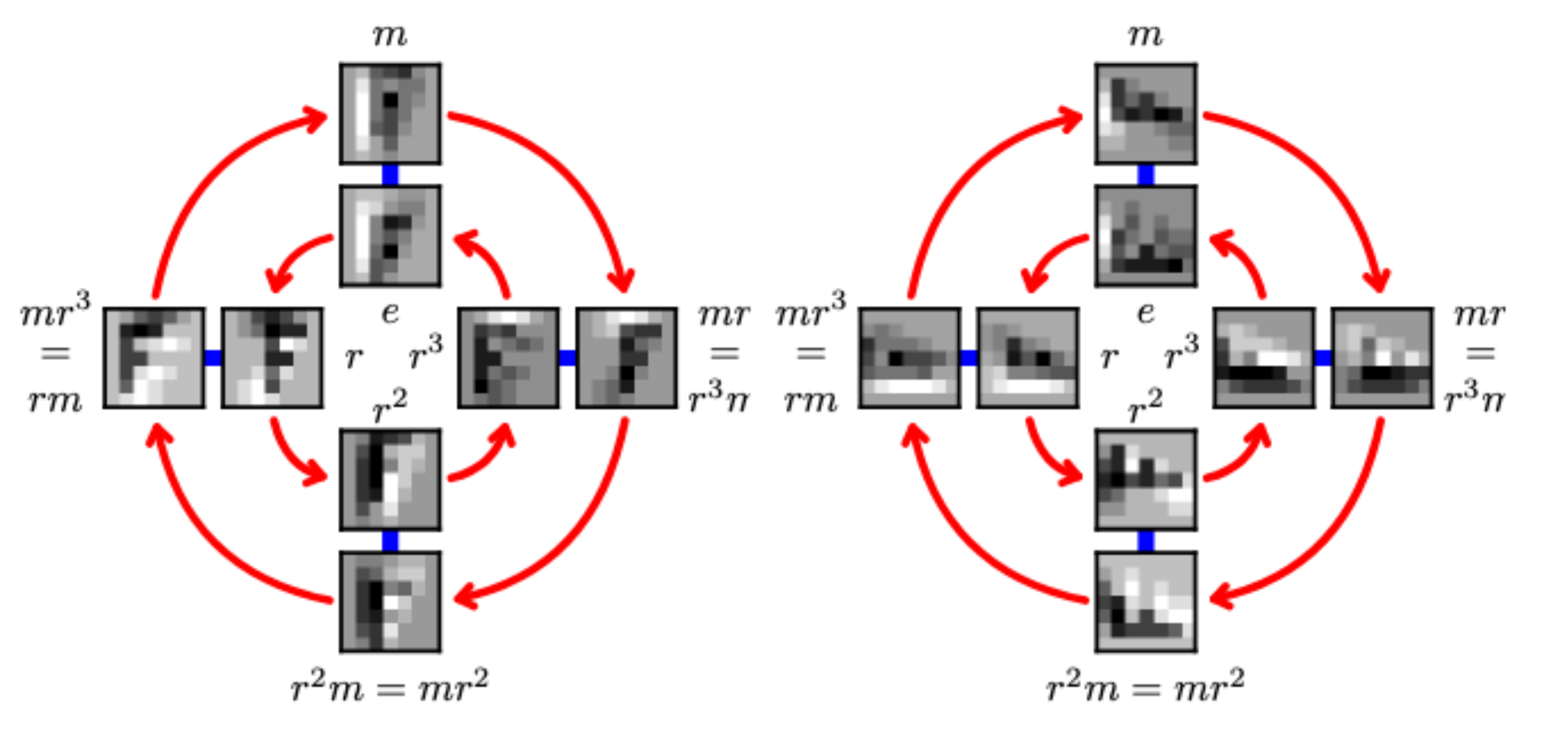


Figure 2. A p4m feature map and its rotation by r.

T.S. Cohen and M. Welling.

- Action on pixel coordinates and group operation: **matrix multiplication.**

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Equivariance of CNNs

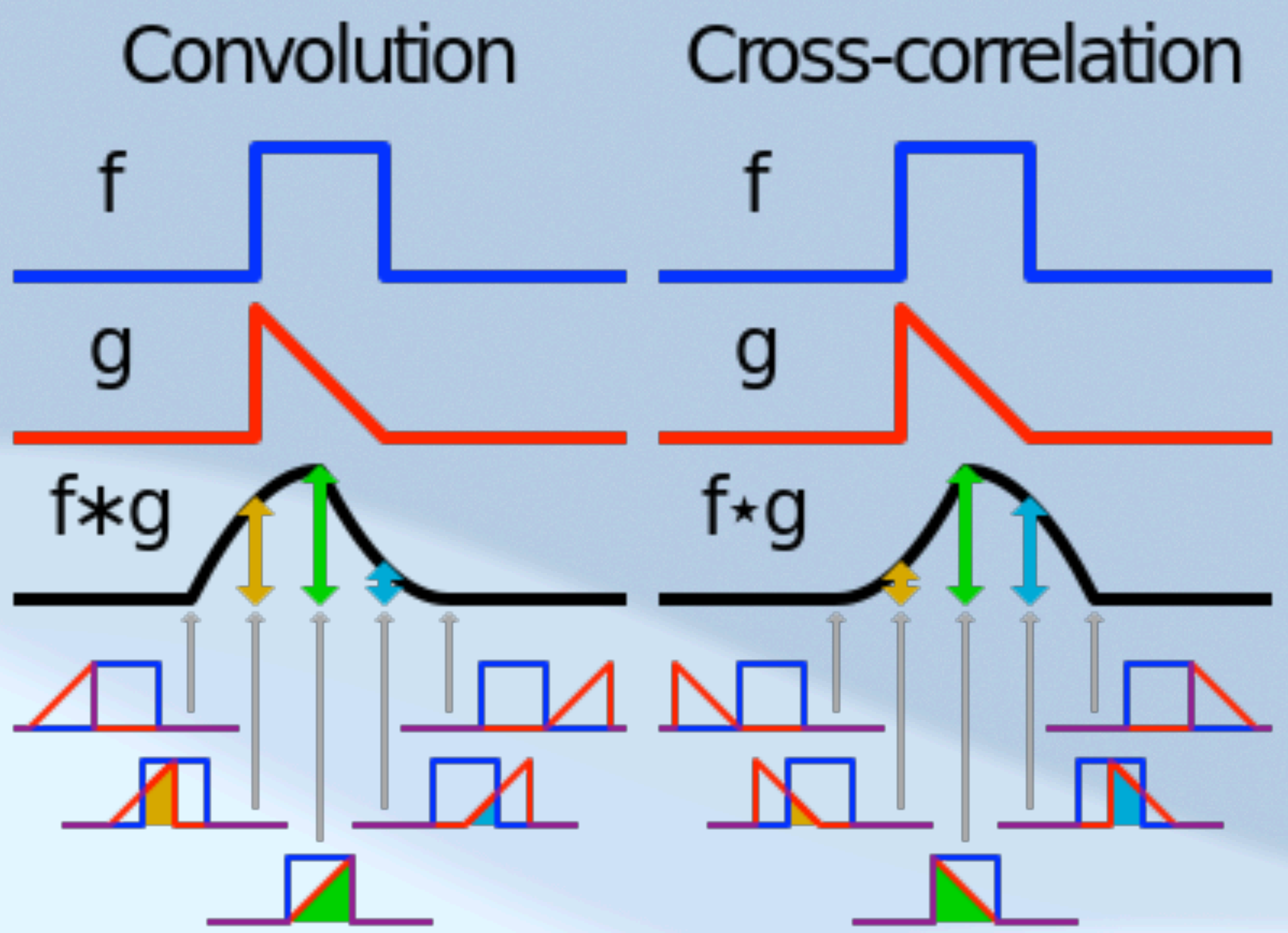
- Before G-CNNs, let's understand standard CNNs' symmetry properties.

- Standard CNNs take stacks of feature maps $f: \mathbb{Z}^2 \rightarrow \mathbb{R}^{K^l}$ as input.

- **Correlation**:

  Convolves with K^{l+1} filters $\psi^i : \mathbb{Z}^2 \rightarrow \mathbb{R}^{K^l}$

  Defined as $[f \star \psi^i](x) = \sum_{y \in \mathbb{Z}^2} \sum_{k=1}^{K^l} f_k(y)\, \psi_k^i\,(y - x)$



Cmglee, Wikimedia Commons.

We will use correlation in forward pass and refer to it generically as convolution.

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Equivariance of CNNs

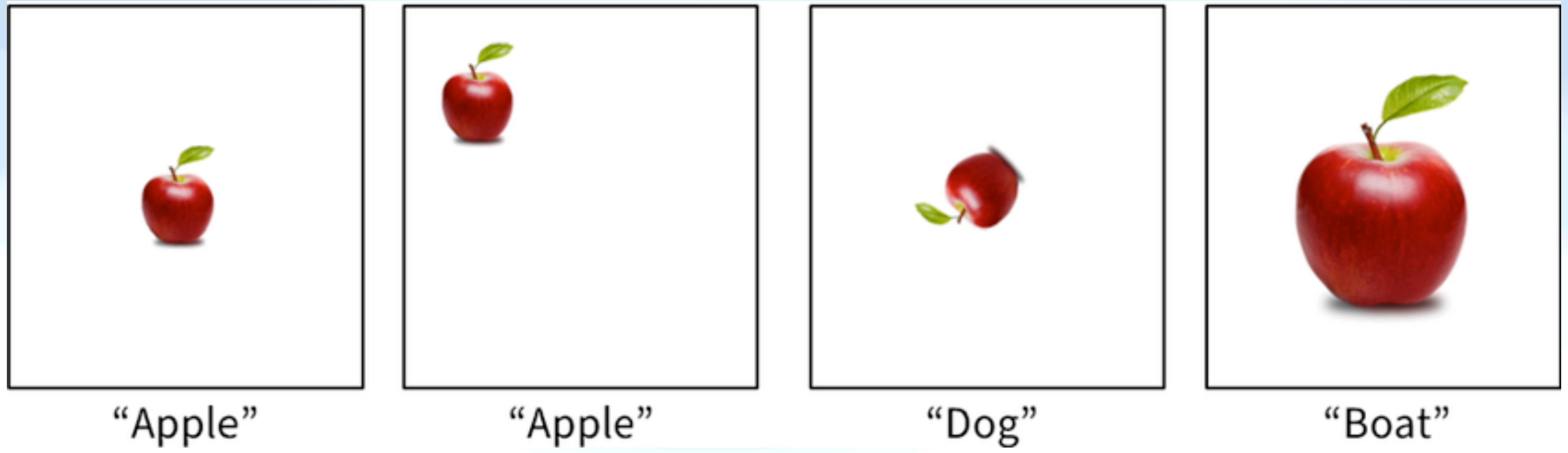
- Translation equivariance:
  - Correlation and Translation commute.
  - Correlation is an equivariant map for the translation group $(\mathbb{Z}^2)$

$[L_t f] \star \psi = L_t[f \star \psi]$: "Translation followed by correlation = correlation followed by translation".

- Lack of rotation equivariance: Standard planar correlation is **not equivariant to rotations.**

$$[[L_r f] \star \psi(x) = L_r[f \star [L_{r^{-1}}\psi]](x)$$

  - This means a standard CNN must learn *rotated copies* of the same filter to achieve rotational equivariance.



Journal of Mathematical Imaging and Vision

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

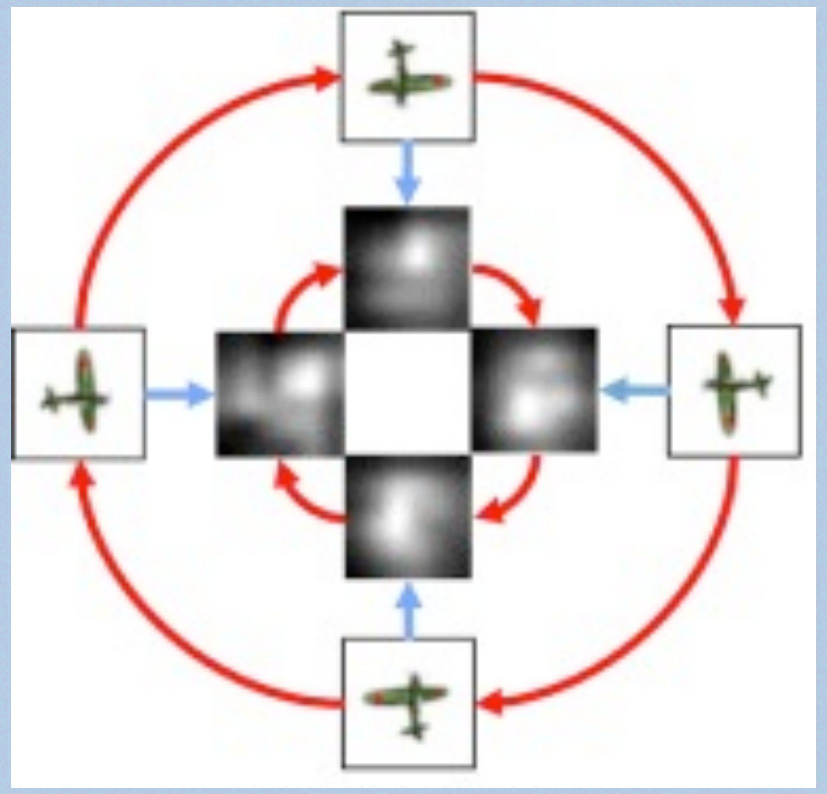
- **First Layer**
  Core idea: Replace the “shift” in standard convolution with a more general transformation from group G.

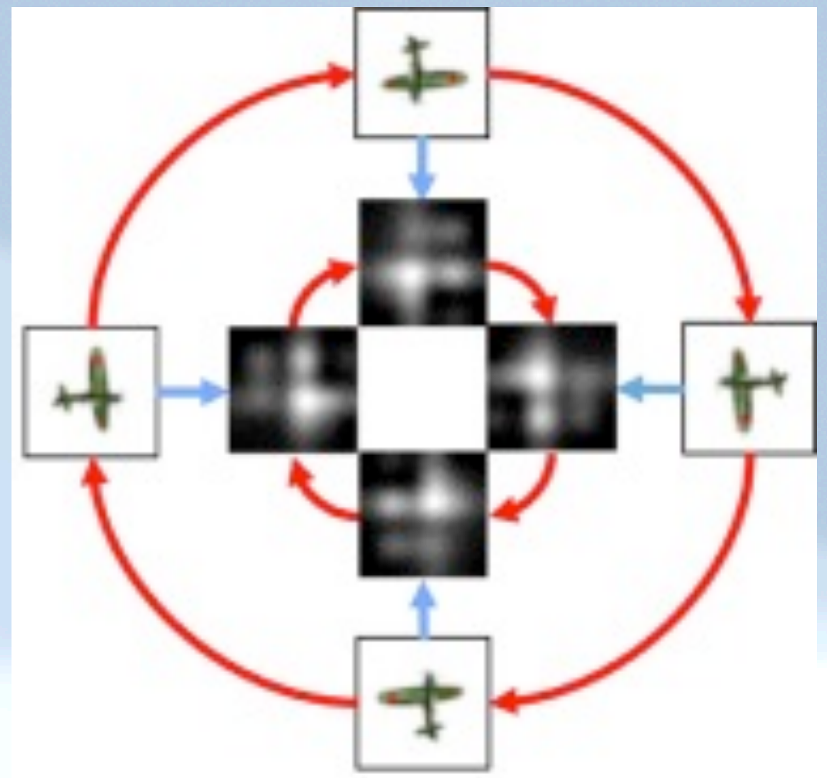
  First Layer: Input $f : \mathbb{Z}^2 \rightarrow \mathbb{R}^K$, Filter $\psi : \mathbb{Z}^2 \rightarrow \mathbb{R}^K$, Output $f \star \psi : G \rightarrow \mathbb{R}^{K'}$.

$$[f \star \psi](g) = \sum_{y \in \mathbb{Z}^2} \sum_k f_k(y)\, \psi_k\,(g^{-1}y)$$

  Lifts the planar image data to the group domain.

- Equivariance in first layer:
  $[L_u f] \star \psi = L_u[f \star \psi]$
  $L_u f$ transforms the input image (function on $\mathbb{Z}^2$).
  $L_u[f\psi]$ transforms the output feature map (function on G).





Deep Rotation Equivariant Network
https://www.sciencedirect.com/science/article/abs/pii/S0925231218301644

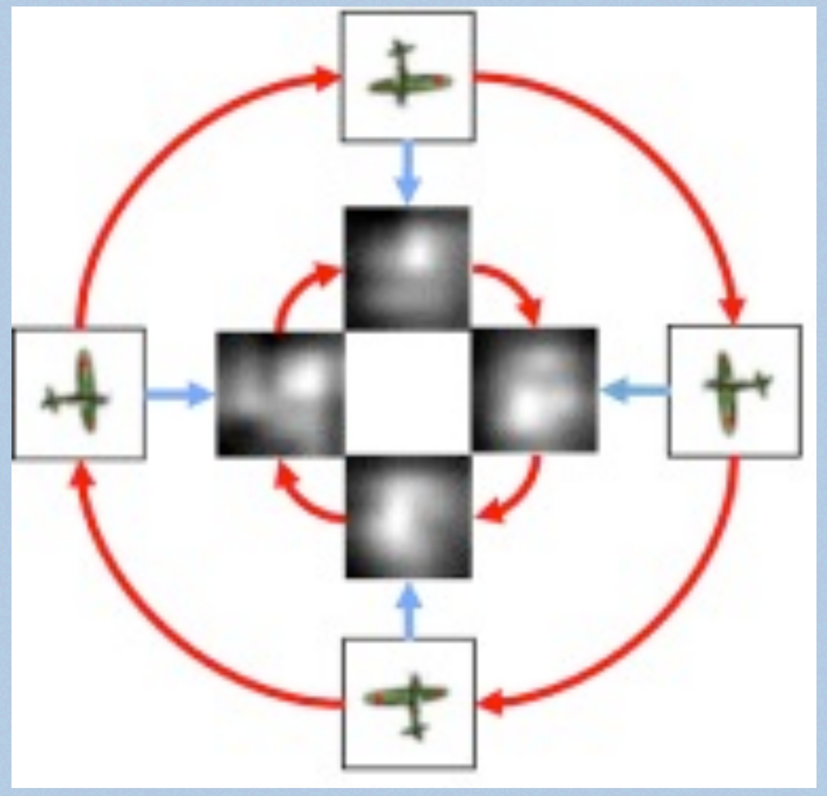
Source: T. S. Cohen and M. Welling, “Group equivariant convolutional networks”, 2016.

G-Equivariant Convolution

- **Subsequent Layers**
  Both input f and filter ψ are now functions on the group G.
  Output $f \star \psi$ remains a function on G.

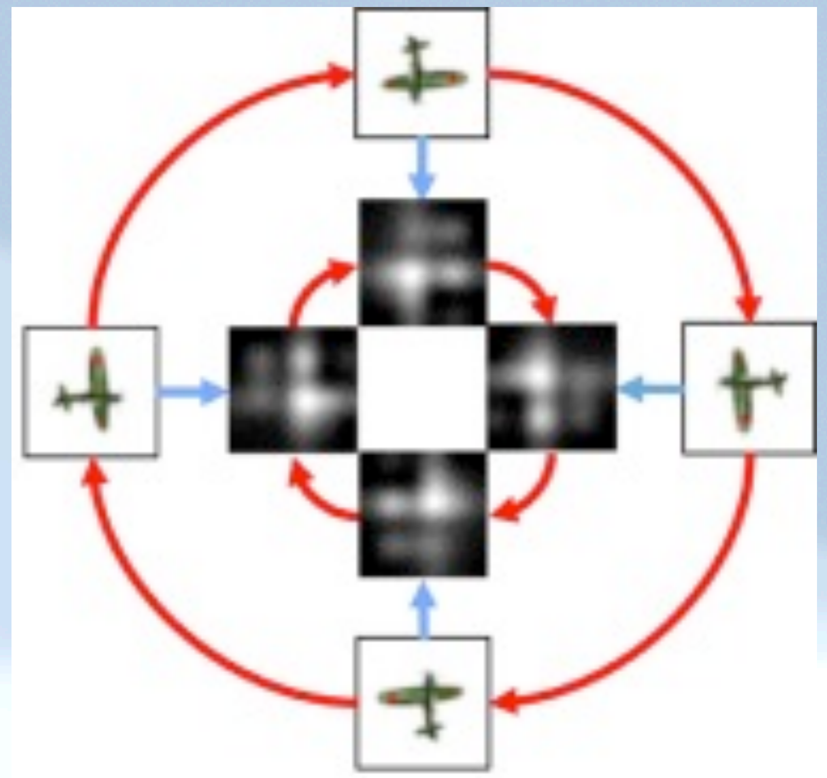
  Input $f : \mathbb{G} \to \mathbb{R}^K$, Filter $\psi : G \to \mathbb{R}^K$, Output $f * \psi : G \to \mathbb{R}^{K'}$.

$$[f * \psi](g) = \sum_{h \in G} \sum_k f_k(h) \psi_k(g^{-1}h)$$



- **Equivariance Property:** G-correlation preserves transformation properties.

$$[[L_u f] * \psi](g) = L_u [f * \psi](g)$$



- **Consistency:** Bias terms, batch norm applied per G-feature map to preserve equivariance.

  Allows use in residual blocks and highway networks.

Deep Rotation Equivariant Network
https://www.sciencedirect.com/science/article/abs/pii/S0925231218301644

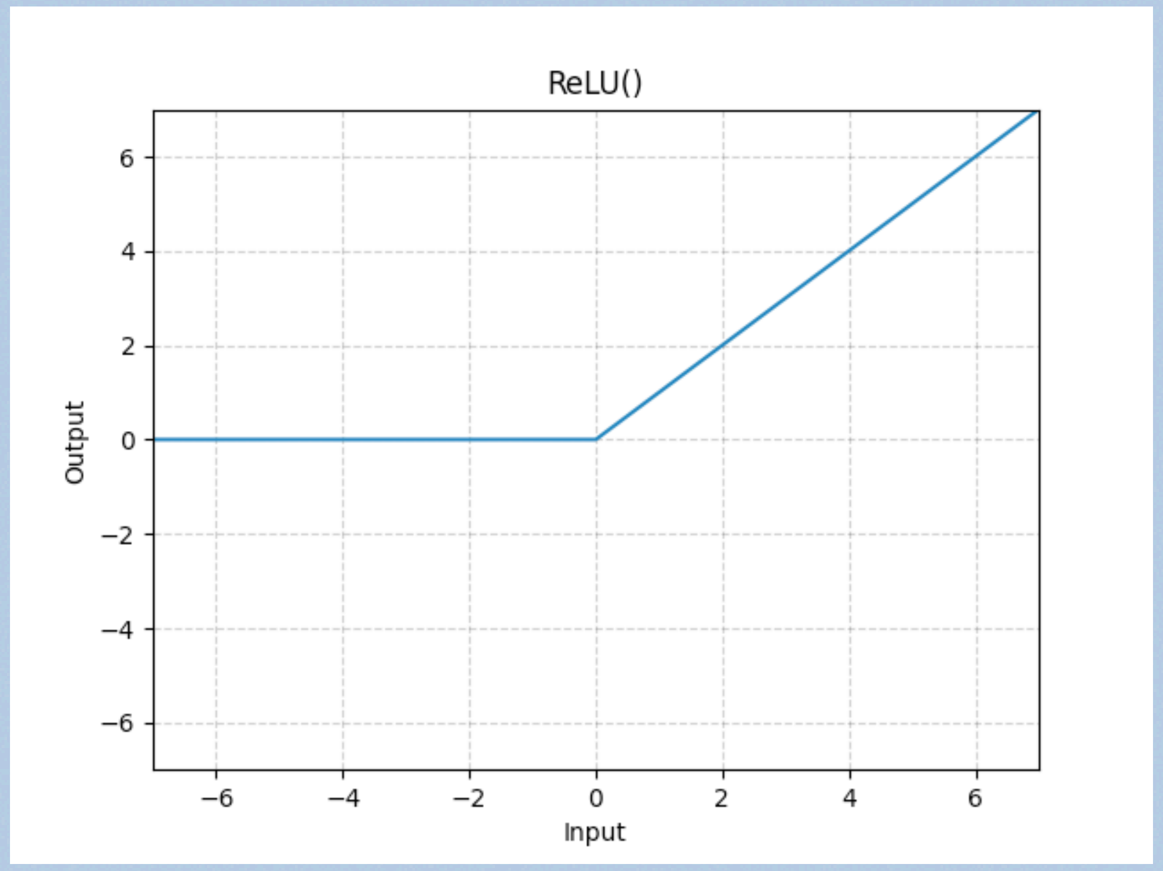
Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Other Equivariant Layers

- ***Pointwise Nonlinearities*** ($\nu : \mathbb{R} \to \mathbb{R}$):
  Applied element-wise to a feature map $f : G \to \mathbb{R}$

  Composition operator: $C_\nu f(g) = [\nu \circ f](g) = \nu(f(g))$

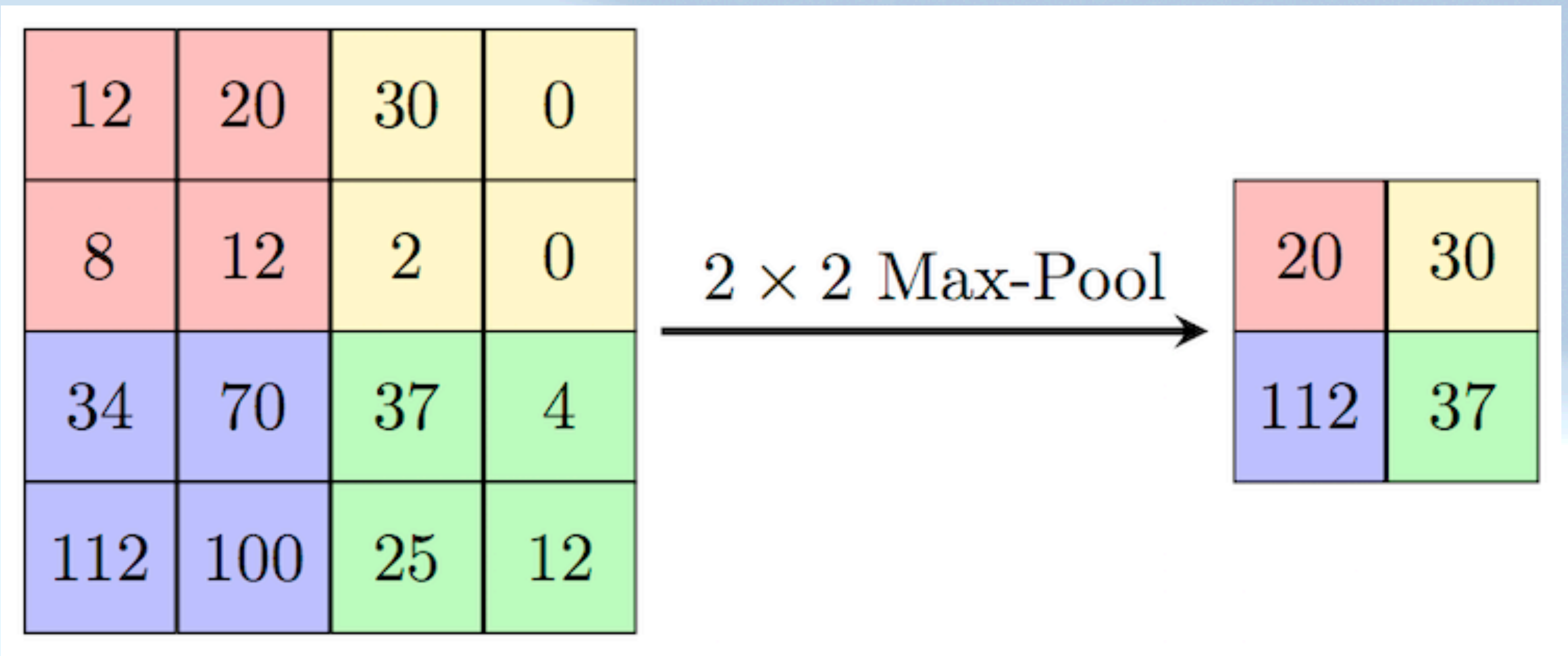
  $C_\nu L_h f = \nu \circ [f \circ h^{-1}] = [\nu \circ f] \circ h^{-1} = L_h C_\nu f.$



PyTorch - ReLU

- ***Pooling (Generalization):***
  Max pooling: $P f(g) = \max_{k \in gU} f(k)$

  Commutes with group action: $PL_h = L_h P$

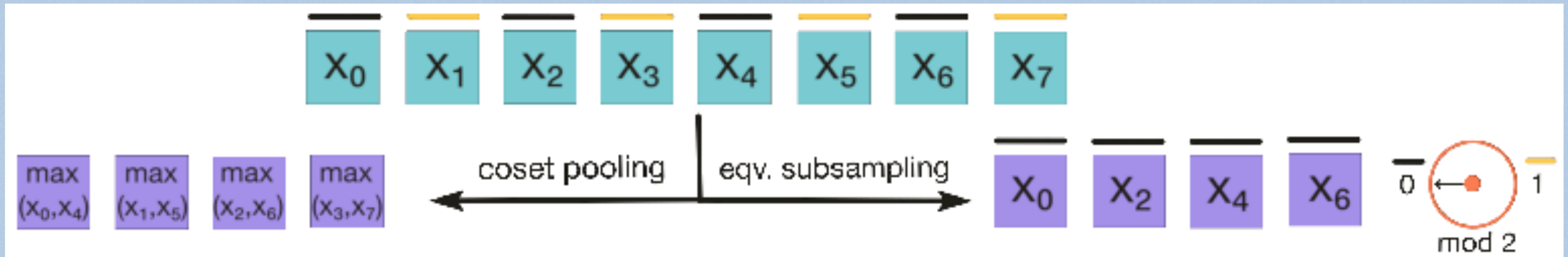


FirelordPhoenix

Source: T. S. Cohen and M. Welling, “Group equivariant convolutional networks”, 2016.

Other Equivariant Layers

- ***Subsampling ("Pooling with stride")***:
  Full G equivariance performed by subsampling on a subgroup $H \subset G$
  Result: Feature map equivariant to H (not necessarily G).



Group Equivariant Subsampling

- ***Coset Pooling***:
  Pooling over **cosets** of a subgroup H (i.e., pooling region U is itself a subgroup H).
  Result: Feature map invariant to right-action of H, functions on G/H.
  Example: p4 pooling over rotations results in a feature map on $\mathbb{Z}^2 \cong p4/R$

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

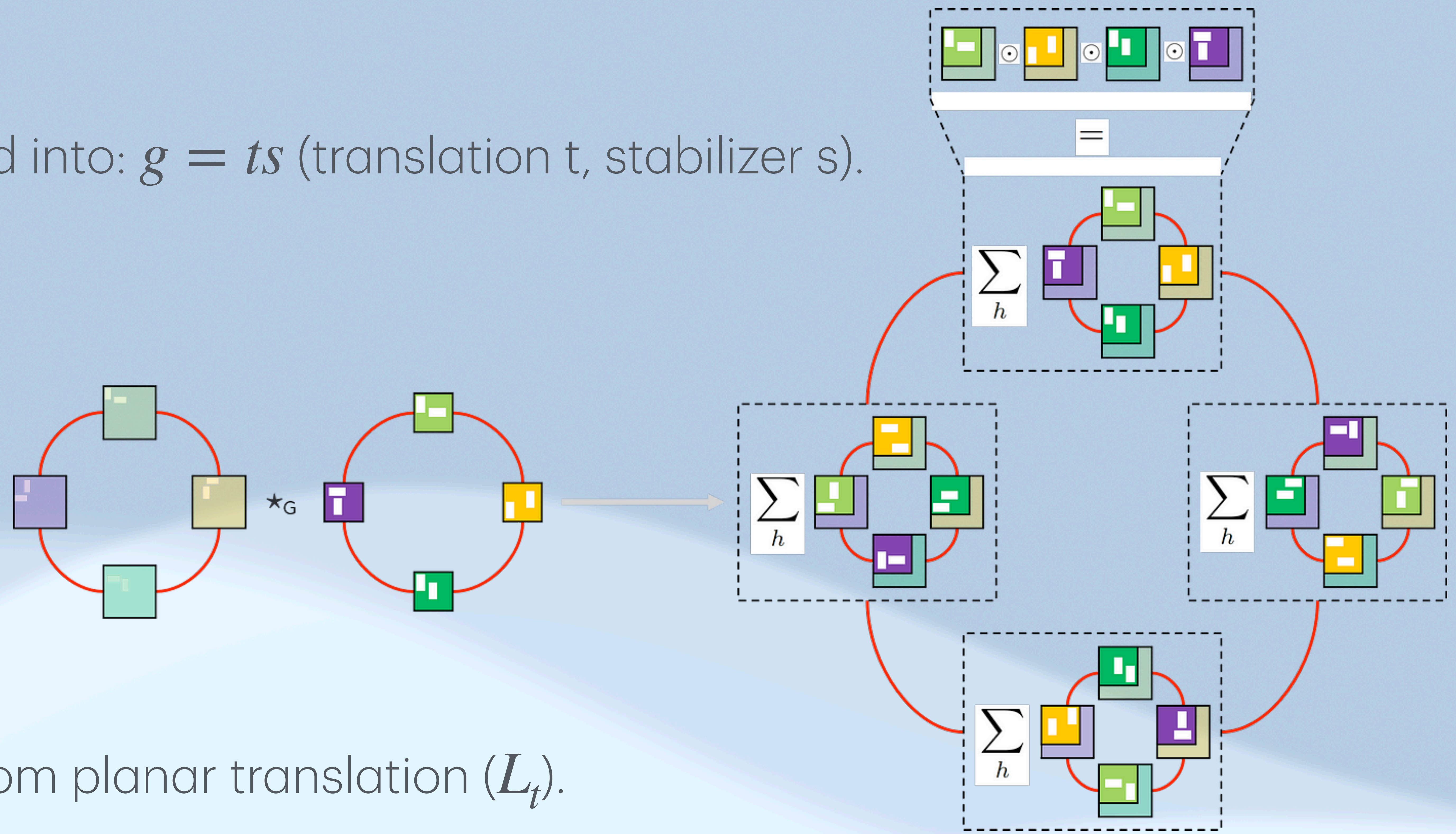
Efficient Implementation: Overview

- Goal: Leverage advances in fast planar convolution (FFTs, Winograd).

- **For "Split Groups" (p4, p4m):**
  Any transformation $g \in G$ can be decomposed into: $g = ts$ (translation t, stabilizer s).

- Key insight: G-correlation can be rewritten:

$$f \star \psi(ts) = \sum_{h \in X} \sum_{k} f_k(h)\, L_t \left[L_s\, \psi_k(h) \right]$$



Casper van Engelenburg

This separates the group transformation (L_s) from planar translation (L_t).

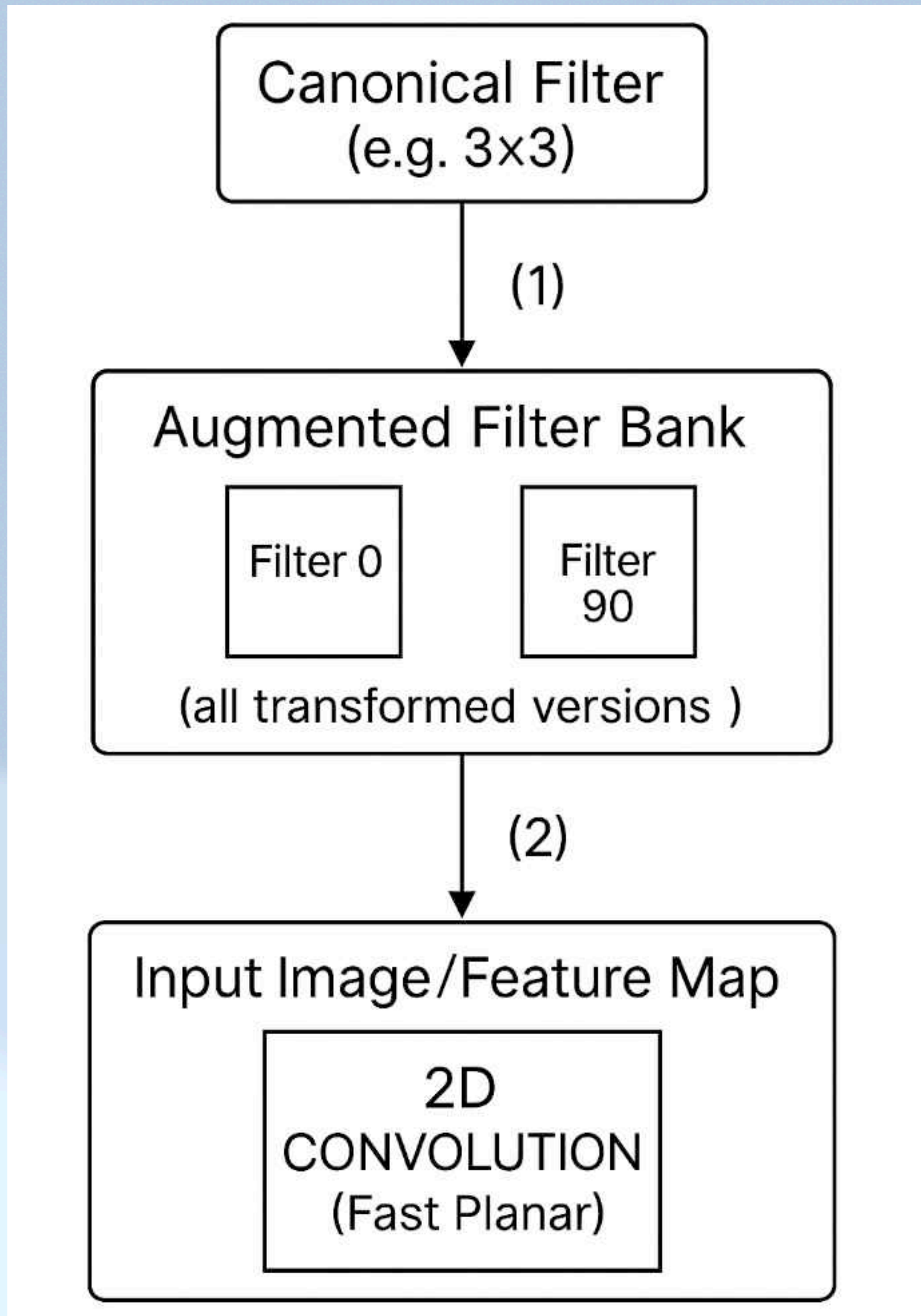
Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Efficient Implementation: Two step process

- Two step process:

1. Filter Transformation $(L_s \psi)$:
Compute all transformed filter versions (e.g., 4 rotations for p4, 8 rotations for p4m.
This is a permutation of filter entries, efficiently implemented on GPU.

2. Fast planar convolution:
Use this augmented filter bank (all transformed filters).
Computational cost comparable to standard planar conv with larger filter bank.



Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Experimental Results: Rotated MNIST

- **Dataset:** Rotated MNIST (62,000 randomly rotated digits).

- Key Takeaways:
  - P4CNN achieved SOTA (2.28% error)
  - Significant improvement over baseline and previous SOTA
  - Premature invariance hurts performance (P4CNNRotationPooling vs P4CNN).

Network	Test Error (%)
Larochelle et al. (2007)	10.38 ± 0.27
Sohn & Lee (2012)	4.2
Schmidt & Roth (2012)	3.98
Z2CNN	5.03 ± 0.0020
P4CNNRotationPooling	3.21 ± 0.0012
P4CNN	**2.28 ± 0.0004**

Table 1. Error rates on rotated MNIST (with standard deviation under variation of the random seed).

T.S. Cohen and M. Welling.

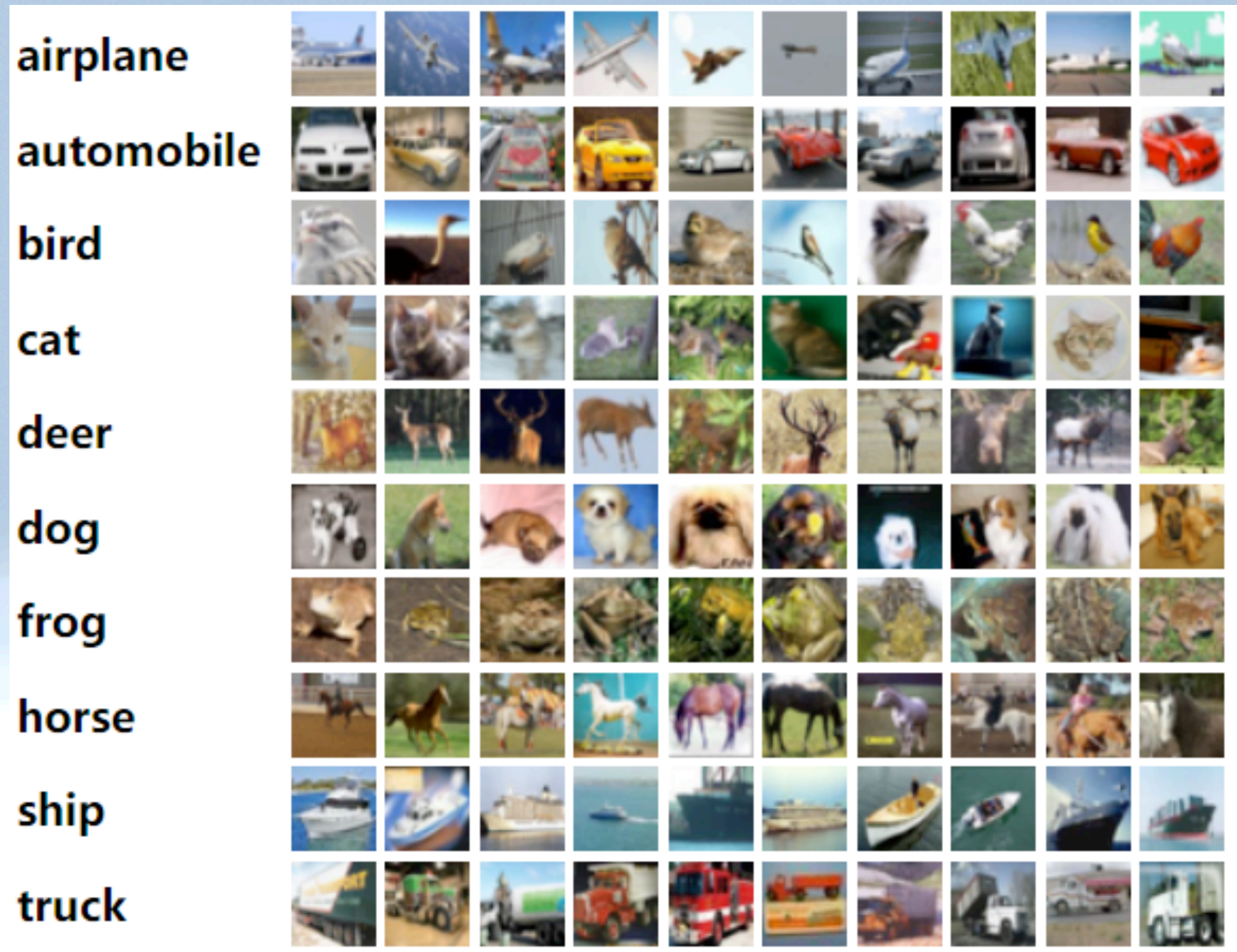
Kaggle: Rotated MNIST

Source: T. S. Cohen and M. Welling, “Group equivariant convolutional networks”, 2016.

Experimental Results: CIFAR-10

- **Dataset:** CIFAR-10 (60k 32x32 images, 10 classes).
  Tested on CIFAR10 (plain) and CIFAR10+ (augmented)

- Key Takeaways:
  - G-CNNs consistently improve results
  - Benefits even on non-symmetric datasets (CIFAR10).
  - P4m-ResNet: competitive SOTA (4.19% on CIFAR10+) with fewer parameters.



University of Toronto: CIFAR 10 dataset

Network	G	CIFAR10	CIFAR10+	Param.
All-CNN	$\mathbb{Z}^2$	9.44	8.86	1.37M
	$p4$	8.84	7.67	1.37M
	$p4m$	7.59	7.04	1.22M
ResNet44	$\mathbb{Z}^2$	9.45	5.61	2.64M
	$p4m$	**6.46**	**4.94**	2.62M

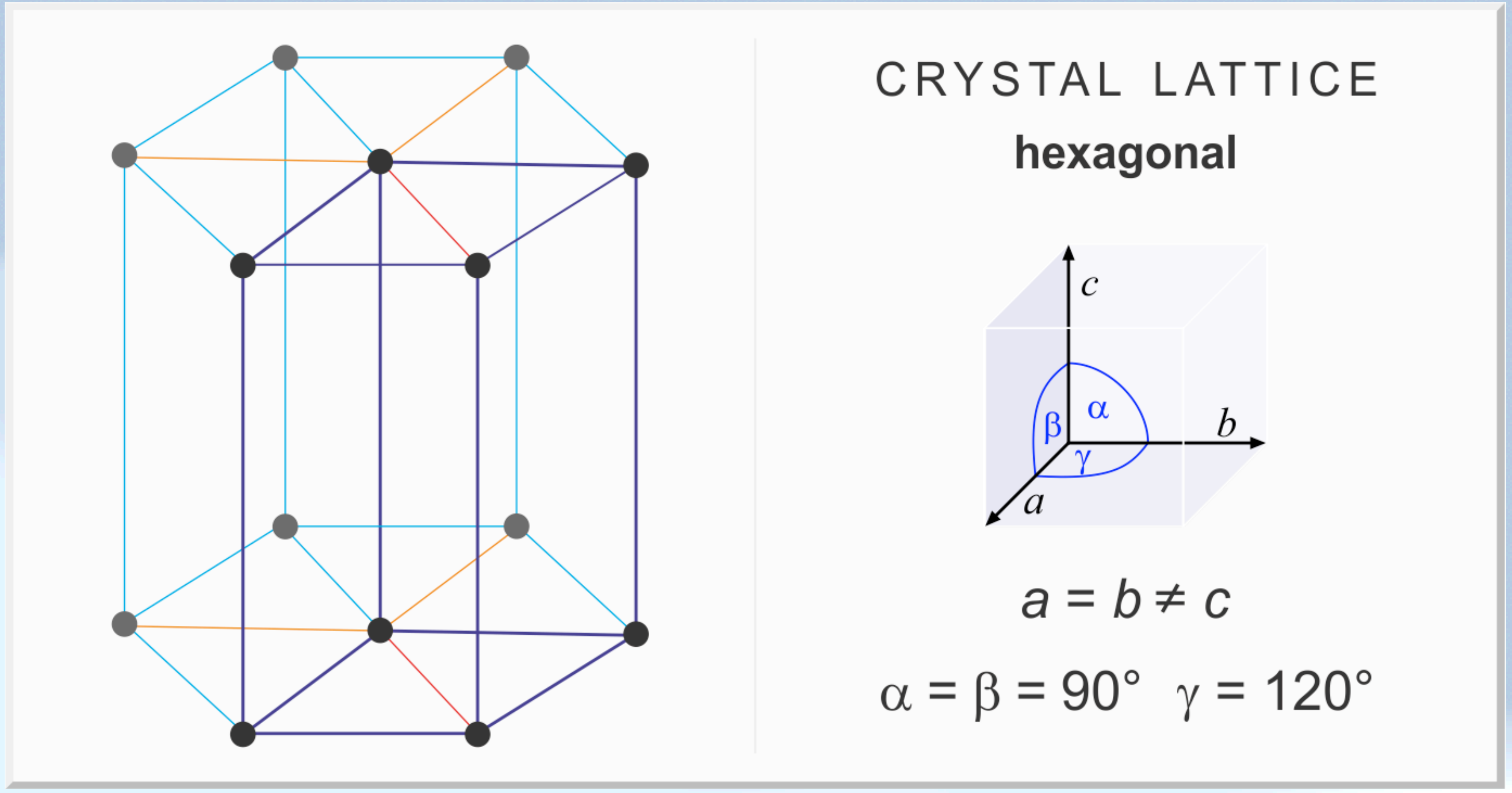
Table 2. Comparison of conventional (i.e. $\mathbb{Z}^2$), $p4$ and $p4m$ CNNs on CIFAR10 and augmented CIFAR10+. Test set error rates and number of parameters are reported.

T.S. Cohen and M. Welling.

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Discussion & Future Work

- Summary: G-convolutions are a drop-in replacement that consistently improves performance.
- Data Augmentation: Still beneficial for G-CNNs, especially with larger augmentation groups.
- Symmetry Not Strict: G-CNNs help even without perfect data symmetry.
- Future Directions:
  - Apply to other groups: hexagonal lattices, 3D space groups.
  - Theory is general; implementation needs group operation/index mapping.
  - Challenges: Continuous groups, very large groups.



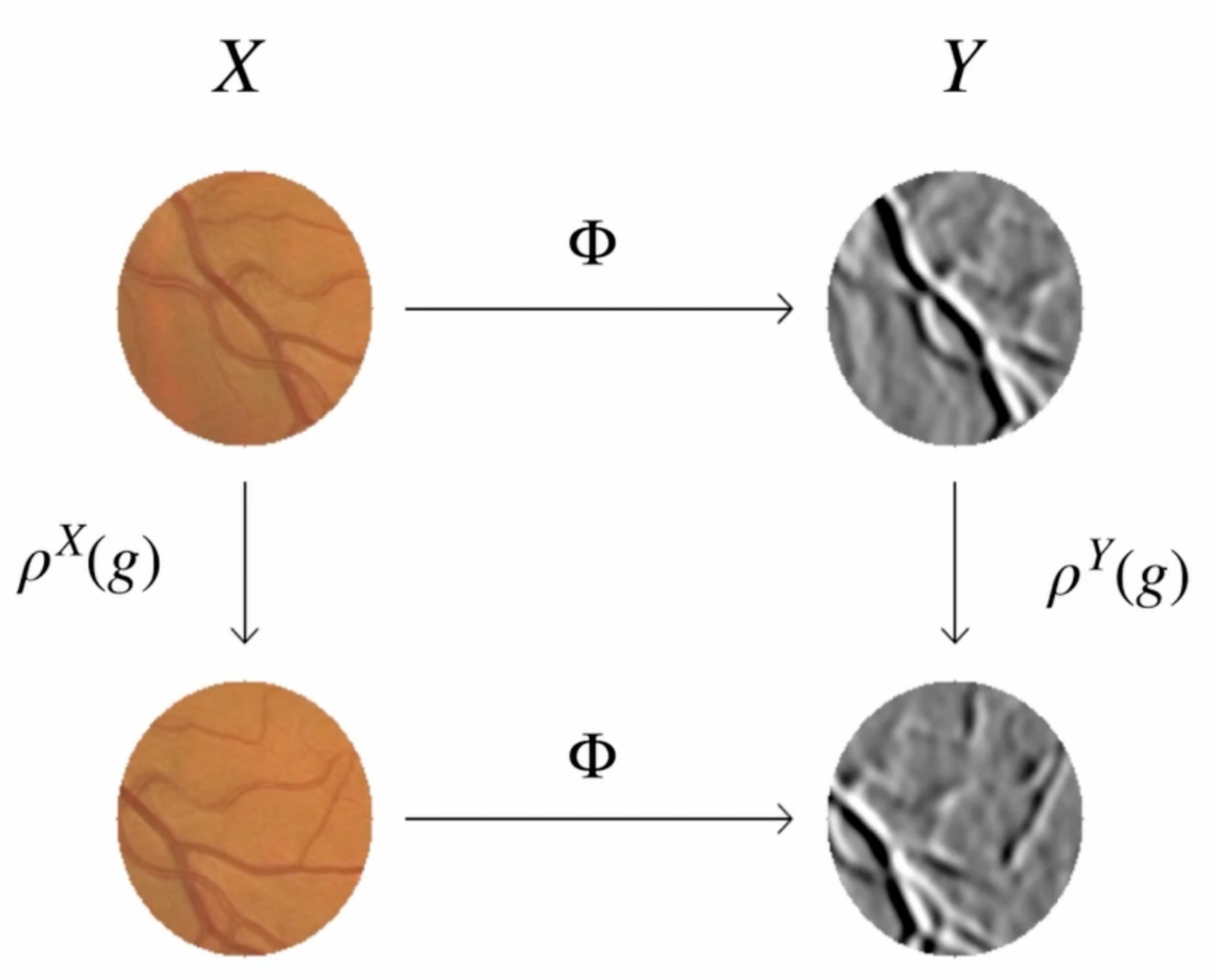
JonathanXVI, Selenium: Importance

- Broader Philosophy: "Structured representations" enhance neural nets' abstract similarity recognition

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Conclusion

- **Introduced G-CNNs:** Generalization of CNNs exploiting symmetries.
- **Proven Equivariance:** All G-CNN layer types are equivariant.
- **Practicality:** Drop-in replacement for spatial convolutions, improving performance without tuning.



Erik Bekkers

- ***Core Benefits***:
  - Increased expressive capacity
  - No parameter increase
  - Achieved state-of-the-art results (Rotated MNIST, CIFAR10)

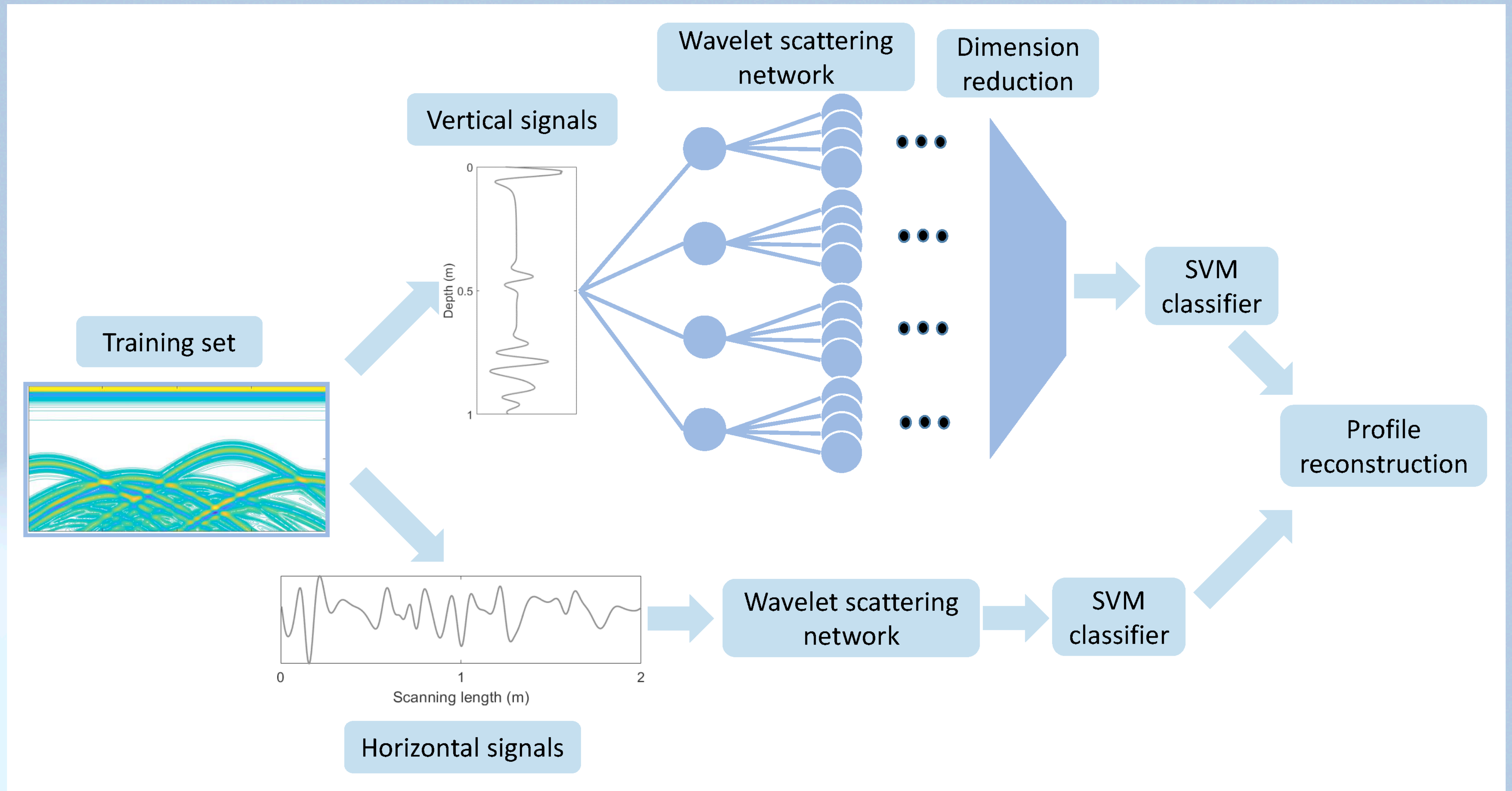
Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

Related Literature

- ***Invariant representations***:
  - Pose normalization (Lowe, Jaderberg).
  - Group averaging (Reisert, Manay).

- ***Scattering Networks***:
  - Wavelet convs, nonlinearities, group averaging (Bruna & Mallat).
  - Extended to translation, rotation, scaling for object/ texture (Sifre & Mallat)Equivariant

- ***Representations (Learning/Construction):***
  - Transforming autoencoders (Hinton).
  - Equivariant Boltzmann machines (Kivinen & Williams).
  - Rotation symmetry in CNNs (Dieleman).
  - Deep Symmetry Networks (Gens & Domingos).



Wavelet Scattering Network-Based Machine Learning for Ground Penetrating Radar Imaging: Application in Pipeline Identification

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.

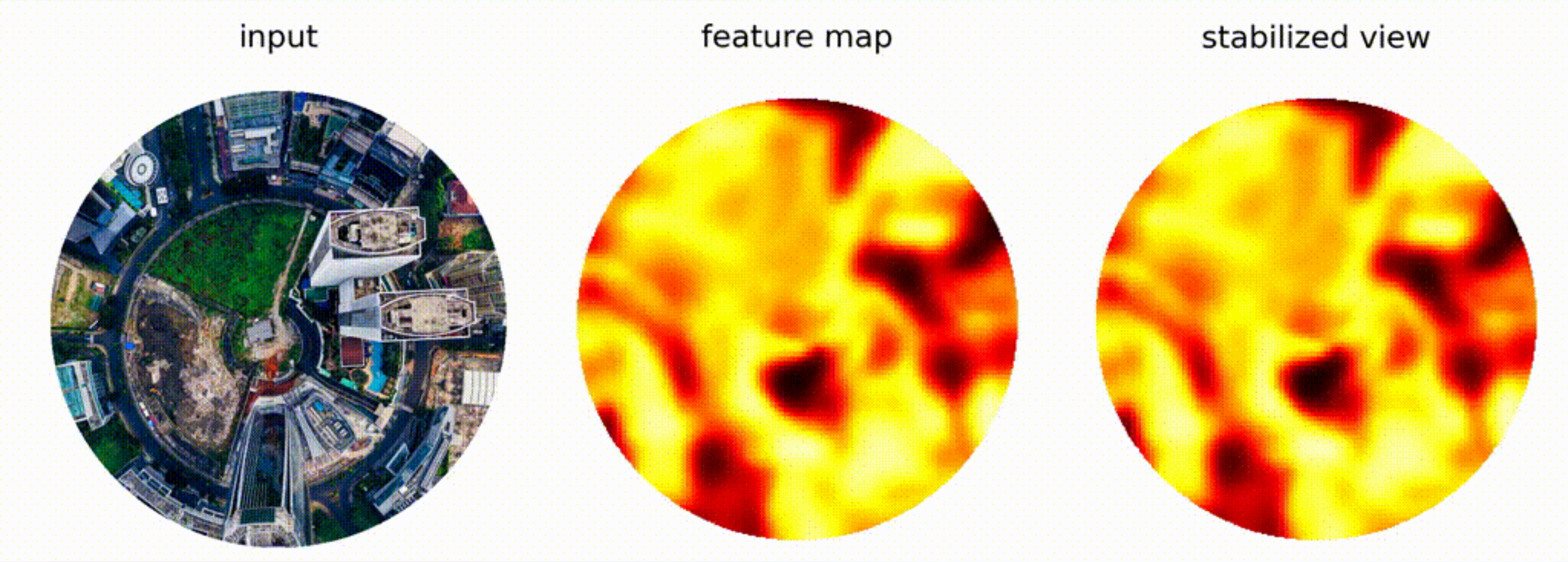
- **Questions?**

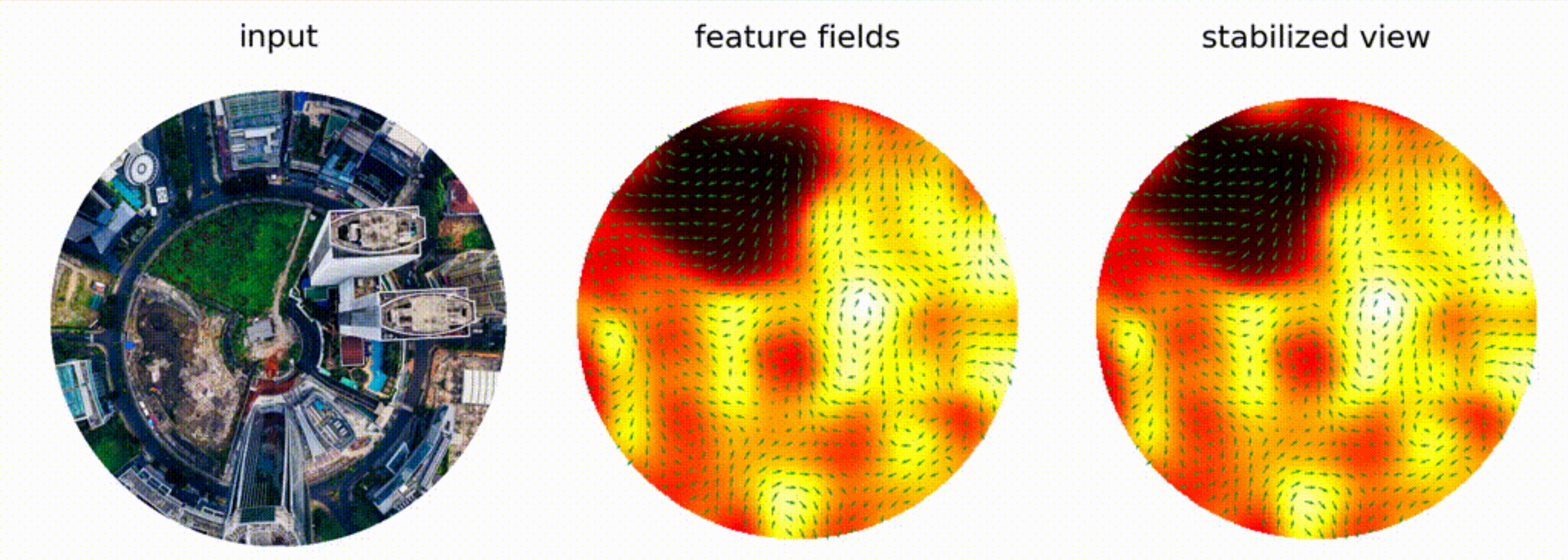
* Practical implications of equivariance vs. invariance?

* G-CNNs on datasets with other symmetries (e.g., scale)?

* Challenges for continuous groups?

* Other beneficial applications of group symmetries?





Figures source: https://github.com/QUVA-Lab/e2cnn

Source: T. S. Cohen and M. Welling, "Group equivariant convolutional networks", 2016.