Equivariant Neural Networks for Inverse Problems

Matthias J. Ehrhardt

Department of Mathematical Sciences, University of Bath, UK

September 29, 2021

Joint work with:
F. Sherry, C. Etmann, C.-B. Schönlieb (all Cambridge, UK),
E. Celledoni, B. Owren (both NTNU, Norway)

The Leverhulme Trust

Engineering and Physical Sciences Research Council

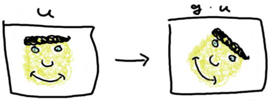
THE FARADAY INSTITUTION

Outline

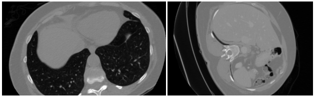
1) Inverse Problems
and Machine Learning

$$x^+ = \Psi_\theta(x - \tau \nabla D(x))$$

2) Equivariance



3) Numerical Results
for CT and MRI



E. Celledoni, M. J. Ehrhardt, C. Etmann, B. Owren, C.-B. Schönlieb, and F. Sherry, "Equivariant neural networks for inverse problems," Inverse Problems, vol. 37, no. 8, p. 085006, 2021.

Inverse Problems and Machine Learning

Inverse problems

$$Au = b$$

u : desired solution

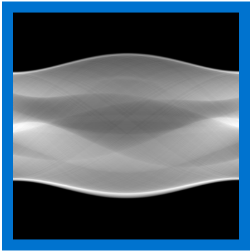
b : observed data

A : mathematical model

Goal: recover u given b

- CT: Radon / X-ray transform $Au(L) = \int_L u(x)dx$

$\rightarrow$



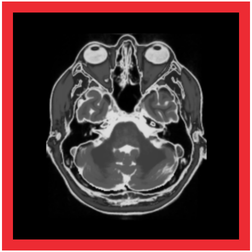
Inverse problems

$$Au = b$$

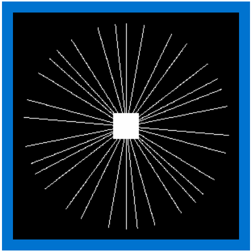
- u : desired solution
- b : observed data
- A : mathematical model

Goal: recover u given b

- MRI: Fourier transform $Au(k) = \int u(x) \exp(-ikx) dx$



$\longrightarrow$



Variational regularization

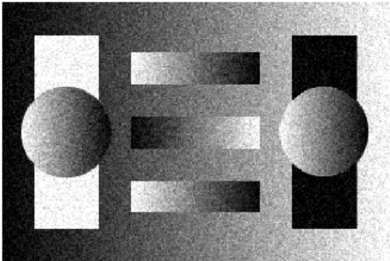
Approximate a solution u^* of $Au = b$ via

$$\hat{u} \in \arg\min_u \left\{ \mathcal{D}(u) + \lambda \mathcal{R}(u) \right\}$$

$\mathcal{D}$ measures **fidelity** between Au and b, related to noise statistics

$\mathcal{R}$ **regularizer** penalizes unwanted features and ensures stability;
e.g. TV Rudin, Osher, Fatimi '92 $\mathcal{R}(u) = \|\nabla u\|_1$,
TGV Bredies, Kunisch, Pock '10 $\mathcal{R}(u) = \inf_v \|\nabla u - v\|_1 + \beta \|\nabla v\|_1$

$\lambda \geq 0$ **regularization parameter** balances fidelity and regularization



Algorithmic Solution

$$\hat{u} \in \arg\min_u \left\{ \mathcal{D}(u) + \lambda \mathcal{R}(u) \right\}$$

Proximal Gradient Descent (PGD) Beck and Teboulle '09

$$u^{k+1} = \mathrm{prox}_{\tau^k \lambda \mathcal{R}}(u^k - \tau^k \nabla \mathcal{D}(u^k))$$

Solution $\Phi(b) := \lim_{k\to\infty} u^k$.
Choose τ^k, λ: $\Phi(b) = \hat{u} \to u^*$ if $\lambda \to 0$

Proximal operator Moreau '62

$$\mathrm{prox}_f(z) := \arg\min_u \frac{1}{2}\|u - z\|^2 + f(u)$$

Algorithmic Solution

$$\hat{u} \in \arg\min_{u} \left\{ \mathcal{D}(u) + \lambda \mathcal{R}(u) \right\}$$

Proximal Gradient Descent (PGD) Beck and Teboulle '09

$$u^{k+1} = \mathrm{prox}_{\tau^k \lambda \mathcal{R}}(u^k - \tau^k \nabla \mathcal{D}(u^k))$$

Solution $\Phi(b) := \lim_{k\to\infty} u^k$.
Choose τ^k, λ: $\Phi(b) = \hat{u} \to u^*$ if $\lambda \to 0$

Proximal operator Moreau '62

$$\mathrm{prox}_f(z) := \arg\min_u \frac{1}{2}\|u - z\|^2 + f(u)$$

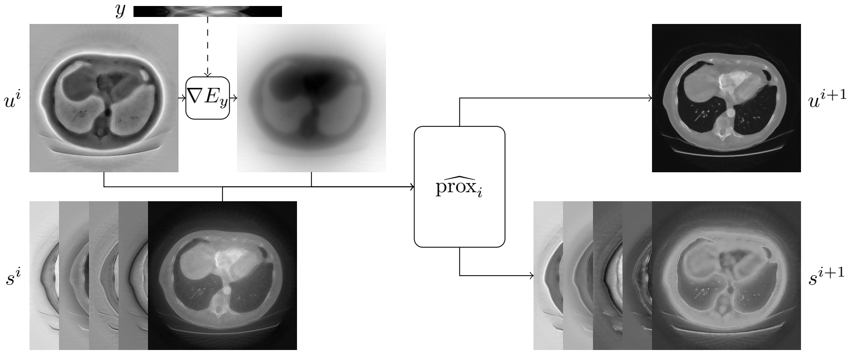
Learned PGD Gregor and Le Cun '10, Adler and Öktem '17, ...

$$u^{k+1} = \widehat{\mathrm{prox}_i}(u^k, \nabla \mathcal{D}(u^k))$$

Solution $\Phi(b) := u^K$, "small" $K \in \mathbb{N}$.
Learn $\widehat{\mathrm{prox}_i}$: $\Phi(b) \approx u^*$

Learned proximal gradient descent with memory

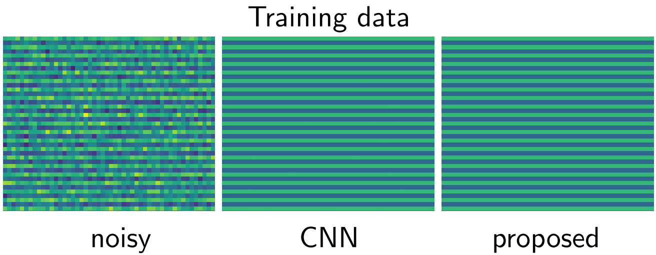
- memory s



Equivariance and Inverse Problems

What happens when data is rotated?

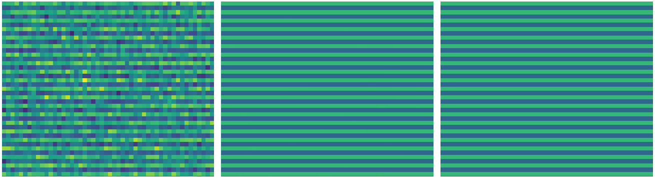
$$\Phi(R_\theta b) \stackrel{?}{=} R_\theta \Phi(b)$$



What happens when data is rotated?

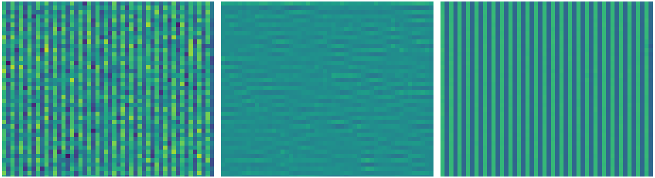
$$\Phi(R_\theta b) \stackrel{?}{=} R_\theta \Phi(b)$$

Training data



noisy CNN proposed

Test data



noisy CNN proposed

How to get “equivariant” mappings?

Example: R_θ rotation by θ, Φ denoising network

$$\Phi(R_\theta b) = R_\theta \Phi(b)$$

How to get "equivariant" mappings?

Example: R_θ rotation by θ, Φ denoising network

$$\Phi(R_\theta b) = R_\theta \Phi(b)$$

- **data augmentation**: e.g. $(b_i, u_i)_i$ becomes $(R_\theta b_i, R_\theta u_i)_{i,\theta}$
  - ✓ **simple to implement** for image-based tasks (e.g. denoising, image segmentation etc)
  - ✗ potentially **computationally costly** since training data is larger
  - ✗ **no guarantees** this will translate to test data
  - ✗ **not always easy/possible** (for inverse problems only viable in simulations or if data is not paired (semi-supervised training))

How to get "equivariant" mappings?

Example: R_θ rotation by θ, Φ denoising network

$$\Phi(R_\theta b) = R_\theta \Phi(b)$$

- **data augmentation**: e.g. $(b_i, u_i)_i$ becomes $(R_\theta b_i, R_\theta u_i)_{i,\theta}$
  - ✓ **simple to implement** for image-based tasks (e.g. denoising, image segmentation etc)
  - ✗ potentially **computationally costly** since training data is larger
  - ✗ **no guarantees** this will translate to test data
  - ✗ **not always easy/possible** (for inverse problems only viable in simulations or if data is not paired (semi-supervised training))
- **equivariance by design** (this talk!)
  - ✓ **mathematical guarantees**
  - ✗ **not trivial** to do

Equivariant neural networks have been studied a lot for segmentation, classification, denoising etc

Bekkers et al. '18, Weiler and Cesa '19, Cohen and Welling '16, Dieleman et al. '16, Sosnovik et al. '19, Worall and Welling '19, ...

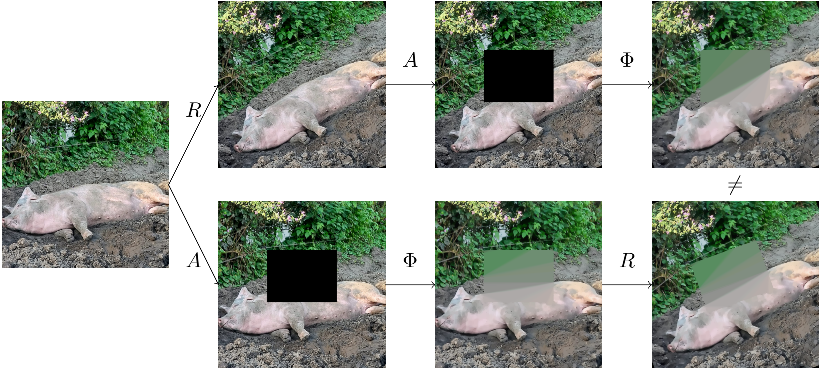
Equivariance and inverse problems

- inverse problem $Au = b$, solution operator: $\Phi : Y \to X$
- **Hope** $\Phi \circ A$ is equivariant, e.g. $R_\theta \circ \Phi \circ A = \Phi \circ A \circ R_\theta$

Equivariance and inverse problems

- inverse problem $Au = b$, solution operator: $\Phi : Y \to X$
- **Hope** $\Phi \circ A$ is equivariant, e.g. $R_\theta \circ \Phi \circ A = \Phi \circ A \circ R_\theta$

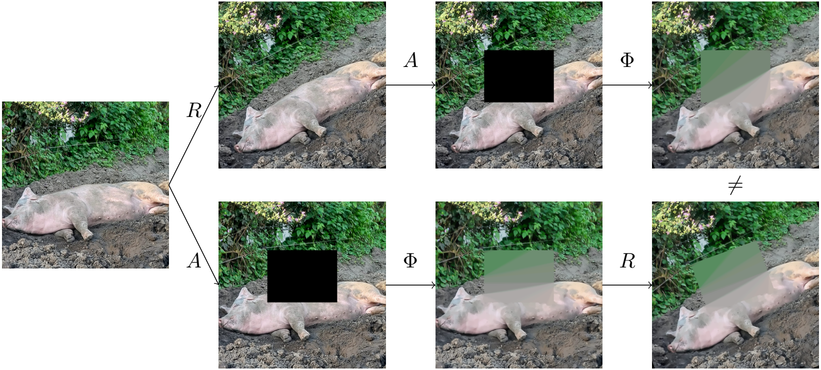
- Even if J is invariant, $\Phi \circ A$ is **not generally equivariant**
- Example: variational TV inpainting



Equivariance and inverse problems

- inverse problem $Au = b$, solution operator: $\Phi : Y \to X$
- **Hope** $\Phi \circ A$ is equivariant, e.g. $R_\theta \circ \Phi \circ A = \Phi \circ A \circ R_\theta$

- Even if J is invariant, $\Phi \circ A$ is **not generally equivariant**
- Example: variational TV inpainting



What about well-behaved kernel: compressed sensing?

Invariant functional implies equivariant proximal operator

Theorem Celledoni et al. '21
Let $X = L^2(\Omega)$ and J be **invariant** with respect to rotations: $J(R_\theta u) = J(u)$.
Then prox_J is **equivariant**, i.e for all $u \in X$

$$\operatorname{prox}_J(R_\theta u) = R_\theta \operatorname{prox}_J(u).$$

- For **example** the total variation (and higher order variants) is invariant to rigid motion

Invariant functional implies equivariant proximal operator

Theorem Celledoni et al. '21
Let $X = L^2(\Omega)$ and J be **invariant** with respect to rotations: $J(R_\theta u) = J(u)$.
Then prox_J is **equivariant**, i.e for all $u \in X$

$$\operatorname{prox}_J(R_\theta u) = R_\theta \operatorname{prox}_J(u).$$

- For **example** the total variation (and higher order variants) is invariant to rigid motion

Since we are interested in Learned Gradient Descent, equivariance of the network is a natural condition.

Equivariance revisited

What is equivariance?

Definition (Group G)

- **associativity:** $\forall g_1, g_2, g_3 \in G : (g_1 \cdot g_2) \cdot g_3 = g_1 \cdot (g_2 \cdot g_3)$,
- **identity:** $\exists e \in G \; \forall g \in G : e \cdot g = g$
- **invertibility:** $\forall g \in G \; \exists g^{-1} \in G : g^{-1} \cdot g = e$

What is equivariance?

Definition (Group G)

- **associativity:** $\forall g_1, g_2, g_3 \in G : (g_1 \cdot g_2) \cdot g_3 = g_1 \cdot (g_2 \cdot g_3)$,
- **identity:** $\exists e \in G \; \forall g \in G : e \cdot g = g$
- **invertibility:** $\forall g \in G \; \exists g^{-1} \in G : g^{-1} \cdot g = e$

Definition (G acts on X)

- **group action:** $G \times X \to X, \quad (g, x) \mapsto g \cdot x$
- **identity:** $e \cdot x = x$
- **compatibility:** $g_1 \cdot (g_2 \cdot x) = (g_1 \cdot g_2) \cdot x$

What is equivariance?

Definition (Group G)

- **associativity:** $\forall g_1, g_2, g_3 \in G : (g_1 \cdot g_2) \cdot g_3 = g_1 \cdot (g_2 \cdot g_3)$,
- **identity:** $\exists e \in G \; \forall g \in G : e \cdot g = g$
- **invertibility:** $\forall g \in G \; \exists g^{-1} \in G : g^{-1} \cdot g = e$

Definition (G acts on X)

- **group action:** $G \times X \to X, \quad (g, x) \mapsto g \cdot x$
- **identity:** $e \cdot x = x$
- **compatibility:** $g_1 \cdot (g_2 \cdot x) = (g_1 \cdot g_2) \cdot x$

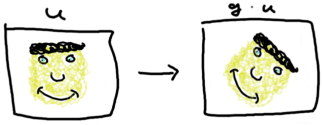
Definition (Equivariance) G acts on X and Y, $\Phi : X \to Y$ is called **equivariant** if for all $g \in G, x \in X$

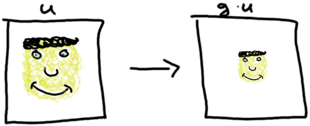
$$g \cdot \Phi(x) = \Phi(g \cdot x)$$

Group actions on functions, e.g. $X = L^2(\mathbb{R}^n, \mathbb{R}^m)$

domain: $(g \cdot u)(x) = u(g^{-1} \cdot x)$

translations, rotations, affine transformations





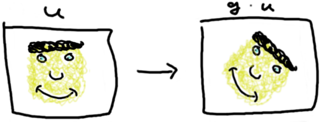
Example: $G = (\mathbb{R}^n, +)$ may act on X via

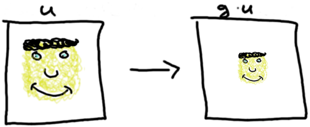
- $(g \cdot u)(x) = u(x - g)$
- $(g \cdot u)(x) = u(x \exp(g))$, if $n = 1$

Group actions on functions, e.g. $X = L^2(\mathbb{R}^n, \mathbb{R}^m)$

domain: $(g \cdot u)(x) = u(g^{-1} \cdot x)$

translations, rotations, affine transformations





Example: $G = (\mathbb{R}^n, +)$ may act on X via

- $(g \cdot u)(x) = u(x - g)$
- $(g \cdot u)(x) = u(x \exp(g))$, if $n = 1$

range: $(g \cdot u)(x) = g \cdot u(x)$

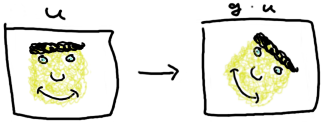
Example: $G = (\mathbb{R}^m, +)$ may act on X via

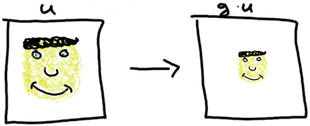
- $(g \cdot u)(x) = u(x) + g$

Group actions on functions, e.g. $X = L^2(\mathbb{R}^n, \mathbb{R}^m)$

domain: $(g \cdot u)(x) = u(g^{-1} \cdot x)$

translations, rotations, affine transformations





Example: $G = (\mathbb{R}^n, +)$ may act on X via

- $(g \cdot u)(x) = u(x - g)$
- $(g \cdot u)(x) = u(x \exp(g))$, if $n = 1$

range: $(g \cdot u)(x) = g \cdot u(x)$

Example: $G = (\mathbb{R}^m, +)$ may act on X via

- $(g \cdot u)(x) = u(x) + g$

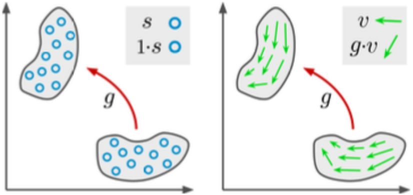
both domain and range: $(g \cdot u)(x) = g \cdot u(g^{-1} \cdot x)$

Acting on domain and range: $(g \cdot u)(x) = g \cdot u(g^{-1} \cdot x)$

- $\overline{G} = \mathbb{R}^n \rtimes H$, $\quad H$ subgroup of the general linear group $\mathrm{GL}(n)$
- $g \cdot x = Rx + t, g = (t, R) \in \overline{G}, t \in \mathbb{R}^n, R \in H$
- $\pi : H \to GL(m)$ representation of H
- $(g \cdot u)(x) = \pi(R) u(R^{-1}(x - t))$

Examples

- **Translations:** $H = \{e\}$
- **Roto-Translations:** $H = \mathrm{SO}(n)$
- **Finite Roto-Translations** $H = Z_M$ (finite subgroup of $\mathrm{SO}(2)$)
- Example: u vector-field, move and transform vectors



Weiler and Cesa 2019

More details: implies equivariant proximal operator

Theorem Celledoni et al. '21

- G acts **isometrically** on X ($\|g \cdot u\| = \|u\|$)
- $J : X \to \mathbb{R} \cup \{+\infty\}$ is **invariant** ($J(g \cdot u) = J(u)$)
- J has **well-defined single-valued proximal operator**

Then prox_J is **equivariant**, i.e for all $u \in X$ and $g \in G$

$$\mathrm{prox}_J(g \cdot u) = g \cdot \mathrm{prox}_J(u).$$

- Proof does **generalize** to variatial regularization with L^2-datafit **if A is equivariant**

Equivariance and Neural Networks

How to get "equivariant" networks?

Proposition Let G be any group.

- The **composition** $\Phi \circ \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **sum** $\Phi + \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **identity** $\Phi(u) = u$ is equivariant.

How to get "equivariant" networks?

Proposition Let G be any group.

- The **composition** $\Phi \circ \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **sum** $\Phi + \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **identity** $\Phi(u) = u$ is equivariant.

Outlook (linearity) There are non-trivial $\overline{G}$-equivariant linear operators.

How to get "equivariant" networks?

Proposition Let G be any group.

- The **composition** $\Phi \circ \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **sum** $\Phi + \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **identity** $\Phi(u) = u$ is equivariant.

Outlook (linearity) There are non-trivial $\overline{G}$-equivariant linear operators.

Proposition (bias) Let $\Phi : X \to X, (\Phi u)(x) = u(x) + b(x)$. For any group G, Φ is equivariant if b is invariant, i.e. $g \cdot b = b$.

How to get “equivariant” networks?

Proposition Let G be any group.

- The **composition** $\Phi \circ \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **sum** $\Phi + \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **identity** $\Phi(u) = u$ is equivariant.

Outlook (linearity) There are non-trivial $\overline{G}$-equivariant linear operators.

Proposition (bias) Let $\Phi : X \to X, (\Phi u)(x) = u(x) + b(x)$. For any group G, Φ is equivariant if b is invariant, i.e. $g \cdot b = b$.

Outlook (nonlinearity) There are $\overline{G}$-equivariant nonlinearities.

How to get "equivariant" networks?

Proposition Let G be any group.

- The **composition** $\Phi \circ \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **sum** $\Phi + \Psi$ is equivariant if Φ and Ψ are equivariant.
- The **identity** $\Phi(u) = u$ is equivariant.

Outlook (linearity) There are non-trivial $\overline{G}$-equivariant linear operators.

Proposition (bias) Let $\Phi : X \to X, (\Phi u)(x) = u(x) + b(x)$. For any group G, Φ is equivariant if b is invariant, i.e. $g \cdot b = b$.

Outlook (nonlinearity) There are $\overline{G}$-equivariant nonlinearities.

Construct $\overline{G}$-equivariant neural networks the usual way:

- layers $\Phi = \Phi_n \circ \cdots \circ \Phi_1$
- $\Phi(u) = \sigma(Au + b)$
- ResNet $\Phi(u) = u + \sigma(Au + b)$

Equivariant linear functions ($\pi_X \equiv id$)

In a nutshell: Linear $\overline{G}$-equivariant operators are convolutions with a kernel satisfying an additional constraint.

Equivariant linear functions ($\pi_X \equiv id$)

In a nutshell: Linear $\overline{G}$-equivariant operators are convolutions with a kernel satisfying an additional constraint.

Theorem paraphrasing e.g. Weiler and Cesa 2019
Let X, Y be function spaces, e.g. $X = L^2(\mathbb{R}^n, \mathbb{R}^m)$, $Y = L^2(\mathbb{R}^n, \mathbb{R}^M)$. The linear operator $\Phi : X \to Y$,

$$\Phi f(x) = \int K(x, y) f(y) dy$$

with $K : \mathbb{R}^n \to \mathbb{R}^{M \times m}$ is $\overline{G}$-**equivariant** iff there is a k such that

$$\Phi f(x) = \int k(x - y) f(y) dy$$

and k is H-invariant, i.e. for all $R \in H$, $x \in \mathbb{R}^n$: $k(Rx) = k(x)$.

Equivariant nonlinearities ($\pi_X \equiv id$)

In a nutshell: There are $\overline{G}$-equivariant nonlinearities.

Equivariant nonlinearities ($\pi_X \equiv id$)

In a nutshell: There are $\overline{G}$-equivariant nonlinearities.

Let $\psi : \mathbb{R} \to \mathbb{R}$ be any non-linear function.

- **Norm nonlinearity** $\Psi_N : X \to X$,

$$[\Psi_N(u)](x) = u(x) \cdot \psi(\|u(x)\|)$$

- **Pointwise and componentwise nonlinearity** $\Psi_P : X \to X$,

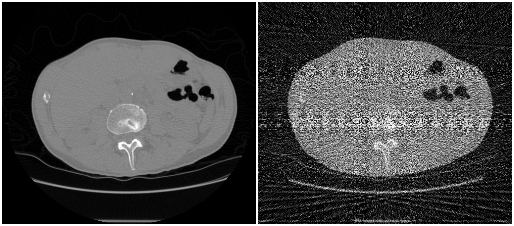
$$[\Psi_P(u)](x) = \vec{\psi}(u(x)), \quad \vec{\psi}(x)_i = \psi(x_i)$$

Lemma Both nonlinearities are $\overline{G}$-equivariant.

Numerical Results

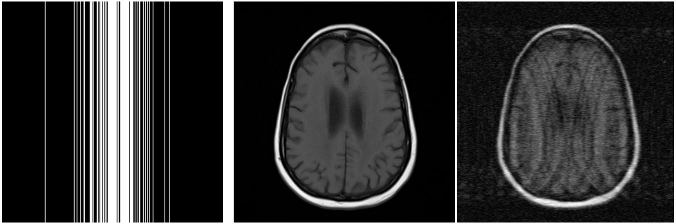
Datasets

- **CT:** LIDC-IDRI data set, 5000+200+1000 images, 50 views



u $\quad$ $\mathrm{FBP}(y)$

- **MR:** FastMRI data set, 5000+200+1000 images



$\mathcal{S}$ $\quad$ u $\quad$ $\mathcal{F}^{-1}(\mathcal{S}^* y)$

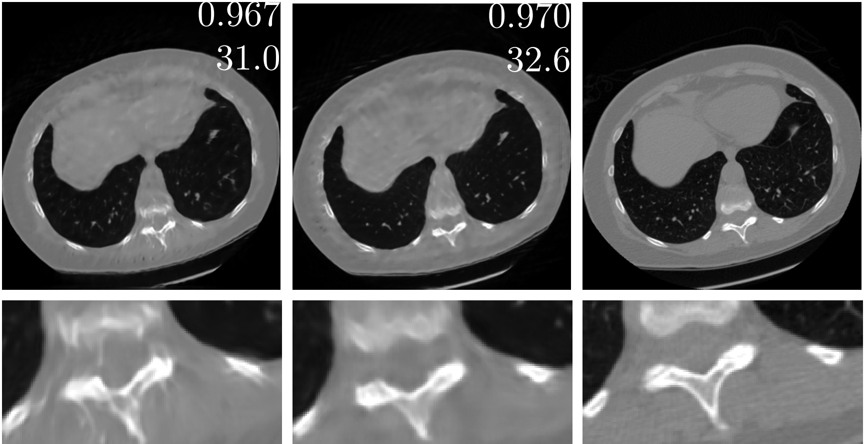
CT Results

Equivariant = roto-translations; Ordinary = translations

Equivariant improves upon Ordinary:

- **higher** SSIM and PSNR
- **fewer** artefacts and **finer** details

Ordinary | Equivariant | Ground truth

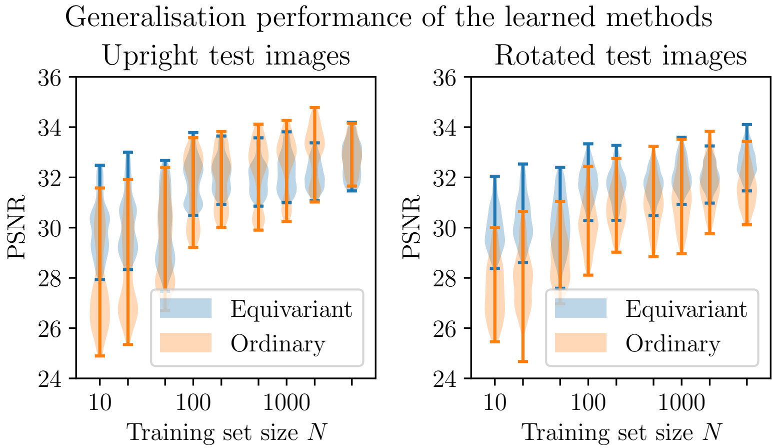


CT Results

Equivariant = roto-translations; Ordinary = translations

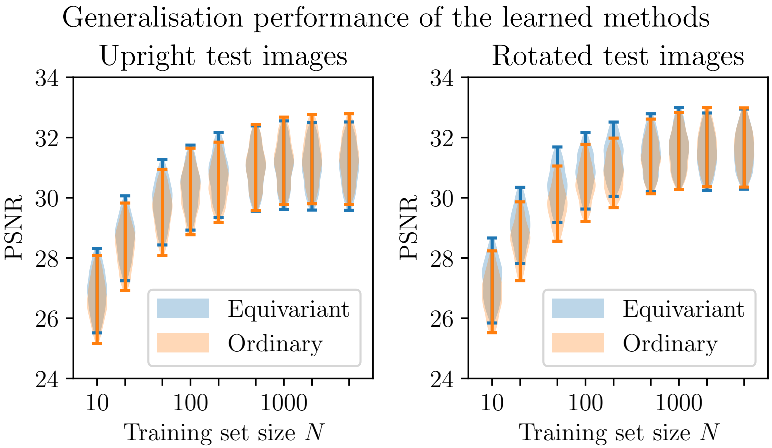
Equivariant improves upon Ordinary:

- **small** training sets
- **unseen** orientations



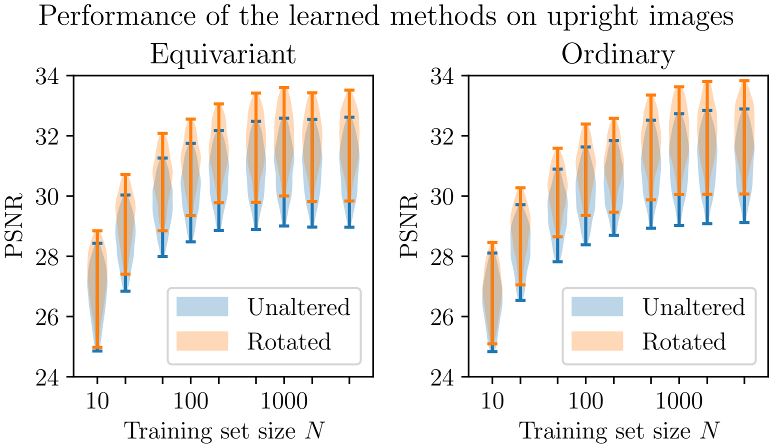
MR Results

- **similar** observations in MR (as in CT); smaller difference
- results for both methods **better on rotated** images



MR Results: Smoothing

- **smoothing helps:** easier to train on smoother images



Conclusions and Outlook

Conclusions

- **no need for data augmentation**: mathematically guaranteed equivariant neural networks exist
- **solution operators** may **not** be equivariant, but **proximal operators** usually are **equivariant**
- computationally **efficient**: as CNNs at run time
- useful for many **applications**: **fewer data** and **robustness**

E. Celledoni, M. J. Ehrhardt, C. Etmann, B. Owren, C.-B. Schönlieb, and F. Sherry, "Equivariant neural networks for inverse problems," Inverse Problems, vol. 37, no. 8, p. 085006, 2021.

Conclusions and Outlook

Conclusions

- **no need for data augmentation**: mathematically guaranteed equivariant neural networks exist
- **solution operators** may **not** be equivariant, but **proximal operators** usually are **equivariant**
- computationally **efficient**: as CNNs at run time
- useful for many **applications**: **fewer data** and **robustness**

Future work

- **other groups**, e.g. scaling of itensities; scaling of domain
- **other inverse problems**, e.g. compressed sensing or trivial kernel
- **higher dimensions** e.g. 3D or dynamic inverse problems

E. Celledoni, M. J. Ehrhardt, C. Etmann, B. Owren, C.-B. Schönlieb, and F. Sherry, "Equivariant neural networks for inverse problems," Inverse Problems, vol. 37, no. 8, p. 085006, 2021.