Code available!

https://github.com/ericlin79119/IC-STN

Inverse Compositional Spatial Transformer Networks

Chen-Hsuan Lin, Simon Lucey   Carnegie Mellon University

Carnegie Mellon University
The Robotics Institute

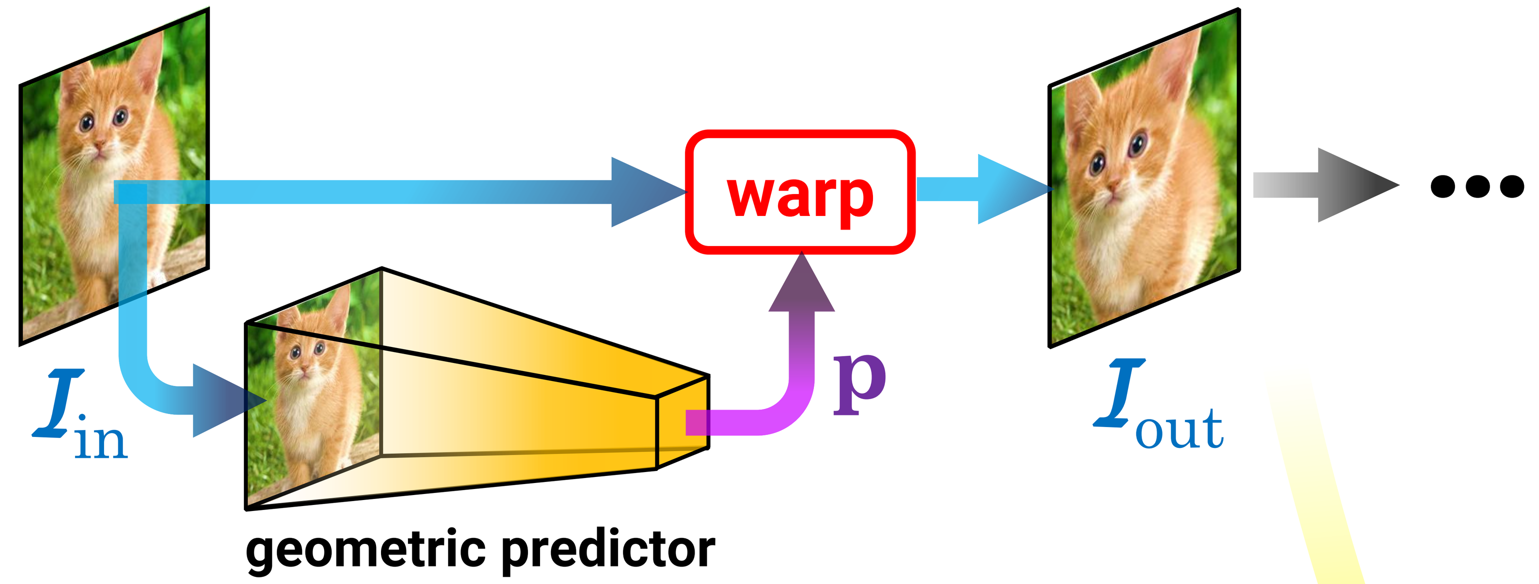
Overview

Typical CNNs **tolerate** spatial variations within data implicitly through
1. **Data augmentation** (generate geometric perturbations)
2. **Spatial pooling** (abstracts semantics, but destroys spatial details)

We propose **IC-STN** to **resolve** spatial variations explicitly via **recurrent** transformations

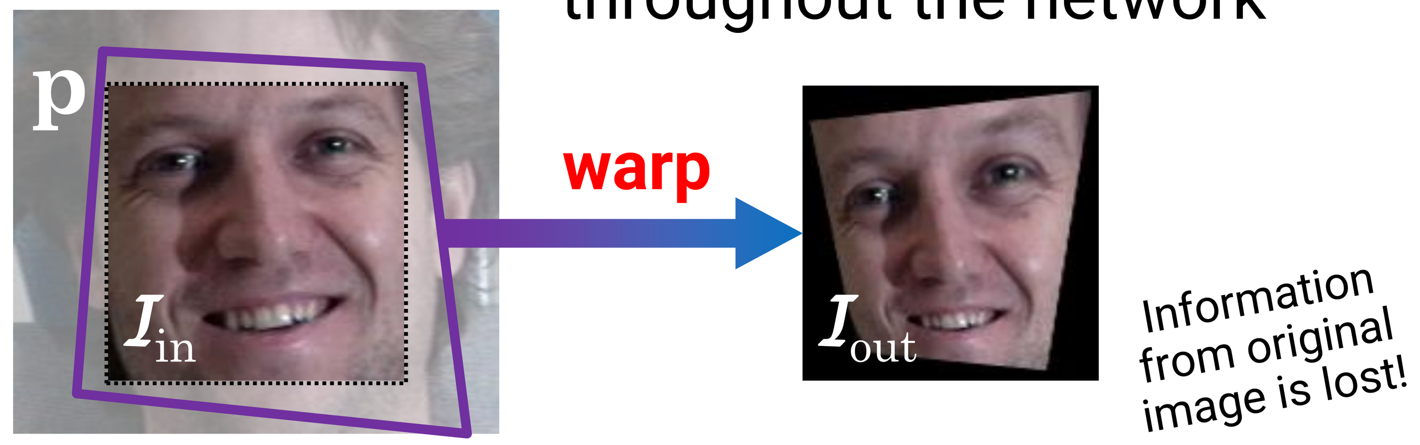
Spatial Transformer Networks (STN)

learns to **resolve** spatial variations explicitly by predicting the geometric transformation $\mathbf{p}$ on the input image $\mathcal{I}_{\text{in}}$

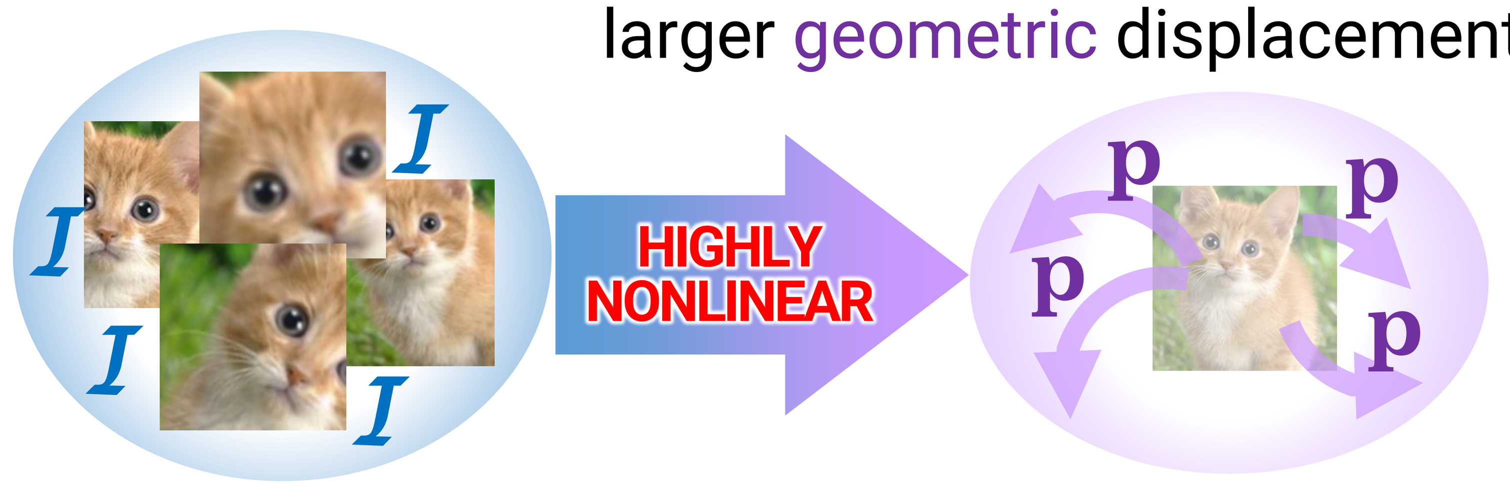


$$\mathcal{I}_{\text{out}}(\mathbf{0}) = \mathcal{I}_{\text{in}}(\mathbf{p}),\ \text{where}\ \ \mathbf{p} = f(\mathcal{I}_{\text{in}}(\mathbf{0}))$$

- **Boundary effect** Geometry is not preserved throughout the network



- **Single prediction** Appearance decorrelates with larger geometric displacements



The Lucas-Kanade (LK) Algorithm

solves for alignment by **iteratively** predicting updates to the warp $\mathbf{p}$ on image $\mathcal{I}$ to match the template $\mathcal{T}$

original form

$$\min_{\Delta\mathbf{p}} \left\| \mathcal{I}(\mathbf{p}+\Delta\mathbf{p}) - \mathcal{T}(\mathbf{0}) \right\|_2^2$$

linearize

$$\min_{\Delta\mathbf{p}} \left\| \mathcal{I}(\mathbf{p}) + \frac{\partial \mathcal{I}(\mathbf{p})}{\partial \mathbf{p}}\Delta\mathbf{p} - \mathcal{T}(\mathbf{0}) \right\|_2^2$$

add / solve

$$\Delta\mathbf{p} = -\left(\frac{\partial \mathcal{I}(\mathbf{p})}{\partial \mathbf{p}}\right)^{\dagger} (\mathcal{I}(\mathbf{p}) - \mathcal{T}(\mathbf{0}))$$

Iteration-specific linear models (dependent on warp state $\mathbf{p}$)

Inverse Compositional (IC)

$$\min_{\Delta\mathbf{p}} \left\| \mathcal{I}(\mathbf{p}) - \mathcal{T}(\Delta\mathbf{p}) \right\|_2^2$$

linearize

$$\min_{\Delta\mathbf{p}} \left\| \mathcal{I}(\mathbf{p}) - \mathcal{T}(\mathbf{0}) - \frac{\partial \mathcal{T}(\mathbf{0})}{\partial \mathbf{p}}\Delta\mathbf{p} \right\|_2^2$$

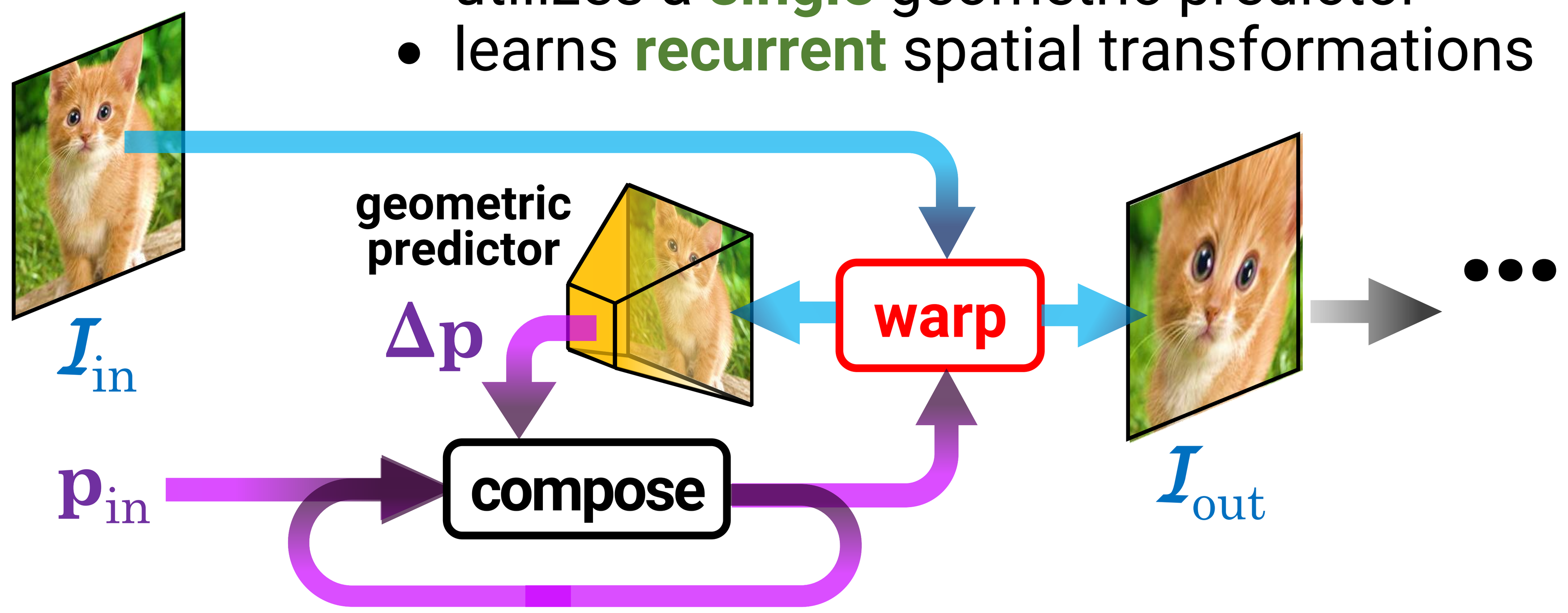
(inversely) compose / solve

$$\Delta\mathbf{p} = \left(\frac{\partial \mathcal{T}(\mathbf{0})}{\partial \mathbf{p}}\right)^{\dagger} (\mathcal{I}(\mathbf{p}) - \mathcal{T}(\mathbf{0}))$$

Static linear model (independent on warp state $\mathbf{p}$)

IC-STN

- preserves geometry and original input image
- utilizes a **single** geometric predictor
- learns **recurrent** spatial transformations



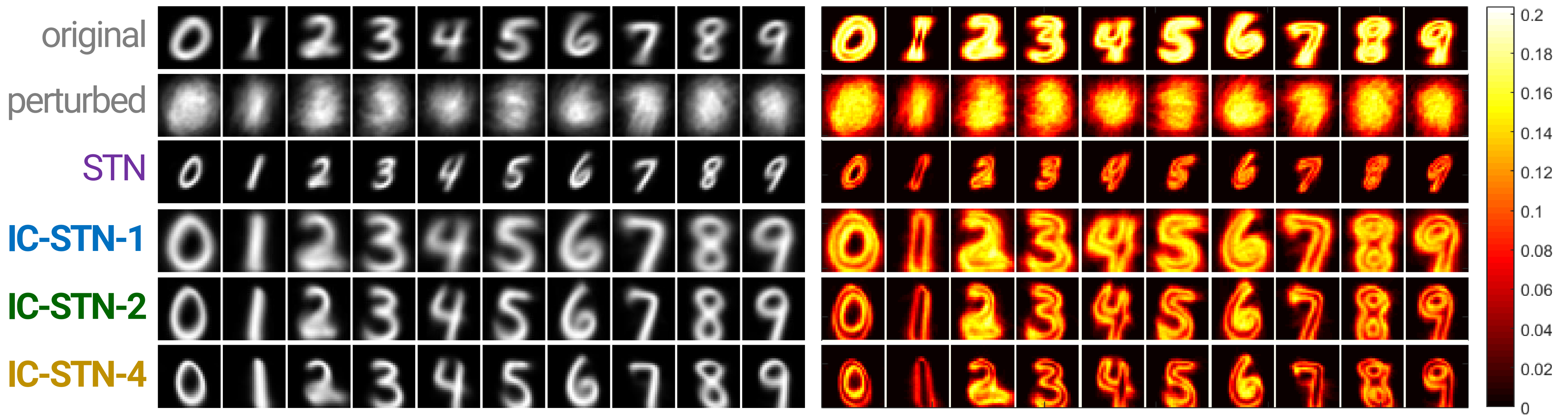
Experiments (perturbed MNIST classification)

Classification error

CNN	6.597%
STN	4.944%
IC-STN-1	3.687%
IC-STN-2	1.905%
IC-STN-4	1.230%

Alignment visualization

Mean appearance / Variance heatmap

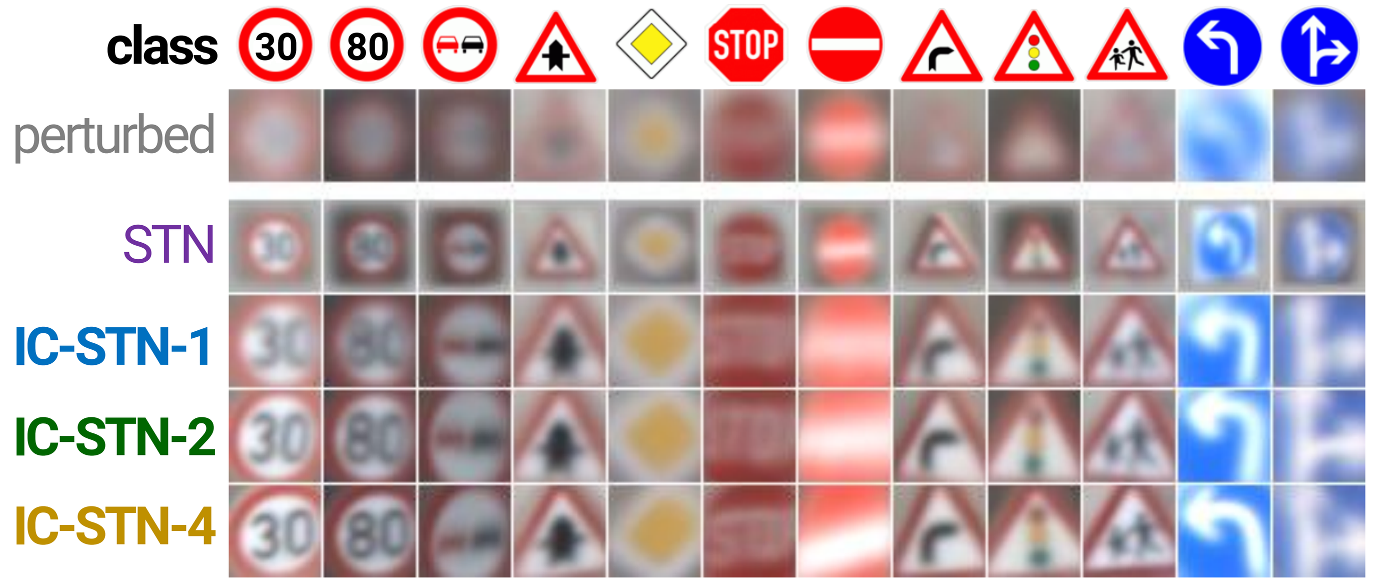


Experiments (traffic sign classification)

Classification error

CNN	8.287%
STN	6.495%
IC-STN-1	5.011%
IC-STN-2	4.122%
IC-STN-4	3.184%

Mean appearance



Discussions

- Theoretical connection: **STN** ⟺ **IC-LK**
- Tolerating data spatial variations needs huge increase of model capacity
- Alignment is more efficient predicting small geometric updates iteratively

Check out our paper and code for more details and discussions!