# Augmentation of brain MRIs using disentanglement and neural network expressivity

Status Seminar

Jon Middleton, Akshay Pai, Mads Nielsen, Stefan Sommer

UNIVERSITY OF COPENHAGEN

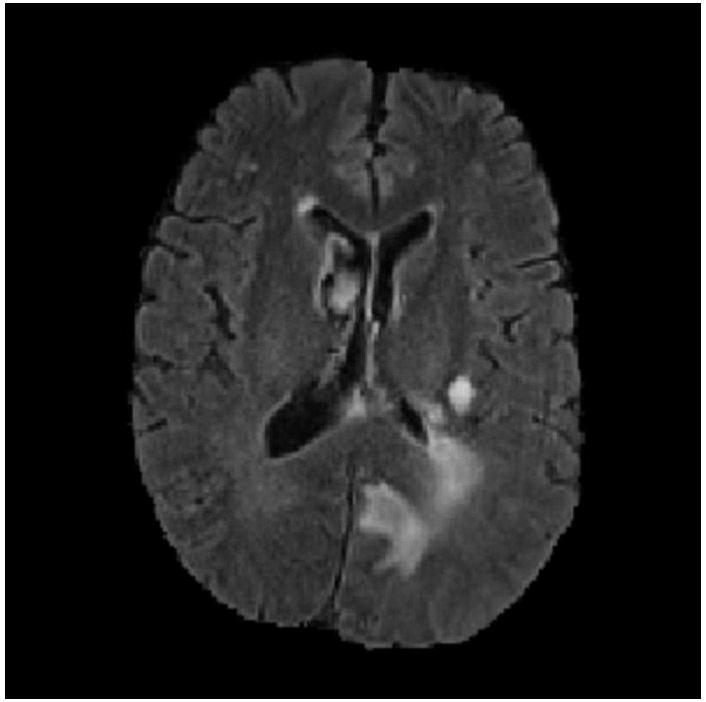
# Pathology Segmentation in Human Brain MRIs

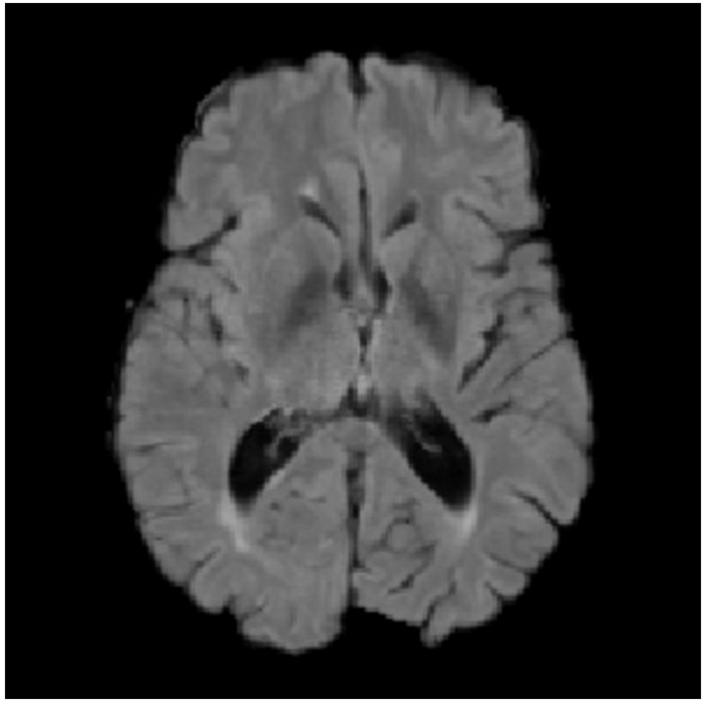
**Task**
Highlight abnormal regions in human brain MRIs

# Pathology Segmentation in Human Brain MRIs

**Task**

Highlight abnormal regions in human brain MRIs

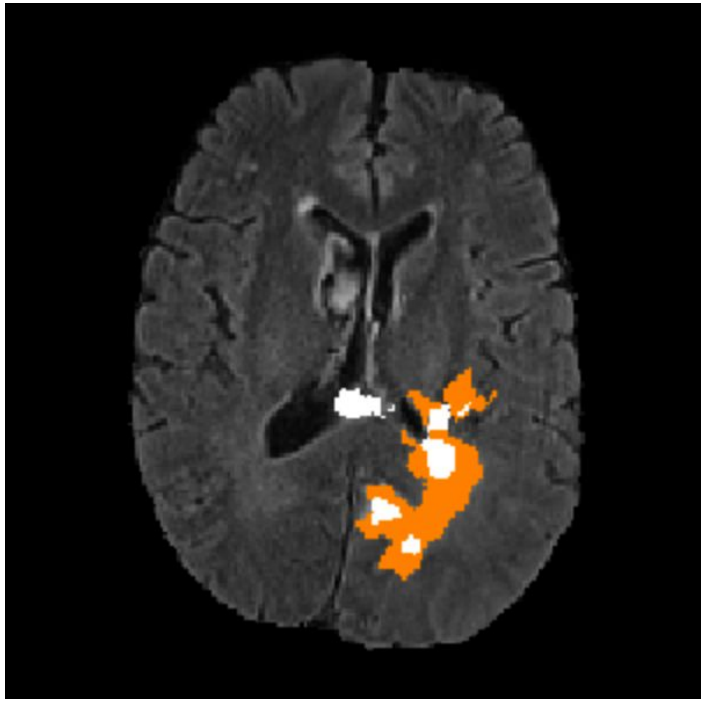


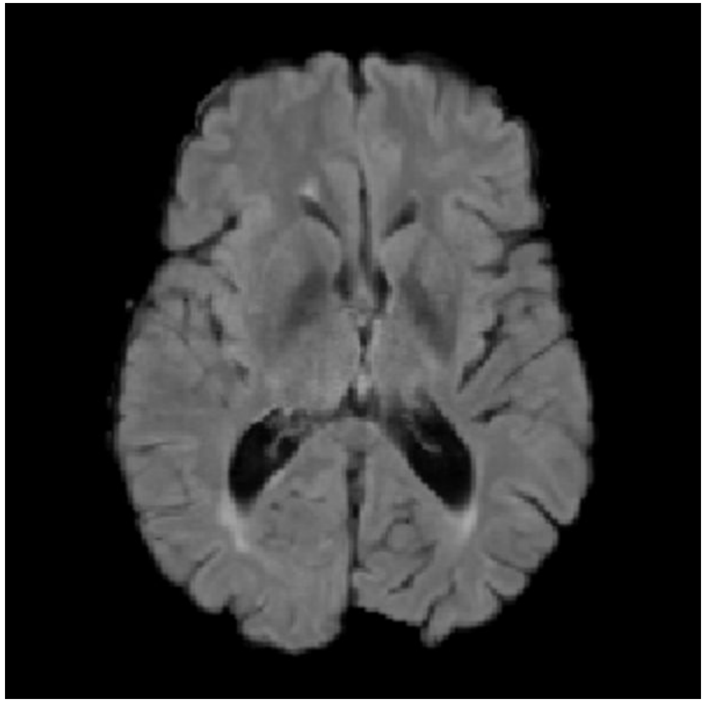


# Pathology Segmentation in Human Brain MRIs

**Task**

Highlight abnormal regions in human brain MRIs



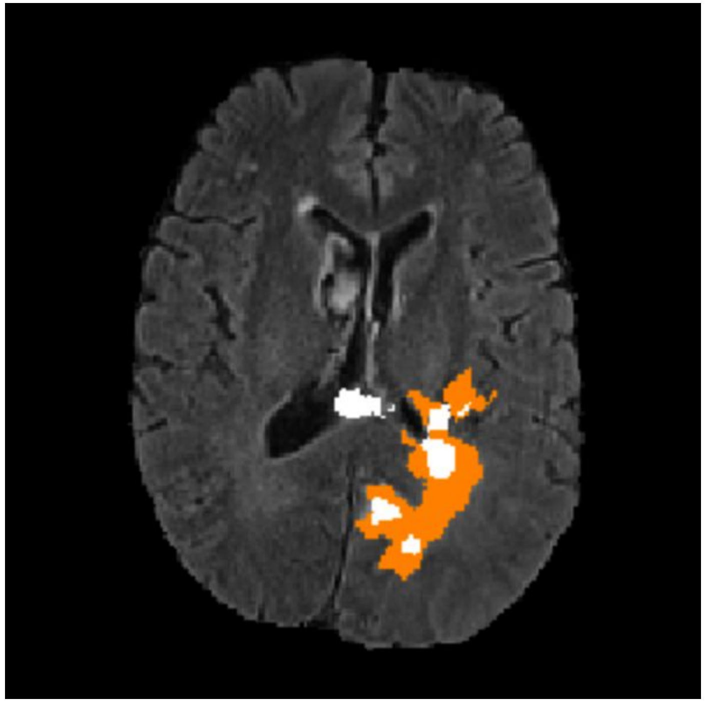


High-grade glioma (BraTS2019)

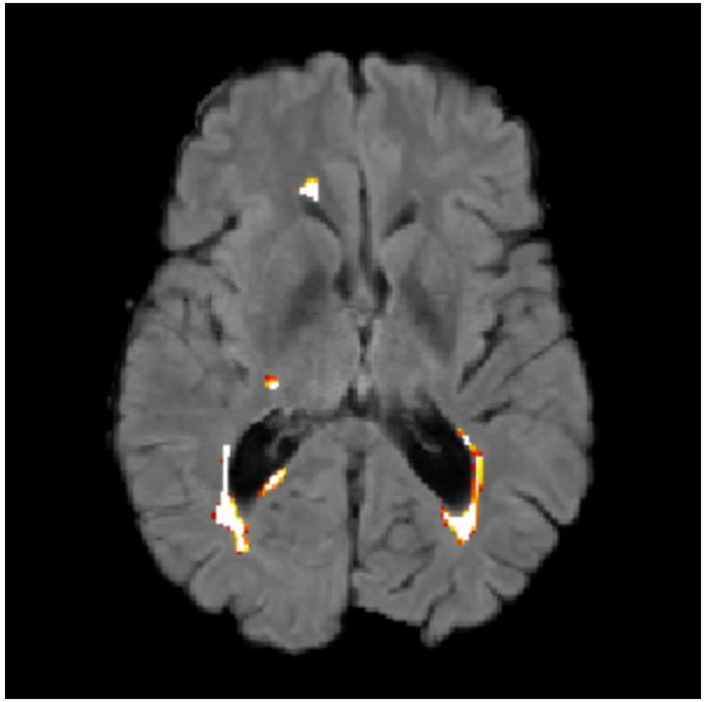
# Pathology Segmentation in Human Brain MRIs

**Task**

Highlight abnormal regions in human brain MRIs



High-grade glioma (BraTS2019)



Multiple sclerosis (MSSEG-2015)

# Pathology Segmentation in Human Brain MRIs

**Problem**

Brain MRI data are scarce; training on scarce data will give an overfit model

# Pathology Segmentation in Human Brain MRIs

**Problem**

Brain MRI data are scarce; training on scarce data will give an overfit model

**Some Approaches**

- *Implicit* data augmentation
  - Inductive bias via model architecture (group-equivariant CNNs)
  - In-network feature augmentation (moment exchange)

# Pathology Segmentation in Human Brain MRIs

**Problem**

Brain MRI data are scarce; training on scarce data will give an overfit model

**Some Approaches**

- *Implicit* data augmentation
  - Inductive bias via model architecture (group-equivariant CNNs)
  - In-network feature augmentation (moment exchange)
- *Explicit* data augmentation
  - Interpolation (mixup)
  - Corruption (cutout, cutmix)
  - Intensification (gamma, bias-field)
  - Deformation (affine, elastic, diffeomorphic)

# Pathology Segmentation in Human Brain MRIs

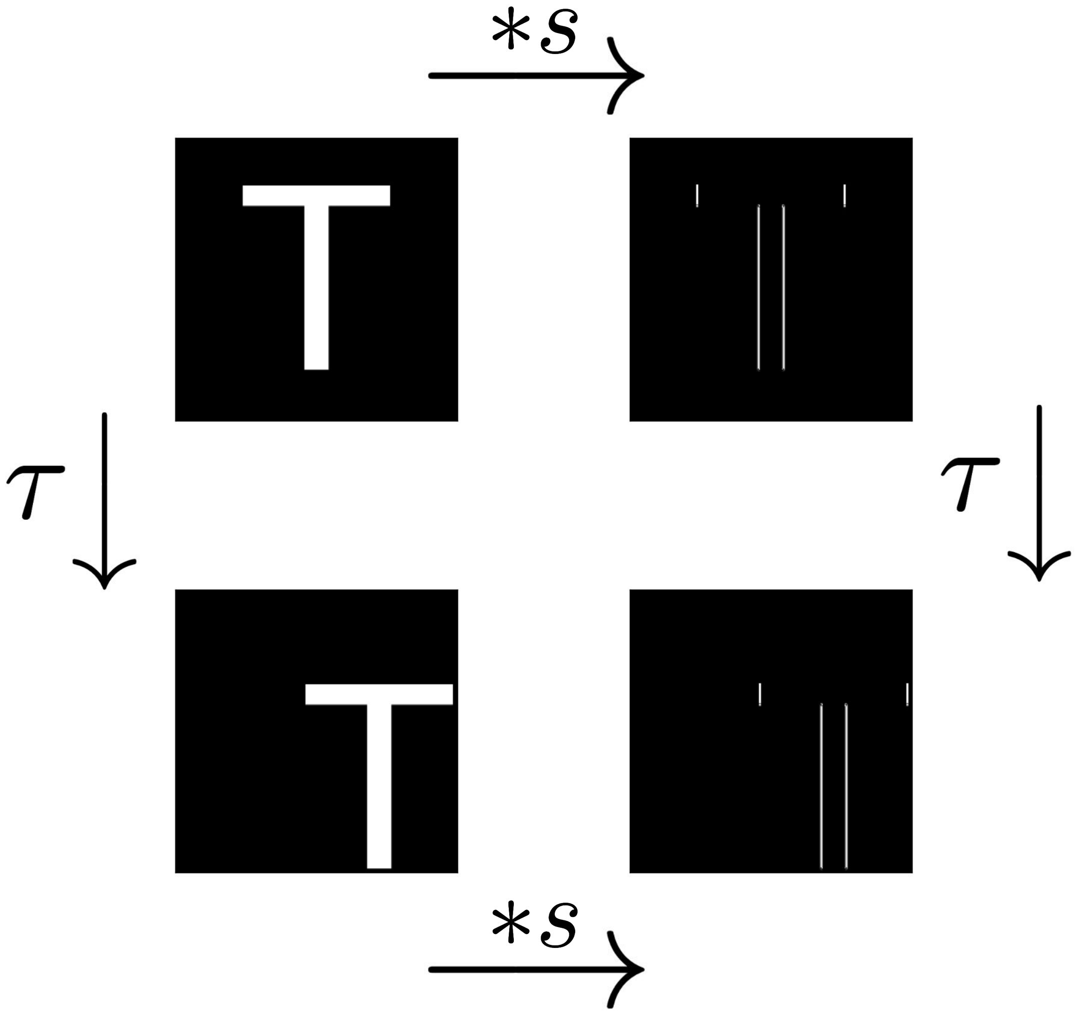
**Problem**

Brain MRI data are scarce; training on scarce data will give an overfit model

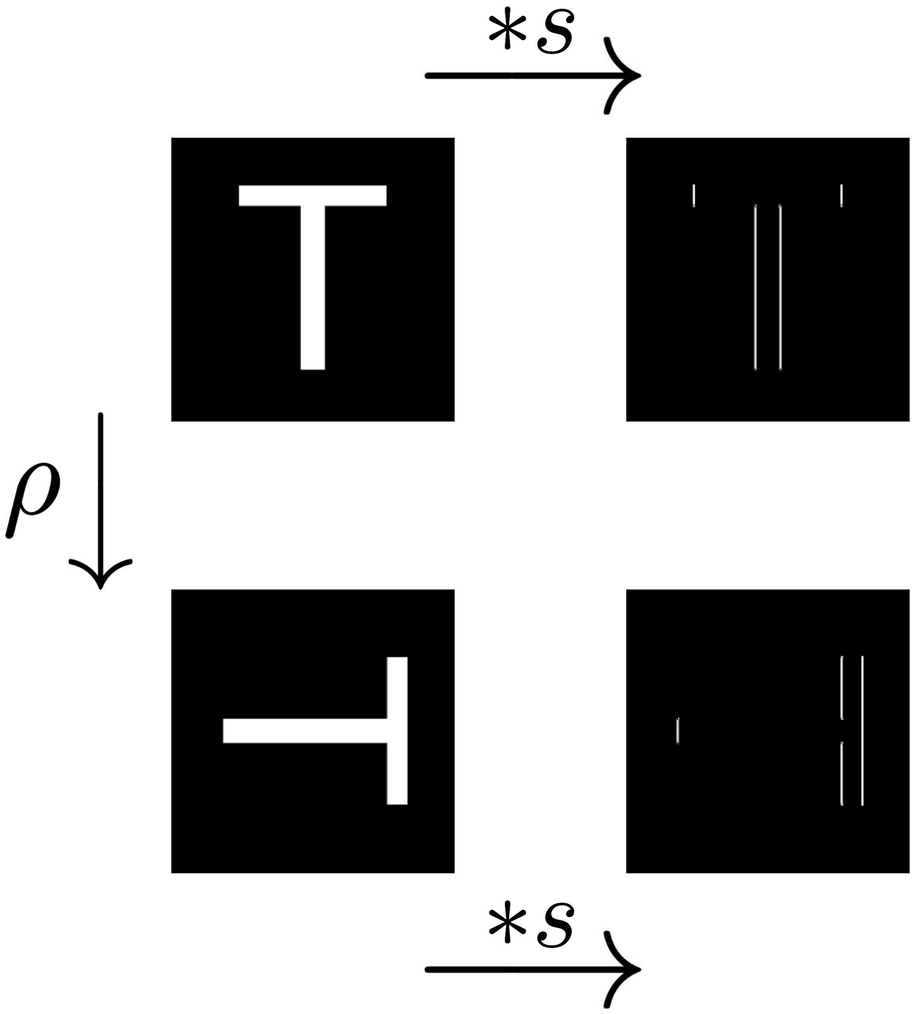
**Some Approaches**

- *Implicit* data augmentation
  - Inductive bias via model architecture (**group-equivariant CNNs**)
  - In-network feature augmentation (moment exchange)
- *Explicit* data augmentation
  - Interpolation (mixup)
  - Corruption (cutout, cutmix)
  - Intensification (gamma, bias-field)
  - Deformation (affine, elastic, **diffeomorphic**)

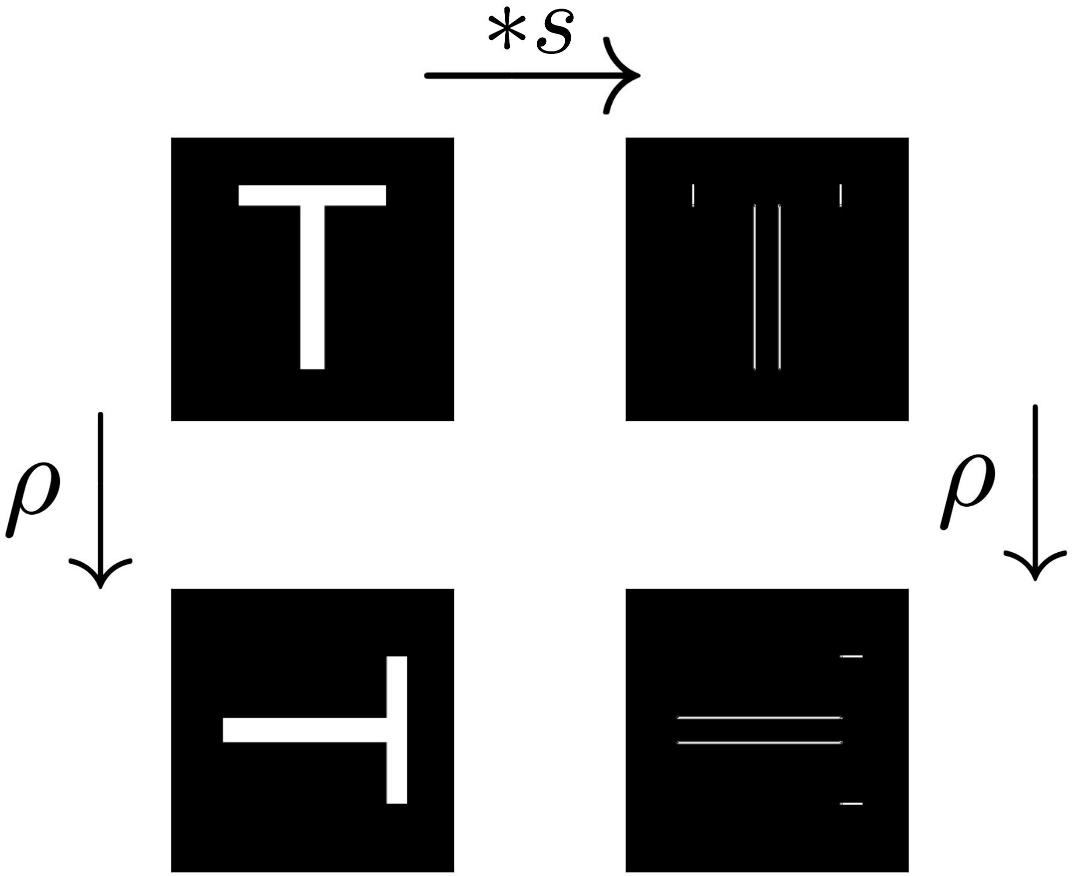
# Translation equivariance in convolutional neural networks



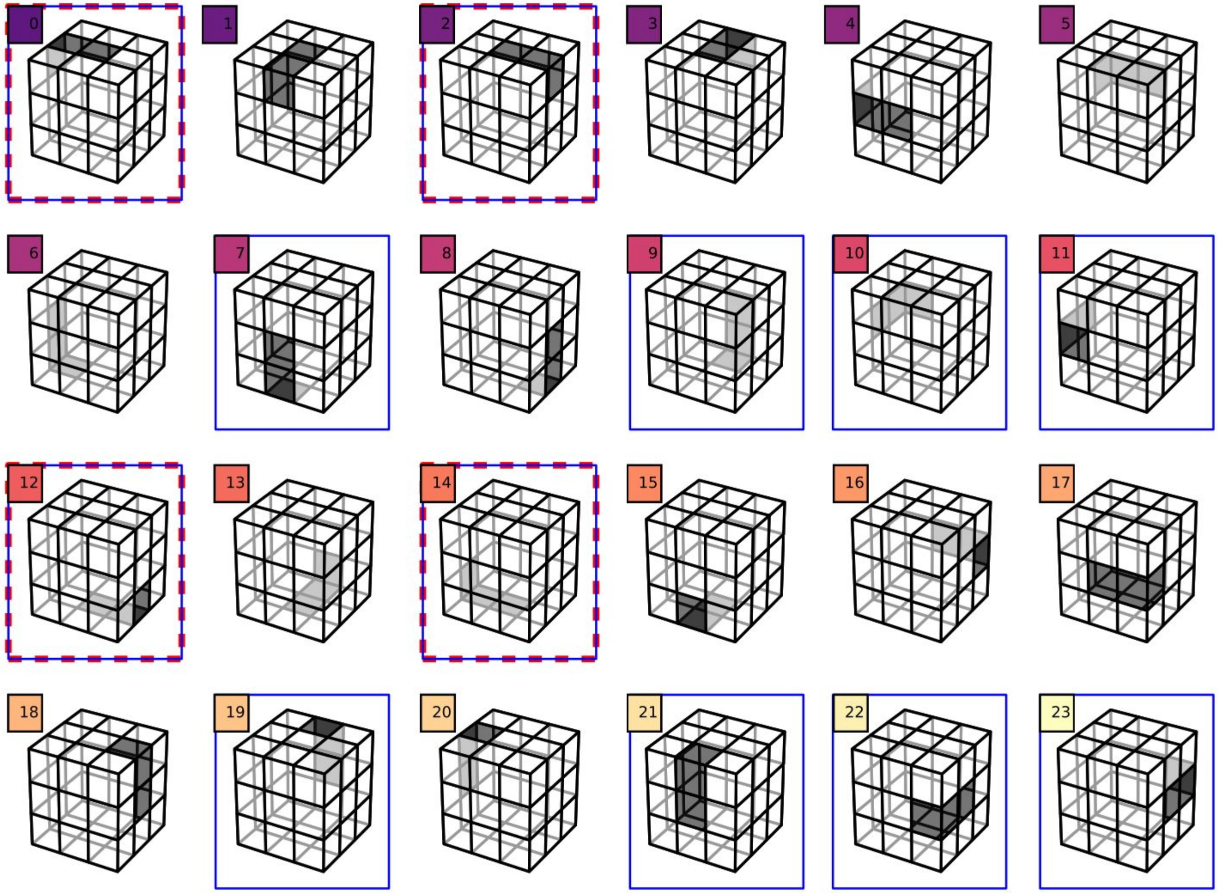
# Finite group-equivariant convolutional neural networks



# Finite group-equivariant convolutional neural networks



# Finite group-equivariant convolutional neural networks




Source: https://openaccess.thecvf.com/content_ECCV_2018/papers/Daniel_Worrall_CubeNet_Equivariance_to_ECCV_2018_paper.pdf

# Open source implementations of G-CNNs

**e2cnn**
https://github.com/QUVA-Lab/e2cnn

**e3nn**
https://github.com/e3nn/e3nn

# Open source implementations of G-CNNs

**GrouPy**
https://github.com/tscohen/GrouPy

**CubeNet**
https://github.com/danielewworrall/cubenet

**Harmonic Networks**
https://github.com/danielewworrall/harmonicConvolutions

# TensorFlow/Keras Implementation of G-CNNs

# TensorFlow/Keras Implementation of G-CNNs

```
class CyclicFour3D(Group):
    """Class for a reducible three-dimensional representation of a cyclic group of any order. This
    group is called 'p4' in the equivariance literature.
    """

    def __init__(self, axes=(0, 1,), name=None):
        group_tuple = (Group.get_matrix([[1, 0, 0], [0, 1, 0], [0, 0, 1]]),
                       Group.get_matrix([[0, -1, 0], [1, 0, 0], [0, 0, 1]]),
                       Group.get_matrix([[-1, 0, 0], [0, -1, 0], [0, 0, 1]]),
                       Group.get_matrix([[0, 1, 0], [-1, 0, 0], [0, 0, 1]]),)
        self.generator = group_tuple[1]
```

# TensorFlow/Keras Implementation of G-CNNs

```
class Octahedral3D(Group):
    """Class for a three-dimensional representation of the octahedral group (a.k.a the group
    of symmetries of the cube, or the fourth symmetric group).
    """

    def __init__(self, inversion=False, name=None):
        self.generators = (Group.get_matrix([[0, 0, 1],
                                             [1, 0, 0],
                                             [0, 1, 0]]),
                           Group.get_matrix([[0, 1, 0],
                                             [-1, 0, 0],
                                             [0, 0, 1]]),)
        group_tuple = ops.generate(self.generators, 6, 3, 24)
```

# TensorFlow/Keras Implementation of G-CNNs

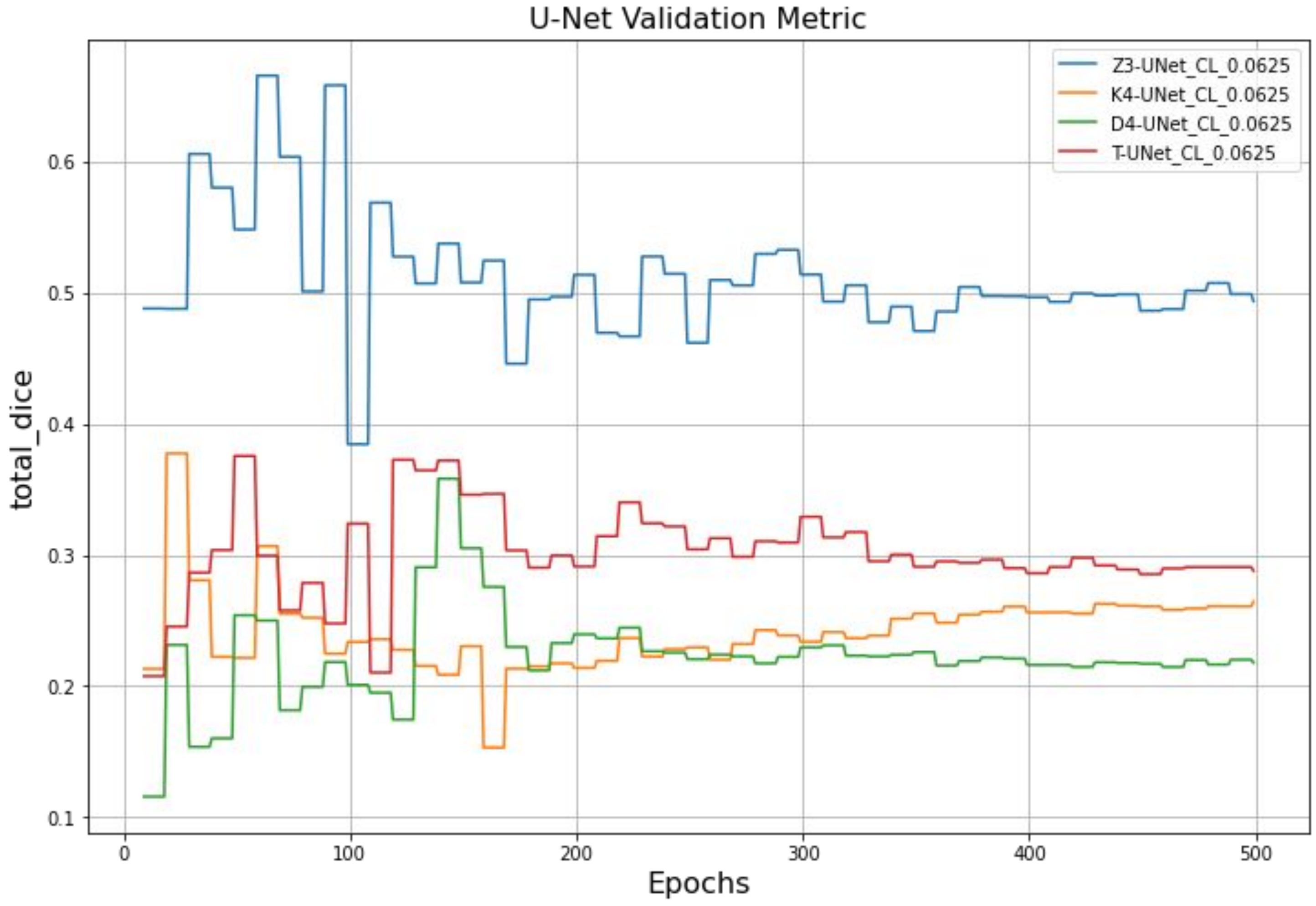
```
class GroupConv3D(Layer):
    """Implementation of an equivariant 3D convolution layer. This implementation maps out of
    Euclidean space and into an output_group of choice.
    """

    def __init__(self, n_channels_in, n_channels_out, kernel_size=3,
                 strides=(1, 1, 1, 1, 1,), padding="SAME", input_group=Trivial(3),
                 output_group=Trivial(3), activation=None, trainable=True):
```

# TensorFlow/Keras Implementation of G-CNNs

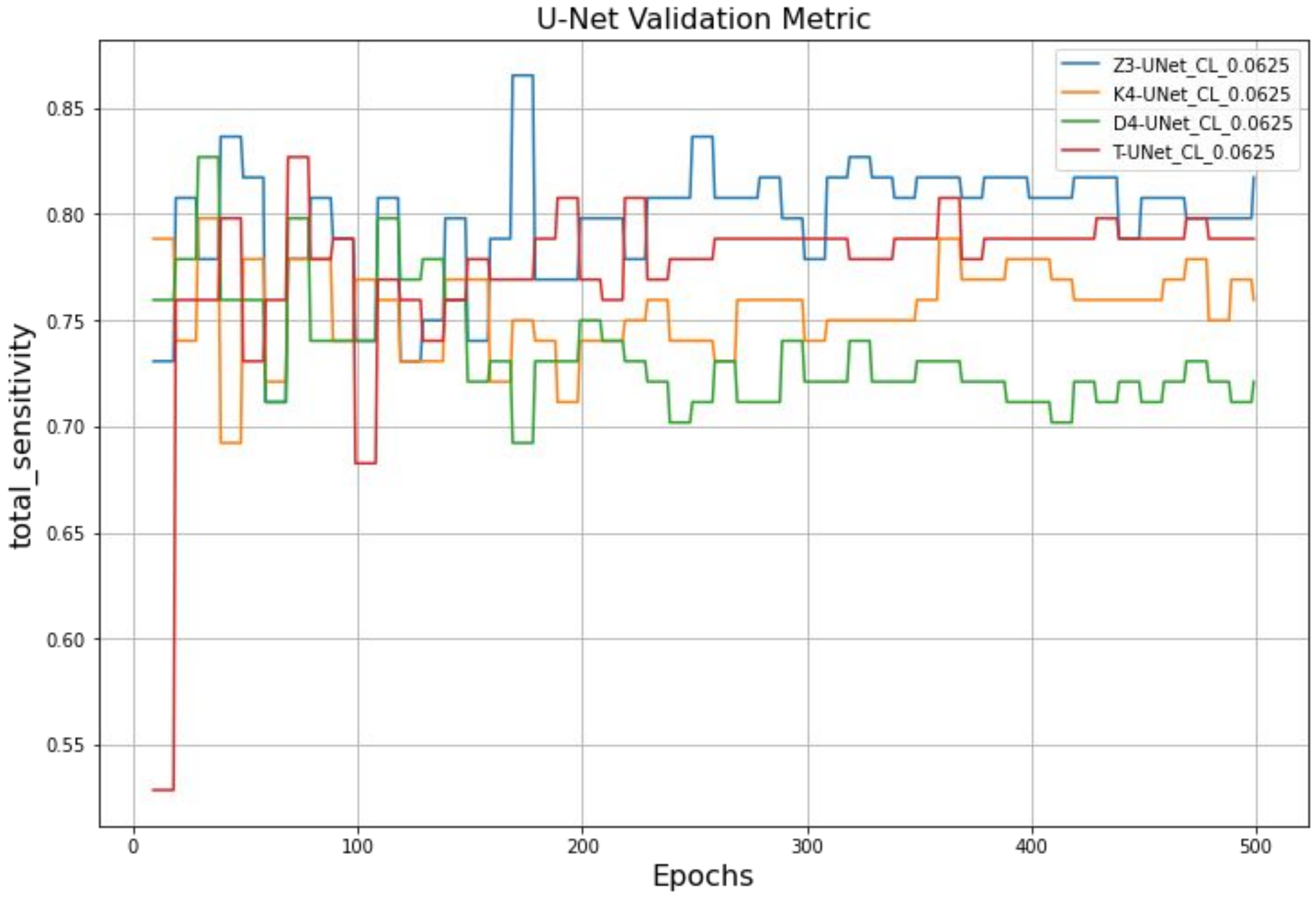
```
class GroupUNet3D(ks.Model):
    """The complete U-net segmentation model"""

    def __init__(self, group, num_classes, n_channels_in, patch_size=(96, 96, 96,),
                 c_factor=1.0, out_act='softmax', kernel_reg=regularizers.l2(1e-6),
                 filters=12, depth=4, LReLU_alpha=0.01, kernel_size=3,
                 max_pool=2, dropout=0.2, padding='same', rotation_params=(False,),
                 blur_params=(False,), transform_params=(False,), cut_off=0.9,
                 trainable=True, label_smoothing=False, label_smoothing_neighbours=5,
                 val_samples=1):
```

# Group U-Net on Truncated BraTS2019

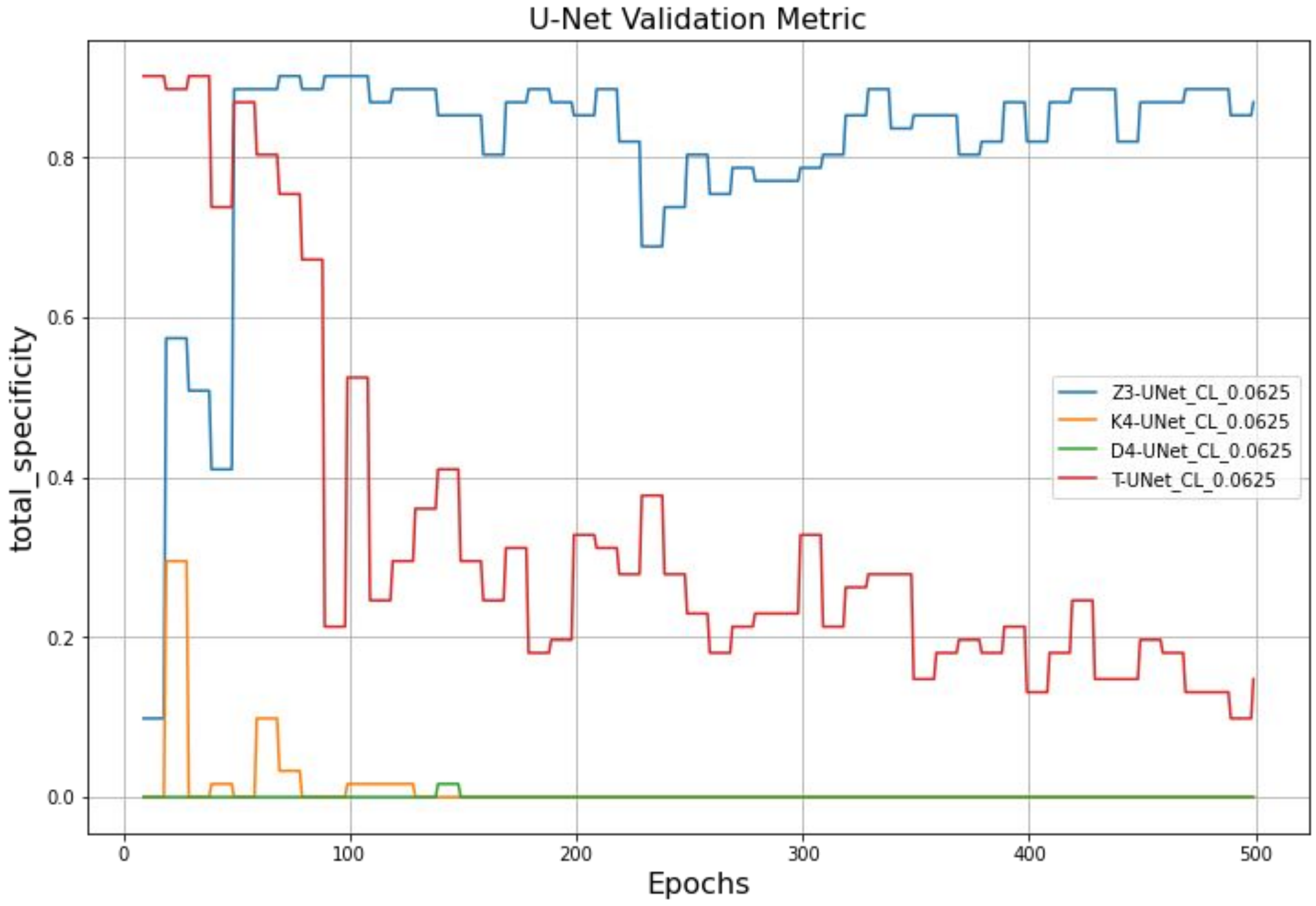
# Group U-Net on Truncated BraTS2019



# Group U-Net on Truncated BraTS2019



# Group U-Net on Truncated BraTS2019



# Group U-Net on Truncated BraTS2019

**Possible cause**
Not enough effective parameters in equivariant U-Nets

**Example**
Equivariance group: C4
Baseline U-Net: 12 filters in the first layer
C4 U-Net: 3 filters in the first layer

# Inferred Transformation VAE for Spatial Augmentation

# Inferred Transformation VAE for Spatial Augmentation

**PADDIT (Orbes-Arteaga, et al. 2019)**
Bayesian inference of a template MRI

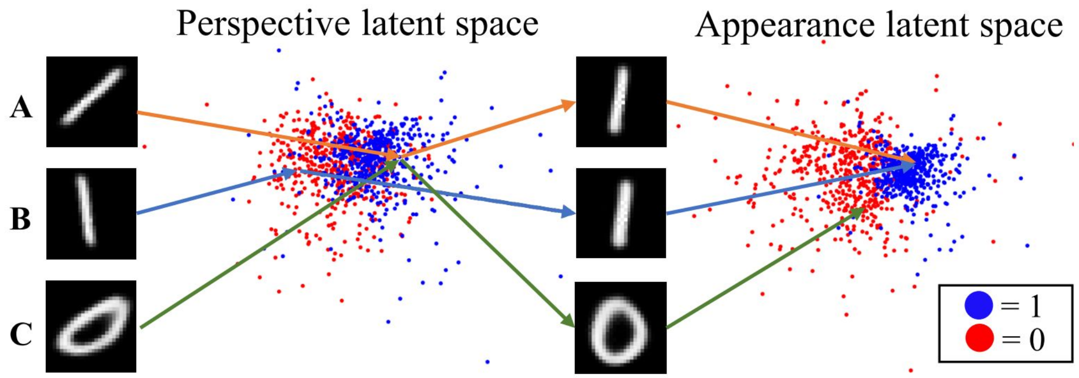
$$p(I_k|\mathbf{v}_k, I_T, \sigma) = \frac{1}{(2\pi)^{V/2}\sigma^V} \exp\left(-\frac{\|I_T - I_k \circ \mathrm{Exp}(\mathbf{v}_k)\|^2}{2\sigma^2}\right)$$

# Inferred Transformation VAE for Spatial Augmentation

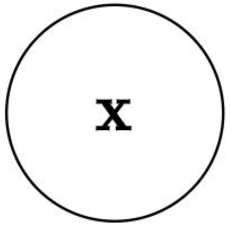
**PADDIT (Orbes-Arteaga, et al. 2019)**
Bayesian inference of a template MRI

$$p(I_k|\mathbf{v}_k, I_T, \sigma) = \frac{1}{(2\pi)^{V/2}\sigma^V}\exp\left(-\frac{\|I_T - I_k \circ \mathrm{Exp}(\mathbf{v}_k)\|^2}{2\sigma^2}\right)$$

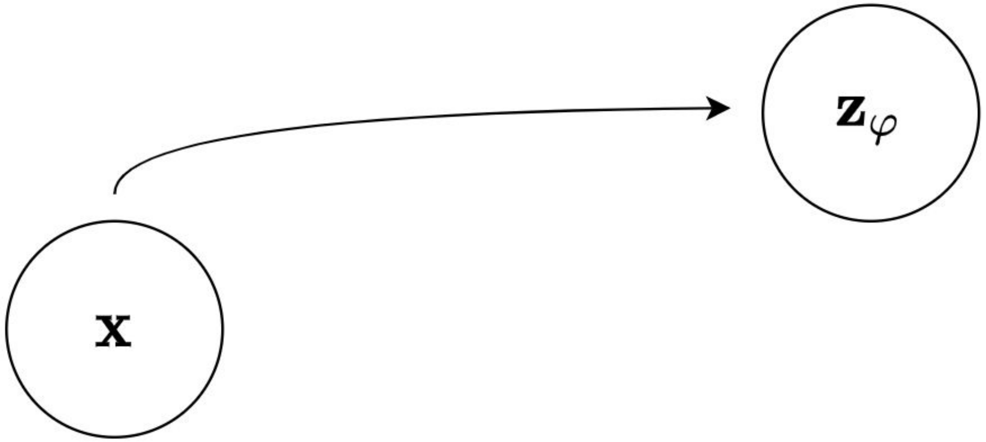
**IT-VAE (Detlefsen and Hauberg, 2019)**
Unsupervised disentanglement of shape and appearance



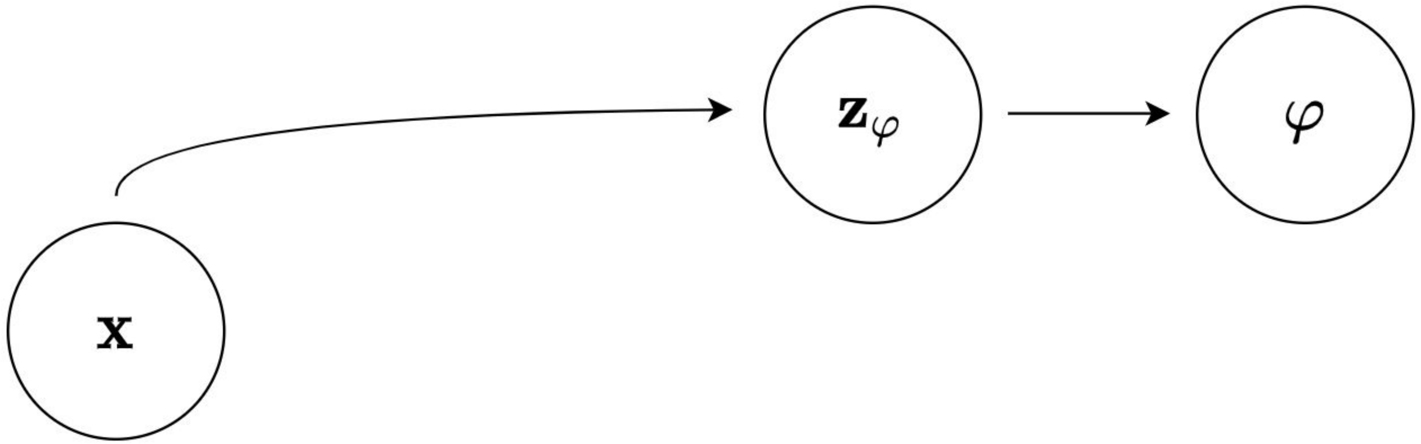
# Inferred Transformation VAE for Spatial Augmentation



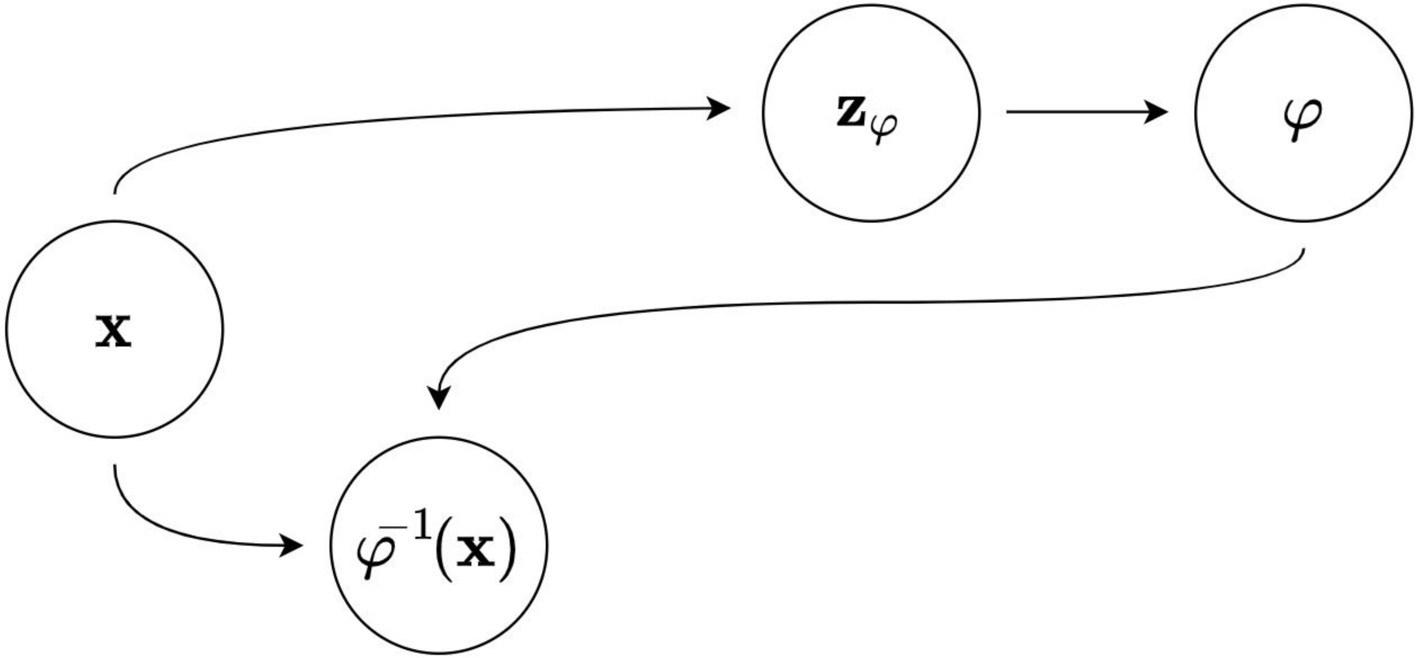
# Inferred Transformation VAE for Spatial Augmentation



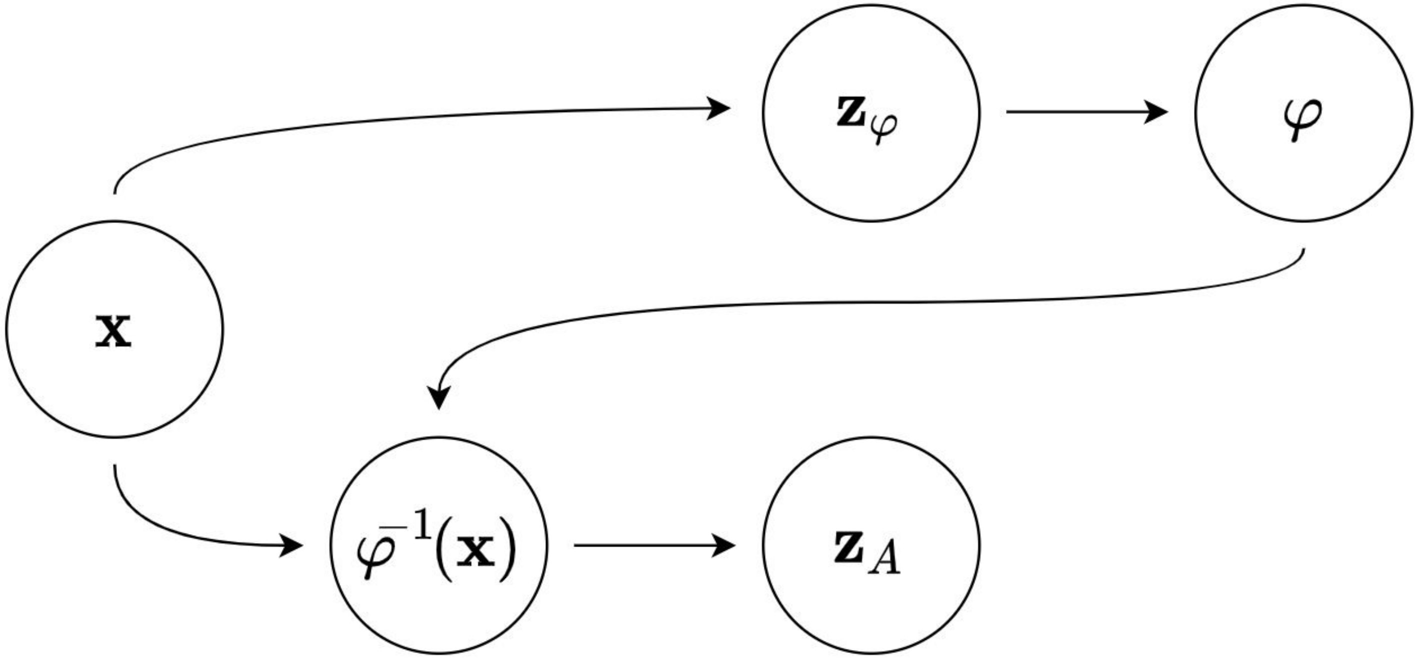
# Inferred Transformation VAE for Spatial Augmentation



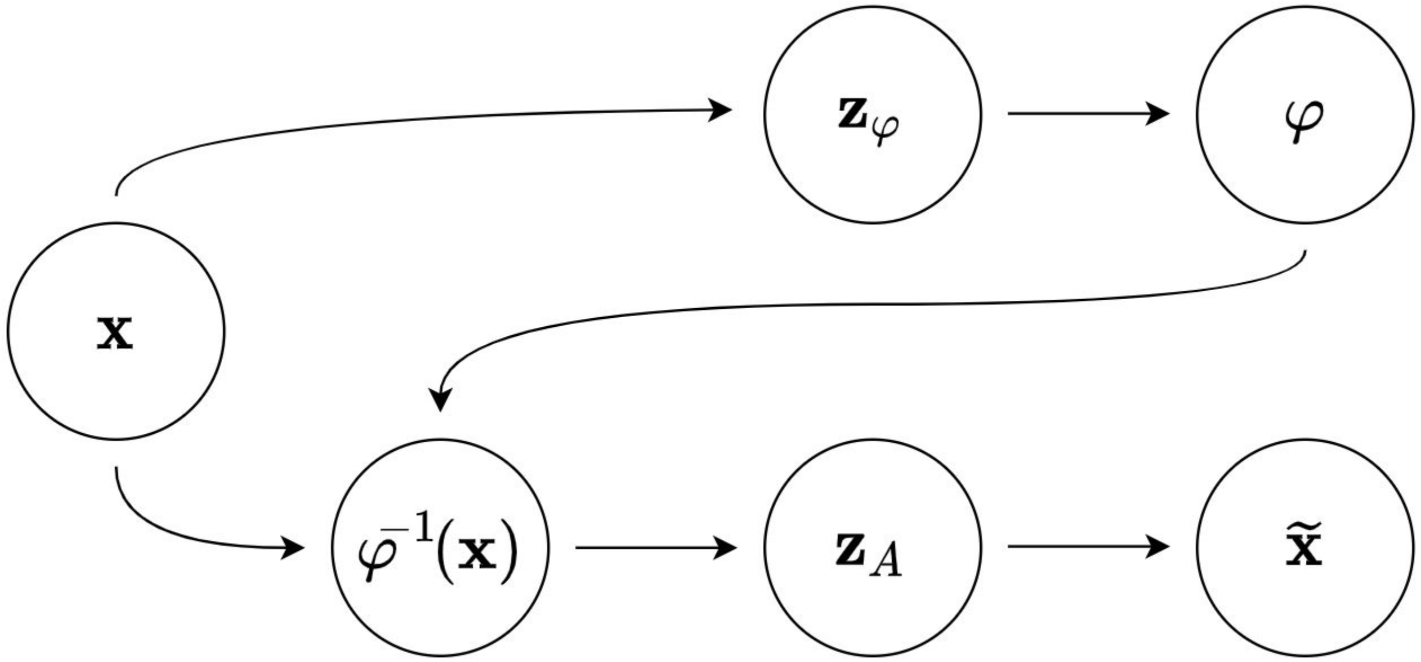
# Inferred Transformation VAE for Spatial Augmentation



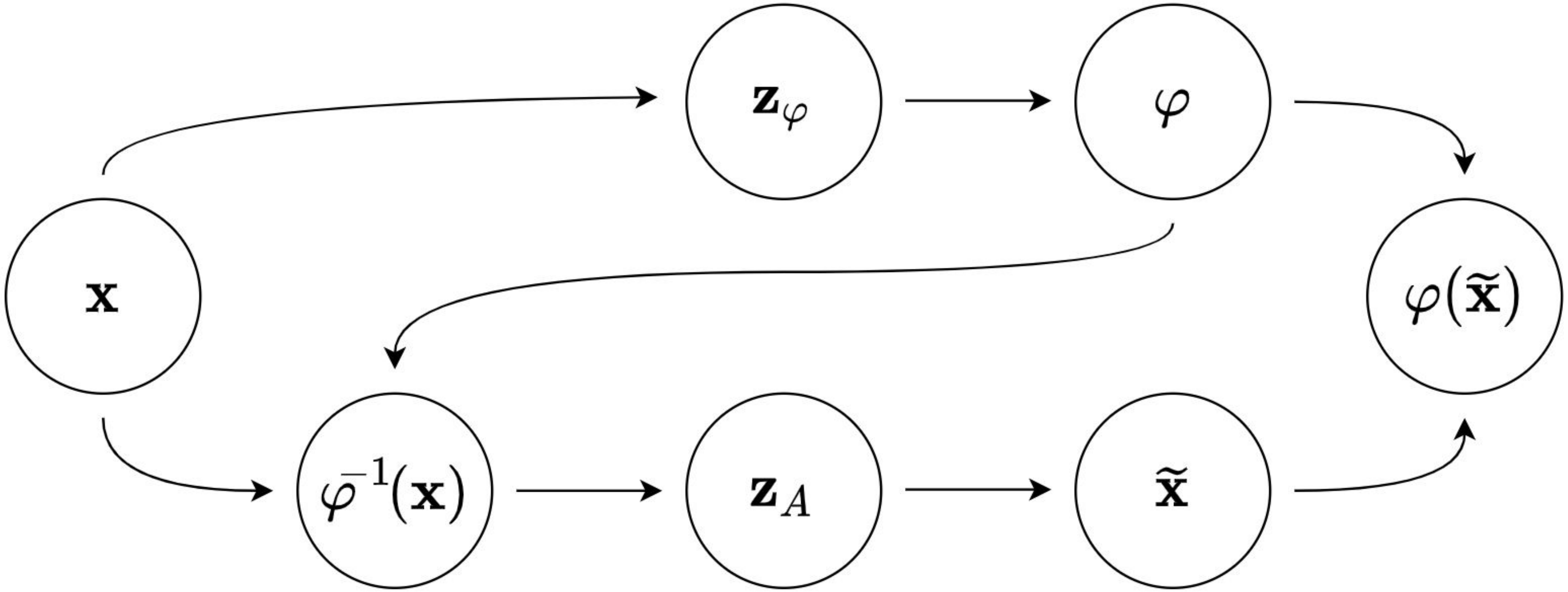
# Inferred Transformation VAE for Spatial Augmentation



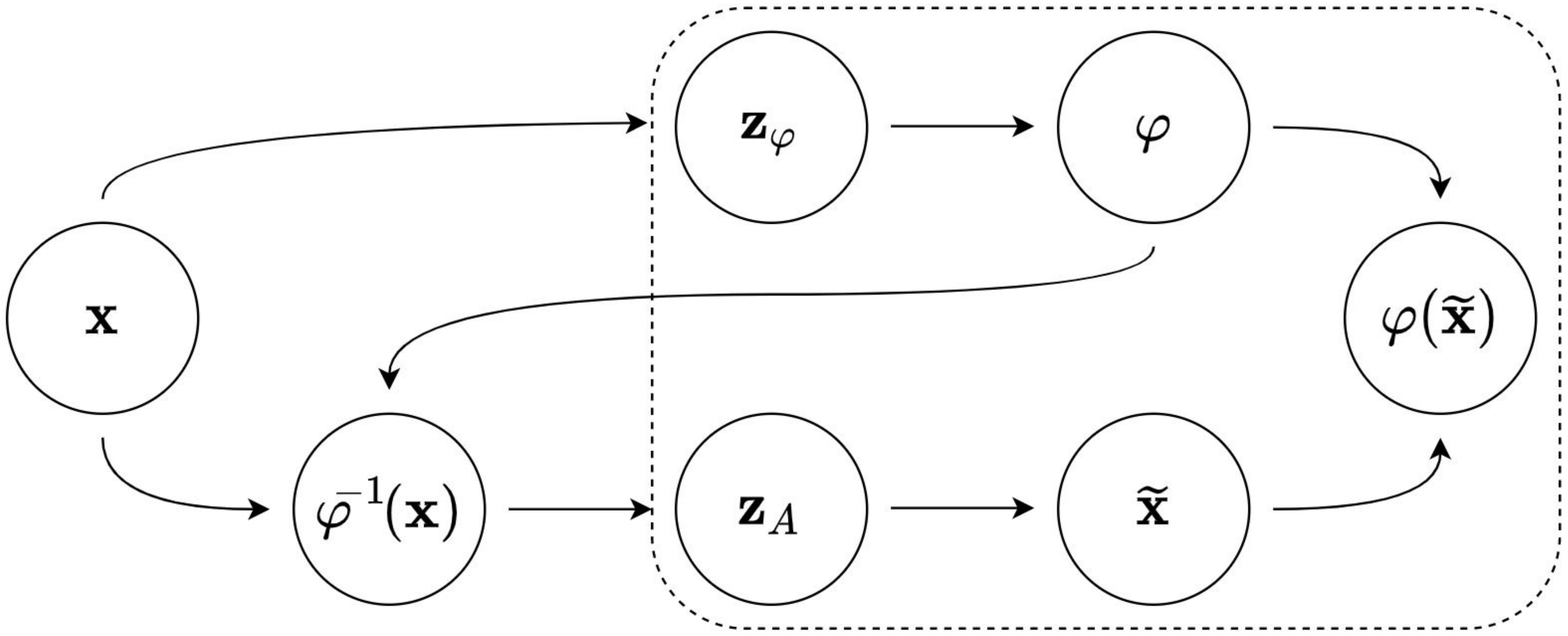
# Inferred Transformation VAE for Spatial Augmentation



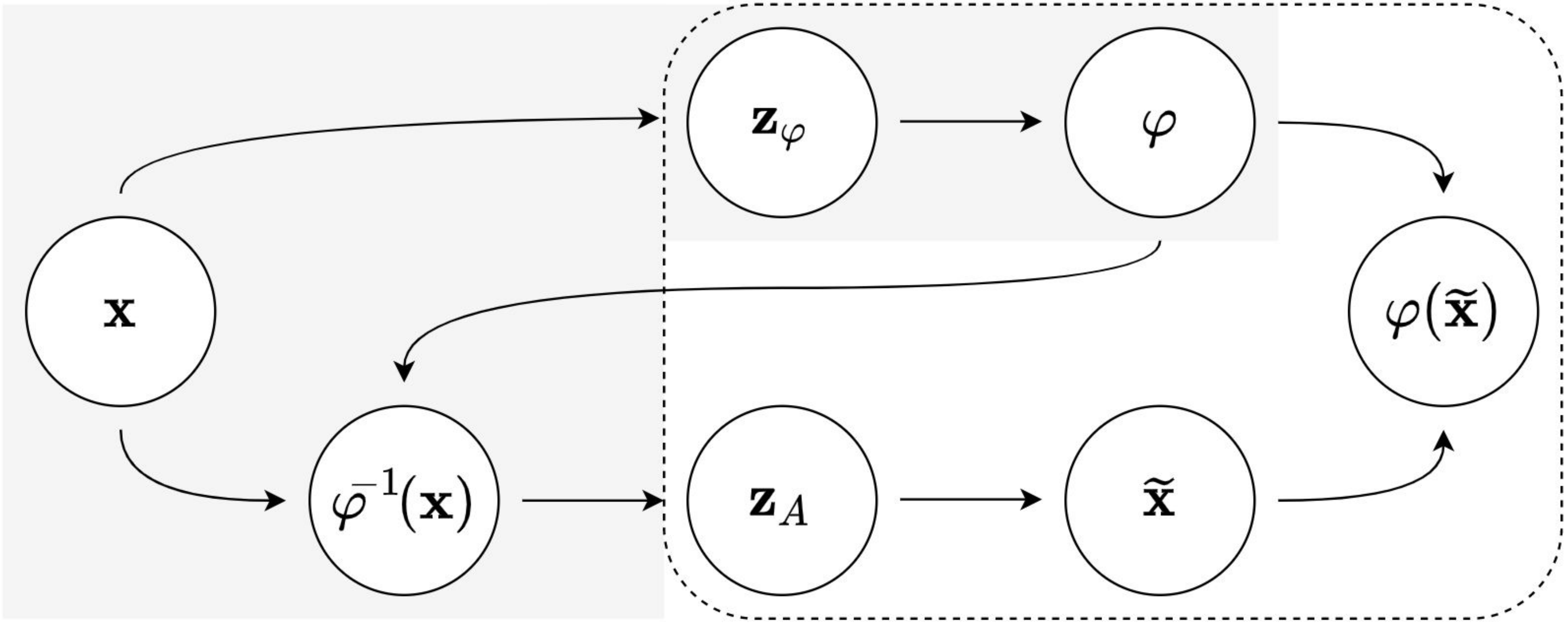
# Inferred Transformation VAE for Spatial Augmentation



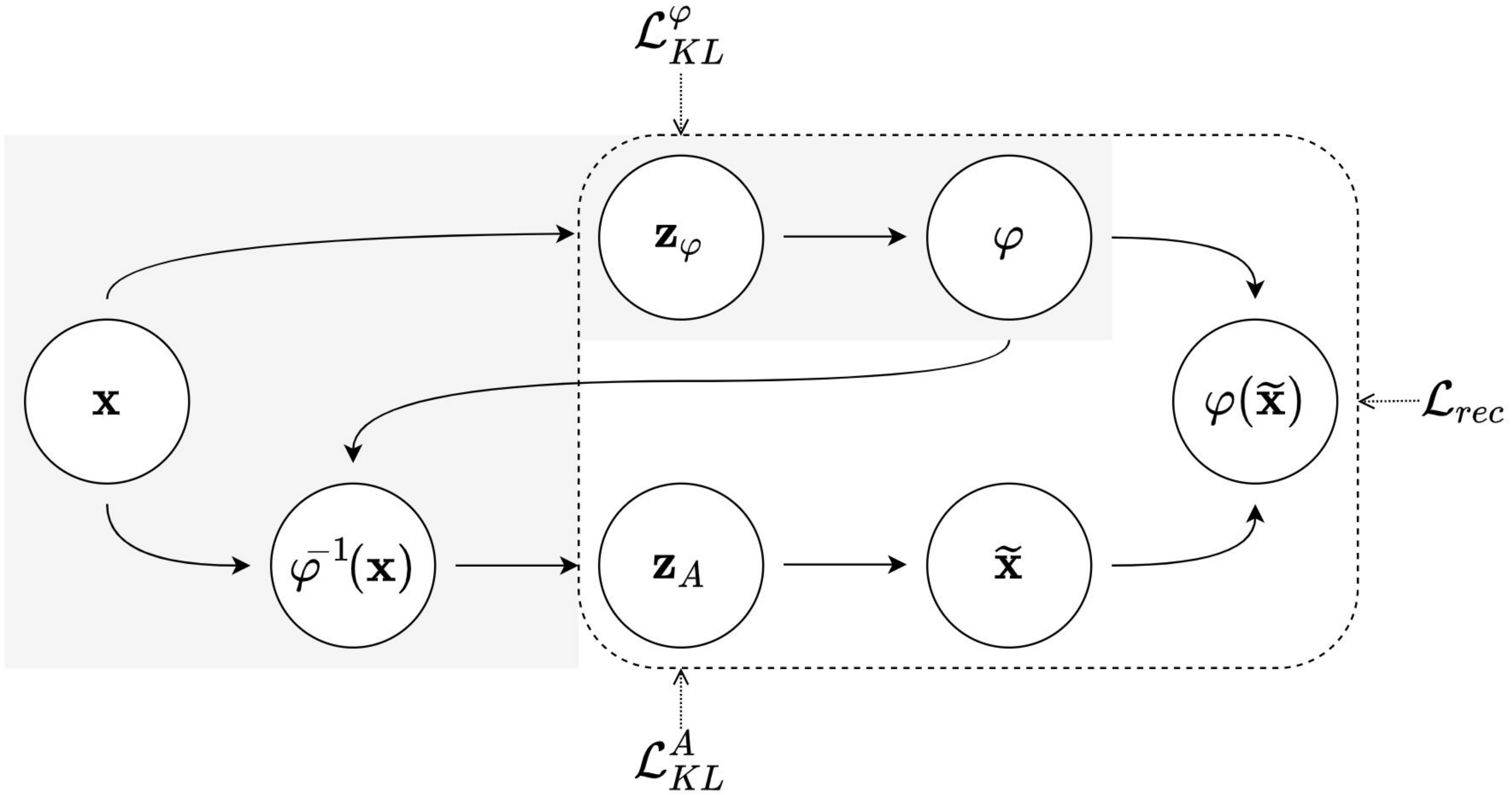
# Inferred Transformation VAE for Spatial Augmentation



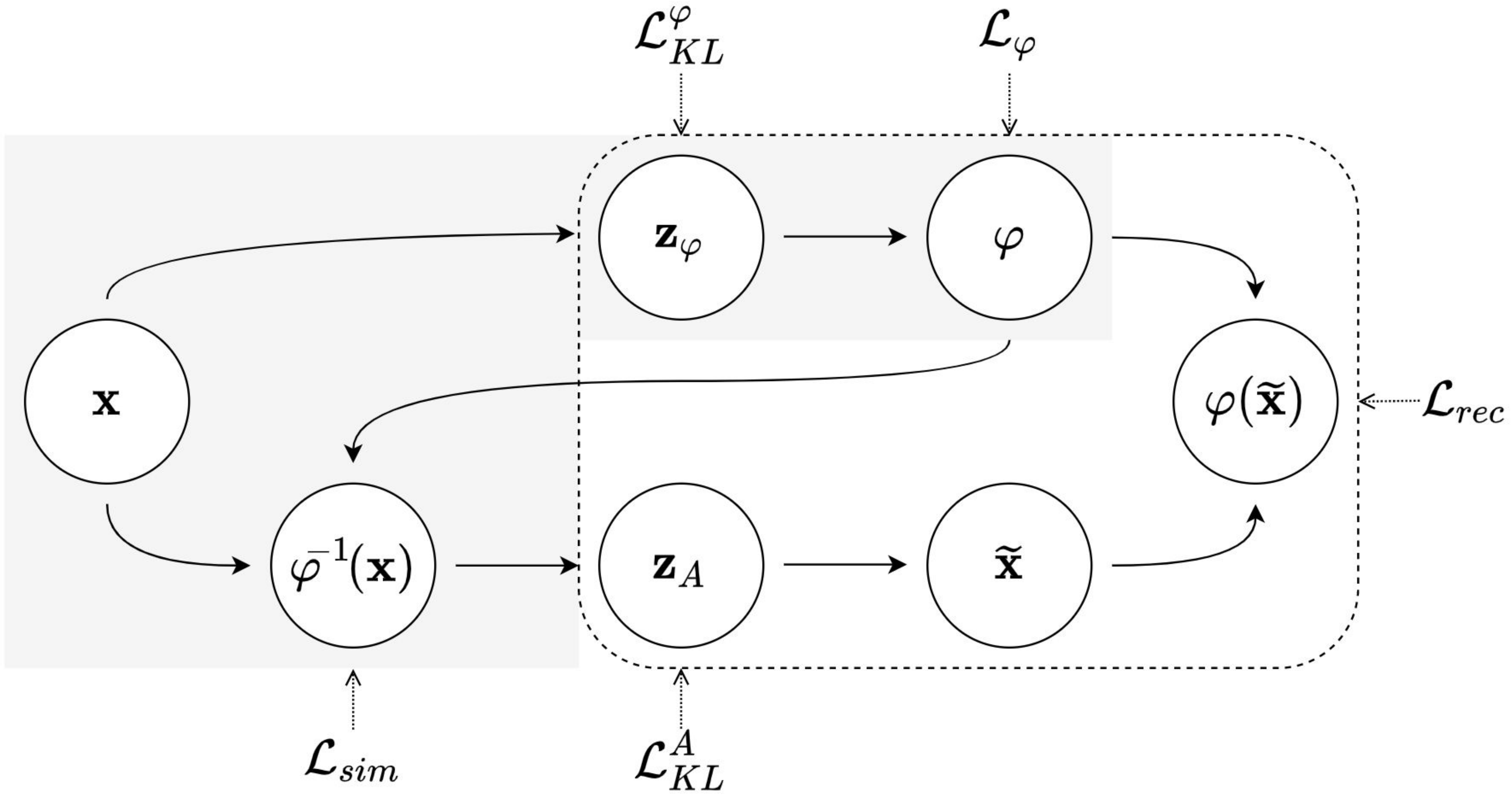
# Inferred Transformation VAE for Spatial Augmentation



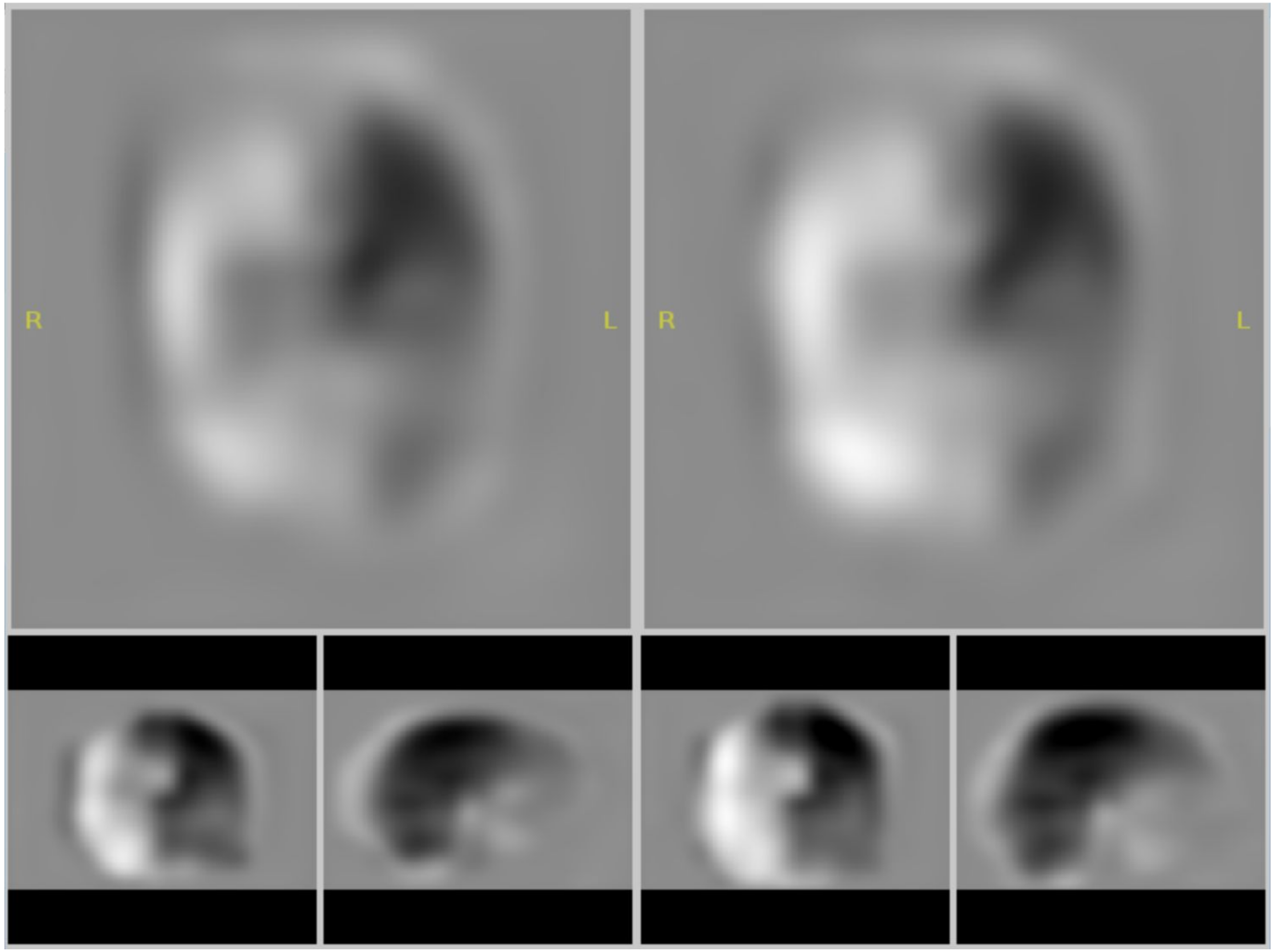
# Inferred Transformation VAE for Spatial Augmentation



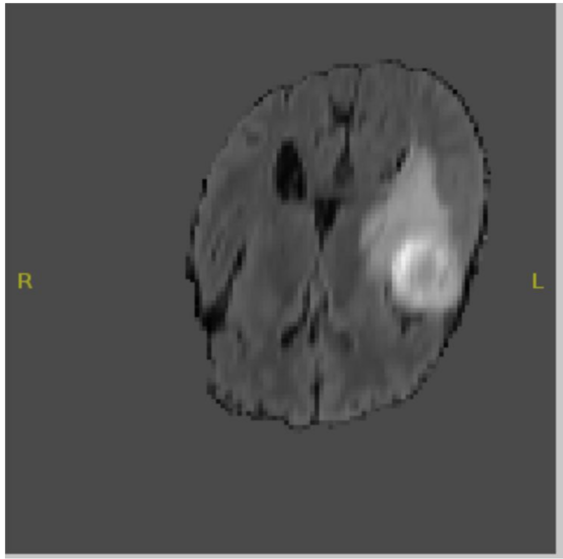
# Inferred Transformation VAE for Spatial Augmentation



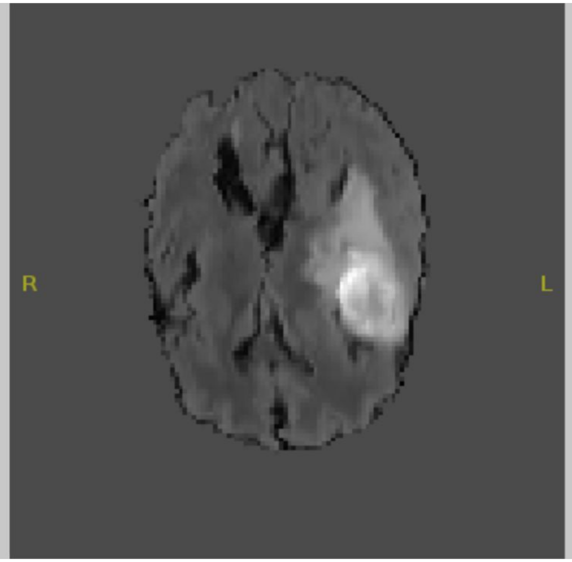
# Decoded samples from conditional latent space distributions



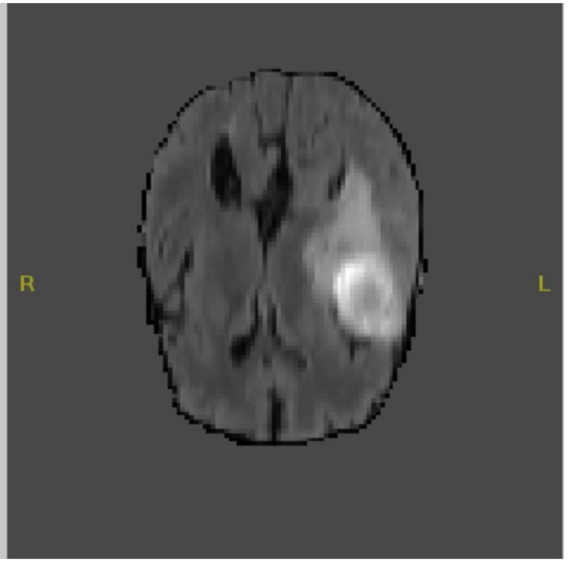
# Inferred Transformation VAE for Spatial Augmentation




Affine




Elastic




IT-VAE

# Inferred Transformation VAE for Spatial Augmentation

## Questions

- What effect do VAE augmentation pipelines have on segmentation networks?

# Inferred Transformation VAE for Spatial Augmentation

**Questions**

- What effect do VAE augmentation pipelines have on segmentation networks?
  - Naive latent space sampling

# Inferred Transformation VAE for Spatial Augmentation

## Questions

- What effect do VAE augmentation pipelines have on segmentation networks?
  - Naive latent space sampling
  - Instance-specific latent space sampling

# Inferred Transformation VAE for Spatial Augmentation

**Questions**

- What effect do VAE augmentation pipelines have on segmentation networks?
  - Naive latent space sampling
  - Instance-specific latent space sampling
  - Mixed-instance latent space sampling

# Inferred Transformation VAE for Spatial Augmentation

**Questions**

- What effect do VAE augmentation pipelines have on segmentation networks?
  - Naive latent space sampling
  - Instance-specific latent space sampling
  - Mixed-instance latent space sampling
- How well can an IT-VAE perform unsupervised groupwise registration?
  - This works for small datasets, but for MRIs?
  - Deformable autoencoders work for MRIs but are unconditioned

# Thanks!

jami@di.ku.dk