



# Normal Similarity Network For Generative Modelling

Jay Nandy, Wynne Hsu, Lee Mong Li



School of Computing

## What is Normal Similarity Network?

- A deep unsupervised generative model using Gaussian-RBF kernels.
- Adopted from CNN architectures where the layers are constructed with *learnable Gaussian-RBF filters*.
- Layers are constructed using strided-RBF filters to down-sample the images.
- Each layer of NSN captures the density of patches via likelihood maximization criterion.
- Final layer captures the density of them in a transformed domain.

## NSN-Gen : Image Generation from NSN

1. NSN-Gen iteratively reconstructs the feature maps starting from the final layer.

2. Final Layer: Sample 1-dimensional random noise vector:  $z \sim N(0, 1)$

3. Hidden Layers: Two step iterative process.  
(a.)  $W_k$  Generation: Use the channel vectors of  $F_{k-1}$  to sample weights for the  $F_k$  patches.

(b.)  $F_k$  Generation:  
• Generate  $F_k$  patches using weighted combinations of filter samples.  
• Stitch these patches to obtain  $F_k$ .

Generate the image in the input layer.

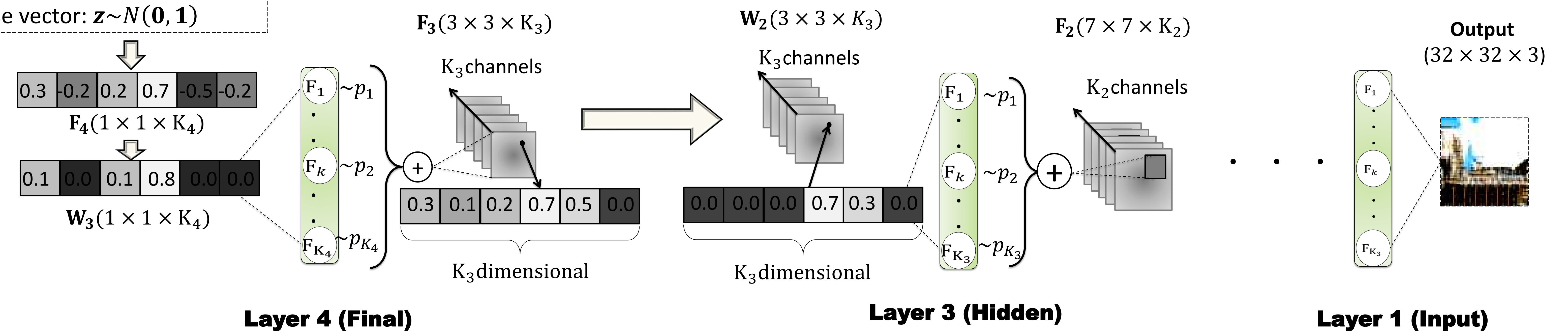
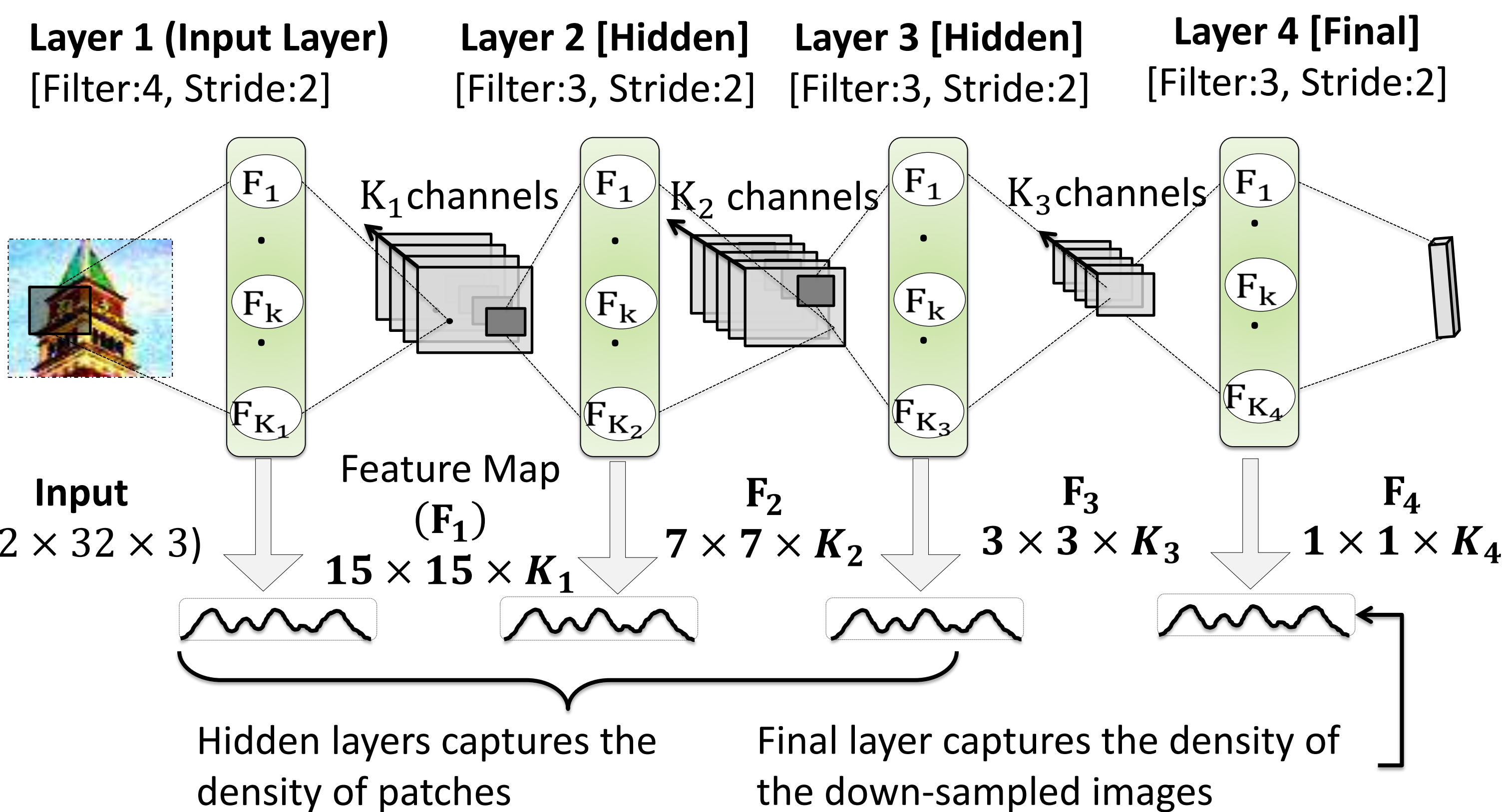


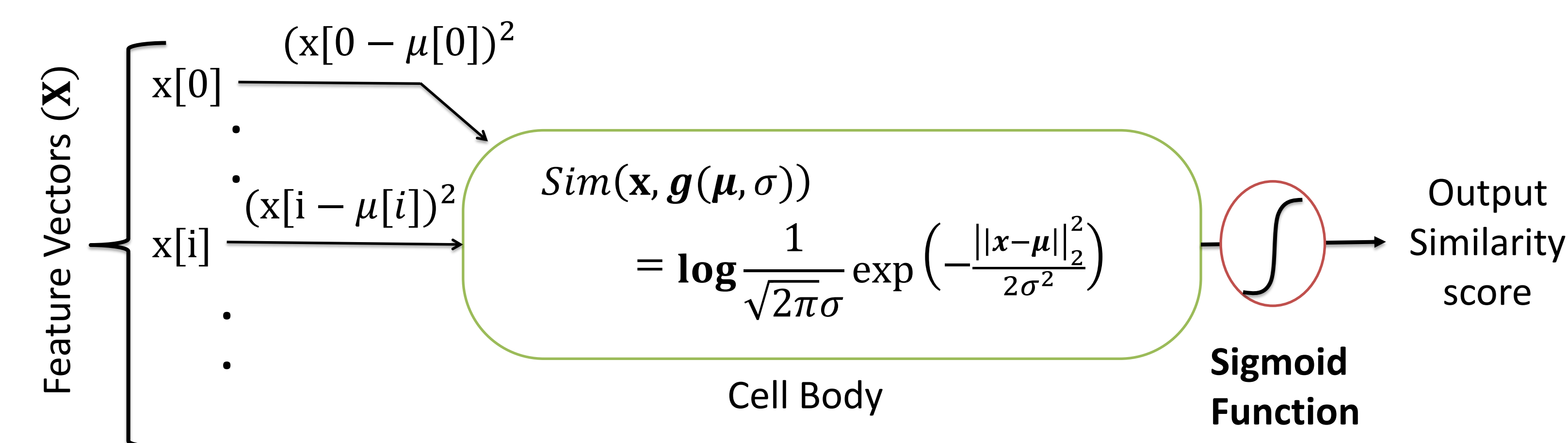
Figure: An example of a 4-layered NSN



## Training Procedure:

- Layer-wise training using a non-parametric variant of Expectation-Maximization algorithm.
- Automatically learns required number of filters.

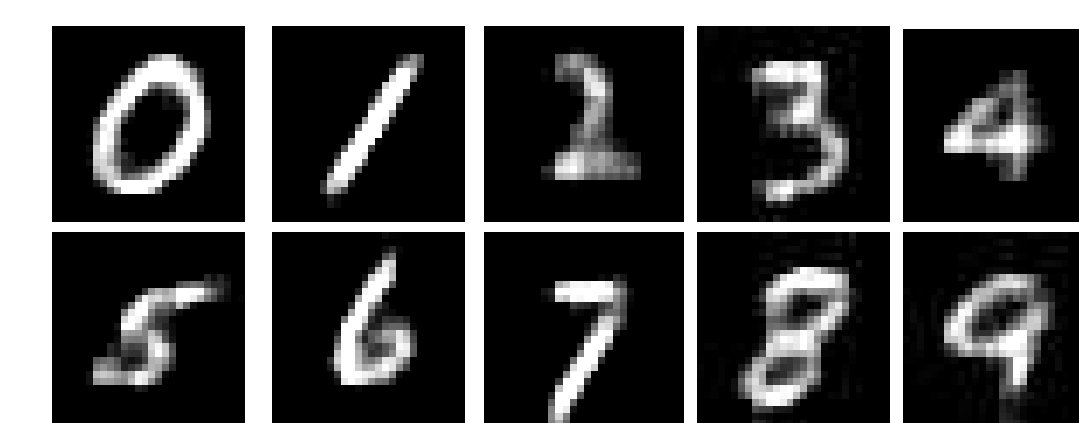
Figure: Schematic diagram of the RBF-filters for NSN



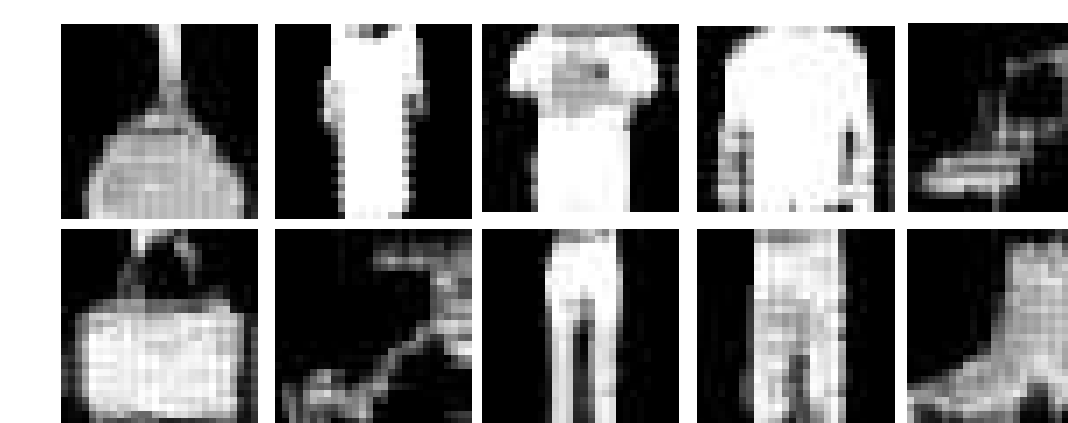
## Generated Image Samples

Table: Comparison of Inception Scores of NSNs with GAN models

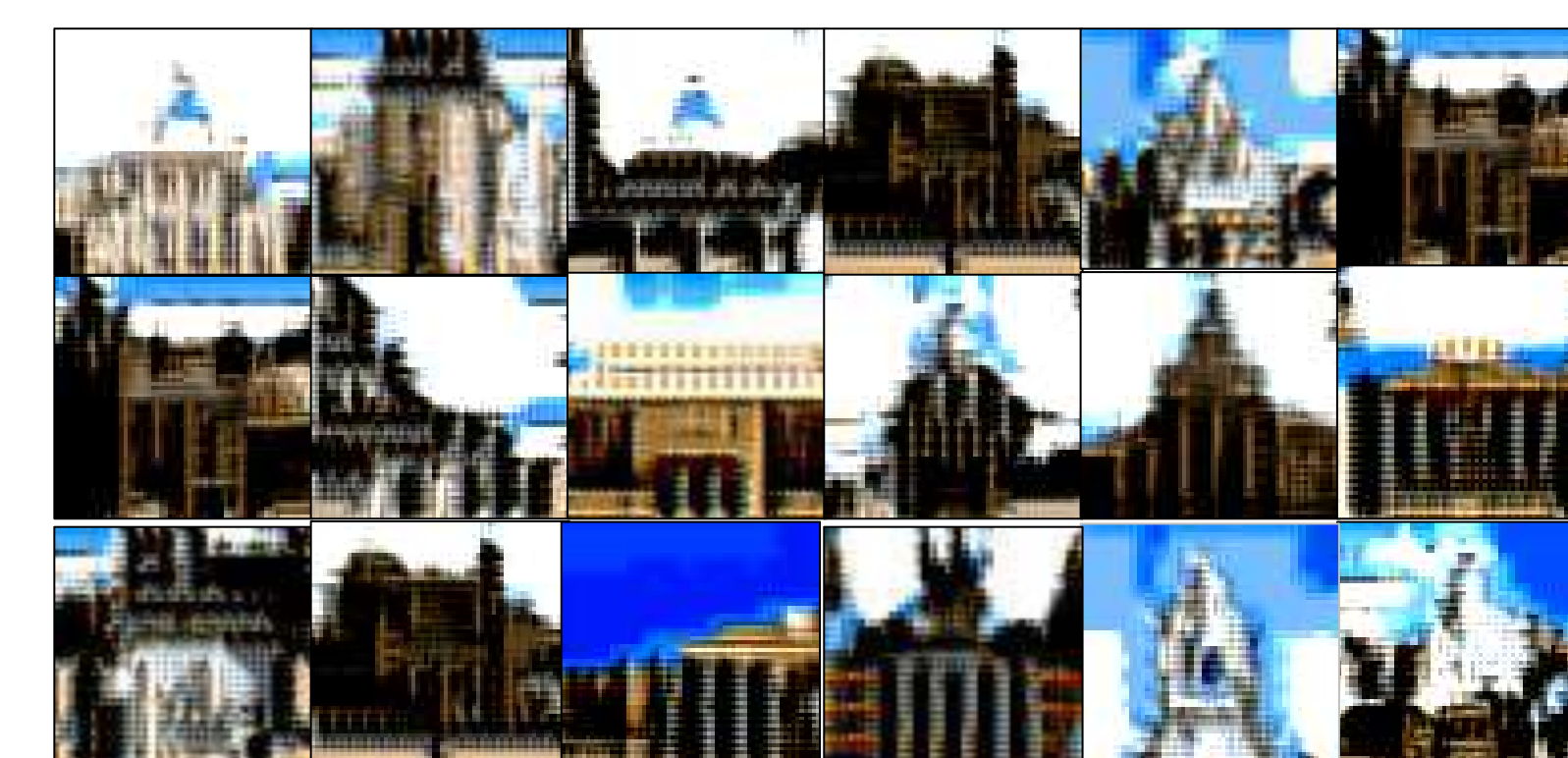
	MNIST	MNIST-F	LFW	Church
Real Data	1.99 ± 0.01	4.25 ± 0.05	3.64 ± 0.28	2.70 ± 0.04
DCGAN	2.27 ± 0.01	4.15 ± 0.04	2.67 ± 0.03	2.89 ± 0.02
WGAN	2.02 ± 0.02	3.00 ± 0.03	2.94 ± 0.03	3.05 ± 0.02
LSGAN	2.03 ± 0.01	<b>4.45 ± 0.03</b>	2.33 ± 0.02	<b>3.44 ± 0.04</b>
<b>NSN (ours)</b>	<b>2.44 ± 0.02</b>	3.85 ± 0.04	<b>3.04 ± 0.10</b>	2.99 ± 0.02



Generated MNIST samples



Generated MNIST-F samples



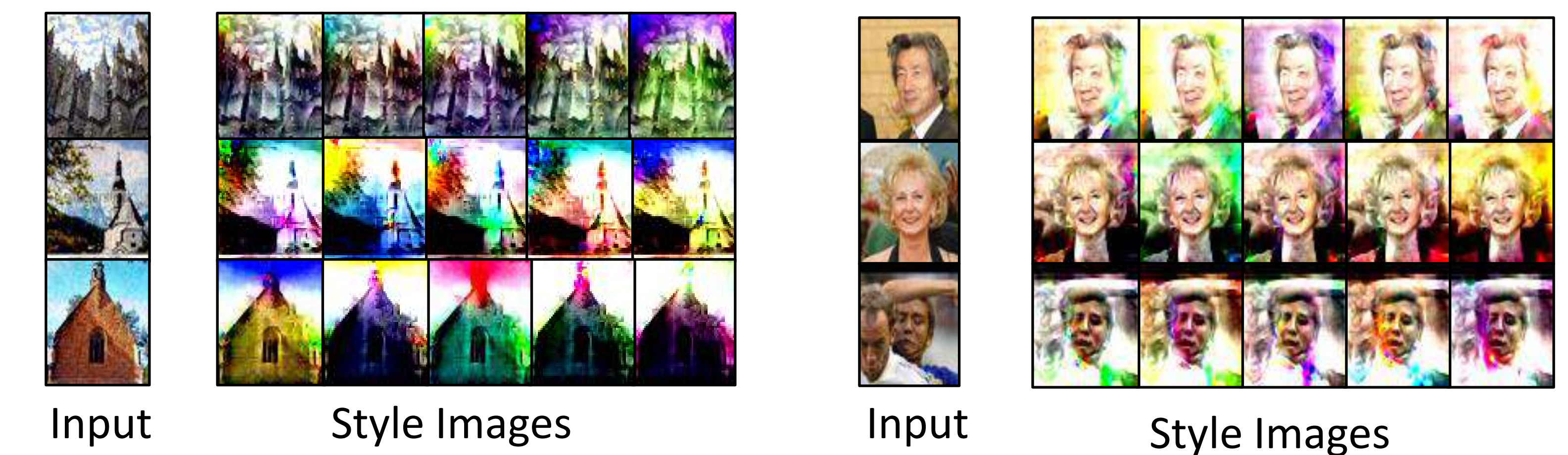
Church samples



Human Faces (LFW) Samples

## More Applications of NSN

### Image Styling



### Reconstructing Occluded Images

