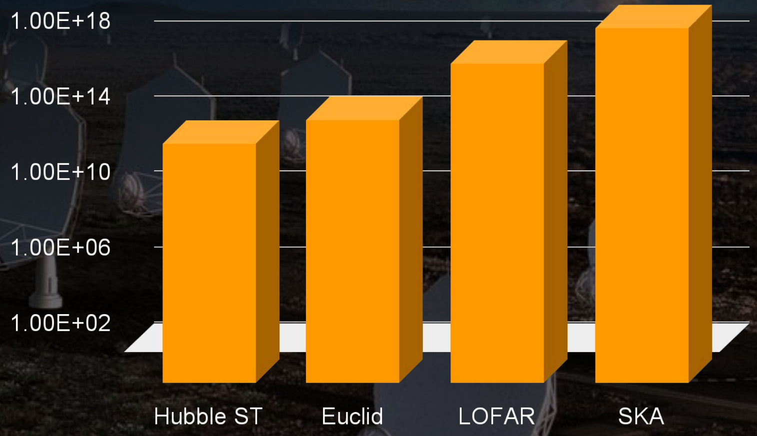
# Scattering-transform-based Galaxy Classification and Image Generation


**Markus Bredberg**
markus.bredberg@epfl.ch
PhD student, LASTRO, EPFL,
Supervisor: Emma Tolley

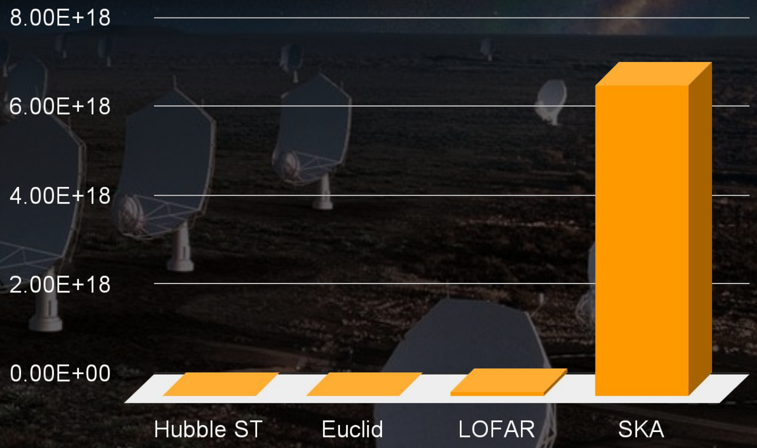
10th of June 2024

# Next generation astronomy will have to deal with a lot of data



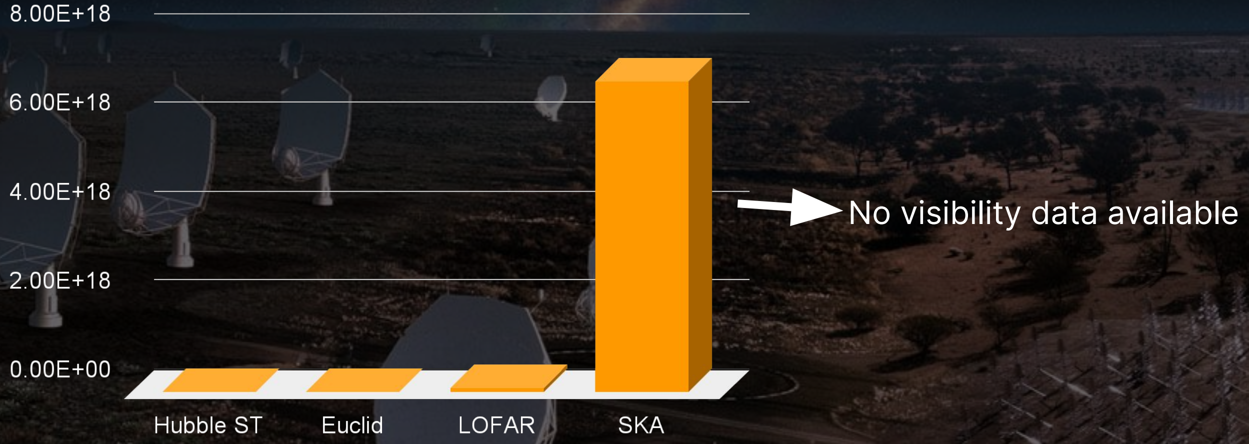

Credit: NASA, ESA, IRA, SKAO

# Next generation astronomy will have to deal with a lot of data



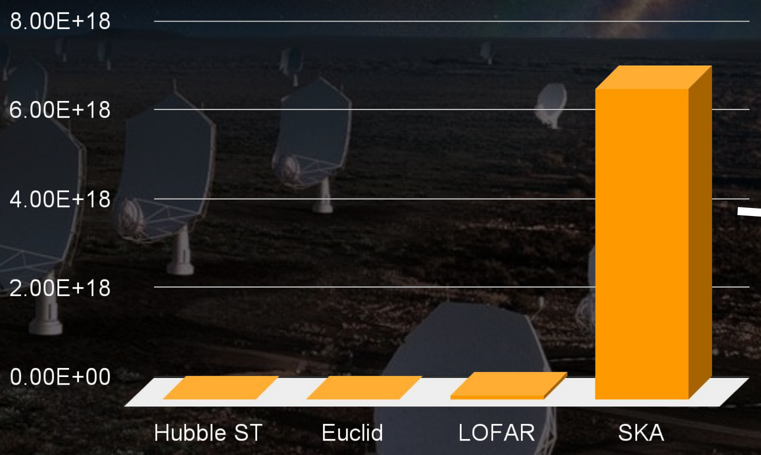

Credit: NASA, ESA, IRA, SKAO

# Next generation astronomy will have to deal with a lot of data




Credit: NASA, ESA, IRA, SKAO

# Next generation astronomy will have to deal with a lot of data

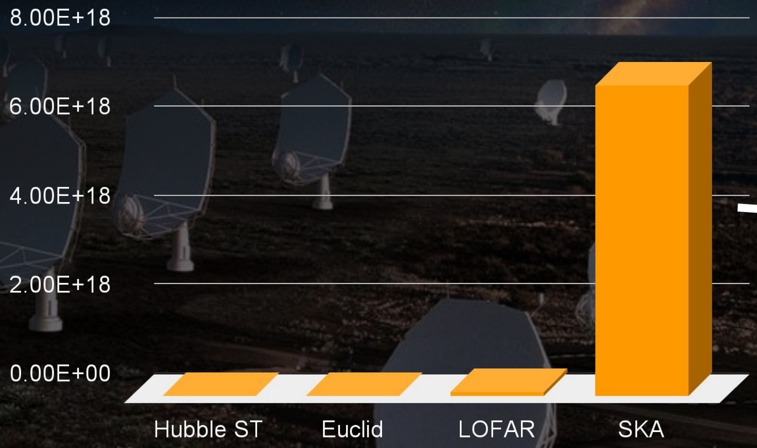



Very time consuming to categorise, label, or tag sources manually

No visibility data available

Credit: NASA, ESA, IRA, SKAO

# Next generation astronomy will have to deal with a lot of data



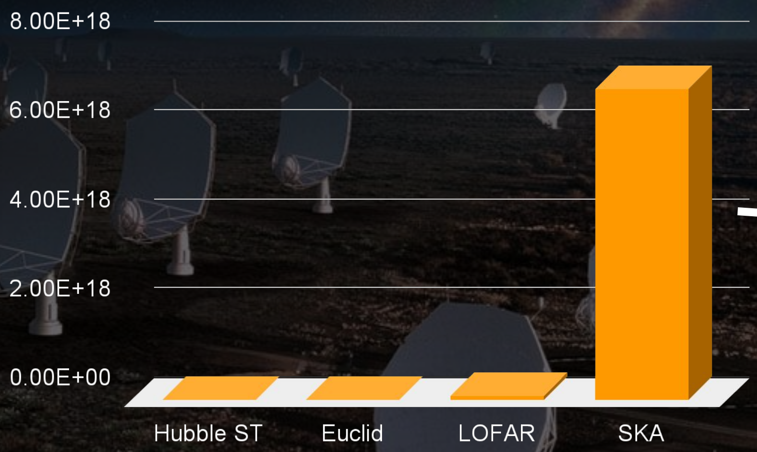

Credit: NASA, ESA, IRA, SKAO

Very time consuming to categorise, label, or tag sources manually

Machine learning

No visibility data available

# Next generation astronomy will have to deal with a lot of data




Credit: NASA, ESA, IRA, SKAO

Very time consuming to categorise, label, or tag sources manually

Machine learning

No visibility data available

Image-based machine learning

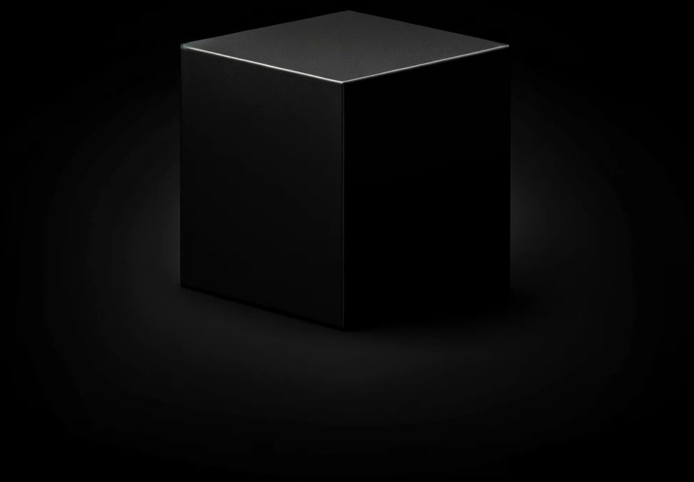
# I hope to ease this transition to machine learning

## Goals:

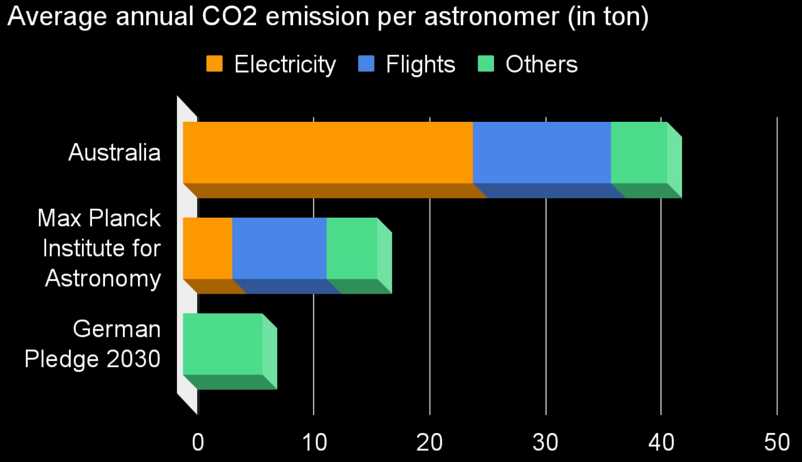
- **Generate mock catalogues** for ML classification
- **Improve classification** of galaxies and radio emission
- **Image reconstruction** and remove artefacts or noise in an image
- **Spillover**: methods can be used for other aims

# Machine learning is not human learning

What's in the black box?



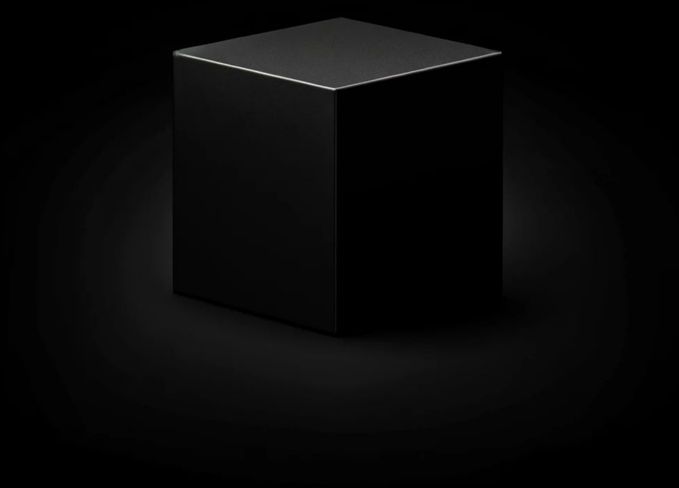
Large-scale computing is expensive



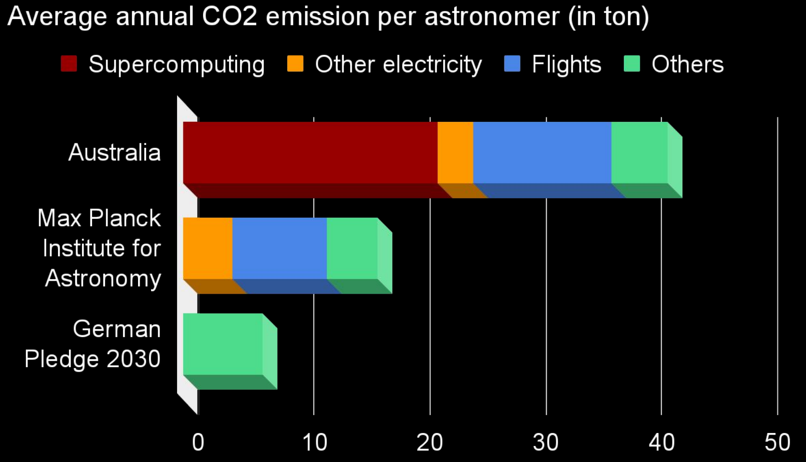

Stevens, A.R.H., Bellstedt, S., Elahi, P.J. et al. 2020. https://doi.org/10.1038/s41550-020-1169-

# Machine learning is not human learning

What's in the black box?

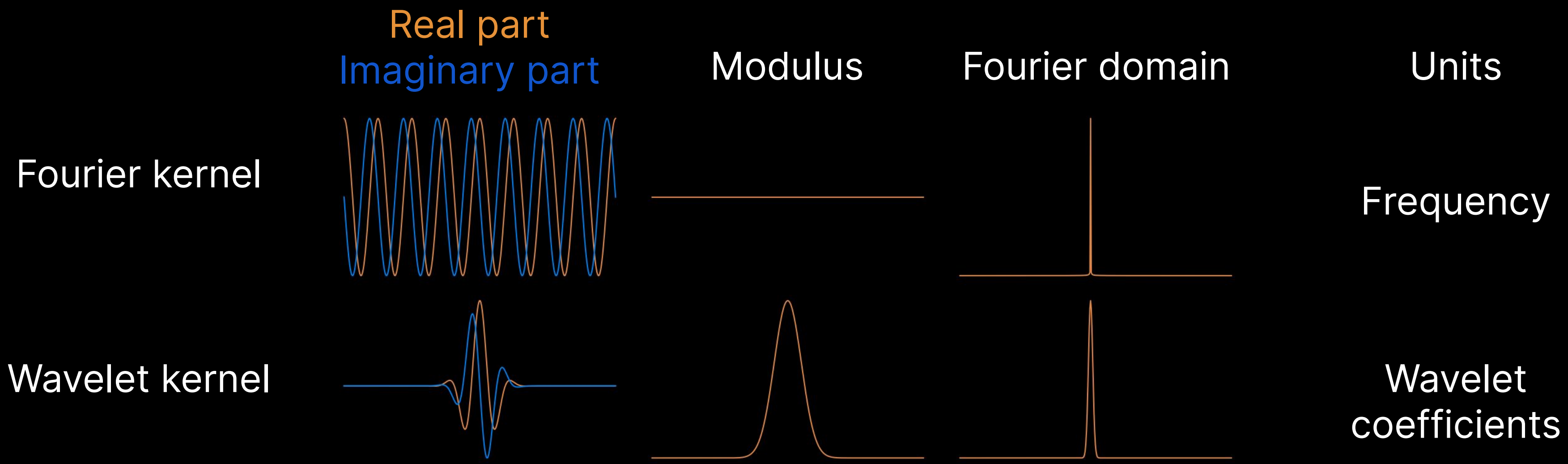


Large-scale computing is expensive

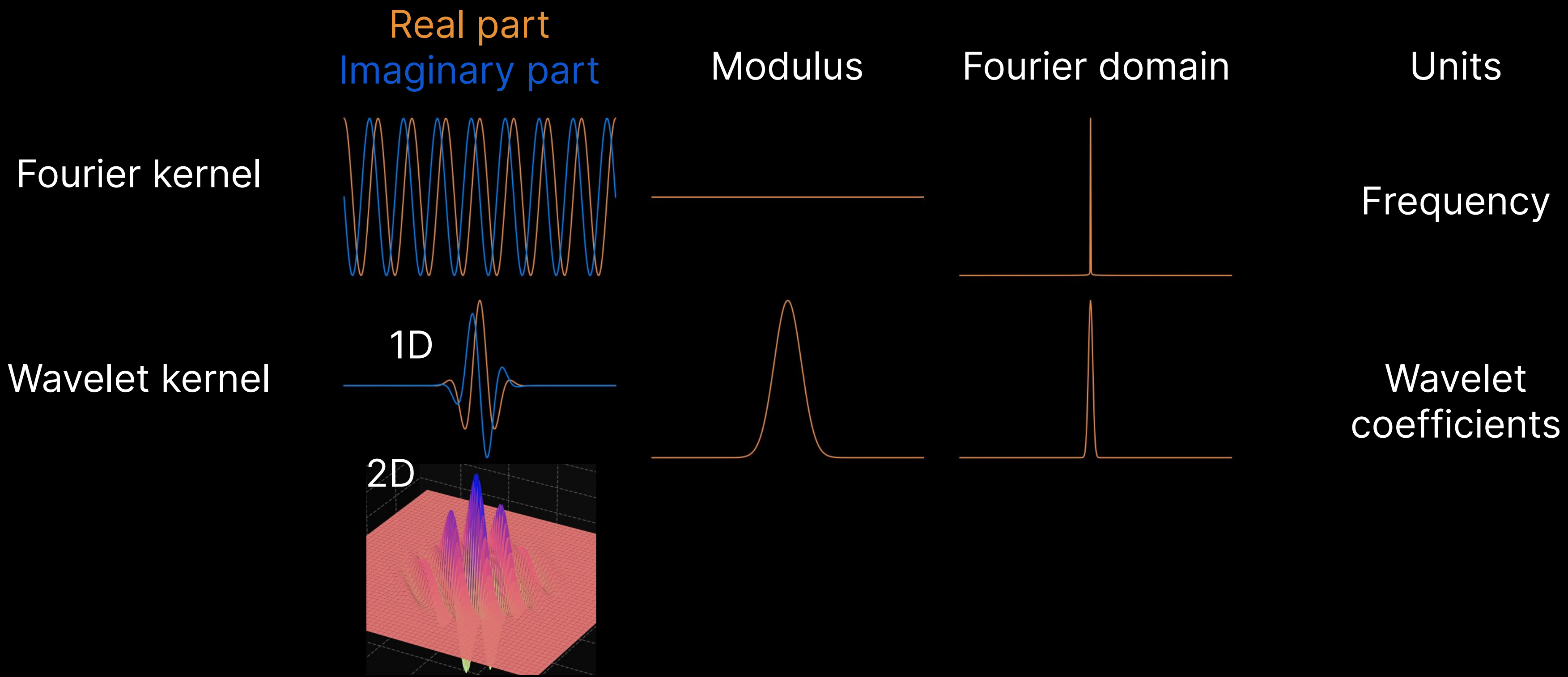



Stevens, A.R.H., Bellstedt, S., Elahi, P.J. et al. 2020. https://doi.org/10.1038/s41550-020-1169-

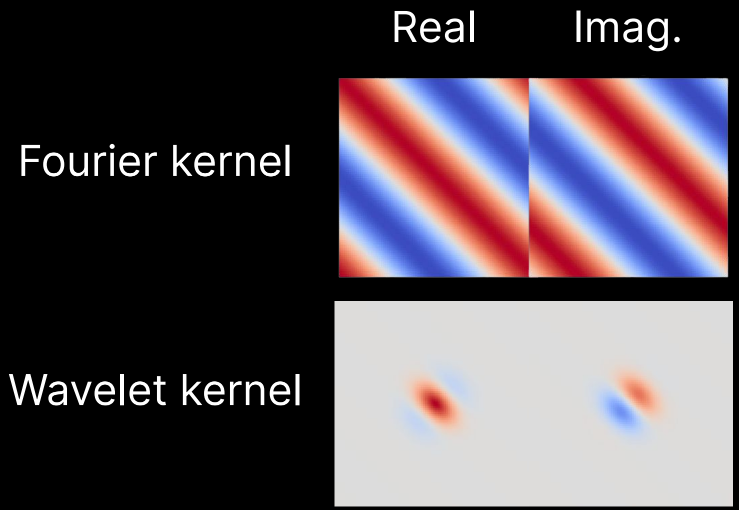
# The scattering transform: an alternative to Fourier transforms and neural networks



# The scattering transform: an alternative to Fourier transforms and neural networks



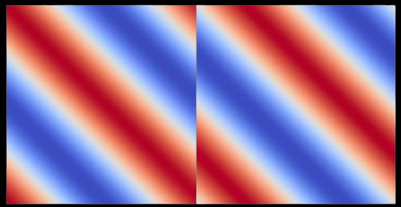
# The scattering transform accounts for spatial information



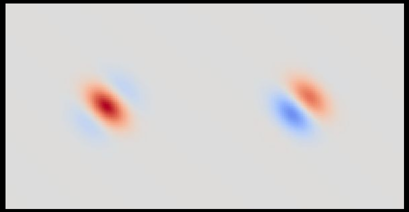
# The scattering transform accounts for spatial information

Real Imag.

Fourier kernel



Wavelet kernel



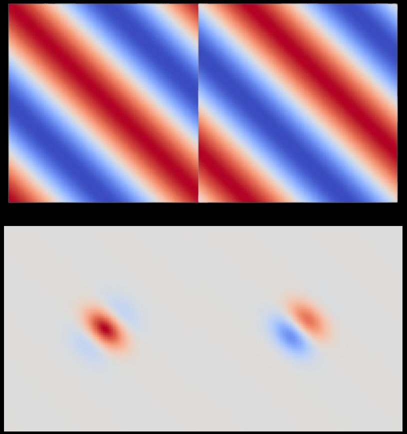
Spatial information

# The scattering transform accounts for spatial information

Real Imag.

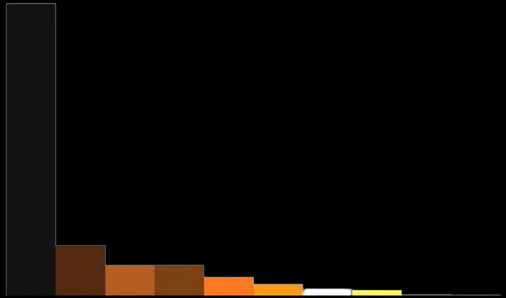
Fourier kernel

Wavelet kernel



Spatial information

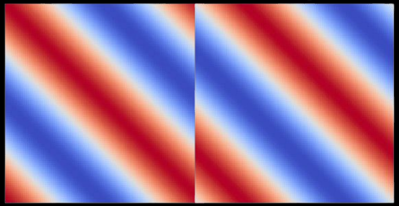
# The scattering transform accounts for spatial information



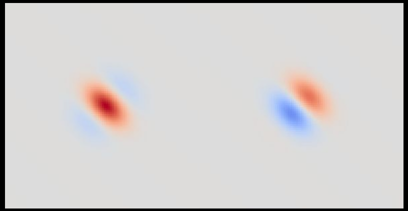
Real

Imag.

Fourier kernel



Wavelet kernel



Spatial information

# The scattering transform is an iterative wavelet transform

Wavelet transform $$WX = X \star \Psi_j(t)$$

Scattering transform $$SX = W|WX|$$

# The scattering transform is an iterative wavelet transform

Wavelet transform $$WX = X \star \Psi_j(t)$$

Scattering transform $$SX = W|WX|$$

Zeroth order $$S_0 = X \star \Phi$$

First order $$S_1(j_1, l_1) = \langle |X \star \Psi_{j_1,l_1}| \star \Phi \rangle$$

Second order $$S_2(j_1, j_2, l_1, l_2) = \langle ||X \star \Psi_{j_1,l_1}| \star \Psi_{j_2,l_2}| \star \Phi \rangle$$

n-th order $$S_n(j_1, j_2, ..., l_n, l_n) = \langle ||X \star \Psi_{j_1,l_1}| \star ...| \star \Psi_{j_n,l_n}| \star \Phi \rangle$$

# The scattering transform is an iterative wavelet transform

Wavelet transform $$WX = X \star \Psi_j(t)$$

Scattering transform $$SX = W|WX|$$

Most diffuse energy

Zeroth order $$S_0 = X \star \Phi$$

First order $$S_1(j_1, l_1) = \langle |X \star \Psi_{j_1, l_1}| \star \Phi \rangle$$

Second order $$S_2(j_1, j_2, l_1, l_2) = \langle ||X \star \Psi_{j_1, l_1}| \star \Psi_{j_2, l_2}| \star \Phi \rangle$$

n-th order $$S_n(j_1, j_2, ..., l_n, l_n) = \langle ||X \star \Psi_{j_1, l_1}| \star ...| \star \Psi_{j_n, l_n}| \star \Phi \rangle$$

# The scattering transform is an iterative wavelet transform

Wavelet transform $WX = X \star \Psi_j(t)$

Scattering transform $SX = W|WX|$

Most diffuse energy

Zeroth order $S_0 = X \star \Phi$

First order $S_1(j_1, l_1) = \langle |X \star \Psi_{j_1,l_1}| \star \Phi \rangle$

Second order $S_2(j_1, j_2, l_1, l_2) = \langle ||X \star \Psi_{j_1,l_1}| \star \Psi_{j_2,l_2}| \star \Phi \rangle$

n-th order $S_n(j_1, j_2, ..., l_n, l_n) = \langle ||X \star \Psi_{j_1,l_1}| \star ...| \star \Psi_{j_n,l_n}| \star \Phi \rangle$

Scale (1-J)

Direction (1-L)

# The scattering transform is an iterative wavelet transform

Wavelet transform $$WX = X \star \Psi_j(t)$$

Scattering transform $$SX = W|WX|$$

Most diffuse energy

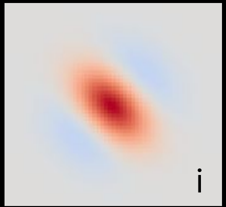
Zeroth order $$S_0 = X \star \Phi$$

First order $$S_1(j_1, l_1) = \langle |X \star \Psi_{j_1, l_1}| \star \Phi \rangle$$

Second order $$S_2(j_1, j_2, l_1, l_2) = \langle ||X \star \Psi_{j_1, l_1}| \star \Psi_{j_2, l_2}| \star \Phi \rangle$$

n-th order $$S_n(j_1, j_2, ..., l_n, l_n) = \langle ||X \star \Psi_{j_1, l_1}| \star ...| \star \Psi_{j_n, l_n}| \star \Phi \rangle$$

Scale (1-J)




Direction (1-L)

# The scattering transform is an iterative wavelet transform

Wavelet transform $$WX = X \star \Psi_j(t)$$

Scattering transform $$SX = W|WX|$$

Most diffuse energy

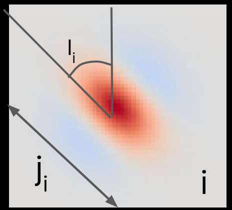
Zeroth order $$S_0 = X \star \Phi$$

First order $$S_1(j_1, l_1) = \langle |X \star \Psi_{j_1,l_1}| \star \Phi \rangle$$

Second order $$S_2(j_1, j_2, l_1, l_2) = \langle ||X \star \Psi_{j_1,l_1}| \star \Psi_{j_2,l_2}| \star \Phi \rangle$$

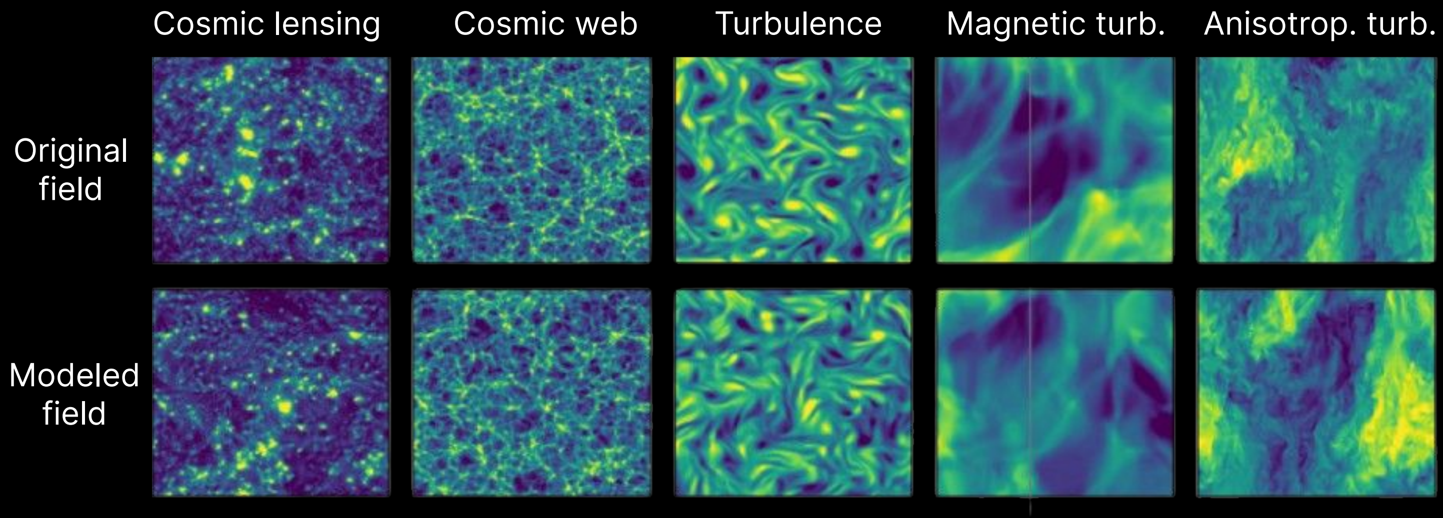
n-th order $$S_n(j_1, j_2, ..., l_n, l_n) = \langle ||X \star \Psi_{j_1,l_1}| \star ...| \star \Psi_{j_n,l_n}| \star \Phi \rangle$$

Scale (1-J)




Direction (1-L)

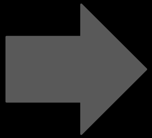
# The scattering transform reproduces field types



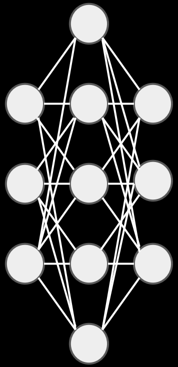

Cheng & Ménard, 2021, arXiv:2112.01288v1

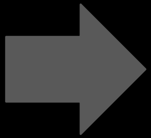
# I want to generate similar galaxies

Input



Method



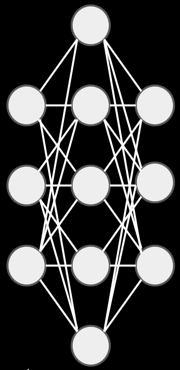


Output

# I want to generate similar galaxies

Input

Method



Output

Scattering transform

# Methods

Classifier

CNN

Scattering Transform
MLP

Generators

VAE

Normalising flows

Loss-based

# Methods

1.

Classifier

CNN

Scattering Transform

MLP

Generators

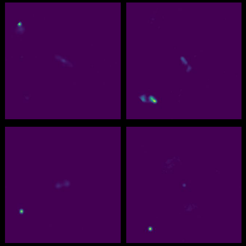
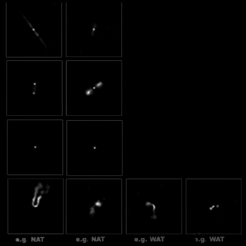
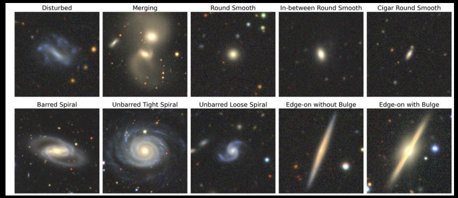
3. VAE

Next time

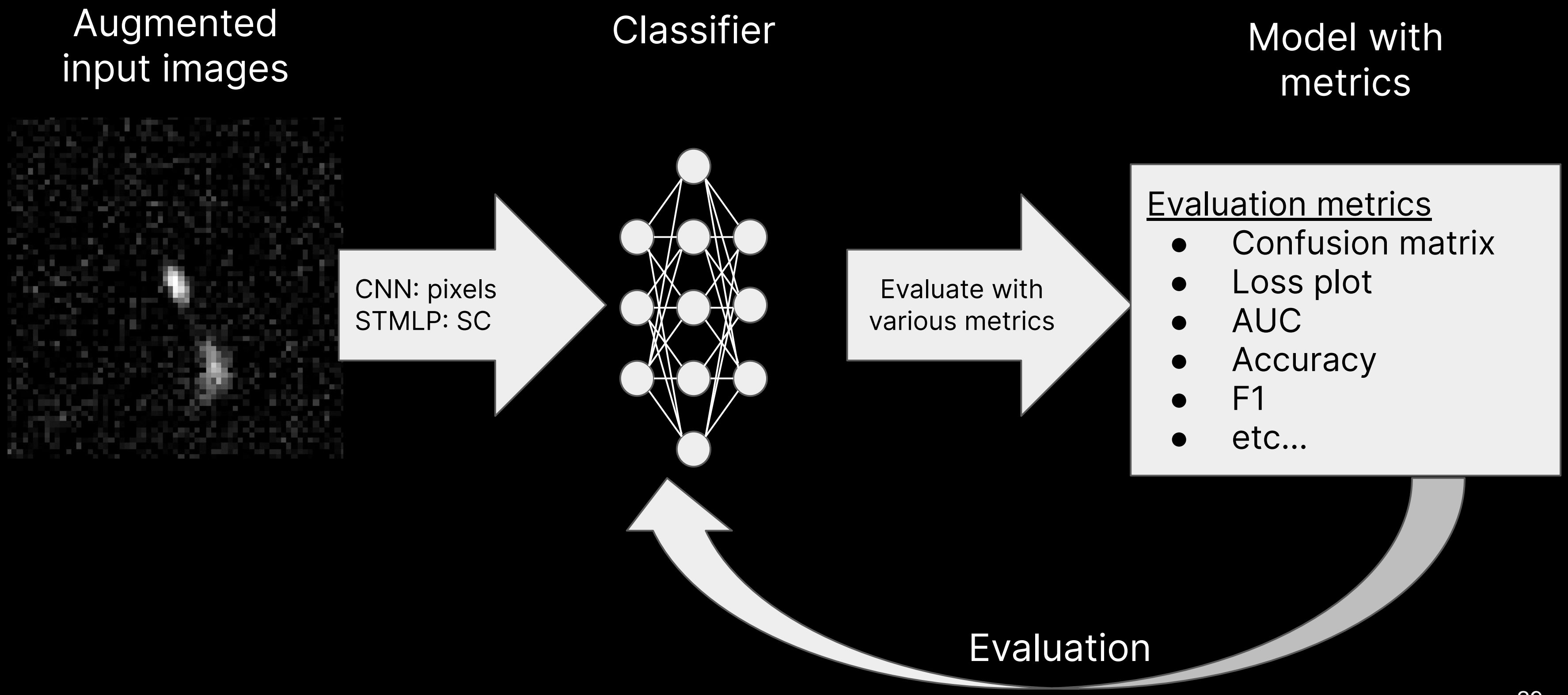
Normalising flows

2. Loss-based

# Four datasets have been used

| Input data | MNIST | Mirabest | FIRST | Galaxy10 |
|---|---|---|---|---|
| **Image size** | 28x28 | 150x150 | 300x300 | 3x256x256 |
| **Number of samples per class** | 6000 | ~500 | ~500 | ~1000 |
| **Example images** |  |  |  |  |
| **# of classes** | 10 | 2 | 4 | 10 |
| **Source** | Deng 2012, doi.org/10.1109/MSP.2012.2211477 | Porter & Scaife 2023, arXiv:2305.11108 v1 | Griese et al. 2023, doi.org/10.1016/j.dib.2023.108974 | Lintott et al. 2011, arXiv:1007.3265 |

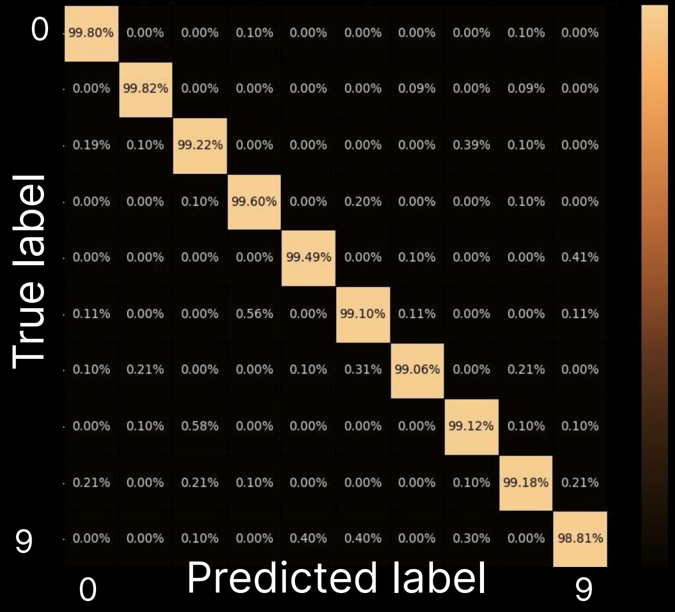
# Classifier - Method



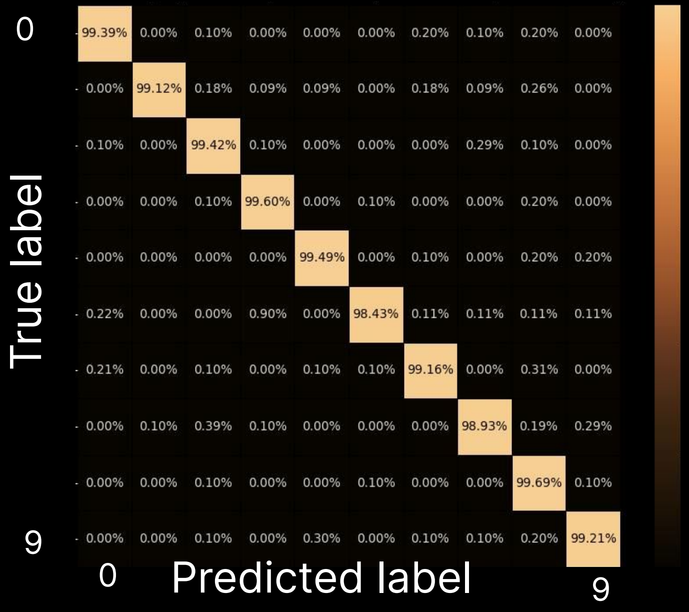
# Classifier - Results

| | |
|---|---|
| Dataset | MNIST |
| Training set per class | 6000 |
| Epochs | 200 |

CNN: 99.33%



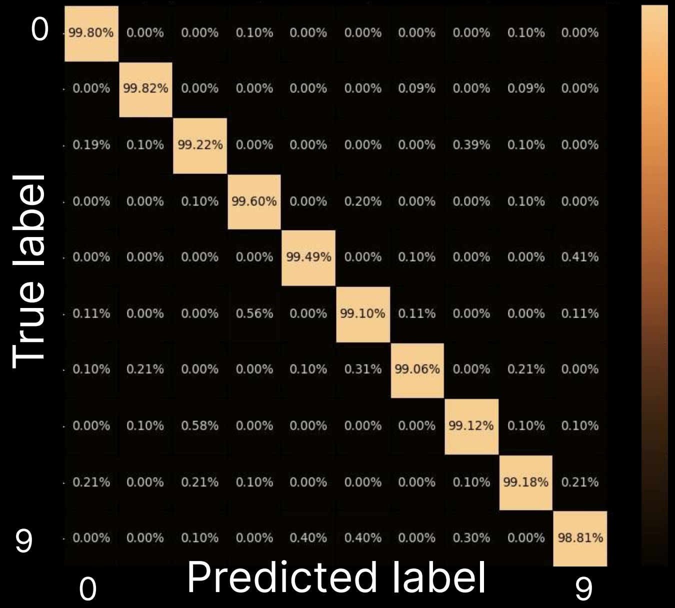

STMLP: 99:25%



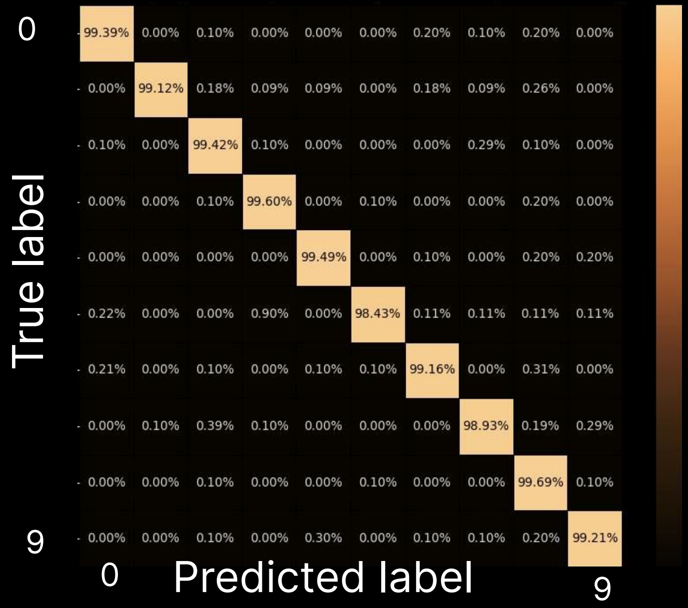
# Classifier - Results

| | |
|---|---|
| Dataset | MNIST |
| Training set per class | 6000 |
| Epochs | 200 |

CNN: 99.33%




STMLP: 99:25%



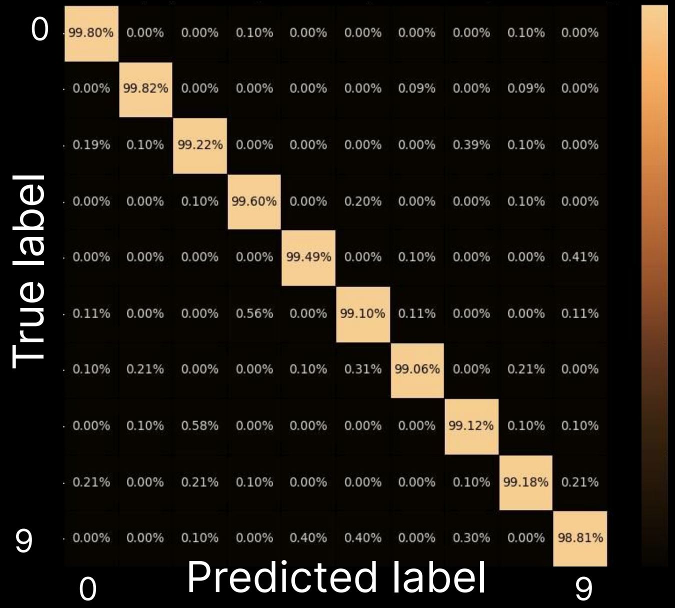

No hidden layers!

# Classifier - Results

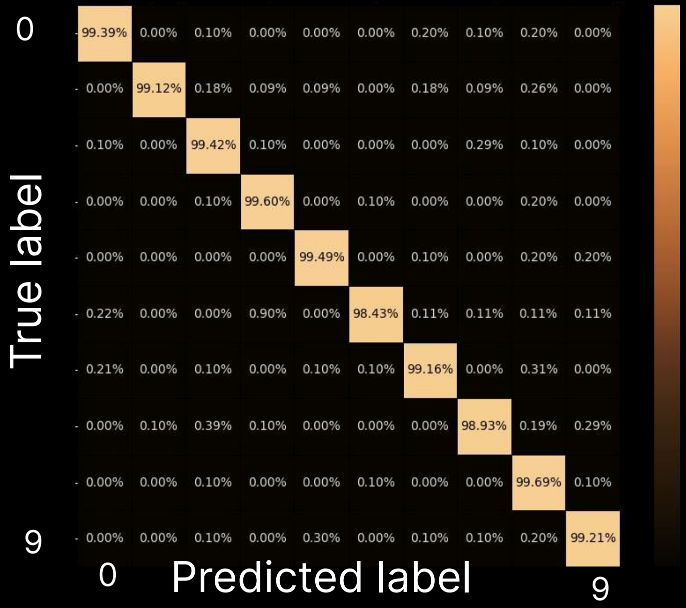
| | |
|---|---|
| Dataset | MNIST |
| Training set per class | 6000 |
| Epochs | 200 |

Smaller, faster, more stable and interpretable, better at small datasets

CNN: 99.33%




STMLP: 99:25%

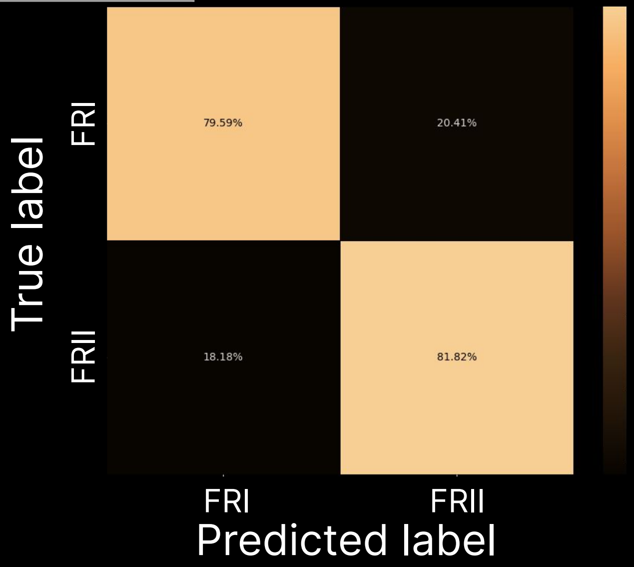



No hidden layers!

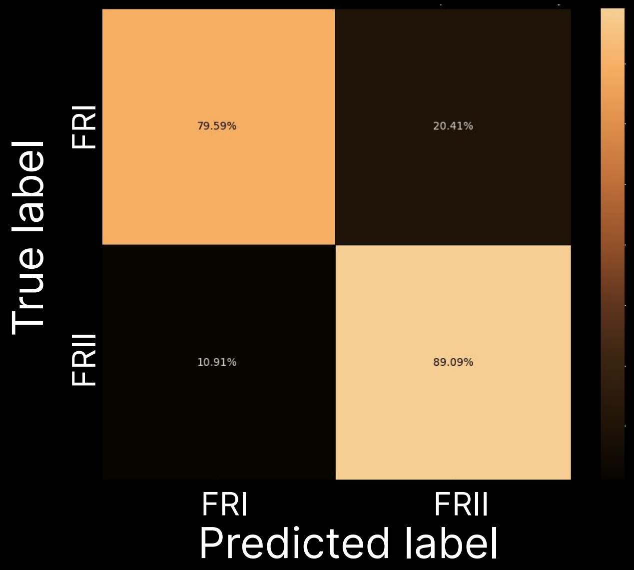
# Classifier - Results

| | |
|---|---|
| Dataset | Mirabest (cropped) |
| Training set per class | 2000 |
| Epochs | 200 |

CNN: 81%




STMLP: 85%

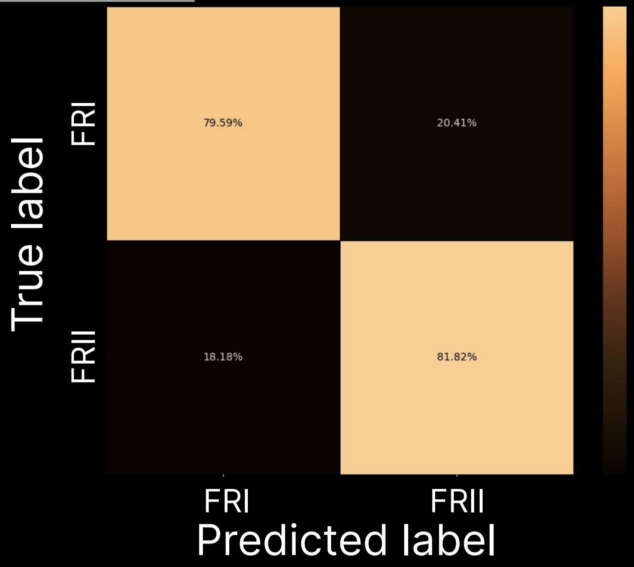


# Classifier - Results

| | |
|---|---|
| Dataset | Mirabest (cropped) |
| Training set per class | 2000 |
| Epochs | 200 |

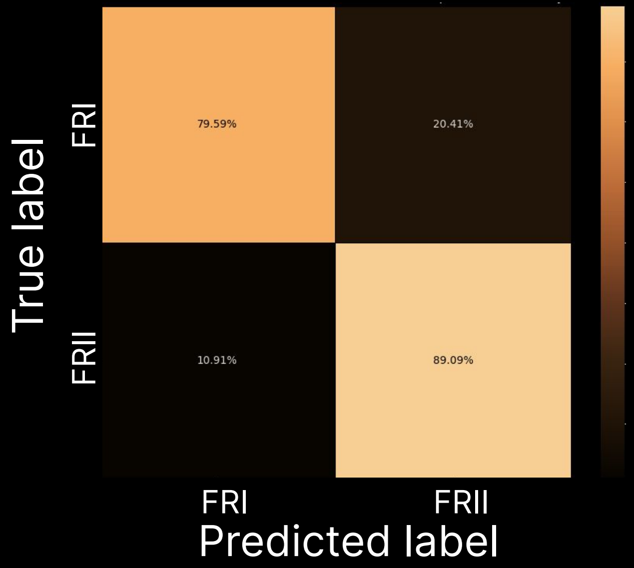
Training time: ~40 min

CNN: 81%

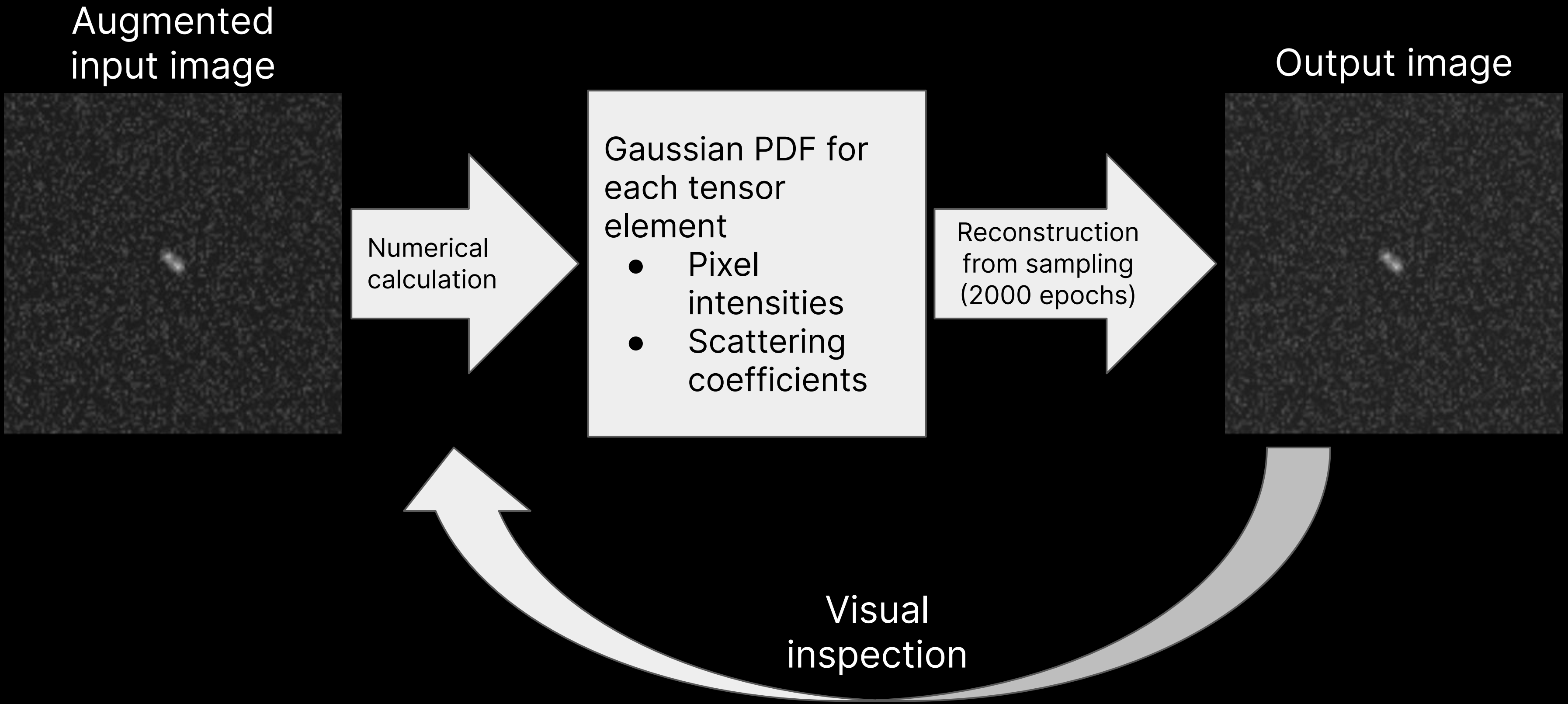



Training time: ~2 min

STMLP: 85%



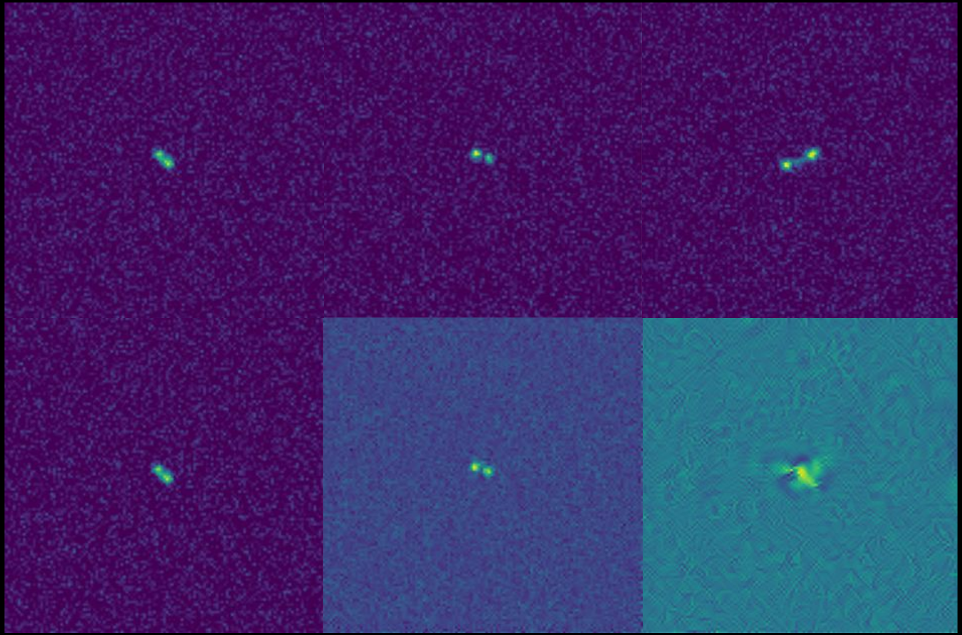
# 2. Generator: Loss-based - Method



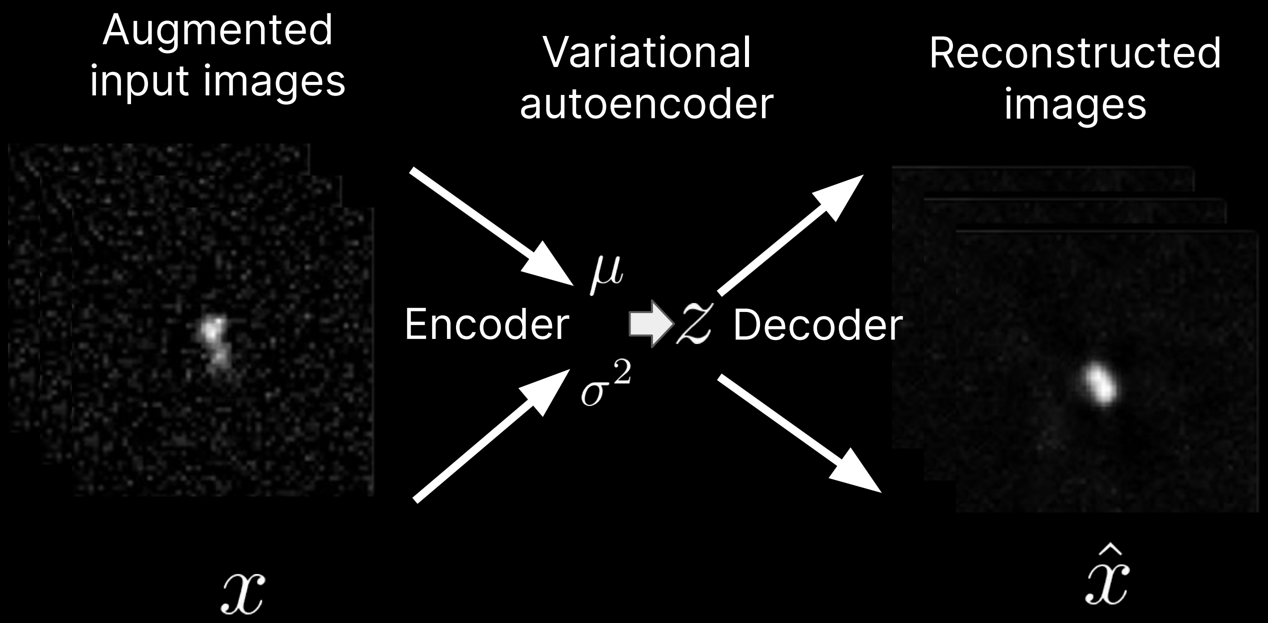
# 2. Generator: Loss-based - Results

Reconstruction of one FRII galaxy

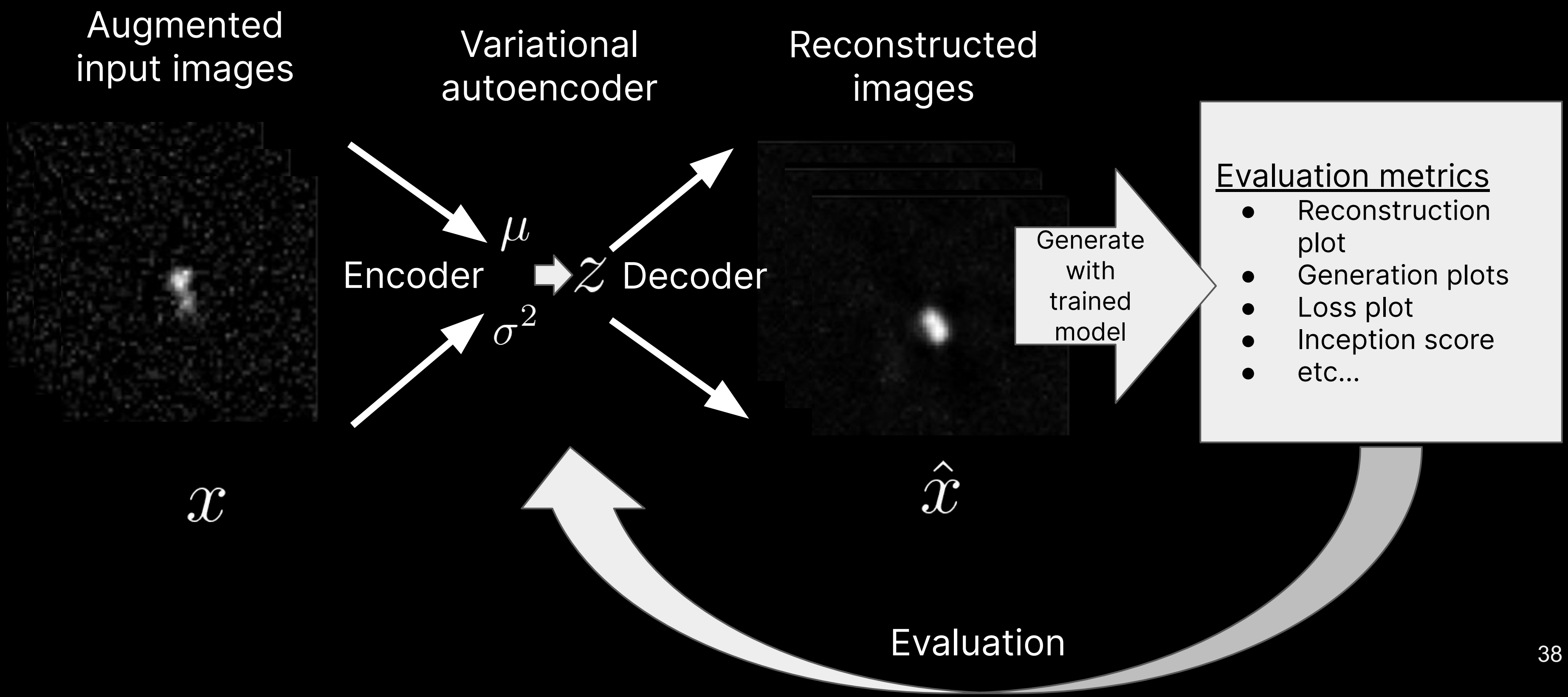
| | Pixels | Scattering coefficients | L-averaged scattering coefficients |
|---|---|---|---|
| Original | | | |
| Reconstruction | | | |



# 3. Method for variational autoencoder

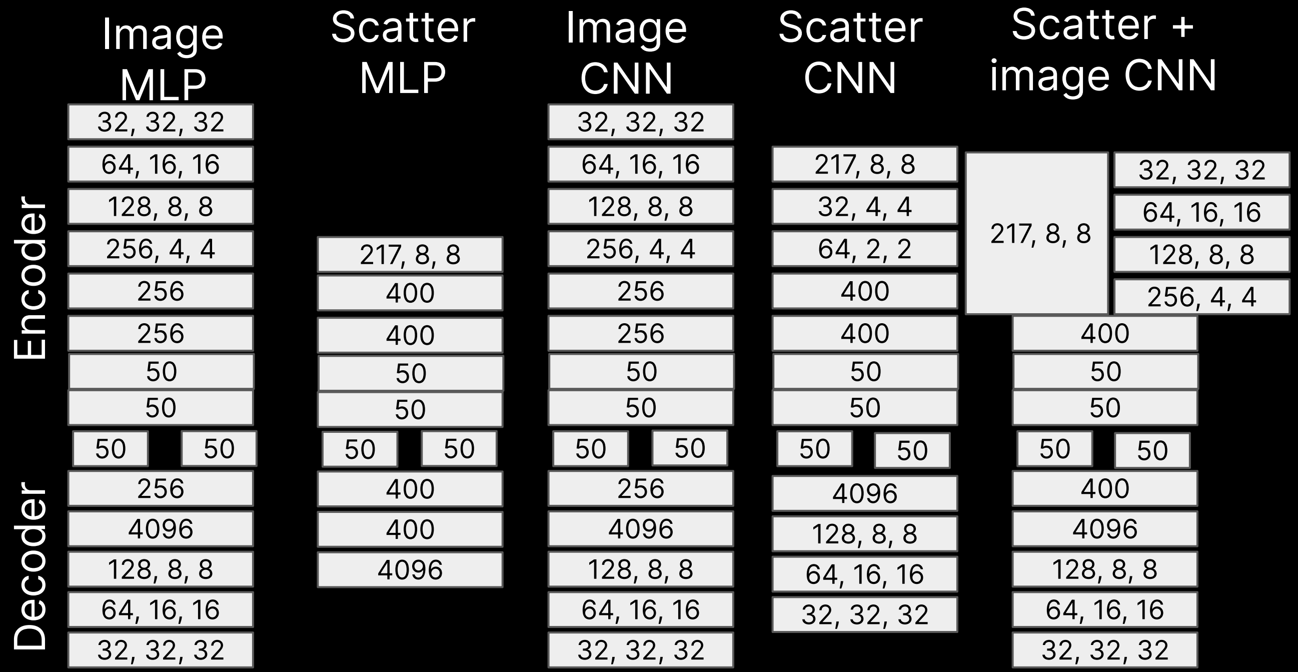


# 3. Method for variational autoencoder



# 3. I used five different variational autoencoders

Architecture (with batch normalisation and dropout)




| | Image MLP | Scatter MLP | Image CNN | Scatter CNN | Scatter + image CNN |
|---|---|---|---|---|---|
| Parameters | 3.6 M | 7.6 M | 3.6 M | 1.3 M | 9.5 M |
| Size | 15.7 MB | 29.1 MB | 15.7 MB | 6.7 MB | 41.4 MB |

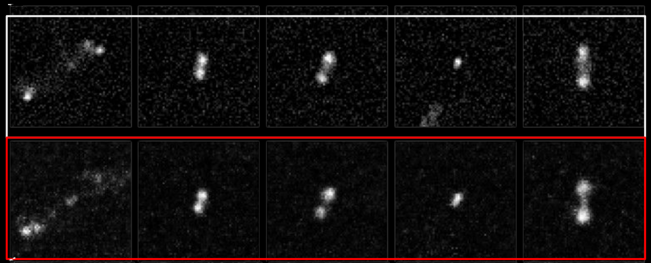
# 3. Generator: Variational Autoencoder - Results

Reconstruction of FRII Galaxies

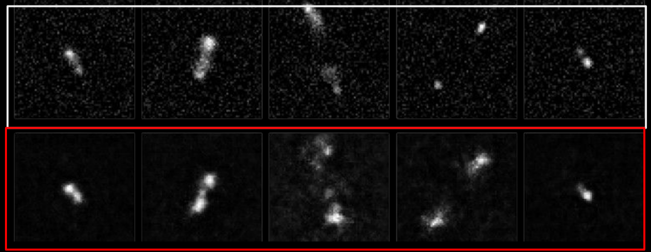
Top row: Original

Bottom row: Reconstructed

Image CNN



Scatter CNN



Scatter + image CNN

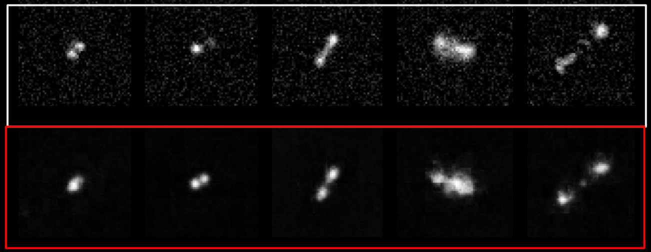
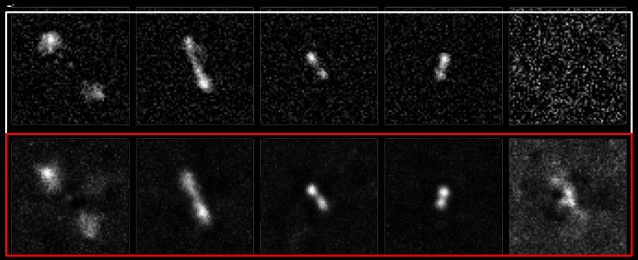


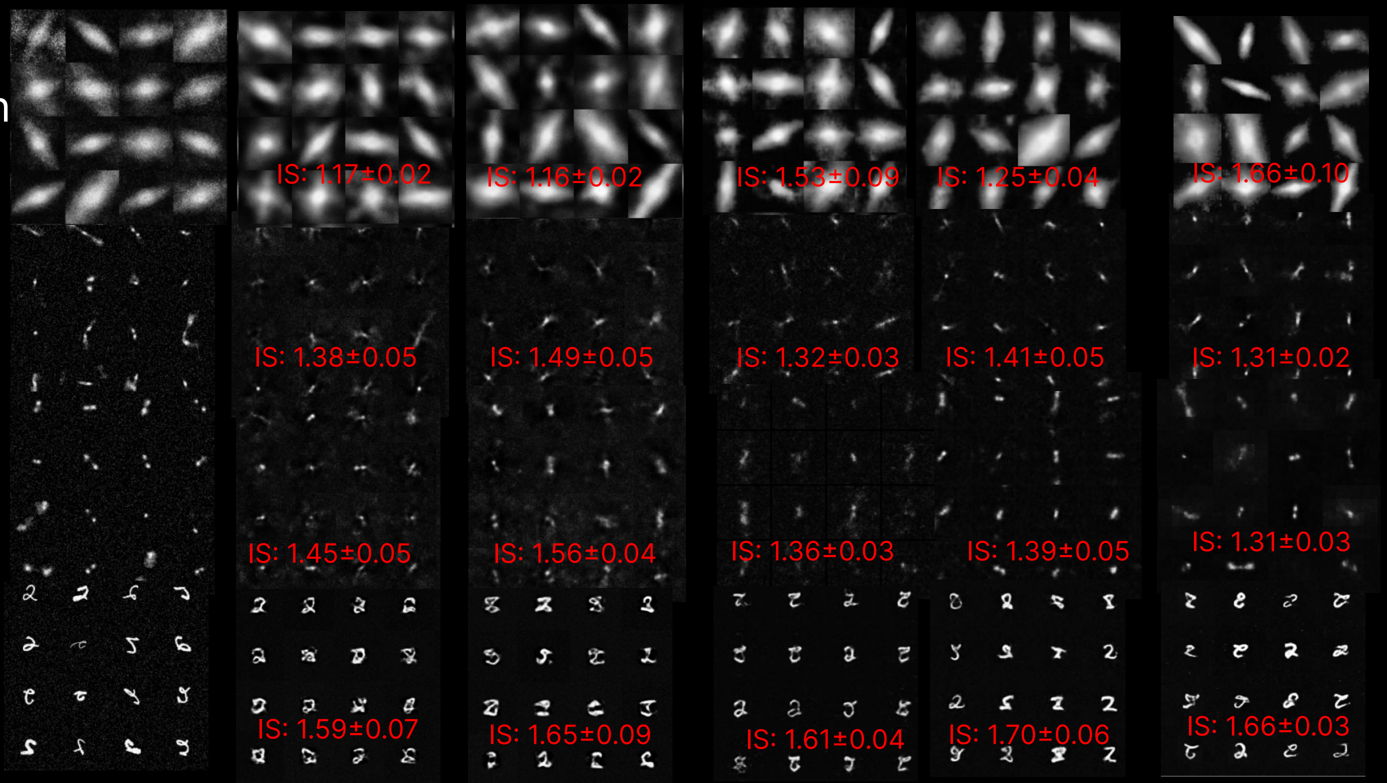
Image MLP



Scatter MLP

# 3. Generator: Variational Autoencoder - Results

Generation from noise



| | Original | Image MLP | Scatter MLP | Image CNN | Scatter CNN | Scatter + image CNN |
|---|---|---|---|---|---|---|
| Edge on galaxies with a bulge | | IS: 1.17±0.02 | IS: 1.16±0.02 | IS: 1.53±0.09 | IS: 1.25±0.04 | IS: 1.66±0.10 |
| FRI | | IS: 1.38±0.05 | IS: 1.49±0.05 | IS: 1.32±0.03 | IS: 1.41±0.05 | IS: 1.31±0.02 |
| FRII | | IS: 1.45±0.05 | IS: 1.56±0.04 | IS: 1.36±0.03 | IS: 1.39±0.05 | IS: 1.31±0.03 |
| Digit 2 | | IS: 1.59±0.07 | IS: 1.65±0.09 | IS: 1.61±0.04 | IS: 1.70±0.06 | IS: 1.66±0.03 |
| | | Tot IS: 5.59±0.19 | Tot IS: 5.86±0.20 | Tot IS: 5.82±0.19 | Tot IS: 5.75±0.20 | Tot IS: 5.94±0.18 |

# Methods

Classifier

CNN

Scattering Transform

MLP

Generators

VAE

Normalising flows

Loss-based

# Methods - a growing list

Classifier

CNN

Scattering Transform
MLP

Generators

VAE

Normalising flows

GAN

Loss-based

Diffusion models

# Future projects

**Classification**

- Dropout
- Regularisation

**Generation**

- GANs
- Diffusion models
- Normalising flows
- Feed in reduced scattering parameters

# Future projects

**<u>Classification</u>**

- Dropout
- Regularisation

**<u>Generation</u>**

- GANs
- Diffusion models
- Normalising flows
- Feed in reduced scattering parameters

Apply on

**<u>Diffuse cluster radio emission</u>**

# Thank you !

## Conclusions:

- The scattering transform is computationally efficient
- Performing on the same level as artificial neural networks, it has nevertheless not succeeded generating realistic sources
- More work is undergoing for classification and generation to be applied on diffuse radio emission

**Have a question?**

Markus Bredberg
markus.bredberg@epfl.ch