Deep learning for classifying and sorting diffraction images

Julian Zimmermann | Max-Born-Institute Berlin

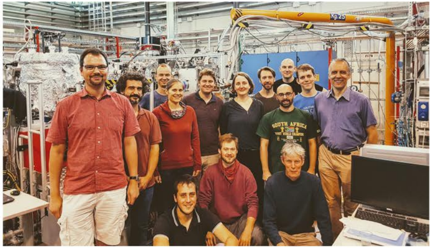
Acknowledgements

Max-Born-Institute:
D. Rupp
M. Sauppe
K. Kolatzki
B. Senfftleben
R.M.P. Tanyag
M.J.J. Vrakking
A. Rouzée

TU Berlin:
B. Langbehn
T. Möller

FERMI collaboration:
R. Cucini
M. Di Fraia
P. Finetti
A. C. LaForge
T. Nishiyama
Y. Ovcharenko
P. Piseri
O. Plekan
K. C.Prince
F. Stienkemeier
K.i Ueda
C. Callegari

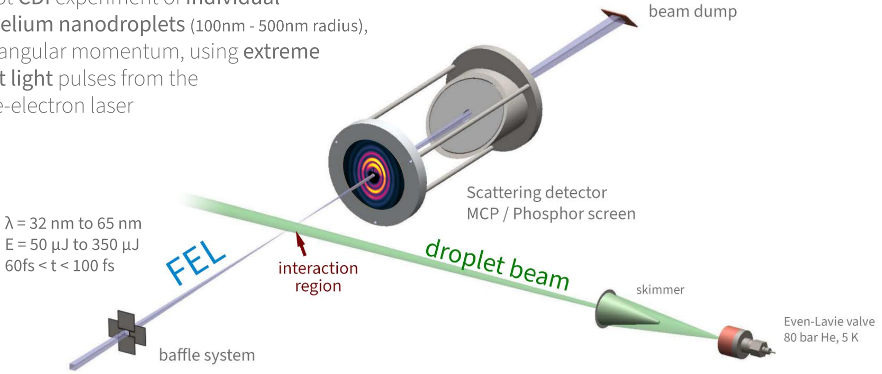
University Rostock:
T. Fennel
C. Peltz
B. Kruse
K. Sander



Experimental setup and recorded data

Experimental setup

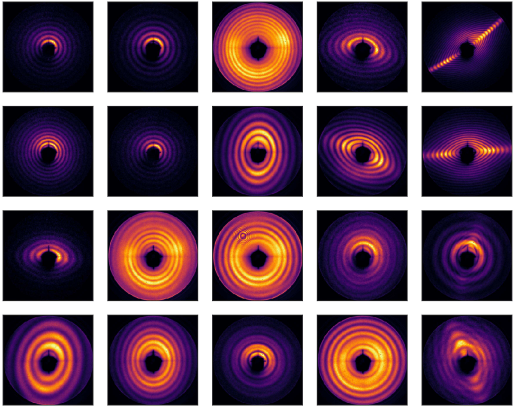
- Single-shot **CDI** experiment of **individual rotating helium nanodroplets** (100nm - 500nm radius), with high angular momentum, using **extreme ultraviolet light** pulses from the FERMI free-electron laser



Langbehn, B et al. Phys. Rev. Lett. 121, 255301 (2018)

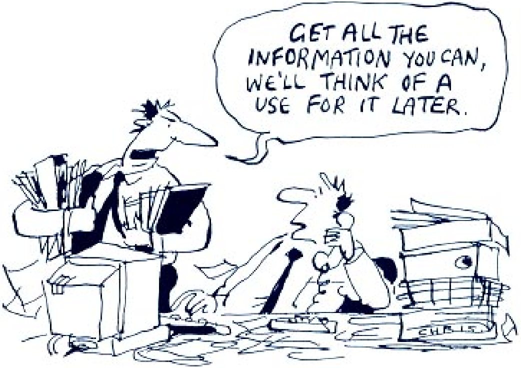
The Problem (2/2)

- From these diffraction images we can **infer the topology of particles in free flight**
- Using pump-probe schemes, we can also **record dynamic processes**



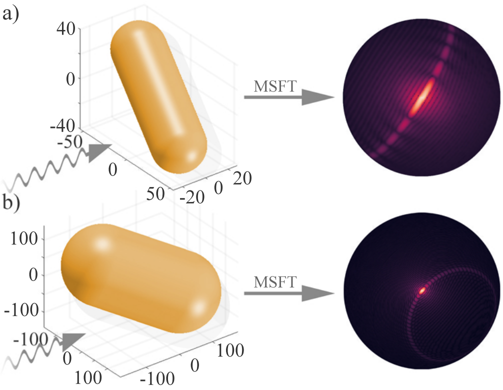
The Problem (1/2)

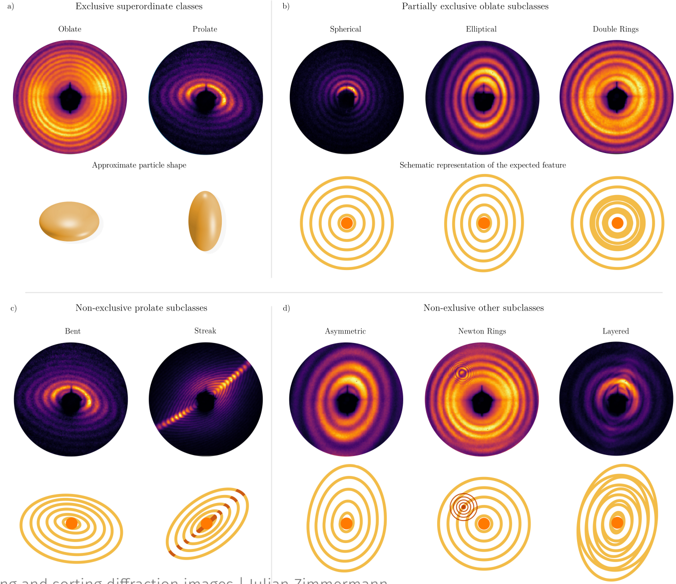
- The high repetition rates at FELs produce large data sets with up to **several million** diffraction pattern



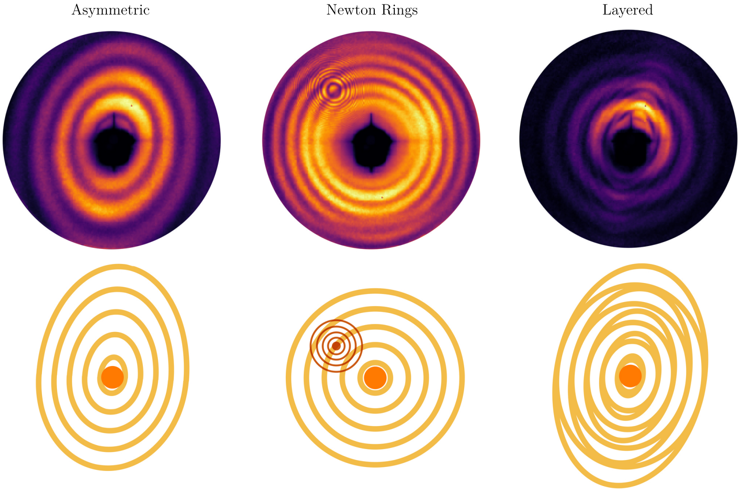
The Problem (2/2)

- Almost **same particle** but **different patterns**
- Handcrafted algorithms can't easily identify that this feature belongs to the same particle




a)
Exclusive superordinate classes
Oblate
Prolate
Approximate particle shape
b)
Partially exclusive oblate subclasses
Spherical
Elliptical
Double Rings
Schematic representation of the expected feature
c)
Non-exclusive prolate subclasses
Bent
Streak
d)
Non-exlusive other subclasses
Asymmetric
Newton Rings
Layered

Non-exlusive other subclasses

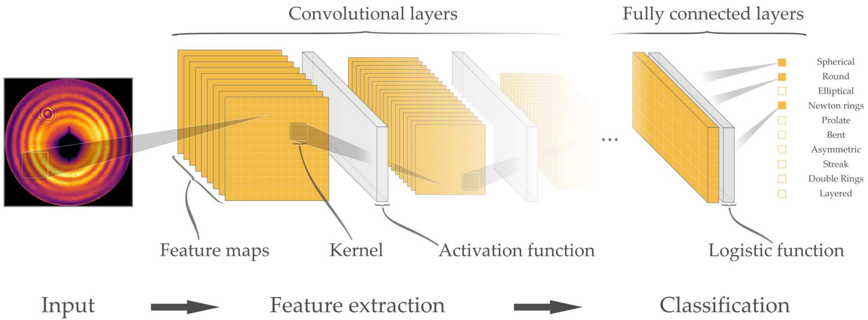


Zimmermann, J. et al. Phys. Rev. E 99, 063309 (2019)

Our first approach:
Supervised learning

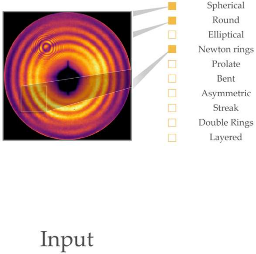
Idea

A directed acyclic graph consisting of hierarchically structured non-linear functions



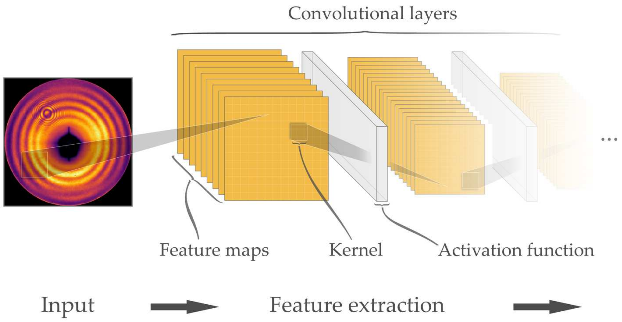
Idea

1. Have a researcher **classify a subset** of the data



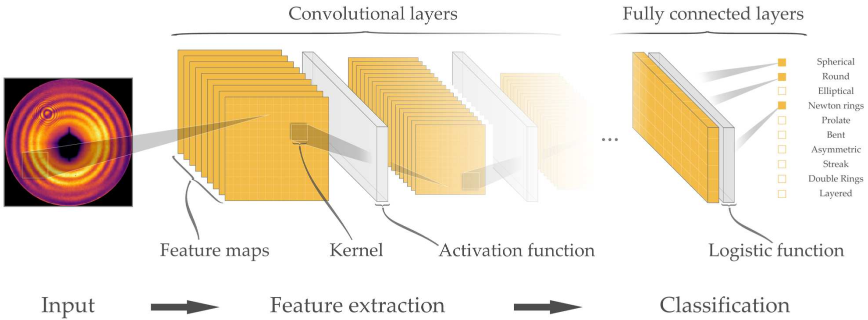
Idea

1. Have a researcher **classify a subset** of the data
2. Use this dataset as **training data** for a convolutional neural network



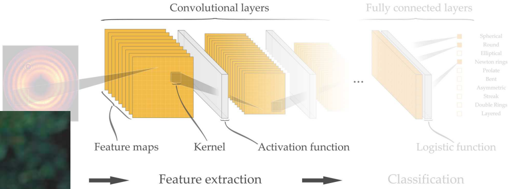
Idea

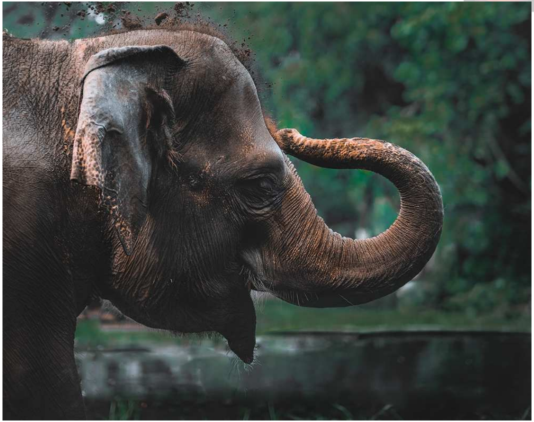
1. Have a researcher **classify a subset** of the data
2. Use this dataset as **training data** for a convolutional neural network
3. Use the trained network to **classify the rest** of the dataset



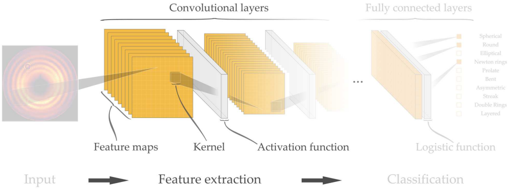
What is inside a neural network?

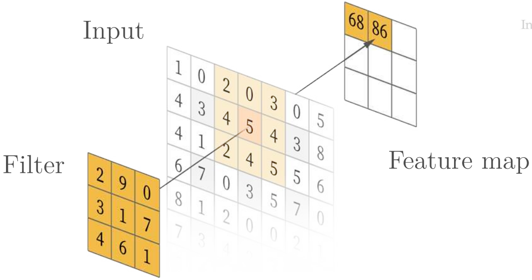
The convolution





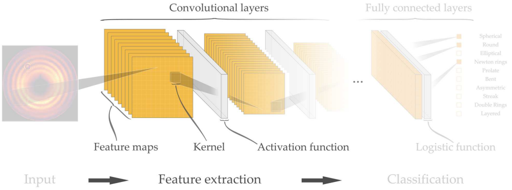
The convolution

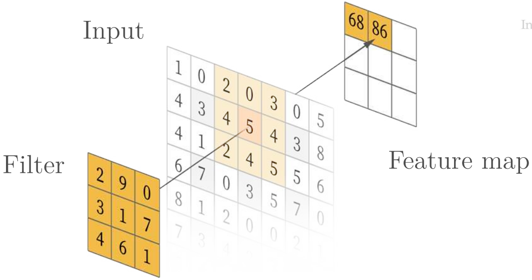




$$4 + 0 + 0 + 12 + 5 + 28 + 8 + 24 + 5 = 86$$

The convolution

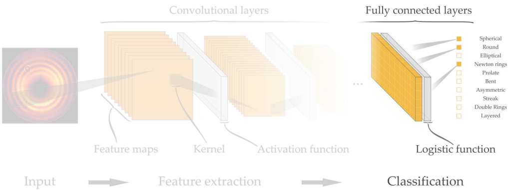




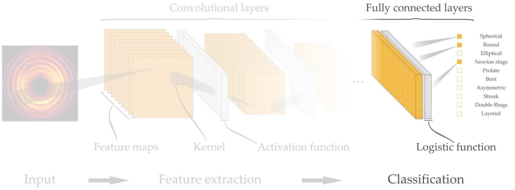
$$4 + 0 + 0 + 12 + 5 + 28 + 8 + 24 + 5 = 86$$

- **Idea**: Nearby pixel are correlated
- Filter is called **receptive field**

The loss function



The loss function



1. Cross entropy

$$E(p,q) = -\sum_i p_i(\mathrm{x}) \log\left(q_i(\mathrm{x})\right)$$

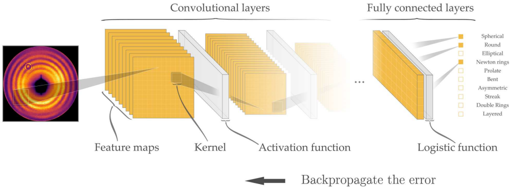
$$\vdots$$

$$= \mathrm{x} - \mathrm{x} \cdot p_i(\mathrm{x}) + \log\left(1 + \exp^{-\mathrm{x}}\right)$$

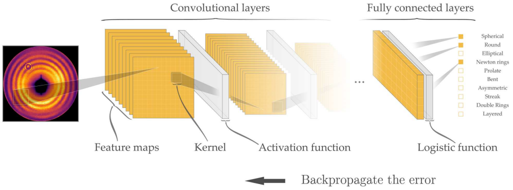
$p_i(\mathrm{x})$:= ground truth,
$q_i(\mathrm{x})$:= sigmoid function,
x := Output of the network

Vector of weights $\to (\hat{\mathrm{x}} \cdot w_{kl})$

Backpropagation



Backpropagation



2. Apply the chain rule

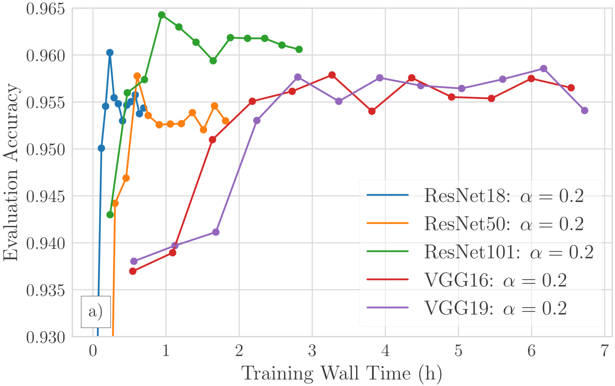
$$\frac{\partial E}{\partial w_{kl}} = \sum_{l} \frac{\partial E}{\partial x_l} \frac{\partial x_l}{\partial w_{kl}}$$

3. Use a gradient descent algorithm to solve for w_{kl}

Our results

Different architectures

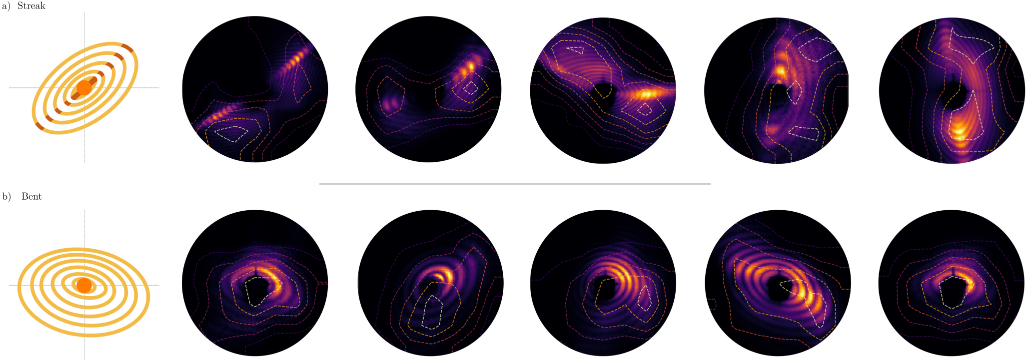
- Even the lowest performing neural network can **outperform previous classification approaches** by a large margin



Zimmermann, J. et al. Phys. Rev. E 99, 063309 (2019)

Interpretation

- **Streak**: The network was able to identify the dominant streak feature regardless of its orientation or size
- **Bent**: Strong resemblance is visible

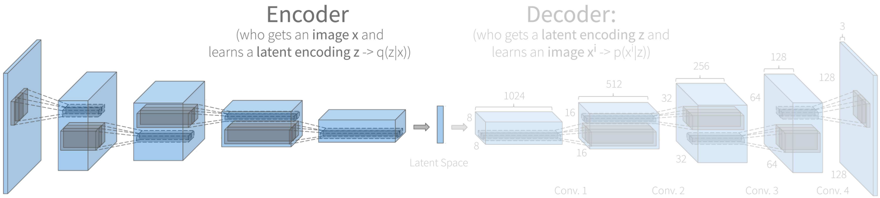


Zimmermann, J. et al. Phys. Rev. E 99, 063309 (2019)
GradCam++: Chattopadhyay, A. et al. IEEE 10.1109/WACV.2018.00097 (2018)

Our second approach: Unsupervised learning

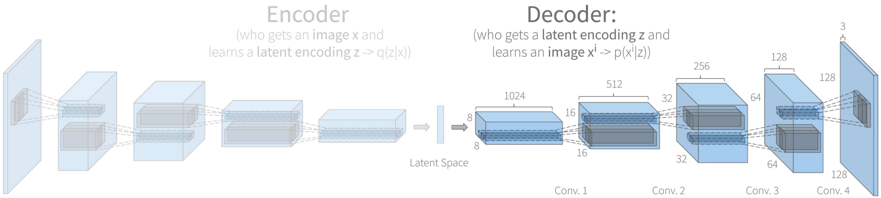
Idea

1. Give a neural network an image and **force the network to reduce the dimensionality**



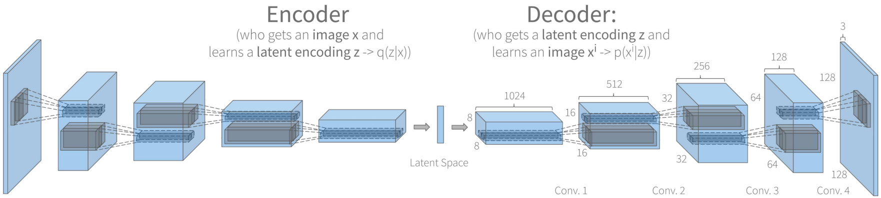
Idea

1. Give a neural network an image and **force the network to reduce the dimensionality**
2. **Reproduce the original image** from this reduced representation



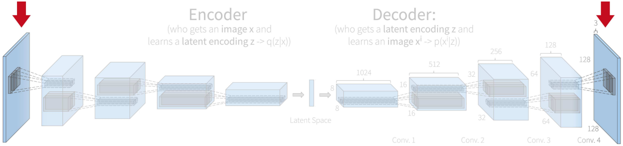
Idea

1. Give a neural network an image and **force the network to reduce the dimensionality**
2. **Reproduce the original image** from this reduced representation
3. This is called **Autoencoder** (We use a β-**TCVAE**)*



*Chen, T. Q. et al. arXiv:1802.04942 (2018)

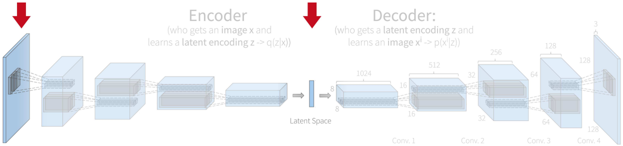
Loss function



1. Minimize the error between the original image and the generated one

$$\mathbb{L} = \underbrace{\mathbb{E}_{q(z|x^i)}\left[\log p(x^i|z)\right]}_{\text{Reconstruction Loss}} - \underbrace{D_{\mathrm{KL}}(q(z|x^i) \,\|\, p(z))}_{\text{Latent space penalty}} - \gamma \underbrace{D_{\mathrm{KL}}(q(z) \,\|\, \overline{q}(z))}_{\text{Total correlation penalty}} \,,$$

Loss function



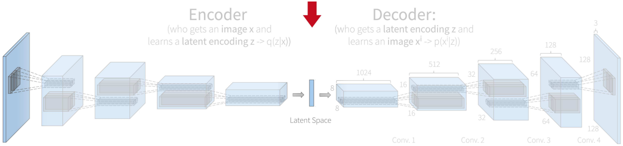
2. Model the latent space as a statistical distribution

$$\mathbb{L} = \underbrace{\mathbb{E}_{q(z|x^i)}\left[\log p(x^i|z)\right]}_{\text{Reconstruction Loss}} - \underbrace{D_{\mathrm{KL}}(q(z|x^i) \parallel p(z))}_{\text{Latent space penalty}} - \gamma \underbrace{D_{\mathrm{KL}}(q(z) \parallel \overline{q}(z))}_{\text{Total correlation penalty}},$$

with D_{KL}, being the Kullback-Leibler divergence:

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right)$$

Loss function



3. Make the latent encodings independent from one another

$$\mathbb{L} = \overbrace{\mathbb{E}_{q(z|x^i)}\left[\log p(x^i|z)\right]}^{\text{Reconstruction Loss}} - \overbrace{D_{\mathrm{KL}}(q(z|x^i) \parallel p(z))}^{\text{Latent space penalty}} - \gamma \overbrace{D_{\mathrm{KL}}(q(z) \parallel \overline{q}(z))}^{\text{Total correlation penalty}} \,,$$

with $\overline{q}(z)$, being the factorial distribution over q(z):

$$\overline{q}(z) = \prod_{j=1}^{d} q(z_j)$$

Evaluate the latent space

1. We use the **same Helium nanodroplets dataset**

Evaluate the latent space

1. We use the **same Helium nanodroplets dataset**
2. We apply an **unsupervised clustering routine on the latent space**, called Gaussian Mixture Model (GMM)

Evaluate the latent space

1. We use the **same Helium nanodroplets dataset**
2. We apply an **unsupervised clustering routine on the latent space,** called Gaussian Mixture Model (GMM)
3. We **compare** our results **with the current state-of-the-art** approach; Spectral clustering on **raw images***

***Yoon, H. C. et al.** Opt. Express 19, 16542-16549 (2011)

Evaluate the latent space

1. We use the **same Helium nanodroplets dataset**
2. We apply an **unsupervised clustering routine on the latent space**, called Gaussian Mixture Model (GMM)
3. We **compare** our results **with the current state-of-the-art** approach; Spectral clustering on **raw images***
   3.1 We examine how well both routines align with the pre-defined labels of our supervised approach.

***Yoon, H. C. et al.** Opt. Express 19, 16542-16549 (2011)

The predicted classes for both routines

We use three metrics for assessment (All normalized between 0 and 1):

We use three metrics for assessment (All normalized between 0 and 1):

1. Homogeneity:
   - A clustering result satisfies homogeneity if all of its clusters contain only data points which are members of a single class

We use three metrics for assessment (All normalized between 0 and 1):

1. Homogeneity:
   - A clustering result satisfies homogeneity if all of its clusters contain only data points which are members of a single class
2. Completeness:
   - A clustering result satisfies completeness if all the data points that are members of a given class are elements of the same cluster

We use three metrics for assessment (All normalized between 0 and 1):

1. Homogeneity:
    - A clustering result satisfies homogeneity if all of its clusters contain only data points which are members of a single class
2. Completeness:
    - A clustering result satisfies completeness if all the data points that are members of a given class are elements of the same cluster
3. Adjusted Rand score:
    - The Rand Index computes a similarity measure between two clusterings by considering all pairs of samples and counting pairs that are assigned in the same or different clusters in the predicted and true clusterings

The predicted classes for both routines

	GMM on VAE encodings	Spectral clustering on raw images
Homogeneity	0.826	0.204
Completeness	0.872	0.221
Adjusted Rand score	0.700	0.321

- Significant improvements:
  - **4.0** times more homogeneous
  - **3.9** times more complete
  - **2.2** times more accurate according to Rand score

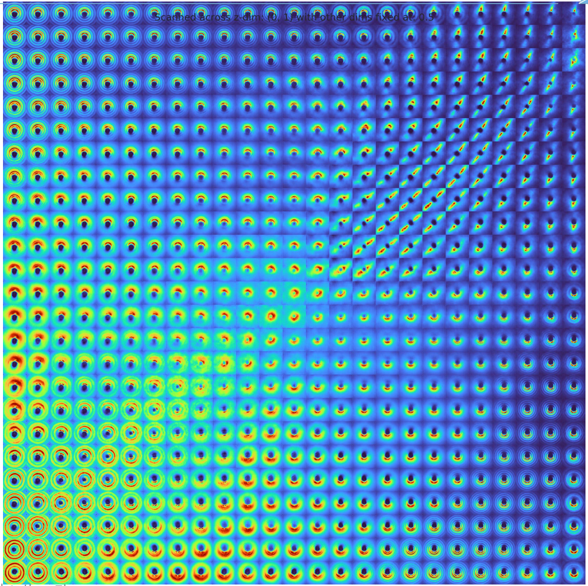
Gaussian Mixture Model clustering on VAE's latent space

Spectral Clustering on raw diffraction images

Outlook

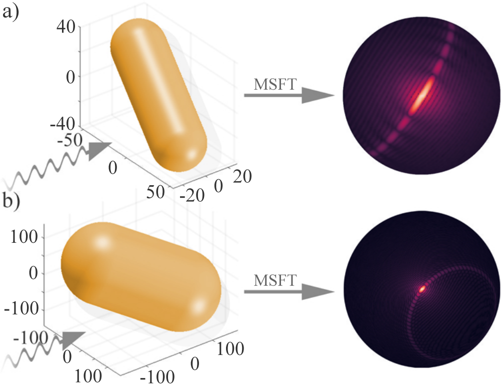
Outlook

- Try to discriminate characteristic features directly in latent space



Outlook

- Extend to simulated data. E.g., in combination with the Multi-Slice-Fourier-Transform (MSFT) method



Outlook

- Extend to online analysis during experiment

Summary

1. We successfully **adapted and published** a state-of-the-art **convolutional neural network** for the domain of diffraction images*
    - The network was able the learn the same features that a researcher identified

* **Zimmermann, J. et al.** Phys. Rev. E 99, 063309 (2019)
  Langbehn, B et al. Phys. Rev. Lett. 121, 255301 (2018)

Summary

1. We successfully **adapted and published** a state-of-the-art **convolutional neural network** for the domain of diffraction images*
    - The network was able the learn the same features that a researcher identified
2. Using unsupervised learning **we improved significantly on the current state-of-the-art**
    - We use a VAE to encode the information in a latent space that we cluster using a density based approach
    - This is still work in progress, but so far the prospect looks promising

* **Zimmermann, J. et al.** Phys. Rev. E 99, 063309 (2019)
Langbehn, B et al. Phys. Rev. Lett. 121, 255301 (2018)

Summary

1. We successfully **adapted and published** a state-of-the-art **convolutional neural network** for the domain of diffraction images*
   - The network was able the learn the same features that a researcher identified
2. Using unsupervised learning **we improved significantly on the current state-of-the-art**
   - We use a VAE to encode the information in a latent space that we cluster using a density based approach
   - This is still work in progress, but so far the prospect looks promising

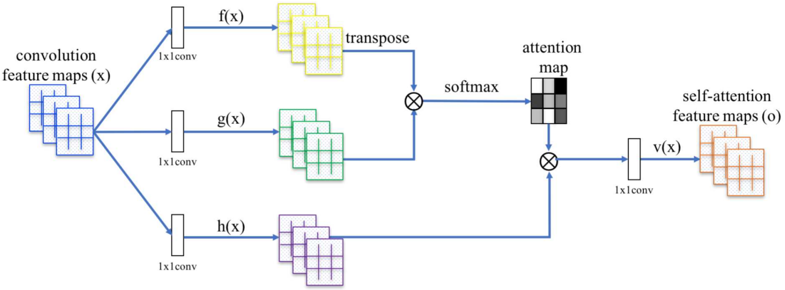
Source code is freely available on Github (written in Python > 3.6 using Tensorflow)

* **Zimmermann, J. et al.** Phys. Rev. E 99, 063309 (2019)
Langbehn, B et al. Phys. Rev. Lett. 121, 255301 (2018)

Thank you!

Self-attention

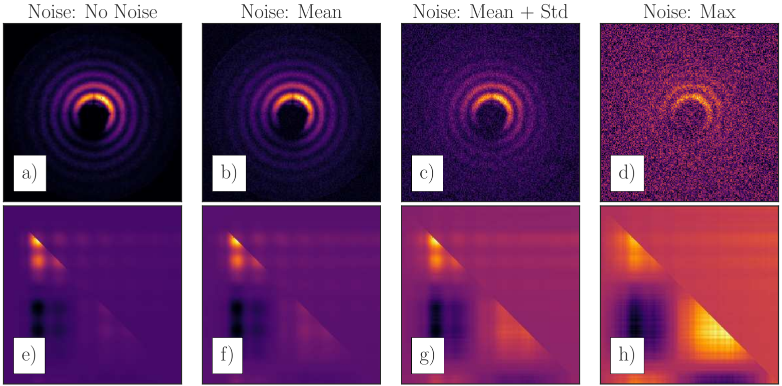
- Let the neural network decide where to look



Zhang, H. et al. arXiv:1805.08318 (2018)

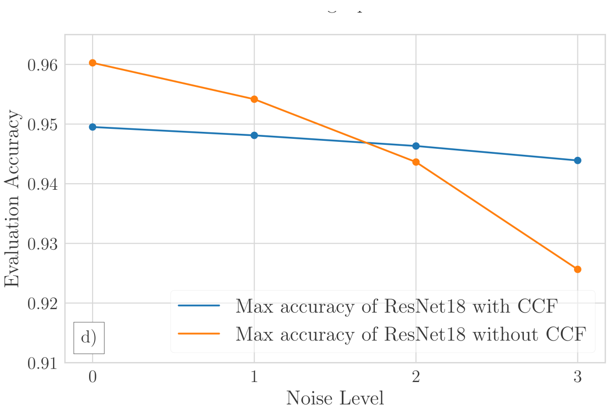
Using cross-correlation function (1/2)

- Higher orders of the two-point cross-correlation function can be used to **increase the signal-to-noise ratio** of a diffraction image

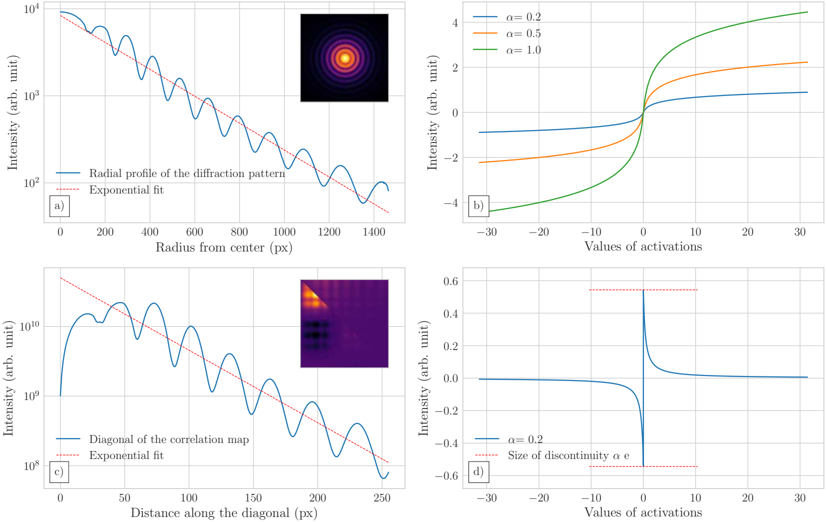


Using cross-correlation function (2/2)

- It is a **viable alternative** to use the two-point cross-correlation images as input to the neural network

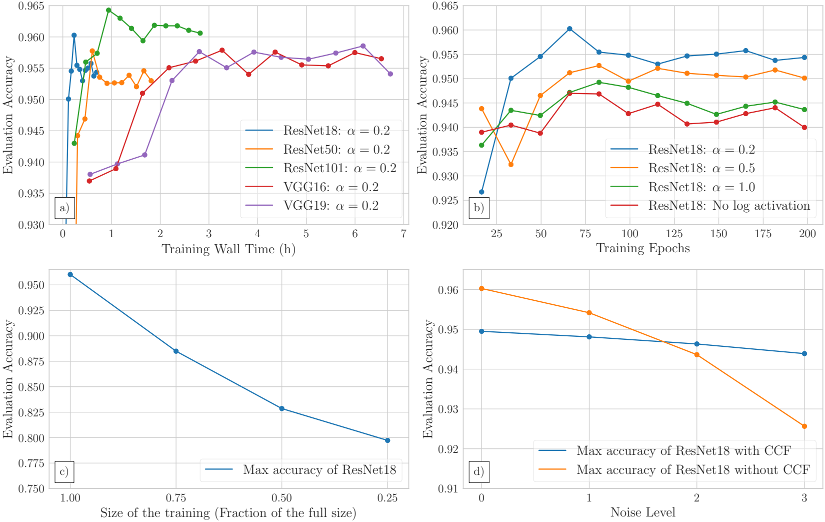


The activ



a)
Radial profile of the diffraction pattern
Exponential fit
Intensity (arb. unit)
Radius from center (px)
b)
α= 0.2
α= 0.5
α= 1.0
Intensity (arb. unit)
Values of activations
c)
Diagonal of the correlation map
Exponential fit
Intensity (arb. unit)
Distance along the diagonal (px)
d)
α= 0.2
Size of discontinuity α e
Intensity (arb. unit)
Values of activations

All accur


0.965
0.960
0.955
0.950
0.945
0.940
0.935
0.930
Evaluation Accuracy
0 1 2 3 4 5 6 7
Training Wall Time (h)
ResNet18: $\alpha = 0.2$
ResNet50: $\alpha = 0.2$
ResNet101: $\alpha = 0.2$
VGG16: $\alpha = 0.2$
VGG19: $\alpha = 0.2$
a)
0.965
0.960
0.955
0.950
0.945
0.940
0.935
0.930
0.925
0.920
Evaluation Accuracy
25 50 75 100 125 150 175 200
Training Epochs
ResNet18: $\alpha = 0.2$
ResNet18: $\alpha = 0.5$
ResNet18: $\alpha = 1.0$
ResNet18: No log activation
b)
0.950
0.925
0.900
0.875
0.850
0.825
0.800
0.775
0.750
Evaluation Accuracy
1.00 0.75 0.50 0.25
Size of the training (Fraction of the full size)
Max accuracy of ResNet18
c)
0.96
0.95
0.94
0.93
0.92
0.91
Evaluation Accuracy
0 1 2 3
Noise Level
Max accuracy of ResNet18 with CCF
Max accuracy of ResNet18 without CCF
d)

[1] **Langbehn, B et al.** Phys. Rev. Lett. 121, 255301 (2018)

[2] **Zimmermann, J. et al.** Phys. Rev. E 99, 063309 (2019)

[3] **Emma, et al.** doi: 10.1038/nphoton.2010.176

[4] **Bostedt, et al.** doi: 10.1103/RevModPhys.88.015007

[5] **Bobkov, S. A. et al.** J. Synchrotron Radiat. 22, 1345–1352 (2015).

[6] **Zhang, H. et al.** arXiv:1805.08318 (2018)

[7] **Kingma, D. P. et al.** Iclr 1–14 (2014) doi:10.1051/0004-6361/201527329

[8] **Chen, T. Q. et al.** arXiv:1802.04942 (2018)