Technische Universität Braunschweig

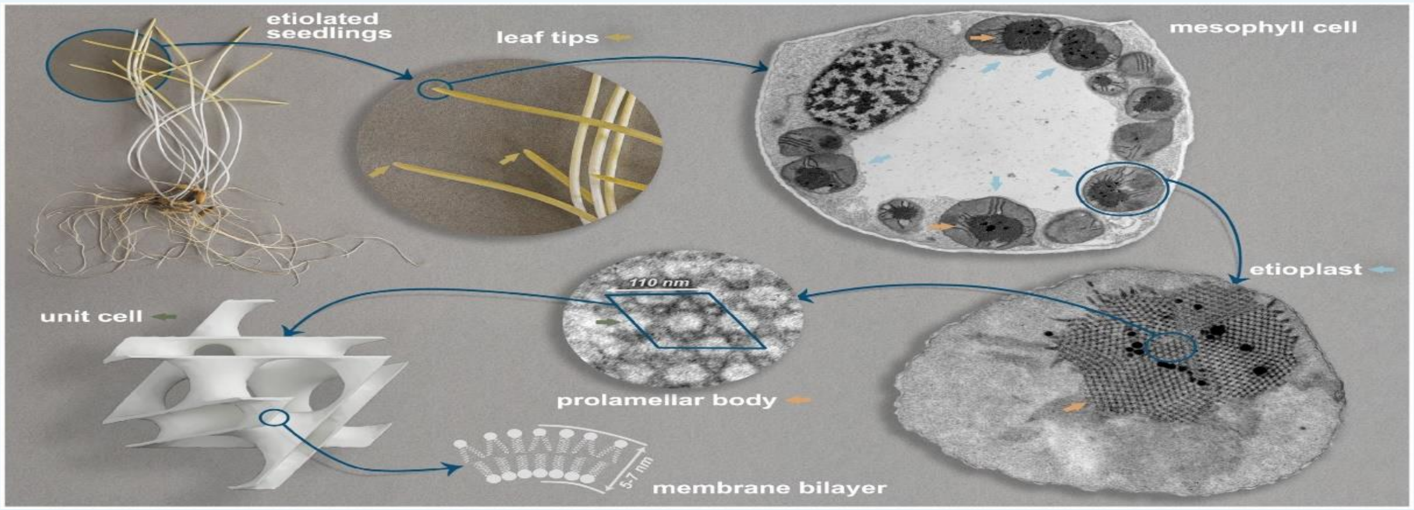
Institut für Analysis und Algebra

Multi-view classification of chloroplast cells

Bhupender Bindal, 13 September 2023

Introduction

- **Motivation:**
  - Better understanding of the nano-scale 3D structure of cubic membranes and its structure-functional relationship
  - Create nature-inspired materials: medicine, material science etc.
  - In this study, 3D structures in etioplasts, which are precursors of chloroplast cells, are studied
- How to study these nanoscale 3D structures?



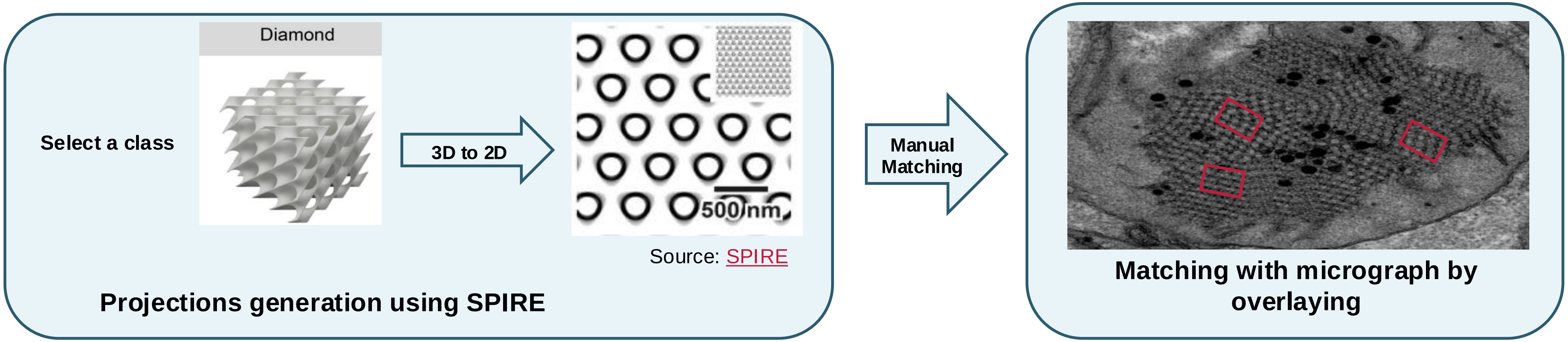
Plant specimen to micrographs

Source: SPIRE

Difficult to identify 3D structures from 2D projections

Introduction

- **Current methodology:** Generate micrographs controlling parameters of the geometry and manually match them with the specimen to identify the type of micrograph



Source: SPIRE

Projections generation using SPIRE

Matching with micrograph by overlaying

- **Problem:** For each micrograph, generate 100s or 1000s micrographs and do manual matching

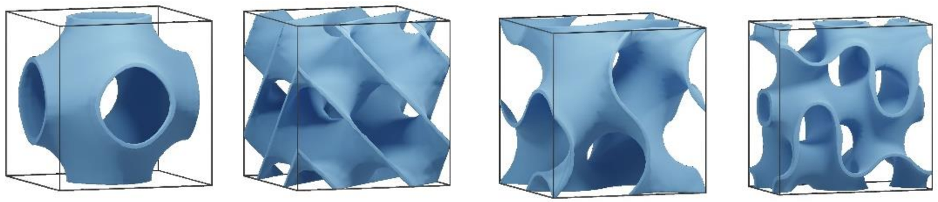
Introduction

- **Proposed approach:** Some function that maps a micrograph to the corresponding class of 3D structure

$$f : \boldsymbol{X}_V \rightarrow \underbrace{\{0, 1, 2, 3\}}_{\text{4 Classes:}}$$

Function
(Neural network)

Micrograph



Primitive Diamond Gyroid Lonsdaleite

- **Scope of this Studienarbeit:**
  a) Classify micrographs of chloroplast cells into 4 classes using a Multi-View Convolutional Neural Network (MVCNN)
  b) Compare the multi-view method with the baseline method of single-view Convolutional Neural Network (CNN)
  c) Assess the applicability of trained models to real micrographs.

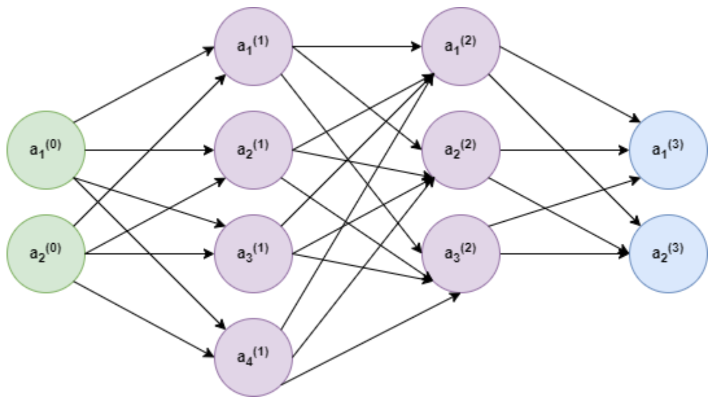
Introduction

The organisation of the presentation:

1. Convolutional Neural Networks
2. Multi-View Convolutional Networks
3. Experimental setup
4. Results
5. Discussion
6. Conclusion

1. Convolutional Neural Networks

- **Why neural networks are so popular? :** It can automatically identify data-specific information given sufficient data
- **Simplest type:** Feed Forward Neural Network (FFNN) ~ a composition of many individual non-linear functions
- Convolutional Neural Network (CNN) is a type of FFNN that processes multi-dimensional array data
- **How does it process multi-dimensional data?:** Convolution operation



$a^{(0)} = x \qquad a^{(1)} \qquad a^{(2)} \qquad a^{(3)} = f(x)$

FFNN

$$\boxed{f(x) = f^{(3)}(f^{(2)}(f^{(1)}(x; W^{(1)}; b^{(1)}); W^{(2)}; b^{(2)}); W^{(3)}; b^{(3)})}$$

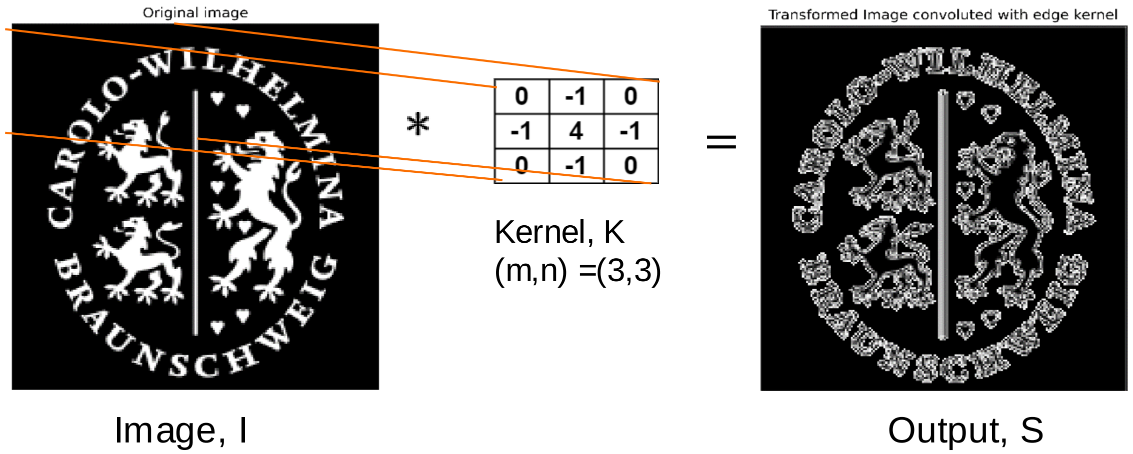
$$a^{(l)} = \underbrace{f^{(l)}}_{\text{Non-linear Activation}}(\underbrace{W^{(l)}a^{(l-1)} + b^{(l)}}_{\text{Affine linear transformation}}), \quad for \quad \underbrace{l}_{\text{Layer}} \in \{1, 2, 3\}$$

Where, $\boxed{a^{(0)} = x \quad \text{and} \quad a^{(3)} = f(x)}$

1. Convolutional Neural Networks

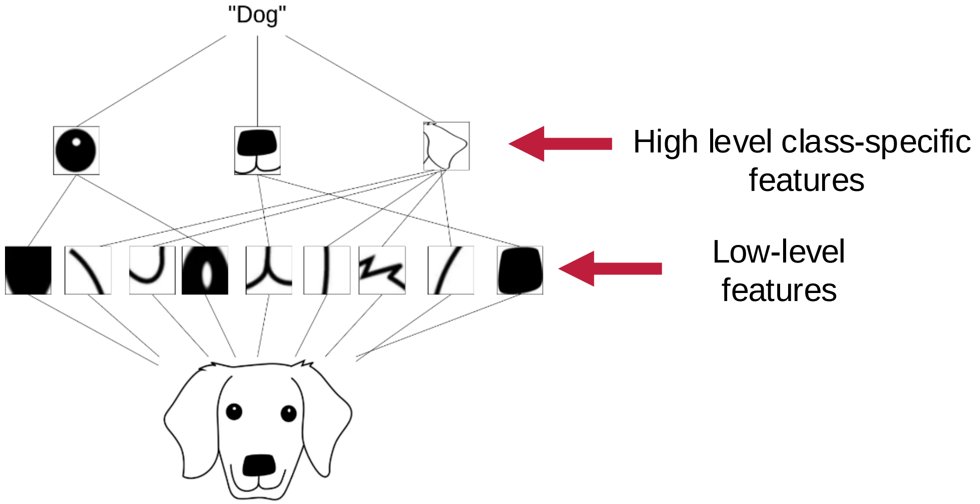
Convolution operation:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(i+m, j+n)K(m,n)$$



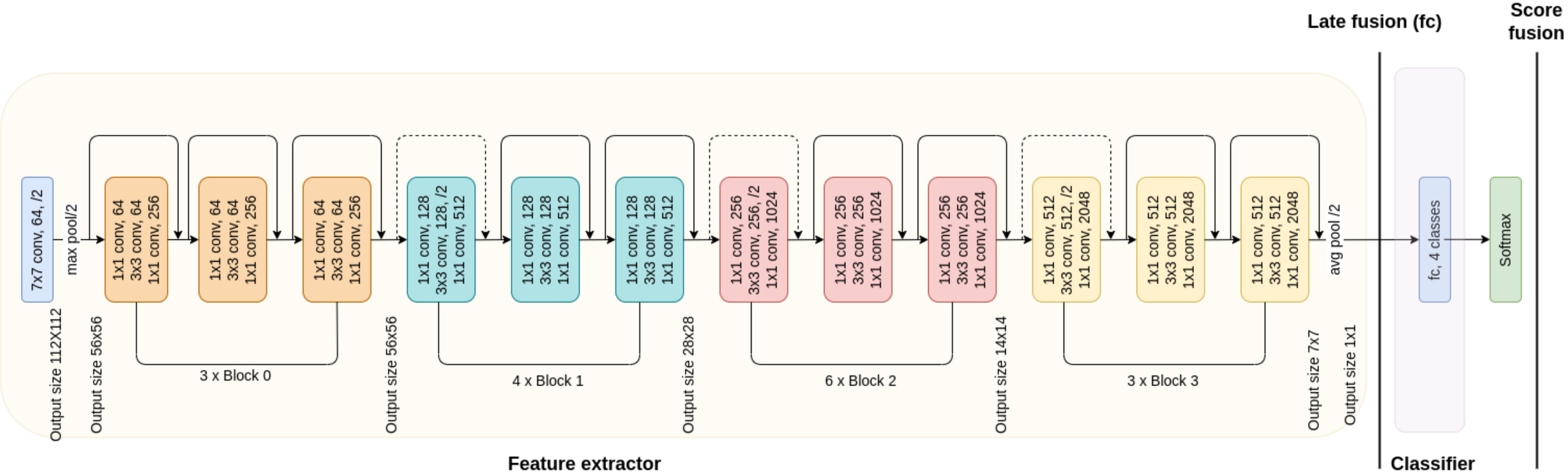
In a CNN:

- Each convolutional layer consists of many such kernels
- Entries of each kernel are learnable parameters of the network
- Extracted features are composed in subsequent layers



1. Convolutional Neural Networks

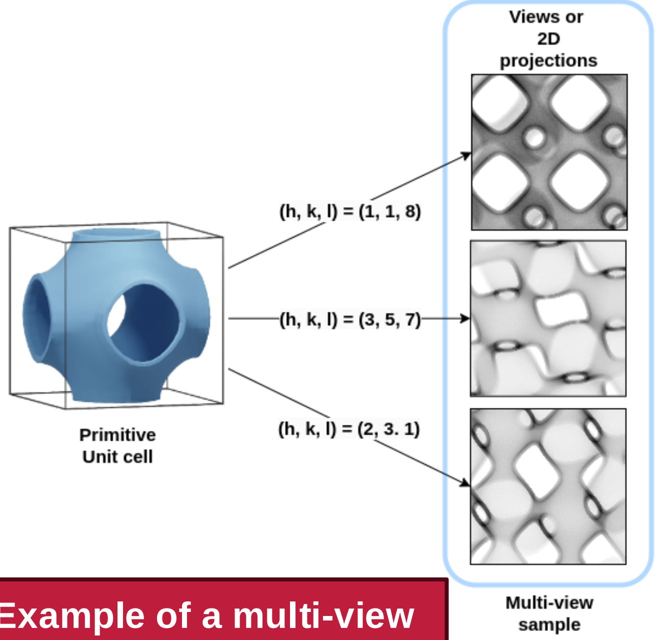
- **We need a CNN architecture:** Transfer learning
  - Model trained and tested on large-scale data
  - **How to use it:** Fine-tuning of the architecture on SPIRE data



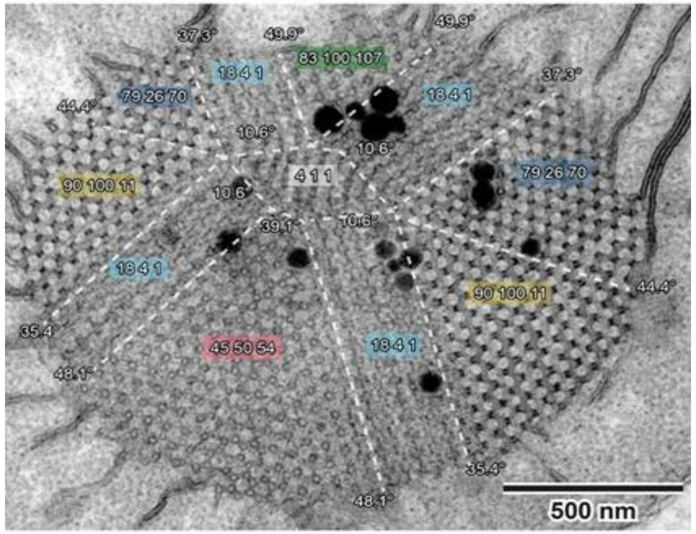
- **ResNet50 Network architecture:** 50 layers and 23.5 Million parameters
- **Pre-trained ResNet50:** Trained on 1.2 million images belonging to 1000 classes

2. Multi-View Convolutional Networks

- **Idea:** Combine information from different views and obtain more informative representations of an object
- **How to fuse? :** CNN layers have different levels of information both in abstraction and size
- **Which fusion strategy to use:**
  a) Late fusion (fully connected) strategy: best according to Seeland's paper
  b) Sum-score strategy: an additional strategy for comparison



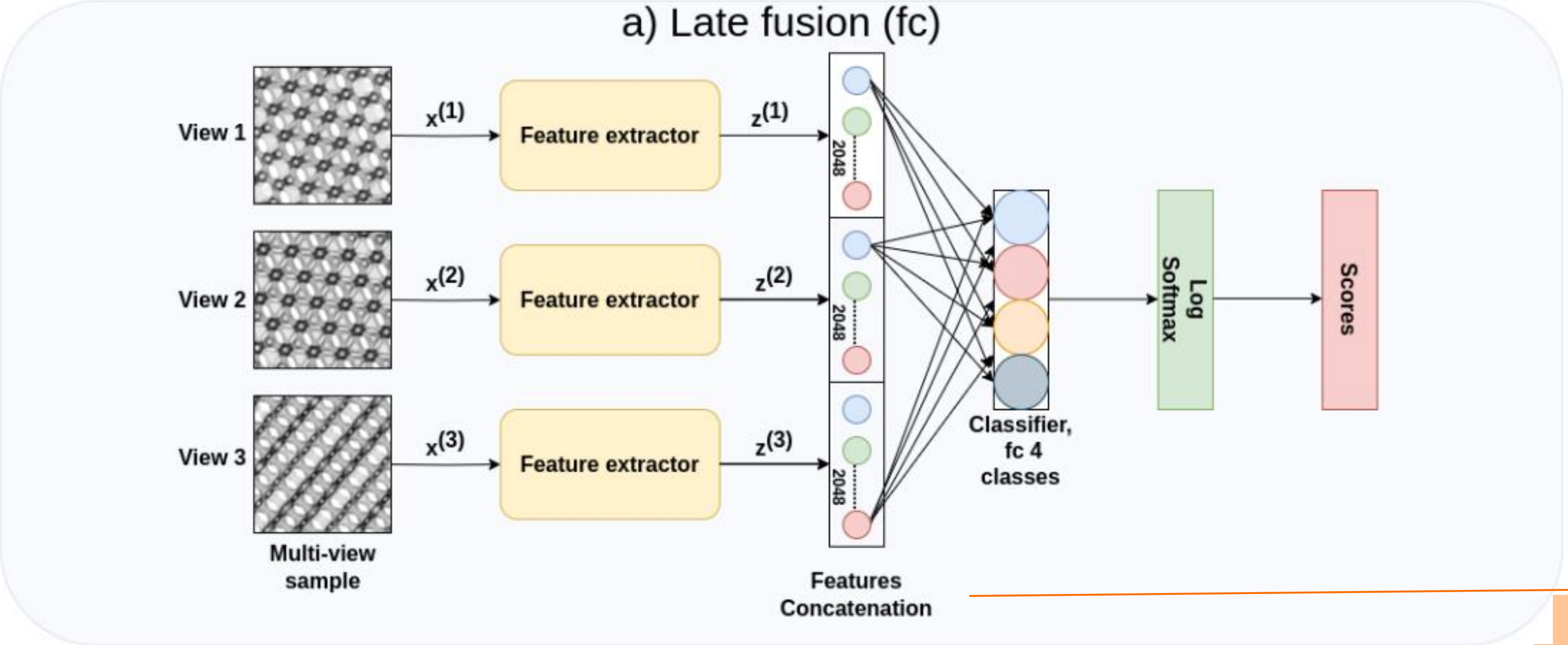
Example of a multi-view sample



Different sections or view in a micrograph

Source: SPIRE

2. Multi-View Convolutional Networks: Late fusion (fc)



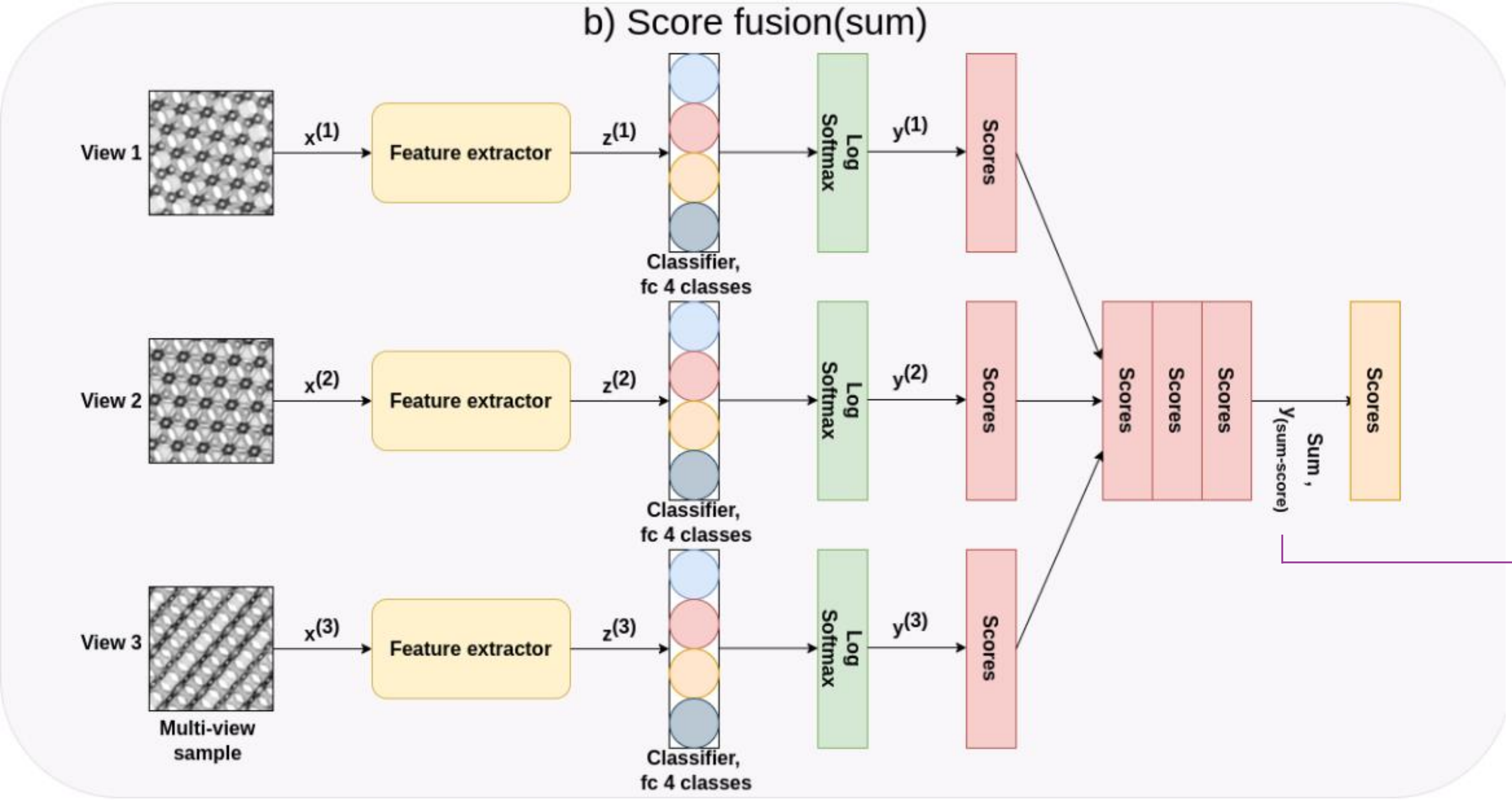
Creates a single feature from 3 views:

$$Z_V = [z^{(1)}, z^{(2)}, z^{(3)}]$$

# of parameters= 2048*3*4 = 24,576

(Length of feature vector * #views * # classes)

2. Multi-View Convolutional Networks: Score-sum fusion



Summing up scores of individual views:

$$\boldsymbol{y}_{(sum-score)} = \boldsymbol{y}^{(1)} + \boldsymbol{y}^{(2)} + \boldsymbol{y}^{(3)}$$

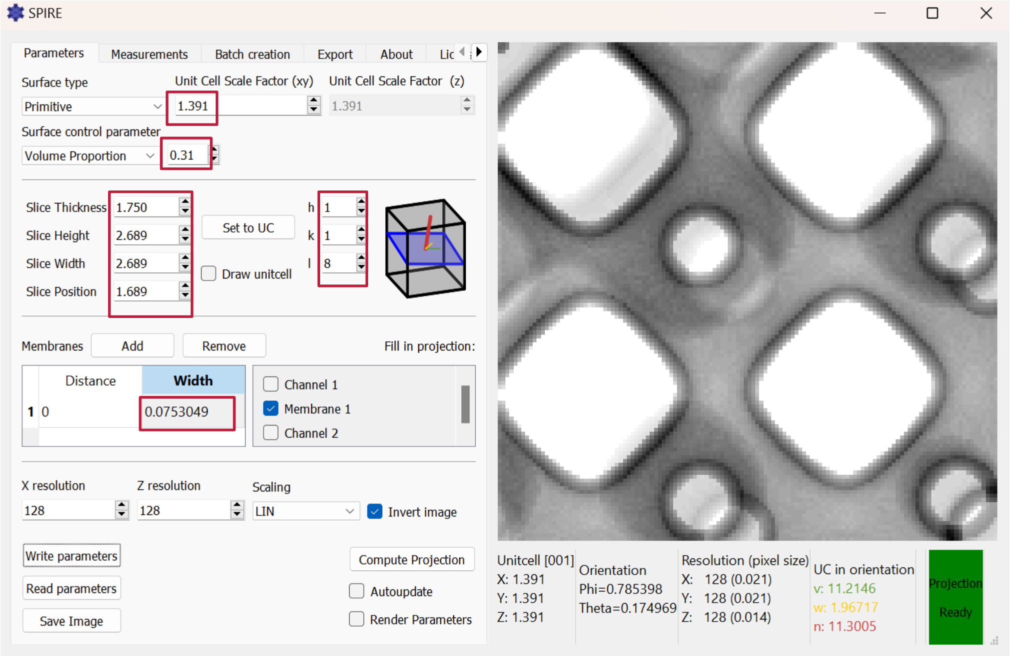
# of parameters= 2048*4 = 8,196

(Length of feature vector * # classes)

3. Experimental setup

Dataset generation

- Quality of data ~ Performance of ML model
- We don't have sufficient TEM micrographs →SPIRE generated data (or synthetic data)
- 10 parameters per class to generate 1 projection
- **Assumptions:**
  - Generated synthetic data matches real data distribution
  - Views in a single micrograph vary only in inclination parameters
- Train: validation: test :: 1000: 100: 100
- **Additional, noisy version of the dataset** to have another data distribution similar to real micrographs



Source: SPIRE

3. Experimental setup

Common data-related steps:

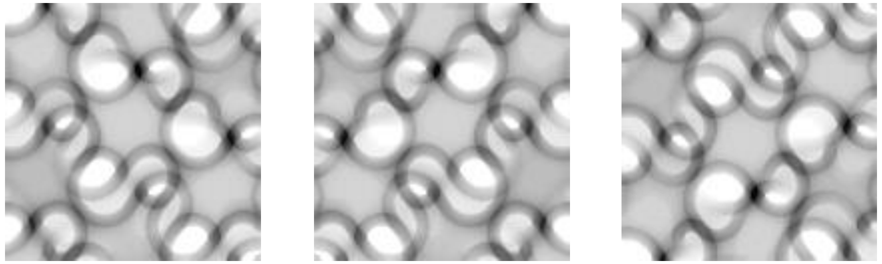
1. Data preprocessing as per ResNet50
2. Random data augmentation: Rotation and flipping (horizontal and vertical)

Training procedure:

1. Training a single view network (fine-tuning)
2. **Reuse** the convolutional base of the single view for multi-view
3. **Freeze** the branches of multi-view
4. Train the remaining layers (fusion layer, classification layer)

Data Augmentation

1. More samples
2. More varied training data distribution



Original   Flipped horizontally   Flipped vertically

3. Experimental setup

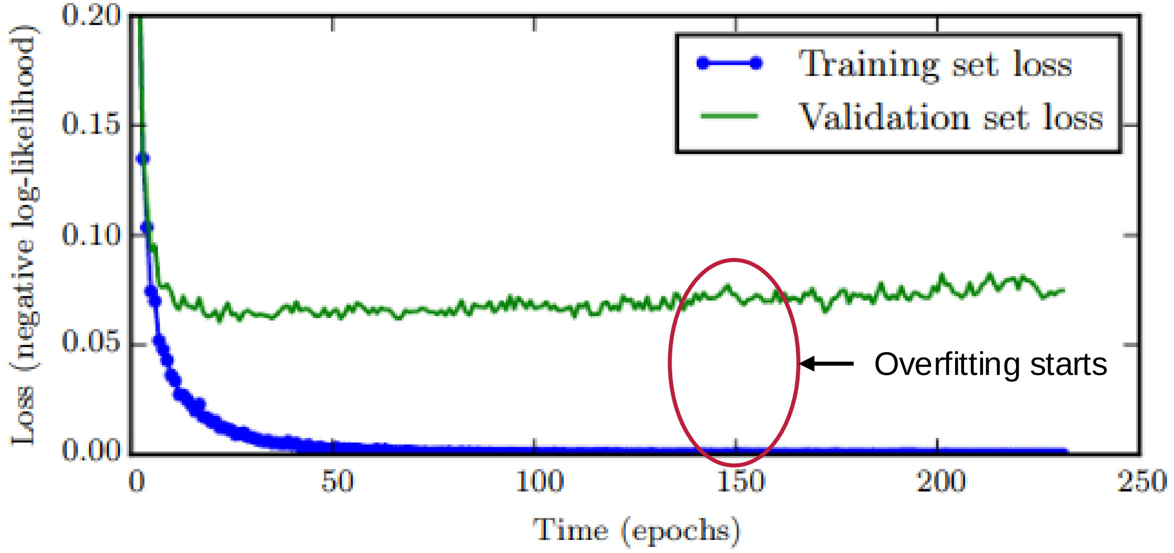
Performance metrics:

1. Classification accuracy → Overall performance
2. Confusion matrix → Class-wise performance
3. Training time → Computational requirements

Important training settings:

- Early stopping with 10 iterations without improvement to avoid overfitting
- Adam optimiser
- Negative Loglikelihood Loss for classification
- Fixed random seed (set to 21) for reproducibility of the same results

Trained 6 models: 3 on clean and 3 on noisy versions of the data



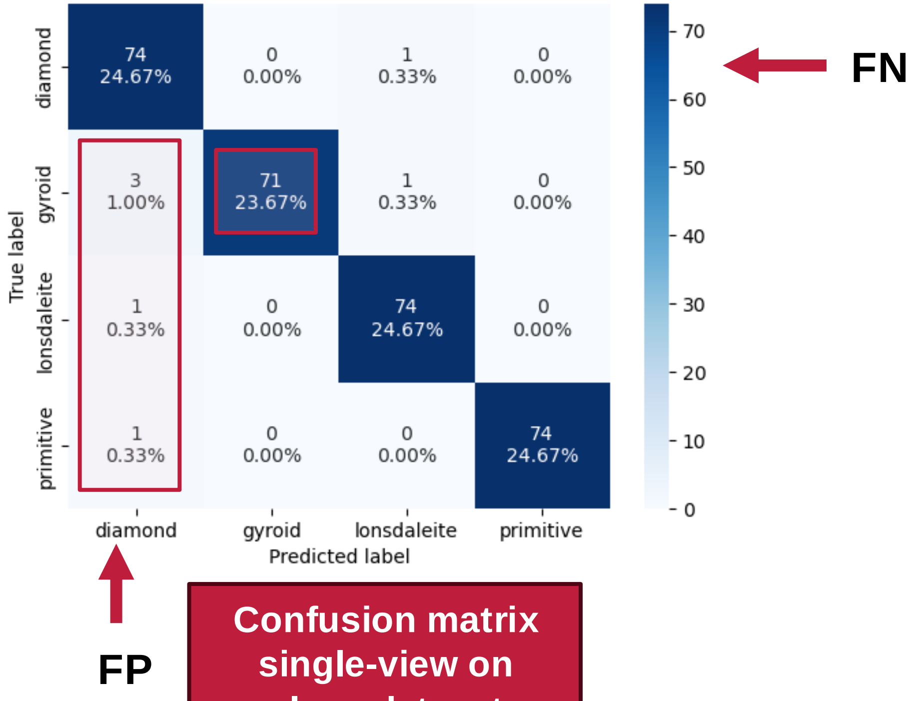
Source: Goodfellow, Bengio, Courville

4. Results

A. Results on SPIRE-generated data

Accuracy

Test results on SPIRE data					
Method	Layer	Test accuracy & δ_{BL} [%]			
		Clean dataset		Noisy dataset	
Avg. across single views		96.76		96.00	
Late (fc)	fc	100.00	3.24	98.00	2.00
Sum-score	softmax	100.00	3.24	98.00	2.00



Confusion matrix single-view on clean dataset

4. Results

B. **Computational requirements:** Training time

Method	Layer	Clean dataset Training time	Noisy dataset Training time
Single view		17.93	9.43
Late (fc)	fc	3.09	140. 68
Sum-score	softmax	3.60	5.91

Hardware: Graphics Processing Unit (GPU): NVIDIA GeForce GTX 1060 with 3GB memory

Training time in minutes

4. Results

C. Results on real TEM micrographs

- Real TEM micrographs could be gathered only for two classes, diamond (40) and gyroid (20).
- One SPIRE-generated synthetic sample each for lonsdaleite and primitive class

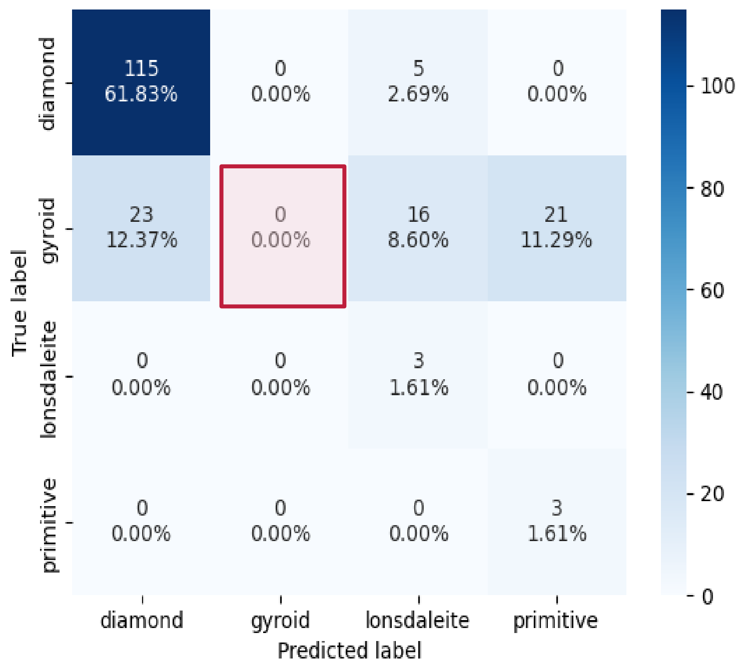
Accuracy

Test results on Real TEM specimens data					
Method	Layer	Test accuracy & δ_{BL} [%]			
		Model trained on Clean dataset		Model trained on Noisy dataset	
Avg. across single views		65.06		21.50	
Late (fc)	fc	67.74	2.68	$\boxed{17.74}$	-3.76
Sum-score	softmax	66.13	1.07	$\boxed{40.32}$	18.82

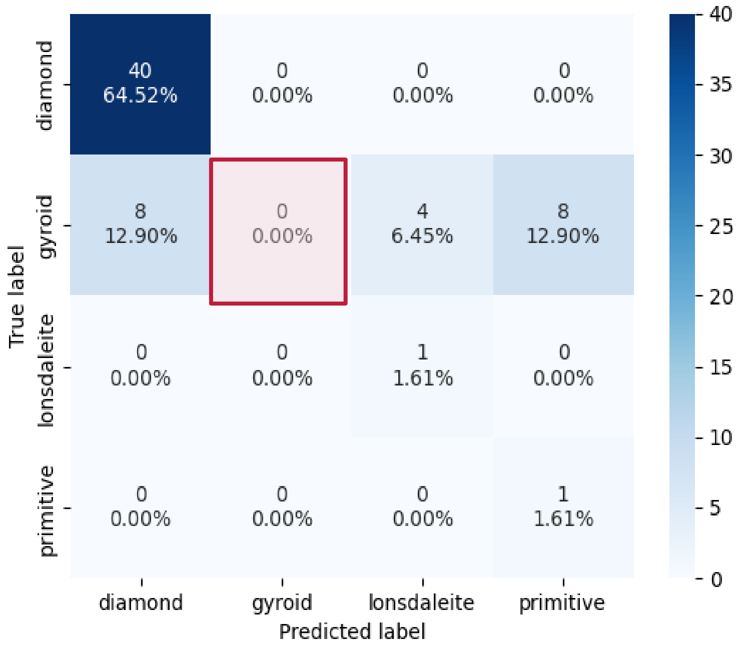
4. Results

. C. Results on real TEM micrographs

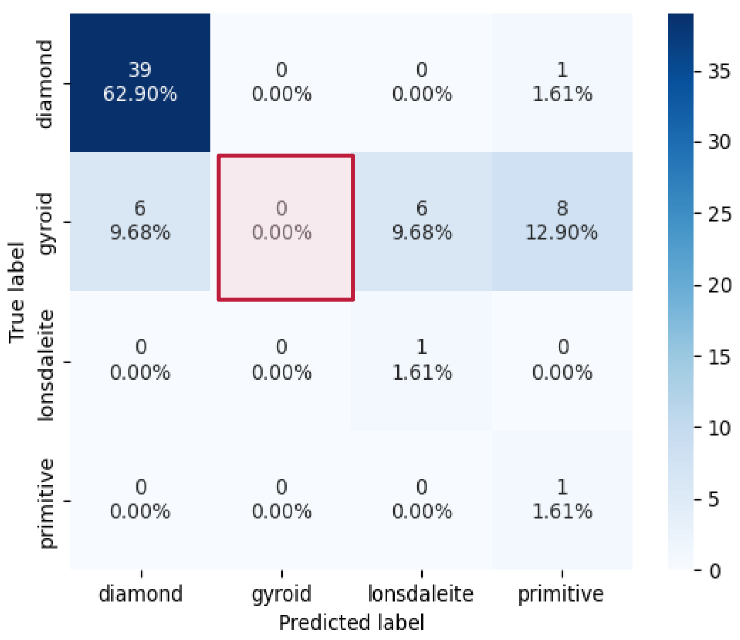
Confusion matrices on clean dataset



Single-view



Late (fc)

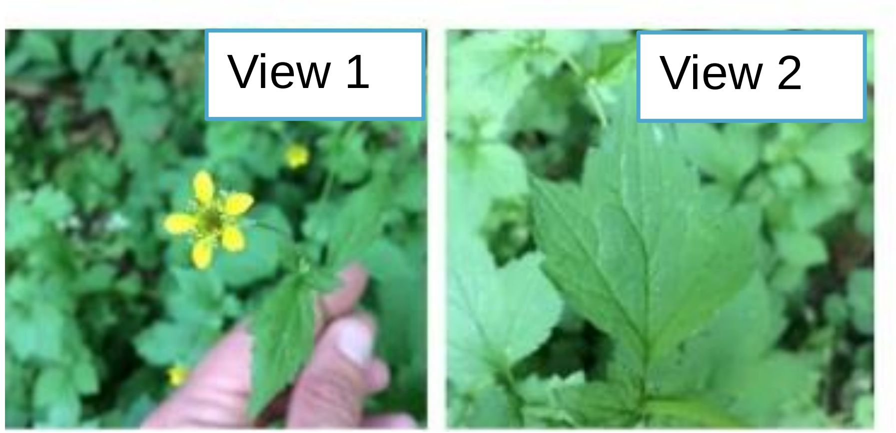


Sum-score

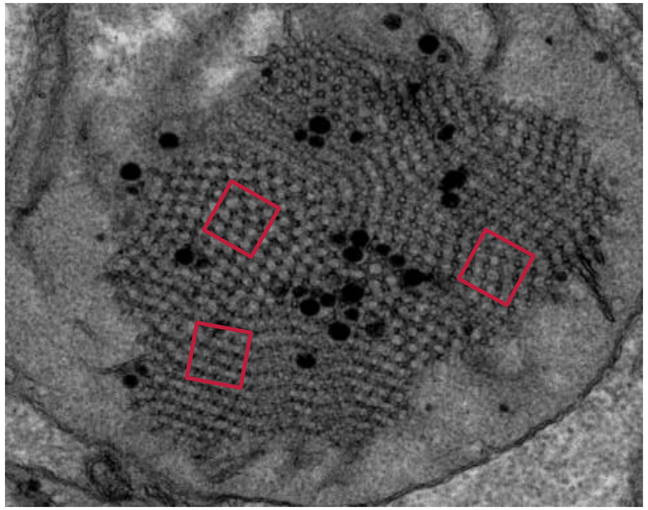
5. Discussion

A. Contradicting results w.r.t. Seeland’s paper

- Late fusion (fc) is not clearly advantageous than sum-score fusion
- Late fusion (fc) did not converge with the noisy dataset
- Late fusion concatenates the extracted features, but input images don’t have a fixed order



Source (Adapted from): Seeland, M., & Mäder, P. (2021).

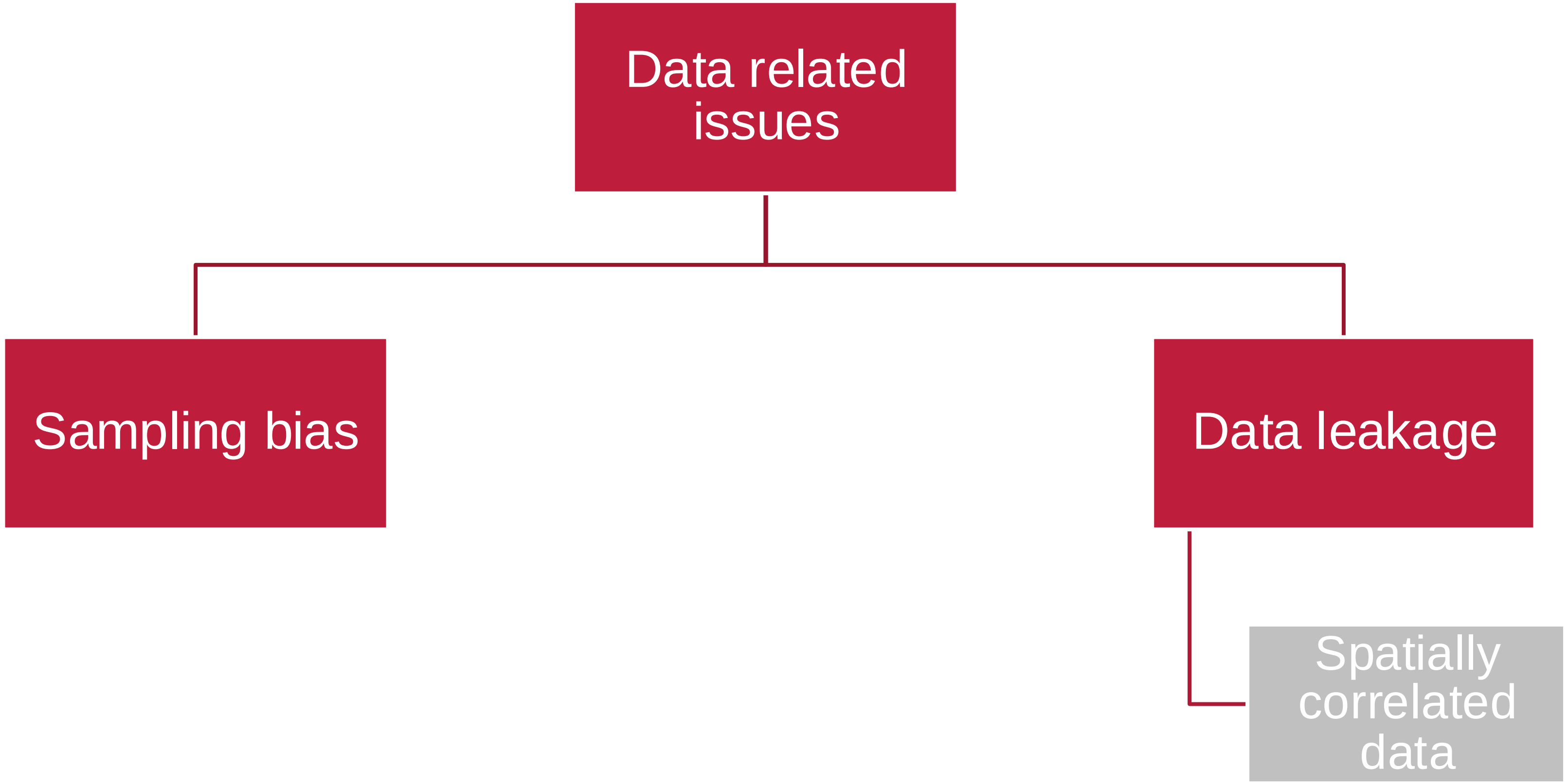


Ordered vs. Unordered

Technische Universität Braunschweig
Institut für Analysis und Algebra

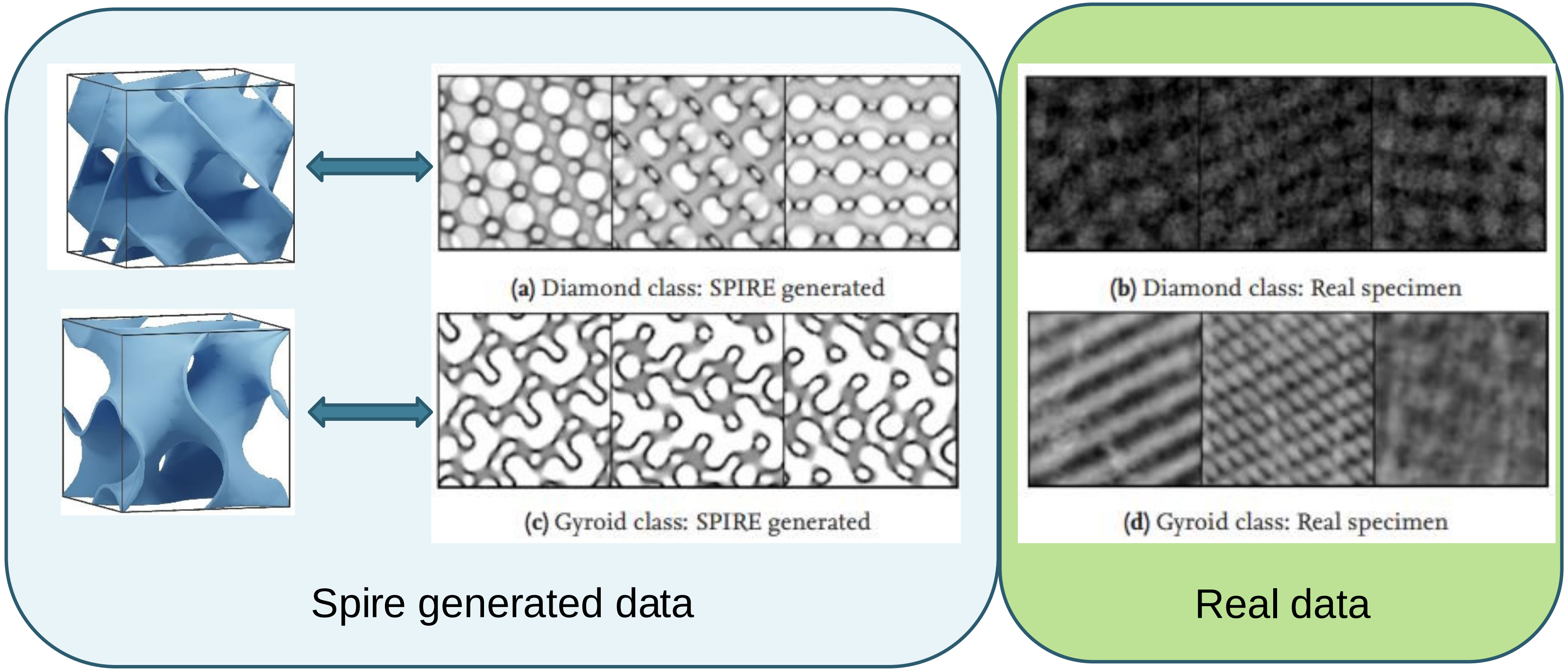
5. Discussion

B. Data-related issues



5. Discussion

a) Sampling bias: Collected data does not sufficiently represent the true data distribution of the underlying problem



- The extent of bias is inconclusive as we don't have real samples from all classes.

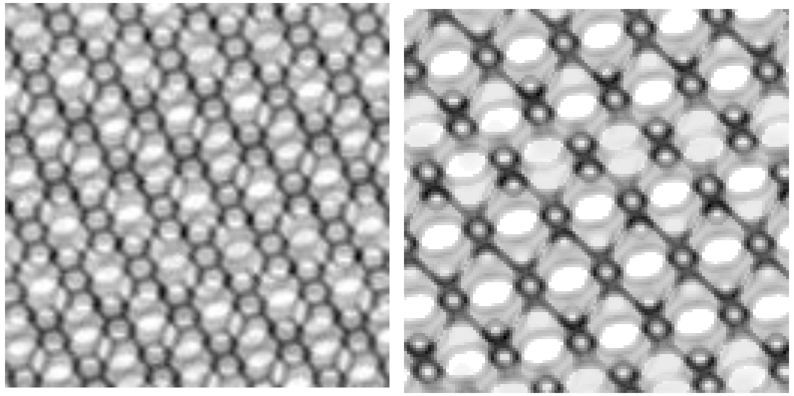
5. Discussion

b) Data leakage: The model has information related to test data which is not available in the practice, resulting in exceptionally good performance.

Source: Bored panda

Training data Test data

Example:
Similar or correlated data samples in train and test → Inflated accuracy



Training data Test data

Diamond class with (h,k,l) : 10,5,9 and rest 7 parameters different

Spatially correlated data

6. Conclusion

- Can CNN classify chloroplast micrographs?: YES, with limited accuracy
- Is MVCNN better than CNN? : YES
- Are models trained on SPIRE data applicable to real micrographs? PROMISING RESULTS, BUT further evaluation is required

Thank you!

References

- Tobias M Hain et al. “SPIRE—a software tool for bicontinuous phase recognition: application for plastid cubic membranes”. In: Plant Physiology 188.1 (Jan. 1, 2022)
- Marco Seeland and Patrick Mäder. “Multi-view classification with convolutional neural networks”. In: PLOS ONE 16.1 (Dec. 1, 2021).