Identifying ancient "insoluables"

Christoffer Øhlenschlæger, Nicklas Christiansen, Magnus Diamant og William Pedersen

All contributed evenly

KØBENHAVNS UNIVERSITET

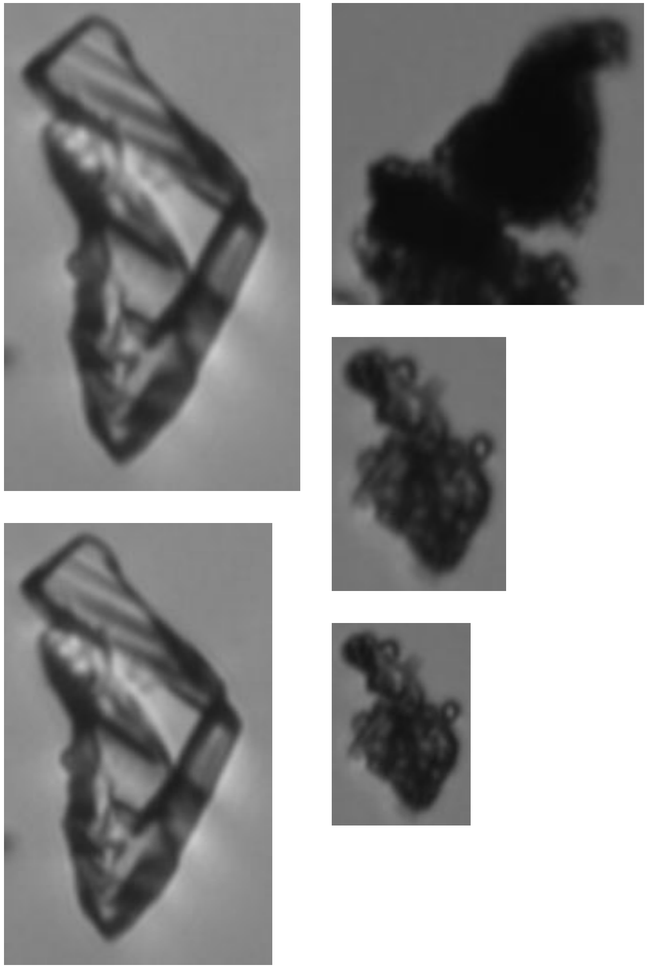
What is the project about?

- Known dataset is obtained from controlled samples
- Unknown dataset is obtained from Peru from a melted ice core filtering process
- Objective 1: What materials (and distribution) is this unknown dataset made of based on the labels from the known dataset?
- Objective 2: What if there are other materials than the 7 classes we've seen in the known dataset? Anomaly/outlier detection

source: https://www.nature.com/articles/d41586-019-02566-9

Overview

- Data exploration
- LightGBM classification on MetaData
- CNN + UMAP
- Autoencoder + UMAP
- Outlier detection with Isolation Forest



Data Exploration

- 147960 samples
- 50 features (target and identification columns excluded)
- Dataset unbalanced - rebalancing the dataset for the models using MetaData only and using weights in the loss function for the NN's
- Length of unknown dataset 102764
- Generated balanced hold-out test (728 cases each)

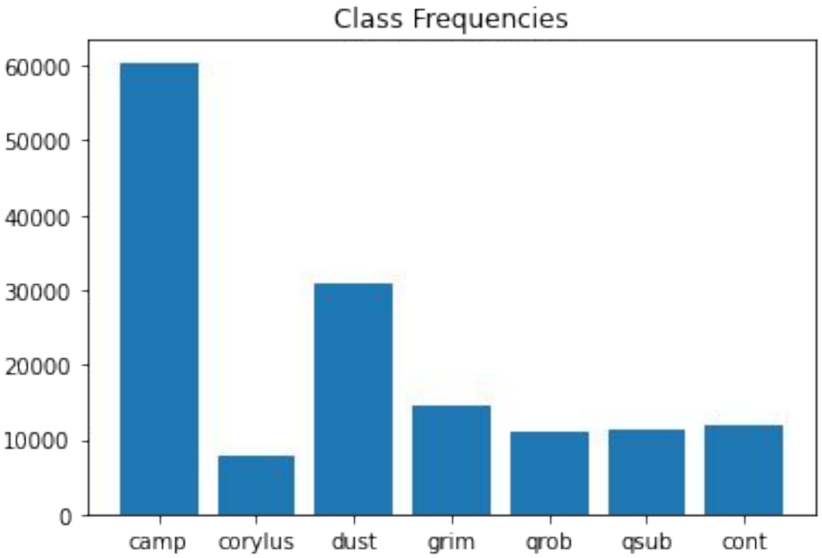
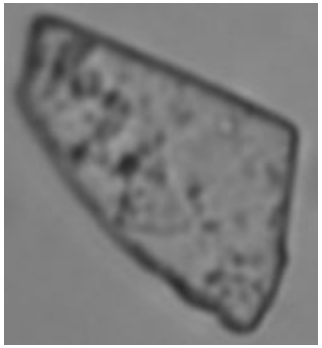


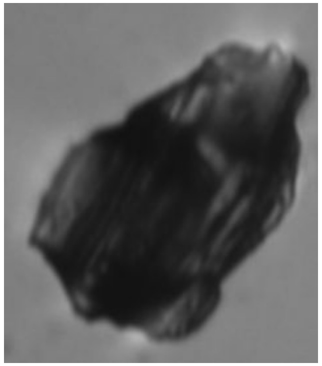
Image class examples

Ash

Campanian



Grimsvotn

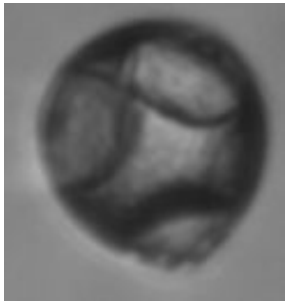


Contamination

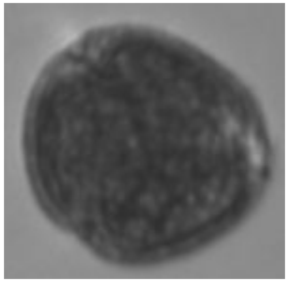
Contamination

Pollen

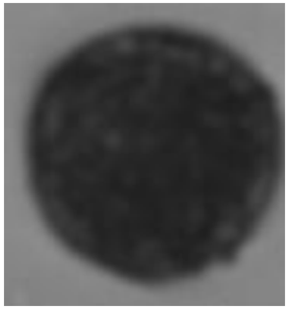
Corylus



Qrubur

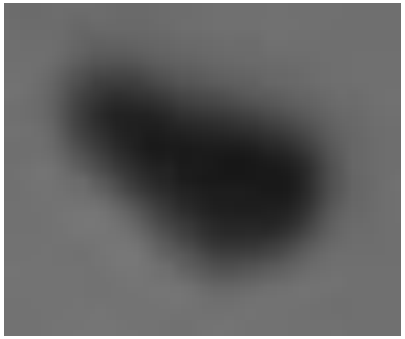


Qsuber



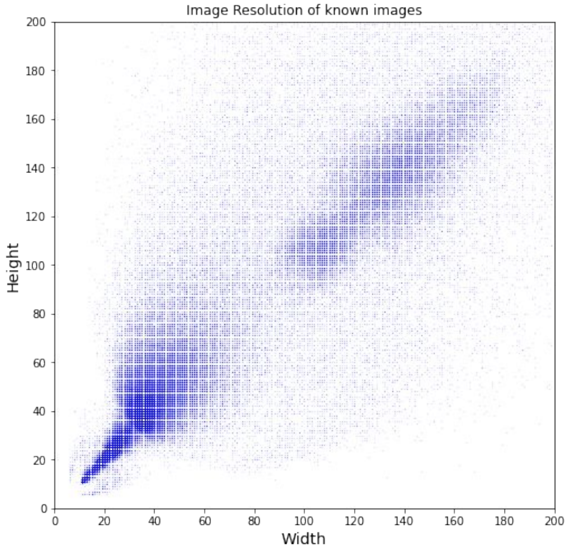
Dust

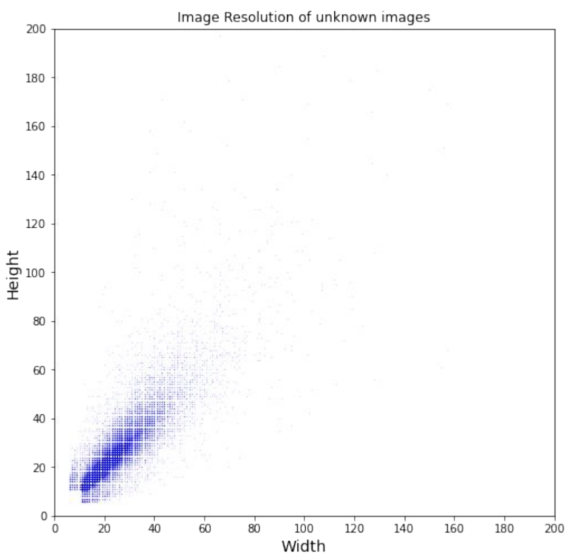
Dust



Data preprocessing

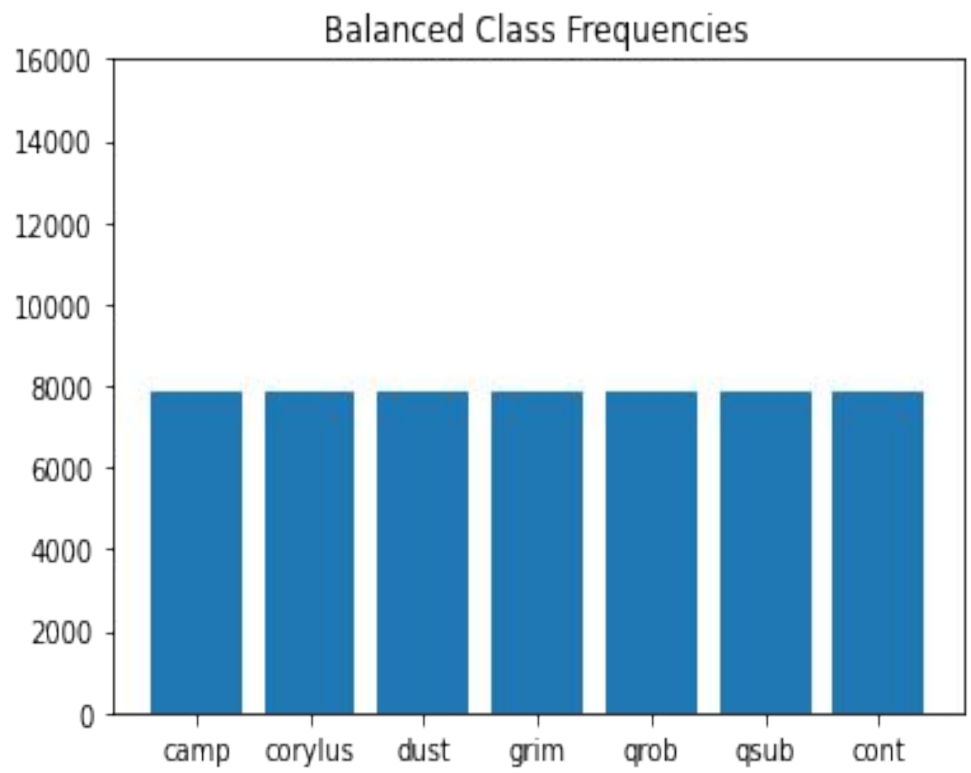
- Resize images to 128x128
- Normalize
- Grayscale





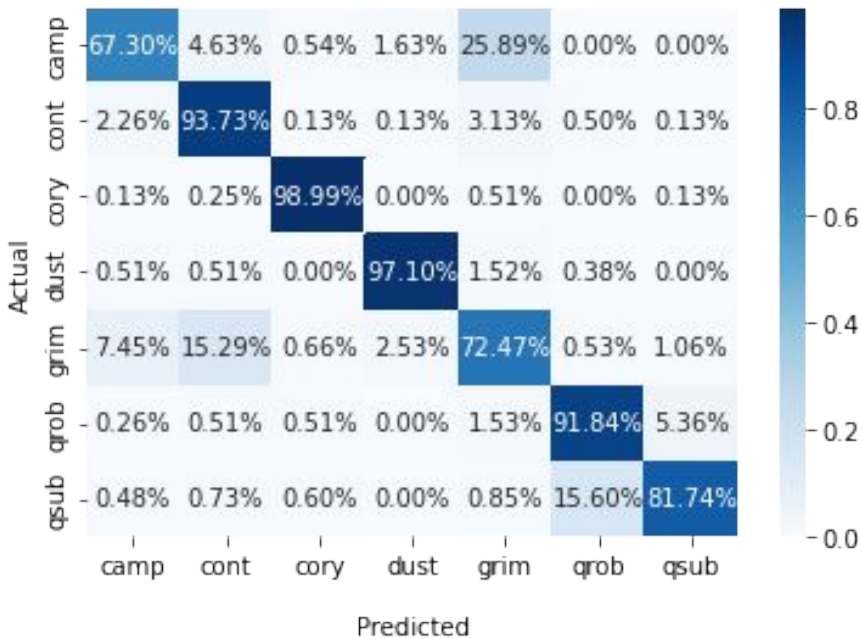
LightGBM on MetaData

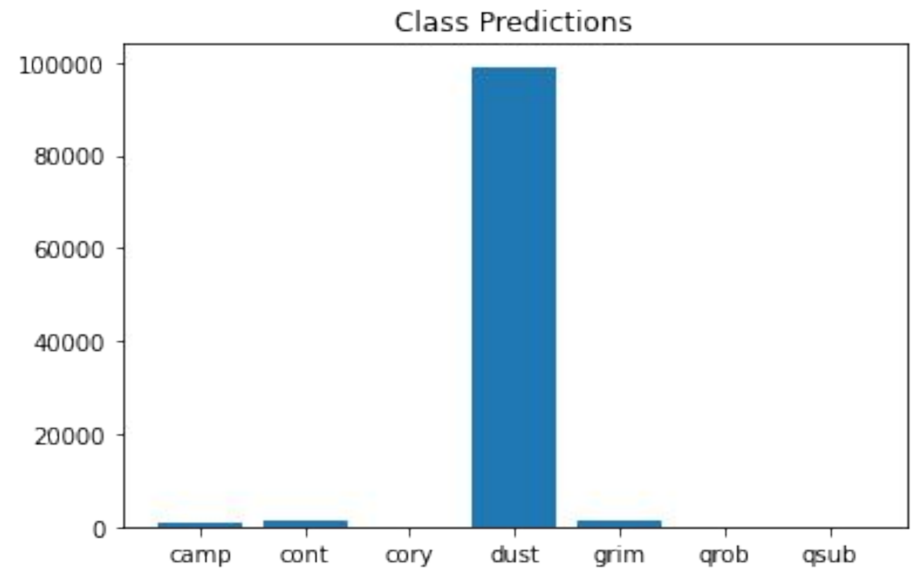
- Tree based gradient boosting algorithm
- Numerical features
- Balanced dataset
- Randomized Search
- KFold Cross Validation



LightGBM results

- Accuracy on 7 classes: 86%
- Predicts that almost all samples in unknown dataset belong to Dust class

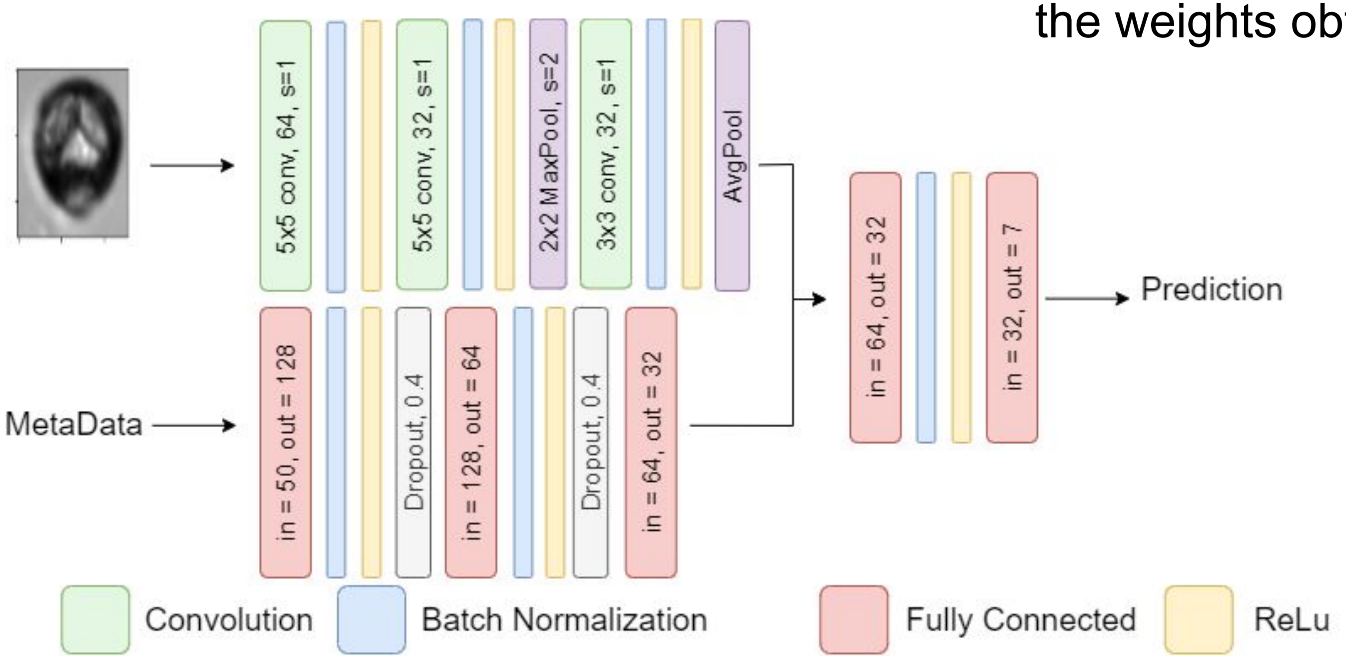




Predictions						
camp	cont	corylus	dust	grim	qrob	qsub
1075	1240	19	99116	1283	24	7

Classification using CNN

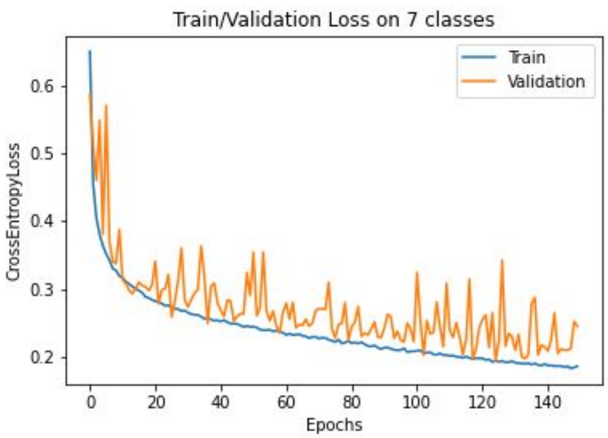
- Build our own CNN network
  - Use weights in the loss function
  - Add Batch Normalization
  - Add MetaData
- Use a pretrained ResNet model
  - Pretrained cnn network on more than a million images
  - 18 layers deep
  - Network is trained on more than 1000 object classifications - we can use the weights obtained



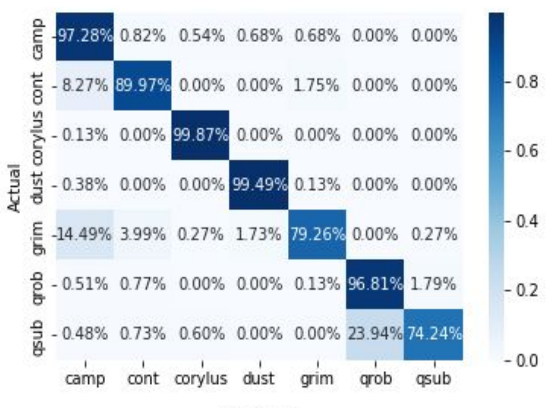
Classification with CNN

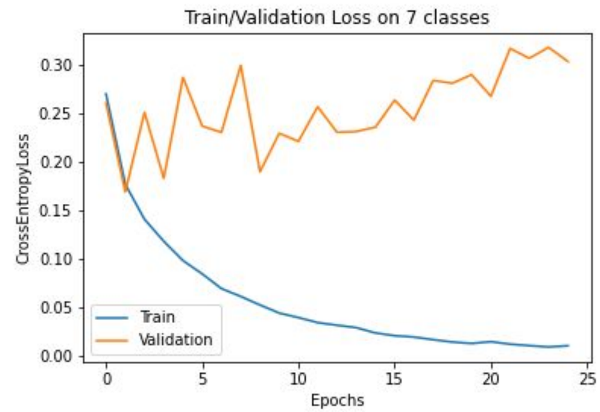
Our own model

ResNet



90.89%

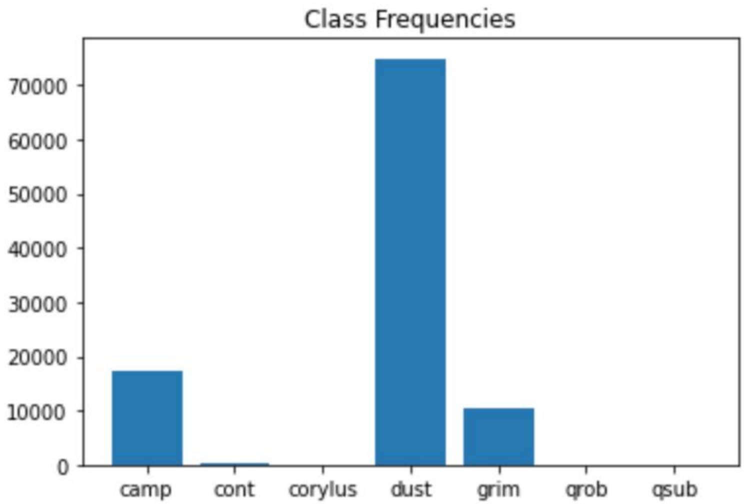




94.55%

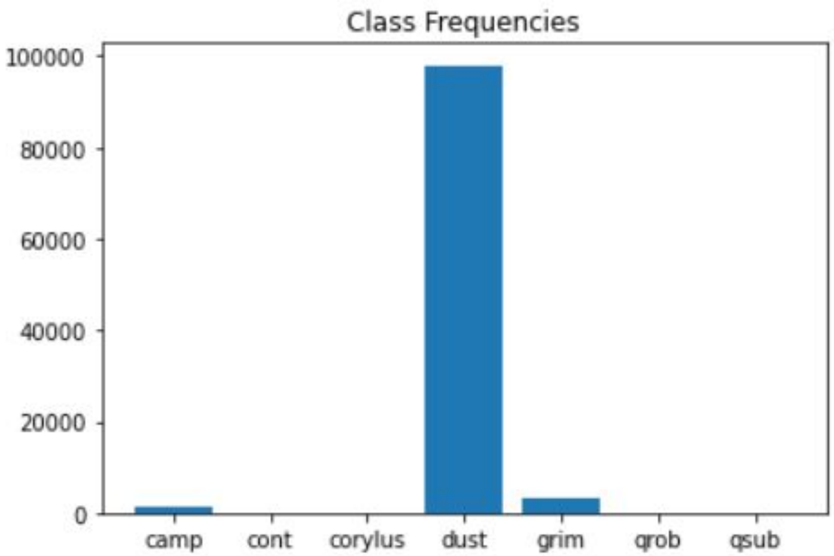
Classification with CNN

Our own model



Predictions						
camp	cont	corylus	dust	grim	qrob	qsub
17279	219	11	74911	10277	37	30

ResNet

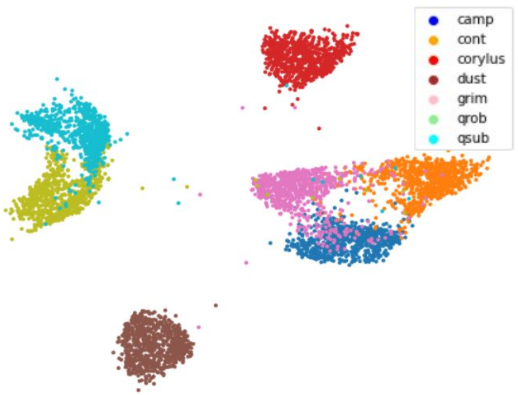


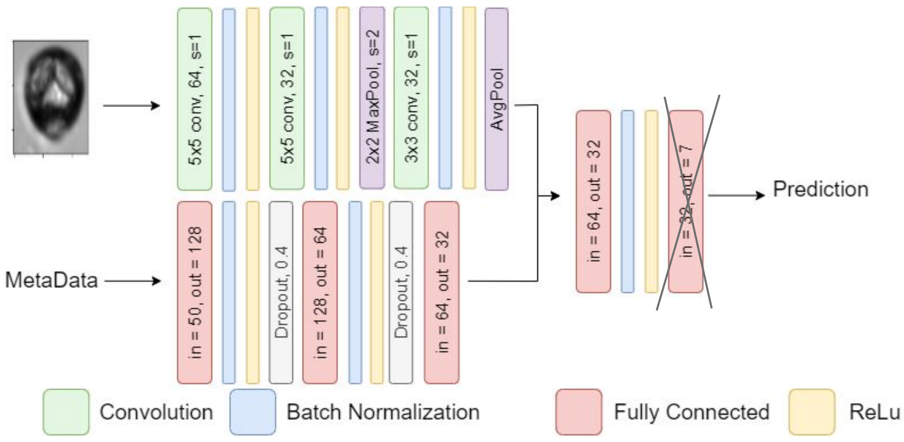
Predictions						
camp	cont	corylus	dust	grim	qrob	qsub
1255	35	122	97960	3342	0	50

Outlier detection using CNN

- Use ParametricUMAP on the second to last layer
- Using ResNet
  - Divide dataset into 4 subclasses
  - Train new model and UMAP
  - Use embedding to check for anomalies in unknown dataset
  - Run experiments with training on excluded classes to see if the model actually can find anomalies

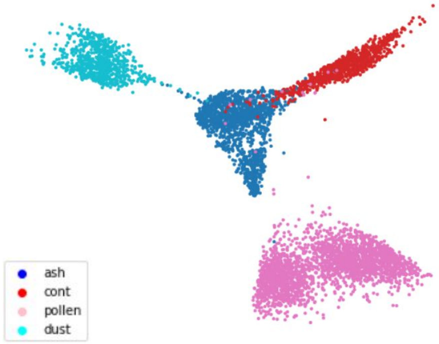
ResNet trained on 7 classes UMAP’d





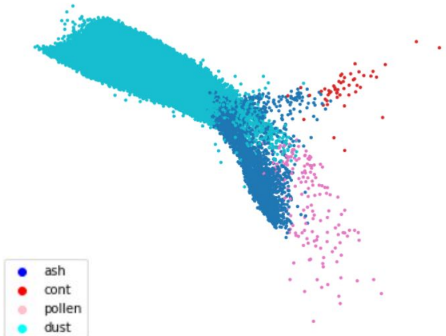
Anomaly detection with CNN

Model trained on all 4 subclasses

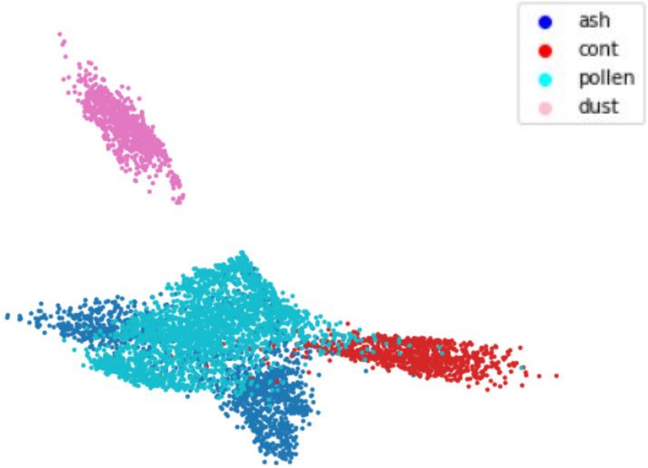


Dust removed in training - used to predict

Used on unknown data

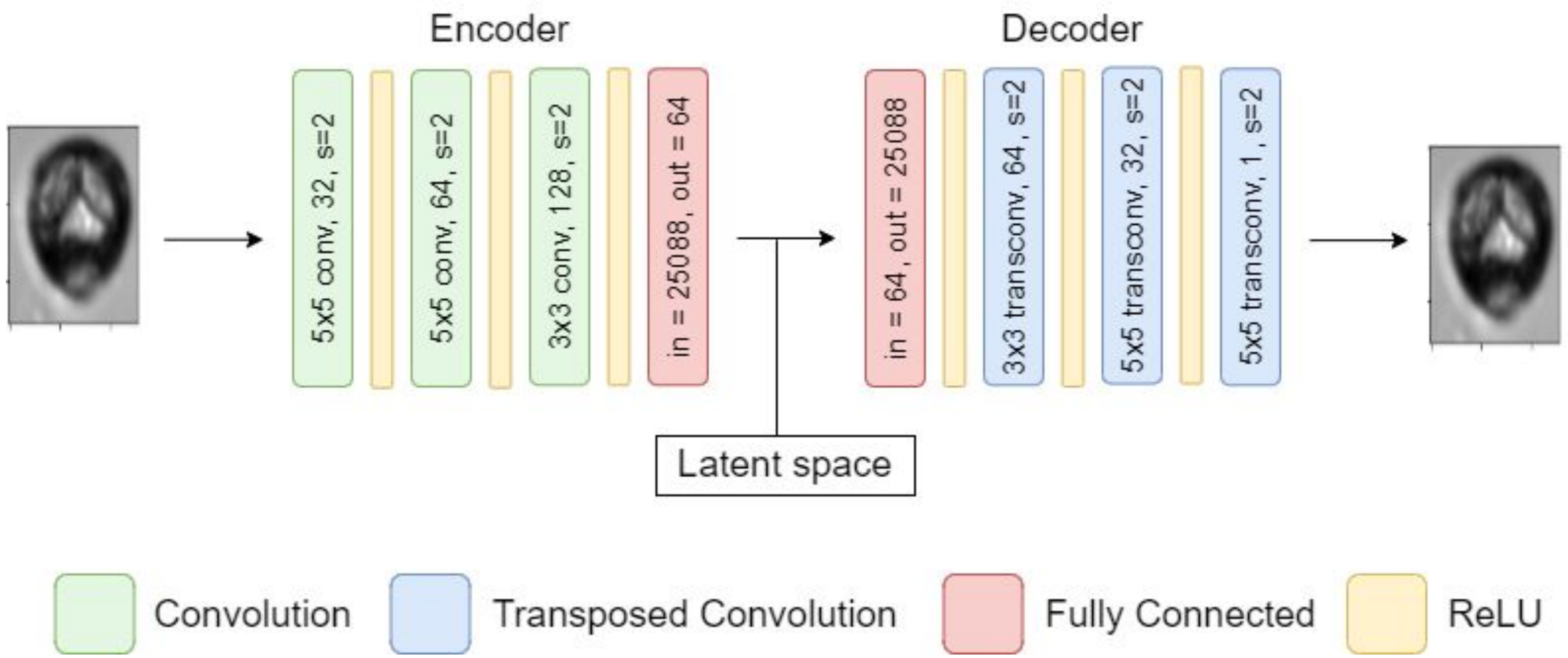


Pollen removed in training - used to predict



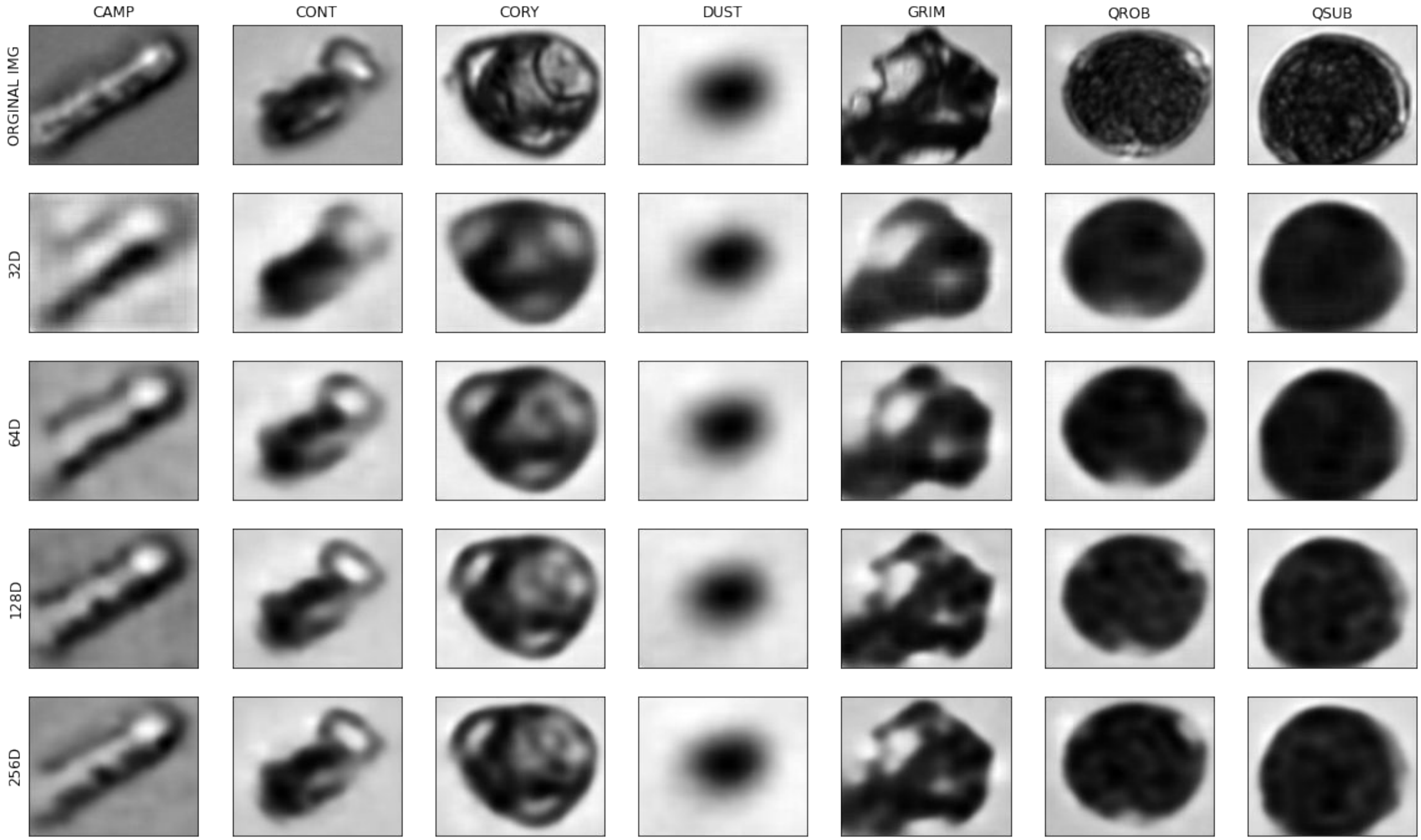
Autoencoders for outlier detection

Encoding the data



Autoencoders for outlier detection

Reconstruction of input with different latent space dimensions



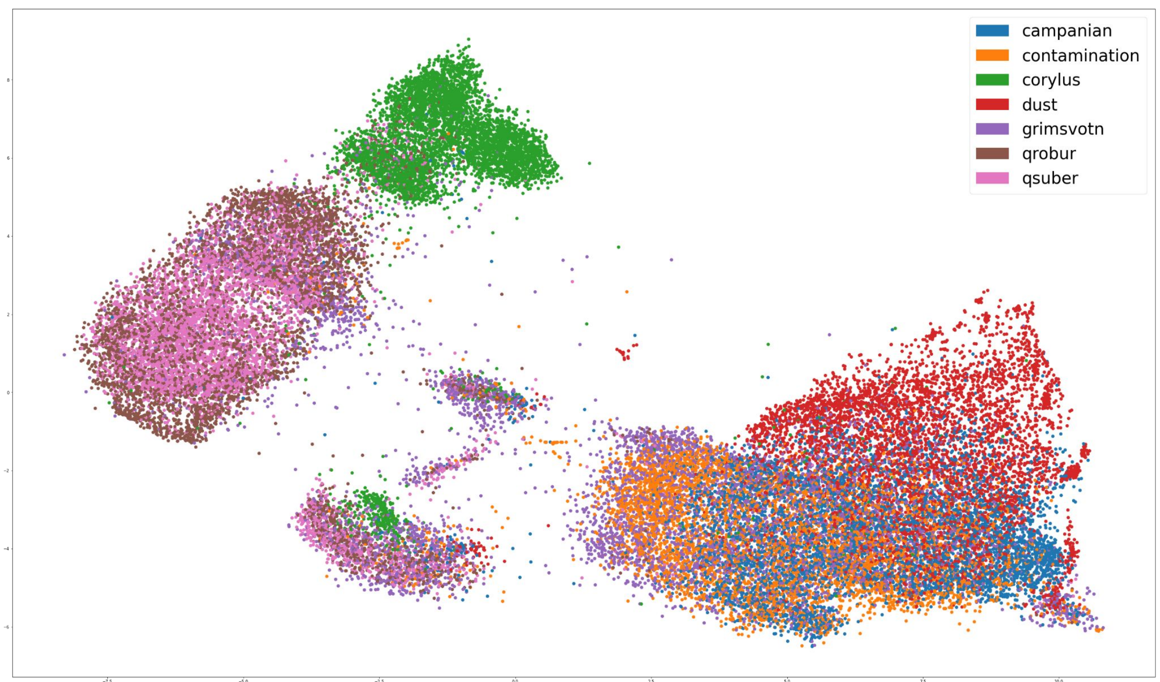
Autoencoders for outlier detection

Finding global outliers

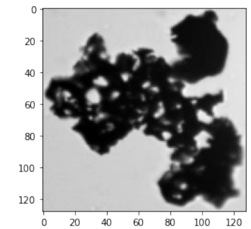
Latent space

↓

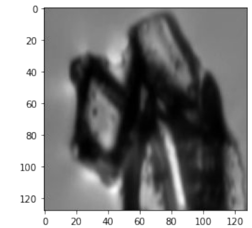
ParametricUMAP



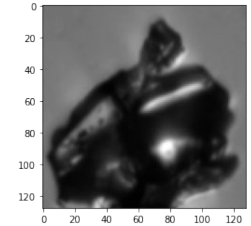
With reconstruction loss



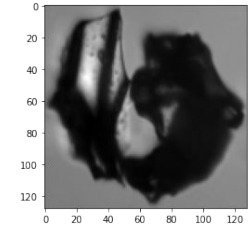
MSE: 0.0349



MSE: 0.0343

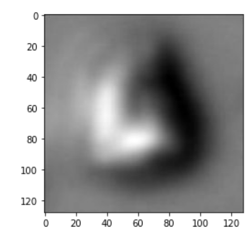


MSE: 0.0316

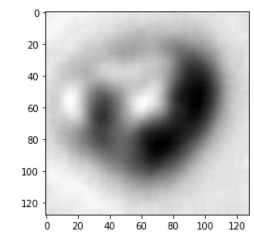


MSE: 0.0301

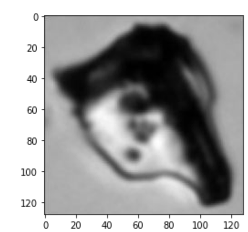
With UMAP and Local Outlier Factor



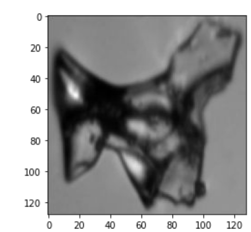
NLOF: -4.98



NLOF: -4.57



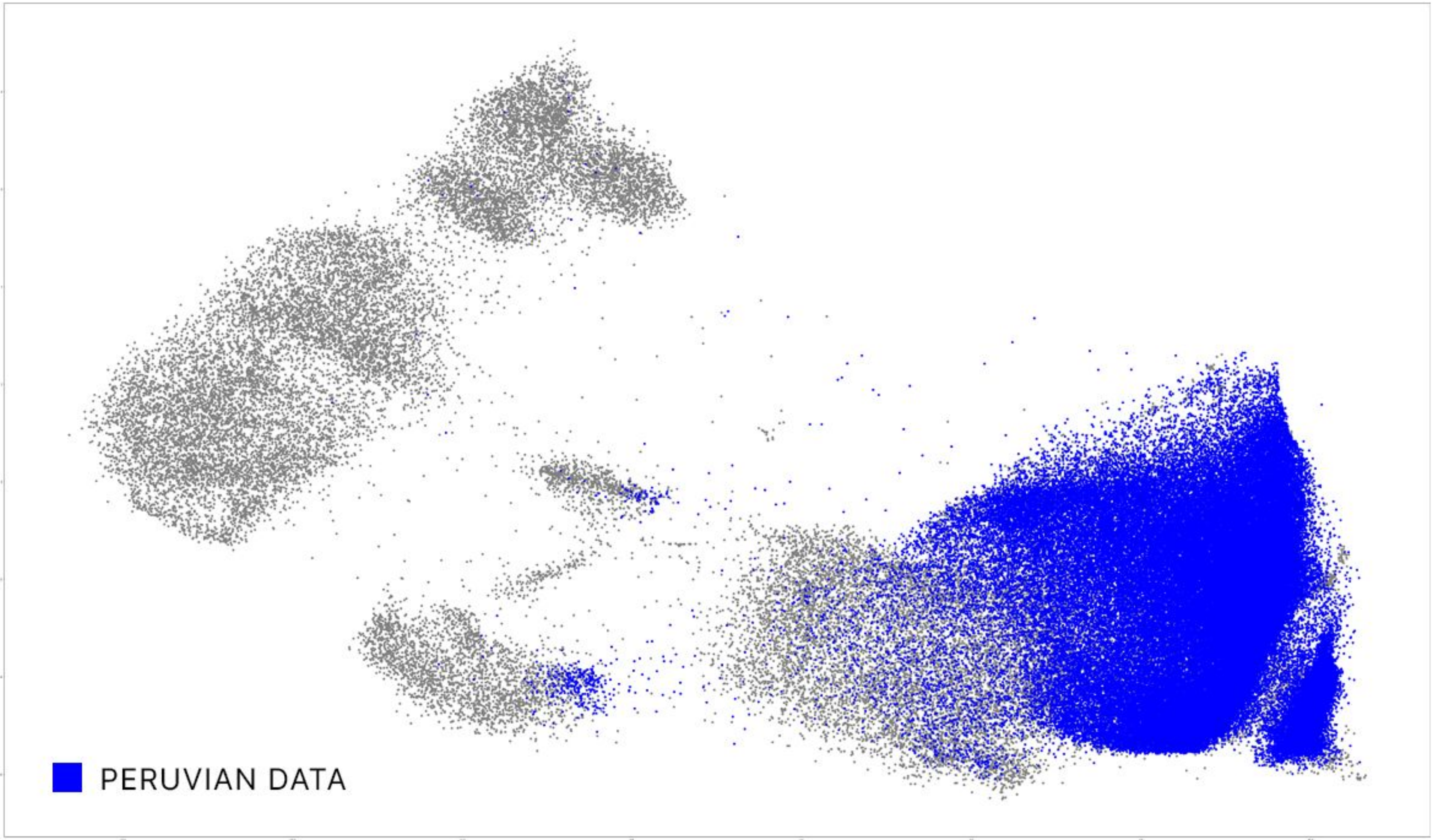
NLOF: -3.84



NLOF: -3.66

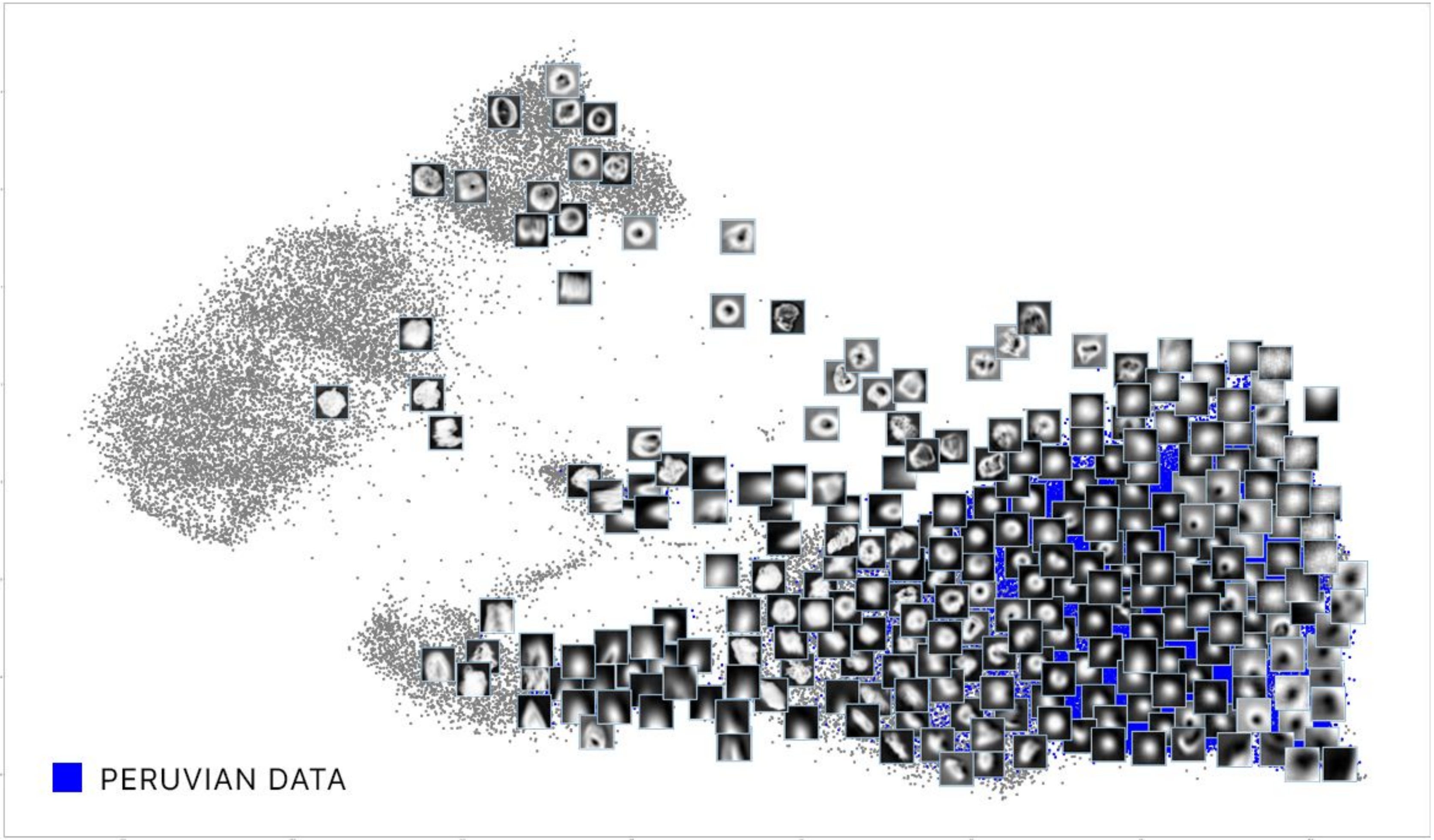
Autoencoders for outlier detection

Finding points of interest in the unknown dataset

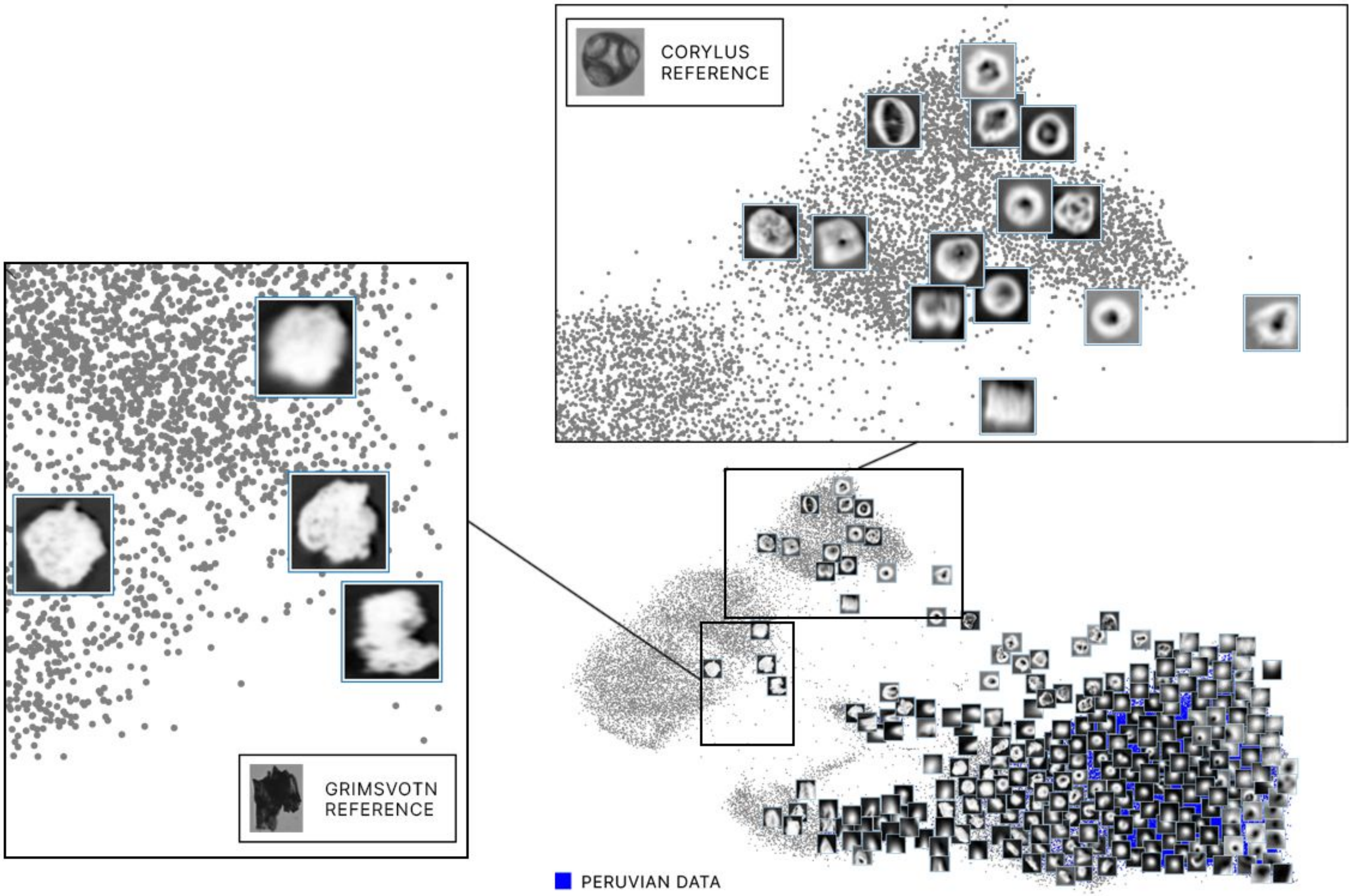


Autoencoders for outlier detection

Finding points of interest in the unknown dataset

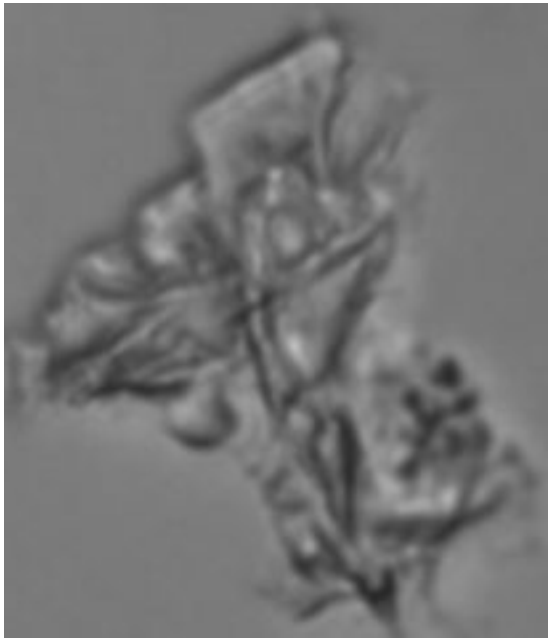


Autoencoders for outlier detection

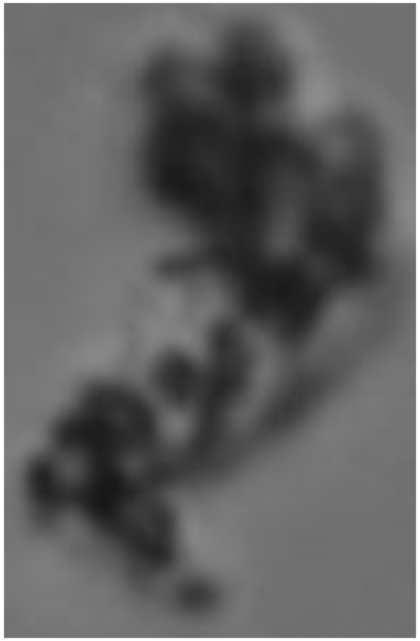


Isolation forest: Top 3 outliers

- Dropping all non-numerical features from unknown data set
- Isolation forest fitted on unknown data set
- Images below got the lowest scores (Outliers)



QCY_27_2_1_31.png



QCY_23_3_4_626.png

QCY_25_6_1_4.png

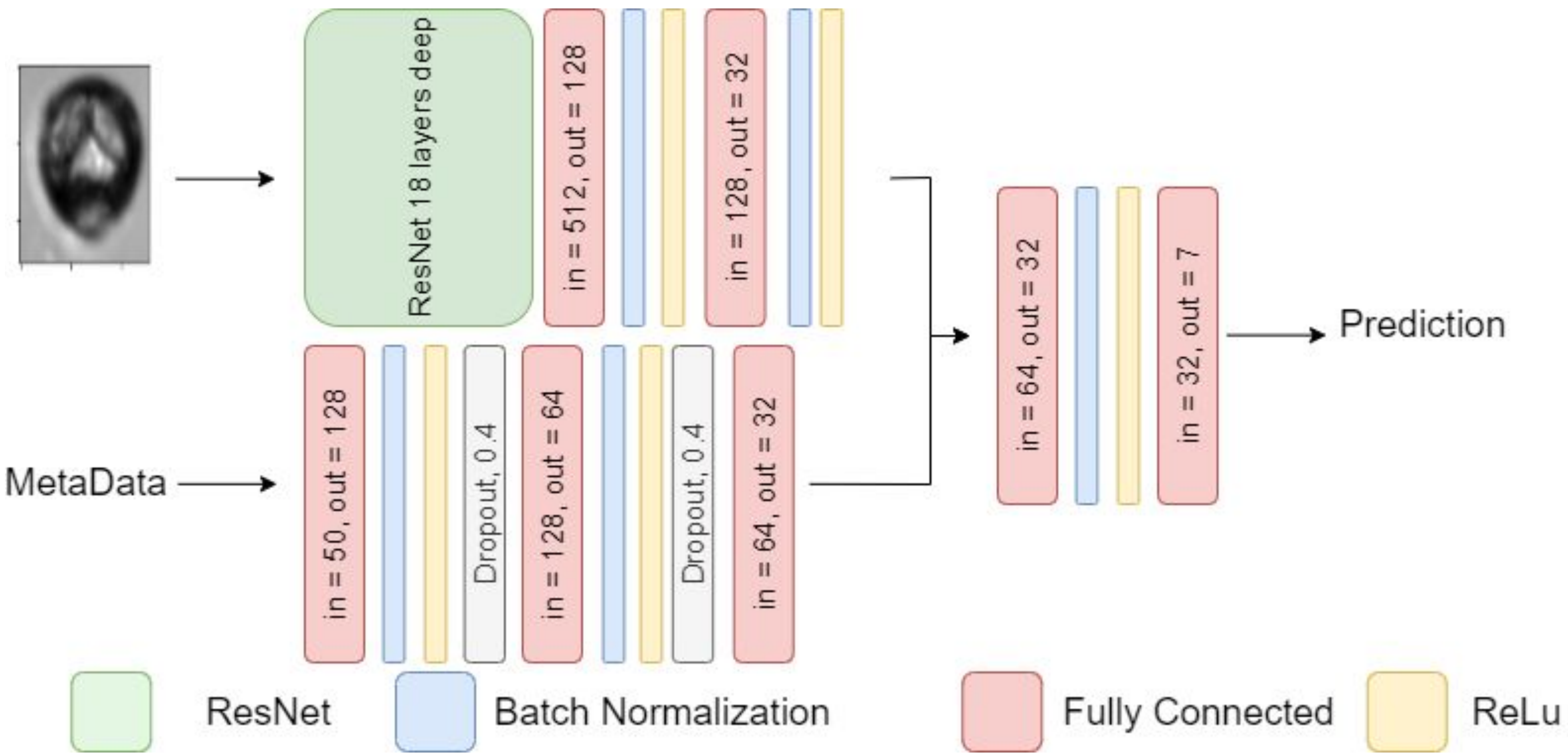
Future work

- More experiments with layers and sizes of the networks
- Experiment with data augmentation - signal and image processing
- More experiments with the CNN + UMAP combination to find interesting images/outliers

Conclusion

- Reasonably good accuracy using tree-based learners on the metadata
- ResNet predicting on holdout testset
    - 7 classes: 95%
    - 4 subclasses: 98%
    - Seems to generalize well on unknown dataset
- Autoencoder detected interesting images in the unknown dataset
- Isolation Forest detected interesting images in the unknown dataset

Appendix: ResNet architecture

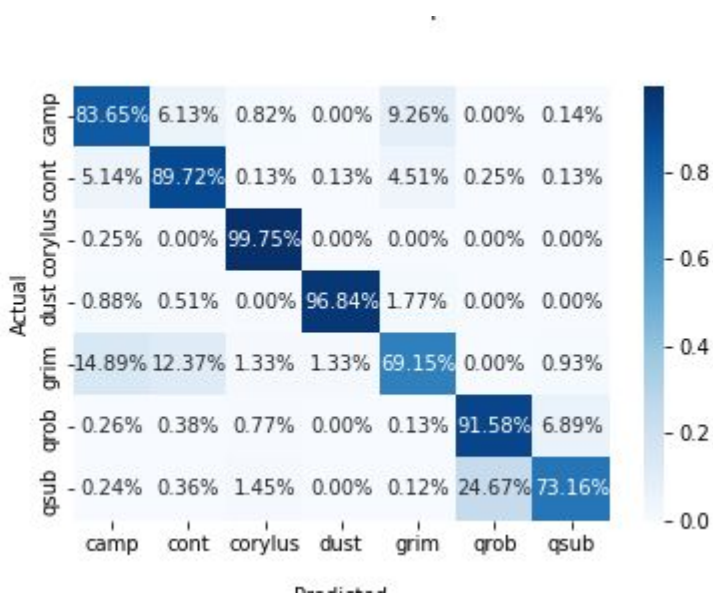


Appendix: CNN experiments

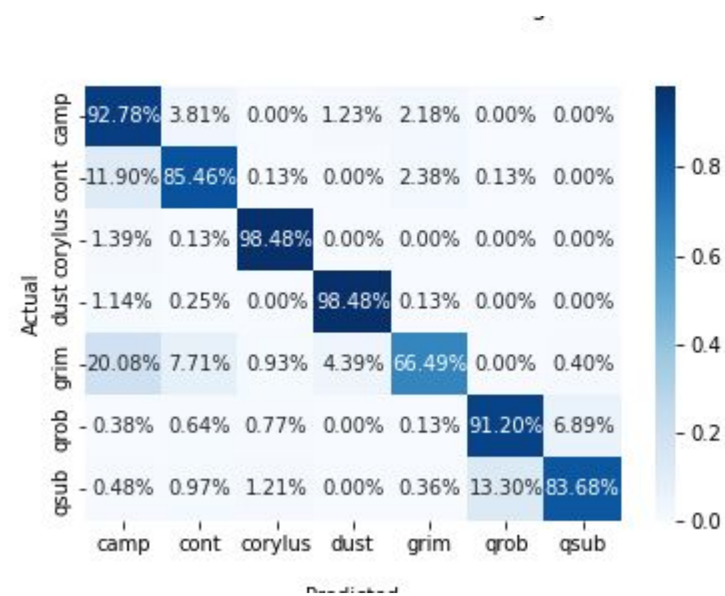
- Each of the following experiments has been conducted with 25 epochs due to limitations in time and computation power
- Each experiment has been conducted 3 times and average accuracy has been reported
- All other settings have been kept equal except for the setting being tested

Appendix: CNN experiment: Downsample dataset / using weights

86.31% accuracy on testset

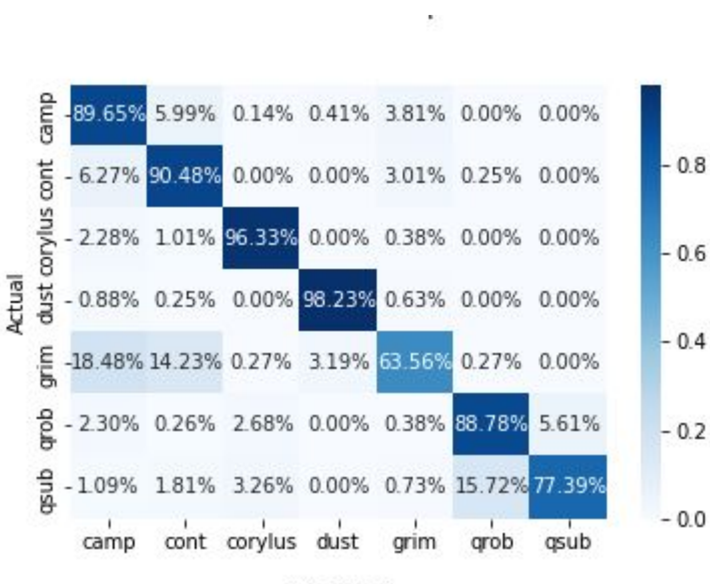


88.15% accuracy on testset

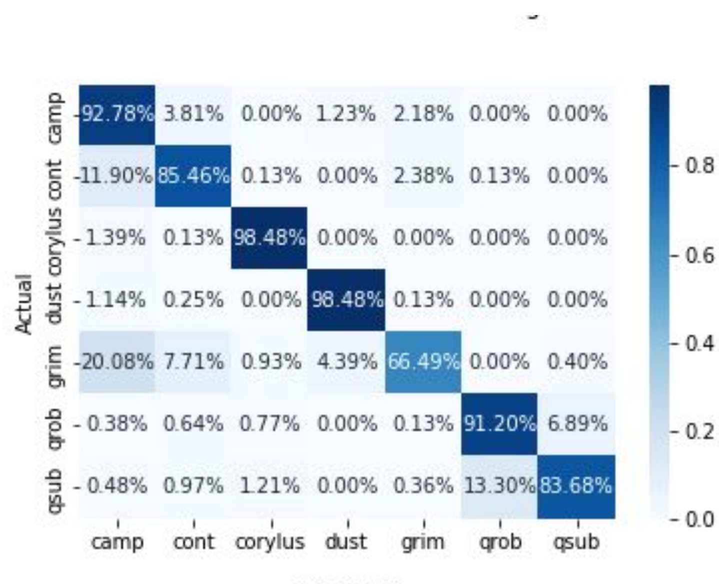


Appendix: CNN experiment: No batch norm / batch norm

86.42% accuracy

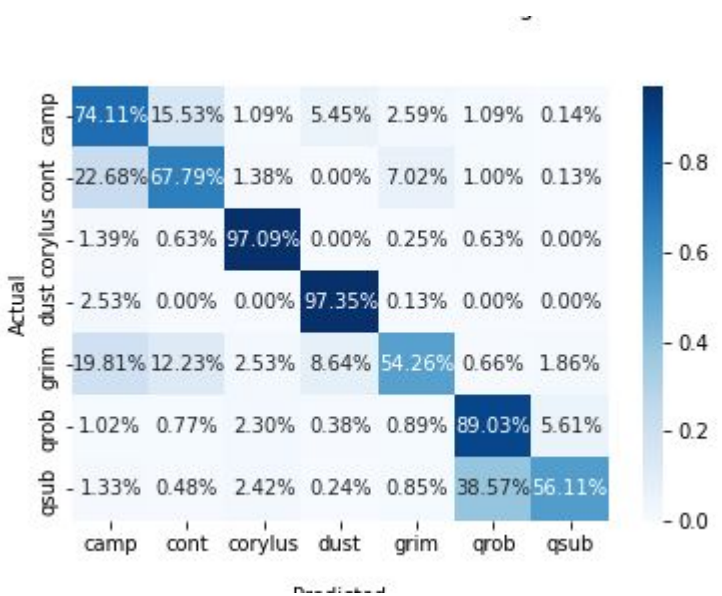


88.15% accuracy on testset

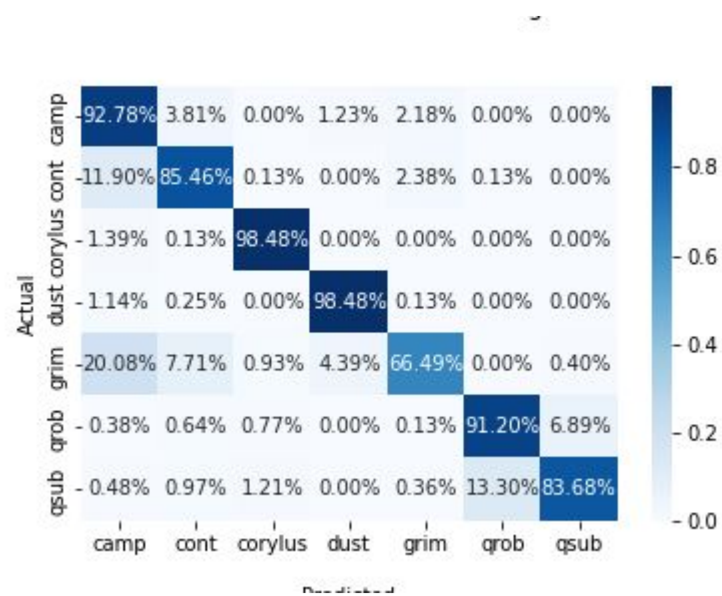


Appendix: CNN experiment: Only CNN / CNN + metadata

76.56% accuracy on testset

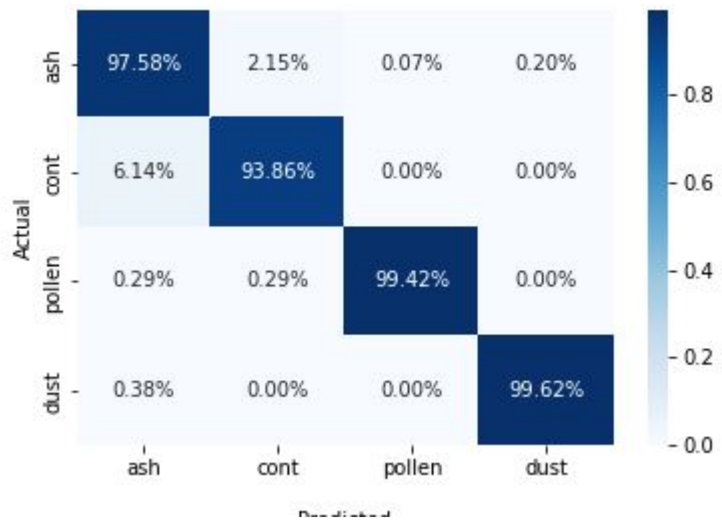


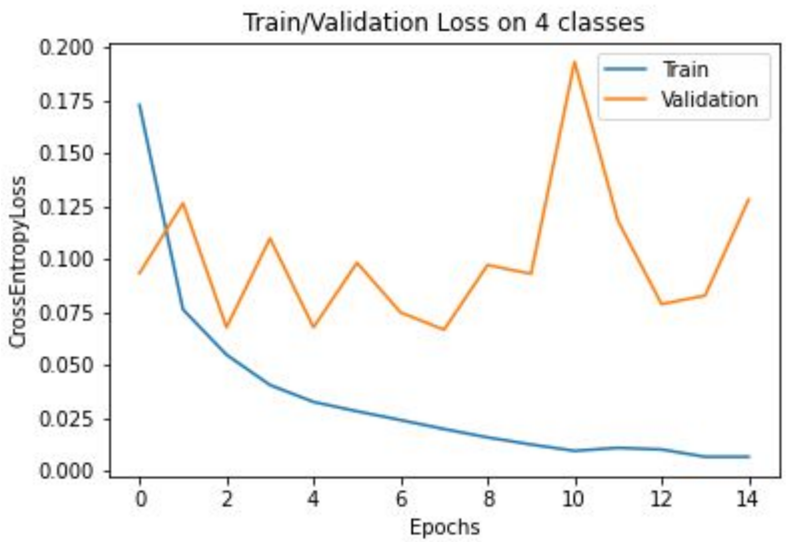
88.15% accuracy on testset



Appendix: Model results for outlier detection with ResNet

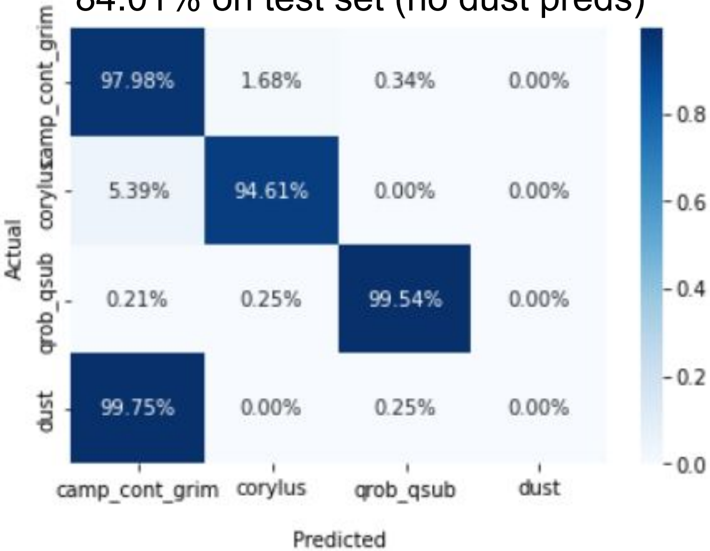
98.45% accuracy on testset

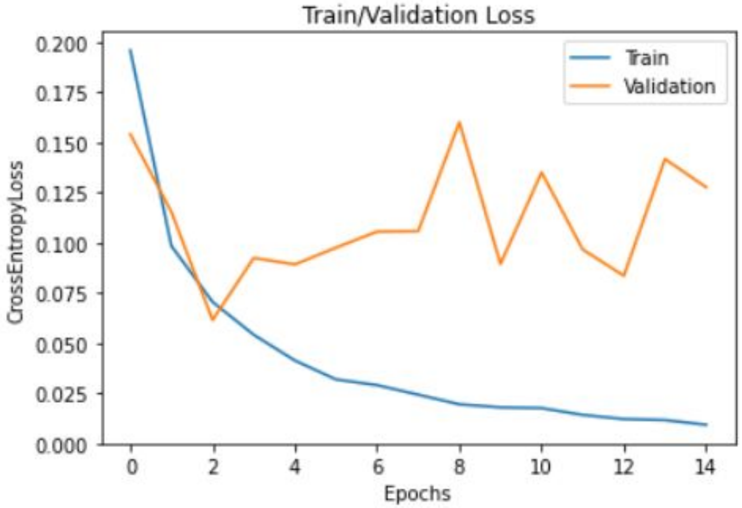




Appendix: Model results for outlier detection with ResNet (4 classes, trained without dust, predicted with dust)

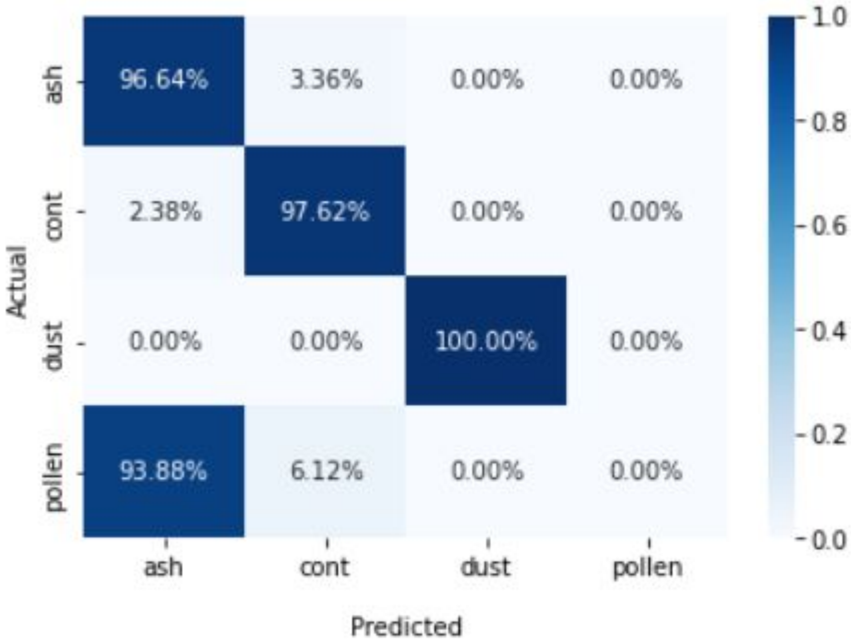
84.01% on test set (no dust preds)

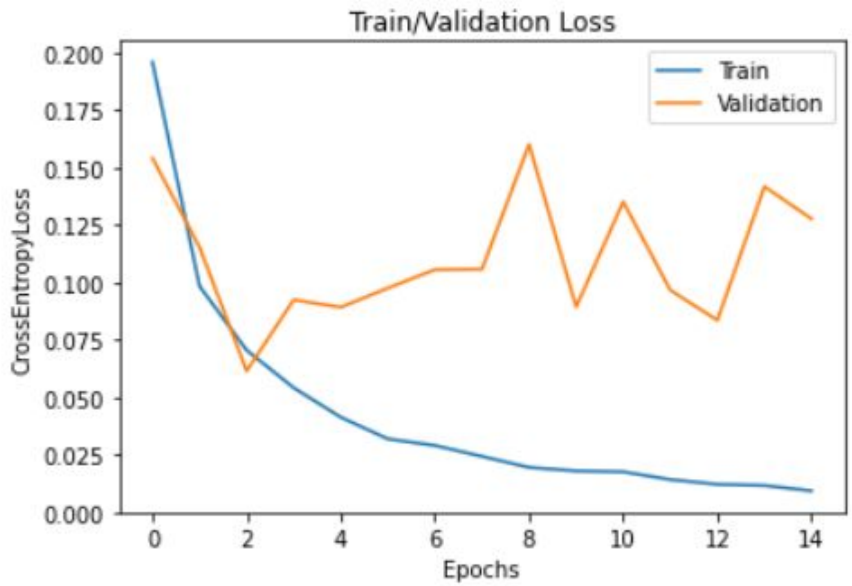




Appendix: Model results for outlier detection with ResNet (4 classes, trained without pollen, predicted with pollen)

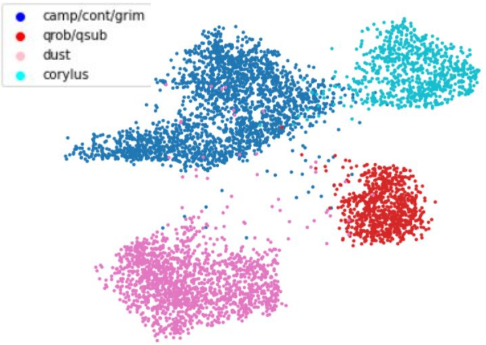
No pollen in training

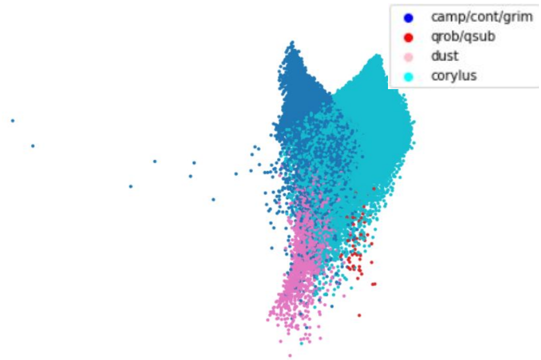


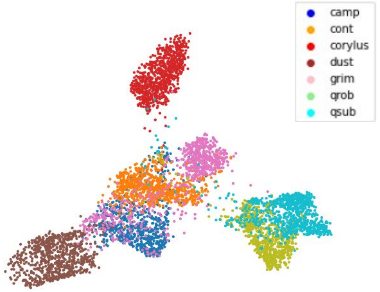


Appendix: Experiment: Outlier Detection: own cnn trained on 7 classes

- Divide dataset into the 4 clusters (because bottom left picture creates 4 clusters-ish)
- Train new model on these 4 clusters
- UMAP again (top right)
- Use embedding on unknown data (middle right)
- Use Local Outlier Factor from SKLearn to get outliers
- Conclusion: This method only found images looking like dust







Particle ID	imgpaths
330	/home/nico/Desktop/MarieCurie/Flowcam/test/QCY/QCY_22_5_1_330.png
114	/home/nico/Desktop/MarieCurie/Flowcam/test/QCY/QCY_22_5_5_114.png
236	/home/nico/Desktop/MarieCurie/Flowcam/test/QCY/QCY_22_5_5_236.png
298	/home/nico/Desktop/MarieCurie/Flowcam/test/QCY/QCY_23_3_1_298.png
790	/home/nico/Desktop/MarieCurie/Flowcam/test/QCY/QCY_23_3_3_790.png
286	/home/nico/Desktop/MarieCurie/Flowcam/test/QCY/QCY_25_7_3_286.png
665	/home/nico/Desktop/MarieCurie/Flowcam/test/QCY/QCY_25_7_3_665.png
377	/home/nico/Desktop/MarieCurie/Flowcam/test/QCY/QCY_27_2_1_377.png

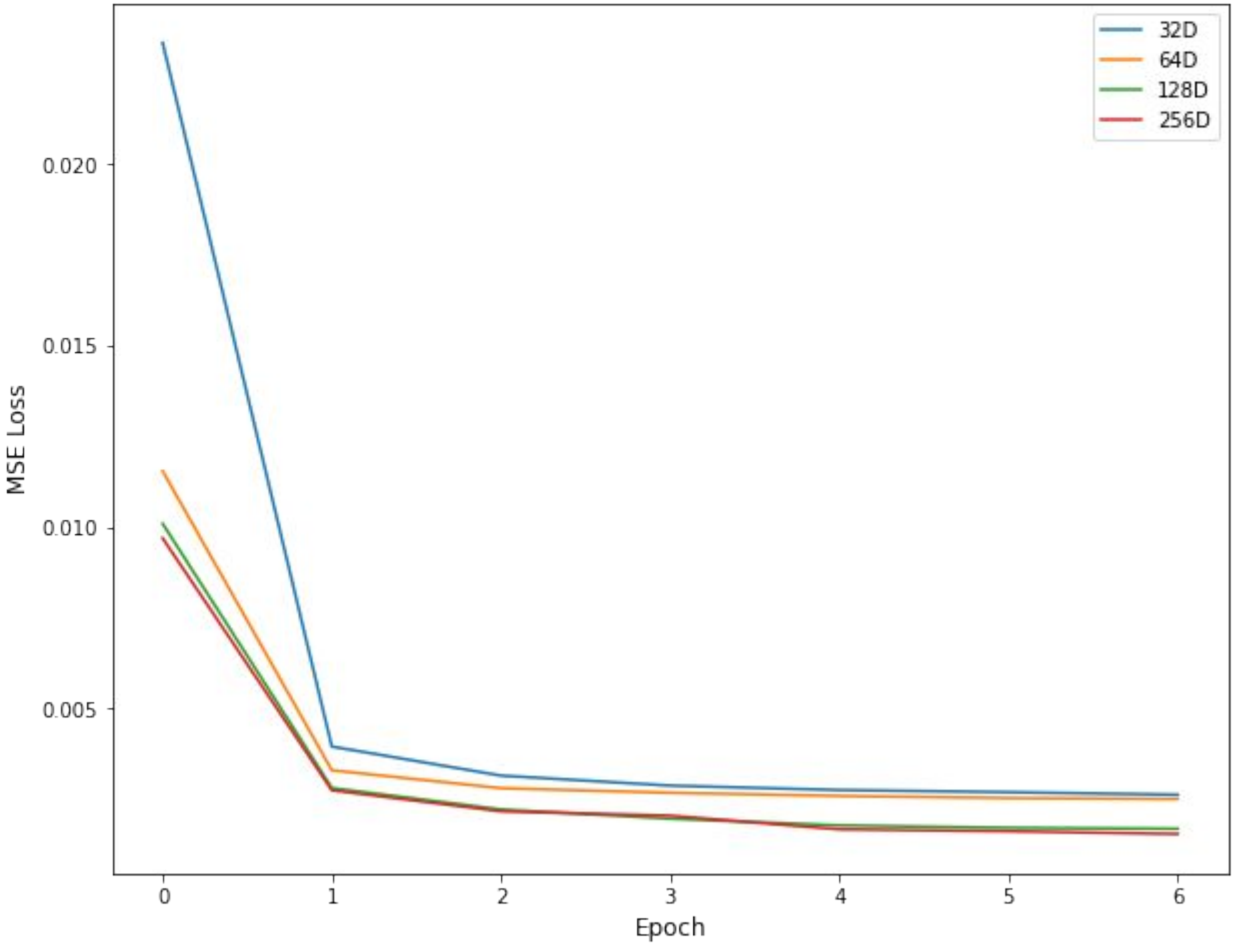
Appendix

Autoencoder model summary

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1           [-1, 32, 62, 62]             832
              ReLU-2           [-1, 32, 62, 62]               0
            Conv2d-3           [-1, 64, 29, 29]          51,264
              ReLU-4           [-1, 64, 29, 29]               0
            Conv2d-5          [-1, 128, 14, 14]          73,856
              ReLU-6          [-1, 128, 14, 14]               0
           Flatten-7                [-1, 25088]               0
            Linear-8                   [-1, 64]       1,605,696
            Linear-9                [-1, 25088]       1,630,720
             ReLU-10                [-1, 25088]               0
        Unflatten-11          [-1, 128, 14, 14]               0
  ConvTranspose2d-12           [-1, 64, 29, 29]          73,792
             ReLU-13           [-1, 64, 29, 29]               0
  ConvTranspose2d-14           [-1, 32, 61, 61]          51,232
             ReLU-15           [-1, 32, 61, 61]               0
  ConvTranspose2d-16          [-1, 1, 125, 125]             801
================================================================
Total params: 3,488,193
Trainable params: 3,488,193
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.06
Forward/backward pass size (MB): 6.60
Params size (MB): 13.31
Estimated Total Size (MB): 19.97
----------------------------------------------------------------
```

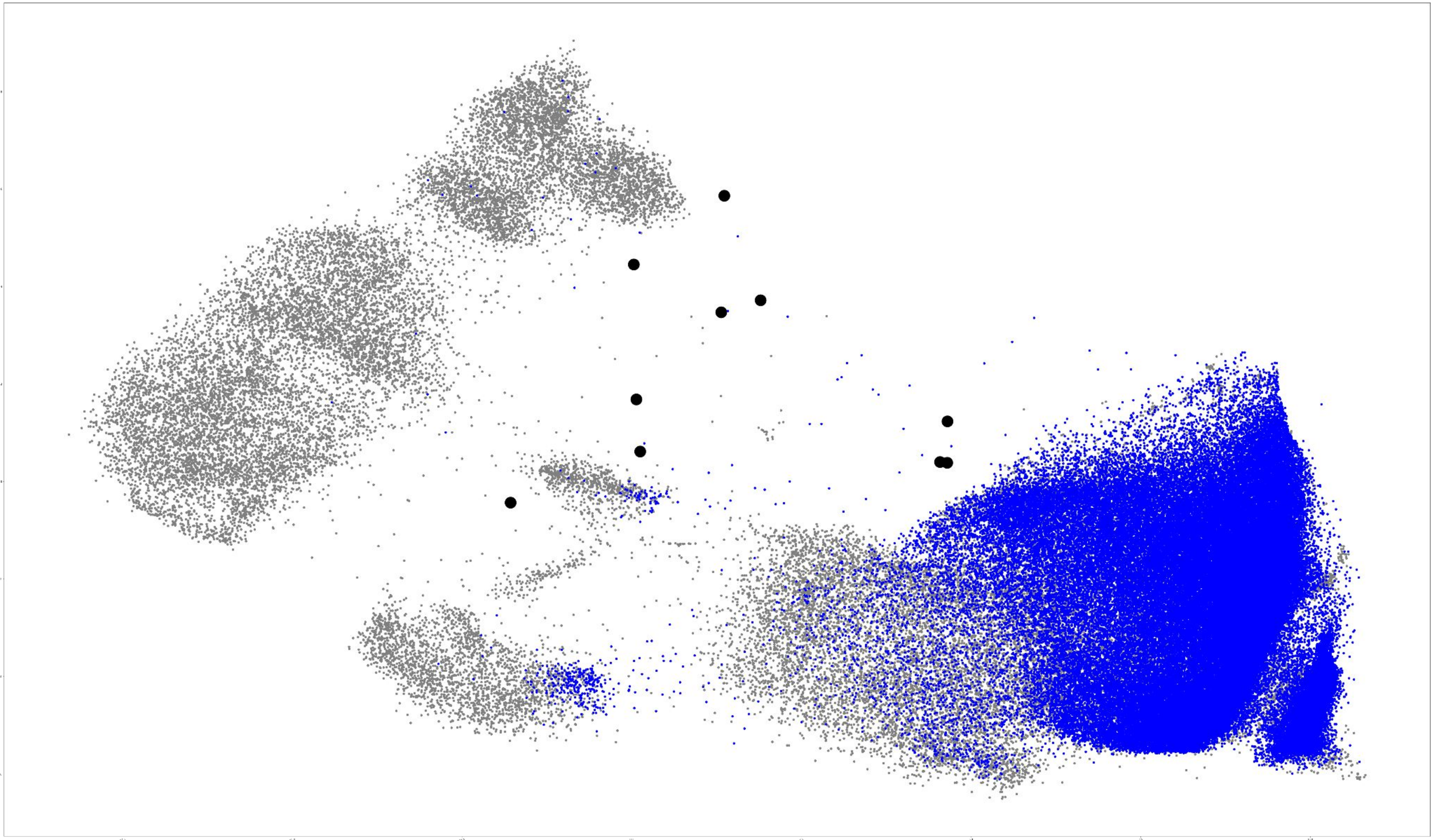
Appendix

Autoencoder train loss

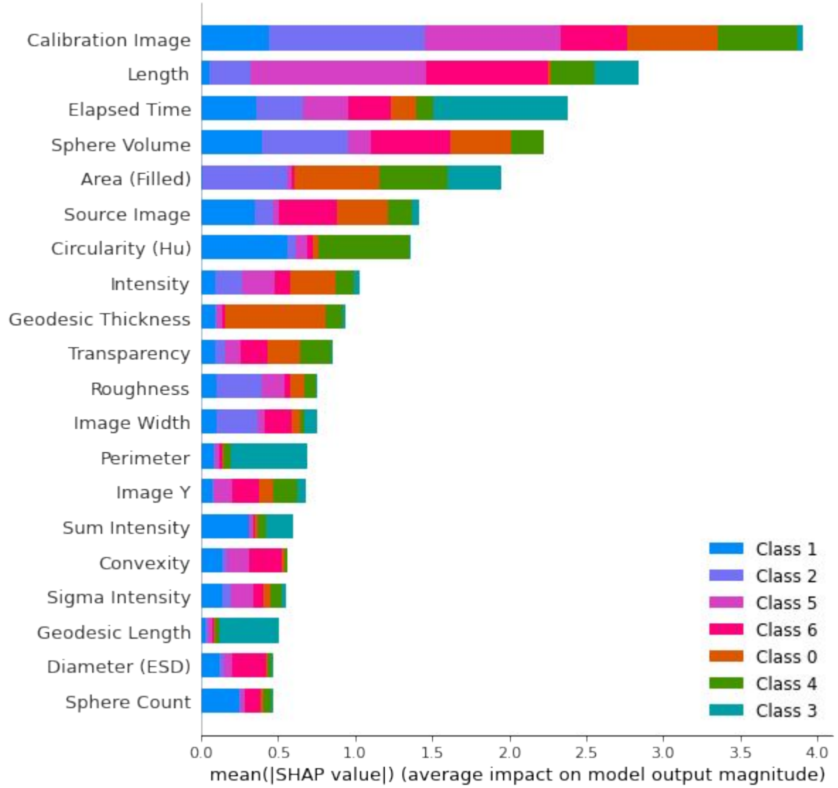


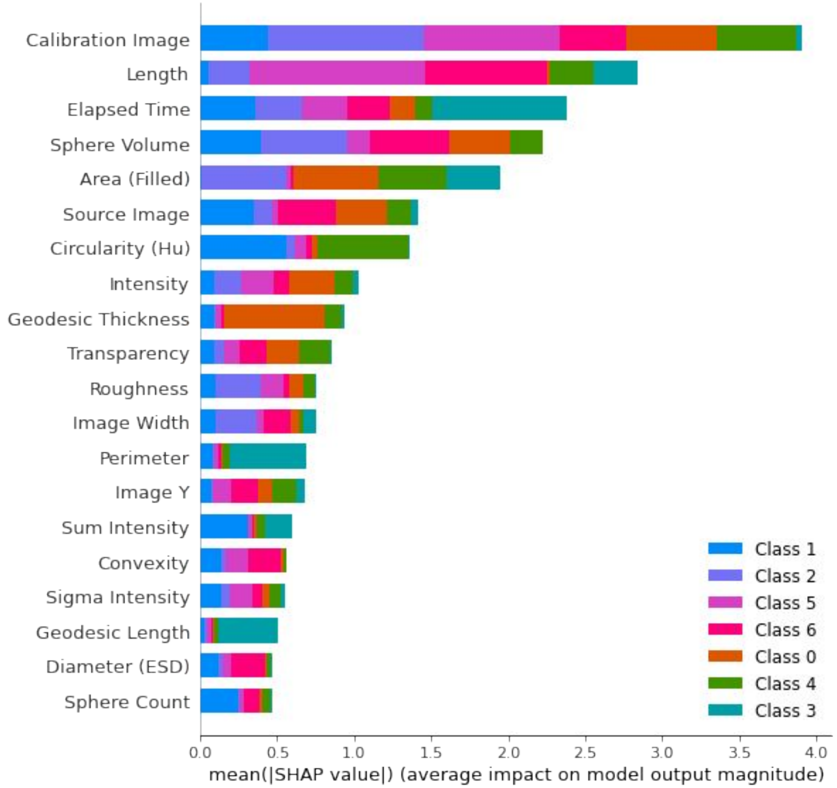
Appendix

The 10 outliers for the peruvian data using Local Outlier Factor



Appendix: SHAP values - LightGMB





Appendix
Isolation forest: Top 10 outliers

```
ID: 31, name: QCY_27_2_1_31.png
ID: 626, name: QCY_23_3_4_626.png
ID: 4, name: QCY_25_6_1_4.png
ID: 19, name: QCY_23_3_3_19.png
ID: 242, name: QCY_24_3_4_242.png
ID: 29, name: QCY_27_3_5_29.png
ID: 20, name: QCY_23_3_1_20.png
ID: 435, name: QCY_23_3_5_435.png
ID: 42, name: QCY_26_4_5_42.png
ID: 163, name: QCY_23_3_3_163.png
```