



Semi-supervised machine-learning method for analyzing images from the Balloon-borne Ice Cloud particle Imager B-ICI

János Stenszky¹ and Thomas Kuhn¹

¹Luleå University of Technology, Department of Computer Science, Electrical and Space Engineering

Correspondence: János Stenszky (janos.stenszky@ltu.se)

Abstract.

Machine learning (ML) has emerged as a promising approach for image analysis, particularly for applications involving specialized or niche datasets. Different imaging techniques result in varying image characteristics and quality. ML techniques that have been previously employed to analyze ice crystals had to be developed for each instrument and imaging technique.

5 The Balloon-borne Ice Cloud particle Imager (B-ICI) collects and images ice particles on a film strip as it ascends through the atmosphere, producing images of ice particles on a background revealing features of the film and its oil coating. To process these raw data obtained from B-ICI, two distinct ML models were developed and trained. The first is a segmentation model, designed to identify ice crystals while filtering out other objects on the image. The second is a shape classification model, which assigns the detected particles to one of four predefined categories. An intermediate step between these two models
10 enables users to visually inspect and, if necessary, correct the segmentation output. This approach facilitates the fine-tuning of the segmentation, allowing for adjustments to model parameters as needed.

In this paper, we present these trained models and explain the validation efforts undertaken to demonstrate their applicability for analyzing data obtained from B-ICI. Comparing the model's performance with manually analyzed new data quantifies the performance of the model. The results of this comparison shows good agreement, in particular for $\gtrsim 50\mu\text{m}$.

15 1 Introduction

Measuring cloud properties provides valuable insight into both large scale climate-effect and local processes occurring within clouds. This can be accomplished through in-situ imaging techniques. This measurement of cloud particles offers the advantage of directly extracting microphysical properties, such as particle size and shape. The advantage of this measurement approach is the ability to obtain particle-by-particle data.

20 Traditionally, image analysis has been performed manually—a process that is both time-consuming and labor-intensive. To accelerate and improve data analysis, techniques that rely more on computational methods and automation are necessary. With recent advancements in computer science, machine learning (ML) approaches have been applied to extract particle information from cloud measurements. Over the past decade several studies have shown that ML can be applied to particle-detection and classification in the field of atmospheric science.



25 Xiao et al. (2019) developed a custom model to classify cloud particles into 11 distinct categories, using data collected
by a Cloud Particle Imager (CPI) mounted on an aircraft. Building on this work, Wu et al. (2020) created a model to detect
and classify particles from their own measurements, recorded using a commercially available Cloud Imaging Probe from
Droplet Measurement Technologies (DMT). Jaffeux et al. (2022) trained machine learning models to identify and classify ice
crystals imaged by two different optical array probes. Their results demonstrated that, with sufficient training data, machine
30 learning models can achieve reliable classification performance. Touloupas et al. (2020) investigated the classification of cloud
particles using data obtained from stationary holographic imaging probes. In Antarctica, a flatbed scanner system was installed
to monitor precipitation. Using data from this setup, Guasta (2022) developed a neural network for the automated analysis of
snow crystals, structuring the process into two sequential steps: image segmentation followed by classification of the detected
particles. Axebrink (2021) developed a neural network, as part of a master's thesis, to analyze snow crystals captured by the
35 Dual Ice Crystal Imager (D-ICI) (Kuhn and Vázquez-Martín, 2020). Additionally, Hicks and Notaroš (2019) and Key et al.
(2021) created models to analyze particles recorded by a Multi-Angle Snowflake Camera (Garrett et al., 2012).

Although the use of ML for identifying and classifying ice particles on image data has attracted considerable scientific
interest over the past decade, a universal model applicable across different instruments has yet to be developed. The literature
demonstrates a variety of approaches to constructing ML models; however, it is important to recognize that the performance
40 and applicability of these models are highly dependent on the source of the crystal imagery.

There are several fundamentally different methods for sampling and imaging ice particles. One approach involves collecting
ice particles and capturing images of them aboard an aircraft (Hobbs et al., 1973) or using a slow-ascent balloon platform
(Kuhn and Heymsfield, 2016). Ramelli et al. (2020) employed a holographic ice particle imager on a tethered, while Chambers
et al. (2024) utilized a similar holographic imaging system on an untethered balloon. Instruments such as the Cloud Particle
45 Imager (CPI), which are typically mounted on aircraft, also provide detailed imagery of individual particles (Connolly et al.,
2007).

Data processing and consequent development of ML algorithm has to be done for that specific dataset. For example, Wolf
et al. (2019) utilized data obtained from a balloon-borne instrument, while Grulich et al. (2021) employed a combination of
aircraft and balloon-based measurements. Xiao et al. (2019) and Wu et al. (2020) used airborne cloud particle imagers mounted
50 on aircraft, whereas Touloupas et al. (2020) based their work on data from a stationary holographic imager. At present, the
variation in image quality between these instruments is significant enough that each setup requires a dedicated ML model
trained specifically on its own data.

In this paper the process is described how to implement these ML models. In chapter 2 the source of the data is explained,
with the description of the resulting images from the B-ICI instrument. In chapter 3 we give a brief introduction of machine
55 learning applied on image analysis, the description of the training process of the segmentation and the classification model. In
chapter 4 the results are shown in a yet-to be processed measurement dataset.

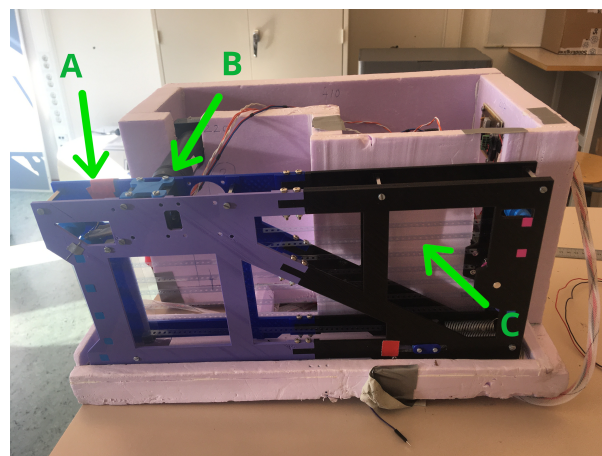
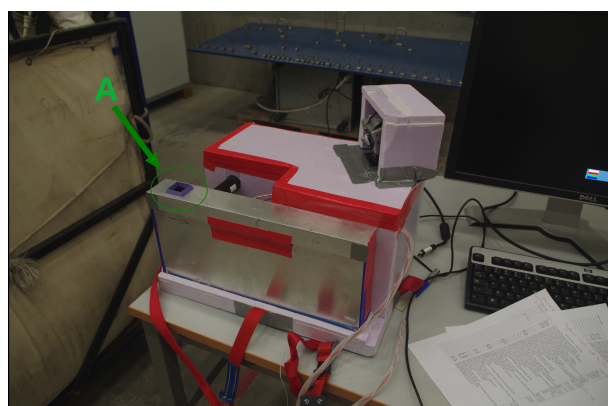


Figure 1. B-ICI in its current state. The location of the inlet (A), the camera (B) and the routing of the collecting film strip (C) can be seen in the figure.

2 B-ICI

2.1 The instrument

The data for this study was collected using the Balloon-borne Ice Crystal Imager (B-ICI), an instrument developed at Luleå University of Technology, described by Kuhn and Heymsfield (2016). B-ICI is equipped with a magnifying lens and a digital camera, capturing high-resolution images of cloud particles as it ascends through the troposphere. The CCD sensor has a resolution of 1280×960 pixels, and the current system achieves an optical resolution of approximately $4 \mu\text{m}$ Kuhn and Heymsfield (2016).

The instrument features an inlet (approximately 31 mm long), as shown in Figure 1, through which it collects cloud particles. Once particles enter the inlet, they are deposited onto the section of a film strip exposed below the inlet opening. The film is coated with silicone-oil in order to facilitate the collection of the particles. This film moves slowly and continuously within the instrument, with only a small section exposed to the inlet at any given time. The advancing film moves collected particles into the field of view of the camera where images are taken every second. The short distance of 31 mm between inlet and camera ensures that particles do not change between deposition and imaging.

This instrument is taking high quality images of ice particles while it ascends through the cloud. It has a collection efficiency of approximately 50% at $12 \mu\text{m}$ and 80% at $25 \mu\text{m}$ Kuhn and Heymsfield (2016). With its pixel resolution of $1.65 \mu\text{m}/\text{pixel}$ particles bigger than $10 \mu\text{m}$ can be detected, and the shape can be determined on particles larger than $20 \mu\text{m}$.

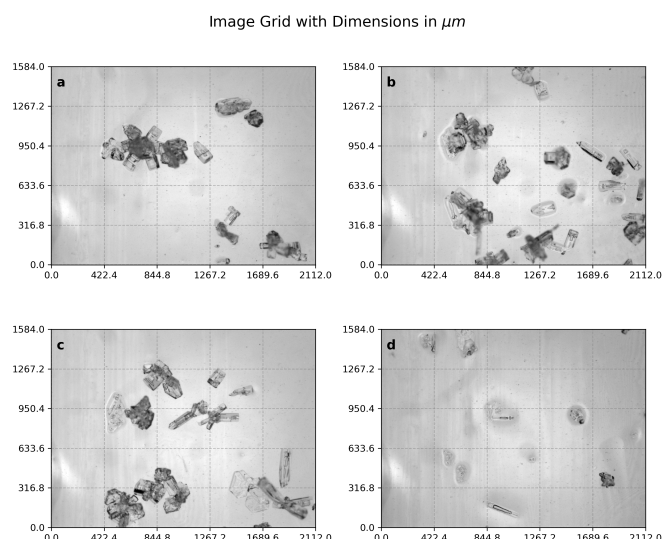


Figure 2. Example images recorded by the B-ICI where the crystals can be easily identified (panels a-d). When the particles are relatively small or the oil coating thicker, the particles sink into the oil, resulting in less clear edges and disturbing contrast around the particles (d).

2.2 Image data

Measurements have been performed on numerous occasions since 2013. Cirrus-cloud regions are identified based on supplemental radiosonde measurements as well as from visual inspection of the data as regions where only ice particles are captured. From these cirrus-cloud regions, 4090 images recorded between 2013 and 2018 have been selected with various numbers of cirrus-cloud particles on it. These images have previously been analyzed manually (Wolf et al., 2018, 2019). and are used here as the training dataset (see Sect. 3.2). Figure 2 presents example images of ice crystals recorded by the instrument.

Due to the instrument's method, it may capture objects that are not ice particles such as dust, fibers, or scratches. Examples of these undesired objects are shown in Figure 3. Such non-particle objects on the images pose challenges for automated image analysis not using ML and result effectively in a non-uniform background. This, combined with changing contrast around particles that sink into the oil coating, makes classical approach of edge detection inefficient (Wolf, 2019). Prior to implementation of ML methods, there was no reliable way to confidently identify and exclude non-ice particles from the analysis.

Even though the background, faults, and extra objects vary in each image, their characteristics remain consistent across the dataset. While not identical, these recurring patterns of variation exhibit enough regularity to allow the development of machine learning models capable of identifying and distinguishing ice crystals from these background artifacts.

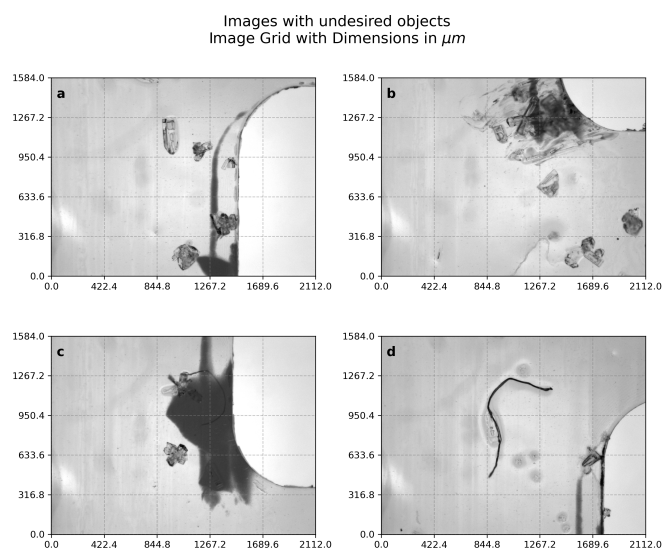


Figure 3. Example B-ICI images with non-particle objects and film defects. One can see imperfections and dirt on the film (panels a, b, c), edges from holes in the film (panels a, b, c, d) and fibers (panel d). Just as with ice crystals, these extra features appear with a variety of patterns, shapes, and sizes; generally however, they are consistently different from ice crystals.

3 Description of the machine learning models

The nature of the problem suggests to separate the image processing into two models, one that segments the images to identify ice particles and the other that classifies the ice particles. A segmentation model has been developed to generate binary mask images, showing where the ice particles are, suitable for confident edge detection. This model analyzes the images and identifies patterns based on the training dataset. It detects ice particles while disregarding other irrelevant features on the images. A classification model has also been created in order to classify the particles according to their shape. This model evaluates each individual particle and assigns it to one of a number of predefined shape categories, as established in the training dataset. Both models have been specifically developed to analyze data from the B-ICI instrument, as previously described.

3.1 Overview of ML model training

Both models use a Convolutional Neural Network (CNN), which is a type of ML that automatically learns the features and patterns on a training dataset. It is using so-called convolutional layers that are detecting local features of an input image.

To facilitate efficient processing, images are resized, allowing the computer to analyze them more effectively. A reduced image size increases the processing speed and therefore reduces the computational capacity at the expense of reducing the



details of the features. To extract these features, a so-called kernel size is used that is defined as a group of neighboring pixels. The smaller this kernel, the faster the process is with the trade-off that it can overlook more features. The main challenge of the training process is to find balanced image and kernel size, where the training can be the fastest without losing information.

Upon receiving an input image, the CNN applies a series of filtering layers to extract relevant features. With filtering one
105 can extract edges of features from an image based on the statistical specification of the training dataset.

The output of each convolutional layer is a two-dimensional matrix representing the activated groups of pixels, commonly referred to as a feature map. The CNN processes this matrix by evaluating each pixel along with its neighboring pixels, assigning values to local pixel groups.

This operation is performed sequentially, scanning from left to right and top to bottom. The resulting feature map is then
110 passed on as input to the subsequent layer. This process continues through multiple layers, with each layer reducing the spatial dimensions and simplifying the representation. The number of layers is tested for the best performance on the dataset.

The output of the model corresponds to the number of desired classes. In the case of segmentation, the output consists of two classes: 1 for particle and 0 for non-particle. For classification, the output consists of four classes, one for each predefined category. With each successive layer, the data is abstracted further until the network produces a final output.

115 Training the model involves using a labeled dataset in which the correct outputs are known – in this case, binary images containing the silhouettes of previously identified particles. The dataset is divided into an 80/20 split: 80% of the data is used for training, while the remaining 20% is reserved for validation.

The source images can be augmented in order to increase the population that can be used in the training process. Image augmentation techniques included horizontal flipping, rotation, zooming, translation, and adjustments to contrast and brightness.
120 With this technique one can generate additional training data. The more images there is the more accurate the model will be.

One complete pass of all training-images through the aforementioned process is referred to as an epoch. At the end of each epoch, the model's performance is evaluated on the validation set by comparing the predicted results with the actual truth. This evaluation is typically conducted using standard performance metrics, *accuracy*, *precision*, *recall* and *loss*.

The unit that the model is handling here referred as data-point. It is a pixel in case of the segmentation model and a particle
125 for the classification model. Each data-point is assigned a category, particle or non-particle for the segmentation and one of the four shapes for the classification. Consequently, all predictions are divided into the four groups: true positives (data-points that have been correctly predicted, or *TP*); true negatives (correctly identified as not belonging to the category, or *TN*); false positives (wrongly predicted data-points, or *FP*); false negatives (wrongly predicted as not belonging to the category, or *FN*).

Accuracy is defined as the ratio of the correct predictions to all predictions:

$$130 \quad Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

Precision tells the reader how much of the data is classified correctly. It is the ratio of correctly classified actual positives to all actual positives, specifies how many of the data-point predictions are actually true. Thus, it is the ratio of correctly predicted data-points to all predicted data-points,

$$Precision = \frac{TP}{TP + FP}. \quad (2)$$



135 *Recall* measures that how many of the data-points have not missed identification. It is defined as the correctly classified true positives to all that classified as positives, specifies how many of the actual particles have been predicted correctly. It is the ratio of correctly predicted data-points to all actual data-points,

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

The *loss* function is described for the segmentation and classification model separately. In summary, a higher *loss* value
140 indicates poorer model performance. For the segmentation model, it corresponds to fewer correctly identified objects, while for the classification model, it reflects a higher rate of crystal shape misclassification.

The training of CNN process is executed over multiple epochs. The number of these epochs can either be predefined or a stop criterion can be applied to terminate the process. The criterion is based on comparing performance parameters over a certain number of epochs, the so-called patience interval. If performance does not improve within the patience interval the process is
145 stopped. It is called early stopping.

The ML models were implemented using TensorFlow, an open-source platform designed for developing machine learning algorithms Abadi et al. (2016). TensorFlow offers a flexible framework for constructing and training neural networks. It supports the use of pre-trained models—such as ResNet, VGGNet, and AlexNet—in a modular, "plug-and-play" manner. Several other studies have evaluated and used these pre-trained models for ice-crystal detection and classification. Xiao et al. (2019) found
150 that Resnet152 works the best for their dataset. Wu et al. (2020) compared 8 models, with their finding that EfficientNet-B6 has the best performance. Zhang et al. (2024) used Resnet50 for their rotated object detection model. These different pre-trained models do not replace the training process on a specific dataset; rather, they serve as a foundation into which data can be integrated and training parameters can be adjusted. A wide variety of such models are available, each differing in how they perform pixel-wise calculations, handle residuals, and manage internal data flows. Through repetitive testing, the best performing
155 pre-trained model on the B-ICI data was found to be EfficientNet-B7. This conclusion is consistent with the results reported by Tan and Le (2019), although their work was not focused on atmospheric science and cloud particles. Both the segmentation and classification model presented in this paper are using the EfficientNet-B7 pre-trained model.

3.2 Training of the segmentation model

B-ICI mages previously obtained between 2013 and 2018 and analyzed manually have been used to train the initial model. All
160 together there are 4090 images with various amountof crystals detected on it. Examples of the original and the corresponding mask-images indicating where particles were identified manually are shown in Figure 4.

For the segmentation the Tversky-loss function was used to measure the loss. It allows the ratio of precision and recall to be adjusted in order to accommodate the dataset (Salehi et al., 2017) This function can be defined as,

$$Tl = \frac{TP}{TP + FN + \beta FP}, \quad (4)$$

165 where β is a weighing factor which can be used to fine-tune the performance of FN and FP . Manipulating β allows the user to influence the fraction of FP values.

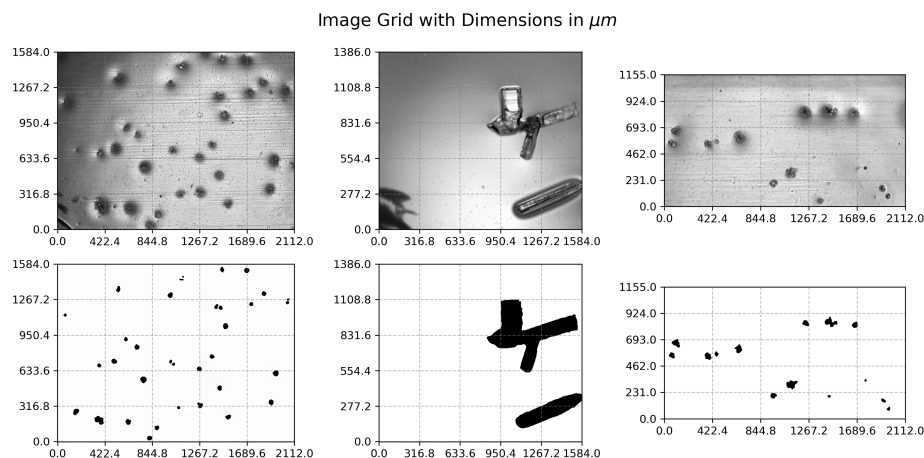


Figure 4. Examples of original and the corresponding masked image. The mask-image is created by hand, identifying the ice-crystals on the image visually.

Equation 4 is used for balancing the image segmentation to find the accuracy that needed for the dataset Basically the model can be adjusted to detect more or fewer objects, depending on the individual cases. After the initial training, 3 different models were created based on 3 different β values:

1. A strict with high β model that identifies less objects,
2. A medium model where the sensitivity is balanced
3. A loose model with low β that identifies more objects.

TensorFlow provides support for pre-trained CNN models, which can significantly accelerate the training process on new datasets. Metrics for the initial segmentation model training are presented in Figure 5. The training goes for 50 epochs with early stopping implemented. It can be seen that *precision* and *recall* fluctuate around 70 – 80% and 80 – 85%, respectively. The *loss* values fluctuate around 30%.

Once the model is trained in the original training dataset, it can be evaluated on new data that was not part of the training dataset. Doing so it produces a mask image of this fresh data, which can be visually inspected and accepted if the segmentation performance is satisfactory. These mask images produced by the model are similar to the manually analyzed images shown in Figure 4. Since unprocessed data from 2018–2023 are available, these can be used to gradually evaluate the various models with different β values and to augment the original dataset of 4090 images with the accepted results, extending the training dataset with each step on the way. This process is shown in Figure 6.

After applying all three models to the new dataset and selecting the correct original-mask image pairs, the accepted images can be incorporated into the training dataset. Using this method, new improved versions of the models can be trained, and the process can be repeated on images where the model previously performed poorly.

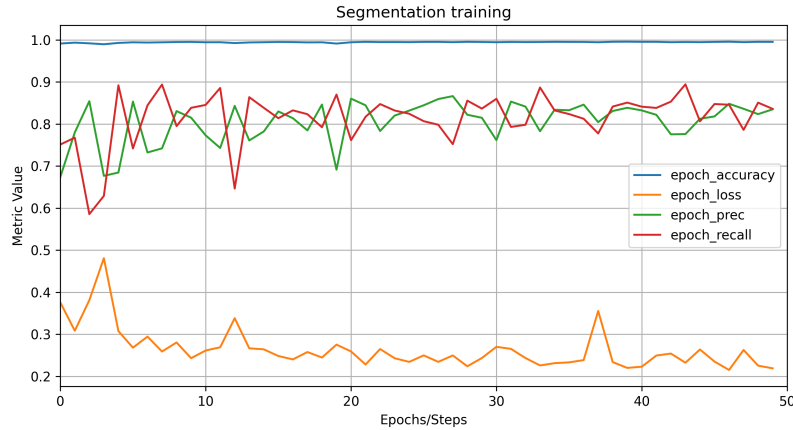


Figure 5. Training of the initial segmentation model on the previously analyzed dataset.

This semi-supervised learning approach allows systematic errors—arising from limitations in the instrument—to be filtered out by user intervention. The user decides whether to accept the generated segmentation and add it to the training set. If too few or too many particle edges are detected, a stricter or more permissive model can be selected to find one that identifies all the particles on the image.

190 Once a satisfactory mask image is obtained, the original image and the corresponding mask are added to the training dataset, extending it for the next generation of model training. Moreover, this method improves the model incrementally without the need for the tedious manual identification of each particle.

The images of the individual particles found by this segmentation are used as input in the classification model on the next step, while the dimensions of the identified objects are collected to a database.

195 3.3 Training of the classification model

Similarly to the segmentation model, classification of the particles has been done by a model trained specifically for classifying the individual particles. This model is using the ice crystals identified by the segmentation model as an input.

In this study, to enable comparison with previous measurements of the same instrument as described by Wolf et al. (2018), four categories were selected: columns, compacts, rosettes, and plates. Examples of these categories are shown in Figure 7.

200 The *Loss* function is used in the classification model is called sparse categorical cross-entropy. it is defined as,

$$Loss = - \sum_{i=1}^C (y_{true} \log y_{pred}), \quad (5)$$

where C is the number of possible categories and y is the category of each particle. This equation the logarithm ensures that misclassified values have a high penalty (Mao et al., 2023).

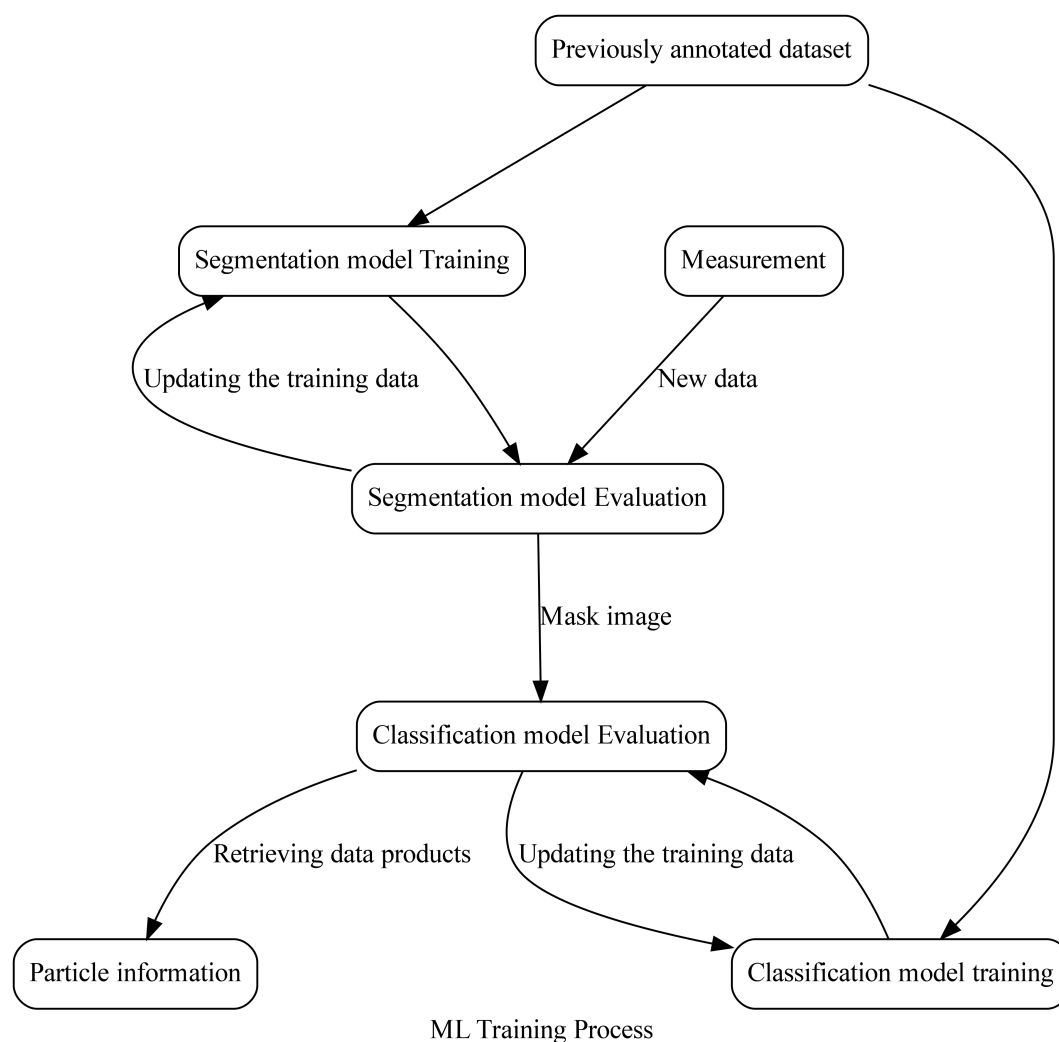


Figure 6. Flowchart of the training process and data processing. After the initial training of the model, it is evaluated on new data. Following visual inspection of the evaluation results, the accepted images are fed back into the training process to create a new iteration of the model, thereby improving the overall accuracy and facilitating the processing of additional data.

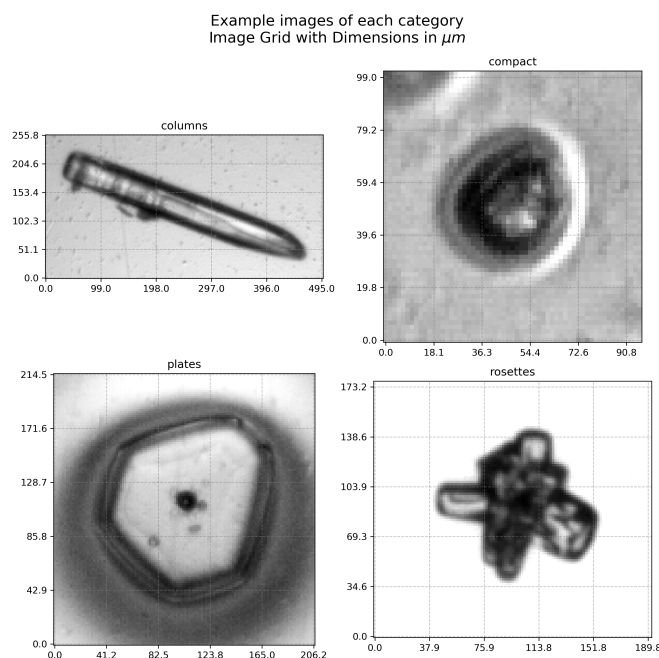


Figure 7. Sample image from the image categories. This shows the main characteristics of these four categories. Columns, compacts, plates and rosettes has easily distinguished characteristics.

Approximately 250 images were available for each category for training. An 80/20 split was applied between the training and validation datasets.

The classification model evaluates individual particles and assigns the probability that each particle belongs to a given category. Similar to the segmentation model, the classification model also utilizes the pretrained EfficientNet-B7 architecture. Training was conducted over 20 epochs with early stopping implemented, resulting in high confidence in the model's performance, as illustrated in Figure 8. After around 10 epoch the *accuracy* of the training plateaus out, and then the model gets over-fitted, meaning that the training reached its best state and additional training only makes it worse. At the same time it can be seen that the validation *loss* value reached its minimum. In this case, the best performing epoch based on validation *loss* is epoch number 6. After that epoch, the training continued for another 10 epochs, the value set as patience interval. Since the performance did not improve in this interval, the weights from epoch nr 6 is loaded and saved for the model.

In addition to standard evaluation metrics such as accuracy and precision, a confusion matrix provides insight into the model's performance in classifying individual objects. It is generated from the model's predictions (on the validation dataset) by comparing the predicted categories with the reference (ground truth) categories. The resulting confusion matrix is shown in Figure 9. It highlights the general tendencies of the model, indicating which categories are classified with high confidence and which are more prone to misclassification. It shows a strong agreement within the predicted and reference categories, with the lowest confidence of 85%. Besides the agreements, it also tells the reader that what categories are the most likely to be

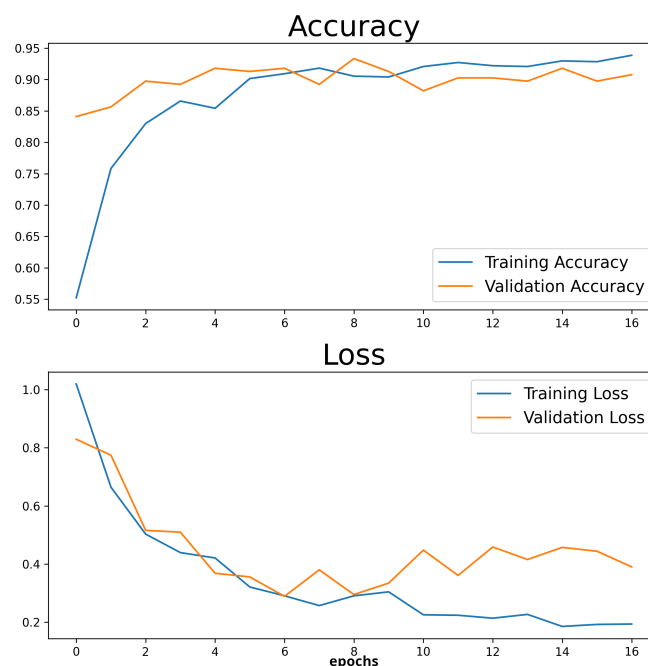


Figure 8. Training from the classification model. The upper figure shows the accuracy while the bottom shows the loss values throughout the iterations.

220 mixed, based on the training-dataset. For example, 8% of rosettes are miscategorized as compacts as well as 5% of plates were identified as compacts as well. This shows us that this model has a tendency to show a higher ratio of compacts than someone visually evaluating would do.

This classification model can be continuously extended with newly labeled data to increase the volume of training data for model development. Additionally, if new particle categories are to be included, the model can be expanded and retrained, 225 assuming a roughly equal number of training examples is available for each category. The model's performance degrades significantly when images from other instruments are introduced, due to variations in image quality across instruments. Subtle visual differences—such as edge sharpness—have a substantial impact on classification performance. For this reason, only particles imaged by the B-ICI instrument were used for training, despite the availability of several open-access databases. This necessitated the development of a dedicated classification model tailored to this specific instrument, which is presented in this 230 paper.

The resulting models are sequentially used on the new data, as it is shown in Figure 6. First the segmentation model is run on the original images, then the classification model categorizing each individual particles larger than 20 μ m, the size limit of B-ICI.

On the detected and categorized particle the *maximum dimension* is measured to approximate the size of the particles. It 235 is defined as the diameter of the smallest circle that encloses the particle.

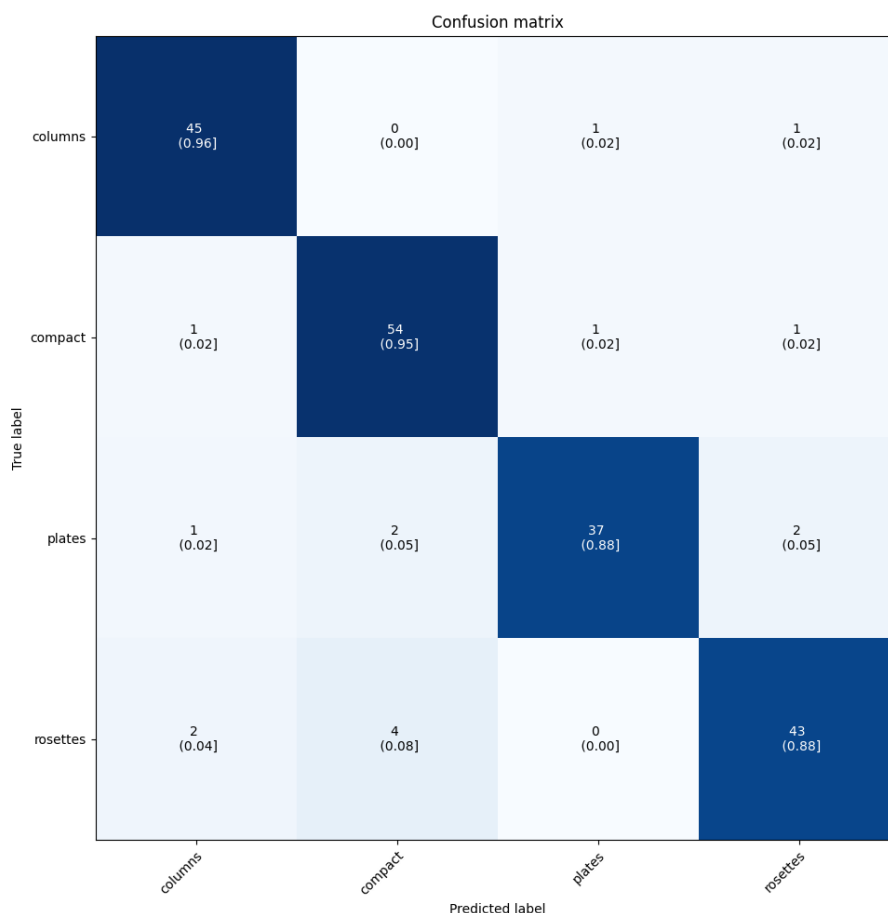


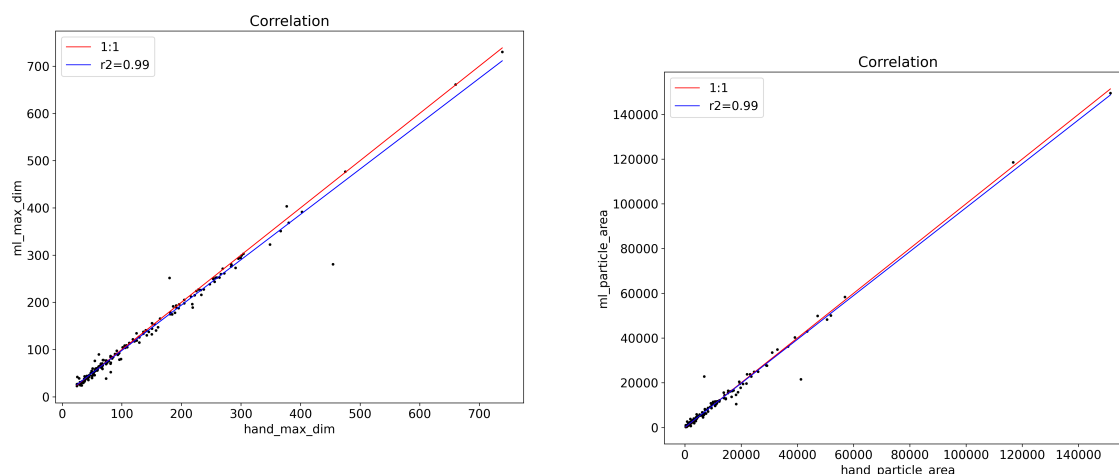
Figure 9. Confusion matrix of the classification model. The resulting output is compared to the reference dataset to see how the model performs.

Class category is produced by the classification model. The model gives a probability score for each category showing the probability that the given particle belongs to that category. The category with the highest score is assigned to the particle.

4 Model Performance

Direct comparison of ML with manual analysis allows us to evaluate the performance of the model on unprocessed data. In this section the output of the segmentation and the classification model is compared with a hand-analyzed dataset.

For this purpose, measurements from a campaign on 2024-03-20 were analyzed both manually and using the machine learning model. Altogether around 200 particles were identified in this measurement. This dataset has been analyzed manually, and was not included in the training of the models.



(a) Correlation of maximum dimensions, that is used to determine particle size in μm .

(b) Correlation of the particle areas. Values are in μm^2

Figure 10. Correlate the hand and ML analyzed dataset. The red line is a plotted 1:1 ratio.

Individual particles' properties determined through manual analysis or from ML segmentation can be compared using simple correlation analysis. Figure 10 shows a strong correlation between the particle size and area identified manually and those identified by the machine learning model. This suggests that the particles detected by both methods are comparable in terms of size and area.

The size distribution from manually detecting particles and the distribution resulting from ML segmentation are shown in Figure 11. Size distributions are shown as the probability density functions. Log-normal fits are shown for each size distribution. The fitted curves are compared in Figure 11 panels c and d to identify the size ranges where the largest discrepancies occur. The results show good agreement between the two methods for particles larger than approximately $100\mu\text{m}$. The greatest differences are observed in the smaller size ranges. The comparison show that the fit to machine learning size distribution predicts approximately 10% more particles in the $40\text{--}100\mu\text{m}$ range, while predicting about 50% fewer particles in the $20\text{--}40\mu\text{m}$ range. This discrepancy is understandable, as distinguishing ice particles from dust or other contaminants becomes increasingly difficult near the resolution limits of the instrument. In the larger size range (above $100\mu\text{m}$), the observed differences can be attributed to the relatively low number of particles; in such cases, each individual particle has a greater influence on the overall distribution compared to the more populated smaller size ranges.

To characterize the discrepancies introduced by the machine learning model in the size distribution, one can compare the two size distributions directly, as can be seen in Figure 12. Thus, the differences between the ML values of the size distribution and the ground truth are evaluated for each size bin by comparing to the uncertainty bounds of the manually analyzed data. These bounds, indicated as uncertainty ranges in Figure 12, are based on simple statistical sampling error. The results show that the model's size distribution falls within or is very close to the uncertainty bounds of the manually analyzed data. Therefore,

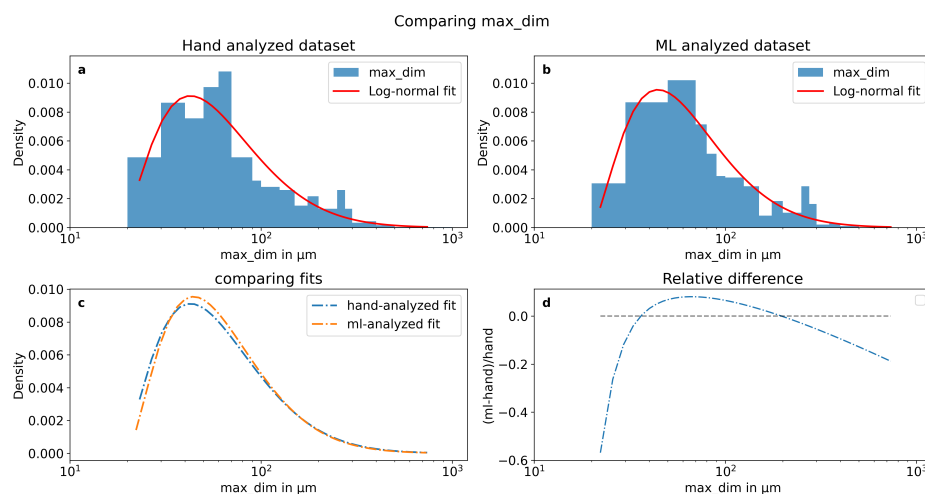


Figure 11. Comparison of size distributions of hand and model analysis of the same dataset (*a* and *b*). Plotting the fits on top of each other on figure (*c*) gives an idea how the 2 dataset compares. On figure *d* relative difference between the two dataset shown, highlighting that the most mismatch can be found in the size-range below $100\mu\text{m}$.

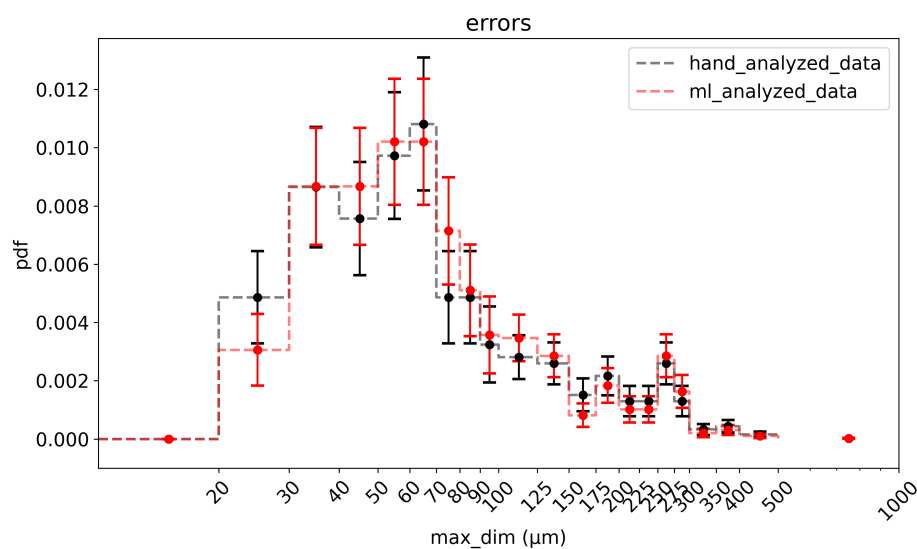


Figure 12. Comparison of size distributions of hand and model analysis. The uncertainty ranges shown are based on the sampling errors.

no significant or systematic error is introduced by the model in all but the smallest sizes. In fact, in this comparison, the only size bin where the model seems to underpredict the size distribution is the $20\text{--}40\mu\text{m}$ range. This could also be seen from the
265 log-normal fits discussed above indicating the model's increasing difficulty near the resolution limit of the instrument.

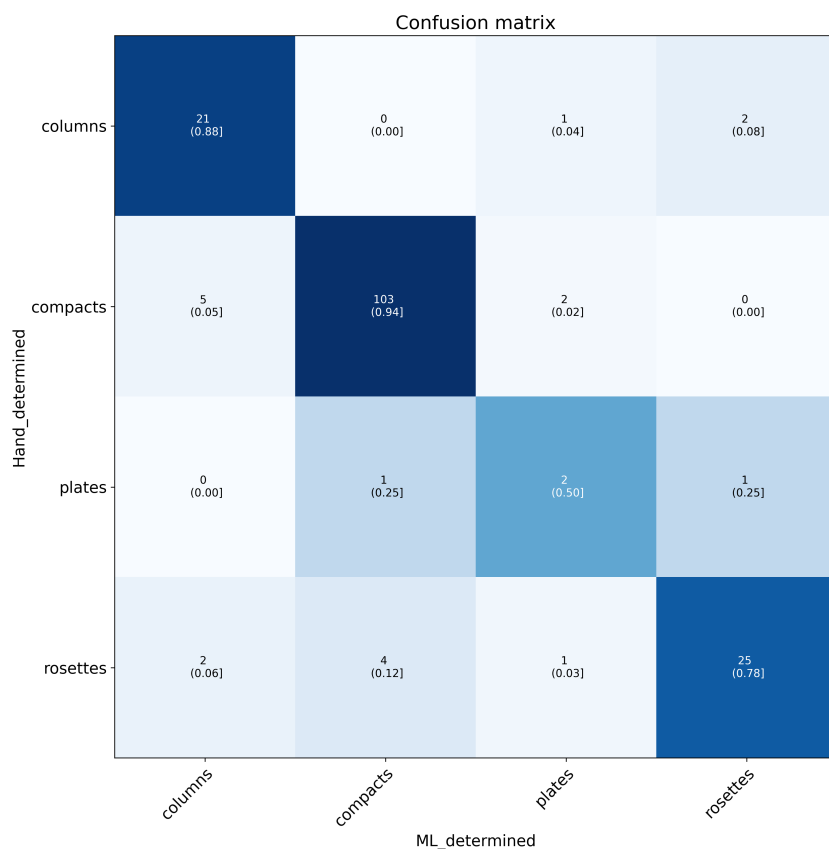


Figure 13. Confusion matrix on the predicted dataset.

Similarly, a comparison has been done with the classification of the particles. Firstly, the shape of each particle has been determined manually by the authors, then the confusion matrix was created that can be seen in Figure 13. The cross-referred categories show strong agreement ($\gtrsim 80\%$) in all but the plate category, with some systematic misclassification. For example, around 10% of rosettes were classified as compacts by ML. It should be noted that these particles are smaller than the average size of those in the rosette category, highlighting the model's sensitivity to particle size in relation to the accuracy of shape classification. Similarly, around 5% of the particles were classified as columns instead of compacts. The most mismatched category is plates, however during this measurement there were only few plates, thus one or two mismatched particles have a large impact on the confusion matrix.



5 Conclusions

275 It has been demonstrated that the ML models developed and trained on images from B-ICI enable processing future new data with a level of automation that was not possible before.

A segmentation model to detect ice particles and a classification model to assign shape categories to the particles have been trained on a dataset that was previously manually analyzed. The resulting models proved to be reliable for the dataset.

280 The segmentation model's output, mask images, is further processed to determine particle properties such as maximum dimension, area, and aspect ratio. Thus, the segmentation model enables automatic particle detection and determination of particle properties. The classification model classifies the shape of the detected particles. Currently particles are classified into four different categories: columns; compacts; plates, and rosettes. A limited number of shape categories that had been chosen in previous studies with B-ICI data have been used here. In the future these can be changed and their number increased depending on the need and the available dataset for training.

285 Comparison of the models' segmentation and classification results with a hand-analyzed dataset—excluded from the training process—provides an intuitive performance assessment in addition to standard evaluation metrics. This comparison highlighted the accuracy, and weakness of the model.

While in general the models performed well, they seem to have a deteriorated performance for smaller particles getting close to $20\mu\text{m}$, the limit of B-ICI for shape recognition. At these sizes it is very difficult to differentiate between particles and 290 other objects on an image. This increasing difficulty with the smallest particles has been observed in both the segmentation and classification models. They are, however, more reliable for particles with sizes $\gtrsim 50\mu\text{m}$.

Therefore, when using these models on datasets containing predominantly smaller particles close to a particle size of $20\mu\text{m}$, one should use caution. The described models have, in particular for $\gtrsim 50\mu\text{m}$, demonstrated good performance and great potential. Through continuous re-training with new B-ICI measurements that are added to the training dataset after having 295 been processed in a semi-supervised way, the general accuracy increases making it easier to automatically process ice crystal particles with each added new measurement.

Code availability. Code will be made available.

Data availability. Data will be made available at the Swedish National Data Service.

Author contributions. JS developed and trained the model. TK collected the data. JS and TK worked on the evaluation of the model on the 300 new data. JS wrote the manuscript and JS and TK worked on the revision of the manuscript.

<https://doi.org/10.5194/egusphere-2025-2818>

Preprint. Discussion started: 7 July 2025

© Author(s) 2025. CC BY 4.0 License.



Competing interests. The authors declare that they have no conflict of interest.

Financial support. This work has been funded by the Swedish National Space Agency (SNSA) - 2020-00122.



References

- 305 Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al.: TensorFlow: A System for Large-Scale Machine Learning, in: 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), pp. 265–283, 2016.
- Axebrink, E.: Automatic Classification of Snow Particles, Master’s thesis, Luleå University of Technology, Department of Computer Science, Electrical and Space Engineering, 2021.
- 310 Chambers, T. E., Reid, I. M., and Hamilton, M.: A lightweight holographic imager for cloud microphysical studies from an untethered balloon, *Atmospheric Measurement Techniques*, 17, 3237–3253, <https://doi.org/10.5194/amt-17-3237-2024>, 2024.
- Connolly, P. J., Flynn, M. J., Ulanowski, Z., Choularton, T. W., Gallagher, M. W., and Bower, K. N.: Calibration of the Cloud Particle Imager Probes Using Calibration Beads and Ice Crystal Analogs: The Depth of Field, *Journal of Atmospheric and Oceanic Technology*, 24, 1860 – 1879, <https://doi.org/10.1175/JTECH2096.1>, 2007.
- 315 Garrett, T., Fallgatter, C., Shkurko, K., and Howlett, D.: Fall speed measurement and high-resolution multi-angle photography of hydrometeors in free fall, *Atmospheric Measurement Techniques*, 5, 2625–2633, <https://doi.org/10.5194/amt-5-2625-2012>, 2012.
- Grulich, L., Weigel, R., Hildebrandt, A., Wand, M., and Spichtinger, P.: Automatic shape detection of ice crystals, *Journal of Computational Science*, 54, 101 429, <https://doi.org/10.1016/J.JOCS.2021.101429>, 2021.
- Guasta, M. D.: ICE-CAMERA: a flatbed scanner to study inland Antarctic polar precipitation, *Atmospheric Measurement Techniques*, 15, 6521–6544, <https://doi.org/10.5194/amt-15-6521-2022>, 2022.
- 320 Hicks, A. and Notaroš, B. M.: Method for classification of snowflakes based on images by a multi-angle snowflake camera using convolutional neural networks, *Journal of Atmospheric and Oceanic Technology*, 36, 2267–2282, <https://doi.org/10.1175/JTECH-D-19-0055.1>, 2019.
- Hobbs, P. V., Farber, R. J., and Joppa, R. G.: Collection of Ice Particles from Aircraft Using Decelerators, *Journal of Applied Meteorology* (1962-1982), 12, 522–528, <http://www.jstor.org/stable/26176719>, 1973.
- 325 Jaffeux, L., Schwarzenböck, A., Coutris, P., and Durore, C.: Ice crystal images from optical array probes: Classification with convolutional neural networks, *Atmospheric Measurement Techniques*, 15, 5141–5157, <https://doi.org/10.5194/amt-15-5141-2022>, 2022.
- Key, C., Hicks, A., and Notaroš, B. M.: Advanced Deep Learning–Based Supervised Classification of Multi-Angle Snowflake Camera Images, *Journal of Atmospheric and Oceanic Technology*, 38, 1399–1414, <https://doi.org/10.1175/JTECH-D-20-0189.1>, 2021.
- 330 Kuhn, T. and Heymsfield, A. J.: In Situ Balloon-Borne Ice Particle Imaging in High-Latitude Cirrus, *Pure and Applied Geophysics*, 173, 3065–3084, <https://doi.org/10.1007/s00024-016-1324-x>, 2016.
- Kuhn, T. and Vázquez-Martín, S.: Microphysical properties and fall speed measurements of snow ice crystals using the Dual Ice Crystal Imager (D-ICI), *Atmospheric Measurement Techniques*, 13, 1273–1285, <https://doi.org/10.5194/amt-13-1273-2020>, 2020.
- Mao, A., Mohri, M., and Zhong, Y.: Cross-Entropy Loss Functions: Theoretical Analysis and Applications, <https://arxiv.org/abs/2304.07288>, 2023.
- 335 Ramelli, F. ., Beck, A. ., Henneberger, J. ., Lohmann, U., Ramelli, F., Beck, A., and Henneberger, J.: ETH Library Using a holographic imager on a tethered balloon system for microphysical observations of boundary layer clouds Rights / license: Creative Commons Attribution 4.0 International Using a holographic imager on a tethered balloon system for microphysical observations of boundary layer clouds, *Atmospheric Measurement Techniques*, 13, 925–939, <https://doi.org/10.3929/ethz-b-000404920>, 2020.



- 340 Salehi, S. S. M., Erdogmus, D., and Gholipour, A.: Tversky loss function for image segmentation using 3D fully convolutional deep networks, <https://arxiv.org/abs/1706.05721>, 2017.
- Tan, M. and Le, Q. V.: EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, <http://arxiv.org/abs/1905.11946>, 2019.
- Touloupas, G., Lauber, A., Henneberger, J., Beck, A., and Lucchi, A.: A convolutional neural network for classifying cloud particles recorded by imaging probes, *Atmospheric Measurement Techniques*, 13, 2219–2239, <https://doi.org/10.5194/amt-13-2219-2020>, 2020.
- 345 Wolf, V.: Analysis of Arctic ice cloud properties using in-situ and remote sensing measurements, Ph.D. thesis, Luleå University of Technology, Space Technology, ISBN 978-91-7790-369-7, ISSN 1402-1544, 2019.
- Wolf, V., Kuhn, T., Milz, M., Voelger, P., Krämer, M., and Rolf, C.: Arctic ice clouds over northern Sweden: Microphysical properties studied with the Balloon-borne Ice Cloud particle Imager B-ICI, *Atmospheric Chemistry and Physics*, 18, 17 371–17 386, <https://doi.org/10.5194/acp-18-17371-2018>, 2018.
- 350 Wolf, V., Kuhn, T., and Krämer, M.: On the Dependence of Cirrus Parametrizations on the Cloud Origin, *Geophysical Research Letters*, 46, 12 565–12 571, <https://doi.org/10.1029/2019GL083841>, 2019.
- Wu, Z., Liu, S., Zhao, D., Yang, L., Xu, Z., Yang, Z., Zhou, W., He, H., Huang, M., Liu, D., Li, R., and Ding, D.: Neural Network Classification of Ice-Crystal Images Observed by an Airborne Cloud Imaging Probe, *Atmosphere - Ocean*, 58, 303–315, <https://doi.org/10.1080/07055900.2020.1843393>, 2020.
- 355 Xiao, H., Zhang, F., He, Q., Liu, P., Yan, F., Miao, L., and Yang, Z.: Classification of Ice Crystal Habits Observed From Airborne Cloud Particle Imager by Deep Transfer Learning, *Earth and Space Science*, 6, 1877–1886, <https://doi.org/10.1029/2019EA000636>, 2019.
- Zhang, H., Li, X., Ramelli, F., David, R. O., Pasquier, J., and Henneberger, J.: IceDetectNet: a rotated object detection algorithm for classifying components of aggregated ice crystals with a multi-label classification scheme, *Atmospheric Measurement Techniques*, 17, 7109–7128, <https://doi.org/10.5194/amt-17-7109-2024>, 2024.