Article

Deep Learning-Based Water Crystal Classification

Hien Doan Thi ^{1,†} , Frederic Andres ^{2,*,†} , Long Tran Quoc ^{1,†} , Hiro Emoto ^{3,†} , Michiko Hayashi ^{3,†} , Ken Katsumata ^{3,†} and Takayuki Oshide ^{3,†}

¹ Department of Computer Science, Vietnam National University, Hanoi 11300, Vietnam; hiendt104@gmail.com (H.D.T.); tqlong@vnu.edu.vn (L.T.Q.)

² National Institute of Informatics, Tokyo 101-8430, Japan

³ I.H.M General Research Institute, Tokyo 103-0004, Japan; hiromasa@hado.com (H.E.); hayashi@hado.com (M.H.); katsumata@hado.com (K.K.); oside@hado.com (T.O.)

* Correspondence: andres@nii.ac.jp; Tel.: +81-3-4212-2542

† These authors contributed equally to this work.

Abstract: Much of the earth's surface is covered by water. As was pointed out in the 2020 edition of the World Water Development Report, climate change challenges the sustainability of global water resources, so it is important to monitor the quality of water to preserve sustainable water resources. Quality of water can be related to the structure of water crystal, the solid-state of water, so methods to understand water crystals can help to improve water quality. As a first step, a water crystal exploratory analysis has been initiated with the cooperation with the Emoto Peace Project (EPP). The 5K EPP dataset has been created as the first world-wide small dataset of water crystals. Our research focused on reducing the inherent limitations when fitting machine learning models to the 5K EPP dataset. One major result is the classification of water crystals and how to split our small dataset into several related groups. Using the 5K EPP dataset of human observations and past research on snow crystal classification, we created a simple set of visual labels to identify water crystal shapes, in 13 categories. A deep learning-based method has been used to automatically do the classification task with a subset of the label dataset. The classification achieved high accuracy when using a fine-tuning technique.

Keywords: water crystal; deep learning; fine-tuning; supervised; classification

Citation: Thi, H.D.; Andres, F.; Quoc, L.T.; Emoto, H.; Hayashi, M.; Katsumata, K.; Oshide, T. Deep Learning-Based Water Crystal Classification. *Appl. Sci.* **2022**, *12*, 825. <https://doi.org/10.3390/app12020825>

Academic Editor: Chuan-Ming Liu

Received: 31 October 2021

Accepted: 10 December 2021

Published: 14 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Along with the development of society, research on the human impact on nature is of greater and greater concern. Water quality [1] has become one of the main challenges that societies will face during the 21st century, as the United Nations brought water quality issues to the forefront of international actions under the Sustainable Development Goal 6. It is important to monitor how human actions will affect water quality and pollution issues. Water has always played an important role in the climatic ecosystem. Because most of our planet is covered by water, 70 to 90% of the human body (depending on age) is water. Testing water quality is simple, but as water can exist in different states or phases (liquid, solid, and gas), it can be made simpler. Advanced research [2] has been completed to understand water phases, resulting in the discovery of a new phase for water liquid. Water quality can be evaluated in each of the four phases. The focus of our research is the solid water phase known as frozen water crystal. We define “Frozen Water Crystal”, a water crystal A microscopic crystal observed at the tip of a protrusion formed when liquid water is dropped in the form of drops onto a petri dish and frozen. The structure of the crystal is three-dimensional, and the crystal structure differs depending on the information possessed by the water. Crystals are formed when water changes to a solid-state, such as when it is frozen at -25 to -30 °C. Depending on the origin of the water and the formation process, crystals are divided into three main types: snow crystals, ice crystals,

and water crystals. From the shape of the crystal, the purity level and the texture are clearly reflected, which then enables us to assess the quality of the water. Depending on the environmental conditions and the impact of the surrounding elements, the same water can give many different shapes. Each type of shape of crystals can be considered to be unique, without repetition. Many studies have been performed on classifying snow crystals and ice crystals [3–6]. In recent years, the application of deep learning on crystals became popular in the research world with the publication of 3D Crystal classification [7]. However, no real classification and understanding of water crystals based on deep learning have been completed until now.

This article innovation is the application of artificial intelligence for the first time on a dataset of photos of water crystals known as “the 5K EPP Dataset” collected with the collaboration of the I.H.M General Research Institute. This dataset is composed of high-resolution photos captured by a microscope camera under laboratory conditions. All the photos have been stored and managed by the I.H.M General Research Institute. A simple water crystal structure definition has been proposed in this research, which references snow crystals classification from [5] and the EPP project. This definition provides an easy understanding of the water structure with a 2D image.

With non-prohibited purposes, the water crystal dataset can support other researchers interested in water with their dataset [8,9]. These results assess the affection of human beings on water crystals initialization and the quality of water.

Deep learning has been widely applied in many research fields and achieved surprising results, especially in image processing. Convolutional neural networks (CNNs) are a special kind of neural network analysing high dimensional features dataset such as image, video, etc. CNNs were developed with image processing in mind, which makes them computationally more efficient when compared to other multi-layer back-propagation neural networks. CNNs can be used to automatically extract features from the dataset, which simplifies the next process. These features are not only useful for specific tasks but also can help in other related tasks. This opens a new era for research with reduced effort to achieve good results. With its well-understood architecture, CNNs are nowadays used widely in many areas, including image and structure classification. However, deep neural networks (DNNs) trained by conventional methods with small datasets commonly show worse performance than traditional machine learning methods [10]. The deeper network requires more data to train. This limitation prevents the wide application of deep learning in any field, in which collecting and assembling big datasets is a challenge. With the 5K EPP dataset, we face the same problem. We use data transformation techniques to enhance our dataset and the fine-tuning method is used to help train the model so that the model can learn better from a small dataset. Class weight is also used in this research to solve the imbalanced dataset problem. To build a classifier with deep learning, we split our work into 2 main steps: feature extractor and classification. We built a deep learning model to extract meaningful features from the EPP dataset, then used those to classify water crystal structures. We used 2 different techniques to extract features from the EPP dataset: convolutional auto-encoder and Fine-tuning. The extracted features are then stacked in convolution layers to make a classifier. The convolutional auto-encoder (CAE) [11] has been widely applied in dimension reduction and image noise reduction. Because the CAE model can keep the spatial information of the original image and extract information gently by using the convolution layer, it is considered to be one of the most state-of-the-art techniques in deep learning nowadays. Furthermore, it is an unsupervised method and can be used with less effort than a supervised one. Fine-tuning is a useful method for improving the performance of a neural network. It helps the researchers achieve higher performance with less effort.

In fine-tuning, a model trained on a given task is used for another similar task. This method reduced the training time and effort to extract meaningful features from the original input. ImageNet pre-trained models have been used for the fine-tuning method. This paper is organized as follows. Related works are in Section 2, the 5K EPP dataset is described in

Section 3, our dataset study approach and methods are in Section 4, Section 5 describes the experimental results, and the conclusions follow in Section 6.

2. Related Works

With a research focus to improve precipitation measurements and forecast for over 50 years, scientific studies of meteorology and weather include the study of snowflakes, ice crystals, and water crystals. Snowflake studies provide some of the most detailed evidence of climate change. It impacts atmospheric science. One of the first attempts to catalog snowflakes was made in the 1930s by Wilson Bentley who created a method of photographing snowflakes in 1931, using a microscope attached to a camera. The Bentley Snow Crystal Collection (<https://snowflakebentley.com/> accessed on 15 October 2020) includes about 6125 items. A general classification of snow crystals $T_a - s$ diagram was proposed by Nakaya [3], which provides the most perfect classification from a physical point of view, with 7 categories. These categories include needles, columns, fern-like crystals developed in one plane, the combination of column and plane crystals, rimed crystals, and irregular crystals. The crystal images were collected from a slope of Mount Takachi, near the center of Hokkaido Island. Magono [4] published an improved version of Nakaya's classification, with the modification of and a supplement to Nakaya's classification of snow crystals. The results were obtained by laboratory experiments and from meteorological observation. The new classification provides temperature and humidity conditions, which can describe the meteorological differences in groups of asymmetric or modified types of snow crystals. It provides 80 categories, modified from Nakaya's categories and adding some new categories as well. Thirty thousand microscopic photographs of snow crystals taken by the Cloud Physics Group were used in their research.

Kikuchi and his team [5] proposed a new classification with 121 categories to classify snow crystals, ice crystals, and solid precipitation particles. They qualified their classification as "global scale" or "global" because their observations were performed from the middle latitudes (Japan) to polar regions. This classification consisted of three levels: general, intermediate, and elementary—which are composed of 8, 39, and 121 categories, respectively. Interestingly, this classification can be used not only for snow crystals but also for ice crystals. The deep learning method has been widely applied in many research fields, especially with image datasets. However, it faces the problem of working from a limited dataset. Fortunately, with the advent of image collection methods, a method to collect snowflake images was proposed: the Multi-Angle Snowflake Camera (MASC) [12]. It was developed to address the need for high-resolution multi-angle imaging of hydrometeors in freefall and has resulted in datasets comprising millions of images of falling snowflakes. Several studies have been published resulting from this development. A new method to automatically classify solid hydrometeors based on MASC images was presented by Praz et al. [13]. In this research, they proposed a regularized multinomial logistic regression (MLR) model to output the probabilistic information of MASC images. That probability is then weighed on the three stereoscopic views of the MASC to assign a unique label to each hydrometeor. The MLR model was trained using more than 3000 MASC images labeled by visual inspection. This model achieved very high performance with a 95% accuracy. Hicks et al. [6] published an automatic method to classify snowflakes, collected via Multi-Angle Snowflake Camera (MASC). The training dataset contains 1400 MASC images. They used a convolutional neural network and residual network which had been pre-trained with ImageNet as a back-bone for their model. Snowflakes were sorted by geometrics and divided into 6 distinct classes. Then, the degrees of rimming was decided by another training process, which has three distinct classes. Although the accuracy of this research is only 93.4%, it does provide a new way to classify snowflakes or nature structures automatically.

Another research with the MASC dataset was proposed by Leinonen et al. [14]. In this research, they aimed to classify large-scale MASC dataset by unsupervised learning method, using generative neural network (GAN) [15] and K-medoids [16]. With the

features extracted from the discriminator part of the GAN model, they used the K-medoids algorithm to cluster all the images (data points) into 16 classes/categories. This method not only shows the hierarchical clustering groups but also requires no human intervention with such a large dataset. However, MASC images mainly show the crystal's degree of riming, but not the crystal's structure. This is because these images were taken during the falling progress of snowflakes.

In this research, we focus on building a new definition for water crystal classification based on previous studies and using deep learning to automatically classify them.

3. The 5K EPP Dataset

The water crystals have been provided by the Emoto Peace Project (EPP) at the I.H.M General Research Institute (Tokyo, Japan). Crystals were produced from water samples collected from many countries and sources, with the help of scientists all around the world. Water samples from each bottle are produced by the same procedure in [9]:

- From each bottle, a drop (approximately 0.5 mL) of water is placed into each of the 50 Petri dishes. So, there are 50 waterdrops from each bottle;
- Those dishes are then placed on a tray in a random position in a freezer maintained at -25 to -30 °C. The random placements helps to ensure that potential temperature differences within the freezer would be randomized among the dishes;
- The dishes are then removed from the freezer, and placed in a walk-in refrigerator (maintained at -5 °C). A water crystal photo is taken on the top of each resulting ice drop using a stereo optical microscope at either $100\times$ or $200\times$, depending on the presence and size of a crystal.

Known as the **5K EPP dataset** [17], this dataset contains 5007 crystal photos in total. Because the 5K EPP dataset contains very high-resolution images (5472×3648 pixels) and water crystals only occupy a small part in the images, we needed to preprocess each image to remove the background. We used Otsu's method [18] to automatically define the border around the crystals. The minimum rectangular box that can cover each water crystal was chosen to crop the background. This helps reduce the image size while retaining the details in the object. Because the size of the water crystal in each image is different, we resized the cut-off images to the same size, to fit with the input of our machine.

The preprocessed dataset was then sorted into 13 categories. Based on the knowledge from the EPP Laboratory experts, we chose those categories that appeared most frequently in the 5K EPP dataset as our labels. We built a tree-like diagram in Figure 1 to demonstrate how we split the 5K EPP dataset into smaller categories. The branches of the tree correspond to the category in the definition. Finally, we obtained 13 branches corresponding to 13 categories. The details are given in Table 1. We chose the most typical images for each category and labeled them with the predefined definition. We split the 5K EPP dataset into the training set and test set with ratios of 80 and 20, respectively. The scikit-learn (<https://scikit-learn.org/> accessed on 15 October 2020) library was used to split the dataset randomly and guarantee the balance in the dataset.

Figure 1. A tree-like diagram to demonstrate the water crystal categories with 5K EPP dataset.

Table 1. The definition for water crystal classes based on the knowledge from [5] classification.

Category	Crystal Example	Definition
Microparticule		Crystal made up of fine particle on a hexagonal plate
Simple plate		Hexagonal crystal with no outer decoration
Fan-like plate		Square plate with a fan-shaped decoration on the outside
Dentrite plate		A square plate with dendritic decoration on the outside
Fern-like dendrite plate		A square plate with fern-like decorations on the outside
Column/Square		Square or columnar crystal/block crystal
Singular Irregular		Square plate with a fan-shaped decoration on the outside
Cloud-particle		A granular decoration on a square plate
Combinations		Multiple square plates assembled together without overlapping vertically
Double plate		Two square plates stacked on top of each other
Multiple Columns/Squares		Multiple square or columnar crystals / Multiple block crystals
Multiple Irregulars		Multiple asymmetrical crystals or crystals that are not fully formed
undefined		Types of water crystals without crystals

4. Proposed Method

4.1. Feature Extractor

4.1.1. Residual Auto-Encoder

A convolutional auto-encoder (CAE) is an efficient technique used to reduce dimensionality and generate high-level representation from raw data. It is an unsupervised learning algorithm using a back propagation algorithm to update parameters. In this model, the targets are equal to the inputs. A convolutional auto-encoder is composed of two models: an encoder and a decoder. The encoder aims to find the latent representation for input data, while the decoder is tuned to reconstruct the original input from the encoder's output.

Considering a dataset X with n sample and m features, the encoder learns the latent representation H and the decoder tries to reconstruct the original input X' from H , by minimizing the differences between X and X' over all samples:

$$\min_{W, W'} \frac{1}{n} \sum_{i=1}^n \|D_{W'}(E_W(X_i)) - X_i\|_2^2.$$

For a convolutional auto-encoder,

$$E_W(X) = \sigma(X * W) = H$$

$$D_{W'}(H) = \sigma(H * W') = X'$$

where W and W' are learnable parameters and σ is the activation function such as ReLU and sigmoid. At the end of the training process, the embedded code H is used as a new representation of input X . Then, H can be fed into a fully connected layer to do classifying or clustering tasks. We proposed a new CAE model to extract latent representation from high-resolution water crystal images. First, in the encoder, 3 convolution layers were stacked on the input images to extract latent representation. Then, the encoder's output was flattened to form a vector, which is an extracted feature. The decoder transformed embedded features back to the original image. The convolution (transpose) layers with stride allow the network to learn spatial subsampling (upsampling) from data, leading to a higher capability of transformation. Therefore, instead of using a convolution layer followed by a pooling layer, we used a convolution layer with a stride in the encoder and a convolution transpose layer with a stride in the decoder. To achieve low-dimension images with high-representation with very high-resolution images was a challenging task. Down-sampling images to get low dimension representation can lead to a vanishing gradient when training a very deep neural network model. With a traditional CAE, the greater number of hidden layers, the hard it is to reconstruct the original image. To solve this problem, we used the skip idea from ResNet [19]: skip connection. Skip connection addresses the problem with vanishing gradient and information lossless. The idea is that instead of letting the model learn underlying mapping, let it learn the residual mapping.

With skip connection, the residual or identity was added to the output. We obtained the output defined as follows: $y = R(x) + x$.

Because we had the identity connection come due to x , the model actually learned the residual $R(x)$. We used two different kinds of residual block to build an encoder block: a regular block and a downsample block. As the regular block, the residual block has 3 convolution layers with the same number of output channels. The downsample block decreases the sampling rate of the input by deleting samples. When the block performs frame-based processing, it resamples the data in each column of the Mi-by-N input matrix independently. When the block performs sample-based processing, it treats each element of the input as a separate channel and resamples each channel of the input array across time. The resample rate is K times lower than the input sample rate, where K is the value of the Downsample factor parameter. The Downsample block resamples the input by discarding $K - 1$ consecutive samples following each sample that is output.

Each convolution layer was followed by a batch normalization layer and a ReLU activation function. Then, we skipped these three convolution operations and added the input directly before the final ReLU activation function. This kind of design requires that the output of the three convolution layers be of the same shape as the input, so they can be added together. The downsample block had the same design as the regular one, but the first convolution layer reduced the image size and had a different number of channels. To add the input before the last ReLU activation function, we used a 1×1 convolution layer, followed by a batch normalization layer, to transform the input into the desired shape for the addition operation. By experimentation, we found that using two convolution layer after the first convolution layer in each block helps the model reconstruct output better.

The first convolution layer has been used to downsize the image by two and the following two were used to learn useful information. Each convolution layer is followed by a batch normalization layer and an activation layer (except the last one). In this research, we chose ReLU as an activation layer. The skip connection used convolution and batch normalization to reduce the size of the input so that it was equal to the output. The final architecture is shown in Figure 2. We used one convolution layer and three residual blocks to build an encoder. The decoder kept the same structure as the origin. The final model is called a residual auto-encoder (RAE). The reconstruction loss was used to evaluate the performance of the RAE model. The parameters of encoder and decoder were updated by minimizing the reconstruction error:

$$L_r = \frac{1}{n} \sum_{i=1}^n \text{Distance}(D_{W'}(E_W(x_i)), x_i).$$

Figure 2. A residual auto-encoder model to extract features from origin images. Each residual block is a combination of a Downsample block and a regular block, respectively.

Instead of using the Euclidean distance to compute the reconstruction error, we used the Spherical distance in [20]. The latent representations extracted from the RAE model were projected into the surface of the unit hyper-sphere. The distance between data points in that surface was then measured by $d_{spherical}$ function, which is defined as follows:

$$d_{spherical} = \frac{\arccos(s_{\cos(z_i, z_j)})}{\pi} = \frac{1}{\pi} \left\langle \frac{z_i}{\|z_i\|_2 + \epsilon}, \frac{z_j}{\|z_j\|_2 + \epsilon} \right\rangle$$

where $\arccos(\alpha)$ is the inverse cosine function for $\alpha \in [-1, 1]$ and ϵ is a very small value to avoid numerical problem.

4.1.2. Fine-Tuning Model

The efficiency of the classification model is based on the power of the features extracted from the training dataset. With the high meaning features, the classifier can achieve good results from the very first training steps. Auto-encoder is a popular strategy used to extract features from the unlabeled dataset. Because it requires no label to train the CNN model, it can perform well on high dimension datasets, especially images and videos. The model then trains with our full dataset and learns the most important information from these images. However,

to choose a good architecture for the auto-encoder and train it from scratch is not easy as it requires a great deal of knowledge about machine learning and specific datasets. A new method was introduced to help to solve problems with feature extracting, called fine-tuning. Fine-tuning is a process that takes a network that has already been trained for a given task and make it perform a second similar task. Many studies have been shown that fine-tuning techniques can get good results with less effort compared to starting from scratch.

For image related tasks, the most common way is fine-tuning the model trained on ImageNet [21] (with 1.2 million labeled images) by continuing to train it on the original dataset. A competition on classification and object detection has been organized to find state-of-the-art techniques to solve those problems on the ImageNet dataset, called ImageNet Large Scale Visual Recognition Challenge (ILSVRC) (<http://www.image-net.org/challenges/LSVRC/> accessed on 15 October 2020). AlexNet [22] is the first larger-scale convolution model that does well on the ImageNet classification task, outperforming all previous non-deep learning-based models by a significant margin, and won the competition in 2012. After that, VGG [23] was proposed with the idea of deeper networks and much smaller filters, which is a significant jump in deep learning. ResNet [19] introduced residual blocks with skip connections, which allow the gradients to backpropagate through to the initial layers without vanishing. That model won the first prize in the ILSVRC 2015 competition with an error rate of just 3.6%. In 2016, a new model was proposed, called SqueezeNet [24]. This model achieved approximately the same level of accuracy as AlexNet with a much smaller number of parameters. So, it is suitable for building mobile applications. In the same year, DenseNet [25], a densely connected network, was proposed to improve higher layer architectures of the previously developed networks. The DenseNet architecture attempts to solve this problem by densely connecting all the layers: each layer gets the input from the previous layer's output. We used all those models as back-bone to build a model to classify our water crystals. All the experiment results are shown in Table 3.

4.2. Classification Model

The features extracted from each previous step are then feed into the classifier layers to build the classification model.

The classifier has 2 main parts: feature extractor and classifier. To build the extractor, RAE pre-trained and Image pre-trained models were used. With the RAE model, we keep only the encoder from the RAE model, which had been pre-trained with the EPP dataset, as a feature extractor. As with the ImageNet pre-trained models, the last layer is removed to get the latest features. The classifier includes three fully connected layers which are added on top of the feature extractor and then trained simultaneously with the labeled dataset. The overview of the final classification model is given in Figure 3.

Figure 3. Classification architecture overview. The feature extractor can be replaced by RAE pre-trained model or ImageNet pre-trained model. The features are used as input for the next step. The classifier contains 3 fully connected layers, each of them is followed by ReLU and a dropout layer. The last FC outputs the predicted probability distribution. Softmax is added to get the final prediction.

Unlike training a model from scratch, we unfreeze early layers and train the whole network. A small learning rate has been chosen to let the classifier learn the patterns from the previously learned convolution layers in the pre-trained network. For further evaluation and improvement, we chose different metrics to compare the performance between different feature extractors. The comparison results are then shown in Section 5.

4.3. Imbalanced Data

Due to the crystal formation process in nature, the amount of data in each class is imbalanced. Therefore, when labeling the 5K EPP dataset, we realized that there is an imbalance between the number of images among the categories. Some categories include approximately 20% dataset while some others include just 2%. The details are provided in Table 2.

Table 2. The 5K EPP dataset summary.

Category	Card(Photo)	Percentage
Microparticle	161	3.2%
Simple plate	104	2%
Fan-like plate	341	6.81%
Dendrite plate	1388	27.72%
Fern-like dendrite plate	674	13.46%
Column/Square	38	7.5%
Singular Irregular	674	13.46%
Cloud-particle	3	0.0006%
Combination	129	2.57%
Double plates	204	4%
Multiple Columns/Squares	172	3.4%
Multiple Irregular	692	13.82%
Undefined	427	8.52%

To guarantee balance and accuracy when training the deep learning model, we used the class weight method. We simply provided a weight for each class which places more emphasis on the minority classes. Following that idea, the model can learn from all classes equally. Each class will be assigned a weight corresponding to the number of images inside. The weight can be calculated as follows:

$$w_i = \frac{N}{C * n_i}$$

where w_i , n_i , C , and N are the weight assigned to class i , the number of images of class i , the number of classes, and the total images of dataset, respectively. We also use F_1 -score metric to evaluate the model performance with an imbalanced dataset besides the standard evaluation metric. Both are described in Section 5.1.

5. Experiments and Results

5.1. Evaluation Metric

5.1.1. Classification Accuracy

We used standard evaluation metrics to evaluate classification results. For all implementation setup, we set the number of classes equal to the number of ground-truth

categories that were used to label the dataset in Section 3. The performance is evaluated by the accuracy metric:

$$ACC = \frac{1}{n} \sum_{i=1}^n (y_i^{true} = y_i^{pred}) \quad (1)$$

where y^{true} is the ground-truth label, y^{pred} is the prediction label, and n is number of images inside the test set. The test dataset is not used when training the model. The best model should have high accuracy for both training and test progress.

5.1.2. F_1 -Score

With the imbalanced dataset, an efficient way to evaluate the model performance is using F_1 -score [26]. Instead of calculating the ratio of true prediction within the total images, F_1 -score measures accuracy by precision p and recall r . The formula for F_1 -score is defined as follows:

$$F_1 = 2 * \frac{precision * recall}{precision + recall} \quad (2)$$

When p is the number of correct positive results divided by the number of all positive results returned by the classifier, and r is the number of correct positive results divided by the number of all relevant samples (all samples that should have been identified as positive).

5.2. Experiments Environment and Setup

In this section, we discuss applying different pre-trained models used in fine-tuning with our 5K EPP dataset. For our experiments, we used an NVIDIA Tesla V100 SXM2 GPU, with 32GB of memory. The server used for running the experiments was Grid5000 [27] (<https://www.grid5000.fr> accessed on 15 October 2020), the French large-scale and flexible experimental grid platform consisting of 8 sites geographically distributed over France and Luxembourg. Each site comprises one or several clusters, for a total of 35 clusters inside Grid5000. This platform is dedicated to experiment-driven researches in all areas of computer science, with a focus on parallel and distributed computing including Cloud, HPC, and Big Data and AI. Our implementation is based on Python and Pytorch (<https://pytorch.org> accessed on 15 October 2020).

5.3. Experiment Results

5.3.1. Residual Auto-Encoder Model (RAE)

We first trained the RAE model with an unlabelled dataset. Adam optimizer [28] was used to update model parameters, with learning rate $\alpha = 10^{-4}$. Regularization was used to reduce the overfitting problem, with $\gamma = 10^{-5}$. We chose the number of images per batch equal to 32. The model was trained with 100 epochs. We used two different loss functions to train the RAE model: one Spherical citetran2019deep metric and the Binary Cross-Entropy (noted BCE). The reconstruct results built with both metrics are shown in Figure 4. Although the BCE loss function can reconstruct an image quite similar to the original one, when zooming out the image, we can see that some parts of the image are blurred and old content cannot be seen. With Spherical, the reconstructed image is the same as the original one. We also used the Structural Similarity Index (SSIM) [29] to assert the similarity among reconstructed images and the input. The average SSIM has been computed for both Spherical and BCE. In overall, Spherical's SSIM is 0.96, while the BCE's SSIM is 0.89. Therefore, we concluded that Spherical outperformed BCE.

Figure 4. Reconstruct image generated by RAE model train with BCE and Spherical metric separately. The SSIM index is calculated with each reconstructed image. Spherical one outperforms the BCE one.

5.3.2. Classification Model

We trained the classification models proposed in Section 4. Stochastic Gradient Descent (SGD) with Nesterov momentum optimizer was used to update parameters, with learning rate $\alpha = 10^{-3}$, momentum $\Delta = 0.9$ and regularization $\gamma = 10^{-4}$. To enrich the dataset, we used transform techniques such as flip image (vertical and horizontal) with probability $p = 0.5$, rotating the image with a random degree in the range from -90 to 90 degree, random cropping. The classification model is first trained with 100 epochs.

As in Section 4, we used 2 kinds of pre-trained models to build classifiers: RAE pre-trained model and ImageNet pre-trained models. RAE model was trained with an unlabeled dataset, as mentioned in the previous result. With ImageNet pre-trained models, we chose the most popular deep learning models, which had won the ILSVRC: AlexNet, VGG, SqueezeNet, DenseNet, and ResNet. All parameters were adjusted during the training progress. The 5K EPP dataset was then divided into a training and a test set with a ratio of 80:20.

After visualizing and doing statistics on the prediction, we realized that the definition with 13 categories for water crystal was not too good and gave unclear instructions. The classification model sometimes misclassified between water crystal and its spatial form. Because we used a 2D image to classify the plates, it is hard to see the differences between a plate with and without space elements. As mentioned in [30], we should apply machine learning in a problem that humans can do well. We modified the definition to eliminate ambiguity in the labeling process. The major change was combining the space misclassified categories. Additionally, we also found a new category named double plates. Finally, we delivered 12 categories, which are defined in Section 3.

Two fully-connected layers are added on the top of modified models, followed by a ReLU activation layer. In addition to regularization, we also used the traditional dropout method to prevent overfitting problems [31]: a dropout layer is put after each fully-connected layer, except the last one. We fine-tuned parameters and the applied data transformation techniques mentioned in Section 3 to enrich the dataset.

With the new dataset, the model has kept training with the same configuration and parameters. The new definition obtained significant performance improvement. The model could overcome the overfitting problem and obtain high accuracy for both training and test progress.

We used different pre-trained models as back-bone and trained the model with the same parameters and set up conditions. The standard accuracy and F_1 -score were calculated and compare among models. The results are shown in Table 3. The model trained with RAE outperforms models that used AlexNet and SqueezeNet as back-bone, with 4% higher than AlexNet and 8% higher than SqueezeNet in F_1 -score metric. Although other pre-trained models (such as VGG, DenseNet) have high accuracy, the F_1 -scores of Alexnet and SqueezeNet are much lower. In addition, the loss values of VGG and DensetNet are two times bigger than the lowest one (e.g., RestNet). ResNet outperforms other models in both loss value and accuracy with 98.50% Top-1 accuracy and 97.25% F_1 -score. We concluded

that the ResNet back-bone is the best solution for our problem. When evaluating the model with test set, the ResNet accuracy is approximately 93%.

Table 3. Top-1 Accuracy and F_1 -score on 5K EPP training set.

Backbone	Loss	Accuracy	F_1 -Score
RAE	0.094	94.35%	91.64%
AlexNet	0.086	93.71%	87.79%
VGG	0.049	96.21%	92.03%
SqueezeNet	0.130	91.16%	83.31%
DenseNet	0.046	96.93%	93.55%
ResNet	0.025	98.50%	97.25%

5.3.3. Comparative Model

To demonstrate the effectiveness of our model, we compared it with Hicks's model [6]. Both methods used ResNet pre-trained models as the backbone and used the fine-tuning method to train all the parameters.

In their study, Hicks et al. implemented a classifier to automatically classify geometrically and riming-degree of the MASC dataset. They used the ResNet pre-trained model to initialize the model parameter and added a new FC layer as a classifier layer. The model outputs the probability of 6 distinct snowflake categories, which is defined by Hicks et al. They used 2 CNN models to do distinct tasks: (1) classify crystal geometrics and (2) classify riming-degree. Based on the crystal structure classification purpose, we compared our model with Hicks' first model. We trained Hicks' model with the 5K EPP dataset and used classification accuracy to compare its performance to ours. The results are shown in Figure 5. Even though our accuracy is just slightly higher than Hicks', the training progress can show that our model is more stable and the convergence of our model is better than that of the Hicks' model.

Figure 5. Our proposed model compared with Hicks's model. Both implementations are trained on the 5K EPP dataset.

6. Conclusions

Based on the EPP water crystal dataset and the previous knowledge about snowflake classification, we proposed a simple water crystal definition, which can be used to classify the EPP dataset. We contributed a new data science dataset, called the 5K EPP dataset, with 5007 images split into 13 classes (12 categories + undefined). We proposed a deep learning-based method to automatically classify this dataset. We compared fine-tuning results between the residual auto-encoder model, trained with unlabelled EPP datasets, and ImageNet pre-trained models, and then selected the best one. With a fine-tuning technique and ResNet pre-trained model, we had a classifier with 93% accuracy. With this result, we are going to extend the 5K EPP dataset by applied the water crystal definition to label

the EPP water crystal 20K dataset. A new approach to using an unsupervised method to deal with the unlabeled dataset and find a new group of the water crystal structure will be targeted in further studies.

Author Contributions: Conceptualization, H.D.T., F.A. and L.T.Q.; methodology, H.D.T. and L.T.Q.; software, H.D.T.; validation, H.D.T., F.A., L.T.Q., H.E., M.H., K.K. and T.O.; formal analysis, H.D.T. and L.T.Q.; investigation, H.D.T., F.A., K.K., and T.O.; resources, K.K. and T.O.; data curation, K.K., T.O., M.H., F.A. and H.D.T.; writing—original draft preparation, H.D.T., F.A. and L.T.Q.; writing—review and editing, H.D.T., F.A., L.T.Q., H.E., M.H. and K.K.; visualization, H.D.T., F.A., L.T.Q., H.E., M.H. and K.K.; supervision, F.A., L.T.Q., H.E., M.H., K.K. and T.O.; funding acquisition, F.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Institute of Informatics (NII) under the GLO Internship program and Emoto Peace Project, Non-profit Organization.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data will be available for the data science.

Acknowledgments: We would like to thank you the National Institute of Informatics (Tokyo, Japan) for the support of the research and I.H.M General Research Institute (Tokyo, Japan) for their help in the Water Crystal classification. We are grateful to D’Orazio and the French Grid5000 programs (<https://www.grid5000.fr> (accessed on 15 October 2020)) for providing the grid infrastructures, advice, and user assistance.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Boyd, C.E. *Water Quality: An Introduction*, 3rd ed.; Springer Nature Switzerland AG: Berlin/Heidelberg, Germany, 2020. [CrossRef]
2. Pollack, G. *The Fourth Phase of Water: Beyond Solid, Liquid and Vapor*; Ebner & Sons: Springfield, OH, USA, 2013.
3. Nakaya, U. *Snow Crystals: Natural and Artificial*; Hokkaido University: Hokkaido, Japan, 1954.
4. Magono, C.; Lee, C.W. Meteorological classification of natural snow crystals. *J. Fac. Sci. Hokkaido Univ. Ser. 7 Geophys.* **1966**, *2*, 321–335.
5. Kikuchi, K.; Kameda, T.; Higuchi, K.; Yamashita, A.; Working Group Members for New Classification of Snow Crystals. A global classification of snow crystals, ice crystals, and solid precipitation based on observations from middle latitudes to polar regions. *Atmos. Res.* **2013**, *132*, 460–472. [CrossRef]
6. Hicks, A.; Notaroš, B. Method for Classification of Snowflakes Based on Images by a Multi-Angle Snowflake Camera Using Convolutional Neural Networks. *J. Atmos. Ocean. Technol.* **2019**, *36*, 2267–2282. [CrossRef]
7. Ziletti, A.; Kumar, D.; Scheffler, M.; Ghiringhelli, L.M. Insightful classification of crystal structures using deep learning. *Nat. Commun.* **2018**, *9*, 2775. [CrossRef] [PubMed]
8. Radin, D.; Hayssen, G.; Emoto, M.; Kizu, T. Double-blind test of the effects of distant intention on water crystal formation. *Explore* **2006**, *2*, 408–411. [CrossRef] [PubMed]
9. Radin, D.; Lund, N.; Emoto, M.; Kizu, T. Effects of distant intention on water crystal formation: A triple-blind replication. *J. Sci. Explor.* **2008**, *22*, 481–493.
10. Feng, S.; Zhou, H.; Dong, H. Using deep neural network with small dataset to predict material defects. *Mater. Des.* **2019**, *162*, 300–310. [CrossRef]
11. Masci, J.; Meier, U.; Cireşan, D.; Schmidhuber, J. Stacked convolutional auto-encoders for hierarchical feature extraction. In *International Conference on Artificial Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 52–59.
12. Garrett, T.; Fallgatter, C.; Shkurko, K.; Howlett, D. Fall speed measurement and high-resolution multi-angle photography of hydrometeors in free fall. *Atmos. Meas. Tech.* **2012**, *5*, 2625–2633. [CrossRef]
13. Praz, C.; Roulet, Y.A.; Berne, A. Solid hydrometeor classification and riming degree estimation from pictures collected with a Multi-Angle Snowflake Camera. *Atmos. Meas. Tech.* **2017**, *10*, 1335–1357. [CrossRef]
14. Leinonen, J.; Berne, A. Unsupervised classification of snowflake images using a generative adversarial network and K-medoids classification. *Atmos. Meas. Tech.* **2020**, *13*, 2949–2964. [CrossRef]
15. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2014; pp. 2672–2680.
16. Jain, A.K.; Murty, M.N.; Flynn, P.J. Data clustering: A review. *ACM Comput. Surv. (CSUR)* **1999**, *31*, 264–323. [CrossRef]
17. Emoto, H.; Doan Thi, H.; Andres, F.; Hayashi, M.; Katsumata, K.; Oshide, T.; Tran, L. 5K EPP Dataset 2021. Available online: <https://ieee-dataport.org/documents/5k-epp-dataset> (accessed on 15 October 2019). [CrossRef]
18. Otsu, N. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern.* **1979**, *9*, 62–66. [CrossRef]

19. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
20. Tran, B.; Le Thi, H.A. Deep Clustering with Spherical Distance in Latent Space. In *International Conference on Computer Science, Applied Mathematics and Applications*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 231–242.
21. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
22. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2012; pp. 1097–1105.
23. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
24. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
25. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
26. Goutte, C.; Gaussier, E. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In *European Conference on Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 345–359.
27. Balouek, D.; Carpen Amarie, A.; Charrier, G.; Desprez, F.; Jeannot, E.; Jeanvoine, E.; Lèbre, A.; Margery, D.; Niclausse, N.; Nussbaum, L.; et al. Adding Virtualization Capabilities to the Grid’5000 Testbed. In *Cloud Computing and Services Science*; Ivanov, I.I., van Sinderen, M., Leymann, F., Shan, T., Eds.; Communications in Computer and Information Science; Springer International Publishing: Berlin/Heidelberg, Germany, 2013; Volume 367, pp. 3–20. [[CrossRef](#)]
28. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
29. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [[CrossRef](#)]
30. Ng, A. Machine Learning Yearning. 2017. Available online: [http://www.mlyearning.org/\(96\)](http://www.mlyearning.org/(96)) (accessed on 15 October 2019).
31. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.