


Article

Auto-Encoder Classification Model for Water Crystals with Fine-Tuning

Hanan A. Hosni Mahmoud ^{1,*}  and Nada Ali Hakami ²

¹ Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh 11671, Saudi Arabia

² Department of Computer Science, College of Computer Science and Information Technology, Jazan University, Jazan P.O. Box 45142, Saudi Arabia

* Correspondence: hahosni@pnu.edu.sa

Abstract: Water is one of the important, though scarce, resources on earth. The 2021 World Water Resource Report claims that environmental challenges threaten the sustainability of water resources. Therefore, it is vital to screen water quality to sustain water resources. Water quality is related to water crystal structure in its solid state. Intelligent models classify water crystals to predict their quality. Methods to analyze water crystals can aid in predicting water quality. Therefore, the major contribution of our research is the prediction of water crystal classes. The proposed model analyzes water crystals in solid states, employing image analysis and the deep learning method. The model specifies several feature groups, including crystal shape factors, solid-state features, crystal geometry and discrete cosine transform coefficients. The model utilizes feature fusion for better training. The proposed model utilized the EP water crystal dataset from the WC image depository and its accuracy was tested with the multi-feature Validation technique. The nature of our data inclined us to utilize F-Measure and sensitivity for the testing phase. Our proposed model outperformed other state of the art water crystal classification models by more than 6% in accuracy and 7% in f-measures, with performance exceeding 11% for triple feature fusion. Furthermore, our model was faster in training time (10% of the training time of the comparative models) and had 1.42 s classification time.

Keywords: deep learning; water crystals; classification model; solid-state features



Citation: Hosni Mahmoud, H.A.; Ali Hakami, N. Auto-Encoder Classification Model for Water Crystals with Fine-Tuning. *Crystals* **2022**, *12*, 1667. <https://doi.org/10.3390/cryst12111667>

Academic Editors: Saskia Schimmel and Natasha Dropka

Received: 28 October 2022

Accepted: 18 November 2022

Published: 19 November 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

Informatics is widely employed in several detection algorithms. Informatics systems usually use deep learning in the classification of water crystals (WCs) [1]. WCs represent progressive environmental issues, with several solid states and solid-state vibrations [2,3]. Due to the importance of water quality, accurate and dependable water crystal classification models are required for the classification of WC cases [4–7].

WC classification is based on identifying features utilizing various instruments. Solid-state disarray is a common feature, and many water crystals face solid-state disarrays. Hence, water crystal classification models using solid-state disarrays are important in research into WC detection [4–8]. In research, multiple solid-state digital crystal processing (dsp) algorithms are utilized to ascertain the significant features of healthy water crystals. Feature extraction process output is used in supervised deep learning models to attain robust decisions in WC detection. Neural networks and deep learning [9–11] are common models in WC detection. These models depend on the feature extraction process from labeled data [12–15]. It is not feasible to extract features manually, as they represent the solid-state features of the water crystals, so the disarray features of the data are used by intelligent classification techniques. Deep Convolutional Neural Networks can generate feature maps that can be utilized as inputs in the learning process. Deep learning exhibits model performances in aspects such as solid-state and image recognition [16–18]. Such

high precision results inspire researchers to employ DAE in WC detection [19]. DAE has the ability to model complex associations from inputs.

In this research, we proposed two techniques to select features from the EP database: the first is a deep learning-based auto-encoder, and the second technique is fine-tuning. The selected features are then fed to the CNN classifier. A deep learning-based auto-encoder (DAE) [11] is usually used in data dimensionality reduction. The DAE model retains the visual information of the input image and selects information using neural computing. DAE is an unsupervised learning model. Fine-tuning is an advantageous technique for enhancing the precision of convolutional networks. It can achieve higher performance with less training time. Pre-trained ImageNet is utilized for the fine-tuning algorithm.

The proposed model for water crystal class detection used solid-state features. The proposed DAE model employed a feature joining mechanism. DAE was employed to directly extract feature states from several feature groups. The feature state representations were fed to the successive convolutional layers (CLs) and fully connected (FC) to finish the classification. The DAE model, with and without fine tuning, could capture the impact of different features in a feature model-centered methodology. The first model is simpler and requires less resources, while the second model requires more resources, but yields better performance, both in accuracy and execution time.

The occurrence of multiple feature recordings per sample in both the training and testing data groups, might produce biased results in experiment evaluation. The data might contain multiple solid-state recordings for both water crystal cases and water disarray cases. Therefore, we employed a cross validation methodology for Leave-Out cases for unbiased testing of our model. In each convolution iteration, instances of data cases were taken out in the test stage, while the remaining instances were employed in the training stage.

The rest of the paper is divided as follows. Section 2 describes the literature survey. Section 3 provides a detailed description of the data and feature groups. Section 4 presents the proposed DAE model, with and without fine tuning, and the evaluation metrics utilized. Section 5 specifies the experimental and comparative results. Section 6 depicts the conclusions.

2. Literature Review

We reviewed recent research on WC classification models that use deep learning techniques and we recapped the recent feature extraction technology in WC classification. A sector of machine learning algorithms are deep learning algorithms, usually used in WC classification. For example, the authors in [20] utilized a smart device to analyze water crystals from images. In their research, the features were represented in a temporal dimension, and fed to the deep learning model. Many models were constructed on pre-trained architecture, such as AlexNet and ImgNet. To compare the accuracy of these models, an open decision tree was well trained with the temporal data. The experiments exhibited good results with high accuracy in learnable ability to extract significant features that could distinguish water crystal cases.

In [21], an auto-encoder model of weighted auto-encoders and a Softmax classifier was presented. While the auto-encoders were used for computing the intrinsic data in solid state, Softmax was utilized to choose the features to auto-classify the cases. The precision of the model was assessed through experiments with two data groups. The performance indicated that the auto-encoder was suitable for the detection of WCs.

The WC detection model depended on the efficacy of the deep learning model in [22]. The image groups utilized solid-state records of water crystals taken by infra-red camera. The OpenSmile tool was used to get two types of features from solid-state records. The on-group features in the model presented in [23] were stored in the AVEC data group, with 2200 records, and the relevance score was used for feature extraction. In [24], the authors implied that features with maximum relevance values from labels induced better classification performance, while reducing redundancy. The feature group in [25] had 80 features from the crystal factor data group. Both feature groups were fed to supervised classifiers having five layers. Classification performance depicted that DAE had the best

accuracy among other models. An accuracy of 87% attained by the model outperformed the mean laboratory classification of value 73.8%.

Since WCs are caused by disarray, solid-state images are key indicators for stage classification. Another detection model for WCs using CNN was presented by the authors in [26]. In [27], solid-state image readings of 23 WC cases were input to a seven-layer architecture for the classification of WC disarray. The measures utilized precision metrics of accuracy and specificity, and this CNN achieved a performance with accuracy of 89.25%, sensitivity 85.71% and specificity of 92.37%.

The authors in [28] detected disarray described by the quality of the water crystals. The research in [29] used devices to capture data from 10 water crystal batches with disarrayed WCs. After attaining multiple solid-state parameters, they were allocated to classes by experts. Labeled case vectors were fed into deep learning phases. Their CNN classifier outperformed traditional learning methods.

Two important WC classification models are as follows: the first one is a deep learning model that employed an SVM machine with Chi-Square stochastic model [30]. This model extracts informative features from various feature groups. The second model utilizes axonal loss in crystal movement of WC cases by employing a supervised learning CNN [31].

For deep learning, class imbalance is defined if the used data group is unbalanced with the count of majority instances being higher than the count of minority class instances [17–19]. Class imbalance impacts the classification performance [30]. Many deep learning methods assume balanced data group distribution. Measuring the classification accuracy of classifiers in cases of data group imbalance requires better testing metrics. Accuracy and precision are usually used as evaluation measures in deep learning research. Nevertheless, for an imbalanced class distribution, accuracy can be a deceptive measure because the majority instances are allocated as the classification value for any instance [29]. Other measures that can quantify how sound a classifier is in its ability to differentiate among classes, even with imbalanced data, are necessary. Therefore, class-established metrics, such as shape factors, were chosen to compute precision in model evaluation in [31]. In [32], the authors proposed validation of the crystallography open database, using the crystallographic information framework, with high accuracy, but with lengthy CPU time. The authors in [33] proposed a neural network for lattice parameters to deduce monoclinic double perovskites. In [34], the proposed model called for ternary halide perovskites for possible optoelectronic applications using computerized support vector machines. The authors in [35] proposed a topological representation of crystalline compounds for the machine learning prediction of material properties with accuracy reaching 95%. Table 1 depicts different machine learning and deep learning models in the classification of crystal structures in different datasets.

Table 1. Summary of different machine learning and deep learning models to detect the crystal structures in different datasets.

Reference	Model	Dataset	Implementation	Training Time (Hours)	Classification Time (Seconds)	Accuracy%	Limitation
[22]	Crystal structure using SVM	Crystal structure images	SVM	16 H	64 s	75.5%	Images were of low accuracy with many false positives
[24]	Detecting crystal structure using machine learning	Captured infrared images	Neural network trained with histogram of features occurrence	25.5 H	30 s	82.4%	High training time

Table 1. Cont.

Reference	Model	Dataset	Implementation	Training Time (Hours)	Classification Time (Seconds)	Accuracy%	Limitation
[25]	Detection of crystal structure using IOT	Water road data	Sensors	No training, it is statistical process	Off time (not applicable)	74%	Data from sensors were impacted by weather condition
[26]	Crystal structure detection utilizing feature fusion	75,000 incidence of water crystallization and non-water crystals	Forest tree model	35.4 H	43 S	84.5%	Binary classification (water crystal or not)
[30]	A deep learning model for crystal structure prediction	Real crystal structure data	Deep learning ANN	29.5 H	38 S	85.3%	Data was unbalanced
[31]	Crystal structure detection using 3-dimensional CNN	Three-Dimensional crystal structure images	3D CNN	69.5 H	43 S	91.3%	Long training time
[32]	A crystal structure detection model	CAD-CVIS dataset	Deep learning method	14.5 H (low training time because of small datasets)	37 S	75.4%	Low accuracy because the dataset was small for machine learning
[33]	Solid-state water crystal model utilizing semantic segmentation	Water crystal large dataset	Attention model	Non applicable	133 S	90%	Time increased by increasing the data (non-extensible)
[34]	Water crystal structure detection in videos	Water crystal in videos	Object model	Feature extraction	Recognition time 500 S	87.4%	Long recognition time
Our proposed model	Solid-state water crystal structure detection in videos	Public dataset of solid-state water crystal structure	Feature Fusion and parallel neural network	12.5 H on average	12.5 S on average	96.7% on average according to feature fusion	Unidentified crystals were not recognized

3. The Dataset

The benchmark dataset contained water crystals delivered by the Emoto Project (EP). The crystals were collected from samples from several countries [9].

From each source, a sample of 60 droplets of 0.5 mL of water was collected.

The samples were then positioned randomly and kept at -22 to -32 °C. This guaranteed several temperatures.

The samples were then taken from the freezer, and were kept at -5 °C temperature. A water crystal image was captured utilizing an optical microscope at $200\times$ to $400\times$, subject to the occurrence and magnitude of the crystal.

The utilized dataset EP [16], contained 5000 crystal images. The 5K EP dataset comprised high-resolution photos (6072×4048 pixels). The WCs inhabited a minute portion of the iced images. We preprocessed each image to take out the image contextual boundaries. We utilized a subtraction algorithm [17,18] to express the boundaries of the water crystals. The smallest box that covered a water crystal was selected to eliminate redundancies. This reduced the data dimensions and kept the details. The dimensions of the water crystals were varied, so we resized all the input images to one size.

The augmented dataset was categorized into 13 classes with the highest frequency in the EP dataset. All instances in the dataset were labeled. We constructed a tree structure

of the categories in the EP dataset, as depicted in Table 2. We selected the most frequent crystal classes and labeled them. We partitioned the EP data into a 70% training subset, a 10% validation subset and a 20% testing subset. Scikit-learn Python was utilized to assure the data balance.

Table 2. Water crystal solid-state classes as defined in [16,17].







	Category	Crystal Example	Shape	Count of Crystals
1	Microparticle		Hexagonal plate	One
2	Simple particle		Hexagon with no outer boundary	One
3	Fan particle		Square with a fan boundary	One
4	Dendrite particle		Square with dendritic boundary	One
5	Fern dendrite particle		Square with fern-like boundary	One
6	Column		Column	multi crystal
7	Asymmetrical particle		Irregular boundary	Single
8	Cloud particle		Granular shape	One
9	Particle Combinations		Multiple squares with vertical overlap	Multiple

Table 2. Cont.

	Category	Crystal Example	Shape	Count of Crystals
10	Double-particle plate		Two stacked squares	Two
11	Multiple Columns		Multiple block columnar crystals	Multiple
12	Multiple Irregular particles		Multiple Irregular (not completely formed)	Multiple
13	Undefined particle		Types of water particles without crystals	

Data and Feature Groups

The data group was recorded by environmental experts. The data set we selected for our experiment had 4873 water crystals and unidentified crystal cases (4573 normal and 300 undefined crystal structure), as depicted in Table 3.

Table 3. Input EP dataset statistics.

	Training	Validation	Testing
Number of all samples	3411 (70%)	488 (10%)	974 (20%)
Unidentified crystal samples	150	50	100

Solid-state features were effectively utilized to evaluate water crystals cases and to screen progression. Shape and count of crystals are the normally utilized solid-state features in WC detection [17,18]. In the acquired images, features are named base parameters [16]. Solid-state parameters, namely solid-state wave features, are shaped with diffraction images from solid-state crystals as the main parameters. These parameters were selected with a Crys open package [15].

Crystal shape factors which simulate the features of a crystal disarray, are employed as a reliable feature selection model for solid-state crystals in several processes, such as solid-state identification [25], feature identification [26] and detection of WC shapes [6]. These extraction techniques utilize trilateral overlapping banks to incorporate crystal shape factors with spectral field partitioning. In WC research, crystal shape factors are used to detect rapid relapses in solid-state movement, like count, which is always affected by WCs [26]. There were 81 crystal features summarized, with the statistical features of thirteen crystal shape factors, such as average, standard deviation, crystals log and their derivatives, in [27–30].

Discrete cosine transform (WDT) is a projecting tool when taking decisions about digital crystals, especially with slight fluctuations. Specific features extracted by WDT from the solid-state basic features (F0) were applied for WC detection in multiple studies. The reason for selecting WDT features is to extract the deviation in solid-state samples [31]. Therefore, impulsive changes in the regularity of shapes in solid-state samples would be

identified. In data gathering, 9-level cosine coefficients are applied to solid-state crystals for selecting WDT features attained by the F0 log coefficient of the F0 outlining. This procedure yields 172 WDT features containing entropy, and fusion energy of the estimated factors.

Discrete cosine is a model utilized for parameter selection which has the benefit of the coefficient converting features in high quality using the solid-state shape. Q-factor is related to the crystal count, and its high value is extracted for crystals with multiple counts. It was assumed that I is the number of decomposition levels in [28–30]. WC cases can undergo distortions in solid-state crystals. Hence, the shape factors of the WDT in the utilized data group were fixed to the temporal features of the solid-state crystals. The WDT parameters were defined as follows: the value of the Q- parameter controlled the temporal distortion. Avoiding ringing in cosines, the R factor was required to be higher than 3.3. To find the highest accuracy the Q–r value pairs, at various levels (I) were investigated in the stated intervals, and the experiments raised 412 WDT features [29–35].

Solid-state crystal features were also utilized to study the impact of noise on solid-state features. These features were summarized as follows: Boundary shape, Geometrical ratio, Solid-state Excitation value and Mode features [31].

Details about features are depicted in Table 4. Before performing experiments with the proposed model, we employed data normalization to convert the numeric values into a normalized scale. Normalization is a data preprocessing procedure to control bias to high values. The feature group taxonomy was inspired by the taxonomy detailed in [34]

Table 4. Details of the Feature Groups.

Feature Type	Metrics	Formal Definition	Details
Base features	<ul style="list-style-type: none"> Crystal count parameters per unit Entropy Fluctuation 	density per unit Entropy: $16.8 \text{ cal}\cdot\text{K}^{-1}\cdot\text{mol}^{-1}$.	Normally utilized as solid-state features in WC detection to detect solid states
Crystal shape factors	<ul style="list-style-type: none"> Average Standard deviation Crystals shape coefficient and their derivatives 	mono- to dinitro derivative	81 crystal related features summarized with the statistical features of the thirteen crystal shape factors, used to detect rapid relapses in solid-state
Discrete cosine transform (F0)	<ul style="list-style-type: none"> DC coefficient F0 log coefficient of the F0 delineation 	DC coefficient is the coefficient with zero frequency in both dimensions	This procedure yields 172 WDT features
Q-factor discrete cosines	<ul style="list-style-type: none"> Q coefficient Redundancy Number of levels to Convert crystals in a high quality using solid-state crystal. 	fractional loss of energy per cycle Cosine functions oscillating at different frequencies	Q-factor is related to the solid-state captured features, and its high value is extracted for crystals with high wave counts
Solid-state Geometry	<ul style="list-style-type: none"> Boundary shape Geometrical ratio Solid-state Excitation value Mode 	Distance between atoms in the crystal	Solid-state vibration features also utilized for studying the impact of noise on solid-state vibrations.
Solid-state features	<ul style="list-style-type: none"> Solid-state density 	Number of water crystals per unit in solid-state	These features are extracted with Praat analysis open software

4. Methodologies and Feature Selection

Auto-encoders (AEs) are efficient methods to decrease dimensionality and produce structured representation from input. We employed an unsupervised learning model through back propagation to compute parameters. In Residual Auto-encoders, the output

is associated to network inputs. Deep learning encoder–decoder has double stages: encoder and decoder. The first stage encoder discovers the hidden representation in the input. The second stage decoder is fine-tuned to redefine the data.

From the input D with s samples, f factors, the auto-encoder extracts the hidden representation R and the decoder rebuilds the previous input D' from R , by optimizing the differences between D and D' for all the samples:

$$\min(P, P') \frac{1}{s} \left(\sum_i^s \|Dec_{P'}(Enc_P(D_i)) - D_i\|^2 \right)$$

For the auto-encoder:

$$Enc_P(D) = \sigma(Z * P) = H$$

For the auto-decoder:

$$Dec_{P'}(D) = \sigma(Z * P') = D$$

where, Enc_p is the encoder, $Dec_{p'}$ is the decoder, P and P' are the learnable parameters and σ is the sigmoid activation function. The last stage, Z is utilized as the computed vector of the dataset D . Z is used as an input to a fully connected (FC) layer to perform classification. We presented a novel auto-encoder to select the hidden features from the images.

The auto-encoder performed as follows:

- The images were fed to the convolution layer utilizing the inputs to compute the hidden representation.
- The output of the encoder was flattened and stored as a vector

The decode performed as follows:

- The inserted features of the input image were transformed.
- The CL facilitated CNN extraction of visual sampling of the images.

Therefore, we employed a convolution layer with strides in the auto-encoder. Smaller pooled images could yield a degradation in the gradient when training a dense neural network. With auto-encoders, the high count of the hidden layers makes it difficult to build the input image. To tackle that, we utilized the skip connection of ImageNet [19]. Skip connection uses vanishing gradient and lossless data by training the residual mapping. The residual value would added to the output and the model would be trained on the residual $R(x)$, as follows:

$$y = R(d) + d$$

We employed two types of residual block to construct the encoder block with pooling. The residual block contained three convolution layers with equal counts of output channels. The pooling layer reduced the sampling rate. This block down-sampled the data of the input vectors M_i by N , where i spans the columns of the input matrix, and each vector represented a column in the array. This block dealt with elements of the input as distinct channels and down-sampled the input matrix over time. The down-sampling rate was lower than the original sample rate by m K times. The down-sampling was performed by removing a $(m - 1)$ successive sample after each output sample. The convolution layers were trailed by the ReLU function. This scheme necessitated that the output had similar shape to the input. The resampling block had a similar scheme to the regular block. To augment the input at the ReLU layer, we utilized a 1×1 CL, trailed by normalization, to convert the input into the required form to perform addition. By experimentation, we established that employing three CL layers in each block created a better output, as depicted in Figure 1.

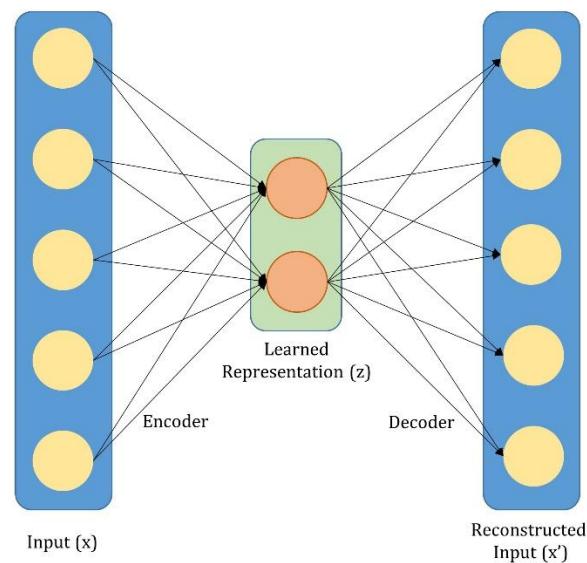


Figure 1. A residual encoder that selects parameters from the input images.

4.1. Fine-Tuning Model

The effectiveness of the classification was grounded on the impact of the selected features from the input images. The representative features allowed the predictor to obtain high output from the initial learning stages. The auto-encoder scheme was utilized to select features in supervised learning. Unsupervised learning attained high performance on high dimensional images. The model at that point started learning from with our whole dataset and extracted the best information. Nevertheless, the selection of the auto-encoder structure is a challenging task as it needs excessive information about learning from particular datasets. Therefore, we proposed a fine-tuning algorithm. The fine-tuning procedure used a CNN that was already trained for one process and employed it for a similar process.

Fine-tuning is usually trained on ImageNet [21] (with more than one million labeled inputs) by ongoing training on the initial dataset. AlexNet [22] is a large-scale CNN that performs well on the ImageNet model. AlexNet's performance is higher than all prior machine learning methods. ResNet [19] proposed a skip residual block-based model that permits back propagating to the primary layers without fading. ResNet earned top place in the ILSVRC 2015 race with a mean square error of 2.96%. SqueezeNet [24] attained the same precision as ResNet with less count of parameters. So, it was appropriate for mobile machine learning applications. DenseNet [25] is a dense CNN that enhanced the higher layer structures of prior CNN networks. DenseNet gave a solution by using all the layers: a layer is fed the input from prior output.

4.2. Classification Model

The features selected in prior steps were used as input to the classifier. The classification process had two main parts: feature selection phase (FSP) and classification phase. To construct the FSP, a pre-trained residual auto-encoder and Image prediction models were utilized. With the residual auto-encoder model, we retained the encoder from the residual auto-encoder, which was pre-trained with the EP database, as a feature selection phase. Similar to the ImageNet, the final layer was detached to extract the final features. The classification process had three FC layers which were utilized with the feature selector and were trained concurrently, using supervised training. The final classifier showing the encoder–decoder with skip connection is shown in Figure 2.

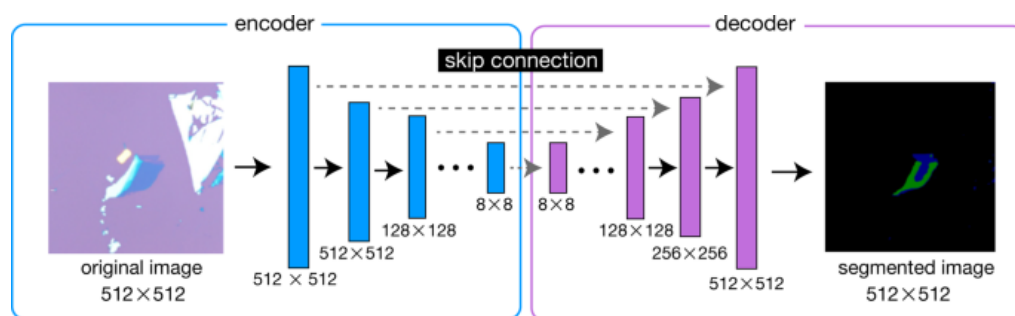


Figure 2. The final classifier showing the encoder decoder with skip connection.

In our model we did not perform learning from the start. We released primary and then utilized the CNN. A minimum learning rate was selected to guarantee that the classifier extracted the patterns from prior learned CL in the pre-trained CNN. For more testing and enhancement, we used various metrics to compute the performance of the various feature selection processes. The experimental results are depicted in Section 5.

4.3. Imbalanced Nature of the EP Dataset

Natural crystal formation is typically unbalanced. Therefore, the count of data items in each category was unbalanced. In the labeling process of the EP dataset, unbalanced categories were found. Some classes had 22% of the data, others had only 3%. The detailed description of the dataset is depicted in Table 5.

Table 5. The detailed description of the dataset.

Class	Count of Images	Ratio
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 assure higher precision in the process, we utilized the scoring algorithm by attaching scores for all classes, which added more weight for the smaller classes. The model would eventually learn from all categories similarly. All classes would be allocated scores matching the count of images in that class. The score was computed as follows:

$$Score_i = M / (C * m_i)$$

where $Score_i$, m_i , C and M are the score allocated to category i , the count of items in category i , the count of categories, and the whole number of photos in D . The experiments applied

F1-score and standard evaluation to test the model precision because these metrics are suitable with imbalanced data.

5. Experiments and Results

5.1. Evaluation Metric

We utilized performance metrics, such as accuracy, to test the classification model. For experimental setup, we fixed the count of classes to the count of actual classes that were utilized for labeling the input images. The accuracy for multiple classes was computed as follows:

$$Acc = \frac{1}{m} \sum_{i=1}^m O_i^{true} = O_i^{Predicted}$$

where O_i^{true} is the actual label, $O_i^{Predicted}$ is the output predicted class, and m is the count of images in the testing subset. The testing subset was not utilized in the training phase.

$$Sensitivity = \frac{TP}{TP + FN}$$

$$F1 - Score = \frac{2 * TP}{2 * TP + FP + FN}$$

TP is true positive, TN is the True negative count, FN is the false negative rate and FP is the false positive rate.

The dropout mechanism was the second technique employed to reduce over-fitting the input. In the learning process, the dropout module retained the environment function of predefined threshold, p , and the environment would be penalized and set to zero. Therefore, the dropout mechanism is a neural sampling technique, and the weights are only changed based on input values [35]. Since DAE studies the spatial correlation among neighbors, we had to alter the order of features in input to mine the connections among the features. We calculated the correlations based on model and feature levels. Then, we employed ordered clustering on the correlation values and clustered the correlated features. The features were reordered according to the clusters in the dendroid.

The classification process is usually modeled in feature-group discipline, influenced by the relevance of the computed features from the input. Nevertheless, this needs more work to acquire protruding features for extracting the hidden attributes [34]. Feature redundancy can yield to computational load in mining valuable information [29]. Machine learning models that have shallow layers cannot model high dimensional data [32]. On the contrary, deep learning techniques have been practical in WC classification, due to their pertinent generalization properties and to their noise tolerant features [23,24]. The aptitude to construct multi representation features and latent associations between data without supervision learning, marks DAE as a prominent option to feature-group models [33].

DAE can capture the inherent properties of the input, employing convolution followed by pooling. In recent research, DAE was employed as a feature selection procedure, preceding steps in training. It decreased the feature dimensionality through the pooling operation. In our research, DAE was utilized as a classification platform.

5.2. Evaluation

Evaluation was required to test the classification performance. Accuracy can result in misleading values in the case of unbalanced data classes. F-measure and sensitivity can distinguish among multiple classes, even in cases of unbalanced data. A confusion matrix articulates the true positives and negatives, as well as the false positives and negatives, for multiple classification. Sensitivity is a measure of the quality of multiple classifications. Sensitivity contemplates true positives and negatives, as well as false positives and negatives, and is considered a balanced metric for unbalanced input. Sensitivity depicts the correlation among labeled and predicted cases in the range ± 1 . A positive 1 is a faultless

classification, while a negative 1 reflects total dissimilarity between the classified and the actual cases.

5.3. Results

Herein we describe the details of the experiment setting output attained by our DAE model. We used a left-out mechanism where, in each iteration of the learning process, data items of one case were left out and utilized as testing input, while the remaining items of the other cases were employed in the training. The number of solid-state recordings per case was five recordings and the case label was computed by a majority function of class labels attached to the solid-states.

Snowflake Python was used to implement the experiments as the deep learning framework [35].

The hardware used was a SUN station with the configuration depicted in Table 6.

Table 6. The Sun station configuration.

GPU	16 cores each of 64 bits @ 3.5 GHz, 32 Gb RAM
CPU	Intel processors
Operating System	UNIX System

As stated, various features were joined in a feature group in the first DAE platform. While experiments were performed with only separate feature groups, the second experiments were performed by joining two, three and four groups of features, in turn. Accuracy, F-score and sensitivity were the scores utilized for testing performance. Table 7 depicts the classification results computed by a single group of features. Discrete cosine transform (F0) features had the highest performance in all measures. Joined features (JF1) that were the joining of base features, solid-state frequency and solid-state vibration tailed the performance measures of the F0 with an accuracy of 80.13% and 81.22% F-Measure. When sensitivity measure was utilized as a discriminative factor of the classification, we could determine that F0 and JF1 classifiers highly discriminated water crystal cases from WC cases.

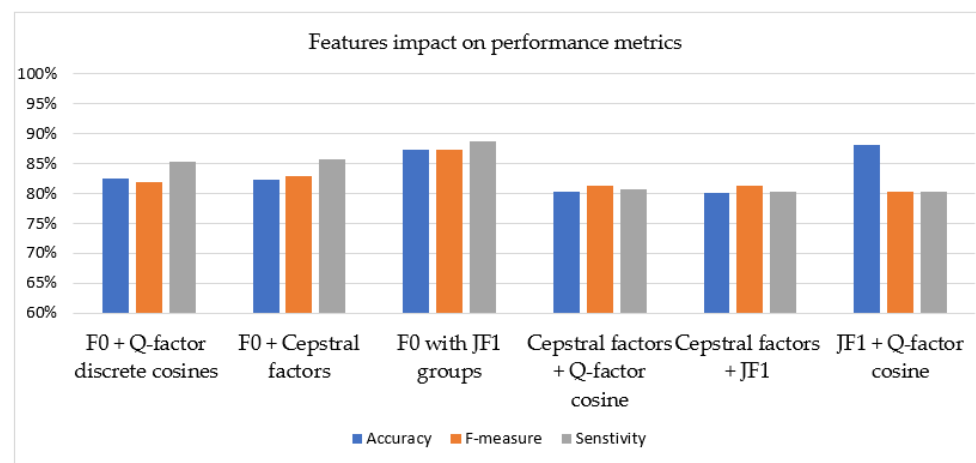
Table 7. Results of Single Feature Groups versus Joined features (JF1) that were the joining of base features, solid-state frequency and solid-state vibration tails for both the DAE and proposed models.

Feature Type	DAE			DAE with Fine Tuning		
	Accuracy	F-Measure	Sensitivity	Accuracy	F-Measure	Sensitivity
Base features	76.20%	76.27%	72.72%	96.40%	96.49%	92.92%
Discrete cosine transform (F0)	77.42%	72.02%	72.72%	92.52%	92.02%	92.82%
crystal shape factors	70.13%	71.33%	70.31%	90.12%	91.42%	90.41%
Solid-state vibration	78.46%	78.22%	73.76%	99.56%	99.42%	92.96%
Solid-state frequency features	76.22%	76.22%	74.76%	91.42%	90.42%	91.86%
Joined features (JF1)	80.13%	81.22%	80.41%	90.12%	91.42%	90.51%

After implementing the experiments with single feature groups, different joint feature group features were utilized. Joint features were fed to the primary DAE as inputs. The results of all permutation of feature groups joining are depicted in Table 8. The results exhibited the fact that the joining of the F0 and Q-factor discrete cosines had the same classification performance with the F0 + crystal shape factors pair. The DAE platform reached 0.925 accuracy with F-Measure of 0.920. Employing the F0 features with the Q-factor cosine or crystal shape factors also increased the sensitivity metric up to 0.95. Joining of F0 with JF1 groups had a good performance measure in terms of all three metrics scores. The accuracy of other feature joining (crystal shape factors + Q-factor cosine, crystal shape factors + JF1 and JF1 + Q-factor cosine) did not enhance at a rate less than 0.800, while their SENSITIVITY metric values were less than 0.740. Figure 3 depicts the results of all permutations of feature group joining.

Table 8. Results of double Feature group Joining for DAE Model.

Feature Type	Accuracy	F-Measure	Sensitivity
F0 + Q-factor discrete cosines	82.53%	82.02%	85.31%
F0 + crystal shape factors	82.33%	83.03%	85.83%
F0 with JF1 groups	87.36%	87.33%	88.76%
crystal shape factors + Q-factor cosine	80.33%	81.33%	80.76%
crystal shape factors + JF1	80.13%	81.33%	80.31%
JF1 + Q-factor cosine	88.13%	80.33%	80.31%

**Figure 3.** The results of all permutation of feature groups joining.

In the final experiments of feature group joining, triple groups were combined for further experiments. The results of the classification performance are depicted in Table 9. The joining of F0, crystal shape factors and Q-factor discrete cosines yielded accuracy of 0.933 and F-Measure of 0.929. The accuracy of the grouping of F0 + crystal shape factors + JF1 was below 0.91 and F0 + Q-factor cosine + JF1 joining were under 0.85. Joining without the F0 group (crystal shape factors + Q-factor cosine + JF1) demonstrated the worst performance in terms of all three metrics. Figure 4 depicts the performance results for triple feature groups for DAE with fine tuning.

Table 9. Results of Triple Feature Groups for DAE.

Feature Type	Accuracy	F-Measure	Sensitivity
F0 + crystal shape factors + Q-factor discrete cosines	87.33%	87.81%	86.31%
F0 + crystal shape factors + JF1	88.33%	88.03%	87.03%
F0 + Q-factor cosine + JF1	84.42%	84.33%	84.82%
crystal shape factors + Q-factor cosine + JF1	88.22%	88.22%	87.88%

We expanded our results using feature group joining, and double and triple groups were fed to the equivalent structures in the proposed DAE with fine tuning. The count of the features in the DAE was defined by the count of the feature groups utilized. Tables 10 and 11 depict the prediction performance from double and triple feature groups.

The experiments indicated that model joining with the double feature groups enhanced the performance according to the feature group combination. JF1 + Q-factor cosine had the higher performance improvement over the first platform, by 1.5%.

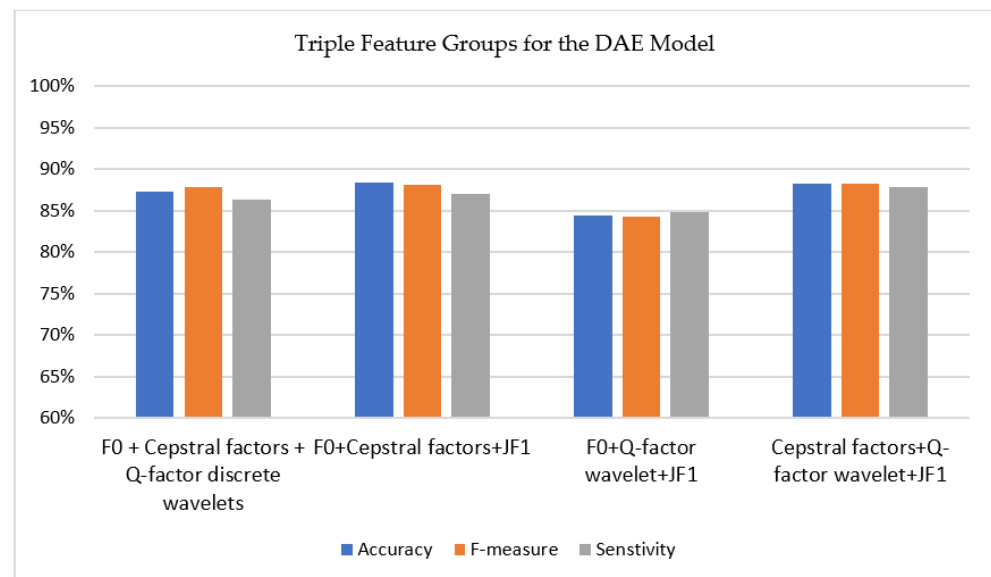


Figure 4. Performance results for triple feature groups for DAE.

Table 10. Model Joining: Double Feature Groups for proposed.

Feature Type	Accuracy	F-Measure	Sensitivity
F0 + Q-factor discrete cosines	95.99%	96.01%	95.91%
F0 + crystal shape factors	95.23%	94.03%	95.23%
F0 with JF1 groups	96.66%	96.63%	93.96%
crystal shape factors + Q-factor cosine	96.63%	96.63%	93.96%
crystal shape factors + JF1	91.13%	92.63%	91.61%
JF1 + Q-factor cosine	91.13%	91.93%	91.61%

Table 11. Model Joining: Triple Feature Groups for DAE with Fine Tuning.

Feature Type	Accuracy	F-Measure	Sensitivity
F0 +crystal shape factors + Q-factor discrete cosines	97.93%	97.02%	97.31%
F0 + crystal shape factors + JF1	95.17%	95.42%	96.53%
F0 + Q-factor cosine + JF1	97.06%	97.13%	97.16%
crystal shape factors + Q-factor cosine + JF1	96.22%	96.92%	93.54%

When the triple feature groups were used, model joining outperformed double joining. Among these joining, F0 +crystal shape factors + Q-factor discrete cosine joining obtained the best performance. This joining had a sensitivity value of more than 0.980, which proved the success of the discriminative classifier power. Although F0 + Q-factor cosine + JF1 and crystal shape factors + Q-factor cosine + JF1 models had the same accuracy and F-Measure scores, F0 + Q-factor cosine + JF1 was in the lead with a higher sensitivity rate. The joining of crystal shape factors + Q-factor cosine + JF1 had the least accuracy in all models.

The model joining enhanced the accuracies up to more than 4% in the triple feature groups than in the feature group results. In addition to the enhancements in the accuracy rates, there were obvious enhancements in the sensitivity values. The sensitivity rate of the F0 + crystal shape factors + JF1 joining outperformed the other joining. The least increase was attained in the crystal shape factors + Q-factor cosine + JF1.

5.4. Comparison with Other Models

Our experimental results were compared with two WC classification models. The first was a deep learning model that employed SVM machine with Chi-Square stochastic model [32]. This model was named Model-1 in our study, and extracted informative type features from the various feature groups. The second model (Model-2) utilized axonal loss in count movement of WC cases versus water crystals by employing a supervised learning CNN [33]. The model extracted wave features from the count movement disarray in water crystal cases. An EP dataset of 65 WC cases and 66 water crystals cases reciting the same solid-state aloud was utilized for extracting wave feature groups. Phoneme automatic segmentation yielded a group of solid-state features. These features were fed to binary neural CNN models. This CNN combined the random forest clustering technique. Their experiments deemed that 5 out of 7 features were statistically sound.

Table 12 depicts the comparison results achieved by only single feature groups for our proposed model versus Model-1 and Model-2. The results showed that our model with the F0 features outperformed the two models while they had comparable performance if we used the other features. When the experiments of the double feature groups were studied (depicted in Table 13), it was established that Model-2 had some advantage over our approach for the joint group of F0 + crystal shape factors feature. Model-1 and Model-2 had comparable results with our model when we used F0 + JF1 and crystal shape factors + JF1 groups. Otherwise, our model outperformed both models by a 5% margin, which established that our parallel structure model was the best with respect to similar state of the art models. Lastly, our classifier with triple feature groups was compared to Model-1 and Model-2 with high performance difference in favor of our model for all triple groups, as depicted in Table 14.

Table 12. Comparison of Accuracy of Model-1, Model-2 and our models (DAE and Fine Tuning) with Individual Feature.

Feature Type	Accuracy		
	DAE and Fine Tuning	Model-1	Model-2
Base features	96.40%	85.47%	85.33%
Discrete cosine transform (F0)	92.52%	82.34%	85.22%
crystal shape factors	90.12%	79.53%	79.77%
Solid-state vibration	99.56%	89.56%	89.76%
Solid-state frequency features	91.42%	86.43%	86.47%
Joined features (JF1)	90.12%	89.53%	89.77%

Table 13. Comparison of Model-1, Model-2 and our model proposed with Triple Feature Groups.

Feature Type	Accuracy		
	DAE and Fine Tuning	Model-1	Model-2
F0 + crystal shape factors + Q-factor discrete cosines	97.93%	91.07%	91.31%
F0 + crystal shape factors + JF1	95.17%	90.47%	89.03%
F0 + Q-factor cosine + JF1	97.06%	78.13%	76.36%
crystal shape factors + Q-factor cosine + JF1	96.22%	76.37%	73.04%

Table 14. Comparison of Model-1, Model-2 and our model proposed with Double Feature Groups.

Feature Type	Accuracy		
	DAE and Fine Tuning	Model-1	Model-2
F0 + Q-factor discrete cosines	95.99%	90.31%	89.41%
F0 + crystal shape factors	95.23%	91.03%	90.23%
F0 + JF1	96.66%	80.22%	80.99%
crystal shape factors + Q-factor cosine	96.63%	80.66%	74.96%
crystal shape factors + JF1	91.13%	74.63%	75.98%
JF1 + Q-factor cosine	91.13%	74.63%	70.41%

5.5. Time Performance

In our proposed model, the time complexity cost was advantageous because features were investigated and trained in parallel. The proposed model had the least training time cost and comparable classification time with the DAE and Model-1. The two models (Model-1 and Model-2) were more time costly in the training phase. Model-2 had the highest classification time. The CPU time is depicted in Table 15. The classification time was calculated in seconds.

Table 15. Computation time cost.

Model	Training Time (Hours)	Average Classifying Time (Seconds)
DAE and fine tuning model with two features	5.5	13.39
DAE and fine tuning model with three features	3.25	11.42
Model-1 [30]	16.85	112.5
Model-2 [31]	15.71	127.35

5.6. Discussion

It can be deduced from the experimental results that the proposed model predicted more than 98% of the samples accurately. From Table 11, it can be concluded that the accuracy of the model increased if there were three features fusion present for the model training. The accuracy metrics of the model proved that it predicted most of the solid-states water crystals predictions correctly by showing a true positive rate of 0.95, on average. On the other hand, the model predicted 95% true negative predictions, on average. Moreover, from Figures 2 and 3, it can be observed that the model predicted the different crystal structures more accurately, by 10% more, when we fused three features together than was the case for two fused features, thereby making the model's concept of combining more features robust and precise. Finally, an F-measure of 93%, on average, proved that the proposed model retained high precision and recall. To validate our results, and offer a comprehensive analysis for evaluating the proposed model, see Figure 5. The distribution of the different class ids is depicted in Figure 6.

These measures show that our proposed model attained accurate predictions of water crystals in twelve classes, which indicated that the proposed model was robust and attained high performance. Tables 12–14 demonstrate the comparison of the proposed model with deep learning models for water crystal classifier. It is clear from the tables that the proposed model outperformed the other two models by more than 7.8% in accuracy, reaching more than 9% accuracy in the triple feature group in favor of our model. The proposed model also needed less training time and classification time, as depicted in Table 15, and outperformed other approaches.

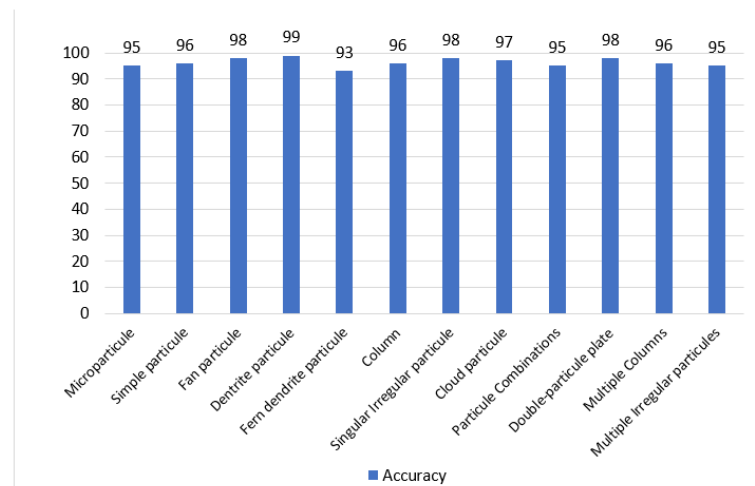


Figure 5. Classification accuracy percentage for the specific output classes.

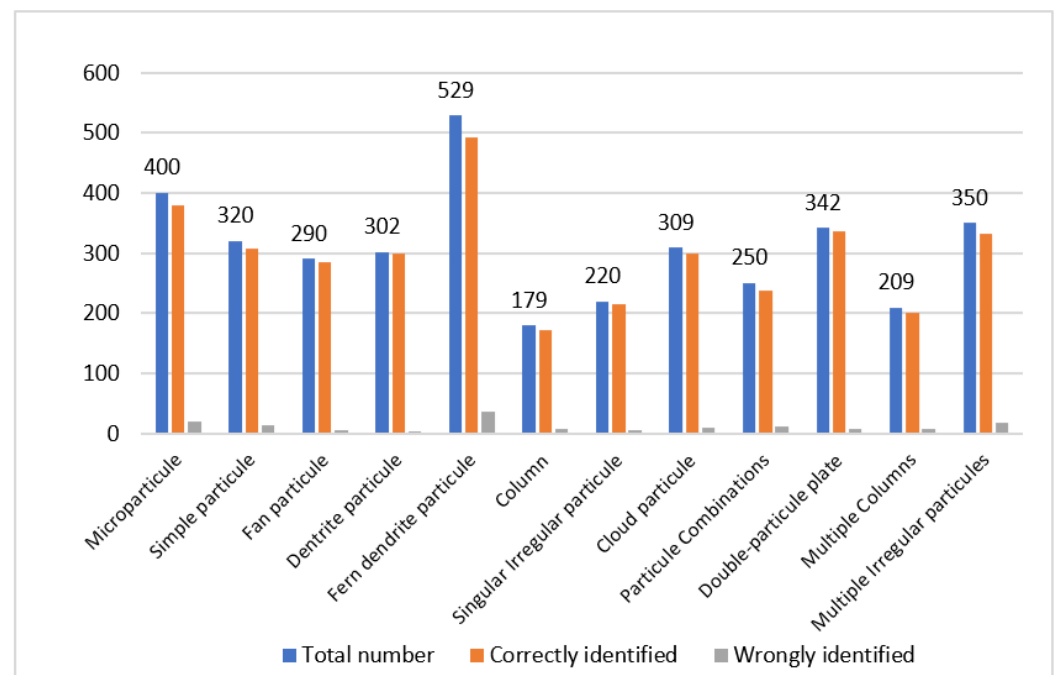


Figure 6. Number of total instances in each class for the specific output classes.

The classification model could be used to design a device to test the clarity of water from the solid state of water crystals with high precision and would display reliable performance.

6. Conclusions

In this paper, we proposed deep CNN models to predict water crystal shapes using a solid-state group of features. We constructed two platforms using deep learning to discriminate between water crystal cases from WC cases with high accuracy. In the first platform DAE, we introduced the feature group joining, and joined various feature groups before feeding to the deep learning DAE. In the second platform proposed, we fed different feature groups to the parallel structure. Then, the deep features were mined from the parallel structure branches at the same time in the training phase. We then merged those features and passed them to the later layers.

Both platforms learned from the EP dataset in the water crystals Learning repository [16]. The key characteristics of the proposed platforms are as follows:

The proposed parallel structure of the DAE produced increased accuracy and reduced time costs.

Each data item had five solid-state recordings per case with a take-out strategy to avoid the unfitting problem.

For future extension, we intend to utilize the parallel layers in the proposed model to apply to multi-modal data. We also aim to use data inputs from devices in WC classification with Long Term Memory models.

Author Contributions: Conceptualization, H.A.H.M. and N.A.H.; methodology, H.A.H.M.; software, H.A.H.M.; validation, H.A.H.M. and N.A.H.; formal analysis, H.A.H.M.; investigation, H.A.H.M.; resources, H.A.H.M.; data curation, H.A.H.M.; writing—original draft preparation, H.A.H.M.; writing—review and editing, H.A.H.M.; visualization, H.A.H.M.; supervision, H.A.H.M.; project administration, H.A.H.M.; funding acquisition, N.A.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R113), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Both platforms learned from the EP dataset in the WC Learning repository [16].

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

References

1. Boyd, C.E. *Water Quality: An Introduction*, 3rd ed.; Springer: Cham, Switzerland, 2020.
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](#)]
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](#)]
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 Proceedings of the International Conference on Artificial Neural Networks, Espoo, Finland, 14–17 June 2011.
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. Emoto, H.; Doan Thi, H.; Andres, F.; Hayashi, M.; Katsumata, K.; Oshide, T.; Tran, L. 5K EP Dataset 2021. Available online: <https://ieee-dataport.org/documents/5k-EP-dataset> (accessed on 1 January 2020).
17. 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. [[CrossRef](#)]
18. 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.

19. Tran, B.; Le Thi, H.A. Deep Clustering with Spherical Distance in Latent Space. In Proceedings of the International Conference on Computer Science, Applied Mathematics and Applications, Hanoi, Vietnam, 19–20 December 2019.
20. 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.
21. 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.
22. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
23. 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.
24. 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.
25. Goutte, C.; Gaussier, E. A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation. In Proceedings of the European Conference on Information Retrieval, Santiago de Compostela, Spain, 21–23 March 2005.
26. 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; Communications in Computer and Information Science*; Ivanov, I.I., van Sinderen, M., Leymann, F., Shan, T., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2013; Volume 367, pp. 3–20.
27. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
28. 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](#)]
29. Liu, J.; Wu, J.; Liu, Q.; Ji, S.; Zheng, X.; Wang, F.; Wang, J. Effect of the Strength of Initial Aluminium on the Bonding Properties and Deformation Coordination of Ti/Al Composite Sheets by the Cold Roll Bonding Process. *Crystals* **2022**, *12*, 1665. [[CrossRef](#)]
30. Vaitkus, A.; Merkys, A.; Gražulis, S. Validation of the Crystallography Open Database using the crystallographic information framework. *J. Appl. Crystallogr.* **2021**, *54*, 661–672. [[CrossRef](#)]
31. Zhang, Y.; Xu, Z. Machine learning lattice parameters of monoclinic double perovskites. *Int. J. Quantum Chem.* **2021**, *121*, e26480. [[CrossRef](#)]
32. Gómez-Peralta, J.I.; Bokhimi, X. Ternary halide perovskites for possible optoelectronic applications revealed by Artificial Intelligence and DFT calculations. *Mater. Chem. Phys.* **2021**, *267*, 124710. [[CrossRef](#)]
33. Jiang, Y.; Chen, D.; Chen, X.; Li, T.; Wei, G.W.; Pan, F. Topological representations of crystalline compounds for the machine-learning prediction of materials properties. *NPJ Comput. Mater.* **2021**, *7*, 28. [[CrossRef](#)]
34. Gunduz, H. Deep Learning-Based Parkinson’s Disease Classification Using Vocal Feature Sets. *IEEE Access* **2019**, *7*, 115540–115551. [[CrossRef](#)]
35. Snowflake/Python Software. Available online: <https://docs.snowflake.com/en/user-guide/python-connector.html> (accessed on 20 June 2021).