

Proceeding Paper

# Cloud-Base Height Estimation Based on CNN and All Sky Images <sup>†</sup>

Emanuele Ogliari <sup>1,\*</sup>, Alfredo Nespoli <sup>1,†</sup>, Elena Collino <sup>2,†</sup> and Dario Ronzio <sup>2,†</sup>

<sup>1</sup> Department of Energy, Politecnico di Milano, Via Giuseppe La Masa, 34, 20156 Milano, Italy; alfredo.nespoli@polimi.it

<sup>2</sup> Ricerca sul Sistema Energetico (RSE S.p.A.), Via R. Rubattino, 54, 20134 Milano, Italy; dario.ronzio@rse-web.it (E.C.); elena.collino@rse-web.it (D.R.)

\* Correspondence: emanuele.giovanni.ogliari@polimi.it

† Presented at the 8th International Conference on Time Series and Forecasting, Gran Canaria, Spain, 27–30 June 2022.

‡ These authors contributed equally to this work.

**Abstract:** Among several meteorological parameters, Cloud-Base Height is employed in many applications to provide operational and real-time cloud-base information to the aviation industry, to initialize Numeric Weather Prediction models and to validate climate models. Moreover, Cloud-Base Height is also useful in the nowcasting (very short-term forecasting) of solar radiation. As cloud movements mainly affect the solar irradiance availability, their characterization is extremely important for solar power applications; an accurate estimation of the ground shadowing requires the knowledge of cloud height and extent. In the present work, the Cloud-Base Height value is estimated starting from sky images acquired from a single All Sky Imager. In order to fulfill this task, a Convolutional Neural Network model is chosen and developed.

**Keywords:** convolutional neural network; CNN; all sky images; cloud-base height; machine learning



**Citation:** Ogliari, E.; Nespoli, A.; Collino, E.; Ronzio, D. Cloud-Base Height Estimation Based on CNN and All Sky Images. *Eng. Proc.* **2022**, *18*, 5. <https://doi.org/10.3390/engproc2022018005>

Academic Editors: Ignacio Rojas, Hector Pomares, Olga Valenzuela, Fernando Rojas and Luis Javier Herrera

Published: 20 June 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

Cloud-Base Height (CBH) estimation is an increasingly crucial task: this parameter is required in many applications [1]. It is exploited to validate climate models [2] and to improve Numeric Weather Prediction (NWP) models [3]; for instance, it improves the definition of the cloud-drift vector field, allowing a more effective modeling of the atmosphere dynamics [4]. Moreover, CBH is also useful in the nowcasting (very short-term forecasting) of solar resources [5] and photovoltaic power plant energy outputs [6]. As clouds are the primary cause of intermittency in solar irradiance, they are of interest for solar power applications: an accurate calculation of the ground shadowing requires the knowledge of cloud height and extent [7].

In the scientific literature, several studies are aimed at estimating CBH through different procedures. In [1], seven All Sky Imagers (ASIs) belonging to the Eye2Sky ASI network, located in the city of Oldenburg, are exploited to estimate CBH. In detail, an independent CBH estimation is derived considering each possible pair of ASIs, and then all the different estimations are properly merged into a final and more reliable one. In [8], the authors combined infrared satellite images and spectral information derived from meteorological sounders to improve the accuracy of the cloud height estimation task. In [9], a method for CBH estimation was developed starting from successions of images captured by ground-based imagers with a hemispherical view of the sky. In [10], a model capable of detecting and tracking individual clouds aimed at creating a 3D model providing cloud attributes such as height, position, size, optical properties and motion was developed. In [11], a newly developed temporal height-tracking (THT) algorithm to the backscatter profiles of two

ceilometers led to retrieved cloud bases that are statistically consistent with each other and ensured reliable detection of CBH, particularly when inhomogeneous cloud fields were present and changing rapidly in time. However, in situ measurements of cloud properties are essential, but they are quite expensive and typically limited in time and spatial location. On the contrary, Machine Learning-based models are capable overcoming physical model limitations, as for solar power plant energy estimation [12]. A CNN is a classification model specifically designed to detect patterns within images. Since the goal of the work was the evaluation of the possibility of using All Sky Imagers in order to detect the Cloud-Base Height, and considering the complexity of the cloud characteristics, this approach was considered ideal to approach this kind of problem. Therefore, the objective of the present work is to estimate, with a Machine Learning algorithm, the CBH value starting from sky images acquired from a single ASI. In order to fulfill this task, a Convolutional Neural Network (CNN) model was chosen and developed.

## 2. Convolutional Neural Network

The method selected to fulfill the objective of the current work is the so-called CNN, a classification model specifically designed to detect patterns within images. In the following, the model is first described from a theoretical point of view. Then, all the characteristics of the model implemented in the present work are discussed and explained.

### 2.1. General Description

CNNs are classification Machine Learning models, nowadays involved, for example, in image search services, self-driving cars, automatic video classification systems, etc. Moreover, their utilization is not restricted to visual tasks: they power many other applications such as voice recognition or natural language processing.

The structure of CNNs derives from the studies of the brain's visual cortex. Several studies and experiments demonstrated that neurons dedicated to vision present a small local receptive field, hence they process only information deriving from a limited region of the visual field. Moreover, the receptive fields of different neurons may overlap, and together they cover the entire visual field. This structure, capable of detecting complex patterns in any region of the visual field, inspired the researcher to develop a Neural Network architecture that gradually evolved into the current CNN.

In further detail, the typical CNN structure consists of a sequence of convolutional and pooling layers:

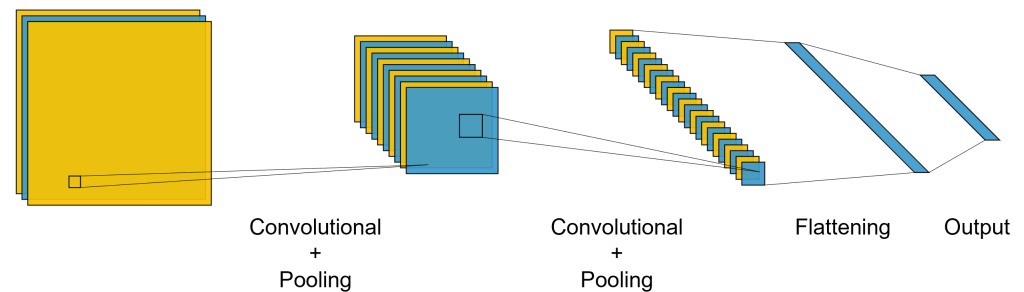
- The convolutional layer is the crucial building block of CNNs: neurons in this type of layer are not connected to every pixel in the input image but only to pixels in their corresponding receptive fields. The weight of a neuron is represented by a filter (or convolution kernel) that, when applied to the image, is able to extract features from it. During the training phase, the convolutional layer learns the best suited filters for a specific task.
- The pooling layer has the goal of subsampling the input image in order to reduce the computational load, the memory usage, and the number of network parameters to be tuned. As in convolutional layers, each neuron is connected to a restricted region of the previous layer. Moreover, neurons in this layer do not have weights: all they perform is the aggregation of the inputs according to a specific aggregation function, such as max or mean.

After being processed in the cascade of convolutional and pooling layers, the information flow is flattened, i.e., it is structured in a suitable format to be further processed. The last step of the classification process takes place in one or more dense layers, providing the final output.

### 2.2. Adopted Structure

The CNN structure adopted is represented in Figure 1. The combination between a convolutional and a pooling layer is exploited two times in order to grant a reasonably

deep feature extraction from the input images. Then, the information flow is flattened and delivered to the final dense layer, aimed at providing the output label corresponding to an unlabeled input image.



**Figure 1.** Adopted CNN structure.

Some characteristic parameters of the CNN structure need to undergo an optimization procedure in order to grant the best possible performance on the considered case study. In detail, a sensitivity analysis is carried out on the number of filters involved in the convolutional layers. The dimension of the dense layer is kept fixed at 16 hidden neurons. This study is aimed at selecting the configuration representing the optimal compromise between model accuracy and complexity. The results of the sensitivity analysis depend primarily on the amount of data involved in training and are reported in the results section.

### 3. Case Study and Available Data

The CBH is a crucial parameter in the characterization of the cloud features and consequently on the determination of the attenuation of solar radiation on the ground. Unfortunately, there are few instruments (ceilometers), which are very expensive, that are able to detect, in an objective way, this parameter. The alternative, considered in this work, is the exploitation of the atmospheric sounding obtained in the correspondence of some airports by means of a radio sounding. This measurement is made with low frequency (generally twice a day at 00 and 12 UTC) because of the weather balloon launch cost which hosts the battery-powered telemetry instrument. In the presented case study, the problem was also to have the radio sounding in proximity of the All Sky Imager. For this reason, the observed dataset was not too large and, therefore, it has been necessary to consider a reduced number of classes. In presence of a consistent dataset of CBH measurements obtained from a ceilometer, it would be possible to increase the number of the classes. Furthermore, in order to train a supervised Machine Learning model, a proper set of data is required. More in detail, it is necessary to have some target data, i.e., representing the quantity that the model is going to estimate, and some input data, i.e., the information on which the estimation is performed. As input data, sky images from an ASI are used. On the other hand, as target data, information derived from radio soundings is available.

#### 3.1. Input Data

All available sky images are acquired through a whole-sky cam and constitute the input to perform the estimation. The images cover two time periods, one comprised between the months of May and September 2020 and one between January 2014 and February 2015, and present an initial resolution of  $1124 \times 1124$  pixels. In order to reduce the computational burden of the estimation algorithm developed, the images are down-sampled to a  $256 \times 256$  pixels resolution.

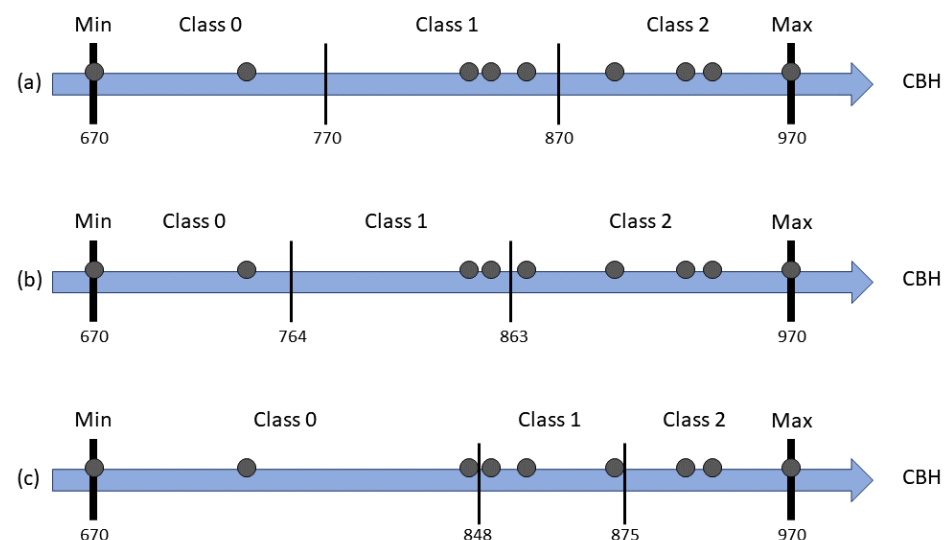
#### 3.2. Target Data

Target data consist of the CBH cloud cover depicted in each image. This value is computed starting from the Pressure of Lifted Condensation Level (PLCL) values, recorded through a radio sounding sensor. However, PLCL alone is not enough to properly estimate

the CBH value because the atmosphere pressure is not constant throughout the year. In order to address this issue, the difference between pressure at sea level and PLCL, representative of the height of the bottom layer of the cloud, must be used to estimate the CBH target value. In the presented case study, radio soundings are carried out once a day, at 12:00: therefore, the useful time frames for CNN training, i.e., those where both PLCL value and the corresponding sky image are available, are not numerous; only 109 sky images with a corresponding target label are available. The CNN is applied, in the present work, as a classification method, while the CBH is a continuous variable; this means it cannot be used as a target variable as it is. However, it was possible to divide the range of the registered CBH values into classes corresponding to a range of values and to train the model to assign each image to its corresponding class. Therefore, the available dataset is divided into 3 classes. The identification of more than 3 classes leads to a critical issue: the number of samples inside each group becomes too small, strongly affecting the classification accuracy. In that case, the algorithm would struggle in recognizing the classes because it does not have a sufficient number of examples to infer their characteristic pattern. On the contrary, if only 2 classes are defined, the range of CBH values corresponding to each class would become too large, reducing the usefulness of the classification.

CBH classes within the interval comprised between the maximum (970) and the minimum (670), can be defined according to different strategies leading to different partitioning of samples. The strategies considered in the current work, and the relevant thresholds, are represented in Figure 2 and are described as:

- Linear: the interval comprised between the maximum and the minimum registered CBH values is divided in 3 evenly spaced intervals (770 and 870).
- Logarithmic: the interval comprised between the logarithm (base 10) of the maximum and the minimum registered CBH values is divided in 3 evenly spaced intervals (764 and 863).
- Equal number of samples: the thresholds dividing CBH classes are set to make three classes with an equal number of samples (848 and 875); this latter partitioning strategy is defined in order to verify the results obtainable with balanced classes in terms of the same number of samples.



**Figure 2.** Different strategies for CBH classes definition and relevant thresholds: (a) linear; (b) logarithmic; (c) equal number of samples.

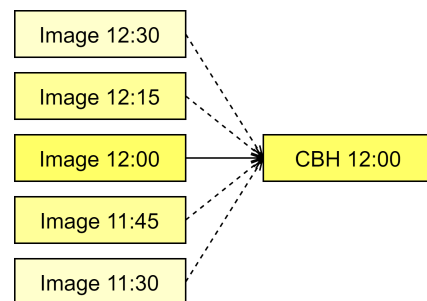
The different partitioning strategies lead to a different number of samples in each class, as reported in Table 1:

**Table 1.** Number of samples in each class according to different partitioning strategies.

Strategy	Class 0	Class 1	Class 2
Linear	39	66	4
Logarithmic	15	70	24
Equal num samples	39	36	34

### 3.3. Oversampling

CNNs, in order to be trained, require a large amount of input images. However, in our case study, the available dataset is not that large. In order to address this issue, an oversampling is performed in order to obtain additional generated input samples, useful to improve the training process of the model. Assuming a negligible time variation of the vertical profile of atmosphere, it is possible to assign to a single radio sounding all the images acquired in the surrounding time frames, as graphically represented in Figure 3.

**Figure 3.** Oversampling strategy adopted to enlarge the amount of training data available.

The impact of oversampling on class population is reported in Table 2:

**Table 2.** Samples per class of the adopted partitioning strategy after oversampling.

Strategy	Class 0	Class 1	Class 2
Linear	201	330	19
Logarithmic	73	360	117
Equal num samples	201	182	167

## 4. Results

The performance evaluation is a crucial step to assess the capability of the model to correctly identify the target classes. Here, the classification performances of a CNN are presented and discussed according to specific evaluation metrics after the class definition strategies previously listed in Section 3.2.

### 4.1. Evaluation Metrics

The evaluation metrics adopted in order to assess the model performances are presented in the following. The Global Accuracy ( $A$ ) measures “how good” a classification model is, returning the fraction of right predictions. It is calculated as in Equation (1):

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

where:

- $TP$  (True Positives) are samples correctly classified by the model as positive;
- $TN$  (True Negatives) are samples correctly classified by the model as negative;
- $FP$  (False Positives) are samples that the model incorrectly classifies as belonging to class  $C$ , while they belong to a different class;

- *FN* (False Negatives) are samples belonging to class *C* that are incorrectly classified as belonging to a different class.

Precision, also denoted as Positive Predictive Value (PPV), for a class *C* is calculated as in Equation (2):

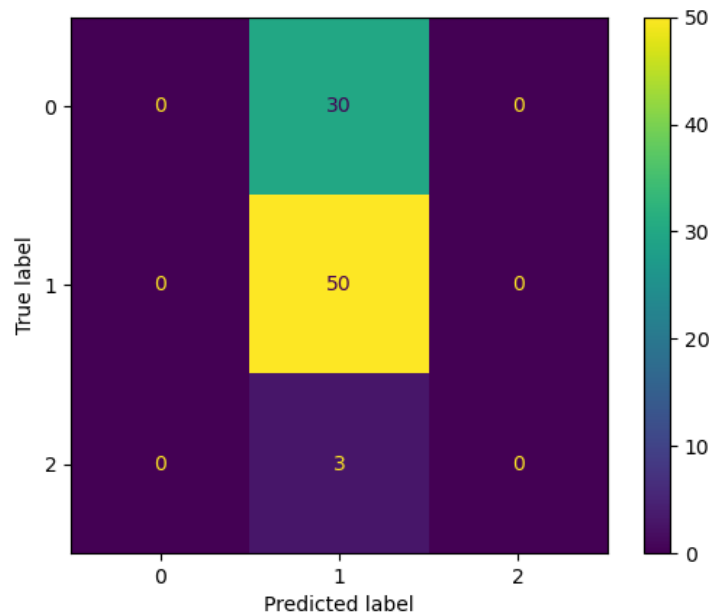
$$\text{Precision} = \text{PPV} = \frac{TP}{TP + FP} \tag{2}$$

Recall, also denoted as sensitivity, for a class *C* is calculated as it is stated in the Equation (3):

$$\text{Recall} = \text{Sensitivity} = \frac{TP}{TP + FN} \tag{3}$$

#### 4.2. Linear Classes Definition

In this case, the classes are defined according to the linear strategy, meaning that samples are defined according to linear partition strategies in the boundary thresholds of the classes, as it is previously described in Section 3.2. The sensitivity analysis carried out in order to identify the optimal network structure indicates the number of filters equal to 64 for both the first and the second convolutional layers. Figure 4 depicts the confusion matrix representing the classification performances in the considered case. A generic cell in row *i* and column *j* represents the number of samples belonging to class *i* that are assigned to class *j* during classification. Table 3 represents the classification performance evaluated through the metrics previously defined.



**Figure 4.** Confusion matrix for the Linear Classes Definition.

**Table 3.** Classification performances in the Linear Classes Definition.

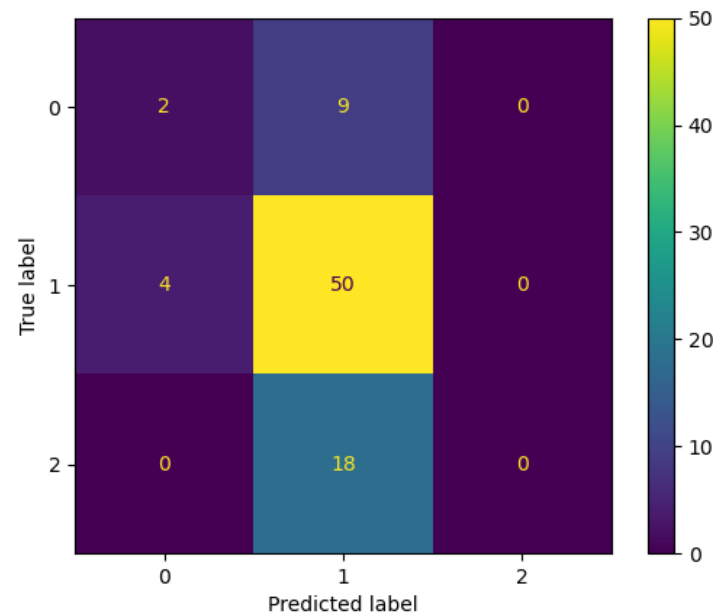
Class	PPV	Recall
0	-	0.00
1	0.60	1.00
2	-	0.00

In this case, the developed classification model is unable to recognize the presence of two classes (classes 0 and 2) out of three. As a matter of fact, the only class appearing in the model output is class 1, i.e., the most numerous one. The performance metrics numerically confirm the coherent results represented by the confusion matrix.

#### 4.3. Logarithmic Classes Definition

In this case, the classes are defined according to the logarithmic strategy. The sensitivity analysis carried out in order to identify the optimal network structure indicates the number of filters equal to 64 for both the first and the second convolutional layers.

Figure 5 depicts the confusion matrix representing the classification performances in the considered case. A generic cell in row  $i$  and column  $j$  represents the number of samples belonging to class  $i$  that are assigned to class  $j$  during classification. Table 4 represents the classification performance evaluated through the metrics previously defined.



**Figure 5.** Confusion matrix for the Logarithmic Classes Definition.

**Table 4.** Classification performances for the Logarithmic Classes Definition.

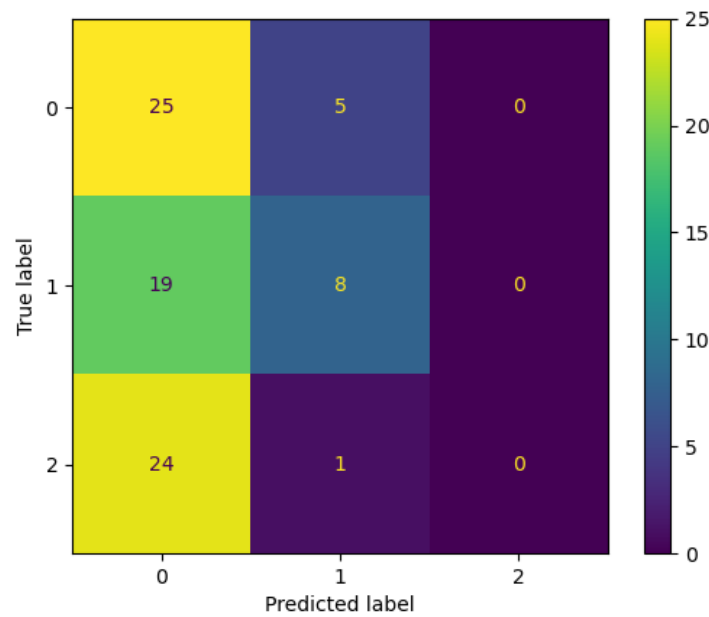
Class	PPV	Recall
0	0.33	0.18
1	0.65	0.93
2	-	0.00

Unlike the previous case, the CNN recognizes the presence of two classes (classes 0 and 1) instead of only one. A small number of samples are assigned to class 0, but only a couple of them really belong to that class. Evaluation metrics highlight coherently the small number of correct classifications in class 0.

#### 4.4. Classes with Equal Number of Samples

In this case, the classes are defined in a way that an equal number of samples belongs to each class. The sensitivity analysis carried out in order to identify the optimal network structure indicates the number of filters equal to 64 for both the first and the second convolutional layers.

Figure 6 depicts the confusion matrix representing the classification performances in the considered case. A generic cell in row  $i$  and column  $j$  represents the number of samples belonging to class  $i$  that are assigned to class  $j$  during classification. Table 5 represents the classification performance evaluated through the metrics previously defined.



**Figure 6.** Confusion matrix for classes defined with an equal number of samples.

**Table 5.** Classification performance for classes with an equal number of samples.

Class	PPV	Recall
0	0.36	0.83
1	0.57	0.30
2	-	0.00

This last case demonstrates that, even with an equal number of samples, the classification performances are slightly worsened compared with the logarithmic class definition. Once again, the CNN recognizes the presence of only two classes out of three and a large number of test samples are misclassified, as highlighted also by the performance metrics. Finally, the Global Accuracy (*A*) for the different class definitions together with the other evaluation metrics adopted in this work is reported in Table 6. In this study case, the strategy of the logarithmic partition of the samples within classes scores the best result in terms of global accuracy (0.63%), even if the limited amount of samples strongly affects the classification precision PPV. On the contrary, linear class definition shows the worst classification results with a global accuracy equal to 0.4, and it is unable to classify samples belonging to class 0 and class 2, while the Equal Amount of Samples strategy indicates almost comparable results (0.6) to the logarithmic partition of the samples within classes. Finally, samples in class 2 are incorrectly classified, allegedly due to the lack of samples belonging to that class, especially in the linear partition strategy.

**Table 6.** Classification performances for the different class definitions.

Class Definition Strategy	Linear		Logarithmic		Equal Number of Samples	
	PPV	Recall	PPV	Recall	PPV	Recall
Class 0	-	0.00	0.33	0.18	0.36	0.83
Class 1	0.60	1.00	0.65	0.93	0.57	0.30
Class 2	-	0.00	-	0.00	-	0.00
Global Accuracy	0.60		0.63		0.40	

### 5. Conclusions

The present work aims at estimating the CBH (Cloud-Base Height) value through a CNN (Convolutional Neural Network) model processing sky images acquired from a



single ASI. The CBH value for each of the training images is estimated starting from the PLCL (Pressure of Lifted Condensation Level) value recorded by radio soundings and the pressure at sea level. Moreover, the model output was not a specific CBH value but a class corresponding to a range of possible CBH values. In total, three classes were defined according to different strategies, namely: linear, logarithmic, and an equal number of samples partition in each group. In order to increase the number of available training samples, an oversampling procedure was carried out. The final best classification accuracy is 63% with the logarithmic classes definition. The reduced number of samples does not allow generalized conclusions to be drawn, even if the classification obtained with the logarithmic class definition seems to be the most promising. Future works will aim at adding more data and will combine the here-presented information contained in sky images with additional exogenous parameters (i.e., from sensors located in situ) to further improve the accuracy of the model.

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

**Funding:** This work has been partly financed by the Research Fund for the Italian Electrical System with the Decree of 16 April 2018.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

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

## References

1. Blum, N.B.; Nouri, B.; Wilbert, S.; Schmidt, T.; Lünsdorf, O.; Stührenberg, J.; Pitz-Paal, R. Cloud height measurement by a network of all-sky imagers. *Atmos. Meas. Tech.* **2021**, *14*, 5199–5224. [[CrossRef](#)]
2. Costa-Surós, M.; Calbó, J.; González, J.A.; Martín-Vide, J. Behavior of Cloud-Base Height from ceilometer measurements. *Atmos. Res.* **2013**, *127*, 64–76. [[CrossRef](#)]
3. Hogan, R.J.; O'Connor, E.J.; Illingworth, A.J. Verification of cloud-fraction forecasts. *Q. J. R. Meteorol. Soc. J. Atmos. Sci. Appl. Meteorol. Phys. Oceanogr.* **2009**, *135*, 1494–1511. [[CrossRef](#)]
4. Kassianov, E.; Long, C.N.; Christy, J. Cloud-base-height estimation from paired ground-based hemispherical observations. *J. Appl. Meteorol.* **2005**, *44*, 1221–1233. [[CrossRef](#)]
5. Rodríguez-Benítez, F.J.; López-Cuesta, M.; Arbizu-Barrena, C.; Fernández-León, M.M.; Pamos-Ureña, M.Á.; Tovar-Pescador, J.; Santos-Alamillos, F.J.; Pozo-Vázquez, D. Assessment of new solar radiation nowcasting methods based on sky-camera and satellite imagery. *Appl. Energy* **2021**, *292*, 116838. [[CrossRef](#)]
6. Nespoli, A.; Niccolai, A.; Ogliairi, E.; Perego, G.; Collino, E.; Ronzio, D. Machine Learning techniques for solar irradiation nowcasting: Cloud type classification forecast through satellite data and imagery. *Appl. Energy* **2022**, *305*, 117834. [[CrossRef](#)]
7. Nguyen, D.A.; Kleissl, J. Stereographic methods for Cloud-Base Height determination using two sky imagers. *Sol. Energy* **2014**, *107*, 495–509. [[CrossRef](#)]
8. Heidinger, A.K.; Bearson, N.; Foster, M.J.; Li, Y.; Wanzong, S.; Ackerman, S.; Holz, R.E.; Platnick, S.; Meyer, K. Using sounder data to improve cirrus cloud height estimation from satellite imagers. *J. Atmos. Ocean. Technol.* **2019**, *36*, 1331–1342. [[CrossRef](#)]
9. Savoy, F.M.; Lemaitre, J.C.; Dev, S.; Lee, Y.H.; Winkler, S. Cloud-Base Height estimation using high-resolution whole sky imagers. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 1622–1625.
10. Nouri, B.; Kuhn, P.; Wilbert, S.; Hanrieder, N.; Prah, C.; Zarzalejo, L.; Kazantzidis, A.; Blanc, P.; Pitz-Paal, R. Cloud height and tracking accuracy of three all sky imager systems for individual clouds. *Sol. Energy* **2019**, *177*, 213–228. [[CrossRef](#)]
11. Martucci, G.; Milroy, C.; O'Dowd, C. Detection of Cloud-Base Height Using Jenoptik CHM15K and Vaisala CL31 Ceilometers. *J. Atmos. Ocean. Technol.* **2010**, *27*, 305–318. [[CrossRef](#)]
12. Ogliairi, E.; Dolara, A.; Manzolini, G.; Leva, S. Physical and hybrid methods comparison for the day ahead PV output power forecast. *Renew. Energy* **2017**, *113*, 11–21. [[CrossRef](#)]