

Article

Artificial Intelligence-Based Detection of Light Points: An Aid for Night-Time Visibility Observations

Zuzana Gáborčíková¹, Juraj Bartok^{2,3}, Irina Malkin Ondík², Wanda Benešová¹, Lukáš Ivica²,
Silvia Hnilicová⁴ and Ladislav Gaál^{2,*}

¹ Institute of Computer Engineering and Applied Informatics, Faculty of Informatics and Information Technologies, Slovak University of Technology in Bratislava, Ilkovičova 2, 842 16 Bratislava, Slovakia

² MicroStep-MIS, Čavojského 1, 841 04 Bratislava, Slovakia

³ Department of Astronomy, Physics of the Earth, and Meteorology, Comenius University in Bratislava, Mlynská Dolina, 842 48 Bratislava, Slovakia

⁴ Institute of Medical Education and Simulations, Faculty of Medicine, Comenius University in Bratislava, Sasinkova 4, 813 72 Bratislava, Slovakia

* Correspondence: ladislav.gaal@microstep-mis.com

Abstract: Visibility is one of the key meteorological parameters with special importance in aviation meteorology and the transportation industry. Nevertheless, it is not a straightforward task to automatize visibility observations, since the assistance of trained human observers is still inevitable. The current paper attempts to make the first step in the process of automated visibility observations: it examines, by the approaches of artificial intelligence (AI), whether light points in the target area can or cannot be automatically detected for the purposes of night-time visibility observations. From a technical point of view, our approach mimics human visibility observation of the whole circular horizon by the usage of camera imagery. We evaluated the detectability of light points in the camera images (1) based on an AI approach (convolutional neural network, CNN) and (2) based on a traditional approach using simple binary thresholding (BT). The models based on trained CNN achieved remarkably better results in terms of higher values of statistical metrics, and less susceptibility to errors than the BT-based method. Compared to BT, the CNN classification method indicated greater stability since the accuracy of these models grew with increasing pixel size around the key points. This fundamental difference between the approaches was also confirmed through the Mann–Whitney U test. Thus, the presented AI-based determination of key points' detectability in the night with decent accuracy has great potential in the objectivization of everyday routines of professional meteorology.

Keywords: meteorological visibility; night-time observations; detectability of light points; convolutional neural network; explainable AI; aviation



Citation: Gáborčíková, Z.; Bartok, J.; Malkin Ondík, I.; Benešová, W.; Ivica, L.; Hnilicová, S.; Gaál, L. Artificial Intelligence-Based Detection of Light Points: An Aid for Night-Time Visibility Observations. *Atmosphere* **2024**, *15*, 890. <https://doi.org/10.3390/atmos15080890>

Academic Editor: Agnieszka Krzyżewska

Received: 30 May 2024

Revised: 18 July 2024

Accepted: 22 July 2024

Published: 25 July 2024



Copyright: © 2024 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

Knowing the exact visibility in all directions at the airports, ports, highways, etc. is the key to safe traffic operations. Low visibility due to weather changes represents a great risk, especially for the drivers of vehicles. As a result of their reduced ability to recognize objects, they are very often exposed to traffic accidents [1,2]. Meng et al. [3] also point to the difficulty of managing the logistics of flight operations at airports as a result of efforts to increase safety in conditions of low visibility. Beyond this, inaccurate airport visibility estimates increase maintenance costs for the airlines and can even lead to fatal aviation accidents [4].

On the other hand, visibility still remains one of the few meteorological variables, the observation of which is nearly unthinkable without trained and experienced human observers. Their work can be supplemented by equipment measurements. These, however,

are by far not perfect: the state-of-the-art measuring tools (e.g., transmissometers and forward scatters) are expensive, and at the same time, they are able to provide measurements in the instruments' location only. Bartok et al. [5] introduced a camera-based observation system that does not necessarily replace but effectively and synergically amends the standard observations. The paper presented human observations from a remote center that allows an observer to report meteorological conditions remotely, using images from cameras installed at the airport. Bartok et al. [5] concluded that for a correct estimation of the prevailing visibility, the standard near-the-runway automated sensors alone are inadequate; however, the camera-aided remote human approach to observations seems to be a promising supplement to eliminate the standard sensors' deficiencies in terms of quality (e.g., high-quality camera records; no more point measurements), objectivity (e.g., a database of archived weather situations), and efficiency (e.g., no need to have an observer physically present at the airport).

A further way to estimate visibility is on the basis of statistical models. They, compared to the equipment measurements, can be cheap in terms of the invested costs; nevertheless, they cannot be considered cheap in terms of the time and human resources necessary for their development and training. The statistical models cover a wide spectrum, starting, for instance, from a simple application of the Koschmieder law, through linear and non-linear models relying on different meteorological variables, up to the highly sophisticated models in the field of image recognition, constructed on the basis of machine learning and deep learning techniques [6].

In recent years, particularly due to the exponential boom in the field of AI technologies, a plethora of papers focusing on AI-based image recognition and/or classification have been published [7,8]. A considerable portion of these studies is based on convolutional neural networks (CNNs) and their innovated architectures, emphasizing their key capabilities such as versatility and robustness [9]. Chen et al. [10] concluded that 'From 2017 to the present, [...] CNNs have increasingly demonstrated irreplaceable superiority in image classification.' The target objects, and consequently, the particular technological equipment and the adopted approaches are manifold, such as medical image recognition [11], face recognition [12], the inspection of road material defects [13], street view image classification [14], or remote sensing image classification [15]. In meteorology and fields closely affected by meteorological conditions (such as road traffic and intelligent transportation systems [16,17]), the major challenge of the AI-based approaches is reliable object recognition in environments with poor visibility conditions, such as haze, fog, dust, dense rainfall, twilight, night-time, etc. [18,19].

Night-time visibility estimation is equally important in aviation and other forms of transportation, as planes, ships, and all forms of public transportation also operate during the night. Good night-time visibility is essential during landing and takeoff (e.g., guided by runway lights, approach lights, and beacons), and also in ground operations such as taxiing. Pedestrians and cyclists are also at higher risk during night-time.

Studies of night-time visibility or light point detection with graphical data as input are, however, often hindered by the lack of high-quality annotated data sets. One such data set was provided in the China Post-Graduate Mathematical Contest in Modeling in 2020 [20]. The data set contained meteorological data and observation video of one particular night (13 March 2020) and was utilized in several studies [20–23]. Nevertheless, beyond the common date, it is hard to track down further details on the data set since different authors refer to different target locations (Chengdu Shuangliu International Airport [22], Nanjing Lukou Airport [20], and unspecified site(s) [21,23]), different time intervals (from approx. 0:00 a.m. to 8:00 a.m. [20,22,23], and from approx. 0:00 a.m. to 12:00 a.m. [21]), and what is a pity, the only web address of the reference data set cited in [20] is no more accessible in 2024.

Quian et al. [21] performed multiple linear and multiple polynomial regressions to model the dependence between ground meteorological observations and reported visibility. Subsequently, the models were used to annotate the video data set split into separate frames

and trained AlexNet [24] to predict visibility. They report less than 0.25% percentage error in their approach in comparison with the ground truth. However, the test data set consisted of the last 129 frames with essentially the same visibility.

Fu et al. [22] used a support vector machine model to classify video frames into four visibility categories. They investigated the influence of the thermodynamic image algorithm and gray histogram for feature extraction on model performance. The reported performance for visibility between 400 m and 800 m contained about the same number of hits and misses for both data processing approaches. For other visibility ranges, both models performed significantly better.

Liu et al. [23] combined two CNN models VGG-16 [25] and Xception [26] to classify fog into four categories with an overall accuracy of 87.64%. However, the lowest visibility category 'thick fog' performed worse, possibly due to insufficient training data.

Chen et al. [20] used a Recurrent Neural Network (RNN) to analyze video sequences and predict visibility. They examined the influence of adding meteorological data as an input to the RNN. The mean relative error of the model decreased from 38.5% to 19.72% by incorporating ground meteorological observations such as temperature, air pressure, and wind speed.

The common features of the above referenced works [20–23] are (i) their focus on night-time conditions, (ii) visibility prediction by means of various AI approaches, and (iii) practically the same, limited data set of inputs. Even though the input figures therein are available with a 15 s frequency, it is questionable whether the video sequences of the length of 8 (12) hours offer sufficient variability of meteorological conditions that are precursors of the low visibility cases. The lack of representativeness is manifested at least by the fixed position of the camera, which does not allow for evaluating the visibility strictly from the aviation perspective, i.e., according to the ICAO (International Civil Aviation Organization) recommendations in all the cardinal directions [27]. The current paper, therefore, aims at developing a trustworthy model, which attempts to mimic the approach of professional observers, similarly to Bartok et al. [5]. Our strategy was also inspired by the study of Pavlove et al. [28] who developed the concept of the 'remote observer' towards more automatized ways of the visibility recognition using various approaches of artificial intelligence (two CNN models based on VNET [29]); however, it was for day-time conditions. It is important to emphasize that unlike the studies above, we are not trying to evaluate the visibility itself or the visibility prediction within different distance intervals relevant to aviation. The current study only deals with the very first (necessary) step towards the automatization of visibility prediction, and it is the analysis of the detectability of light sources in night-time conditions.

The paper is organized as follows: The international standards/requirements for visibility observation, and the camera-based system designed for it, are described in Section 2 (Materials and Methods). The same section also introduces the principles of the creation of the basic data set of camera images, characterizes the standard and novel approaches of image processing, presents the methods adopted for the automated visibility prediction, and finally, lists the statistics and metrics adopted to evaluate the performance of the constructed models. Section 3, after a short introduction related to the analysis of low visibility circumstances at the target location, presents the performance of the three constructed models for the prediction of night-time visibility. The first of the models predicts the visibility using a traditional approach (binary thresholding) with modified images as inputs, whereas the core of the remaining two models is an AI-based approach (convolutional neural network) that makes decisions both on the basis of original and modified camera images. The Discussion Section (Section 4) attempts to inter-compare and interpret the performance of the individual models, and underline their strengths and weaknesses, and the last section (Section 5) delineates some pathways to extend the current research in the future.

2. Materials and Methods

2.1. Definition of Visibility

Prevailing visibility is one of the most important meteorological variables at airports. The glossary of the International Standards and Recommended Practices, Meteorological Service for International Air Navigation (Annex 3) [27] adopted by the Convention on International Civil Aviation Organization (ICAO) defines the prevailing visibility as

The greatest visibility value, observed in accordance with the definition of ‘visibility’, which is reached within at least half the horizon circle or within at least half of the surface of the aerodrome. (These areas could comprise contiguous or non-contiguous sectors).

The visibility as defined for aeronautical purposes is the greater of the following:

- (a) the greatest distance at which a black object of suitable dimensions, situated near the ground, can be seen and recognized when observed against a bright background;
- (b) the greatest distance at which lights in the vicinity of 1000 candelas can be seen and identified against an unlit background.

Currently, reports of prevailing visibility are typically produced by a human observer at regular time intervals. Typically, (a) is applied during the day and (b) during the night.

In the case of determining visibility during the day, i.e., in daylight conditions, meteorological observers recognize specific objects and their outlines. In night-time conditions, however, the observers rely on the ability to determine the visibility of light sources. In the dark, when observing with the eye, instead of the entire visible scenery, the observer only focuses on light points or light sources. As a result, there is no information available on the edges of the target objects, which makes the AI-based prediction of night vision much more challenging.

2.2. Camera Observation Methodology

The general architecture of the camera-based observation system is as follows: A high-resolution camera with accessories is installed on a higher elevated point at the Poprad–Tatry Airport (Slovakia; ICAO identifier: LZTT) with an obstacle-free view. It sends 8 images of the horizon covering all the cardinal directions (N, NE, E, SE, S, SW, W, and NW) to a central server with a 5 min frequency (this periodicity is customizable). The rotation of the camera is provided by a rotator, which is assumed to always stop precisely in pre-defined positions after turning. If this does not happen (for instance, due to wind gusts or some mechanical failures), slightly shifted camera shots may result, and such camera shifts represent one of the several possible factors influencing the accuracy of automated visibility estimation.

As soon as a 360° camera rotation is completed, pre-defined distance markers (landmarks) are added to the individual camera images. With the help of these markers (by switching their statuses either to ‘visible’ or ‘obscured’), the remote (human) observer can determine the prevailing visibility and save the results to the server [5]. Figure 1 illustrates the results of such an evaluation both for daylight and night-time observations.

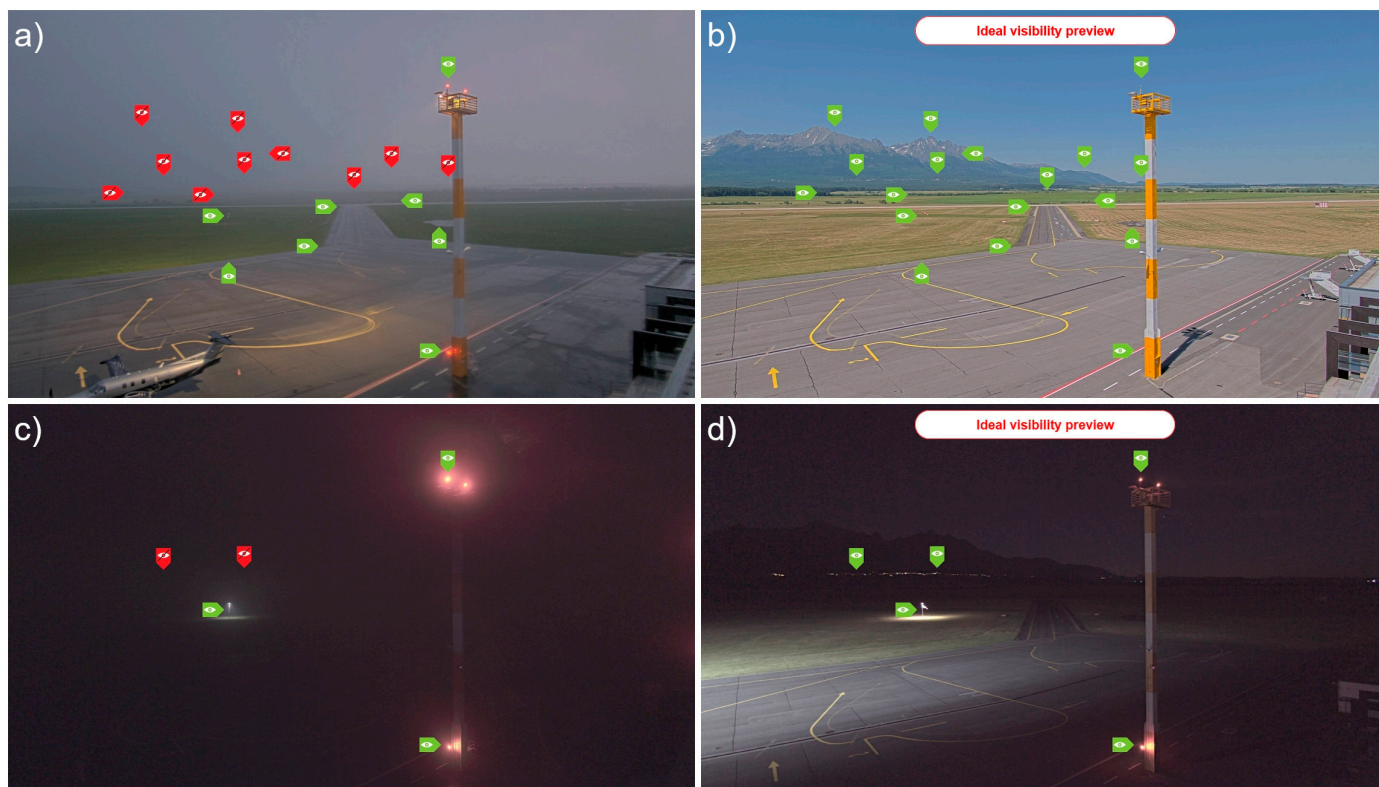


Figure 1. Adding markers to estimate the overall prevailing visibility both in daylight and night-time conditions—examples from the Poprad–Tatry Airport, in the northern direction. (a) Decreased visibility during the day, (b) ideal visibility during the day, (c) decreased visibility during the night, and (d) ideal visibility during the night. The green markers indicate visible targets, whereas the red ones denote obscured ones.

Landmarks with distance labels for all the directions represent an aiding tool for the remote observer. They have to be selected carefully to cover a variety of distances in each direction. Table 1 shows some basic statistics of the landmarks at the Poprad–Tatry Airport where the marker counts were divided into four distance categories used by the ICAO [27]. Furthermore, Figure 2 presents the spatial distribution of the markers in the vicinity of the airport, with the focus on the nearest ones, within the two circles with radii of 0.6 and 1.5 km, corresponding to Table 1.

Table 1. The number of landmarks used to identify directional visibility (regardless of whether they serve in daytime and/or night-time conditions) by the camera-based system at the Poprad–Tatry Airport, stratified by the ICAO distance categories.

Distance Interval [m]	Number of Landmarks
0–600	56
600–1500	26
1500–5000	30
>5000	41
Total	153

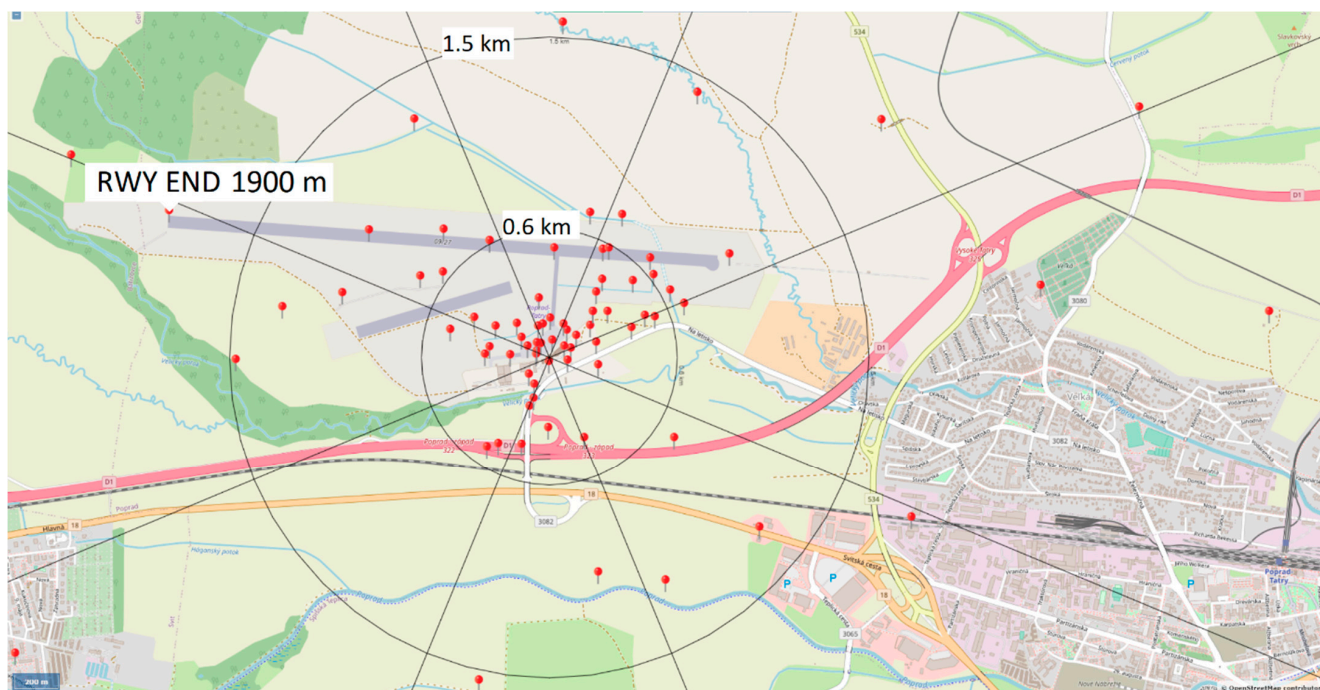


Figure 2. Spatial distribution of the landmarks with distance labels used to identify directional visibility (regardless of whether they serve in daytime and/or night-time conditions) in the vicinity of the Poprad–Tatry Airport.

The basic principles of visibility estimation based on the set of markers assigned to the camera images are as follows:

1. If all the markers are visible in a given direction, then the visibility is larger than the distance of the most distant marker in this direction.
2. If some markers are not recognizable in a given direction, then the visibility is determined by the distance of the nearest visible marker preceding the first obscured one.

These principles mimic the procedure that is required of a professional aeronautical observer.

The evaluation of the camera images by a remote (human) observer serves as the ground truth in the analysis of the automated approaches to night-time visibility estimation. Such an evaluation accompanies each camera image, usually in a text file of an XML format. The associated text file involves information on the light sources (their approximate coordinates and distance from the camera), information on the date and time of the camera shooting, information on the quality of the individual camera images, and, as mentioned, the expert decision of the human observer on which of the light sources are visible and which are not.

2.3. Classical Methods in Image Processing

The cornerstone of the classical methods of image processing (e.g., texture analysis, pattern recognition/classification) is the detection and description of its key points, and parallel to this, the definition of the adequate descriptors that characterize the structure of the image. Herein, a similar approach is adopted, with (a) the detection and description of the key points, and (b) the morphological reconstruction of the original images. Both procedures will be described in detail in the following two subsections.

2.3.1. Key Point Detection and Description

In the current study of night visibility, light spots represent the key points. We attempted to identify and describe them by means of some standard, widely accepted approaches, i.e.:

- SIFT (Scale-Invariant Feature Transform) [30];
- BRIEF (Binary Robust Independent Elementary Features) [31];
- ORB (Oriented FAST and Rotated BRIEF) [32].

Our preliminary analysis indicated that out of the above-listed approaches, the SIFT method yielded the most acceptable results.

A further step in the standard approach of image processing is feature matching. Its goal is to find matches between the descriptors of the recognized key points in two or more tracked images. From the wide palette of the available methods of feature matching, we used the Brute Force method [33]. Finally, we adopted the RANSAC (Random Sample Consensus) [34] algorithm to search for homography and then filter out incorrectly matched pairs of image features.

However, the preliminary testing of this approach revealed that it was unable to identify a significant number of key points, predominantly due to blurred images or a significant presence of fog. Conversely, the application of image sharpening filters led to excessive noise. In some cases where the points were correctly matched, we encountered inaccuracies in the image overlay process. We, therefore, arrived at a conclusion that the standard methods of key point detection and description are unsuited for the identification of light spots in our data set. As a viable alternative to this, we turned our attention to methods of image modification as tools for a binary identification of light spots as detectable or undetectable in a relatively straightforward way.

2.3.2. Image Modification

The term ‘mathematical morphology’ is used to classify methods to extract the useful features of an image that describe the shape of the examined image patches [35]. A specific case of the application of mathematical morphology is morphological image reconstruction. It is used to extract important features from an image by removing noise, highlighting edges, and filling gaps. It uses two inputs: (i) the input image itself to be modified, and (ii) the structuring element (or structural element or kernel), which is usually a small image (e.g., 3×3 or 7×7 pixels), and serves as a pattern that defines the details of the effect of the operator on the input image.

Image reconstruction is performed in an iterative manner while trying to preserve important features from the original image. There are several algorithms commonly used in morphological image reconstruction, and one of them is erosion [35,36]. Erosion is a morphological operation that reduces the boundaries of a white object in a binary image by removing pixels from its edges. Thus, from a technical point of view, the process of morphological erosion is a unification of the input image and the structuring element.

From the practical point of view, morphological erosion was applied on each image patch with a light spot. As mentioned above, the most important feature of the method is its ability to remove noise and highlight important elements in the original image (mostly light spots). Beyond this, the morphological erosion also filtered out, to some extent, the background of the light spots.

Different sizes of the structuring element were used in our preliminary testing; however, the lowest noise and/or gray artifacts were detected at a value of 7, which then was further used.

From now onward, we will term the results of the processing algorithm by the morphological erosion as ‘modified’ images or image patches, in contrast with the ‘original’ ones.

2.4. Binary Classification of Detectability of Light Spots

Having the original images modified by morphological erosion, the overall task of the prediction of meteorological visibility is reduced to a task of binary classification of the light spots, i.e., to decide whether they are detectable (equivalent to ‘visible’, class ‘1’) or, conversely, undetectable (equivalent to ‘obscured’, class ‘0’). In line with this concept, in the remainder of the paper, we will use the terms ‘prediction of detectability’ in relation to the approaches of the automated detection of light spots from the camera images, and ‘ob-

servation of visibility' in relation to the activities of human observers at the meteorological observatory (ground truth).

To determine whether the light spots are detectable or undetectable, two conceptually different approaches were adopted: a traditional one (binary thresholding) and one making use of a concept of machine learning (using a convolutional neural network).

2.4.1. Binary Thresholding

Binary thresholding (BT) [35] is a segmentation method used in image processing. It converts black-and-white or gray images into binary ones, containing only pixels with black or white color. It works on a simple principle in which each pixel in the image is replaced by a black pixel if its intensity value is lower than a specified threshold value, and conversely, the pixel is substituted with a white pixel if its intensity value is higher or equal than the threshold. An image is classified as 1 (visible, detectable) if there is at least one white pixel in the binary image. Otherwise, when the image contains only black pixels, it is classified as 0 (invisible, undetectable).

2.4.2. Convolutional Neural Networks

Convolutional neural networks (CNNs) [37] are one of the most important deep learning methods used to solve problems in the field of computer vision. Convolutional neural networks were inspired by the structure of the visual system, specifically drawing inspiration from the models proposed by Hubel and Wiesel [38]. Convolutional neural networks constructed in this manner have proven to be effective in processing two-dimensional data with a grid-like topology, such as images and videos. These layers were designed for feature extraction by applying convolution to the input data.

Convolutional neural networks are typically composed of the following types of neuronal layers [39]:

1. Convolutional layers. Convolutional layers are used for feature extraction by combining the input of each neuron with the local receptive field of the previous layer [40]. The goal of this process is to discover meaningful internal representations of objects and signs in the image. Voulodimos et al. [39] emphasize the importance of applying convolution to the entire image and activation maps or feature maps using kernels of different sizes. This approach leads to the generation of diverse activation maps that highlight local flags, thereby facilitating the detection of their interconnections.
2. Pooling layers. The essence of pooling layers is the mapping of extracted symptoms [39]. They reduce the spatial dimensions of the input image for another convolutional layer. Simultaneously, the dimensions of width and height are reduced, while the depth remains unaffected. This is commonly referred to as downsampling since size reduction in the input data leads to a loss of information. This loss leads to computational overhead for other layers, and also acts against information overload.
3. Flattening layer. This layer converts multidimensional arrays (output from previous layers) to a single linear vector, a format suitable for dense layers.
4. Dropout layer. The dropout layer is designed to prevent the overfitting of deep learning models. It randomly selects inputs and sets them to 0 with a customizable frequency called the dropout rate.
5. Dense or fully connected layers. In CNN architectures, fully connected layers are applied after several convolutional and pooling layers. These fully connected layers serve as the basis for the final classification of the input images [39]. In these layers, neurons are fully connected to all the activations in the previous layer. Thus, their activation can be calculated by matrix multiplication, followed by a deviation correction. Finally, the fully connected layers convert the obtained 2D feature maps to a 1D feature vector.

Herein, a convolutional neural network was proposed as the other approach to a binary classification of the detectability of light points. As part of the proposed approach,

we built and trained a model of a CNN with the simplest possible architecture of this network (Figure 3).

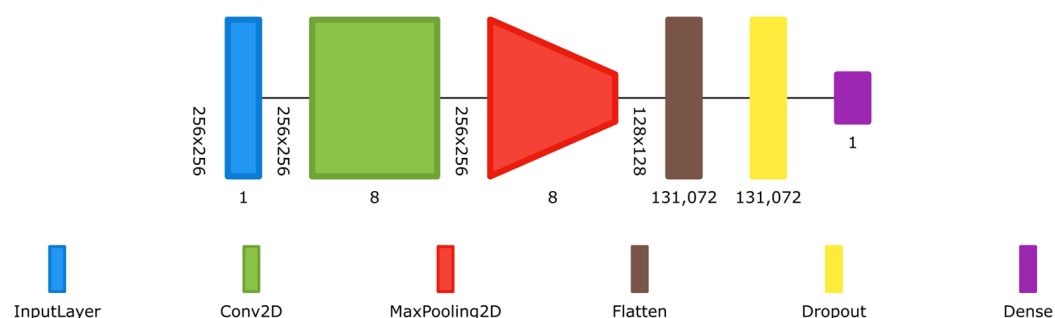


Figure 3. Architecture of the convolutional neural network used in the study. The schematic was created by means of the Net2Vis tool [41].

The architecture consists of one convolutional layer with 8 filters of size 3×3 , followed by a max pooling layer with a 2×2 pool size. The output of the convolutional and pooling layers is flattened into a 1D vector, and then passed through a dropout layer to reduce overfitting. Finally, a dense layer with a single neuron and sigmoid activation function is used for binary classification. This architecture is simple yet effective for tasks involving image data with binary outcomes. More specific details about these layers are provided in Appendix A.

2.5. Data Set Creation

2.5.1. The Initial Data Set

The measurements of visibility took place at the professional meteorological observatory at the Poprad–Tatry Airport (Slovakia). Both the database of the images from the camera observation system (available with a 5 min frequency in 8 cardinal directions; see Section 2.2), and the evaluation of visibility by professional human observers are available for a 39-month long period, starting with January 2018 and ending with March 2021.

2.5.2. Relevant Sampling

To create a relevant data set, one needs the records of light points found in photos that evenly cover all the periods of the year. The original data set, however, is not balanced since it contains uneven amounts of photos from different months of the year. For this reason, we attempted to pseudo-randomly select a certain amount (~1000) of records for each month over the approximately three-year period to achieve data set relevance. The principles of the procedure are as follows.

The governing rule of the selection was to keep the balance of the selected data set in terms of the visibility/invisibility of the light points. This initial data set, previously estimated by professional human observers and regarded as the ground truth, was available as a collection of XML files. According to the number of records in the minority group (i.e., those with the label '1'), the same number of records was selected from the majority group (with the label '0'), thus the new data set was created in a 1:1 ratio in terms of the visibility of light points. As a result of processing XML data from the initial data set (see Section 2.2), an intermediate data set was created with the number of records of approximately 12,000. Nevertheless, this data set had to be further refined/reduced for one of the following three reasons: (a) there was no camera image available to be assigned to the records from the intermediate data set, (b) although the corresponding camera image existed in the intermediate data set, it was either corrupted or was not a black-and-white image (i.e., taken, for instance, during the dusk or dawn), or (c) the overall goal was, again, to reach a balanced status of the data set in terms of visibility of light points by eliminating some records from the intermediate data set. This procedure resulted in the final data

set with about 7000 items; more precisely, it consisted of 6892 camera images and the corresponding text/numeric information.

2.5.3. Choice of Sizes of Image Patches

The original camera images were processed by means of the concept of key point images with a default setting, where the individual key points were approximately centered within the square-shaped cut-out of the original image. After experimenting with different sizes of the neighborhood, we have chosen 256×256 pixels, which is robust with respect to camera shifts and includes both a light spot and its sufficiently small surroundings.

An example of a night-time image is presented in Figure 4a. It was taken on 30 August 2019, at 3:00 a.m., at the Poprad–Tatry Airport. The camera, pointing towards the northern direction, shows the silhouette of the High Tatras Mountain in the background, with some night-time public lighting in the villages at the foot of the mountains. In the foreground, there are two significant objects for visibility estimation: the wind bag and the mast, both illuminated at night. All these points in the background and the foreground might be identified as key points using the procedures described in Sections 2.3.2 and 2.4. Figure 4b emphasizes one of the key points (the wind bag) cut out of the original image with its neighborhood, delineated according to the default settings. Figure 4c serves as an illustration of a modified image, which is the result of the morphological erosion adopted in Figure 4b with the aim of highlighting the light spot in the image. Finally, Figure 4d is the binary image created on the basis of Figure 4c.

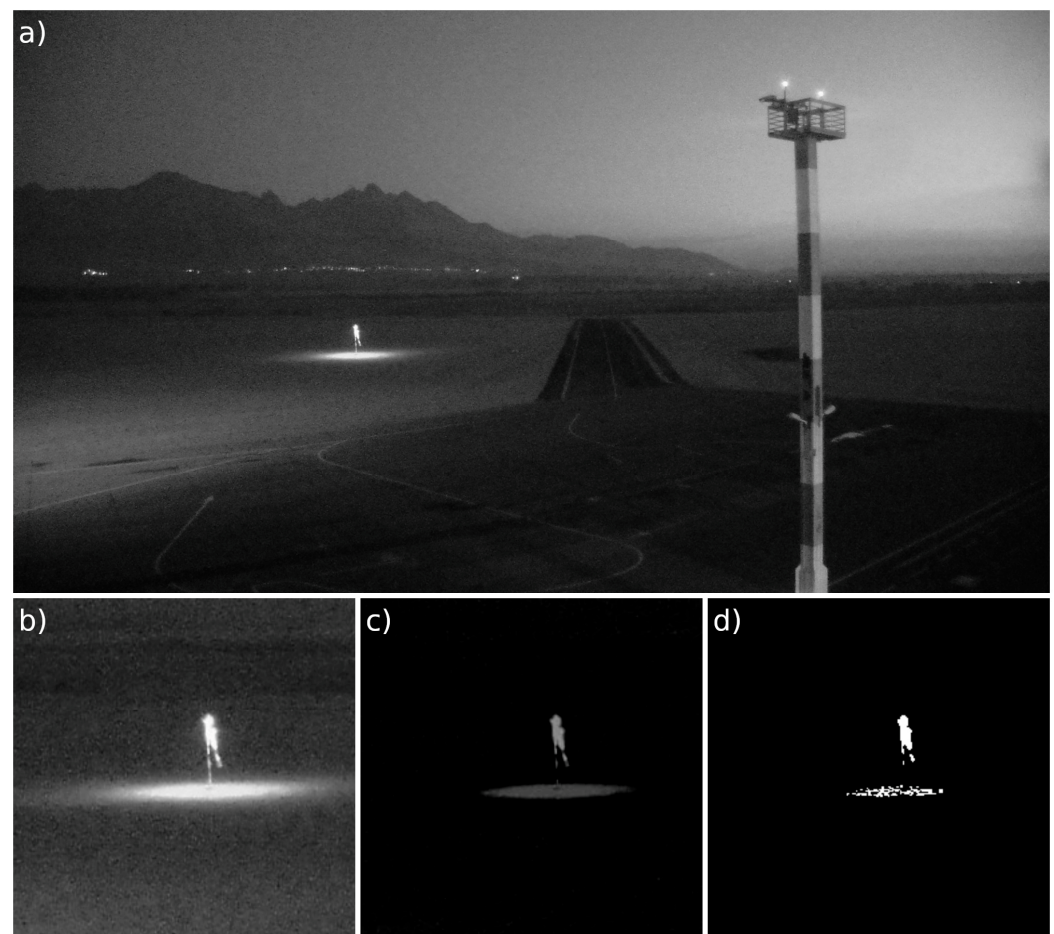


Figure 4. (a) An original camera image, taken at the Poprad–Tatry Airport (30 August 2019, 3:00 a.m.) in the northern direction. (b) An example of a key point neighborhood around the illuminated wind bag, as identified in (a). (c) Key point neighborhood as a result of the modification of (b) by morphological erosion. (d) Key point neighborhood in the form of a binary image.

To evaluate the effect of the default settings of the patch size on the final outcomes of the analysis, an experiment using 15 different neighborhood sizes was also conducted. Herein, various neighborhood sizes were defined, ranging from 30 to 450 pixels, with a step of 30 pixels (i.e., a square patch of the size of 30×30 , 60×60 , 90×90 pixels, etc.). In the end, for each of the ~7000 key points from the final data set, 16 different image patches were prepared, including the default size (256) and an additional 15 from the described experiment (from 30 to 450 pixels). The same procedure was applied both on the ‘original’ and the ‘modified’ images.

2.6. Verification Methodology

Widely known scores were used to evaluate the performance of the proposed models for classifying the detectability of the key points and their neighborhoods. The assessment started with the standard 2×2 contingency table, which reflects the goodness of the given constructed model in the light of the observations (ground truth):

- True Positives (TP)—They indicate the number of cases when the key point is visible according to the human observers, and also the model-based prediction.
- False Negatives (FN)—The key point in the image is visible, but the model finds it undetectable (i.e., type II error).
- False Positives (FP)—The key point in the image is invisible; nevertheless, the model declares it detectable (i.e., type I error).
- True Negatives (TN)—The key point in the image is invisible, and it is correctly predicted by the model as undetectable, too.

Based on these scores, further, more complex statistical metrics are defined [42]:

- Accuracy (A)—It is the ratio of the number of correct predictions to the total number of predictions.

$$A = (TP + TN)/(TP + TN + FP + FN) \quad (1)$$

- False Positive Rate (FPR)—It is the share of the number of false positive predictions from the total number of negative observations. In other words, it describes what proportion of samples that should have been classified as negative are not in that class.

$$FPR = FP/(TN + FP) \quad (2)$$

- False Negative Rate (FNR)—It is the share of the number of false negative predictions from the total number of positive observations. In other words, it describes what proportion of samples that should have been classified as positive are not in that class.

$$FNR = FN/(TP + FN) \quad (3)$$

- True Positive Rate (TPR)—It is the share of the number of true positive predictions from the total number of positive observations. In other words, it describes what proportion of samples that should have been classified as positive are in that class. This statistic is also known as Probability of Detection (POD).

$$TPR = TP/(TP + FN) \quad (4)$$

- True Negative Rate (TNR)—It is the share of the number of true negative predictions from the total number of negative observations. In other words, it describes what proportion of samples that should have been classified as negative are in that class.

$$TNR = TN/(TN + FP) \quad (5)$$

False positives lead to an overestimation of night visibility since some non-visible light points are marked as ‘detected’. This can potentially lead to a safety issue, as an aircraft could try to land in worse weather conditions than expected.

False negatives lead to an underestimation of the night visibility since visible light points are marked as undetected. This can potentially lead to an unnecessary flight rerouting to a different airport. The consequences would be additional costs for the airline and, increased fuel consumption and CO₂ emissions.

In the domain of aviation safety, the metrics that focus on minimizing false positives, namely TNR and FPR, are more relevant than the ones that are aimed at minimizing false negatives. In the remainder of the study (and in the upcoming ones in the future), we, therefore, propose to use the metric FPR as the one to be minimized, and the metric FNR as a satisfying one. This would imply setting an acceptable value of FNR and trying to lower FPR as much as possible.

2.7. Explainable AI

The standard statistical scores and metrics described in Section 2.6 are excellent tools to inter-compare different models in terms of figures; nevertheless, they do not offer any hint for why they perform the way they do. The possible reasons for the good or bad performance of the individual models have to be examined in a different way, by an analysis of the individual cases of the correctly and incorrectly predicted detectability of key points and their surroundings.

Explainable AI is a novel and useful tool, which generally offers further insight into the processes of how machine learning algorithms make decisions in individual cases [43,44]. There is a wide variety of explainable AI tools such as Saliency [45], Vanilla-gradient [46], Occlusion Sensitivity [47], CAM (Class Activation Mapping) [48], and Grad-CAM (Gradient-Weighted CAM) [49] algorithms. We have decided not to use CAM because Grad-CAM is considered to be a superior alternative addressing CAM's drawbacks [49]. We have tried out the rest of the mentioned explainable AI algorithms during the pilot stage of our study. Both Vanilla-gradient and Saliency visualized the interpretability results based on individual pixels rather than regions on the image, which were often too low-level and out of context for our purposes. In addition, many outputs were not 'readable' since an entirely black image was returned as a result. Even though Occlusion Sensitivity was able to highlight interesting regions, it returned both reasonably interpretable and confusing outputs. Moreover, this method was the most time consuming in terms of computational cost. We have thus decided to proceed with the Grad-CAM algorithm since it returned acceptable results, is well studied, robust, relatively fast, and takes into account more of the 'context'.

The Grad-CAM algorithm uses gradients of any target object, which pass into the final convolutional layer. Its goal is to create a rough localization map highlighting important areas in the image, which serves as the basis for any inference to be made about the studied object.

Aside from the Grad-CAM algorithm, the results from the last layer of the convolutional neural network were analyzed using the Sigmoid activation function. In general, the role of the activation functions (or transfer functions) in machine learning is to transform the input (or several inputs) into an output. In other words, the activation function decides whether to activate the neuron or not, i.e., whether the input information is useful and should be transferred to the next node. The Sigmoid function is a widely used activation function in neural networks, which can be characterized by a non-linear behavior [50,51].

In the traditional model of binary thresholding where neither the explainable AI nor any activation functions can apply, the selected cases of correctly and incorrectly predicted visibility were analyzed through a calculation of the maximum pixel intensity in the modified images.

2.8. Computational Details

In the Methods section, two data sets (original and modified images, Section 2.3) and two methods of the light spots' detectability prediction (binary thresholding and convolutional neural network, Section 2.4) were introduced. Herein, we describe the

resulting combinations of data sets and methodological approaches (termed as Models #1, #2, and #3) as well as the technical computational details involved. Note that it was not possible to evaluate all 4 combinations of two data sets and two methods since the binary thresholding only works with modified images.

Model #1 to classify key point detectability was constructed based on the method of binary thresholding (Section 2.4.1) and using the final data set (Section 2.5.2) of the modified image patches (Section 2.3.2). For this model, we have decided to keep the data set as is, e.g., it was not split. We compared the BT-based predictions of detectability with the true visibility labels for each key point of the data set. Note that to allow for a proper comparison of the outcomes across different models, these results were normalized to the number of the test set records from models based on the convolutional networks, which is 690 (10% of the size of the data set 6892).

Model #2 utilized a convolutional neural network (CNN, Section 2.4.2) with the modified images (i.e., pre-processed by a morphological erosion) from the final data set (Section 2.5.2). Model #3 also followed the design of the experiments using CNN; however, the original images from the final data set were used as inputs. For these models a Monte Carlo cross-validation approach [52] was adopted, where the training, testing, and validation data sets were created from the final data set in the ratio of 80:10:10. Such a procedure was repeated five times, each time with a different random seed value. The values provided in the tables represent an average of these 5 trainings. All the input images were scaled to the range 0 to 1. The images in the training and validation sets were shuffled. Binary cross-entropy was adopted as the loss function, and 'Adam' as the optimizer. The learning rate was standard, namely 0.001. The batch size during training was 32, and the training lasted a maximum of 10 epochs. The method of early stopping was implemented.

3. Results

3.1. Low Visibility at Night

The Poprad–Tatry Airport and its vicinity have recently been the subject of several meteorological–climatological analyses, which adopted both traditional (physics-based) and AI-based approaches [5,53,54]. Especially, Bartok et al. [55] evaluated fog characteristics at the target site from the analyzed 39-month period. Figure 5 shows the frequency of fog occurrence at the target site, stratified according to the hours of the day and the individual months, and restricted to the night hours, i.e., between 7 p.m. and 8 a.m. Herein, the observed low-visibility phenomena were classified as fog as soon as the METAR (Meteorological Aerodrome Report) message reported visibility below 1000 m and any type of fog. The color shading of the cells (and the figures within) in Figure 5 indicates the frequency of the occurrence of fog, expressed in the percent of the total available number of METAR messages for the given hours and months. For instance, there are three Octobers in the selection (2018, 2019, and 2020) with a sum of 93 days. Since the METAR messages are issued with a 30 min frequency, there were $2 \times 93 = 186$ METAR messages distributed for each hourly interval. For instance, the maximum value of the plot (15%) is a ratio of the 28 occurrences of the fog from the total of 186 METAR messages.

Figure 5 underpins that fog occurs most frequently in the cold half of the year, i.e., from October to March (however, in March, the fog occurrence is not significant). Figure 5 further reveals that fog at the target site typically occurs in the period from midnight to 8 a.m., which can be explained by the most intensive cooling of the Earth's surface in the night-time. The secondary maximum occurs in the late evening hours.

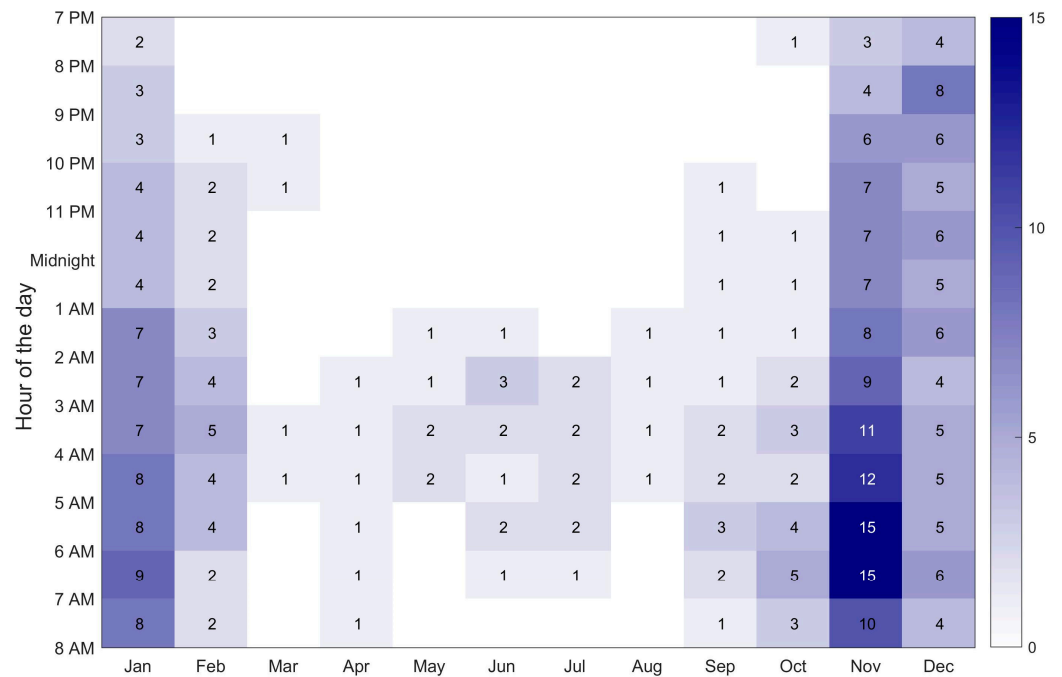


Figure 5. Frequency of fog occurrence during the night hours (between 7 p.m. and 8 a.m.), expressed as a percent ratio of the total available number of METAR messages for the given hour (vertically) and month (horizontally) at the Poprad–Tatry Airport, in the period of January 2018–March 2021. Low visibility was declared as fog as soon as the METAR message included a report of any type of fog, and the visibility was below 1000 m.

3.2. Performance of the Models of Visibility Prediction

Herein, the performance of Models #1, #2, and #3 is presented and analyzed.

The most important findings related to Model #1 (=binary thresholding + modified image patches) are summarized in Figure 6. On one hand, it presents the effect of the patch size selection (with different key point neighborhood sizes, see Section 2.5.3) on the performance of Model #1 in terms of accuracy. On the other hand, Figure 6 also includes the corresponding intensity threshold values.

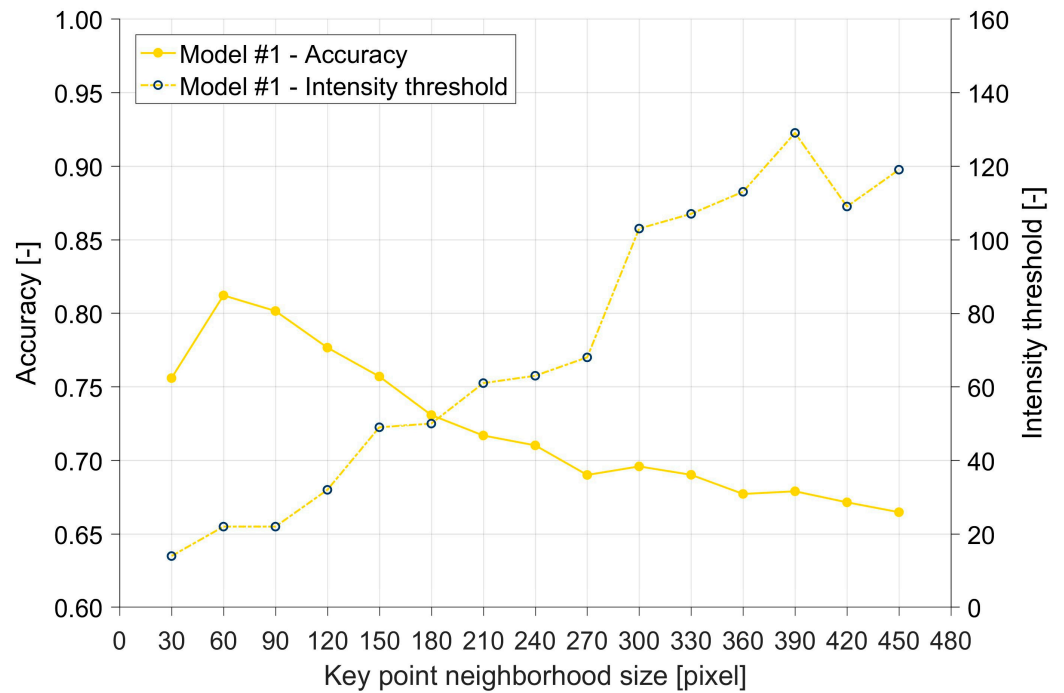


Figure 6. Accuracy of the performance (left y -axis) and intensity threshold values of the binary thresholding (right y -axis) as the functions of the key point neighborhood size.

The pattern of the model's accuracy in Figure 6 (left y -axis) seems to be clear. First, the accuracy increases, and it reaches its maximum at one of the smallest image patch sizes (60×60 pixels). From this point onward, however, the accuracy gradually decreases with increasing image patch size. On the other hand, a further characteristic, the threshold value of the binary thresholding (namely the intensity value, which defines which part of the image is considered as detectable/undetectable) shows an opposite behavior: it increases practically monotonously as the key point neighborhood size increases (Figure 6, right y -axis).

Generally, in images with larger key point neighborhoods, multiple points of light of similar intensity were usually captured. Figure 6, therefore, indicates that capturing more than one key point in an image resulted in a higher threshold value, and consequently, in a lower accuracy of the model. On the other hand, the highest accuracy was achieved in the case when the intensity threshold value was ~ 21 .

An additional plot related to the peak intensity of the modified images (Figure 7) presents a further important aspect of the analysis. It indicates that in most images, the most frequently occurring peak values were less than the intensity threshold of ~ 21 . They, however, represent noise or unfiltered artifacts left over from the morphological reconstruction of the image.

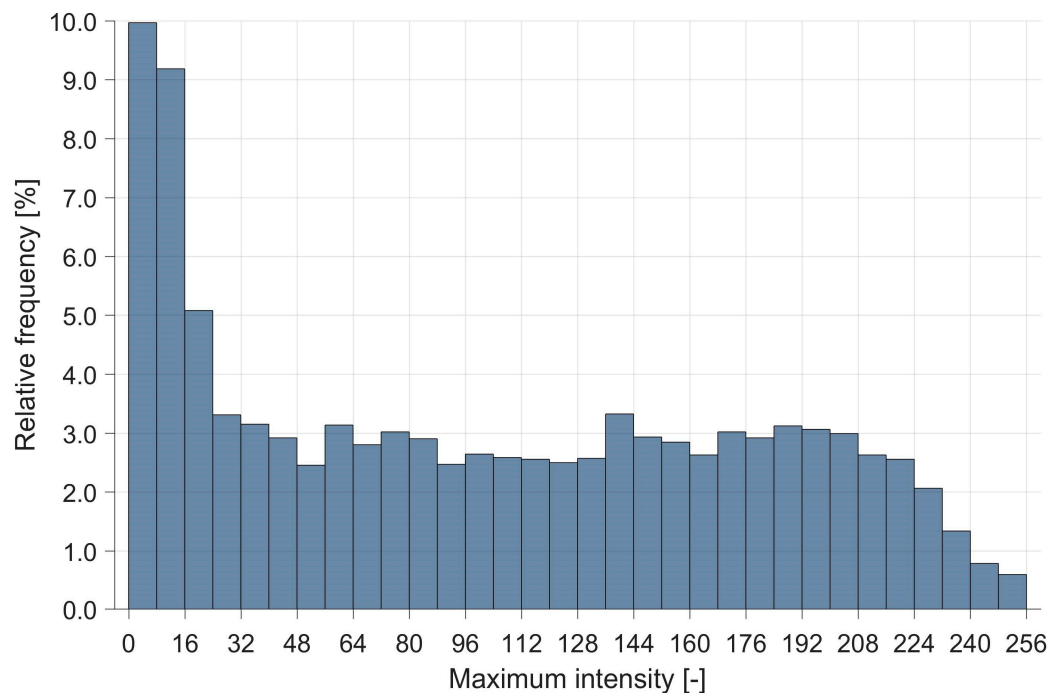


Figure 7. Distribution of the peak intensities detected on all of the image patches with different neighborhood sizes in the analysis of the performance of Model #1.

Overall, the obtained results indicate the low stability of the binary thresholding method. With Model #1, the average accuracy of key point visibility classification reached $\sim 72\%$ (averaged through all the 16 different neighborhood sizes, from 30×30 to 450×450 pixels). Nevertheless, with an appropriately chosen image patch size (namely, 60 pixels), the accuracy of the BT-based prediction may exceed $\sim 81\%$.

The test to verify the dependence of the accuracy on the image patch size (with 15 different sizes) was carried out for both the CNN-based models; their results are summarized in Figure 8 (along with those for Model #1). It can be seen that with the increasing patch size, the accuracy of Model #2 increases monotonously, with a single exception indicated by a minor drop at the value of 270 pixels. Nevertheless, the most relevant feature of this function is that the AI-based Model #2 performs better and better with increasing patch size, whereas for Model #1, the exact opposite statement applies.

The performance of Model #3 resembles the one of Model #2. First, with the increasing size of the image patches (from 30 to 120 pixels), the accuracy of Model #3 sharply increases. Then, in the region of the values of 120 to 270 pixels, the accuracy oscillates around the same value, with a slightly increasing trend. The last third of the plot (from 270 to 450 pixels) exhibits higher values than the second one; however, one can distinguish a moderately increasing trend, indicating a certain kind of saturation.

The overall features of the plot are similar to those of Model #2 (Figure 8); however, Model #3 can be characterized by somewhat larger figures within the entire spectrum of the considered key point neighborhood sizes. Obviously, Model #3 also outperforms Model #1.

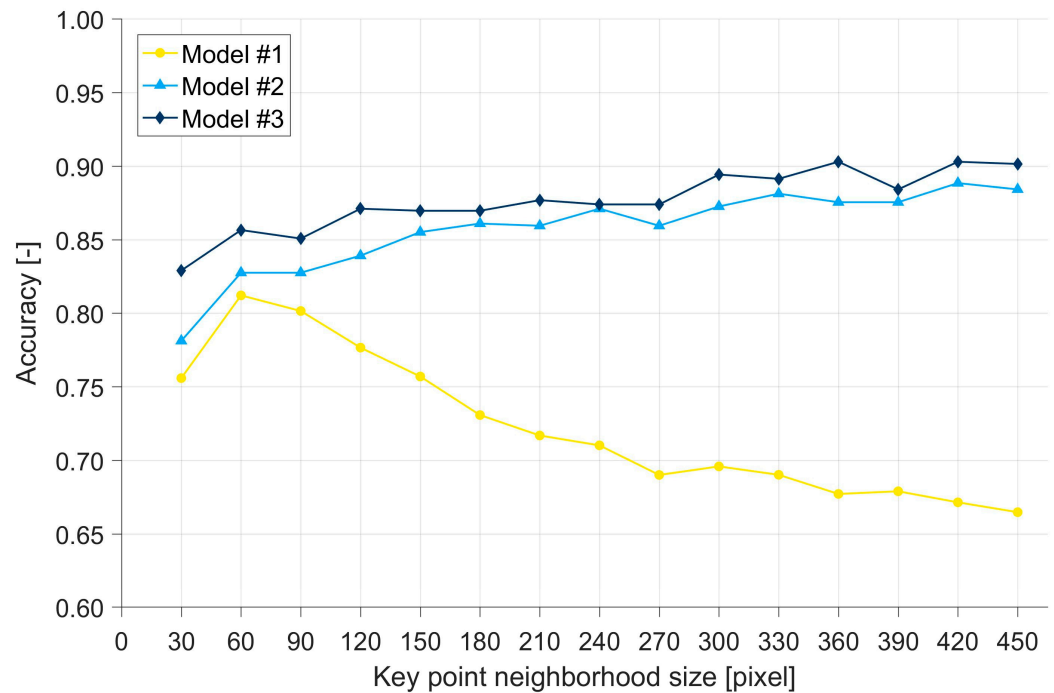


Figure 8. Accuracy of the performance of Models #2 and #3 as a function of the key point neighborhood size. Similar results related to Model #1 are included for a comparison.

The patterns of Figure 8 suggest that with the increasing image patch size, Model #3 was able to cope better with camera shifts. On the other hand, the images with larger patches captured several key points or objects, which might have complicated the model’s decision on the detectability of the key points. A more elaborated analysis related to the performance of the individual models will take place in the Discussion (Section 4).

Table 2 presents the average classification results of Models #1, #2, and #3 where the images with the default key point patch size (256 × 256 pixels) served as the input.

Table 2. Statistics of the visibility/detectability classification of the light points (the first four lines of the table), and the derived metrics for the examined models. Sign ↑ denotes ‘the higher figure the better model’, whereas the opposite holds true for the sign ↓. Bold indicates the best model.

Statistics, Metrics	Model #1	Model #2	Model #3
Visible/Detected (TP) ↑	278	285	298
Visible/Undetected (FN) ↓	67	63	50
Obscured/Detected (FP) ↓	140	27	34
Obscured/Undetected (TN) ↑	205	315	308
Accuracy (A) ↑	0.70	0.87	0.88
False Positive Rate (FPR) ↓	0.33	0.09	0.10
False Negative Rate (FNR) ↓	0.25	0.17	0.14
True Positive Rate (TPR) ↑	0.81	0.82	0.86
True Negative Rate (TNR) ↑	0.59	0.92	0.90

Table 2 presents the summary results of the visibility predictions by Models #1, #2, and #3. An inter-comparison of the scores of the contingency tables (the first four rows of Table 2) indicates that the biggest difference appears in the case of the number of false positives. Model #1 relying on a standard approach had many more FPs (140) than the other two models based on the AI technique (27 and 34 for Model #2 and #3, respectively). The observed difference in the number of FPs (i.e., error type I) may result from the potential presence of other light points with the same intensity in the modified images. A further striking difference between the models is also related to this finding, and it can be found in

the numbers of correctly predicted negatives (205, 315, and 308, for Models #1, #2, and #3, respectively). Such a poor performance of Model #1 might be explained by the fact that the presence of any light point in the modified images makes it impossible for this BT-based (i.e., traditional) approach to classify the key point as undetectable.

The average values of the accuracy in Table 2 are in perfect agreement with the overall features of the plots for the different key point neighborhood sizes presented in Figure 8: the worst performer is Model #1, whereas Models #2 and #3 are better, with a slight difference between them.

Relatively huge differences also appear in the derived statistical metrics (the last four rows of Table 2). Overall, the most striking contrast emerges between Model #1, making use of a standard approach of binary thresholding, and the remaining pair of models relying on AI-based visibility prediction, both using a convolutional neural network. In line with this finding, the equivalent metrics for Models #2 and #3 only showed a little difference, sometimes pushing Model #2 to the first position, in other cases Model #3.

In Model #2, we used images that mostly contained only the visible points of light since the background information was suppressed due to the use of morphological reconstruction. Thus, when predicting the visibility of the light spots, the convolutional neural network was not influenced by the possible appearance of dominant points in the background. This can be both an advantage and a disadvantage. The absence of a dominant object with visibility other than that of the key point can help the prediction. On the contrary, the lack of information that the network can extract from the background can harm the prediction. Table 2 indicates that despite the missing information from the background of the key points, Model #2 achieved results comparable to those of Model #3.

4. Discussion

4.1. Statistical Significance of Results

As Figure 8 indicates, the differences between some models are striking, especially between those based on the AI approaches in comparison with the traditional ones. We, therefore, decided to perform statistical tests to verify the significance of the differences between the individual models. The predicted accuracy values related to different key point neighborhood sizes served as the input data sets in statistical testing. Two pairs of models were compared:

- Model #2 vs. Model #1 (i.e., to verify statistical significance resulting from differences in methods);
- Model #3 vs. Model #2 (i.e., to verify the statistical significance resulting from the difference in inputs processed by the same methodology).

To determine the significance of the differences between the pairs of models, we primarily decided to perform a statistical *t*-test [56]. Considering the independence of the available samples and the goal of the statistical test itself, the *t*-test for independent samples was chosen. Its assumptions, however, were not met. Neither the normal distribution of the data, tested by the Shapiro–Wilk test [57], nor the equality of variances, tested by Levene’s test [58], was fulfilled. Given the fulfillment of the assumption of sample independence, we subsequently decided to use an alternative non-parametric test, the Mann–Whitney U-test [59], which is resistant to the violations of the assumption of normality of the data and capable of comparing the medians of two independent groups.

The results of the Mann–Whitney U-test showed:

- Statistically significant difference between Models #2 and #1;
- Statistically insignificant difference between Models #3 and #2.

4.2. Interpretability of Results

Neural networks traditionally belong to uninterpretable methods, i.e., they are generally considered as ‘black boxes’. Their internal structure can be complex and difficult to interpret directly. When data are fed into a neural network, they pass through several layers

of interconnected nodes, each performing some operations on the data. Understanding exactly how these computations lead to a particular output can be challenging, especially in the case of multi-layered deep learning systems. Nevertheless, efforts are being made to increase the interpretability of neural networks to provide insights into which parts of the input data are most influential in producing a given output.

We attempted to interpret the results by means of (i) the Grad-CAM algorithm within the CNN approach (Section 2.7), (ii) the analysis of the results from the last layer of the CNN using the Sigmoid activation function (again, Section 2.7), and (iii) the analysis of individual cases of binary thresholding.

4.2.1. Grad-CAM

An example of sample images analyzed by the Grad-CAM algorithm is presented in Figure 9.

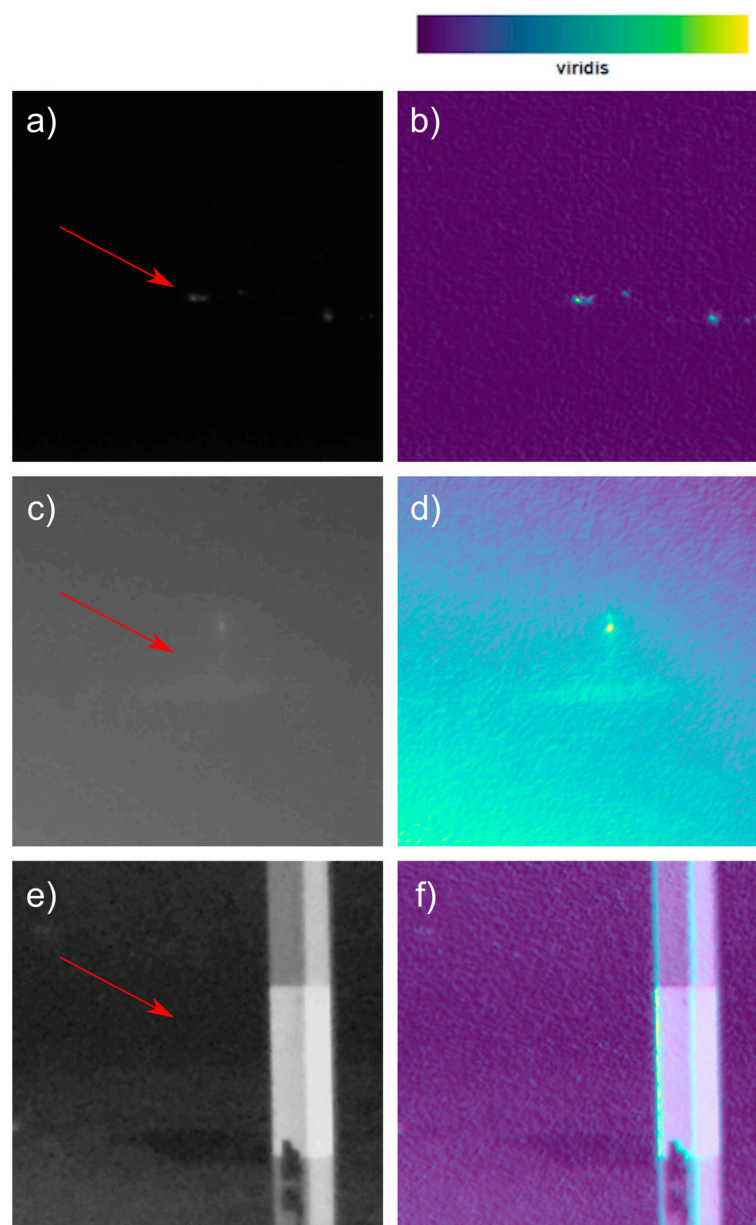


Figure 9. Selected pairs of the original images of key points and their surrounding areas (left; a,c,e) and the corresponding outputs of the Grad-CAM algorithm (right; b,d,f) for the key points labeled as 'Tatranska Lomnica' (top; a,b), 'Wind Cone' (middle; c,d), and 'Runway Light 1' (bottom; e,f). The

Grad-CAM outputs are visualized using the ‘viridis’ colormap, where the purple color indicates that the given pixel was ignored by the neural network, whereas the pixels highlighted with yellow color were considered with the highest weight. The red arrows point to the individual key points.

The visible key point labeled as ‘Tatranska Lomnica’ that was correctly predicted by the CNN approach as detectable (i.e., TP) is shown in Figure 9a. Figure 9b represents the output of the Grad-CAM algorithm. It puts emphasis on the light sources and their immediate surroundings that represent the monitored key point. In addition to them, however, other light sources are also highlighted, but with lower intensity.

The key point named ‘Wind_Cone’ in Figure 9c was captured in foggy conditions. We consider the visibility of this point controversial since in some other images in the data set, with meteorological circumstances similar to those in Figure 9c, the key point was sometimes declared as visible, other times as invisible. According to the output of the Grad-CAM algorithm shown in Figure 9d, the key point itself was marked as a significant area. In addition to this, further points were identified by the algorithm as significant areas; however, these might represent the fog reflection in the wind cone surroundings. In the end, the AI predicted the detectability in this case as 0, which differs from the true visibility value of this key point observed by humans (i.e., FN).

Figure 9e shows the key point labeled as ‘Runway Light 1’ with incorrectly predicted visibility. Here, the AI approach declared the key point as detectable, whereas, in fact, it was invisible (i.e., FP). According to the Grad-CAM algorithm (Figure 9f), the most significant area was the part of the mast that did not belong to the key point itself. In fact, the mast represents another key point that is used in determining daytime visibility at the Poprad–Tatry Airport.

This analysis indicates that to achieve the optimal results, a trade-off should be made between the following:

- The advantages of larger image patches, with
 - Higher accuracy of the CNN approach (underpinned by Figure 8);
 - Lesser susceptibility to camera shifts (due to vibrations in gusty wind).
- The advantages of smaller image patches, with
 - Lesser contamination by nearby key points or dominant objects.

4.2.2. Sigmoid Activation Function

The Sigmoid activation function was adopted to examine the results from the last layer of the trained CNN, especially the cases with incorrectly predicted visibility. The results from this analysis were highly variable. Depending on the randomly selected subset of the data, the probability of predicting the detectability of key points correctly varied greatly. Their distribution in some cases only approached the extremes of 0 or 1, whereas in others it was around the threshold value of 0.5 (i.e., the probability of a correct prediction was close to random).

4.2.3. Maximum Pixel Intensities

The performance of the binary thresholding method was examined through the maximum pixel intensity values of the modified images that could represent light points. The maxima served to estimate the threshold at which these light points would be classified as detectable or undetectable using the method of binary thresholding. Figure 10, which shows three different cases of visibility of the key point ‘Wind Cone’, represents an example of such an approach. Each subfigure has a value of maximum intensity assigned, derived from the corresponding modified images. The analysis suggests that the binary thresholding method is interpretable in a more straightforward way, as one can clearly evaluate specific threshold values at which the target objects might be classified as detectable/undetectable.

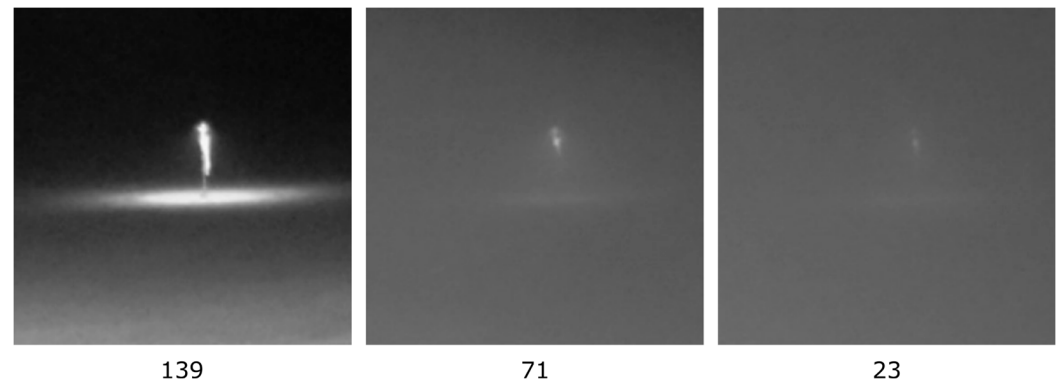


Figure 10. Original images related to the key point labeled as ‘Wind Cone’, in three different circumstances of visibility. The figures below the individual images indicate the maximum pixel intensity, detected in the corresponding modified image.

4.3. Findings in Terms of Related Works

As Ait Ouadil et al. [6] review, there is an enormous pile of AI-based visibility prediction methods; however, the vast majority of them are devoted to daytime conditions. Our results cannot be compared with them at all, since there is a fundamental difference between the visibility prediction during the day vs. night. The complexity of daytime prediction stems from the wide variability of the objects’ features (shape, size, color, shade, hue, saturation, etc.), although the approaches allow for defining an arbitrary number of auxiliary points (landmarks) for a more accurate prediction. In contrast to this, there are no object features or the possibility to define landmarks in night-time prediction. The difficulties of the night-time approaches follow from the limited amount of available information, i.e., whether the methods are able to detect light sources and distinguish them from the background (surrounding pixels).

We found a very limited number of studies focusing on night-time visibility prediction by various AI approaches [20–23]. As mentioned in the Introduction, all four studies used data from one particular day (March 13, 2020 from 0:00 to 8:00) both for the training and testing of the proposed models. We do not consider this data set to be representative of different weather conditions. Aside from this fact, the idea of feeding the data from the Chinese airports into our model indeed emerged, just to see the performance of all the models on the same data set. Nonetheless, as emphasized above, it was impossible to obtain the discussed data set since the only web address cited in [20] was no more accessible in the period of our model developing and manuscript preparation (2023–2024). Comparing the performance of deep learning models on different data sets is not a viable alternative: it can lead to very misleading results.

Finally, there is one more point which prevented us from a direct comparison with the above referenced four works [20–23]. In all of them, the evaluation of the performance was based on visibility—either on visibility as a continuous variable or by means of visibility intervals that are relevant in aviation. In the case of our model, the relatively low number and the spatial distribution of the available light spots in the target location did not let us carry out a similar statistical analysis, allowing at least a rough comparison with the referenced works.

4.4. Potential Drawbacks

Some of the more disputable aspects of our study can be highlighted as follows:

Light spot distribution. The number and the distribution of light spots at the target location have a significant influence on the possibility to evaluate the uncertainty in visibility predictions. Generally, the number/distribution of light spots is affected by the local geographical conditions (hills, valleys, lake/sea/ocean shore, etc.), the population density, the level of industrialization, etc., and these aspects were out of our control.

One testbed, one camera model. The data set was collected at one airport (one geographical location) and with a particular camera model and settings. The inclusion of images collected at multiple locations, with different camera settings could prevent future model drifts, and ensure better transferability by diversifying the training data set. In addition, a more realistic estimate of the real-life performance of the developed models would be possible.

CNN architecture. We adopted a computationally efficient but rather simple CNN architecture. Adding more layers to the CNN will increase both the computational cost and hardware demands, but can lead to more favorable results, especially when combined with a more diverse data set.

No auxiliary data used. The study was only based on graphical inputs and visibility data as the ground truth from professional observers. No other ground meteorological observations were incorporated. Nevertheless, as Chen et al. [20] report, the inclusion of additional variables can greatly improve the model's performance.

4.5. Influential Factors and Further Considerations

The quality of the results may have been affected by a number of objective or subjective factors. These are, among others, as follows:

Camera shifts. The correctness of the AI-based prediction of night-time visibility is conditioned on the proper functioning of the camera system. The critical part of the observing system is the rotator, which may provide camera images of lower quality either due to temporary effects (e.g., shaky images due to the wind gusts) or due to long-term effects, which may be caused by gradual wear of some mechanical parts of the rotator. For this reason, a novel 'static' solution with no moving parts is being tested recently at a selected testbed. The new construction of the system consists of five high-resolution cameras at fixed positions, which are aimed at covering all four cardinal directions. Five cameras (instead of four that one would naturally assume) are expected to supply satisfactory coverage of the entire horizon with a sufficient reserve for overlapping the neighboring pictures, including the possibility of getting rid of some possible distortions at the edges of the camera pictures. Note that a potential solution with a single (and static) fisheye-lens camera covering the entire hemisphere does not represent a viable solution to our problem. Such an optical tool has widespread applicability in camera-based observing systems; however, they suit better to observing cloud coverage (with a dominant focus mostly in vertical directions) than horizontal visibility (with the areas of the key interest occurring at the edges of a fisheye-lens cameras).

External pollution of the camera. Insects and/or spiders represent occasional disturbing elements in the camera view. Their occurrence can be reduced by a preventive spraying of insecticides around the camera objective. Raindrops and snowflakes are further and more frequently occurring factors affecting the quality of the camera images. Water droplets from the camera lens can be removed by heating the target area; nevertheless, the effect of such an operation manifests itself gradually, resulting in a few images with decreased quality. A further possible solution is the usage of blowing devices that can blow water droplets/pollution from camera lenses just before the moment of image capture; however, the development and deployment of such tools can be expensive. Based on our experience we can underpin that with the usage (and combinations) of adapted HW, camera shields, and water-repellent layer on camera lenses, the effect of extreme weather on the quality of camera images can be significantly reduced. The ultimate effect of these interventions, however, depends on the character of local weather and climatic conditions.

Technical parameters of the camera. It is beneficial to use a camera with a strong microchip that allows at least full HD or higher resolution. Larger microchips are able to detect a higher portion of light, and therefore, the resulting images are less affected by noise, particularly in night-time conditions. Furthermore, it would also be beneficial a higher dynamic range (10 bits or more) of the camera that could better capture the details of the bright and dark parts of the image.

Subjective character of the ground truth. The evaluation of visibility carried out by professional meteorological observers is generally considered as the ground truth. Nevertheless, since the subjective character of the procedure cannot be fully eliminated, the ground truth remains, at least to some degree, questionable. Not seldom there are not fully clear weather conditions where the visibility is determined by different observers in a different way. Moreover, it may also happen that a borderline case can be judged by a given meteorological observer in a different way, when evaluated repeatedly, for instance, using some form of backup. Adoption of machine learning techniques, on the other hand, is a significant step towards the objectivization of the decision procedure since a trained neural network will always decide the same way when evaluating identical camera images repeatedly several times.

Biomedical aspects of night vision. Some humans have better night vision than others due to variations in the structure of their eyes and the functioning of their retinas. The ability to see in low-light conditions, such as at night, is primarily determined by the presence of specialized cells in the retina called rods. Rod cells are highly sensitive to light and are responsible for our ability to see in dim lighting. Individual differences in night vision can be influenced by factors such as the density of rod cells in the retina, the efficiency of the visual processing pathways in the brain, and genetic variations that affect the functioning of these components. Additionally, environmental factors like exposure to bright lights or certain medications can also impact night vision [60]. Although many studies have confirmed wide variation in night vision in young healthy volunteers, some factors were explained but the majority of the variation in night vision remains unexplained by obvious non-genetic factors and suggesting a strong genetic component [61]. This medical knowledge gives importance to our solution of the automatic determination of night visibility with cameras because it can bring significant accuracy and objectification.

5. Conclusions

In the presented work, we focused on the issue of the detectability of light points at night. We worked with a data set consisting of (1) camera photos from a selected location (Poprad–Tatry Airport, Slovakia) and (2) the information on whether the key points (light spots) were detectable or not by professional observers.

Multiple sets of images were created for each key point by a cut-out from the original camera images with a number of different key point neighborhood sizes (termed as ‘original’ data set). The ‘modified’ data set of images was created from the ‘original’ one by applying morphological erosion to emphasize the dominant features of the images and suppress detected noise.

The detectability of light points in the images was classified in two ways. The first one was a traditional approach using simple binary thresholding (BT), while the second one was based on one of the progressive AI approaches (convolutional neural network, CNN, with a simple architecture). The individual cases of predicted detectability were analyzed in the light of true visibility values that were determined by human observers.

We primarily compared the use of the standard BT approach with CNN. The two models based on trained CNN achieved remarkably better results in terms of more favorable values of metrics, greater stability, and less susceptibility to errors than the classification by the BT method. The difference in the input image sets (modified vs. original) did not affect the results considerably—some scores/statistics were more favorable for the modified images, others for the original ones. This emphasizes the benefits resulting from image modification, which are mainly the highlighting of dominant features and a lower significance of key points’ background.

We also examined the accuracy of the predictions as a function of the image size around the key points. In these experiments, the CNN classification method showed greater stability since the accuracy of these models grew with increasing pixel size around the key points, whereas an opposite behavior was discernible for the BT method. This finding was also confirmed statistically: the Mann–Whitney U test indicated a significant

difference between the CNN and the standard BT classification methods. On the other hand, the difference between the two models based on CNN classification but with different inputs (original vs. modified images) was statistically insignificant.

The rate of misclassification using CNN was lower, and the errors were caused mostly due to the following two reasons: (i) questionable annotations of true visibility of the key points, or (ii) the presence of another dominant point with different visibility. The error rate of classification by binary thresholding was manifested to a greater extent, especially in cases of the presence of other light points, or parts of the gray image with higher intensity.

In the paper, we managed to point out the potential of an automated and AI-based determination of key points' detectability in the night. The method has shown stability and decent accuracy, which is important in meteorology, and has great perspectives in the everyday routine of professional meteorology, especially in an automated estimation of prevailing visibility. In the current study, we did not make attempts to convert the light points' detectability into actual values of visibility (expressed in meters). It would be interesting to evaluate the uncertainties in the determination of visibility, stratified according to distance categories (see Table 1); however, the low number and the spatial distribution of the available light spots in the target location did not allow for a satisfactory statistical analysis. Such a task remains one of the key plans for the future, either by having collected a larger database covering a much longer period or by setting up a further experiment at a different testbed.

While the proposed solution may offer advantages, it is also important to acknowledge its limitations. Here, we list some issues with the potential to enhance the performance of the proposed approach for future work.

Quality control of the annotated data set. One of the foreseeable problems is the difficulty of obtaining an annotated data set with a well-defined visibility of key points. In the current work, several cases were discovered where the annotated true visibility did not correspond to reality, or, at least, it was questionable. Although being aware of this deficiency, we did not ignore these annotations, which could have caused the results to be distorted. We, therefore, suggest a more thorough screening of the annotated data set to improve the automated predictions' accuracy.

Quality control of the camera images. Another limitation is that we did not directly control the quality of the camera images. As part of their pre-preparation, the images were filtered using additional information about their quality, which we fully relied on. Further undiscovered and unfiltered, low-quality photos (for instance, damaged ones or with low resolution) in the data set could have increased the bias in the results. Therefore, within the field of extension, we call for the investigation of ways to improve the quality of camera images or the implementation of measures to control their quality.

Camera shifts. Our solution does not consider camera movements that could occur over several years due to various unfavorable weather conditions. This could cause shifts between the co-ordinates of the key points and their location in the photos. To overcome this problem, we propose either a software-based or a hardware-based solution. The SW-based solution would involve key point motion analysis techniques in sequential images and their adaptation, for example, using the methods of image registration. The HW-based solution could be a replacement of the critical (moving) part of the system (i.e., a camera mounted on a rotator) by construction with several fixed and high-quality cameras, as indicated in the discussion above.

Deep learning. The aim of the current study was to devise a light-weight neural network suitable for deployment with a low computational cost and moderate hardware requirements. Increasing the complexity of the neural network is expected to lead to better overall performance for accuracy, FNR, and FPR. Although we expect our model to be transferable to other airports in similar meteorological environments, this issue remains to be investigated in further research. In this study, we have decided to address the imbalance of the data set by keeping all the class 0 data and randomly adding class 1, e.g., ensuring that both classes are equally distributed. Other options could be data augmentation [62] or a higher misclassification cost for wrongly classified minority events [63–66].

Author Contributions: Conceptualization, I.M.O., L.I. and J.B.; methodology, Z.G., I.M.O., W.B. and J.B.; software, Z.G.; validation, Z.G., I.M.O. and W.B.; formal analysis, Z.G.; investigation, Z.G., I.M.O. and W.B.; resources, W.B.; data curation, Z.G., I.M.O. and L.I.; writing—original draft preparation, Z.G., S.H. and L.G.; writing—review and editing, L.G.; visualization, L.G.; supervision, J.B., I.M.O. and W.B.; project administration, L.I.; funding acquisition, J.B. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the project AI4EOSC ‘Artificial Intelligence for the European Open Science Cloud’, which is funded by the European Union’s Horizon Europe research and innovation program under grant agreement number 101058593.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Acknowledgments: This work was supported by the Slovak Research and Development Agency under contract no. APVV-22-0107 (AeroCloud).

Conflicts of Interest: Author Juraj Bartok, Irina Malkin Ondík, Lukáš Ivica and Ladislav Gaál was employed by the company MicroStep-MIS. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Appendix A

The following summary (in the form of bullet points) provides a description of the CNN architecture adopted in the study:

- 0 Model initialization
 - The model is initialized using `tf.keras.Sequential()`, which indicates a linear stack of layers.
- 1 First layer: Convolutional Layer
 - Type: Convolutional Layer (Conv2D)
 - Number of Filters: 8
 - Kernel Size: 3×3
 - Activation Function: ReLU (Rectified Linear Unit)
 - Input Shape: $256 \times 256 \times 1$ (for grayscale images) or $256 \times 256 \times 3$ (for RGB images)
 - This layer extracts 8 feature maps from the input image using 3×3 filters, applying the ReLU activation function to introduce non-linearity.
- 2 Second layer: Max Pooling Layer
 - Type: Max Pooling Layer (MaxPooling2D)
 - Pool Size: 2×2
 - This layer reduces the spatial dimensions (height and width) of the feature maps by a factor of 2, helping to reduce the number of parameters and computation in the network, as well as controlling overfitting.
- 3 Third layer: Flatten Layer
 - Type: Flatten Layer (Flatten)
 - This layer flattens the 2D feature maps into a 1D vector, which can be fed into fully connected (dense) layers.
- 4 Fourth layer: Dropout Layer
 - Type: Dropout Layer (Dropout)
 - Dropout Rate: 0.1
 - This layer randomly sets 10% of its input units to 0 at each update during training time, which helps prevent overfitting by ensuring that the model does not rely too heavily on any individual neurons.

- 5 Fifth layer: Dense Layer
 - Type: Dense Layer (Dense)
 - Number of Units: 1
 - Activation Function: Sigmoid
 - This layer is a fully connected layer with a single neuron, using the sigmoid activation function to output a value between 0 and 1, suitable for binary classification tasks.

References

1. Theofilatos, A.; Yannis, G. A Review of the Effect of Traffic and Weather Characteristics on Road Safety. *Accid. Anal. Prev.* **2014**, *72*, 244–256. [[CrossRef](#)] [[PubMed](#)]
2. Walker, S.A.; Strader, S.; Dziubla, D.C.; Haberlie, A. Driving Blind: Weather-Related Vision Hazards and Fatal Motor Vehicle Crashes. *Bull. Am. Meteorol. Soc.* **2015**, *96*, 755–778. [[CrossRef](#)]
3. Meng, Y.; Qi, F.; Zuo, H.; Chen, B.; Yuan, X.; Xiao, Y. Multi-Step LSTM Prediction Model for Visibility Prediction. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020. [[CrossRef](#)]
4. Boudala, F.S.; Isaac, G.A.; Crawford, R.W.; Reid, J. Parameterization of Runway Visual Range as a Function of Visibility: Implications for Numerical Weather Prediction Models. *J. Atmos. Ocean. Technol.* **2012**, *29*, 177–191. [[CrossRef](#)]
5. Bartok, J.; Ivica, L.; Gaál, L.; Bartoková, I.; Kelemen, M. A Novel Camera-Based Approach to Increase the Quality, Objectivity and Efficiency of Aeronautical Meteorological Observations. *Appl. Sci.* **2022**, *12*, 2925. [[CrossRef](#)]
6. Ait Ouadil, K.; Idbraim, S.; Bouhsine, T.; Bouaynaya, C.N.; Alfergani, H.; Johnson, C.C. Atmospheric Visibility Estimation: A Review of Deep Learning Approach. *Multimed. Tools Appl.* **2024**, *83*, 36261–36286. [[CrossRef](#)]
7. Talaei Khoei, T.; Ould Slimane, H.; Kaabouch, N. Deep Learning: Systematic Review, Models, Challenges, and Research Directions. *Neural Comput. Appl.* **2023**, *35*, 23103–23124. [[CrossRef](#)]
8. Archana, R.; Jeevaraj, P.S.E. Deep Learning Models for Digital Image Processing: A Review. *Artif. Intell. Rev.* **2024**, *57*, 11. [[CrossRef](#)]
9. Ahmed, S.F.; Alam, M.S.; Hassan, M.; Rozbu, M.R.; Ishtiaq, T.; Rafa, N.; Mofijur, M.; Shawkat Ali, A.B.; Gandomi, A.H. Deep Learning Modelling Techniques: Current Progress, Applications, Advantages, and Challenges. *Artif. Intell. Rev.* **2023**, *56*, 13521–13617. [[CrossRef](#)]
10. Chen, L.; Li, S.; Bai, Q.; Yang, J.; Jiang, S.; Miao, Y. Review of Image Classification Algorithms Based on Convolutional Neural Networks. *Remote Sens.* **2021**, *13*, 4712. [[CrossRef](#)]
11. Mall, P.K.; Singh, P.K.; Srivastav, S.; Narayan, V.; Paprzycki, M.; Jaworska, T.; Ganzha, M. A Comprehensive Review of Deep Neural Networks for Medical Image Processing: Recent Developments and Future Opportunities. *Healthc. Anal.* **2023**, *4*, 100216. [[CrossRef](#)]
12. Adjabi, I.; Ouahabi, A.; Benzaoui, A.; Taleb-Ahmed, A. Past, Present, and Future of Face Recognition: A Review. *Electronics* **2020**, *9*, 1188. [[CrossRef](#)]
13. Apostolopoulos, I.D.; Tzani, M.A. Industrial Object and Defect Recognition Utilizing Multilevel Feature Extraction from Industrial Scenes with Deep Learning Approach. *J. Ambient. Intell. Human. Comput.* **2023**, *14*, 10263–10276. [[CrossRef](#)]
14. Liu, C.; Wang, Y.; Li, W.; Tao, L.; Hu, S.; Hao, M. An Urban Built Environment Analysis Approach for Street View Images Based on Graph Convolutional Neural Networks. *Appl. Sci.* **2024**, *14*, 2108. [[CrossRef](#)]
15. Mehmood, M.; Shahzad, A.; Zafar, B.; Shabbir, A.; Ali, N. Remote Sensing Image Classification: A Comprehensive Review and Applications. *Math. Probl. Eng.* **2022**, *2022*, 5880959. [[CrossRef](#)]
16. Outay, F.; Taha, B.; Chaabani, H.; Kamoun, F.; Werghe, N.; Yasar, A. Estimating Ambient Visibility in the Presence of Fog: A Deep Convolutional Neural Network Approach. *Pers. Ubiquit. Comput.* **2021**, *25*, 51–62. [[CrossRef](#)]
17. Sirohi, D.; Kumar, N.; Rana, P.S. Convolutional Neural Networks for 5G-enabled Intelligent Transportation System: A Systematic Review. *Comput. Commun.* **2020**, *153*, 459–498. [[CrossRef](#)]
18. Yang, W.; Yuan, Y.; Ren, W.; Liu, J.; Scheirer, W.J.; Wang, Z.; Zhang, T.; Zhong, Q.; Xie, D.; Pu, S.; et al. Advancing Image Understanding in Poor Visibility Environments: A Collective Benchmark Study. *IEEE Trans. Image Proc.* **2020**, *29*, 5737–5752. [[CrossRef](#)] [[PubMed](#)]
19. Al Sobhahi, R.; Tekli, J. Comparing Deep Learning Models for Low-Light Natural Scene Image Enhancement and Their Impact on Object Detection and Classification: Overview, Empirical Evaluation, and Challenges. *Signal Process-Image* **2022**, *109*, 116848. [[CrossRef](#)]
20. Chen, J.; Yan, M.; Qureshi, M.R.H.; Geng, K. Estimating the Visibility in Foggy Weather Based on Meteorological and Video Data: A Recurrent Neural Network approach. *IET Signal Process.* **2022**, *17*, e12164. [[CrossRef](#)]
21. Qian, W.; Cheng, L.; Bin, T.; Xichen, W. Research of Deep Learning-Based Visibility Prediction Model for Foggy Days in Airport. In Proceedings of the 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP), Xi'an, China, 9–11 April 2021; pp. 1101–1107. [[CrossRef](#)]
22. Fu, J.; Chen, M. Research on Airport Visibility Detection under Complex Conditions. In Proceedings of the IVSP '21: 2021 3rd International Conference on Image, Video and Signal Processing, Singapore, 19–21 March 2021; pp. 20–25. [[CrossRef](#)]

23. Liu, Z.; Chen, Y.; Gu, X.; Yeoh, J.K.; Zhang, Q. Visibility Classification and Influencing-Factors Analysis of Airport: A Deep Learning Approach. *Atmos. Environ.* **2022**, *278*, 119085. [[CrossRef](#)]
24. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [[CrossRef](#)]
25. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556. [[CrossRef](#)]
26. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258. [[CrossRef](#)]
27. ICAO—International Civil Aviation Organization. Meteorological Service for International Air Navigation. In *Annex 3 to the Convention on International Civil Aviation*, 20th ed.; ICAO: Montreal, QC, Canada, 2018.
28. Pavlove, F.; Lucny, A.; Malkin Ondik, I.; Krammer, P.; Kvassay, M.; Hluchý, L. Efficient Deep Learning Methods for Automated Visibility Estimation at Airports. In Proceedings of the 2022 Cybernetics & Informatics (K&I), Visegrád, Hungary, 11–14 September 2022. [[CrossRef](#)]
29. Abdollahi, A.; Pradhan, B.; Alamri, A. VNet: An End-to-End Fully Convolutional Neural Network for Road Extraction from High-Resolution Remote Sensing Data. *IEEE Access* **2020**, *8*, 179424–179436. [[CrossRef](#)]
30. Lowe, D.G. Object Recognition from Local Scale-Invariant Features. In Proceedings of the 7th IEEE International Conference on Computer Vision, Kerkyra, Greece, 20 September 1999; Volume 2, pp. 1150–1157. [[CrossRef](#)]
31. Calonder, M.; Lepetit, V.; Strecha, C.; Fua, P. BRIEF: Binary Robust Independent Elementary Features. In *Computer Vision—ECCV 2010*; Daniilidis, K., Maragos, P., Paragios, N., Eds.; ECCV 2010. Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6314. [[CrossRef](#)]
32. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An Efficient Alternative to SIFT or SURF. In Proceedings of the International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571. [[CrossRef](#)]
33. Bradski, G. The OpenCV Library. *Dr. Dobbs J.* **2000**, *120*, 122–125.
34. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **1981**, *24*, 381–395. [[CrossRef](#)]
35. Gonzalez, R.C.; Woods, R.E. *Digital Image Processing*, 4th ed.; Pearson: New York, NY, USA, 2018.
36. Šikudová, E.; Černeková, Z.; Benešová, V.; Haladová, Z.; Kučerová, J. *Computer Vision; Object Detection and Recognition*; Wikina: Prague, Czech Republic, 2014; 397p. (In Slovak)
37. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [[CrossRef](#)]
38. Hubel, D.H.; Wiesel, T.N. Receptive Fields, Binocular Interaction and Functional Architecture in the Cat’s Visual Cortex. *J. Physiol.* **1962**, *160*, 106–154. [[CrossRef](#)]
39. Voulodimos, A.; Doulamis, N.; Doulamis, A.; Protopapadakis, E. Deep Learning for Computer Vision: A Brief Review. *Comput. Intel. Neurosc.* **2018**, *2018*, 7068349. [[CrossRef](#)] [[PubMed](#)]
40. Liu, W.; Wang, Z.; Liu, X.; Zeng, N.; Liu, Y.; Alsaadi, F.E. A Survey of Deep Neural Network Architectures and Their Applications. *Neurocomputing* **2017**, *234*, 11–26. [[CrossRef](#)]
41. Bäuerle, A.; van Onzenoedt, C.; Ropinski, T. Net2Vis—A Visual Grammar for Automatically Generating Publication-Ready CNN Architecture Visualizations. *arXiv* **2019**, arXiv:1902.04394. [[CrossRef](#)]
42. Powers, D.M.W. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation. *arXiv* **2020**, arXiv:2010.16061. [[CrossRef](#)]
43. Xu, F.; Uszkoreit, H.; Du, Y.; Fan, W.; Zhao, D.; Zhu, J. Explainable AI: A Brief Survey on History, Research Areas, Approaches and Challenges. In *Natural Language Processing and Chinese Computing*; Tang, J., Kan, M.Y., Zhao, D., Li, S., Zan, H., Eds.; NLPCC 2019. Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2019; Volume 11839. [[CrossRef](#)]
44. Linardatos, P.; Papastefanopoulos, V.; Kotsiantis, S. Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy* **2021**, *23*, 18. [[CrossRef](#)] [[PubMed](#)]
45. Hou, X.; Zhang, L. Saliency Detection: A Spectral Residual Approach. In Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8. [[CrossRef](#)]
46. Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv* **2013**, arXiv:1312.6034. [[CrossRef](#)]
47. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. In *Computer Vision—ECCV 2014. ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014*; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2014; Volume 8689. [[CrossRef](#)]
48. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning Deep Features for Discriminative Localization. In Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2921–2929.
49. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks Via Gradient-based Localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 618–626. [[CrossRef](#)]

50. Szandala, T. Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks. In *Bio-Inspired Neurocomputing*; Bhoi, A., Mallick, P., Liu, C.M., Balas, V., Eds.; Studies in Computational Intelligence; Springer: Singapore, 2020; Volume 903. [\[CrossRef\]](#)
51. Dubey, S.R.; Singh, S.K.; Chaudhuri, B.B. Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark. *Neurocomputing* **2022**, *503*, 92–108. [\[CrossRef\]](#)
52. Xu, Q.S.; Liang, Y.Z. Monte Carlo Cross Validation. *Chemometr. Intell. Lab.* **2001**, *56*, 1–11. [\[CrossRef\]](#)
53. Vorobyeva, O.; Bartok, J.; Šišán, P.; Nechaj, P.; Gera, M.; Kelemen, M.; Polishchuk, V.; Gaál, L. Assessing the Contribution of Data Mining Methods to Avoid Aircraft Run-Off from the Runway to Increase the Safety and Reduce the Negative Environmental Impacts. *Int. J. Environ. Res. Public Health* **2020**, *17*, 796. [\[CrossRef\]](#) [\[PubMed\]](#)
54. Šipková, V.; Hluchý, L.; Dobrucký, M.; Bartok, J.; Nguyen, B.M. Manufacturing of Weather Forecasting Simulations on High Performance Infrastructures. In Proceedings of the 2016 IEEE 12th International Conference on e-Science (e-Science), Baltimore, MD, USA, 23 October 2016; pp. 432–439. [\[CrossRef\]](#)
55. Bartok, J.; Šišán, P.; Ivica, L.; Bartoková, I.; Malkin Ondík, I.; Gaál, L. Machine Learning-Based Fog Nowcasting for Aviation with the Aid of Camera Observations. *Atmosphere* **2022**, *13*, 1684. [\[CrossRef\]](#)
56. Student. The Probable Error of a Mean. *Biometrika* **1908**, *6*, 1–25. [\[CrossRef\]](#)
57. Shapiro, S.S.; Wilk, M.B. An analysis of Variance Test for Normality (Complete Samples). *Biometrika* **1965**, *52*, 591–611. [\[CrossRef\]](#)
58. Brown, M.B.; Forsythe, A.B. Robust Tests for the Equality of Variances. *J. Am. Stat. Assoc.* **1974**, *69*, 364–367. [\[CrossRef\]](#)
59. Mann, H.B.; Whitney, D.R. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *Ann. Math. Stat.* **1947**, *18*, 50–60. [\[CrossRef\]](#)
60. Barbur, J.; Stockman, A. Photopic, Mesopic, and Scotopic Vision and Changes in Visual Performance. In *Encyclopedia of the Eye*; Dartt, D.E., Besharse, J.C., Dana, R., Eds.; Elsevier: Oxford, UK, 2010; pp. 323–331. [\[CrossRef\]](#)
61. Bartholomew, A.J.; Lad, E.M.; Cao, D.; Bach, M.; Cirulli, E.T. Individual Differences in Scotopic Visual Acuity and Contrast Sensitivity: Genetic and Non-Genetic Influences. *PLoS ONE* **2016**, *11*, e0148192. [\[CrossRef\]](#) [\[PubMed\]](#)
62. Chao, X.; Zhang, L. Few-shot imbalanced classification based on data augmentation. *Multimed. Syst.* **2023**, *29*, 2843–2851. [\[CrossRef\]](#)
63. Ting, K.M. A Comparative Study of Cost-Sensitive Boosting Algorithms. In Proceedings of the 17th International Conference on Machine Learning, Stanford, CA, USA, 29 June–2 July 2000; pp. 983–990.
64. Jiang, L.; Li, C.; Wang, S. Cost-sensitive Bayesian Network Classifiers. *Pattern Recogn. Lett.* **2014**, *45*, 211–216. [\[CrossRef\]](#)
65. Jiang, L.; Qiu, C.; Li, C. A Novel Minority Cloning Technique for Cost-sensitive Learning. *Int. J. Pattern Recogn.* **2015**, *29*, 1551004. [\[CrossRef\]](#)
66. Qiu, C.; Jiang, L.; Kong, G. A Differential Evolution-based Method for Class-imbalance Cost-sensitive Learning. In Proceedings of the International Joint Conference on Neural Networks (IJCNN), Killarney, Ireland, 12–17 July 2015; pp. 1–8. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.