

Article

# Weather Classification by Utilizing Synthetic Data

Saad Minhas <sup>1,\*</sup>, Zeba Khanam <sup>1,†</sup>, Shoaib Ehsan <sup>1</sup>, Klaus McDonald-Maier <sup>1</sup> and Aura Hernández-Sabaté <sup>2,3</sup>

<sup>1</sup> School of Computer Science & Electrical Engineering, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, UK; zeba.khanam@essex.ac.uk (Z.K.); sehsan@essex.ac.uk (S.E.); kdm@essex.ac.uk (K.M.-M.)

<sup>2</sup> Computer Vision Centre, Universitat Autònoma de Barcelona, Plaça Cívica, 08193 Bellaterra, Spain; aura@cvc.uab.cat

<sup>3</sup> Departament de Ciències de la Computació, Universitat Autònoma de Barcelona, 08193 Bellaterra, Spain

\* Correspondence: ssminh@essex.ac.uk; Tel.: +44-781-7023-125

† These authors contributed equally to this work.

**Abstract:** Weather prediction from real-world images can be termed a complex task when targeting classification using neural networks. Moreover, the number of images throughout the available datasets can contain a huge amount of variance when comparing locations with the weather those images are representing. In this article, the capabilities of a custom built driver simulator are explored specifically to simulate a wide range of weather conditions. Moreover, the performance of a new synthetic dataset generated by the above simulator is also assessed. The results indicate that the use of synthetic datasets in conjunction with real-world datasets can increase the training efficiency of the CNNs by as much as 74%. The article paves a way forward to tackle the persistent problem of bias in vision-based datasets.

**Keywords:** weather classification; synthetic data; dataset; autonomous car; computer vision; advanced driver assistance systems; deep learning; intelligent transportation systems



**Citation:** Minhas, S.; Khanam, Z.; Ehsan, S.; McDonald-Maier, K.; Hernández-Sabaté, A. Weather Classification by Utilizing Synthetic Data. *Sensors* **2022**, *22*, 3193. <https://doi.org/10.3390/s22093193>

Academic Editor: George D. Magoulas

Received: 31 January 2022

Accepted: 16 March 2022

Published: 21 April 2022

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



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

## 1. Introduction

Simulation and virtual testing will play an increasingly relevant role, as they provide a very effective way to deal with the high number of scenarios Connected and Automated Driving (CAD) vehicles will encounter [1]. Road accidents related to adverse weather conditions play a huge part in disrupting the flow of traffic in a busy city environment [2–4]. The data available at present contains a large amount of variation. Figuring out a particular weather condition is a straight forward task for a normal human being but can be quite challenging for a computer vision system [5–7]. To overcome the challenges, neural networks, in recent decades, have revolutionized computer vision systems to detect the weather condition using images as an input. Indeed, Convolutional Neural Networks (CNN) have been deployed in various fields such as ship detection [8–13], object tracking in endoscopic vision [14,15], nuclear plant inspection [16–18], transport systems [19,20], and other complex engineering tasks [21,22]. Yet, there is still a lot of ground to cover. In the case of weather recognition on roads, the main challenges are: variability in elements such as camera placement and road layouts [23] and the machine learning methods such as CNN. Under such circumstances, there is a need to explore more methods of filling the gaps in between real world images; ideally, a set of images recorded in the same location but with different weather conditions would maximize the efficiency of a machine learning system. This is the main reason why the use of synthetic data can be more productive as compared to the real-world counterpart. In this paper, there are two main objectives: the first is to assess the modifications made to the driver simulator, which was previously used for driver vehicular interactions [24]. The second objective is to test the performance of the generated dataset with other comparable groundtruth datasets.

A custom-built virtual simulator that specializes in varying weather systems was previously implemented [25]. It utilizes Unity3d to simulate the weather with accurate lighting effects. The simulator environment was based on the real-world location of central Colchester, United Kingdom. It features a good mix of wide-open road and inner-city roads. An autonomous car was driven at different hours of the day in weather conditions ranging from sunny, cloudy, or rainy to foggy with different camera angles. The final dataset comprises 108,333 images, with approx. 35,000 images per class. The results show that state-of-the-art CNN architectures trained on the synthetic dataset were able to achieve an accuracy as high as 74% when tested on a real-world dataset [26].

The paper is organized in four sections. In Section 1, the main topic of the article is introduced in the light of recent advances made in this field, the simulator used, and the datasets used for weather classification training and testing. Section 2 presents a detailed description of the weather classification pipeline and the methodologies. Section 3 evaluates the experimental results and inferences that can be drawn. Finally, Section 4 concludes the research work presented in this article that can be used as a guide for future works.

### 1.1. Related Work

Most of the previous research includes the use of polarized and infrared cameras. The use of such cameras can provide some plausible data, but the installation costs can easily be substantial [27]. In order to overcome this issue, the use of RGB cameras is preferred because they are simpler and cost-effective, hence making it viable for mass production.

A study performed by Omer and Fu [28] used color cues to add illumination variance; however, their approach required the detection of white road lines to detect the road area, which can be quite challenging in severe winter weather conditions.

Most of the studies aimed toward driver assistance systems have been performed with regard to rainy weather classifications [29,30]. A study performed by Lu et al. [31] dealt with two class weather classification which included sunny and cloudy. In that study, the authors proposed a new data augmentation scheme to substantially enrich the training data, which was then used to train a latent SVM framework to make the solution insensitive to global intensity transfer. Another study [32] dealt with multi-class weather classification, which only used fixed camera points.

With regard to synthetic datasets, there is a lot of research being carried out to fill the gaps between real world data with synthetic data. There are some driver simulators that can fill in the void by generating synthetic datasets for weather classification. CARLA [33] is one such simulator that aides in autonomous research. It comprises a built in weather system that can be used to generate weather classification datasets. The Synthia dataset [34] is another example, which comprises 200,000 plus images with varying dynamic situations such as a clear sky, rain, and night. Hao et al. [35] developed a weather simulator that could replicate the weather at a given time in a virtual environment. However, it lacked visual fidelity for our experiments. The article also had specific requirements for the camera position's location on the driver's car.

Another plausible direction of the research into weather monitoring has been the use of microwave-based Synthetic Aperture Radar (SAR) imaging. Unlike optical sensors, this tool is unaffected by weather conditions. This is the main reason that they have been used for high speed ship detection [36,37]. The SAR images are used as input to a grid convolutional neural network (G-CNN) to detect ships and their speeds. Another prominent work has used a depthwise separable convolution neural network (DS-CNN) to detect high speed ships [38]. Another direction of research has focused on small sized ships [39] and the unperceived imbalance problem [40]. However, connection to a satellite is not always possible; thus, in this research, optical sensors were considered.

### 1.2. Contributions

Currently, autonomous driving cars deploy a series of sensors to classify the weather condition. However, this solution is not holistic and requires additional computational resources in terms of power and memory. More often, each weather type requires a specific sensor. For instance, rain is detected using a humidity sensor that plays no role in detecting sunny weather. Thus, using a camera to capture images and detecting weather using a single image is not only economical but also computationally less expensive. Finally, to our knowledge, most of these datasets are not aimed toward weather specifically.

For all these reasons the main technical contributions of this paper are the following:

1. **Holistic Weather Classification Solution.** The solution proposed in this paper is a holistic solution in the sense that has been built from the ground up focusing on weather simulations;
2. **Simulator-based Training Dataset.** Detection of weather using a single image requires a large images dataset with varying features for training CNNs, which turns out to be a bottleneck. In this work, a synthetic dataset for training CNNs is proposed;
3. **Real-Time Evaluation.** The state-of-the-art CNNs are evaluated on real-time testing datasets to detect the accuracy of weather classification. Moreover, our classification approach differs from others in the sense that instead of processing only a portion of the image, it processes the entire image. This specific approach allows the system to record subtle changes in light intensity and color variations, which can be crucial in distinguishing between different weather conditions, such as cloudy and sunny.

### 1.3. The Simulator

The simulator that this study utilizes was previously used for the study of driver vehicular interaction. Ref [25] as shown in the Figure 1. Although it is capable of recording a driver's behavior in detail, the setup was altered for this study. In particular, it is capable of recording multi-camera vehicle viewpoints. It was modified to reflect the real-world location of Colchester, United Kingdom. This includes a good variation of two-lane as well as single-lane roads. It also comprises a highly detailed interior as shown in Figure 2 as well as exterior, so the simulator capabilities can be extended even further if required for future studies. Moreover, a virtual camera was fitted to the top of the windshield just above the rear view mirror to better capture the environment ahead. Figure 3 shows the autonomous car that was used to record the necessary viewpoints for this study.

The virtual environment was roughly based on the small town of Colchester located in the Eastern United Kingdom. The initial goal was to construct a virtual motorway section based on the M25, but that was later amended for a more varied road environment. The total travel distance was over one mile, which consisted of a fair balance of double lane roads, roundabouts, and inner town single lane roads as shown in Figure 4.



Figure 1. Simulator.



**Figure 2.** Virtual Interior.



**Figure 3.** Virtual Car.

The virtual world also contained a wide range of randomly generated traffic cars that added to the realism and complexity of its real-world counterpart. The simulator was designed in such a way that it provided a considerable amount of modification control for the researcher. The goal was to make it accessible for anyone to start generating varied weather conditions for autonomous research. In addition, it has the ability to adjust for a virtually unlimited number of weather variations and the amount of images that can be generated at any given time.

#### 1.4. The Dataset

The generated dataset provides a plausible amount of varied weather conditions. The main classes include sunny, cloudy, foggy, and rainy. Each class then contains further subclasses involving the same class captured every hour from 9 AM to 4 PM. This methodology provided the most efficient learning material for CNNs and deep learning algorithms, as it provides the same location within varied lighting and weather conditions. For each recording session, the virtual car was allowed to run through the circuit, which resulted in the capture of approximately 2600 images. Each session was recorded on an hourly basis, i.e., for a clear day, weather for each session of driving was captured at 9:00, 10:00, 11:00, 12:00, 1:00, 2:00, 3:00, and 4:00. This provided a much needed variation in the overall shadow and lighting conditions for a varied dataset generation. Figure 5 shows the four main classes captured at various locations through different sessions. Table 1 shows the distribution of images per class. The resolution of each image was recorded at  $1280 \times 720$ ; the channels used were red, green, and blue. Notice that the validation images for the foggy class only consist of five images; this is because a foggy image is by far the most specific in color tone and channel information. Moreover, the quantity of validation images was set by the creators of the Berkeley Deep Drive dataset.





Figure 4. Proposed Environment.

Table 1. No. of Training images (Our dataset) and Testing images (BDD) per class distribution.

Class	Training	Testing
Clear	9613	1764
Cloudy	38,949	1677
Foggy	29,914	5
Rainy	29,857	396
Total	108,333	3842

Moreover, extensive care was taken to simulate secondary imperfections such as water droplets on the camera lens for distortion, traffic car signal bloom effects, and water shower behind traffic car wheels. Additional camera angles, such as left view, right view, and back view, were also captured to meet the challenge of the diverse task and absence of discriminatory features among various weather conditions.

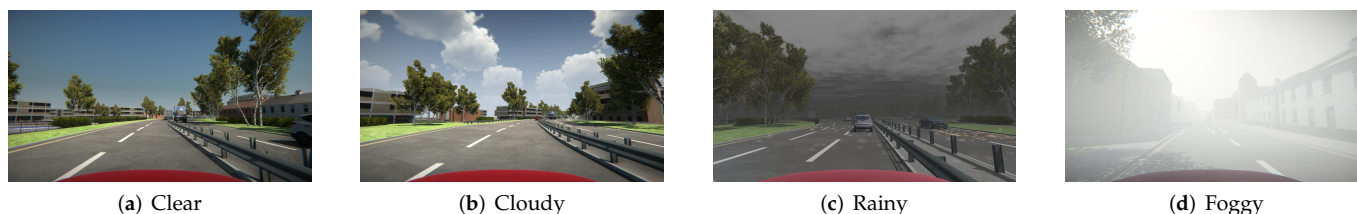


Figure 5. Synthetic Weather Dataset.

Our synthetic dataset was evaluated on the Berkeley Deepdrive dataset [26], because it provides a considerable variation of varying weather conditions in a fairly balanced annotated pattern as shown in the Figure 6.

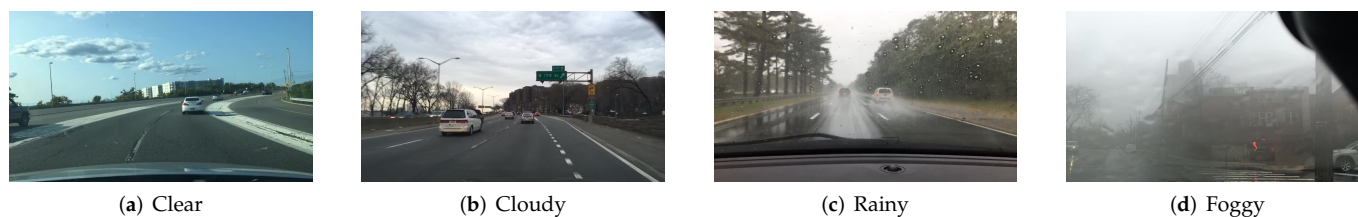


Figure 6. BDD (Berkeley Deep Dive) Dataset.

## 2. Weather Classification

To check to what extent our synthetic dataset was useful for weather classification, we applied a number of deep learning networks to test the dataset. One of the most famous deep learning architectures, Convolutional Neural Networks (CNN) have been able to perform various vision tasks with capabilities comparable to humans. However, CNN's performance is highly dependent upon the large size of the training data. This problem intensifies for a weather classification task as the real-time weather variation data availability for self-driving cars is difficult [41] to attain. Based on this problem, we tried to gauge whether different CNN architectures trained using synthetic data were good enough to classify the weather captured in real time.

Transfer learning is a powerful machine learning technique, which allows re-usage of a model for different tasks. It has gained immense popularity for computer vision tasks where pretrained CNN architectures are used as the standard starting point given the vast resources in terms of computation and time required to develop CNNs from scratch.

The pipeline used for the work described in this paper is visually represented in Figure 7. The pipeline operates in a fashion where the weights of the entire pretrained network were frozen except the classification layers at the end. The softmax layer is added for multi weather classification. The softmax layer (4,1) is added because the number of classes is 4. The classifier layers of the pretrained networks were retrained on the proposed synthetic weather dataset. The test real-time images were passed through the retrained CNN models to extract predict the network's accuracy.

The classifier layers were trained on the synthetic images and tested on the real-world dataset Berkeley DeepDrive [26]. After performing the experiment, the mean Average Precision (mAP) was calculated for each of the models.

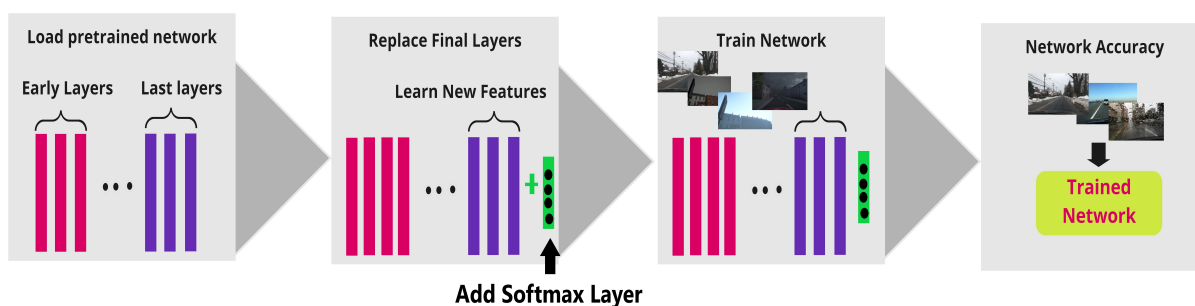


Figure 7. Pipeline: Step 1: Load the pretrained network, Step 2: Unfreeze the classification layers and add a softmax layer (4,1), Step 3: Train the weights of the classification layers with the synthetic dataset, Step 4: Test the network accuracy with a real time test dataset.

### Pretrained Model

The pretrained models used for predicting weather are described in depth in the following subsections:

#### 1. AlexNet

AlexNet [42] can easily be considered as a breakthrough network that has popularized deep learning approaches against traditional machine learning approaches. With eight layers, AlexNet won the famous object recognition challenge known as called the

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012. It is a variant of an artificial neural network, where the hidden layers comprise convolutional layers, pooling layers, fully connected layers, and normalization layers. A few of its standout features are the addition of nonlinearity, use of dropouts to overcome overfitting, and a reduction in network size due to overfitting.

2. **VGGNET**  
VGGNET [43], a 19-layer network, was proposed as a step forward from AlexNet and was a runner up in the ILSVRC-2014 challenge. As an improvement, the large kernel size of the first and second convolutional layers of AlexNet net were replaced by multiple  $3 \times 3$  size kernel filters. The small-size filters allow the network to have a large number of weight layers. Nonlinearity in decision making was incremented by adding  $1 \times 1$  convolution layer.
3. **GoogleLeNet**  
GoogleLeNet [44], a 22-layer network, was the winner of the ILSVRC-2014 challenge. It was proposed as a variant of an inception network to reduce the computational complexity of traditional CNNs. The inspection layer had variable receptive fields to capture sparse correlation patterns in the feature map.
4. **Residual Network**  
Residual Network [45] was the winner of the ILSVRC-2015 challenge. It was proposed with the aim of overcoming the problem of a vanishing gradient in ultra-deep CNN by introducing residual blocks. Various versions of Residual Network (ResNet) were developed by varying the number of layers as 34, 50, 101, 152, and 1202. The popular Residual Networks ResNet50 and ResNet101 were used in our experiment.

### 3. Results

In this section, we evaluate the various CNN models trained on our proposed synthetic dataset and compare their performance on the BDD dataset. The synthetic dataset contained images annotated with four weather classes. The number of epochs was set to 500. The learning rate of the stochastic gradient descent (SGD) optimizer for cross-entropy minimization was set to 0.0001. These parameters were deduced empirically by analyzing the training loss. As a regularization strategy during the training phase, two data augmentation techniques were used for all architectures. The first technique took random crops of training images, and the second technique applied rotation to the training images. All the algorithms were implemented using MATLAB, and the experiments were performed on a Tesla K80 with 12GB GPU memory and 916.77 GB storage.

Each experiment for calculating accuracy for the given pretrained model on the testing dataset was conducted 10 times. Then, the average accuracy for each model was calculated and denoted as mean Average Precision (mAP). The results tabulated in Table 2 show that the mAP for all the architectures varied between 60% and 74%. The accuracy variation over each epoch is shown in Figures 8 and 9.

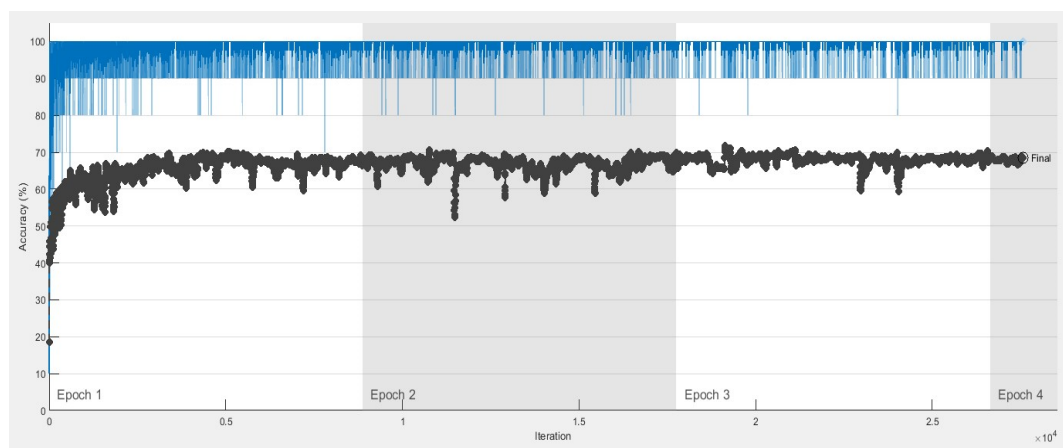
**Table 2.** Results from CNN evaluations.

Architecture	mAP	Trainable Parameter	Time (min)
AlexNet	$0.6856 \pm 0.012$	61M	986
VGGNET	$0.7334 \pm 0.023$	138M	2930
GoogleLeNet	$0.6034 \pm 0.009$	7M	618
ResNet50	$0.6183 \pm 0.025$	26M	1020
ResNet101	$0.63 \pm 0.006$	44M	1242

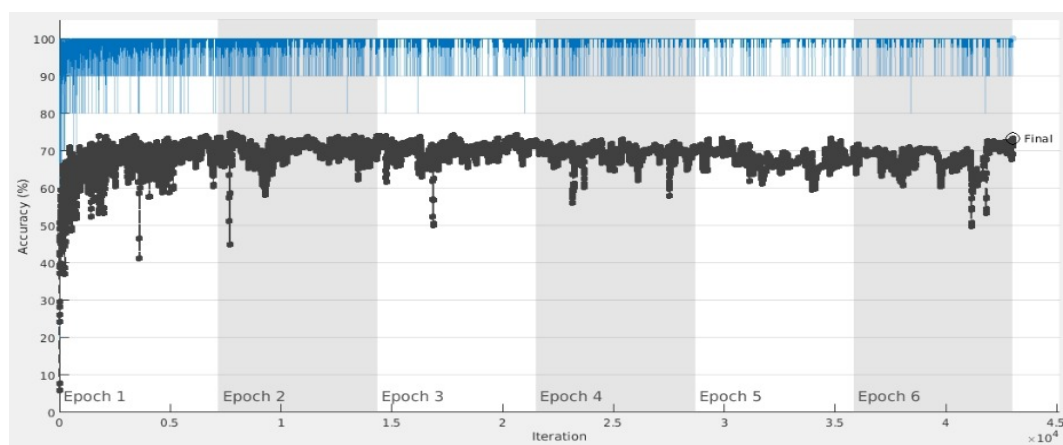
ResNet architectures achieved the lowest accuracy due to their complicated multi-branch designs, i.e., residual addition in ResNet, as the fine tuning of hyper-parameters and other customization becomes difficult. Given the constraint of hardware in self-driving cars, the inference is slowed down along with the reduction in memory utilization [46].

The most efficient weather classification accuracy on the testing dataset was achieved by the VGGNet architecture. These results indicate that the optimization achieved by the inclusion of smaller kernel filters at the initial convolutional layers had a positive effect on the overall task of weather classification. The universal effectiveness of the performance of VGGnet to extract deep features has also been affirmed previously by the state-of-the-art PFGFE-Net [13] that uses VGGNet as a backbone.

The training times from Table 2 reveal that they were directly proportional to the parameter due to the backpropagation process to retrain the weights of classification layers. However, with a closer look, one can conclude the training of the weather classification process for self driving cars was performed in the cloud, and it was a one time process. In the particular case of VGG, the training was time intensive, but it was a one time task. The testing time for determining weather from a single image on average using VGG was 15.67 fps, which is real-time efficient. Concluding the potential of this type of architecture on a classification task with a paucity of datasets draws attention to the possibility of more experimentation by training on a larger synthetic dataset with more diverse classes.



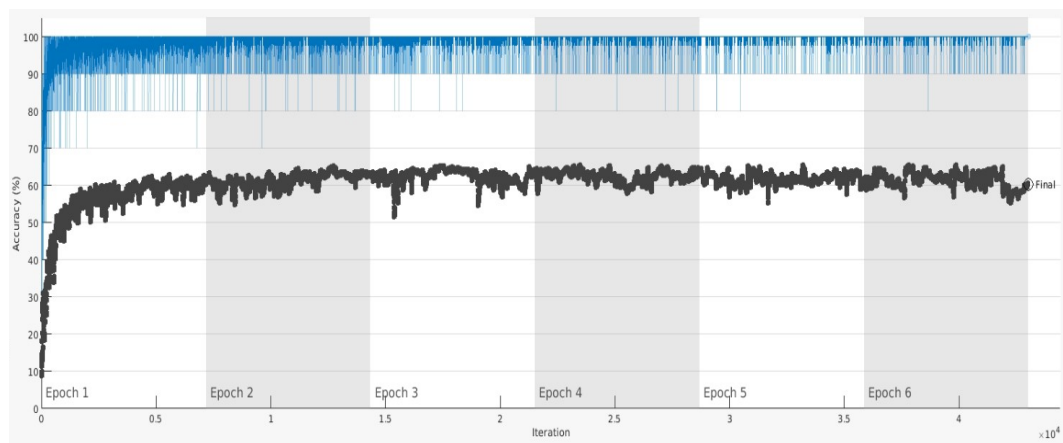
(a) AlexNet



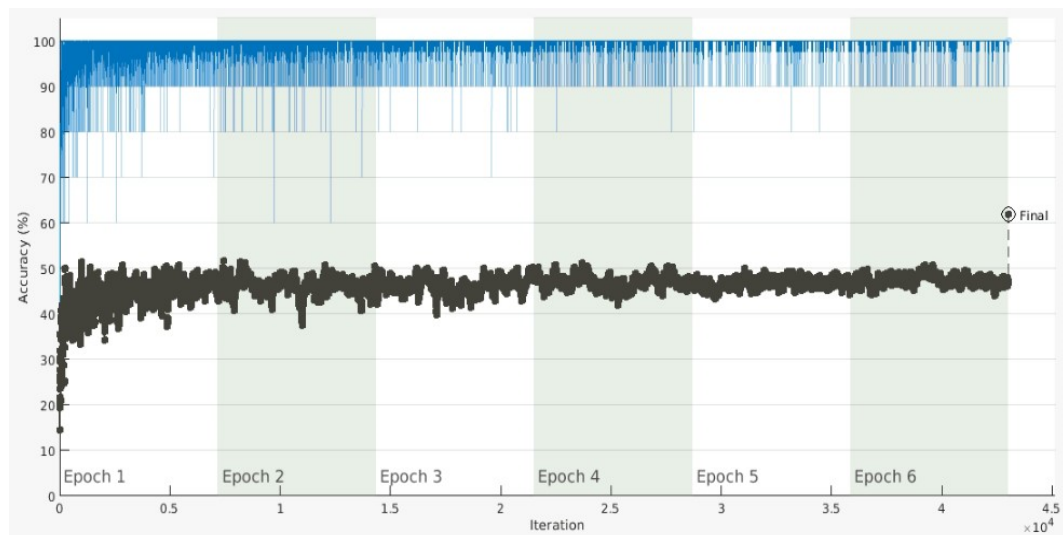
(b) VGG

Figure 8. Cont.

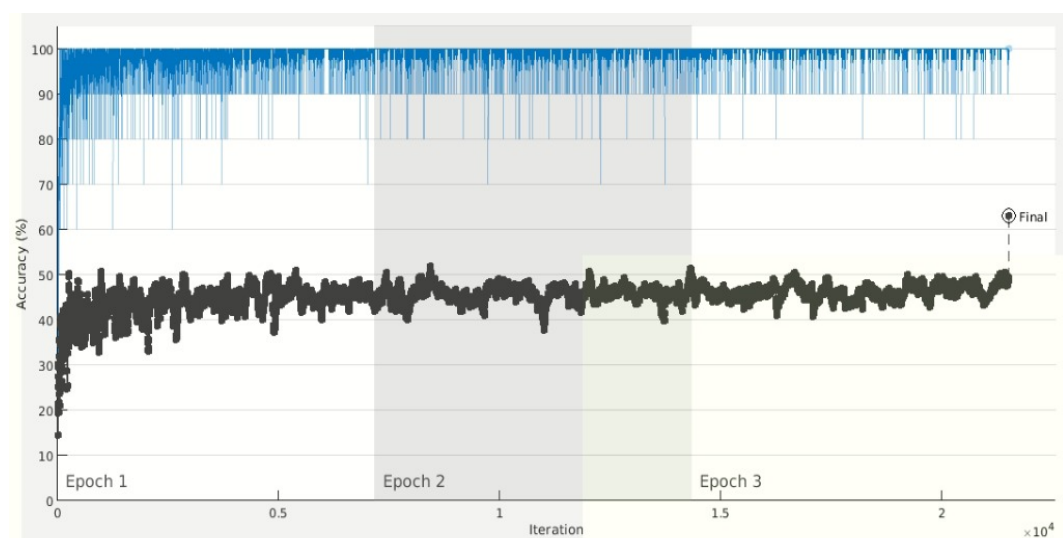




(c) GoogleNet

**Figure 8.** Accuracy variation over each epoch for (a) AlexNet, (b) VGG, and (c) GoogleLeNet models.

(a) ResNet50



(b) ResNet101

**Figure 9.** Accuracy variation over each epoch for Residual Networks (a) ResNet50 and (b) ResNet101.

#### 4. Conclusions

This paper highlighted the development of a custom driver simulator that was able to produce complex weather scenarios in immaculate detail. The manuscript also highlighted the possibility of using synthetic datasets to train a classifier in the context of weather classification and provided a synthetic dataset validated on the real-world Berkeley Deep-Drive [26]. The proposed dataset was also hybrid in nature as synthetic images from different camera angles were taken. The weather classification accuracy was derived by testing classifiers on different real-time datasets, which allowed the persistent problem of bias in vision datasets to be tackled. The study proves that a persistent visual fidelity is important in generating realistic datasets for computer vision-based datasets. Furthermore, with advent of computer graphics it will be possible to achieve advanced photorealism in the generated datasets game engines such as Unity and Unreal are embedding new visualization techniques to further enable data scientists to generate accurate synthetic data for vision based tasks.

**Author Contributions:** Conceptualization, S.M. and Z.K.; Data curation, S.M. and Z.K.; Formal analysis, S.M.; Investigation, S.M. and Z.K.; Methodology, S.M. and Z.K.; Project administration, S.M.; Resources, S.M. and Z.K.; Software, S.M. and Z.K.; Supervision, S.E., K.M.-M. and A.H.-S.; Validation, S.M. and Z.K.; Visualization, Z.K.; Writing—original draft, S.M. and Z.K.; Writing—Review and editing, S.M. and Z.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the UK Engineering and Physical Sciences Research Council through Grants EP/R02572X/1, EP/P017487/1 and EP/V000462/1. This work was also supported by Ministerio de Ciencia e Innovacion (MCI), Agencia Estatal de Investigacion (AEI) and Fondo Europeo de Desarrollo Regional (FEDER), RTI2018-095209-B-C21 (MCI/AEI/FEDER, UE); Agencia de Gestio d'Ajuts Universitaris i de Recerca grant numbers 2017-SGR-1597; and CERCA Programme/Generalitat de Catalunya.

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

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The proposed weather dataset will be available freely in due course.

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

#### References

1. Position Paper on Road Worthiness. Available online: [https://knowledge-base.connectedautomateddriving.eu/wp-content/uploads/2019/08/CARTRE-Roadworthiness-Testing-Safety-Validation-position-Paper\\_3\\_After\\_review.pdf](https://knowledge-base.connectedautomateddriving.eu/wp-content/uploads/2019/08/CARTRE-Roadworthiness-Testing-Safety-Validation-position-Paper_3_After_review.pdf) (accessed on 26 January 2022).
2. Cools, M.; Moons, E.; Wets, G. Assessing the impact of weather on traffic intensity. *Weather. Clim. Soc.* **2010**, *2*, 60–68. [CrossRef]
3. Achari, V.P.S.; Khanam, Z.; Singh, A.K.; Jindal, A.; Prakash, A.; Kumar, N. I 2 UTS: An IoT based Intelligent Urban Traffic System. In Proceedings of the 2021 IEEE 22nd International Conference on High Performance Switching and Routing (HPSR), Paris, France, 7–10 June 2021; pp. 1–6.
4. Kilpeläinen, M.; Summala, H. Effects of weather and weather forecasts on driver behaviour. *Transp. Res. Part F Traffic Psychol. Behav.* **2007**, *10*, 288–299. [CrossRef]
5. Lu, C.; Lin, D.; Jia, J.; Tang, C.K. Two-class weather classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 3718–3725.
6. Roser, M.; Moosmann, F. Classification of weather situations on single color images. *IEEE Intell. Veh. Symp.* **2008**, *10*, 798–803.
7. Zhang, Z.; Ma, H. Multi-class weather classification on single images. In Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, 27–30 September 2015; pp. 4396–4400.
8. Zhang, T.; Zhang, X. ShipDeNet-20: An only 20 convolution layers and <1-MB lightweight SAR ship detector. *IEEE Geosci. Remote Sens. Lett.* **2020**, *18*, 1234–1238.
9. Zhang, T.; Zhang, X.; Shi, J.; Wei, S.; Wang, J.; Li, J.; Su, H.; Zhou, Y. Balance scene learning mechanism for offshore and inshore ship detection in SAR images. *IEEE Geosci. Remote Sens. Lett.* **2020**, *19*, 4004905. [CrossRef]
10. Zhang, T.; Zhang, X.; Ke, X.; Liu, C.; Xu, X.; Zhan, X.; Wang, C.; Ahmad, I.; Zhou, Y.; Pan, D.; et al. HOG-ShipCLSNet: A novel deep learning network with hog feature fusion for SAR ship classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5210322. [CrossRef]
11. Zhang, T.; Zhang, X. Squeeze-and-excitation Laplacian pyramid network with dual-polarization feature fusion for ship classification in sar images. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 4004905. [CrossRef]

12. Zhang, T.; Zhang, X.; Ke, X. Quad-FPN: A novel quad feature pyramid network for SAR ship detection. *Remote Sens.* **2021**, *13*, 2771. [[CrossRef](#)]
13. Zhang, T.; Zhang, X. A polarization fusion network with geometric feature embedding for SAR ship classification. *Pattern Recognit.* **2022**, *123*, 108365. [[CrossRef](#)]
14. Dhiraj; Khanam, Z.; Soni, P.; Raheja, J.L. Development of 3D high definition endoscope system. In *Information Systems Design and Intelligent Applications*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 181–189.
15. Khanam, Z.; Raheja, J.L. Tracking of miniature-sized objects in 3D endoscopic vision. In *Algorithms and Applications*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 77–88.
16. Aslam, B.; Saha, S.; Khanam, Z.; Zhai, X.; Ehsan, S.; Stolkin, R.; McDonald-Maier, K. Gamma-induced degradation analysis of commercial off-the-shelf camera sensors. In Proceedings of the 2019 IEEE SENSORS, Montreal, QC, Canada, 27–30 October 2019; pp. 1–4.
17. Khanam, Z.; Saha, S.; Aslam, B.; Zhai, X.; Ehsan, S.; Cazzaniga, C.; Frost, C.; Stolkin, R.; McDonald-Maier, K. Degradation measurement of kinect sensor under fast neutron beamline. In Proceedings of the 2019 IEEE Radiation Effects Data Workshop, San Antonio, TX, USA, 8–12 July 2019; pp. 1–5.
18. Khanam, Z.; Aslam, B.; Saha, S.; Zhai, X.; Ehsan, S.; Stolkin, R.; McDonald-Maier, K. Gamma-Induced Image Degradation Analysis of Robot Vision Sensor for Autonomous Inspection of Nuclear Sites. *IEEE Sens. J.* **2021**, *1*. [[CrossRef](#)]
19. Gil, D.; Hernández-Sabaté, A.; Enconniere, J.; Asmayawati, S.; Folch, P.; Borrego-Carazo, J.; Piera, M.À. E-Pilots: A System to Predict Hard Landing During the Approach Phase of Commercial Flights. *IEEE Access* **2021**, *10*, 7489–7503. [[CrossRef](#)]
20. Hernández-Sabaté, A.; Yauri, J.; Folch, P.; Piera, M.À.; Gil, D. Recognition of the Mental Workloads of Pilots in the Cockpit Using EEG Signals. *Appl. Sci.* **2022**, *12*, 2298. [[CrossRef](#)]
21. Yousefi, A.; Amidi, Y.; Nazari, B.; Eden, U. Assessing Goodness-of-Fit in Marked Point Process Models of Neural Population Coding via Time and Rate Rescaling. *Neural Comput.* **2020**, *32*, 2145–2186. [[CrossRef](#)] [[PubMed](#)]
22. Azizi, A.; Tahmid, I.; Waheed, A.; Mangaokar, N.; Pu, J.; Javed, M.; Reddy, C.K.; Viswanath, B. T-Miner: A Generative Approach to Defend Against Trojan Attacks on DNN-based Text Classification. *arXiv* **2021**, arXiv:2103.04264.
23. Qian, Y.; Almazan, E.J.; Elder, J.H. Evaluating features and classifiers for road weather condition analysis. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 4403–4407. [[CrossRef](#)]
24. Minhas, S.; Hernández-Sabaté, A.; Ehsan, S.; McDonald-Maier, K.D. Effects of Non-Driving Related Tasks During Self-Driving Mode. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 1391–1399. [[CrossRef](#)]
25. Minhas, S.; Hernández-Sabaté, A.; Ehsan, S.; Díaz-Chito, K.; Leonardis, A.; López, A.M.; McDonald-Maier, K.D. LEE: A Photo-realistic Virtual Environment for Assessing Driver-Vehicle Interactions in Self-Driving Mode. In *Computer Vision—ECCV 2016 Workshops, Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016*; Hua, G., Jégou, H., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 894–900.
26. Yu, F.; Chen, H.; Wang, X.; Xian, W.; Chen, Y.; Liu, F.; Madhavan, V.; Darrell, T. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. *arXiv* **2020**, arXiv:1805.04687.
27. Lim, S.H.; Ryu, S.K.; Yoon, Y.H. Image Recognition of Road Surface Conditions using Polarization and Wavelet Transform. *J. Korean Soc. Civ. Eng.* **2007**, *27*, 471–477.
28. Kawai, S.; Takeuchi, K.; Shibata, K.; Horita, Y. A method to distinguish road surface conditions for car-mounted camera images at night-time. In Proceedings of the 2012 12th International Conference on ITS Telecommunications, Taipei, Taiwan, 5–8 November 2012; pp. 668–672. [[CrossRef](#)]
29. Kurihata, H.; Takahashi, T.; Ide, I.; Mekada, Y.; Murase, H.; Tamatsu, Y.; Miyahara, T. Rainy weather recognition from in-vehicle camera images for driver assistance. In Proceedings of the IEEE Proceedings. Intelligent Vehicles Symposium, Las Vegas, NV, USA, 6–8 June 2005; pp. 205–210. [[CrossRef](#)]
30. Yan, X.; Luo, Y.; Zheng, X. Weather Recognition Based on Images Captured by Vision System in Vehicle. In *Advances in Neural Networks—ISNN 2009*; Yu, W., He, H., Zhang, N., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 390–398.
31. Lu, C.; Lin, D.; Jia, J.; Tang, C. Two-Class Weather Classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2510–2524. [[CrossRef](#)]
32. Song, H.; Chen, Y.; Gao, Y. Weather Condition Recognition Based on Feature Extraction and K-NN. In *Foundations and Practical Applications of Cognitive Systems and Information Processing*; Sun, F., Hu, D., Liu, H., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 199–210.
33. Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; Koltun, V. CARLA: An Open Urban Driving Simulator. In Proceedings of the 1st Annual Conference on Robot Learning, Mountain View, CA, USA, 13–15 November 2017, pp. 1–16.
34. Ros, G.; Sellart, L.; Materzynska, J.; Vazquez, D.; Lopez, A.M. The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes. In Proceedings of the The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
35. Feng, H.; Fan, H. 3D weather simulation on 3D virtual earth. In Proceedings of the 2012 IEEE International Geoscience and Remote Sensing Symposium, Munich, Germany, 22–27 July 2012; pp. 543–545. [[CrossRef](#)]
36. Zhang, T.; Zhang, X. High-speed ship detection in SAR images based on a grid convolutional neural network. *Remote Sens.* **2019**, *11*, 1206. [[CrossRef](#)]

37. Zhang, T.; Zhang, X.; Shi, J.; Wei, S. HyperLi-Net: A hyper-light deep learning network for high-accurate and high-speed ship detection from synthetic aperture radar imagery. *ISPRS J. Photogramm. Remote Sens.* **2020**, *167*, 123–153. [[CrossRef](#)]
38. Zhang, T.; Zhang, X.; Shi, J.; Wei, S. Depthwise separable convolution neural network for high-speed SAR ship detection. *Remote Sens.* **2019**, *11*, 2483. [[CrossRef](#)]
39. Zhang, T.; Zhang, X.; Ke, X.; Zhan, X.; Shi, J.; Wei, S.; Pan, D.; Li, J.; Su, H.; Zhou, Y.; et al. LS-SSDD-v1. 0: A deep learning dataset dedicated to small ship detection from large-scale Sentinel-1 SAR images. *Remote Sens.* **2020**, *12*, 2997.
40. Zhang, T.; Zhang, X.; Liu, C.; Shi, J.; Wei, S.; Ahmad, I.; Zhan, X.; Zhou, Y.; Pan, D.; Li, J.; et al. Balance learning for ship detection from synthetic aperture radar remote sensing imagery. *ISPRS J. Photogramm. Remote Sens.* **2021**, *182*, 190–207. [[CrossRef](#)]
41. Guerra, J.C.V.; Khanam, Z.; Ehsan, S.; Stolkin, R.; McDonald-Maier, K. Weather Classification: A new multi-class dataset, data augmentation approach and comprehensive evaluations of Convolutional Neural Networks. In Proceedings of the 2018 NASA/ESA Conference on Adaptive Hardware and Systems (AHS). IEEE, Edinburgh, UK, 6–9 August 2018; pp. 305–310.
42. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [[CrossRef](#)]
43. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
44. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
45. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
46. Ding, X.; Zhang, X.; Ma, N.; Han, J.; Ding, G.; Sun, J. Repvgg: Making vgg-style convnets great again. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13733–13742.