A Benchmark and Investigation of Deep-Learning-Based Techniques for Detecting Natural Disasters in Aerial Images

Demetris Shianios[0009−0005−8266−0727], Christos Kyrkou[0000−0002−7926−7642], and Panayiotis S. Kolios[0000−0003−3981−993X]

KIOS Research and Innovation Center of Excellence, University of Cyprus, Cyprus
https://www.kios.ucy.ac.cy/
{shianios.demetris,kyrkou.christos,kolios.panayiotis}@ucy.ac.cy

Abstract. Rapid emergency response and early detection of hazards caused by natural disasters are critical to preserving the lives of those in danger. Deep learning can aid emergency response authorities by automating UAV-based real-time disaster recognition. In this work, we provide an extended dataset for aerial disaster recognition and present a comprehensive investigation of popular Convolutional Neural Network models using transfer learning. In addition, we propose a new lightweight model, referred to as *DiRecNet*, that provides the best trade-off between accuracy and inference speed. We introduce a tunable metric that combines speed and accuracy to choose the best model based on application requirements. Lastly, we used the Grad-CAM explainability algorithm to investigate which models focus on human-aligned features. The experimental results show that the proposed model achieves a weighted F1-Score of 96.15% on four classes in the test set. When utilizing metrics that consider both inference time and accuracy, our model surpasses other pre-trained CNNs, offering a more efficient and precise solution for disaster recognition. This research provides a foundation for developing more specialized models within the computer vision community.

Keywords: Natural Disasters Recognition · Image Classification · UAV (Unmanned Aerial Vehicle) · Deep Learning · Benchmark · Grad-CAM

1 Introduction

Natural disasters have been on the rise worldwide in recent years, with ecological and socioeconomic consequences. According to the United Nations Office for Disaster Risk Reduction, there were 7,348 disaster incidents between 2000 and 2019, resulting in 1,23 million deaths and US $3 trillion in economic losses [22]. The World Meteorological Organization, claims that over the last 50 years, a disaster related to weather, climate, or water hazard has occurred every day, killing 115 people and inflicting US $202 million in losses [4].

Unmanned Aerial Vehicles (UAVs) such as drones have emerged as effective tools for the early identification of these disasters due to their low cost, wide coverage area, and low risk to personnel. However, on-board processing presents its own set of issues, due to limited computational resources and low-power limits imposed by UAVs. As a result, the operational performance of the underlying computer vision algorithm is critical for autonomous UAVs to detect disasters in real-time. With the combination of Deep Learning, drones can be used for disaster classification, which can quickly and accurately identify affected areas, assess damage severity, and prioritize response efforts.

One crucial aspect of the successful implementation of deep learning models is the availability of a sufficient amount of dataset, which plays a significant role in their overall performance. However, gathering data, in the event of an emergency, is time-consuming and expensive, as it frequently involves human data processing and expert evaluation. The potential for evaluating deep learning models in such situations is constrained by the dearth of comprehensive datasets related to natural disasters. Furthermore, while there has been considerable research on algorithms for natural disaster detection in aerial images, explainable AI has not been extensively investigated for this domain in the existing literature. By providing an explainable visual representation of the image regions on which the model is focusing, image explainability algorithms can help emergency responders quickly identify the location and extent of a natural disaster, allowing them to respond more effectively and efficiently.

This work addresses these gaps by extending aerial image datasets for disaster recognition, including four classes; normal, earthquakes, floods, and wildfires encompassing a total of 16,723 images. We propose the DiRecNet CNN model and compared it to widely known pre-trained models such as EfficientNet-B0 [21], MobileNet-V2 [17] ResNet50 [8], VGG16 [19], DenseNet121 [9], Inception-ResNetV2 [20] NASNetMobile [23] and Xception [5] using transfer learning. The proposed CNN achieved a weighted F1-score of 96.15% in the test set and outperformed other pre-trained CNNs when considering inference time. In our study, we also conducted experiments on the explainability of the image using Gradient-weighted Class Activation (Grad-Cam) technique, with the objective of improving the explainability of the model. Using Grad-CAM, we better understood common failures or errors by emphasizing the significant areas of an image that contribute to a certain prediction. Overall, CNN-based deep learning models exhibit strong potential for real-time natural disaster detection.

2 Background and Related Work

Several innovative solutions have been developed for visual disaster recognition in recent years, which can be crucial for rapid response operations. Gadhavi et al. [7] proposed a model that uses transfer learning to recognize natural disasters using a video dataset. Aamir et al. [1] developed a binary model to detect the existence of a disaster and a classification model to identify different types of disasters. Agrawal and Meleet [2] fine-tuned the ResNet-50 model for disaster recognition and tested it on real-time and pre-recorded videos. Alam et al. [3] used transfer learning with various pre-trained CNN models to classify the MEDIC dataset.

Li et al. [14] used YOLOv3 for detection and various neural networks, including VGG, ResNet, and MobileNet, for classification on the LADI dataset [15].

The current state-of-the-art methods for disaster detection typically focus on identifying a single type of disaster. Some recent techniques aim for multi-class disaster detection, but their models are too large and have too many parameters for effective execution on unmanned aerial vehicles (UAVs) onboard hardware. Therefore, developing custom models that are tailored to the specific constraints and requirements of embedded systems on drones is crucial to achieving efficient and effective disaster recognition. It should be noted that most existing models may not incorporate explainable AI techniques, which can limit their usefulness in providing valuable insights to first responders in the field.

Previous studies have suffered from a lack of diversity in their datasets, some containing limited images or not aligning well with UAV viewpoints. Our work aims to address these limitations and establish a benchmark. Moreover, the lack of aerial perspective images in current datasets hinders natural disaster recognition. Some datasets focus exclusively on a single type of disaster, failing to represent the full spectrum of real-world scenarios. Geographic or temporal bias can further compromise representativeness, as certain datasets can draw from a restricted range of locations or events. Our proposed methodology aims to mitigate the limitations of biased datasets by incorporating diverse aerial imagery and promoting transparency in the decision-making process of our models. Furthermore, our model is optimized for deployment in embedded systems, such as drones, and achieves a favourable balance between speed and accuracy.

3 Proposed Approach

3.1 Dataset for Disaster Recognition using UAVs

Our aim was to create a benchmark for aerial natural disaster recognition suitable for UAV applications. To do so we start initially from the AIDER database [12,13] which had a similar purpose but a smaller number of images per disaster class which can result in overfitting, and poor generalization. In addition to these samples, we extracted images as frames of videos downloaded from YouTube searched using queries like "aerial" + "disaster","flood","collapsed building".

The data collection process involved scanning images to match the visual perspective of the UAV, and filtering out any irrelevant images, such as those that were blurred or not related to the disaster. The mean resolution ($width \times height$) for each class is; earthquake 667×1018, flood 595×884, normal 553×395, wildfire 1557×834. Overall, the images collected belong to commonly occurring natural disasters, earthquakes/collapsed buildings, floods, and wildfire/fire with an additional class, the normal case. Normal images do not reflect events, disasters, or any other aspects that could be related to catastrophic events. Fig. 1 shows samples from the dataset, while the summary of the data is explained in the Tab. 1. As a result, our contribution compared to the state-of-the-art is a newer, larger dataset containing a set of images of natural disasters that are also suitable for use in UAV applications for aerial disaster recognition.

	Earthquakes	**Floods**	**Wildfire/Fire**	**Normal**	**Total**
Train	1927	4063	3509	3900	13399
Validation	239	505	439	487	1670
Test	239	502	436	477	1654
Total	2405	5070	4384	4864	**16723**

Table 1. Proportion of images in each class within the train, validation, and test set.

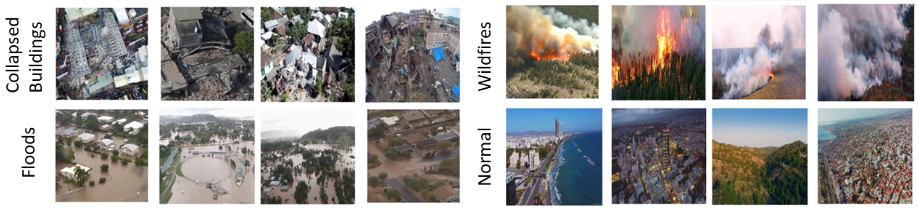

Fig. 1. Overview of aerial images from the Database.

3.2 Disaster Recognition Network Architecture

To enhance the operation of a UAV in emergency response, it is necessary to have lightweight algorithms that provide a good trade-off of complexity and accuracy. To this end and to motivate more work towards this area, we proposed the design of a custom CNN designed from scratch to be efficient by tailoring the use of convolutional layers and kernel sizes.

The custom CNN called *DiRecNet* consists of four main blocks, making it feasible for the model to learn hierarchical feature representations without reducing the feature map resolution too much. On the first two blocks, we use normal convolutional layers to extract richer low level features, while on the last two blocks we utilized separable convolutions to account for the fact that the channel size increases and has more efficient computations with a reduced number of operations and parameters.

In more detail, the model first passes the scaled images onto two consecutive normal convolutional layers. The former with a kernel size of 7×7 pixels and 16 filters, while the latter with 5×5 pixels and 16 filters. This follows modern network trends that apply larger kernels [16]. The smaller channel number is used to offset the larger kernel size. Batch normalization is applied just after these convolutions, with a max-pooling operation of stride 2×2 after that. The data points are then passed to the next block of two convolutional layers with kernel size of 3×3. The first convolution involves 32 filters, while the second has 64 convolution filters. Again, batch normalization is applied before the next max pooling layer. The third block involves two separable convolutions with 128 and 256 filters respectively and 3×3 size, followed by a batch normalization layer. A max-pooling operation is also applied with a pool size of 2×2. The last block is designed with two identical separable convolutions of the 512 filter and the size 3×3. Finally, a global average pooling layer is applied to flatten the

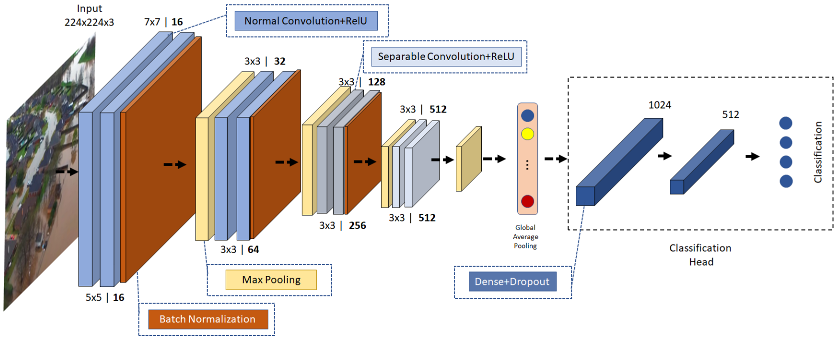

Fig. 2. Proposed Convolutional Neural Network Architecture.

features. These are then passed to a fully connected layer of 1024 neurons, before a dropout of 0.7. Another fully connected layer is applied with 512 neurons with a dropout of 0.5. The last layer of the model is a fully connected layer of size 4 as the number of classes. An overview of the model is depicted in Fig. 2.

3.3 Baseline Designs

To provide a useful benchmark for the constructed dataset, we compare the performance of various CNN models. We use transfer learning approach to modify and fine-tune CNNs trained on the ImageNet large scale dataset [11] to perform image disaster recognition. The transfer learning CNN models investigated in this work are: EfficientNet-B0 [21], MobileNet-V2 [17], ResNet-50 [8], VGG-16 [19], DenseNet-121 [9], InceptionResNet-V2 [20], NASNetMobile [23] and Xception [5]. These models capture a wide range of architectural design choices.

During the experiments, we freeze some layers of the pre-trained models and add some others to be trained. Specifically, in our experiments, we remove the last fully connected (FC) layer of each model and Global Average Pooling (GAP) was attached. On top of that, we added three fully connected layers with two dropouts in between. In general, the classification head architecture attached to the transfer learning models is the same as in the proposed model. For each pre-trained model, a pre-processing function was implemented using the TensorFlow library to standardize the input images based on the ImageNet dataset [6].

3.4 Data Pre-Processing and Training Process

The images in our data collection were scaled to 224 × 224 × 3 and standardized for DiRecNet therefore to change the distribution to have a mean of zero and a standard deviation of one. Random augmentations were applied to expand the diversity of the dataset and combat overfitting. Specifically, we applied rotation, zoom, horizontal shift, vertical shift, horizontal flip, and shear. We experimented with different color spaces, but chose RGB for the final experiments. We used

Models	PARAMS	Weighted F1	FPS	Score 1	Score 1	Score 1	Score 2
	(M)	(%)	$(1/s)$	Biased FPS	Biased F1	Balanced	
EfficientNet-B0 [21]	5.89	95.82	11.72	0.74	0.82	0.78	819.64
MobileNet-V2 [17]	4.10	93.77	**15.37**	**0.90**	0.77	0.84	259.57
ResNet50 [8]	26.21	**96.98**	6.87	0.47	0.77	0.62	1073.61
VGG16 [19]	15.77	94.50	5.22	0.29	0.55	0.42	146.22
DenseNet121 [9]	8.61	95.07	7.46	0.45	0.65	0.55	310.21
InceptionResNetV2 [20]	56.43	88.09	5.48	0.12	0.11	0.11	1.81
NASNetMobile [23]	5.88	90.65	12.96	0.66	0.49	0.57	25.18
Xception [5]	23.49	92.44	5.48	0.25	0.42	0.33	36.81
DiRecNet (Proposed)	**1.53**	96.15	14.05	0.89	**0.91**	**0.90**	**1235.12**

Table 2. Performance evaluations for disaster predictions.

slightly different training regimes for the pre-trained models and the DiRecNet model. The pre-trained models were frozen until the feature extraction layer, before attaching the global average pooling layer and fully connected layers. This implies that the initial layer weights are fixed and cannot be changed so as to preserve learned features. Then they were fine-tuned for 40 epochs with a learning rate of $1e-3$ and weight initialization based on ImageNet ILSVRC Challenge [11]. On the contrary, the proposed DiRecNet was trained from scratch for 300 epochs, with a reduced learning rate of $1e-4$. The batch size was set to 32, and Adam optimizer was selected for both DiRecNet and pre-trained models.

3.5 Explainability through Grad-CAM

Understanding how a deep learning model works and why it predicts a specific classification outcome is highly important for critical applications such as emergency management. Consequently, we move beyond the "black box" of CNN predictions and acquire a deeper understanding of how these models arrive at their decisions. This was achieved through experimentation with an explainable AI technique, known as Gradient-weighted Class Activation Mapping (Grad-CAM) [18]. The algorithm creates a coarse localization map that highlights key areas in the image for class prediction, by using the gradients of each target as they flow into the final convolutional layer. In this way, we can identify classes that are more challenging for the different models and understand whether additional context is needed and whether current state-of-the-art methods are suitable for the application of disaster recognition.

4 Experimental Evaluation and Results

4.1 Configuration and Evaluation Metrics

The experiments were carried out on the Linux operating system using the Tesla V100 Graphics Processor Unit, with 64GB RAM and CUDA version 10.2. We use TensorFlow [1] 2.4.1 as the deep learning framework along with Python [2]

[1] http://www.tensorflow.org
[2] http://www.python.org

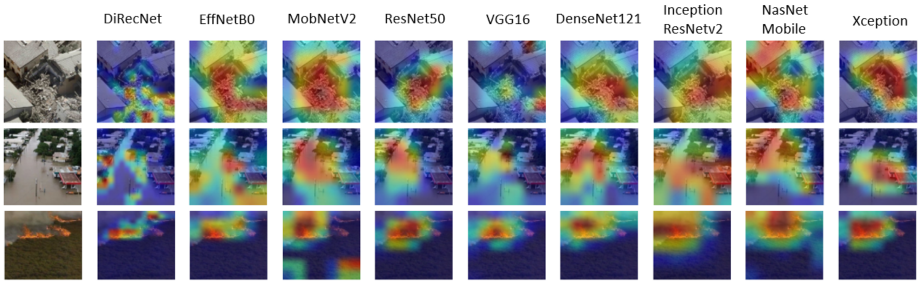

Fig. 3. Results of Grad-Cam algorithm for the different models. The heat-maps show that the classification's importance is dominated by the pixels associated with the disastrous occurrence.

version 3.8.0. To evaluate the performance of the models, we investigated two key performance indicators. These are the weighted F1 score and frames per second (FPS). This is particularly important because both performance and speed are crucial to detect natural disasters in real-time.

We then formulated a parametrizable score function as shown in Eq. 1 in a way to allow for choosing the trade-off between accuracy and speed. By setting the λ value, we can identify the model that performs best for a particular setting. In this work, we have chosen λ to be 0.7 to bias towards more accurate models, and 0.3 to bias towards speed. Additionally, to provide a more extensive evaluation we benchmark the models using a modified version of the scoring formula proposed in [10] for evaluating the combined effect of speed and accuracy as shown in Eq. 2, where we set the normalizing constant C to $1e27$.

$$\text{Score1} = \lambda \times F1_{\text{norm}} + (1 - \lambda) \times FPS_{\text{norm}} \tag{1}$$

$$\text{Score2} = \frac{2^{\text{F1}} \times \text{FPS}}{C} \tag{2}$$

However, prior to this, since the values of FPS and F1 have different ranges, we normalize them across all models by using the formula in Eq. 3, where values in x are squeezed into the range $[a, b]$ where a was set to 0.1 and b at 1, thus making the variables comparable to each other.

$$x_{norm} = (b - a)\frac{x - min(x)}{max(x) - min(x)} + a \tag{3}$$

4.2 Disaster Classification Evaluation

The general evaluation of the disaster classification performance of the models is shown in Tab.2. In summary, the models' weighted F1-Score ranges between 88% and 97%. ResNet-50 demonstrates optimal performance in terms of accuracy, while MobileNet-V2 exhibits the highest FPS, rendering it appropriate for processing multiple streams concurrently. By evaluating the performance of the models using a balanced approach that considers both speed and accuracy ($\lambda = 0.5$) according to the metric score in Eq. 1, our proposed model surpasses other models, achieving a score of 0.9, with MobileNet-V2 ranking second at 0.84. When biasing for FPS or F1, the proposed model remains the first or a very close second. Specifically, when prioritizing FPS ($\lambda = 0.3$), MobileNet-V2 achieves a score of 0.9, while the proposed model reaches 0.89. Conversely, when emphasizing F1, the proposed model leads with a score of 0.91, followed by EfficientNET-B0 at 0.82. Additionally, with respect to the metric presented in Eq. 2, the proposed model demonstrates superior performance compared to other methods. The proposed model achieves an overall score of 1235.12, while the second-best performing model, ResNet50, achieves a score of 1073.61. This shows that the heterogeneous design of mixing normal and separable convolutions provides a well-balanced solution with fewer parameters than other models.

4.3 Gram-CAM Evaluation

We interpret the decision of each model using Grad-CAM. In Fig. 3, all models predict the right class, and the heat-map produced by Grad-Cam is displayed. First, comparing the pre-trained models, we observe that the majority create a coarse grain heat map except for the VGG model. In contrast, while the proposed model correctly predicts the disaster type, it does so with a much sparser heat map. For example, in the collapsed building image, the region focuses more on the rubble rather than the building structure, while in the flood image, the model seems to distinguish the flood class based on the presence of surrounding buildings. In most pre-trained models, larger regions are emphasized, but for fire-related decisions, models exhibit more similar characteristics. The experiment exposed Grad-CAM's limitations, as highlighted regions may not always clarify disaster presence, like in collapsed building cases. In collapsed buildings, highlighting adjacent structures does not effectively explain the disaster's presence. We expect that this research can drive more efforts toward specialized explainability techniques for such applications.

5 Conclusion and Future Work

In this work, we presented a new larger dataset, offering 16,723 images, for aerial image recognition of disasters. We have explored the direct application of various existing CNN pre-trained models on this dataset to provide an initial benchmark. More importantly, we have shown that a heterogeneous CNN with mixed normal and separable convolutions can provide adequate trade-off between accuracy and speed and can thus be an optimal choice for these kinds of applications. Through this process, we have formulated a tunable metric to evaluate models. Based on the various scoring schemes, the proposed model still outperforms traditional pre-trained CNNs. Lastly, the gradient-weighted class activation mapping (Grad-CAM) method was used to visualize the input regions crucial for class predictions, demonstrating that different models provide a varying degree of granularity in explanations.

The experimental findings indicate that we were able to obtain classification outcomes that offered promising results for real-time disaster recognition from aerial images. Those initial results are encouraging, but there are still some challenges. Further, improvements and further investigation on more lightweight models are possible based on the experiments in this paper. Furthermore, an approach for multi-task scenarios where classification is combined with segmentation to provide more localized and precise identification of disasters is desired. Finally, it is worth investigating non-supervised approaches, since data for emergency management applications are scarce and difficult to annotate.

Acknowledgements

This work is supported by the European Union Civil Protection Call for proposals UCPM-2022-KN grant agreement No 101101704 (COLLARIS Network). The work is partially supported by the European Union's Horizon 2020 research and innovation program under grant agreement No 739551 (KIOS CoE - TEAMING) and from the Republic of Cyprus through the Deputy Ministry of Research, Innovation and Digital Policy.

References

1. Aamir, M., Ali, T., Irfan, M., Shaf, A., Azam, M.Z., Glowacz, A., Brumercik, F., Glowacz, W., Alqhtani, S., Rahman, S.: Natural disasters intensity analysis and classification based on multispectral images using multi-layered deep convolutional neural network. Sensors **21**(8), 2648 (2021)
2. Agrawal, T., Meleet, M., et al.: Classification of natural disaster using satellite & drone images with cnn using transfer learning. In: 2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES). pp. 1–5. IEEE (2021)
3. Alam, F., Alam, T., Hasan, M., Hasnat, A., Imran, M., Ofli, F., et al.: Medic: A multi-task learning dataset for disaster image classification. arXiv preprint arXiv:2108.12828 (2021)
4. Association, W.M., et al.: Wmo atlas of mortality and economic losses from weather, climate and water extremes (1970–2019). Tech. rep., Technical Report (2021)
5. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1251–1258 (2017)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
7. Gadhavi, V.B., Degadwala, S., Vyas, D.: Transfer learning approach for recognizing natural disasters video. In: 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS). pp. 793–798. IEEE (2022)

8. 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. pp. 770–778 (2016)
9. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4708 (2017)
10. Ignatov, A., Malivenko, G., Timofte, R.: Fast and accurate quantized camera scene detection on smartphones, mobile ai 2021 challenge: Report. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2558–2568 (2021)
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Communications of the ACM **60**(6), 84–90 (2017)
12. Kyrkou, C., Theocharides, T.: Deep-learning-based aerial image classification for emergency response applications using unmanned aerial vehicles. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 517–525 (2019). https://doi.org/10.1109/CVPRW.2019.00077
13. Kyrkou, C., Theocharides, T.: Emergencynet: Efficient aerial image classification for drone-based emergency monitoring using atrous convolutional feature fusion. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **13**, 1687–1699 (2020). https://doi.org/10.1109/JSTARS.2020.2969809
14. Li, Y., Wang, H., Sun, S., Buckles, B.: Integrating multiple deep learning models to classify disaster scene videos
15. Liu, J., Strohschein, D., Samsi, S., Weinert, A.: Large scale organization and inference of an imagery dataset for public safety. In: 2019 IEEE High Performance Extreme Computing Conference (HPEC). pp. 1–6 (Sep 2019). https://doi.org/10.1109/HPEC.2019.8916437
16. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11976–11986 (June 2022)
17. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4510–4520 (2018)
18. 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. pp. 618–626 (2017)
19. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
20. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-first AAAI conference on artificial intelligence (2017)
21. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. pp. 6105–6114. PMLR (2019)
22. Yaghmaei, N.: Human Cost of Disasters: An Overview of the Last 20 Years, 2000-2019. UN Office for Disaster Risk Reduction (2020)
23. Zoph, B., Vasudevan, V., Shlens, J., Le, Q.V.: Learning transferable architectures for scalable image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8697–8710 (2018)