Patch-based classification for building damage assessment using satellite imagery of natural disasters

Jie Wei, Ivan Miller

Abstract

Historically assessment of damage to buildings located in the areas affected by natural disasters was being performed manually by experts in particular type of disaster. Such an approach creates a bottleneck when disaster relief teams may not be able to properly prioritize the relief efforts in order to deliver help where it is most needed.

An automated solution capable of performing accurate classification of building damage is needed to allow for faster and more efficient prioritization of relief efforts. However, most of the research around building damage assessment focuses on semantic segmentation when satellite or aerial image is being analyzed in its entirety. Unfortunately, this method does not allow for targeted assessment of damage when the location and coordinates of the building of interest are known, which may be especially important to locate and evacuate to safety schools, hospitals, and other buildings that may have larger populations of vulnerable categories of people.

The paper presents an efficient way of classifying building damage with high level of accuracy using cropped-out patches of original satellite images that only contain one building per image together with a relatively small amount of area immediately close to each building. First, a new dataset for binary classification of building damage was created. Second, various data augmentation methods were being applied to the images from the new dataset and the data was then used for training of several deep learning models to achieve great performance.

Additionally, the winning classifier demonstrated high computational efficiency, plus, the small size of the weights of the learned model could allow the inference process to be effectively run on a conventional laptop by first responders directly in the field.

1. Introduction

After natural disasters, before responders can act in the affected area, they need to identify areas/buildings affected the most to prioritize the relief efforts. Historically the assessment of the damage was manually performed by experts which could be quite time-consuming, since the events qualified as natural disasters usually affect large territories. Fast and accurate assessment of the damage is critical to an effective response to disasters which ultimately saves lives

On top of that, since some buildings such as schools, hospitals, retirement homes etc., may need to be prioritized by disaster response teams to save potentially vulnerable people, there is a need for an automated solution capable of quickly and accurately classifying building damage when locations of the buildings of interest are known. In addition to FEMA similar approach may also be of interest to the military. Since manual process is extremely laborious and time consuming, the goal of this project is to reduce the time necessary to perform building damage assessment by creating a machine learning classifier based on deep learning methods that achieves an f1 score of at least 80%.

The data for this project comes from the xView2 competition hosted by the Defense Innovation Unit in 2019, it was collected from 19 natural disasters of 6 different types around the world (such as fire, wind, earthquake etc.) using pairs of pre- and post-event satellite images from Maxar. The dataset was created through a rigorous process of manual annotation of 850,73 building polygons performed by disaster response experts with specializations in various disaster types implemented to produce high-quality annotations of building damage using four ordinal labels of damage level: “no damage”, “minor damage,” “major damage,” and “destroyed” [1].

Fig.1 Example of a pre- and post-event original satellite images from the xBD dataset.

To achieve the above-mentioned goals, we made two novel contributions:

1) instead of using original satellite images from the xBD dataset we worked on cropping out each annotated building polygon together with the area within 50x50px patch that

was immediately close to the footprint of each building to produce building patches and consolidate them into a new dataset.
2) We took advantage of transfer learning and fine-tuned a pre-trained MobileNetV3 model on the newly created dataset and, as a result, presented the classifier that beats the performance benchmark.

2. Related Work

Most of the research in building damage assessment uses pre- and post-disaster event images and could be divided into three major groups:

1) Semantic segmentation that is usually built with encoder-decoder architecture [2]
2) Algorithms detecting change between pre- and post-even images [3]
3) Patch-based image classification [4].

As buildings look differently throughout the world so does the damage caused by a specific type of disaster, for example damage to a building from a wildfire will look completely different from a building damaged by flooding. As for the pre- and post-event images, the visual characteristics of the buildings significantly change in cases of serious damage. Destruction of a building causes normally seen because of shadows strong geometric shapes outlining the footprint of the building to disappear completely (see Fig.3). Also, for classification algorithms the spatial features such as texture and were more important than the spectral information [6].

3. Methods

Our goal was to build a machine learning algorithm that can be used for a fast and accurate assessment of damage to the buildings based on building coordinates using pre- and post-event satellite images. The proposed system is specifically focusing on small cropped-out images of buildings and areas that are immediately close to them.

3.1. New Dataset from xBD building patches

This project differentiates itself from a more common task of semantic segmentation, where the satellite image is evaluated in its entirety and the model first needs to differentiate the buildings from the background before getting to the main task of damage assessment. For patch-based classification the model is being trained on the data that went through a robust data preprocessing when each building polygon is being placed in the center of a much smaller patch with most of the area of the image occupied by the building itself (see Fig. 2).

Fig.2. Zoomed in example of the pair of original images with areas of future pre- and post-event building patches was outlined in red.

We experimented with several different sizes of the side of the training patch 150px, 100px, 64px, and 50px and achieved the highest performance with the smallest, 50x50px patch. We also implemented several filtering techniques and applied thresholds in order to:

1) Control for the building size relative to the size of the patch to create a more uniform set of building patches. For example, we would like to avoid situations when a building polygon occupied an area less than 30% of the image patch.
2) Applied thresholds on the pixel values in the building patch to account for satellite shifts and/or cloud coverage by filtering out extremely bright/dark images.

Additionally, to be able to perform a deeper analysis of classification results, for each building patch we recorded the corresponding satellite metadata, so that mapping between building polygons and original satellite images was possible (that includes information about each individual disaster, disaster type, satellite metadata etc.).

The newly created dataset contained 220,728 squared patches of the original satellite images each representing a centered building polygon and the area immediately surrounding the building, with each individual patch having a side of 50px. Even though only a fraction of all created patches was included in the training set due to multiple filtering conditions being employed, the total number of training images increased by 10x, making the classification task at hand more computationally expensive, which posed an additional challenge.

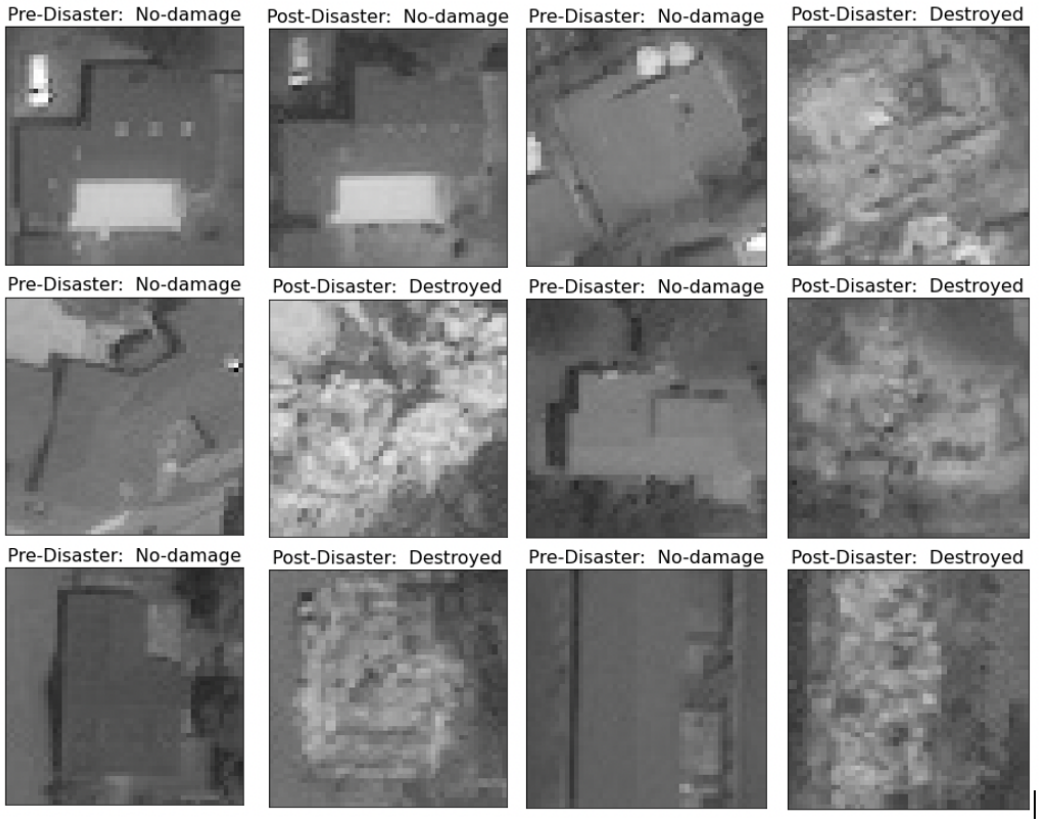


Fig. 3. 50x50px patches from Santa Rosa wildfire, 2017 (holdout dataset used for benchmarking and assessment of performance). Note a drastic difference in visual representation between pre- and post-event images that led to the highest validation accuracy of 95.97% among all disaster types.

3.2.Data augmentations

One of the common challenges in image classification is the lack of variability in the training data, so in addition to conventional transformations of the input images such as random flips, shifts, crops etc., we took advantage of the Albumentations library to incorporate a series of more sophisticated augmentations in the training pipeline created with PyTorch: we performed random shifts of values for each RGB channel (RGBShift), randomly changed of brightness and contrast (RandomBrightnessContrast), added noise by multiplying each image to a random number of array numbers (MultiplicativeNoise), and randomly changed hue and saturation (HueSaturationValue).

3.3. Transfer learning

We took advantage of transfer learning using pre-trained models of three different architectures: MobileNetV3, ResNet50, and ViT. We experimented with two main techniques 1) fixed feature extractor - freezing the weights of the pre-trained model and replacing the last layer with a new one initialized with random weights, 2) loading the weights of the pre-trained model and fine-

tuning the whole network on a new dataset [5]. Since fine-tuning showed significantly better performance, we chose that technique to be used with all three models.

We then used a 50% subset of the dataset to train each model with different combinations of loss functions, optimizers, and an adaptive learning rate for 10 epochs each to obtain some performance baseline. Initially the training was performed on a conventional laptop only equipped with CPU where one training epoch was taking anywhere from approximately 45min to complete (MobileNetV3 and ResNet50 models), up to 1h and 30min for ViT. Due to the prohibitive amount of time necessary to effectively train deep learning models on our relatively large dataset of building patches, we had to switch to NVIDIA A100 GPU. That yielded an immediate 10x increase in training speed, which allowed us to incorporate significantly more robust image augmentations, so it was decided to retrain each model for 10 epochs to a establish uniform baseline.

Table 1. Comparison of performance after fine-tuning MobileNetV3, ResNet50, and ViT models on building patches created from xBD dataset. Results include training during benchmarking (10 epochs on GPU):

Model	F1 Score	Precision	Recall	Validation Accuracy	Training Time*	Number of Epochs	Model Size	Inference Time
ResNet50_10	72.87%	78.79%	67.78%	86.56%	60m 27s	10	94.4Mb	4m 36s
MobileNet_v3_10	72.45%	78.23%	67.47%	86.62%	58m 59s	10	22.1Mb	5m 26s
ViT l 32_10	64.90%	61.00%	69.34%	74.31%	101m 26s	10	1.23Gb	13m 21s

ViT showed the worst performance across all tracked benchmarks with 1.7x longer time needed to train the model, 2.6x longer time to run inference on the holdout set, plus, the size of the model itself was over 55x larger when compared to the MobileNetV3 that showed the second best F1 score of 72.45%. Due to the above-mentioned factors, we decided that removing ViT from the experiment would allow us to free up computational resources and shift them to a longer training of ResNet50 and MobileNetV3 models, since both showed more promising results during the benchmarking (see Table 1 for details).

Both ResNet50 and MobileNetV3 were then retrained on the full training set (2x increase compared to the “benchmarking” phase) for 40 epochs each using NVIDIA A100 GPU, and the process took approximately 7 hours for each model. In the Fig.4 you could see a comparison of training and validation accuracy side-by-side with training and validation loss for both MobileNetV3 and ResNet50 models. Note a similar looking drop in validation accuracy/ spike in validation loss that happened during training of both models around the same time: epoch #6 / epoch #4 for MobileNet and ResNet respectively.

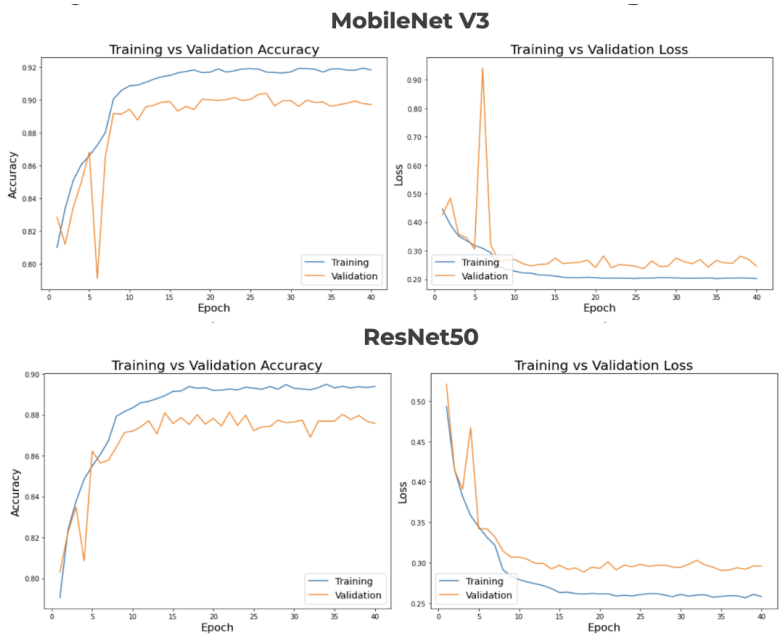


Fig.4. Training vs. validation accuracy and loss for MobileNetV3 and ResNet50 model during training for 40 epochs on NVIDIA A100 Tensor Core GPU.

3.4. Holdout Dataset

We separated 7,550 building patches into a holdout set to be used for analyses of performance of each model on unseen data. With this dataset we also took advantage of the metadata saved during preprocessing of the xBD dataset which allowed us to create mapping between each patch representing a building polygon and original satellite images and bring in data on disaster type, satellite metadata etc. for each patch. Fig. 7 shows the breakout pre- and post-event by damage type.

3.5. Comparison of performance of ResNet50 and MobileNetV3

Longer training of the ResNet50 model on a larger amount of data brought noticeable gains in recall, which increased by 5.12%, however, even after more than 7 hours of training the improvement in F1 score and precision was only 3.07% and 0.45% respectively.

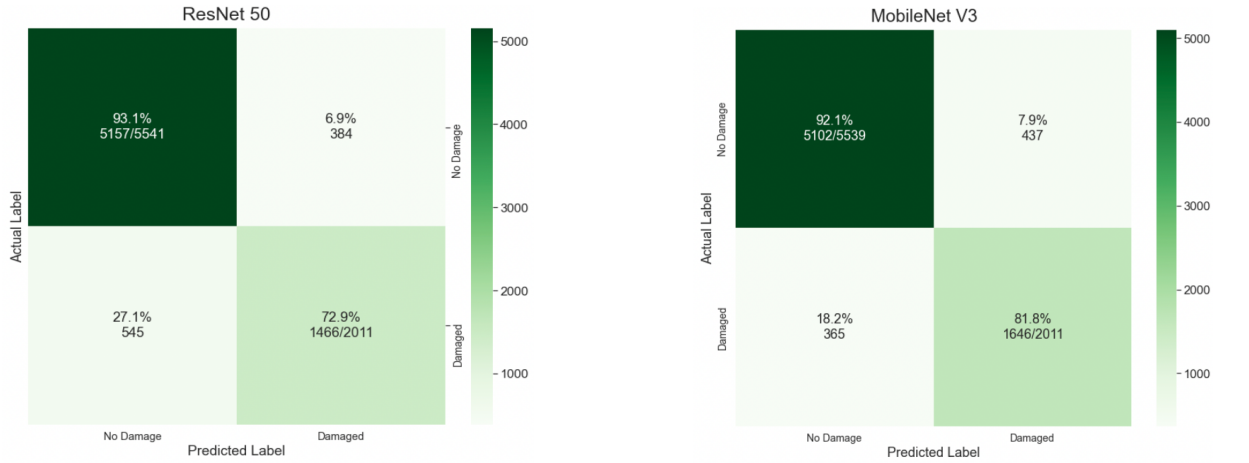


Fig.5. confusion matrix visualizing the results of inference on a 7,550-image holdout set by ResNet50 (left) and MobileNetV3 (right) models.

As seen in Fig.5 ResNet50 still struggled to recognize the damage and misclassified over 27% of the data. At the same time, MobileNetV3 showed the best overall result, achieving an 80.41% F1-score (and effectively beating the 80% benchmark) and clocking in an impressive 89.35% validation accuracy (see detailed results for both models in Table 2).

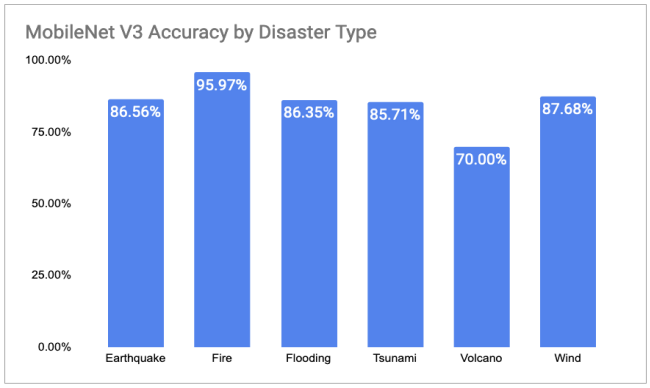
Table 2. Results of fine-tuning of MobileNetV3, ResNet50, and ViT models on building patches created from xBD dataset. Results include training during benchmarking (10 epochs) and final training for 40 epochs for MobileNetV3, ResNet50.

Model	F1 Score	Precision	Recall	Validation Accuracy	Training Time*	Number of Epochs	Model Size	Inference Time
MobileNet_v3_40	80.41%	79.02%	81.85%	89.35%	427m 21s	40	22.1Mb	5m 33s
ResNet50_40	75.94%	79.24%	72.90%	87.70%	426m 31s	40	94.4Mb	4m 44s
ResNet50_10	72.87%	78.79%	67.78%	86.56%	60m 27s	10	94.4Mb	4m 36s
MobileNet_v3_10	72.45%	78.23%	67.47%	86.62%	58m 59s	10	22.1Mb	5m 26s
ViT l 32_10	64.90%	61.00%	69.34%	74.31%	101m 26s	10	1.23Gb	13m 21s

* All models were trained on NVIDIA A100 Tensor Core GPU (200GiB RAM, 30 vCPUs)

3.6. Analysis of misclassified patches using satellite metadata

When looking at a breakout of the classification results by the type of disaster, we see that, despite imbalance of the holdout dataset by disaster type, outside of extreme examples, such as “Volcano” disaster type represented by only 10 patches, the model achieves a minimum accuracy of 85.71%. Interestingly, the highest validation accuracy of 95.97% was achieved across image patches that belong to the “Fire” disaster type even though it was not a majority group within the holdout dataset (see Fig. 6).



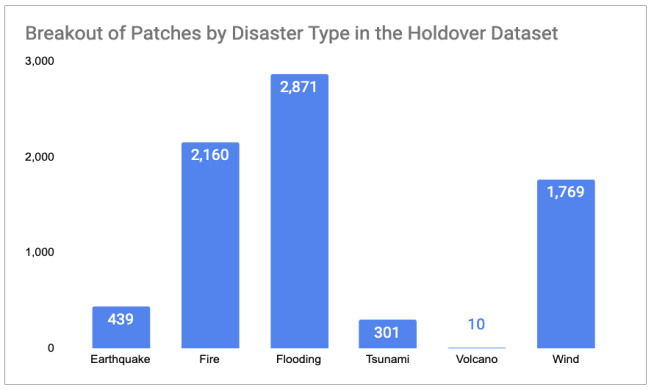


Fig.6 Breakout of classification results by the type of disaster achieved on the holdout set by MobileNetV3 (top). Total number of image patches by disaster type in the holdout dataset (bottom).

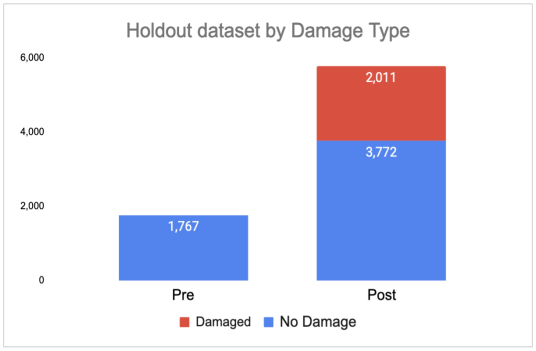


Fig.7 Breakout of images in the holdout dataset by pre- and post-event and damage type.

4. Conclusion

This paper describes a deep learning-based machine learning classifier that can be used for fast and accurate assessment of damage to the buildings based on building coordinates using pre- and post-event satellite images. The proposed system is specifically focusing on small cropped-out patches of buildings and areas that are immediately close to them. Using pre-trained MobileNetV3 network and transfer learning we managed to achieve computational efficiency by significantly reducing the size of an input image to 50x50px and achieved great performance ultimately beating the desired benchmark of 80% in F1-score.

4.1. Future work

Further work with our newly created dataset of building patches from xBD includes conditioning on disaster type, satellite metadata and other dimensions that could be mapped on to the building patches using metadata in the newly created dataset, which would allow us to analyze the performance beyond results of binary classification.

We are also currently working on developing the application to streamline the process of running inference of our trained model on the unseen images with three main objectives:

1) Simplify cropping of the target building patch to the input image specifications.
2) Run the created patch through our trained MobileNetV3 model.
3) Display damage class predicted by the model.

Additionally, plans for future work include creating a hybrid model where our trained neural network will be working in tandem with non-deep learning methods such as K-Means clustering or Random Forest.

5. References and Materials

[1] "xBD: A Dataset for Assessing Building Damage from Satellite Imagery" by Ritwik Gupta, Richard Hosfelt, Sandra Sajeev, Nirav Patel, Bryce Goodman, Jigar Doshi, Eric Heim, Howie Choset, and Matthew Gaston

[2] Bai, Y et al "Towards operational satellite-based damage- mapping using u-net convolutional network: A case study of 2011 Tohoku earthquake-tsunami." *Remote Sensing*, 10(10), 2018. https://www.mdpi.com/2072-4292/10/10/1626.

[3] Doshi, J. , Basu, S., Pang, G, "From satellite imagery to disaster insights". AI for Social Good workshop (NeurIPS 2018), 2018. https://arxiv.org/abs/1812.07033.

[4] Fujita, A., et al "Damage detection from aerial images via convolutional neural networks." In 15th IAPR International Conference on Machine Vision Applications (MVA), pp. 5–8. IEEE, 2017. https://ieeexp lore.ieee.org/abstract/document/7986759

[5] "Transfer Learning and Fine-tuning Convolutional Neural Networks" Stanford University CS231n: Deep Learning for Computer Vision

[6] Cooner, A. et al, "Detection of urban damage using remote sensing and machine learning algorithms: Revisiting the 2010 Haiti earthquake." Remote Sens. 2016, 8, 868 https://doi.org/10.3390/rs8100868

[7] Bai, Y.; Hu, J.; Su, J.; Liu, X.; Liu, H.; He, X.; Meng, S.; Mas, E.; Koshimura, S. Pyramid Pooling Module-Based Semi-Siamese Network: A Benchmark Model for Assessing Building Damage from xBD Satellite Imagery Datasets. Remote Sens. 2020, 12, 4055. https://www.mdpi.com/2072-4292/12/24/4055

[8] "Assessing out-of-domain generalization for robust building damage detection" by Vitus Benson and Alexander Ecker. Published at NeurIPS 2020 Workshop on Artificial Intelligence for Humanitarian Assistance and Disaster Response (AI+HADR 2020)