Using Deep Learning and Satellite Imagery to Assess the Damage to Civil Structures After Natural Disasters

Scott Jones and Jafar Saniie
Department of Electrical and Computer Engineering
Illinois Institute of Technology, Chicago, Illinois, USA

***Abstract*— Since 1980, millions of people have been harmed by natural disasters that have cost society over three trillion dollars. After a natural disaster has occurred, the creation of maps that identify the damage to buildings and infrastructure is imperative. Currently, many organizations perform this task manually, using pre- and post-disaster images and well-trained humans to infer the degree and extent of damage. This manual task can take days to complete. We propose to do this task automatically using post-disaster satellite imagery. We use a pre-trained neural network, SegNet, and replace its last layer with our own damage classification scheme. The final layer of the network is re-trained using cropped segments of the satellite image of the disaster. Our test results show that it is possible to create these maps quickly and efficiently.**

Keywords—Damage assessment, damage grading, deep learning, satellite imagery, neural networks

I. INTRODUCTION

Natural disasters are sudden, harmful events arising from the normal working of the earth's biosphere [1]. Typical events include hurricanes, tsunamis, fires, floods, earthquakes, droughts, and volcanic eruptions. They often result in deaths and severe destruction to both infrastructure and communities. According to EM-DAT, The International Disaster Database [2], since 1980, approximately 2.6 million people have died, 6.8 million people have been injured, 158 million people have been displaced, and 6.8 billion people have been otherwise negatively impacted by natural disasters. The total cost to property, livestock, and crops has been over 3 trillion dollars. And the frequency of natural disasters is increasing, from 460 natural disasters per decade between 1940 to 1979, to 3050 per decade from 1980 through 2017. This is an increase of about 600% over 40 years and a harbinger for a world living with climate change.

The work required to help communities recover after a natural disaster is a large and complicated effort [3]. One of the artifacts that emergency managers use to manage this complexity is a map of the area affected by a natural disaster. To this map they add the location of people, street closings (due to downed power lines, excess water, or storm debris), staging areas, and building damage. This allows them to route first responders and aid to the most heavily impacted places first. Currently this process is a manual one. It involves many organizations and people (for example Copernicus [4] in the EU, FEMA [5] in the USA, and the OpenStreetMap Foundation [6] for the world) who review before and after pictures of the disaster area and complete the damage grading maps by hand. It is a process that can take days. If the time to create these maps could be reduced, recovery efforts could provide aid more effectively and efficiently to those in need.

As a first step, we investigate the feasibility of making these maps automatically. The task we address is a very specific one: the automatic creation of damage grading maps from satellite imagery of the natural disaster using pre-trained neural networks. As the name implies, damage grading maps are aerial maps of the disaster zone's terrain over which the damage to structures has been applied. Our goal is to use a single satellite image of the damaged area and, from that image, to determine which structures have been damaged, the extent of the damage, and, if possible, to what degree each structure has been compromised. Prior to beginning this work, we need to understand current research in the field and the issues that remain to be addressed.

II. RESEARCH

A. Current Research

Work in several fields is converging to solve the problem of damage grading using neural networks from satellite imagery. Researchers in Computer Vision have been using neural networks for the past three decades to segment and classify images [7, 8]. Researchers in Remote Sensing have been using neural networks to classify groundcover from satellite imagery since the mid 1990's [9, 10]. In addition, some other fields, such as Geographical Information Systems, Wind Engineering, and Aerodynamics, [11, 12]have begun to use neural networks to classify images.

More recently, a great deal of this work has focused on deep learning, increasing the depth and breadth of the network and on ever more complex architectures [13, 14] that implement the fusion of different types of data[15, 16], different spatial resolutions of data [15, 17], and different times at which the data is gathered [11, 18, 19].

This approach faces several problems. First, it requires a large amount of training data—up to a million images for some very deep networks [14, 20, 21]. This quantity of training data is not easy to find and is even harder to systematically examine and annotate. Second, quality data in ground truth damage maps is difficult to find [11, 21, 22]. In fact, some groups estimate that their damage grading maps are at best 60% to 70% accurate [4, 19].

Third, it is unclear if there are any requirements for the number of classes to train, the labelling of the ground truth data, and the size of the patches that are used for training relative to the size of the object being classified. In our

978-1-7281-0927-5/19/$31.00 ©2019 IEEE

literature review we found no articles that discussed any of these issues, most likely because researchers have focused on the deep network and its architecture and they used whatever ground truth, image size, and classification structure was available for the task.

Fourth, if the type of damage from all natural disasters in all geographies is similar then it is plausible to train a single network. However, if the damage from dissimilar natural disasters is different (for instance, from hurricanes versus earthquakes), or if the building materials and architectures used in disparate geographies vary, then specialized networks will need to be trained.

B. Our Approach

Given the limitations in the data and the short window of time for creating the maps, we take another approach. To control for issues one and four above, we use pre-trained networks as the starting point and then apply transfer of training techniques to re-purpose them to our classification task. This approach allows us to dramatically reduce the number of images used to train the network, since we are only training the pixel classification layer. Using pre-trained networks also will allow us to quickly adopt to any type of disaster in any geography by changing the last layer of the network to evaluate the classes that are most useful and training on the data from previous disasters in the same geography.

With regard to the quality of the ground truth data used, issue two above, we use the best quality data available to us from the Copernicus EMS. This is not a terribly satisfactory answer because Copernicus estimates that their ground truth may only be 60% accurate.

The major contribution of this paper lies in the proof of concept for examining the variables posed by the third issue above. Since there was little information in the literature about the size of the patches that are used for training relative to the size of the object being classified, the labelling of the ground truth, or the number of classes on which to train, we examine one patch size (134 x 146 pixels), two sets of classifications (2-class versus 5-class), and one type of ground truth labeling (roof and building outline). We also examine whether all patches or only patches with buildings in them should be used as the training set. If this proves successful, further testing is warranted.

C. The Choice of Deep Neural Network

We reviewed three deep neural networks to see which one would be the best starting block for damage grading: U-Net [23], SegNet [24], and DeepLab [25].

U-Net [23] is a fully convolutional network that was designed to segment and classify biomedical images. It uses data augmentation to overcome the limited training data available, an overlapping tile strategy to manage the large images that are used, and a novel weighting scheme to discern the borders between cells. It uses a 572 x 572 pixel grayscale input image.

SegNet [24] was designed for general scene recognition tasks, such as those found in driving videos or indoor scene recognition. This Deep Convolutional Neural Network (DCNN) retains boundary information to discern small objects based on shape, is based on the VGG architecture, and uses median frequency balancing to handle unequal class sizes. It uses a 224 x 224 pixel RGB input image. Note that no special data augmentation measures are needed because there are ample training examples for this type of network.

DeepLab [25] was also designed for general scene recognition. It increases feature resolution by replacing max-pooling with filter up sampling, accommodates objects at different scales by using several filters with differing receptive fields, and increases localization accuracy by using fully connected Conditional Random Fields (CRF). It uses a 224 x 224 pixel RGB input image.

Although U-Net best handles the issues with limited training data and large images, the SegNet architecture potentially can find buildings based on a small building footprint because it can locate small objects by shape, has a native image input size similar to our patch size, and uses a more standard method to balance for class frequencies. Also, the issues with limited training data and large images can be handled through training rather than through network architecture. The deep neural network that best meets the needs of our task is SegNet.

III. Methods and Procedures

A. Data Preparation

We obtained data from the Copernicus EMS program which provides earth observation capabilities for monitoring the environment. We used their damage grading maps which estimate the damage to buildings and infrastructure from a disaster. Damage to buildings is graded on a four-step scale: possibly affected, moderately affected, highly affected, and destroyed. Each damage grade is assigned a color and each building is outlined in the color that corresponds to its level of damage. We analyzed information from Typhoon Haiyan [26] which made landfall over Guiuan in Eastern Samar, the Philippines, as a category 5 typhoon early on the morning of November 7, 2013 UTC and traversed the Philippines in a west northwesterly direction. Haiyan was one of the strongest typhoons to ever make landfall with winds up to 190 mph. Sea level surge was expected to be two meters or higher with the potential for widespread destruction. The damage to Tacloban city was devastating. Over 90% of the city was destroyed and the lack of electricity, clean water, and a functioning civil government led many people to evacuate.

For this study, three channel (RGB) damage grading maps from the Copernicus EMS were used as the base images. They were stored as 300 dpi PDF files. Each map was imported into Adobe Acrobat® where the base terrain and the damage overlay, that is the ground truth, were separated and stored as two files in PNG format. The terrain image contained the base RGB raster satellite imagery and the AOI boundary. The ground truth image contained the vector formatted file of the damage grading and the AOI boundary. Each file was 6834 x 6728 pixels. Both files were further edited in Adobe Illustrator®. The terrain image was cropped to remove the AOI boundary. The ground truth image file was similarly cropped and then edited to in-fill all the buildings with the colors of their respective outlines.

Each file was imported into MATLAB® Deep Learning Toolbox [27]. The terrain images did not require cleaning. The ground truth images were cleaned to correctly label the four categories of building damage (possibly affected, moderately affected, highly affected, and destroyed) and the final category of background.

As mentioned above, damage was graded on a four-step scale: possibly affected, moderately affected, highly affected, and destroyed. Each damage grade was assigned a color and

each building was outlined in the color that corresponds to its level of damage on the ground truth image. Any remaining unclassified pixels were classified as background.

Since a neural network that could analyze a native image of approximately 6500 square pixels would take a very long time to train, both the terrain and ground truth images were tiled into patches. The patch size was set at 134 x 146 pixels. This was done to reduce the size of the network while still maintaining distinguishable visual features (see Fig.1.)

(a) Terrain Patch (b) Ground Truth Patch Overlaid on the Terrain

Fig. 1. Example of (a) Terrain and (b) Ground truth images for a single tile.

B. Metrics

There are many metrics in use for analyzing data from semantic segmentation studies. All of them rely upon a confusion matrix generated from the comparison of the ground truth labels to the predicted labels for each pixel in the image or dataset. The data for the confusion matrix for an image *I* in the dataset is calculated as

$$C_{i,j} = \sum_{p \in I} |\{L_{pred}(p) = i \; and \; L_{gt}(p) = j\}| \quad (1)$$

where $L_{pred}(p)$ is the predicted label of pixel p in the image and $L_{gt}(p)$ is the ground truth label of the same pixel. The main diagonal of the confusion matrix indicates the number of correct classifications, while the off-diagonals indicate the number of mis-classifications.

Sometimes the confusion matrix is augmented with additional information such as the percentage of the values in the cells compared to the total number of pixels as well as sums of the rows and the columns, and their percentages. In this case, the sums of the rows $G_j = \sum_{i=1}^{N} C_{ij}$ for each class specify how many pixels, out of all classes, were correctly predicted, while the sums of the columns $P_i = \sum_{j=1}^{N} C_{ij}$ specify how many pixels, out of all positive classes, were correctly predicted [28]. In each of these equations, *N* is the number of classes.

Most of the literature on semantic segmentation relies on the definition of the confusion matrix from Eq. 1 [28]. We follow this calculation for the Global Accuracy, Per Class Accuracy, and Mean Per Class Accuracy and average across all images so that results can be compared across articles.

1) *Global Accuracy:* The first of the metrics, Global Accuracy, measures how many pixels were correctly classified compared to the total number of pixels. It is a simple measure of how well the classifier works on all the test images. For the simple 2-class system, the accuracy is calculated as (TP+TN)/(TP+TN+FP+FN), where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives. Extending this into the multi-class case, we have

$$GA = \frac{\sum_{i=1}^{N} C_{ii}}{\sum_{i=1}^{L} G_i} \quad (2)$$

where the numerator is the sum of the diagonal and the denominator is the total number of pixels.

2) *Per Class Accuracy and Mean Per Class Accuracy:* Per Class Accuracy measures, for each class, the number of pixels correctly identified compared to the number of pixels for the class according to the ground truth. It is a simple measure of how well each class in the classifier performs. The Per Class accuracy is defined as

$$PC_i = \frac{C_{ii}}{G_i} \; for \; i = 1 \ldots N \quad (3)$$

Mean Per Class Accuracy is another measure of how well the classifier performs. It is the simple average of the accuracy of all classes

$$PC = \frac{1}{N} \sum_{i=1}^{N} \frac{C_{ii}}{G_i} \quad (4)$$

IV. Experiments

We used SegNet [24] Network, which is included as part of MATLAB® Deep Learning Toolbox Release 2018b [27], as the basis for the neural network. The only alteration was to remove the final layer and graft our own pixel classification layer onto the network. The classification layer was connected to all the nodes in the previous layer via a softmax function with a cross-entropy loss.

Each time the network was trained, four parameters were used. The weights were optimized using stochastic gradient descent with momentum over four mini-batch sizes (4, 8, 16, and 32 patches). The momentum was set at 0.9, the initial learning rate was 0.001, and the L2 regularization was set at 0.0005. These parameters were chosen to minimize the changes to the existing weights while training the final layer. The maximum number of epochs was 100 and the mini-batch size varied as described above. Iterations per epoch was determined by the algorithm by dividing the number of training examples by the mini-batch size. The number of images in the classes was balanced using the median frequency class weights.

We looked at three experimental conditions, the first of which was data augmentation. Other studies found that data augmentation provided more robust and generalizable results and we were curious to know if it had the same effect on our data. Data augmentation was implemented through the *imagedataaugmenter* function in MATLAB®. This function was used to perform three types of augmentation: random reflection around the x-axis where the image was reflected in the left to right direction 50% of the time; random translation through the x-axis where the image was moved according to a randomly chosen integer from a uniform distribution between -10 and +10 pixels; and random translation through the y-axis where the image was moved according to a randomly chosen integer from a uniform distribution between -10 and +10 pixels.

The second experimental condition was the number of classes on which the neural network was trained. The network was trained either on all five of the original categories (possibly affected, moderately affected, highly affected, destroyed, and

background) or on only two categories (damaged and background). The damaged class was composed of all four of the damage grading classes. If the network can be trained equally well on two classes and five classes, it will be capable of estimating the extent and degree of damage. If it can only be well trained on two classes, it is only possible to estimate the extent of the damage.

The third and final experimental condition was the data on which the network was trained. The network was either trained on all the patches created from the terrain and ground truth images or on only those patches where a damaged building was found. In other words, patches that had only background in them were removed from the training set. Since there were 35 million pixels in the background class, we wanted to see if performance would improve if the ratio of damaged pixels to background pixels was enriched. We call these two groups "all patches" or "building patches."

The network was trained four times for each of the conditions and the results for Global Accuracy, Per Class Accuracy, and Mean Per Class Accuracy were averaged. Mini-batch sizes of 4, 8, 16, and 32 patches were each used once for a total of four training runs per data set.

V. Results and Discussion

For the three experimental conditions discussed, we only report the Global Accuracy score since the scores for Mean per Class Accuracy and Class Accuracy followed similar patterns. However, before examining the mathematical results, we look at the results visually. Fig. 2(a). shows the ground truth for five levels of damage, while the figure to its right shows the prediction in varying shades of green and the ground truth in varying shades of purple. The buildings with greater damage, those in red and orange, are more likely to be correctly classified than are the other buildings. Also notice that the prediction tends to incorrectly identify some of the background as damaged structure.

The results for the first experimental condition were unsurprising. Data Augmentation slightly improved in the results. The Global Accuracy for the data augmented images was 86.3% versus 85.7% for the images without augmentation. For the remainder of the analysis, all images used data augmentation.

(a) Ground Truth Patch (b) Predicted Patch

Fig. 2. Example of (a) Ground truth and (b) Predicted images for a single tile.

The second experimental condition, the number of classes used in training, provided more interesting results. In this case, performance improved when two classes were used for training rather than five classes. As Fig. 3. shows, Global Accuracy was 81.3% when 2 classes were trained and 73.0% when 5 classes were trained. Even though the number of images in the classes was balanced using the median frequency class weights, the sheer number of pixels in the background category biased the network towards correctly classifying everything as background.

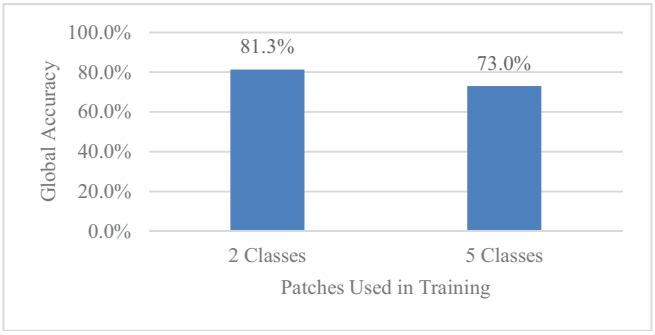

Fig. 3. Global accuracy for number of classes.

The third experimental condition, the type of patches used for training, was also informative. In this case, performance improved when patches with only background information were used for training. As Fig. 4. shows, Global Accuracy was 86.4% when all patches were used and 67.9% when only patches with buildings were used. When we tried to equalize this by eliminating the patches with only background data, performance decreased. However, this decrease could have been caused by the reduced number of patches that were used to train the network. In either case, the results indicates that the network is more effectively trained on all patches.

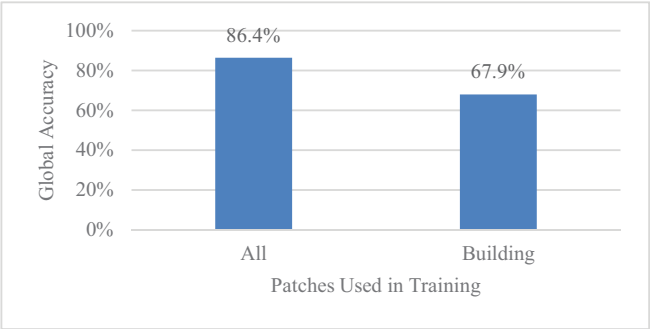

Fig. 4. Global accuracy for patches.

VI. Conclusion

Our test results have shown that it is possible to create maps that show the extent of the damage after a natural disaster using post-disaster satellite imagery. In addition, the time taken to perform this analysis, after the satellite image is provided, is less than one hour. This proves that the damage map creation procedure is an operationally useful technique for helping emergency managers maintain situational awareness and direct resources to the most critical areas during an emergency.

To increase the accuracy of the prediction, we will focus our future efforts on comparing the different types of ground truth labeling (building sides, building roofs etc.); the types of deep networks used for training, such as U-Net [23] or DeepLab [25]; and the contribution of different resolutions of satellite imagery. In addition, we plan to examine the similarity of damage from different types of natural disasters in different regions. If the type of damage from all natural disasters in all geographies is similar, then it is plausible to train a single

network. However, if the damage from dissimilar natural disasters is different (for instance, from hurricanes versus earthquakes), or if the building materials and architectures used in disparate geographies vary, then specialized networks will need to be trained.

REFERENCES

[1] Merriam-Webster. *natural disaster*. Available: https://www.merriam-webster.com/dictionary/natural%20disaster

[2] H. Ritchie and M. Roser. (2019). *Natural Disasters*. Available: https://ourworldindata.org/natural-disasters

[3] FEMA, "Incident Management Handbook," F. E. M. Agency, Ed., ed, 2017.

[4] *Copernicus Emergency Management Service*. Available: https://emergency.copernicus.eu/

[5] A. United States. Federal Emergency Management, *FEMA disaster program information*: Revised Feb. 2008. Washington, DC : FEMA, 2008., 2019.

[6] O. F. contributors. *Main Page*. Available: http://wiki.osmfoundation.org/w/index.php?title=Main_Page&oldid=6067

[7] X. Wang, H. Ma, X. Chen, and S. You, "Edge Preserving and Multi-Scale Contextual Neural Network for Salient Object Detection," *IEEE Transactions on Image Processing,* vol. 27, pp. 121-134, Jan 2018.

[8] H. M. Bui, M. Lech, E. Cheng, K. Neville, and I. S. Burnett, "Object Recognition Using Deep Convolutional Features Transformed by a Recursive Network Structure," *IEEE Access,* vol. 4, pp. 10059-10066, 2016.

[9] M. F. Augusteijn, L. E. Clemens, and K. A. Shaw, "Performance Evaluation of Texture Measures for Ground Cover Identification in Satellite Images by Means of a Neural-Network Classifier," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 33, pp. 616-626, May 1995.

[10] P. B. Zhang, Y. H. Ke, Z. X. Zhang, M. L. Wang, P. Li, and S. Y. Zhang, "Urban Land Use and Land Cover Classification Using Novel Deep Learning Models Based on High Spatial Resolution Satellite Imagery," *Sensors,* vol. 18, p. 21, Nov 2018.

[11] A. Piscini, V. Romaniello, C. Bignami, and S. Stramondo, "A New Damage Assessment Method by Means of Neural Network and Multi-Sensor Satellite Data," *Applied Sciences-Basel,* vol. 7, p. 10, Aug 2017.

[12] S. Radhika, Y. Tamura, and M. Matsui, "Cyclone damage detection on building structures from pre- and post-satellite images using wavelet based pattern recognition," *Journal of Wind Engineering and Industrial Aerodynamics,* vol. 136, pp. 23-33, Jan 2015.

[13] K. R. Nia and G. Mori, "Building Damage Assessment Using Deep Learning and Ground-Level Image Data," in *2017 14th Conference on Computer and Robot Vision (CRV)*, 2017, pp. 95-102.

[14] Q. Dung Cao and Y. Choe, "Detecting Damaged Buildings on Post-Hurricane Satellite Imagery Based on Customized Convolutional Neural Networks," *CoRR,* vol. abs/1807.01688, 2018.

[15] D. Duarte, F. Nex, N. Kerle, and G. Vosselman, "Multi-Resolution Feature Fusion for Image Classification of Building Damages with Convolutional Neural Networks," *Remote Sensing,* vol. 10, p. 26, Oct 2018.

[16] L. Gomez-Chova, D. Tuia, G. Moser, and G. Camps-Valls, "Multimodal Classification of Remote Sensing Images: A Review and Future Directions," *Proceedings of the IEEE,* vol. 103, pp. 1560-1584, Sep 2015.

[17] T. Hermosilla, L. A. Ruiz, J. A. Recio, and J. Estornell, "Evaluation of Automatic Building Detection Approaches Combining High Resolution Images and LiDAR Data," *Remote Sensing,* vol. 3, pp. 1188-1210, Jun 2011.

[18] S. W. Myint, M. Yuan, R. S. Cerveny, and C. P. Giri, "Comparison of Remote Sensing Image Processing Techniques to Identify Tornado Damage Areas from Landsat TM Data," *Sensors,* vol. 8, p. 1128, 2008.

[19] J. Thomas, A. Kareem, and K. W. Bowyer, "Automated Poststorm Damage Classification of Low-Rise Building Roofing Systems Using High-Resolution Aerial Imagery," *Ieee Transactions on Geoscience and Remote Sensing,* vol. 52, pp. 3851-3861, Jul 2014.

[20] Y. B. Bai, E. Mas, and S. Koshimura, "Towards Operational Satellite-Based Damage-Mapping Using U-Net Convolutional Network: A Case Study of 2011 Tohoku Earthquake-Tsunami," *Remote Sensing,* vol. 10, p. 17, Oct 2018.

[21] A. Vetrivel, N. Kerle, M. Gerke, F. C. Nex, and G. Vosselman, "Towards automated satellite image segmentation and classification for assessing disaster damage using data-specific features with incremental learning," in *Proceedings of GEOBIA 2016 : Solutions and synergies, 14-16 September 2016, Enschede, Netherlands*, N. Kerle, M. Gerke, and S. Lefevre, Eds., ed. Enschede: University of Twente, Faculty of Geo-Information Science and Earth Observation (ITC), 2016/9/14, pp. 1-5.

[22] N. Kerle and R. R. Hoffman, "Collaborative damage mapping for emergency response: the role of Cognitive Systems Engineering," *Natural Hazards and Earth System Sciences,* vol. 13, pp. 97-113, 2013.

[23] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *CoRR,* vol. abs/1505.04597, 2015.

[24] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 39, pp. 2481-2495, 2017.

[25] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 40, pp. 834-848, Apr 2018.

[26] C. E. M. Service. *[EMSR058] Tacloban City: Detailed Grading Map 2*. Available: https://emergency.copernicus.eu/mapping/system/files/components/EMSR058_02TACLOBANCITY_GRADING_DETAIL02_v2_300dpi.pdf

[27] Mathworks. (2019). *Deep Learning Toolbox*. Available: https://www.mathworks.com/products/deep-learning.html

[28] G. Csurka, D. Larlus, and F. Perronnin, "What is a good evaluation measure for semantic segmentation?," in *BMVC*, 2013.