Assessing Scale of Building Damage Post Disasters

Abhirag Nagpure   Pranav Gujarathi   Shubham Gaikwad
Luddy School of Informatics, Computing and Engineering
Indiana University
{anagpure, pgujarat, shgaikwa}@iu.edu

Abstract

Object detection and recognition from satellite imagery has made significant progress, owing its success to deep learning methodologies, mainly convolutional neural networks (CNNs). Change detection has also shown considerable progress, mainly due to algorithms like optical flow and Siamese networks. We focus on the xBD dataset, introduced along-with the xView2 challenge, to detect the scale of damage done to buildings using pre and post disaster image pairs. We experiment with multiple architectures for semantic segmentation and multi-class classification, and also introduce a novel modified architecture of the existing Faster RCNN object detection model.

1. Introduction

To facilitate faster human assistance to affected geographical areas and its population after a disaster, identifying affected areas is absolutely urgent so as to initiate the deployment of response activities. Remote sensing and satellite imagery has long been used for this purpose due to their accessibility and wide coverage of areas. Manual analysis is possible, but is highly prone to erroneous and subjective methodologies and calculations, and at minimum, is a tedious and time-consuming process. Automating this analysis pipeline with accurate reports would greatly improve assistance to affected areas.

Change detection is a relatively open field, with most advances being in specific domains like tracking an object by following its position. Image similarity, a very similar problem has also been tackled using Siamese Networks and one-shot learning. Unfortunately, none of these approaches give a quantitative measure of the scale of change in images.

The xView2 challenge[1] introduces the xBD dataset [7], which consists of pre and post disaster image pairs, spread across 7 types of disasters : tsunamis, volcanoes, hurricanes, tornadoes, earthquakes, floods and bushfires/wildfires. The task of the challenge is split into two parts:

1. Localize buildings
2. Classify Building Damage: Assign a class of damage from :
   (a) undamaged building
   (b) building with minor damage
   (c) building with major damage
   (d) destroyed building

Our project seeks to experiment upon modified existing Siamese networks for image similarity and tackle the xView2 challenge. We also propose a modified Faster-RCNN model with a 6 channel input, over the traditional 3/4 channels that usual CNNs work upon.

2. Related Work

Convolutional neural networks (CNNs) were the first breakthrough in the computer vision domain for automated image analysis with the introduction of the backpropogation algorithm [16]. The popularity of CNN-based architectures sky-rocketed with the results produced by AlexNet on the ILSVRC challenge. [10] [6]

Originally, Siamese networks were introduced as an approach to tackle the problem of signature verification [3]. The basic idea behind this network is the sharing of weights by two different sub-networks so as to calculate the similarity between two comparable inputs using a novel contrastive loss function that penalizes inputs from different classes and vice versa. Following their success, they were applied to a wide variety of problems ranging from face detection to speech emotion recognition. [11] [4]

Image matching/similarity has been an interesting topic in the deep learning and computer vision research community. [2] proposes a multi-scale deep Siamese network which creates an embedding from both upper and lower level CNN layers outputs. It also uses a pair mining strategy

[1] The xView2 challenge is hosted at https://xview2.org/

Figure 1. A pair of pre and post disaster images

which is inspired from curriculum learning which claims to produce better results.

Another relevant technical contribution [5] applies traditional machine learning techniques, namely random forests and multi-layer perceptron classifier trained on image representations derived from various structural and textural features.

Class Label	#Samples
un-classified	845
no-damage	17232
minor-damage	144
major-damage	231
destroyed	927

Table 1. Patch Class Distribution in Training Data

3. Dataset

The xBD dataset spans over 19 disasters, each belonging to one of the following category: tsunamis, volcanoes, hurricanes,tornadoes, earthquakes, floods and bushfires/wildfires. Pre and post disaster image pairs are provided, as are their corresponding data files in the JSON format. Each image is 1024*1024 RGB image in the PNG format. A JSON file for the pre-disaster image contains one or more polygon annotations for buildings and a corresponding unique identifier key for each polygon. A JSON file for the post-disaster image includes all of the previously stated fields and a class label that describes the nature of damage on a scale of 1-4: 'no-damage', 'minor-damage', 'major-damage' and 'destroyed'.

4. Experiments

4.1. Setup

Due to constraints imposed by all of time, size of data, and limited computing resources, we restrict the scope of our experiments to the Portugal Wildfire disaster only. It consists of 1869 image pairs, which we randomly split into 1500 for training and 369 for testing. Within 1500 images, the distribution of classes of patches is shown in Table 1.

We drop all 'un-classified' samples from both training and testing data, since they will not contribute towards classification.

All experiments are performed on a personal workstation equipped with a Intel® Xeon(R) E-2276M CPU, a 16 GB Nvidia® Quadro RTX 5000 GPU and 128 GB RAM.

4.2. Baseline Model

The baseline model[2] released as a part of the xView2 challenge consists of two models: SpaceNet Building Detection[3] (pre-trained on the SpaceNet dataset[1]) for detecting buildings and a custom ResNet[8] based CNN architecture for classifying damage.

[2] https://github.com/DIUx-xView/xview2-baseline
[3] https://github.com/motokimura/spacenet_building_detection

Figure 2. Localization Output after training for 10 epochs

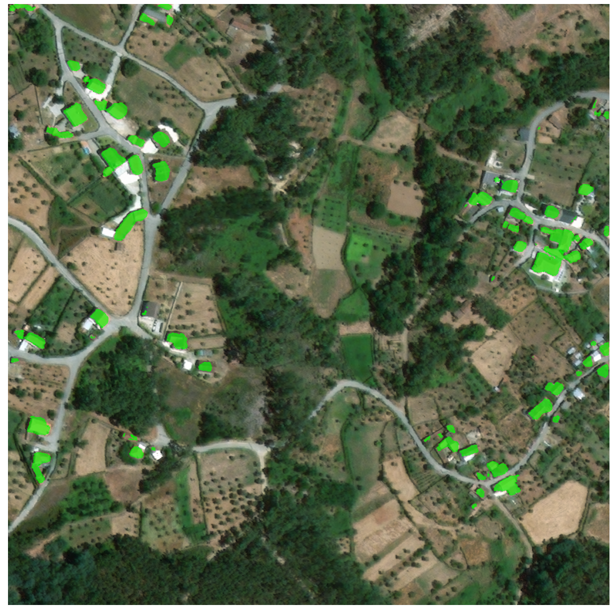

Figure 3. Localization Output after training for 20 epochs

4.3. Our contribution

4.3.1 Semantic Segmentation

We choose the SpaceNet Building Detection model used in the baseline since it is already pre-trained to recognise buildings from satellite imagery in the SpaceNet dataset and hence, the transfer learning methodology can be used here. The intuition behind this is a segmentation model trained for detecting buildings will have learnt features more relevant to the problem here, than a model that is trained for more generic features based on the ImageNet or MS-COCO dataset.[6] [12]

4.3.2 Patch Classification

1. Learning Classifiers on features extracted from CNNs:

   (a) Using Convolutional features from VGG-19: We pass each pair of pre and post patches through a VGG-19 model[15] pre-trained on the ImageNet dataset, and extract features from the third convolutional block. We calculate Manhattan distance between these two feature vectors[13] and train a Random Forest classification model by undersampling the majority class so that all classes are almost equally represented. Results on testing images are shown in Table 2.

   (b) Using Bottleneck Features: We train a multilayer perceptron classifier on the Euclidean and Manhattan distances between the feature vectors extracted for both the pre and the post imae patch from the ResNet[8] architecture but results are worse than the previous approach.

Class	F-1 Score	#Samples
no-damage	0.51	5017
minor-damage	0	54
major-damage	0	68
destroyed	0.16	267

Table 2. Performance of Random Forest on Cosine Similarity of Bottleneck Features

Class	F-1 Score	Precision	Recall
no-damage	0.67	0.99	0.50
minor-damage	0.02	0.01	0.25
major-damage	0.04	0.02	0.28
destroyed	0.32	0.21	0.66

Table 3. Performance of our custom Siamese network

2. Siamese Network for Image Similarity:

   We develop a custom Siamese network architecture, using a 3-layer CNN and a dense layers. For every pair of pre and post image patches which are classified as 'no-damage', we assign a score of 0, and 1 if for the rest. We use the binary cross-entropy loss function and treat the network as a classifier. During prediction, we take the image similarity score and transform it using suitable ranges into our target classes. Results obtained are shown in Table 3.

3. 6 Channel Faster-RCNN:

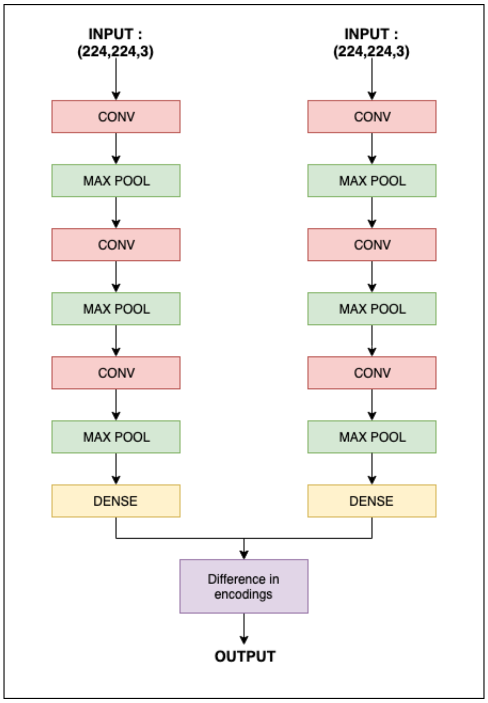

Figure 4. Our Siamese Network Architecture

The original Faster-RCNN model[14] uses a Region Proposal Network (RPN) that shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals. An RPN is a fully convolutional network that simultaneously predicts object bounds and object-ness scores at each position. The RPN is trained end-to-end to generate high-quality regions of interest (ROI) proposals, which are used by Faster R-CNN for detection. We further merge RPN and Faster-RCNN into a single network by sharing their convolutional features—using the recently popular terminology of neural networks with 'attention' mechanisms, the RPN component tells the unified network where to look.

Our goal of applying Faster RCNN is not exactly to boost performance on localization or classification selectively, but unlike other approaches, develop an end to end pipeline for both the tasks. Hence due to high object size variance and imbalance in the data we will not be looking at the classification/localization scores as we did with the earlier approaches but rather the four losses in the Faster-RCNN structure, namely the ROI classification loss, ROI Regression loss, RPN Classification loss and finally the RPN Regression loss.

Since predicting the scale of damage is conveniently based on the pre and post pairs, we want to not only localize region which has buildings, but also predict relative damage. We solve this by concatenating the pre disaster and post disaster image in the dimension of number of channels, hence we have 6 dimensional image for the object detection. For the classification part of the exercise we utilize the MobileNet [9] architecture which is trained from scratch. Unfortunately, for some strange reason, we keep running out of CUDA memory but all the four losses steadily decrease.

5. Conclusions and Future Work

The problem we chose has a huge dataset and requires a lot of computing power and time for each experiment. We tried to implement few techniques on a subset of the data. The experiments of calculating the normalized dot products on the embeddings of pre and post disaster image pairs did not give good results. The classification results improved when we tried a custom Siamese network on the part of the data. This network can further be improved with some hyper-parameter tuning and adding more data. One of the bottlenecks we faced was class imbalance in 'no-damage' and other classes. The 'no-damage' data points account for 20 times more than the 'minor-damage' and 'major-damage' classes.

We tried to implement the Faster-RCNN model to perform localization and classification in single shot. We were successful to implement at a very basic level. But again due to lack of resources and time were not able to bring to fruitful conclusion. We plan to work with this approach further as we think this novel approach can give some better results. We also plan to train a U-Net based architecture to perform localization and classification in a single shot.

Acknowledgements

We thank Prof. Kim for his ideas and constant words of encouragement.

References

[1] Spacenet on amazon web services (aws). "datasets." the spacenet catalog. last modified april 30, 2018.

[2] Srikar Appalaraju and Vineet Chaoji. Image similarity using deep cnn and curriculum learning, 2017.

[3] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a "siamese" time delay neural network. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems 6*, pages 737–744. Morgan-Kaufmann, 1994.

[4] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546 vol. 1, June 2005.

[5] Yang; Campbell James B. Cooner, Austin J.; Shao. Detection of urban damage using remote sensing and machine learning algorithms: Revisiting the 2010 haiti earthquake., 2016.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[7] Ritwik Gupta, Bryce Goodman, Nirav Patel, Ricky Hosfelt, Sandra Sajeev, Eric Heim, Jigar Doshi, Keane Lucas, Howie Choset, and Matthew Gaston. Creating xbd: A dataset for assessing building damage from satellite imagery. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[9] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, May 2017.

[11] Zheng Lian, Ya Li, Jianhua Tao, and Jian Huang. Speech emotion recognition via contrastive loss under siamese networks. *Proceedings of the Joint Workshop of the 4th Workshop on Affective Social Multimedia Computing and first Multi-Modal Affective Computing of Large-Scale Multimedia Data - ASMMC-MMAC'18*, 2018.

[12] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014.

[13] M. D. Malkauthekar. Analysis of euclidean distance and manhattan distance measure in face recognition. In *Third International Conference on Computational Intelligence and Information Technology (CIIT 2013)*, pages 503–507, Oct 2013.

[14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2015.

[15] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[16] J. S. Denker D. Henderson R. E. Howard W. Hubbard L. D. Jackel Y. LeCun, B. Boser. Backpropagation applied to handwritten zip code recognition.