# Destroyed Buildings Detection during the Syrian War using Satellite Imagery and CNN

**Projet - Informatique ENSAE 2020**

César Roaldès, Morgane Hoffmann & Paul Graffan

April 2020

## 1 Problem Framing

The Syrian conflict has been the biggest humanitarian crisis in recent time taking the live of millions of people and destroying urban centers and infrastructures in the country. Due to restricted access, security issue as well as dysfunction of main institutions the scale of the destruction remains to be estimated accurately. This work aims at giving a clue on the later. Building on satellite imagery and the huge work of the UNITAR-UNOSAT[1] we construct a classifier that recognizes destructed infrastructures and buildings in main Syrian urban centers. Based on this algorithm we then propose a mapping of destruction for several cities in Syria were information is still very scarce. Further development are finally proposed.

## 2 Protocol

In recent years, open access to satellite imagery has been developed allowing for interesting applications in computer vision. However, performance of computer vision tasks such as detection or classification rely on large labeled training data which are often not available at low costs. We overcome this challenge by relying on multi-step transfer learning approach. Our transfer learnig pipeline involve three steps. First we start with a convolutional neural network (CNN) model ResNet [2] that has been pretrained on ImageNet [1], a large labeled classification dataset. This pre-work allows our model to learn low-level image features that are common to many vision tasks. Because satellite imagery have particular features, we then train our model again on a labeled satellite imagery dataset EuroSAT [3; 4]. Finally we build on the knowledge gained from this classification task and fine-tune the CNN for our specific task : predict the intensity of damage of Syrian urban centers at high level geographic aggregation.

For this very last step, daytime satellite imagery have been collected from the Google Static Maps API which makes available images with relatively high resolution (0.3 meters per pixel approximately) while each image size is 600x600. Labels were collected from UNITAR-UNOSTAT damage assessment of Syrian urban centers. In this dataset each damaged building is referenced with its latitude and longitude. Transfer from the latter to labeled satellite imagery involves determining if the image contains or not a damaged building. We therefore end up with a binary classification : damaged if the image contains a damaged building and not damaged otherwise. Our collected images contain really few buildings and we believe that damages are more likely to be concentrated. Therefore, if this labelling is far from being optimal it allows us nevertheless to spot damaged locations.

In order to illustrate the labelling process, Figure 1 shows an example of a damaged labelled image (right) and a non-damaged labelled image (left).

## 3 Results

As presented in the previous section, three steps were undertaken. Models were trained using 70% of our sample and evaluated on the remaining 30%. As a benchmark model for performance comparison we use a ResNet50 CNN pretrained on ImageNet but not retrained on EuroSAT. Both models are fitted during 50 epochs. We use the cross entropy loss and Adam [5] optimizer to optimize the CNN parameters with a starting learning rate of 0.001 decreased by a factor 10 every 15 epochs.

[1] UNITAR-UNOSAT is the United Nations satellite imagery programme


Non-damaged satellite imagery Ez-Zhor, Syria

Damaged satellite imagery Ez-Zhor, Syria

**Figure 1:** We clearly see a big difference between destroyed and non-destroyed locations in the city Ez-Zhor, Syria

## 3.1 Achieved Performance

The evolution of the performance for both models during the training phase are presented below in Figure 2. Both models reach convergence at roughly the same speed and we can't assess that the second step of training involving EuroSAT has been significantly improving results.

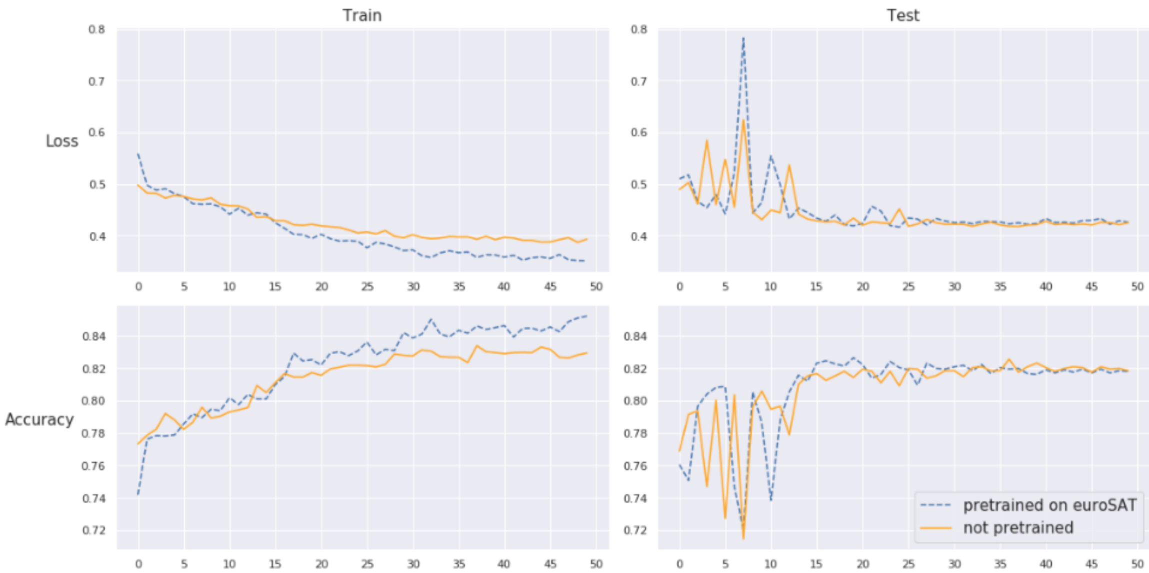



**Figure 2:** Performance evolution during training phase

Next we compare both models on the validation set composed of 2164 satellite images, with 62% images labeled as 0 (no damaged buildings in the image) and 38% labeled as 1 (presence of damaged buildings) in accordance to the training set. We use for performance metrics the accuracy of predictions and the AUC as presented in Table 1.

| Model | accuracy | AUC |
|---|---|---|
| not pretrained | 0.825 | 0.802 |
| pretrained EuroSAT | 0.826 | 0.802 |

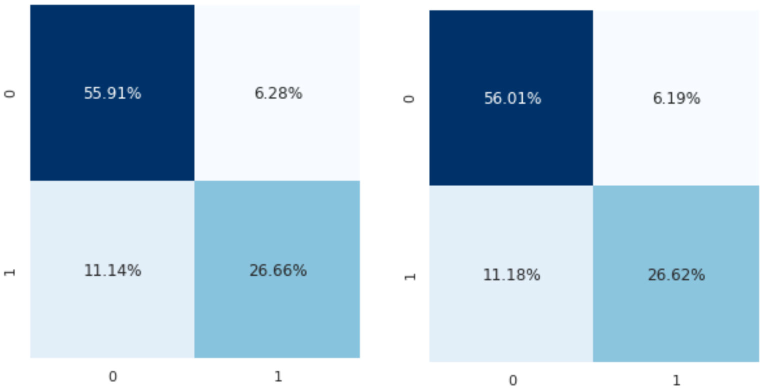
**Table 1:** Models accuracy and AUC on validation set

The pretrained model on EuroSAT did not outperform our benchmark. However, both CNN achieve strong results with an accuracy of 82% on validation set. To refine our analysis, we display in Table 2 the precision, recall and f1-score of both models.

| Model | precision | recall | f1-score |
|---|---|---|---|
| not pretrained | 0.81 | 0.71 | 0.75 |
| pretrained EuroSAT | 0.81 | 0.70 | 0.75 |

**Table 2:** Models precision, recall and f1-score on validation set

Considering recall as our measure of performance, one can observe that the not pretrained model has slightly outperformed the EuroSAT pretrained model. Furthermore, confusion matrices in Figure 3 show that the actual difference in prediction accuracy for both model is insignificant. We cannot conclude that one model better detects destroyed buildings in an image than the other. The transfer learning pipeline did not improve the quality of predictions of the original ResNet50. Section 3.3 attempts to analyse the decision process of our models and section 4 proposes potential improvements.




not pretrained pretrained EuroSAT

**Figure 3:** Confusion matrices on validation set

## 3.2 Mapping Damaged Urban Cities

As use case, we run the model on Aleppo's satellite images on which we display models' predictions with identified destroyed buildings location given by UNITAR-UNOSAT in Figure 4.

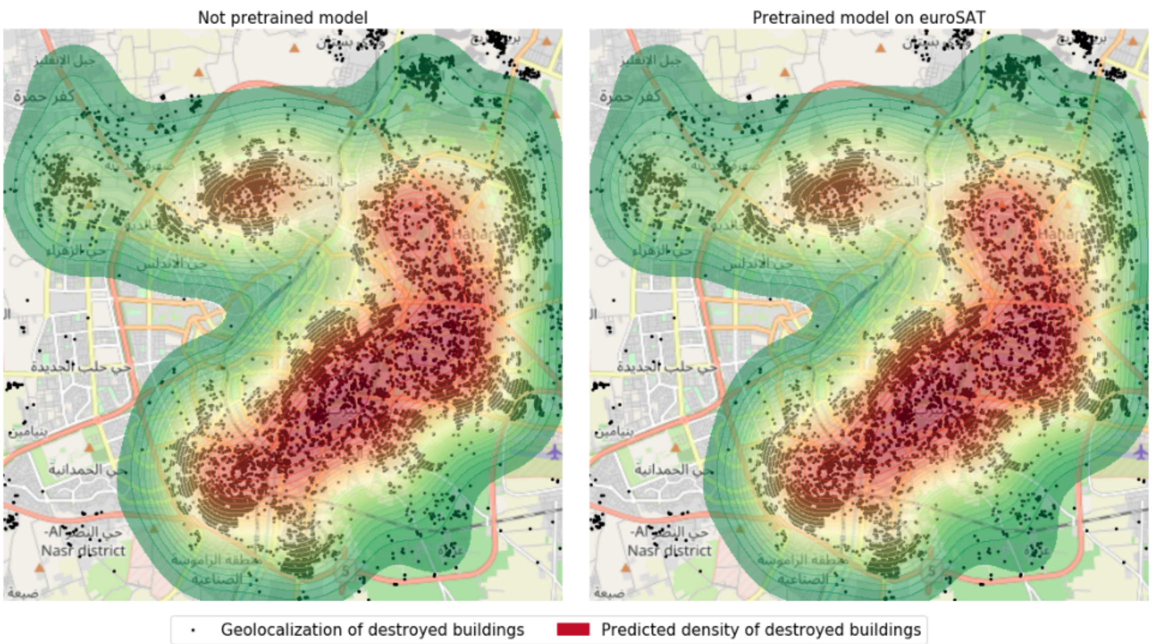
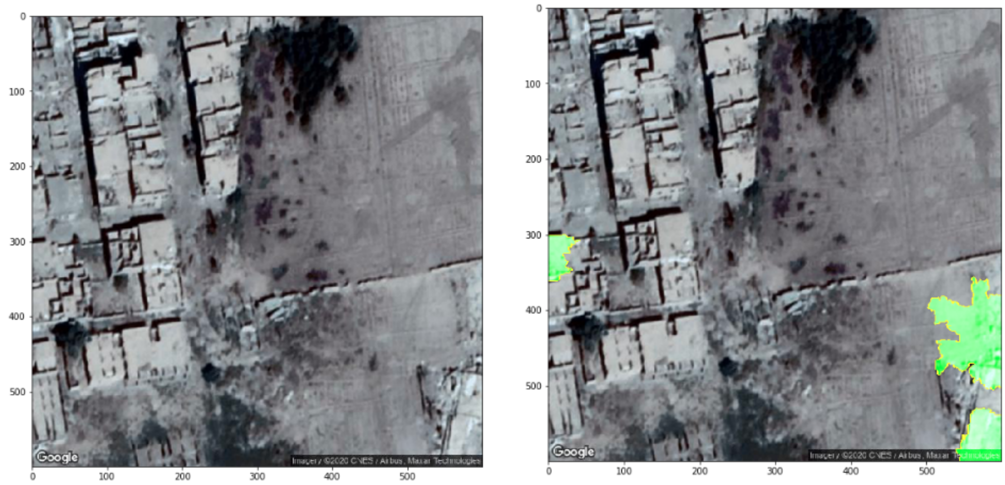



**Figure 4:** Models predictions on Aleppo

### 3.3 Understanding the Black Box : LIME

LIME (standing for *Local Interpretable Model-agnostic Explanations*) [7] is an explanation technique that explains the predictions of any classifier. When it comes to image classification, LIME grays out super-pixels to detect parts of the image having the greatest impact on its classification by the model. By making these explanations understandable, LIME allows us to compare the prior beliefs we have on the most important features a "good" model should base its decisions on with the features it actually uses. For our problem, we expect our model to target parts of the image containing destroyed buildings characteristics when it labels 1.

Figure 6 and 5 show two images containing destroyed building, and the LIME analyses of the most influential super-pixels (in green) in the models decision-making processes. Note that for both images, our models correctly predict the images labels. Unfortunately, the low resolution of the images makes it quite difficult, even to a human observer, to identify destroyed buildings, making it difficult to interpret the results. However, the super-pixels highlighted don't seem to specifically target destroyed buildings characteristics and lead us to doubt the generalizability of our models.



Original image LIME analysis

**Figure 5:** Not pretrained model analysis

Original image LIME analysis

**Figure 6:** Pretrained EuroSAT model analysis

# 4 Further Developments and Discussion

First concerns relate to the methodology we choose in order to label satellite images which likely creates noise during the training process. UNITAR-UNOSAT damage assessment database gathers information on buildings damage. In particular, it registers, for each building, its damage intensity as well as its exact geographic position (latitude and longitude). We used this information not to label one building but an entire image potentially containing several other non-damaged buildings making the training process much harder. We were aware of this issue before running the model and we were actually surprised by its good results despite this obvious issue. To overcome this challenge we propose the following methodology : First, one should use a building detection ML model to identify all buildings in the area of interest. Model detection using Faster-RCNN [6] architectures are good candidates for this task. Second, one could crop images around each building and label all buildings using the UNITAR-UNOSAT damage assessment database. Unfortunately, we were not able to investigate this proposed framework in this project.

Another caveat relates to the gap between the year damages were assessed and the year the satellite imagery was taken. Given the limited availability of high-resolution time series of daytime imagery (at low cost) we had to rely on Google Map API which gave us small image samples at one point in time (February, 2020). However, the last damage assessment from UNOSAT was undertaken at the end of 2017. If we can believe that the majority of destroyed buildings were not rebuilt in such a short period of time we think nevertheless that using images from 2017 would yield better results. Finally, because of satellite imagery access limitations we were not able to train our model on a large amount of training data. This step will be crucial to enable further generalization.

We believe that remote sensing combined with ML is an extremely powerful tool for humanitarian crisis management from armed conflict to natural disasters. Automating the process of damage structures registration not only gives a clue on the amount of material loss but also represents a good proxy to localize affected and in-need populations. Developing this framework appears to us extremely important and work has still to be done in this field.

# Bibliography


[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *arXiv e-prints*, page arXiv:1512.03385, December 2015.

[3] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Introducing eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 204–207. IEEE, 2018.

[4] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2019.

[5] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, page arXiv:1412.6980, December 2014.

[6] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *arXiv e-prints*, page arXiv:1506.01497, June 2015.

[7] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “Why Should I Trust You?”: Explaining the Predictions of Any Classifier. *arXiv e-prints*, page arXiv:1602.04938, February 2016.