Man-made Structures Detection from Space

Authors: **Eduard Ribas Fernández**, **Peter Weber**

Advisor: **Jordi Vitrià** (Universidad de Barcelona), **Marco Bressan** (Satellogic)

Fundamentals of Data Science Master's Thesis - Universitat de Barcelona - July 2019

All the files used in this project, including the image datasets build and code generated, are available online:

- The images datasets are published in Google Drive (https://drive.google.com/open?id=1Hjod1ZTuSlW3VNO2IuGoq_iagl3imnJQ%7D)
- All the code produced and used in these analysis is available in a GitHub repository (https://github.com/peterweber85/MasterThesis). It includes Python libraries generated, scripts and Jupyter Notebooks.

1. Introduction

Human kind exerts an ever increasing pressure on natural and ecological systems due to the associated consequences of the explosion of human population. The exploitation of the earth manifests itself in extraction of natural ressources, proliferation of human-made infrastructure and waste, and increasing production land use for crop and pastry land.

An essential prerequisite to mitigate human threat to nature is the access to data that allows for spatial and temporal mapping of human activity. To this end, the last decades have brought about developments of affordable and recurrent remote sensing technology. In particular, we now have public and continuous access to overhead imagery data (from satellites or from airborne sensor systems) for earth observation in different levels of detail, ranging from 100m to 0.01m. Additionally, remote sensing technologies open up the road for applications in agricultue, disaster recovery, urban development, and environmental mapping.

On the other hand, the computer vision community has largely benefited from recent advances in deep learning ultimately leading to the outsourcing of convolutional neural networks pretrained on massive datasets. In the remote sensing community, researchers are recently also starting to follow this pathway. However, pretrained models are not yet widely available, so that many works in the remote sensing field are based on fine-tuning neural networks pre-trained on traditional computer vision tasks.

Together with Satellogic (a company that provides earth observation data and analytics as a service to enable better decision making for industries, governments, and individuals), we aim this thesis to provide an answer to the question:

> **What is the optimal resolution to detect human impact in satellite imagery, having in mind the economical cost of acquiring and processing the information?**

Determining this value is important not only for designing optimal satellite sensors but also to use optimal data sources when developing data-based remote sensing products. The goal here is not to build the top performance, state-of-the-art model to detect all sorts of human impact in satellite images, but rather to analyze the feasibility and cost of doing so at different resolutions. Of course, better algorithms could be trained on larger datasets to accurately identify certain types of human impact, but we consider a more general problem.

In this work we cover the process of building the datasets needed for our investigation, some Deep Learning techniques to model the problem at different resolutions, and an extensive discussion of the results obtained and costs associated to the entire pipeline.

2. Building datasets

In order to tackle our problem, we need a good dataset from which we can build reliable models. The goal is to detect human impact on aerial images and determine the dependency on resolution per pixel of a chosen evaluation metric. Ideally, the range for the resolutions should scale from a few tens of centimeters to a few tens of meters, whereas the images with low resolution can be generated from the high resolution images by downsampling. Therefore, the dataset has to fulfill the following conditions:

- Provide imagery data with labels that can be used to clearly distinguish between existing and non-existing human impact, respectively. This impact might be classified pixel wise, or as binary classification for the entire image, or as multi-class classification that can be translated into binary labelling.
- Balanced dataset of approximately the same number of images for both classes, and a large variety of different terrains within each class.
- Third, the images need to have a resolution per pixel which is equal or better than 1m. Also, the height and width of the images should measure at least 500×500 pixels, so that one has enough room for downsampling.

There exist some relevant, publicly available remote sensing datasets with ground truth labels, ranging from pure low-resolution satellite imagery (Sentinel-2) to high-resolution images taken with an aircraft (USGS) to a mix of different image sources (Google Earth). The table below summarizes some of these datasets:

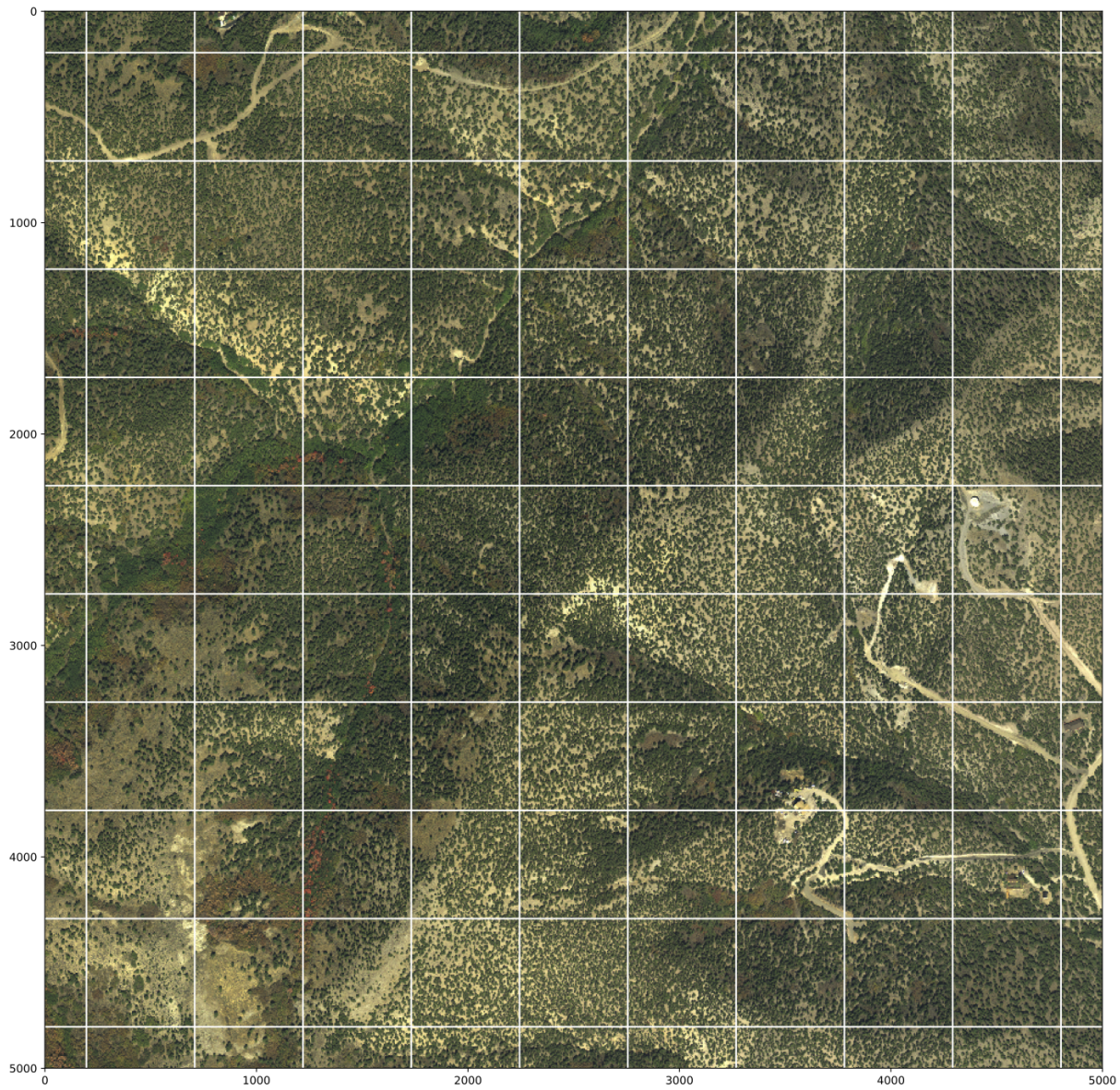
name	source	images	resolution (m)	size (pixel)	categories
BigEarthNet	Sentinel-2	590,326	10, 20, 60	120, 60, 20	~ 50
EuroSAT	Sentinel-2	27,000	10	64	10
UCMerced	USGS	2100	0.3	256	21
DeepSat	USGS	405,000	1	28	6
AID	Google Earth	10,000	0.5 - 8	600	30
PatternNet	Google Earth	30,400	0.06 - 4.69	256	38

Overall, the main issue with these datasets stems from the fact that non of them was collected with the purpose to analyze the human footprint. Therefore they are very unbalanced, and do not contain sufficient variety of images for the classes without human influence. We hence decided to collect and label images by ourselves, finally opting for the USGS aerial imagery collection (Earthexplorer USGS (https://earthexplorer.usgs.gov/)).

USGS land cover

With this datasource and the help of the USGS Land Cover Viewer (https://gis1.usgs.gov/csas/gap/viewer/land_cover/Map.aspx) we were able to construct a balanced and representative dataset. For the determination of relevant geographic locations we excluded cities and highly developed urban areas, and instead focussed on unpopulated areas. Specifically, we limited our image search to the four land use categories **agriculture**, **shrubland-grassland**, **semi-desert**, and **forest-woodland** that can be found in the USGS Land Cover Viewer. Note that these categories served as a rough geographic orientation to pin down geolocations of interest.

With this given locations, we constructed two datasets with 0.3m and 1m resolution (from the High Resolution Orthoimagery datasource and the National Agriculture Imagery Program (NAIP) respectively). The images download were large images of thousands of pixels, as the one below, so we implemented a pipeline to process them:

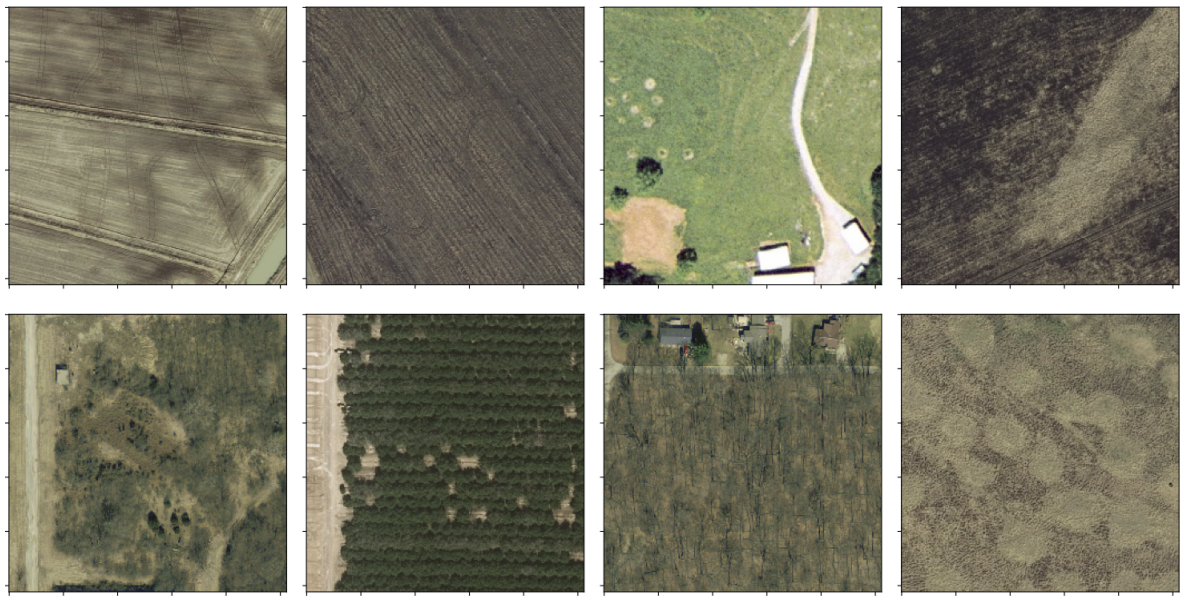


Data processing and labeling

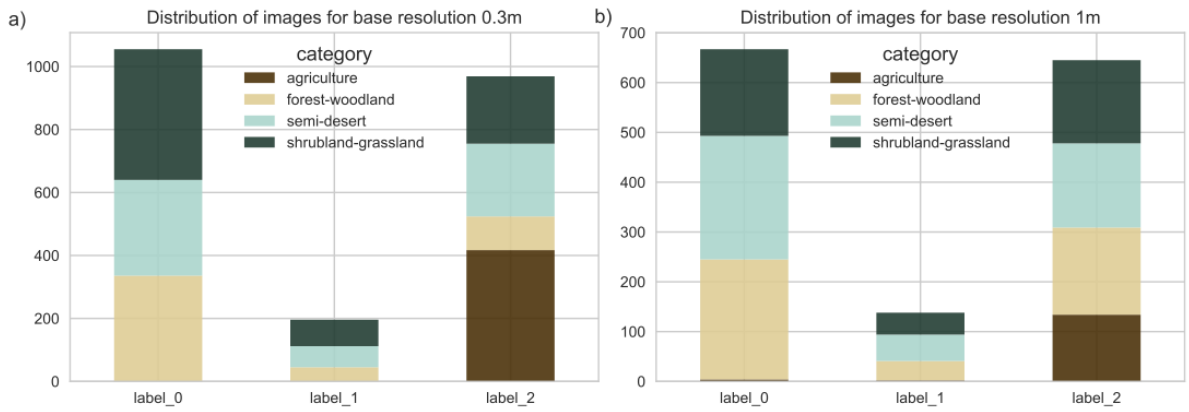
The datasets were build following these steps:

1. Download large raw images.
2. Crop images of size 512×512 pixel from the large on (as the grid in the picture above).
3. Label images with either zero (no human impact), one (minimal human impact), two (clear human impact).
4. Degrade images i.e. reduce number of pixels and thereby resolution per pixel.

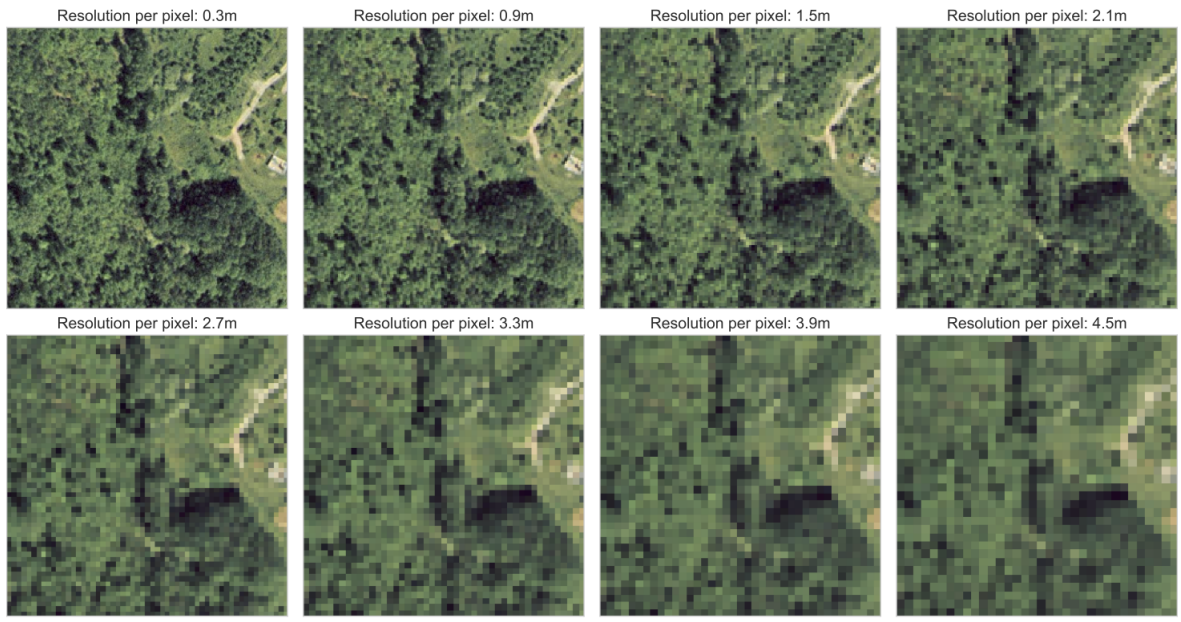
That way, we obtained smaller labelled images for each category. The image below shows examples of the agriculture images:



Annotating the images with labels was performed following certain rules, in order to ensure the consistency and variety of the datasets. The plots below show the distribution of the resulting datasets, from which we developed our models:



Finally, the processed and labeled images are downsampled with a Lanczos filter, an example of which is shown in the image below:



For this particular image one can observe how certain image features disappear as the image quality is decreases: the building on the right side is not identifiable above 3m resolution, or the texture of the track is blurred above 4m resolution. This shows how different elements in an image are not recognizable anymore once the resolution approaches their characteristic size.

3. Deep Learning

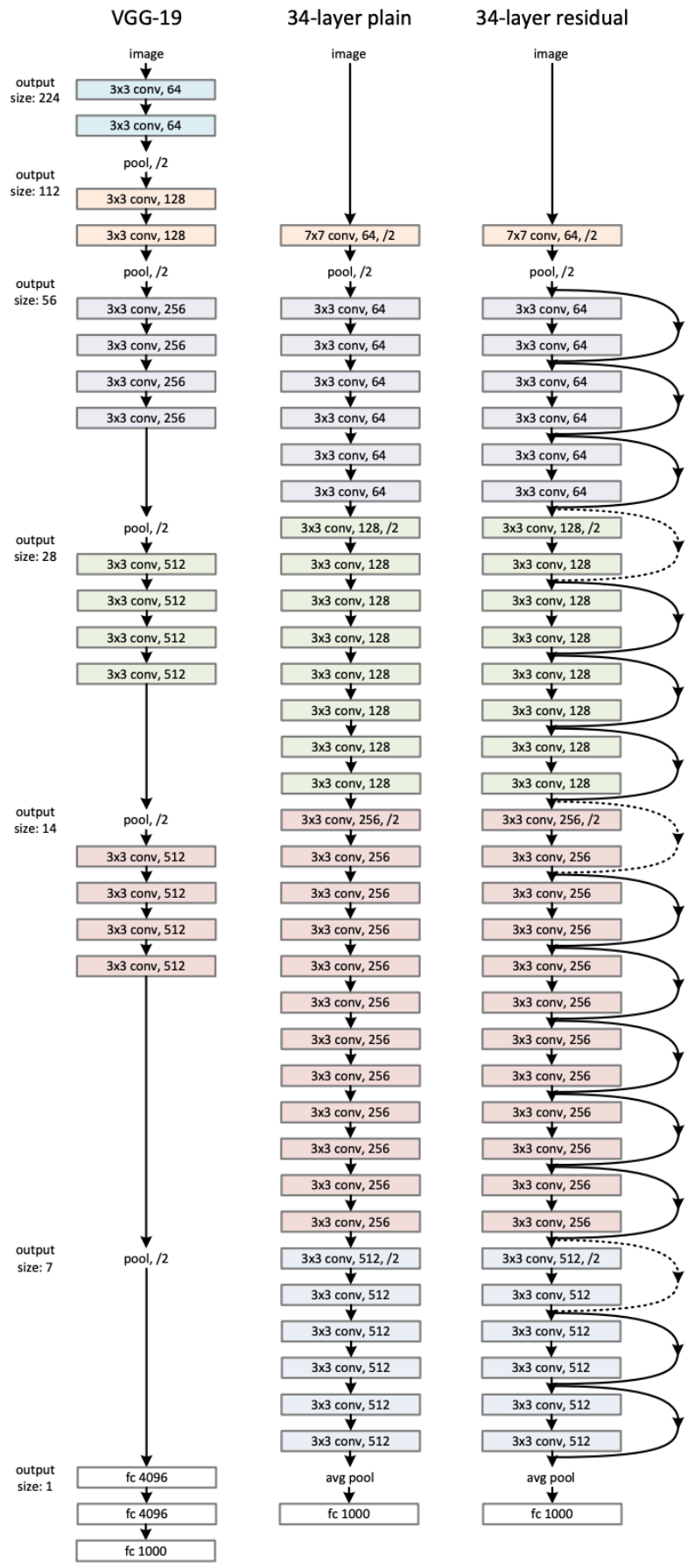
Deep Learning (DL) models have led to vast performance improvements in a large variety of domains, and therefore have gained substantial popularity over the last decades. These models were initially inspired by the human brain and analogies in neuroscience, which is why this class of algorithms was coined **Neural Networks** (NN).

One of the most common architectures in NN are the **Convolutional Neural Networks**, which have driven major breakthroughs in visual object recognition, and image, video and audio. CNNs are specifically designed to process input data that has the shape of multiple arrays, such as the pixel values of a 2-dimensional image with three color channels. This is accomplished by using additional layers to preserve spatial structure. In general, a CNN is composed of several convolutional layers followed by a nonlinearity. These are often followed by a pooling layer, and a fully connected layer is used as the last layer of the network. With this design, CNNs take advantage of the natural properties of images. The central element here is the convolutional layer, which takes into account that local pixel values are highly correlated, and that the local statistics of images are invariant to translation.

CNN architectures

There exist several CNN architectures that have achieved remarkable results recently:

- The **AlexNet** was the first convolutional Neural Network that achieved remarkable results in the ImageNet classification task in 2012. It halfed the error in comparison to all competing non Deep Learning based approaches.
- The winner in 2013 was the **ZFNet**, which basically had improved hyperparameters compared to the AlexNet.
- In 2014, two networks were developed that were significantly deeper than previous networks: the **VGG** network (with 19 layers) and the **GoogleNe**t (with 22 layers).
- A significant improvement in the ImageNet challenge was achieved in 2015 by Kaiming He et al., when they submitted **ResNet**. They introduced residual blocks that contained an identity mapping in parallel to the convolutional layer, which allowed them to train a deeper network without compromising its training and accuracy. Taking advantage of this approach they were able to design networks with a depth of up to 152 layers, which allowed for halfing the error rate of the ILSVRC challenge in 2015.



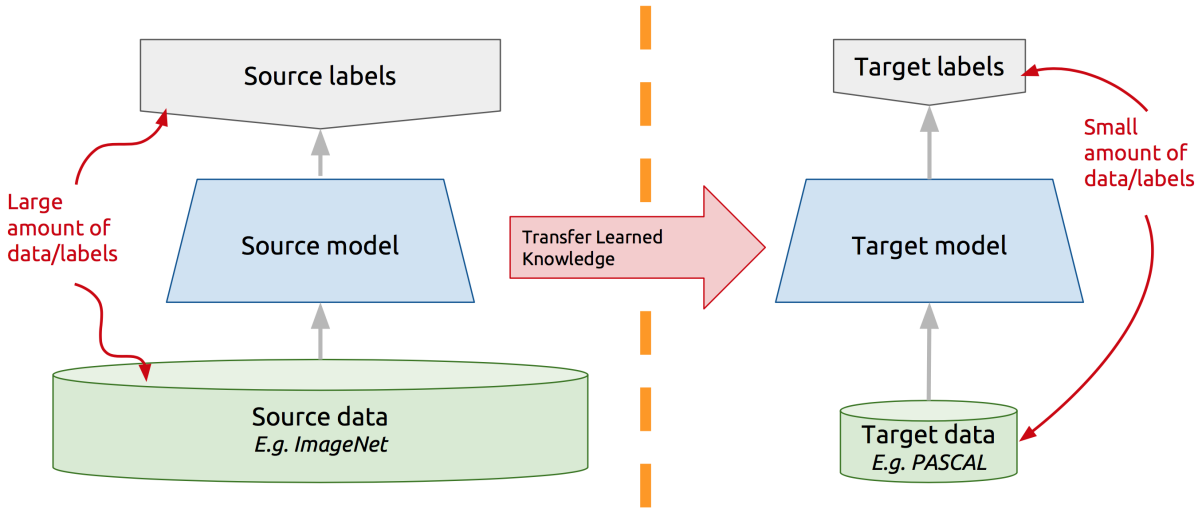
In recent years there have been developed many networks that go beyond ResNet. Some of them are extensions of ResNet (**ResNext**), or combinations of ResNet with other architectures (**Inception-V4**), or networks that do not make use of a residual block and instead use layer dropout (**FractalNet**). Nevertheless, ResNet is still a state-of-the-art network, and that is why we have used it for the experiments in this work.

4. Proposed approach

In order to train a model based on images, some sort of features need to be extracted. Traditionally, this image feature extraction was based on a set of hand-crafted detectors aimed to detect edges, corners, blobs and other feature descriptors. Some of these detectors are the Sobel filter, Laplacian of Gaussian (LoG), Difference of Gaussians (DoG), Determinant of Hessian (DoH), SIFT, SURF, Histograms of Oriented Gradients (HOG) and Gabor filters.

More recent approaches to image classification using Neural Networks have benefited from the existing and increasing computational power, and deep Convolutional Neural Networks have been able to achieve higher performance than traditional models.

Yet, training a deep CNN from scratch for a particular problem requires a large and exhaustive dataset along with a huge amount of computational power. However, it has been shown that the architectures of pre-trained NN can be reused for other purposes and achieve equally great performance. This is known as **Transfer Learning** (see Figure below).



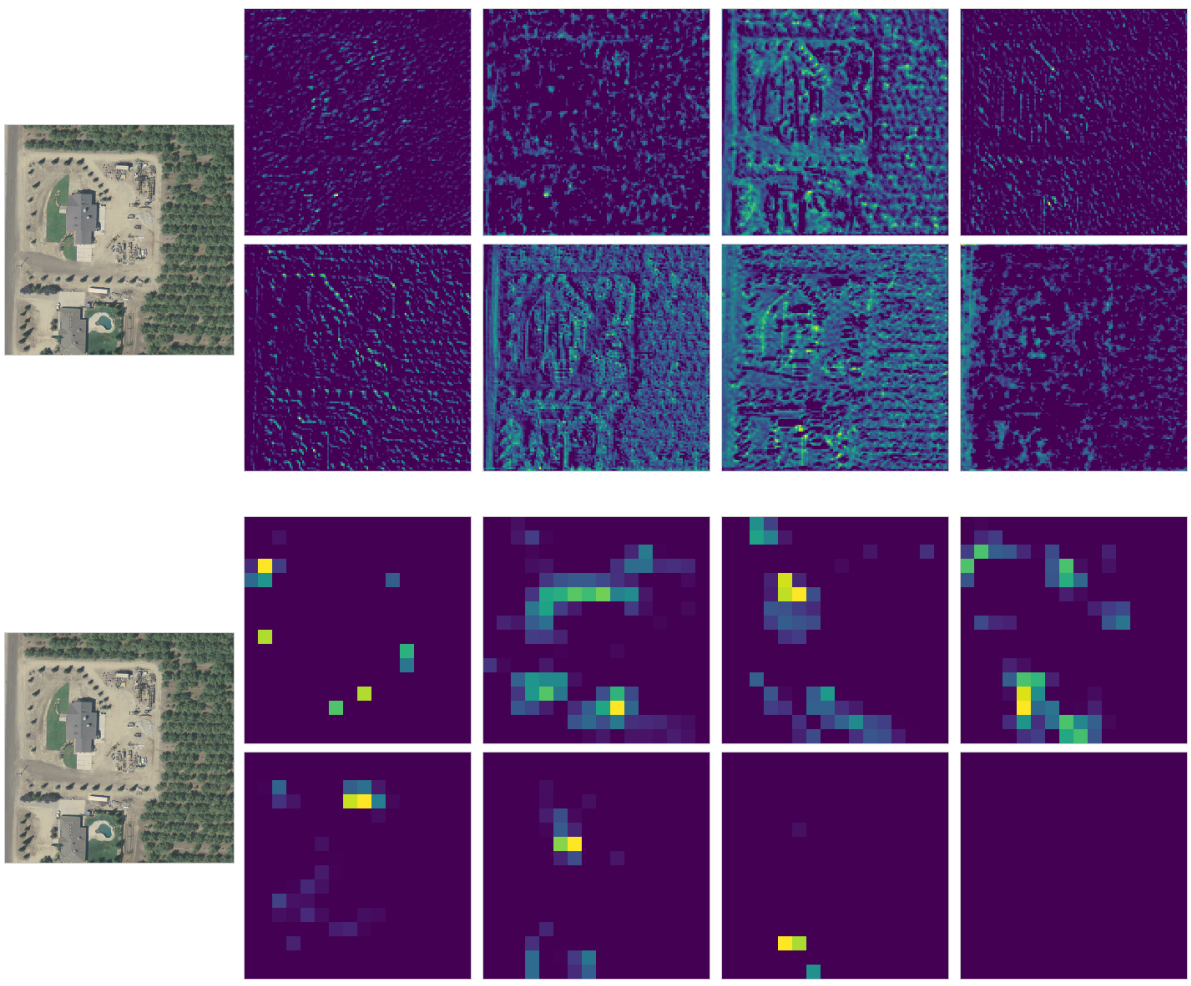
These pre-trained architectures can be re-purposed by reusing the learned weights and either replacing the final layers of the net by some other classifier, or even fine-tuning all the layers for the specific problem. In any case, the initial layers of the Neural Network provide a great image feature extractor.

Architecture

The architecture we propose for our problem consists om the activation layers of a ResNet, which act as the feature extractors of our images, followed by a shallow classifier made of a single dense (fully connected) layer. The ResNet we consider (ResNet50) has a total of 49 activation layers, so the output at each of them is different in size. Initial layers are able to recognize edges, textures and patterns while keeping an image size similar to the input. On the other hand, deeper activation layers show convolutions of higher order hierarchical structures.

For instance, for an input image of (tensor) size $512 \times 512 \times 3$ (a $512x512$ image with 3 RGB channels), the output of the first activation layer is of size $256 \times 256 \times 64$, the 10^{th} gives a $128 \times 128 \times 256$ tensor, and the last 49^{th} activation layer outputs $16 \times 16 \times 2048$.

The images below show 8 activation maps both for the 10^{th} (top) and the 49^{th} (bottom) layer of a sample image from the dataset (from the agriculture category). Some of the 10^{th} activations are particularly sensitive to edges, shadows, or textures, which later translate into more abstract outputs at the 49^{th} layer.



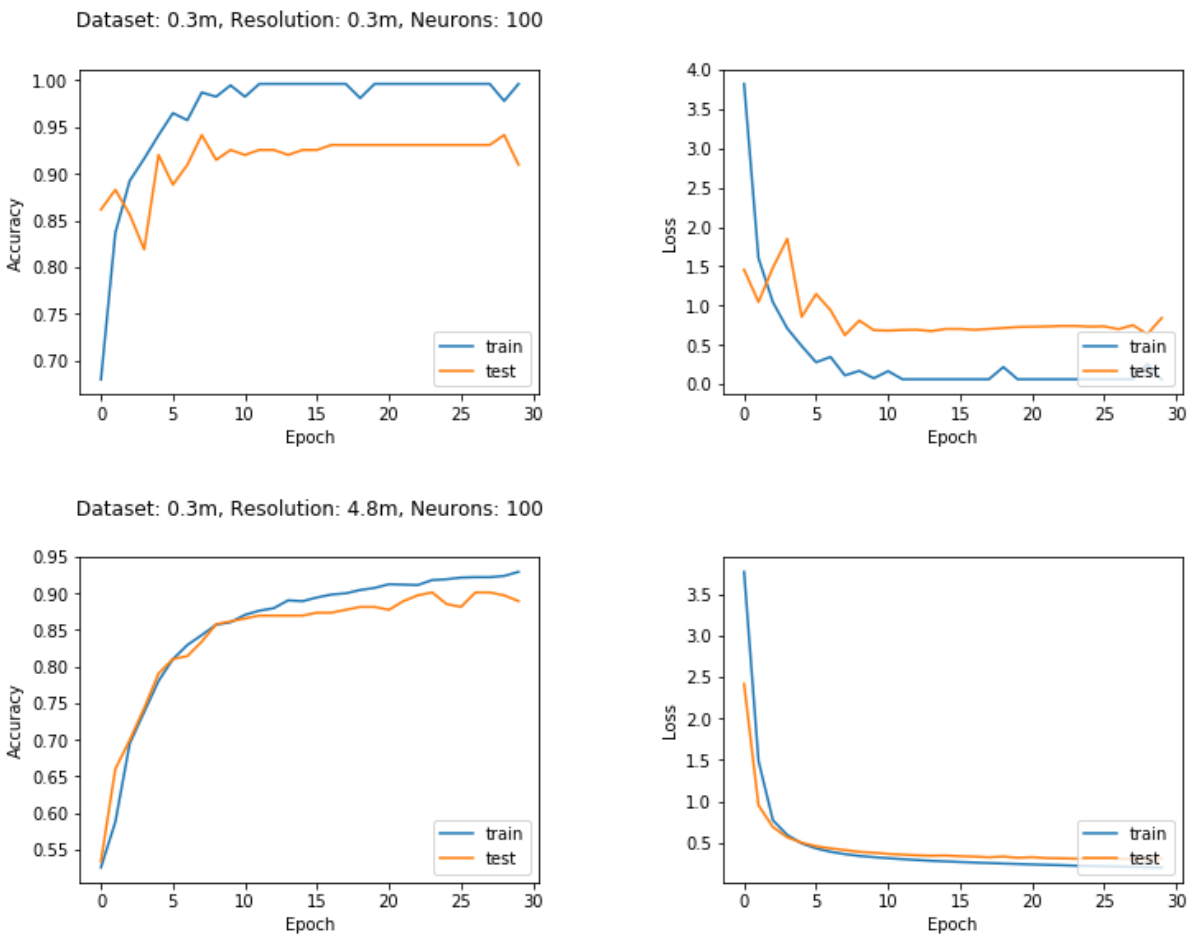
Our complete architecture consists on taking the output of the last (49^{th}) activation layer of the ResNet as the features of our images and passing them through a single dense layer of 100 or 200 neurons with ReLU activation, followed by a single dense node with a Sigmoid activation acting as the classifier. This model is trained on the dataset with RMSprop optimizer and a binary cross-entropy loss function.

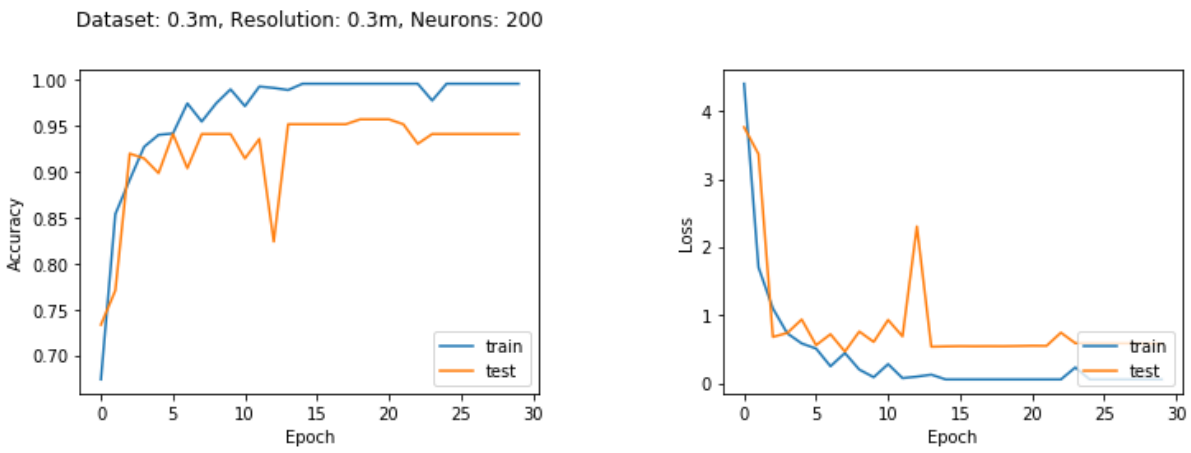
Training pipeline and experiments

This architecture is used to train models for several image resolutions obtained from our datasets. To do so, we do the following steps for each of the datasets:

1. Load the original images (at the original resolution) from disk along with the human impact labels and categories.
2. Downsample the images to the desired resolution.
3. Compute the ResNet activations (at the 49^{th} activation layer) of the resulting downgraded images.
4. Consider a stratified KFold split of the dataset (with 8 splits) for cross-validation. That means, the dataset is split into 8 sets with labels 0-1 equally distributed.
5. Train the model separately for each combination of the 7 training sets considering 30 epochs. The remaining set is used as a validation set to assess the accuracy. After training on all folds, the results of the 8 experiments are averaged in order to obtain more consistent measures.
6. Repeat the process for all downgraded resolutions.

The plots below show the convergence of some of the models trained for the $0.3m$ dataset. In general, the NN is able to converge and achieve a good accuracy (as shown in the plots), although for some particular splits of the data, it fails to converge and stays in a low accuracy point. This is probably due to the fact of having a relative small dataset. Hence, these particular folds are not going to be considered when computing the final averaged results.





5. Results

In this chapter we discuss the results obtained in this thesis. First, we analyze how the model works for a few given resolutions, so that we can be confident about its performance. Then, we consider how the accuracy changes with the resolution and within the categories. Finally, we discuss the cost around Satellite imagery to analyze once the entire earth.

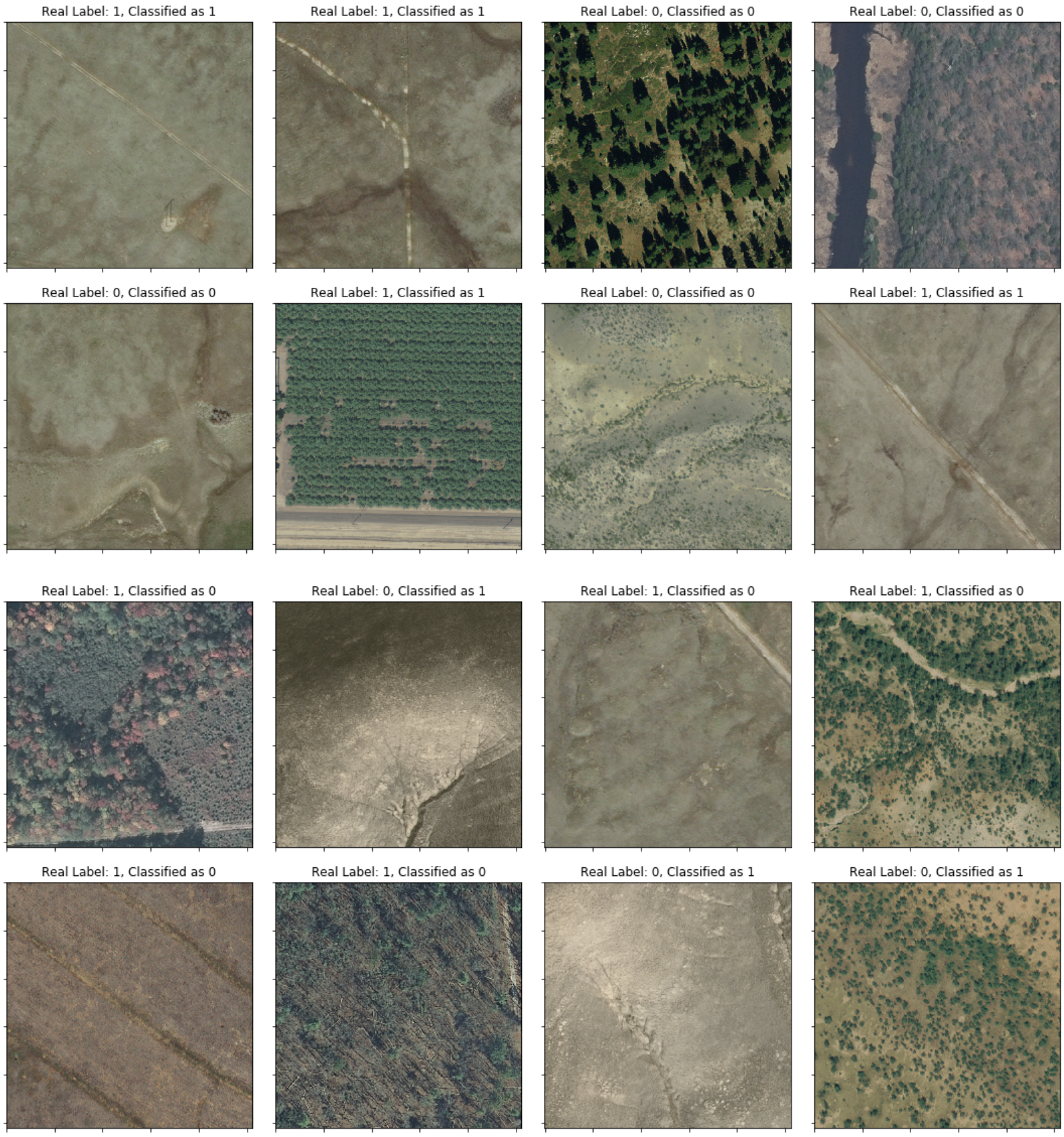
Transfer learning on aerial imagery

The first thing we want to investigate from our experiments is whether the approach followed is able to achieve good results. That is, we want to evaluate if using a pre-trained ResNet as a feature extractor for aerial images allows the trained model to properly discriminate the existence of human impact.

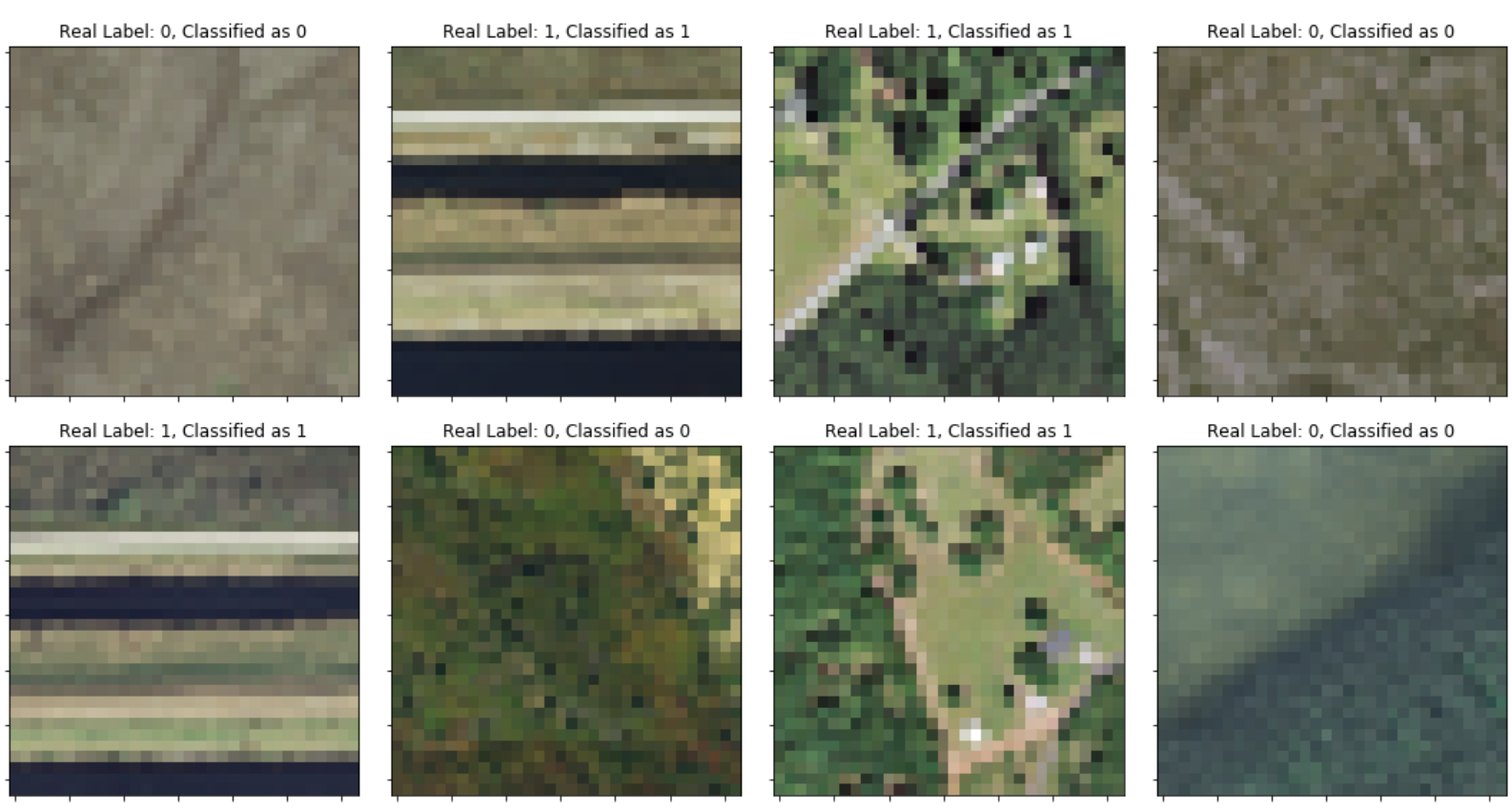
Tables ??? - ??? in Appendix ??? show that the accuracies obtained with all the experiments are indeed remarkable. All these results are discussed in more detail in the next section, but let us begin by focusing on few cases of the $0.3m$ dataset in order to understand how the models are behaving. In table ??? we can see that an accuracy of around 0.9226 is achieved on the base resolution ($0.3m$), while it drops to 0.8690 of the last downgraded resolution of the dataset ($4.8m$).

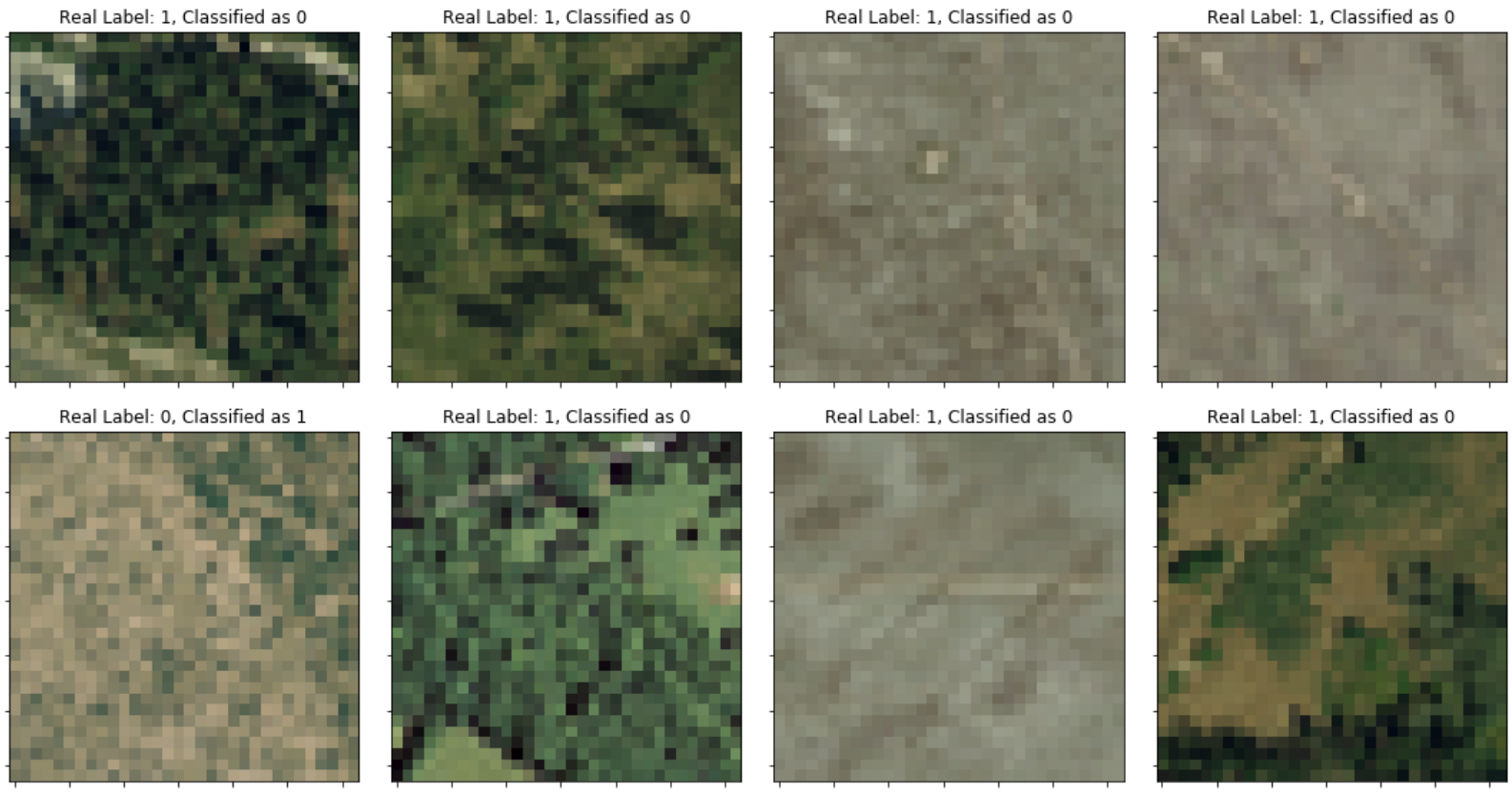
Let us consider first some examples of correctly and wrongly classified images at the base resolution (for one of the cross-validation folds), which are shown in Figures ??? and ??? respectively. The first set of samples shows that the model accurately detects clear human impact related to agriculture (2nd picture in the second row) and paths. On the other hand, the second set shows that it might fail to detect it when the impact is subtle, covering a small region of the image, or when it can even be confused with natural structures (or vice versa).

Note that, from this point, \textit{label 1} refers to images with clear human impact, which were defined as \textit{label 2} when building the datasets in Chapter ???.

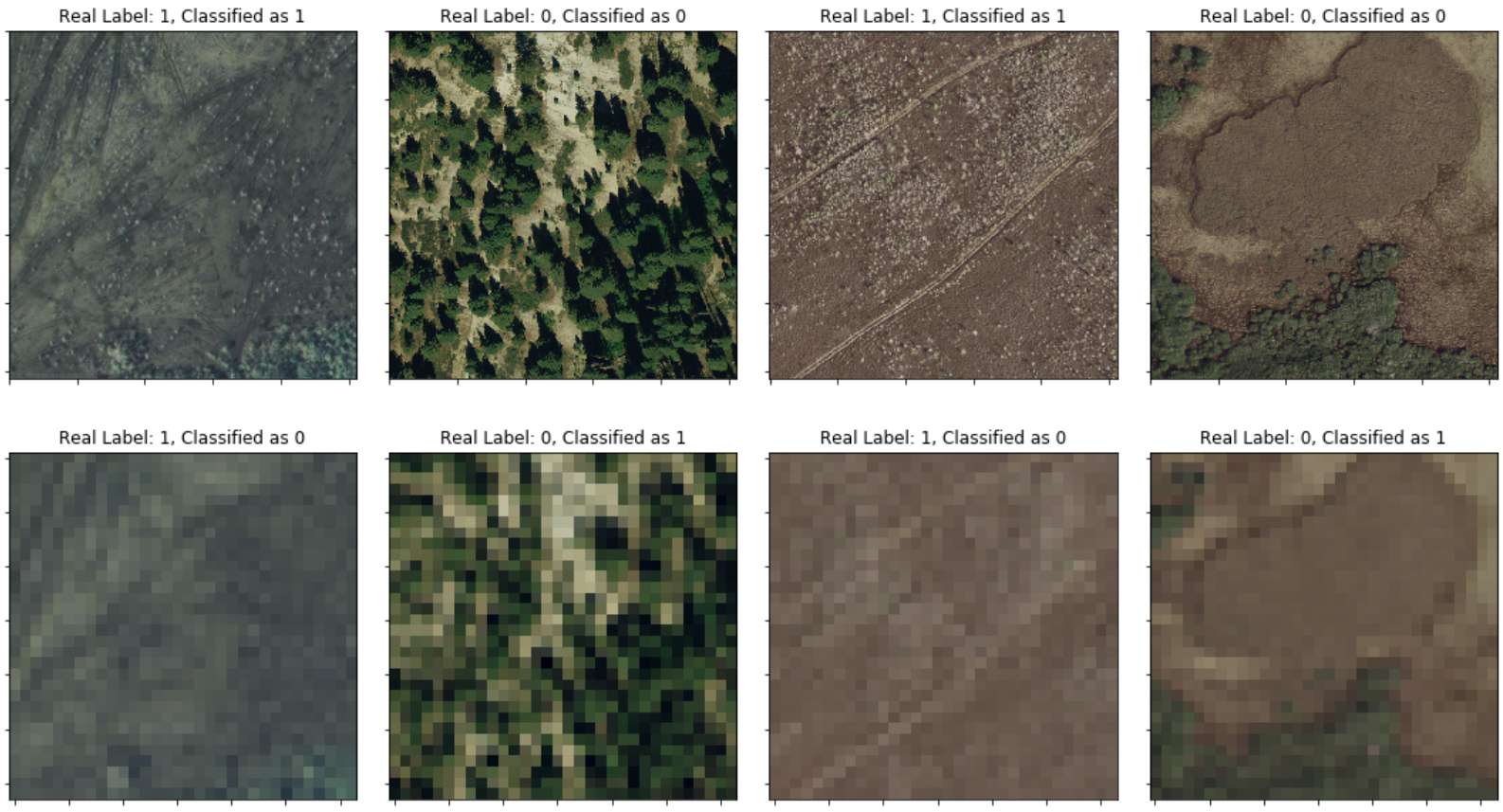


The same kind of analysis can be done for the last resolution, $4.8m$. Figures ??? and ??? show correctly and wrongly classified images at this resolution. The first set shows that the model detects human impact when it is still evident, even with the low resolution, but the second set indicates that it commits mistakes when the human impact evidences are lost with the downgrade process. Similarly, it might classify as man-made structures patterns that are indeed natural.

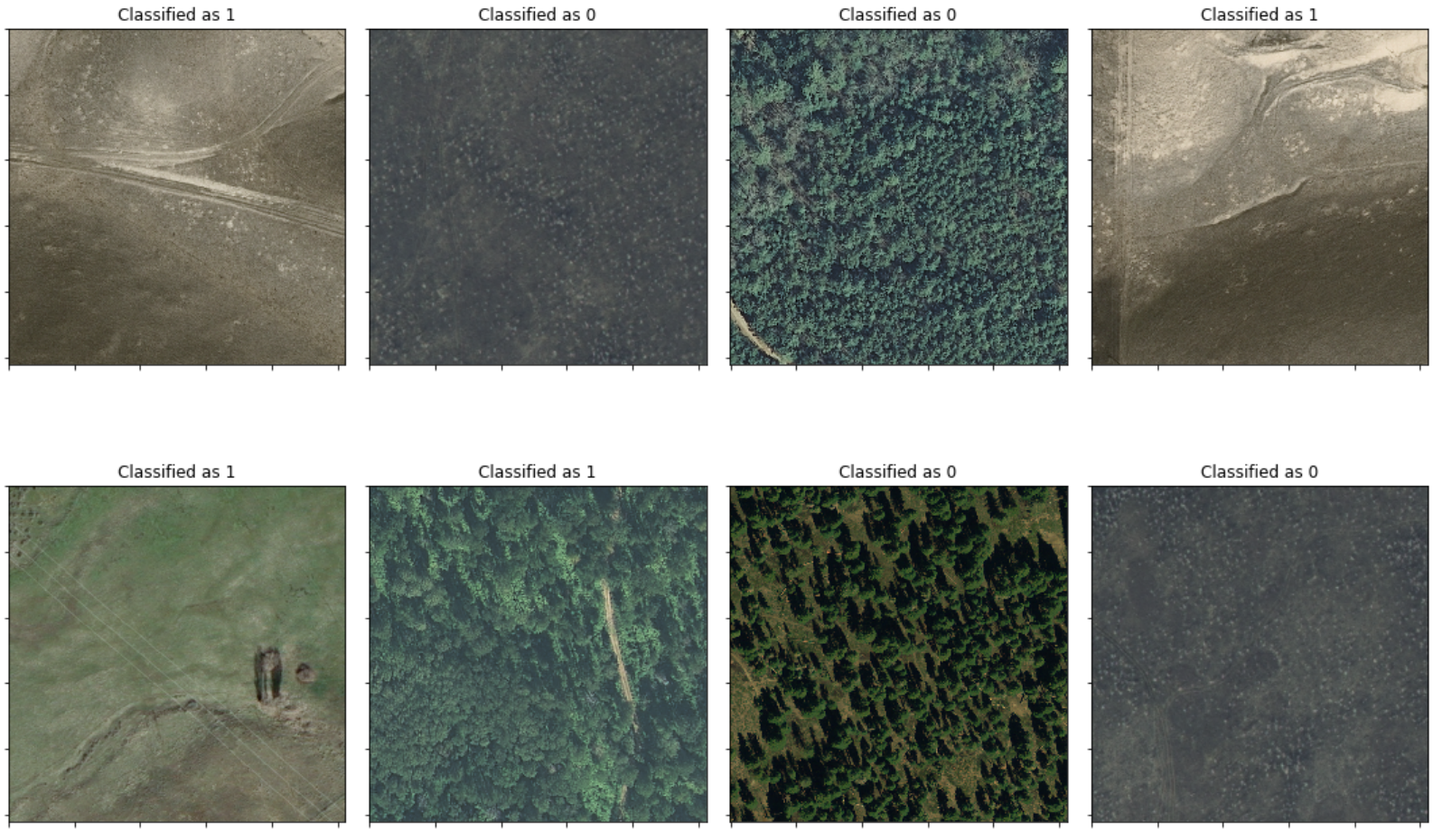




This can also be seen with Figure ???, which shows images that are correctly classified at $0.3m$ resolution (top row) but wrongly classified at $4.8m$. The first and third pair of images demonstrate that, when the human impact is subtle, the model missed it in the downgraded resolution. Conversely, non human activity can also be misclassified at lower resolutions (second and fourth images).



Finally, we can investigate how the model behaves with images where human impact is very subtle. For this purpose, we consider the images with the intermediate label (\textit{label 1} in Chapter ???, Figure ???). The model has never been faced with these images, so this can give a good perception of whether the models have been able to learn relevant features of human impact. Figure ??? shows some of these images with the label predicted by the model in the title. Even if man-made structures in these pictures are small, the model is able to detect straight lines and shapes as human activity.

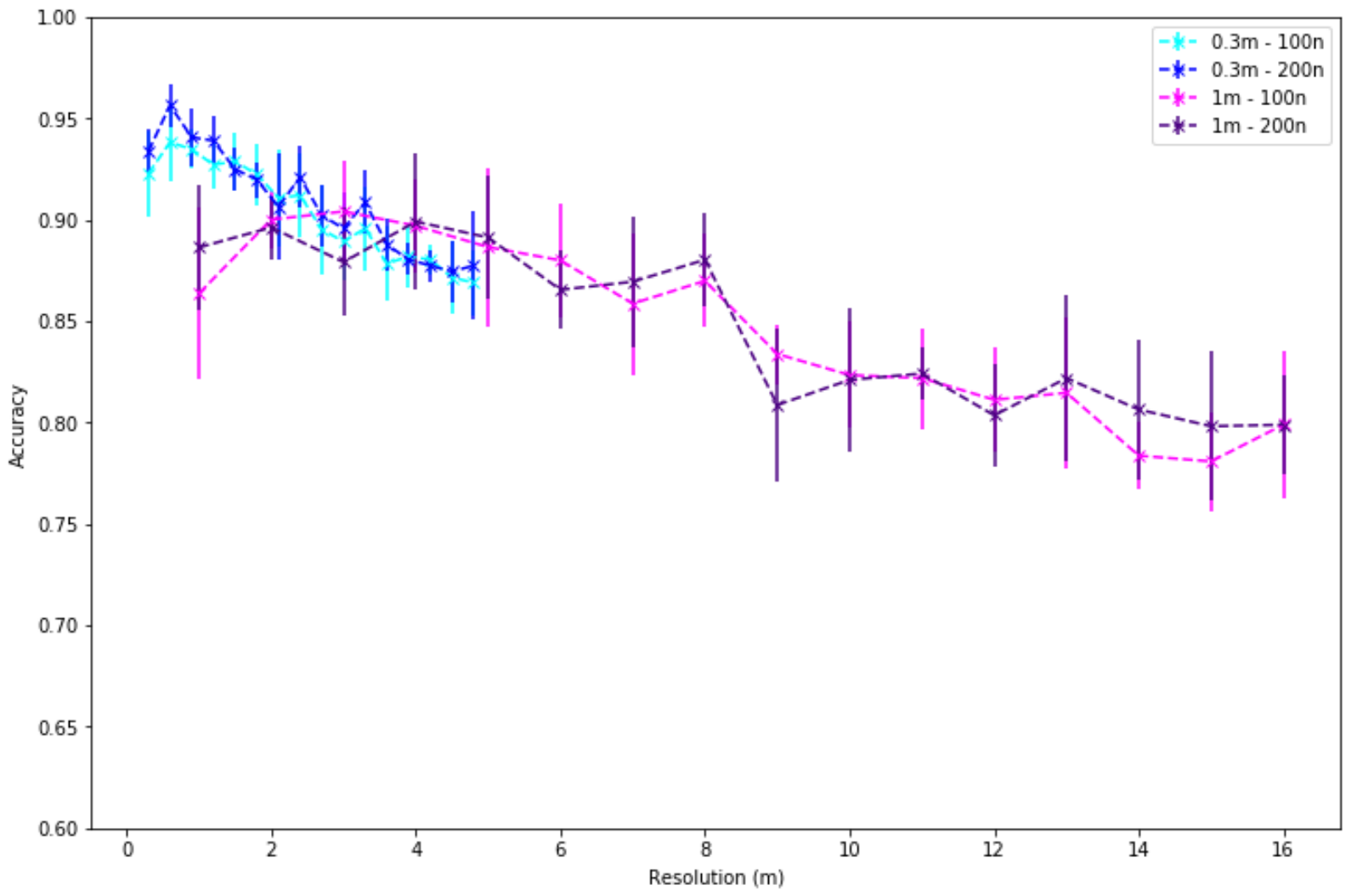


All in all, we can conclude that the model is able to achieve a great accuracy. It is able to generalize to unseen, subtle images, and still be accurate for lower resolutions, were the amount of pixels and information per images become much smaller.

Man-made structures detection at different scale

Now that we know that the trained models are able to detect, with high accuracy, human impact from our datasets, we are ready to analyze the results for different resolutions. From all the experiments (datasets, architectures, resolutions and cross-validations), the results have been stored and aggregated. As mentioned in the previous chapter, the models were not able to converge for some particular splits of the datasets. Hence, these few folds have been ignored when aggregating the results. Tables ??? - ??? in Appendix ??? summarize the accuracies obtained for each of the datasets ($0.3m$ and $1m$) and downgraded resolutions, both the overall accuracy and by category.

Similarly, Figure ??? shows the overall accuracies obtained for all resolutions. From this we can see that similar accuracies are achieved for both datasets on the same degraded resolution, which means that both datasets are comparable and can be considered together. Also, we realize that increasing the size of the model from 100 neurons to 200 does not have a big impact on the accuracy, but tends to perform slightly better. Hence, for other results later we will focus on this architecture only.

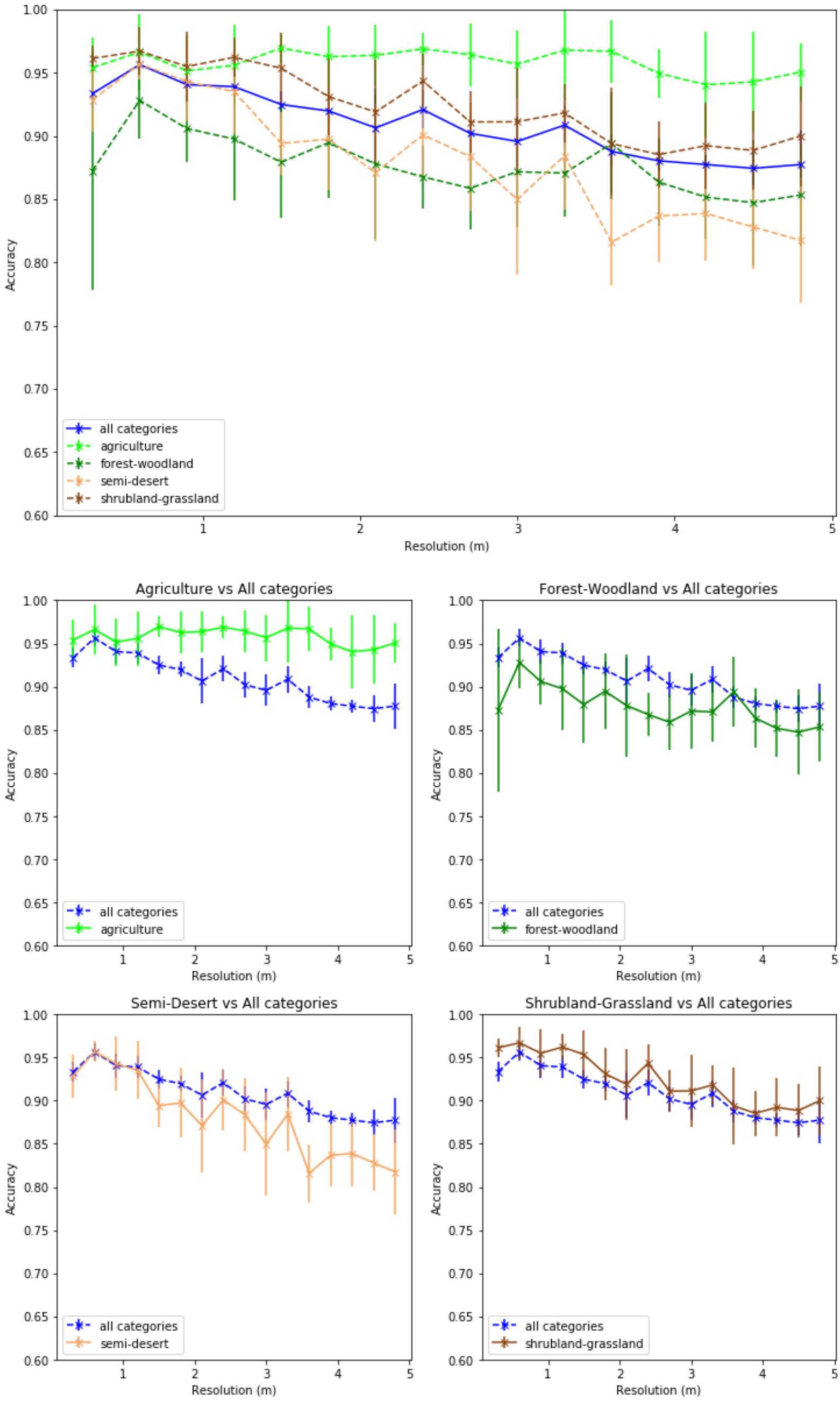


On closer inspection, we also detect that the accuracy on the base resolution ($0.3m$ and $1m$) is always slightly worse than the next resolution. This is probably because the input image size at the base resolution is quite large (512×512), which makes the dense layer much more complex to train. Indeed, Figures ??? and ??? in Chapter ??? already suggest that the models struggle to be optimized. More sample images would be required in order to compensate for the complexity and achieve a greater accuracy.

Finally, the overall conclusion from this plot is that, as expected, better resolutions (less than $2m$) allow for greater accuracies, of over 90%, which means a great success considering that the images in the datasets are balanced between having or not human impact. Furthermore, accuracy is still good for resolutions between $2m$ and $8m$, staying between 85% and 90%. From $8m$ onwards, accuracy drops to 80% and the model is not able to

detect more subtle elements of man-made structures.

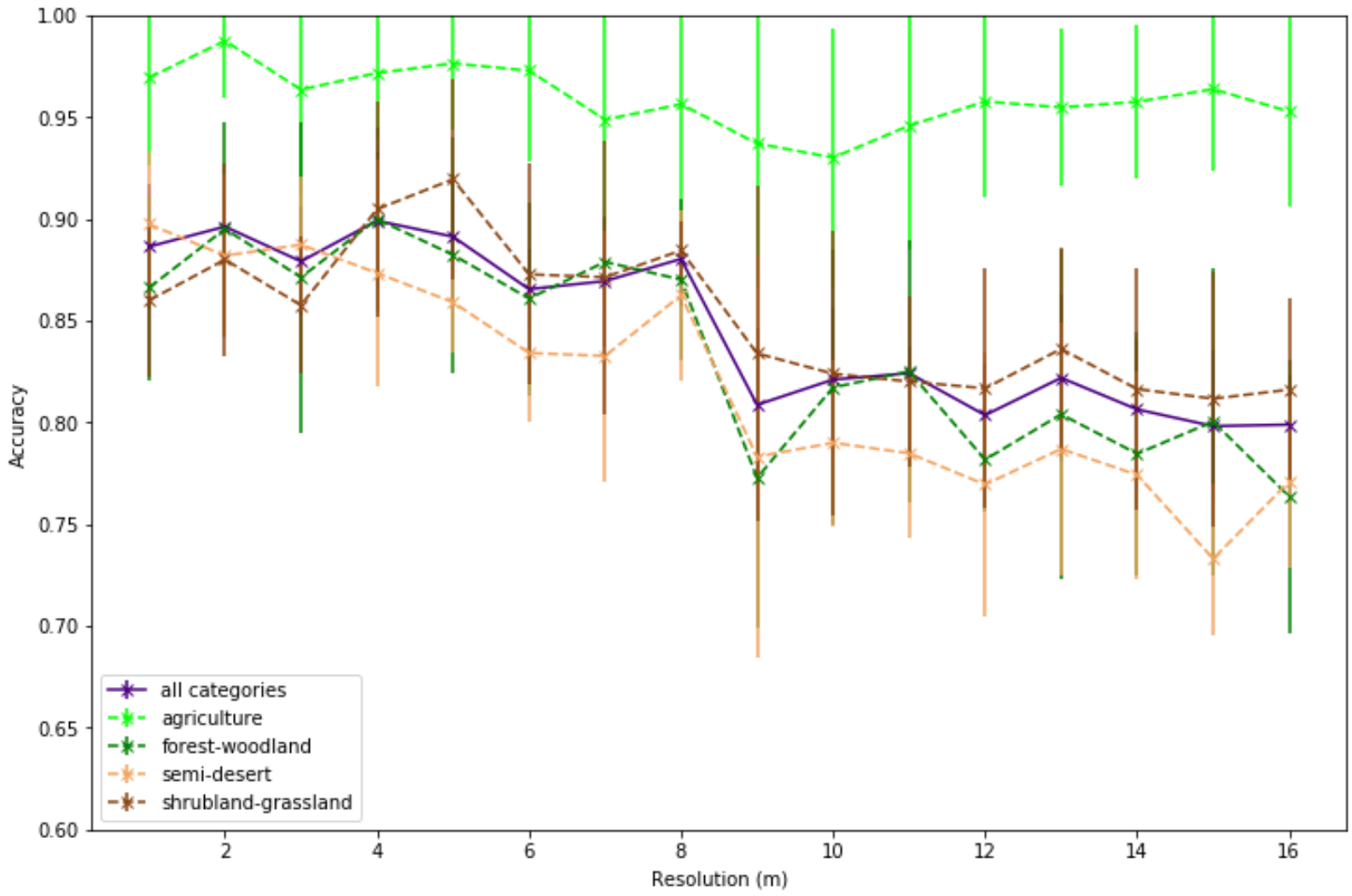
Now let us consider how these accuracies behave for each of the land use categories in the datasets. As discussed in section ???, these categories are rough approximations of the kind of terrain and human impact, with images that could be exchanged between categories, but overall these can give and idea of the accuracies when analyzing different kind of terrains. Indeed, Figures ??? and ??? show that, for the $0.3m$ dataset (and 200 neurons model), accuracy differs substantially between categories. Fig. ??? shows the global comparison between categories, while Fig. ??? allows for a better comparison of each category with respect to the overall accuracy.

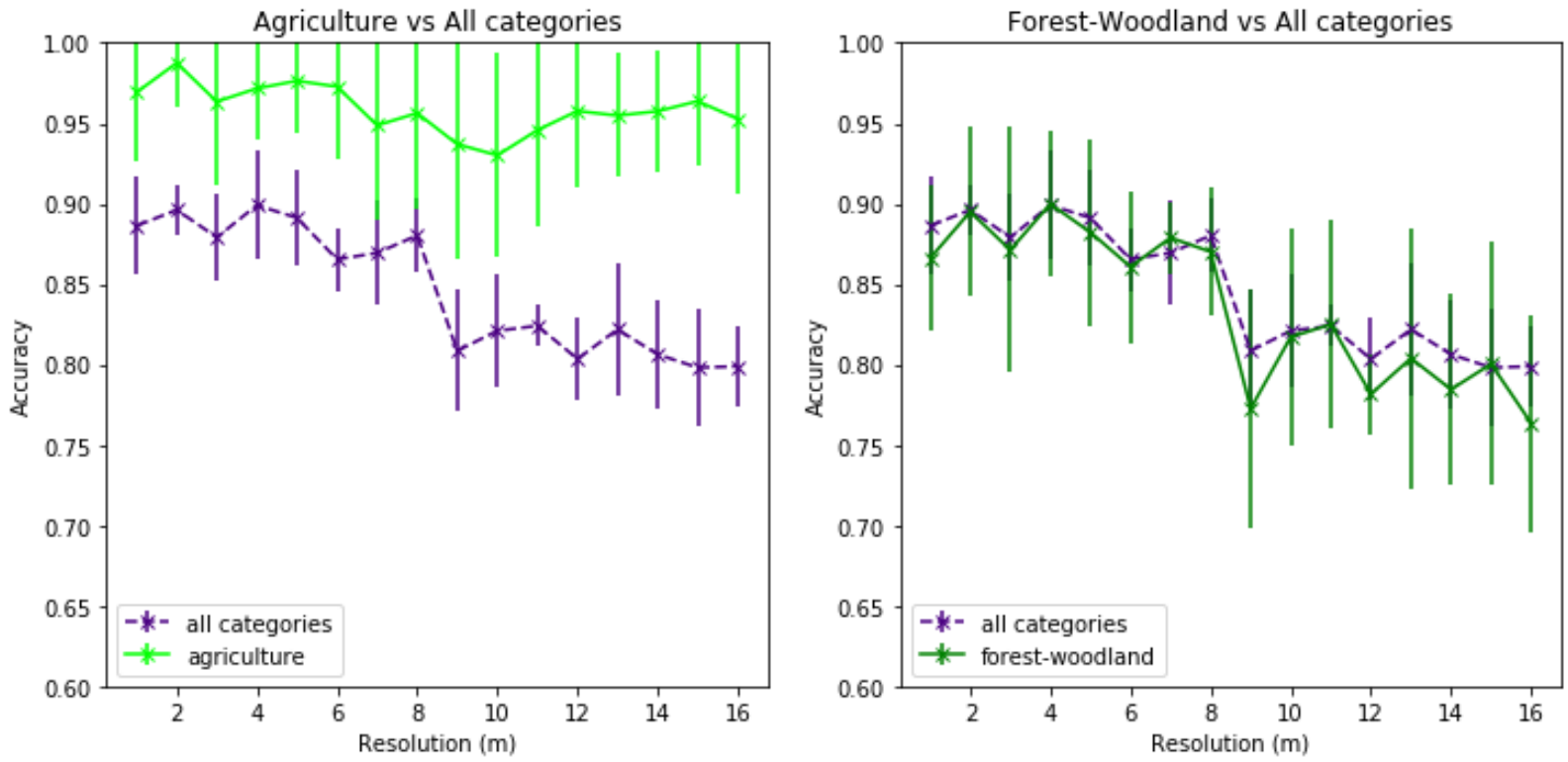


These plots have been obtained once the model was trained for all the categories. Then, the accuracy on the validation set was computed for each individual category and over all images in the set. The validation set in each iteration of the cross-validation was small, consisting of few hundred images, and a homogeneous representation of the categories was not imposed (validation samples were picked randomly, only preserving proportion between having or not human impact), which explains the large variability for each experiment (vertical lines in the plots).

Although the category is not taken into account when training, we can see that the models are capable of detecting agriculture-related human impact with high accuracy (over 95%), without really being affected by the drop in accuracy. Shrubland-grassland category also has a good accuracy, while the model performs worse in semi-desert and forest-woodland images. That means that the models are able to capture textures and patterns related to agriculture quite easily, while the other categories have more variable features.

Similar results are obtained for the $1m$ dataset, which are shown in Figures ??? and ???. The models are able to achieve a great accuracy when detecting agriculture-related human impact (over 95%), independently of the resolution considered, but the performance drops for the other 3 categories. The biggest drop is observed at $8m$ consistently over the three non-agriculture categories.





Cost estimation

We discuss the financial cost associated to building and launching a satellite, and to renting infrastructure for performing the entire image processing pipeline. We further study the cost as a function of pixel resolution. However, our estimates are very rough approximations because many factors are involved and large variations occur between them. To give an example, choosing one material over the other might change the cost of manufacturing and launching a satellite by one order of magnitude. It is also completely different to have a satellite for 3 years in space, or to target a lifespan of 20 years.

Having this in mind, we follow laws from physics to estimate the dependency of the satellite cost on resolution. First, the cost of launching a satellite into the orbit scales linearly with its mass, which is given by the amount of fuel needed. Second, the mass of the satellite scales quadratic with resolution so that overall we obtain a cubic dependency of financial cost on resolution. The latter increase in cost is associated with the optical instruments used. As a reference for the satellite cost we use a Skysat satellite from Planet \parencite{skysat_planet} that has a resolution of about 1m and a value of $\$30$ million. This amount was provided to us by Satellogic and includes construction, launch and maintenance during the satellite's lifespan.

Our final goal is to give an estimation of the expenditure to monitor once the entire surface of the earth (about 149 million km^2). To this end, we multiply the satellite cost by the ratio: time needed to scan the earth over the satellite's lifespan. Further, a satellite can map 1 million km^2 at 1m resolution in 4.2 days \parencite{satellogic_youtube}. We hence can calculate the satellite cost per km^2. With $area\ per\ lifespan = 10^6 \times \frac{10 \cdot 365}{4.2}\ km^2$ we obtain $cost\ satellite\ per\ km^2 = cost\ satellite/area \approx 0.035\ \$/km^2$.

\begin{table}[h!]

```
\begin{tabular}{l | l | l | l | l}
      description & cost & unit & cost ($/km$^2$) & cost ($/pixel) \
      \hline
      process raw data & & & 0.004 & $4 \times 10^{-9}$ \
      hot storage  & $72\times 10^{-6}$ & $/(km$^2$/month) & 0.000864 & $8.64\times10^{-10}$ \
      cold storage  & $36\times 10^{-6}$ & $/(km$^2$/month) & 0.000432 & $4.32\times10^{-10}$ \
      archive storage  & $9\times 10^{-6}$ & $/(km$^2$/month) & 0.000108 & $1.08\times10^{-10}$ \
      download data & 8 & $/Gb & 0.021 & $2.1  \times 10^{-8}$\
      serving to final client & 0.09 & $/Gb & 0.000236 & $4.7232 \times 10^{-10}$\
      prediction (AWS) & 0.05 & $\sim$6 & $/h & s/km$^2$ & 0.00145 & $1.45 \times 10^{-9}$\
  \end{tabular}
```

\caption{\textbf{Costs for image data processing.} All costs except the prediction are provided by Satellogic.}

\end{table}

Another cost intensive block when capturing satellite imagery involves image data processing for which the cost scales quadratic with resolution. For example, an operation that costs $100/km2at1mresolutionwillcostonly1$/km^2 at 10m resolution. The data processing step consists of multiple parts: transformation of raw data into image pixels, storing data in a hot, cold, and archive storage, downloading data from the satellite, serving it to

the final client, and in our case predicting human impact. These costs are summarized for 1m resolution in table ???. Note that we used the conversion factor 0.002624 for an image to convert from Gigabytes to km^2 (2× compressed) and we assume 12 months of data storage. The prediction step is estimated by loading 4 images that each have an area of about $500 \times 500m^2$, calculating the ResNet activations of the final layer, and predicting the class using the models trained in chapter~??? in an ensemble fashion. This part amounts to a processing time of 6s for an area of 1km^2, which can be converted into costs per km^2 assuming 0.05$/h of AWS EC2 compute~\parencite{aws}.

To finally obtain the dependence of the resolution on the total financial cost we sum the data cost per km^2 and the satellite cost per km^2 at 1m resolution, and convert to cost per pixel ($\times 10^{-6}$). We then multiply with the number of pixels necessary to cover the entire surface of the earth. Here the satellite cost per km^2 is a cubic function and the earth surface in pixel is a quadratic function in resolution. The result is shown in Fig.~???. We obtain a cost of about $15 million dollars at 1m resolution with a very step slope towards better resolutions. At 0.3m resolution the cost is two orders of magnitude higher than at 1m while for worse resolutions the cost decreases by two orders of magninute when the cost is a factor 10 larger. We conclude that for worse resolutions the data processing cost is the dominating cost whereas for very good resolutions the satellite cost dominates.

6. Conclusions

In this final chapter we discuss and close the different aspects of the project, from the problem definition itself, to the datasets build and models developed, and we conclude our thesis with some further work ideas to continue and enhance our approach.

The Problem

The intial phase during the development of the project consisted on clearly defining the problem to study. The goal was to investigate which satellite imagery resolutions allowed for an accurate detection of man-made structures, and what would be the cost associated. For that, we needed to define the scope of human activity to consider, look for suitable datasets for this study (which eventually lead to building our own datasets), and define the actual technical problem to be modeled, in order to evaluate the accuracy by resolution.

Both for the datasets and the problem, we needed them to be feasible enough to not require high technical and computational efforts (which would be, for instance, trying to detect every type of human activity in the images, providing their position and shape, and classifying them into several more categories). On the other hand, we needed it to be a realistic situation, so the results obtained could be extrapolated to other, more complex scenarios.

All in all, having a well-defined problem scope and a good approach to tackle it allowed us to achieve remarkable and realistic results.

The Datasets

After investigating existing datasets of labeled satellite images, we could not find the one suitable for our purpose, as most of them were mainly focused towards urban areas, or were just build for some other different goal that would not work for us. Hence, we decided to build it ourselves. It had to be representative enough to pose a challenge for our models, yet feasible to be build with our available time and resources. The four categories considered (agriculture, shrubland-grassland, semi-desert, and forest-woodland) and the balance between non-existent and existent human impact images allowed us to build a good and representative dataset, which makes reasonable to eventually generalize the results obtained to other scenarios.

Of course, we acknowledge that having a larger dataset of images, annotated with higher degree of detail, like position, shapes and types of man-made or natural structures, would be great to build high performance models, capable of detecting all sorts of human impact with far better accuracy. Nevertheless, this high-precision goal could not be fitted into our general purpose.

The Models and Results

Using a pre-trained CNN like the ResNet helped us to speed up the training process and achieve good results without requiring a very large dataset and computational power. The binary classification problem considered turned out to be feasible and representative of how accuracy is affected with a decrease in the image resolution.

From the results we observe that, as expected, the higher the resolution the better, but also that there seems to be sweet spot between 1-$2m$ and $8m$ where, except for the more subtle forms of human activity, most of the man-made structures studied are detectable with good accuracy. This trade-off with the resolution allows to consider more cost-economic satellite solutions without dramatically compromising accuracy and utility. For instance, operating a satellite at $2m$ (or $8m$) instead of $1m$ reduces the cost approximately by a factor 6 (or 100).

Further work

With all that said, we realize that there is still plenty of space for further work and investigations, so let us now indicate some of these ideas.

First of all, having a better dataset could help improving the investigations and opening new lines to explore. It could be improved and enlarged with more variate images, with a better and more consistent classification, and including more detailed annotations of the position and type of objects or structures appearing. This would allow to train more accurate models capable of detecting all these kind of human impact.

Regarding the model, other techniques for feature extraction could be studied, like other pre-trained Neural Networks, and the parameters itself (like the number of activations to consider, the architecture of the model or the training phase) could be further fine-tuned. Additionally, the pipeline could be made more robust so that it could ingest a larger amount of data, as part of the improvements suggested for the dataset. And, of course, having a powerful computational cluster would allow to speed up the processes and target more ambitious goals.

A more in depth study of the results could help to understand more precisely on which images the algorithms fail, what kind of information are learning (patterns, colors, shapes, etc) and how to enhance them.

Finally, it would be interesting to have a more detailed analysis of the costs associated to all this solution, from data gathering, processing and modeling to the production implementation itself. Also, taking into account other related factors, like infrastructure needed, legal aspects and ecological footprint \parencite{Strubell2019} would give a more complete idea of the viability of global satellite image analysis.

In conclusion, this project has allowed us to investigate a relatively new topic, covering from the data gathering to the technical implementation using state-of-the-art tools, and leaving the door open to further investigations.

In [ ]: