Mapping Urban Areas Leveraging the Analysis of Ground-Level Imagery with Convolutional Neural Networks

Henrique Metelo Rita de Almeida

Abstract

The technological advancements in mobile devices have allowed people to have the ability to easily take pictures and share them on the web. Within platforms like Flickr[1] and the Geograph.uk[2], we are presented with an almost infinite and ever growing number of community-shared pictures, often containing the location where they were taken, which might provide precious information for a variety of tasks. This paper proposes a deep learning approach that aims to exploit such information for mapping tasks for which top-down imagery is often not enough and where ground-level photos and close-up views can be the key to achieve accurate results. The proposed procedure is tested by recurring to ancillary data sources that provide land-use and scenic beauty geographic data in order to train a model that is able to efficiently and effectively automate terrain mappings. The experimental results show that by combining information from multiple photographs, and making use of deep convolutional and recurrent neural architectures, the envisioned model is able to produce automated land-use and scenic-beauty mappings, which can be considered a viable alternative to the currently used mapping methods that require hand-annotations.

1 Introduction

The recent growth in the number of community-shared geotagged ground-level images, mainly due to the increasing use of smartphones for taking photos, has provided a new source of information which may help identifying geographical features of the surroundings in which each photograph was taken. Several projects, such as the Flickr Creative Commons Dataset and the Geograph Dataset, were created with the intent of collecting and aggregating large sets of georeferenced pictures which might prove useful for a vast array of tasks, such as terrain mapping tasks.

In urban areas, mapping land use is critical for local governments that intend to execute smart and informed city planning initiatives to provide sustainable growth of the urban fabric. Land usage stands for the use that is given by humans to a certain occupied area. Some examples of land use categories are residential areas, industrial areas, sports facilities, or commercial zones. The task of creating these mappings cannot usually be accomplished using satellite level images as, although certain areas can be analysed relying on top-down images (e.g. sports facilities by identifying soccer fields), most of the times, certain details that can only be observed from a ground-level perspective, such as building facades, are the key to obtain such mappings. As a consequence, the production of this type of maps usually requires hand-annotations obtained by recurring to survey-based methods, which results in a huge upkeep cost due to the need for manual labour, and makes it impossible to keep the available data up to date.

This work follows on previous efforts such as that from Newsam and Leung (2019), exploring the possibility of leveraging community-shared ground-level images to fully automatize this type of task, by using deep convolutional neural networks (CNNs) for image analysis and by introducing a sequence analysis component that can help improve the quality of the obtained results by combining information from multiple photographs. This is achieved by using a recurrent neural architecture that is fed with a sequence of feature vectors extracted from a set of images, which are then processed in order to extract an output land-use class or scenicness score. The obtained results show that this method is a viable approach to improve the mapping results. The data and code used in this work are available in a public repository[3].

The rest of this paper is organized as follows: In Section 2, I provide a brief explanation about some basic concepts related to deep neural network models, which play a key role in this work. I also present several previous studies which address the use of community-shared ground-level images to produce mappings for a variety of tasks, such as land use, land cover or scenic beauty. Section 3 presents a definition of the objectives of the proposed work, along with a short description of the methodology and data sources. Afterwards, Section 4 presents a brief summary of the obtained results. Lastly, Section 5 presents a brief conclusion of the work and provides possible ideas that can be developed in future work.

[1] `http://www.flickr.com`

[2] `http://Geograph.org.uk`

[3] `github.com/Henrique97/ThesisFinalWork`

2 Fundamental Concepts and Related Work

The most simple neural network structure that is commonly used is the Multi-Layer Perceptron (MLP), composed by several connected layers of basic units, called perceptrons, each of them representing a processing level in the network's hierarchy. A perceptron is capable of generating an output from a given set of inputs by obtaining an intermediate value resulting from a linear combination of the inputs, and then passing it through a non-linear function that is usually referred to as an activation function:

$$p = \sum_{i=1}^{n} w_i \times x_i + b \tag{1}$$

In the previous equation, $w = \{w_1, ..., w_n\}$ is used for representing the weights associated with each input, and b represents a bias term. By increasingly adding hidden layers we can increase the processing power of the network. A MLP containing a large number of hidden layers is often referred to as a deep neural network. The network weights can be trained through a mechanism called backpropagation, which consists in the calculation of an error associated with the network's prediction and the propagation of this same error through the network, in order to tune each of the weights associated with each layer.

Convolutional Neural Networks (CNNs) are a type of neural network specialized in dealing with high-dimensional data (e.g., images). Although the principles behind this type of network are the same as those from standard MLPs, CNNs can provide dimensional filters that allows us to identify patterns in an image, while keeping the number of learnable parameters significantly low when compared to standard MLPs. To achieve this, CNNs employ the use of convolution operations to deal with the dimensionality problem. The convolutional operation makes use of filters, which are composed of trainable weights, and convolves them with the input. In this work a state-of-the-art CNN architecture, known as EfficientNet (Tan and Le, 2019), will be used to extract image features.

Recurrent Neural Networks (RNNs) are an alternative architecture for processing variable sized inputs, due to the way they generate an output by combining both the current input and the result from a previous state. This behaviour can be very valuable for certain cases, such as video processing, where it enables the network to adapt its number of layers according to the number of frames that need to be processed. Overall, this type of networks are usually used in order to try to and make a predicton based on the contents of a sequence, such as predicting the next frame of an image sequence. The results presented in this work make use of advanced recurrent layers such as Long Short-Term Memory (LSTM) units (Hochreiter and Schmidhuber, 1997), which are a type of RNN which is works based on a two stream approach, in which one stream is responsible for keeping long-term information while the other stream only weights the information extracted from the last inputs.

Some previous studies have explored the idea of using deep learning in order to develop models for building maps from ground-level images. In Zhu et al. (2019), the authors propose the use of a CNN architecture to map land usage for the city of San Francisco, resorting to large image sharing services, such as Flickr. The proposed CNN consists in a two stream architecture, where one stream is responsible for object recognition, with the other being used for scene recognition. In Srivastava et al. (2020) another approach was taken in the use of ground-level images for land use mapping, by using a set of multiple pictures and combining its results using an aggregator, which would weight the features of each picture in the final mapping results.

The proposed approach follows the results of previous studies and is based on the idea that the principle of using multiple images to characterize a region can be further expanded by exploring possible relations between images as a way to further improve the mapping results. For this purpose, this work makes use of recurrent network layers to process sequences of images.

3 The Proposed Approach to Land-Use Mapping

Only recently have ground-level photographs started to be used in order to provide an alternative automated solution for mapping tasks, which were usually achieved by recurring to methods based in manual annotations, and/or remote sensing through top-down images. Studies such that from Newsam and Leung (2019) have established a strong base on how leveraging available ground-level community shared photos could impact tasks such as land-use mapping. Srivastava et al. (2019) worked on expanding this idea and improved the current methods by combining multiple pictures in order to further improve the accuracy of mappings. However, none of these previous approaches has explored the possibility of a model that could look at a sequence of images that represents a region, instead of only analysing images individually. My proposal consists in organizing photographs belonging to the same region into sequences, and make use of a Recurrent Neural Network (RNN), together with a Convolutional Neural Network (CNN) model, in order to better leverage possible links between pictures that might help further improve the accuracy of the mappings.

3.1 Procedure

The considered step-by-step procedure associated to the proposed approach is summarized in Figure 1. In brief, we have that the first step consists in selecting a study region based on available land-use data, provided via hand-annotations. This is important

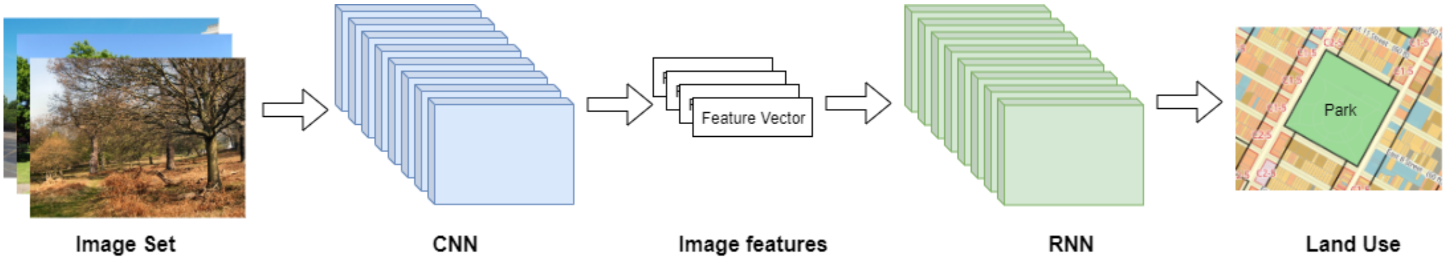

Figure 1: Illustration of the model and procedure used to process ground-level imagery and extract a land usage mapping

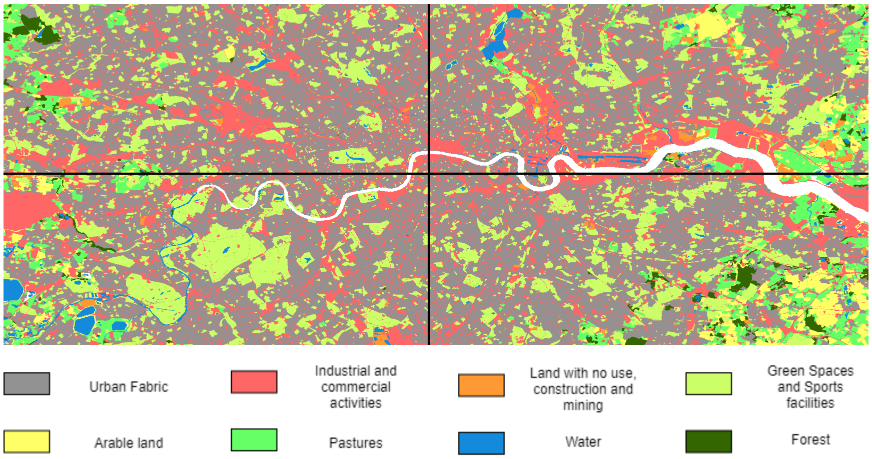
Urban Fabric
Industrial and commercial activities
Land with no use, construction and mining
Green Spaces and Sports facilities
Arable land
Pastures
Water
Forest

Figure 2: Sub-region division for the Land-use and Scenic Beauty tasks, colored according to the Land-Use classes.

in the context of collecting ground-truth data for supporting model training. The information is then aggregated into a meaningful set of land-use classes that can be used in order to train and test the envisioned model. The collected map, which is originally available in vector format, is then transformed into a raster based on a defined resolution ($25m \times 25m$). When a pixel with conflicts between different mapping classes is found, the area is mapped according to the majority class. Due to the complexity of land-use mapping, some of the originally available classes may be aggregated in order to create new and more meaningful superclasses. One simple example is the aggregation of classes representing different residential fabric densities into a residential super-class. An example for the resulting map, considering a region corresponding to the surroundings of London, UK, and original data from the Urban Atlas 2012[4], is presented in Figure 2.

With a region selected, the next step consists in collecting ground-level photographs which provide geospatial information about the place they were taken, and that are within the study region. The geospatatial data can then be used to overlay the photographs' locations with the raster created in the previous step, in order to attribute a class to each picture, according to the area it was taken in. In our tests, we collected photos from Geograph.uk.

In the context of tests assessing the performance of the method, for cross-validation purposes, the study region can then be divided into four sub-regions, as observed in Figure 2. This allows us to train models using the photos from 3 sub-regions, afterwards assessing the ability of the model to generalize to a forth region. The photographs are used to train a CNN in a simple classification task, in which the target classes are the ones extracted from the raster. The basic network architectures used in this work are the DenseNet (Huang et al., 2016) and EfficientNet (Tan and Le, 2019) models.

After training the CNNs, for each of the 25x25m raster cells, we produce a sequence of photographs in descending order according to the distance to the centre of the cell. For this ordering and based on the available coordinates for each picture, we can make use of a metric such as the haversine distance to calculate which photographs are closer to the centre of each square and can better depict its content:

$$D(x,y) = 2 \arcsin \sqrt{\sin^2(\frac{x_1 - y_1}{2}) + \cos x_1 \cos y_1 \sin^2(\frac{x_2 - y_2}{2})} \tag{2}$$

In the previous equation, x and y represent the locations between which we calculate the distance, and x_1/x_2 and y_1/y_2 are the latitude and longitude of each of those points, respectively.

The sequences of images can then be used to train a recurrent neural network model in a classification task in which the input are the sequences of images and the target classes are the land use classes attributed to each cell based on the data from

[4] http://land.copernicus.eu/local/urban-atlas/urban-atlas-2012

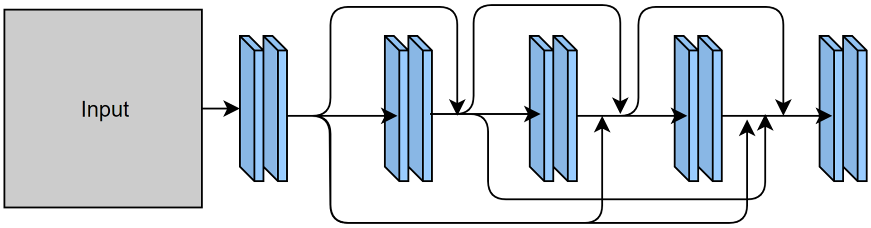

Figure 3: DenseNet Block. The arrows represent the flow of feature vector maps between convolution layers layers.

the raster. Part of the complete model (i.e., the CNN part in the CNN RNN combination) is initiated with the weights obtained in the image pre-training phase. These weights are locked in order to reduce memory usage and allow the tuning of the remaining weights by recurring to bigger sequences. I will also model similar to the one used in Srivastava et al. (2020) to evaluate the results of combining the features of different images via an aggregator which makes use of average pooling.

In our tests, for each of the cross-validation steps, we can then use the sub-region left out during training to evaluate the model's performance and create a new raster for this area based on the extracted land-use classes, in order to create a visual representation of the results that is easily comparable to the original raster.

Besides land cover mapping, a similar procedure can be used to approach the task of scenic beauty mapping. In this case, the first step consists in collecting photographs within the study region and attribute a score to each one of them. In our tests we used the same study region as for land cover mapping but, unlike the in the previous task, the scenic scores are obtained based on evaluations for each individual photograph, instead of having to be attributed based on a ground-truth raster map. We can then train a CNN on a regression objective in order to predict the scenicness of each image.

Using the inverse procedure to the land-use task, we can attribute a scenic score to a map cell based on the averaging of the scenic scores of the pictures contained within its region. Using sequences of images extracted for land-use mapping, we can train a RNN model in the same linear regression problem, in order to generate a scenic beauty map of the study region. In this case, instead of considering all raster cells for training, as in the case of land cover mapping, we use only the cells containing photos with a ground-truth scenicness score.

3.2 Neural Network Architectures for Image Analysis

For the image analysis this work makes use of two deep CNN architectures, namely the DenseNet and EfficientNet models.

The DenseNet model (Huang et al., 2016) introduced the idea of dense shortcut connections. Shortcut connections had already been used in several previous models, such as the ResNet model (He et al., 2015), and enabled the network layers to produce an output combining both the pre-processed and post-processed data, which proved to result in a more stable learning. The main innovation in the DenseNet model was the fact that each layer was able to receive the output from all the preceding layers, as it is shown in Figure 3. This enables the learning process to tackle the loss of information in deeper neural models, while allowing a better propagation of the error during the training phase, by avoiding the vanishing gradient problem. Regarding the merging of the outputs from previous layers, unlike previous models where the pre-processed data was added to the output, in the DenseNet model all the outputs are concatenated, resulting in a clean representation of the results from all the preceding layers.

More recently, the EfficientNet architecture was introduced along with a new scaling method, named compound scaling, with the main objective of optimizing the scaling of networks.The principle behind this idea is that given a certain value for available memory and available computing power (FLOPS), the objective is to maximize the accuracy obtained by the model by adjusting the network's depth (d), width (w) and the resolution of the input (r). Due to the fact that there is a dependency between all the variables, it becomes difficult to find the optimal solution to this problem, which constitutes the reason why most of the networks used to be extended in only one dimension. The compound method introduces a new procedure to obtain the optimal values for the variables, solving the following expression where α, β, γ are fixed hyperparameters and ϕ establishes the constant growth for each dimension.

$$\begin{aligned}
&d = \alpha^{\phi}, w = \beta^{\phi}, r = \gamma^{\phi} \\
&s.t.: \\
&\alpha.\beta^2 - \gamma^2 \approx \\
&\alpha \geq 1, \beta \geq 1, \gamma \geq 1
\end{aligned}$$

To begin with, a base network architecture was built to serve as a baseline model. Next, the authors fixed ϕ=1 and performed a grid search for the optimal values of the hyperparameters α, β, γ. Then, the effectiveness of this method was tested by scaling the network according to ϕ.

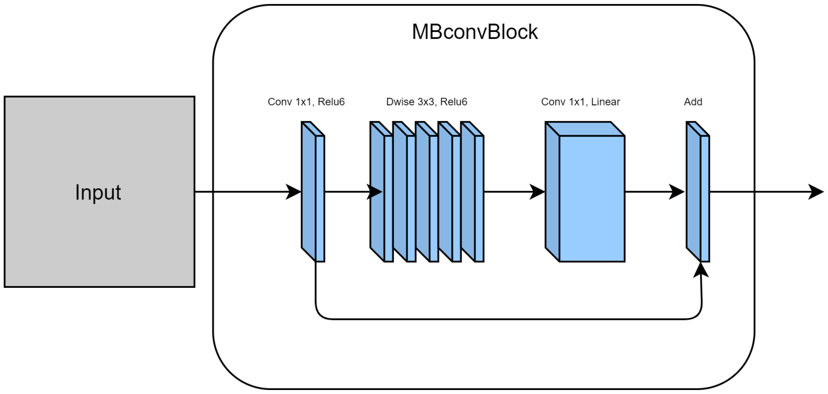

Figure 4: MBconv Block. The arrows represent the flow of feature vector maps between layers.

The base network architecture, named EfficientNet-B0, is mainly constituted by Mobile Inverted Bottleneck convolutional blocks (MBConvs), which are presented in Figure 4. The MBConv blocks work around the principle of depthwise separable convolutions. Unlike normal convolutions, in which we begin by applying each filter directly to all input channels, and then combine the results of all channels, in depthwise separable convolutions we start by applying one single filter to each input channel and only then combine the outputs using a 1×1 filter. This results in a great reduction of parameters, and a reduction in the number of multiplication operations in the order of 8 to 9 times the number of multiplications executed in standard convolutions.

To further increase the performance of MBConv blocks, the authors also make use of both inverted residuals and linear bottlenecks, which are based on the premise that a low dimensional subspace can capture the information from feature maps, and that non-linear activations, such as the ReLU, lead to information loss.

In order to improve the gradient flow to the initial layers, the blocks use residual connections such as the ones found in the DenseNet. However, unlike previous networks where the residual connections are used between layers with a high number of channels, these connections are made between narrow layers, which results in a decrease in memory and computational cost. This technique works based on the aforementioned premise that the valuable information is still contained in the bottlenecks, allowing us to connect these type of layers.

Taking into account the loss of information that results from the use of non-linear activations, and noting the fact that MBConv blocks use residual connections between narrow layers, it is expected that these blocks suffer from a considerable reduction in accuracy. To tackle this issue the authors proposed the use of a convolution with a linear activation before merging the output with the initial activations. This allows to tackle the loss of information generated by non-linear function, without the need to increase the number of channels in the narrow layers that contain the residual connections.

3.3 Tools and Data

The entire procedure was implemented in the Python[5] programming language, as there already many packages that facilitate the collection of data and the implementation of the desired model. Naming some of the most important tools, this work has made extensive use of the GDAL[6] package, which allowed me to translate and edit vector and raster geospatial data formats, and the Tensorflow [7] and Keras [8] packages, which facilitated the creation and training of deep neural models based on the proposed approach. Some of the more recent models and methods were also tested by recurring to specific packages, such as an EfficientNet Tensorflow implemention[9] based on the original paper (Tan and Le, 2019).

To train the model for the identification of land use classes, the first step consisted in collecting a large number of photographs. For this purpose, I have made use of the *Geograph.uk dataset* as a data source, as it provides low noise community-shared photographs of the scenery for the study region.

For the land-use task, I have also built a ground-truth map based on data collected from the *Urban Atlas 2012*, which provided land-use data for the city of London based on hand-annotations collected for the *Copernicus Programme*[10]. Finally, for scenic-beauty mapping, the scenicness scores were obtained based on the game Scenic-Or-Not[11], which provides scores for photographs available in the *Geograph.uk* dataset.

[5] http://www.python.org/

[6] http://gdal.org/

[7] http://www.tensorflow.org/

[8] http://keras.io/

[9] github.com/qubvel/efficientnet

[10] http://land.copernicus.eu/

[11] http://scenicornot.datasciencelab.co.uk

3.4 Methodology Details

The CNN architectures used in this work are based on the ones presented in the original papers. The weights are initiated according to the results obtained in the ImageNet challenge, as both of the studied tasks relate to the classification of ground-level photos and they can leverage this previously obtained knowledge to speed up the training phase. The final layers of the CNN networks are replaced by either a dense layer with a softmax activation function, so that the output can be representative of 1 of the 8 land-use classes, or a dense layer with a output, so that the result can be representative of a scenic-beauty score. The cross-entropy and mean-squared error loss functions are used for the tasks of land-use and scenic beauty mapping, respectively.

Besides the basic CNN architectures I also evaluate the performance of the following methods to help in model training: the auto-augment (Cubuk et al., 2019) augmentation technique; the AugMix (Hendrycks et al., 2019) augmentation technique; the supervised contrastive learning method (Khosla et al., 2020).

The auto-augment method consists in a data augmentation technique which aims to automatically find the optimal augmentation policies for a certain dataset. The result of this method consists in a set of sub-policies which are randomly applied to data during the training phase. The authors of this paper applied this method to several datasets, including the ImageNet dataset, and were able to improve the state-of-the-art accuracy and the transfer learning capability amognst different datasets. Taking this into consideration, I have made use of the list of sub-policies applied by this method in the ImageNet dataset, which include a mixture of the following augmentations: automatic addition of contrast; histogram equalization; posterization; solarization; shearing; translation; sharpness change; brightness change; color changes; image inversion.

The AugMix data augmentation method combines ideas from several previous approaches to increase the robustness of a network, allowing it to better deal with unforeseen corruptions in the testing data. To achieve this, new training samples are formed by performing several augmentations of a raw image, which are made by applying a chain of transformations to it, such as a translation or rotation, and then mixing them together, along with the original image, using elementwise convex combinations. Furthermore, the authors of this method propose the use of a Jensen-Shannon Divergence consistency loss during training, in combination with the original supervised training loss, to help the network obtain similar results for the augmented and original samples. For this work, this technique was applied by considering the following augmentations: automatic addition of contrast; histogram equalization; posterization; solarization; shearing; translation.

The contrastive learning technique consists in a two step training approach. In the first step, the top classification layer of the CNN model is removed, leaving the set of features as output of the model. The model is then trained with the intent that pictures from the same class are close in the feature space and that pictures from different classes are as far away as possible. For this purpose, a max margin contrastive loss (Hadsell et al., 2006) is used. This loss function essentially equates the euclidean distance between feature vectors, in order to generate clusters in the feature space that should aggregate the photographs from each class. The second step consists in adding the classification layer back and make use of the pre-trained weights to extract classification results based on the aggregated feature sets.

Following the two step training approach, the techniques mentioned above are applied during the CNN pre-training phase. For the sequence analysis, I only make use of non-augmented image sequences and initialize the CNN weights according to the results of the best performing method.

4 Experimental Evaluation

This work focused on land-use and scenic beauty mapping for the urban area of the city of London and its surroundings. The testing and evaluation process was split into 5 steps as presented bellow:

1. Train and test a CNN model in a classification task for land-use mapping by recurring to community-shared photograps from Geograph.uk and ground-truth land-use data collected from the Urban Atlas for 2012.

2. Train and test a CNN model in a regression task for scenic-beauty mapping by recurring to community-shared photograps from Geograph.uk and ground-truth data collected from the game Scenic-or-Not.

3. Use the tuned weights from Step 1 in order to train a RNN architecture in a classification task for land-use mapping by recurring to community-shared photograps from Geograph.uk and ground-truth land-use data collected from the Urban Atlas for 2012. The study region is divided into cells and for one of them, a sequence of ten photographs in ascending order based on distance is generated.

4. Use the tuned weights from Step 2 in order to train a RNN architecture in a regression task for scenic beauty mapping, by recurring to community-shared photographs from Geograph.uk and produce a ground-truth scenic beauty map based on the scores of individual photographs collected from the Scenic-or-Not game. The study region is divided in a similar way to the previous step, but instead of attributing a class to each cell, we will only consider the cells that have photographs with scenic score inside their area. The average score is attributed as the ground-truth to the cell.

	Sub-region 1		Sub-region 2		Sub-region 3		Sub-region 4	
	Train	Test	Train	Test	Train	Test	Train	Test
Urban	93422	44138	130187	7373	116752	20808	133340	4220
Industrial and Commerce	104670	39176	126526	17320	124382	19464	140329	3517
No use, Construction and Mining	1688	516	2179	25	2027	177	2194	10
Green Spaces and Sports	38007	14727	51413	1321	40507	12227	51713	1021
Arable	11699	380	12079	0	11496	583	12079	0
Pastures	21974	867	22840	1	22081	760	22841	0
Water	7565	2262	9552	275	7512	2315	9810	17
Forest	13283	548	13831	0	13220	611	13831	0

Table 1: Support for each land-use classed based on the sub-region used for testing.

	DenseNet-169			EfficientNet-B0		
	Precision	Recall	F-score	Precision	Recall	F-score
Urban	0.578	0.680	**0.618**	0.590	0.623	0.604
Industrial and Commerce	0.608	0.535	0.565	0.596	0.588	**0.589**
No Use, Construction and Mining	0.0	0.0	0.0	0.057	0.010	**0.012**
Green Spaces and Sports	0.589	0.512	0.549	0.625	0.493	**0.552**
Arable	0.196	0.029	0.044	0.167	0.064	**0.090**
Pastures	0.209	0.305	0.231	0.191	0.331	**0.241**
Water	0.460	0.308	0.361	0.401	0.357	**0.373**
Forest	0.219	0.261	0.235	0.171	0.365	**0.235**
Macro Average	0.357	0.329	0.325	0.350	0.354	**0.337**
Accuracy			0.57			**0.58**

Table 2: Results for the simple architectures evaluated in the land-use task.

5. Produce a raster for the study region based on the obtained results for both the scenic-beauty and land-use tasks.

The task of land-use mapping presented in this work consists in a simple classification problem where the goal is to accurately extract a class based on the information extracted from a single image or set of images. For this purpose, a way to evaluate the performance of different models consists in the use of metrics such as recall, precision and F1-score.

For the task of scenic-beauty, instead of mapping a class, the objective is to extract a real value which represents the scenicness that can be observed in a certain picture. A possible way to evaluate the results is to estimate the difference between the obtained predictions and the original value. For this purpose, we can use various statistics, such as the Root Mean Square Error (RMSE) between the predicted and original values, or the Mean Absolute Error (MAE).

Due to memory usage constraints, this work has been divided in two segments in order to improve the end results of the experiments. Beginning by experimenting with individual image analysis allowed me to use more memory intensive procedures (e.g., larger batch sizes) and more easily test certain methods, such as data augmentation techniques. I begin by presenting the results for the following experiments with image analysis: use of a DenseNet169 architecture with batch size 32 for information extraction; use of an EfficientNet-B0 architecture with batch size 32 for information extraction; for the task of land-use mapping, use of an EfficientNet-B0 architecture with batch size 32 along with augmentations generated by the AutoAugment method; use of an EfficientNet-B0 architecture with batch size 16 together with the AugMix method; use of an EfficientNet-B0 architecture with batch size 32 along with a supervised contrastive learning technique as presented in Khosla et al. (2020).

To provide a more accurate evaluation of the tested models, I have divided the study region into 4 sub-regions to follow a 4-fold cross validation approach, which consists in feeding 1 of the sub-regions as the testing set and using the remaining as training set, repeating the procedure 4 times. As a result, the model behaviour can then be better estimated by averaging the evaluation metrics obtained for each of the folds. A map of the sub-regions, colored according to the classes for land-use classification is presented in Figure 2. In order to help and improve the results, some extra photographs were collected from an outer region, and added to the training set. These images were not used in the following image sequence due to the fact that generating and using the extra cells in this region would drastically increase the time that it takes to produce data and train a model. Table 1 presents the support for each sub-region divided by the 8 classes considered for this task.

The results obtained for the experiments are presented in Table 2 and Table 3, according to the metrics of accuracy, precision, recall and F1-score, and also according to the results obtained separately for each of the 8 classes.

As shown in Table 2, both the DenseNet model and EfficientNet Model performed almost similarly, as expected, based on their similar performance in the ImageNet challenge. The only major difference between these two architectures corresponds to a 34% decrease in the time taken for each training epoch for the EfficientNet Architecture.

Regarding the alternative techniques applied along the standard architecture, the AugMix augmentations provided the most increases in performance for some of the classes, which should be correlated with the fact that the original paper reports that this technique tends to allow to deal better with data corruption and improve the transfer of learning between training and test datasets which do not present the same data distribution. For the remaining evaluated methods, there was no noticeable improvements

	Auto-Aug			AugMix			Contrastive		
	Precision	Recall	F-score	Precision	Recall	F-score	Precision	Recall	F-score
Urban	0.585	0.644	0.608	0.760	0.841	**0.796**	0.588	0.633	0.608
Industrial and Commerce	0.600	0.559	0.576	0.572	0.636	**0.596**	0.597	0.575	0.579
No use, Construction and Mining	0.028	0.000	0.007	0.135	0.007	**0.007**	0.035	0.007	0.007
Green Spaces and Sports	0.630	0.500	**0.554**	0.638	0.488	0.551	0.630	0.481	0.548
Arable	0.170	0.046	0.066	0.166	0.058	**0.084**	0.148	0.046	0.066
Pastures	0.198	0.353	**0.252**	0.247	0.191	0.217	0.182	0.354	0.236
Water	0.417	0.371	**0.391**	0.441	0.246	0.313	0.424	0.354	0.383
Forest	0.195	0.320	**0.240**	0.186	0.324	0.236	0.181	0.318	0.235
Macro Average	0.353	0.349	0.337	0.393	0.349	**0.350**	0.348	0.346	0.333
Accuracy			0.570			0.57			**0.58**

Table 3: Results for the complementary methods and techniques evaluated in the land-use task.

	RMSE	MAE
DenseNet-201	0.90	0.70
EfficientNet-B0	0.89	**0.68**
EfficientNet-B0 with auto-augment	**0.88**	0.69

Table 4: Results for the complementary methods and techniques evaluated in the scenic-beauty task.

when using the auto-augment augmentations or the contrastive learning pre-training technique. Taking all into consideration, it was decided that the weights learned for the test with the EfficientNet-B0 architecture, along with AugMix augmentations were to be saved and used for the training and mapping phases of the remaining work.

For the task of scenic beauty, due to reduced number of pictures in the study region, I have opted to evaluate the models by considering, for testing purposes, all the photos within the dataset used for the previous task that have scenicness scores, and for training, the remaining photographs of the UK territory for which I was able to collect scenicness scores. Unlike in the previous task, the data for the sceniccess score is extracted directly from the train and test dataset without recurring to a secondary data source for this purpose which helps reducing the error associated with the ground-truth data. However, the scenicness of a photo is not a concrete measure but rather a subjective term that can influenced by a variety of factors, as explored in Seresinhe et al. (2018). By averaging a set of scores attributed by different individuals we can more accurately portrait the scenicness of a photograph, though it will still be a measure heavily influenced by the human expert's perspective of scenic-beauty. As referred before due to the desired output being a real value between 0 and 10, the classification layer of the tested models will be replaced by a linear regression layer.

The results obtained for the experiments are presented in Table 4 according to the metrics of RMSE and MAE. The models behave in a similar way to the previous task, though by looking at the differences between the basic EfficientNet-B0 and the Auto-Augment model it is possible to observe that this augmentation technique helps smoothing the predictions and reduce the RSME. This increase in performance is to be expected, as we initialized the CNN with the weights from obtained from the ImageNet challenge and, as the authors of the auto-augment technique report, applying this method usually helps the transfer of learning between different but similar task.

Regarding the creation of the photographic sequences for the land-use task, the raster map obtained from the *Urban Atlas* is divided into 25x25m cells, to which a class is assigned based on the land-use data. Next, for each cell, the photos are sorted according to the distance to its center, a sequence of the 10 closest images are attributed to it.

A major issue faced during this work was the fact that due to the fact that the Geograph.uk dataset is generated via voluntary contributions, the photographs are not equally distributed throughout the study region. As such, the quality of the produced sequences of images when it comes to the representing the contents of the cells varies significantly according to the availability of photographs. Initial tests suggested that there was a large decay in terms of the metrics used to measure the model's performance, mainly due to the fact that even the closest image to the cell's center was in an area that did not belong to the same class as the cell it was trying to represent. A possible solution to this could be reducing the granularity of the map, by increasing the cell size and, as a result, increase the chances of having a photograph within each cell. However, this would seriously reduce the support for some of the minority classes, as if we were presented with a bigger cell which, according to the Urban Atlas, could be identified as 20% a minority class and 80% a majority class, the cell would only be seen as belonging to the majority class. As such, the alternative found for this problem consisted in filtering the cells and excluding those for which there is not at least one photograph within 2 kilometers of the cell and inside a region with the same class as the target output. Although this method is applied during the training process, the final rasters and result will also consider all the available cells.

For sequence processing I evaluate the results of 4 different models: 1 model which uses an aggregator to combine multiple image features using an average pooling layer; 3 RNN architectures which are fed with 3,5 and 10 sequences of photographs.

The results for each of the tests are presented in Table 5. The evaluation process also uses the same cross-validation procedure that was used in the individual image analysis and the results represent the average values obtained for the 4 sub-regions.

	Aggregator			RNN 3 photos			RNN 5 photos			RNN 10 photos		
	Precision	Recall	F-score	Precision	Recall	F-score	Precision	Recall	F-score	Precision	Recall	F-score
Urban	0.769	0.888	0.826	0.784	0.877	0.826	0.789	0.884	**0.830**	0.792	0.863	0.827
Industrial and Commerce	0.666	0.489	0.563	0.635	0.519	0.572	0.659	0.513	**0.578**	0.597	0.553	0.573
No use, Construction and Mining	0.000	0.000	0.000	0.008	0.004	0.004	0.072	0.004	**0.018**	0.061	0.008	0.012
Green Spaces and Sports	0.682	0.667	0.669	0.665	0.674	0.674	0.699	0.659	0.677	0.719	0.665	**0.690**
Arable	0.314	0.056	0.096	0.146	0.100	0.120	0.174	0.131	**0.149**	0.178	0.087	0.110
Pastures	0.493	0.495	**0.486**	0.513	0.400	0.446	0.494	0.454	0.471	0.516	0.446	0.481
Water	0.578	0.463	0.513	0.583	0.513	0.544	0.530	0.556	**0.547**	0.549	0.524	0.532
Forest	0.485	0.390	0.429	0.500	0.277	0.353	0.474	0.469	0.456	0.520	0.420	**0.458**
Macro Average	0.498	0.431	0.448	0.479	0.420	0.442	0.486	0.459	**0.466**	0.491	0.446	0.460
Accuracy			0.714			0.713			**0.718**			0.715

Table 5: Land-use task results using photographic sequences

	RMSE	MAE
Aggregator	1.55	1.15
RNN 3 photos	1.13	0.90
RNN 5 photos	1.11	0.88
RNN 10 photos	**1.10**	**0.87**

Table 6: Scenic-Beauty task results using photographic sequences

As it can be observed in the presented results, the use of a sequence of images resulted in improved results when compared to the use of a single image. When increasing the size of the sequence, it is possible to notice that the model provides increased accuracy, especially when looking at smaller sequences (between 3 and 5 pictures). However, it is also possible to verify that this improvement associated with the increase in size of the available information starts to fade away when we get to longer sequences (10 images), in which can notice small decreases and increases in the results from different classes, based on the considered metrics. The most likely reason for this behaviour is that due to the fact that we make use of a not so vast dataset of pictures.The quality of the sequences will also suffer with the increase in the number of pictures used for each cell, as the most distant pictures will often be far way from the cell's area and will not provide any meaningful grounds to what should be the class attributed to the image set. Comparing the RNN with the use of average pooling in the model with the aggregator, the improvements in the results obtained by analysing sequences with a RNN are mainly connected to better predictions in the regions with lower density of photographs, while still maintaining the same accuracy for more densely packed regions.

For the task of scenic beauty, there was no map available that could provide scenic scores based on the region, and the ground-truth data had to be collected based on the results obtained for individual photographs. As such, in a similar way to what happened in the previous task, only some cells were selected for the evaluation process, based on the existence of a classified photo within its region. Running the same recurrent neural network models provided the results presented in Table 6.

Looking at the results, these present similarities to the ones obtained for the previous task. However, the difference between the RNN architecture and the Aggregator is more prominent than in the previous task. This is most likely related to the fact that the number of samples for this task is lower than the ones used for land-use mapping, which also affects the image density in the studied areas, which will have a bigger effect in the Aggregator, due to the use of a Average Pooling layer to combine the features from different images.

In Figure 5 I present the result of the automated land-use mapping for 2 sub-regions of the sub-regions along with the ground-truth map extracted from the Urban Atlas 2012. For the task of scenic beauty, we can also make use of the trained model, along with the sequences generated for the land-use task and produce a scenic beauty map, as presented in Figure 6. Altough, it is important to consider that there's certainly a lack of accuracy associated with these predictions, due to the subjectiveness of the target value and due to the lack of ground-truth data, the maps allow us to observe that there is an apparent correlation between the land-use and scenic beauty of a place. For example, taking a look at the top left corner of the map, we observe higher levels of scenic beauty, which is expected for a region which the most prominent class are nature related (green-urban areas; forest; pastures). On the contrary, regions closer to the city centre with a concentration of urban fabric and commercial and industrial areas are portrayed as a region with low scenic scores.

5 Conclusions and Future Work

In this work, I presented the results of a new approach for terrain mapping based on the analysis of sequences of ground-level images, using convolutional neural networks, recurrent neural networks, photos obtained via the Geograph.uk initiative, and land-use and scenic-beauty data extracted from the Urban Atlas 2012 and Scenic-Or-Not game, respectively. The envisioned procedure was tested in the urban area of the city of London, and allowed the creation of rasters, which portrait the region according to 8 land-use classes (Urban Fabric; Industrial and commercial activities; Land without use, construction sites and mining facilities; Green Spaces and Sports facilities; Arable land; Pastures; Water; Forest) and according to a scenic-beauty scale from 0 to 10.

Regarding future work, a possible way of expanding the presented method would be trying to increase the used set of images,

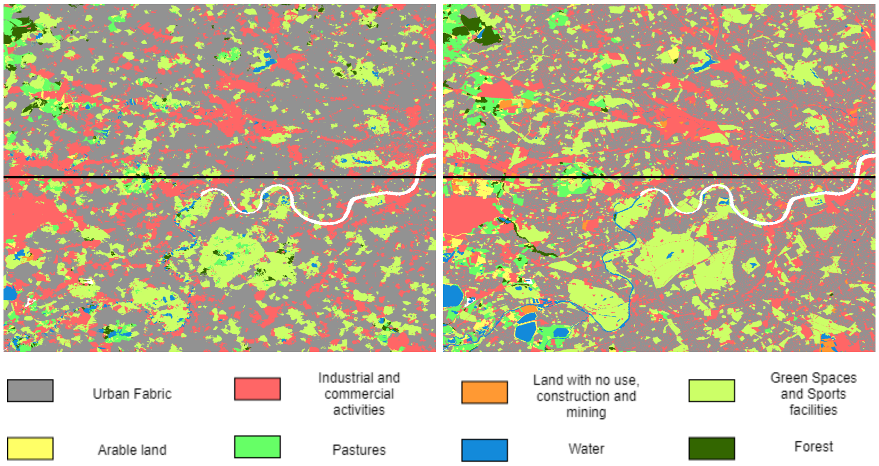
Urban Fabric
Industrial and commercial activities
Land with no use, construction and mining
Green Spaces and Sports facilities
Arable land
Pastures
Water
Forest

Figure 5: On the left, part of the raster automatically generated based on the obtained predictions for land-use, using the sequence analysis approach. On the right, part of the raster created based on the land-use data obtained via the Urban Atlas 2012 data.

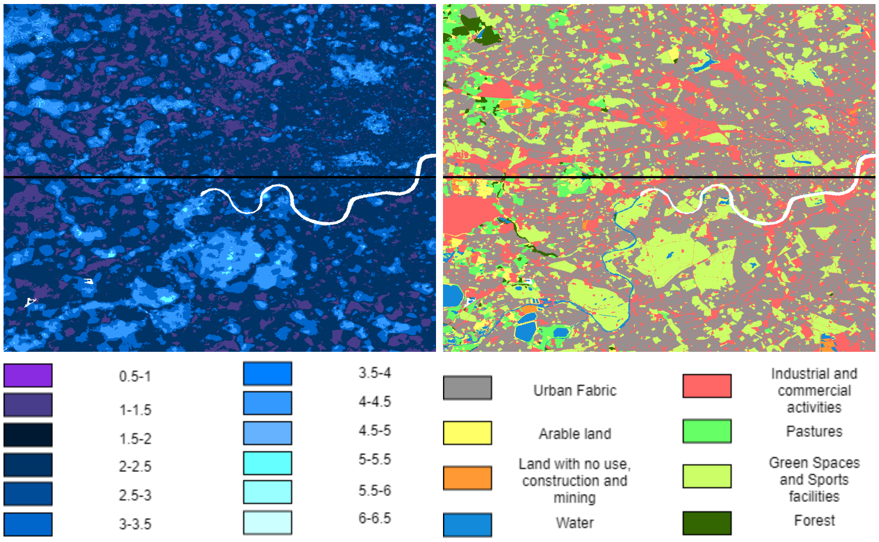
0.5-1
1-1.5
1.5-2
2-2.5
2.5-3
3-3.5
3.5-4
4-4.5
4.5-5
5-5.5
5.5-6
6-6.5
Urban Fabric
Industrial and commercial activities
Arable land
Pastures
Land with no use, construction and mining
Green Spaces and Sports facilities
Water
Forest

Figure 6: On the left, part of the raster automatically generated based on the obtained predictions for scenic beauty, using the sequence analysis approach. On the right, part of the raster created based on the land-use data obtained via the Urban Atlas 2012 data. It is possible to observe a correlation between the scenic beauty and use of a place.

in order to obtain bigger and more accurate sequences that could help improve the represention of each mapping tile. This could be done by recurring to Google Street View API[12] as explored in Srivastava et al. (2019). Another possibility consists in using other augmentation methods, such as the one presented in Yang and Soatto (2020), which consists in a Fourier Domain Adaptation method, which allows the augmentation of images, using the inverse Fourier Transform from pairs of images with different lighting conditions.

Given that the density of photographs for each area lead to differences in the results that were verified when processing sequences of images, a possible alternative for sequence processing that could lead to improvements would be to replace the cell by cell analysis with the combination of multiple sequences of pictures from neighbouring cells, through a convolutional approach on top of the RNN results. This way, the mapping for each cell could weigh in the results obtained for its neighbours which would reduce the chance of errors associated with incorrect predictions for a single cells' images.

Finally, another idea that would be interesting to explore related to the use of digital elevation maps (DEMs), such as the *EU-DEM*[13] in order calculate the viewshed from the position of each photograph, based on the height of the place where they were taken. This would allow to tune the sequences of images that represent a cell not only based on the distance to the center, but also considering which photographs better capture the surrounding area.

[12] http://developers.google.com/maps/documentation/streetview/overview
[13] https://land.copernicus.eu/imagery-in-situ/eu-dem/eu-dem-v1.1

References

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. (2019). Autoaugment: Learning augmentation policies from data.

Hadsell, R., Chopra, S., and LeCun, Y. (2006). Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, abs/1512.03385.

Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., and Lakshminarayanan, B. (2019). Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9:1735–80.

Huang, G., Liu, Z., and Weinberger, K. Q. (2016). Densely connected convolutional networks. *CoRR*, abs/1608.06993.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. (2020). Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*.

Newsam, S. and Leung, D. (2019). Georeferenced social multimedia as volunteered geographic information. In *CyberGIS for Geospatial Discovery and Innovation*, pages 225–246. Springer.

Seresinhe, C. I., Moat, H. S., and Preis, T. (2018). Quantifying scenic areas using crowdsourced data. *Environment and Planning B: Urban Analytics and City Science*, 45(3):567–582.

Srivastava, S., Vargas Munoz, J. E., Lobry, S., and Tuia, D. (2020). Fine-grained landuse characterization using ground-based pictures: a deep learning solution based on globally available data. *International Journal of Geographical Information Science*, 34(6):1117–1136.

Srivastava, S., Vargas-Muñoz, J. E., and Tuia, D. (2019). Understanding urban landuse from the above and ground perspectives: A deep learning, multimodal solution. *Remote sensing of environment*, 228:129–143.

Tan, M. and Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946.

Yang, Y. and Soatto, S. (2020). Fda: Fourier domain adaptation for semantic segmentation.

Zhu, Y., Deng, X., and Newsam, S. (2019). Fine-grained land use classification at the city scale using ground-level images. *IEEE Transactions on Multimedia*, 21(7):1825–1838.