DETECTING LANDFILLS USING MULTI-SPECTRAL SATELLITE IMAGES AND DEEP LEARNING METHODS

Anupama Rajkumar
Continental Automotive Kft
Budapest, 1092, Hungary
rajkumar.anupama@continental-corporation.com

Tamas Sziranyi & Andras Majdik
Machine Perception Research Laboratory
Institute for Computer Science & Control (SZTAKI)
Budapest, 1111, Hungary
{sziranyi.tamas,majdik}@sztaki.hu

ABSTRACT

The cropping up of illegal landfills across the world and more so in developing countries has become a prevalent malice. It has become essential to come up with innovative solutions to this challenge, one that goes beyond detecting these spots manually. In recent times, remote sensing along with deep learning and computer vision has been applied to solve many problems. Through this paper, we would like to present a very high resolution Landfill dataset created from satellite images and demonstrate that by applying suitable deep learning methods, landfills can be detected even with a constrained and limited dataset. Through this paper we also make the Landfill dataset available to the research community.

1 INTRODUCTION

Landfills are sites designated for dumping rubbish, garbage or other sorts of solid wastes. Landfills are usually designated areas by municipalities where garbage is dumped and methods are in place to control the toxicity, emitted gases etc. in the vicinity. However, over time, multiple spots of illegal garbage dumps have cropped up all across the world, more so in the developing countries. These unchecked and illegal landfills don't have adequate safeguards and this causes considerable harm to the environment by contaminating the soil and water which can eventually pose serious health hazard to the inhabitants of the neighboring area which includes both humans and wildlife.

Manually detecting these landfills is a slow, expensive and inaccurate process. Through this paper, we propose the possibility of using satellite imagery along with the application of modern deep learning methods to detect landfills using semantic segmentation.

Through the rest of the paper, we describe the process for collecting the satellite images and processing them to create the landfill dataset. Further, we explain the models on which this dataset was applied for training and the experimental setup. We demonstrate the prediction results and the performance metrics obtained and finally present our conclusion and future scope for this work.

2 DATA COLLECTION

Availability of good quality data is essential to solve any machine learning problem. There are multiple datasets derived from satellite images like UC Merced land use dataset (Newsam & Yang, 2010), DeepSat (Basu et al., 2015), Urban Atlas and BigEarthNet (Charfuelan et al., 2019) that are used extensively for remote sensing applications. However, these datasets mostly cover general land use categories and are hence, not useful for training our models to detect landfills. Hence, we decided to create our own. To this end, as a first step, we identified a few major landfills across Asia, Europe and South America. Since the size of a landfill can vary from a few meters to miles, we

Figure 1: Panchromatic, Multi-spectral and Pan-sharpened Images of landfill in Vinca, Serbia

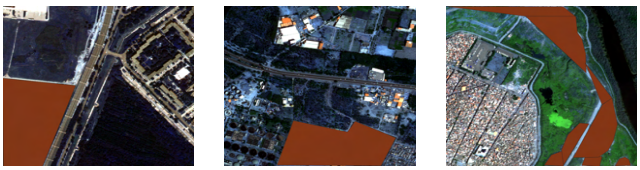

Figure 2: Landfills with masks in Ercsi in Hungary, Ahmedabad and Mumbai in India respectively

decided to use very high resolution multi-spectral satellite images from WorldView-3, WorldView-2 and GeoEye-1 satellite missions which provides high spatial and spectral resolution. We obtained images of 13 major landfills from these satellites which comprised our dataset.

3 Data Preparation

3.1 Image Processing Step

WorldView and GeoEye satellite images are available in 2 forms - a grayscale panchromatic image with very high spatial resolution (upto 30 cms) but low spectral resolution and a raterized multi-spectral image with high spectral resolution (upto 8 bands) but low spatial resolution. We perform the image processing technique of pan-sharpening the images so as to obtain resulting images with high spatial and spectral resolution as shown in Figure 1. We use these pan-sharpened images for the Landfill dataset.

3.2 Labelling Methodology

In order to create accurate labels for landfill images, in addition to visual inspection of the known landfill locations, we make use of vegetation indices like NDVI which tells about the strength of vegetation in the area within and surrounding the landfill. Along with vegetation indices, we calculated spectral signatures of the images which allowed us to create accurate semantic segmentation masks for the landfill images as shown in Figure 2.

3.2.1 Dataset Specifications

Since the original landfill images were very large, we split them into several non-overlapping patches of 512x512 pixels. After splitting these images, we obtained 3241 image patches in all. However, of these only 242 patches contained pixels with landfill information. This made the dataset highly imbalanced. In fact, even within the 242 patches, landfill information is scarce. Hence, we discarded the patches with no landfill information and create a final dataset with the 242 patches containing landfill information.

4 Experiments

4.1 Proposed Deep Learning Models

In order to detect landfills from the landfill dataset proposed in Sections 2 and 3, we apply semantic segmentation models. The deep learning models that we propose are based on Fully Convolutional Network (FCN) and U-Net proposed by Darrel et al. (2015) and Ronneberger et al. (2015) respectively.

We create variations of these architectures by adding two popular feature extraction models namely VGG16 (Simonyan & Zisserman, 2015) and ResNet34 (He et al., 2015). As a result, our experiments were performed on 4 deep learning architectures namely : FCN-VGG16, FCN-ResNet34, UNet-VGG16, UNet-ResNet34 as shown in Figure 3, 4, 5 and 6 respectively.

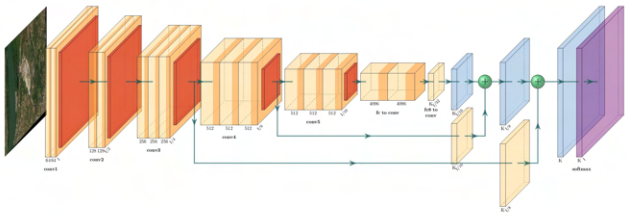

Figure 3: Overview of FCN-VGG16 architecture.

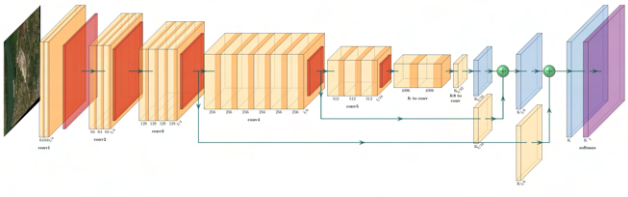

Figure 4: Overview of FCN-ResNet34 architecture.

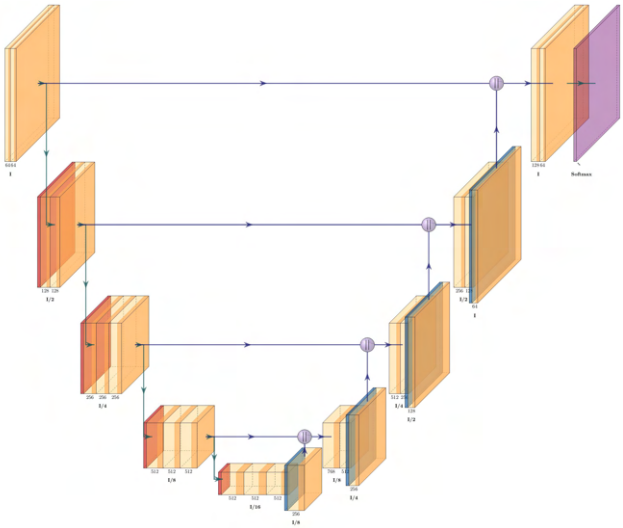

Figure 5: Overview of UNet-VGG16 architecture.

4.2 Experimental Setup

In order to train the models, the pan-sharpened Landfill dataset was split into 80% training and 10% validation set in a stratified manner. 10% of images were earmarked as test set on which the performance of the models were evaluated.

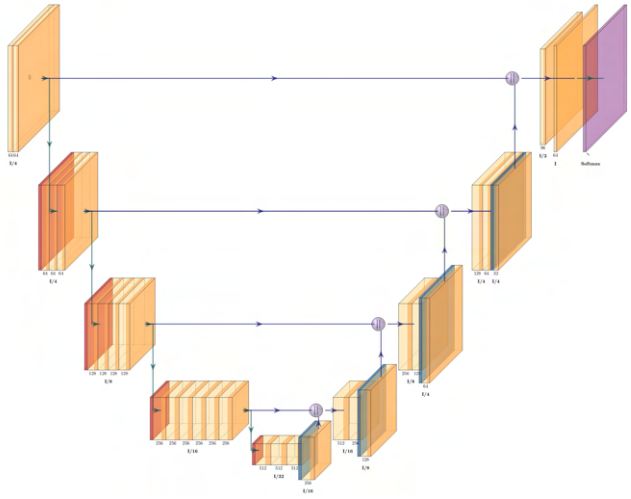

Figure 6: Overview of UNet-ResNet34 architecture.

4.2.1 PRE-TRAINING THE MODELS

Our proposed models have a large number of trainable parameters whereas the size of the Landfill dataset is extremely small. This makes it impractical to train the models end-to-end and from scratch. To this end, we use pre-trained models. For our experiments, we use classifications models that were pre-trained on ImageNet (Deng et al., 2009) and SpaceNet (Bacastow et al., 2018) datasets.

Our experiments rely heavily on transfer learning and while training, we found that freezing the higher layers and re-training and fine-tuning the parameters of the deeper layers of the network led to best results.

4.2.2 IMPLEMENTATION DETAILS

All the models were implemented using the PyTorch framework on Google Colab Pro. Nvidia Tesla P100 GPU was used for training and testing.

In addition to fine-tuning of hyperparameters, early-stopping was used to prevent the models from overfitting. The values of tuned hyperparameters along with the number of trainable parameters are listed in the table in Appendix A.1.

4.3 RESULTS

The performance of models is calculated using the metrics that are typical for a semantic segmentation problem, namely, pixel accuracy, precision, recall, specificity and mean intersection over union (mIoU). The definition of these metrics have been provided in A.2. The performance metrics of each model has been listed in Table 1 with the best performance marked in bold. These values have been listed for models pre-trained on SpaceNet which performs better than the models pre-trained on ImageNet.The corresponding loss-accuracy plots can be found in the Appendix A.3.

Table 1: Performance Metrics

Model	Pixel Accuracy	Precision	Recall	Specificity	mIoU
FCN-VGG16	0.762	0.602	0.421	0.843	0.330
UNet-VGG16	0.786	0.439	**0.550**	0.834	0.332
FCN-ResNet34	0.803	0.642	0.549	0.838	**0.401**
UNet-ResNet34	**0.830**	**0.645**	0.480	**0.919**	0.396

Figure 7 depicts the prediction results of a few landfill patches from the test set for each model.

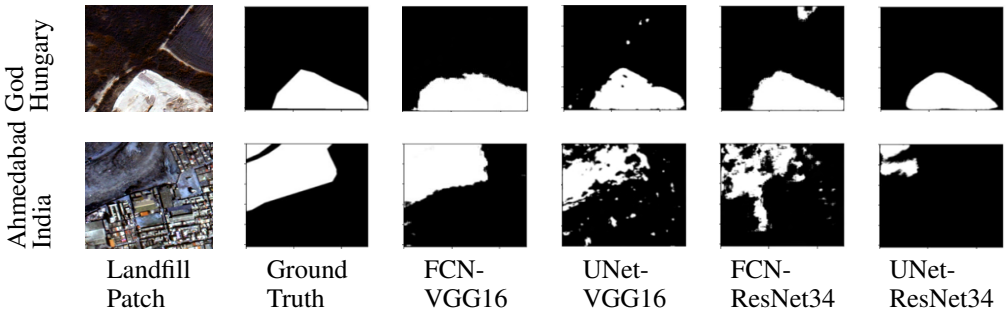

Figure 7: Prediction results

The models trained on the very high resolution Landfill dataset can be used to predict landfills from lower resolution images effectively as shown in Figure 8 where lower resolution, non pan-sharpened multi-spectral unseen patches from WorldView and GeoEye satellite images were used.

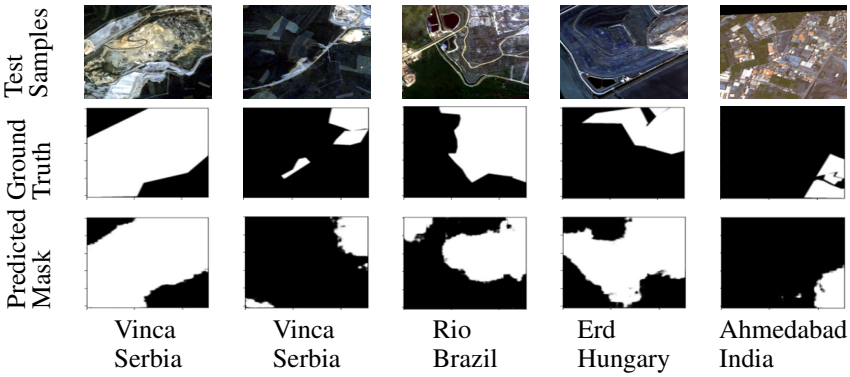

Figure 8: Transferability on low-resolution images

5 Conclusions and Discussion

Through the results that we obtained from our experiments, it can be shown that with remote sensing and appropriate deep learning methods, even a small dataset can be used to effectively detect landfills with a considerable accuracy. A very useful finding of our experiment was that our models trained on very high resolution images predicted the landfills very well even with images having low spatial resolution. This transferability proves useful since most of the satellite images available have low spatial resolution. However, the images and labels in our dataset does not cater to a specific category of waste and rather detects waste in general. While this is suitable to detect landfills that crop up illegally and contains all types of waste, it would be useful to have datasets that help identify specific types of waste.

5.1 Future Scope

Thanks to the rise in satellite missions, research community has become privy to a large repository of images for earth observation. Remote sensing images can be used equally well to detect and study the pollution in water bodies like rivers and lakes which is a very big problem in countries like India or to monitor deforestation or illegal mining. This could usher a new era where earth observation along with deep learning can help in aiding the efforts towards environmental awareness.

ACKNOWLEDGMENTS

We would like to thank European Space Agency (ESA) for making the requisite very high resolution multi-spectral satellite images from WorldView and GeoEye satellite missions available for research purposes. In addition, we would like to thank the Ministry of Innovation and Technology along with the National Research, Development and Innovation (NRDI) Office within the framework of the Artificial Intelligence National Laboratory Program for supporting this research.

REFERENCES

Todd M. Bacastow, Dave Lindenbaum, and Adam Van Etten. Spacenet: A remote sensing dataset and challenge series. *CoRR*, abs/1807.01232, 2018.

Saikat Basu, Robert DiBiano, Sangram Ganguly, Manohar Karki, Supratik Mukhopadhyay, and Ramakrishna Nemani. Deepsat - a learning framework for satellite imagery, 2015.

Marcela Charfuelan, Begüm Demir, Volker Markl, and Gencer Sumbul. Bigearthnet: A large-scale benchmark archive for remote sensing image understanding. pp. 5901–5904, 2019.

Trevor Darrel, Jonathan Long, and Evan Shelhame. Fully convolutional networks for semantic segmentation. *Neural Computation*, pp. 1527–1554, 2015.

Jia Deng, Wei Dong, Fei-Fei Li, Li-Jia Li, Kai Li, and Richard Socher. Imagenet: A large-scale hierarchical image database. pp. 248–255, 2009.

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

Shawn Newsam and Yi Yang Yang. Bag-of-visual-words and spatial extensions for land-use classification, 2010.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. 2015.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. 2015.

A APPENDIX

A.1 HYPERPARAMETERS

The values of tuned hyperparameters as listed in Table 2.

Table 2: Model hyperparameters

Hyperparameter	**FCN-VGG16**	**UNet-VGG16**	**FCN-ResNet34**	**UNet-ResNet34**
Batch size	5	5	5	5
Epochs	40	40	70	70
Learning rate	$1e^{-3}$	$1e^{-3}$	$2e^{-4}$	$2e^{-4}$
Loss function	BCE	BCE	BCE	BCE
Optimiser	Adam	Adam	Adam	Adam
Weight decay	$1e^{-5}$	$1e^{-5}$	$1e^{-5}$	$1e^{-5}$
Scheduler	StepLR	StepLR	StepLR	StepLR
Gamma	0.5	0.5	0.5	0.5
Input dims	512x512x3	512x512x3	512x512x4	512x512x4
Bands used	R,G,B	R,G,B	R,G,B,NIR	R,G,B,NIR
Trainable parameters	18,646,749	37,466,626	22,893,090	30,008,770

A.2 PERFORMANCE METRICS

The performance metrics used in the project can be defined as:

Pixel Accuracy or PA:

$$PA = \frac{TP + TN}{TP + TN + FP + FN}. \tag{1}$$

Precision:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Recall/Sensitivity/True Positive Rate

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

Specificity/True Negative Rate

$$Specificity = \frac{TN}{TN + FP} \tag{4}$$

Mean Intersection over Union (mIoU)

$$IoU = \frac{TP}{TP + FN + FP} \tag{5}$$

where:

TP : True positive, correct positive prediction

FP : False positive, incorrect positive prediction

TN : True negative, correct negative prediction

FN : False negative, incorrect negative prediction

A.3 LOSS-ACCURACY PLOTS

Loss-accuracy plots can be found in Figure 9.

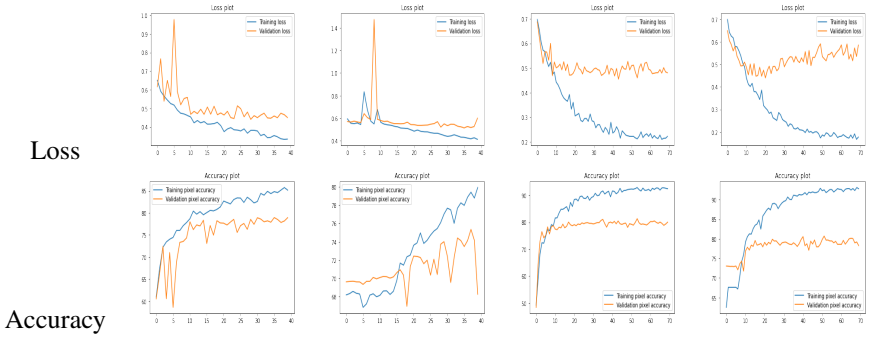

Figure 9: Loss-accuracy plots of models used