River segmentation for flood monitoring

Laura Lopez-Fuentes
University of the Balearic Islands, Spain
Autonomous University of Barcelona, Spain
AnsuR Technologies, Norway
Balearic Islands Health Research Institute, Spain
Email: l.lopez@uib.es

Claudio Rossi
Istituto Superiore Mario Boella, Italy
Email: rossi@ismb.it

Harald Skinnemoen
AnsuR Technologies, Norway
Email: harald@ansur.no

***Abstract*—Floods are major natural disasters which cause deaths and material damages every year. Monitoring these events is crucial in order to reduce both the affected people and the economic losses. In this work we train and test three different Deep Learning segmentation algorithms to estimate the water area from river images, and compare their performances. We discuss the implementation of a novel data chain aimed to monitor river water levels by automatically process data collected from surveillance cameras, and to give alerts in case of high increases of the water level or flooding. We also create and openly publish the first image dataset for river water segmentation.**

***Keywords*-Deep Learning, flood, water, segmentation, emergency, monitoring;**

I. Introduction

Floods report global annual economic losses of $96 billion [1]. They are mainly due to river water overflows, which are caused either by heavy precipitations or by rapid snow melting. To improve river monitoring and early warnings, video cameras can be installed in riverbeds to assess the water level status. The most straightforward way to visually determine if an alert threshold is reached is to use static cameras pointed toward the riverbed and compare the water level against historical observations. However, the manual monitoring of video cameras is very costly. In this work, we propose to use a water segmentation technique to analyze video streams in real-time in order to automatically detect anomalies such as sudden water extend increases. Hence, we train and test three different Deep Learning algorithms for the task of water segmentation and compare their performances.

In Section II we make a brief summary of computer vision algorithms for flood prevention and detection, together with object segmentation algorithms. In Section III we describe the three algorithms used to perform the water segmentation, while in Section IV we introduce the dataset created for water segmentation. Section V is devoted to the explanation of the evaluations carried out to compare the selected algorithms, and of the performance metrics used. Also, we discussion the obtained results and the potential use of this tool to implement early warning in case of floods. Finally, in Section we present our conclusions and outline future works.

II. Related work

The incremental usage of surveillance cameras in areas prone to natural disasters has raised interest in such events in the scientific community, especially in the computer vision [2] domain.Background subtraction techniques together with morphological operations and color probability has been used to determine water presence in videos [3]. When it comes to static images, most algorithms are based on light, texture and color features, and on clustering or classification models to segment the regions containing water [4]. The main drawback of these algorithms is the usage of handcrafted features which work well only the specific context in which they were created, because they are dependent on image characteristics such as lighting conditions, water color, etc. Moreover, the comparison among such algorithms is difficult because all previous studies are evaluated on non-publicly-available data.

For the water segmentation task we investigate the use of Artificial Neural Networks, which are composed by a collection of connected computational units called neurons. The connection of several neurons forms a neural network, which is characterized by a set of weights (one for each connection) and biases (one for each neuron). Weights and biases are the parameters that define the differentiable function representing the neural network, and they are generally learned through a supervised approach. The neural network is commonly organized in layers, where the greater number of layers enable for more complex models. When the networks are composed by more than one layer they are referred as Deep Neural Networks and they are part of the Deep Learning family of machine learning algorithms. Deep Learning methods have been applied in many different fields such as medicine and medical diagnosis, data mining or pattern recognition.

In the last few years, the breakthrough of Deep Learning has stirred the field of computer vision. Recently, an algorithm based on deep features was introduced to determine the presence of flood in images from social media [5]. The results reported are promising, however the output is a probability of the image containing evidence of a flood. Therefore, this algorithm can only be used once the disaster has already occurred and does not support flood monitoring.

Semantic segmentation of images is the process of classifying with a semantic class every pixel of an image according to the human perception. Semantic segmentation has been a classic topic in computer vision, which has seen great advances through the application of Deep Learning algorithms. Initially, algorithms that performed pixel-wise labeling through visual features were introduced [6], [7]. Then, fully convolutional networks were successfully applied to the problem of semantic segmentation, improving performances over previous methods. As a general rule, deeper networks extract higher semantic information but at the same time they lose pixel location information. Several techniques have been studied for solving this drawback, such as bilinear interpolation [8], unpoooling operations [9], [8] or skip-layers, which combine fine information from early layers with coarse information from deeper layers [10], [11]. Algorithms based on Adversarial Networks have also been applied to segmentation problems [12], [13].

Despite the advances of Deep Learning semantic segmentation algorithms, to the best of our knowledge they have not been applied to the problem of water identification in images, yet. In this paper, we train three different state-of-the-art algorithms [10], [11], [12] on a novel publicly available dataset of river images, which we introduce in Section IV.

III. Semantic segmentation algorithms

In order to anticipate floods caused by water overflow in rivers, we propose to automatically detect an increase in river water levels through water segmentation from videos taken by surveillance cameras installed near riverbeds. The algorithm outputs a water percentage, which can be mapped to the water level increase. When such percentage goes above a certain threshold, which is computed from historical observations, a warning can be triggered. Therefore, it is crucial to achieve a fine water segmentation in order to perceive variations in water levels between frames in a time series. Convolutional Neural Networks (CNN) typically use subsampling to keep filters small and to reduce computational costs. By doing so, the output feature map from deeper layers is reduced, resulting in an segmentation prediction of a smaller size of the input image. Early algorithms based on CNN for semantic segmentation were normally composed by a CNN phase to output a fixed size coarse segmentation map, followed by an upsampling algorithm, and in some cases a refinement phase. This algorithms were not trainable end-to-end and could only work with images of a fixed input size due to the presence of fully connected layers operating with a fixed sized input. Later, Fully Convolutional Networks (FCN) were introduced to solve the problem of semantic segmentation [10], [14], [11]. Using a fully convolutional network allows to process input images of variable size. In order to do so, fully connected layers are transformed into convolutions with kernels which cover the entire input region. Moreover, in this architectures upsampling layers were also introduced to obtain an output of the same resolution of the input and *skip* layers, which combine finer information from earlier layers with semantically more relevant information from deeper layers, making the whole algorithm trainable end-to-end. Given the advantages of FCN we have chosen two stat-of-the-art algorithms [10], [11] based on this technique which have reported top performances in well known segmentation datasets such as PASCAL [15] and CamVid [16]. Additionally, we compare the performances obtained with a segmentation algorithm based on Conditional Adversial Networks [12], which has shown very good results in several computer vision tasks, including semantic segmentation. To the best of our knowledge this algorithm has not yet been quantitatively compared with FCN segmentation algorithms.

Next, we briefly explain the three different algorithms chosen for the task:

1) Fully convolutional networks for semantic segmentation (FCN-8s) [10]: One of the first proposed FCN for semantic segmentation, adapting contemporary classification networks such as AlexNet [17], VGGnet [18] and GoogLeNet [19] into FCN. To produce dense predictions they propose to do bilinear interpolation and upsampling using convolutions with an input stride of $1/f$, where f is the factor needed in order to reverse the forward and backward passes of the convolution. The deconvolutional filters are also learned. Skip layers are added to combine the prediction layer with previous layers with finer strides [1].

2) Fully Convolutional DenseNets for Semantic Segmentation (Tiramisu) [11]: Similarly to FCN8-s, in Tiramisu a CNN network proposed for image classification is adapted into a FCN network with skip and upsampling layers to perform semantic segmentation. However, Tiramisu is based on an CNN architecture called DenseNet which was first introduced in [20]. This architecture is built from a series of *dense blocks* that are iterative concatenation of previous feature maps, naturally introducing skip connections and multi-scale supervision. Additionally, upsampling operations called *transition up* are added to recover the resolution of the input image. This layers consist in a transposed convolution that upsamples the previous feature maps. In order to avoid the linear growth in the number of features due to the upsampling paths, the input of a dense block is not concatenated with its output and thus the transposed convolution is applied only to the feature maps obtained by the last dense block.

3) Image-to-Image Translation with Conditional Adversarial Networks (Pix2Pix) [12]: These networks learn to

[1] https://github.com/shelhamer/fcn.berkeleyvision.org

map an input image to an output image and the loss function to train this mapping. They can be applied to multiple tasks such as synthesizing photos, reconstructing objects from edge maps or semantic segmentation among others. The main advantage of this architecture is that it does not rely on a manually fixed loss function that forces the network to learn the task. The network is composed by a generator G which produces outputs which ideally can not be distinguished from the "real" images and a discriminator D which learns to distinguish between the images generated by G and the "real" ones. In order to overcome the losses in resolution due to the progressive downsampling of the features, they also use skip connections which consist of a concatenation of all the channels from layer i with those from layer $n - i$, where n is the total number of layers. Moreover, they combine the the GAN objective with an L1 distance to the ground truth in order to impose to the generator to generate an output similar to the ground truth [2].

IV. Dataset

We present a publicly available dataset for water segmentation in rivers [21]. It has been created using self-gathered images, images retrieved from Google, and images from surveillance cameras in riverbeds and labeled by a human annotator. The dataset contains a total of 300 images of different rivers. For every image there is the corresponding ground truth file which consists of a two-dimensional binary matrix of zeros for the pixels which contain background information and ones for the pixels which contain water information. In Figure 1 (a-c) we show some examples of the images in the dataset, while in Figure 1 (d-f) the corresponding water ground truth is displayed. Water regions are represented in white and the background in black. The dataset has a big variance among the images in terms of water color, turbulence, angle, and illumination. In Table I we report further information about the size of the images from the dataset and the amount of water present in the images.

Smallest image	118x158 pixels
Biggest image	2448x3264 pixels
Mean size of image	550x826
Min percentage of water	6.31%
Max percentage of water	91.37%
Mean percentage of water	39.57%

Table I: Meaningful information about the dataset.

V. Experiments

We divided the dataset in 75% training and 25% in test. For training We have used the default parameters suggested by the authors for each algorithm.

We evaluate the results of the three algorithms on the test set using two common semantic segmentation metrics, the Mean Intersection over Union (MIoU) and the pixel-wise accuracy (Pa), which are given by:

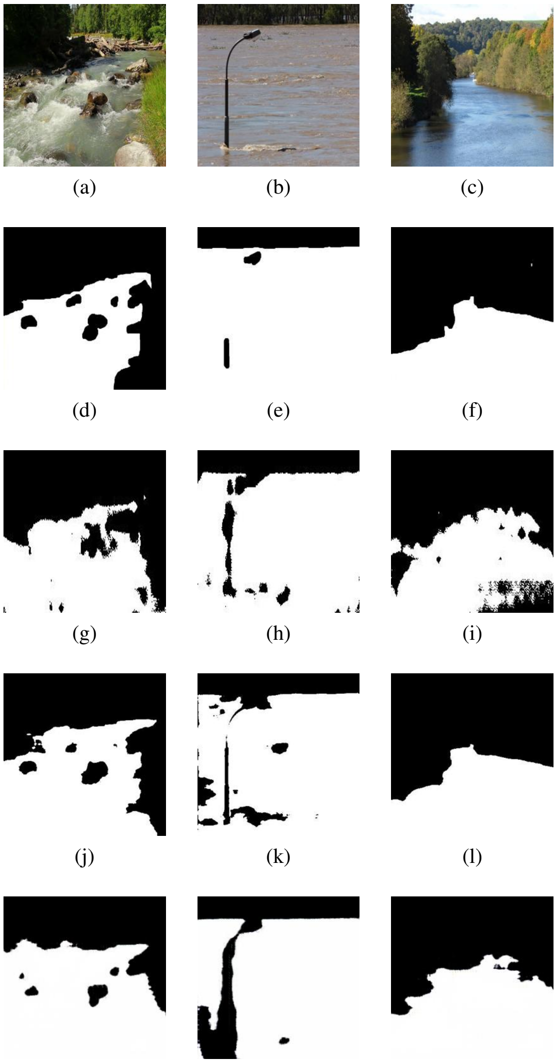

Figure 1: Qualitative results from the river segmentation, first row corresponds to the original image, the second is the ground truth, the third is the result from the FCN-8s, the fourth from the Tiramisu and in the last row from Pix2Pix.

$$\text{MIoU} = \frac{(1/C)\sum_i n_{ii}}{t_i + \sum_j n_{ji} - n_{ii}}, \quad \text{Pa} = \frac{\sum_i n_{ii}}{\sum_i t_i},$$

where n_{ij} corresponds to the number of pixels from class i which have been wrongly classified as belonging to class j, n_{ii} represents the pixels from class i which have been correctly classified, C is the total number of classes and finally t_i is the total number of pixels of class i. In Table II we report quantitative results over the test set, specifically statistic measures (mean and standard deviation) associated with the two semantic metrics. The best results in terms of

[2] https://github.com/phillipi/pix2pix

both mean IoU and accuracy are obtained by the Tiramisu framework, which obtains over 5% better performance in both metrics than the second best performing algorithm, i.e., Pix2Pix. Moreover, Tiramisu achieves the lowest standard deviation, which suggests that the results are consistent among the different images. In Table III we show the best and worst performing algorithms on individual images considering the previous two metrics. Again, Tiramisu is the algorithm having the highest number of bests and the lowest of worsts. In Figure 1 we show some qualitative visual results: the results from Tiramisu seem to resemble more the ground truth while the FCN-8s algorithm seems to perform worst.

	MIoU [%]		**Pa**[%]	
	Mean	Std	Mean	Std
FCN-8s	70.05	14.92	82.80	11.04
Tiramisu	**81.91**	**13.74**	**90.47**	**9.16**
Pix2Pix	72.25	14.27	84.7	10.05

Table II: Tiramisu algorithm obtains the best mean in both metrics with lower standard deviation.

	Number of images			
	MIoU		**Pa**	
	Worst	Best	Worst	Best
FCN-8s	37	6	37	7
Tiramisu	**10**	**57**	**11**	**56**
Pix2Pix	28	12	27	12

Table III: Tiramisu is the best performing algorithm: around 75% of the bests and less less than 15% of worsts, considering both metrics.

VI. Conclusions and Future Works

In this work we have studied three state-of-the-art algorithms for semantic segmentation and applied them to water segmentation in rivers. We have introduced a new dataset for water segmentation which we have used to train and test the algorithms. Tiramisu was the best performing algorithm for water segmentation, both quantitatively and qualitatively. This work is particularly relevant for the implementation of an automatic detection of water level increases using cameras in riverbeds in order to improve flood early warnings.

Acknowledgment

This work was partially supported by the Spanish Grants TIN2016-75404-P AEI/FEDER, UE, TIN2014-52072-P and the European Commission H2020 I-REACT project no. 700256. Laura Lopez-Fuentes benefits from the NAER-INGSPHD fellowship of the Norwegian Research Council under the collaboration agreement Ref.3114 with the UIB.

References

[1] "Worlds 15 countries with the most people exposed to river floods," 2015. [Online]. Available: goo.gl/pZo7su

[2] L. Lopez-Fuentes, J. van de Weijer, M. Gonzalez-Hidalgo, H. Skinnemoen, and A. D. Bagdanov, "Review on computer vision techniques in emergency situation."

[3] A. Filonenko, D. C. Hernández, D. Seo, K.-H. Jo *et al.*, "Real-time flood detection for video surveillance," in *Industrial Electronics Society, IECON 2015-41st Annual Conference of the IEEE.* IEEE, 2015, pp. 004 082–004 085.

[4] C. Lai, J. Yang, and Y. Chen, "A real time video processing based surveillance system for early fire and flood detection," in *Instrumentation and Measurement Technology Conference Proceedings, 2007. IMTC 2007. IEEE.* IEEE, 2007, pp. 1–6.

[5] L. Lopez-Fuentes, J. van de Weijer, M. Bolaños, and H. Skinnemoen, "Multi-modal deep learning approach for flood detection," in *Working Notes Proceedings of the MediaEval 2017 Workshop.* CEUR-WS, 2017, pp. 1–3.

[6] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.

[7] M. Mostajabi, P. Yadollahpour, and G. Shakhnarovich, "Feedforward semantic segmentation with zoom-out features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3376–3385.

[8] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *arXiv preprint arXiv:1511.00561*, 2015.

[9] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1520–1528.

[10] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[11] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).* IEEE, 2017, pp. 1175–1183.

[12] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *arXiv preprint arXiv:1611.07004*, 2016.

[13] C. Gan, N. Wang, Y. Yang, D.-Y. Yeung, and A. G. Hauptmann, "Devnet: A deep event network for multimedia event detection and evidence recounting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2568–2577.

[14] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention.* Springer, 2015, pp. 234–241.

[15] M. Everingham and J. Winn, "The pascal visual object classes challenge 2011 (voc2011) development kit," *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep*, 2011.

[16] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *European conference on computer vision.* Springer, 2008, pp. 44–57.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[20] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," *arXiv preprint arXiv:1608.06993*, 2016.

[21] L. Lopez-Fuentes and C. Rossi, "River segmentation dataset," Oct. 2017. [Online]. Available: https://doi.org/10.5281/zenodo.1003085