CLASSIFICATION OF UNDERWATER PIPELINE EVENTS USING DEEP CONVOLUTIONAL NEURAL NETWORKS

Felipe R. Petraglia, **Advisor:** *José Gabriel R. C. Gomes*

Federal University of Rio de Janeiro
COPPE - Electrical Engineering Program
Rio de Janeiro, Brazil
fpetraglia@pads.ufrj.br, gabriel@pads.ufrj.br

ABSTRACT

Automatic inspection of underwater pipelines has been a task of growing importance for the detection of a variety of events, which include inner coating exposure and presence of algae. Such inspections might benefit of machine learning techniques in order to accurately classify such occurrences. This article describes a deep convolutional neural network algorithm for the classification of underwater pipeline events. The neural network architecture and parameters that result in optimal classifier performance are selected. The convolutional neural network technique outperforms the perceptron algorithm, for different event classes, reaching on average 93.2% classification accuracy, while the accuracy achieved by the perceptron is 91.2%.

***Index Terms*—** Convolutional neural network, event classification, feature extraction, wavelet transform, perceptron.

1. INTRODUCTION

The intensification of subsea oil and gas field exploitation has turned the inspection of underwater pipelines into a progressively demanding task. Usually conducted with the use of ROVs (Remotely Operated Underwater Vehicles), which employ sensors and cameras and are controlled through radio or cable connections [1], visual inspection by humans is a tedious endeavor, particularly in the cases of long inspections, low image quality and search for multiple targets [2]. In contrast to ROVs, Autonomous Underwater Vehicles (AUVs) are able to automatically detect and track underwater pipelines. In this regard, event classification methods based on machine learning can be used in order to automatically inspect the pipelines.

Classic neural network techniques, such as the multilayer perceptron (MLP), are strongly dependent on feature extraction methods, which are often manually carried out. Recently, deep learning algorithms have been able to iteratively extract their own features from original data. This paper focuses on the application of one of these recent techniques, namely convolutional neural network (CNN), to event classification. A method consisting of a perceptron preceded by a wavelet-based feature extractor is also described, and the results obtained using the perceptron and the deep CNN architectures are compared.

This work was supported by Brazilian higher education and research funding agencies: CAPES, CNPq, and FAPERJ.

2. EVENT CLASSES AND DATA SETS

The classifier developed in this work was used to detect four different event types. Inner coating exposure (ICE) occurs when the pipeline surface is damaged. The outer cover disruption is caused by the object impact and by natural circumstances, such as waves, sea currents, among others. Visually, it can be described as a texture region containing parallel stripes, possibly surrounded by homogeneous regions.

The presence of algae can be characterized by a variety of shapes, colors and textures. This event might hide damages on the pipeline surface, hampering their detection.

Flanges are structures commonly found at pipeline junctions, and they are used for holding pipeline sections together. When they are seen from a frontal view, these events are outlined by hexagons surrounding cylinders. These formations can also be seen from a side view, and in that case they are characterized by thinner rectangles emerging from thicker structures.

Concrete blankets (CB) are structures placed under or over the pipelines, and they are constructed to give support or protect the pipelines from vibrations. These events are usually identified by a regular brick array.

In order to train and test the implemented classification system, windows containing event samples were extracted from high resolution images (1280×720), thus composing databases that are used as neural network inputs. Windows that do not contain any of these classes were also extracted, to compose the negative sample database. For each event class, positive and negative samples were mixed, so that the system would perform binary classification. 60×60 pixel windows were extracted for ICE, algae and CB samples, whereas 80×80 pixel windows were extracted for flanges, due to the need to include their entire geometry in each sample.

Fig. 1 shows, in each row, samples belonging to each class. In this figure, windows containing flanges were resized to 60×60 pixels, to match the other samples visual aspect.

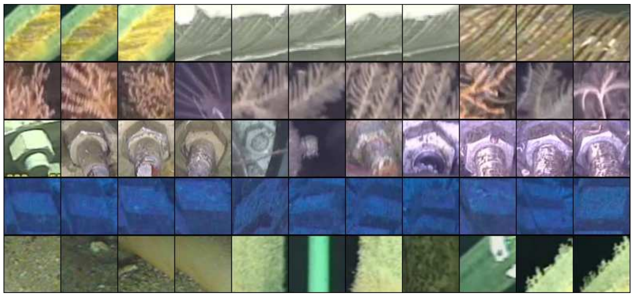

Fig. 1. Samples from different event datasets. From top to bottom rows, ICE, algae, flange, CB and negative samples.

3. CONVOLUTIONAL NEURAL NETWORK

The implemented CNN topology consists of two convolutional, two max pooling and three fully connected (FC) layers. The first layer is a 32 kernel convolutional layer [3], which detects relatively simple features, which can be easily recognised and interpreted. After that, a pooling layer [3] is used, so that max pooling is applied to 2×2 regions, with strides [3] of 2. Subsequently, another 32 kernel convolutional layer is applied, to detect more abstract and detailed features, which are usually present among the ones from the previous layer. A pooling layer applies max pooling once again to 2×2 regions, with strides of 2. Three FC layers are subsequently applied. They map the previous layer outputs into deeper features, to allow the best classification performance. The FC layers have, respectively, 512 outputs, 256 outputs and 1 output.

At each convolutional layer output, batch normalization [4] is performed. The optimizer utilized was Adam [5], with learning rate set to 0.001. Rectified linear unit (ReLu) activation function was used after each convolutional and FC layer. After every FC layer, dropout regularization [6] was used, with dropout probability of 50%. The loss function chosen was cross entropy. The batch size was set to 100. The CNN was implemented using Keras API.

4. RESULTS

The CNN classification accuracy was compared to the one from a system comprised of a multilayer perceptron preceded by a wavelet-based feature extractor [7]. A 3-level Daubechies 2 (Db2) wavelet was employed, and the mean and the variance of the wavelet coefficients at each level were used as features for the neural network [7].

Results obtained by the CNN and by the MLP for the four classes of events are shown in Table 1. For each event class, 100,000 positive and 100,000 negative samples were randomly mixed. Among these, 162,000 samples were used for training, 20,000 for testing and 18,000 for validating the networks. Before being applied to the neural network input, each window is converted to grayscale, in order to eliminate color dependence.

Table 1. Classification accuracy for the four different event classes.

Accuracy (%)	ICE	Algae	Flange	CB
CNN	96.5	98.3	83.0	95.0
MLP	94.6	97.0	82.4	90.8

5. CONCLUSIONS

The CNN was shown to efficiently classify underwater pipeline events in comparison with the MLP based on wavelet-computed features. Without the need of manually selected feature extraction, the CNN obtained a higher classification accuracy on all four event classes that were considered in this paper, achieving 93.2% on average, whereas the perceptron accuracy reached 91.2% on average.

6. REFERENCES

[1] J. Antich and A. Ortiz, "Underwater cable tracking by visual feedback," in *Proceedings of Pattern Recognition and Image Analysis*. Lecture Notes in Computer Science, 2003.

[2] M. Jacoby and D. Karimanzira, "Underwater pipeline and cable inspection using autonomous underwater vehicles," in *MTS/IEEE OCEANS - Bergen*, 2013.

[3] M.D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*, 2014.

[4] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *32nd International Conference on Machine Learning*, 2015.

[5] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *5th International Conference on Learning Representations*, 2015.

[6] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," in *Journal of Machine Learning Research*, 2014.

[7] E. Medina, M.R. Petraglia, and J.G.R.C. Gomes, "Neural-network based algorithm for algae detection in automatic inspection of underwater pipelines," in *15th International Conference on Artificial Intelligence*, 2016.