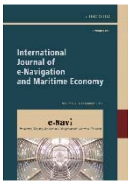
Available online at www.e-navigation.kr
e-Navigation Journal



Original article

Fine-grained Boat Classification using Convolutional Neural Networks

Michele FIORINI[a], Domenico D. BLOISI [b*], Ali YOUSSEF[c], Andrea PENNISI[d]

[a] The Institution of Engineering and Technology (IET), Italy Network, mfiorini@theiet.org
[b*] Dept. of Computer Science, University of Verona, Italy, domenico.bloisi@univr.it, Corresponding Author
[c] Dept. of of Computer, Control, and Management Engineering, Sapienza University of Rome, Italy, youssef@diag.uniroma1.it
[d] Dept. Electronics & Informatics (ETRO), Vrije Universiteit Brussel (VUB), apennisi@etrovub.be

Abstract

The use of radar-based systems for vessel monitoring is not suitable in populated areas, due to the high electromagnetic emissions. In this paper, a camera based vessel recognition system for application in the context of Vessel Traffic Services (VTS) and Homeland Protection (HP) is proposed. Our approach is designed to extend the functionality of traditional VTS systems by permitting the classification of both cooperative and non-cooperative targets, using camera images only. This allows enhancing the surveillance function in populated areas, where public opinion is strongly concerned about electromagnetic emissions and therefore antennas are suspiciously observed and radars are not allowed. Experiments have been carried out on a publicly available data set of images coming from the ARGOS boat traffic monitoring system in the City of Venice (Italy). The obtained classification accuracy of 89.6% (with 11 different classes of boats) demonstrates the effectiveness of the proposed approach.

Keywords: Boat Data Set, Optical Tracking, Vessel Classification, Green VTS, Convolutional Neural Network

1. Introduction

Vessel Traffic Services (VTS) systems contribute to the safety and efficiency of navigation, safety of life and protection of the environment. These tasks may demand specific traffic management to minimise incident, improve better use of ports and navigation facilities promoting positive economic results. VTS are in charge of acquisition, processing, and analysis of data in order to provide monitoring and navigational advices from inland waters up to territorial waters (12 NM). Classic VTS are equipped with Radar and U/VHF radio in different forms. This means that automatic classification is possible only for cooperative targets, i.e., the ones equipped with the Automatic Identification System (AIS), while non-cooperative (non-AIS) targets have to be identified manually by human operators. Moreover, the use of radars can be problematic in populated areas, where public opinion is strongly concerned about electromagnetic emissions and therefore antennas are suspiciously observed and radars are not allowed.

In this paper, we describe a vision-based classification approach for application in the context of VTS and Homeland Protection (HP). In particular, we adopt a robust deep learning technique for boat classification, using real images captured by ARGOS system for training and testing (Bloisi et al., 2009), an advanced automatic traffic monitoring system operating 24x7x365 in the City of Venice, Italy (see Figure 1).

The main contribution of the proposed approach consists in the use of a two-step strategy: First, a binary (i.e., boat/no-boat) classification, based on off-the-shelf Convolutional Neural Network (CNN) features, is carried out. Then, a multi-class classification, using a training set containing seventeen classes, is performed. To validate quantitatively the proposed approach, we use the *ARGOS classification data set*, which is part of the publicly available Maritime Detection, Classification, and Tracking (MarDCT) database (Bloisi et al., 2015). The ARGOS classification data set is unique in its nature and it is very challenging due to the presence of boat wakes, waves, reflections, and boats navigating very close each other.

The rest of the paper is organized as follows. Section II contains an overview of recent deep learning approaches for image classification. Section III contains the details of the proposed algorithm, while qualitative and quantitative results are shown in Section IV. Finally, conclusions are drawn in Section V.

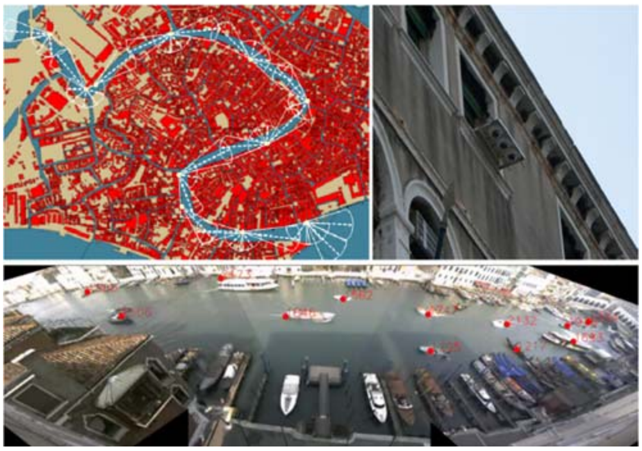


Figure 1: The ARGOS system in Venice

2. Related Work

This section contains, in the first part, a discussion about recent deep learning solutions for the general problem of object detection that are applicable to the problem of boat detection. Then, we discuss some specific features for the boat classification problem.

2.1. CNN based Classification

Convolutional Neural Network (CNN) has shown impressive performance on image classification (Krizhevsky et al., 2012) and object detection (Erhan et al., 2014). CNNs can handle the presence of multiple instances of the same object in the processed image. In the group of CNN based methods, DetectorNets (Szegedy et al., 2013) and OverFeat (Sermanet et al., 2013) perform the object detection on a coarse set of sliding-windows. On the other hand, Region-based Convolutional Neural Networks (R-CNNs), proposed by Girshick et al. (2016), work on the extraction of proposal regions that are a subset of all the possible image locations. Detection is carried out by applying a classification on different regions (patches) extracted from the original image. The patch with high probability does not represent only the class of that region, but also gives its location in the image. The second stage of R-CNN involves improving the localization accuracy by minimizing the error of the predicted coordinates against the ground truth coordinates. To this end, a linear regression layer is optimized. Then, a non-maximal suppression technique is used to merge highly overlapping regions, which are predicted to be of same class.

2.2. Boat Classification

A number of approaches have been proposed in the literature to provide solutions for classifying vehicles based on Computer Vision and CNN approaches (Bousetouane and Morris, 2015). The number of the different boat types navigating in the City of Venice is very high, this make the recognition and classification tasks very challenging (Bloisi et al., 2007).

In this work, we use a data set made of fixed size snapshots acquired from real cameras placed in the Grand Canal water channel in Venice. It has been generated by using the detection and tracking functionalities of the ARGOS system. Moreover, we use the classification method described in (Bloisi et al., 2013), as baseline to evaluate the performance of our approach.

The boat classification process has been designed to be a starting point for obtaining benchmark results within the MarDCT data sets. The main challenges in the classification task are represented by:

1. Static elements classified as boats.
2. Partial view of the captured boat(s).
3. Multiple boats captured in the same snapshot.
4. False positive detections caused by waves and reflections.

A drawback of the ARGOS classification data set is the presence of some labels not accurately placed. These annotations do not distinguish among cases in which more than one boat is present. In such a case, a label named *multiple_occurence* is assigned and no distinction among the type of vessels present in the image is given. Moreover, the data set has a label called *water*, which includes false positive snapshots that are incorrectly classified as containing boats.

In this paper, we focus on improving the boat detection deployed in the ARGOS system by handling the above listed challenges and the image acquisition (i.e., by filtering out background elements like water or boat cropped partially). This led to train the CNN model on an increased data set for boat recognition. Since the number of labelled images is greater, partial view and multiple boats in the captured snapshots can be taken into account. As a result, it is possible to improve the classification performance.

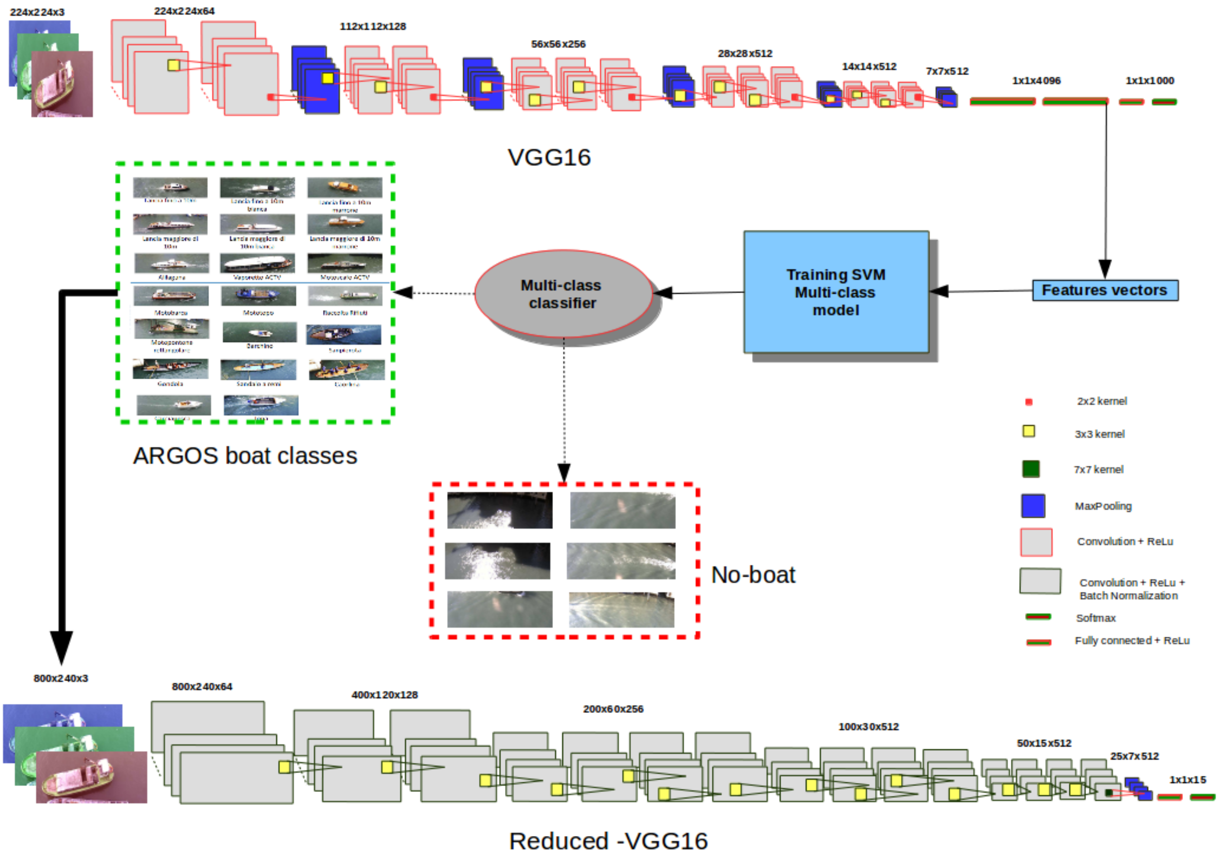


Figure 2: Functional architecture of the proposed approach

3. Proposed Method

The functional architecture of the proposed method is shown in Figure 2, with all the different modules involved in the computational pipeline. We use a pre-trained VGG16 deep neural network in two different ways:

1. For a coarse-grained detection (i.e., boat/no-boat classification).
2. For a fine-grained multi-class detection procedure.

The aim of this two-step procedure is to reduce the search space. Since the ARGOS classification data set contains also images without boats, the coarse-grained classification is needed to handle undesired images. In particular, our method first scans the captured image, extracts a feature descriptor of salient object present in it, and then classifies it into specific categories of boats.

3.1. Transfer Learning

A transfer learning strategy is used to address the need of a large data set for modeling the CNN during the training stage and to deal with the computational time and resources needed. As anticipated in the previous section, instead of training the deep network from scratch with random weight initialization, we use a pre-trained VGG16 net for object detection. The VGG16 structure needs to be modified for dealing with our fine-grained classification problem. Since the CNN filters extract generic features (e.g., edges and colours) at earlier layers, our idea is to modify the final layer of the CNN, which tends to be more class-specific. Summarizing, the same CNN structure of VGG16 has been adopted in two transfer learning steps:

1. Replacing the classifier at the last layer of VGG16.
2. Fine-tuning of the weights of the trained model.

In order to adapt the VGG16 classifier to our needs, we removed the last fully-connected layer of the network by extracting the features at the fully-connected layer FC7, and by training a linear Support Vector Machine (SVM). Since FC7 has an output size of 4096, the extracted features are included in vectors of size 4096, which are used for training a multi-class SVM model on the ARGOS data set, and for testing the capability of the obtained CNN.

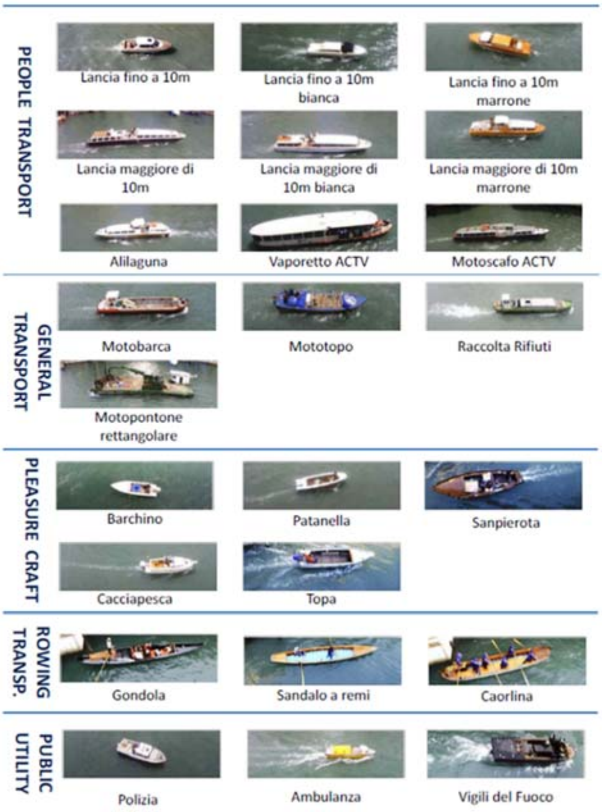


Figure 3: Boat categories in the ARGOS data set

3.2. SVM Model Training

For training the multi-class SVM model, we have divided the boat data set with a 7:3 train to test ratio. A cross-validation like procedure is used in order to test all the images in a single class. Different training steps are carried out to test all the images and to extract the bounding boxes and the visual features related to the objects of interest. The linear SVM model is trained using stochastic gradient descent (SGD) and it is then used for binary classification among the extracted object. In such a way, the input of the CNN is a set of images containing only the desired object (i.e., a boat) with a limited area containing background information (e.g., water and boat wakes). This step allows to accurately train the CNN.

3.3. Training Data

The training data for our approach are extracted from the publicly available image and video database MarDCT. In particular, the used ARGOS classification data set contains 24 different classes (see Figure 3), which are exploited as positive images for the classification procedure.

A special *water* class is present that may help to reduce the search space or to ignore the false positives caused by waves and wakes. The snapshots have a fixed size of 800×240 pixels. Due to this big size, it is possible to have different objects in the snapshot in addition to boats, especially in the case of small and middle size boats.

Our data set represents a unique domain due to the presence of a number of boat types that are specific to the City of Venice (e.g., gondolas). However, from a general point of view, our data set is not extremely different in context from the data set used for training VGG16 model. As stated before, the VGG16 model is able to capture general features in its early layers, which are relevant and useful for boat classification.

A slight modification is made on the VGG16 structure to improve training time with no effect on classification performance. We kept the size of the input image (i.e., 800×240) to avoid any distortion. The amount of computational by convolutional layers will greatly increase as initial set of convolutions are occurring over the entire input image. The first convolutional layer is replaced by a 3x3 strided convolutional layer. The pooling layers are removed to transfer learning through kernel associations instead of fixing the pooling operation. Springenberg et al. (2014) to reduce the computational burden have presented a similar work. A single classification layer replaces the fully connected hidden layers. Moreover, to speed up the training, a batch normalization layer is added after each convolutional layer (Ioffe et al., 2015).

Data set augmentation, obtained by increasing the number of the training samples, is used in order to allow the network to be slightly more invariant, and to avoid overfitting. The horizontal flipping, blurring, adding noise and denoising images allow us to obtain a data set four times larger than the original. Furthermore, dropout (Srivastava et al., 2014) is used on top of the last convolutional layer during training to improve the generalizability of the model. The data set is shuffled and divided with the ratio 5:1 train to validation test.

Figure 4 shows the learning rate curve, the training and test losses and the validation accuracy as a function of the number of iterations.

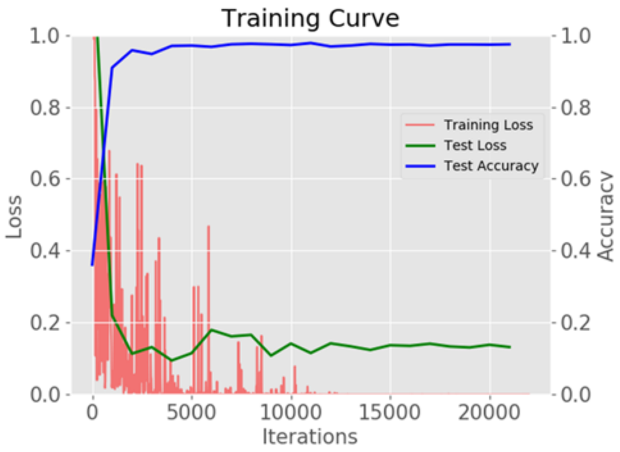


Figure 4: Training and test losses, together with test accuracy, visualized as a function of the number of iterations

The model achieves a top 3 validation accuracy of 97% with a value of 0.1 for the validation loss. Both curves converge after 15000 iterations.

4. Experimental Results

Figure 5 shows eight test images with the corresponding classification results. Incorrect results (see Figure 5B) are mostly due to the presence of multiple vessel in the same image and to the noise added by the motion of the boats (e.g., boat wakes).

For training the SVM multi-class classifier, we used 17 classes, while for training the reduced VGCC16 model, we used 11 classes, selected by excluding the *water* class and other categories not having enough samples. The boat samples includes boats used for transportation of people (e.g., *vaporetto*) and goods (e.g., *mototopo*), public utility boats (e.g., *ambulance*), boats used for pleasure and tourism (e.g., *gondola*).

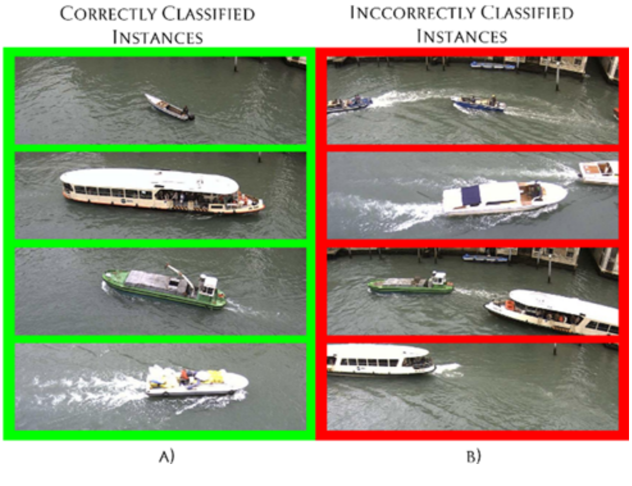

Correctly Classified Instances
Inccorrectly Classified Instances
A)
B)

Figure 5: Classification examples. A) Correctly classified vessels. B) Incorrect classified samples

Table 1: Number of images per boat type

Class	Train images	Test images	Total
Alilaguna	376	75	452
Ambulanza	284	56	340
Barchino	374	74	448
Lanciafino	2917	583	3500
Motobarca	717	143	860
Mototopo	2927	585	3512
Patanella	930	186	1116
Polizia	237	47	284
Raccolta rifiuti	314	62	376
Topa	260	52	312
Vaporetto	3164	632	3796

Table 1 shows each categories and the number of samples used in both training and validating the model. The use of data coming from the scenario of the City of Venice allows for analyzing a large variety of boat categories. In particular, the used data set contains 24 different categories of boats navigating in the Grand Canal (see Figure 3).

We use the one-versus-all (OVA) approach for this multi-class problem. The overall accuracy of the multi-class task is computed as:

$$acc = \frac{TP + TN}{P + N} \quad (1)$$

where *acc* represents the classification accuracy over the specified number of classes, *TP* are the true positive classified samples, *TN* is the number of the true negative classified samples, *P* is the number of all positive samples, and *N* is the number of all negative samples.

Results are given in Figure 6, where the confusion matrix shows that the SVM multi-class model achieves an accuracy up to 98% on some categories (e.g., vaporetto, gondola) and the overall accuracy on 17 classes is 82.4%. It is worth to be noted that in our previous work (Bloisi et al., 2015), we obtained an accuracy of 73% with traditional classification approaches, namely Random Forest (RF), Decision Tree Learning Algorithm (J48), and K-Nearest Neighbor (KNN).

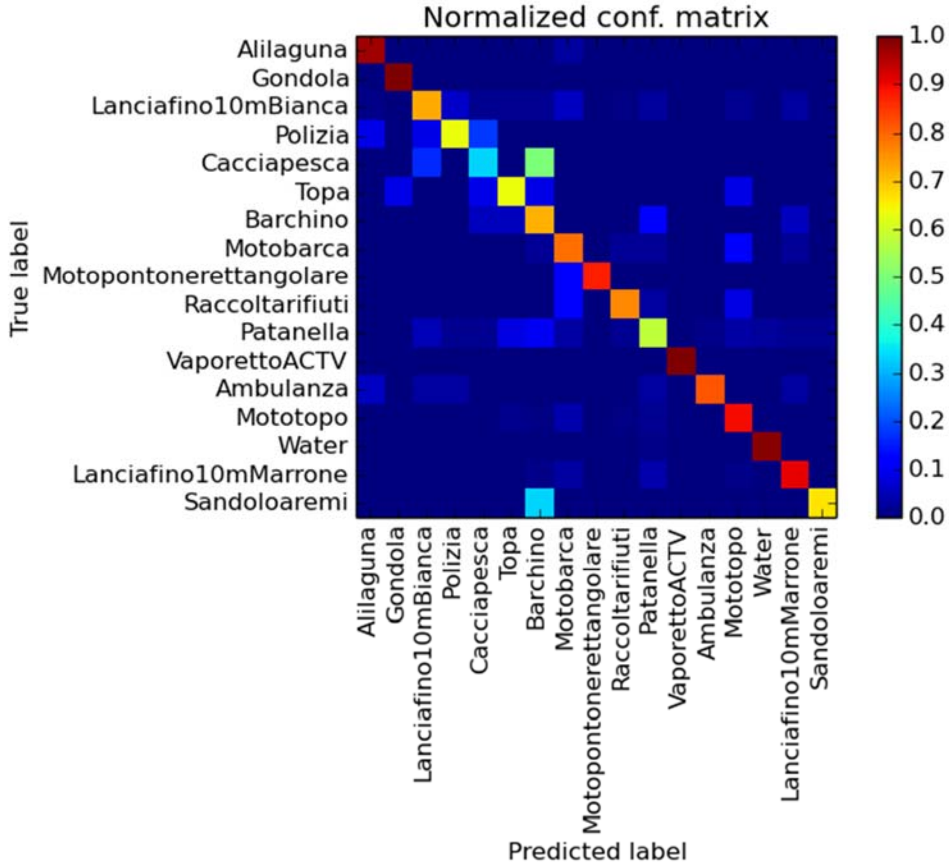


Figure 6: Confusion matrix for 17 classes

Table 2: Classification results

Model	acc top 1	acc top 3
SVM 17 classes	0.824	-
SVM 15 classes	0.860	-
VGG16 reduced	**0.896**	**0.965**

The reduced VGG16 network is trained on 11 boat categories using the Caffe framework (Jia et. al., 2014). The multinomial logistic loss for a one-of-many classification task is computed over real-valued predictions probability distribution over classes, which are given by SoftMax. The training loss and test loss are measured. The test accuracy is measured as top 1 accuracy (the model gives the highest prediction to the correct class) and the top 3 accuracy. Table 2 shows the quantitative classification results by adopting a pre-processing step for filtering out possible outliers.

Figure 7 shows the achieved improvement on the classification procedure.

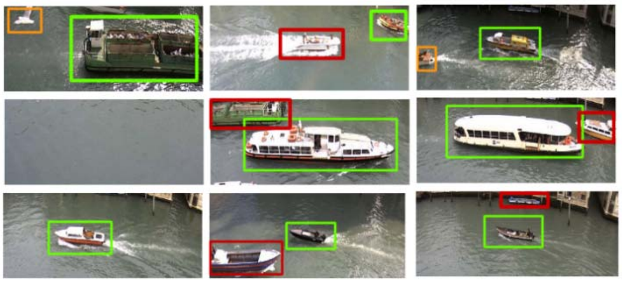


Figure 7: Classification of multiple boats in the same image

The *water* class can be ignored, and the images with multiple boats can be processed correctly. Moreover, the background elements are removed, and the dataset can be used for further training routines.

5. Conclusions

In this paper, we have presented a vision-based method for classifying cooperative and non-cooperative boats in populated areas. The proposed approach can be used to enhance the traditional surveillance functions in existing VTS systems.

Experimental results demonstrate that it possible to achieve an accuracy of 89.6% and to deal with challenging situations, including the presence of multiple boats or partial occlusions (some of the images used for testing contain only portions of vessels). The experimental validation has been carried out by making use of real data coming from the ARGOS system installed in the challenging scenario of the City of Venice in Italy. The complete data set used for the evaluation, together with additional image sequences captured in real VTS sites across Europe, are made available by the authors of this paper through the MarDCT database (Bloisi et al., 2015).

As future work, we intend to test the proposed approach in other challenging real world scenarios, such as the Hong Kong bay, where different type of vessels are present.

Submitted : July 16, 2017
Accepted : November 15, 2017

Acknowledgements

This work has been possible thanks to the support of Venice Municipal Authorities and Archimedes Logica Srl. We thank them for the permission of capturing and processing video streams with the ARGOS system and for helping in the production of the data set.

References

Bloisi, D. D. and Iocchi, L. (2009), ARGOS - A video surveillance system for boat traffic monitoring in Venice. *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 23, No. 7, pp. 1477–1502.

Bloisi, D. D., Iocchi, L., Leone, G. R., Pigliacampo, R., Tombolini, L. and Novelli, L. (2007), A distributed vision system for boat traffic monitoring in the Venice Grand Canal. In: *Int. Conf. on Computer Vision Theory and Applications (VISAPP)*, pp. 549–556.

Bloisi, D. D., Iocchi, L., Pennisi, A. and Tombolini, L. (2015), Argos-Venice boat classification. In: *Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–6.

Bousetouane, F. and Morris, B. (2015), Off-the-shelf CNN features for fine-grained classification of vessels in a maritime environment. In: *Int. Symp. on Visual Computing*, pp. 379–388.

Chatfield, K., Simonyan, K., Vedaldi, A. and Zisserman, A. (2014), Return of the devil in the details: Delving deep into convolutional nets. In: *British Machine Vision Conference (BMVC).*

Erhan, D., Szegedy, C., Toshev, A. and Anguelov, D. (2014), Scalable object detection using deep neural networks. In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 2147–2154.

Fiorini, M. and Lin, J.-C. (2015), *Clean Mobility and Intelligent Transport Systems*, The IET, Transportation Series 1, pp. 237–263.

Girshick, R., Donahue, J., Darrell, T. and Malik, J. (2016), Region-based convolutional networks for accurate object detection and segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 142–158.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: *Int. Conf. on Machine Learning*, pp. 448-456.

Jia, Y. Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S. and Darrell, T. (2014), Caffe: Convolutional architecture for fast feature embedding. In: *22nd ACM International Conference on Multimedia*, pp. 675–678.

Krizhevsky, A., Sutskever, I. and Hinton, G. E. (2012), Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 1097–1105.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C. and Reed, S. E. (2015), SSD: Single shot multibox detector, arXiv:1512.02325.

Ren, S., He, K., Girshick, R. and Sun J. (2015), Faster R-Cnn: Towards real-time object detection with region proposal networks. In: *28th Int. Conf. on Neural Information Processing Systems*, pp. 91–99.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R. and LeCun, Y. (2013), Overfeat: Integrated recognition, localization and detection using convolutional networks, arXiv:1312.6229.

Springenberg, J. T., Dosovitskiy, A., Brox, T., & Riedmiller, M. (2014). Striving for simplicity: The all-convolutional net. arXiv preprint arXiv:1412.6806 .

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, Vol. 15, No. 1, pp. 1929-1958.

Szegedy, C., Toshev, A. and Erhan, D. (2013), Deep neural networks for object detection. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 2553–2561.

Zeiler, M. D., Ranzato, M., Monga, R., Mao, M. Z., Yang, K., Le, Q. V., Nguyen, P., Senior, A. W., Vanhoucke, V., Dean, J. and Hinton, G. E. (2013), On rectified linear units for speech processing. In: *Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3517–3521.

There is no conflict of interest for all authors.