2020 IMEKO TC-19 International Workshop on
Metrology for the Sea
Naples, Italy, October 5-7, 2020

Lightweight and efficient convolutional neural networks for recognition of dolphin dorsal fins

Gianvito Losapio[1], Rosalia Maglietta[2], Tiziano Politi[1], Ettore Stella[2], Carmelo Fanizza[3], Karin Hartman[4], Roberto Carlucci[5], Giovanni Dimauro[6], Vito Renò[2]

[1] *Department of Electrical and Information Engineering (DEI), Polytechnic University of Bari, Via Orabona 4, 70125 Bari, Italy; gvlosapio@gmail.com; tiziano.politi@poliba.it*
[2] *Institute of Intelligent Industrial Systems and Technologies for Advanced Manufacturing, National Research Council of Italy (CNR STIIMA), Via Amendola 122 D/O Bari, Italy; rosalia.maglietta@cnr.it; ettore.stella@cnr.it; vito.reno@cnr.it*
[3] *Jonian Dolphin Conservation, Viale Virgilio 102 - 74121 Taranto, Italy; carmelo@joniandolphin.it*
[4] *Nova Atlantis Foundation, Risso's Dolphin Research Centre, Rua Dr. A. F. Pimentel 11, 9930-309, Santa Cruz das Ribeiras, Pico, Azores, Portugal; oceanwatch@gmail.com*
[5] *Department of Biology, University of Bari, Via Orabona 4 - 70125 Bari, Italy; roberto.carlucci@uniba.it*
[6] *Department of Computer Science, University of Bari, Via Orabona 4 - 70125 Bari, Italy; giovanni.dimauro@uniba.it*

***Abstract* – The study of cetaceans is of vital importance to infer biological information useful to drive sustainable action plans aimed at preserving the marine environment and its biodiversity. In a recent study, we developed a novel algorithm for the detection of dorsal fins in the context of a fully automated pipeline for the photo-identification of Risso's dolphins. A lightweight convolutional neural network (CNN) architecture was proposed to recognize fins among cropped images, filtering the inputs for the photo-identification algorithm. In this paper, we compare the performances of that custom CNN to another extremely efficient architecture: Shufflenet. Training an efficient classifier is a key effort to speed up the first part of the photo-identification pipeline, enabling the feasibility of large scale ecological studies. The experiment confirms that both architectures provide a robust feature extraction capability for the problem in hand, even with a significantly smaller number of parameters with respect to other popular state-of-the-art CNNs.**

I. INTRODUCTION

Photo-identification of specimens is arguably one of the best non-invasive methods to estimate several parameters which describe a population of cetaceans: abundance, spatial distribution and site fidelity, to name a few. This knowledge is of vital importance to infer biological information useful to drive sustainable action plans aimed at preserving the marine environment and its biodiversity [1–8].

Risso's dolphin is a particularly well-suited species for the use of photo-identification. Analogously to human fingerprints, sub adult and adult individuals exhibit patterns of scarring and variations in dorsal fin shape as long-lasting identifiable natural marks [9].

Two state-of-the-art algorithms for the automated photo-identification of Risso's dolphins have been recently proposed: SPIR (Smart Photo Identification of Risso's dolphin) [11] and NNPool (Neural Network Pool) [12].

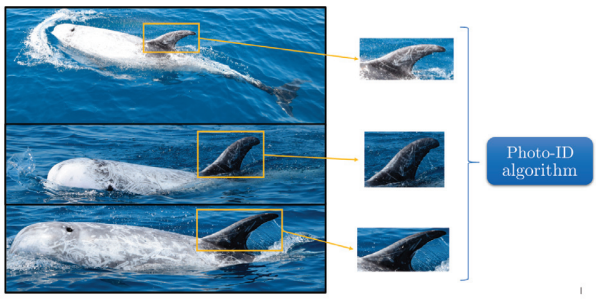

Fig. 1. The crucial role of cropping dorsal fins out of full frame images in the Risso's dolphin photo-identification pipeline.

Similarly to any modern photo-identification algorithm, their effectiveness is guaranteed only if an appropriate area of interest has been designated beforehand in the image. Figure 1 intuitively illustrates such problem. Extracting relevant regions out of full frame images is indeed the main bottleneck in processing data from photographic capture-recapture surveys [13, 14]. For this reason, a novel algorithm has been proposed in our recent work [15] to automatically detect dorsal fins in the context of Risso's dol-

phin photo-identification.

Particular attention was devoted to the design of a convolutional neural network (CNN) classifier, trained to recognize fins among proposed regions with the objective of filtering the inputs for the photo-identification algorithm. A custom CNN was introduced, with a *lightweight* architecture characterized by a lower number of parameters and a reduced computational demand if compared to the other state-of-the-art architectures [16, 17].

In this paper, a new experiment is conducted in order to compare the performances of that custom CNN to another extremely efficient architecture, which recently gained lots of popularity in the field of convolutional neural networks for mobile devices: Shufflenet [18, 19].

Training an efficient classifier is a key effort to speed up the first part of a more ambitious fully automated Risso's dolphins photo-identification pipeline, enabling the feasibility of large scale ecological studies [15].

II. MATERIALS AND METHODS

A. Data acquisition

The data used in this work consist of Risso's dolphins images collected by two private reasearch associations in two different study areas across various time periods:

- $7,881$ pictures taken in the Gulf of Taranto (Northern Ionian Sea) between 2013 and 2018 by marine mammals observers aboard a 40 *ft* catamaran during standardized surveys.
- $2,840$ pictures taken near Pico island (Atlantic Ocean) in 2018 during ocean based surveys, using a 5.8 m long zodiac, equipped with a 50 HP outboard engine.

B. Methodology

The image preprocessing algorithm proposed in [15] using *3D polyhedron-based color segmentation* is exploited to create datasets for the deep learning classifier. Given a set of images $(x_i)_{i=1}^n$ cropped according to this procedure, the following classes are created by manually assigning a label to each sample x_i:

- class *Fin*: images x_i containing at least one clearly visible dorsal fin (label $y_i = 1$);
- class *No Fin*: all the remaining images (label $y_i = 0$).

Figure 2 summarizes such dataset creation procedure $\mathcal{D} = (x_i, y_i)_{i=1}^n$.

The considered classifier - in charge of addressing fins recognition among cropped images - is the following regularized cross entropy minimizer

$$\min_{w,b} \sum_{i=1}^{t} \log \left(\frac{e^{f_{y_i}(x_i;w,b)}}{\sum_{j=1}^{2} e^{f_j(x_i;w,b)}} \right) + \lambda \|w\|^2 \qquad (1)$$

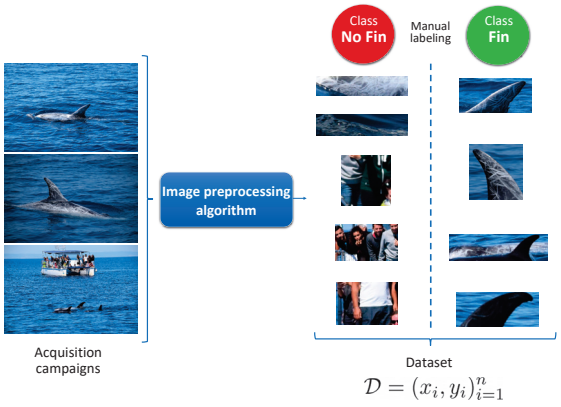

Fig. 2. Dataset creation procedure for the binary classification problem Fin vs No Fin.

where:

- (x_i, y_i) is a sample from the training set $\mathcal{T} \subseteq \mathcal{D}$, $i = 1, \ldots, t$;
- $f(x_i; w, b)$ is the two-dimensional output of a convolutional neural network, x_i being the input and w, b being the weights and the biases considered in the architecture. The subscript f_k is used to denote its k-th entry;
- $\log$ denotes the natural logarithm and its argument represents the softmax function applied to the output layer of the network;
- λ is the L2-regularization parameter used to prevent overfitting.

The minimization problem (1) is solved through an iterative gradient descent algorithm. The canonical binary classification metrics (i.e. accuracy, sensitivity, specificity) are considered to assess performances.

III. EXPERIMENTS AND RESULTS

A. Datasets

Three different datasets were created, with the corresponding sizes reported in table 1:

- a random split was performed on the images taken in the Gulf of Taranto: 80% to be used as the training set $\mathcal{T}$ for the problem 1, while the remaining 20% to be used as a test set (later referred as *Taranto test set*). The percentages were balanced upon each class;
- the pictures from Pico island have been used as a second test set (later referred as *Azores test set*).

Table 1. Number of images contained in each dataset.

	Fin	*No Fin*	Total
Training set $\mathcal{T}$	4,302	3,054	7,356
Taranto test set	1,076	764	1,840
Azores test set	2,411	2,383	4,794

B. Custom CNN

Similarly to the other popular convolutional neural networks [16–18], our custom architecture consists of repeated building blocks with the same structure. Dealing with a relatively straightforward binary classification problem, the peculiar design principle is maximum simplicity and clearness, as already proposed in [23].

The total number of layers is 23. There are three blocks of convolutional layers with small 3×3 kernels combined with the rectified linear unit (ReLU) activation function and a max pooling operation which reduces the block size. Later, three fully connected layers are used to get a final binary prediction out of the extracted features. A more detailed analysis of each single layer is presented in table 2.

The third and last max pooling operation (denoted as *MAXPOOL-3*) was designed to perform a more aggressive downsampling of features with respect to the original architecture proposed in [15]. The effect is to reduce the number of parameters required at the next fully connected layer in order to make the total number of parameters of our CNN comparable to ShuffleNet.

C. ShuffleNet

The pre-trained version of ShuffleNet available in Matlab has been used in our experiment [21]. It is composed of 172 layers, similar to the architecture denoted as ShuffleNet $1\times$, $g = 4$ in the original paper [18].

Addressing the binary classification task in hand with ShuffleNet required changing the network output size, from 1000 to 2. This led to a dramatic drop of the number of parameters in the last fully connected layer, from 545,000 to only 1,090 - resulting in a nearly 40% reduction of the total learnable parameters.

Despite a large number of layers, this architecture exploits the potential of a few specific operations - pointwise grouped convolution, channel shuffling and depthwise separable convolution - to greatly reduce both the computation cost and the number of parameters yet maintaining the accuracy.

D. Learning process

Common training options were used for both CNNs: the *stochastic gradient descent with momentum* algorithm, with a minibatch dimension of 30, a total number of epochs (i.e. full pass of the training set) of 30 and constant learning rate of 0.0003. The regularization parameter was fixed to $\lambda = 10^{-4}$. The following settings are instead specific to each architecture:

- the custom CNN was trained from scratch by using the so-called *Glorot initialization* [22];
- the pre-trained ShuffleNet was fine-tuned with transfer learning. A multiplicative factor of 10 was applied to the learning rate associated to the novel bidimensional output layer, in order to speed up the update of parameters in this layer. On the contrary, the learning rate of the first 10 layers was set to zero, *freezing* the initial parameters to keep the same starting features extraction.

The training process took about 39 minutes for the custom CNN and 68 minutes for ShuffleNet, using the *multiple GPUs* mode offered by Matlab on a laptop equipped with Intel Core i7-8750H CPU operating at 2.20 GHz, 8 GB RAM and Nvidia GeForce GTX 1050 Ti with 4GB of memory as graphics card. The quantitative results are reported in table 3, based on the datasets described in section III.A.

The performances on the Taranto test set are very similar for both architectures. A modest difference can be observed on the Azores test set, where ShuffleNet shows slightly better performance as reported in table 3. Considerably, the accuracy value is greater than 90% for both approaches. Designed with an 86% lower number of layers but with a similar number of parameters, the custom CNN reported a 42% speed up in the training time compared to ShuffleNet. This is a remarkable result because, generally speaking, fine-tuning a network with transfer learning should be faster and easier than training a network from scratch with randomly initialized weights (e.g. the gradients of the frozen layers do not need to be computed). However, our custom CNN outperformed ShuffleNet in terms of time required for the training.

Concerning generalization, the experiment confirms that both CNNs provide a robust feature extraction capability for the problem in hand, even with a significantly smaller number of parameters with respect to the other popular state-of-the-art architectures [16, 17].

IV. CONCLUSION AND FUTURE WORKS

A comparison between two lightweight convolutional neural networks was assessed for the task of dorsal fins recognition in the context of a Risso's dolphins photo-identification pipeline: a custom architecture trained from scratch versus a pre-trained ShuffleNet model. Overall, both CNNs achieved good performances, conferming the efficiency of such lightweight architectures for the binary classification task in hand.

Compared to ShuffleNet, the main advantages of the

Table 2. Quantitative details of the custom CNN architecture. A name is assigned to each layer, according to the following conventions: (i) CONV is used for convolutional layers, MAXPOOL for max pooling layers, FC for fully connected layers; (ii) The first index is used to keep track of an increasing order in which same types of layer appear in the architecture; (iii) The second index of the labels CONV represents the number of kernels. Note that Rectified Linear Unit (ReLU) layers placed after every single CONV layer and FC layer complete the architecture (with the exception of a softmax layer after FC-3).

Layer name	Kernel size	Weights	Bias	Output size
Input	-	-	-	224×224×3
CONV1-16	3×3×3×16	432	16	224×224×16
CONV2-16	3×3×16×16	2,304	16	224×224×16
MAXPOOL-1	2×2	-	-	112×112×16
CONV3-32	3×3×16×32	4,608	32	112×112×32
CONV4-32	3×3×32×32	9,216	32	112×112×32
MAXPOOL-2	2×2	-	-	56×56×32
CONV5-64	3×3×32×64	18,432	64	56×56×64
CONV6-64	3×3×64×64	36,864	64	56×56×64
MAXPOOL-3	8×8	-	-	9×9×64
FC-1	128×5184	663,552	128	1×1×128
FC-2	128×128	16,384	128	1×1×128
FC-3	2×128	256	2	1×1×2

Table 3. Quantitative results of the experiment. Acc, Sens, Spec are short versions for Accuracy, Sensitivity and Specificity, respectively.

	Specifications			Taranto test set			Azores test set		
CNN	Layers	Parameters	Training time	Acc	Sens	Spec	Acc	Sens	Spec
Custom	23	752,530	39 m	95.38	95.91	94.63	90.38	86.15	94.67
ShuffleNet	172	862,802	68 m	95.38	96.37	93.98	93.37	90.54	96.22

proposed custom architecture are a significantly lower number of layers - with benefits in the interpretability of its structure - and a faster training time while mantaining similar generalization properties.

Possible future experiments may consist of: (i) comparing our custom CNN to other efficient state-of-the-art architectures designed for low-cost hardware (e.g. MobileNet [24]; (ii) including in our custom architecture the same efficient convolution operations used in ShuffleNet, still preserving a reduced number of layers.

Finally, a benchmark on real hardware shall be considered, i.e. an off-the-shelf ARM-based computing core, with the ultimate goal of deploying the automated photo-identification pipeline on mobile devices with limited computational power. A very interesting use case is indeed real-time identification of individuals during sighting campaigns.

REFERENCES

[1] Carlucci, R.; Fanizza, C.; Cipriano, G.; Paoli, C.; Russo, T.; Vassallo, P. *Modeling the spatial distribution of the striped dolphin (Stenella coeruleoalba) and common bottlenose dolphin (Tursiops truncatus) in the Gulf of Taranto (Northern Ionian Sea, Central-eastern Mediterranean Sea). Ecol. Indicat.* **2016**, *69*, 707–721.

[2] Carlucci, R.; Baş, A.A.; Liebig, P.; Renò, V.; Santacesaria, F.C.; Bellomo, S.; Fanizza, C.; Maglietta, R.; Cipriano, G. *Residency patterns and site fidelity of Grampus griseus (Cuvier, 1812) in the Gulf of Taranto (Northern Ionian Sea, Central-Eastern Mediterranean Sea). Mammal Res.* **2020**, 1–11. doi:10.1007/s13364-020-00485-z.

[3] Azzolin, M.; Arcangeli, A.; Cipriano, G.; Crosti, R.; Maglietta, R.; Pietroluongo, G.; Saintingan, S.; Zampollo, A.; Fanizza, C.; Carlucci, R. *Spatial distribution modelling of striped dolphin (Stenella coeruleoalba) at different geographical scales within the EU Adriatic and Ionian Sea Region, central-eastern Mediterranean Sea. Aquatic Conserv. Mar. Freshw. Ecosyst.* **2020**. doi:10.1002/aqc.3314.

[4] Azzellino, A.; Fossi, M.C.; Gaspari, S.; Lanfredi, C.; Lauriano, G.; Marsili, L.; Panigada, S.; Podesta, M. *An index based on the biodiversity of cetacean species*

to assess the environmental status of marine ecosystems. Mar. Environ. Res. **2014**, *100*, 94–111.

[5] Pace, D.; Tizzi, R.; Mussi, B. *Cetaceans value and conservation in the Mediterranean Sea. J. Biodiv. Endang. Spec.* **2015**, *2015*. doi:10.4172/2332-2543.S1-004.

[6] Arcangeli, A.; Campana, I.; Bologna, M.A. *Influence of seasonality on cetacean diversity, abundance, distribution and habitat use in the western Mediterranean Sea: implications for conservation. Aquatic Conserv. Mar. Freshw. Ecosyst.* **2017**, *27*, 995–1010.

[7] Nowacek, D.P.; Christiansen, F.; Bejder, L.; Goldbogen, J.A.; Friedlaender, A.S. *Studying cetacean behaviour: new technological approaches and conservation applications. Anim. Behav.* **2016**, *120*, 235–244.

[8] Baş, A.A.; Öztürk, B.; Öztürk, A.A. *Encounter rate, residency pattern and site fidelity of bottlenose dolphins (Tursiops truncatus) within the Istanbul Strait, Turkey. J. Mar. Biol. Assoc. UK* **2019**, *99*, 1009–1016.

[9] Hartman, K.L. *Risso's Dolphin: Grampus griseus.* In *Encyclopedia of Marine Mammals*; Elsevier: 2018; pp. 824–827.

[10] Maglietta, R.; Renò, V.; Cipriano, G.; Fanizza, C.; Milella, A.; Stella, E.; Carlucci, R. *DolFin: An innovative digital platform for studying Risso's dolphins in the Northern Ionian Sea (North-eastern Central Mediterranean). Sci. Rep.* **2018**, *8*, 17185. doi:10.1038/s41598-018-35492-3.

[11] Renò, V.; Dimauro, G.; Labate, G.; Stella, E.; Fanizza, C.; Cipriano, G.; Carlucci, R.; Maglietta, R. *A SIFT-based software system for the photo-identification of the Risso's dolphin. Ecol. Inform.* **2019**, *50*, 95–101. doi:10.1016/j.ecoinf.2019.01.006.

[12] Maglietta, R.; Renò, V.; Caccioppoli, R.; Seller, E.; Bellomo, S.; Santacesaria, F.C.; Colella, R.; Cipriano, G.; Stella, E.; Hartman, K.; et al. *Convolutional Neural Networks for Risso's dolphins identification. IEEE Access* **2020**, 1. doi:10.1109/ACCESS.2020.2990427.

[13] Buehler, P., Carroll, B., Bhatia, A., Gupta, V., & Lee, D.E. (2019). *An automated program to find animals and crop photographs for individual recognition*. Ecol. Informatics, 50, 191-196.

[14] Schofield, D., Nagrani, A., Zisserman, A., Hayashi, M., Matsuzawa, T., Biro, D., Carvalho, S. (2019). *Chimpanzee face recognition from videos in the wild using deep learning*. Science Advances. 5. 10.1126/sciadv.aaw0736.

[15] Renò, V.; Losapio, G.; Forenza, F.; Politi, T.; Stella, E.; Fanizza, C.; Hartman, K.; Carlucci, R.; Dimauro, G.; Maglietta, R. *Combined Color Semantics and Deep Learning for the Automatic Detection of Dolphin Dorsal Fins.*, Electronics 2020, 9, 758. doi:10.3390/electronics9050758.

[16] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Rabinovich, A. (2015). *Going deeper with convolutions*. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).

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

[18] Zhang, X., Zhou, X., Lin, M., Sun, J. (2018). *Shufflenet: An extremely efficient convolutional neural network for mobile devices*. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 6848-6856).

[19] Ma, N., Zhang, X., Zheng, H. T., Sun, J. (2018). *Shufflenet v2: Practical guidelines for efficient cnn architecture design*. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 116-131).

[20] Russakovsky, O., Deng, J., Su, H. et al. *ImageNet Large Scale Visual Recognition Challenge*. Int J Comput Vis 115, 211-252 (2015). https://doi.org/10.1007/s11263-015-0816-y

[21] `https://mathworks.com/help/deeplearning/ref/shufflenet.html`, accessed on May 10, 2020.

[22] Glorot, X., Bengio, Y. (2010, March). *Understanding the difficulty of training deep feedforward neural networks*. In Proceedings of the thirteenth international conference on artificial intelligence and statistics (pp. 249-256).

[23] Renò, V.; Mosca, N.; Marani, R.; Nitti, M.; D'Orazio, T.; Stella, E. *Convolutional neural networks based ball detection in tennis games*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18-22 June 2018; pp. 1758-1764. doi:10.1109/CVPRW.2018.00228.

[24] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko,W.Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: *Efficient convolutional neural networks for mobile vision applications.* arXiv preprint arXiv:1704.04861, 2017.