Improving Right Whale Recognition by Fine-tuning Alignment and Using Wide Localization Network

AbdulWahab Kabani
Computer Science Department
University of Western Ontario
London, Ontario, Canada, N6A 3K7
Email: akabani5@uwo.ca

Mahmoud R. El-Sakka
Computer Science Department
University of Western Ontario
London, Ontario, Canada, N6A 3K7
Email: melsakka@uwo.ca

***Abstract*—Right Whales can be recognized by the callosities pattern on their heads. They are an endangered species with an estimated 450 whales remaining. Marine biologists regularly perform manual recognition of the whales while monitoring the population but the process is slow and time consuming. Deep learning methods achieved state-of-the-art results on several visual recognition tasks. However, training deep learning models on this task is very difficult because the number of training images is low. We propose a wide localization network which can be used to localize the region of interest in image. Once the region of interest is localized, a deep learning model can be used to classify the whales. The solution we describe in this paper achieves an accuracy score of 78.7% and ranks as one of the best 3 solutions on this dataset.**

***Index Terms*—Whale Localization , Whale Detection , Whale Recognition , Deep Learning , Convolutional Neural Network , Localization , Detection , Recognition , Image Classification**

I. Introduction

The North Atlantic Right whale [1], [2] is an endangered species due to harsh hunting. These whales were a preferred target by hunters because of their tendency to live close to the shore, being rich in whale oil, and the fact that their bodies float when killed. After they became a protected species, the main source of decrease in population include collision with ships.

There are several difficulties associated with this dataset. First, the size of each image is huge making it very difficult to fit the data into the GPU without losing important features. Second, the features (whale head callosities) in many images are not very clear due to environmental conditions. Finally, the number of individual whales (the classes we want to predict) is large (447) while the number of training images is limited (4,544 images).

The process of manually recognizing whales involves a marine biologist looking into the head callosities pattern to determine the name (or ID) of the whale. This is because right whales can be recognized using the pattern of the callosities on their heads. Since the head of the whale occupies only a small area in the whole image, localizing the region of interest is very important. Localizing the region of interest can help the classification model focus on the most discriminative features and avoid irrelevant features such as features in the surrounding water. Normally, a deep learning classification model can learn to ignore irrelevant features by training it on large datasets. However, because this dataset [1] is very small, it is important to train the classification model only on the important features.

In this paper, we will start by introducing related work in Section II. Then, we will introduce the overview of our solution in Section III. After that, we will talk about the network that we used for head localization in Section V. In Sections VI, we will describe how to perform head orientation and alignment, respectively. The classification network will be described in Section VII. Finally, the results will be introduced in Section VIII and we will conclude our work in Section IX.

II. Related Work

Deep learning involves stacking several layers of units called neurons in order to achieve a high level of abstraction. Typically, the model extracts several types of features that include low level features and high level features. During training, an objective function is optimized using the back-propagation algorithm [3].

A convolutional neural network [4] (CNN or convnet) is a network that contains convolutional layers. A convolutional layer differs from a fully-connected layer because it has restricted connectivity. This can be very useful because it reduces the amount of parameters that need to be tuned. These types of networks are suitable for many visual recognition problems.

CNNs achieve the state-of-the-art performance on many datasets such as MNIST [5] and the ImageNet large scale classification challenge [6],[7]. For instance, several architectures [8], [9], [10] showed excellent results on the ImageNet challenge [6],[7]. The success was possible thanks to the development in computing power, regularization techniques such as Dropout [11],[12], initialization methods [13], ReLU activations [14], and data augmentation.

The dataset we are reporting our results on is based on a Kaggle competition [2] sponsored by National Oceanic and Atmospheric Administration [1]. Just like a fingerprint, the callosities pattern found on the right whale head can be used to identify each whale. What is unique about this dataset is that it contains a very low number of training images (4,544) while the testing set has 6,926 images. Deep learning solutions

978-1-5090-5538-8/17/$31.00 ©2017 IEEE

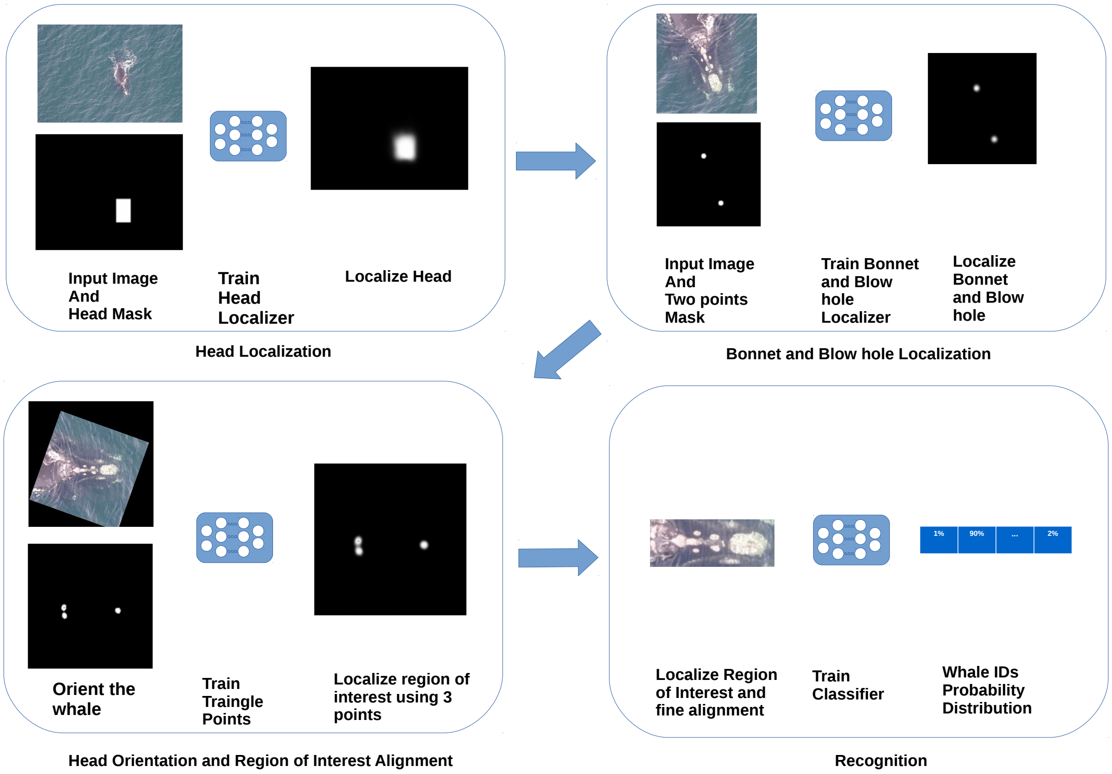

Fig. 1. The overview of the method: the first stage involves performing head localization. The second stage involves identifying the bonnet and blow hole. Using these two points, the head is oriented east. In stage 3, we perform further fine tuning alignment and a bounding box is extracted. Once the region of interest is identified, the region of interest is passed to the recognition network to predict the ID based on the callosities pattern.

are very effective on problems with large datasets. In [15], it was shown that orienting the whale heads with respect to the bonnet and blow hole can alleviate this requirement. Our previous work [16] -which was ranked in the top 5 best solutions on this dataset- relied on that idea and produced head patches. The head patches were then passed to a leaky neural network for classification.

Most of the solutions that ranked high on the leaderboard followed a similar idea as suggested in [15]. Inspired by the work in [17], the team that was ranked in the second position [18] used a multi-stage that involves regressing a bounding box to localize the whale head followed by an alignment process. The classification models were based on the Visual Geometry Group network (VGG) [10] and ResNet [19].

Finally, the team that have the highest score on this dataset [20] used an ensemble of classification models trained on a passport-like images of the whales heads.

III. Method Overview

Training the recognition model on the raw images is unlikely to yield good results. When training the recognition model on the original images, we could not achieve good results due to overfitting. The max accuracy we were able to achieve is around 8%. Since the number of target labels is large (447) and the number of training data is small (4,544), the recognition model will pick up on non-discriminative features from the surrounding water. In addition, when resizing the images to fit the model, the most discriminative features (head callosities) will be very small making it very difficult to recognize the whale. To avoid this, we need to produce head patches that show only the head callosities with as little background as possible.

First, we start by training a wide convolutional neural network (wide net) to localize the head of the whale. Once the network is trained, a bounding box is predicted and used to produce a head image. This is similar to what was suggested in [20]. However, we used a different localization network called wide net. The main advantage of this network is that it has the ability to predict region of interests with different shapes (rectangles, circles, or any arbitrary shaped region of interest). In addition, it has a low memory requirement making it ideal for use as a pre-processing step.

In the second stage, a wide net model is trained to localize the bonnet and blow hole. As suggested in [15], we use these two points to orient the head in one direction. In the third stage, we train a wide net model to predict three points (bonnet and two points representing the post blow hole callosities). These points are used to perform further fine-tuning alignment. In addition, we can use them to extract a very tight bounding

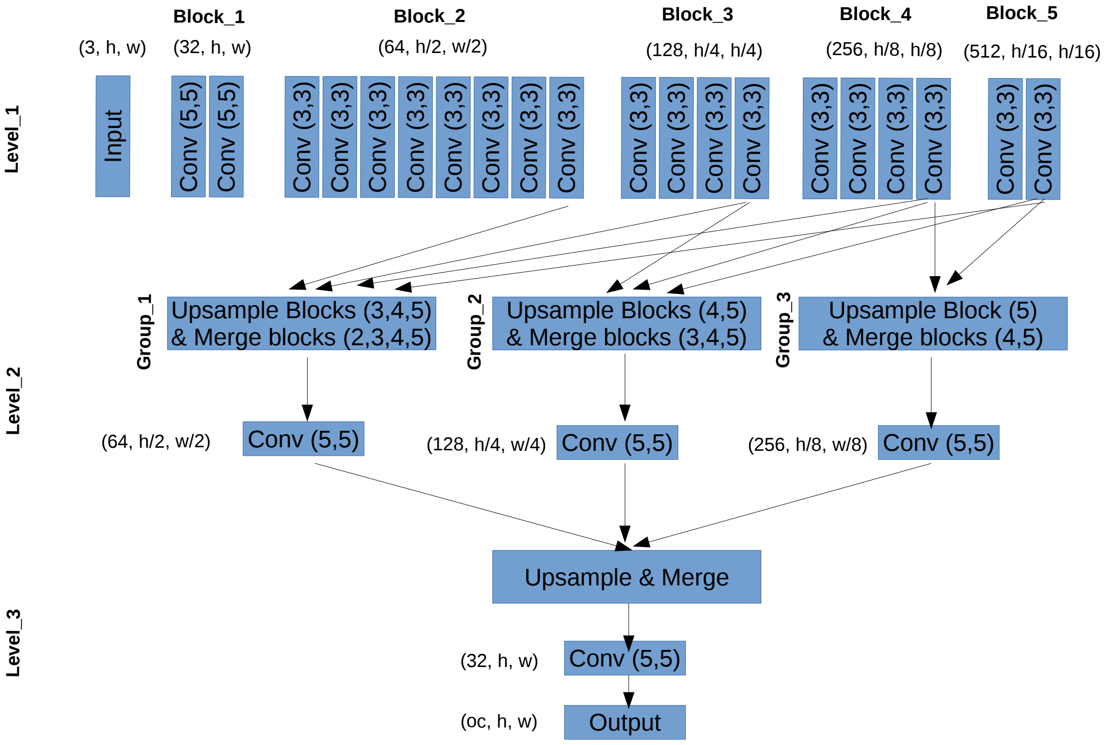

Fig. 2. The architecture of wide net: the network is composed of three levels. Level_1 is just like any typical recognition network. Level_2 involves merging the last layer from each block in level_1 in three different ways. Finally, in level_3, the outputs of level_2 are combined and a mask is produced. Abbreviations: h is the height of the image, w is the width of the image, oc is the output channel size.

box that only shows the most discriminative features.

Once a bounding box is extracted, it is used to train an ensemble of deep learning models to classify and identify each image. What is unique about our solution is that we use deep learning model called the wide net, which we will describe in Section IV. In addition, we perform an extra fine-tuning step after head orientation using three support points.

IV. Head Localization

In order to localize the whale head, we propose an efficient deep learning model called the wide convolutional neural network (wide net). The architecture of this model is shown in Figure 2. This network is capable of performing segmentation and producing region of interests at a very low processing power. The network is composed of three levels: Level_1 is simply a set of convolutional and pooling layers. Level_2 involves merging the last layer from each block in level_1 in three different ways. Three types of merging is done: type_1 involves merging (block_2, block_3, block_4, block_5), type_2 involves merging (block_3, block_4, block_5), and finally type_3 (merging block_4, and block_5). Type_1 carries a combination of high resolution features along with highly abstract features. On the other hand, Type_3 carries only abstract features. Finally, in level_3, the outputs of level_2 are combined and a mask is produced.

The last layer in the network has a sigmoid activation to map the pre-activation values into values between 0 and 1. Rectified linear units (ReLU) activation [14] is used after each layer. In order to extract more abstract features, we use Max-pooling in level_1. As a result, the layers in each block in level_1 will have different sizes. In order to combine them in level_2, we use upsampling to ensure that the layers can be merged. This network has a low processing footprint and was previously tested on a device with only 3.5 GPU memory.

Before passing the image to the network, it is important to standardize the images. We opted for channel-wise mean subtraction and standard deviation division. All images were resized to $128(height) \times 192(width)$. The ground truth for this network involves a mask that contains zero pixels everywhere except in the region of interest.

Because we are using a sigmoid activation in the output layer, our loss function is binary cross-entropy. The learning rate was set at 0.001 and decreased by 50% if the validation error does improve after 20 epochs. The head localization model is trained for 100 epochs. After that, we use it to predict the bounding box for the whale head. We are now ready for the bonnet and blow hole localization.

V. Bonnet and Blow Hole Localization

The bonnet and blow hole localization stage (shown in Figure 1) involves training a deep learning model to locate the two points. This was proposed in [15] as a means to speed up the training process. We also noticed that it is useful in reducing the amount of pixels that represent the surrounding water. These pixels can lead to overfitting because the training set is not large enough. Therefore, the model may be tricked into thinking that these pixels influence the kind of target we are interested in.

The training process and network architecture (Figure 2) is exactly the same as we described in Section IV. However, the input images were resized into $128(height) \times 128(width)$. In addition, this network will output a mask with 2 channels. One channel produces the location of the bonnet while the other channel predicts the location of the blow hole.

Once these two points are localized, we can easily rotate the images. This is because the angle of the whale with respect to the x axis can be estimated by Equation 1.

$$\theta = tan^{-1}\left(\frac{y_{bonnet} - y_{blowHole}}{x_{bonnet} - x_{blowHole}}\right), \qquad (1)$$

VI. Head Orientation and Region of Interest Alignment

After orienting all whale heads into one direction, we train the same localizer network described in Section IV and VI to detect three points (the bonnet and the two points that represent the post blow hole callosities). We perform this step for two main reasons. First, these points can be used to perform even more fine tuning alignment than we achieved in the previous step. Second, these points can help in extracting very small region of interest which is what we are trying to achieve. The average distance between the bonnet and the two post blow hole callosities is the width of the region of interest. The distance between the two post callosities points is the width of the image.

VII. Recognition

As mentioned earlier, right whales can be recognized using the callosities patterns on their heads. We used an ensemble of two architectures (a VGG-like [10] structure without the fully connected layers and a simpler architecture). These two architectures are shown in Figure 3. The ensemble includes training these two networks with three kinds of non-linearities: leaky non-linearity [21] with leakiness value of 0.1, leaky non-linearity [21] with leakiness 0.3, and a rectified linear units [14]. We set the dropout rate [11] at 75%. All layers were initialized randomly. Table I shows the kind of data augmentation that we used to alleviate overfitting. All networks were trained for 400 epochs, if the validation error does not improve after 25 epochs, it is automatically decreased by 50%.

VIII. Results

As mentioned earlier, the training set has 4,544 images while the testing set has 6,925 images. We extracted 10% of the training set and create a validation set so that we can monitor the training process. The number of images for each class varies from 1 training image per class to 47 training images per class. There are 24 classes with only 1 training image. In order to process the images, we used scikit-image [22] and openCV [23]. The deep learning models were developed using Theano [24], [25] and Keras [26].

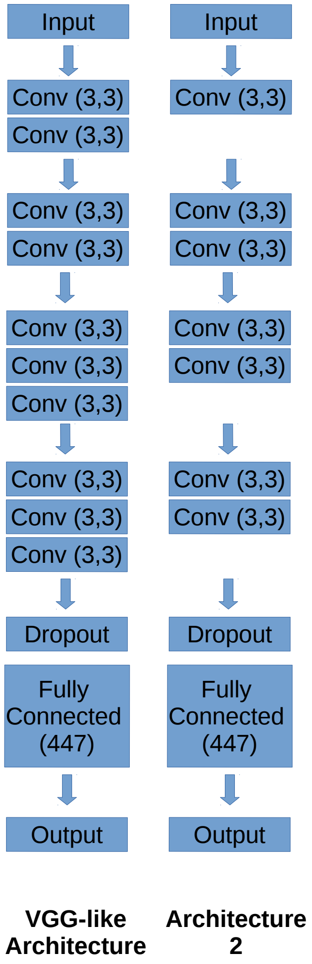

Fig. 3. Recognition Architectures: we use two architectures. One architecture is a VGG-like structure [10] but without the fully connected layers. The second architecture is a simpler architecture with less layers.

TABLE I
Recognition Data augmentation: Random transformations along with parameters. These transformations are applied randomly to each image before sending it to the GPU.

Transformation	Parameters
Rotation	Angle between -10 and +10
Horizontal Flip	Randomness=50%
Vertical Flip	Randomness=50%
Horizontal Shift	Up to 19 pixels
Vertical Shift	Up to 12 pixels
Gaussian Blurring	Up to $\sigma = 1$
Contrast Rescaling	Randomly stretch/shrink intensity

TABLE II
RESULTS OF ALL MODELS WE DESCRIBED IN THE PAPER. ALL RESULTS ARE REPORTED ON THE VALIDATION SET UNLESS MENTIONED OTHERWISE. THE TEST SET LOG LOSS IS REPORTED AS DISPLAYED BY THE EVALUATION SERVER. BECAUSE WE DO NOT HAVE THE GROUND TRUTH, WE CANNOT REPORT THE ACCURACY.

Model	IoU	Average Distance (in pixels)	Log Loss	Accuracy
Head Localization	71.8%	2.41	_	_
Bonnet and Blow hole	_	(bonnet: 2.02, blow hole: 2.64)	_	_
Bonnet and Two Post-callosities Points	_	(Bonnet: 1.26 Point_1: 2.01 Point_2:1.90)	_	_
Recognition (Validation Set)	_	_	1.16	78.7%
Recognition (Test Set)	_	_	1.10	Unknown

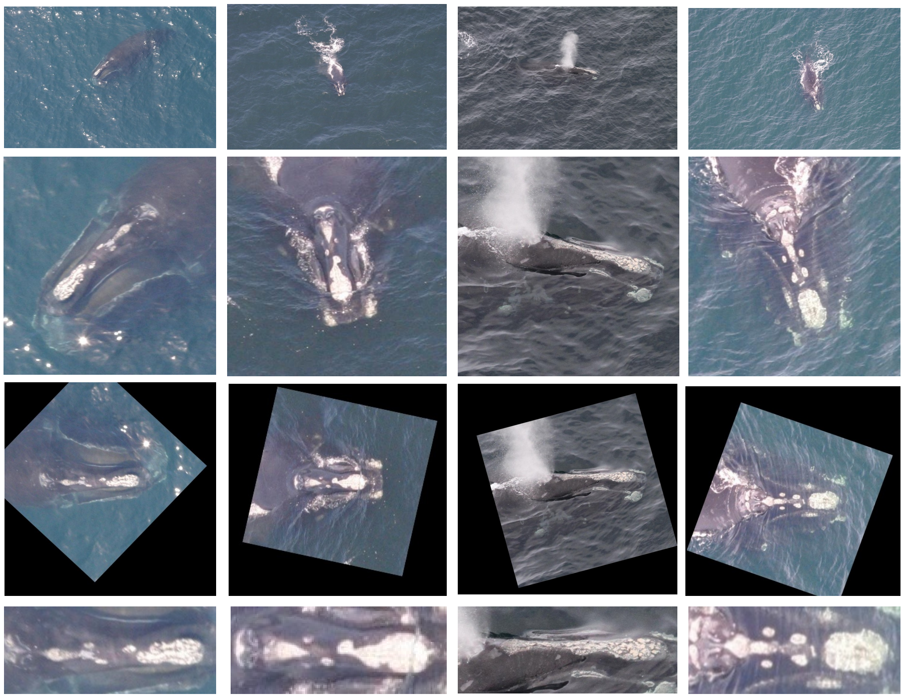

Fig. 4. A sample of input images along with the localization after each stage.

Figure 4 shows the sample output of each stage. Training the localization networks took around 5 hours per network. Training the recognition networks took around 3 hours per network. We trained an ensemble of 12 networks. Table II shows a summary of the results that we achieved. The head localization model achieved an intersection over union score (IoU) of 71.8%. The average absolute distance error from the center of the true bounding box to the predicted one is 2.41 pixels. The log loss score (lower is better) of the recognition ensemble is 1.16 on the validation set and 1.10 on the test set as reported by the evaluation server. The accuracy score on the validation set is 78.7%. We cannot compute the accuracy on the test set because the ground truth is not provided. Table III shows how our method stacks against other solutions. The table only shows the top 10 teams out of 364 teams. The solution we described in this paper ranks in the top 3 positions. It is worth noting that the solution that was ranked in the fifth position is our old method which we described in [16].

IX. CONCLUSION

Deep learning can be used on problems with large training data. We proposed a wide localization network which can be used to localize a region of interest in an image. We demonstrated how it can be used to localize the callosities pattern. After that, we trained an ensemble of deep learning models to classify the images.

ACKNOWLEDGMENT

This research is partially funded by the Natural Sciences and Engineering Research Council of Canada (NSERC). This support is greatly appreciated.

We would also like to thank Kaggle and the National Oceanic Atmospheric Administration Fisheries for providing this data set. Also, we would like to thank Nervana Systems for sharing their approach and annotating the bonnet and the blow hole locations. Their contribution helped us complete this research.

TABLE III
TEAMS RANKING: THIS TABLE SHOWS THE RANKING OF THE SOLUTION WE DESCRIBE IN THE PAPER. THE SOLUTION RANKED IN THE $3^{rd} position$. THE TABLE ONLY SHOWS THE TOP 10 TEAMS WHILE THE NUMBER OF TEAMS THAT PARTICIPATED IN THE COMPETITION IS 364. THE TEAM THAT RANKED IN THE FIFTH POSITION IS OUR OLD SOLUTION WHICH WE DESCRIBED IN [16].

Ranking	Team Name	Score
1	deepsense.io [20]	0.59600
2	felixlaumon [18]	1.07585
Post_competition	**ours**	**1.1025**
3	SKE	1.14982
4	threedB	1.33648
5	AbdulWahab [16]	1.46909
6	Tsakalis Kostas	1.51900
7	bawdyb .	1.55823
8	Left Whales	1.75764
9	Anil Thomas [15]	1.80178
10	Doug Koch	2.13797

REFERENCES

[1] C. Khan, P. Duley, A. Henry, J. Gatzke2, and T. Cole1, "North atlantic right whale sighting survey (narwss) and right whale sighting advisory system (rwsas) 2013 results summary," *US Dept Commer, Northeast Fisheries Science Center Reference Document*, pp. 14–11, 2014.

[2] Kaggle, "Right whale recognition," https://www.kaggle.com/c/noaa-right-whale-recognition, [Online; accessed 19-January-2016].

[3] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Cognitive modeling*, vol. 5, no. 3, p. 1, 1988.

[4] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.

[5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[7] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[8] 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.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *arXiv preprint arXiv:1409.4842*, 2014.

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

[11] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[12] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[13] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[14] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.

[15] A. Thomas, "whale-2015," https://github.com/anlthms/whale-2015, 2015, [Online; accessed 19-January-2016].

[16] M. El-Sakka and A. Kabani, "North atlantic right whale localization and recognition using very deep and leaky neural network," *Mathematics for Applications*, vol. 5, no. 2, pp. 155–170, 2016.

[17] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[18] F. Lau, "Recognizing and localizing endangered right whales with extremely deep neural networks," http://felixlaumon.github.io/2015/01/08/kaggle-right-whale.html, [Online; accessed 04-August-2016].

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[20] R. Bogucki, "Which whale is it, anyway? face recognition for right whales using deep learning," http://deepsense.io/deep-learning-right-whale-recognition-kaggle/, [Online; accessed 04-August-2016].

[21] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML*, vol. 30, 2013.

[22] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, and the scikit-image contributors, "scikit-image: image processing in Python," *PeerJ*, vol. 2, p. e453, 6 2014. [Online]. Available: http://dx.doi.org/10.7717/peerj.453

[23] G. Bradski, "Opencv," *Dr. Dobb's Journal of Software Tools*, 2000.

[24] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio, "Theano: new features and speed improvements," Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

[25] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio, "Theano: a CPU and GPU math expression compiler," in *Proceedings of the Python for Scientific Computing Conference (SciPy)*, Jun. 2010, oral Presentation.

[26] F. Chollet, "keras," https://github.com/fchollet/keras, 2015.