Humpback Whale Identification

Identifying a whale by its tail

Vitalii Duma
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

Serhii Tiutiunnyk
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

***Abstract*—This document is a Machine Learning project report dedicated to the humpback whale identification problem which was set as a Kaggle competition "Humpback Whale Identification". The approach used for solving the problem is a Siamese Neural Network [1] with some modifications and heuristics.**

***Index Terms*—Siamese Neural Network, image embedding**

I. Introduction

A. Importance of the problem

The problem of recovering whale populations is still actual nowadays due to the adapting to warming oceans, intense whaling during the last several centuries and competition with the fishing industry for food.

The scientists use the photo from surveillance system to register the whales activity and migration. They use the shape of the whales' tails and special marks there to identify the species of the whale.

The vast majority of this work is being done manually by scientists. So, the goal of this project is automation whale identification which will improve the monitoring process, help to get rid of routine job and increase the scientists' performance.

B. Potential impact

Solution of this problem can be applied for migration monitoring of other animals which might help scientists to take care about endangered species.

Along with it, some new heuristics and approaches applied to Siamese Neural Network (SNN) can improve existed solutions for other problems, such as:

- One-Shot Image Recognition [1]. It is very similar to this case dataset as the vast majority of classes have only one example. Since siamese networks for the first time study discriminatory functions for a large concrete data set, they can be used to summarize this knowledge and for completely new classes and distributions.
- Pedestrian tracking for CCTV [2]. In this project SNN is being used together with size and position features of images to detect several persons in the camera view area. SNN learns associations between several frames and trajectories.
- Matching resumes to jobs [3]. In this use case, SNN tries to match job offers and candidate's resumes. Here can be applied natural language processes (NLP) to retrieve deep contextual information from offer description and resumes, compare its embedding and force to increase distance between unsuitable pairs.

Siamese Neural Networks are widely used for solving problems in different ares. So some enhancements in solution whale identification task can lead to improvements contiguous tasks which were mentioned above.

II. Data collection and preparation

The data for the whale identification task was provided by the organozators on the competition platform `kaggle.com` [4]. The dataset contains thousands of images of humpback whale flukes. Individual whales have been identified by researchers and given an `id`. The target is to predict the `id` of images from the test set which will be used for evaluation. What makes this task such a challenge is that there are only a few examples per instance of whale `id` (3,000+ whale Ids).

A. Data preparation

The biggest improvement in solving this problem was done by the method of generating appropriate pairs of images for the training CNN. Each batch of images consists of pairs in which half of them match the same whale `id` and another half of pairs of different whales. As it was mentioned earlier, there are only a few image instances per whale `id` so each images was used for both matching the same whales and differentiation. Furthermore, the pictures which differs whales i.e. with different ids, are selected to be difficult to distinguish. The similar approach is widely used in adversarial training [5]. There are several approaches how to evaluate the difference (distance) between the images. The linear sum assignment [6] method was used to find the most difficult pairs of whales images for matching.

Firstly, there are identical images in the training and test set. Some of them are pixel to pixel identical and the others differ by brightness background color, contrast, size etc.

Along with it, there were several methods applied to images to increase the number of samples for the same whale. There

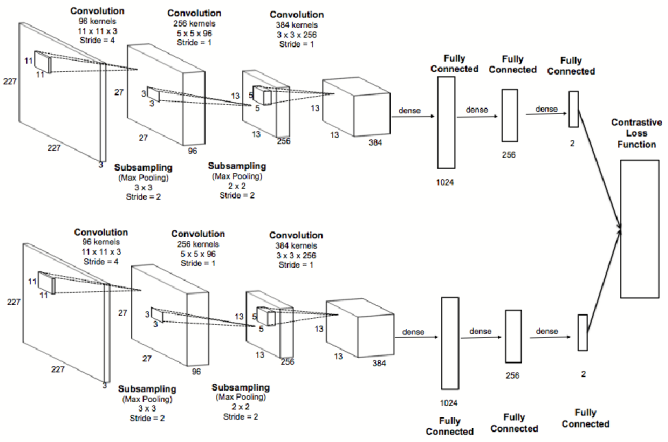

Fig. 1. SNN architecture

are image rotation and affine transformation [7]. Affine transformation maps a rectangular image to a square. In addition, during training data augmentation is conducted by applying a random shift, zoom, shear and rotation.

Finally, the image is normalized i.e. set zero mean and unit variance.

III. Modeling

Siamese neural network consists of two identical Convolutional Neural Network (CNN) which transform the whale images to the vector. It is a general architecture represented on Fig. 1. The input of the model is two images of whales. The CNN is trained to increase the distance between outcome vectors if the input images have different ids i.e. belong to different whales. In the other words, the same CNN with the same weights is being used for both input images.

By testing each image from the test set with each image from the training set it is possible to determine the most likely whales by sorting them by the probability of a match.

The part of model which compares output vectors is a field for experiments. The simplest way is just cosine distance measure with threshold and a loss function. There are pros and cons of applying the distance measure.

First of all, the distance as a measure of vectors similarity is intuitively understandable. Moreover, swapping the the order of input images does not change the result which is also makes sense.

Along with it, there some drawback of applying the distance measure.

- The distance measure will recognize two objects with a zero values as a perfect match of two images.
- At the same time, two vectors with large but a little bit different values will be considered as good match but not perfect as they are not equal.
- Also distance measure makes applying ReLU (Rectified Linear Unit) activation inappropriate as it zeros negative values which leads to loosing some part of information and force network to work only in the positive part of the space.
- The distance measure does not provide a negative correlation between result vectors.

There were several experiments conducted to choose the embedding vectors comparison.

- Cosine distance measure with threshold built into loss function.
- Difference between vectors passed to the input of logistic regression.
- More complicated transformations under the vector's features like sum, product, absolute difference, squared difference etc. passed to the logistic regression.
- The same transformation as in the previous point but passed to the neural network.

The last configuration showed the best results. There was the same output neural network used for each feature parallely like it is on the convolutional stage for images. The output neural network learns weighs between matching zeros and close non-zero values.

The output of the network is a logistic regression.

A. CNN model

The CNN consists from 6 blocks as the training data is not big enough to increase the number of parameters and still keep the model expressive enough. Each block processes maps with lower and lower resolutions, with intermediate layers of the pooling layers.

- Block 1 has a single convolution layer with stride 2 followed by 2x2 max pooling. Because of the high resolution, it uses a lot of memory, so there is a minimum of work done to save memory for subsequent blocks.
- Block 2 has two 3x3 convolutions similar to VGG. These convolutions are less memory intensive then the subsequent ResNet blocks, and are used to save memory. Note that after this, the tensor has dimension 96x96x64, the same volume as the initial 384x384x1 image, thus we can assume no significant information has been lost.
- Blocks 3 to 6 perform ResNet like convolution. I suggest reading the original paper, but the idea is to form a subblock with a 1x1 convolution reducing the number of features, a 3x3 convolution and another 1x1 convolution to restore the number of features to the original. The output of these convolutions is then added to the original tensor (bypass connection). I use 4 such subblocks by block, plus a single 1x1 convolution to increase the feature count after each pooling layer.
- The final step of the branch model is a global max pooling, which makes the model robust to fluke not being image always well centered.

References

[1] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov. "Siamese neural networks for one-shot image recognition." ICML Deep Learning Workshop. Vol. 2. 2015.

[2] Laura Leal-Taixe, Cristian Canton-Ferrer, and Konrad Schindler. "Learning by tracking: Siamese CNN for robust target association." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2016.

[3] Maheshwary, Saket, Hemant Misra. "Matching Resumes to Jobs via Deep Siamese Network." Companion of the The Web Conference 2018 on The Web Conference 2018. International World Wide Web Conferences Steering Committee, 2018.

[4] Humpback Whale Identification competition dataset https://www.kaggle.com/c/humpback-whale-identification/data

[5] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, Ken Nakae1,Shin Ishii "Distributional smoothing with virtual adversarial training" Graduate School of Informatics Kyoto University.

[6] Burkard R.E., Derigs U. (1980) The Linear Sum Assignment Problem. In: Assignment and Matching Problems: Solution Methods with FORTRAN-Programs. Lecture Notes in Economics and Mathematical Systems, vol 184. Springer, Berlin, Heidelberg

[7] Martin G.E. (1982) Affine Transformations. In: Transformation Geometry. Undergraduate Texts in Mathematics. Springer, New York, NY