Classifying Plankton Species with Deep Learning and Computer Vision

Matthew Graham[1,2], Gavin Gray[1,2], Scott Lowe[1,2], Finlay Maguire[3,4], Alina Selega[1,2] and Dragos Stanciu[1,5]

THE UNIVERSITY of EDINBURGH informatics

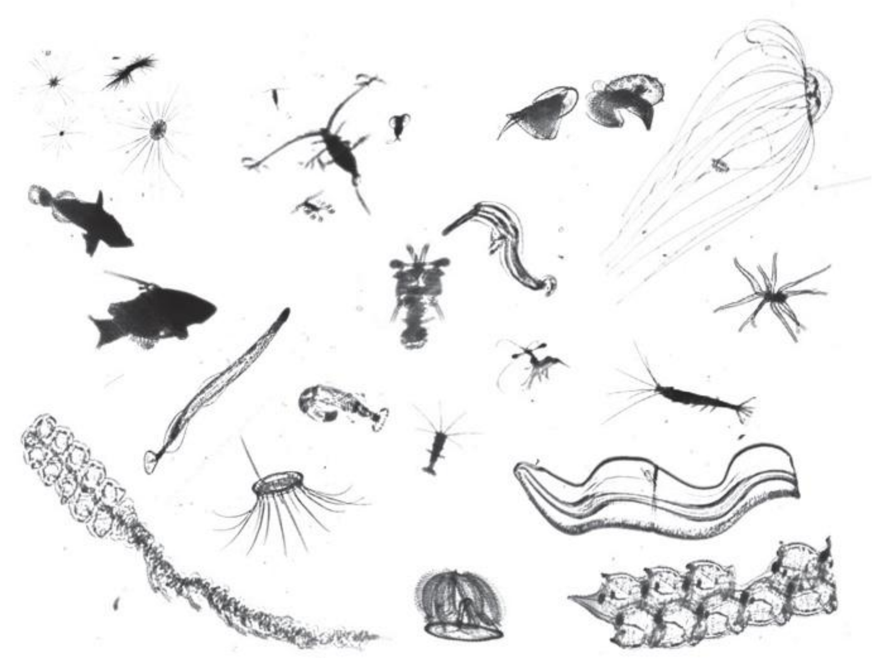
1. Neuroinformatics DTC, Sch. of Informatics, U. of Edinburgh, Edinburgh, UK. 2. ANC, Sch. of Informatics, U. of Edinburgh, Edinburgh, UK.
3. University College London, London, UK. 4. National History Museum, London, UK. 5. IPAB, Sch. of Informatics, U. of Edinburgh, Edinburgh, UK.

UCL NATURAL HISTORY MUSEUM

>INPUT

Plankton are an importand part of the biosphere, comprising half of global carbon fixation. The population composition of plankton is a key indicator of ecosystem function in marine environments. Marine biologists at the Hatfield Marine Science Center, and elsewhere can collect many thousands images of these microscopic organisms every day, but classifying these captured organisms remains time consuming. To automate this process, the researchers sought a solution through a machine learning competition, hosted through Kaggle.

Raw Data: Plankton Images



The dataset contained 30,336 labelled and approximately 20,000 unlabelled images of plankton, classified into 121 different groups.

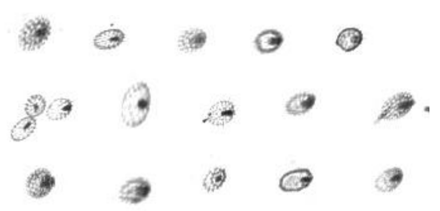
Problematically, the dataset featured a large number of classes, an imbalanced distribution of class sizes (9 to 2000 examples), and a wide range of image sizes (from <40 to >400 pixels in length).

Test yourself!

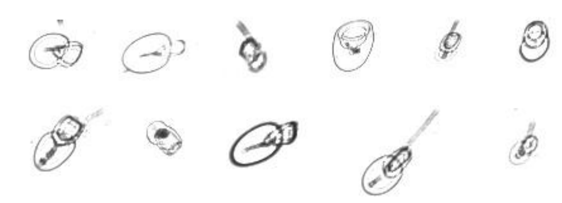
Can you classify the plankton?

Given the two sets of example images as your training data, try to identify which group each organism in the test images belongs to (answers below).

protist fuzzy olive

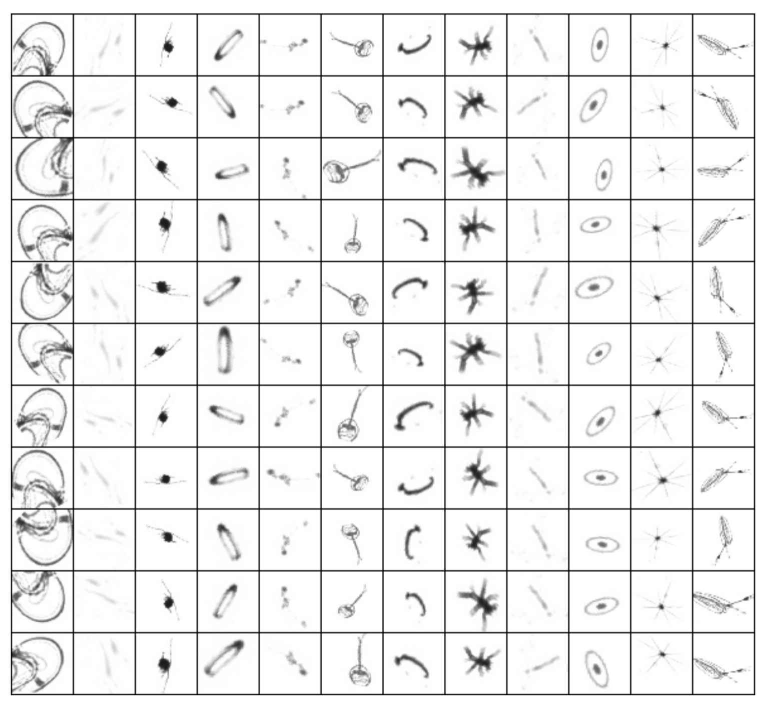


siphonophore calycophoran sphaeronectes young



combined: classify these examples

Augmentation



To prevent the model from overfitting to the training data, we augmented the input images with affine transformations — rotation, reflection, translation, scaling, stretching and shearing. The result is a continuous stream of unique input stimuli.

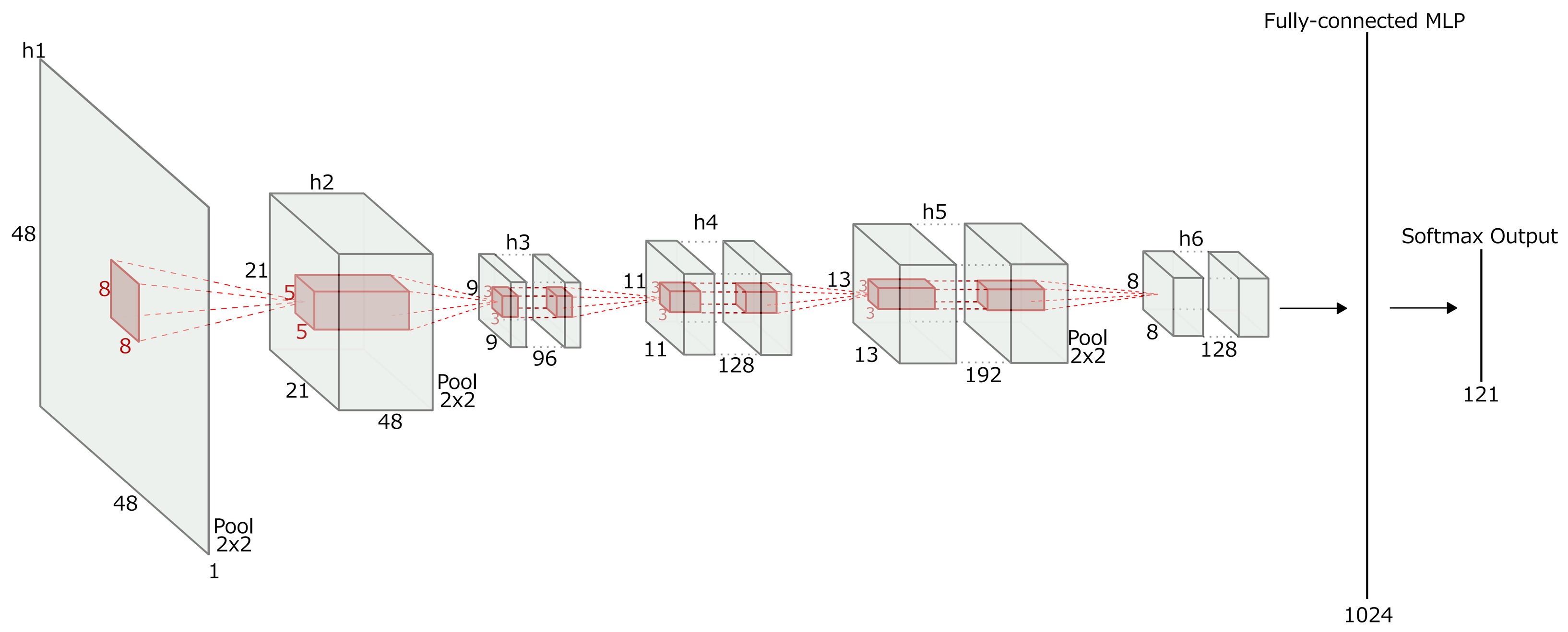
We resized all images to 48x48 to provide a consistent input to the convnet.

EPSRC Engineering and Physical Sciences Research Council · MRC Medical Research Council · BBSRC · kaggle · NATIONAL DATA SCIENCE BOWL

>MODEL

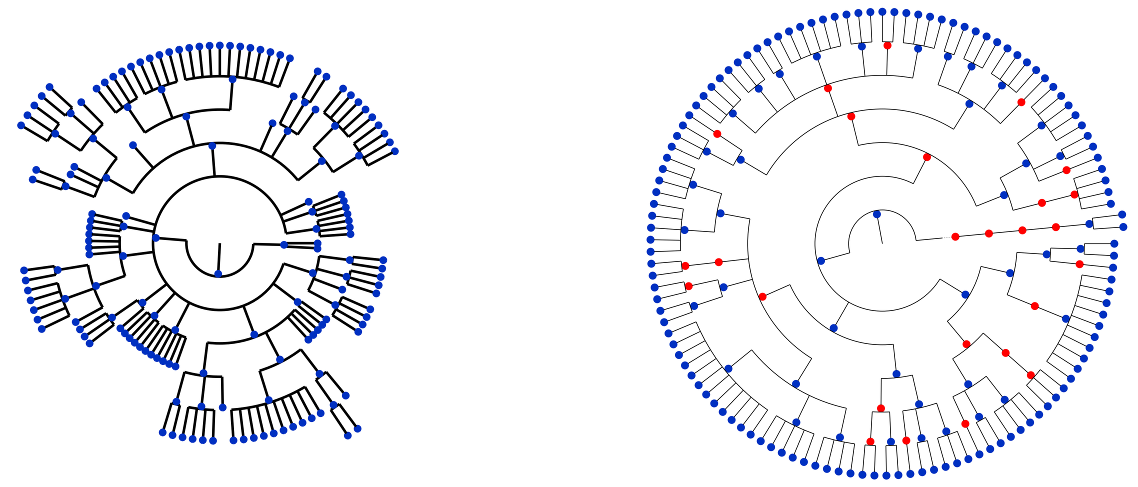
Convolutional Neural Network

For image classification, convolutional networks are state of the art. Our initial structure was based on the successful AlexNet design. After experimenting with many proposed changes, including siamese networks, deeper networks and alternative pooling strategies, our best performing network differed from AlexNet only by one additional convolutional layer.



Hierarchical Modelling

The classes have an intrinsic hierarchy given by their visual relationships (below left). This could be useful for providing additional information in the back-propogated error signal. We converted the hierarchy into a series of six *1-of-k* vectors (right) which could be supplied to six parallel softmax output layers.



This improved the initial learning rate, but had no impact on the final performance of the trained network.

Computer Vision Techniques

We extracted descriptions of global and local texture features with traditional computer vision techniques.

Global Features
- Haralick features
- Grey-Level Co-occurance Matrix attributes
- Zernike moments
- Parameter-Free Threshold Adjacency Stats
- Contour Moments and Hu Moments
- Within-contour histogram of grey-level intensity

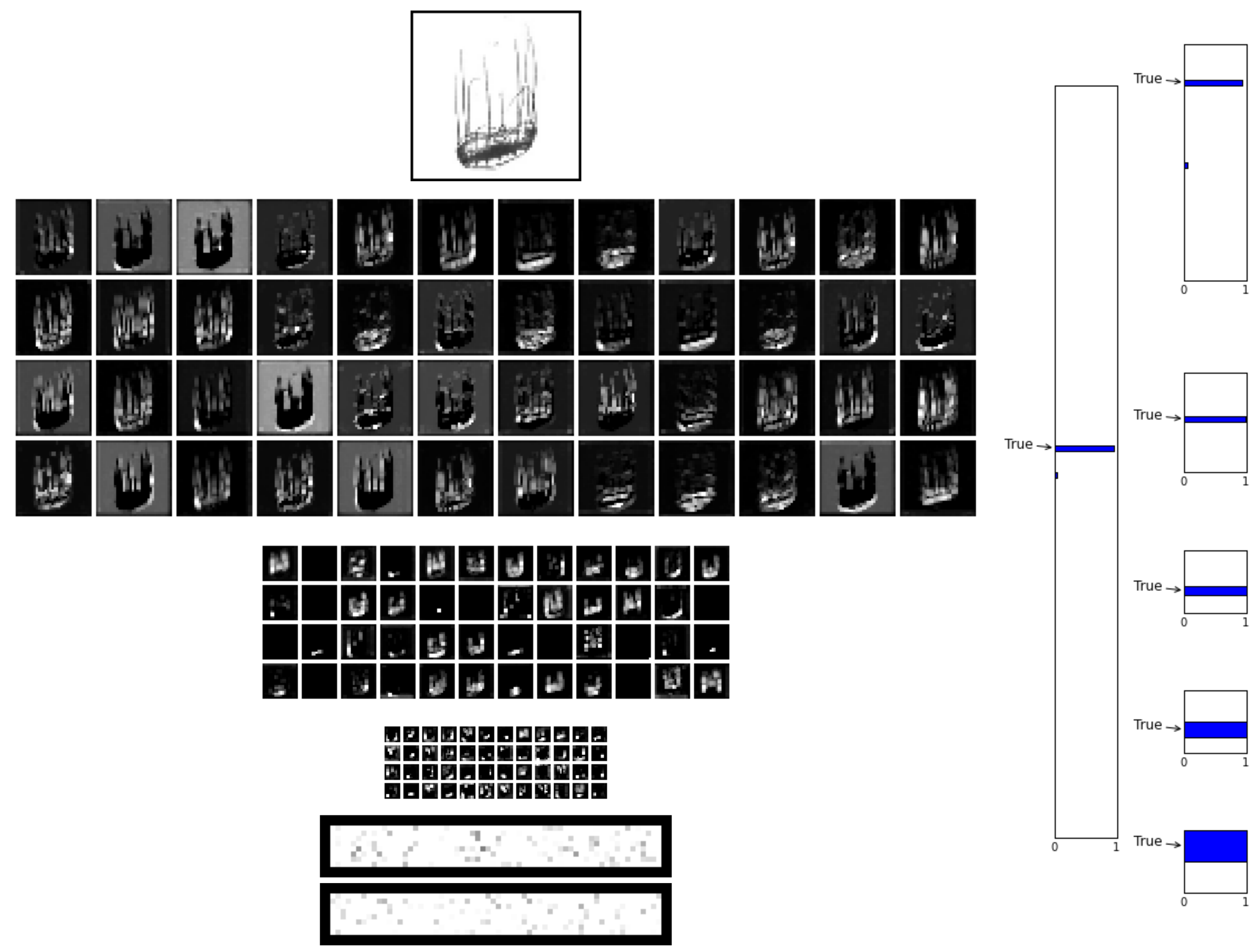
Local Features
- ORB keypoints
- MSER keypoints

Keypoint descriptions were clustered into words, or used to directly classify the image, with the collection of keypoints forming a voting ensemble.

Classifiers built solely with these features as training data did not perform as well as the convolutional networks.
We trained a convnet with CV features alongside training images, but this did not improve our performance.

>OUTPUT

Here we can see the convolutional network classifying an image. Each panel shows the activation of the neurons in response to applying their kernel the feedfoward inputs.



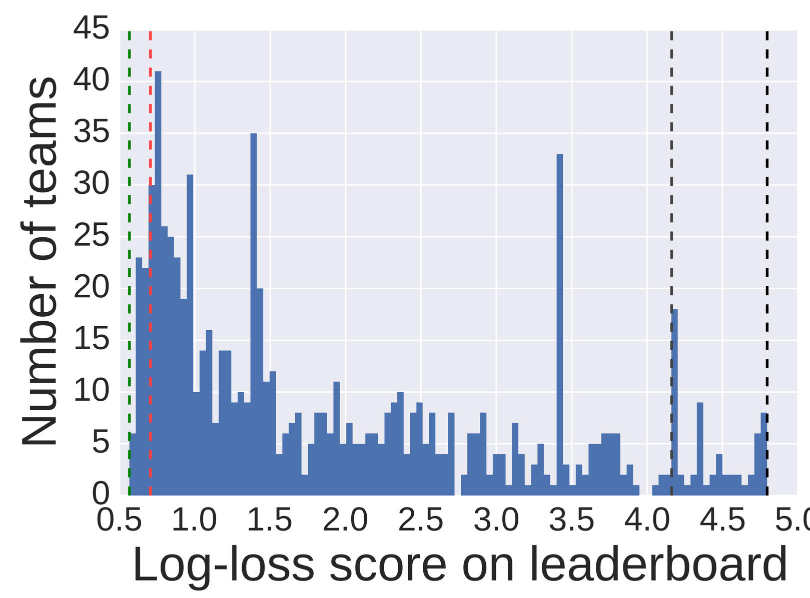
On the right, we can see the output probabilities for each of the classes and superclasses in the hierarchy.

Results

Our final competition submission was an ensemble average of the output probabilities from three networks. Two had the same architecture but different training schedules, and the third was a hierarchically trained model.

Model weights were determined by performing stochastic gradient descent on heldout data.

Our log-loss score of 0.704 placed us 57th out of 1054 entrants on the leaderboard. Right: distribution of better-than-baseline leaderboard scores (green: leader; red: us; grey: prior distribution).



Conclusions

The ideas we attempted were very similar to those used by the winning team. Our implementation was held up due to our inexperience with convnets. Without the time contstraint of the competition, our approach would likely have been able to perform much better.

Acknowledgements

We would like to thank Kaggle, Booz Allen Hamilton and the Hatfield Marine Science Center for organising the competition, and thank Chris Williams, Bob Fisher and Amos Storkey for providing useful comments.