USING DEEP LEARNING TO AUTOMATE BOAT DETECTION FOR FISHING EFFORT ESTIMATES

ISSAM H. LARADJI, MARK SCHMIDT, BRETT T VAN POORTEN, THERESA GODIN, SAUL GREENBERG

MOTIVATION

Estimating angling efforts is an important statistical measure for managing recreational fishing on lakes. One common strategy is to capture lake images over long periods of time and count the number of anglers. However, analyzing lake images takes a vast amount of human effort and time. We are developing a Computer Vision algorithm (based on Machine Learning) that attempts to perform these counts automatically, and here we report on our progress to date.

PROBLEM

Machine learning algorithms often require a vast amount of training samples that are carefully annotated. Since anglers and lakes come in many variations, a machine learning model needs to train on enough samples in order to learn these variations before it can confidently make predictions. As a consequence, we have these 3 main challenges:

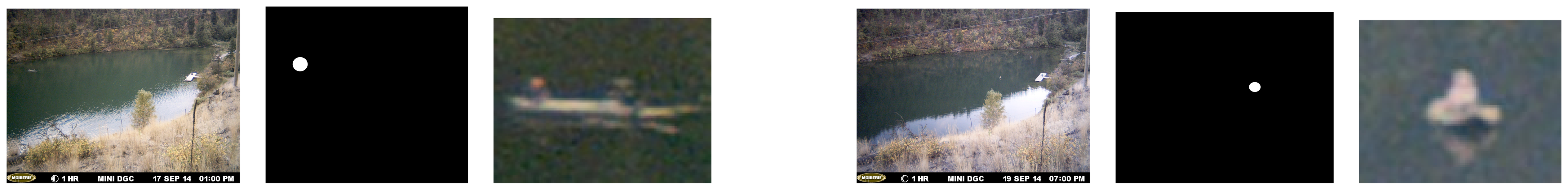
1. Anglers can come in many forms, shapes, poses and sizes.
2. Lakes can change dramatically over a period of time - lightning changes, season changes, etc.
3. Lakes can look significantly different, which makes it difficult for a model to predict anglers on new lakes.

THE DEEP LEARNING MODEL

The deep learning model used is a fully-convolutional neural network which consists of 4 convolutional and 2 max pooling layers. It takes as input the lake image and outputs a heatmap where the values are between 0 and 1 at each pixel location. Higher values means that it is more likely that there is a boat at the corresponding pixel. The source code, the full description of the model, and a basic running example is available at `https://github.com/IssamLaradji/Boat_Detection_WRFC8`

DATASET

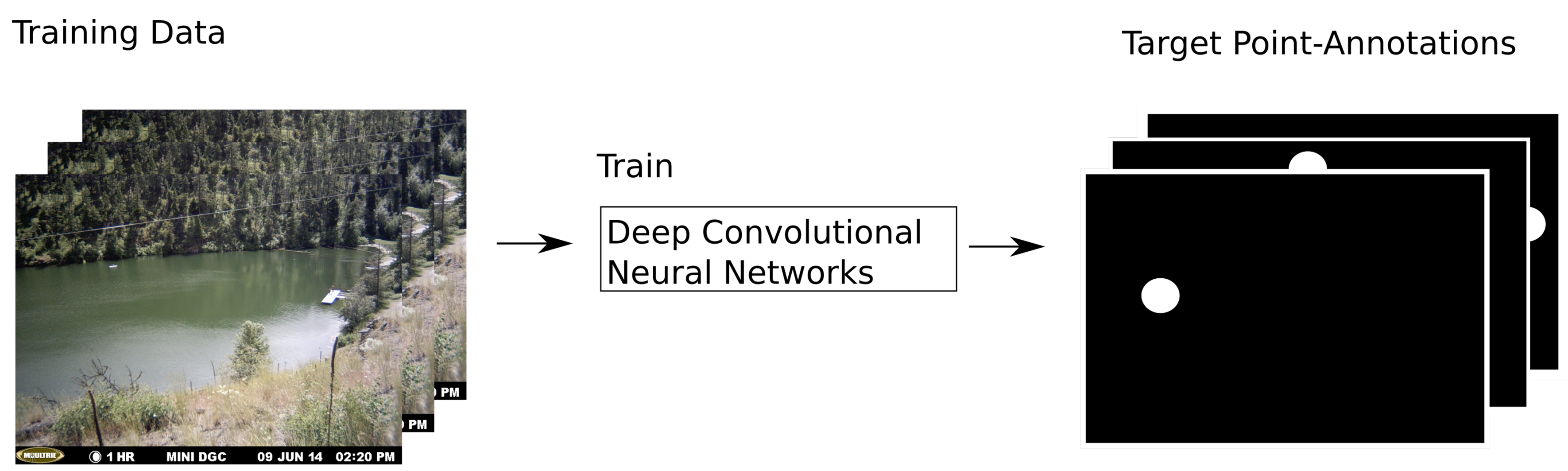
We are given a set of snapshot images of a lake that were taken every hour. The labels are given as point-annotations. These annotations indicate where the anglers are in the image.



Two lake images, their corresponding point-annotations, and the boats at close up.

TRAINING PHASE

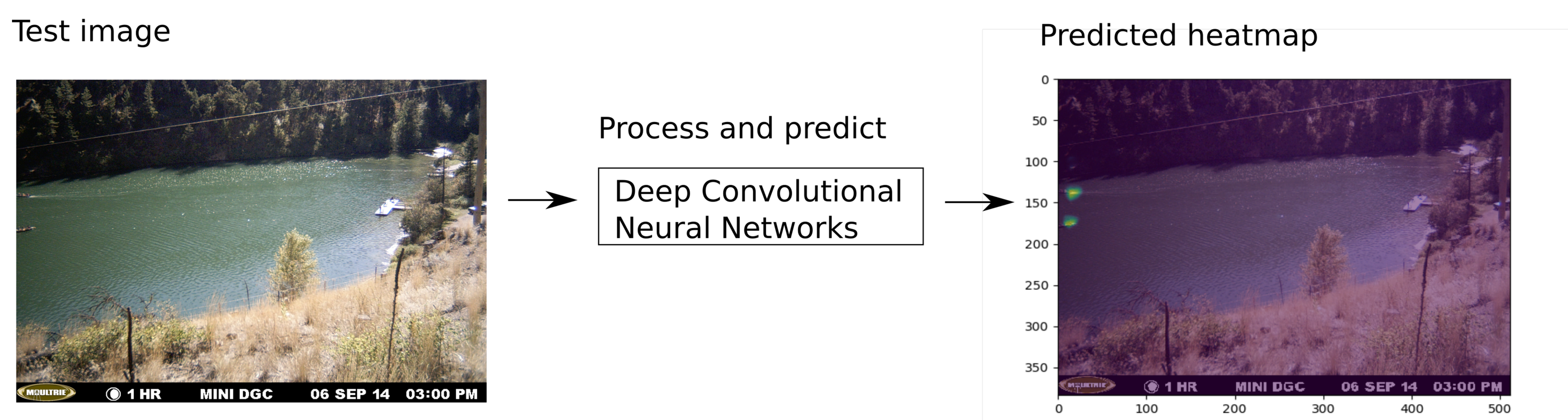
The model, which is a fully-convolutional neural network, trains on a set of lake images and optimizes its predictions so that they match the given target point-annotations.



The model learns to predict the locations of the boats in the annotated images.

PREDICTION PHASE

After training, the model is given an unseen lake image for which it predicts the boat locations. The model outputs a heatmap where each pixel is a value between 0 and 1. The value corresponds to the confidence level as to whether that pixel location belongs to a boat. A value closer to one means the model is more confident that the pixel belongs to a boat.



The greener the colors in the predicted heatmap, the closer the values are to 1. The heatmap shows that the model predicted 2 boats that are on the left side of the lake image.

ACKNOWLEDGEMENTS

We thank MITACS and the Freshwater Fisheries Society of BC for their generous support and we thank the authors for the helpful discussions.

RESULTS

We tested the algorithm on the Yellow Docks lake in BC. We divided the dataset into 3 sets.

1. The training set which contains the first 100 images of the lake - it has 15 boats;
2. the validation set which contains the next 50 images of the lake; and
3. the test set which contains the next 300 images of the lake.

We trained the model (using the GTX 1070 GPU) on the training set and achieved around 10% false positive rate and false negative rate on the test set. We have a false positive when an image patch is considered a boat when there is no boat; and a false negative when a boat has not been identified by the model.

FUTURE DIRECTIONS

Our work to date has exposed many challenges that remain to be overcome, but the progress is promising. Our future work will test the algorithm on a variety of data sets and will analyze images for shore anglers as well as boats. If successful, we expect automation to significantly reduce the workload on human experts, where their job is to confirm or reject predictions rather than laboriously search for and count anglers and boats in lake images. The following is a list of items that we plan to tackle in the near future:

1. Achieve improved performance on more difficult lakes;
2. train a model that can perform well on unseen lakes; and
3. train a model that can predict more than just boats (people, tents, etc.)

If we address the above items, we might integrate the learning algorithm with Timelapse2, an image analyzer for camera traps, which is found here `http://saul.cpsc.ucalgary.ca/timelapse/`