Machine Learning Strategies for Image Based Instream Water Flow Classification

Timothy Becker\textsuperscript{a} and Mary Becker\textsuperscript{b}

https://github.com/timothyjamesbecker/eco_image

\textsuperscript{a} Department of Computing Sciences, University of Hartford, West Hartford CT
\textsuperscript{b} Monitoring and Assessment Program, Connecticut Department of Energy and Environmental Protection, Hartford CT
Altered stream flow

Flow Regime
- Magnitude
- Frequency
- Duration
- Timing
- Rate of Change

Water Quality
Energy Sources
Physical Habitat
Biotic Interactions

Ecological Integrity

1979 minimum stream flow regulation adopted. Diversion Act of 1982 excluded diversions prior to act

2002 Waterbury Vs. Washington

PA 05-142 'An Act Concerning the Minimum Water Flow Regulations' that resulted in adopted stream flow regulations in 2012

Legislators exclude groundwater diversions from regulations

Critical of DEEP for not having a comprehensive list of flow impaired waters

Statewide Water Plan
Traditional stream connectivity studies are resource intensive and only cover localized areas.

CT DEEP developed a method to effectively use images to capture stream connectivity.

This effort began by routinely taking standardized ‘grab sample’ photos resulting in a few hundred images a year.

The next year trail cameras were placed out at a few sites to take 2 photos daily resulting in a few thousand images a year.

In 2018 trail cameras were set out at 24 sites taking hourly photos resulting in over 120,000 images.
**Metadata**

**Standardized File Name**

A folder is created for each deployment with a standardized naming convention.

A tool written in Python is used to rename files and update description tag.
Metadata

Exchangeable Image File Format (EXIF)

Contains meta data about the image such as date/time, camera type, camera name

Developed a tool in Python to perform QA on meta data and extract needed information

'Image Copyright': (0x8298) ASCII=Copyright 2012 @ 286,
'Image DateTime': (0x0132) ASCII=2018:09:19 10:22:40 @ 266,
'Image ExifOffset': (0x8769) Long=830 @ 150,
'Image GPSInfo': (0x8825) Long=1668 @ 162,
'Image ImageDescription': (0x010E) ASCII=MOLTRIE DIGITAL GAME CAMERA @ 182,
'Image Make': (0x010F) ASCII=MOLTRIE @ 214,
'Image Model': (0x0110) ASCII=M-999i @ 230,
'Image Orientation': (0x0112) Short=35 @ 54,
'Image PrintIM': (0xC4A5) Undefined=[80, 114, 105, 110, 116, 73, 77, 45, 48, 51,
'Image ResolutionUnit': (0x0128) Short=Pixels/Inch @ 90,
'Image Software': (0x0131) ASCII=Ver 2.0 @ 258,
'Image XResolution': (0x011A) Ratio=96 @ 242,
'Image YCbCrPositioning': (0x0213) Short=Cosited @ 126,
'Image YResolution': (0x011B) Ratio=96 @ 250,
Image Category labels

Category 1 (Dry)

Category 2 (No Flow)

Category 3 (Minimal Flow)

Category 4 (Well Connected)

Category 5 (Just Below Bankfull)

Category 6 (Above Bankfull)

Category 0 (Can’t Identify Category)
Assess stream connectivity

Incorporate frequency, magnitude, timing and duration into assessment process.
In Progress QA – Human Readers

QA Goal – Independent Reader 10% of Images

Current Status – 2018 ~ 3% of Images QCed

~ 70 % Readers in Agreement

~ 19 % Reader 1 assigned a category Reader 2 assigned category 0 (can not identify category)

~ 11 % Reader 1 and Reader 2 within 1 category (Most discrepancies between categories 3 & 4 and 2 & 3)
Machine learning dataset

Machine learning dataset

<table>
<thead>
<tr>
<th>Category</th>
<th>Images (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>927</td>
</tr>
<tr>
<td>2</td>
<td>1,709</td>
</tr>
<tr>
<td>3</td>
<td>5,086</td>
</tr>
<tr>
<td>4</td>
<td>43,088</td>
</tr>
<tr>
<td>5</td>
<td>6,616</td>
</tr>
<tr>
<td>6</td>
<td>1,061</td>
</tr>
</tbody>
</table>
Classification Problem: Train

large set of labeled images

m-class classification model
Classification Problem: Predict

new unlabeled image

m-class classification model

<table>
<thead>
<tr>
<th>Class</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>0.2</td>
</tr>
<tr>
<td>3</td>
<td>0.1</td>
</tr>
<tr>
<td>4</td>
<td>0.1</td>
</tr>
<tr>
<td>5</td>
<td>0.6</td>
</tr>
<tr>
<td>6</td>
<td>0.0</td>
</tr>
</tbody>
</table>

wrong!
Classification Strategies: (1) 2D-DCNN

each pixel is a feature
Classification Strategies: (1) 2D-DCNN
Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning
Mohammad Sadegh Norouzzadeh, Anh Nguyen, Margaret Kosmala, Alexandra Swanson, Meredith S. Palmer, Craig Packer, Jeff Clune
original images have:
(1) varying aspect-ratio
(2) varying pixel resolution
(3) varying footer overlays

eye need to be:
1) as low-resolution as possible
   (256x256 used in the PNAS paper)
2) uniform from one camera to
   another camera and site

Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning
Mohammad Sadegh Norouzzadeh, Anh Nguyen, Margaret Kosmala, Alexandra Swanson, Meredith S. Palmer, Craig Packer, Jeff Clune
Image Processing: Pixel Resolutions

- 200x112
- 400x225
- 600x200
Image Processing: Labeled Processed

category 1

category 2

category 3

category 4

category 5

category 6
m-class training: partitioning

If you randomly partition the test and training sets, you achieve a super high performance (that is also a terrible predictor given a new site)

(1) Starting with the lowest frequency category, select sites such that they keep the highest consistency with respect to the test and training partition spectrums (select all of site 15244 images)

(2) Select all of a category for a site (so that the training never sees the category 1 from site 15244)
m-class training: balancing

When label spectrums are biased, the training will not learn less frequent categories effectively

(1) Down sample higher frequency labels

(2) Up sample lower frequency labels (requires image manipulation code to on-the-fly alter the images which is known as data augmentation)
m-class training: hyperparameter search

(1) What resolution to use?
(2) Should we use color or grayscale?
(3) How many kernel filters to use?
(4) What batch size to use for training?
(5) When should the weight updates be stopped (epochs)?
(6) What balancing factor should we use?
(7) Should we use data augmentation to up sample?
(8) Can I get better performance by combining or altering categories?

Solution: Search the hyperparameter space!
class training: 2-class f1 = 0.73
class training: 2-class f1 = 0.73
(1) DCNN strategy has limited effectiveness
(2) DCNN needs more data to learn what water is
(3) DCNN learns the site textures more than anything
(4) Certain categories are poorly defined (IE confused)
(5) A different strategy is needed to accurately classify 1 and 6 (they have the fewest data points and contain subtle difference to adjacent categories)
Acknowledgements

Thanks to...

CT DEEP Water Quantity and Monitoring and Assessment program staff for their dedication to high quality collection, data management and image reading.

In particular, Melissa Czarnowski, Chris Bellucci, Ali Hibbard, Laura Robbins and Corinne Fitting.

U.S. EPA Region 1 for providing equipment that aided the collection of data in this study.

University of Hartford for providing grant funding for GPU resources needed for image processing and model development
Joe Spada and Nick Muise for structuring the metadata