ICML 2013 Whale Challenge Method Documentation

Yun-Chiao Lee and Pei-Hao Su

(Big thanks to Sz-Rung Siang and Po-Wei Chou for their help)

Foreword

- We survey and found the key factor of Kaggle winner last year is **Template Matching**!
- Therefore we modified his released code to fit our data
- Surprisingly, the winner of the last year win this year's ICML whale challenge again with same methods and more positive template
- Small conclusion: image processing methods are useful for whale sound detection!

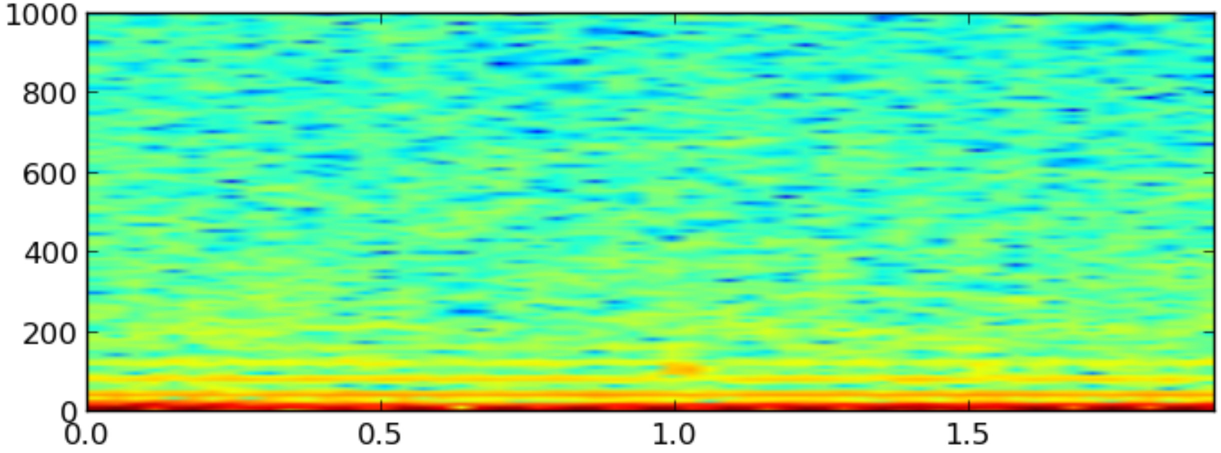
Whale Sound detection

- Feature extraction:
  - Template Matching
- Classification Algorithm:
  - Gradient Boosting Classifier
- Language:
  - python
- Tool:
  - opencv package, scikit learn package

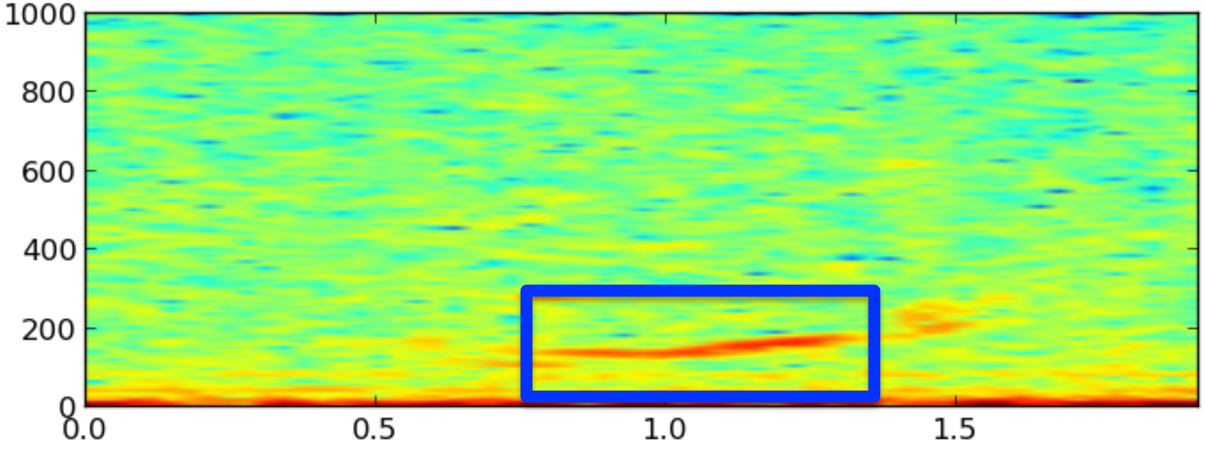
Spectrogram

params = {'NFFT':256, 'Fs':2000, 'noverlap':192}

- Clip without whale sound

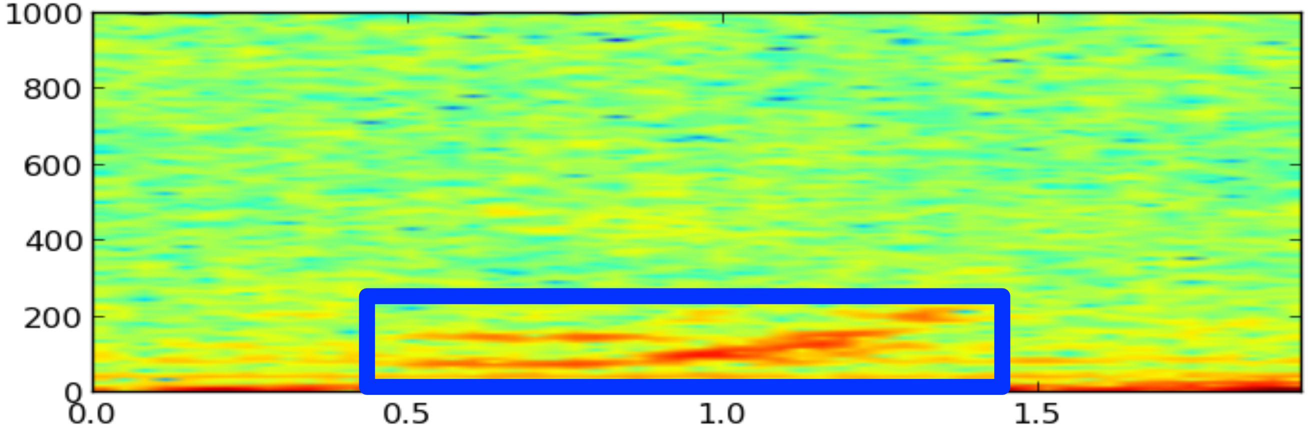


- Clip with whale sound



Preprocessing

- Cut margin
  - We choose clip with whale sound and cut the margin of positive part within spectrogram manually (blue part)
    - Totally 30 manual templates



- Contrast enhancement on each spectrogram to make sound part more explicit

Template Matching

- Up-sweep trajectory in spectrogram
  - Calculate Max correlation and find X,Y location centTime, bwTime, skewTime, tvTime (centroid, width, skew, and total variation)…
- We found that most whale sounds spread at frequency between 100-250Hz
- Some audio clips had very low SNR ratio

Classifier

- 4-fold cross validation
- We tried several machine learning classifiers
  - 3-Nearest Neighbors
  - Linear SVM (C=0.025)
  - RBF SVM (gamma=2,C=1)
  - Decision Tree (depth=5)
  - Random Forest(max_depth=5, n_estimators=10, max_features=1)
  - Naive Bayes
  - LDA
  - QDA
  - Gradient Boosting (ensemble of 15000 decision tree)
- Gradient Boosting yields the best result!