Visual Clues Location

Instructor: Dr. Anshumali Shrivastava, Dr. Krishna V. Palem
Mingchao Jiang, Joseph Hwang, Peter Elmers, Alexander Hansen
{mj33, jzh3, pe4, ajh9}@rice.edu

Abstract

The primary contribution of this project is to attempt to improve known image-based location prediction algorithms by introducing feature weighting and selection to better handle repetitive and spurious features within the images. The two primary approaches are using an Inverse Document Frequency weight on image features and also applying geometric constraints and heuristics to better select features.

Introduction

The goal of this project is to provide a tool to use 1 picture to determine user location with a given 2D map. The tool has a low requirement for camera and image quality. Also, no GPS, sensor data, or cloud computing was involved, which would protect user privacy and save energy. This tool can be used for interactive navigation in public places: hospitals, shopping malls, museums, etc.

Dataset

The dataset was manually collected from the basement of Fondren library using the frontal camera of an iPhone. We developed a 45 ft* 65ft map with 25 observing points and collected images from 8 directions at each observing point for the training set. We chose 12 additional observing points for the testing set (287 pictures in total). The images were manually filtered to remove uninformative images: images of just walls or of extremely close up features.

Model

The initial image location model extracted a set number of Oriented FAST and rotated BRIEF (ORB) features per image. The match score between two images is the number of ORB features in the test image that are less than a threshold hamming distance (learned on the training set) away from the closest feature in a given candidate training image. The predicted location is the location of the training image with the most matches. Yet initial benchmarks with this algorithm provided subpar results due to the occurrence of repetitive features (lights, tiles, bookshelves). To handle this situation, we used two different approaches:

1. **Feature-cleaning**

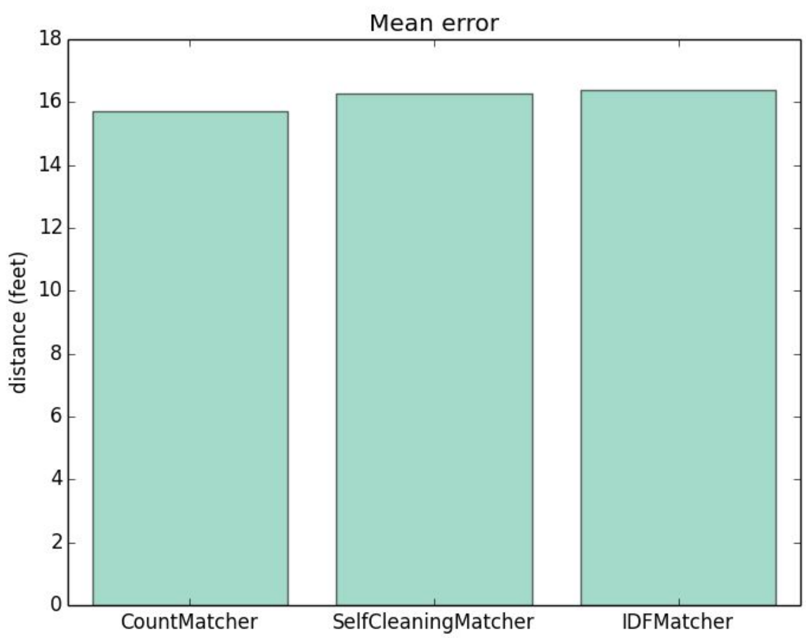
   Match the features in a given image against all the other features in the same image. If a feature has more than a thresholded number of matches it is discarded. The program also performs this same cleaning by discarding features that have too many matches in illegal images. An illegal image is determined by geometric constraints: such as images that were taken facing the opposing direction. As far as we know, this technique has not been applied to image-based location prediction yet.

2. **Inverse Document Frequency Weighting**

Weight each feature by the log of its inverse frequency across the entire corpus of images when summing up the number of matches. This will cause more frequently occurring features to have a lower weight. Though other works have utilized Tf-Idf for image matching (Chum 08), our contribution was applying this technique to image-based location prediction. We also did not use a Bag of Words approach and instead relied on a distance threshold, which ran much faster and provided better results, see figure 3.

Results

We implemented each of the techniques described in the previous section and tested their accuracy on our data set. First we split our images into a training set and a test set. We use the training set to learn error-minimizing parameters and test accuracy on the test set. We compute a model's error as the mean difference between the expected and actual distance from each query point. The *CountMatcher* image matcher implements the initial approach described above. The other matchers (*IDFMatcher* and *SelfCleaningMatcher*) implement the previously described models to find weights on each feature instead of only a global threshold. Again, we learned threshold values on the training set before applying on the testing set to produce results.



Conclusion and Future Work

Unfortunately, even with tuned parameters, our new models performed slightly worse than the naive approach. While the *CountMatcher* achieves average error 15.7 feet, the *SelfCleaning* achieves 16.3 feet and the *IDFMatcher* attains 16.4 feet. Though similar works (Chum 08) have been able to use TF-IDF successfully in image matching, we were unable to show similar results for image-based location prediction. Future tests may be to use these algorithms on larger datasets to see if they perhaps provide better results on those.

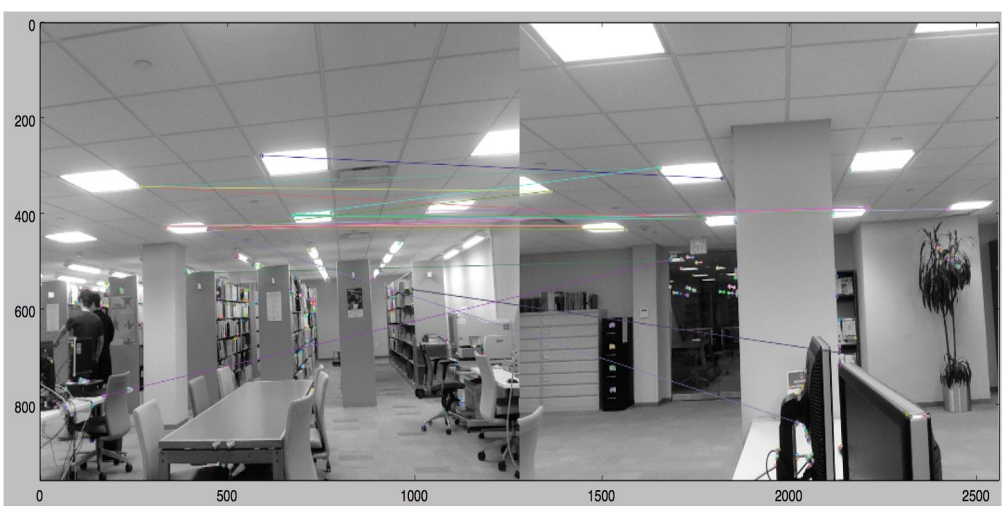
References

[1] Anshumali Shrivastava et. al, “CaPSuLe: A Camera-based Positioning System Using Learning,” 2016.

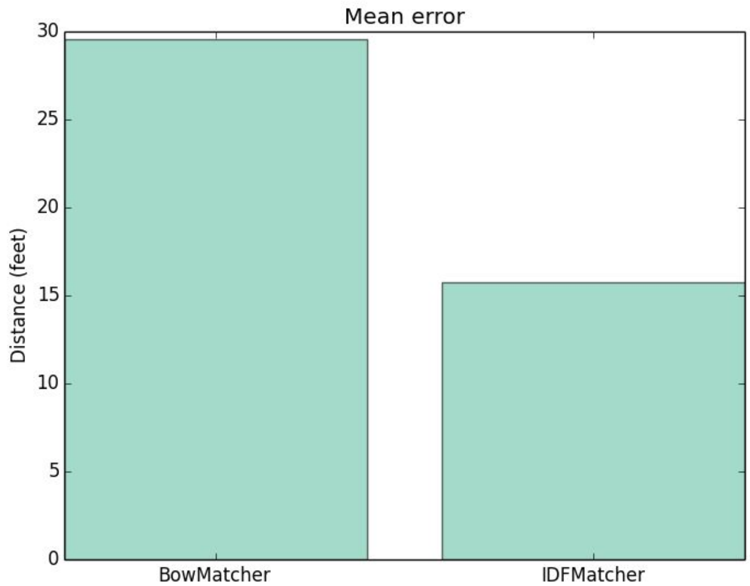
[2] Chum et. al, “Near Duplicate Image Detection: min-Hash and tf-idf Weighting,” 2008.

Appendix

[1] Map of Fondren library and photograph positions



[2] Example of failed match on repetitive features



[3] Bag of Words Matcher vs IDFMatcher mean error