Diving Deeper into IM2GPS

Huda Alamri
School of Interactive Computing
Georgia Institute of Technology
Atlanta, Georgia
Email: halamri@gatech.edu

Julia Deeb
School of Interactive Computing
Georgia Institute of Technology
Atlanta, Georgia
Email: jdeeb3@gatech.edu

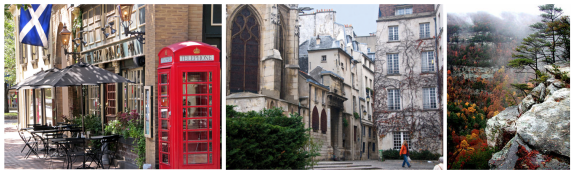

Fig. 1. Where might these photos have been taken? Ans: (from left to right) Savannah, GA; Paris, France; Cloudland Canyon State Park, GA

***Abstract*—Predicting the location an image was taken at is a long standing research problem in computer vision. Given the recent successes with using deep learning techniques, we present a series of approaches towards image geo-location using a variety of deep learning techniques. This report describes our implementation and experimental set-up of training the GoogLeNet and AlexNet architectures on a sample sub-section of the im2gps dataset. Our results achieve competitive performances as compared to the current state of the art. Thus, our major contribution for this report is a methedological bridge between the existing frameworks fro image geo-location.**

Keywords*—computer vision, classification, location, deep learning.

I. Introduction

Consider the images in figure 1 – Where mights these photos have been taken? As a human, we might be able to reason about the possible locations there photos might have been taken. Especially if we have previously been to those areas, as we can use our prior knowledge of the region to reason about that area. Even if we haven't been to that particular area, we can use our knowledge of the world geography and culture to try to guess as to possible locations for such an image. To illustrate this, lets walk through the example established in 1. Consider the left most image - we can see that the image was taken in a urban location which would exclude all remote locations. The flag in the upper part of the image along with the red phone–booth might point us towards Scotland or a region with a large number of Scottish inhabitants. Given these constraints we might correctly identify the image as being taken in Savannah, Georgia – a city with a high number of Scottish descendants.

Now lets consider the middle photo: the contents of the middle photo might lead us to similar conclusion as the first photo - that the photos is taken in an urban location and based on the architecture likely in Europe. Perhaps, one might note that the roofing on the buildings is reminiscent of France. This would be right on both accounts as the photo is taken in Paris, France.

The last image shows a more complicated example to reason about. Without flags and architectural landmarks, it is harder to determine more specifics beyond a forested region. The photo was taken in Cloudland Canyon State Park in Georgia; however, visually there is nothing inherent about the image that might point us towards that location.

While this is a small example, this illustrates the difficulties and complexity of try to geo–locate an image without any context or meta-data. Even the content of the images can be ambiguous or misleading or worse yet too generic to point towards one location or another.

Yet, there are several compelling applications for an automatic system that could do just this. For example, knowing the location of an image would provide context for the image such as average

rainfall, climate, or per–capita income that might otherwise be indistinguishable from images alone.

While humans still struggle with this problem, perhaps computer can leverage the large amounts of data present on-line and use the vast resources available to predict a likely location. The approach we present in this paper attempts to do just that - leverage a large data set gathered from the Internet to build a classifier for specific regions.

II. Background

Estimating the precise geographic location (or even the distribution across likely locations) from visual cues in a photograph is an open area of research in the domain of computer vision. It is a task that requires semantic reasoning about the scene depicted in an image as well as an accurate detector of salient objects that might help determine the location.

We approached this problem as a classification problem; thus our research question is "given an image can we estimate which city it was taken in?". This more closely resembles the research questions from [3] but in our case we look at a broader set of classes.

Our work is heavily inspired by [4] which leverages a large scale Internet dataset of labeled images to determine a likely location of an unknown image. The work in [4] uses traditional, handcrafted features which we hope to extend by using modern deep features. Given the success of the features in the results from [4] as well as the recent successes in computer vision in using deep learning, we believe that we can improve the results by using deep features.

Our intuition is further confirmed by the more recent work in [9] which uses deep learning for the same task. Like our approach, [9] approaches this problem as a classification task - however their approach uses a more sophisticated set of labels that span all regions of the inhabited world. This paper introduces a new dataset that contains over 100 MIL images. Surprisingly however, the results in [9] show only a small improvement over the results in [4]. Thus, our contribution over these previous works is to provide a methodological bridge between [4] and [9].

Though our work is base around using a single view image alone, the context surrounding an image may provide additional and useful information. The approach in [6] is based on this idea and so uses meta–data such as travel priors to narrow down likely locations. In future, this would be an exciting angle to pursue especially with the release of the Yahoo 100MIL dataset [7].

III. Data

For our experiments, we use the data-set from [4] which contains 6 million images collected from flickr. The images are accompanied by a GPS tag as to where the photo was originally taken.

As mentioned earlier, we approached this problem as a classification problem - thus we use the GPS coordinates to determine which city the images were located in and use those city names as labels. At this point, one thing we noted about the data-set is that the data was not balanced across each of the cities. This is not surprising given that popular tourist cities would receive more photographic attention than smaller and\or remote cities.

To balance the data-set, we limit the number of cities to those that contain at-least 11,000 images – this results in 37 non-overlapping cities. Then we randomly sample those images so that each city label has the same number of images. This results in a data-set representing 37 cities spanning nearly each continent (see figure 2). In total the data-set contains 373,618 images which was split 90-10 into a training set and a test set.

IV. Methods

In this section, I will describe the different methods we experimented with for our approach. We began first by establishing a baseline using two different simple machine learning algorithms. Then for comparison, we experimented with two deep learning approaches. Software for the deep learning models used in these experiments is available at [2].

Fig. 2. cities considered in our dataset, note how they span a larger variety of cultures and locations

A. *Using Deep Features*

Our initial approach is heavily based on the conclusions from [5] which found that by using a lazy learning approach with more sophisticated features, the results were doubled. This indicated that by using even more sophisticated features, we might be able to further improve those results. Thus, we experiment with deep features from convolutional neural network architectures.

We ran two experiments using the fc7 features extracted from the AlexNet model trained on the Places 205 data–set from [10]. Our intuition behind using these features is that the model from [10] was trained to perform the similar task of scene recognition and might therefore have the best discriminatory features for our purposes. Experiments in [8] also pint towards not using a model trained on ImageNet or other object oriented models for a more scene classification task.

Once extracted, we use k-nearest neighbor and SVM classifiers to predict the location for the test set of the images. For K-nearest neighbor's we use the L2 distance function. And for the SVM, we used a multi-class linear SVM with the L2-loss function. We chose these particular methods so that their set-ups would match those in [5]. That way in future, we can run our experiments on the full dataset to get a full comparison.

B. *Fine-Tuning Deep Networks*

Due to the size of our training set, instead of training a whole new deep model we fine-tune an existing deep network. Fine-tuning is achieved by retraining the classifier on our training dataset and updating the weights within the existing architecture by continuing back-propagation. For our experiments, we updated using small batch gradient descent. For our final two experiments, we fine-tuned two models: the shallower AlexNet from [10] and the much deeper GoogLeNet from [1]. Both models were trained on the same dataset (Places-205) which was designed for the problem of scene classification.

We chose these two models so as to compare the differences between using a shallower model and a deeper model. We predict that the deeper GoogLeNet will outperform the shallower AlexNet due to the fact that GoogLeNet already out-performs AlexNet in tasks such as scene recognition [1].

V. Results

our results for each of the aforementioned methods can be seen in table I. Our results are promising especially considering the limitations of our dataset. Note that we were able to see a huge improvement over random chance and that with fine tuning alone we are able to improve our results substantially.

Classification Method	Results
Random Chance	2.70%
fc7 features + KNN	2.90%
fc7 features + SVM	18.77%
AlexNet	22.03%
GoogLeNet	**33.30**%

TABLE I. TESTING RESULTS FOR OUR EXPERIMENTS

Surprisingly, however the results from using the fc7 features with KNN preformed near random chance. Perhaps this is due to the limited size of our training set. As mentioned in [4], the size of the training set matters for the accuracy. If there is not enough data in the training set for the new image to compare to, then there is little hope of that new image matching something in the dataset. We are likely in this case to see improvements by simply repeating this experiment on the full dataset rather than the smaller subset.

It seems fine-tuning the GoogLeNet network performed by–and–large the best. This points towards the idea that by using a deeper network we can achieve better results over its shallower counterparts. One major downfall towards using this model that we ran into is that the deeper GoogLeNet had a much higher computational cost and memory consumption.

VI. Limitations

Due to time constraints and the constraints of our methodology, we could not run our experiments on the larger data–set in [4]. This means that we can not compare our results to the previous work or the current state of the art. Our next planned set of experiments would be to run our experiment set ups on the full dataset so that we could compare the results against the established baselines.

Our results show promising potential to improve on the results from [4] as well as to provide an additional methodology compared to the state of the art in [9] that would require less training and have a faster run time.

VII. Conclusion

In this report, we describe our experimental set-up for training sophisticated classifiers for the task of geo-locating images within a city. The fine-tuned Places205-GoogLeNet and Alex-Net models achieve competitive performances as compared to the current state-of-the-art.

Given these results, we establish the following as future work. First and foremost, we are going to repeat this establish experimental set-up using the full dataset so as to establish an actual comparison between our results and the results in [5]. Given time, we'd also like to extract deep features from GoogLeNet to see if those feature might also improve the performance. Finally, given time we'd like to fully train a new deep model from scratch given the full 6MIL dataset rather than only fine-tune.

References

[1] http://places.csail.mit.edu/user/leaderboard.php.

[2] https://github.com/bvlc/caffe/wiki/model-zoo.

[3] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. What makes paris look like paris? *ACM Transactions on Graphics*, 31(4), 2012.

[4] James Hays and Alexei A Efros. Im2gps: estimating geographic information from a single image. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[5] James Hays and Alexei A Efros. Large-scale image geolocalization. In *Multimodal Location Estimation of Videos and Images*, pages 41–62. Springer, 2015.

[6] Evangelos Kalogerakis, Olga Vesselova, James Hays, Alexei A Efros, and Aaron Hertzmann. Image sequence geolocation with human travel priors. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 253–260. IEEE, 2009.

[7] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. The new data and new challenges in multimedia research. *arXiv preprint arXiv:1503.01817*, 2015.

[8] Limin Wang, Sheng Guo, Weilin Huang, and Yu Qiao. Places205-vggnet models for scene recognition. *arXiv preprint arXiv:1508.01667*, 2015.

[9] Tobias Weyand, Ilya Kostrikov, and James Philbin. Planet-photo geolocation with convolutional neural networks. *arXiv preprint arXiv:1602.05314*, 2016.

[10] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014.