Assisted Annotation of Visual Landmarks in Urban Wayfinding

Sean M. Arietta
Department of Computer Science
University of California, Berkeley

Maneesh Agrawala
Department of Computer Science
University of California, Berkeley

Ravi Ramamoorthi
Department of Computer Science
University of California, Berkeley

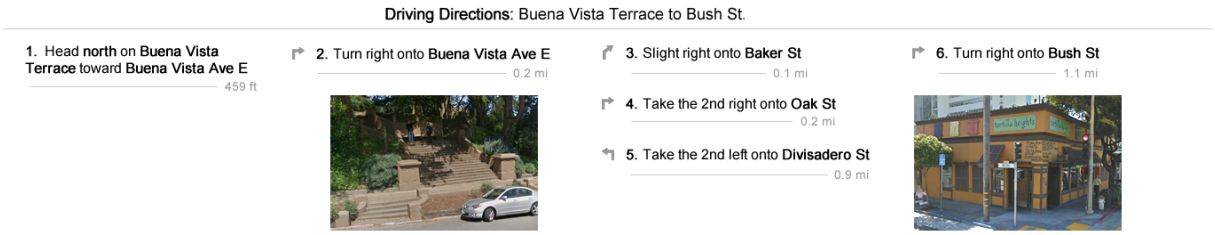

Figure 1: One application of automatically detecting visual landmarks is the ability to create new wayfinding directions with visually important cues. Here the turn-by-turn directions have been annotated with images of the detected visual landmarks near some of the turns.

Abstract

We propose a method for automatically determining visual landmarks in urban environments. We define a visual landmark as a visual element that would be considered important to an unfamiliar human if he/she were attempting to navigate a wayfinding route. Our system trains a machine learning model from user-annotated panoramic images via a popular crowd sourcing platform ensuring a varied set of human biases. Our model shows promising performance in its ability to predict the probability of visual landmarks in novel views across an entire urban environment from the sparse training data.

1 Introduction

Wayfinding is the process of navigating a physical environment, usually expressed as a series of point-to-point directions from an origin to a destination. Recent developments in mapping technology and the ubiquity of the Internet have facilitated the incorporation of wayfinding aids into everyday life. People commonly print maps or use in-car navigation systems to reach a destination. However, there are still numerous challenges that wayfinding still poses in visualizing and describing routes.

Currently, most wayfinding applications provide human-readable text or detailed roadmaps showing a route through an environment. While this information describes the general layout of a route, it often lacks a crucial element that is present in almost all human-described routes: visual landmarks.

As a simple, illustrative example, consider describing directions to your home. It is very likely that you will include various visual landmarks (e.g. an oddly colored building at a congested corner) to help an unfamiliar navigator successfully traverse the route. These landmarks are especially useful in crowded urban environments where street signs and road markers may be obscured. In essence, the visual landmarks provide alternate means for navigators to verify that they are correctly following the route.

We approach this problem by developing a method for automatically determining these visual landmarks in an urban environment. Figure 1 demonstrates a potential application of our detection results. Annotating existing wayfinding directions with visual landmarks provides a much more intuitive way for people to navigate their environments and is more consistent with the way people describe directions to one another.

Our method relies on user-annotated visual landmarks to serve as training data for a machine learning algorithm that produces a "model" for visual landmarks. We can predict the visual landmarks in novel views of an urban environment by testing agreement with our derived model. In this work we concentrate on 360-degree panoramas of cities scraped from Google StreetView[1], however other sensing modalities (LIDAR, aerial imagery) may be incorporated in later work.

2 Related Work

There have been a number of effortss over the last decade that have considered solutions to providing effective wayfinding directions. Summarizing each of these techniques is beyond the scope of this paper. Readers interested in an overview of these more classical approaches should refer to the principled study presented by Lovelace et al. [1999]. More recent research has considered the benefits of incorporating visual landmarks into wayfinding directions, but have relied on manual specification of landmarks [Raubal and Winter 2002] or models trained for specific scenes [Millonig and Schechtner 2007]. We extend these approaches by automatically detecting visual landmarks in novel scenes using a small set of training data.

The training data that we use to perform the detection is collected through Amazon's Mechanical Turk[2] crowd sourcing system. There has been a trend in the last decade to rely on these so-called crowds of workers to complete tasks that used to require expensive and relatively small user studies. This crowd sourcing model has been applied in many problem domains [Kittur et al. 2008]. Crowd sourcing itself has been studied as it presents challenges in verifying collected data, motivating workers [Mason and Watts 2010; Horton and Chilton 2010], and communicating tasks effectively.

Detecting visual landmarks can be thought of as an attempt to detect the "salient" regions in urban environments, although our goal

[1] http://www.google.com/streetview
[2] http://www.mturk.com

Figure 2: The set of panoramas that were used to collect user-defined visual landmarks. We tried to choose a set of images that was representative of the types of elements an urban environment might contain.

is more aligned with detecting the Schelling Points of these environments [Schelling 1980]. There has been extensive work in automatically detecting salient regions in images with various goals such as human attention prediction [Judd et al. 2009] [Itti et al. 1998], image summarization [Goferman et al. 2010], and recognition [Gao and Vasconcelos 2004]. Similar to the work of Caduff and Timpf [2008], we focus specifically on determining visual landmarks. Our work can be seen as a practical implementation of their theoretical results.

One important feature of our work is the ability to reliably detect the configurations of urban elements (buildings, signs, cars, people, etc). Although our current system ignores the potential gains from incorporating knowledge of these elements, we plan to consider them in future work. Previous research has attempted to automatically detect buildings - a prolific visual landmark - either from single images [Hoiem et al. 2007] or from LiDAR returns [Carlberg et al. 2009]. There have also been encouraging results in detecting roads [Ünsalan and Boyer 2005], cars [Felzenszwalb et al. 2008], people [Felzenszwalb et al. 2008], and trees [Secord and Zakhor 2007; Leckie et al. 2003]. Our system will use these approaches to automatically exclude large portions of urban environments from the prediction step resulting in a more efficient and accurate approach.

3 Collecting the Training Data

Our approach to automatically detecting visual landmarks in urban environments is to train a machine learning model capable of reliably predicting whether pixels in panoramas are visually important or not. In order to train a machine learning model to perform this task, a set of training data is required. Specifically, for our particular machine learning algorithm, we require a set of panoramas with hand-annotated regions containing visual landmarks. In order to produce a large enough training set, we rely on a crowd of workers to annotate the set of panoramas in Figure 2 with this information.

We implemented our crowd-sourced solution via Amazon's Mechanical Turk - an online system allowing a "requester" to post tasks in the form of web applications that are completed by "workers". We prompted workers to, "...mark a single region in [each panorama] that you consider most important if you were giving someone directions through that intersection." The web application we built allowed users to pan over a 360-degree panorama and mark rectangular regions with their mouse. Figure 4 shows a view snapshots of the interface presented to the workers.

Each annotation specified by the workers was saved to a database asynchronously to avoid refreshing the browser and causing potential loss of fidelity in workers' interest. The machine learning algorithm described in Section 4 uses this data to train a machine learning model capable of predicting visual landmarks.

Although the purpose of collecting the training data was primarily for training our model to predict visual landmarks, one auxiliary conclusion of our results is that people tend to mark the same regions despite their inability to communicate with one another about the task. Figure 3 shows a visualization of all of the annotations received for one panorama. Notice that almost all of the workers in this case marked the fountain as being to most important visual landmark for guiding *someone else* through the intersection.

Figure 3: A visualization of the set of user-annotations for one our training panoramas (see Figure 2). Notice that almost all of the users marked the same area, providing strong evidence that our model is trained to predict Schelling Points [Schelling 1980].

The agreement amongst the set of non-communicating workers is most directly explainable by the concept of a Schelling Point [Schelling 1980]. A Schelling Point is a feature that two people who have a common goal, but cannot communicate, consistently choose as being important. Thomas Schelling gives the example of two parachuters who become separated but have the same map. Each is asked to choose a point on the map to meet; a point they would predict the other parachuter would also choose. In his work, Schelling showed that people tend to agree on these features despite their inability to coordinate with one another. Similarly, our system is effectively attempting to find the visual Schelling Points of urban environments.

4 Building a Model for Visual Landmarks

Our approach to predicting visual landmarks in novel urban intersections is to derive a machine learning model of visual importance based on the set of user-defined annotations we gathered from our collection system discussed in Section 3.

Given a set of regions in a set of panoramas that have been marked as visual landmarks, we use the framework provided by Fan et al. [2008] to train a support vector machine capable of determining whether pixels in a novel panorama (one not included in our training) are part of a visual landmark or not. Specifically, we minimize the function:

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N+M}\zeta\left(\mathbf{w};\mathbf{x_i},y_i\right) \quad (1)$$

(a) The initial view presented to a user (b) A panned view (down and left) (c) A user-annotation of the visual landmark

Figure 4: In our online user-annotation system a user is initially (a) presented with a random view of a 360-degree panorama. The user is able to (b) pan around the panorama and (c) mark the visual landmark with a rectangular box. The annotation is saved to a database and eventually used to train our machine learning model.

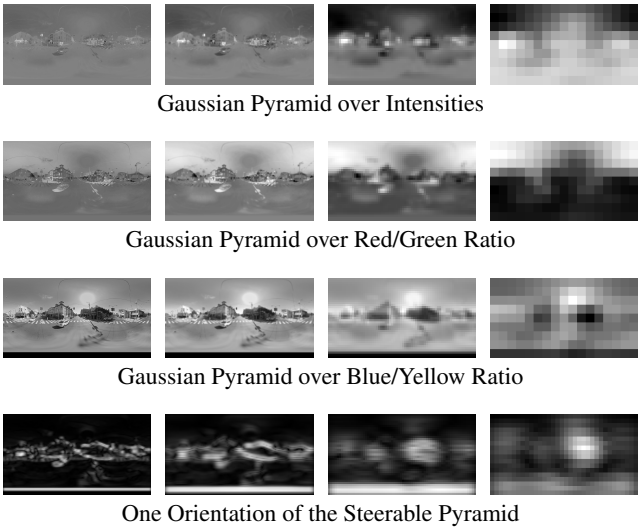

Figure 5: A representative set of features used as input to our machine learning algorithm. The top row are some of the levels in a Gaussian pyramid over intensities, the middle two rows are the same levels for the chromaticities, and the last row is all of the levels in a steerable pyramid for one filter orientation.

where y_i are the training labels ($+1$ or -1), $\mathbf{x_i}$ are the features, and $\mathbf{w}$ are the weights of the model. The explicit parameters of this optimization are the cost function ζ, the weighting parameter C, the number of positive examples N, and the number of negative examples M. Additionally, solving this equation usually involves an approximation technique introducing an error parameter ϵ controlling the accuracy of the solution.

As with all machine learning models, the choice of which features to extract from the input is extremely important. Although we considered many combinations of different features, ultimately we settled on the same features used by Itti and Koch [Itti et al. 1998]. These include the Gaussian pyramids of the intensities and chromaticities of each panorama, concatenated with the responses of a set of steerable filters over a range of spatial scales (similar in spirit to the steerable pyramid [Simoncelli and Freeman 1995]). The intensities were derived by computing the mean of the pixels in RGB space. The chromaticities were computed by taking the ratio of the red/green and blue/(red-green) channels of the original image. Figure 5 shows some of these features for the top-left image in Figure 2.

5 Detection Results

The primary goal of this work is to generate a model capable of predicting whether a pixel belongs to a visual landmark or not. We tried many combinations of machine learning approaches, parameters to those approaches, and features. Documenting all of those combinations is beyond the scope of this paper. The reader can refer to our supplemental materials page for an exhaustive list[3].

Ultimately we chose to train a linear support vector machine using the liblinear library developed by Fan et al. [2008]. For the cost function in Equation 1 we used the L1-regularized logistic regression form:

$$\min_{\mathbf{w}} ||\mathbf{w}||_1 + C \sum_{i=1}^{N+M} \zeta\left(\mathbf{w}; \mathbf{x_i}, y_i\right)$$

Where,

$$\zeta\left(\mathbf{w}; \mathbf{x_i}, y_i\right) = \log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x_i}))$$

We used a cost penalty (C) of 10, an error threshold (ϵ) of 0.01, and a set of 4×10^4 / 8×10^4 positive/negative samples respectively as parameters to the model training.

To test the performance of our generated model we generated predictions for the set of five panoramas in Figure 6. These panoramas were not included in our training set, however we did collect user annotations for them as described in Section 3 for ground-truth comparisons.

Figure 6 shows the results of our predictions. In these images, brighter areas correspond to more likely candidates of visual landmarks. We derive this likelihood by computing the dot product between the extracted features in the input image and the model weights derived during training.

In many cases our model is capable of predicting visual landmarks consistent with humans. For instance, in the top images in Figure 7 the tall orange building and the blue wall mural are labeled as visual landmarks - consistent with the manual annotations. However, our model is not robust enough to handle scenes where there is a lot of occlusion and objects that may have similar statistics to visual landmarks. The bottom images in Figure 7 demonstrate this limitation. Note that the car is being labeled as the most important visual element in this panorama rather than the brightly colored building.

[3] http://aether.cs.berkeley.edu/silicon/training.results.php

Figure 6: The results of our machine learning model. The images on the left were treated as input to our prediction approach, which consists of extracting features from the input and computing the dot product between those features and our model weights. Brighter areas in the right images correspond to more likely visual landmarks.

Figure 7: A comparison of a success case (top) and a failure case (bottom). Notice that the model is capable of picking out buildings that colored significantly different, except in the presence of distracting objects (cars, people, etc). We hope to remedy these shortcomings by eliminating these superfluous elements as a pre-process.

We hope to fix these types of errors by first segmenting out transient parts of the scene (cars, people, etc) using machine learning models trained individually for those elements.

6 Conclusions

We have described a system for automatically predicting where visual landmarks exist in panoramic images of urban environments. We collected user-defined visual landmarks from a small set of panoramas using Amazon's Mechanical Turk crowd sourcing platform to train a support vector machine capable of predicting visual landmarks in novel panoramas. Our results indicate that we can reliably predict these landmarks in cases where the input contains mostly buildings, but fail to do so when there are cars, people, and significant tree coverage. We are confident that future research will be able to avoid many of the pitfalls of the current system to create a robust predictor.

7 Future Work

Currently our method treats all pixels in novel panoramas as being equal. We expect to gain a significant leap in prediction performance by incorporating "priors" into the process. Specifically, we plan to automatically rule out pixels where cars, people, trees, roads, and sky are present. Additionally, our current system performs its training and detection on unrectified panoramas. Projecting these spherical images into their respective orthographic forms should also increase the prediction performance of our system.

Another direction of future work is to incorporate more sensing modalities into the training and prediction phases of our system. LiDAR has been used extensively in the community for detection purposes and could provide us with similar benefits.

References

CADUFF, D., AND TIMPF, S. 2008. On the assessment of landmark salience for human navigation. *Cognitive Processing 9*, 249–267.

CARLBERG, M., GAO, P., CHEN, G., AND ZAKHOR, A. 2009. Classifying urban landscape in aerial lidar using 3d shape analysis. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, 1701 –1704.

FAN, R.-E., CHANG, K.-W., HSIEH, C.-J., WANG, X.-R., AND LIN, C.-J. 2008. Liblinear: A library for large linear classification. *J. Mach. Learn. Res. 9* (June), 1871–1874.

FELZENSZWALB, P., MCALLESTER, D., AND RAMANAN, D. 2008. A discriminatively trained, multiscale, deformable part model. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 1 –8.

GAO, D., AND VASCONCELOS, N. 2004. Discriminant saliency for visual recognition from cluttered scenes. In *In Proc. NIPS*, 481–488.

GOFERMAN, S., ZELNIK-MANOR, L., AND TAL, A. 2010. Context-aware saliency detection. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2376–2383.

HOIEM, D., EFROS, A. A., AND HEBERT, M. 2007. Recovering surface layout from an image. *Int. J. Comput. Vision 75* (October), 151–172.

HORTON, J. J., AND CHILTON, L. B. 2010. The labor economics of paid crowdsourcing. In *Proceedings of the 11th ACM conference on Electronic commerce*, ACM, New York, NY, USA, EC '10, 209–218.

ITTI, L., KOCH, C., AND NIEBUR, E. 1998. A model of saliency-based visual attention for rapid scene analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 20*, 11 (November), 1254–1259.

JUDD, T., EHINGER, K., DURAND, F., AND TORRALBA, A. 2009. Learning to predict where humans look. In *IEEE International Conference on Computer Vision (ICCV)*.

KITTUR, A., CHI, E. H., AND SUH, B. 2008. Crowdsourcing user studies with mechanical turk. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ACM, New York, NY, USA, CHI '08, 453–456.

LECKIE, D., GOUGEON, F., HILL, D., QUINN, R., ARMSTRONG, L., AND SHREENAN, R. 2003. Combined high-density lidar and multispectral imagery for individual tree crown analysis. *Canadian Journal of Remote Sensing 29*, 5, 633–649.

LOVELACE, K., HEGARTY, M., AND MONTELLO, D. 1999. Elements of good route directions in familiar and unfamiliar environments. In *Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science*, C. Freksa and D. Mark, Eds., vol. 1661 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 751–751.

MASON, W., AND WATTS, D. J. 2010. Financial incentives and the "performance of crowds". *SIGKDD Explor. Newsl. 11* (May), 100–108.

MILLONIG, A., AND SCHECHTNER, K. 2007. Developing landmark-based pedestrian-navigation systems. *Intelligent Transportation Systems, IEEE Transactions on 8*, 1 (March), 43–49.

RAUBAL, M., AND WINTER, S. 2002. Enriching wayfinding instructions with local landmarks. In *Geographic Information Science*, M. Egenhofer and D. Mark, Eds., vol. 2478 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 243–259.

SCHELLING, T. 1980. *The strategy of conflict*. Harvard University Press.

SECORD, J., AND ZAKHOR, A. 2007. Tree detection in urban regions using aerial lidar and image data. *Geoscience and Remote Sensing Letters, IEEE 4*, 2 (april), 196 –200.

SIMONCELLI, E. P., AND FREEMAN, W. T. 1995. The steerable pyramid: A flexible architecture for multi-scale derivative computation. 444–447.

ÜNSALAN, C., AND BOYER, K. L. 2005. A system to detect houses and residential street networks in multispectral satellite images. *Computer Vision and Image Understanding 98*, 3, 423 – 461.