

# Geolocating Static Cameras

Nathan Jacobs, Scott Satkin, Nathaniel Roman, Richard Speyer, and Robert Pless  
Department of Computer Science and Engineering  
Washington University, St. Louis, MO, USA  
{jacobsn, satkin, ngr1, rzs1, pless}@cse.wustl.edu

## Abstract

A key problem in widely distributed camera networks is geolocating the cameras. This paper considers three scenarios for camera localization: localizing a camera in an unknown environment, adding a new camera in a region with many other cameras, and localizing a camera by finding correlations with satellite imagery. We find that simple summary statistics (the time course of principal component coefficients) are sufficient to geolocate cameras without determining correspondences between cameras or explicitly reasoning about weather in the scene. We present results from a database of images from 538 cameras collected over the course of a year. We find that for cameras that remain stationary and for which we have accurate image timestamps, we can localize most cameras to within 50 miles of the known location. In addition, we demonstrate the use of a distributed camera network in the construction a map of weather conditions.

## 1. Introduction

A global network of tens of thousands of outdoor networked cameras currently exists. Individuals and groups mount cameras for surveillance, for observing weather and particulate matter, and for viewing the natural beauty of scenic locations. Linking these cameras to the world wide web is a cheap and flexible method for individual camera owners to view images and to share these images with a wide audience. These cameras form a large, and growing, free global imaging network. Accurate localization of unknown cameras is an important first step in using this network of webcams. We address the localization problem and demonstrate the use of this network to construct a map of weather conditions.

This paper explores the following localization problem: Given static cameras that are widely distributed in a natural environment with no known landmarks, no ability to affect the environment, and perhaps no overlapping sensing areas (“fields of view”), discover the positions of the

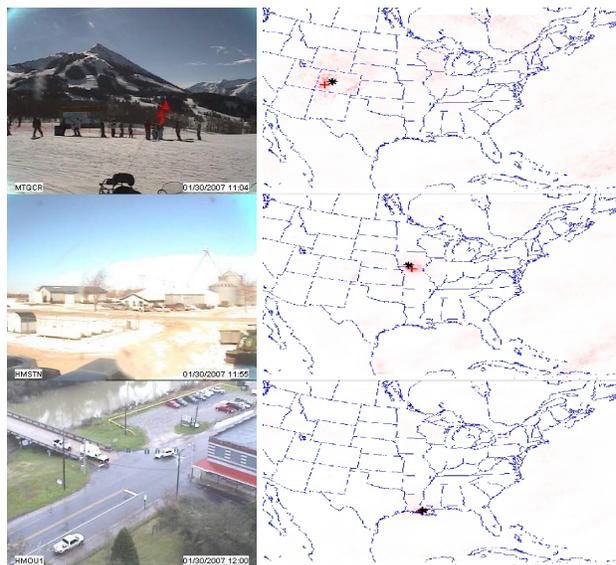


Figure 1. It is possible to geolocate an outdoor camera using natural scene variations, even when no recognizable features are visible. (left) Example images from three of the 538 static webcams that have been logged over the last year. (right) Correlation maps with satellite imagery; a measure of the temporal similarity of the camera variations to satellite pixel variations is color coded in red. The cross shows the maximum correlation point, the star shows the known GPS coordinate.

cameras. The key point in the above problem definition is that we consider real, natural environments. Sensing data from natural environments has useful properties for localization. First, variations in natural environments happen at many time scales, examples include changes due to daylight, weather patterns, and seasons. Second, because these phenomena are spatially-localized, over a long period of time the time-course of these variations is unique to a particular geographic location.

We present two localization methods that use natural temporal variations: one based on correlations of camera images to geo-registered satellite images and the other based on correlations with cameras with known locations.

We find that by using natural temporal variations our methods gives an accurate estimate of the camera location when other methods are likely to fail, specifically, it is robust to imaging distortion and works well even with a limited field of view.

## 2. Related Work

Finding the extrinsic calibration (the positions and orientations) of a network of cameras is a precursor to most applications in geometric vision. This problem has been extensively studied in the case where there are feature correspondences matching points in images from multiple cameras [12, 5], or matching features in the image to features computed from a digital elevation map [17, 8, 16].

Within the deployment of more distributed camera networks, distributed versions of these geometric calibration have been proposed and implemented [9, 13]. Various cues (beside feature correspondences) have been proposed to define camera topologies or approximate relative camera positions, based on object tracks [14], or statistical correlation of when objects enter and exit the camera views [18], both of which allow inference of camera locations when the camera fields of view do not overlap, although the cameras must be close enough so that objects appear or disappear between cameras, and there is a low entropy distribution of differences between departure times (from one camera) and arrival times (in another camera).

To our knowledge, the only other work to geolocate a camera using natural scene variations is based on explicit measurements of the sun position ([7]), which was followed by work in computing absolute camera orientation [19]. These techniques require special hardware to ensure the sun is in the field of view and accurate camera calibration to determine the angle of the sun.

However, the general desire for knowing the geolocation of a large set of webcams is highlighted in community efforts to build such a list, including lists with manually entered locations [4] and estimated locations using IP-address reverse lookup [1].

## 3. Consistent Natural Variations in Outdoor Scenes

The consistent causes of image variations in static outdoor cameras are the diurnal cycle and the weather. Recently, it was found that even if cameras view different scenes, there are consistent patterns to how these images vary over time. In particular, the PCA decomposition of images from each camera creates image components (which are scene dependent), and coefficients (whose daily pattern of variation are nearly independent of the scene) [11]. Creating a camera localization method based on these coefficients eliminates any camera-specific feature specification.

The data from each camera can be summarized as a data matrix  $I \in \mathbb{R}^{p \times T}$  where each column is an image of  $p$  pixels of the same scene at time  $t$ . Singular Value Decomposition decomposes this matrix as  $I = U\Sigma V^T$ , where the columns of  $U$  are the principal components and the columns of  $V$  (we use the first three columns for all experiments) are the time-series of principal component coefficients. A large-scale statistical study of images from 538 outdoor scenes [11] finds that the matrix of components  $U$  and singular value matrix  $\Sigma$  are scene dependent but the matrix of coefficients  $V$  is much less so. Figure 2 shows coefficient trajectories from several cameras (i.e, three leading columns of  $V$ ) for one day.

In the majority of outdoor cameras the leading principal component encodes the difference between day and night. As such, the coefficient trajectory of this component make a sharp transition at dawn and dusk. Differences in dawn and dusk time due to geolocation or natural seasonal variation cause the times of the sharp transitions to change. The second and third components have coefficient trajectories that indicate difference due to sun position. The scene specific components highlight appearance changes between the sun facing east and west, and differences between dawn and dusk and the middle of the day. The ordering of these components is not fixed; this is a problem we address later. The coefficients of these components are significantly affected by the weather (e.g., the magnitude of the coefficients is lower when it is cloudy).

The temporal variations in the PCA coefficients are related to natural scene variations and are consistent across many cameras. The remainder of this paper implicitly uses these variations to geolocate widely-distributed static cameras.

## 4. Camera Localization

How can we determine the geographic coordinates of a static camera? Potential solutions depend on what external information is available, and this explores three scenarios for camera localization. First, even when there is no other information, weak camera localization is possible by correlating image variations with a map of solar illumination. Second, if the camera is in a region with satellite coverage, localization is possible by seeking the region of the satellite image that correlates most with the image variations. Third, when there already exists a network of cameras with known locations, a position of a new camera can be estimated by finding the existing cameras with correlated image variations and interpolating their known location.

**Evaluation Dataset** All experiments were performed on a database of over 17 million images captured over the last year from 538 static outdoor cameras [11] located across

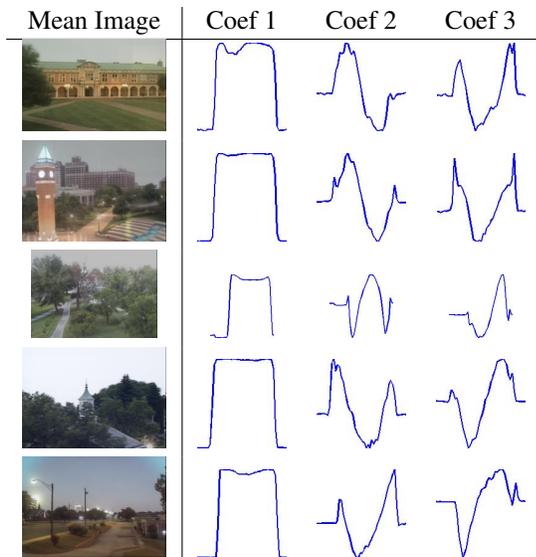


Figure 2. The principal component coefficients of static images of outdoor scenes have consistent patterns. This figure shows the mean image and plots of the first three PCA coefficients for one day for several camera. The horizontal axis of each plot is the time of day and the vertical axis is the coefficient value. The coefficients for different cameras are similar despite the fact that the corresponding scenes are very different.

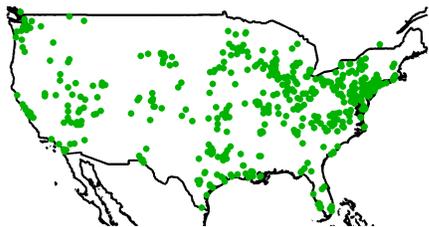


Figure 3. A scatter plot of the locations of cameras used to validate our algorithms.

the United States (see Figure 3). The cameras in the dataset were selected by a group of graduate and undergraduate students and many come from the Weatherbug camera network [2]. We selected the cameras with published latitude and longitude coordinates (which we assumed to be correct). Cameras which moved (including rotation or zoom) during the two testing time frames (April 2006, February and March of 2007) were rejected from the dataset.

#### 4.1. Absolute Camera Localization

This section describes a method for estimating the location of a camera using natural appearance variations and geo-registered satellite imagery. Using straightforward statistical techniques we show that this is possible using only a small number of principal component coefficients of images from the camera and satellite images taken at the same time. Since the mapping from satellite image coordinates to

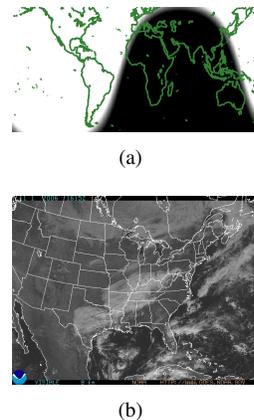


Figure 4. Examples of geo-registered images that we use to localize cameras. (a) A synthetic satellite image in which intensity corresponds to the amount of sunlight. (b) A visible-light image from a geostationary satellite.

a global coordinate system is known, the localization problem reduces to determining which pixel in the satellite image is the most likely location of the camera.

For a collection of  $T$  different time points we find the satellite images and camera images taken closest to each time point. The geo-registered satellite images are combined into a matrix of  $S \in \mathbb{R}^{p \times T}$ , where each column is an image. The camera image data is decomposed using incremental SVD [6] to approximate the first  $k$  PCA components of the camera images. The corresponding coefficients define the matrix  $V \in \mathbb{R}^{T \times k}$ , where each column is the time-course of one coefficient from the camera we are attempting to localize.

For a given camera, we compute the correlation score of each pixel in the satellite image. This score is defined as the correlation of the individual pixel time-series signal (the rows of  $S$  which encode how that pixel of the satellite varies through time), and a signal constructed as a projection of the PCA coefficient matrix  $V$ . We construct this projection as the linear combination of the rows of  $V^T$  that is closest to the satellite pixel signal in the least-squares sense. This score can be computed for all pixels at once as:

$$\text{diag}(S(SV(V^T V)^{-1}V^T)^T)$$

Allowing for the pixel to correlate with a linear combination of the PCA coefficients provides robustness to the ordering of these PCA coefficients. Computing this score for every pixel yields a false-color satellite image in which pixel intensities correspond to the temporal similarity of the pixel to the camera. Examples of these images for two types of satellite images are shown in Figures 1 and 5.

**Using a synthetic daylight map.** As a baseline for comparison we consider the case in which no satellite coverage

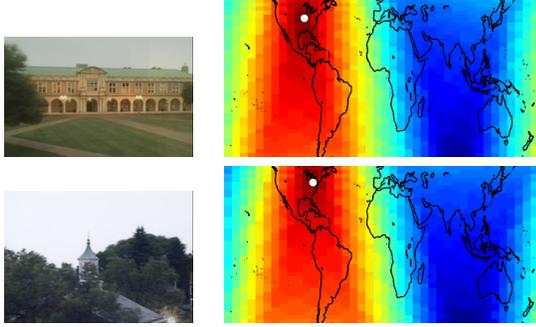


Figure 5. The correlation of the first PCA coefficient with pixels from the synthetic daylight map shown in Figure 4(a). The region with the highest correlation corresponds to the location of the camera (white dot).

is available. We use the algorithms described above without modification on a synthetic daylight map where intensities correspond to the amount of sunlight (Figure 4(a) shows an example). These images are generated by thresholding the solar zenith angle  $z$  for a given time and location. Pixels intensities are as follows: black if  $z > 100$ , white if  $z < 90$ , and varying linearly between the thresholds. Examples of correlation maps generated from this dataset are shown in Figure 5. This method gives very similar results to an algorithm which specifically searches for dawn and dusk in the image data and uses the length of the day and the dawn time to calculate position.

**Using visible satellite images.** We now present results of localizing cameras using images from the NASA Geostationary Operational Environmental Satellite [3]. See Figure 4(b) for an example image from the satellite dataset. We tested on two 300 image datasets from two satellite views: one of the Maryland area and one of the Pennsylvania area. We find that by using visible satellite images our algorithm localizes most cameras within 50 miles of the known location. Figure 8 shows a histogram of errors in the predicted locations. Figure 6 shows the actual position and our estimates for cameras in Pennsylvania. The mean localization error over all cameras is 44.6 miles; this is skewed by dramatic errors in a few cameras, dropping the 8 outliers reduces the mean to 23.78 miles.

## 4.2. Relative Camera Localization

Global localization, using the methods described above, depends on the availability of a set of signals with known mappings to a global coordinate system. In this section we eliminate this requirement by solving the problem of localizing a camera relative to other cameras. One distinct advantage of this approach is that accuracy is not dependent on the spatial discretization of the global signal; adding more cameras would give more accurate localization.

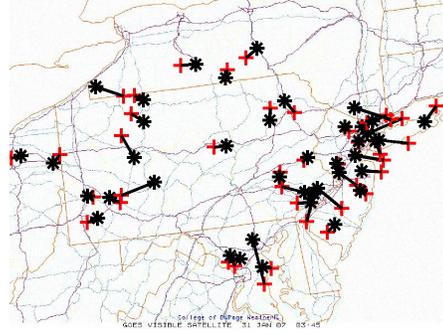


Figure 6. A comparison of the estimated (red crosses) and actual locations (black stars) for the Pennsylvania areas cameras.

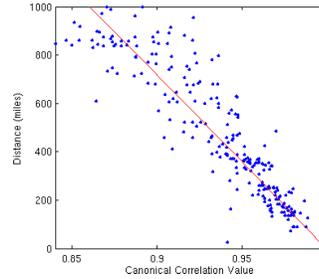


Figure 7. A scatter plot of the canonical correlation and the distance between a single camera and the 402 remaining cameras. There is a strong linear relationship between distance and canonical correlation, especially at small distances.

Our approach is based on the intuition that geographically close cameras will have similar weather patterns and hence similar PCA coefficient trajectories (the columns of the matrix  $V$ ). The problem with directly comparing the PCA coefficients of cameras is that the trajectory patterns may be permuted or split between several components. To overcome this difficulty we use canonical correlation analysis (CCA) [10] to solve for linear combinations of the PCA coefficients of each pair of cameras that maximizes the diagonal of the cross-correlation matrix. Specifically it solves for the projections  $p_{i,j}, p_{j,i}$  of the PCA coefficient matrices  $V_i$  and  $V_j$  that maximize the correlation  $\rho$  between the two signals

$$\rho = \max \text{corr}^2(V_i p_{i,j}, V_j p_{j,i}).$$

Figure 7 shows the linear relationship between the largest canonical correlation and the known distance for many pairs of nearby cameras. Using this relationship we can predict the distance given only the canonical correlation.

To determine the absolute location of the camera we assume that the locations of all other cameras are known and calculate the canonical correlation  $\rho_i$  between the new camera and each localized camera. The estimated location of the new camera is the  $\rho_i$  weighted average of the known locations of the three cameras with the highest canonical correlation.

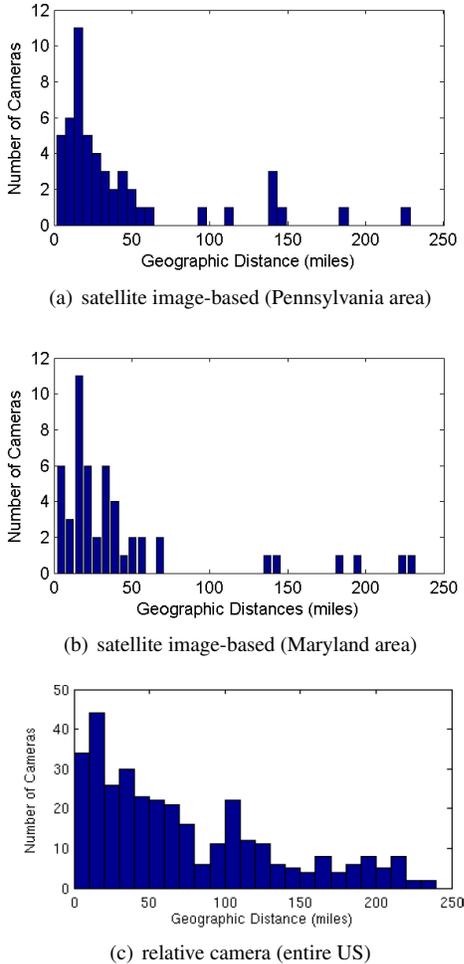


Figure 8. The distribution of errors in location prediction, using the satellite image based method, for a sets of cameras in the Pennsylvania and Maryland areas. (c) The distribution of errors using the relative localization method for 403 cameras located across the United States.

We tested the relative localization algorithm on a dataset of 403 static cameras using images sampled every five minutes over a one week period. Locations were estimated separately for each camera by localizing relative to the remaining cameras. Figure 8(c) shows the absolute error in the location estimates, the mean error was 91.3 miles. We find that the accuracy of the estimates is weakly correlated with the distance to the nearest neighbors, and when neighboring cameras are geographically close the accuracy is similar to the satellite correlation algorithm.

## 5. Generating Satellite Images from Many Webcameras

In the previous section we solve for camera locations by finding the maximum correlation between variation in cam-

era images and variations in pixels of the satellite image. This leads us to consider the reverse question; could a collection of widely distributed cameras allow us to predict an unknown satellite image? In this section we demonstrate the ability to construct visible satellite images.

We take the supervised approach by using regularized linear regression to learn a mapping from a set of images from webcams to a satellite image. Each training example consists of a satellite image  $S(t)$  and a set of webcam images  $I_{c,t}$  taken at the same time  $t$ . We first reduce the dimensionality of webcam images  $I_{c,t}$  separately at each camera using PCA and use the first  $k$  PCA coefficients as predictors (the results shown use  $k = 3$ ).

To learn the mapping we construct a matrix of satellite images  $S \in \mathbb{R}^{p \times T}$ , where each column is a satellite image. The camera data is summarized as a matrix of PCA coefficients  $V \in \mathbb{R}^{T \times k}$ , where each row contains the first  $k$  PCA coefficients for all cameras for images captured at given time. We then solve for set of coefficients  $F = SV(V^T V + \lambda \mathbf{I})^{-1}$  (we use  $\lambda = .01$ ). Using  $F$  we can predict an unseen satellite image from a set of camera PCA coefficients  $V_t^T$  by multiplying by the coefficient matrix  $FV_t^T$ .

We evaluated this method using a set of 1700 visible satellite images from four consecutive months and 42 webcams in the Maryland, Virginia area (the set shown in Figure 8). We use 1400 of these satellite images to define the linear regression model. Figure 9 shows that prediction of satellite images from web camera images is feasible using these methods.

## 6. Discussion

This work was in part inspired by the Weather and Illumination Database (WILD) dataset [15], which captured a long series of high-resolution images of the same scene over 6 months, and reasoned explicitly about weather conditions and atmospheric optics to create surface normal and depth estimates of a complicated urban scene. Our results indicate that the time series of PCA coefficients is strongly correlated with weather. This is efficient to compute and works in cases where the scene in view is too close to be noticeably affected by diffusion effects (as in Figure 1, bottom). This doesn't allow for reasoning about scene structure, but does offer a convenient method for camera localization.

We admit that a network of cameras which are localized with an error 24 miles is not likely to be useful for classical approaches to computing scene structure. Instead, the algorithms presented in the previous section are intended to demonstrate that location information is available without finding corresponding points or tracking corresponding objects. Furthermore, using the geographic information inherent in natural scene changes can be done based on image statistics alone, without creating explicit algorithms to com-

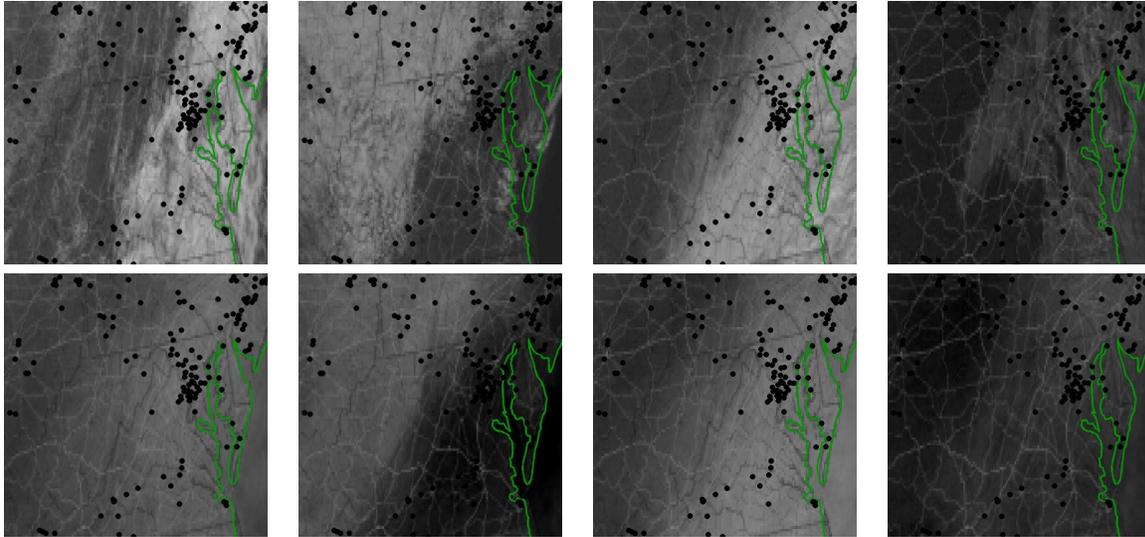


Figure 9. (top) Satellite images from the Washington D.C. area. (bottom) Predicted satellite image using PCA coefficients of webcams (located at black dots) for the corresponding time.

pute cloudiness or sun position. Thus, this offers a scalable solution to organizing the camera resources that continue to be added to the web. We believe that similar statistical representations of image variation will find interesting correlations at longer timescales (such as variations due to snowfall and tree foliage) and with other signals (such as wind velocity maps).

## References

- [1] <http://bbs.keyhole.com/ubb/download.php?number=12084>.
- [2] <http://weatherbugmedia.com/>.
- [3] <http://www.goes.noaa.gov/>.
- [4] <http://www.opentopia.com/hiddenecam.php>.
- [5] P. Baker and Y. Aloimonos. Calibration of a multicamera network. In *Omnivis 2003: Omnidirectional Vision and Camera Networks*, 2003.
- [6] M. Brand. Incremental singular value decomposition of uncertain data with missing values. In *Proc. European Conference on Computer Vision*, pages 707–720, 2002.
- [7] F. Cozman and E. Krotkov. Robot localization using a computer vision sextant. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, pages 106–111, Nagoya, Japan, May 1995.
- [8] F. Cozman and E. Krotkov. Automatic mountain detection and pose estimation for teleoperation of lunar rovers. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 1997.
- [9] D. Devarajan, R. J. Radke, and H. Chung. Distributed metric calibration of ad hoc camera networks. *TOSN*, 2(3):380–403, 2006.
- [10] H. Hotelling. Relations between two sets of variates. *Biometrika*, 28:321–377, 1936.
- [11] N. Jacobs, N. Roman, and R. Pless. Consistent temporal variations in many outdoor scenes. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, June 2007.
- [12] J. Jannotti and J. Mao. Distributed calibration of smart cameras. In *Workshop on Distributed Smart Cameras*, 2006.
- [13] W. Mantzel, H. Choi, and R. Baraniuk. Distributed camera network localization. In *Proc. Signals, Systems and Computers*, volume 2, 2004.
- [14] D. Marinakis and G. Dudek. Topology inference for a vision-based sensor network. In *Canadian Conference on Computer and Robot Vision (CRV)*, pages 121–128, 2005.
- [15] S. G. Narasimhan, C. Wang, and S. K. Nayar. All the images of an outdoor scene. In *Proc. European Conference on Computer Vision*, pages 148–162, 2002.
- [16] F. Stein and G. Medioni. Map-based localization using the panoramic horizon. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Nice, France, 1992.
- [17] W. Thompson, T. Henderson, T. Colvin, L. Dick, and C. Valiquette. Vision-based localization. In *ARPA Image Understanding Workshop*, pages 491–498, Washington D.C., 1993.
- [18] K. Tieu, G. Dalley, and W. E. L. Grimson. Inference of non-overlapping camera network topology by measuring statistical dependence. In *Proc. International Conference on Computer Vision*, pages 1842–1849, 2005.
- [19] A. Trebi-Ollennu, T. Huntsberger, Y. Cheng, E. T. Baumgartner, B. Kennedy, and P. Schenker. Design and analysis of a sun sensor for planetary rover absolute heading detection. *IEEE Trans. on Robotics and Automation*, 17(6), 2001.