Webcam2Satellite: Estimating Cloud Maps from Webcam Imagery

Calvin Murdock
Washington Univ. in St. Louis
calvinmurdock@wustl.edu

Nathan Jacobs
University of Kentucky
jacobs@cs.uky.edu

Robert Pless
Washington Univ. in St. Louis
pless@cse.wustl.edu

Abstract

We consider the problem of estimating the current satellite cloud map from a collection of broadly distributed, ground-based webcams. The approach uses historical, georeferenced satellite imagery to learn a mapping between the satellite image and the ground imagery. We explore representational choices for inferring the cloud status based on the ground-level imagery and consider several alternatives for spatially interpolating these sparse measurements to give a complete map. Proof of concept results show that this gives plausible estimates of satellite imagery.

1. Introduction

Satellite imagery is a cornerstone of weather and environmental imaging. It captures large-scale, calibrated, and regularly sampled imagery, supporting weather prediction and environmental monitoring. In this paper we consider one satellite data product – the cloud imagery commonly used for weather prediction – and explore algorithms to allow terrestrial sensors to estimate this imagery by locally estimating the cloud measures from ground-based webcams and interpolating these measurements over the continental scale. These approaches probably will not replace satellites in the short run, but the increasing danger from space debris [19] and direct targeting of satellites [3, 16] makes it important to consider alternative methods of creating large-scale weather maps.

The idea of creating satellite maps from ground-based image sensors was first introduced as an application of the Archive of Many Outdoor Scenes [7], a dataset of images from publicly available webcams from around the world. While their approach was very simple and had problems that we discuss later in this paper, it highlighted the potential to interpolate large-scale satellite maps from geo-calibrated cameras.

Relative to satellite imagery, webcams offer an alternative with an array of positive and negative features. Webcams are deployed along highways, parks, resorts, beaches, and schools. They are sparsely and unevenly distributed and may not be optimally placed for the purpose of weather estimation, but they have update rates on the order of seconds instead of the 15-60 minute update rates of satellites. Thus, this paper explores how to exploit satellite imagery available today in order to capture statistics of cloud appearance and derive related imagery if/when only ground-based webcam data is available.

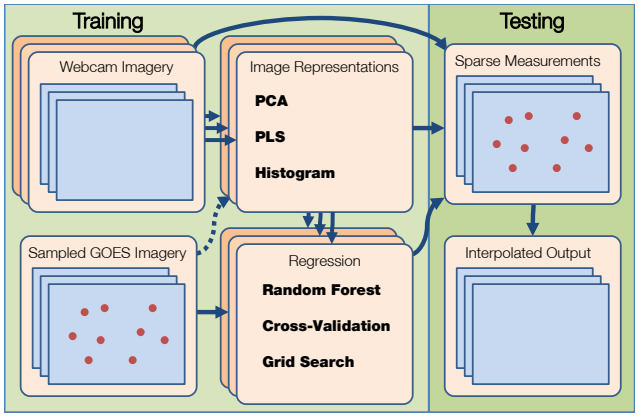


Figure 1. An overview of our approach. Given five months of satellite imagery and daytime webcam images from 2772 geo-located cameras, we learn low-dimensional image representations and then predict scalar cloudiness measurements taken from coincident satellite images using multiple regression techniques. Using 2-fold cross-validation and grid search, we select the best model for each camera. Finally, we predict sparse cloudiness measurements that are then interpolated to construct a complete cloud map for the entire region. Experiments include comparison of different low-dimensional image representation and evaluation on reconstructing one month of cloud images not in the training data.

The contribution of this paper is to explore modeling choices in learning the mapping from ground-based imagery to satellite imagery. In that context we consider:

- Algorithms for automatically learning camera-specific regression models to predict satellite image pixels from ground level imagery.
- Evaluation of spatial interpolation choices to estimate a continental-scale cloud map based on local estimates.

This work rests on several assumptions. Unlike previous work, we do not assume that a camera remains static; the image features and regression alternatives that we consider include some that apply to cameras that pan and tilt; however, we assume that the types of variation seen in the training period apply to the testing period. Second, some camera images may show clouds that extend beyond a single pixel in the satellite image, even though satellite images have a resolution where one pixel covers multiple kilometers in distance. Thus, we only attempt to predict a scalar intensity at the camera location to remove potential ambiguity between cloud elevation and distance.

1.1. Background and Previous Work

There is substantial interest in creating large-scale maps from ground level data. One example problem is estimating ground-cover classification (e.g. urban, farmed, forest) from uncontrolled sources, such as geo-located pictures uploaded to Flickr [10]. Related work seeks to map the scenicness of scenes, based both on the image content and the density of pictures taken at different locations, which provides an interesting combination of social and visual features [21, 4].

Large scale webcam data archives [9, 7] have been used as a test case for camera geo-location, geo-orientation, and some work in phenology [5, 14]. Analysis of such datasets [7] has shown that the PCA coefficients computed from a month of images captured at local noon from an outdoor camera have a strong dependence on local cloudiness, and a simple linear regression model was used to estimate continental scale cloud maps [8]. However, this paper did not evaluate the ability to reliably predict cloud maps that are significantly outside of the training set.

Supervised learning methods have also been used to predict other environmental properties from webcam imagery. Canonical correlations analysis was used to predict wind velocity in scenes with visible trees or flags [5], water vapor pressure was inferred in scenes with large depth of field using an image feature based on contrast [6], and semi-supervised tools closer to those used in this paper have been proposed to estimate atmospheric visibility from webcam data [20]. However, these approaches rely on simple linear regression models, require collocated environmental sensors, and are only effective for certain scenes.

2. Overview of Approach

We propose a method for estimating the current satellite image from a collection of webcam images (Figure 1). We first assign each webcam to a satellite pixel based on its geographic location. In the training phase, we learn a regression model that predicts the satellite pixel value from a concurrent webcam image. To do this, we extract several types of low-dimensional features (Section 3) and use cross-validation to select the feature that supports the most accurate regression estimate (Section 4). In the testing phase, the satellite pixel estimates made by the camera-specific regression model are interpolated to give a continental-scale satellite cloud map (Section 5). Experimental results show the feasibility of this approach and encourage continued work towards a ground-based global network for atmospheric monitoring.

3. Low-Dimensional Image Representation

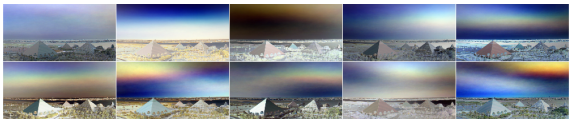
The regression problem of predicting cloudiness from webcam imagery is challenging due to the complexity and diversity of the scenes we consider. For example, weather cameras usually view wide expanses of sky while traffic cameras tend to focus on much smaller segments of land with more significant movement and variation. In this section, we discuss methods of handling two critical issues: the high-dimensional nature of our feature space and the temporal dependence of image appearance. To ease computational requirements and reduce overfitting, we explore three alternative low-dimensional image representations in an attempt to convert an individual image into a small number of features that retain information about the local cloudiness. We begin with a description of these methods and then describe our method for handling temporal variability.

In the context of this variation, there are a number of different cues to cloud cover. If the sky is visible, clouds can be viewed directly, but for cameras looking only at the ground, cloud cover may be implied by the presence of shadows or changes in lighting intensity. Additionally, many webcams move over time – either shaking due to wind or by remote controls. These variations make it unclear which image properties best represent cloud cover, so we explore a variety of image representations and evaluate their performance using cross-validation.

Principal Component Analysis (PCA) is one method of unsupervised dimensionality reduction that finds a low-dimensional subspace accounting for the greatest amount of variance in the training dataset. In the case of static cameras, PCA decompositions tend to be highly correlated with lighting changes due to sun position and cloud cover [8]. However, PCA does not explicitly take into account the goal of quantifying cloud cover. Instead, it encodes whatever the dominant causes of change are in the scene, including the presence of cars and the motion of trees.

We can address this issue by separating the sky and ground pixels – using the canonical component analysis approach in [7] – and take the PCA decomposition of both independently. Additionally, unlike past work with webcams, we take the PCA decomposition of color images, where the RGB channels at all pixels are stacked to form a single vector.

Partial Least Squares. To incorporate the specific goal



(a) PCA components

(b) PLS components

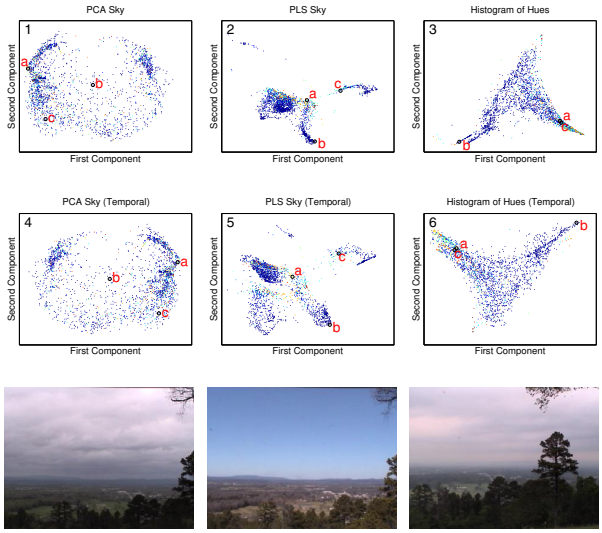
Figure 2. The first 10 PCA (a) and PLS (b) components for 5 months of daytime images from a webcam. Note that the PCA components encode color variations, often at sunset, while the PLS components encode a basis that is more relevant to cloudiness.

of quantifying cloud cover, it is appropriate to use a supervised method that considers known outputs on training data. Partial Least Squares (PLS) is a regression technique that projects both the predictor and response variable(s) onto low-dimensional subspaces that optimize regression performance between them [15]. For webcam imagery with corresponding satellite cloud pixels, this results in a representation that is most sensitive to cloud appearance and less sensitive to other variations such as sunset color. Figure 2 shows a representative example.

Histogram of Hues. Both PCA and PLS create linear decompositions of scene appearance that can fail in cases of camera motion and jitter. Thus, we also compute a color-based cue based on the observation that clouds alter the color of the sky. We transform the images to the HSV color space and construct a histogram that quantizes the frequency of hues in the sky pixels, ignoring saturation and value, which are more related to lighting conditions. We make two such representations: the first computes the histogram over the sky region using the computed sky-mask and the second considers only the top 20 rows of pixels in the image, which often contain segments of sky with clouds that are closest to the camera location even when cameras move. When this is not the case, the regression model will fail and the cross-validation step will choose an alternative model.

Encoding Temporal Information. Daily variation in sun position, and yearly variation in scene appearance are two of the most dominant appearance changes in many scenes. We augment the low-dimensional features to encode this temporal information with cyclic variables, encoded as a 2D unit vector. For the time of day, measured in hours, this vector is: $\left(\sin(\frac{2\pi \text{ time of day}}{24}), \cos(\frac{2\pi \text{ time of day}}{24})\right)$, and a similar 2D unit vector encodes the time of year.

These variables are appended to the low-dimensional image representations described in the previous sections.



(a) March 19, 13:59 (b) February 26, 18:59 (c) April 27, 12:29

Figure 3. A visualization of the complex structure of image representations by embedding the features onto two dimensions using Isomap [18]. Each image is represented as a single point whose color corresponds to the cloudiness measurement in the coincident satellite image. Original features (1-3) and those concatenated with temporal components (4-6) are shown for PCA (1,4), PLS (2,5), and histogram of hues (3,6). Three example images are shown, labeled (a) through (c). Note that a slight image translation causes images (a) and (c) to be separated with PCA and PLS despite their similar cloudiness, while the histogram of hues is unaffected. Also, concatenation of temporal information has the effect of spreading out similar points, which potentially increases the discriminability and resolution of a regression model.

Since the magnitude and intrinsic dimensionality of these feature vectors differ from the remainder of the low-dimensional representation, the temporal components are first multiplied by constants learned during cross-validation.

4. Estimating Satellite Image Pixels

Given low-dimensional representations of ground imagery, quantifying cloud cover becomes a regression problem. One simple method of regression assumes that the response variable varies linearly with the predictor variables. Partial Least Squares attempts to linearize the data by constructing a basis that is highly correlated with the response variables. However, a linear image basis that correlates with cloud cover is unlikely due to the complexity with which cloudiness is represented in images. Introducing nonlinearities into the regression model can result in a more accurate representation. We have found that random forest regression, a robust, nonparametric regression technique, is the most effective choice at estimating satellite cloudiness pixels. Furthermore, for the satellite images we consider,

several pre-processing operations significantly improve our prediction accuracy.

4.1. Satellite Imagery Preprocessing

In addition to cloud cover, satellite image appearance can be affected by a number of unrelated factors, including ground temperature. In infrared satellite imagery, this is expressed as regular, low-rank variations in intensity over the course of the day[1].

Because clouds tend to be sparse while diurnal intensity variations are fairly uniform, Robust Principal Component Analysis (RPCA) is an appropriate method to factor out these low rank intensity changes [11]. RPCA has a single parameter that controls the level of sparsity; a value of $\lambda = 0.002$ was found to be most effective.

Restricting the regression response variable to a single pixel in a satellite map simplifies computation and avoids ambiguities related to unknown camera calibration and orientation. However, sparsely sampling a high-frequency signal can result in unexpected distortion that does not accurately represent the underlying trend in the data. For example, some clouds in satellite images are very small and are not always accompanied by larger fronts. If an individual cloud happens to be directly overhead at the time a webcam image is taken, the resulting measurement will be large even if there are no clouds in the surrounding area and the webcam image shows a mostly clear sky.

In order to address this issue, we first filter out high frequency components of the satellite images with a Gaussian filter, whose variance approximates the observed correlation pattern. Figure 4 shows the stages of the complete process and Figure 5 shows statistics of unfiltered and filtered satellite imagery.

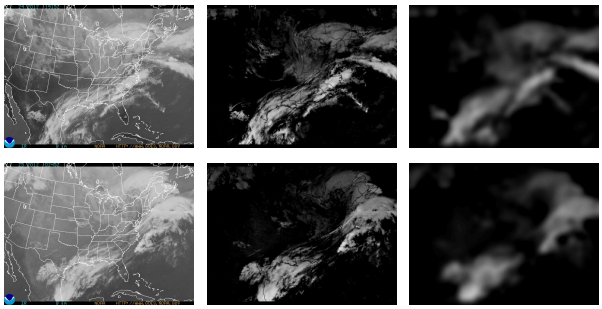
4.2. Random Forest Regression

Because the complexity and quantity of our data is large, a single regression model may not be able to capture an accurate representation of the relationship between webcam images and cloud map intensity, especially in the presence of noise and outliers. Random forests are instances of ensemble learning that robustly combine the outputs of multiple regression trees that consider random subsets of training data – using bootstrap aggregation – and features [2]. This was found to result in better performance than linear regression or support vector regression with a Gaussian kernel, as shown in Figure 10b. Within our implementation, we vary the number of trees in the forest and the maximum number of features that are considered.

Cross-Validation and Grid Search: In order to get the best possible performance considering the large variability in the webcam content and image representations, we use grid search and 2-fold cross-validation to select the image representation and model parameters that result in the lowest mean squared prediction error on a subset of the data. We first normalize the training data so so that the mean is zero and the standard deviation is one, which reduces the number of possible parameters to search through.

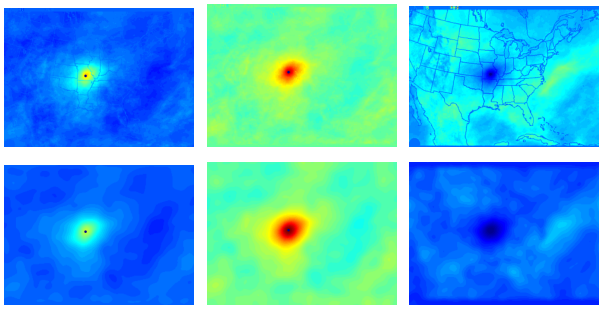
In total, we consider 8 image representations; PCA and PLS computed on the full image, the sky pixels only, and the ground pixels only, and the histogram of hues for the sky pixels only and for the top 20 rows in the image. The parameters that we consider are the weights of the temporal components and the number of regression trees in the forest and the maximum number of random features.

[1]This is wonderfully illustrated in the following YouTube video (not ours): http://www.youtube.com/watch?v=nhJ3KuJbjsY



(a) original (b) RPCA (c) filtered

Figure 4. (a) Infrared satellite images recorded at two different times. Notice the large variation in the background intensity due to changes in temperature. (b) The same images preprocessed using Robust PCA to remove these low rank intensity variations. Small, high frequency clouds are retained. (c) After Gaussian filtering with a standard deviation of $\sigma = 5$ allowing for more robust, single pixel measurements.



(a) covariance (b) correlation (c) variogram

Figure 5. Statistics of RPCA-processed, unfiltered (top) and filtered (bottom) satellite imagery. (a) The sample covariance computed from 5 months of satellite images in reference to the pixel indicated by the black dot. (b) The corresponding sample correlation. (c) The corresponding experimental semivariogram, which is used with Kriging.

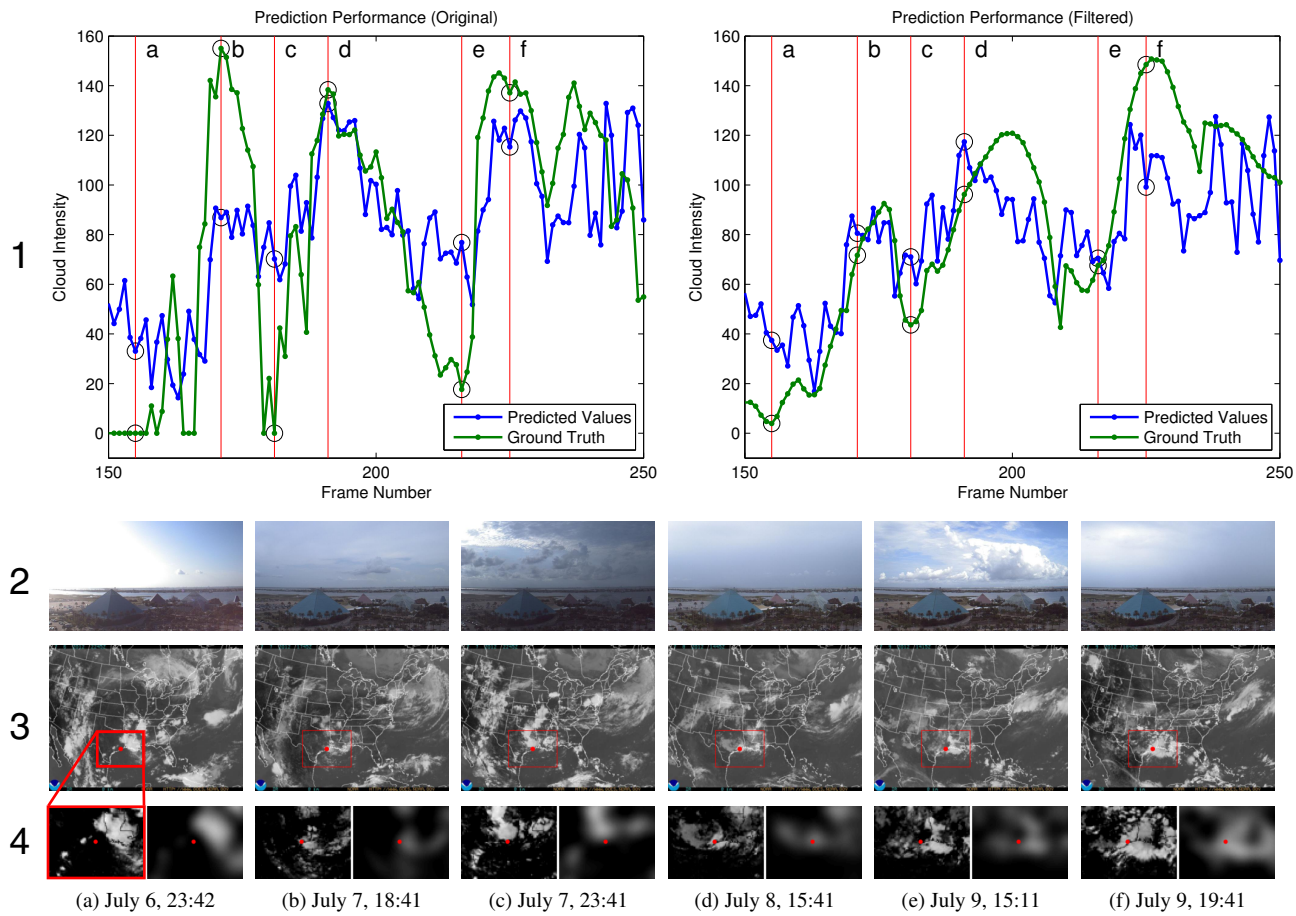


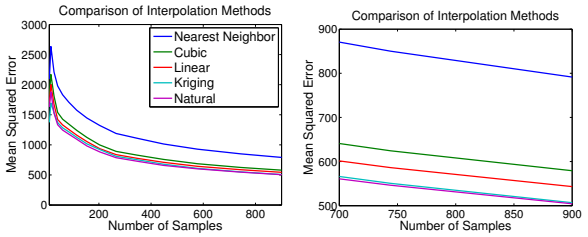
Figure 6. Row 1: A comparison between ground truth cloudiness measurements and those predicted using random forest regression with a PLS image representation from only ground pixels. The model on the left was trained using the samples from the original satellite images processed only with RPCA and the model on the right used Gaussian filtered images. While the predicted values are very similar for both, they match more closely to the Gaussian filtered measurements. This indicates that random forest regression has an effect similar to that of a low-pass filter. Specific measurement times, labeled (a) through (f), are indicated with a red vertical line. Row 2: The input webcam images at the indicated times. Row 3: The coincident, unprocessed satellite images at the indicated times. The location of the webcam is shown as a red dot. Row 4: A zoomed-in view of the corresponding processed satellite images. The left is unfiltered, thus retaining sparse, high frequency clouds, while the right is filtered. The predictions with discrepancies between ground truth and predicted values using unfiltered measurements correspond to satellite images where the measurement happened to be right at the edge of a cloud. Filtering the satellite image gives a more plausible target for the regression step, and is sensible because interpolating sparse webcam estimates will in any case give low resolution output.

5. Interpolating Webcam Measurements

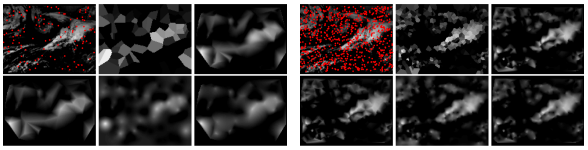
We use the camera-specific regression models to predict the cloudiness at individual satellite pixels from current webcam images. Since webcams are sparsely distributed, this results in a sparse set of estimated pixels that we combine to form a complete satellite image using interpolation. The best method for interpolation depends on the nature of the data. In this work, we limit our consideration to three purely spatial interpolation approaches. However, we plan to explore spatio-temporal interpolation methods in future work.

Spline Interpolation is a technique that fits piecewise polynomials to measurements in order to smoothly interpolate values at other points [13]. In short, the underlying interpolants are derived by enforcing continuity constraints on the function and its derivatives, which results in different levels of smoothness. In particular, we consider functions that are linear, which only requires that the functions be continuous, and cubic, which requires that the second derivatives be continuous.

Neighbor Interpolations construct an interpolated value solely from neighboring measurements. We consider



(a) full (b) zoomed-in



(c) 115 random samples (d) 744 random samples

Figure 7. A performance comparison of interpolation methods using ideal measurements sampled randomly from the original satellite image. In (a) and (b), the mean squared interpolation error is compared as a function of the number of samples. Example interpolation comparisons using different numbers of samples (c and d) show the original image (top left), nearest neighbor interpolation (top center), cubic interpolation (top right), linear interpolation (bottom left), Kriging interpolation (bottom center), and natural neighbor interpolation (bottom right).

nearest-neighbor and natural-neighbor interpolation [17], which calculates each value as a linear combination of neighboring measurements with weights determined by the local Voronoi tessellation.

Kriging is another method of interpolation that takes into account the previously observed correlation between data points [12]. This method relies on the semivariogram, which quantifies the variance of the difference between the values at two locations in the field. An experimental approximation of this function can be computed from the sample covariance; both are shown in Figure 5. For simplicity, we assume that the process is isotropic and fit the experimental values to a theoretical spherical variogram model [22].

6. Experimental Application

The methodology described in the previous section was implemented using a collection of publicly accessible outdoor webcams distributed across the United States. For training data, quantitative measures of cloud cover were sampled from coincident satellite images. Using approximately five months of cross-validated training data, the best image representation and regression model were selected and used to extrapolate one month of testing data.

6.1. GOES Infrared Satellite Imagery

Our ground truth cloud measurements come from infrared imagery obtained from Geostationary Operational Environmental Satellites (GOES) [1]. Unlike visual imagery that measures reflected photons within the visual spectrum, infrared imagery is highly correlated with temperature, so it is less affected by lighting variations that are unrelated to cloud cover. The intensity of a cloud in an infrared satellite image can be caused by a number of factors, including its type, elevation, and thickness, which contributes to the underlying complexity.

6.2. The Archive of Many Outdoor Scenes

The Archive of Many Outdoor Scenes (AMOS) is a collection of approximately 20,000 outdoor webcams from around the world with images recorded every 30 minutes [5]. Many of these cameras are also associated with geolocations that were either provided by the maintainer or inferred from their IP address. Within the United States, we selected 2772 geo-located cameras with a sufficient number of images recorded during the training and testing periods. Since webcams often have very poor low-light performance and are usually unable to capture appearance changes due to cloud cover at night, we only consider images that were recorded during the day as determined by the sunrise and sunset times derived from the camera geolocation and timestamp.

6.3. Sources of Error

There are a number of issues inherent with these data sources that could potentially affect regression performance. Since webcam imagery is only recorded every 30 minutes, the associated timestamp could have an error of up to 30 minutes depending on the refresh rate of the camera. Thus, we consider webcam and satellite images to be coincident if their timestamps differ by at most 15 minutes. If we assume that cloud motion is entirely translational, these temporal errors are equivalent to small spatial errors, which are bounded by the relatively low cloud speeds observed in the satellite imagery. Since a pixel in a satellite image is highly correlated with its neighbors and high frequency clouds are filtered out, these additional errors do not significantly affect performance.

In some cases, very large errors are present in the data. For example, geo-location metadata could be completely wrong, especially if they were inferred from server IP addresses. In addition, some cameras do not fit well with any of the image representations. This can occur if the camera is highly unstable or if no cloudiness cues are visible in the images. In these cases, there can be no meaningful regression model, which results in a very high cross-validation error. Since the size of our data set is so large, we can disregard these cameras without issue.

6.4. Results

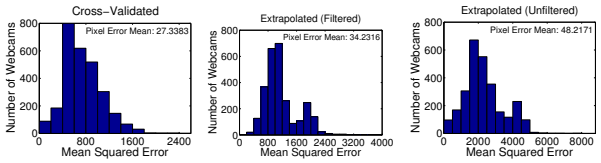
Of the 2772 cameras that were evaluated, we selected 2000 to be used in the cloud map construction, excluding outliers and low quality cameras that would reduce prediction accuracy. For all cameras, the average error of cross-validated training data was 27.34, or 10.7% of the dynamic range, while the average error of extrapolated data using Gaussian filtered satellite imagery was 34.23, or 13.4%. The slight increase can be attributed to the time difference between training and testing data and the varying availability of training data for different cameras. Because of anomalous high frequency clouds, the average error of extrapolated data using unfiltered satellite imagery was significantly higher. The complete results are summarized in Figure 8, and example predicted satellite maps for a variety of different conditions are shown in Figure 11.

For the majority of the cameras, the best regression performance was achieved with the PLS image representation using full webcam images, thus taking advantage of cues that appear in both ground and sky pixels. Most of the cameras in which the histogram of hues approach was more successful included substantial camera motion. PCA never performed better than PLS since PLS explicitly solves for the linear basis that gives the best correlation to the cloud estimates.

We also compare the regression error in using the PLS image decomposition on sky pixels and ground pixels. We find it interesting that these results are so well correlated. In Figure 9, we show an anecdotal exploration of cases where the errors are different. In these cases, camera (a) had lower error using only ground pixels, potentially due to the presence of stable foreground object shadows and the visibility of cloud shadows in the distance. Camera (b) shows a foreground dominated by transient, moving cars, which led to better regression performance using just the sky pixels.

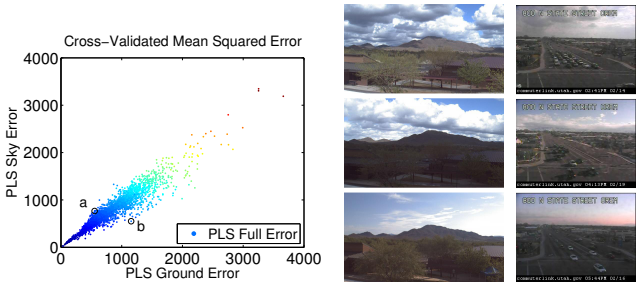
The final interpolation results are shown in Figure 10. Because natural neighbor interpolation only considers pixels within the convex hull of measurement locations, only pixels common to both interpolations were used in the comparison. For extrapolated testing data, the ideal Kriging interpolation performance was better, especially using filtered measurements. However, with measurements predicted from webcam imagery, the performance was very similar for natural neighbor and Kriging interpolation methods using both unfiltered and filtered training data. This is likely due to the low-pass filtering effects of random forest regression.

Since random forest regression consistently overestimated and underestimated extreme cloudiness measurements, we made constant contrast adjustments to better fit the range of possible values.



(a) (b) (c)

Figure 8. (a) The cross-validation error of training data using the best regression algorithm for each camera. (b) The testing error with a regression model trained on filtered measurements. (c) The testing error with a regression model trained on unfiltered measurements. Note that the mean error in (a) and (b) are very similar, while the mean error in (c) is significantly higher, indicating decreased generalizability due to high frequency sampling errors. Note the different x-axis scales.



(a) ground (b) sky

Figure 9. A comparison of regression performance using the PLS image representation on just sky pixels, just ground pixels, and the entire image. Each camera is represented as a point with coordinates and color determined by its cross-validation error. Performance is highly correlated, but we show example images on the right from cameras (a) and (b) that had lower error using just ground pixels and just sky pixels, respectively.

(a) (b) (c)

Figure 10. (a) The image representations with the lowest mean squared error for each camera. (b) Average cross-validated error for different regression methods using the best performing image representation for each camera. Outliers were ignored when computing the mean errors. (c) Interpolation error using different methods as compared to unfiltered satellite imagery. A baseline error, which always assumes completely clear skies, is shown as a red line. "Blur" indicates that the predictions were made using regression models trained on filtered samples.

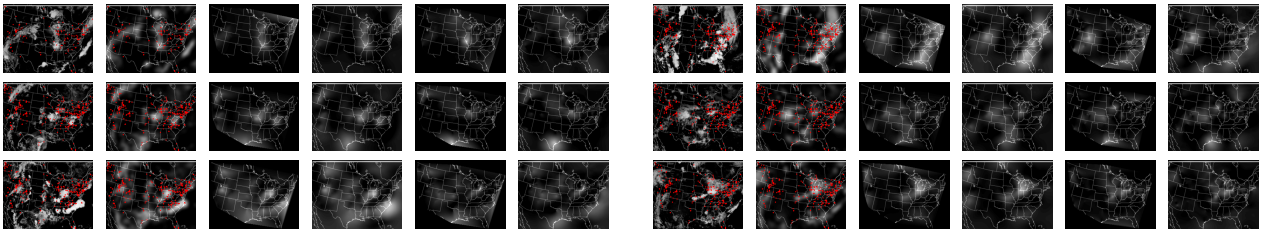


Figure 11. Final interpolation results. (a) Original satellite image. Camera locations are shown as red dots. (b) Filtered satellite image. (c) Natural neighbor interpolation using ideal measurements sampled from filtered satellite images. (d) Kriging interpolation using ideal measurements sampled from filtered satellite images. (e) Natural neighbor interpolation using values predicted from regression models trained on filtered measurements. (f) Kriging interpolation using values predicted from regression models trained on filtered measurements.

7. Conclusions

Throughout this paper, we presented a method for coordinating thousands of geo-located, outdoor webcams in order to estimate a cloud map with potential applications to weather forecasting. We presented a robust choice for low-dimensional image representations that retain cloudiness information. Using these features, we used random forests trained on historical satellite imagery to predict a sparse set of localized measurements, which were interpolated to fill in the remainder of the map. While unlikely to replace satellite imagery in the short term, the results demonstrate the plausibility of leveraging the vast, existing network of webcams for large-scale environmental monitoring.

One major obstacle in producing more accurate interpolations is the availability and distribution of webcams. Even ideal measurements sampled from actual satellite imagery can result in poor interpolation performance if there are large regions without webcams. Similarly, if a single prediction is incorrect, errors can propagate a large distance if there are no nearby measurements. While a larger dataset of geo-located cameras could alleviate these issues, knowledge of spatial and temporal correlations and prediction uncertainty could also result in improved performance.

References

[1] http://www.goes.noaa.gov. 6

[2] L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001. 4

[3] W. J. Broad and D. E. Sanger. China tests anti-satellite weapon, unnerving U.S. *New York Times*, 2007. 1

[4] D. J. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg. Mapping the world's photos. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 761–770, New York, NY, USA, 2009. ACM. 2

[5] N. Jacobs, W. Burgin, N. Fridrich, A. Abrams, K. Miskell, B. H. Braswell, A. D. Richardson, and R. Pless. The global network of outdoor webcams: Properties and applications. In *Proc. ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL GIS)*, Nov. 2009. 2, 6

[6] N. Jacobs, W. Burgin, R. Speyer, D. Ross, and R. Pless. Adventures in archiving and using three years of webcam images. In *IEEE CVPR Workshop on Internet Vision*, 2009. 2

[7] N. Jacobs, N. Roman, and R. Pless. Consistent temporal variations in many outdoor scenes. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, June 2007. 1, 2

[8] N. Jacobs, S. Satkin, N. Roman, R. Speyer, and R. Pless. Geolocating static cameras. In *Proc. IEEE International Conference on Computer Vision*, Oct. 2007. 2

[9] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan. Webcam clip art: Appearance and illuminant transfer from time-lapse sequences. *ACM Transactions on Graphics*, 28(5), December 2009. 2

[10] D. Leung and S. Newsam. Proximate sensing: Inferring what-is-where from georeferenced photo collections. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2010. 2

[11] Z. Lin, M. Chen, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *ArXiv e-prints*, 2010. 4

[12] M. A. Oliver and R. Webster. Kriging: a method of interpolation for geographical information systems. *International journal of geographical information systems*, 4(3):313–332, 1990. 6

[13] C. Reinsch. Smoothing by spline functions. *Numerische Mathematik*, 10:177–183, 1967. 5

[14] E. Riordan, E. Graham, E. Yuen, D. Estrin, and P. Rundel. Utilizing public internet-connected cameras for a cross- continental plant phenology monitoring system. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1501–1504, July 2010. 2

[15] R. Rosipal and N. Krmer. Overview and recent advances in partial least squares. In *in Subspace, Latent Structure and Feature Selection Techniques, Lecture Notes in Computer Science*, pages 34–51. Springer, 2006. 3

[16] T. Shanker. Pentagon is confident missile hit satellite tank. *New York Times*, 2008. 1

[17] R. Sibson. A brief description of natural neighbor interpolation. In V. Barentt, editor, *Interpreting mulitvariate data*, pages 21–26. John Wiley and Son, 1981. 6

[18] J. B. Tenenbaum, V. d. Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. 3

[19] D. Wright. Space debris. *Physics Today*, 60(10):35–40, 2007. 1

[20] L. Xie, M. A. Carreira-Perpinan, and S. Newsam. Semi-supervised regression with temporal image sequences. In *Proc. IEEE International Conference on Image Processing*, 2010. 2

[21] L. Xie and S. Newsam. Im2map: deriving maps from georeferenced community contributed photo collections. In *Proceedings of the 3rd ACM SIGMM international workshop on Social media*, pages 29–34, 2011. 2

[22] http://www.mathworks.com/matlabcentral/fileexchange/25948. 6