Exploratory Analysis of Time-Lapse Imagery with Fast Subset PCA

Austin Abrams, Emily Feder, Robert Pless
Washington University in St. Louis
One Brookings Drive, Campus Box 1045, St. Louis MO 63130
abramsa@cse.wustl.edu, {eafeder, pless}@wustl.edu

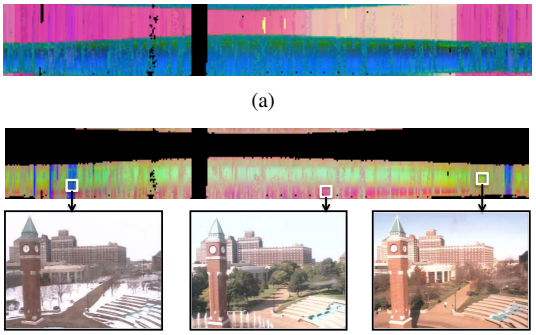
Abstract

In surveillance and environmental monitoring applications, it is common to have millions of images of a particular scene. While there exist tools to find particular events, anomalies, human actions and behaviors, there has been little investigation of tools which allow more exploratory searches in the data. This paper proposes modifications to PCA that enable users to quickly recompute low-rank decompositions for select spatial and temporal subsets of the data. This process returns decompositions orders of magnitude faster than general PCA and are close to optimal in terms of reconstruction error. We show examples of real exploratory data analysis across several applications, including an interactive web application.

1. Introduction

Projects like the Archive of Many Outdoor Scenes [8], Webcam Clip Art [11] and the Weather and Illumination Database [13] have documented efforts to find and archive webcam images. However, using these image archives for practical applications remains challenging. Although the global network of webcams have been used to monitor tree health [6] and geolocate cameras [9], the first step in both of these papers is to find a set of webcams that don't suffer from severe camera motion. In [5], the authors make use of webcams to estimate scene depth, but their algorithm assumes that the images being taken on a partly cloudy day. The process of exploring large archives of time-lapse imagery to discover interesting patterns is still very manual, often requiring hours of searching by an experienced professional.

This paper considers the problem of exploratory data analysis in large sets of time-lapse data captured from a fixed viewpoint. Exploratory data analysis seeks to provide tools to allow a user to explore volumes of data quickly to uncover intrinsic patterns. Part of the challenge of applying exploratory data analysis to an archive of webcam images is the sheer volume of the data: a year's worth of low-resolution webcam imagery occupies more than 200 MB over 16,000 images.



(a)

(b)

Figure 1. Exploratory data analysis tools support users looking for unknown patterns or events in the data. A recently proposed summary image (a) decomposes a year's worth of images from a webcam using 3 component PCA, and color codes each location in a time-of-day vs. time of year using those coefficients. The coding is dominated by large scale image changes (here, coding the illumination changes from day to night). This paper offers an interactive tool that pre-computes a many component PCA, then interactively chooses the 3-dimensional subspace that captures variation of any subset of pixels and images. Applied to daytime images and focused on a select group of pixels covering portions of the ground, the false color coding (b) highlights snowfall (in blue) and changes in leaf color (red and green).

Throughout this paper, we make use of principal component analysis (PCA) to discover trends in data and visualize patterns in imagery. PCA is an often-used tool for exploratory data analysis for large volumes of data, because very high-dimensional data can be visualized in a low-dimensional space, such as 2D plots or as a single pixel in an RGB summary image. Previously, PCA has been used on large sets of webcam imagery to find the dominant ways in which a particular webcam image changes over one or several years [8, 7]. Figure 1 shows one example visualization that captures the variation of a scene over the course of a year in a false color plot; the summary image captures

variation over times of day and times of year on different axes and color codes each location by the top 3 PCA coefficients computed for the image taken at each time (see Section 5.1 for more details).

However, PCA is a rather blunt instrument, capturing only the most important variations over the entire time-extent and over the entire image. The ability to detect local changes in the data, such as the blossoming of flowers, is hidden by global variations, such as the difference in appearance between night and day images. In the exploratory framework, it is useful to focus in on parts of the image or parts of the day or parts of the year that are most relevant. Classically, this would require recomputing the PCA components over the relevant subset of data, which is a time consuming task on tens of thousands of large images (even with fast, incremental PCA algorithms).

Although local changes will be dominated by global variations, if enough principal components are calculated up front, these local variations will appear perhaps as a mixture of the less important coefficients. This paper shows how, given arbitrary subsets of the original data source, we can rotate principal component bases to highlight the most important variations within a subset of the data. Computing this rotation is much faster than recomputing components from the subset, allowing real-time responsive behavior within a web application. The contributions of this paper are: deriving this rotation; showing, on a small but representative set of cameras, that rotating bases is almost as good as recomputing from original data; and highlighting the use of this to find interesting events in webcam archives. In this regard, we can visualize the (say) three top principal components and visualize how a particular spatial or temporal aspect of the data set varies, while still achieving a large enough speedup so that data exploration across tens of thousands of webcam images can be implemented interactively.

Section 2 reviews other papers that explore visualization of large scale data sets. Then, Section 3 presents the mathematical background and derivation for Fast Subset PCA, a new PCA formulation that quickly discovers variations in subsets of imagery. Section 4 discusses the experiments we performed to validate these derivations, and Section 5 describes the possible applications that follow from this work, as well as a web application that uses these results.

2. Previous Work

Other papers [2, 18] describe anomaly detection frameworks on long-term webcam images. These systems attempt to score the "normalcy" of the current frame of a webcam image in terms of previously-seen images, and can reliably select the few most anomalous images from a long-term time-lapse sequence. However, these papers leverage similarity scores across the entire data set, and so cannot perform anomaly detection on either spatial or temporal subsets of the original data set. In this paper, we develop a system for selecting exemplar images which give similar results to the abnormalities presented in [2], with the added ability to quickly analyze arbitrary spatiotemporal subsets of the original data.

A large effort has been made to condense large data sets to create meaningful images that provide an easy-to-visualize summary of the entire data set. In [14], Oh et al. survey a series of video cataloging tools, collectively known as video abstraction, including various tools to select the few frames that summarize long portions of a video sequence. Summarizing long sections of video sequences has been done by detecting motion in spatio-temporal volumes and compositing different temporal segments of the video into a single image describing the overall motion in the image [15]. In each case, the authors focus on summarizing image sequences with high refresh rates. In this paper, we aim to summarize time-lapse sequences, where any two frames in the sequence are taken half an hour apart. Several papers [3, 4, 16] discuss ways of compositing aesthetically-pleasing summary images by slicing spatiotemporal volumes of time-lapse imagery. These techniques can be applied for exploratory data analysis of time-series webcam data, but are more appropriate for at most a few days of imagery.

The authors of [12] presented the concept of a time-of-day vs. time-of-year plot, as in Figure 1. In this paper, they use their "Spatio-Temporal Irradiation Map" to model sunlight variations in architectural design. The authors of [10] also use similar maps to model solar intensity for lighting design.

In [7], Jacobs et al. document their efforts to make large webcam data sets more accessible to the research community. In the paper, the authors make use of PCA to generate year-long summary images, which describe how a webcam's image changes due to high-level daily and seasonal variations. This paper allows these plots to be interactively refocused on subsets of this data.

Our sample data is from camera time lapses archived and shared through the Archive of Many Outdoor Scenes (AMOS) [8].

3. PCA and Fast Subset PCA

In this section, we give a brief background on PCA, and show how to extend its formulation to quickly compute PCA decompositions for subsets of the original data. Through these derivations, we introduce Fast Subset PCA and show how it can be used to quickly find local spatial or temporal variations in the data.

3.1. Background

As applied to a sequence of images taken from a static camera, PCA reduces the high dimensionality of image space into a few basis images whose linear combinations approximately span the variation of initial images.

More formally, assume we are given some sequence of images $I_1, \ldots I_n \in \mathcal{R}^p$ (where p is the number of pixels and n is the number of images), as well as a predetermined number of bases b. Then, if $\vec{\mu}$ is the p-element mean image vector, then the goal is to find a set of basis images $U \in \mathcal{R}^{p \times b}$ and a set of coefficient vectors $V_1, \ldots V_n \in \mathcal{R}^b$ such that

$$I_i \approx \vec{\mu} + UV_i^\top \quad \forall\, i \in \{1, \ldots, n\}. \tag{1}$$

Typically, this is done by creating a data matrix $D = [I_1 | \ldots | I_n] \in \mathcal{R}^{p \times n}$, subtracting off $\vec{\mu}$ from each column, and performing the singular value decomposition (SVD) of D so that

$$D - \vec{\mu} \approx USV^\top \tag{2}$$

In a slight abuse of notation, we define the subtraction of a $p \times 1$ vector $\vec{\mu}$ from a $p \times n$ matrix D as the subtraction of $\vec{\mu}$ from every column of D.

Unless $b \geq n$ or $b \geq p$, then the above equation is only approximate; D can not be reconstructed perfectly from U, S, V, and $\vec{\mu}$. However, the SVD provides the best rank b approximation of the matrix D in terms of reconstruction error.

In future sections, we will derive approximations for U, S, and V that model the variation of spatial and temporal subsets in D, although not as accurately as the standard SVD. Because the SVD gives back the 'best' matrices for U, S, and V, we will treat these matrices as the optimal case in terms of low-rank reconstruction error.

3.2. Fast Subset PCA

In this section, we introduce *Fast Subset PCA*, or FSPCA, which takes advantage of precomputed global PCA decompositions to quickly and accurately compute local PCA decompositions for temporal and spatial subsets of the data. In our experiments, computing Fast Subset PCA is orders of magnitude faster than naïvely recomputing SVD over subsets of the original data set, while still achieving near-optimal accuracy.

In our formulation, we assume that we have been given a data set D of n images with p pixels each. We represent D as a $p \times n$ matrix where the ith column in the matrix is the ith image. As a first step, we use SVD to find

$$D - \vec{\mu} \approx USV^\top, \tag{3}$$

where $\vec{\mu}$ is the p-element vector containing the mean image, U is a $p \times b$ orthonormal matrix of basis vectors, S is a $b \times b$ diagonal matrix of singular values, V is an $n \times b$ orthonormal matrix of coefficient vectors, and b is the predetermined number of basis vectors. In our experiments, $b = 25$.

In practice, we deal with thousands of images that each have many thousands of pixels each, so to alleviate memory limitations, we perform an incremental variant of SVD [1] that never stores D all at once.

3.2.1 Selecting Subsets of Images

Suppose we now select some subset of images $N \subseteq \{1, \ldots, n\}$ from the dataset D, and we want to measure the intrinsic variance of this subset of the data. Let D_N be the $p \times |N|$ submatrix of D defined by taking columns out of D as specified in N:

$$D_N = [\ldots c_i \ldots]_{i \in N}\,, \text{ where } c_i \text{ is the } i\text{th column of } D. \tag{4}$$

The goal is to find a new $\hat{U}, \hat{S}, \hat{V}$, and $\hat{\mu}$ that capture the variation of the images in D_N, so that $D_N - \hat{\mu} = \hat{U}\hat{S}\hat{V}^\top$. The simplest solution is to simply perform PCA on this new data matrix D_N (as in Equation 3) to find the best $\hat{U}, \hat{S}, \hat{V}$ and $\hat{\mu}$. The resulting decomposition is guaranteed to find the basis vectors that minimize reconstruction error, and is thus the "best" decomposition. However, this additional computation can be very time intensive, since D_N is $p \times |N|$ and p is usually very large.

Instead, we can construct a new $|N| \times b$ coefficient matrix V' by selecting only the rows out of V that correspond to the selection N:

$$V' = \begin{bmatrix} \cdots \\ r_i \\ \cdots \end{bmatrix}_{i \in N}, \text{ where } r_i \text{ is the } i\text{th row of } V. \tag{5}$$

Suppose that N was chosen in a way that eliminated some aspect of variation from the set of images (for example, only choosing the daytime images). Then, the columns of V' will cover a smaller set of coefficient space than V did. To capture the variance in this new coefficient space, we perform a second-order PCA step on $SV'^\top$ to find which ways the reduced coefficient space vary. By denoting the mean of all columns of $SV'^\top$ as $\vec{\mu}_{SV}$, we get:

$$SV'^\top - \vec{\mu}_{SV} = U_N S_N V_N^\top, \tag{6}$$

Since the size of $SV^\top$ is relatively small compared to D_N, this decomposition is much faster than recomputing the full SVD for D_N. Whereas before we only computed the SVD up to b basis vectors, here we can quickly compute the full SVD and decompose $SV'^\top$ exactly (so that U_N and S_N are both $b \times b$ and $V_N^\top$ is $b \times |N|$). By rearranging Equation 6, we can apply this result to Equation 3:

$$D_N - \vec{\mu} + U\vec{\mu}_{SV} \approx UU_N S_N V_N^\top \tag{7}$$

(a) Camera 90 (b) Camera 4 (c) Camera 33

Figure 2. Example images from each camera in the experimental dataset. Cameras 4 and 33 are shown with the subset of pixels that FSPCA uses when using spatial subsets of the data.

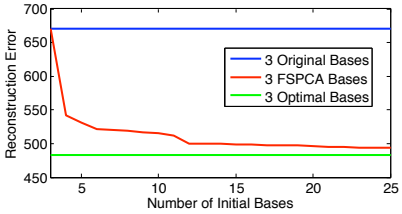


Figure 3. Reconstruction error of FSPCA from 3 bases as a function of b, the initial number of bases (red), with respect to the original PCA bases (blue, see Equation 3) and optimal bases (green). This shows that as the number of initial bases increase, the mean squared reconstruction error over all pixels and all images from FSPCA asymptotically approaches optimal, and using 25 initial bases gives results close to optimal.

Using this approach, we find $\hat{U} = UU_N$, $\hat{S} = S_N$, $\hat{V} = V_N$, $\hat{\mu} = \vec{\mu} - U\vec{\mu}_{SV}$. Not only is U_N orthonormal (by definition from SVD), it is square, so we can treat U_N as a b-dimensional rotation. Geometrically, this derivation shows that Equation 7 uses U_N to rotate U to more adequately cover the coefficient space of V'.

3.2.2 Selecting Subsets of Pixels

The same approach can be taken to find approximate PCA bases and coefficients that capture the variance from a selected portion of pixels in a scene.

Assume that we have found $\vec{\mu}, U, S$, and V over the entire dataset D, as in Equation 3. Given some $P \subseteq \{1, \ldots, p\}$ as subset of pixels from the image, the goal is to find $\hat{\mu}, \hat{U}, \hat{S}$, and $\hat{V}$ that best explain the variation found in D_P, the submatrix of D defined by selecting only the P rows from D:

$$D_P = \begin{bmatrix} \cdots \\ r_i \\ \cdots \end{bmatrix}_{i \in P}, \text{ where } r_i \text{ is the } i\text{th row of } D. \quad (8)$$

Similarly, we define $\vec{\mu}_P$ as the subvector of $\vec{\mu}$ whose elements are defined by P. Then, we construct U' as the $|P| \times b$ submatrix of U taken by the selection of P.

$$U' = \begin{bmatrix} \cdots \\ r_i \\ \cdots \end{bmatrix}_{i \in P}, \text{ where } r_i \text{ is the } i\text{th row of } U. \quad (9)$$

ID	Start	End	n	p	size
90	12/1/08	12/31/08	1,470	640×360	115 MB
4	4/1/08	6/30/08	4,367	352×240	42 MB
33	1/1/07	12/31/07	16,508	320×240	226 MB

Figure 4. Statistics about the cameras used in the experimental results.

ID	$\lvert N\rvert$	$\lvert P\rvert$	PCA time (sec)	FSPCA time (sec)	Optimal time (sec)
90	169	-	893.1	0.24	91.60
4	-	6900	882.0	0.36	174.32
33	8736	4875	3264.69	0.78	336.48

Figure 5. Statistics about the computation time to compute PCA for each method. Here, $|N|$ is the size of the temporal subset, and $|P|$ is the size of the spatial subset, and the last three columns show the time taken to compute PCA bases for each method. Cells that are dashed out correspond to selecting all elements along a spatiotemporal dimension (so $|N| = n$ or $|P| = p$).

We find the SVD of $U'S$ as $U'S = U_P S_P V_P^\top$, which gives the ways in which $U'S$ vary. Then, we can apply this result to Equation 3:

$$D_P - \vec{\mu}_P = U_P S_P V_P^\top V^\top \quad (10)$$

Now, we find that $\hat{U} = U_P$, $\hat{S} = S_P$, $\hat{V}^\top = V_P^\top V^\top$, and $\hat{\mu} = \vec{\mu}_P$. Note that, as before, $V_N^\top$ is a square orthonormal matrix, so this transformation effectively rotates the coefficient space of the original decomposition so that the new decomposition better explains the variation found in D_P.

3.2.3 Spatial and Temporal Subsets

Finally, these methods can be composed to find decompositions that explain the variance in both spatial and temporal subsets of the original data. Each FSPCA decomposition takes as input U, S, V, and μ returns a new $\hat{U}, \hat{S}, \hat{V}, \hat{\mu}$ that better account for the variation in some spatial or temporal subset of the data. So, to find the FSPCA decomposition for a spatial and temporal subset of the original data, we first perform FSPCA on the spatial subset, and then again on the temporal subset (or vice versa).

4. Experiments

To validate the work in Section 3, we use data available from AMOS and compute PCA bases and coefficients using $b = 25$ original components. The small set of selected cameras point in different geographic directions and have different distances to objects in the scene. Figure 2 shows an example image from each camera, Figure 4 gives some statistics about the sizes of each data set, and Figure 5 shows

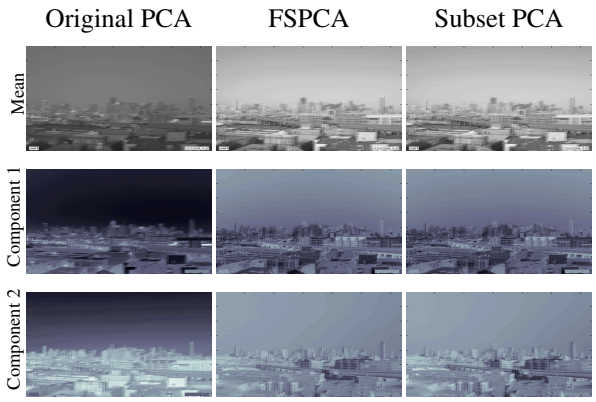


Figure 6. Comparing the mean, first principal component, and second principal component between the original set of PCA bases (left column), FSPCA bases (middle column), and optimal bases over the subset (right column).

that in all experiments, computing FSPCA is substantially faster than recomputing PCA over the new subset.

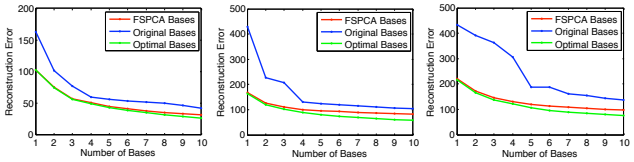
The higher b is, the better our FSPCA bases will be, in terms of reconstruction error with respect to the optimal case. Figure 3 shows that, in practice, the reconstruction error of FSPCA asymptotically approaches optimal as b increases, and that $b = 25$ is appropriate for our experiments.

We first chose a subset of clear daytime images from camera 90 and applied the FSPCA algorithm. Figure 7(a) shows the reconstruction error of FPSCA with respect to the optimal case. When reconstructing images from 3 basis vectors, the FSPCA bases gave a reconstruction error 2.1 percent worse than optimal, while the original bases gave a reconstruction error more than 38 percent worse than optimal.

When running FSPCA, we calculate a new mean and set of basis vectors as described in Equation 7. Figure 6 shows that these new means and basis vectors are semantically meaningful, in that they capture high-level variation from the selected subset of images.

Next, we ran PCA on camera 4, and then applied FSPCA to a mask of pixels covering the ground regions of the scene, as shown in Figure 2(b). Figure 7(b) shows the reconstruction error using this method. When reconstructing images from 3 basis vectors, the FSPCA bases gave a reconstruction error 1.095 times the optimal error, while the original bases gave a reconstruction error twice as large as in the optimal case.

As a final experiment, we computed PCA on camera 33, and then chose subsets both spatially, as shown in Figure 2(c), and temporally (only selecting daytime images). Figure 7(c) shows the reconstruction error for this experiment. Reconstructing with 3 FSPCA bases gave a reconstruction error 6.0 percent worse than optimal, and the reconstruction from 3 original bases gave a reconstruction error 164 percent worse than optimal.



(a) Camera 90 (b) Camera 4 (c) Camera 33

Figure 7. Measuring squared reconstruction error per pixel per image as a function of number of bases for each experiment, using global PCA bases (blue), FSPCA bases trained from $b = 25$ original bases (red), and optimal bases (green).

While reconstruction is not the main purpose of this method, having a low reconstruction error gives confidence that using the top 3 components in the FSPCA rotated basis is nearly as good as PCA applied to just that subset, and therefore is effective for visualization purposes.

5. Example Applications

In the previous section, we show that FSPCA can quickly and accurately approximate PCA bases over arbitrary spatiotemporal subsets of the original data set. Thus, FSPCA is a useful tool for data visualization, because it allows users to quickly perform ad-hoc queries on large sets of imagery and visualize the first few components of decompositions tied to that area. In this section, we provide sample data visualization applications that leverage the previous results.

5.1. Year-Long Summary Images

In order to quickly visualize the results from a PCA decomposition, we can construct year-long summary images that describe the major modes of variation across a year. To create a summary image, we create a $k \times 365$ RGB image, where k is the number of images taken in a day (for our experiments, $k = 48$). Thus, the ith image from any given year corresponds to the ith pixel in the summary image for that year. In our experiments, we set the value of pixel i to be the projection of the first three normalized PCA coefficients onto the RGB cube. Using PCA coefficients in year-long summary images, as in [7], tends to highlight daily and seasonal changes through time; an example is provided in Figure 1(a).

If the original set of PCA coefficients are displayed in a year-long summary image, then the set of coefficients in RGB have to capture the 3 most dominant changes that take place over the span of a year. Most likely, the first component will capture the difference between night and day, and the next two components will capture variations in sun positions through the day. However, as was done with camera 33, we can select only the daytime images from a set of the original images, and this will eliminate one mode of variation from the data. By further refining the data to only

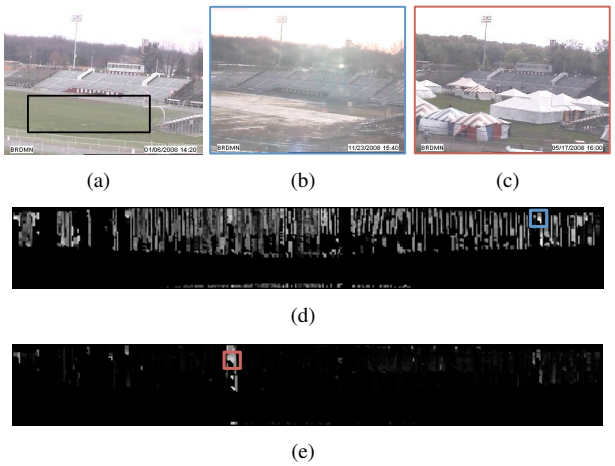


Figure 8. (a) shows an example webcam image taken from a year's worth of imagery, along with the pixels selected for FSPCA. We run FSPCA over these pixels, as well as only the daytime images. (d)-(e) Year-long reconstruction error images, where brighter pixels correspond to a higher reconstruction error in the spatiotemporal subset of data; (d) was generated using the original PCA decomposition, and (e) was generated using FSPCA. (b)-(c) Example images from each summary, shown as blue and red squares in the summary images.

contain some of the pixels (as shown in Figure 2), such as a tree in the image, other modes of variation will disappear, and subtler changes in the data become more apparent. Figure 1 shows the year-long PCA summary from camera 33, before and after we use FSPCA to remove spatial and temporal subsets of the original data set.

These summary images can also be used to visualize reconstruction error across large data sets. By coloring each pixel in the image by the corresponding reconstruction error, we can quickly visualize which images are not well-explained by PCA, and are thus the most anomalous. Figure 8 shows that using FSPCA with year-long reconstruction error summary images is an easy way to spot anomalous behaviour in spatiotemporal subsets of the original data set.

5.2. Iterative Refinement

Many webcams "in the wild" experience severe camera motion, due to camera operators deliberately changing their orientation, or strong weather conditions. The dominant modes of variation from the resulting image sequence will come from camera motion, rather than changes within the scene itself, as noted by [7]. Although the top few of these coefficients only account for global changes in the scene, they may provide a first step in selecting new images upon which to run FSPCA.

If a user wants to view only the modes of variation that occur under a single camera orientation, then he or she can select a seed image, select the closest images (in terms of coefficient space), and perform FSPCA to remove the other camera orientations. We can use this iterative refinement algorithm to quickly remove unwwanted modes of variation.

To find the set of images which are closest to a seed image, we perform k-means clustering on the 3-element coefficient space from PCA, with $k = 2$. Then, we select the cluster of images containing the seed image and perform FSPCA on this subset. This process can be repeated iteratively, so that users can quickly refine their selection by choosing appropriate seed images at each level. Figure 9 shows a camera with major camera motion, and how iterative refinement can be used to quickly remove unwanted variation. In this case, the user stops when selecting images from only sunny days with no camera motion. Because it eliminates major forms of camera motion, this process can be used as preprocessing for algorithms that make the static camera assumption, such as factored time-lapse video [17] or camera geolocation [9].

5.3. Selecting Exemplar Images

In order to quickly visualize the space of images that the camera can view, we propose a method to select the top k exemplar images from a large data set. Choosing the most anomalous (least well-reconstructed) images most often finds camera failures, raindrops on the lens, and lens flare, so we consider instead points whose coefficients are extreme. In practice, most of the images in a time-lapse sequence will be 'boring', where the image doesn't deviate far from the mean image. However, some images contain large-scale events which deviate from the mean but are still well-reconstructed. These images represent the outer boundary of what the PCA decomposition can express, and finding these exemplar images tells what kinds of extreme variations the data set can account for, regardless of how often these variations naturally occur.

To find the top k exemplar images, we follow [7], and employ a greedy algorithm to approximate a maximally separated set. We first find the image whose PCA coefficients are farthest from the origin and add it to the exemplar set. Then, incrementally add images whose coefficients are farthest from any existing exemplar's coefficients. By constructing the exemplar set this way, we include only the images that are far away from each other in coefficient space.

If one uses this exemplar selection algorithm with the original formulation of PCA, then the resulting top images will show the exemplar images when considering all pixels in the image. However, if the user masks off some set of pixels in the scene, and performs FSPCA, the resulting exemplar images will account for only that section of the image. This would be useful for those wanting to study the variations from only a certain portion of the scene (e.g., only one direction of traffic on a traffic camera), rather than

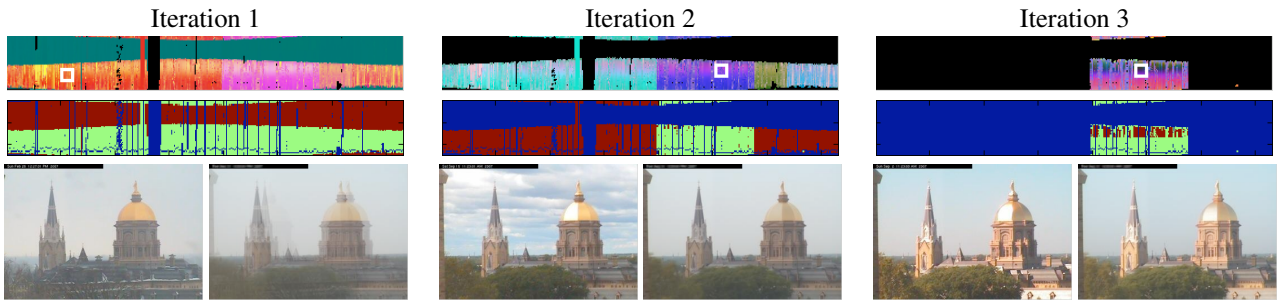


Figure 9. Using FSPCA for iterative refinement of a scene with substantial camera motion. Each iteration shows: the year-long PCA summary image (top); the k-means clustering summary image, where red and green represent the two clusters and blue is an absence of data (middle); the selected image (lower-left, shown as a white box in the summary image); and the mean of all images in the selected cluster (lower-right).

across the entire image. Figure 11 shows that FSPCA can be used with this method to find the extreme variations found in a subset of the original data, in this case, capturing the graduation event in May 2008.

5.4. Web Application

We have developed a web application that allows users to specify spatial or temporal subsets from a year of webcam images upon which to perform FSPCA. Jobs can be specified through the web browser by dragging rectangular areas on exemplar webcam images or year-long summary images. The FSPCA bases are then calculated and displayed to the user as a year-long summary image. Because PCA basis vectors from the full data set have been pre-computed, any FSPCA request can be completed almost immediately, which gives the user incredibly fast tools to refine the set of images both spatially and temporally. Furthermore, the user can also select an image from the data set and ask for all of the images which are less than some thresholded distance to the selected image in coefficient space. Figure 10 describes the web interface and an example summary image.

6. Conclusion

In this paper, we present a new framework to allow users to quickly visualize PCA decompositions for arbitrary subsets of time-lapse imagery. This method, which takes advantage of a large number of precomputed PCA basis vectors, is orders of magnitude faster than naïvely recomputing PCA for arbitrary subsets of the data, while still achieving near-optimal results in terms of reconstruction error. Fast Subset PCA can be used for a variety of visualization applications, such as creating year-long summary images, quickly removing modes of variation from the data set (including substantial camera motion), and selecting exemplar images with respect to certain areas of the image. These applications allow users to quickly and easily explore long-term time-lapse imagery, where there may be tens of thousands of images taken in a year.

(a)

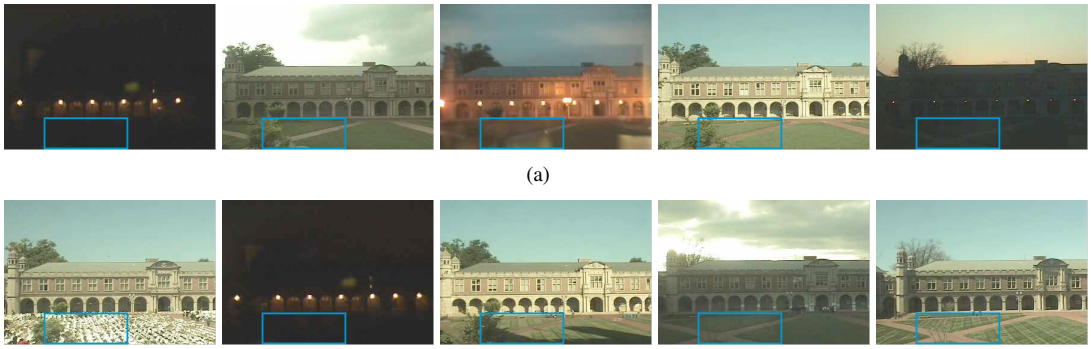
(b)

Figure 10. A screenshot from the web interface, where a user has marked a section of the image and a section from the year-long summary image of mean values. (b) The two exemplar images show that the PCA coefficients have coded for the blossoming of flowers.

References

[1] M. Brand. Incremental singular value decomposition of uncertain data with missing values. In *European Conference on Computer Vision*, pages 707–720, 2002.

[2] M. Breitenstein, H. Grabner, and L. Van Gool. Hunting Nessie - real-time abnormality detection from webcams. In *Computer Vision Workshops (ICCV Workshops)*, pages 1243 –1250, Sep. 2009.



(a)

(b)

Figure 11. The top 5 exemplar images found using (a) the original PCA decomposition and (b) FSPCA over the pixels highlighted in the blue rectangle. The first set of exemplar images account for variation across the entire image, while FSPCA finds variation within the selected region.

[3] M. F. Cohen, A. Colburn, and S. Drucker. Image stacks. Technical report, Microsoft Research, 2003.

[4] M. T. Gabriel, M. Terry, G. J. Brostow, G. Ou, J. Tyman, and D. Gromala. Making space for time in time-lapse photography. In *International Conference on Computer Graphics and Interactive Techniques*, 2004.

[5] N. Jacobs, B. Bies, and R. Pless. Using cloud shadows to infer scene structure and camera calibration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2010.

[6] 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 *ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL GIS)*, Nov. 2009.

[7] 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*, June 2009.

[8] N. Jacobs, N. Roman, and R. Pless. Consistent temporal variations in many outdoor scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2007.

[9] N. Jacobs, N. Roman, and R. Pless. Toward fully automatic geo-location and geo-orientation of static outdoor cameras. In *IEEE Workshop on Applications of Computer Vision (WACV)*, Jan. 2008.

[10] S. Kleindinest, M. Bodart, and M. Andersen. Graphical representation of climate-based daylight performance to support architectural design. *LEUKOS - The Journal of the Illuminating Engineering Society of North America*, 5:39 –61, July 2008.

[11] 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 (SIGGRAPH Asia 2009)*, 28(5), December 2009.

[12] J. Mardaljevic. Precision modelling of parametrically defined solar shading systems: Pseudo-Changi. In *Eighth International IBPSA Conference*, August 2003.

[13] S. Narasimhan, C. Wang, and S. Nayar. All the Images of an Outdoor Scene. *European Conference on Computer Vision (ECCV)*, III:148–162, May 2002.

[14] J. H. Oh, Q. Wen, S. Hwang, and J. Lee. Video abstraction. In *Video Data Management and Information Retrieval*, pages 321–346, 2004.

[15] Y. Pritch, A. Rav-acha, A. Gutman, and S. Peleg. Webcam synopsis: Peeking around the world. In *International Conference on Computer Vision (ICCV)*, 2007.

[16] R. Raskar, A. Ilie, and J. Yu. Image fusion for context enhancement and video surrealism. *Non-Photorealistic Animation and Rendering (NPAR)*, pages 85–152, 2004.

[17] K. Sunkavalli, W. Matusik, H. Pfister, and S. Rusinkiewicz. Factored time-lapse video. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 26(3), Aug. 2007.

[18] H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages II: 819–826, 2004.