AUTOMATICALLY ORGANIZING DIGITAL PHOTOGRAPHS USING TIME AND CONTENT

Matthew Cooper, Jonathan Foote, and Andreas Girgensohn

FX Palo Alto Laboratory
3400 Hillview Ave. Bldg. 4
Palo Alto, CA 94304
{cooper, foote, andreasg}@fxpal.com

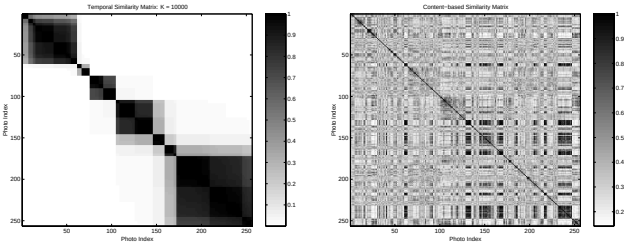
ABSTRACT

We present similarity-based methods to cluster digital photos by time and image content. This approach is general, unsupervised, and makes minimal assumptions regarding the structure or statistics of the photo collection. We describe versions of the algorithm using temporal similarity with and without content-based similarity, and compare the algorithms with existing techniques, measured against ground-truth clusters created by humans.

1. INTRODUCTION

Digital cameras are coming into widespread use, and allow users to amass increasingly larger collections of digital photographs. There is thus a demand for automatic tools to help users manage, organize, and browse these collections. Unlike film, digital photographs typically include meta-data, such as the time and date, in a standard image header such as Exif (EXchangeable Image File [1]). In a recent report, Stanford researchers have found that organizing photos by time significantly improves users' performance in a series of retrieval tasks [2]. Furthermore, consumers often wish to organize their photos in terms of "events" both for browsing and retrieval, as well as for sharing selected subsets of photos with others. Events are difficult to define quantitatively or consistently, but most commonly, photographs from the same event were taken in relatively close proximity in time. Events tend to exhibit little coherence in terms of low-level image features, and it is not uncommon for visually dissimilar photos to belong to the same event. For example, pictures from a trip to the beach could include photos of widely different subjects (beach, ocean, vehicle) taken at at different times of day.

To start, we examine clustering the photos by timestamp alone. We adapt a similarity-based media segmentation algorithm [3, 4] to hierarchically cluster photographs with similar (i.e. proximal) timestamps. This approach makes no assumptions about the distribution of the timestamps. We then extend the algorithm to consider content-based features in addition to temporal data. This approach can be extended to include any other types of meta-data or image features which may prove useful. We also note that this clustering technique can be applied to any time-ordered data given a measure of similarity.



(a) (b)

Fig. 1. Panel (a) shows a temporal similarity matrix for 256 photographs. Panel (b) shows the corresponding image similarity matrix, computed from DCT coefficients. Darker hues indicate increasing similarity.

2. RELATED WORK

Automatic digital photo organization has received increased attention in recent years. The algorithms in [2, 5] operate using an adaptive local threshold applied to the inter-photo time interval. Researchers at Kodak have developed an event segmentation algorithm based on clustering time differences using a two class version of K-means [6]. All time differences in the cluster with the greater mean are labelled as event boundaries. Our approach is similar in spirit to the scale-space media segmentation approach of [7] but is coarse-to-fine and doesn't require segment boundaries to be "traced back" from smaller scales to larger scales. Our approach is multi-resolution, and uses a similarity-based con-

fidence measure to assess clustering performance at the different resolutions. Clustering the photos at varying time resolutions can also enable flexible user interfaces, allowing users to organize their photo collections at different time scales.

3. ALGORITHMIC DETAILS

3.1. Pre-processing

For each photo, the Exif headers are processed to extract the timestamp (if Exif information is not available, we rely on the modification time of the digital image file, which will not generally be reliable for photo clustering). The N photos in the collection are then ordered in time so the resulting timestamps, $\{t_i : i = 1, \cdots, N\}$, satisfy $t_1 \leq t_2 \leq \cdots \leq t_N$. (N.B. Throughout, we index sequences and matrices by photo index in time order, not by absolute time.) For content analysis, we transform each photo to the Ohta colorspace [8] and compute the discrete cosine transform (DCT) of each channel. For each photo, we concatenate the 25 low frequency DCT coefficients from each channel to form a set of time-ordered feature vectors: $\{v_1, \cdots, v_N\}$. Any features which consistently quantify similarity can be substituted or integrated into the analysis. The sole requirement is that similar images produce similar features.

3.2. Distance matrix embedding

We use a multi-scale approach to determine the temporal structure in the photo collection. Using the timestamps, we construct $N \times N$ similarity matrices according to

$$\mathbf{S}_T^{(K)}(i,j) = \exp\left(-\frac{|t_i - t_j|}{K}\right). \quad (1)$$

To measure the image similarity between photos, we use a similarity measure based on exponential the cosine distance between the photos' DCT features:

$$\mathbf{S}_C(i,j) = \exp\left(\frac{< v_i, v_j >}{\|v_i\|\|v_j\|} - 1\right) \quad . \quad (2)$$

Fig. 1 shows example similarity matrices computed from 256 digital photographs taken over five months' time. Panel (a) shows the temporal similarity matrix of Eq. (1) for $K = 10000$ minutes, while panel (b) shows the image similarity matrix computed from Eq. (2). The time-ordered index runs along the rows (top to bottom) and columns (left to right) of the matrices. Dark blocks of high similarity along the main diagonal indicate clusters of sequentially similar photographs. Corners between the dark squares along the main diagonal indicate boundaries between two groups of photos.

The parameter K in Eq. (1) controls the sensitivity of the exponential similarity measure. By varying K, we assess temporal similarity over different time extents. For joint temporal and content-based clustering, we form a second family of similarity matrices indexed by K:

$$\mathbf{S}_J^{(K)}(i,j) = \begin{cases} \mathbf{S}_T^{(K)}(i,j) & \text{if } |t_i - t_j| > 48 \text{ hours} \\ \max(\mathbf{S}_C(i,j), \mathbf{S}_T^{(K)}(i,j)) & \text{otherwise.} \end{cases} \quad (3)$$

By construction, this matrix emphasizes temporal similarity, which we generally find to be most reliable for organization. However, image similarity can dominate for photos with sufficient temporal proximity. The left column of Fig. 2 shows three similarity matrices computed using Eq. (3) for different values of K. As expected, coarser clusterings of the photos are visualized in the matrices for larger values of K. As K decreases, finer dissimilarities between groups

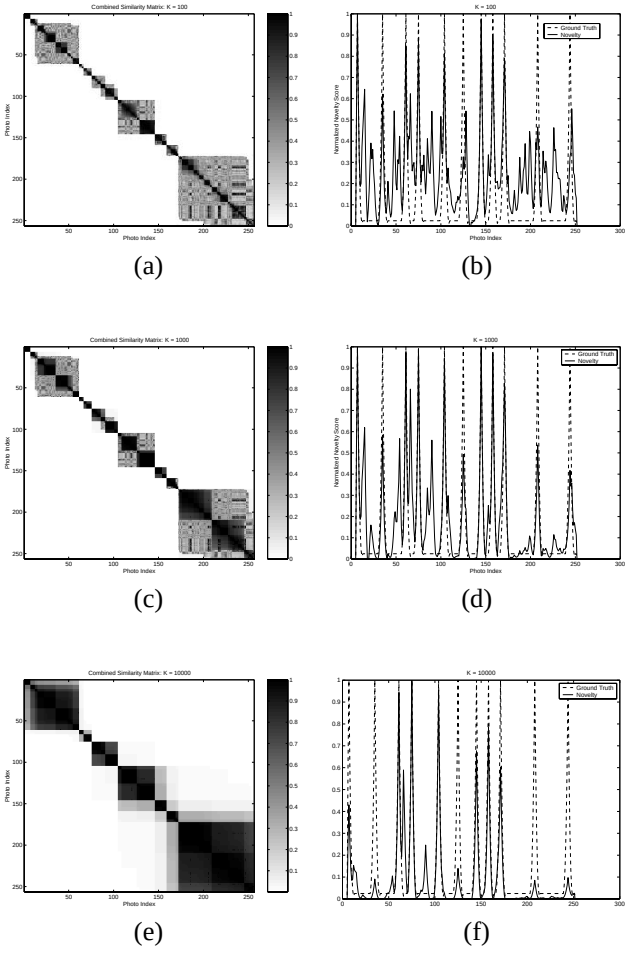


Fig. 2. The left column shows the similarity matrices $\mathbf{S}_J^{(K)}$ for $K = 100$ (a), $K = 1000$ (c), and $K = 10000$ (e). Panels (b), (d), and (f) show the corresponding novelty scores computed using a gaussian checkerboard kernel (solid), with the ground truth novelty score (dashed).

of timestamps become apparent, and the content-based similarity is more prominent.

3.3. Photo clustering

In Fig. 1, clusters are visible to the eye as dark blocks on the main diagonal. To cluster the collection into groups of similar photos, we travel along the diagonal and calculate a measure of how much a particular region looks like a boundary, that is like a 2×2 checkerboard [3]. This is done using a matched filter approach: we correlate a Gaussian-tapered 11×11 checkerboard kernel, denoted g, along the main diagonal of each $\mathbf{S}_J^{(K)}$ to calculate the "novelty score"

$$\nu_K(i) = \sum_{l,m=-5}^{5} \mathbf{S}_J^{(K)}(i+l, i+m) g(l,m) \quad . \tag{4}$$

(For clustering, we need only compute the portion of the similarity matrix around the main diagonal with the same width as the kernel, reducing computational complexity to order N.)

The right column of Fig. 2 shows the novelty scores computed for $K = 10^2, 10^3, 10^4$ minutes. While the matrices reveal structure at different resolutions, the peaks in the corresponding novelty scores (solid plots) comprise a set of cluster boundaries between contiguous groups of similar photos. In the plots of the right column, the ground truth novelty score is the superimposed dashed plot. The ground truth score is computed from a binary similarity matrix whose $(i,j)^{th}$ element is one if photos i and j were placed in the same event folder by the photographer and zero otherwise.

For clustering, we locate peaks in the novelty score at each scale (K), performing the analysis from coarse scale to fine (decreasing K). To build a hierarchical set of event boundaries, we include boundaries detected at coarse scales in the boundary lists for all finer scales. At each scale we detect peaks by finding zeros in the first difference of the novelty score. We threshold detected peaks as a function of the maximum novelty for a data-independent approach.

3.4. Selecting a "best" scale

This procedure results in a list of cluster boundaries and strengths at multiple resolutions. Ultimately, we wish to present users with the boundaries from a the single, best, resolution level. To determine the "goodness" of the boundaries at a given time scale, we calculate a confidence measure from the average within-class similarity and the between-class dissimilarity of the data. Denote the detected boundaries at each level, $\mathcal{B}^{(K)} = \{b_1, \cdots, b_{n_K}\}$, indexed by photo: $\mathcal{B}^{(K)} \subset \{1, \cdots, N\}$. For convenience, assume that $b_1 = 1$ and $b_{n_K} = N$. We then compute the confidence score

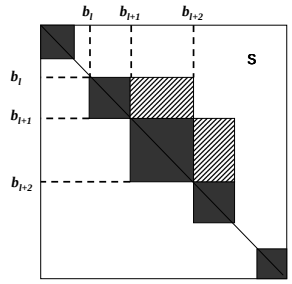


Fig. 3. Computing a confidence score for clustering. The dark regions represent within-cluster similarity, while the gray regions represent between-cluster similarity.

$$\begin{aligned} C(\mathcal{B}^{(K)}) &= \sum_{l=1}^{|\mathcal{B}_K|-1} \sum_{i,j=b_l}^{b_{l+1}} \frac{\mathbf{S}_J^{(K)}(i,j)}{(b_{l+1}-b_l)^2} \\ &- \sum_{l=1}^{|\mathcal{B}_K|-2} \sum_{i=b_l}^{b_{l+1}} \sum_{j=b_{l+1}}^{b_{l+2}} \frac{\mathbf{S}_J^{(K)}(i,j)}{(b_{l+1}-b_l)(b_{l+2}-b_{l+1})} \end{aligned} \tag{5}$$

The first term above quantifies the average within-class similarity between the photos within each cluster. The second term quantifies the average between-class similarity between photos in adjacent clusters. By negating this term, the confidence measure thus combines each cluster's average self-similarity and the dissimilarity between adjacent clusters. Fig. 3 illustrates the idea graphically. The within-class similarity terms are the means of the terms of darker regions along the main diagonal. The between-class terms are the means of the off-diagonal gray regions. Algorithm 1 details the computational steps. We have also experimented with a purely time-based approach in which $\mathbf{S}_J^{(K)}$ is replaced by $\mathbf{S}_T^{(K)}$ in the algorithm.

Algorithm 1 [Hierarchical Photo Clustering]

1. *Extract and sort photo timestamps, $\{t_1, \cdots, t_n\}$, and compute DCT features $\{v_1, \cdots, v_N\}$.*
2. *For each K in decreasing order*
   (a) *Compute the similarity matrix $\mathbf{S}_J^{(K)}$ using Eq.* (3).
   (b) *Compute the novelty score ν_K of Eq.* (4).
   (c) *Detect peaks in the novelty score.*
   (d) *Form event boundary list using event boundaries from previous iterations and newly detected peaks.*
3. *Compute confidence score using list of event boundaries, $\mathcal{B}^{(K)}$ for each K following Eq.* (5).
4. *Select event boundary list for K maximizing the confidence score.*

Table 1. The table summarizes our experimental results.

Collection I			
Alg.	Precision	Recall	F-score
[6]	0.39	1.0	0.56
[2]	0.38	1.0	0.55
Thresh.	0.72	0.95	0.82
Time	0.79	0.88	0.83
Joint	0.9	0.79	0.84

Collection II			
Alg.	Precision	Recall	F-score
[6]	0.42	1.0	0.6
[2]	0.29	1.0	0.45
Thresh.	1.0	0.85	0.92
Time	0.77	0.94	0.85
Joint	0.84	0.89	0.86

4. EXPERIMENTAL RESULTS

For evaluation, we applied five algorithms to two separate photo collections. Collection I consists of 1036 photos taken over 15 months, and Collection II consists of 413 photos taken over 13 months; all photos had accurate timestamps. Photos were assigned to meaningful events by the respective photographers. Photos in each event were sequential, and event classifications were used as ground truth for our clustering experiments. We compare our algorithms to two other approaches described in the literature [2, 5] and to a simple (constant) threshold. [2] and [5] compare the time difference between successive photographs to a variable threshold based on the logarithm of the average inter-photo time difference over a local window. Event boundaries occur where the time difference between photos exceeds the threshold. To determine if this worked better than simple thresholding, we skipped the thresholding step and looked at the first level of the hierarchy it created. For each algorithm, the precision, recall [9], and F-score[1] for the detected event boundaries are presented in Table 1. The "Thresh.", "Time", and "Joint" algorithms refer to the simple threshold, purely temporal, and joint time-content versions of Algorithm 1, respectively. Note also that the threshold is manually selected to maximize the F-score for [2], [5], and the simple thresholding, while the similarity-based approaches are fully automatic.

At most time intervals, the self-similarity algorithms slightly outperform the simple threshold, with the added advantage of not requiring an *a priori* threshold. It also supplies a hierarchy of event boundaries. The joint and temporal similarity algorithms exhibit almost identical performance, suggesting that temporal analysis alone may be sufficient.

[1]The F-score is computed for given precision p and recall r as F-score $= (2 \times p \times r)/(p + r)$.

5. SUMMARY

Our approach has significant advantages over existing techniques. Besides integrating temporal and image content information, our approach does not rely on a preset threshold and should generalize better to different image collections. While existing approaches typically only consider the similarity between adjacent photos (such as comparing their time difference to a threshold), our novelty measure is based on similarity comparisons between *all possible* photo pairs in a local neighborhood. Ultimately, this more comprehensive analysis will provide more robust clustering.

6. REFERENCES

[1] Digital Still Camera Image File Format Standard. Japan Electronic Industry Development Association, 1998. http://www.pima.net/standards/it10/PIMA15740/exif.htm

[2] A. Graham, H. Garcia-Molina, A. Paepcke, and T. Winograd. "Time as the essence for Photo Browsing Through Personal Digital Libraries." *Proc. Joint Conf. on Digital Libraries*, 2002.

[3] J. Foote, "Automatic Audio Segmentation using a Measure of Audio Novelty." *Proc. IEEE Intl. Conf. on Multimedia and Expo*, 2000.

[4] M. Cooper and J. Foote. "Scene Boundary Detection Via Video Self-Similarity Analysis." *Proc. IEEE Intl. Conf. on Image Processing*, 2001.

[5] J. Platt, M. Czerwinski, and B. Field. "PhotoTOC:Automatic Clustering for Browsing Personal Photographs." Microsoft Research Techincal Report MSR-TR-2002-17, 2002.

[6] A. Loui and A. Savakis. "Automatic Image Event Segmentation and Quality Screening for Albuming Applications." *Proc. IEEE Intl. Conf. on Multimedia and Expo*, 2000.

[7] M. Slaney, D. Ponceleon, and J. Kaufman. "Multimedia Edges: Finding Hierarchy in all Dimensions." *Proc. ACM Conf. on Multimedia*, 2001.

[8] Y-I Ohta, T. Kanade, and T. Sakai. "Color Information for Region Segmentation." *Comp. Graphics & Image Processing*, **13**:222-241, 1980.

[9] R. Korfhage. *Information Storage and Retrieval*. John Wiley, 1997.