Capture-time Classification of Mobile Sunset Photos Leveraging Strong Spatiotemporal Cues

Ali Almajed
Computer Science and Software Engineering
Rose-Hulman Institute of Technology
Terre Haute, IN
almajeas@gmail.com

Matthew Boutell
Computer Science and Software Engineering
Rose-Hulman Institute of Technology
Terre Haute, IN
boutell@rose-hulman.edu

ABSTRACT

Advancements in the field of mobile photography provide new contextual cues to enhance capture-time scene classification. In this paper we present a novel approach to improve an existing sunset photo classifier by using the photo's spatiotemporal cues. First, we classify the photo based on visual cues using a support vector machine. Second, we use spatiotemporal cues – geolocation, date, and time – to calculate the range of times when sunset photos are expected to be taken that day; the probability distribution is learnt from a large set of geotags obtained from Flickr. We then classify the photo using those spatiotemporal cues. Finally, we obtain a classification from the posterior probability using both visual and spatiotemporal cues. We present a new mobile camera app that classifies photos as sunset or non-sunset at capture time, and demonstrate the effectiveness of our application on a large dataset.

Categories and Subject Descriptors

I.4.9 [**Image Processing and Computer Vision**]: Applications.

Keywords

Geotag, GPS, Mobile, Sunset, Classification, Spatiotemporal.

1. INTRODUCTION

The field of mobile photography continues to grow at a rapid pace. The ubiquity of camera phones has made it easy for people to snap photos in any place and at any time. While photographs taken with camera phones are generally of lower quality than those taken with professional cameras, accurate information captured by the camera about the location of the photographer and the time the photo was taken create new opportunities for scene classification and annotation.

We consider semantic classification of natural photos, those in which the background, rather than a person or an event, dominates the classification. In earlier years, classification of such photos was done using pixels only [1]. Later, others used context such as camera settings (like focal length and flash) captured in the image metadata [2]. However, the usage of timestamps in that work was limited to the elapsed time between photos in collections, because photographers often forgot to set the times on their cameras, or forgot to accommodate for daylight saving time. Others have used geotags [3], but those tags, when entered by the user at upload time in photosharing sites like Flickr, are often not as accurate as those added at capture time.

Mobile phones can reduce user error by auto-synchronizing the time and time zone and by automatically geotagging photos at *capture time*, leading to highly accurate, strong spatiotemporal cues. In this paper, we explore a situation where accurate location and time has much potential to improve classification: sunset photos. Sunset photos are taken at a time of day that depends both on geolocation and date. While others have improved semantic classification of events and landmarks using location only (such as photos of the Eiffel tower) [4], or time elements only (such as calendar events for Christmas parties or fireworks on holidays) [5], sunset photos are of interest because of the interplay between both time and location.

Capture-time photo tagging has interesting applications. If some tags are already present when the photo is uploaded to a photosharing site, that site could simply ask the user to confirm them, reducing the effort of manual tagging. Sunset detection can also be used to help amateur photographers. While many cameras have a sunset scene mode designed to preserve the vivid colors of sunsets, some photographers forget to use it. A positive sunset detection could activate this mode automatically.

Given a geolocation and a date, we can calculate the exact time of the setting sun and derive from it a range of times when photographers can take a picture and still call it a sunset photo. On one hand, this spatiotemporal information is very strong: one can say definitively that a photo captured 4 hours after sunset is not a sunset photo, regardless of the image content. But on the other hand, it is not foolproof in isolation; just because a photo was taken in San Francisco at 8:00 pm on a March evening doesn't mean the photographer was even outdoors! Visual cues must be used as well.

We contribute the following. First, we present and analyze statistics from a set of geotagged Flickr photos showing how far from the time of sunset that people capture sunset photos, using both camera phones and arbitrary cameras. Second, we compare classification accuracy using spatiotemporal cues alone, visual cues alone, and the two cues fused using *maximum a posteriori* (MAP) probabilities. Third, we develop a new Android camera app that tags a photo as sunset on the fly if it believes the user is currently taking a photo of a sunset. The app's computations are done quickly enough to operate in real time, as it makes use of low-resolution image features and a simple classification scheme.

2. METHODOLOGY

We restrict implementation of our app to sunsets. While sunrise detection is similar, we restrict our experiments to sunsets for three reasons. First, and of highest importance, sunsets are photographed much more often than sunrises, so the training data is more plentiful. Second, sunset colors are more salient because they are warmer. Third, it simplifies the presentation.

GeoMM'14, November 7, 2014, Orlando, FL, USA.
Copyright 2014 ACM 978-1-4503-3127-2/14/11…$15.00.
http://dx.doi.org/10.1145/2661118.2661120

The key to our system is calculating the photo capture time relative to the time the sun is setting that day. Photos taken before sunset feature the sun in the sky, while those taken after sunset feature color-filled skies (Figure 1). Intuitively, good photos of sunsets should be taken within a short time of the time the sun actually sets[1], while those taken outside of a large enough window cannot be sunsets.

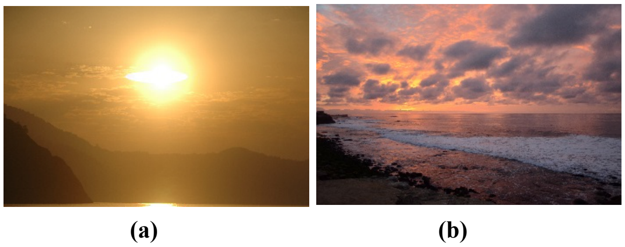

(a) (b)

Figure 1. Photo taken (a) before the sun has set and (b) after.

2.1 Calculating Time until Sunset for Flickr Photos

How we calculate the sunset time depends on whether the photo is obtained from a photosharing site (for training purposes) or whether it is being classified on a device (at capture time). We discuss capture-time classification later with the rest of the app. In the case of pictures acquired from Flickr, the *geolocation* and *date taken* were served to the Google TimeZone service [6] to acquire the time zone and daylight-saving adjustments. These were joined with the local *time taken* in order to compute the exact time taken in UTC. Then the geolocation and date were served to a sunset calculator [7] to compute the official sunset time of the day in UTC[2]. The time until or since sunset is acquired by taking the signed difference of the two numbers.

We obtained 137,648 sunset images taken by a large number of photographers from Flickr simply using a search of images tagged "sunset" [9] [10]. Of those, 37,804 were geotagged and had some timestamps. Although some of the images had private EXIF data, we obtained the GPS coordinates through Flickr's API. We examined these and found the following distribution of times as in Figure 2a. Most of the images were taken close to sunset time, but two peaks can be seen around the 1-hour mark before and after sunset. We believe this noise is produced as a result of the photographers not setting the daylight saving time on non-smartphone cameras. We then used a subset of 1,041 that were tagged as being from smartphones, specifically, any of these models[3]: iphone, nexus, galaxy, gt-i, gt-n, gt-s, sch-i, sgh-t, sgh-d, sph-d. Since smartphones sync time automatically and geotag the photos at capture time, these should be more accurate (Figure 2b).

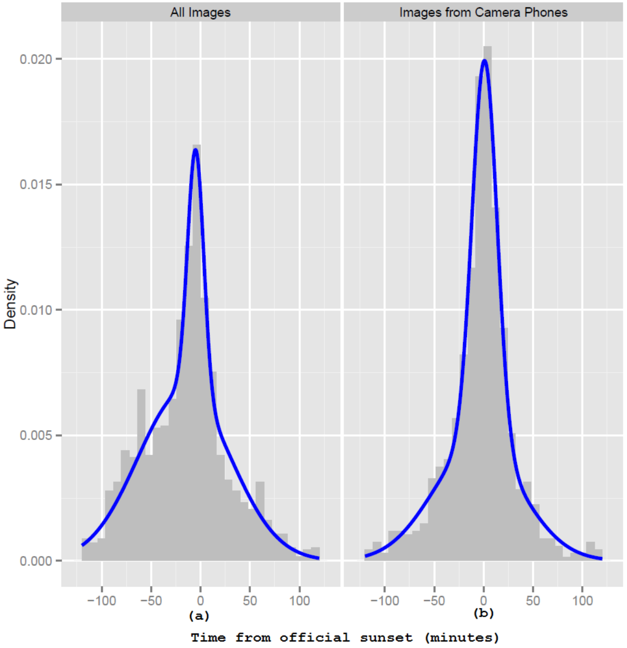


Figure 2. Distributions of times of sunset photos relative to official sunset times. Negative times are those taken before the sun sets. (a) General geotagged photos. (b) Photos taken using smartphones. The approximate Gaussian model is superimposed over the discrete distribution.

This data can be approximated by a mixture of Gaussians. While one could model it as a discrete distribution by normalizing the counts, we chose to fit it. Using the EM algorithm, we fit the distribution with two Gaussians and obtain the following.

$$f(x,\theta) = \frac{P}{\sigma_1}\varphi\left(\frac{x-\mu_1}{\sigma_1}\right) + \frac{1-P}{\sigma_2}\varphi\left(\frac{x-\mu_2}{\sigma_2}\right) \quad (1)$$

Given only the spatiotemporal cues, one can classify a photo as sunset or not by applying MAP estimation. We discuss this in Section 2.3.

2.2 Sunset Detection Using Visual Cues

Sunset detection has been studied previously as part of scene classification systems [1] [11]. Typical baseline systems use color features extracted from the image, for example, spatial color moments extracted from an n x n grid, as in Figure 3, and classified using learning vector quantization (LVQ) [1] or support vector machines (SVMs) [11].

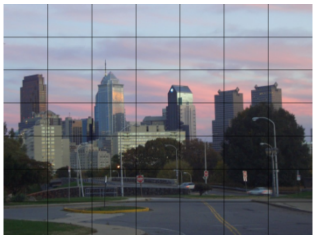


Figure 3. Spatial moments computed using n = 7.

We built a baseline classifier as done in [11], but used RGB space for simplicity since the non-linear transformation to LSV space is costlier and our experiments showed the choice of color space did not affect accuracy significantly. The parameter n can be varied also. Intuitively, a finer grid (larger n) should give higher

[1] This does vary by season especially at polar extremes.

[2] Various sunset times can be calculated: official (earliest), civil, nautical, and astronomical (latest) [8]. We use official, as it seems most consistent with our definition of sunset.

[3] This list was acquired from Flickr's cameras page [9].

accuracy if a large enough training set is used, while a courser grid yields few features and thus faster classification time.

SVMs output a single number, which is the signed distance from the decision boundary - the magnitude is a confidence in the classification, because those close to 0 are in the margin and include the support vectors, those hardest to classify. We classified an independent set of 2523 photos from the Flickr set and plotted the distribution of distances from the SVM margin as Figure 4 shows. Even after removing some obviously mis-tagged photos, this is still a challenging data set, with many weakly-colored sunsets, as evidenced by the number of data points close to the margin and the resulting amount of overlap in the distributions.

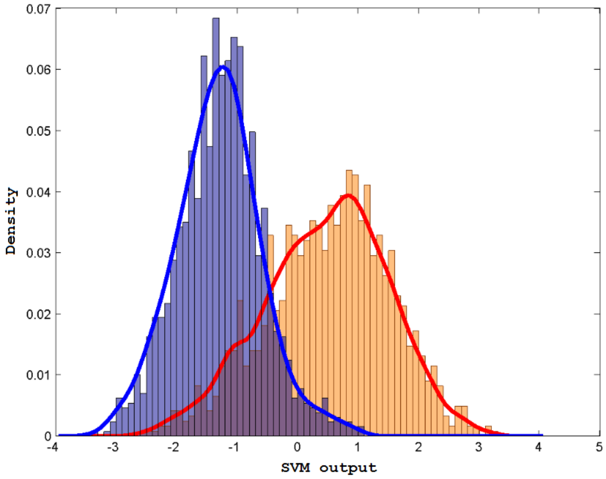


Figure 4. Distributions of SVM output for sunsets (orange) and non-sunsets (blue) on a large dataset from Flickr.

Binary classification decisions are made by thresholding the distance, typically at 0. In our dataset (Figure 4), the threshold that optimizes overall accuracy is -0.4, but other thresholds can be chosen to achieve any desirable true positive rate on sunsets. For fusion, one can model P(S) in the probability domain by converting to a belief in the range [0, 1] using a sigmoid function. Instead, we use a posterior function, as discussed next, to combine it with the probability from the spatiotemporal cues.

2.3 Evidence Fusion Using MAP Estimation

The posterior probability is the probability of an event S provided the evidence. Our event, S, is that a photo is of a sunset. We calculate the posterior probability using Bayes Rule,

$$P(S|E) = \frac{P(E|S)P(S)}{P(E)} \qquad (1)$$

where E is the available evidence of the class of the photo. Expanding $P(E)$ using the definition of joint probability yields

$$P(E) = P(E,S) + P(E,\bar{S}) \qquad (2)$$

where $\bar{S}$ is a nonsunset photo. Applying the definition of conditional probability to (2) and substituting back into (1) we get

$$P(S|E) = \frac{P(E|S)P(S)}{P(E|S)P(S) + P(E|\bar{S})P(\bar{S})} \qquad (3)$$

The priors can be estimated from another data set, say a random set of Flickr images. This ratio $P(S)/P(\bar{S})$ is expected to be very small (e.g., 1:49 if 2% of images were sunsets), making it extremely unlikely that any photo could be classified as a sunset. We thus ignore priors as done in [3]; after ignoring priors, we obtain:

$$P(S|E) = \frac{P(E|S)}{P(E|S) + P(E|\bar{S})} \qquad (4)$$

The evidence is a joint distribution of the visual (SVM) evidence, *v*, and the spatiotemporal evidence, *t*. Substituting yields

$$P(S|E) = \frac{P(t,v|S)}{P(t,v|S) + P(t,v|\bar{S})} \qquad (5)$$

While the visual evidence and the spatiotemporal evidence are not independent, we reasonably assume that they are *conditionally* independent, given the class of the photo. This yields:

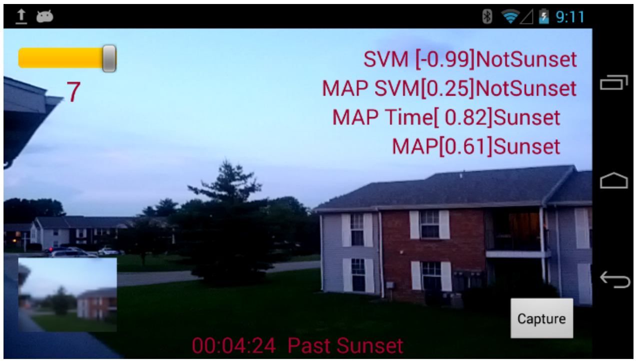
$$P(S|E) = \frac{P(t|S)P(v|S)}{P(t|S)P(v|S) + P(t|\bar{S})P(v|\bar{S})} \qquad (6)$$

The visual probabilities, $P(v|S)$ and $P(v|\bar{S})$, are acquired from the SVM distributions in Figure 4, since the SVM classifier uses pixel data. The spatiotemporal probability of sunset, $P(t|S)$, is acquired from the time distribution in Figure 2. For non-sunsets, $P(t|\bar{S})$, we assumed a uniform distribution, since we observed from a large random photoset that non-sunset photos are captured uniformly at all times of day.

2.4 Android Application

We designed a camera app for Android. Upon starting the app, it loads an SVM that had been trained offline and calculates the official sunset time for that day at that location. The required geolocation and current time are simply acquired through the mobile device's internal services.

At capture time, the app computes and displays the time until sunset on the preview. It also classifies the video feed as sunset or not using the visual cues, spatiotemporal cues, and the fused cues – see Figure 5.



SVM [-0.99]NotSunset
MAP SVM[0.25]NotSunset
MAP Time[0.82]Sunset
MAP[0.61]Sunset

00:04:24 Past Sunset

Figure 5. Screenshot of the Android app.

4. EXPERIMENTAL RESULTS

We test our system for two qualities. Classification accuracy using visual cues is computed offline on a Flickr data set. Performance (runtime) is measured on the mobile device itself.

4.1 Accuracy on Flickr Set

We prune our data sets to meet the following criteria. Sunset skies must have some warm color: red, orange, yellow, pink, or purple, but do not need to include the sun itself. Sunset images can have a variety of objects in the foreground. We exclude photos obviously taken with filters and those where the foregrounds have not begun to dim, as they seem to occur earlier in the afternoon. We require the skies in the photos in our training set to have significant color, but we allowed the colors in the test photos to be much weaker, which makes the set challenging to classify – Figure 6 shows that the baseline classifier obtains about 80% accuracy.

We trained an SVM on a set of 2,490 non-smartphone images, and tested on the 2523 smartphone images described earlier in the paper – each image has GPS and timestamps. We compare the joint effects of visual and spatiotemporal evidence with the effect of each in isolation (setting all the probabilities involving the other to 1) – see Figure 6. As expected, the spatiotemporal cues improve the classification accuracy consistently. Interestingly, we observe that, in isolation, time until sunset is a less salient cue than color. Sunset photos have a higher likelihood ($P(t|S) \geq 0.5$) from 27 minutes before sunset until 31 minutes after sunset. Unfortunately, many sunsets in the test set, while taken with smartphones, seem to be taken outside that range, causing them to be misclassified. Thus the spatiotemporal cues in isolation were weaker than anticipated.

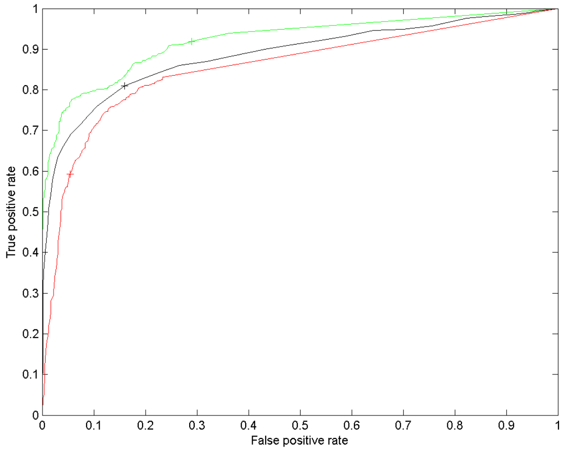


Figure 6. Combined ROC curve for 2523 smartphone images using visual cues only (black-baseline), spatiotemporal cues only (red-lowest), and the combined cues (green-highest). The AUC for each are 0.882, 0.855, and 0.918, respectively.

4.2 Performance of Mobile App

The results of the mobile application were matched against the results from the Flicker set to verify their accuracy. Additionally, we measure the responsiveness of the app, measured by frame rate of classification. We measured accuracy and performance for various values of *n*, the grid size. The most accurate results were found when *n* = 7, which matches our expectation. However, the app ran consistently at 4 frames per second for all values of *n* (on a Google Nexus 4). Since *n* affects the dimensionality of the SVM, we conclude that the SVM classification time is not the major factor in the runtime, and keep *n* = 7 in our app. While our code runs at less than the standard 30 fps, it is still fast enough to capture and tag photos smoothly. The app is available at https://github.com/RoseMobileVision/Sunset/ .

5. CONCLUSION

We have presented evidence that one can leverage spatiotemporal cues available on smartphones to improve the detection of sunset photos at capture time, using distributions learnt from a large geotagged data set. We also developed an app to classify photos in real time. One next step is to experiment with classification schemes like boosting, that have smaller memory footprints than SVMs. With boosting, one can also add the spatiotemporal features directly to the feature vector to compare early fusion with the late fusion technique we used. Another future direction is to investigate data from sensors like accelerometers, compasses, and gyroscopes. For sunsets, the camera should be pointing west – this should help classification if the data is available and reliable. Finally, a reasonable extension to this work to other natural photos is to exploit GIS information from an online service, for example to help classify beach scenes or open water scenes.

6. ACKNOWLEDGMENTS

We thank Eric Reyes and Diane Evans for assistance fitting the mixture of Gaussians.

7. REFERENCES

[1] A. Vailaya, M. Figueiredo, A. Jain, H. Zhang. Content-based hierarchical classification of vacation images. *Proceedings of International Conference on Multimedia Computing and Systems*, Florence, Italy, 1999.

[2] J. Luo, M. Boutell, and C. Brown. Pictures are not taken in a vacuum: An overview of exploiting context for semantic scene content understanding. *IEEE Signal Processing Magazine*, 23(2), pp. 101-114, March 2006.

[3] D. Joshi and J. Luo. Inferring Generic Activities and Events from Image Content and Bags of Geo-tags. *Proceedings of International Conference on Image and Video Retrieval*, Niagara Falls, Ontario, 2008.

[4] Y. Li, D. Crandall, D. Huttenlocher. Landmark classification in large-scale image collections. *Proc. ICCV*, Kyoto, Japan, 2009.

[5] Gallagher, A., Neustaedter, C., Cao, L., Luo, J., and Chen, T. Image Annotation using Personal Calendars as Context. *Proceedings of ACM Multimedia*, ACM Press, 2008.

[6] Google Time Zone API.

https://developers.google.com/maps/documentation/timezone

[7] M. Reedell (2010) Sunrisesunsetlib-java (Version 1.0). https://github.com/mikereedell/sunrisesunsetlib-java/.

[8] Astronomical Applications Department of the U.S. Naval Observatory. (2011, Oct 7). Rise, Set, and Twilight Definitions. http://aa.usno.navy.mil/faq/docs/RST_defs.php

[9] Flickr APIs. https://www.flickr.com/services/api/

[10] OpenIMAJ FlickrCrawler. http://sourceforge.net/p/openimaj/wiki/The%20FlickrCrawler%20Tools/

[11] M. Boutell, J. Luo, and R.T. Gray: Sunset scene classification using simulated image recomposition. *Proc IEEE Int'l Conf Multimedia & Expo*, Baltimore, MD, 2003