Shadow Detection and Sun Direction in Photo Collections

Scott Wehrwein Kavita Bala Noah Snavely
Cornell University
{swehrwein, kb, snavely}@cs.cornell.edu

Abstract

Modeling the appearance of outdoor scenes from photo collections is challenging because of appearance variation, especially due to illumination. In this paper we present a simple and robust algorithm for estimating illumination properties—shadows and sun direction—from photo collections. These properties are key to a variety of scene modeling applications, including outdoor intrinsic images, realistic 3D scene rendering, and temporally varying (4D) reconstruction. Our shadow detection method uses illumination ratios *to analyze lighting independent of camera effects, and determines shadow labels for each 3D point in a reconstruction. These shadow labels can then be used to detect shadow boundaries and estimate sun direction, as well as to compute dense shadow labels in pixel space. We demonstrate our method on large Internet photo collections of scenes, and show that it outperforms prior multi-image shadow detection and sun direction estimation methods.*

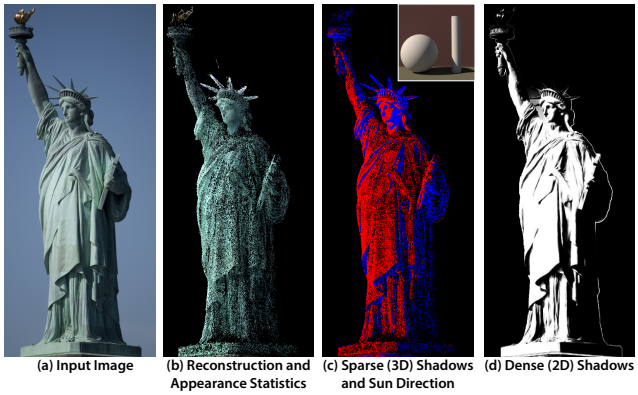

Figure 1. Our technique begins with photos from and Internet photo collection (a). After performing 3D reconstruction, we use appearance statistics (average intensity, (b)) to produce a shadow label for each reconstructed 3D point (c). We then use these shadow labels to estimate the direction of the sun (inset), and dense shadows in pixel space using the matting Laplacian (d).

1. Introduction

A long-standing goal in computer vision is photo-realistic modeling of scenes from images. While we have made great progress in *geometric* reconstruction, through structure from motion (SfM), bundle adjustment, and multi-view stereo, geometry is just part of the story. To get truly detailed, realistic, relightable scene reconstructions, we also need to solve for appearance—e.g. texture maps and materials—which means that we also must reason about surface reflectance, camera response, and, critically, *illumination*.

Modeling all of these aspects of scene appearance together is challenging, especially in unstructured photo collections. Many existing methods involve large-scale nonlinear optimization jointly over many parameters, including illumination and albedo (*e.g.*, [12, 23]). These methods can sometimes work well, but are subject to local minima without a good initial guess. In this paper we focus on simpler methods for estimating illumination properties, in particular modeling shadows and sun direction in Internet photo collections. We propose a straightforward and optimization-free technique, yielding illumination information that is useful in its own right but could also easily serve as a good initial guess for illumination in a larger reconstruction pipeline.

Relative to a comprehensive appearance model, shadows and sun direction are simple properties, yet they are powerful in their ability to explain outdoor scene appearance. Shadow maps can be used for more realistic texture mapping of highly detailed 3D models [23]. Shadows and sun direction also play an important role in outdoor photometric stereo, which has been shown to capture fine geometric details better than multiview stereo [24]. Shadows and sun direction together can be used as a geometric cue for 3D reconstruction as in [3].

Sun position is also useful for correcting timestamps in Internet images, where time zone differences and incorrectly set camera clocks make the image metadata unreliable. Recent work has used raw timestamps to coarse and noisy image timestamps to reconstruct photo collections in 4D by adding a temporal dimension [18, 17], but these methods are limited by the quality of these timestamps. Our method can be used to determine accurate timestamps from estimated

sun directions, potentially enabling much finer-grained 4D reconstructions of much more lively events at the scale of hours, rather than months or years.

Figure 1 illustrates our method, which works by leveraging appearance statistics within a photo and across a collection, along with the physics of natural illumination, to detect shadows on a sparse 3D reconstruction of the scene. To achieve robustness to camera effects, we analyze illumination ratios of pairs of points, which are used in a voting sceme to classify which 3D scene points are in shadow and which in sun for a given image. Shadow boundaries are then detected and used to estimate sun direction. Finally, we can also combine image matting techniques [16] with the sparse reconstruction to obtain dense shadow labels in pixel space. We evaluate our method on large-scale photo collections of outdoor scenes, and show that it outperforms prior work on shadow detection and sun position estimation. We demonstrate that our method also works on aligned image stacks (e.g., timelapse data).

2. Related Work

There has been a rich body of research on shadow detection, both as a standalone task on a single image and within the context of other problems such as photometric stereo, 3D reconstruction, and intrinsic images. Because we are focused on applications involving photo collections, where many images are available but speed and robustness are key, we primarily review multi-image techniques designed for photo collections and webcams.

Shadows in image stacks. Recent work has focused on shadow detection and illumination decomposition in registered image stacks, such as outdoor webcams. As part of a video factoring method, Sunkavalli *et al*. detect shadows based on simple pixel statistics and thresholding [25]. Ackermann *et al*. [4] and Abrams *et al*. [2] apply photometric stereo to webcam data, and use variants of Sunkavalli's thresholding method (*e.g*., Abrams *et al*. propose a time-varying intensity threshold). Like our method, [14] compares pixel values to their mean for shadow estimation. These methods are well-suited for image stacks with limited changes in camera exposure, or when the images are in a known temporal sequence. However, they are less effective in Internet photo collections where cameras vary.

Abrams *et al*. [3] use correspondences between cast shadows and occluding geometry in outdoor scenes to estimate depth information; this work makes use of an in-house shadow detector based on expectation-maximization [1]. While this method outperforms other techniques, it requires careful geographic camera calibration and accurate per-image timestamps.

Single-image shadow detection. One closely related single-image method is that of Guo *et al*. [7], which finds pairs of image regions that are predicted to be the same material, and combines these with learned per-region shadow classifiers in a CRF. Other recent techniques have also used learning to predict shadows using a combination of local and pairwise features [26, 15]. In our work, we use pairwise comparisons among pixels, but use many images from a photo collection to compute statistics that can be used to discount the effect of materials and cameras.

Image collections. A growing body of work estimates illumination in unstructured Internet image collections. One related method is that of Laffont *et al*. [12] for estimating intrinsic images from photo collections. Like our method, Laffont's is based on analyzing pairs of points, but they have to find pairs of points with similar surface normal and illumination; from this, they build a graph on points that requires adequate connectedness and good spatial and normal distribution. We focus on the more constrained problem of shadow detection but devise a simpler approach.

Other work incorporates shadow detection into larger illumination estimation methods. Shan *et al*. estimate lighting and reflectance from photo collections, including an explicit model of shadows [23]. However, they do not evaluate performance on these tasks, as their goal is to obtain perceptually accurate rendered images. In addition, their method involves a complex optimization procedure, as do related methods (e.g. *et al*. [8]). Our method is simple and easily implemented, and can improve such estimation methods.

Sun position from illumination. In addition to detecting shadows, we use attached shadows to estimate sun direction (and timestamps). Prior work has also sought to estimate illumination properties from either shadows cast on a planar surface [22, 20, 13, 21], or from shading information [19]. In a closely related method, Hauagge *et al*. [10] use an intrinsic image technique that considers ambient occlusion [9] to determine shading and illumination outdoors. They then compare the illumination estimate against renderings with a physically-based sun/sky model to determine sun direction. Our method is simpler, reasoning about occlusion in the form of shadows, and relying only on attached shadow boundaries for sun direction. We achieve comparable performance without full illumination estimates or sun/sky models.

3. Illumination Ratios

In this section, we analyze illumination ratios and show how they can be used to detect shadows in photo collections. We begin with a set of images and a set of sparsely reconstructed 3D points (obtained using SFM and MVS), where each point is visible in a subset of the images. For each 3D point x, we sample the grayscale intensity value from all images that observe x. We also assume that the MVS reconstruction includes an estimated surface normal; we do not require normals for shadow detection, but will use them

for sun direction estimation. For now, our goal is to label each 3D point as sunlit or in shadow.

A simplistic but intuitive heuristic for guessing whether a point is sunlit or not is to answer the question "is the point brighter than usual?" Most points in an outdoor scene are sometimes shaded and sometimes sunlit. Because we have a collection of observations of the same point, we could answer the question by comparing an observation of a point x in image i to its average value over all observations:

$$R(x) = \frac{I_i(x)}{\mathcal{E}[I(x)]} \tag{1}$$

However, a given observation depends on camera effects as well as illumination. A higher-than-average intensity could indicate a sunlit point, but it could just as easily arise due to an overexposed image. The above ratio fails to isolate illumination effects from camera effects.

We can eliminate camera exposure by comparing this ratio for two points x and y in image i:

$$R(x,y) = \frac{\frac{I_i(x)}{\mathcal{E}[I(x)]}}{\frac{I_i(y)}{\mathcal{E}[I(y)]}} \tag{2}$$

In this ratio of ratios, exposure and any other linear camera effects cancel, and relative illumination of x and y is isolated. This ratio is large if x is unusually bright relative to y in image i—namely, if x is sunlit and y is in shadow.

A natural way to use the illumination ratio to estimate shadow labels is to threshold it. In particular, using the intuition above we can use a threshold T to classify pairs as follows:

A. If $R(x,y) > T$ then x is sunlit and y is in shadow.

B. If $R(x,y) < 1/T$, then x is in shadow and y is sunlit.

C. If $1/T \leq R(x,y) \leq T$, then x and y are similarly lit.

Note that because the ratio deals with relative illumination, Case **C** is ambiguous—the points may be either both sunlit or both shaded. The first two cases are symmetric, so WLOG, we focus on analyzing the first.

3.1. Image Formation Model

To determine how to set the threshold T, we consider the ratio's behavior under a simple image formation model, where an observation of x in image i is the product of its diffuse albedo (ρ_x), the camera exposure (E_i), and the illumination incident on x in image i ($L_{i,x}$):

$$I_i(x) = \rho_x E_i L_{i,x} \tag{3}$$

The lighting L_i is further decomposed into a direct (sun) term L_d and an ambient (sky) term L_a. L_d is modulated by the cosine of the angle $\phi_{x,i}$ between the surface normal and the sun direction, and a binary sunlight indicator $C_{x,i}$, which is zero if point x is in shadow in image i.

$$I_i(x) = \rho_x E_i [C_{x,i} L_d \cos(\phi_{x,i}) + L_a] \tag{4}$$

Notice that the shadow detection problem amounts to finding $C_{x,i}$ for all points x observed in image i.

Although directly plugging this image formation model into the illumination ratio (2) does not yield immediate insights, we can simplify a great deal by approximating a point's albedo by the point's average observed intensity. Other albedo estimation methods could also be used, but we found that this approximation works well enough in practice. Our robust voting scheme (discussed in the next section) helps to eliminate most of the errors introduced by this approximation.

At this point all quantities relate to a single image i, so we drop the subscripts. Assuming $\mathcal{E}[I(x)] \approx \rho_x$, we can plug our image formation model (4) into (2) and the albedos cancel. We can also simplify further by substituting $f = L_a/L_d$, i.e., the ratio of intensities of ambient and direct illumination, to obtain:

$$R(x,y) = \frac{\frac{\rho_x E[C_x L_d \cos\phi_x + L_a]}{\rho_x}}{\frac{\rho_y E[C_y L_d \cos\phi_y + L_a]}{\rho_y}} = \frac{C_x \cos(\phi_x) + f}{C_y \cos(\phi_y) + f} \tag{5}$$

3.2. Choosing a Threshold

We see that (5) boils down to a ratio of the cosine shading terms of the two points, with f added to both the top and bottom. Because f varies throughout the day as the sun and sky vary in relative brightness, we cannot solve for a perfect threshold T that always classifies pairs correctly. Points with ϕ_x near 90° (that is, they are only obliquely lit by the sun) also pose a challenge since their appearance can be very close to that of shaded points. Our approach is to choose a threshold T that classifies pairs correctly under most common circumstances, and then later use a voting scheme that considers many pairs of points, overruling incorrect pairwise classifications.

We can draw insight about f from the CIE Sky Standard [11]. For a clear, sunny sky (CIE Model #12), Figure 2 shows that at most times of day, f ranges from 0.2 (when the sun is directly overhead) to 0.5 (when the sun is about 20 degrees above the horizon) [5]. As the sun sets (L_d goes to zero), f goes to infinity.

Using the threshold as described above amounts to a classification: the illumination for a pair of points is classified either as different (Cases **A** and **B**) or the same (Case **C**). In our discussion we refer to **A** and **B** as a "positive" classification—that is, x and y should have opposite labels. Case **C** is a "negative" classification, where the points are said to have same label. As usual, our choice of threshold involves a trade-off between false positives and false negatives.

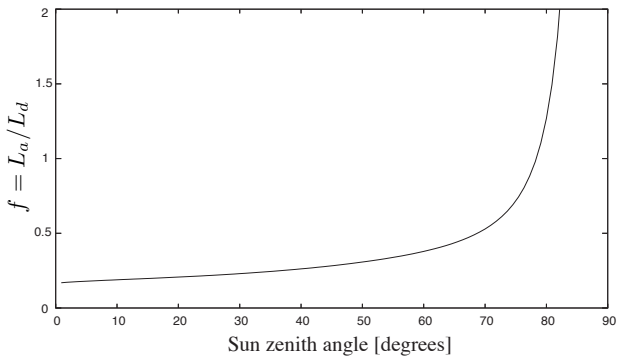

Figure 2. The relative illuminance of the sky and the sun as a function of sun zenith, according to CIE clear sky model #12. For most sun angles, f falls between 0.2 and 0.5. Figure adapted from [5] with permission.

Figure 3 shows that a higher threshold limits false positives, whereas a lower threshold minimizes false negatives. In Figure 3(a) we consider two sunlit points x, y. Each plot shows for which ϕ_x, ϕ_y the points x, y get correctly classified as the same (negative) by a given threshold T for different values of f; for larger T, the false positive region shrinks. Figure 3(b) assumes x is sunlit and y is shaded, and plots the value of $R(x, y)$, with three example thresholds drawn. Here, a lower T is better—the portion of each curve above the threshold corresponds to true positives, while the portion below corresponds to false negatives.

Empirically, we found that a threshold of about 3 worked best, although the results were not very sensitive to slightly different values due to the voting process. Although Figure 3(b) suggests that $T = 3$ produces exclusively false negatives for values of $f \geq 0.5$, we found that in practice, 3 works well, even for larger f. This is likely due to real-world deviations from the image formation model and the fact that only false positives cast incorrect votes, while false negatives are simply ignored in our voting scheme.

4. Estimating Shadow Labels

We have seen that the illumination ratio is a useful, but somewhat noisy indicator of pairwise relative illumination. In this section, we propose a simple algorithm that considers the illumination ratio among many pairs of scene points in a voting scheme to robustly estimate a shadow label for each point.

For each point x visible in an image, we pick a set of K (we use $K = 1000$ in our experiments) other points visible in the same image and cast votes according to their thresholded illumination ratio, as described in the previous section. In particular, for each of the K points y_k, a vote for x being sunny is cast if $R(x, y_k) > T$, and shadowed if $R(x, y_k) < 1/T$. Otherwise, the vote is considered uninformative and thrown out. We aggregate all informative votes into a shadow score for each point, as illustrated in Figure 4, which shows the votes cast by all other image points when compared to the three labeled example points. By selecting a sufficiently large set of random points, we are highly likely to get points with differing illumination, which should cast informative votes. Outliers due to noise or unusual surface normal configurations are outvoted by the majority of well-behaved pairs.

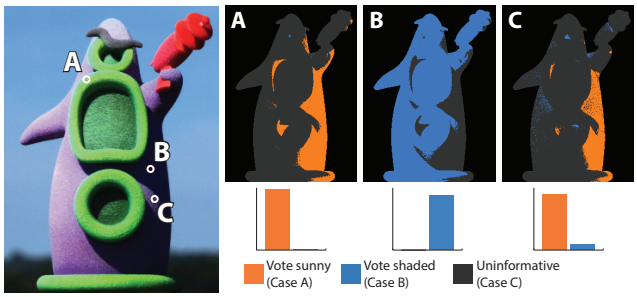

Figure 4. An illustration of thresholded illumination ratios comparing three points to all other points in the image. Image A shows the votes cast by the thresholded value of $R(A, y)$ for all other points y in the image, and likewise for B and C. Point A is sunlit, so most of the informative scores come from shaded points (orange). A few outliers provide shady scores (blue). The opposite is true for point B. Point C's normal is relatively far from the light direction, so the scores are not unanimous but still conclude that it is sunlit.

The votes are aggregated into a shadow score:

$$S(x) = \frac{\ell_x}{\ell_x + s_x} \tag{6}$$

where ℓ_x is the number of sunlit votes and s_x is the number of shadowed votes. This score is 1 if all pairs vote sunlit, 0 if all pairs vote shaded, and 0.5 if an equal number of votes went each way. As such, the score's distance from 0.5 gives a measure of how confidently we can label the point. For sun direction estimation, we are willing to leave low-confidence points unlabeled, so we choose a minimum confidence c_0. The final label of x is sunlit if $S(x) > 0.5 + c_0$ and shadowed if $S(x) < 0.5 - c_0$. If neither holds, x remains unlabeled. An example shadow labeling is shown in Figure 1(c).

4.1. Dense Shadow Maps

We have proposed a technique for determining shadow labels for points in a sparse 3D reconstruction. We now show that we can use an image matting technique to simultaneously propagate the sparse shadow labels to all image pixels and produce continuous labels indicating the extent of the cosine shading. This is useful for applications of shadow detection (*e.g.*, shadow removal, texture mapping) where a sparse binary shadow map is not sufficient.

We draw inspiration from the recent work of Guo *et al.* [7] and Laffont *et al.* [12]. Guo *et al.* estimate a dense binary

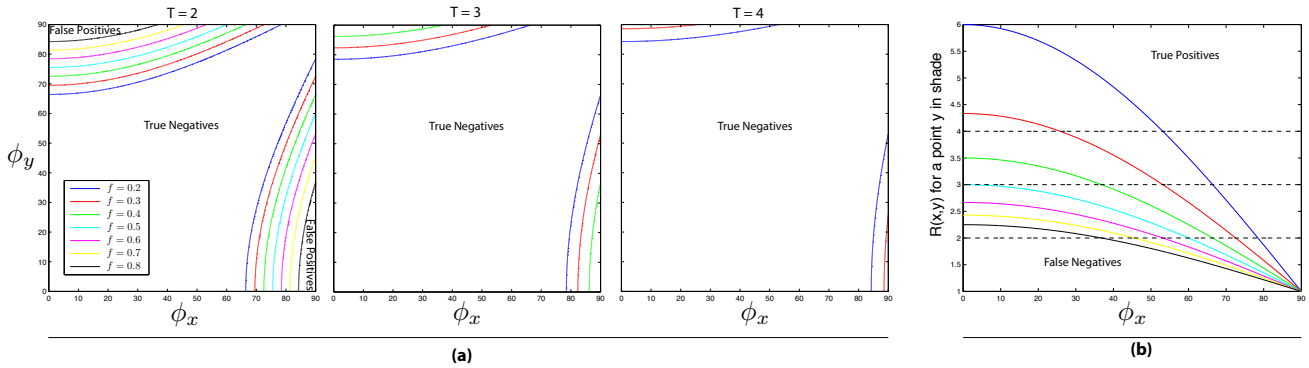

Figure 3. An illustration of the trade-off in threshold choice. **(a)** For $T = 2, 3, 4$, we consider two sunny points with ϕ_x and ϕ_y, showing which pairs of angles result in correct classifications. True negatives, in the large central area, occur when the two sunny points are classified as having the same illumination. The contour lines delineate the regions where false positives occur for different values of f. **(b)** For sunlit x and shadowed y, we plot the ratio score as a function of ϕ_x for different values of f. Thresholds of 2, 3, and 4 are shown with black dashed lines; the threshold separates each curve into true positives (above the threshold) and false negatives (below).

shadow map and then use the matting Laplacian [16] to produce continuous labels near shadow region boundaries. In the context of intrinsic images, Laffont *et al*. use a similar technique to go from a sparse set of reflectance constraints to a dense reflectance image.

We found that the matting Laplacian performs well on shadow maps even when simultaneously propagating sparse to dense labels and estimating continuous labels. As in [7], the dense shadow map α is obtained by solving

$$(\mathcal{L} + \lambda D)\alpha = \lambda d\hat{\alpha} \tag{7}$$

where D is a diagonal matrix with 1 in the positions corresponding to the known sparse shadow labels and 0 elsewhere, d is the vector of elements from D's diagonal, $\hat{\alpha}$ contains the known shadow label values (1 for sunlit, 0 for shadow), and $\mathcal{L}$ is the matting Laplacian matrix constructed as in [16].

To pose our problem in this framework, we obtain a sparse 2D labeling by projecting the 3D points' shadow labels into the image using the camera pose from the structure-from-motion reconstruction. Because the matting solver fills in unknown values, we set $c_0 = 0.25$ so that only points with high-confidence labels are considered known values. Because some parts of an input image (*e.g.*, the sky) may not be part of the reconstructed scene, we use a standard image segmentation technique [6] and assign a fixed label of 0 to regions of the image where very few or no scene points project. Figure 7(bottom) shows several dense shadow maps computed using this approach.

5. Sun Position Estimation

In this section we show how to use the sparse shadow labels from the previous section to estimate the sun direction in an image. Given that our MVS reconstruction provides us with a surface normal estimate for each point, our key idea is to use a well-known property of directional lighting: a surface normal lying on an attached shadow boundary must be orthogonal to the light direction. Or in other words, attached shadow boundaries occur where the light direction is tangential to the surface. In principle, if we can identify two distinct normals with this property, their cross product gives the sun direction up to a $180°$ ambiguity. However, there are two challenges involved in using this property to find the sun direction: we must (1) identify shadow boundaries in a sparse set of labeled points and (2) discard cast shadow boundaries, which do not have the above property.

5.1. Sun Direction From Shadows

It is important to distinguish between two types of shadows. A point lies in a *cast shadow* if some object lies between the point and the sun. If the point's surface normal faces away from the sun direction and therefore receives no sunlight, this point lies in an *attached shadow*.

Although cast shadows can provide information about sun direction (as with a sundial), this generally requires precise geometry of the casting object. Because our 3D model is sparse and may not include all geometry (*e.g.*, a building across the street from a landmark of interest), such correspondence is difficult to find. Instead, we focus on attached shadow boundaries, whose surface normals provide weak indicators that can be aggregated into a final sun direction estimate. Our overall approach is to detect shadow boundaries of all types, then find a robust consensus with the assumption that some shadow boundaries will be outliers.

5.2. Detecting Shadow Boundaries

To detect 3D points that lie on shadow boundaries, we consider its neighboring points. Intuitively, if a point is on a

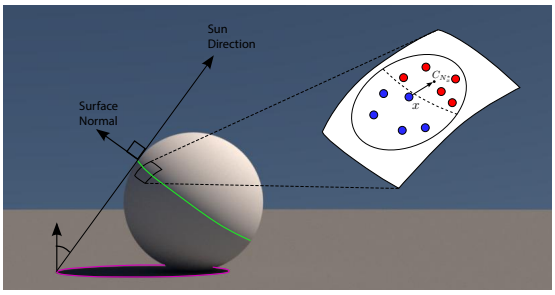

Figure 5. Normals on an attached shadow boundary (green) are orthogonal to the light direction, whereas normals on a cast shadow boundary (purple) are unconstrained. The inset shows the boundary vector pointing from a shadowed point x to the centroid of its sunlit neighbors $C_{N_x^*}$.

shadow boundary, it should have many neighbors with the opposite shadow label. These neighbors should:

1. lie on the same smooth surface and thus have similar normals
2. lie in the same general direction from the point

To capture this intuition, we define a boundary score as a function of the positions, normals, and shadow labels of the points in the neighborhood of x.

Let N_x be the neighborhood of x, defined as the set of points within a 3D radius r of x. We define the boundary neighbor set $N_x^* \subseteq N_x$ to be the neighbors that lie across a shadow boundary on the same surface; in particular, these are the points with shadow labels opposite from x's and normals forming an angle less than Θ_{crease} with x's normal.

Let the *boundary vector* B_i of x be the displacement vector from x to the centroid $C_{N_x^*}$ of the points in N_x^* scaled by the fraction of neighbors in the boundary neighbor set:

$$B_i(x) = c_{sign} \frac{|N_x^*|}{|N_x|} (C_{N_x^*} - x) \tag{8}$$

We use an indicator c_{sign}, set to -1 if x is sunlit, or 1 if it is in shadow so that the vector always points towards the sunny side of the boundary. $B_i(x)$ captures the intuition described above—if a point is surrounded by opposite-labeled points on all sides the distance to the centroid will be small, and if only a few neighbors have opposite labels the scale factor will be small. If neighbors have very different normals, they likely lie across a crease in the geometry, they are discounted. The direction of the boundary vector indicates which side of the shadow boundary the sun should be on, which we use later. Figure 5 illustrates the boundary term on an attached shadow edge.

We choose r manually for each dataset based on the scale of the scene and the density of the point cloud. We tag x as a boundary point in image i if $||B_i(x)||_2$ is greater than a threshold B_0. We set $B_0 = 0.2$ and $\Theta_{crease} = 45°$ for all results in this paper.

5.3. Estimating Sun Direction

We now have a set of shadow boundary points, some of which have surface normals orthogonal to the sun direction (inliers) and some of which do not satisfy this constraint (outliers). We use RANSAC to find the inliers and estimate the sun direction. We use the cross product of the normals of two random boundary points as the hypothesis, and count each boundary point as an inlier if its normal is within $10°$ of orthogonal to the sun direction and its boundary vector (which points towards the sunny side of the boundary) has a positive dot product with the hypothesis sun direction.

RANSAC assumes that the inlier (attached shadow) normals lie in a single plane, while the normals at cast shadow boundaries do not. This assumption may not hold in the case of a large cast shadow on a flat surface; to handle this case, we quantize the normals and count inliers only once per normal bin. This does no harm to the inlier estimate (coincident normals add no information) but prevents heavily-weighted collections of outliers from forming an incorrect consensus.

Constrained sun path algorithm. We have described a robust method for estimating sun direction given a set of shadow boundary normals, but if the scene is geo-registered we can improve our method by considering only astronomically possible sun directions. We can further constrain the possible sun paths to those occurring on the date that the image was taken. In this case, the set of sun directions is constrained to a single path through the sky. Although the EXIF timestamp is often incorrect due to time zone differences and improperly set camera clocks, the date field is often more reliable. Furthermore, small date errors produce only slightly different sun paths, so such inaccuracies can be tolerated.

Given a 1D sun path constraint, a brute-force approach is feasible: rather than randomly sampling pairs of boundary normals, we simply check the inlier count for each of a set of sun directions sampled along the sun path. Figure 6 summarizes our sun direction estimation process, showing an example result on the TENTACLE dataset.

6. Results

In this section we show both qualitative and quantitative evaluations of our methods against ground truth where possible, as well as against other methods. We begin by describing the datasets used, then show results for shadow maps, sun direction, and comparisons to existing methods.

Datasets. We evaluate our method on four datasets: two photo collections and two image stacks. STATUE is a photo collection reconstructed from about 80,000 photos. CASTLE is a scene reconstructed from about 33,000 photos. For

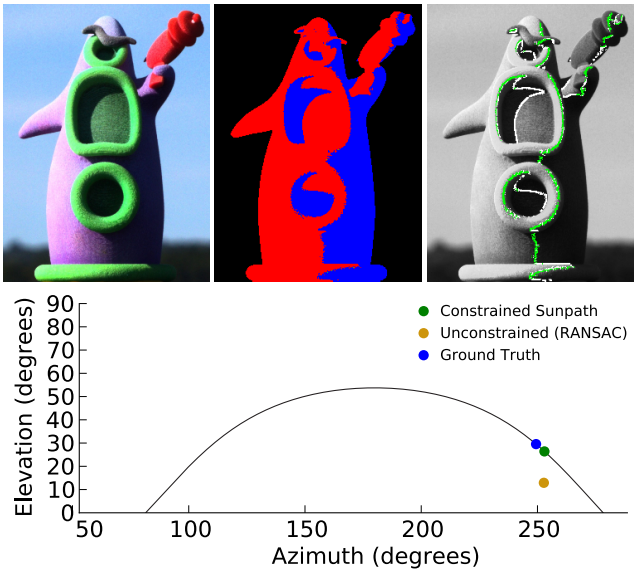

Figure 6. An illustration of our sun direction method. Given the shadow labels (middle), we compute a boundary score for each point (right, overlay pixels). The white points show all boundary points, while the green points are the inlier boundary points for the estimated sun direction. The algorithm correctly finds inlier points that primarily correspond to attached shadow boundaries.

efficiency, we computed the average intensity of points in these datasets over a subset of 10,000 images. For evaluation, we chose 115 images for STATUE and 200 for CASTLE that were taken under sunny or strongly directional illumination conditions at different times of day. Our other two datasets are sequences of images taken by a single static camera over the course of a day. TEMPLE is a 571-image webcam dataset made available by Sunkavalli *et al*. [25], which we use for comparison with their method.

Finally, the TENTACLE dataset allows quantitative evaluation of our methods. We used a 3D printed model of a figure and photographed it under outdoor illumination at 5-minute intervals throughout a sunny day. This allowed us to create raw linear images with proper exposure at each interval. We manually geo-registered the scene, then computed the precise sun direction based on the image timestamps. We also generated ground truth shadow maps by rendering the model under directional illumination corresponding to the sun direction. TENTACLE is useful because it is a non-synthetic dataset with many properties usually only available in synthetic datasets, including dense 3D geometry, surface normals, shadow maps, and sun directions.

Shadow detection in photo collections. Figure 7 shows shadow detection results on STATUE and CASTLE. From top to bottom, we show the input image, sparse shadows, and dense shadows. Notice that even fine details, such as the tight folds of the statue's cloak and the ridges at the top of

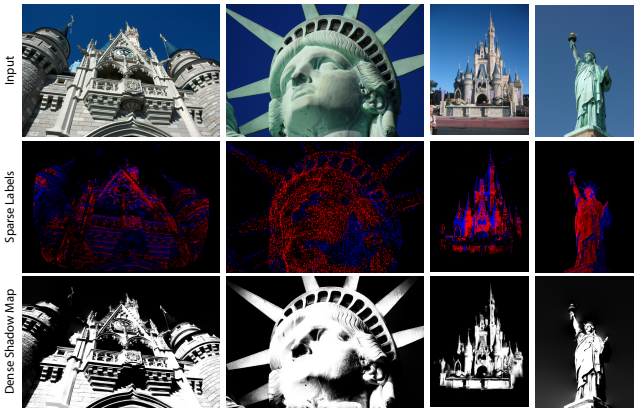

Figure 7. Input images (top), sparse shadow maps (middle), and corresponding dense shadow maps generated using the matting Laplacian (bottom). In the sparse shadow maps, red points indicate sunlit labels and blue points indicates shadow labels; in the dense map, white indicates a point in full sun and black indicates a fully shadowed point.

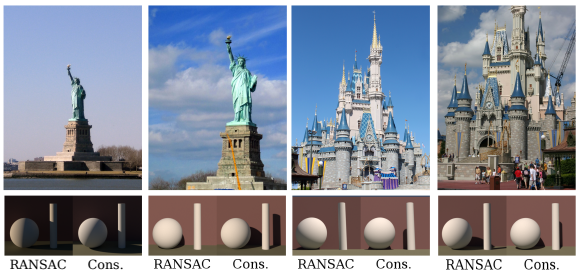

Figure 8. Qualitative sun direction results on CASTLE and STATUE. The top row shows an input image, and the bottom row shows a synthetic scene lit under the estimated sun direction using our two proposed sun position estimation approaches.

its base are correctly labeled in the sparse shadow map. The CASTLE dataset has more complicated geometry, but our method correctly identifies detailed shadow regions such as the ridges on the castle's facade. Using these accurate sparse shadow labels, the matting Laplacian is able to propagate shadow labels across the entire image, meanwhile choosing a continuous α value that indicates the extent of the cosine shading at each pixel.

Sun direction. It is difficult to obtain ground truth sun directions in large photo collections. Although sun direction can be inferred from geographic pose and accurate timestamps, the timestamps accompanying Internet photo collections are unreliable and cannot be used as ground truth. Instead, we show qualitative sun direction results on images from STATUE and CASTLE, and provide quantitative results on TENTACLE where ground truth is available.

Figure 8 shows qualitative results for sun direction estima-

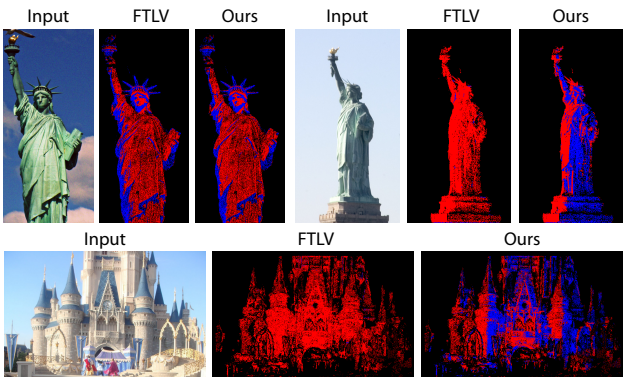

Figure 9. A comparison between our shadow detection method and FTLV's. Their performance is comparable on well-exposed images, but our method performs better on improperly exposed images.

tion from photos in STATUE and CASTLE. For each image, we run the RANSAC and constrained sun path algorithms. To illustrate the estimated sun direction, we show a simple rendered scene from a camera with the same viewpoint lit using a sun-sky illumination model. For images with plentiful and pronounced attached shadows, both approaches produce plausible sun direction estimates; in more challenging cases, the constrained approach produces a better estimate.

On the TENTACLE dataset, we are able to estimate the sun direction with a median error of 17.5° using our RANSAC-based approach. Given a rough geolocation, correct orientation, and an accurate date, our constrained sun path approach is able to obtain a median error of 10.9°, with very few errors over 25°. This is competitive with the results of the more complicated approach of [10] on the same dataset.

Comparisons. Most existing multi-image shadow estimation techniques assume a static camera and known time-ordering of images or precise timestamps [2, 3], and therefore cannot be applied to uncalibrated photo collections. One notable exception is Sunkavalli *et al*. [25], which decomposes time-lapse videos into intrinsic images. Their shadow detection method assumes no temporal ordering, using statistics over all observations to set a threshold which determines the shadow label of a point. This method works well on webcam sequences taken by a single camera, but fails on datasets with widely varying exposure and other camera inconsistencies.

Figure 9 compares our method with FTLV [25] on CASTLE and STATUE. For a fair comparison, we computed average intensity (used in our method) and the threshold (used in their method) using only the evaluation set. Their method performs nearly as well as ours on well-exposed images, but fails on images with slightly incorrect exposure because they do not account for camera variation.

We compared our shadow detection to FTLV on the TENTACLE and TEMPLE datasets. Figure 10 shows a qualitative comparison on TEMPLE; a similar comparison for TENTACLE is shown in the supplemental material. Table 1 shows a quantitative comparison with FTLV and the single-image method of Guo *et al*. [7] on TENTACLE. Guo's results are comparable, while FTLV performs slightly better than our method on the well-exposed TENTACLE set. However, our photo collection results show that our method is more robust overall in our target application.

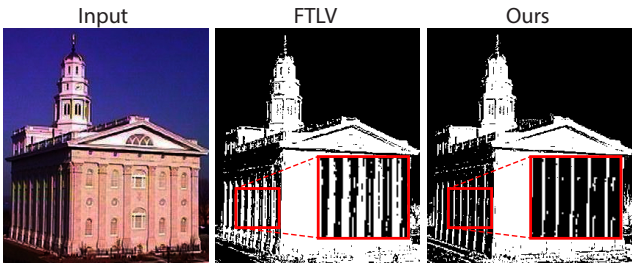

Figure 10. Example comparison of our results with FTLV on TEMPLE. Red insets show detail of a region where the performance differs.

	Ours	FTLV	Guo *et al*.
All points	87.16	91.65	86.73
$\cos\phi_x < 0.25$	94.15	95.14	89.48

Table 1. Percent of points classified correctly on TENTACLE.

The main difference in performance on TENTACLE arises from the inclusion of almost-shadowed points. Our method performs poorly on points that are only obliquely lit by the sun ($\cos\phi_x$ is small) because of the thresholding. When contribution of the sunlight is small, the point looks more like a shadowed point than a sunny point. If we discount points with $\cos\phi_x < 0.25$ our method performs comparably to FTLV, while Guo improves much less. TENTACLE also has some highly occluded points that are never sunlit during the day; this is a failure case for both our method and FTLV. TEMPLE lacks ground truth, but we observe qualitatively that our method provides cleaner shadow masks; we suspect this is due to TEMPLE's greater variation in exposure as compared to TENTACLE.

Limitations. Our analysis makes a number of assumptions. In particular, we have assumed that the scene is Lambertian, the weather conditions are clear, and the average observed intensity is a good approximation of the albedo. While our results on Internet data suggest that our method is robust to some deviations from these assumptions, our method breaks down in the presence of large specularities (*e.g*., the roof of CASTLE), heavy occlusion (*e.g*., the shadowed mouth area of TENTACLE), and heavy cloud cover. Our method performs well in the presence of noise found in Internet photo collections, but other shadow detection methods (*e.g*.,

[7, 25]) may perform as well or better in single-image and more controlled multi-image settings.

7. Conclusions

We have presented a method for shadow detection in photo collections suitable both for unstructured photo collections and image stacks, and demonstrated its applicability to automatically estimating sun position in photographs using attached shadows. Although our analysis makes several simplifying assumptions, the method is surprisingly effective in practice. We believe that robust shadow labels and sun direction in photo collections can enable or improve a variety of applications in outdoor reconstruction, appearance modeling, and scene understanding.

Our approach could be extended in a variety of ways. First, we could devise a more sophisticated scoring approach that considers the entire graph of votes between pixel pairs globally, or considers neighboring labels. Second, once we compute average intensities, we estimate shadow masks independently for each image; one could instead jointly predict shadows across the entire photo collection at once, making use of richer statistical information available among images. Third, one challenging case for our method are points that are nearly always in shadow; we could explore using statistics such as variance to identify how frequently a point is in shadow as a prior on shadow likelihood. Finally, we plan to use our techniques in further estimation of illumination and appearance in photo collections in the wild.

Acknowledgements This work was supported in part by grants from the NSF (IIS-1111534), Amazon AWS for Education, Google, and the Intel Science and Technology Center for Visual Computing. The authors thank Paul Upchurch and Daniel Hauagge for their help with figures, and Joe Kider for assistance in capturing the TENTACLE dataset.

References

[1] A. Abrams, C. Hawley, K. Miskell, A. Stoica, N. Jacobs, and R. Pless. Shadow estimation method for "the episolar constraint: Monocular shape from shadow correspondence". *CoRR*, abs/1304.4112, 2013.

[2] A. Abrams, C. Hawley, and R. Pless. Heliometric stereo: Shape from sun position. In *ECCV*, 2012.

[3] A. Abrams, K. Miskell, and R. Pless. The episolar constraint: Monocular shape from shadow correspondence. In *CVPR*, 2013.

[4] J. Ackermann, F. Langguth, S. Fuhrmann, and M. Goesele. Photometric stereo for outdoor webcams. In *CVPR*, 2012.

[5] A. Bouthors. *Realistic rendering of clouds in realtime*. PhD thesis, Grenoble Universities, 2008.

[6] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004.

[7] R. Guo, Q. Dai, and D. Hoiem. Single-image shadow detection and removal using paired regions. In *CVPR*, 2011.

[8] T. Haber, C. Fuchs, P. Bekaert, H.-P. Seidel, M. Goesele, and H. P. Lensch. Relighting objects from image collections. In *CVPR*, 2009.

[9] D. Hauagge, S. Wehrwein, K. Bala, and N. Snavely. Photometric ambient occlusion. In *Proceedings of CVPR*, 2013.

[10] D. Hauagge, S. Wehrwein, P. Upchurch, K. Bala, and N. Snavely. Reasoning about photo collections using models of outdoor illumination. In *Proceedings of BMVC*, 2014.

[11] R. Kittler and S. Darula. CIE general sky standard defining luminance distributions. In *Proc. of Canadian Conf. on Building Energy Simul.*, 2002.

[12] P.-Y. Laffont, A. Bousseau, S. Paris, F. Durand, and G. Drettakis. Coherent intrinsic images from photo collections. *SIGGRAPH Asia*, 2012.

[13] J.-F. Lalonde, A. Efros, and S. Narasimhan. Estimating natural illumination from a single outdoor image. In *CVPR*, 2009.

[14] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan. Webcam Clip Art: Appearance and illuminant transfer from time-lapse sequences. *SIGGRAPH Asia*, 28(5), December 2009.

[15] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan. Detecting ground shadows in outdoor consumer photographs. In *ECCV*, 2010.

[16] A. Levin, D. Lischinski, and Y. Weiss. A closed form solution to natural image matting. In *CVPR*, 2006.

[17] R. Martin-Brualla, D. Gallup, and S. M. Seitz. Time-lapse mining from internet photos. 2015.

[18] K. Matzen and N. Snavely. Scene chronology. In *Proc. European Conf. on Computer Vision*, 2014.

[19] P. Nillius and J.-O. Eklundh. Automatic estimation of the projected light source direction. In *CVPR*, 2001.

[20] T. Okabe, I. Sato, and Y. Sato. Spherical harmonics vs. haar wavelets: Basis for recovering illumination from cast shadows. In *CVPR*, 2004.

[21] A. Panagopoulos, T. Vicente, and D. Samaras. Illumination estimation from shadow borders. In *IEEE Color and Photometry In Computer Vision Workshop*, pages 798 –805, 2011.

[22] I. Sato, Y. Sato, and K. Ikeuchi. Illumination from shadows. *PAMI*, 25(3), 2003.

[23] Q. Shan, R. Adams, B. Curless, Y. Furukawa, and S. M. Seitz. The Visual Turing Test for scene reconstruction. In *3DV*, 2013.

[24] B. Shi, K. Inose, Y. Matsushita, P. Tan, S.-K. Yeung, and K. Ikeuchi. Photometric stereo using internet images. *3DV*, 2014.

[25] K. Sunkavalli, W. Matusik, H. Pfister, and S. Rusinkiewicz. Factored time-lapse video. In *SIGGRAPH*, 2007.

[26] J. Zhu, K. G. G. Samuel, S. Masood, and M. F. Tappen. Learning to recognize shadows in monochromatic natural images. In *CVPR*, 2010.