The Episolar Constraint: Monocular Shape from Shadow Correspondence

Austin Abrams, Kylia Miskell, Robert Pless
Washington University in St Louis
{abramsa|klm4|pless}@cse.wustl.edu

Abstract

Shadows encode a powerful geometric cue: if one pixel casts a shadow onto another, then the two pixels are colinear with the lighting direction. Given many images over many lighting directions, this constraint can be leveraged to recover the depth of a scene from a single viewpoint. For outdoor scenes with solar illumination, we term this the episolar constraint, which provides a convex optimization to solve for the sparse depth of a scene from shadow correspondences, a method to reduce the search space when finding shadow correspondences, and a method to geometrically calibrate a camera using shadow constraints. Our method constructs a dense network of nonlocal constraints which complements recent work on outdoor photometric stereo and cloud based cues for 3D. We demonstrate results across a variety of time-lapse sequences from webcams "in the wild."

1. Introduction

A pixel under shadow has a dramatically different intensity than the same pixel under direct lighting. Vision applications often incorporate shadows into their models, either by treating them as noise to be detected and ignored [8, 23], exploiting them as cues for camera calibration [5, 13], or incorporating them into larger image formation models [1, 3].

In this paper, we treat shadows as a strong geometric cue: if a pixel is under shadow, then it must be the case that some other object along the lighting direction is casting a shadow onto it. For outdoor imagery, a geolocated camera and accurate timestamps cause this colinearity to have a known georeferenced direction. If the camera also has known geometric calibration, we can express this property as a linear constraint over the depth of each pixel involved. From this geometry, we derive three novel results:
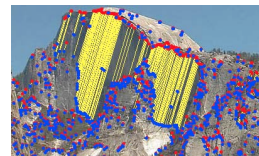
- An image-space constraint between a shadow and its occluder,

- An approach to geometrically calibrate a camera from shadow correspondences, and
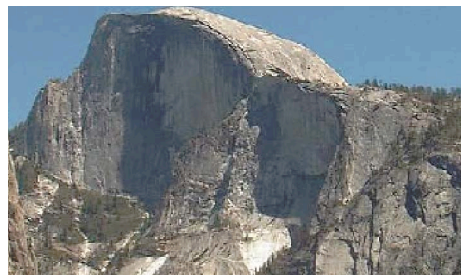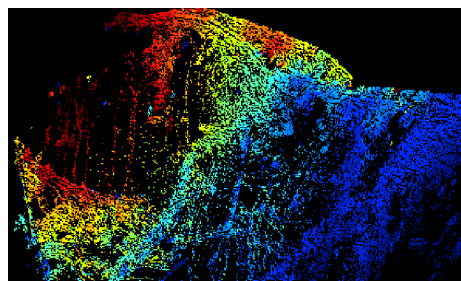

(a)

(b)   (c)

(d)

Figure 1. In this paper, we exploit the inherent structure of cast shadows to recover shape from a single view. Given a time-lapse sequence from a geographically-calibrated camera (a), we create correspondences (shown as a yellow line) between a shadow (blue) and its occluding object (red) (b). Repeated across the image (c) and across many lighting directions, these tens of thousands of correspondences can be used as a cue to recover a sparse depth map from a single viewpoint (d). Depth increases from blue to red.

- A convex optimization to solve for the unknown depths for a sparse set of pixels from shadow correspondence.

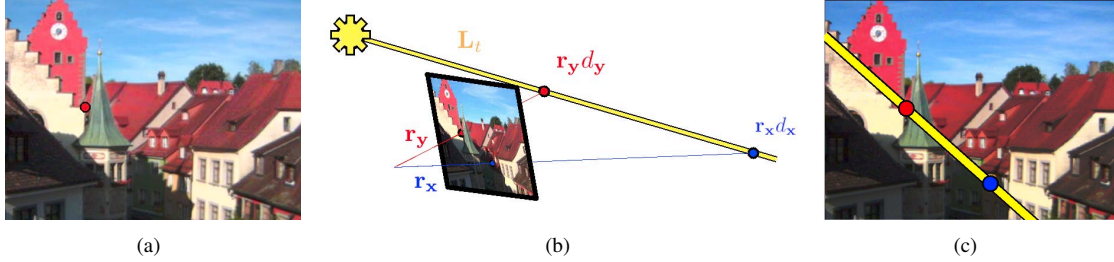Inferring depth from shadow correspondences has sev-

Figure 2. Visualizing the episolar constraint. Where could the red point in (a) cast a shadow in the scene? This point must lie on the plane spanned by the 3D pixel ray for the red point and the 3D lighting direction, shown in yellow (b). This solar plane intersects the image plane, defining the episolar line (c). Finding the correct shadow correspondence therefore constrains the relative depth of each point.

eral desirable properties over other monocular cues for depth. First, shadow correspondences capture general shape: we do not require the ground to be planar or even visible, nor do we require the depth surface to be smooth or continuous. Next, since we work directly with binary shadow masks, rather than intensities, we do not need to account for real-world photometric distortions such as variable exposure and radiometric response, so long as the shadow extraction pipeline is sufficiently robust. Finally, as demonstrated in Figure 1, we derive constraints which do not suffer from the aperture problem commonly found in other correspondence problems.

2. Related Work

A large body of work focuses on recovering shape from shadow-based cues. Early work focused on interpreting shadows from line drawings: Shafer and Kanade [22] introduced a general theory for describing the orientation of surfaces by the shadows they cast onto each other. Lowe and Binford [17] build a reasoning system to infer structure from line drawings, where one cue leverages manually-specified correspondences between a shadow and its caster.

Most shape-from-shadows approaches find depth surfaces consistent with some shadow-or-not labeling across many images. Early work by Hatzitheodorou and Kender [9] introduces an approach to recover the shape of a one-dimensional surface slice from the shadows it casts on itself, extended by Raviv et al. [19] to work with 2D surfaces.

Later, Savarese et al. [21] leveraged epipolar geometry to carve out a surface from shadow labels across multiple views (see [15] for a survey on space carving). Although we work in the single-viewpoint scenario, we borrow concepts from epipolar geometry in a similar way, by treating the light source as a secondary camera.

Shadowgrams [7], shadow graphs [24], and shadow/antishadow constraints [6] all encode a constraint similar to the one presented in this paper: all pixels on the image-space line between a shadow and its occluder

should have a height below the corresponding 3D line. In contrast, we do not place any constraint on the intermediate pixels between a shadow and its occluder, which removes the assumption that the depth surface is terrain-like. Also, these works assume an orthographic camera, whereas we work with pinhole cameras.

Kawasaki and Furukawa [14] treat shape-from-shadows as a kind of structured light, where a wand is waved in front of the light source, and recover depth by constraining that the group of pixels shaded by the wand in any particular frame are coplanar in 3D. In this work, we do not place assumptions on the shape of the object that casts shadows in each frame.

Recently, Bamber et al. [4] implement a single-view shadow carving algorithm suitable for long-term time-lapses. However, they make the assumption that the ground plane is large and visible. In contrast, we work with scene where the geometry is unknown a priori and the ground plane may not be visible.

3. Episolar Geometry

In this section, we derive the geometric constraints between a shaded pixel and the pixel that cast its shadow. The geometry of this constraint is equivalent to considering the sun as an orthographic camera.

We denote pixels as boldface vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$. We assume that we know the lighting direction $\mathbf{L}_t \in \mathbb{R}^3$ at each time t, which can be recovered using a solar position algorithm, given accurate timestamps and GPS [20]. For this work, we assume the camera is centered at the origin and has been geo-calibrated, giving each pixel's ray into space $\mathbf{r_x} \in \mathbb{R}^3$. For clarity, we assume that the lighting direction and each pixel ray is a unit direction vector $||\mathbf{L}_t|| = ||\mathbf{r_x}|| = 1$.

The goal is to find the per-pixel depth $d_\mathbf{x}$. Throughout the paper, we treat $\mathbf{y}$ as the object that casts the shadow, and $\mathbf{x}$ as the object that receives the shadow. In an abuse of terminology, the phrase "$\mathbf{y}$ casts a shadow onto $\mathbf{x}$" should be interpreted as "the 3D object that projects onto the image
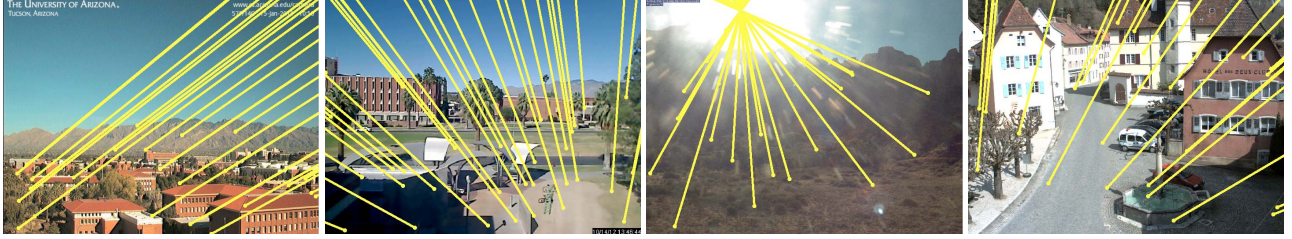
Figure 3. Example episolar lines emerging from a small sample of randomly-selected pixels. Notice that the "episole"—where all lines intersect—lies on the sun, if it is in view, or exactly opposite the sun, if it is behind the camera.

at $\mathbf{y}$ casts a shadow onto the 3D object that projects onto the image at $\mathbf{x}$". For consistency, in all figures, $\mathbf{y}$ and $\mathbf{x}$ are represented as a red and blue points, respectively.

Suppose that some pixel $\mathbf{y}$ casts a shadow onto some other pixel $\mathbf{x}$ for some lighting direction $\mathbf{L}_t$; we denote such a correspondence as $\mathbf{y} \rightsquigarrow_t \mathbf{x}$. Assuming directional lighting, this correspondence emplaces a constraint on the depths d of pixels at $\mathbf{x}$ and $\mathbf{y}$:

$$\mathbf{r_x} d_\mathbf{x} + \mathbf{L}_t \alpha_\mathbf{xy} = \mathbf{r_y} d_\mathbf{y}, \tag{1}$$

where $\alpha_\mathbf{xy}$ is the unknown 3D distance between pixels $\mathbf{x}$ and $\mathbf{y}$. This constraint takes the form of a linear constraint involving the unknown depth of each pixel and the 3D distance between $\mathbf{x}$ and $\mathbf{y}$. This property holds a close relationship with well-known epipolar geometry, so we denote Equation 1 as the *episolar constraint*. See Figure 2 for a visualization of the episolar constraint.

Notice that this property is true for all types of geometry. Nowhere do we make the assumption that our scene has a substantial ground plane, or that the depth surface is smooth or continuous.

We take advantage of this linear relationship in three distinct ways. First, this property defines an image-space constraint between an object and its shadow. This reduces the search space to 1D when determining shadow correspondences. Second, we derive a nonlinear optimization to geometrically calibrate a camera from shadow correspondences. In contrast to previous work, this calibration does not place any assumption on the underlying geometry nor require that the camera sees the sky. Finally, given correspondences from a variety of lighting directions, we derive a convex optimization procedure which recovers the depths of all pixels involved.

3.1. The Episolar Line

Generating correspondences between a shadow $\mathbf{x}$ and its occluder $\mathbf{y}$ is a challenging problem, but Equation 1 sheds some light on the shadow correspondence problem. If $\mathbf{y}$ casts a shadow onto some unknown location $\mathbf{x}$, then the point $\mathbf{r_x} d_\mathbf{x}$ must lie in the linear subspace spanned by $\mathbf{r_y}$ and $\mathbf{L}_t$. This linear subspace corresponds to a plane in

3D[1] which intersects the image as a line passing through $\mathbf{y}$. Therefore, if a pixel $\mathbf{y}$ casts a shadow, then its corresponding pixel $\mathbf{x}$ must lie on this *episolar line*.

Although this constraint alone does not dictate *where* on the episolar line the shadow truly comes from, it dramatically reduces the search space necessary for shadow correspondence. In Section 4 we describe how to complete this correspondence by taking advantage of self-shading priors.

Of practical interest is that the episolar line does not suffer from the common aperture problem seen in other correspondence problems. For example, linking a roofline to its horizontal shadow would be ambiguous without using this constraint; any point on the roof could conceivably produce a shadow anywhere on the shadow edge. However, this horizontal shadow will cross the episolar line at exactly one point, disambiguating the aperture problem. Figure 7 has several examples of this behavior.

This is especially useful in generating a dense network of constraints. If we could only generate correspondences on shadow corners, the constraint set might not be dense enough to use reliably. However, since we can create correspondences across shadow edges, our overall correspondence set is much more informative of the underlying geometry. In Section 5, we explore the connectedness properties of real scenes and show that, provided there are enough images, the resulting constraints form large connected components across the image.

3.2. Episolar Calibration

Notice that in order to generate the episolar line, we need estimates of the camera's calibration to determine pixel rays $\mathbf{r}$ in the same coordinate frame of $\mathbf{L}$ (in our case, the East-North-Up space). However, estimating the geometric calibration of an outdoor camera is nontrivial. Various approaches exist for calibration from outdoor cues such as sky color [16] or shadow trajectories cast onto the ground plane [5, 13]. Webcams "in the wild" often do not have these features, as the sky might occupy only a small portion of the image, and the ground might not be planar or visible.

---

[1] A similar plane forms the basis for much of the work in the shadow carving approach presented in [21].

However, cast shadows are abundant in most outdoor scenes. Here, we leverage user-supplied shadow correspondences to calibrate a camera. Through the episolar constraint, we find the camera calibration parameters θ that define a pinhole camera which produces episolar lines most consistent with the given correspondences.

More formally, if a user supplies a set of ground truth shadow correspondences $G = \{\mathbf{y}_i \leadsto_{t_i} \mathbf{x}_i\}$, and $e_\theta(\mathbf{x}, t) \in \mathbf{R}^2$ defines the unit-vector episolar direction for a pixel $\mathbf{x}$ at time t under camera parameters θ, we solve the nonlinear optimization

$$\theta^* = \arg\min_{\theta, \beta} \sum_{i \in G} ||\mathbf{x}_i + \beta_i e_\theta(\mathbf{x}_i, t_i) - \mathbf{y}_i||^2, \quad (2)$$

where β_i is the distance between $\mathbf{x}_i$ and $\mathbf{y}_i$ along the episolar line (analogous to α in Equation 1). In practice, we do not optimize over β, but rather substitute the least-squares solution of β given θ:

$$\beta_i^* = e_\theta(\mathbf{x}_i, t_i)^\top (\mathbf{y}_i - \mathbf{x}_i). \quad (3)$$

After substitution, 2 becomes a nonlinear optimization over the camera parameters θ.

This optimization is nonconvex, so we seed the initialization by trying 1000 random settings of camera parameters, choosing the one that gives the lowest error. From there, we run a Levenberg-Marquardt optimization [18] to simultaneously optimize for the camera's pan, tilt, roll, and focal length. The correspondences used for calibration are not used for any other step.

In contrast to previous work, our calibration approach does not require any of the sky to be in view, and it does not make assumptions about the underlying geometry of the scene. For many outdoor scenes, such as the camera shown in Figure 4, these assumptions would be too restrictive.

3.3. Episolar Integration

Given shadow correspondences C across a variety of lighting directions, the episolar constraint yields a depth inference process which can be cast as a constrained convex program:

$$\arg\min_{d, \alpha} \sum_{\mathbf{y} \leadsto_t \mathbf{x} \in C} ||\mathbf{r}_\mathbf{x} d_\mathbf{x} + \mathbf{L}_t \alpha_{\mathbf{xy}} - \mathbf{r}_\mathbf{y} d_\mathbf{y}||^2 \quad \text{s.t.} \quad d \geq 1. \quad (4)$$

Notice that we constrain the solution so that $d \geq 1$. This both sets the scale of the system and prevents the trivial solution $d = 0$. Since our goal is to recover the depths d, we can again express the optimal $\alpha_{\mathbf{xy}}^*$ in terms of the following linear system:

$$\mathbf{L}_t \alpha_{\mathbf{xy}}^* = \mathbf{r}_\mathbf{y} d_\mathbf{y} - \mathbf{r}_\mathbf{x} d_\mathbf{x} \quad (5)$$

$$\alpha_{\mathbf{xy}}^* = \mathbf{L}_t^\top (\mathbf{r}_\mathbf{y} d_\mathbf{y} - \mathbf{r}_\mathbf{x} d_\mathbf{x}) \quad (6)$$
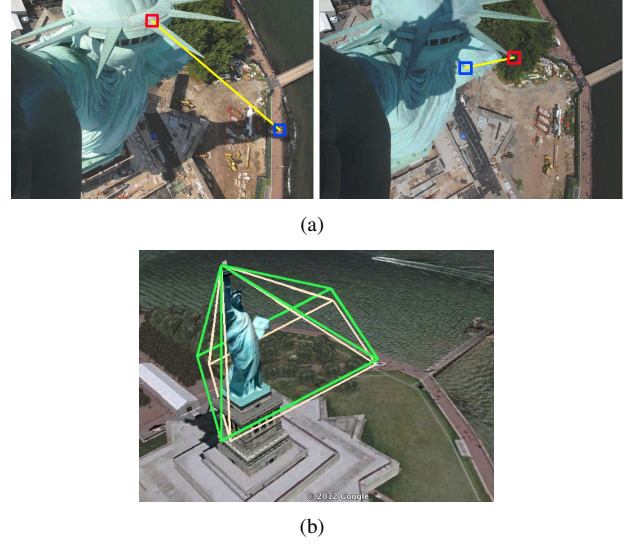

(a)

(b)

Figure 4. Calibrating a camera through episolar constraints. Given a few ground truth correspondences (examples shown in (a)), we find the camera position most consistent with those correspondences (b). We compare our results (beige frustum) to the results from [2] (green frustum), which uses hand-selected 3D-to-2D correspondences as determined by Google Earth geometry.

By substitution of α^* into Equation 4, we can express the problem only in terms of the unknown depth d:

$$\arg\min_{d \geq 1} \sum_{\mathbf{y} \leadsto_t \mathbf{x} \in C} ||(\mathbf{r}_\mathbf{x} - \mathbf{L}_t \mathbf{L}_t^\top \mathbf{r}_\mathbf{x}) d_\mathbf{x} - (\mathbf{r}_\mathbf{y} - \mathbf{L}_t \mathbf{L}_t^\top \mathbf{r}_\mathbf{y}) d_\mathbf{y}||^2 \quad (7)$$

Although Equations 4 and 7 are mathematically equivalent, the removal of the α has enormous practical benefits. In the scenes we work with, there are tens of thousands of correspondences, each one with its own α. Optimizing only on the depth yields a much smaller optimization problem.

Although this system of equations has a well-defined global solution with only one correspondence, it also extends to a network of linked constraints. That is, if some pixel $\mathbf{y}$ casts a shadow onto both $\mathbf{x}$ and $\mathbf{x}'$ at different times, this formulation constrains the relative depth of $\mathbf{x}, \mathbf{x}'$, and $\mathbf{y}$. Similarly, if $\mathbf{x}$ is shaded by two different pixels $\mathbf{y}$ and $\mathbf{y}'$ at different times, this places constraints onto $\mathbf{x}, \mathbf{y}$, and $\mathbf{y}'$. This network of constraints is demonstrated in Figure 5.

Because this process solves for a depth surface consistent with a set of depth differences, we denote the optimization in Equation 7 as *episolar integration*.

4. Correspondence Generation

Although the episolar line reduces the search space for shadow correspondence to be along a line, it remains an open problem to robustly link a shadow to its caster. In this paper, we use a fairly simple correspondence generation
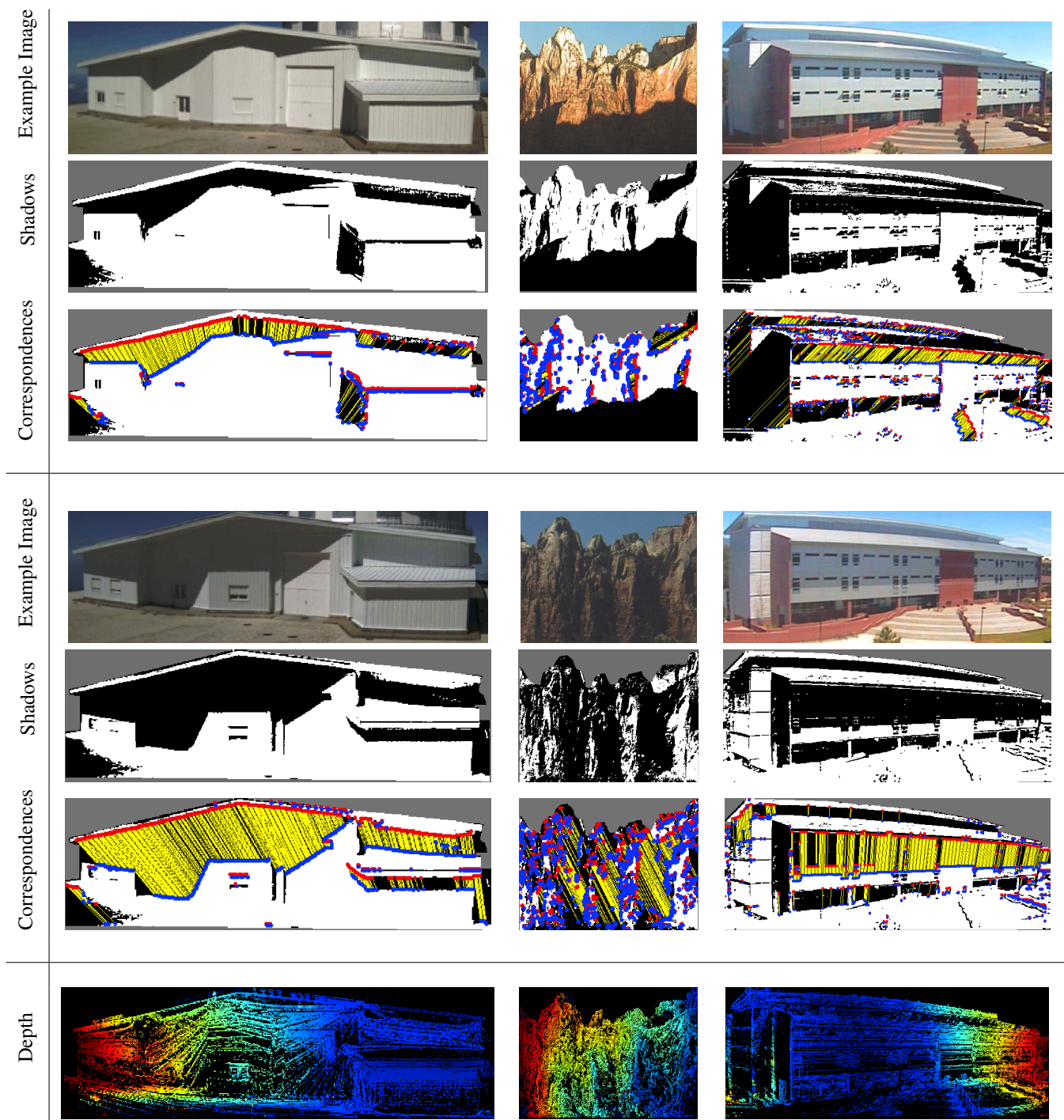
Figure 7. Results from our depth inference process on cameras from the AMOS dataset. From top to bottom, we show a crop from an example image, its shadow mask, and the extracted shadow correspondences (for two images). The bottom row shows the recovered depth. Correspondences are shown as connections (yellow line) between an occluder (red point) and its shadow (blue). Depth increases from blue to red. Notice that the episolar line provides enough constraints to overcome the aperture problem, common in other correspondence problems.
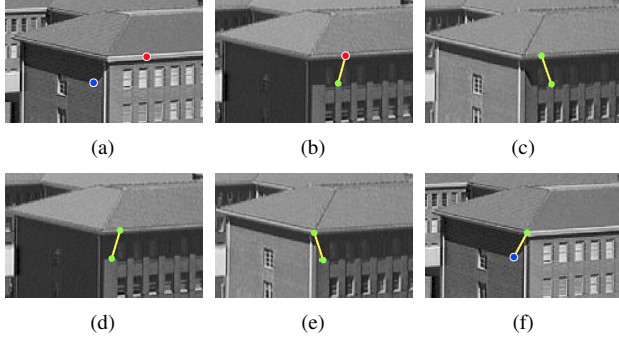
Figure 5. Visualizing a small portion of the constraint graph. Although there is never a time when the red point directly casts a shadow onto the blue point (a), there are enough intermediate constraints (b)-(f) to implicitly constrain the relative depths of the two points, and all intermediate points involved (green).

rule which works well for most cases of self-shading.

Given an input sequence of imagery from a diverse set of lighting directions, we first apply an in-house shadow estimation approach which returns a shadow-or-not label for all pixels in sequence.

When this method classifies some pixel $\mathbf{y}$ on a shadow edge as under direct illumination at time t, our goal is to find which pixel—if any—receives the shadow produced by $\mathbf{y}$. We employ a greedy strategy by taking incremental steps along the episolar line emerging from $\mathbf{y}$. If the first step away from $\mathbf{y}$ is under shadow, we walk along the episolar direction until we find a pixel $\mathbf{x}$ which is directly lit again. We then create the correspondence $\mathbf{y} \rightsquigarrow_t \mathbf{x}$. However, if the first step away from $\mathbf{y}$ is still directly lit, no correspondence is generated (i.e., contiguous lit regions only generate correspondences on their edges). We repeat this process for all lit pixels $\mathbf{y}$ at all times t.

In natural scenes, shadow correspondences tend to start in the same locations in the images (rooflines, convexities in mountain ridges, etc.), but end in many different locations. From this observation, we use a simple heuristic to remove correspondences which begin or end in unlikely locations. Using the full set of correspondences, we estimate the probability of a correspondence starting or ending at any given pixel $\mathbf{z}$:

$$P_{start}(\mathbf{z}) = \sum_{\mathbf{z} \rightsquigarrow_t \mathbf{x}} \frac{1}{n}, \qquad P_{end}(\mathbf{z}) = \sum_{\mathbf{y} \rightsquigarrow_t \mathbf{z}} \frac{1}{n}, \quad (8)$$

where n is the number of images. We remove any correspondence $\mathbf{y} \rightsquigarrow_t \mathbf{x}$ where $P_{start}(\mathbf{y}) \leq 0.1$ or $P_{end}(\mathbf{x}) \geq 0.1$. The correspondence generation pipeline is described in Figure 6.

This heuristic helps to remove two common error modes. First, if a shadow is cast on the ground far away from its occluder, as in Figure 6(a), correspondences will be gener-
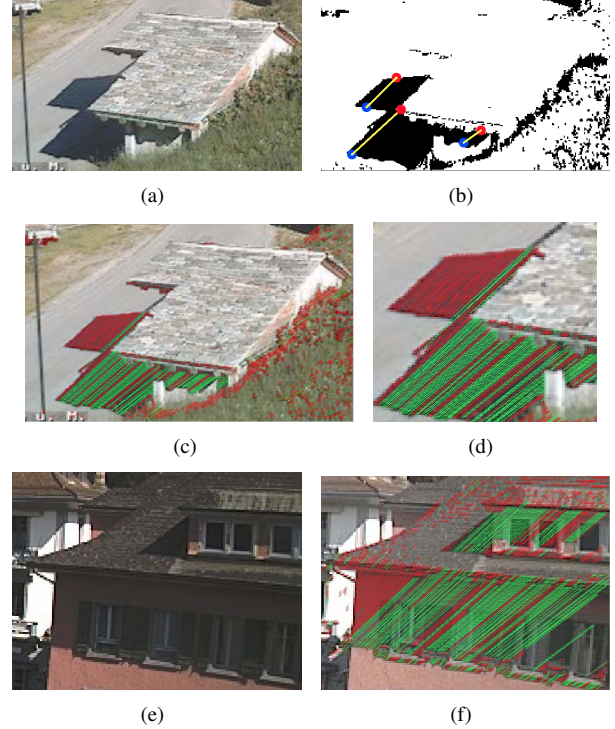

Figure 6. Generating shadow correspondences. From a time-lapse sequence (one example image shown in (a)), we extract a shadow-or-not labeling for each image (b). For all lit pixels on shadow boundaries, we follow their episolar lines until we find another pixel which is directly illuminated (three examples shown). From here, we remove any correspondence that starts or ends in an unlikely place (c), detail crop in (d) (All correspondences marked in green are kept, red are removed; see text for details). In this case, all correspondences that start on the ground are removed. In (e) and (f), all correspondences that stop at the vertical edge of the building are removed.

ated from one side of the cast shadow to the other. However, since "correspondences" rarely start in the middle of the ground plane, they will be filtered out. Second, the initial rule will stop many correspondences at geometry edges when the background is lit and the foreground is not, as in Figure 6(f). These false correspondences will be filtered out because it is rare for a true correspondence to stop in the same place repeatedly.

5. Results

We test our approach on several real-world cameras from the AMOS dataset [12] and show qualitative results in Figure 7. For each camera, we select 100 images from a diverse set of lighting directions with clear skies, and use a multi-scale alignment procedure adapted from [11] to remove small jitter. To recover lighting directions, we use the solar position algorithm from [20]. When calibrating the

(a) Example image   (b) Recovered correspondences

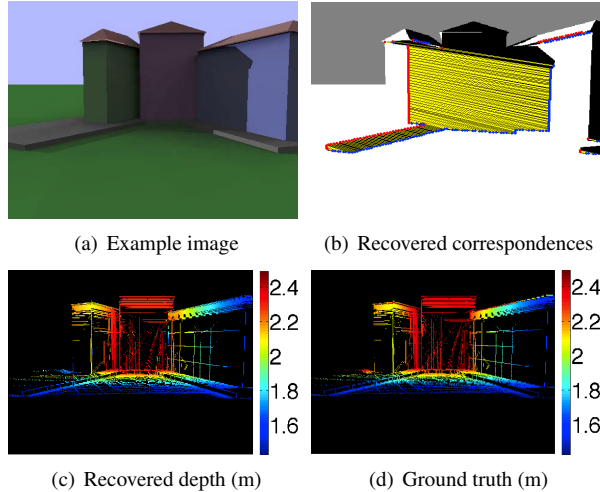
(c) Recovered depth (m)   (d) Ground truth (m)

Figure 8. Experiments on a synthetic dataset; example image shown in (a). The correspondences recovered from this scene (b) are rich enough to extract a depth map (c) very close to the ground truth (d). The recovered depth map differs from the ground truth by an average of 0.05 meters.

(a) Example image   (b) Recovered correspondences

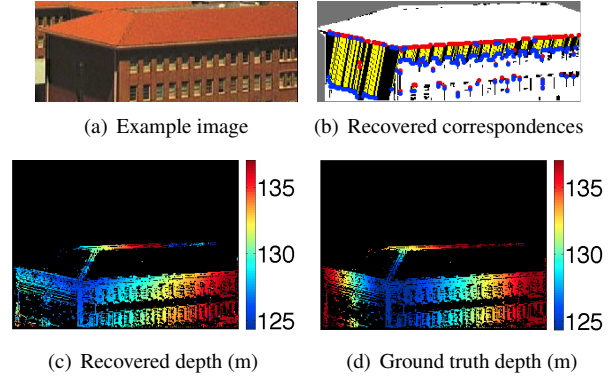(c) Recovered depth (m)   (d) Ground truth depth (m)

Figure 9. Quantitative evaluation of recovered depth. Given a sequence of images (example in (a)), we recover shadow correspondences (b) and a depth map (c). We compare our results to Google Earth models (d). For this structure roughly 120 meters away from the camera, almost all pixels are less than 4 meters away from their ground truth location (using the ground truth to set the scale). This corresponds to a 3.2% error.

camera using the nonlinear optimization in Section 3.2, we optimize over 50 manually-chosen correspondences.

Notice that our approach reliably extracts depth from a variety of complicated geometry and that although the resulting depth map is sparse, the network of constraints covers a large portion of the scene. To give scale, a typical scene has roughly 70,000 constraints across 30,000 pixels.

Our runtime is largely dependent on the complexity of the shadow masks and image resolution, but we report timing with respect to a camera with 135,000 pixels on a 2.53 GhZ Intel Core 2 Duo with 8GB of memory. The most time-consuming aspect is in computing the shadow masks, which took 4m40s. Creating and filtering correspondences takes another 42 seconds, and solving for depths took 23 seconds.

We also generated a synthetic sequence with rendered cast shadows on a complex scene, shown in Figure 8. Our recovered depth surface is almost exactly the ground truth.

To measure quantitative error on real scenes, we compare our depth maps to Google Earth geometry. Since our depth maps are known only up to an unknown scale, we use the ground truth to resolve the scale and compare relative error. Figure 9 demonstrates that our depth maps closely match the ground truth.

Finally, to emphasize the importance of using many images, in Figure 10 we explore how the connectedness of the constraint graph increases with more images. This shows that up to roughly 20-30 images, most correspondences form relatively local clusters constraining few pixels. There is then a transition where groups merge and the size increases quickly. After that, shadows that cast onto new parts
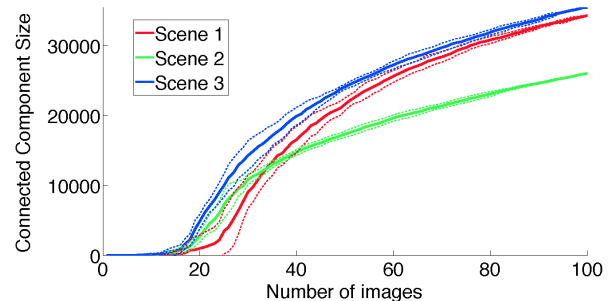

Figure 10. Exploring the connectedness of the constraint graph. For the three cameras in Figure 7, we show the size of the largest connected component in the constraint graph as a function of number of images used. We average results over 10 trials, selecting random subsets of the original imagery and showing the mean and 1 standard deviation as solid and dotted lines, respectively.

of the scene are incorporated into the model and the size of the largest connected component grows linearly. This suggests we have not yet reached diminishing returns, in that we can continue to add more imagery and expect more of the scene to be incorporated.

6. Conclusions

Of course, there are cases in which our admittedly naïve correspondence generation technique will not work correctly. For example, the shadow labeling between the tip of a vertical pole to its shadow on the ground plane will almost certainly not be entirely shaded, thus creating a false correspondence. We anticipate that enforcing appearance similarity priors for nearby lighting directions will help leverage correspondence generation for more complicated cases.

Despite this limitation, our simple rule works well for most cases of self-shading.

Our approach only gives a sparse representation of the depth, reconstructing the depths of pixels which cast a shadow or had shadows cast onto them. While this network of constraints still covers a large portion of the image, an ideal solution would merge this constraint with other depth inference processes such as outdoor photometric stereo [1, 3] or shape-from-clouds [10] to "fill in the gaps."

In this paper, we present an approach for recovering the depth surface of an outdoor scene by treating the sun as a second camera and establishing correspondences between a shadow and its caster. This provides a nonlocal depth integration algorithm, as well as an image-space constraint which dictates which potential correspondences are geometrically feasible. These constraints are particularly useful for shape reconstruction, because the correspondence step does not suffer from the aperture problem, and our derivation makes no assumptions on the shape of the depth surface.

References

[1] A. Abrams, C. Hawley, and R. Pless. Heliometric stereo: shape from sun position. In *Proc. European Conference on Computer Vision*, 2012.

[2] A. Abrams and R. Pless. Webcams in context: Web interfaces to create live 3d environments. In *Proc. ACM SIGMM International Conference on Multimedia (ACMMM)*, 2010.

[3] J. Ackermann, F. Langguth, S. Fuhrmann, and M. Goesele. Photometric stereo for outdoor webcams. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[4] D. C. Bamber, J. D. Rogers, and S. F. Page. A method for 3d scene recognition using shadow information and a single fixed viewpoint. In *Visual Information Processing*, 2012.

[5] X. Cao and H. Foroosh. Camera calibration and light source orientation from solar shadows. *Computer Vision and Image Understanding*, 105:60–72, Jan 2007.

[6] M. Chandraker, S. Agarwal, and D. Kriegman. ShadowCuts: Photometric stereo with shadows. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[7] M. Daum and G. Dudek. On 3-d surface reconstruction using shape from shadows. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 1998.

[8] R. Guo, Q. Dai, and D. Hoiem. Single-image shadow detection and removal using paired regions. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[9] M. Hatzitheodorou and J. Kender. An optimal algorithm for the derivation of shape from shadows. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 1988.

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

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

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

[13] I. Junejo and H. Foroosh. Using solar shadow trajectories for camera calibration. In *Proc. IEEE International Conference on Image Processing*, pages 189 –192, oct. 2008.

[14] H. Kawasaki and R. Furukawa. Shape reconstruction and camera self-calibration using cast shadows and scene geometries. *International Journal of Computer Vision*, 83(2):135–148, 2009.

[15] K. Kutulakos and S. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

[16] J.-F. Lalonde, S. G. Narasimhan, and A. A. Efros. What do the sun and the sky tell us about the camera? *International Journal of Computer Vision*, 88(1):24–51, May 2010.

[17] D. Lowe and T. Binford. The interpretation of three-dimensional structure from image curves. In *International Joint Conference on Artificial Intelligence*, 1981.

[18] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11(2):431–441, 1963.

[19] D. Raviv, Y.-H. Pao, and K. Loparo. Reconstruction of three-dimensional surfaces from two-dimensional binary images. *IEEE Transactions on Robotics and Automation*, 5(5):701 –710, oct 1989.

[20] I. Reda and A. Andreas. Solar position algorithm for solar radiation applications. In *NREL Report No. TP-560-34302*, 2003.

[21] S. Savarese, H. Rushmeier, F. Bernardini, and P. Perona. Shadow carving. In *Proc. IEEE International Conference on Computer Vision*, 2001.

[22] S. A. Shafer and T. Kanade. Using shadows in finding surface orientations. *Computer Vision, Graphics, and Image Processing*, 22(1):145 – 176, 1983.

[23] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma. Robust photometric stereo via low-rank matrix completion and recovery. In *Proc. Asian Conference on Computer Vision*, 2010.

[24] Y. Yu and J. T. Chang. Shadow graphs and surface reconstruction. In *Proc. European Conference on Computer Vision*, 2002.