Vertical Parallax from Moving Shadows

Yaron Caspi*        Michael Werman

October 7, 2004

Abstract

This paper presents a simple method for capturing and computing 3D parallax, where by 3D parallax we mean parallax vertically relative to the to the ground plane, height. The method is based on analyzing shadows of vertical poles (e.g a tall building’s contour) that sweep the object. Unlike existing beam scanning approaches (shadow or structured light) that recover the distance of a point from the camera, our approach measures the height from the ground plane directly. The constraints used by previous methods are based on intersecting a plane, defined by the light source and the straight casting object with the object we wish to reconstruct. Such a triangulation formulation requires accurate knowledge of the light source position. In contrast, our approach intersects two (unknown) planes generated separately by two casting objects. This omits the need to precompute the location of the light source. Furthermore, it allows a moving light source to be used.

The proposed setup is particularly useful, when the camera cannot face the scene orthogonally or when the object is far away from the camera. A good example is an urban scene captured by a single webcam. Focusing on parallax rather than depth is not limited to outdoor setups. The paper also describes a simple desktop setup/device that accurately captures height. Another candidate for our approach are structured light-based 3D reconstruction systems. We describe modifications to the projected patterns and to the capturing setup that enable height rather than depth to be captured.

*Yaron Caspi was supported by the Vallazi-Pikovsky Fellowship.
1. Introduction

Many of shadow or structured light-based methods for reconstructing three dimensional structure of objects have been proposed in the past ([1, 3, 4, 9] to list but a few). By shadow or structured light methods we refer to methods that combine a single camera with a source of light, e.g., a laser beam, light strip (or stripes) from a projector or a cast shadow edge. Just as in stereo applications, structured light methods tend to produce a depth map, i.e., a representation that captures the distance of each scene point from the camera.

A natural alternative for depth map is Plane + Parallax (P+P) representation. Plane + Parallax (P+P) has been touted as an excellent representation for 3D reconstruction (e.g., [11, 13, 16, 17]). In contrast to depth maps (i.e., the distance of each scene point to the camera) commonly used in stereo applications, a P+P representation recovers point parallax. The term parallax refers to the residual misalignment after alignment of the reference plane. Given that this parallax is proportional to the point’s height from the reference plane, it provides an alternative means of capturing 3D structure.

A three dimensional Plane + Parallax (P+P) representation is typically recovered from point matches. To the best of our knowledge, 3D reconstruction from cast shadow edges or light strips, were limited to recovery of depth maps, as their intermediate representation. This paper attempts to bridge between shadow/structured light and Plane + Parallax.

The paper presents a simple method for computing a P+P representation from moving shadows (or structured light). It recovers a dense parallax map of objects lying on a planer surface (the reference plane). Furthermore, the output is a special case of P+P. We recover a ”vertical parallax”. We construct an intermediate representation denoted in the paper as ”vertical mapping”, which is a dense mapping that assigns a vertically corresponding point on the reference plane to each scene point (the point on the ground plane that is exactly below it).
The proposed method input consists of two image sequences of a scene, static except for moving shadows cast by vertical obstructions, where either/both the light source and obstructor(s) moves. To illustrate these requirements our canonic scenario is when an object in an outdoor scene is ”swept” by shadow edges twice, each time by a different vertical obstruction (e.g., two buildings). The ”sweep” results from the motion of the sun.

The cast shadow of a building edge on the reference plane is a line. The 3D object on the reference plane deforms this line. Combining the information of two such deformations enables the construction of a simple constraint that captures each point’s parallax.

Once our setup and the accompanying reconstruction is completed, we can assign real-world Euclidean units (up to a global scale factor) to every visible point above the reference plane. For this task known theoretical results on P+P representation can be applied. For example, we employed in our setup results with the method described by Criminisi et al. [6]) The benefit of our approach is that the vertical correspondence is generated automatically for every scene point. This replaces the need to manually mark the point correspondences as in ([6]).

We also describe a stratification approach, where we first affinely calibrate the reference plane and then derive affine equations. Furthermore, we describe a shadow based calibration approach. Alternatively, we analyze the induced error if such a calibration is omitted.

The main benefit of the proposed approach over previously proposed shadow-based or other structured light systems is when the objects are far away from the camera (e.g., outdoor scenes). In these cases, depth variations are fairly small with respect to the distance to the camera. Therefore, depth maps are difficult to compute. In contrast, the recovered parallax map (height from the reference plane) using a P+P representation is not affected by the distance to the camera, and is therefore more accurate in these cases.

Moreover, the computation of classic depth maps is based on triangulation, i.e., on intersecting rays from the light source with rays from the camera. This constrains the light source to be in a
fixed and known position (typically computed in a preprocessing calibration step). On the other hand, the method proposed in this paper does not have constraints on the location of the light source and/or the obstruction. Therefore, it can use the sun (and its motion) as the source of cast shadow.

To illustrate that the P+P approach can be applied to indoor scenes as well, we describe two indoor capturing setups. The first one inspired by Bouget and Perona work [1, 3], describes a small desktop size setup that supports the parallax capture. The critical observation for this setup is that our formulation does not require that the casting object be in a fixed position - it can move as long as it remains vertical. The second setup combines a projector that replaces the cast shadows. Here we follow the similarity between methods that use cast shadows and methods that project light stripes ([9, 15] to list but a few recent publications). We describe how to design a structured light pattern and setup that will support parallax capture.

The paper is organized as follows:
2. Capturing Parallax

Before describing the setup and its associated geometry, we briefly describe a few of the notions that will be used in this paper. Assume a tall pole (or a building) casts a shadow on a plane with an object on it as illustrated in Fig. 1.(a). The shape of the shadow’s edge (transition from dark to bright) results from the intersection of a vertical plane with the object and ground plane. This plane is illustrated by transparent blue in Fig. 1.(a). This intersection defines two notions that will be used in this paper: "shadow curve" and "shadow line". Both are illustrated in Fig. 1.(b). The shadow curve is the trace of the vertical plane shadow on the object and on the ground. The shadow line is the line generated by the intersection of the shadow vertical plane with the ground plane. Note that the shadow line might not be seen at some points on the ground plane, yet it is still well defined.
Figure 3: The vertical mapping. This figure illustrates the vertical mapping and its properties, on a real scene. The left hand side (a) superimpose two shadows captured at different times. For each shadow we mark the shadow curves (red) and shadow lines (green). The line and curve intersection points are marked by $Q$ and $P$, respectively. Finally the vertical mapping $P \rightarrow Q$ is illustrated by an arrow. The center figure (b) illustrates why this is indeed a vertical mapping. It shows two poles. The shadows cast by the left pole must be on the blue vertical plane. The shadows cast by the right pole must be on the red vertical plane. Because both $P$ and $Q$ belong to both planes, they must be on their intersection. The intersection of two vertical planes is a vertical line denoted by a dark arrow. The right hand side (c) illustrates properties of the projection of the vertical mapping, that is - the arrows of the vertical mapping intersect at the vertical vanishing point (the projective projection of the infinity point below the scene). In this example the vertical vanishing point can be recovered by intersecting the lines of the table legs. Indeed all three lines (2D projections of 3D vertical lines) intersect at the same point. Finally, (c) also shows that if the vertical vanishing point is known, a single sweep suffices (i.e., we need only one pole). Simply because the intersection of the line connecting $p$ and vpp with the single shadow line defines the point $q$.

2.1. The Setup

The capturing "scenario" is illustrated in Figure 2. It illustrates (a and b) two tall buildings casting shadows on a small object lying on the ground plane. The goal is to model the small object (the cylinder). This figure illustrates a fairly common scenario in urban environments, where two buildings (or two corners of the same building) cast shadows on the same objects (at different times during the day).

In general, it suffices to assume that the poles (building contours) that cast the shadows are parallel and not necessarily orthogonal to the ground plane. However, in most cases they will be vertical, so we refer to them as vertical poles\(^1\). The ground plane need not be horizontal it can

\(^1\)If the poles are only parallel but not vertical, the "height" reconstruction will be along their joint direction.
be slanted. In this case the recovered height will be with respect to point below it.

Given 2 images captured at different times (e.g, 9am and 1pm), but using the same static camera, we have two shadow lines and two shadow curves. These are denoted by $l_{9am}$ and $l_{1pm}$ and by $c_{9am}$ and $c_{1pm}$, respectively, and are displayed in Fig 2 (c).

Given these 2 lines and 2 curves, we define a real intersection point (curve intersection) and a virtual intersection point (line intersection), both are displayed in Fig 2 (d). The real (curves) intersection point is denoted by $P$ and belongs to both shadow curves. The virtual (lines) intersection point is denoted by $Q$ and is defined by the intersection of the two shadow lines. Assuming that the lines are not parallel (the shadows are not from the same time of the day, nor exactly 12 hours apart) then $Q$ is well defined. This allows us to define the "vertical mapping". The vertical mapping is the mapping from $P$ to $Q$ (see Fig 2. (d)). The vertical mapping is a mapping from real intersection points (intersection of curves, possibly not on the plane) to the virtual lines intersection point on the plane. The $P$s - curve intersection are always visible, but the $Q$s (line intersections may sometimes be occluded - covered by the object. For example, all visible points on the plane are mapped by the vertical mapping onto themselves.

Shadow curves may intersect at more than one point. Shadow lines always intersect at a single intersection point. Such a many-to-one mapping represents the case, where multiple scene points are above, the same point on the ground plane (i.e, a vertical wall in the scene). Note, that the vertical mapping is well defined, as it is a mapping from scene points into projections of planar points. For applications that require inversion of this mapping, we will use the highest real (curves) intersection point.

A further illustration and explanation of vertical mapping and its properties are illustrated in Fig. 3. It explains why such a setup (two vertical poles) provides the desired output (i.e., a vertical mapping in 3D), the crucial observation is that the mapping is parallel to the vertical poles. This is because it is induced by an intersection of two vertical planes. Again, this result holds even if
the poles are just parallel and not vertical. Then the parallax mapping vectors will be parallel to both poles (but not vertical).

The above explanation focuses on a single scene point. To generate a dense mapping (i.e., a vertical mapping for every scene point), we use two sequences of shadow images each sweeping the scene/object once. The temporal length of such sequences depends on the distance of the object from the obscuring object (the time each shadow edge sweeps the object). In our outdoor experiment, each sweep was 15 min long, and they were taken about 3 hours apart.

The recovery of shadow lines and shadow curves from images is similar to the method proposed by Bouguet and Perona [1]. Some differences result from: (a) The long exposure time and translation of the source of light, particularly when using the sun as the light source. This, requires more robust recovery methods. (b) In our case, all shadow lines intersect at a single point (the point where the pole intersects the ground plane). These differences from method of [1] are briefly described in Sec 6.

So far, we have not considered the camera. If we image such a scene, we only have access to the projection of $P$ and $Q$ and projections of shadow curves and lines. The geometry of the projection of the vertical mapping into 2D images is discussed next.

3. Three Dimensional Reconstruction

Thus far, we have not considered the camera. Given a sequence of images of such a scene with cast shadows, we only have access to projections of $P$ and $Q$. In this section we discuss what can be computed about real 3D values (height), once the projection of the vertical mapping pairs of points are known. We describe several possible approaches: (i) A complete projective setup, where large projective distortion is taken into account. (ii) A stratification approach, where we first affinely calibrate the reference plane and then derive affine equations. (iii) Finally, we analyze the error in assuming an affine camera (i.e., apply affine equations without prerectifying the reference plane).
It is shown that in many cases these errors are likely to be small.

3.1. Projective Equations

Many previous theoretical results on plane + parallax may be used to analyze the geometry of the vertical mapping presented in this work (See for example [11, 12, 13, 16, 17], or [14] for recent results). They all show that the 3D height of a point from the reference plane can be determined up to global scale factor. The work which is most related to this paper is titled ”single view metrology” [6, 7]. There, the parallax is also vertical. However, pairs of points above each other are selected manually. Height of such a manually selected point from its corresponding point on the reference plane is computed using the cross ratio. The cross ratio is a projective invariance defined by 4 points (on a line). In this case these are: $q$ the point on the reference plane, $p$ a point above $q$ whose height we want to recover, the vertical vanishing point ($vvp$), and the $c$ intersection of the line defined by $p$ and $q$, and the line at infinity: $c = l_\infty \times \vec{pq}$. The resulting projective invariance (the cross ratio) is:

\[
\frac{[p,q]}{[q,c]} = \frac{[Q,P]}{[Q,C]},
\]

where $[,]$ indicates the distance between two points, and capital letters denote the corresponding 3D points. Setting $Q$ on the reference plane to be at height 0, rearranging terms, and cancelling $\infty/\infty$ we get:

\[
\text{height } P = \text{height } C \cdot \frac{[p,q]}{[q,c]}.
\]

Criminisi et al., [6] showed that the height of $C$ is the same for all scene points and equals the height of the camera from the reference plane. This is an explicit representation of the global scale factor that has to be assumed, known (e.g., a using an object with known height in the scene)
or guessed (e.g., manual tuning). Note that the use of the line at infinity may be viewed as an implicit calibration. The next section describes a stratification approach where the rectification of the reference plane is explicit.

### 3.2. Affine Equations

A stratification approach basically decomposes the reconstruction problem into two steps: (i) affine calibration/rectification of the reference plane, and (ii) reconstruction using affine equations. The affine calibration step is performed using the line at infinity \( l_\infty = [l_1, l_2, l_3]^T \) (Two methods of extracting such information from shadows are described in Section 4.1). Once the line at infinity \( l_\infty = [l_1, l_2, l_3]^T \) is known standard methods (see [10]) can be used to calibrate the scene. This section focuses on the second step - height recovery once the reference plane is affinely calibrated.

As we are only interested in height, we have chosen a coordinate system that simplifies the derived equations. We induce the rectified image affine coordinate system \((X, Y)\) to the ground plane. The \(Z\) axis is in the direction of the poles (i.e., vertical direction) with the ground plane being at \(Z = 0\). As a result, we get a simple form for the camera projection matrix.

Denote by \(p = (u, v, 1)\) and \(q = (u', v', 1)\) the images of a pair of scene points \(P = (X, Y, Z, 1)\) and \(Q = (X, Y, 0, 1)\), i.e., \((p \cong MP\) and \(q \cong MQ)\). Our special coordinate system implies that \(u = X\), and \(v = Y\), thus \(M\) the \(3 \times 4\) camera matrix has the following form:

\[
M = \begin{pmatrix}
1 & 0 & \alpha & 0 \\
0 & 1 & \beta & 0 \\
0 & 0 & \gamma & 1
\end{pmatrix}
\]  

(2)

The simple form results from the fact that using this coordinate system \(M\) preserves the \(X\) and \(Y\) values of points on the reference plane (with \(Z = 0\)).

Applying \(M\) on \(P\) and \(Q\) yields:

\[
u = \frac{X + \alpha Z}{1 + \gamma Z}, \quad v = \frac{Y + \beta Z}{1 + \gamma Z}
\]  

(3)
and

\[ u' = \frac{X + \alpha \cdot 0}{1 + \gamma \cdot 0} = X, \quad v' = \frac{Y + \beta \cdot 0}{1 + \gamma \cdot 0} = Y. \] (4)

Eliminating \( Z \) from 3 and substituting \( X \) and \( Y \) gives:

\[ u\beta + u' v\gamma - u' \beta - v\alpha - v' u\gamma + v' \alpha = 0, \] (5)
a linear homogeneous equation in \((\alpha, \beta, \gamma)\), so that 2 pairs of points are sufficient to compute \((\alpha, \beta, \gamma)\) up to scale. Once \( \alpha, \beta, \gamma \) are found, we can compute \( Z \):

\[ Z = \frac{v' - v}{v\gamma - \beta} = \frac{u' - u}{u\gamma - \alpha}. \] (6)

Again, the results are up to one global scale that has to be assumed, known or guessed. It is interesting to note that if we do not precalibrate the reference plane, Eq. 5 and 6 do not hold as the induced coordinate system is not even a projective coordinate system. This is why we call them affine equations.

### 3.2.1 Geometric illustration of the affine equations

Equation 6 can also be written as:

\[ \frac{\alpha/\gamma - u}{-1/\gamma} = \frac{u' - u}{Z}. \] (7)

Using this form provides the following simple geometric interpretation. Inducing the coordinate system of the image on the reference plane is equivalent to placing the camera image plane on the ground plane. See Fig. 4.(a). Now Eq. 7 is a similarity relation between two right angle triangles shown in Fig. 4.(b), with the camera center being at: \([-\alpha/\gamma, -\beta/\gamma, -1/\gamma]^T\). This "new" location corresponds to changing the origin to be at the reference plane (and not on the camera).
\[
\begin{align*}
\mathbf{t}_{\text{new}} &= \begin{pmatrix} 1 & 0 & \alpha \\ 0 & 1 & \beta \\ 0 & 0 & \gamma \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -\alpha/\gamma \\ -\beta/\gamma \\ -1/\gamma \end{pmatrix}.
\end{align*}
\]

To summarize, using a slightly different coordinate system (a camera matrix with camera center \((M[4] = t_{\text{new}}))\), reveals that the constraint of Eq 6 is a similarity between two right angle triangles.

### 3.3. Affine Equations without Affine Calibration

This section analyzes the following strategy: Apply the affine equation 5 and 6 directly. i.e., without affine calibration of the plane. Obviously the results are not accurate. However, we show here that in many cases the induced errors are small.

In the projective setup, the values of the line in infinities are folded into the reconstruction equations. To analyze their contribution, we will compare their values with and without affine calibration. After affine calibration, the cross ratio used in Eq. 1 becomes:

\[
\text{height } P = \text{height } C \cdot \frac{[p, q]}{[q, c]}.
\]

This is because the \(vvp\) is transformed to infinity by affine calibration, and thus the last term becomes one \([vpp, p]/[vpp, c] = 1\). Therefore, the absolute error that omitting the affine calibration introduces is \(\text{error} = [vpp, p]/[vpp, c]\). We, however, are only interested in the relative error, as our reconstruction is only up to unknown scale factor. Thus the error is:

\[
\text{relative error} < \max_{p_1,p_2} \frac{[vpp, p_1]}{[vpp, c_1]} \frac{[vpp, c_2]}{[vpp, p_2]}.
\]

In our experiments, this term is usually dominated by the location of \(vvp\) - the vertical vanishing point. Thus, ignoring it does not significantly affect the accuracy of the final reconstruction.
Figure 4: The induced coordinate system. (a) Illustrates the geometry of the induced coordinate system on the ground plane. This can simply be thought of as a homography mapping between two planes through the camera center. (b) Illustrates the geometrical interpretation of the equations for deriving $Z : \frac{\alpha/\gamma - u}{1/\gamma} = \frac{w - u}{Z}$. It is a simple ratio between 2 right angled triangles with joint edges: the first one results from the camera center ($t$), and the second is the scene point with height $\hat{Z}$.

It is interesting to note that there are applications that do not require any calibration. Chuang et al. [5] proposed a similar setup to the one proposed here. They also sweep the scene/object twice by a shadow, generated by translating the obstructing object (a pole) in front of the camera. Their algorithm generates a mapping from the reference plane to scene points, that is later used for insertion of a novel shadow.

Since they focus on image and shadow rendering in the specific light+camera setup, their method does not require calibration. This implies that the results are limited to the particular constellation of the camera and sun at the time of taking the input sequences. In comparison to their approach, our method reverses the roles of the pole and the source of light i.e., we move the sun and fix the pole (we require vertical poles). These changes allow us to apply their results to arbitrary light directions.

4. An Outdoor Example

This section illustrates the applicability of our method to address scenarios that are highly challenging for other methods. We downloaded 5 hours of data from a remote webcam\(^2\), at intervals of 5 seconds, on a sunny day. Representative images of the input data are shown in Fig. 5. The task

\(^2\)Data was taken from a camera at http://www.damstaete.nl/damsite. To the best of our knowledge the camera was temporarily removed.
Figure 5: **Outdoor example.** The first two rows display representative frames downloaded from the webcam in Amsterdam (from www.damstaete.nl/damsite). The shadows used in this example are from the pole in the center of the square (around 11:30 am), and from a tall building invisible in the frame (around 1:30 pm). We reconstruct the 3D structure of the the statue of a lion marked by a red circle in (a). The lion is mounted on a circular base, we see its back. A magnified view is displayed in (b). The reconstructed parallax is shown in (c). We can see the lion’s back and the circular shape of its base.
is to reconstruct the 3D structure of a statue of a lion. It is marked by a red circle in Fig. 5.(a) (the lion is facing the other direction). Fig. 5.(b) shows a magnified view. The algorithm was applied to two 15min sequences (around 11 : 30 am and 1 : 30 pm), when shadow edges were ”sweeping” the statue. Before, presenting the results, we would like to discuss the challenges of this scene. 1) Only a single webcam is used. 2) The shallow viewing angle (i.e., nonorthogonal imaging condition). 3) The object is textureless. 4) The distance from the camera (about 100m). 5) The poor input resolution, object dimensions are 60 by 80 pixels.

The reconstructed height map is shown in Fig. 5.(c). One can clearly see the lion’s back and the circular shape of its base.

4.1. Shadow-based Calibration

Complete Euclidian calibration of the reference plane requires 4 known points. Such calibration patterns are typically available in an indoor setup, where it can be drawn on the reference plane. However for outdoors (where a calibration pattern may not be available), we propose exploiting sun-based cues.

1) **Identifying parallel lines**: If in some frames two (or more) shadows induced by different vertical poles are visible simultaneously, their corresponding shadow lines on the reference plane are parallel. The intersection of parallel lines provides a point on the line at infinity \( l_\infty \). Given 2 pairs (or more) of parallel lines, it is possible to compute \( l_\infty \). Once \( l_\infty = [l_1, l_2, l_3]^T \) is known, it can be used directly (Sec. 3.1). Alternatively, we can calibrate the reference plane using the following homography \( H_{affine\ calibration} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{bmatrix} \) (see [10]). Then we can use the affine equations of of Sec. 3.2.

2) **Identifying right angles**: The time the images are captured (at a resolution of minutes) can be used to retrieve angles between lines on the ground plane (actually this is how a sundial
Figure 6: **Shadow-based affine calibration.** Figures (a) and (b) display pairs of parallel shadow lines (red lines). Their intersection points are on the line at infinity \((l_\infty)\). Thus it can be retrieved (c).

works). For example, two shadows captured exactly 6 hours apart produce a right angle (Earth circles the sun every 24 hours). Five such angles are sufficient for Euclidean calibration of the reference plane. See [10] for details.

### 5. Indoor Systems

This section describes a couple of indoor setups for accurate capture of parallax. Using two examples, light stripe-based systems and shadow-based systems, we illustrate how previously proposed methods that compute depth maps can be modified to compute parallax.

#### 5.1. Light Stripes - Lasers and Projectors

Many light-stripe approaches have been proposed in the past. In these systems "shadow edges" and "shadow curves" are replaced by the trace of a projected light stripe (commonly light stripe). Geometrically, shadows and light stripes are equivalent. Both use a line projected on the plane and on the object. Using two (or more) points on the reference plane and pre-computed/calibrated information about the light source location, the light/shadow 3D plane can be computed. Now, a point’s depth is recovered by intersecting a ray originated at the camera center (passing through the retrieved point) with the shadow/light 3D plane. See [2] for detailed derivation when using shadows, or [19] for a system that use a laser to generate a light stripe, and share the same geometry. Representative examples of projector based systems, that use the same geometry may
be found in [8, 18, 15]. The benefit that a projector provides over a laser or shadows is that multiple stripes can be used to speed up the process. Typically, a set of vertical (or horizontal) stripes is used. Different stripe configurations are used to reduce the ambiguity that simultaneous lines/stripes generate.

To apply our parallax approach to projected stripes we need to enforce ”verticality” of the planes defined by the light stripes. This means ensuring that all 3D planes generated by the laser/projector are vertical to the reference plane. Denote by $O$ the projector center, and by $o$ its closest point on the reference plane $\pi$, $o \in \pi$ is the vertical projection $O$ on the reference plane $\pi$). A vertical plane originating from point $O$ must pass through point $o \in \pi$. Thus, we will use a pencil of light planes. The common 3D line for all planes in the pencil is the line $\vec{O}o$. It is the shortest distance from the projector to the reference plane. Therefore, a ”star-like” pattern should be projected. It replaces the series of parallel lines that are typically used. The center of the star projects to the point $o$.

Note, that all other useful suggestions to speed up or/and simplify the capturing process, such as using Gray patterns or color (for example [4] [9] ) can be adopted here.

5.2. Parallax on Your Desktop

Inspired by Bougout and Perona, [1, 2, 3] system, we presents a parallax capturing system. Both systems consist of a static lamp and a static video camera opposite a plane (a desktop) with an object on it. The user sweeps the object with a shadow of a narrow moving object (e.g. a pencil). While Bougout and Perona moved the pencil freely (i.e., the pencil could have arbitrary orientation), we slide it carefully and maintain the verticality of the obstructing object (”the pencil”). Fig 7 displays our system. The human hand holds the platform that maintains the pole verticality. To combine two different light locations, we used two lamps, both are marked by red circles.
It should be appreciated that only the "verticality" of the poles is required for capturing parallax. Therefore, both the pole and/or the light source may move or change. Moving the pole and not the light source simplifies the shadow edge extraction, and thus contributes to reconstruction accuracy. Reconstruction results using this system are displayed in Fig. 8.

5.2.1 Accuracy of the indoor setup

Fig. 8(a) and (b) present two representative images when the left light source or the right light source were used, respectively. (c) and (d) display the reconstructed height map, from front and side views. Note, that the slanted front face of the object was recovered accurately. To measure the accuracy of the reconstruction results, we compare them to manually measured dimensions of the reconstructed object. The object was a piece of wood with a trapezoid profile. Its wide base is 6.8 cm, and its narrow base is 3.65 cm. One edge has a right angle with a height of 3.2 cm. The overall width of the piece is 16.6 cm. To measure the errors, we used the planer phases of the object. Since our reconstruction is only up to a scale factor, we match a plane to the phases and compute RMS error with respect to that plane. The error was less than 1%, which translates to less than 0.16 mm.

Summary: Unlike outdoor scenarios, where the advantage of the proposed approach is clear, i.e, other methods cannot be applied, for indoors we do not argue that our approach is always preferable. The comparison with depth map approaches truly depends on the camera and reference plane configuration. If the camera principal axis is perpendicular to reference plane, our method cannot be applied (this is a singular point, where the vertical mapping cannot be measured). However, if the camera principal axis tends to be parallel to the reference plane, and the object is far away from the camera, measuring height is more reliable then measuring depth from the camera.
Figure 7: **Parallax on your desk - the setup**. This figure illustrates the setup used for capturing sequences with a moving pole. The red circles mark the two lamps used in this setup (in this frame the right one is in use). The green ellipse marks the camera. The hand in the center is sliding the vertical brown pole. The task is to recover the shape of the orange object, placed on the white table.

Figure 8: **Parallax on your desktop results**. This figure presents results obtained by the setup described in Fig. 7. (a) and (b) are representative input images when different lamps were used. (c) and (d) represent the reconstructed parallax. Note that the slanted front face of the object was recovered accurately.

## 6 Extracting Shadow Lines and Curves

Our shadow extraction algorithm is based on temporal analysis as proposed in Bouguet and Perona [2]. Namely: (i) For each pixel, find its minimal and maximal value along time. (ii) The time a shadow edge passes a particular pixel is when its intensity value equals the average between min and max values. This simplified approach suffices when we moved the pole e.g., Fig. 8. Some modifications were needed when light source was moved. This is because the min and max values of a given along time are not well defined. Thus additional cues have to be involved. First we used all planar points to estimate shadow lines (and not just two points as in [2]). Second we enforce a common intersection point for all retrieved lines. Lines that did not pass trough the common intersection point are pruned out. Finally, hole filling (interpolation) is applied where
Figure 9: **Moving the light source** The top row (a.) and (b.) display three representative frames from the first and second input sequences (images when shadows are cast by the left and right pole, respectively). The sequences were captured while turning a lamp around the room, with light facing the objects. The second row displays the recovered height map. The dark diagonal lines in the reconstructed output correspond to shadow lines that did not intersect the joint intersection points (the table legs).

Furthermore, for the outdoor setup we also enforce smooth angular speed of shadow motion (the sun moves in a constant angular speed). Given the pole location at the image (previous paragraph), we transform the input images (only a region of interest) to a polar coordinate system with origin at the pole. In this coordinate system (angle, radius, time) shadows move vertically. For each row of resulting 3D semi-polar volume (an $xt$ cut of this volume) was analyzed using dynamic programming to find a smooth shadow motion. The dynamic programming cost function was the distance from average of min and max temporal values. The results where warped back to the image coordinate system.

Finally, for the outdoor example, outliers resulting from the presence of people (people climbed on the statue during our experiment) were manually detected, and values in occluded areas were interpolated.
Conclusions

We presented an approach for capturing parallax, (height from the ground plane) from moving shadows cast by vertical obstructors. The main benefits of the presented method are for outdoor scenes, when the object is far from the camera e.g., the webcam example in Fig. 5. Previously proposed shadow-based (or structured light) approaches are impractical in these cases. Since our method (i) recovers height and not depth, and (ii) does not constrain the location of light source, it can use the sun as the cause of shadow and is less affected by the distance to the camera. It is intersecting to note that methods that are based on inputs from multiple cameras (e.g., stereo) will also face difficulties in such outdoor examples. The ratio between depth variations and distance from the camera/s will require an impractical base line between the cameras. Therefore, for such a scenario, the proposed method subsumes previously published approaches.

References


