Three-Dimensional Structure
Extraction of a Projecting Object from its Shadows

Ares Ribó Mor
Facultat de Matemàtiques i Estadística
Universitat Politècnica de Catalunya

Advisor: Jordi Saludes Closa
Computer Vision Center
Universitat Autònoma de Catalunya

1 Introduction

In this technological project we study the viability of reconstructing the three-dimensional geometry of the projecting elements from a façade using their shadows projected on the wall at two different time instants.

This problem arose from an initiative of the City Council of Barcelona in order to prevent loosenings from the façades of several old buildings of the city, which were frequent at the end of the nineties.

Another problem studied in stereo vision is the recovering of the three-dimensional coordinates of an object given two images obtained by cameras of known relative positions and orientations.

The advantage of our method is that we do not need to know either the position of the cameras or their relative positions. They can be positioned everywhere, and hence we have a wide range of movement, having the possibility of taking the photographs far away from the building. The only required information are the date and the time at which the two photographs have been taken and the declination of the wall.

We obtain the reconstruction of the three-dimensional object up to a homothecy, since the distance between the façade plane and the retinal plane is unknown.

We also have studied several segmentation algorithms for the extraction of the shadow projected on the wall.

Our algorithms are implemented in MATLAB and we have built a MATLAB-based GUI (Graphical User Interface).

2 Collineations of the projective plane

Any system of lenses can be approximated by a system that performs a perspective projection of the world onto a plane. We look at such a system projectively. It allows us to locate the two images of an object taken from two unknown different places in the same position in the photograph.

For this purpose we have implemented a function that, given two photographs of the same plane, maps one image to the other.
2.1 Projective spaces

Let be $\mathbb{P}^n$ the n dimensional projective space, associated to a $n + 1$ dimensional vector space $E$.

Remember that a projective basis $\mathcal{R} = [P_0, \ldots, P_n; U]$ of $\mathbb{P}^n$ is a set of $n + 2$ points $P_0, \ldots, P_n, U$ such that no $n + 1$ of them are linearly dependent. For example, the set $e_i = [0, \ldots, 1, \ldots, 0]^T$, $i = 0, \ldots, n + 1$, where 1 is in the $i$th position, and $e_{n+2} = [1, \ldots, 1]^T$, is a projective basis, called the standard projective basis.

Any point $x$ of $\mathbb{P}^n$ can be described as a linear combination of the $n + 1$ points of the basis, i.e. $x = \sum_{i=0}^{n+1} x_i e_i$. The point $x$ is represented by a non zero $n+1$ vector of projective coordinates $x = [x_1, \ldots, x_{n+1}]^T$. Two $n+1$ vectors $[x_1, \ldots, x_{n+1}]^T$ and $[y_1, \ldots, y_{n+1}]^T$ represent the same point if and only if there exists a nonzero scalar $\lambda$ such that $x_i = \lambda y_i$ for $1 \leq i \leq n + 1$. Therefore, the correspondence between points and coordinate vectors is not one to one.

A linear transformation or collineation from $\mathbb{P}^n$ into $\mathbb{P}^m$ is defined by a nonsingular $(n + 1) \times (n + 1)$ matrix $A$ up to a nonzero scalar factor.

Fundamental Theorem of Projective Geometry. Let be $\mathcal{R}$ and $\mathcal{R}'$ two projective bases of $\mathbb{P}$ and $\mathbb{P}'$ respectively. Then exists a unique collineation $\varphi: \mathbb{P} \rightarrow \mathbb{P}'$ such that $\varphi(\mathcal{R}) = \mathcal{R}'$.

In particular, a linear transformation from $\mathbb{P}^2$ into itself is uniquely defined by two 4-sets of points in general position, $\mathcal{R} = [P_0, P_1, P_2; U]$ and $\mathcal{R}' = [P'_0, P'_1, P'_2; U']$, represented by the coordinate vectors $x_i$ and $y_i$, $i = 1, \ldots, 4$. The proof of the following proposition give us the matrix that defines this collineation.

Proposition. If $x_1, x_2, x_3, x_4$ and $y_1, y_2, y_3, y_4$ are two 4-sets of coordinate vectors such that in either set no three vectors are linearly dependent, i.e. form two projective bases, then there exists a nonsingular $3 \times 3$ matrix $P$ such that $Px_i = \rho_i y_i$, $i = 1, \ldots, 4$, where the $\rho_i$ are scalars, and the matrix $P$ is uniquely determined apart from a scalar factor.

Constructive proof. Let be $e_i$, $i = 1, \ldots, 4$, the standard projective basis of $\mathbb{P}^2$.

Then there exist nonsingular matrices $A$ such that $Ae_i = \lambda_i x_i$, $i = 1, \ldots, 4$, where the $\lambda_i$ are non zero scalars; any two matrices with this property differ at most by a scalar factor.

To proof this, note that the matrix $A$ satisfies the three conditions

$$Ae_i = \lambda_i x_i \quad i = 1, 2, 3$$

if and only if it can be written $A = [\lambda_1 x_1, \lambda_2 x_2, \lambda_3 x_3]$. We must show that we can choose the values of $\lambda_i$, $i = 1, \ldots, 4$ in such a way that the equation $Ae_4 = \lambda_4 x_4$ is also satisfied. But this is equivalent to

$$[x_1, x_2, x_3] \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \lambda_4 x_4$$

By the hypothesis concerning the linear independence of the vectors $x_i$, the matrix $[x_1, x_2, x_3]$ is of rank 3. Thus the ratios of the $\lambda_i$ are uniquely determined.
and, furthermore non of the \( \lambda_i \) is zero. The matrix \( \mathbf{A} \) is uniquely determined up to a scalar factor and is clearly nonsingular. Assuming \( \lambda_1 = 1 \) the values for \( \lambda_1, \lambda_2, \lambda_3 \) are uniquely determined, since they are the solution of a \( 3 \times 3 \) system of linear equations.

Therefore, we can choose a nonsingular matrix \( \mathbf{A} \) and four nonzero scalars \( \lambda_1, \lambda_2, \lambda_3, \lambda_4 \) such that

\[
\mathbf{A} \mathbf{e}_i = \lambda_i \mathbf{x}_i \quad i = 1, \ldots, 4
\]

Similarly we can choose \( \mathbf{B} \) and \( \mu_1, \mu_2, \mu_3, \mu_4 \) so that

\[
\mathbf{B} \mathbf{e}_i = \lambda_i \mathbf{y}_i \quad i = 1, \ldots, 4
\]

Then

\[
\mathbf{BA}^{-1} \mathbf{x}_i = \frac{\mu_i}{\lambda_i} \mathbf{y}_i \quad i = 1, \ldots, 4
\]

and taking \( \mathbf{P} = \mathbf{BA}^{-1} \) and \( \rho_i = \frac{\mu_i}{\lambda_i} \) the existence is proved. To prove the uniqueness, if \( \mathbf{P} \mathbf{x}_i = \rho_i \mathbf{y}_i \) and \( \mathbf{Q} \mathbf{x}_i = \sigma_i \mathbf{y}_i \), then \( \mathbf{PA} \mathbf{e}_i = \lambda_i \rho_i \mathbf{y}_i \) and \( \mathbf{Q} \mathbf{A} \mathbf{e}_i = \mu_i \rho_i \mathbf{y}_i \), and hence \( \mathbf{PA} = \tau \mathbf{QA} \), i.e., \( \mathbf{P} = \tau \mathbf{Q} \) for some \( \tau \in \mathbb{R} \setminus \{0\} \). \( \square \)

2.2 The algorithm that maps the first image to the second

We select four points in each of the two images \( I_1 \) and \( I_2 \) and we think of them as projective points of \( \mathbb{P}^2 \). Let be \( \mathbf{x}_i \) and \( \mathbf{y}_i \), \( i = 1, \ldots, 4 \), the coordinate vectors representing these four pairs of points.

Following the steps of the previous proof we construct the matrix \( \mathbf{P} \) associated to the collineation.

Let be \( f \) the function calculated from \( \mathbf{P} \) which maps \( I_1 \) to \( I_2 \). Let be \( (i_1, j_1) \) a pixel of \( I_1 \). Note that every pixel \( (i, j) \) of an image satisfies \( i, j \in \mathbb{Z} \). The problem is that, in general, \( f(i_1, j_1) \) does not have integer coordinates. For practical purposes what we do is for any pixel \( (i_2, j_2) \in I_2 \) we compute its corresponding point in \( I_1 \), and we interpolate the value of the image in \( f^{-1}(i_2, j_2) \) using nearest neighbour interpolation. We do not work with pixels \( (i_2, j_2) \in I_2 \) such that \( f^{-1}(i_2, j_2) \notin I_1 \). We set their value to zero; this is the reason why sometimes some black strips can be appreciated at the edges of the transformed image.

Figures 1 and 2 show two photographs of the same balcony. Figure 3 shows the second image mapped to the first.

Thanks to this function we can take the photographs of the façade from every place.

3 Solar coordinates

We need to know the solar coordinates, i.e. the direction in which the shadow is cast by our projecting object on the wall.

An accuracy of 0.01 degree is sufficient, according to the precision of our data. Hence, we calculate the position of the Sun by assuming a purely elliptical motion of the Earth; that is, we neglect the perturbations by the Moon and the planets.
Figure 1: Lleida, September 18th, 1999, at 16° 48’.

Figure 2: Lleida, September 28th, 1999, at 19° 15’.

3.1 Systems of astronomical coordinates

In astronomy, a star position is determined by spherical coordinates. Several systems of astronomical coordinates take part in our algorithm to compute the solar coordinates. The algorithms return the horizontal coordinates of the Sun; they are the azimuth $A$ (measured westward from the South) and the altitude $h$ (positive above the horizon, negative below).
3.2 Algorithm to compute the horizontal coordinates

This algorithm, extracted from the book [Mee91], calculates the azimuth and the apparent altitude of the Sun given a day, an hour and the geographic coordinates of the place.

The principal steps of the method are the following:

**Step 1** Conversion of the date and the time into the corresponding Julian (Ephemeris) Day.

**Step 2** Calculus of the time $T$, measured in Julian centuries of $36525$ ephemeris days from the epoch J2000.0

**Step 3** Ecliptical coordinates: longitude $\lambda$ and latitude $\beta$

(a) Geometric mean longitude of the Sun $L_0$

(b) Mean anomaly of the Sun $M$

(c) Equation of the Sun of the centre $C$

(d) True geometric longitude $\Theta$ of the Sun referred to the mean equinox of the date.

(e) Longitude of the ascending node of the mean orbit of the Moon on the ecliptic, $\Omega$, measured from the mean equinox of the date.

(f) Apparent longitude $\lambda$ of the Sun referred to the true equinox of the date ($\Theta$ corrected for the nutation and the aberration)

(g) Assume $\beta=0$

(h) We obtain $\lambda$ (measured from the vernal equinox along the ecliptic) and $\beta$ (positive if north of the ecliptic, negative if south).
Step 4 Equatorial coordinates: right ascension \( \alpha \) and declination \( \delta \)

(a) Mean obliquity of the ecliptic \( \varepsilon_0 \)
(b) True obliquity of the ecliptic \( \varepsilon \) (\( \varepsilon_0 \) corrected for the nutation)
(c) We obtain of \( \alpha \) and \( \delta \)

Step 5 Hour coordinates: local hour angle \( H \) and declination \( \delta \)

(a) Local sidereal time \( \theta \)
(b) Sidereal time at Greenwich \( \theta_0 \)
(c) We obtain of \( H \) (measured westwards from the South)

Step 6 Horizontal coordinates: azimuth \( A \) and (true) altitude \( h \).

Step 7 Effect of the atmospheric refraction. We obtain of the apparent altitude of the Sun \( h_0 \)

4 Three-dimensional reconstruction of one point given two projections on the façade

Let be \( M \) a point of unknown coordinates. Let be \( m_1 \) and \( m_2 \) two known points which are the shadow cast by \( M \) on the façade at two different time instants \( t_1 \) and \( t_2 \).

We search the spatial coordinates of \( M \), the recovered three-dimensional point, given the two-dimensional coordinates of \( m_1 \) and \( m_2 \).

Assume that the camera is parallel to the plane \( \pi \) of the wall, which is totally vertical respect to the ground.

Suppose that we have applied the collineation function and hence we have the point \( M \) located in the same position in the two images. Let us also suppose that the directions of the Sun at \( t_1 \) and \( t_2 \) have been computed, by using the algorithm described in the previous section.

4.1 A camera model. Affine study of our geometric system

We fix the world coordinate system \([O, X, Y, Z]\) for the three-dimensional space and the two-dimensional coordinate system \([O, u, v]\) for the retinal plane as indicated in Figure 4. The origin points of the two reference coordinate systems are placed in the same point of the wall. The axes \( X \) and \( Y \) point East and North respectively, while \( Z \) points to the zenith.

We abuse of notation and identify the plane \( \pi \) of the façade and the image plane (i.e. the retinal plane), since they are parallel planes; we use the same coordinate system \([O, u, v]\) in both.

The plane \( \pi \) is defined by the equation \( Y = X \tan \alpha \), where \( \alpha \) is the angle between \( \pi \) and the \( X \) axis.
4.1.1 To measure the declination of the façade $\alpha$

Since the magnetic North does not coincide with the true astronomical North due to the magnetic declination, we cannot measure the angle $\alpha$ with a simple magnetic compass. We use two methods to find $\alpha$:

Method 1: Cartographic data

Method 2: Astronomical data

The declination of a wall $\pi$ is the number of degrees East or West of South made by a perpendicular from the surface of the wall.

To find the declination of $\pi$ at any time of the day, we need to determine the direction $b$ of the Sun from $\pi$ and the azimuth $A$, which we already know.

To determine $b$ we use a simple instrument constructed as shown in the Figure 5. Tack a piece of paper to a flat board and drive a nail $EF$ into the board perpendicular to the face with about 3cm left above the surface. Draw a straight line $EW$ through the base of the nail. Hang a weight by a thin thread at the base of nail $E$. To use it, place the board against the wall and rotate till the plump line coincides with the line $EW$. Mark the end of the shadow cast by the nail, $C$, and note the time for later use. Measure the perpendicular distance $CD$ to the vertical line $EW$. Then the angle of the Sun from $\pi$ is $b = \arctan \frac{CD}{EF}$.

Compute the azimuth $A$ for the registered time.

The algorithm to compute the declination $d$ is the following:

1. If the observation was before local apparent noon:

   (a) If the Sun was to the right of the wall:
2. If the observation was after local apparent noon:

   (a) If the Sun was to the left of the wall:
      i. \( d = b + A \) to East
      (b) If the Sun was to the right of the wall:
         i. \( b > A, d = b - A \) to East
         ii. \( b < A, d = A - b \) to East

Then \( \alpha = d - 90^\circ \) if \( \pi \) declines East or \( \alpha = 270^\circ - d \) if \( \pi \) declines West.

4.1.2 Geometric study of our system

We can see a camera as a system which operates as a linear transformation from \( \mathbb{R}^3 \) to \( \mathbb{R}^2 \).

Let be \( q = (q_x, q_y, q_z) \) the coordinates of \( M \) referred to the world coordinate system \([O, X, Y, Z]\) and \( p_1 = (u_1, v_1) \) \( p_2 = (u_2, v_2) \) the coordinates referred to \([O, u, v]\) of the points \( m_1 \) and \( m_2 \) of shadow cast by \( M \) onto the image plane.

From each of the two images we can deduce two linear equations in \( q \) which can be written in matrix form as

\[
A_i q = p_i \quad i = 1, 2
\]
where \( A_i \) are \( 2 \times 3 \) matrices which depend on the position of the Sun at the moment that each photograph has been taken. This is, \( A_i \) are two linear transformations from \( \mathbb{R}^3 \) to \( \mathbb{R}^2 \).

Considering these to linear systems simultaneously we obtain a \( 4 \times 3 \) system of linear equations

\[
A q = p
\]

where

\[
A = \begin{bmatrix}
A_1 \\
- & - \\
A_2
\end{bmatrix}
\quad p = \begin{bmatrix}
p_1 \\
- \\
p_2
\end{bmatrix}
\]

**Explicit computation of \( A \)**

Let be \( a_1 \) and \( a_2 \) the position vectors of the Sun at time instants \( t_1 \) and \( t_2 \). We find \( a_1 \) and \( a_2 \) computing the horizontal coordinates of the Sun (azimuth \( A \) and the apparent altitude \( h_0 \)) by using the method described in the third section and transforming them into our rectangular coordinate system \([O, X, Y, Z]\), by using the following formulae:

\[
a(1) = \cos(270 - A) \cos(h) \\
a(2) = \sin(270 - A) \cos(h) \\
a(3) = \sin(h)
\]

The parametric equations of the lines passing through \( M \) with direction vectors \( a_1 \) and \( a_2 \) are

\[
r_i : q + a_i t_i = 0 \quad t_i \in \mathbb{R}, \quad i = 1, 2
\]

The spatial coordinates of \( m_i \) are given by the intersections \( r_i \cap \pi, \ i = 1, 2; \) these intersections are given by

\[
q_y + a_i(2)t_i = (q_x + a_i(1)t_i) \tan \alpha
\]

and hence

\[
t_i = \frac{q_x \tan \alpha - q_y}{a_i(2) - a_i(1) \tan \alpha}
\]

Then, replacing \( t_i \) in the equation of \( r_i \) we obtain the three-dimensional coordinates of \( m_i, \ i = 1, 2. \)

We can rewrite these equations in matrix form, calling \( B_i \) to the \( 3 \times 3 \) matrix which transforms linearly \( q \) into the three-dimensional coordinates \( m_i \). Let be

\[
B = \begin{bmatrix}
B_1 \\
- & - \\
B_2
\end{bmatrix}
\]

Let be \( P \in \pi \) a point of spatial coordinates \((p_x, p_y, p_z)\) referred to \([X, Y, Z]\) and retinal coordinates \((p_u, p_v)\). We have

\[
\begin{bmatrix}
\cos \alpha & \sin \alpha & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
p_x \\
p_y \\
p_z
\end{bmatrix} = \begin{bmatrix}
p_u \\
p_v
\end{bmatrix}
\]

Hence, \( A = RB \) is the \( 4 \times 3 \) matrix such that

\[
A q = p
\]

9
where
\[
R = \begin{bmatrix}
\cos \alpha & \sin \alpha & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & \cos \alpha & \sin \alpha & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

4.2 Solving the overdetermined system of linear equations

We solve the overdetermined linear system
\[
Aq = p
\]
by using the method of least squares.

It consists of finding \( x \) which minimizes the amount \( ||Ax - p||^2 \), that is
\[
q = \min_x ||Ax - p||^2
\]

The method of least squares gives us
\[
q = (A^T A)^{-1} A^T p
\]
which is the three-dimensional reconstruction of the point \( M \).

To estimate the statistical error on a set of \( N \) approaches of \( q \) we measure the RMS (root-mean-square) error.

5 Shadow segmentation

We want to determine a region of shadow \( O \) which is of our interest. In most of our cases, this region of shadow has almost a polygonal shape. We have implemented with MATLAB several methods in order to estimate the shape \( O \).

5.1 Manual segmentation

Manual input of the vertices of \( O \). \( O \) is the region inside the polygonal line which connect these points.

5.2 Automatic segmentation

We have implemented in MATLAB two different algorithms to determine \( O \) automatically.

First of all we convert our \( RGB \) image into its equivalent \( HSV \) image.

5.2.1 Method 1: Flooding

Given an \( HSV \) image \( I \) and a starting pixel of \( O \), the algorithm fills the region \( O \) by a flooding, following an established distance criterion based in the value and the hue of the pixels. In the flooding we store the non-visited neighbours classified as belonging to \( O \) in a queue. The algorithm finishes when the queue is empty.

Notice that the flooding does not proceed diagonally, not only for velocity reasons, but also because the four-neighbour flooding is more robust.

The disadvantage of this method is its high computational cost for big images.
5.2.2 Method 2: Classification pixel by pixel

Given an HSV image $I$ and a set $P$ of pixels manually selected, the method consist of analyzing every pixel of $I$ and deciding if their belong to the shadow region or not, according to some predetermined characteristics of $O$.

$P$ is a set of pixels of interest which we compare with the rest of the pixels of $I$. We split the elements of $P$ into two groups: the points that are interior to $O$ and the ones that are exterior.

We classify every pixel of $I$ as inner or external to $O$ according to the label of the nearest element of $P$, following a distance criterion previously established, and we obtain a binary image.

Finally we select the connected region of shadow, $O$.

The complexity of this method is linear in the number of pixels of the image.

5.2.3 Morphological operators

Once we have applied one of the two previous methods, we have the segmentation of the connected region of shadow $O$ in a binary image. Now we can improve our results by using mathematical morphology.

We have tested many morphological operators (as dilation, erosion, openings and closings) using different structuring elements (squares, bars with several orientations, ...).

We have implemented with MATLAB a function which applies a sequence of basic morphological operations. It is very difficult to establish a general criterion which works in all cases. Our GUI also allows us to choose the morphological operators and the structuring elements appropriate in each case.
6 Three-dimensional reconstruction of the whole projecting object

Once we have the segmentation of the regions of shadow $O_1$ and $O_2$ cast by the projecting object at $t_1$ and $t_2$, we can compute the three-dimensional reconstruction in two different ways.
6.1 Method 1: Reconstruction from the vertices

If $O_1$ and $O_2$ are segmented manually, we compute the three-dimensional reconstruction of the chosen pairs of vertices and connect them in the 3-D space, obtaining the perimeter of the projecting object.

![Figure 9: Reconstruction from the vertices.](image)

If the shadows $O_1$ and $O_2$ were segmented automatically we could extract the vertices and compute the 3-D reconstruction of these pairs of vertices.

6.2 Method 2: Reconstruction by sweeping the shadows

The projecting object can be seen as a curve in $\mathbb{R}^3$ with origin and ending at the plane $\pi$. Let be $c_i$ the perimeter curve which enclose $O_i$ at the instant $t_i$, $i = 1, 2$.

The following equation is satisfied:

$$m_1 + t_1a_1 = m_2 + t_2a_2 = M$$

Then

$$m_1 - m_2 = -t_1a_1 + t_2a_2 \in \langle a_1, a_2 \rangle$$

Let be $r$ the intersection line of $\pi$ and the plane $\langle a_1, a_2 \rangle$. We call $r$ the sweeping line.

$$m_1 - m_2 \in \pi \cap \langle a_1, a_2 \rangle = r$$

This algorithm consists of sweeping pixel by pixel the curves $c_1$ and $c_2$ according to $r$, identifying pairs of points. We obtain the whole three-dimensional curve reconstructing every pair of points.

13
7 Real world examples and applications

7.1 Validation and examples

In order to examine the correctness of our calculations, we have constructed a model of known sizes.

The proportion between the edges of the projecting object in the real model is 1.9636, while the proportion between the edges of our three-dimensional reconstruction varies between 2.0698 and 2.3674. In our reconstruction, the angles between the four edges of the projecting object are 88°7427, 89°4737, 87°0153 and 91°6944.

We think that the reconstruction is good. There is a small error due to our possible lack of precision when we select the pixels and to the RMS error of the least squares solution of the overdetermined linear system of equations.

7.2 Applications

Our reconstruction allows the extraction of some three-dimensional properties of a projecting object from a façade, as its shape and proportions.

If the projecting object is a balcony, the knowledge of these properties can help us to predict its condition. The study of its state could prevent some accidents.

As an example, the reconstruction of the balcony of Figures 11 and 12 is shown in Figure 13. We can clearly appreciate that it can be a public danger if it is not repaired.
8 Conclusions and remarks

8.1 Analysis results

Our 3-D reconstruction needs less information than other stereo vision methods. That is one of the reasons why it is a very flexible and versatile method. Its difficulty is that it cannot be applied for all cases. We depend on the façade of every building. For example, in some cases the bricks of the wall, some
window or a door deform the shape of the shadow and the reconstruction would not be the best.

About shadow segmentation, we think that in this moment the manual method is more practical, although recently we got very interesting results by using automatic segmentation.

8.2 Future work

Recently we have improved the results of both methods of automatic segmentation. Actually we are able to extract satisfactorily the shadow region in most cases.

Now we could investigate the improvement of the results when we work with more accurate data, measured by using a higher quality optical system and photographic negatives of high resolution. For example, with better images the effect of the half-light could also be studied.

Nowadays, the method exposed in this technological project maybe is not still a very practical method to evaluate the state of a façade, but we think that in a future it can become to be a good one.