CSE 252B: Computer Vision II

Lecturer: Serge Belongie
Scribe: Jayson Smith

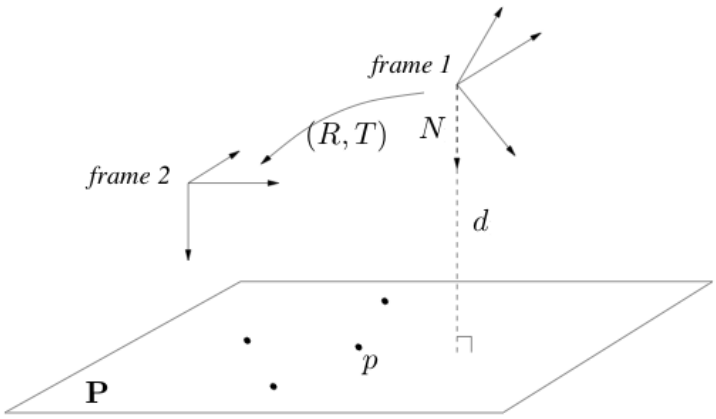
LECTURE 4
Planar Scenes and Homography

4.1. Points on Planes

This lecture examines the special case of planar scenes. When talking about the 8-point algorithm in two views, we saw that there are configurations of points lying on **critical surfaces** that cause the algorithm to fail. One example is when all points lie on a plane. We need an alternative algorithm to handle this important special case.

[1]Department of Computer Science and Engineering, University of California, San Diego.

April 7, 2004



The above figure (from an early draft of MaSKS) is a special case of the two-view configuration of points on a plane:

$$\boldsymbol{X}_2 = R\boldsymbol{X}_1 + \boldsymbol{T}$$

In this case, the vector normal from camera 1 is a perpendicular distance d from the plane P:

$$\boldsymbol{N}^\top \boldsymbol{X}_1 = n_1 X + n_2 Y + n_3 Z = d$$

or,

$$\frac{1}{d}\boldsymbol{N}^\top \boldsymbol{X}_1 = 1 \quad \forall\, \boldsymbol{X}_1 \in P$$

4.2. Homography

Using the definition of the normal vector above, and multiplying and dividing by d (which equals $\boldsymbol{N}^\top \boldsymbol{X}_1$), the expression for the transformation becomes

$$\boldsymbol{X}_2 = R\boldsymbol{X}_1 + \boldsymbol{T}\frac{1}{d}\boldsymbol{N}^\top \boldsymbol{X}_1 = H\boldsymbol{X}_1$$

where

$$H = R + \frac{1}{d}\mathbf{T}\mathbf{N}^\top, \quad H \in \mathbb{R}^{3\times 3}$$

H is known as the **planar homography matrix**. Previously, we had only the epipolar constraint which mapped a point in one view to a line in the other view:

$$\boldsymbol{x}_2^\top E \boldsymbol{x}_1 = \boldsymbol{x}_2^\top \widehat{T} R \boldsymbol{x}_1 = 0$$

A homography is stronger: it is a point-to-point mapping from one view to another.

4.3. From 3D to 2D Coordinates

Under homography, we can write the transformation of points in 3D from camera 1 to camera 2 as:

$$\boldsymbol{X}_2 = H\boldsymbol{X}_1 \quad \boldsymbol{X}_1, \boldsymbol{X}_2 \in \mathbb{R}^3$$

In the image planes, using homogeneous coordinates, we have

$$\lambda_1 \boldsymbol{x}_1 = \boldsymbol{X}_1, \quad \lambda_2 \boldsymbol{x}_2 = \boldsymbol{X}_2, \quad \text{therefore} \quad \lambda_2 \boldsymbol{x}_2 = H\lambda_1 \boldsymbol{x}_1$$

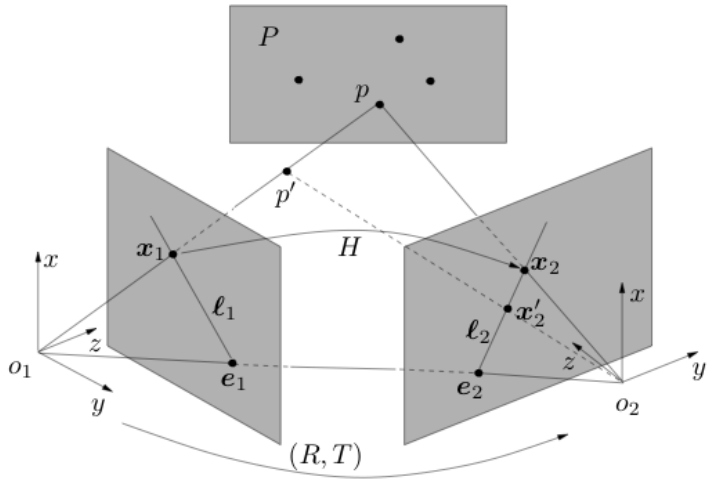
This means that $\boldsymbol{x}_2$ is equal to $H\boldsymbol{x}_1$ up to a scale (due to universal scale ambiguity). The consequence of this ambiguity is that $\boldsymbol{T}$ and d can't be individually extracted from H. We can only extract $\frac{\boldsymbol{T}}{d}$.

Note that $\boldsymbol{x}_2 \sim H\boldsymbol{x}_1$ is a *direct mapping* between points in the image planes.

If it is known that some points all lie in a plane in the scene, the image can be rectified directly without needing to recover and manipulate 3D coordinates.

4.4. Induced Homography

We say the homography H is *induced* by coplanar points in the scene.



The pose $(R, \boldsymbol{T})$ is a 3D relationship, but H is a shortcut that directly relates points in the image planes. When the homography is applied to a point p' that does not actually lie on plane P, it will get mapped in image plane 2 as if it were "splatted" on P along the ray $\overline{o_1 p'}$.

For a point p' not on plane P, the epipolar lines are given by:

$$\boldsymbol{l}_2 \sim \widehat{x}_2' H \boldsymbol{x}_1 \quad \text{and} \quad \boldsymbol{l}_1 \sim H^\top \boldsymbol{l}_2$$

$\boldsymbol{x}_2'$ is the image of point p' in image plane 2, and $H\boldsymbol{x}_1$ is where the homography maps $\boldsymbol{x}_1$ in image plane 2. These two points lie on $\boldsymbol{l}_2$, and their cross product gives us the $\boldsymbol{l}_2$.

The transformation of lines under the homography H arises as follows:

$$\begin{aligned} \boldsymbol{l}_2^\top \boldsymbol{x}_2 &= 0 \quad \forall\, \boldsymbol{x}_2 \\ \Rightarrow \boldsymbol{l}_2^\top H\boldsymbol{x}_1 &= 0 \quad \forall\, \boldsymbol{x}_1 \quad (\boldsymbol{l}_2^\top H \text{ must be the epipolar line through } \boldsymbol{x}_1) \\ \Rightarrow \boldsymbol{l}_1 &= H^\top \boldsymbol{l}_2 \\ \Rightarrow \boldsymbol{l}_2 &= H^{-\top} \boldsymbol{l}_1 \end{aligned}$$

Thus we have means to relate points and lines from camera 1 to camera 2:

$$\begin{aligned} \boldsymbol{x}_2 &= H\boldsymbol{x}_1 \quad \text{(contravariant)} \\ \boldsymbol{l}_2 &= H^{-\top}\boldsymbol{l}_1 \quad \text{(covariant)} \end{aligned}$$

The important point here is that we can recover the epipolar lines without knowing the essential matrix E.

Note: There is a "Six point algorithm" (which is not its official name) that uses 4 points on a plane to induce a homography, and two points not on the plane to find the epipolar lines. This appears in the next lecture.

4.5. Homography Estimation

To estimate H, we start from the equation $\boldsymbol{x}_2 \sim H\boldsymbol{x}_1$ and cross both sides with $\boldsymbol{x}_2$:

$$\begin{aligned} \widehat{x_2}\boldsymbol{x}_2 &\sim \widehat{x_2} H\boldsymbol{x}_1 \\ \Rightarrow \widehat{x_2} H\boldsymbol{x}_1 &= \boldsymbol{0} \end{aligned}$$

since any vector crossed with itself goes to $\boldsymbol{0}$. This is the *planar homography constraint*.

What would happen if we ignored the fact that these points were on a plane and tried to estimate E?

Since $\boldsymbol{x}_2 \sim H\boldsymbol{x}_1$, then for every $\boldsymbol{u} \in \mathbb{R}^3$, we know:

$$\begin{aligned} \boldsymbol{u} \times \boldsymbol{x}_2 = \widehat{u}\boldsymbol{x}_2 \perp H\boldsymbol{x}_1 & \\ \Rightarrow (\widehat{u}\boldsymbol{x}_2)^\top H\boldsymbol{x}_1 &= 0 \quad \text{(note: } \widehat{u} \text{ is skew-symmetric)} \\ \Rightarrow -\boldsymbol{x}_2^\top \widehat{u} H\boldsymbol{x}_1 &= 0 \quad \text{(since homogeneous, can drop minus sign)} \\ \Rightarrow \boldsymbol{x}_2^\top \widehat{u} H\boldsymbol{x}_1 &= 0 \\ \Rightarrow \boldsymbol{x}_2^\top E\boldsymbol{x}_1 &= 0 \end{aligned}$$

This final equality is true for *a family* of matrices $E = \widehat{u}H \in \mathbb{R}^{3\times 3}$ *besides* the true $E = \widehat{T}R$. This will cause the 8-point algorithm to crash.

4.6. Purely Rotating Camera

A camera that is rotating but has no translation maps points in 3D as

$$\boldsymbol{X}_2 = R\boldsymbol{X}_1$$

This is a special case of homography:

$$H = R + \frac{1}{d}\boldsymbol{T}\boldsymbol{N}^\top \quad \text{with} \quad \boldsymbol{T} = \boldsymbol{0},$$

so

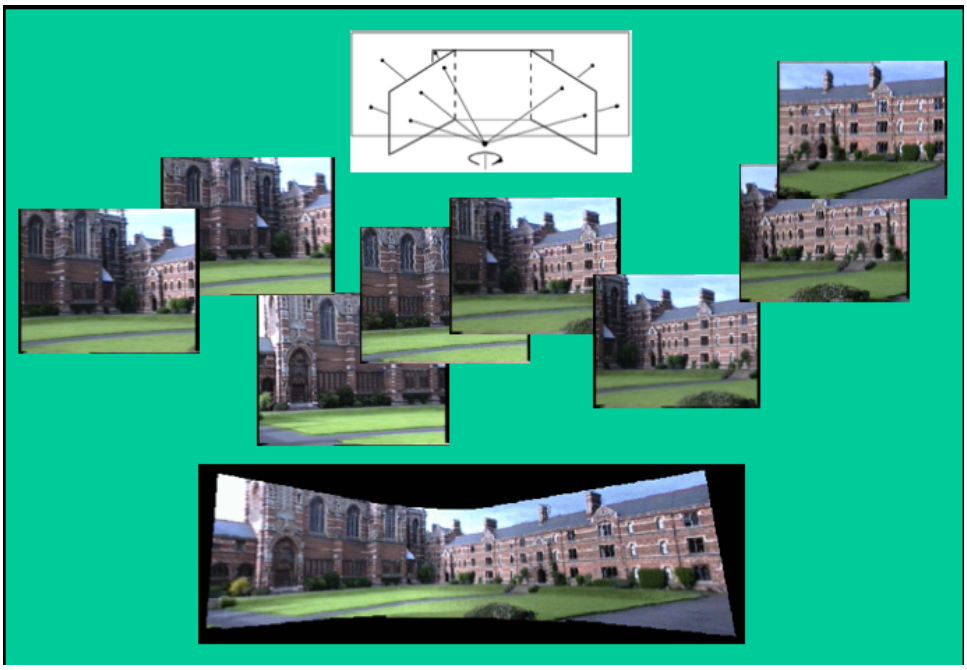
$$\widehat{x_2} H \boldsymbol{x}_1 = \boldsymbol{0}$$

becomes

$$\widehat{x_2} R \boldsymbol{x}_1 = \boldsymbol{0}$$

A camera rotating about its optical center captures images of a 3D scene as if the scene were painted on a plane infinitely far away from the camera. No depth can be perceived without a translation between the two views. Depth cues (parallax) can only be recovered when $\boldsymbol{T}$ is nonzero. Looking at the homography equation, the limit of H as d approaches infinity is R. Thus any pair of images of an arbitrary scene captured by a purely rotating camera is related by a planar homography.

A *planar panorama* can be constructed by capturing many overlapping images at different rotations, picking an image to be a reference, and then finding corresponding points between the overlapping images. The pairwise homographies are derived from the corresponding points, forming a mosaic that typically is shaped like a "bow-tie," as images farther away from the reference are warped outward to fit the homography. The figure below is from Pollefeys and Hartley & Zisserman.



4.7. Second Derivation of Homography Constraint

The homography constraint, element by element, in homogenous coordinates is as follows:

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \Leftrightarrow \boldsymbol{x}_2 = H\boldsymbol{x}_1$$

In inhomogenous coordinates ($x_2' = x_2/z_2$ and $y_2' = y_2/z_2$),

$$x_2' = \frac{H_{11}x_1 + H_{12}y_1 + H_{13}z_1}{H_{31}x_1 + H_{32}y_1 + H_{33}z_1}$$
$$y_2' = \frac{H_{21}x_1 + H_{22}y_1 + H_{23}z_1}{H_{31}x_1 + H_{32}y_1 + H_{33}z_1}$$

Without loss of generality, set $z_1 = 1$ and rearrange:

$$x_2'(H_{31}x_1 + H_{32}y_1 + H_{33}) = H_{11}x_1 + H_{12}y_1 + H_{13}$$
$$y_2'(H_{31}x_1 + H_{32}y_1 + H_{33}) = H_{21}x_1 + H_{22}y_1 + H_{23}$$

We want to solve for H. Even though these inhomogeneous equations involve the coordinates nonlinearly, the coefficients of H appear linearly.

Each corresponding point $\boldsymbol{x}_1^i \overset{corresponds}{\longleftrightarrow} \boldsymbol{x}_2^i$ gives two equations. H has 8 degrees of freedom, so we need 8 equations = 4 correspondences to get H (up

to a scale factor). This should match intuition in that one needs 4 corners to describe the mapping of a square under perspective projection.

To solve, stack H into $\boldsymbol{H}^s \in \mathbb{R}^9$ (i.e. `H(:)` in matlab). Write

$$\widehat{x_2} H \boldsymbol{x}_1 \quad \text{as}$$
$$a^\top \boldsymbol{H}^s = \mathbf{0}, \quad \text{where}$$
$$a \doteq \boldsymbol{x}_1 \otimes \widehat{x_2} \in \mathbb{R}^{9\times 3}$$

Think of this as stuffing all the necessary cross terms between the two sets of coordinates into blocks of a matrix. This allows one to make explicit the linear dependence of the values in H.

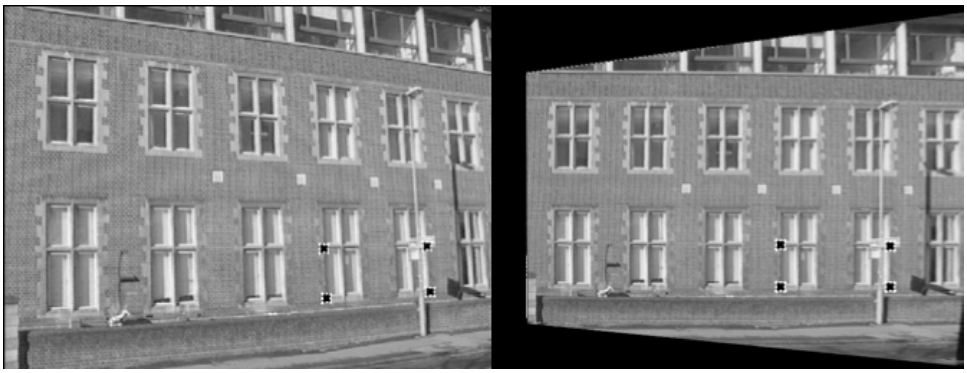
Note: since the rank of a 'hatted' matrix is 2, the third equation added by the Kronecker product is a linear combination of one of the other two, and is thus redundant. Also, as convenient as the Kronecker product is here, it does not carry over to the derivation for 3D case, which we'll need later on in the class.

Collect the a's for each correspondence into a "design matrix" χ,

$$\chi \doteq [a^1\ a^2\ \cdots\ a^n]^\top \in \mathbb{R}^{3n\times\ 9}, \quad \text{then}$$
$$\chi \boldsymbol{H}^s = \mathbf{0}$$

$\text{rank}(\chi) = 8$; solve for $\boldsymbol{H}^s$ as the null vector, then reshape it into H.

The process described above is known as the 4-point algorithm or the Direct Linear Transform (DLT). It appears as Algorithm 5.2 in MaSKS.[1] The pair of images below (from Hartley & Zisserman) shows an image before and after projective distortion correction using H obtained from the four indicated points in the left image and their known coordinates in the scene.



H is homogeneous, specified up to a universal scale factor. In MaSKS Section 5.3.3, they give the decomposition of H into $\{R, \frac{1}{d}\boldsymbol{T}, \boldsymbol{N}\}$. There are 4 possible solutions, two of which are physically possible in the sense of positive depth w.r.t. the two image planes.

[1]Here is a question to think about. Why can't you just solve for H via a pseudoinverse on $X_2 = HX_1$, where $X_i = [\boldsymbol{x}_i^1, \ldots, \boldsymbol{x}_i^n]$?