Lecture 11: Projection and Motion

A.L. Yuille

February 23, 2012

1 Introduction

2 Projection: Perspective and Approximations

We consider a pinhole camera (see Szeliski's book for more sophisticated models).

This is defined by the position of the pinhole $\vec{C}$, the focal length f, and a right-handed coordinate system $\vec{n}, \vec{p}, \vec{q}$. The unit vector $\vec{n}$ specifies the direction of view of the camera. The plane of the camera is specified by the equation $\vec{x} \cdot \vec{n} = k$. k is a constant. By requiring that the center of projection point $\vec{x}_0 = \vec{C} - f\vec{n}$ lies on the plane, we calculate $k = \vec{C} \cdot \vec{n} - f$. The vectors $\vec{p}, \vec{q}$ will specify a two-dimensional coordinate system on the plane. See figure (1).

Hence the plane is

$$\vec{x} \cdot \vec{n} = \vec{C} \cdot \vec{n} - f, \quad \text{or } (\vec{x} - \vec{C}) \cdot \vec{n} + f = 0. \tag{1}$$

A point $\vec{X}$ in space will be projected onto the plane. This can be computed by calculating where the straight line joining $\vec{X}$ to $\vec{C}$ intersects the plane. Points on the line are specified in terms of a parameter λ by:

$$\vec{X}(\lambda) = \vec{C} + \lambda(\vec{X} - \vec{C}). \tag{2}$$

The line intersects the plane for λ^* such that:

$$\lambda^* \vec{n} \cdot (\vec{X} - \vec{C}) + f = 0, \quad \lambda^* = \frac{-f}{\vec{n} \cdot (\vec{X} - \vec{C})}. \tag{3}$$

Hence the projection $\vec{x}_p$ of point $\vec{X}$ is given by:

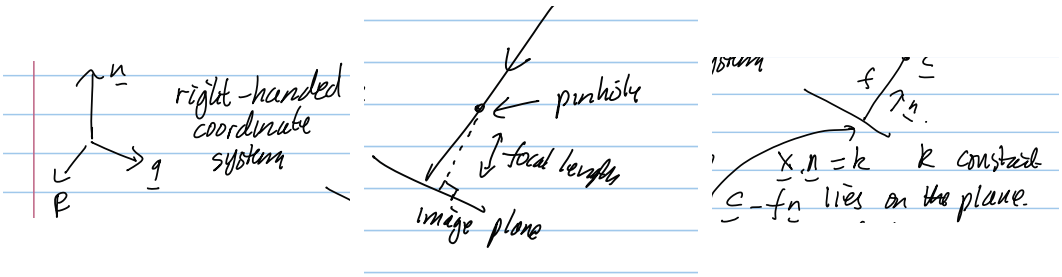

Figure 1: A right-handed coordinate system (left panel). The pinhole camera model (center panel). The image plane (right panel).

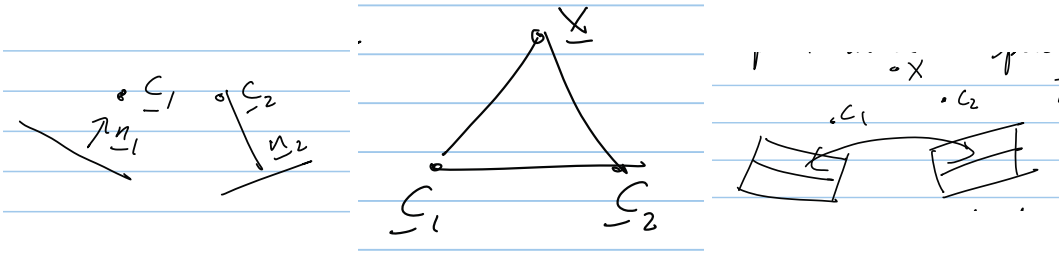

Figure 2: Two cameras with pinhole positions $\vec{C}_1$, $\vec{C}_2$ and pointing in directions $\vec{n}_1$, $\vec{n}_2$ (left panel). A point $\vec{X}$ in space together with the pinhole positions $\vec{C}_1, \vec{C}_2$ defines a plane. This intersects the image planes in two straight lines (center panel). Points on these lines are the epipolar lines (right panel) – points on them come from the same plane in apsce. Hence we can only match points on the corresponding epipolar lines. Camera callibration is required to estimate the camera parameters and the corresponding epipolar lines.

$$\vec{x}_p = \vec{C} - \frac{f(\vec{X} - \vec{C})}{(\vec{X} - \vec{C}) \cdot \vec{n}}. \tag{4}$$

The center of projection is at $\vec{x}_0 = \vec{C} - f\vec{n}$ (all points in space which lie on the line $\vec{C} + \mu\vec{n}$, $\forall \mu$ get projected to this point). The projection of X relative to the center of projection is specified by:

$$\vec{x}_p - \vec{x}_0 = f\vec{n} - \frac{f(\vec{X} - \vec{C})}{(\vec{X} - \vec{C}) \cdot \vec{n}} = \frac{f}{(\vec{X} - \vec{C})}\{\vec{n}\{(\vec{X} - \vec{C}) \cdot \vec{n}\} - (\vec{X} - \vec{C})\}. \tag{5}$$

Observe that $(\vec{x}_p - \vec{x}_0) \cdot \vec{n} = 0$, so $\vec{x}_p - \vec{x}_0$ lies in the two-dimensional spaced spanned by $\vec{p}$ and $\vec{q}$. Hence we can express:

$$\begin{aligned} \vec{x}_p - \vec{x}_0 &= \vec{p}\{\vec{p} \cdot (\vec{x}_p - \vec{x}_0)\} + \vec{q}\{\vec{q} \cdot (\vec{x}_p - \vec{x}_0)\}, \\ \vec{x}_p - \vec{x}_0 &= x_p\vec{p} + y_q\vec{q}, \\ \text{where } x_p = \vec{p} \cdot (\vec{x}_p - \vec{x}_0) &= -\frac{f(\vec{X} - \vec{C}) \cdot \vec{p}}{(\vec{X} - \vec{C}) \cdot \vec{n}} \\ y_p = \vec{q} \cdot (\vec{x}_p - \vec{x}_0) &= -\frac{f(\vec{X} - \vec{C}) \cdot \vec{q}}{(\vec{X} - \vec{C}) \cdot \vec{n}}. \end{aligned} \tag{6}$$

3 Epipolar Lines, Camera calibration

If we have two cameras, or two eyes, there is an epipolar line constraint which relates points in the two images, see figure (2). There is a special case if the two cameras point straight ahead and are parallel to each other then the epipolar lines are parallel to each other, see figure (3)

Theoretical Results: if we have N views of M points then we can solve for the camera parameters. This requires knowing which points can be matched.

Note: see Szeliski or the many books on these topics.

4 Linear Projection Models: Taylor series analysis

The perspective projection equations are nonlinear. In many cases it is helpful to use linear approximations. The most popular are orthographic projection and weak affine. These approximations hold in certain

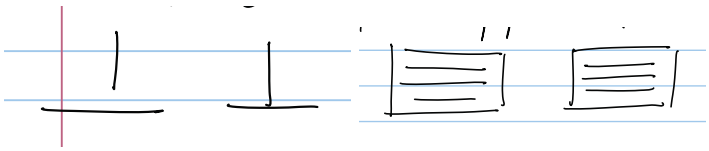

Figure 3: If both cameras point in the same direction (left panel) then the epipolar lines are parallel (right panel).

situations, such as when the depth variations of objects/scene being viewed is much less than the average distance to the objects.

To derive these linear approximations, we perform Taylor series analysis by computing the expansion about a fixed point $\vec{X}_0$ in space. We compute the projections for points $\vec{X} = \vec{X}_0 + \Delta\vec{X}$ where $|\Delta\vec{X}|$ is assumed to be small and will drop the $O|\Delta\vec{X}|^2$ terms in the expansion.

We express:

$$\begin{aligned} x_p &= -f\{(\vec{X}_0 - \vec{C}) \cdot \vec{p} + \Delta\vec{X} \cdot \vec{p}\}\{(\vec{X}_0 - \vec{C}) \cdot \vec{n} + \Delta\vec{X} \cdot \vec{n}\}^{-1}, \\ y_p &= -f\{(\vec{X}_0 - \vec{C}) \cdot \vec{q} + \Delta\vec{X} \cdot \vec{q}\}\{(\vec{X}_0 - \vec{C}) \cdot \vec{n} + \Delta\vec{X} \cdot \vec{n}\}^{-1} \end{aligned} \tag{7}$$

We perform Taylor's expansion keeping only the first order terms in $|\Delta\vec{X}|$ (using $(a+\epsilon)^{-1} = (1/a)(1 + \epsilon/a)^{-1} \approx 1/a - \epsilon/a^2$).

This gives:

$$\begin{aligned} x_p &= -f\frac{(\vec{X}_0 - \vec{C}) \cdot \vec{p}}{(\vec{X}_0 - \vec{C}) \cdot \vec{n}} - \frac{f\Delta\vec{X} \cdot \vec{p}}{(\vec{X}_0 - \vec{C}) \cdot \vec{n}} + \frac{f\{(\vec{X}_0 - \vec{C}) \cdot \vec{p}\}\Delta\vec{X}_0 \cdot \vec{n}}{\{(\vec{X}_0 - \vec{C}) \cdot \vec{n}\}^2} \\ y_p &= -f\frac{(\vec{X}_0 - \vec{C}) \cdot \vec{q}}{(\vec{X}_0 - \vec{C}) \cdot \vec{n}} - \frac{f\Delta\vec{X} \cdot \vec{q}}{(\vec{X}_0 - \vec{C}) \cdot \vec{n}} + \frac{f\{(\vec{X}_0 - \vec{C}) \cdot \vec{q}\}\Delta\vec{X}_0 \cdot \vec{n}}{\{(\vec{X}_0 - \vec{C}) \cdot \vec{n}\}^2} \end{aligned} \tag{8}$$

To understand this approximation, we need to see what terms are being neglected. The approximation ignores higher order terms in the expansion of the denominator – these are of form

$$\frac{-f\{(\vec{X}_0 - \vec{C}) \cdot \vec{q}\}|\Delta\vec{X}_0 \cdot \vec{n}|^2}{\{(\vec{X}_0 - \vec{C}) \cdot \vec{n}\}^3} \tag{9}$$

Hence the requirement is that $\Delta\vec{X} \cdot \vec{n}/\{(\vec{X}_0 - \vec{C}) \cdot \vec{n}\}$ is small. In other words, that the variation in depth on the object along the line of sight direction $\vec{c}$, relative to a fixed point $\vec{X}_0$, is smaller than the depth of the object. This can happen, for example, if we let $\vec{X}_0$ be a point on the object and require that the variation in depth of the object is small. There are, of course, many cases where this approximation is not good – see examples of vanishing points in the last section– but the simplicity of using linear equations helps compensate for the approximation.

A simpler approximation can happen if we choose $\vec{X}_0$ to be on the line of sight (i.e. $\vec{X}_0 = \lambda\vec{n} + \vec{C}$ for some λ). In this case terms like $(\vec{X}_0 - \vec{C}) \cdot \vec{p} = (\vec{X}_0 - \vec{C}) \cdot \vec{q} = 0$. Then we obtain:

$$\begin{aligned} x_p &= -f\Delta\vec{X} \cdot \vec{p}\{(\vec{X}_0 - \vec{C}) \cdot \vec{n} + \Delta\vec{X} \cdot \vec{n}\}^{-1} \\ &\approx -f\frac{\Delta\vec{X} \cdot \vec{p}}{(\vec{X}_0 - \vec{C})\vec{\cdot n}} + f\frac{\Delta\vec{X} \cdot \vec{p}\Delta\vec{X} \cdot \vec{n}}{\{(\vec{X}_0 - \vec{C})\vec{\cdot n}\}^2} \end{aligned} \tag{10}$$

If we ignore the higher order terms then we get *orthographic projection* up to a constant scaling factor given by $f/\{(\vec{X}_0 - \vec{C})\}$. The higher order terms will be small provided $\Delta\vec{X} \cdot \vec{n}/\{(\vec{X}_0 - \vec{C}) \cdot \vec{n}\}$ is small.

5 Estimating Depth

Suppose we have a sequence of points where we know the correspondence between points at different times. We also assume that all these points are observed. We also assume that the points are moving rigidly (i.e. rotating and translating).

This can occur if we are looking at an object from different viewpoints (e.g. Ullman and Basri) or if we are doing structure from motion (Kontsevich et al, Tomasi and Kanade). Ullman and Basri pointed out that different views of the same object lay on linear subspaces (provided orthographic or weak affine was used). Tomasi and Kanade (and earlier Kontsevich et al) showed that this linear assumption enabled structure to motion to be formulated as a least square fitting problem which is bilinear in the positions of the viewed 3D points and the camera parameters, which could be solved by linear algebra techniques (Singular Value Decomposition) up to some ambiguities.

More formally, we have N points $\{\vec{X}^\mu : \mu = 1, ..., N\}$ in space. We have M projections specified by the camera parameters $cam = \{(f_i, \vec{c}_i, \vec{n}_i, \vec{p}_i, \vec{q}_i) : i = 1, ..., M\}$. Note that each camera has seven parameters.

By the previous sections, the projections are specified by:

$$x_{\mu i} = \frac{-f_i(\vec{X}^\mu - \vec{C}_i) \cdot \vec{p}_i}{(\vec{X}^\mu - \vec{C}_i) \cdot \vec{n}_i}, \quad y_{\mu i} = \frac{-f_i(\vec{X}^\mu - \vec{C}_i) \cdot \vec{q}_i}{(\vec{X}^\mu - \vec{C}_i) \cdot \vec{n}_i} \tag{11}$$

We can specify a generative model for the data given by:

$$P(\{(x_{\mu i}, y_{\mu i}\}|\{\vec{X}^\mu\}, \{cam_i\}) = \frac{1}{Z} \exp\{-E[x, y, \vec{X}, cam]/T\}. \tag{12}$$

The simplest model is where we assume that the generative process is corrupted by additive Gaussian noise. This gives an energy function:

$$E[x, y, \vec{X}, cam] = \sum_{\mu i}\{x_{\mu i} + \frac{f_i(\vec{X}^\mu - \vec{C}_i) \cdot \vec{p}_i}{(\vec{X}^\mu - \vec{C}_i) \cdot \vec{n}_i}\}^2 + \sum_{\mu i}\{y_{\mu i} + \frac{f_i(\vec{X}^\mu - \vec{C}_i) \cdot \vec{q}_i}{(\vec{X}^\mu - \vec{C}_i) \cdot \vec{n}_i}\}^2. \tag{13}$$

Then ML estimation of the positions of the points $\{\vec{X}^\mu\}$ and the camera parameters $\{cam_i\}$ can be obtained by minimizing $E[x, y, \vec{X}, cam]$ with respect to $\{\vec{X}^\mu\}$ and $\{cam_i\}$ simultaneously. This leads to the Sturm-Triggs algorithm and modifications by Oliensis and Hartley.

Of course, it is not clear that the quadratic energy – e.g., Gaussian assumption made here – is correct. It simplifies the mathematics but should be altered to include, for example, missing datapoints and other forms of noise. In addition, prior assumptions can be included on $\{\vec{X}^\mu\}$ and the camera parameters $\{cam_i\}$.

6 Linearization and Structure from Motion

There are disadvantages to directly minimizing $E[x, \vec{X}, cam]$. It may have local minima – and it may be sensitive to small errors. So instead we linearize the projection.

Let $\Pi(\vec{X}^3_\mu, K^m)$ be the projection of a point $\vec{X}^3_\mu$ for a camera with parameter K^m. If we assume a linear projection model then $\vec{x}^p_{\mu,m} = K^m\vec{X}^3_\mu$ where K^m is a 2×3 matrix and $\vec{X}^3_\mu$ is a point in three-dimensional space.

We assume that the motion of the viewed scene can be modeled as if it is perfectly rigid. Empirically this method gives reasonable results even if the object is only semi-rigid.

More formally, we assume that the pixels $\{(x_\mu, y_\mu) : \mu \in \mathbf{D}\}$ in the reference image correspond to points $\{(x_\mu, y_\mu, z_\mu) : \mu \in \mathbf{D}\}$ in three-dimensional space, where the $\{z_\mu : \mu \in \mathbf{D}\}$ are unknown and need to be

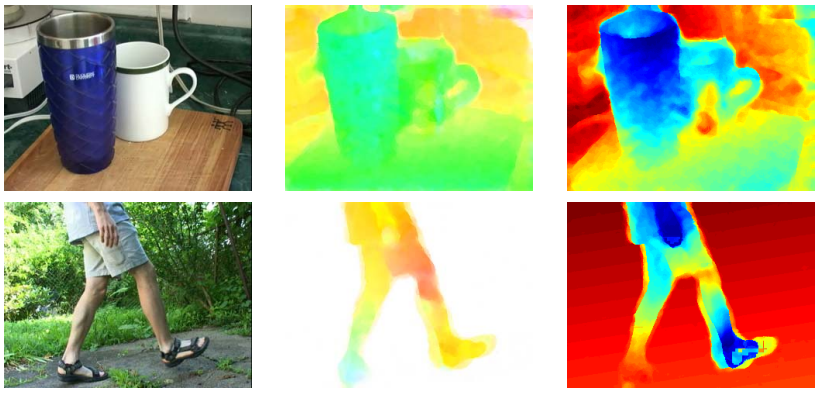

Figure 4: Two examples of dense flow estimation (based on Middlebury flow-color mapping, middle column) and dense depth estimation (right column).

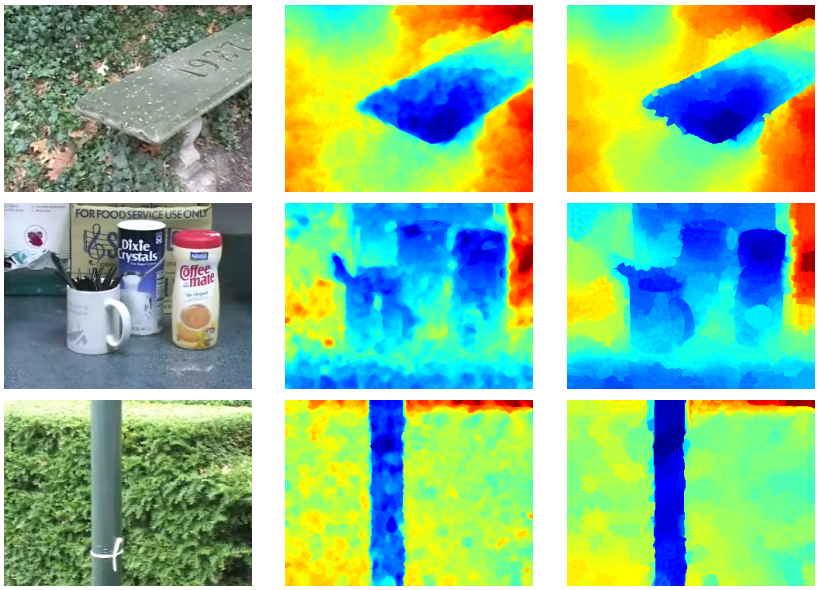

Figure 5: Three examples of dense flow field inferred from the superpixel MRF model. Left: Reference image;Middle: Raw depth estimation from SVD; Right: Smoothed depth field.

estimated (i.e. we have used an optical flow algorithm to solve the correspondence between different image frames).

We assume that the other two images $\mathbf{I}_{-m}$ and $\mathbf{I}_m$ are generated by these points $\mathbf{X}^3$ using scaled orthographic projection with camera coefficients $\mathbf{C}_{-m}, \mathbf{C}_m$. Hence the positions of these points $\mathbf{X}^3$ in images $\mathbf{I}_{-m}, \mathbf{I}_m$ is given by $\Pi(\mathbf{X}^3; \mathbf{C}_{-m})$ and $\Pi(\mathbf{X}^3; \mathbf{C}_m)$ respectively, where $\Pi(.,.)$ is the projection. We seek the camera coefficients $\mathbf{C}_{-m}, \mathbf{C}_m$ and depths $\{z_\mu\}$ so that the projections best agree with the correspondences between $\mathbf{I}_{-m}, \mathbf{I}_0, \mathbf{I}_m$ estimated by the motion flow algorithm. For computational simplicity, we formulate this as minimizing a quadratic cost function:

$$\begin{aligned} E[\{z_\mu\}; \mathbf{C}_{-m}, \mathbf{C}_m] = \quad & \textstyle\sum_\mu |\mathbf{x}_\mu + \mathbf{v}_\mu^f - \Pi(\mathbf{X}_\mu^3; \mathbf{C}_m)|^2 \\ & + \textstyle\sum_\mu |\mathbf{x}_\mu - \mathbf{v}_\mu^b - \Pi(\mathbf{X}_\mu^3; \mathbf{C}_{-m})|^2. \end{aligned} \tag{14}$$

This minimization can be solved algebraically using singular value decomposition to estimate $\{z_\mu^*\}$ and $\mathbf{C}_{-m}^*, \mathbf{C}_m^*$ (Kontsevich et al 1987, Tomasi and Kanade 1992). For the scaled orthographic approximation there is only a single ambiguity $\{z_\mu^*\} \mapsto \{\lambda z_\mu^*\}$ where λ is an unknown constant (but some estimate of λ can be made using knowledge of likely values of the camera parameters). We do not attempt to estimate λ and instead use the method described in Tomasi and Kanade (1992) which implicitly specifies a default value for it. Note, if we use a more general affine camera model then there is a larger class of ambiguities.