CS 280, Computer Vision (Prof. Jitendra Malik) March 3, 1999
Lecture #12 (Epipolar geometry and shape from texture.) **DRAFT** Notes by Dimitri Chernyak

---

- Epipolar Geometry.
- Shape from Texture.

1 Epipolar Geometry

1.1 Geometric aside.

If vectors $\vec{a}$, $\vec{b}$,and $\vec{c}$ are coplanar, then:

$$\vec{a} \cdot (\vec{b} \otimes \vec{c}) = 0$$

or

$$\vec{a^T} \cdot (\hat{b} \cdot \vec{c}) = 0$$

where $\hat{b}$ is the skew-symmetric matrix associated with $\vec{b}$.

Note that the magnitudes of the vectors don't make any difference. Only the directions matter.

1.2 Epipolar geometry

Consider the two cameras and the point X in the outside world as shown on Figure 1. Points C_1 and C_2 are the focal points of the left and right cameras respectively, and x and x' are the projections of point X onto the corresponding image planes.

Let $\vec{t}$ denote the translational component between the cameras and R, - the rotational one.

Plane formed by $C_1 X C_2$ is called an *epipolar plane*. The lines of intersection of epipolar plane with the camera planes are called *epipolar lines*.

Projection of point C_1 onto the image plane of the right camera is point E_r, which is called an *epipole*. All epipolar lines pass through this epipole. Similarly, for the left camera there is a corresponding epipole E_l, which is projection of point C_2.

In a special case when the cameras are parallel ($R = I$ identity matrix), the epipolar lines are parallel, since the epipole for each camera is at infinity.

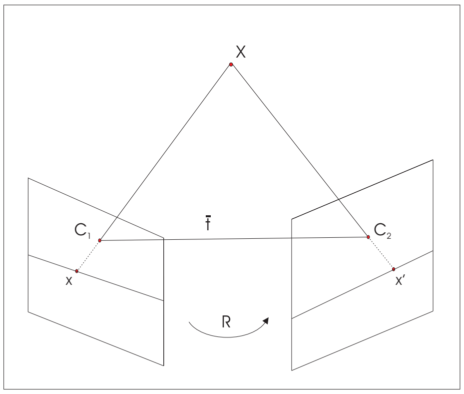

Figure 1: Epipolar geometry

Lets choose the coordinate system of the left camera to be the coordinate system for the epipolar constrain with origin at point C_1. Vectors $\vec{x}$, $\vec{t}$,and $\vec{Rx'}$ are coplanar, so the following must hold:

$$\vec{x} \cdot (\vec{t} \otimes \vec{Rx'}) = \vec{x^T} \hat{t} R \vec{x'} = \vec{x^T} E \vec{x'} = 0,$$

where $E = \hat{t}R$ and $\hat{t} = \begin{bmatrix} 0 & t3 & -t2 \\ -t3 & 0 & t1 \\ t2 & -t1 & 0 \end{bmatrix}$

This is the epipolar constrain equation.

1.3 Epipolar constrain

Epipolar constrain is very useful for solving the correspondence problem for stereopsis. Let A be a point in the image plane of the left camera. We can find the epipolar line l passing through A based on the relative positions of the two cameras. Using the epipolar constrain we can find the corresponding epipolar line l' in the image plane of the right camera. Therefore, we can restrict the search for the corresponding point A' to the line l', i.e. we have to do one dimensional search instead of the full two dimensional one. This property is preserved under projective transformation because of the preserved co-planarity.

2 Shape from texture

2.1 Basic task

Texture is a spatially repeated pattern of elements. These can be regular (such as floor tiles) or stochastic (such as animal skin or field of flowers). Given a surface with repetitive or quasi-repetitive elements, texture elements would look different on a projective plane. Shape from texture uses these distortions to recover the actual surface geometry in the world.

Shape From		
Motion	Stereo	Texture
Measure optical flow (apparent change in position)	Measure disparities	Measure texture distortion
Compute structure from motion	Compute structure from disparities	Compute structure from texture distortions

Table 1: 3D shape recovery.

2.2 Shape from variations.

Using apparent variations in texture, one can recover the surface using F^* backprojection mapping. The problem is quite similar to recovering shape from motion or stereo by considering relative changes in appearance of elements. Table 1 summarizes the similarities:

2.3 What is texure distortion?

If the change between two nearby texture patches on the image plane is small, it is likely that these patches are nearby on the actual surface. Given a texture patch on the image plane we can fit texture distortion for each direction with some affine transformation. What is being preserved? For regular textures it's the texture element; for stochastic texture it's the statistical properties.

2.4 3D shape computation.

In case of motion, translation, rotation and depth (Z-value) variations give rise to optical flow. In case of texture, slant, tilt, distance, surface normal, curvatures at point $\vec{p}$ give rise to affine texture distortions in different direction from $\vec{p}$.