18CSE390T
Computer Vision

S2-SLO2-Self-Calibration

Self-calibration

- *Auto-calibration* is developed for covering a projective reconstruction into a metric one, which is equivalent to recovering the unknown calibration matrix K_j associated with each image.
- In the presence of additional information about scene, different methods can be applied.
- If there are parallel lines in the scene, three or more vanishing points, which are the images of points at infinity, can be used to establish homography for the plane at infinity, from which focal length and rotation can be recovered.

Self-calibration (cont).

- In the absence of external information:

 consider all sets of camera matrices $P_j = K_j[\ \mathrm{R_j} \mid \mathrm{t_j}\ ]$ projecting world coordinates $p_i=(X_i,Y_i,Z_i,W_i)$ into screen coordinates $x_{ij} \sim P_j p_i$.

- Consider transforming the 3D scene $\{p_i\}$ through an arbitrary 4✖ 4 projective transformation $\tilde{H}$, yielding a new model consisting of points $p'_i = \tilde{H} p_i$.
- Post-multiplying each other P_j matrix by $\tilde{H}^{-1}$ still produces the same screen coordinates and a new set of calibration matrices can be computed by applying RQ decomposition to the new camera matrix .

Self-calibration (cont).

- A technique that can recover the focal lengths (f_0, f_1) of both images from fundamental matrix F in a two-frame reconstruction.
- Assume that camera has zero skew, a known aspect ratio, and known optical center.
- Most cameras have square pixels and an optical center near middle of image and are likely to deviate from simple camera model due to radial distortion
- Problem occurs when images have been cropped off-center.

Self-calibration (cont).

- Take left to right singular vectors $\{u_0, u_1, v_0, v_1\}$ of fundamental matrix $\boldsymbol{F}$ and their associated singular values $\{\sigma_0, \sigma_1)$ and form the equation:

$$\frac{\boldsymbol{u}_1^T \boldsymbol{D}_0 \boldsymbol{u}_1}{\sigma_0^2 \boldsymbol{v}_0^T \boldsymbol{D}_1 \boldsymbol{v}_0} = -\frac{\boldsymbol{u}_0^T \boldsymbol{D}_0 \boldsymbol{u}_1}{\sigma_0 \sigma_1 \boldsymbol{v}_0^T \boldsymbol{D}_1 \boldsymbol{v}_1} = \frac{\boldsymbol{u}_0^T \boldsymbol{D}_0 \boldsymbol{u}_0}{\sigma_1^2 \boldsymbol{v}_1^T \boldsymbol{D}_1 \boldsymbol{v}_1},$$

two matrices:

$$\boldsymbol{D}_j = \boldsymbol{K}_j \boldsymbol{K}_j^T = \mathrm{diag}(f_j^2, f_j^2, 1) = \begin{bmatrix} f_j^2 & & \\ & f_j^2 & \\ & & 1 \end{bmatrix}$$

Self-calibration (cont).

- Encode the unknown focal length. Write numerators and denominators as:

$$
\begin{aligned}
e_{ij0}(f_0^2) &= \boldsymbol{u}_i^T \boldsymbol{D}_0 \boldsymbol{u}_j = a_{ij} + b_{ij} f_0^2, \\
e_{ij1}(f_1^2) &= \sigma_i \sigma_j \boldsymbol{v}_i^T \boldsymbol{D}_1 \boldsymbol{v}_j = c_{ij} + d_{ij} f_1^2.
\end{aligned}
$$

Application: View Morphing

- Application of basic two-frame structure from motion.
- Also known as view interpolation.
- Used to generate a smooth 3D animation from one view of a 3D scene to another.
- To create such a transition: smoothly interpolate camera matrices, i.e., camera position, orientation, focal lengths. More effect is obtained by easing in and easing out camera parameters.
- To generate in-between frames: establish full set of 3D correspondences or 3D models for each reference view.

Application: View Morphing

- Triangulate set of matched feature points in each image .
- As the 3D points are re-projected into their intermediate views, pixels can be mapped from their original source images to their new views using affine projective mapping.
- The final image then composited using linear blend of the two reference images as with usual morphing.