CIS 5800

Machine Perception

Instructor: Lingjie Liu

Lec 11: March 3, 2025

Administrivia

- Get started on HW2 ASAP! Due March 5, 11:59pm ET.

Summary of Tools for Single-View Geometry (1/2)

- Finding vanishing points:
  - Using parallel lines
  - Using homographies
  - Using “cross ratios”
- Using VPs to compute horizon and determine camera orientation w.r.t. ground (assuming standard intrinsics K)
- Finding homographies from planes:
  - from 2 orthogonal VPs plus correspondences for the 2 points (0, 0), and (1, 1).
  - from 4 non-square points using inversion and composition of 2 homographies.
  - from >=4 arbitrary non-collinear point correspondences.
- Using homographies for:
  - Virtual billboard display
  - Finding camera pose and refining with Kabsch (given intrinsics K)

Summary of Tools for Single-View Geometry (2/2)

- Using cross ratios to measure real lengths:
  - Along a line, given 2 world lengths.
  - Along a line, given 1 world length and the VP.
  - Using VPs on ground plane, vertical VP, and vertical cross ratios for length transfer.
- Using pose and intrinsics (full projection matrix) to project AR objects onto world.
- Finding extrinsics from 3 2D->3D point correspondences (P3P problem)
  - What if we had depth images?
  - What if we only had standard RGB images?
- Finding extrinsics from n>3 2D->3D point correspondences not on a plane (PnP problem)
- Finding camera intrinsics (camera calibration):
  - Finding intrinsics K from 3 orthogonal VPs (camera calibration part 1)
  - Overview of more general camera intrinsics calibration under radial distortion

Next, onwards to 2 Views!

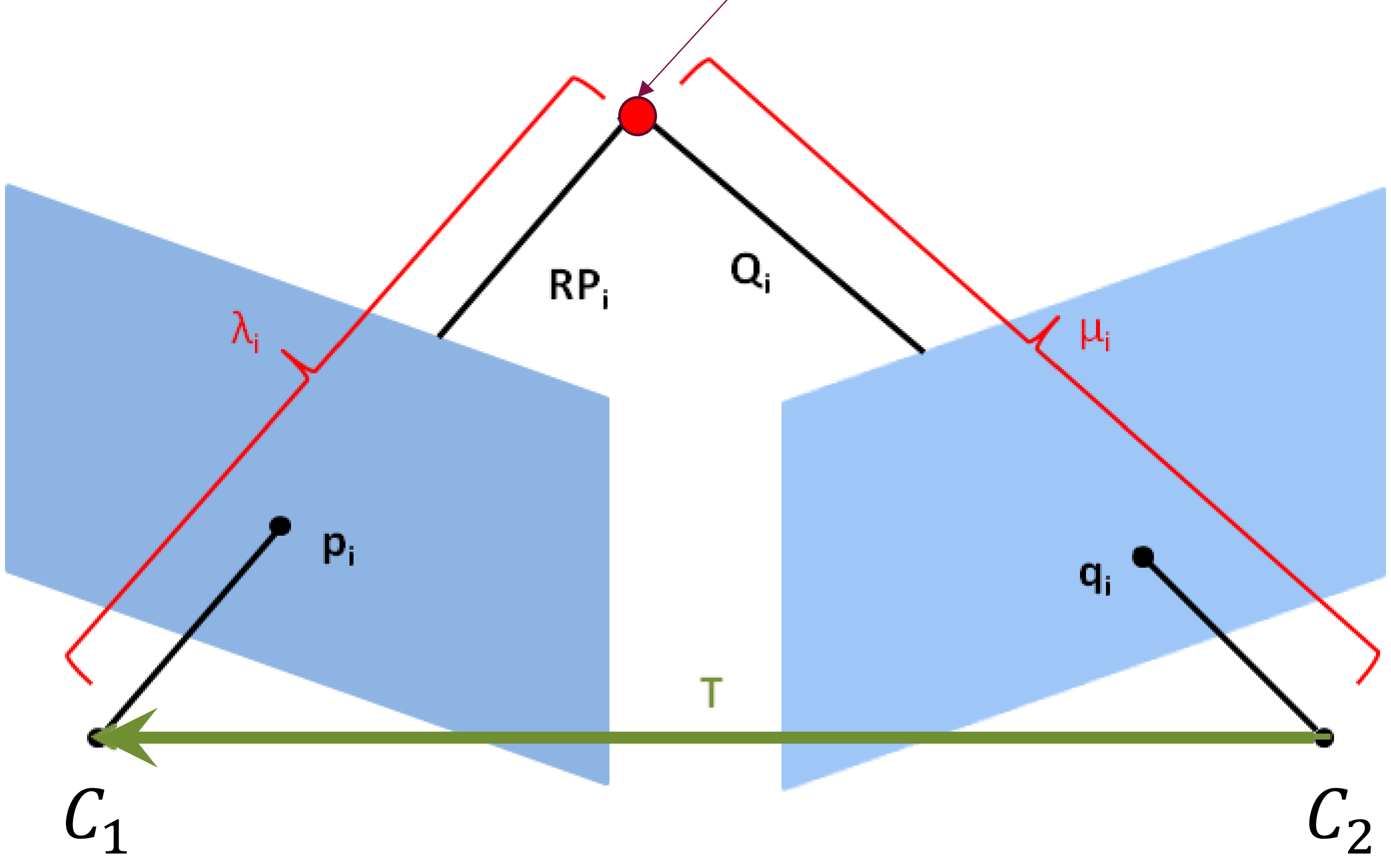
3D Motion from Two Views or Structure from Motion (SfM)

Input: Two Calibrated Views of the same 3D scene

Intrinsics known,
So we can always stick to "calibrated coordinates" like we used in P3P, rather than pixel coordinates.

Input: Two Calibrated Views of the Same 3D Scene

$$\boxed{[\lambda \boldsymbol{p}]_{\text{in } C1 \text{ camera coords}} = [\boldsymbol{\mu q}]_{\text{in C2 camera coords}}}$$



$$[X]_{C2} = R[X]_{C1} + T$$

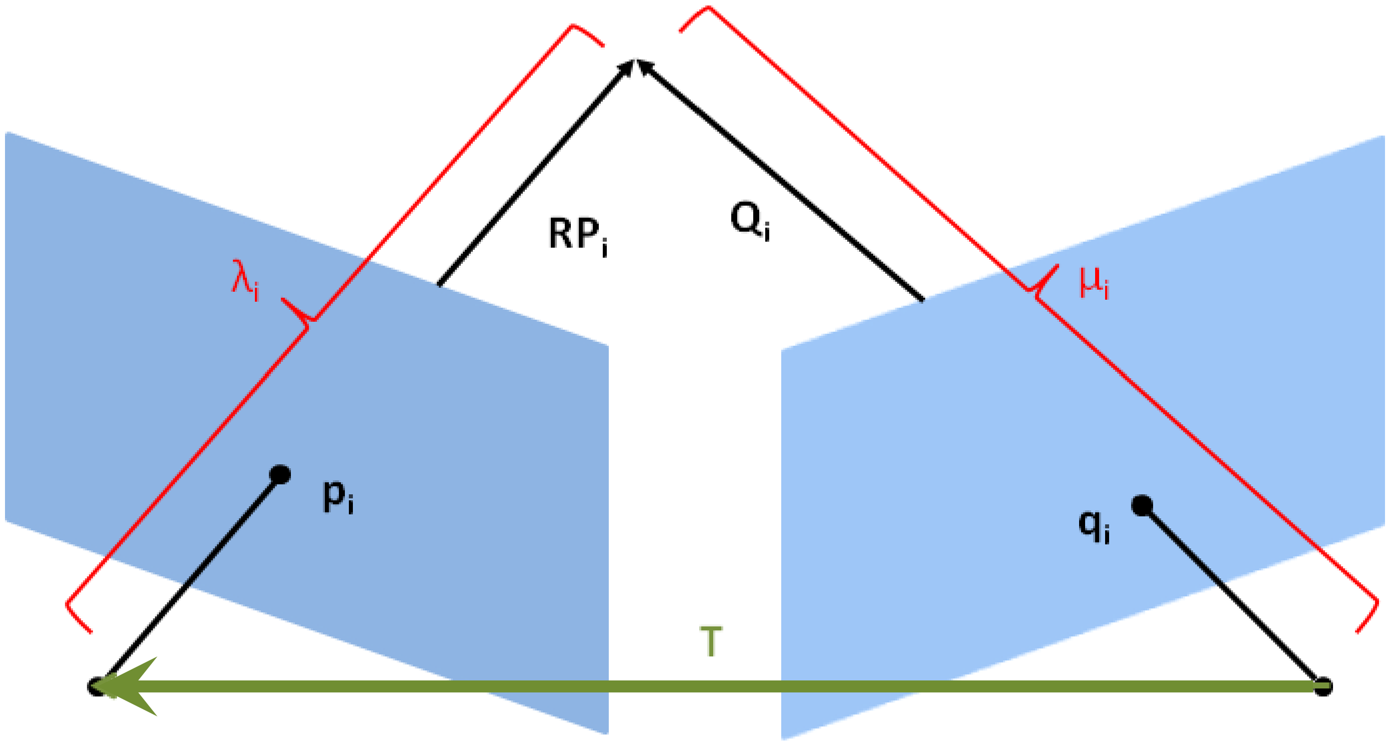
Now, any 3D coordinates in C_1 frame:
$[X]_{C1} \rightarrow [R[X]_{C1} + T]_{C2}$

So $[\lambda p]_{C1} \rightarrow [R(\lambda p) + T]_{C2} = [\mu q]_{C2}$

Or now that we are in the same coordinate system:

$$R(\lambda p) + T \ = \mu q$$

Two Calibrated Views of the Same 3D Scene



$$R(\lambda p) + T \ = \mu q$$

Given 2D correspondences (p,q)

Find motion R, T and depths λ, μ.

PnP vs. 2-View Structure from Motion (SfM)

PnP

1. Gn. "world frame" points P_i and corresponding calibrated coordinates $K^{-1}x_i$
2. Set camera frame 3D coordinates (with scale/depth ambiguity) to $\lambda_i K^{-1} x_i$
3. Then solve (for R, T, λ):

$$\lambda_i K^{-1} x_i = R P_i + T \ \forall i$$

Where R, T denotes transformation between camera and world.

Structure from Motion

1. No "world frame". Instead, just calibrated image plane coordinates $p_i = K_1^{-1} x_{i1}$ and $q_i = K_2^{-1} x_{i2}$ **of the same 3D point $\boldsymbol{P_i}$.**
2. So corresponding camera frame 3D coordinates in the two frames are: $\lambda_i p_i$ and $\mu_i q_i$.
3. Now solve for "motion between cameras" R, T and the scales λ_i, μ_i (which permit getting 3D coordinates of P_i in either camera frame)

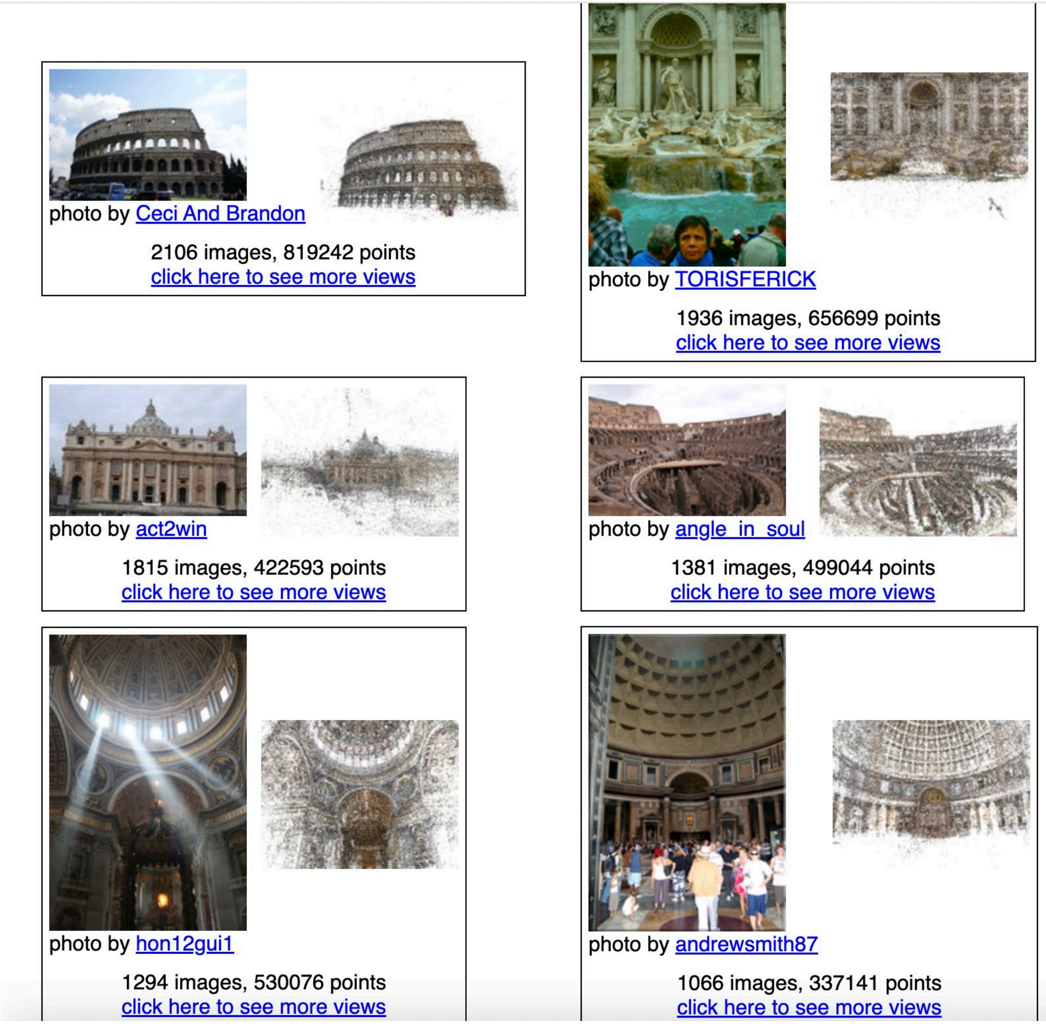
SfM, SLAM, Visual Odometry

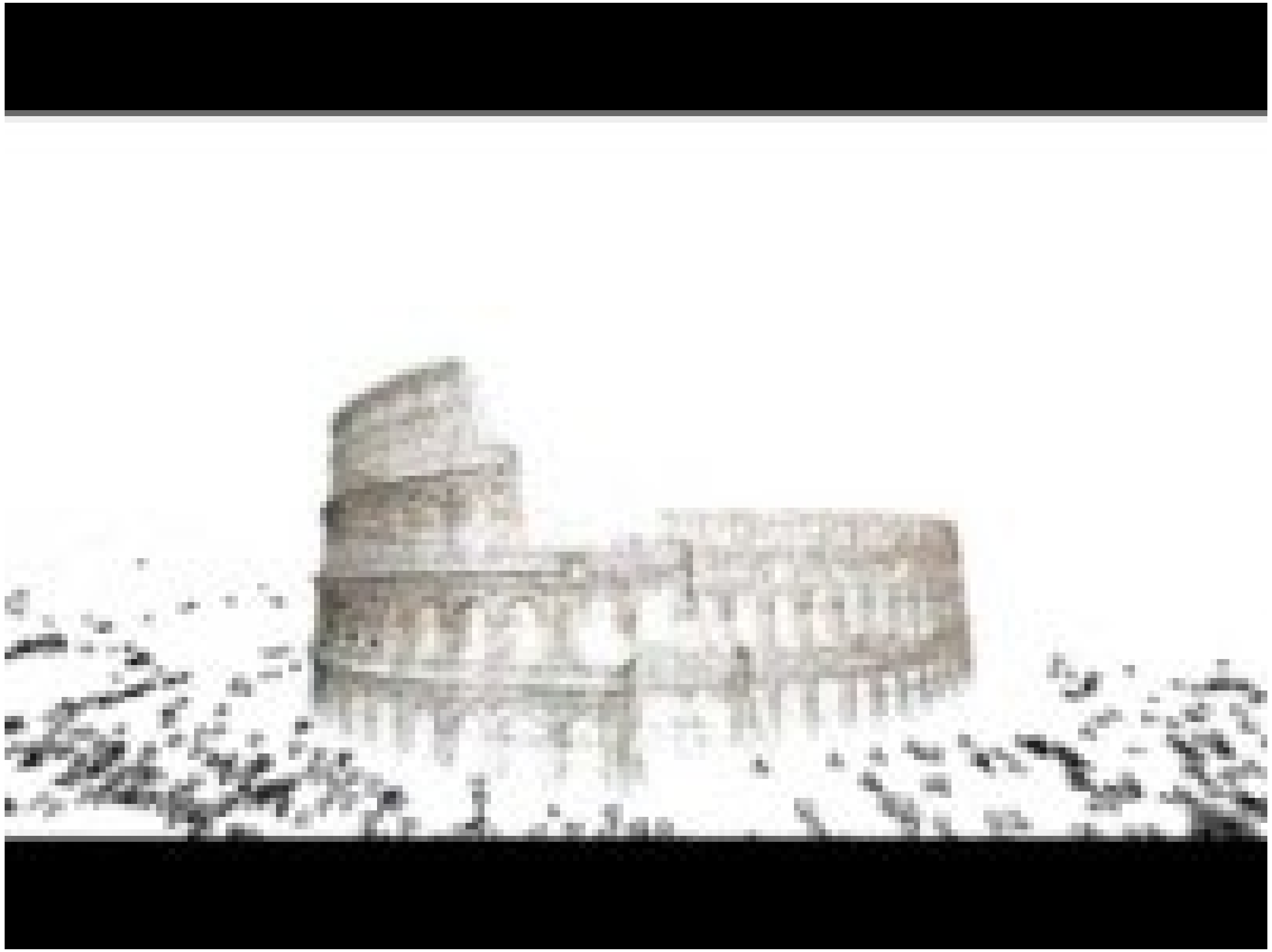
- **"SfM" (structure from motion):** graphics and computer vision folks interested mainly in building 3D models of scenes/objects. Images could be unordered, from multiple different cameras taking pictures of the same 3D scene/object. Usually performed offline.
- **"VO" (visual odometry):** mainly interested in localizing and tracking the robot alone, i.e. how far has it traveled in what direction, what speed etc. over short time window. Images usually frames of a video captured from a moving camera in a static / close-to-static world. Often performed online, as images stream in. Usually a single pre-calibrated camera.
- **"SLAM" (simultaneous localization and mapping):** Jointly estimating 3D models of scenes and localizing the robot (camera) w.r.t. that scene. Images similar to VO, also streaming, but emphasis on long-term map consistency. VO++. We'll see an example of a SLAM system later.

Methods largely the same, with differences in emphases depending on which output (depths, transformations, or both) matter most.

SfM Application: Building Rome in a Day

SfM techniques applied to Flickr image collections!





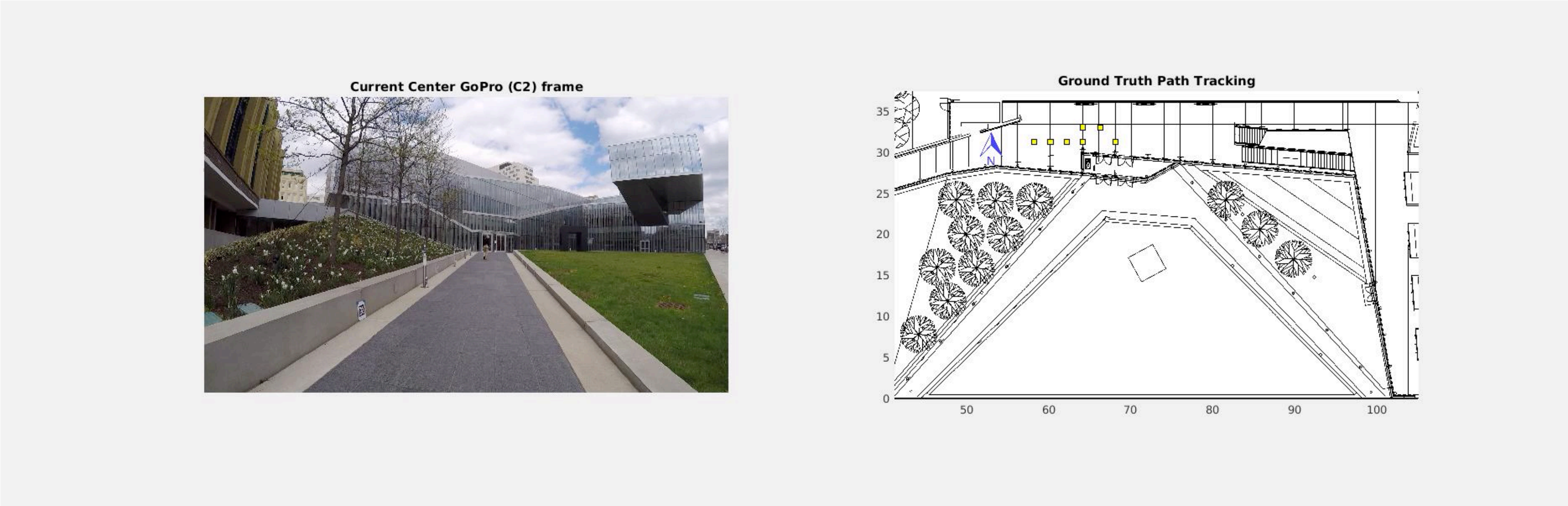
https://grail.cs.washington.edu/rome/ (2009-11)

SfM Application: 3D Presidential Portrait

Really MVS rather than SfM, since the camera locations are fully known.

https://www.youtube.com/watch?v=4GiLAOtjHNo

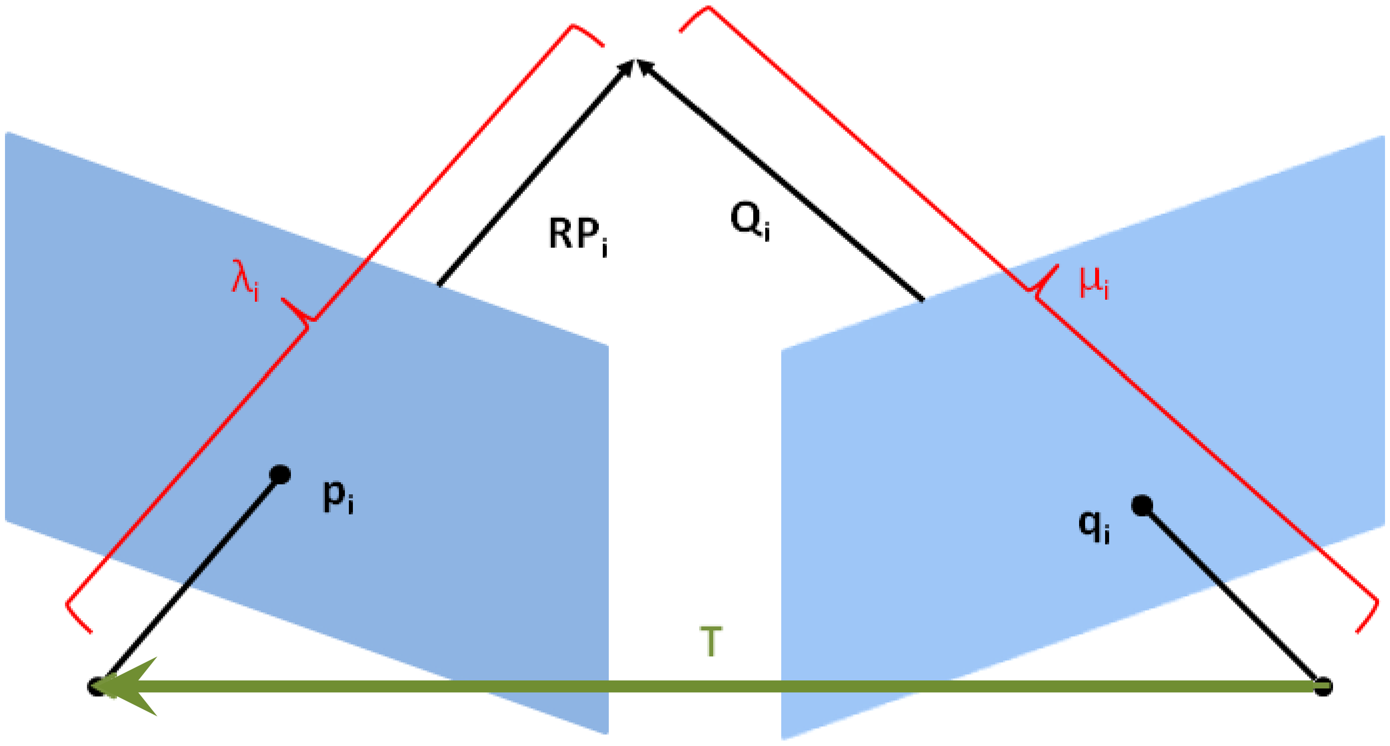
SfM is the main ingredient of Visual Odometry



SLAM for Driving in DARPA Grand Challenge (circa 2007)

Kostas Daniilidis, UPenn 4th place entry in the DARPA Grand challenge 2007

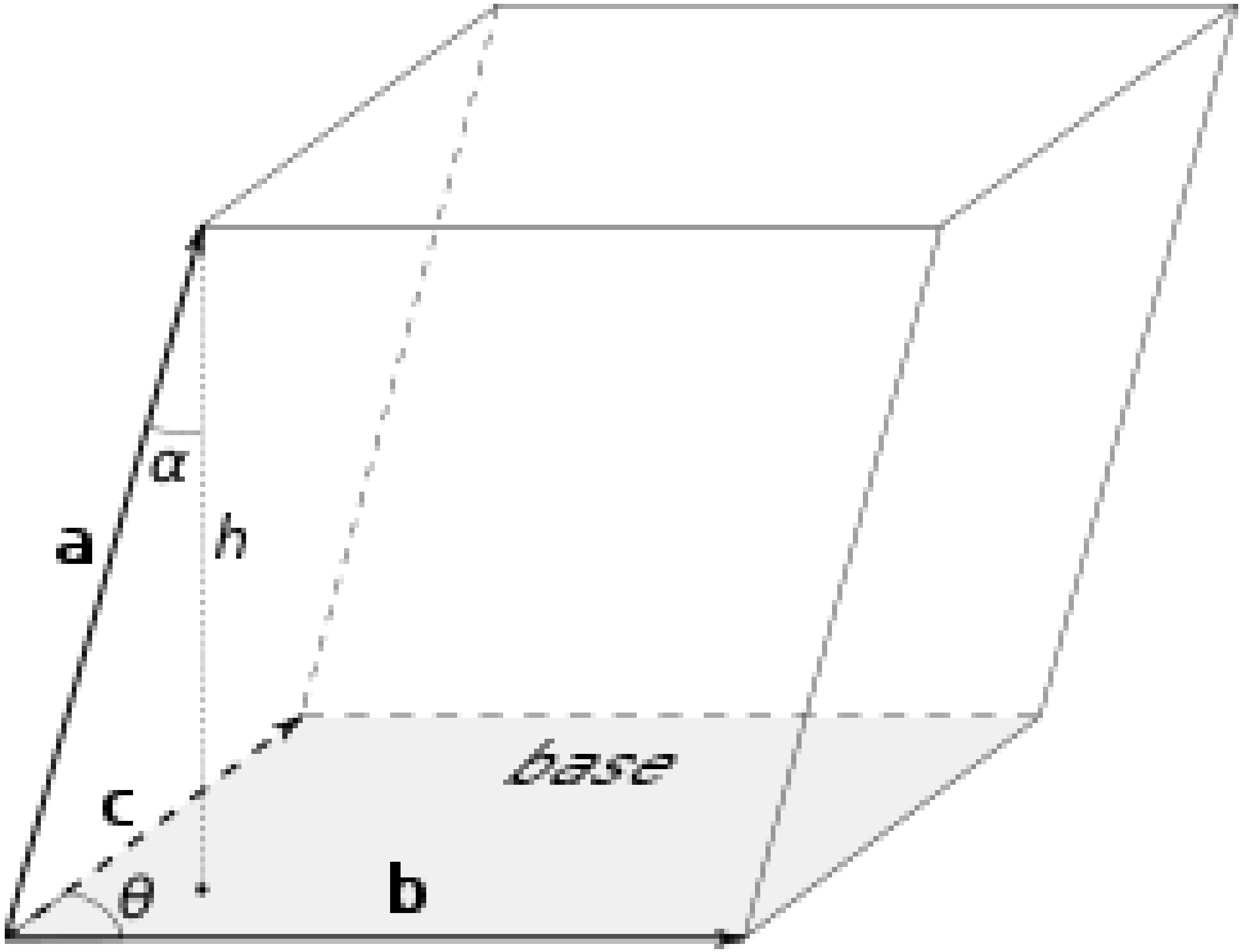
“Epipolar Constraints” Between Two Views of a Scene



We can eliminate the depths from $R(\lambda p) + T = \mu q$ and obtain the epipolar constraint:

$$\boldsymbol{q}_i^T(T \times R\boldsymbol{p}_i) = 0$$

Review: Mixed Product = Volume of Parallelepiped



$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) = \mathbf{b} \cdot (\mathbf{c} \times \mathbf{a}) = \mathbf{c} \cdot (\mathbf{a} \times \mathbf{b})$$

Wikipedia

In our setting, q, Rp, T are edges of a triangle because

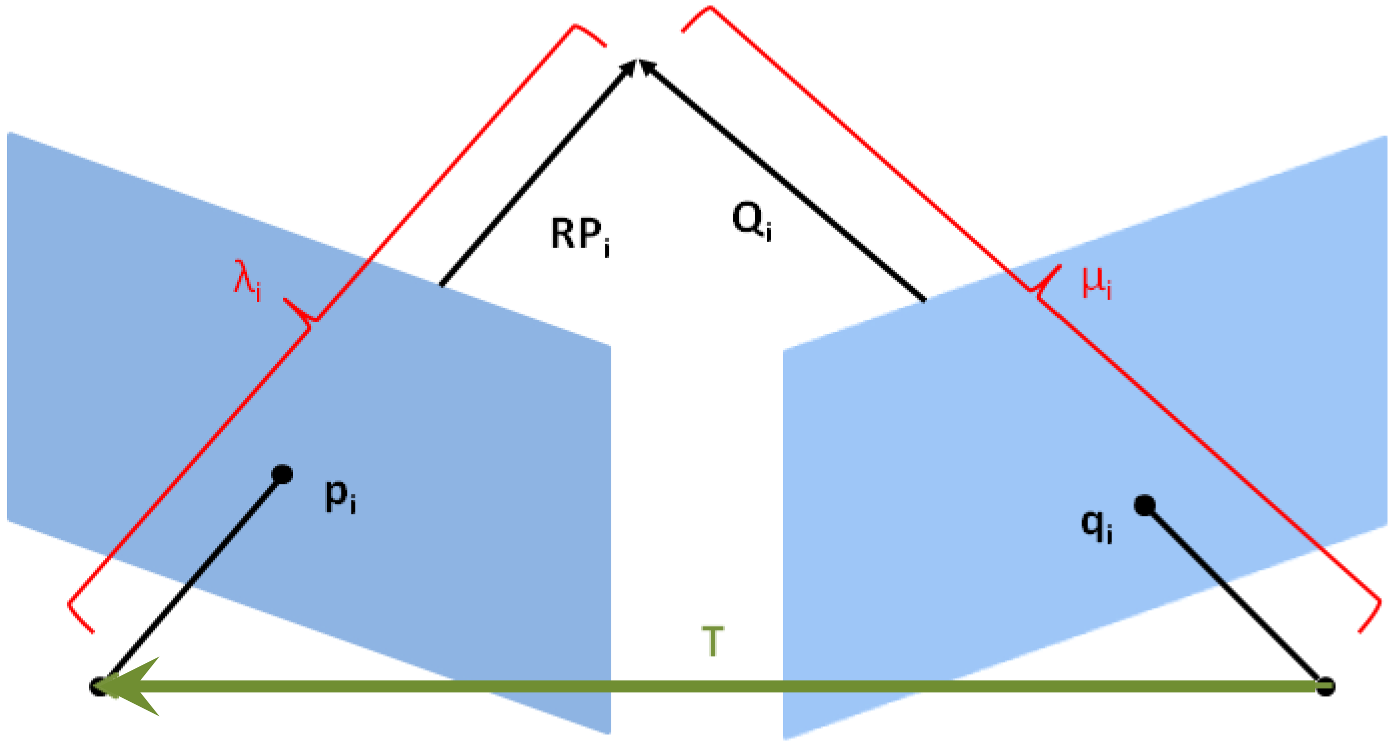
$$\lambda R\boldsymbol{p} + \boldsymbol{T} = \mu \boldsymbol{q}$$

(which is the triangle rule for vectors).

And triangles are planar!

Fun fact: this is also the determinant of a matrix with columns **a**, **b**, **c**.

Geometric Meaning of Epipolar Constraint



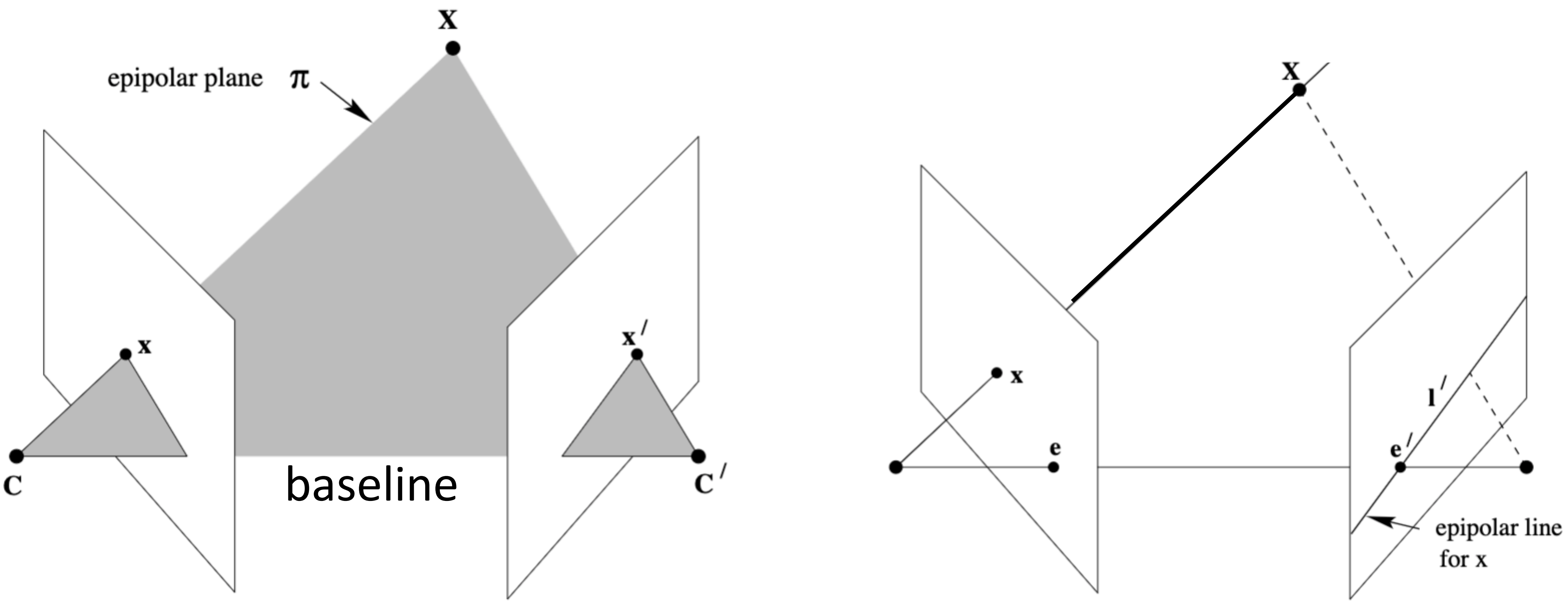
The two rays q and Rp intersect in space if and only if they are coplanar with the translation vector T. Three vectors are coplanar if their mixed product vanishes:

$$q^T(T \times Rp) = 0$$

Epipolar constraint = *"image rays from the world point to the two cameras lies on the same plane as the baseline (translation vector) connecting the camera centers."*

Again, these are just the edges of a triangle with the 2 camera centers and the world point as vertices, so naturally coplanar!

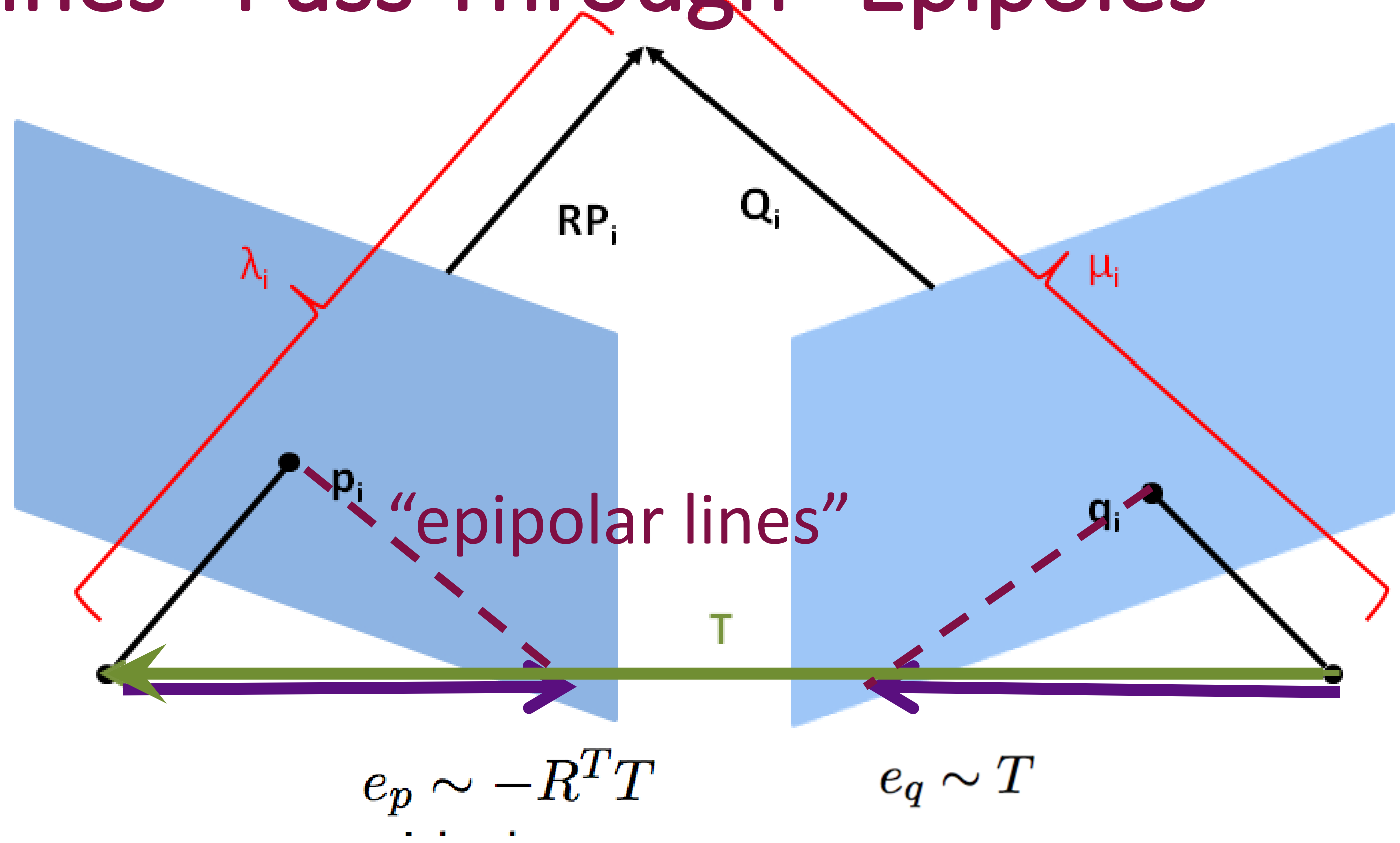
Epipolar “Planes” and “Lines”



Epipolar planes are planes containing the baseline.
Any 3D point induces a corresponding epipolar plane.
Intersection of this plane with an image plane = epipolar line for that 3D point.

ZH Fig 9.1

“Epipolar Lines” Pass Through “Epipoles”



$e_p \sim -R^T\ T$ and $e_q \sim T$ are the “epipoles” = images of the other camera center on each plane = intersections of baseline T with the two planes = VP of the translation direction in each plane.

All epipolar lines in each image plane pass through its epipole.

Epipolar Constraints

$$\boldsymbol{q}_i^T(T \times R\boldsymbol{p}_i) = 0$$

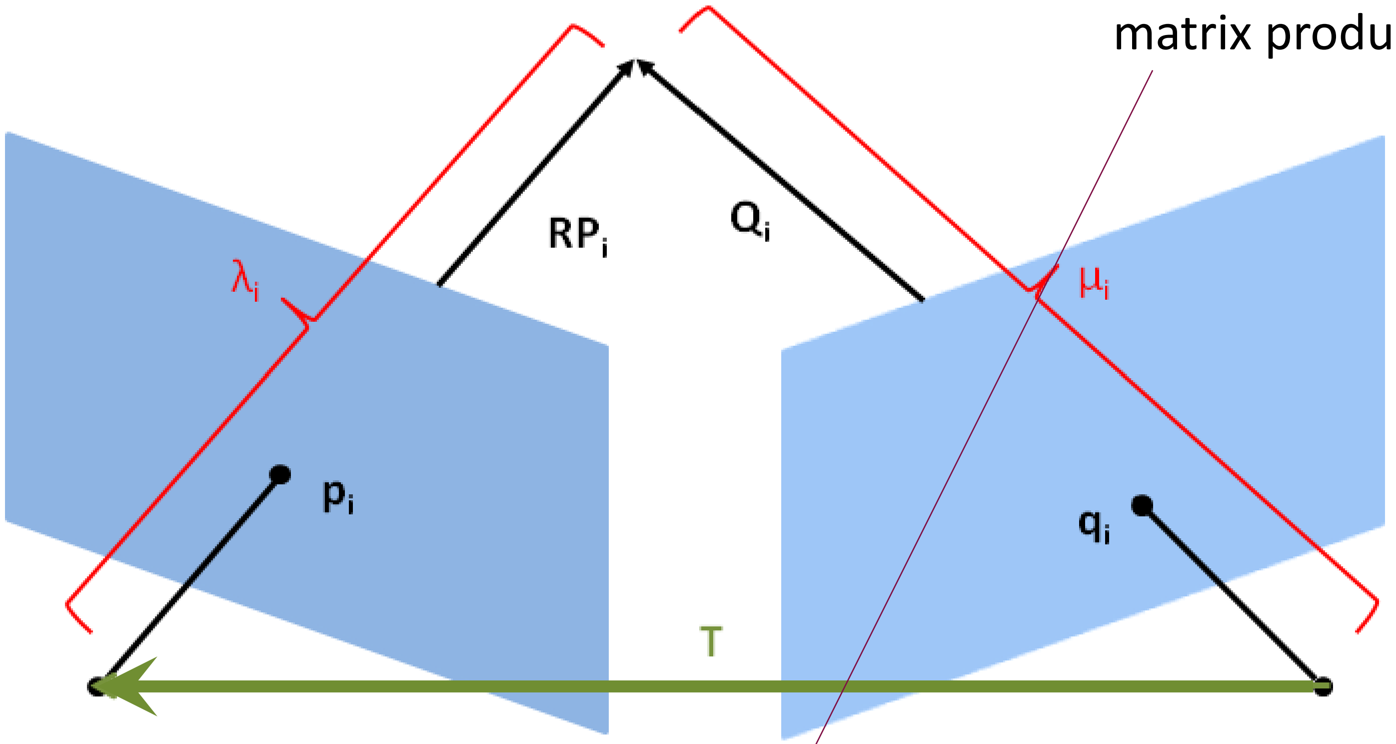
Gives us an equation in unknowns: R, T => route to an SfM solution?

Q: If this were linear in R, T, we could solve it with enough (p_i, q_i) pairs i.e. 2D->2D correspondences. Is it linear though?
A: No, but through a change of variables, we will soon make the equation linear in some new unknowns.

Also note: we are not only interested in R, T (the "motion"). We also want to solve for the depths λ_i, μ_i afterwards (the "structure").

Epipolar Constraints

Want to solve for R, T.
Can we make this look more like standard linear equations containing matrix products etc.?



We can eliminate the depths from $R(\lambda p) + T = \mu q$ and obtain the epipolar constraint:

$$q^T (T \times Rp) = 0$$

Review: Cross-products through skew-symmetric matrices

The vector cross product also can be expressed as the product of a skew-symmetric matrix and a vector:

$$\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_{\times} \mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Sometimes written as $\hat{a}$. Remember, this is now a 3×3 matrix, while original a, b were 3x1 vectors.

Wikipedia on cross product

The Essential Matrix E

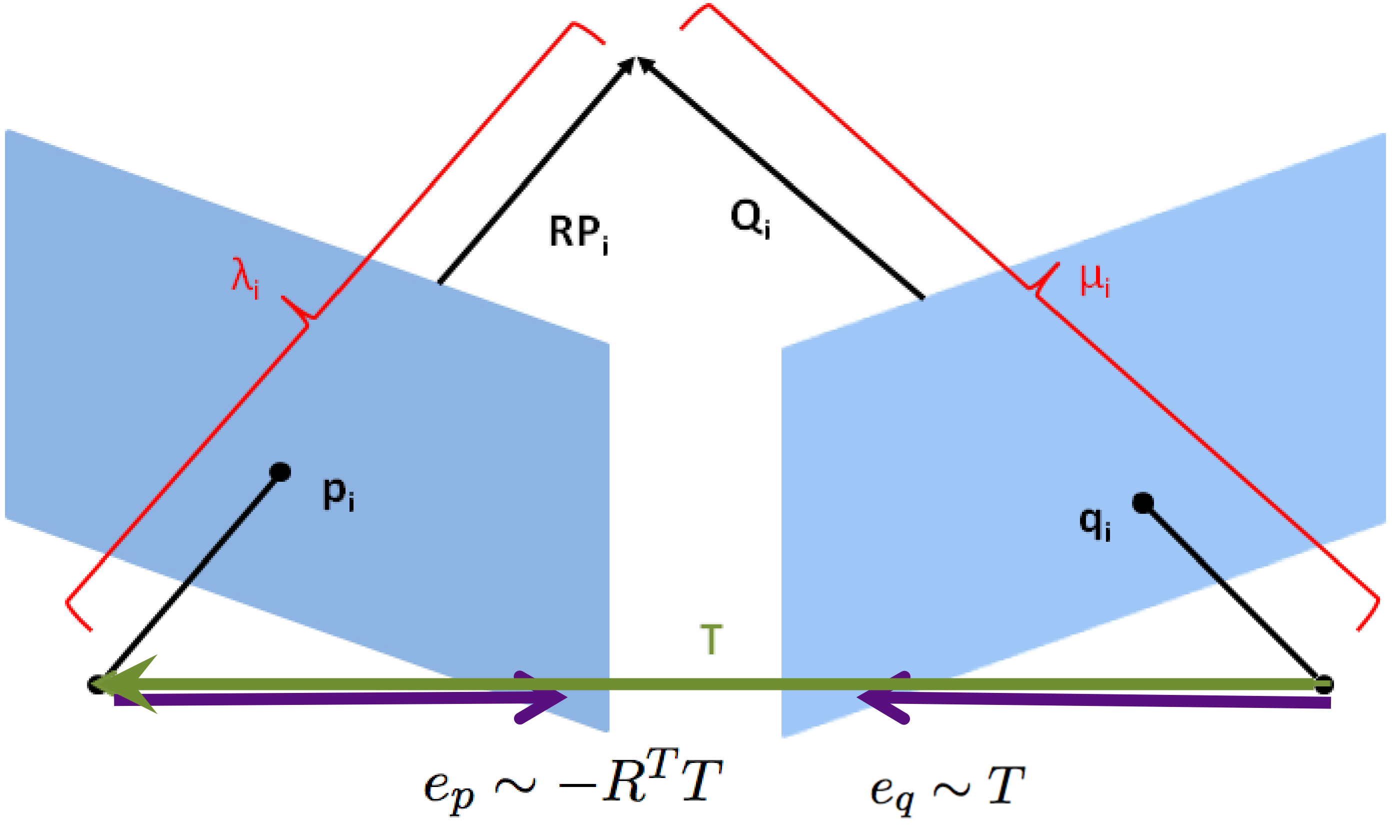
We had: $\boldsymbol{q}_i^T(T \times R\boldsymbol{p}_i) = 0$

$$\Rightarrow \boldsymbol{q}_i^T(\hat{T}R)\ \boldsymbol{p}_i = 0$$

Renaming $E = (\hat{T}R)$:

$$\boldsymbol{q}_i^T E\ \boldsymbol{p}_i = 0$$

"Essential matrix"



Now linear in the new unknowns $E_{3\times3}$! But will need to recover $T_{3\times1}$, $R_{3\times3}$ later.

Is the epipolar constraint really linear in E?

$q_i^T E_{3\times3} p_i = 0$ is a single equation that is linear in the elements of E

Can write this out explicitly as below.

If

$$E = \begin{pmatrix} e_1 & e_2 & e_3 \end{pmatrix}$$

then epipolar constraint can be rewritten as

$$\begin{aligned} q^T \begin{pmatrix} e_1 & e_2 & e_3 \end{pmatrix} \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} &= q^T \left(p_x e_1 + p_y e_2 + p_z e_3 \right) \\ &= \begin{pmatrix} p_x q^T & p_y q^T & p_z q^T \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix} = 0 \end{aligned}$$

This equation is linear

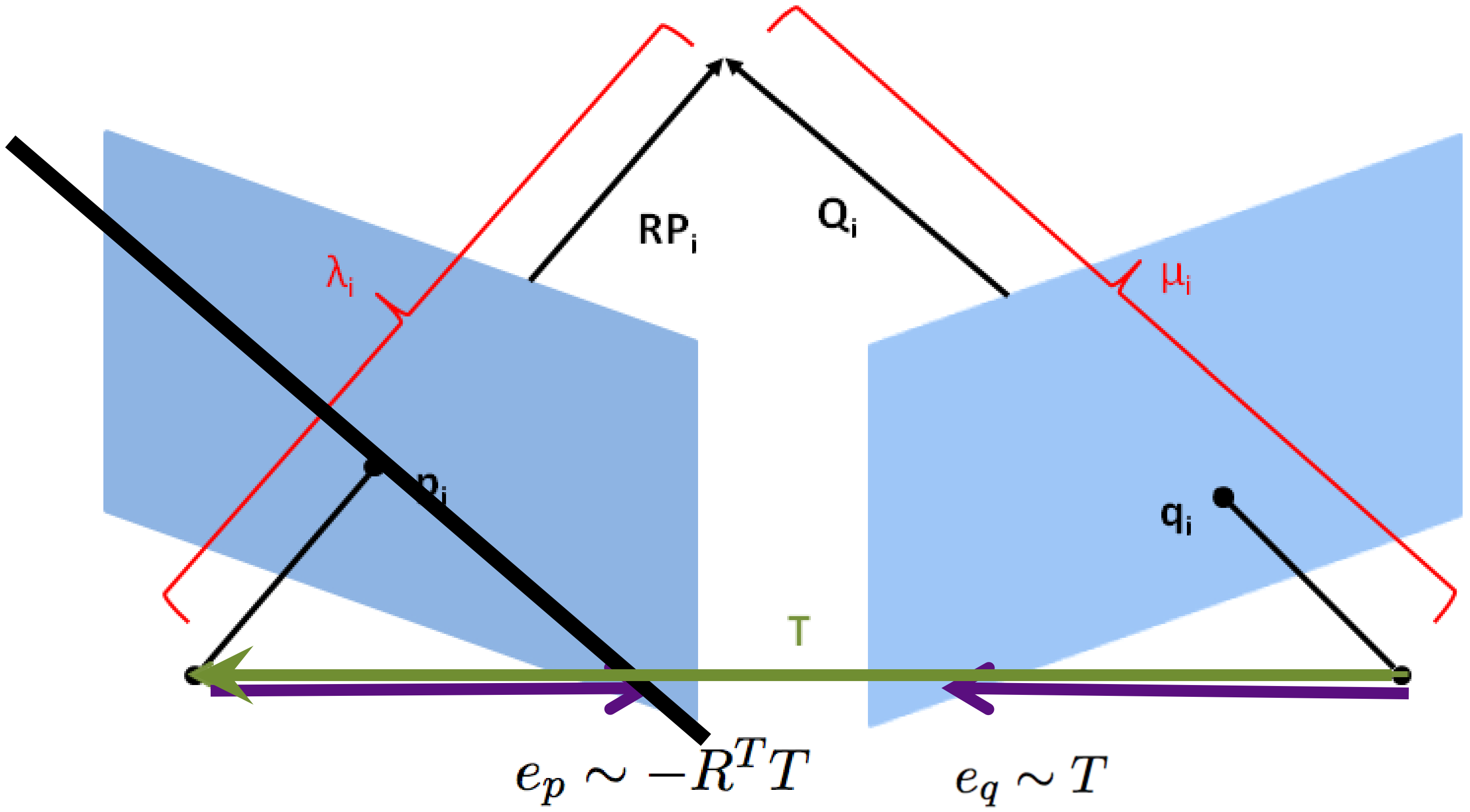
Longuet-Higgins 1981

Fundamental matrix

- Essential matrix E connects image plane coordinates i.e. "calibrated coordinates" for points $p = K_p^{-1}(u_p, v_p, 1)^T$ and $q = K_q{}^{-1}(u_q, v_q, 1)^T$.
- Fundamental matrix F assumes that we are still in "pixel land"
- So:

$$q^T E p = 0 \qquad \text{can be written as:} \qquad (K_q q)^T F (K_p p) = 0$$

Epipolar Lines in Essential Matrix Notation

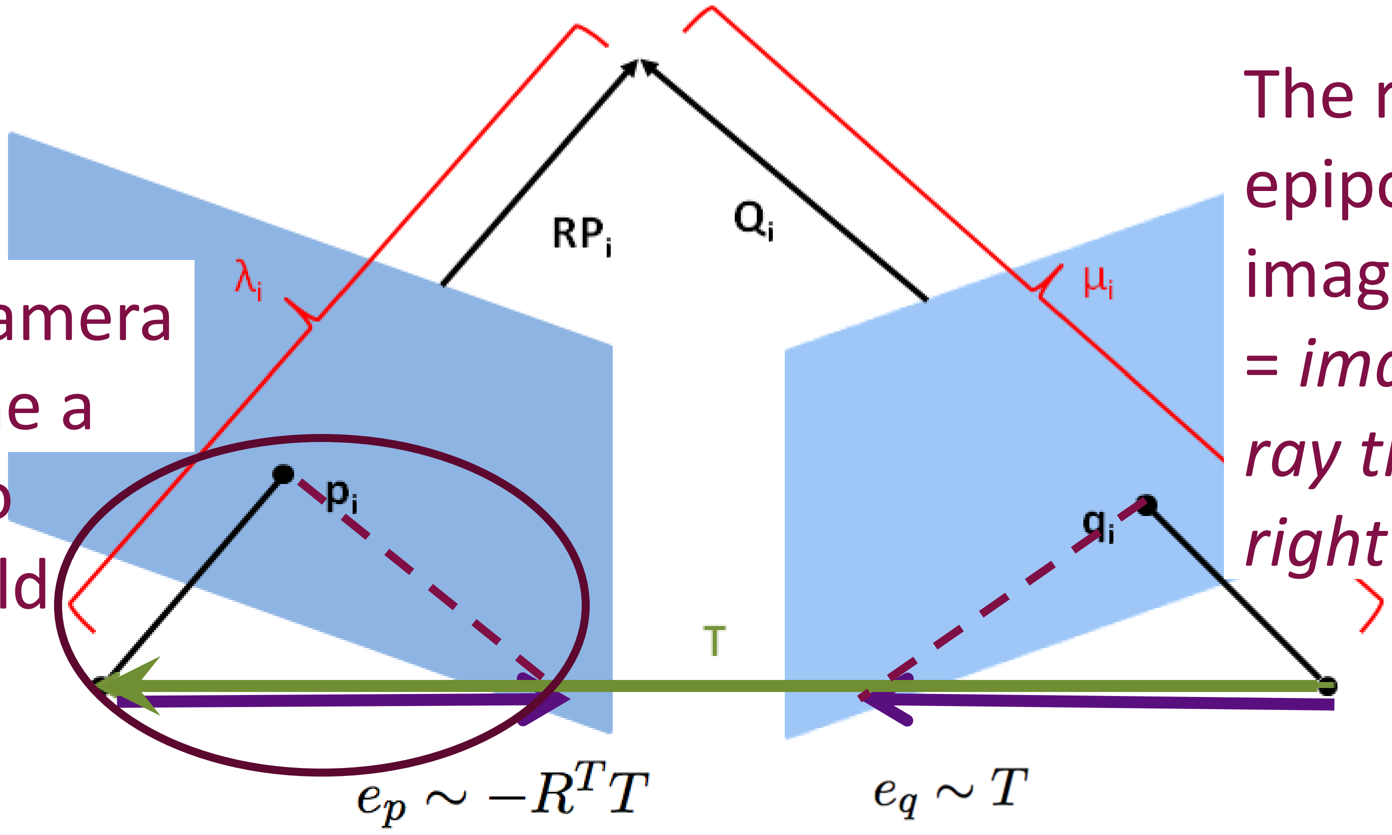


Equation $q^T E p = 0$ is a line equation in the p-plane with line coefficients $E^T q$. It is called the epipolar line in p-plane.

Exercise: What about in q-plane?

Epipolar Lines Constrain Point Correspondences!

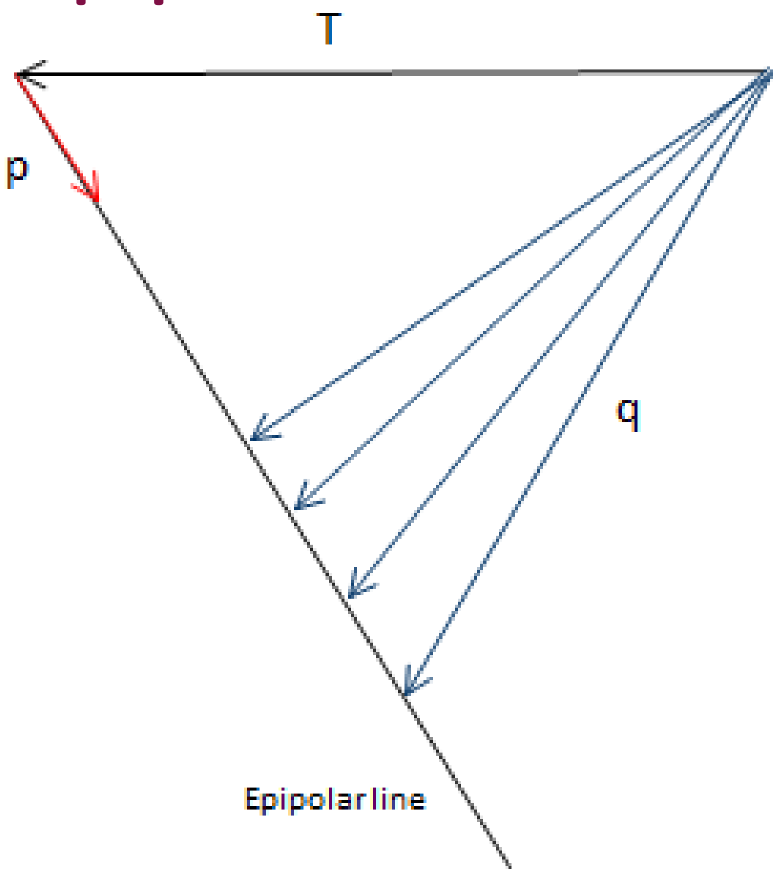
p_i and the two camera centers determine a plane (which also contains the world point P_i)

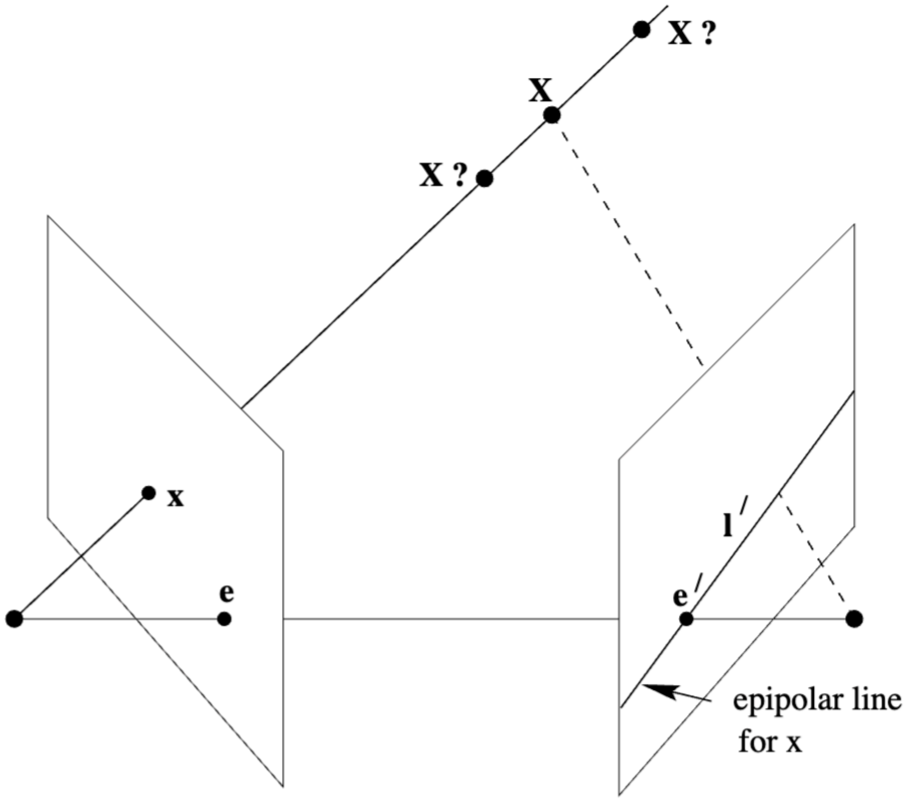


The relevant right epipolar line is the image of this plane = *image of the left ray through the right camera.*

Given a point p = left image of a world point, the right image of that point is constrained to lie on the corresponding right epipolar line, and vice versa. (depending on depth of the world point).

Epipolar Lines Constrain Point Correspondences!

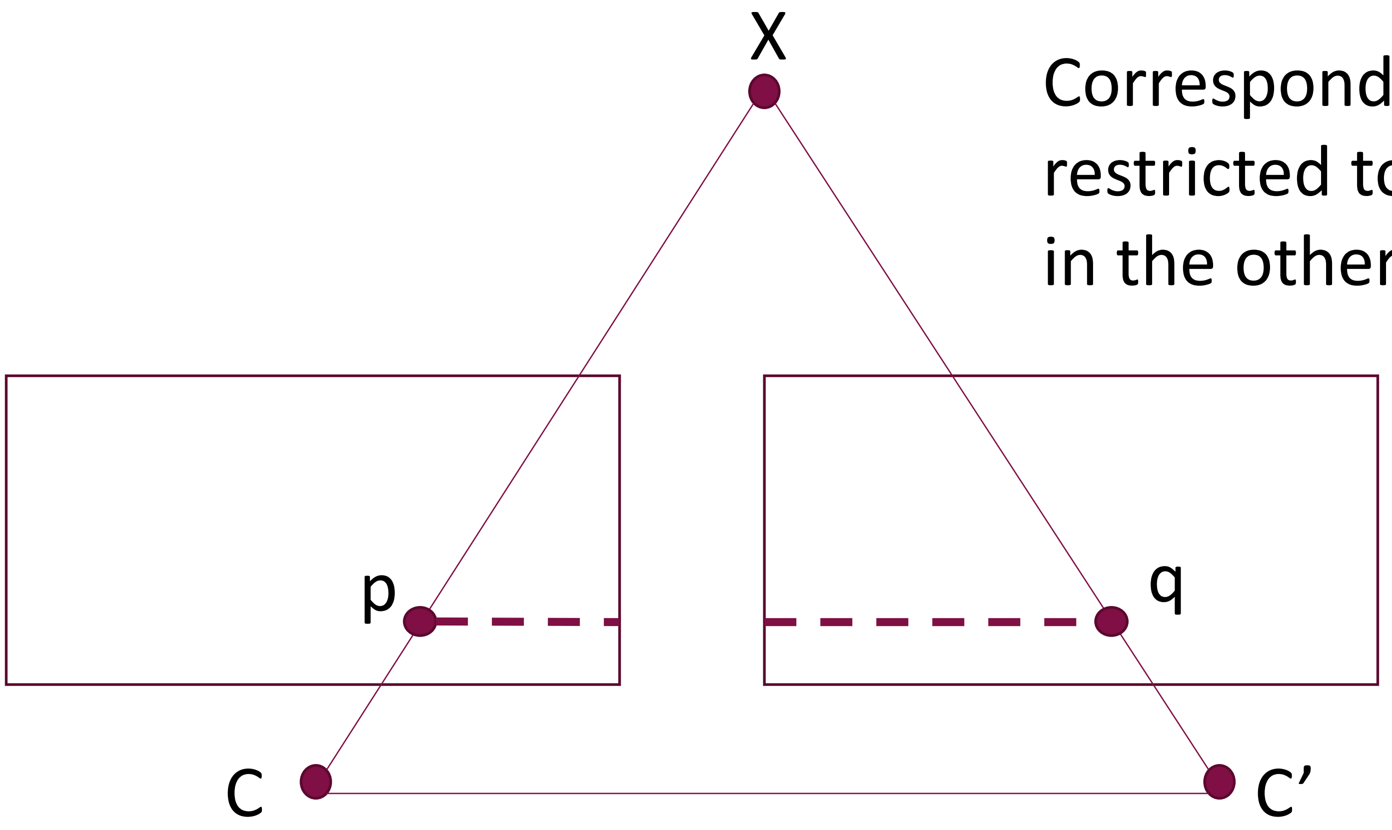




Knowledge of the E-matrix allows us to search for points q corresponding to points p along the epipolar line, reducing correspondence to $1D$-search.

Position of the corresponding point q along epipolar line varies with depth of the 3D points which is still constrained to lie on the ray through p.

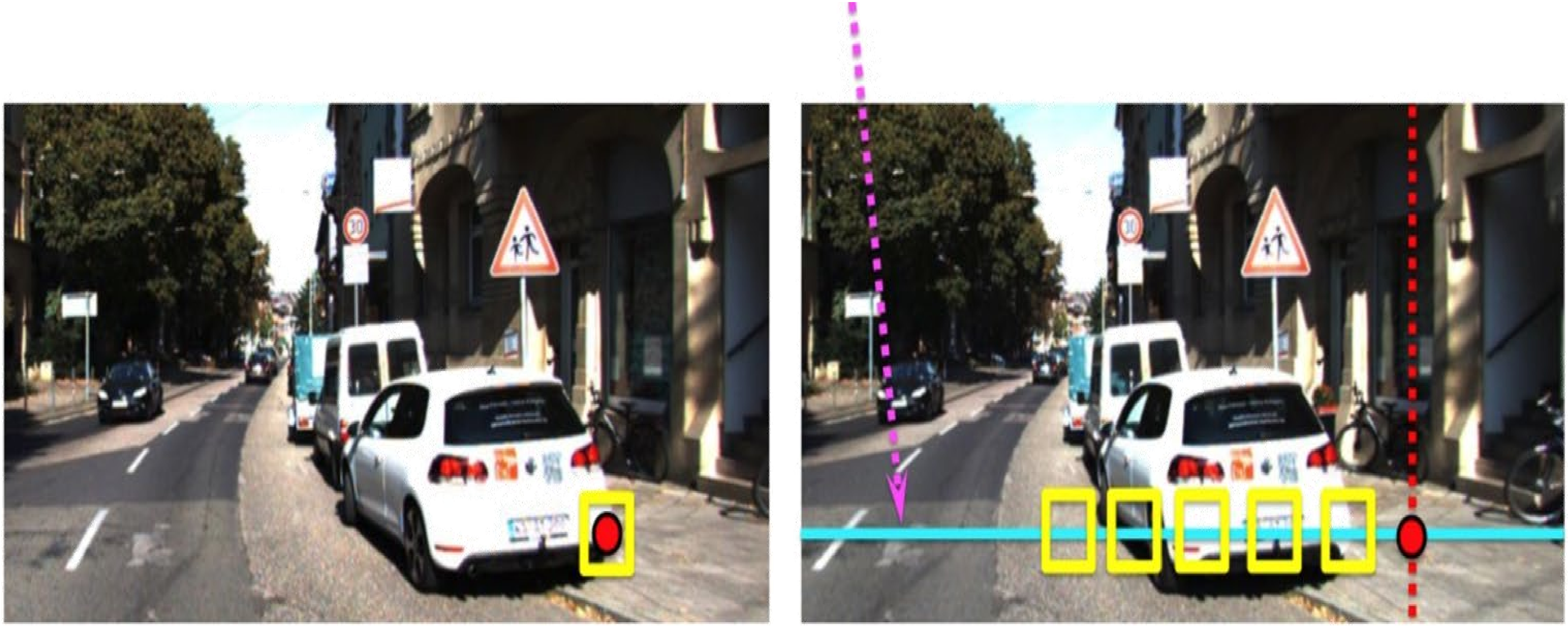
Special case: “frontoparallel” / “parallel stereo” cameras



Correspondences are restricted to the “same” row in the other image!

Q: Where is the epipole = image of other camera center in each image?
A: Baseline is parallel to image plane(s) => epipole is at infty along x, and epipolar line is horizontal.

What if the cameras are “frontoparallel” / “parallel stereo”?



left image right image

Collecting Epipolar Lines into Epipolar “Pencils”

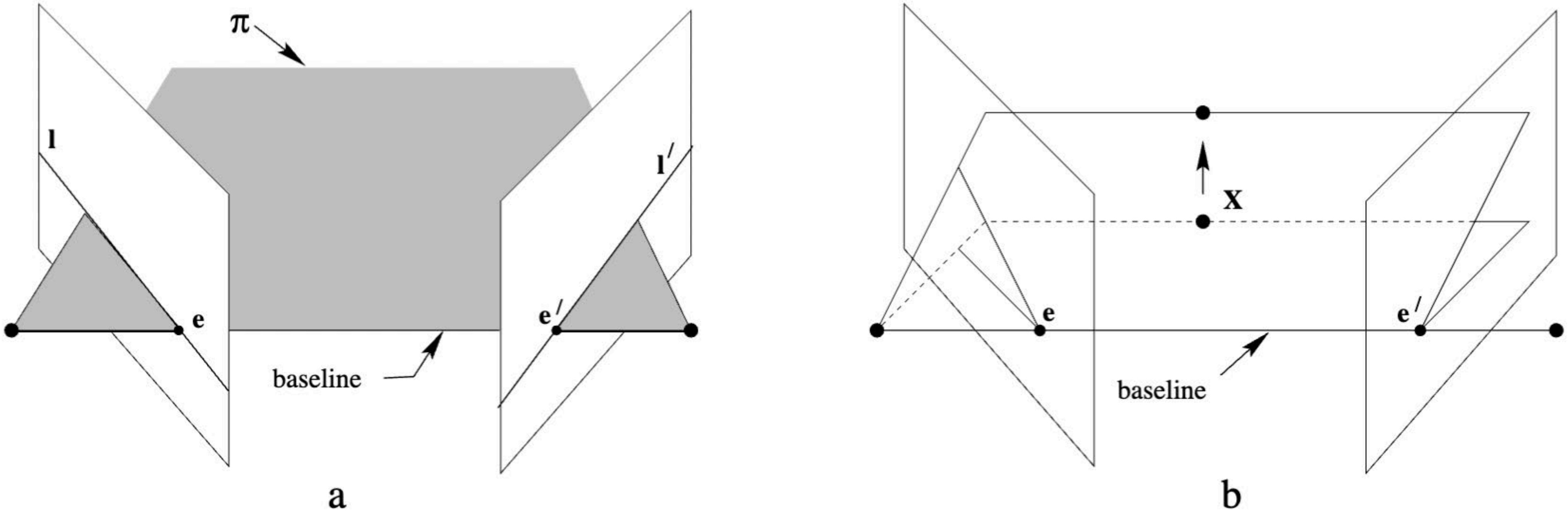
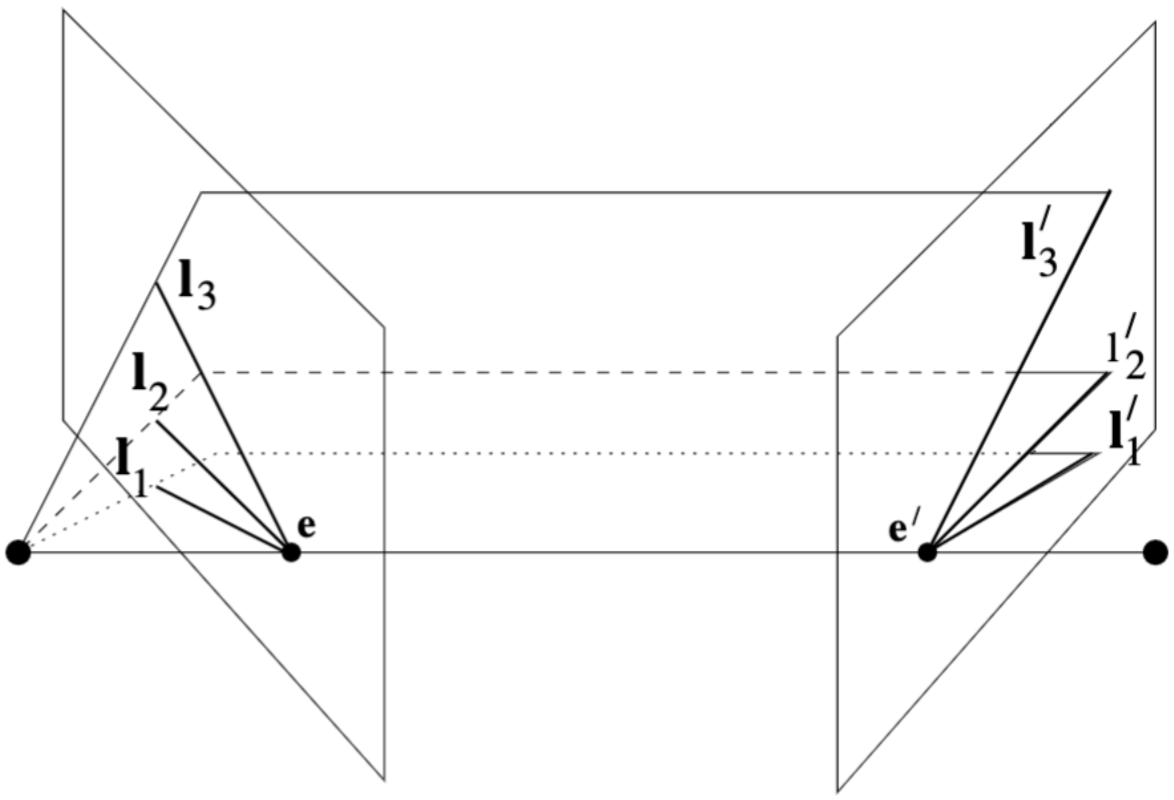


Fig. 9.2. **Epipolar geometry.** *(a) The camera baseline intersects each image plane at the epipoles* **e** *and* **e**′. *Any plane* π *containing the baseline is an epipolar plane, and intersects the image planes in corresponding epipolar lines* **l** *and* **l**′. *(b) As the position of the 3D point* **X** *varies, the epipolar planes “rotate” about the baseline. This family of planes is known as an epipolar pencil. All epipolar lines intersect at the epipole.*

Epipolar "Pencils"



The epipolar lines in each image plane form a "pencil" whose tip is the epipole i.e. image of the other camera center / intersection of baseline with image plane. (Because all epipolar planes contain the baseline, after all)

Epipolar Pencils of Frontoparallel Cameras

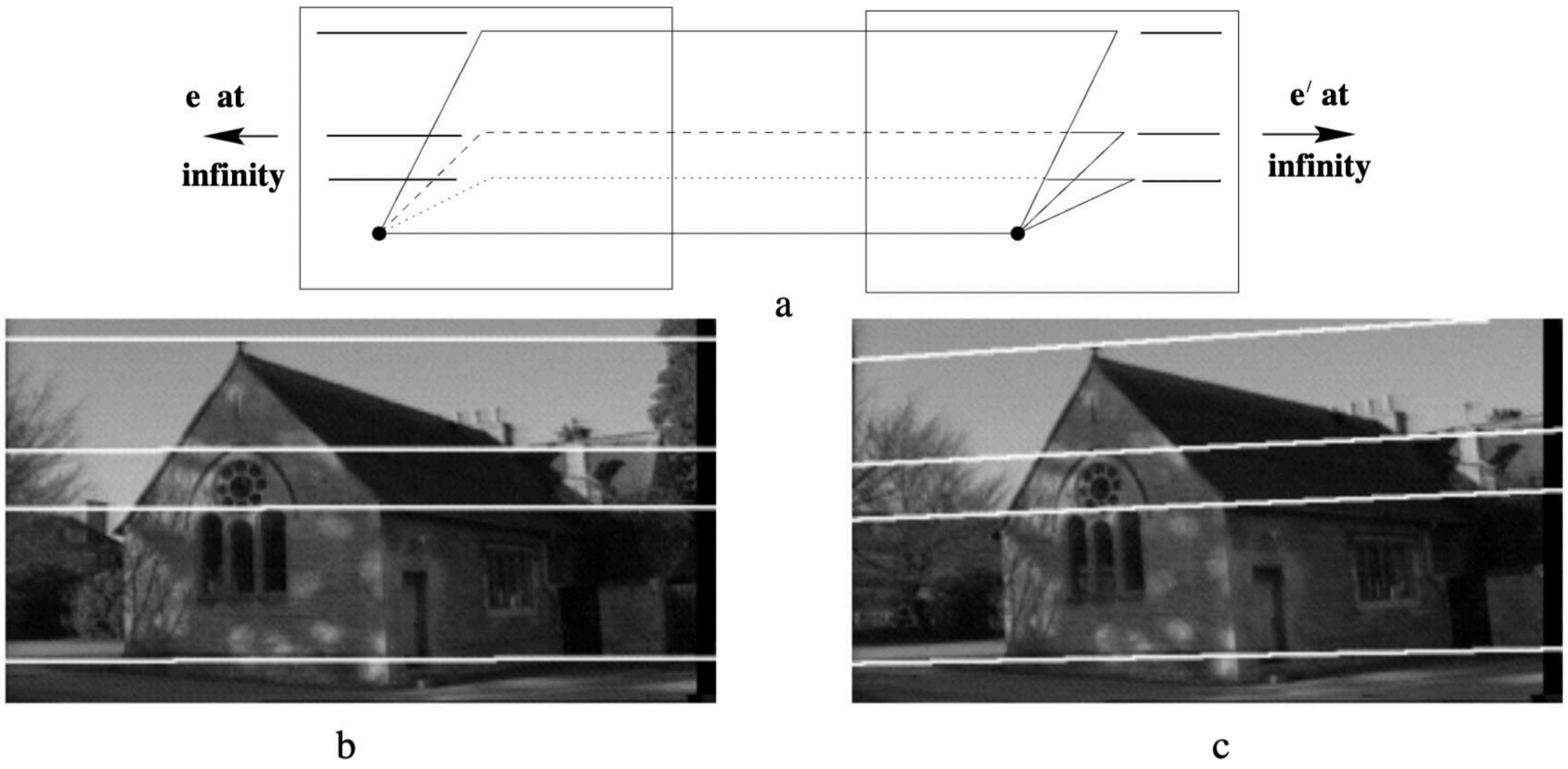
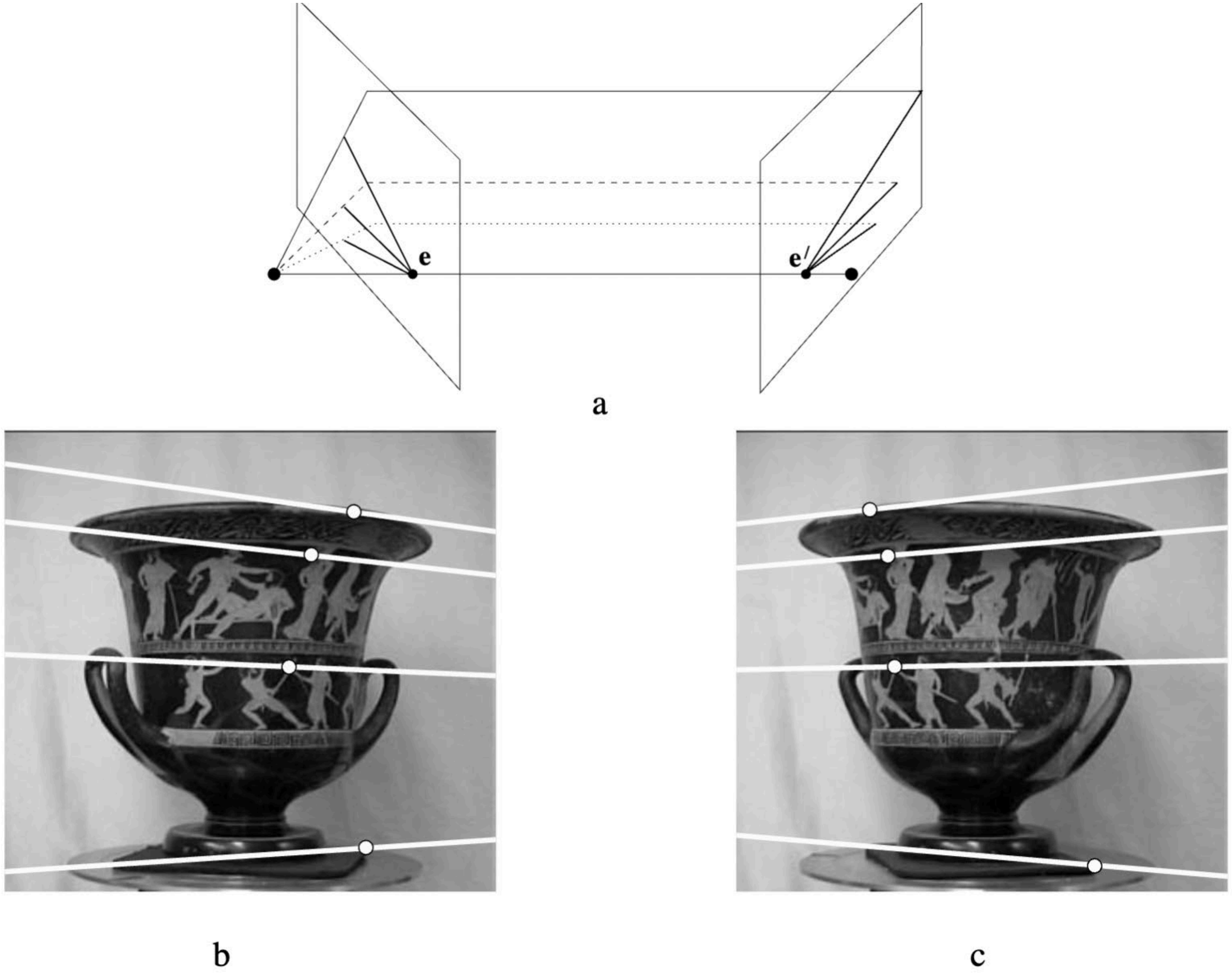


Fig. 9.4. **Motion parallel to the image plane.** *In the case of a special motion where the translation is parallel to the image plane, and the rotation axis is perpendicular to the image plane, the intersection of the baseline with the image plane is at infinity. Consequently the epipoles are at infinity, and epipolar lines are parallel. (a) Epipolar geometry for motion parallel to the image plane. (b) and (c) a pair of images for which the motion between views is (approximately) a translation parallel to the x-axis, with no rotation. Four corresponding epipolar lines are superimposed in white. Note that corresponding points lie on corresponding epipolar lines.*

Epipolar Pencils visually identify camera orientations



Epipolar Pencils visually identify camera orientations

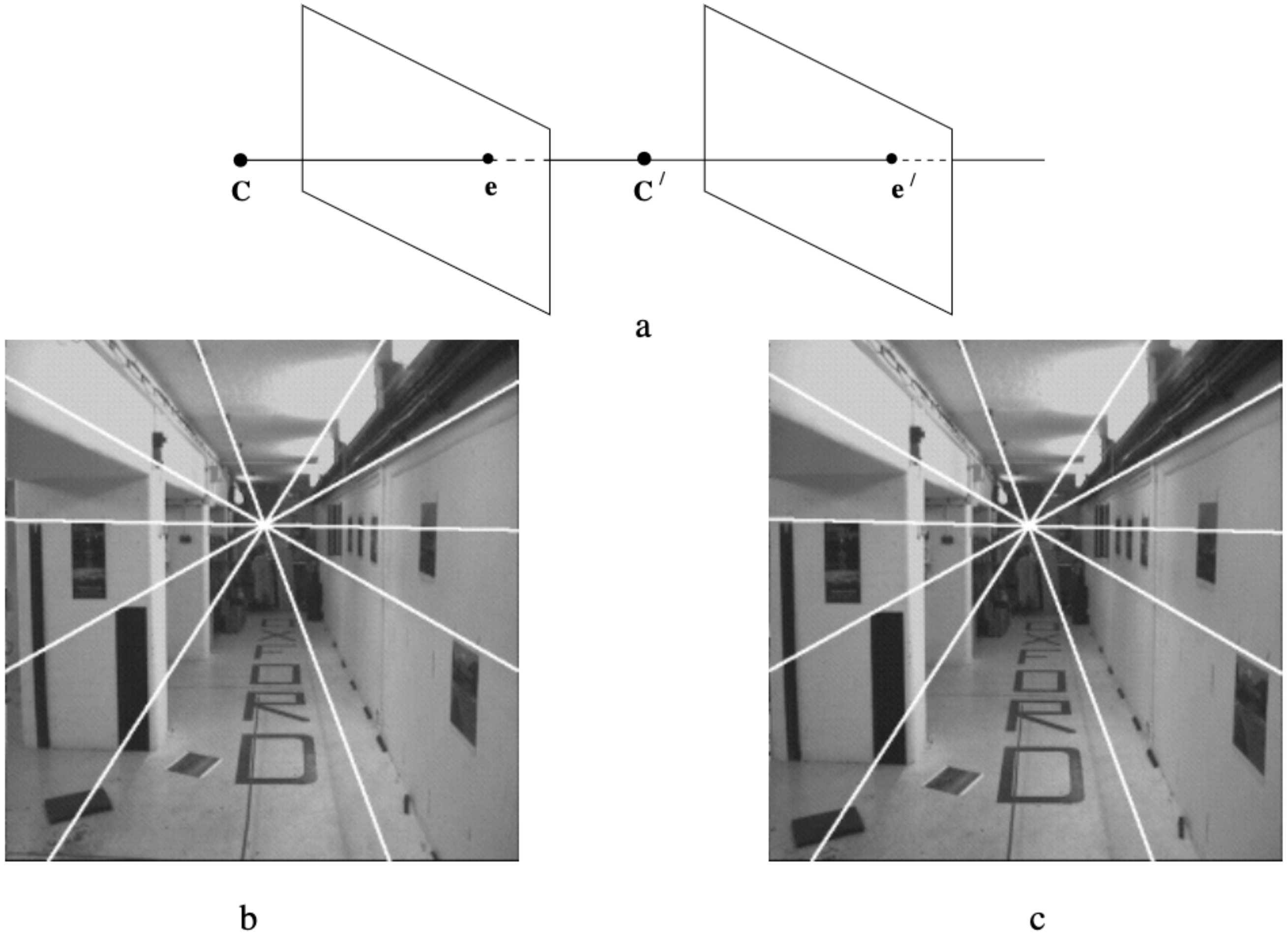


Fig. 9.8. **Pure translational motion.** *(a) under the motion the epipole is a fixed point, i.e. has the same coordinates in both images, and points appear to move along lines radiating from the epipole. The epipole in this case is termed the* Focus of Expansion *(FOE). (b) and (c) the same epipolar lines are overlaid in both cases. Note the motion of the posters on the wall which slide along the epipolar line.*

Summary so far (Essential matrix and Epipolar Stuff)

- Essential matrix contains information about R, T between two views, provides “epipolar constraint” equations that might help identify E and eventually solve the SfM problem.

- Geometric intuitions connected to epipoles, epipolar planes, lines, pencils. Just visualizing epipolar pencils already gives us qualitative information about the relative orientations of the two camera views, hinting at the utility of these concepts for SfM.

- Epipolar lines directly constrain point correspondences, will be useful for later on in the course.