6.891

Computer Vision and Applications

Prof. Trevor. Darrell

Lecture 9: Affine SFM

- Geometric Approach
- Algebraic Approach
- Tomasi/Kanade Factorization

Readings: F&P Ch. 12; (except 12.1 is optional)

Lecture	Date	Description	Readings	Assignments	Mater
1	2/3	Course Introduction Cameras, Lenses and Sensors	Req: FP 1.1, 2.1, 2.2, 2.3, 3.1, 3.2	PS0 out	
2	2/5	Image Filtering	Req: FP 7.1 - 7.6		
3	2/10	Image Representations: pyramids	Req: FP 7.7, 9.2		
4	2/12	Texture	Req: FP 9.1, 9.3, 9.4	PS0 due	
	2/17	Monday Classes Held (NO LECTURE)			
5	2/19	Color	Req: FP 6.1-6.4	PS1 out	
6	2/24	Local Features			
7	2/26	Multiview Geometry	Req: FP 10	PS1 due	
8	3/2	Multiview Geometry II			
9	3/4	Affine Reconstruction	FP 12, except 12.1	PS2 out	
	3/9	Projective Reconstruction	FP 1		
11	3/11	Model-based Object Recognitio		PS2 due	
12	3/16	Project Previews		EX1 out	
13	3/18	(no class -- Horn lecture on 3/10 instead)		EX1 due	
	3/23-3/25	Spring Break (NO LECTURE)			

3/10

Horn Lecture
D.I Seminar
Wed 1pm
NE43-8th fl.

Horn Lecture: *Perspective Projection Properly Models Image Formation*

Date: 3-10-2004 Time: 1:00 PM - 2:00 PM Location: NE43-814

Methods based on projective geometry have become popular in machine vision because they lead to elegant mathematics, and easy-to-solve linear equations.

It is often not realized that one pays a heavy price for this convenience. Such methods do not correctly model the physics of image formation, require more correspondences, and are considerably more sensitive to measurement error than methods based on true perspective projection.

In this talk we find that for the example of exterior orientation: (i) Methods based on projective geometry are fundamentally different from methods based on perspective projection; (ii) Methods based on projective geometry yield a transformation matrix T that in general does not correspond to a physical imaging situation that is, a rotation, translation and perspective projection; (iii) Optimization methods based on the real physical imaging equations (true perspective projection) produce considerably more accurate results.

Last Time

Instantaneous Essential Matricies

Fundamental Matrix and the 8-point algorithm

Tri-focal geometry

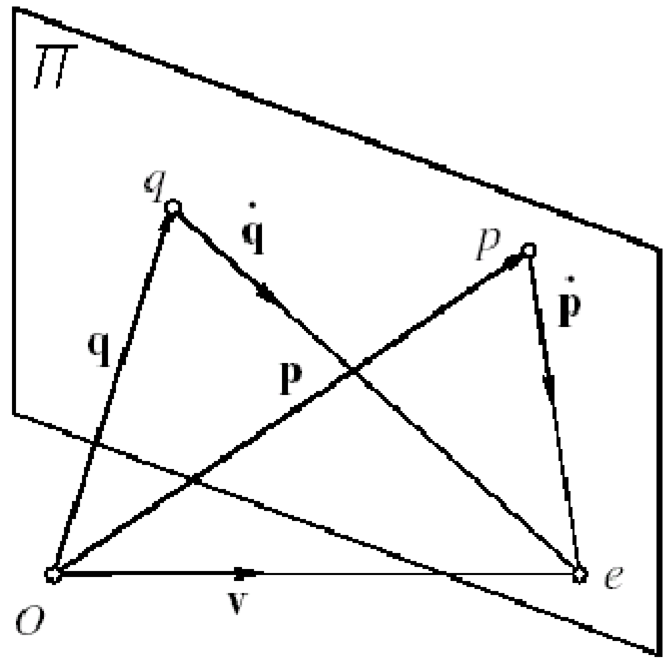
Translating Camera

$$p^T([v_\times][\omega_\times])p - (p \times \dot{p}).v = 0$$

$$\omega = 0$$

$$(p \times \dot{p}).v = 0$$

p, $\dot{p}$, and v are coplanar



Focus of expansion (FOE): Under pure translation, the motion field at every point in the image points toward the focus of expansion

Fundamental matrix

Essential matrix for points on normalized image plane,

$$\hat{p}^T \mathcal{E} \hat{p}' = 0$$

assume unknown calibration matrix:

yields:

$$p = K\hat{p}$$

$$\boxed{\boldsymbol{p}^T \mathcal{F} \boldsymbol{p}' = 0} \qquad \mathcal{F} = \mathcal{K}^{-T} \mathcal{E} \mathcal{K}'^{-1}$$

The 8 point algorithm

8 corresponding points, 8 equations.

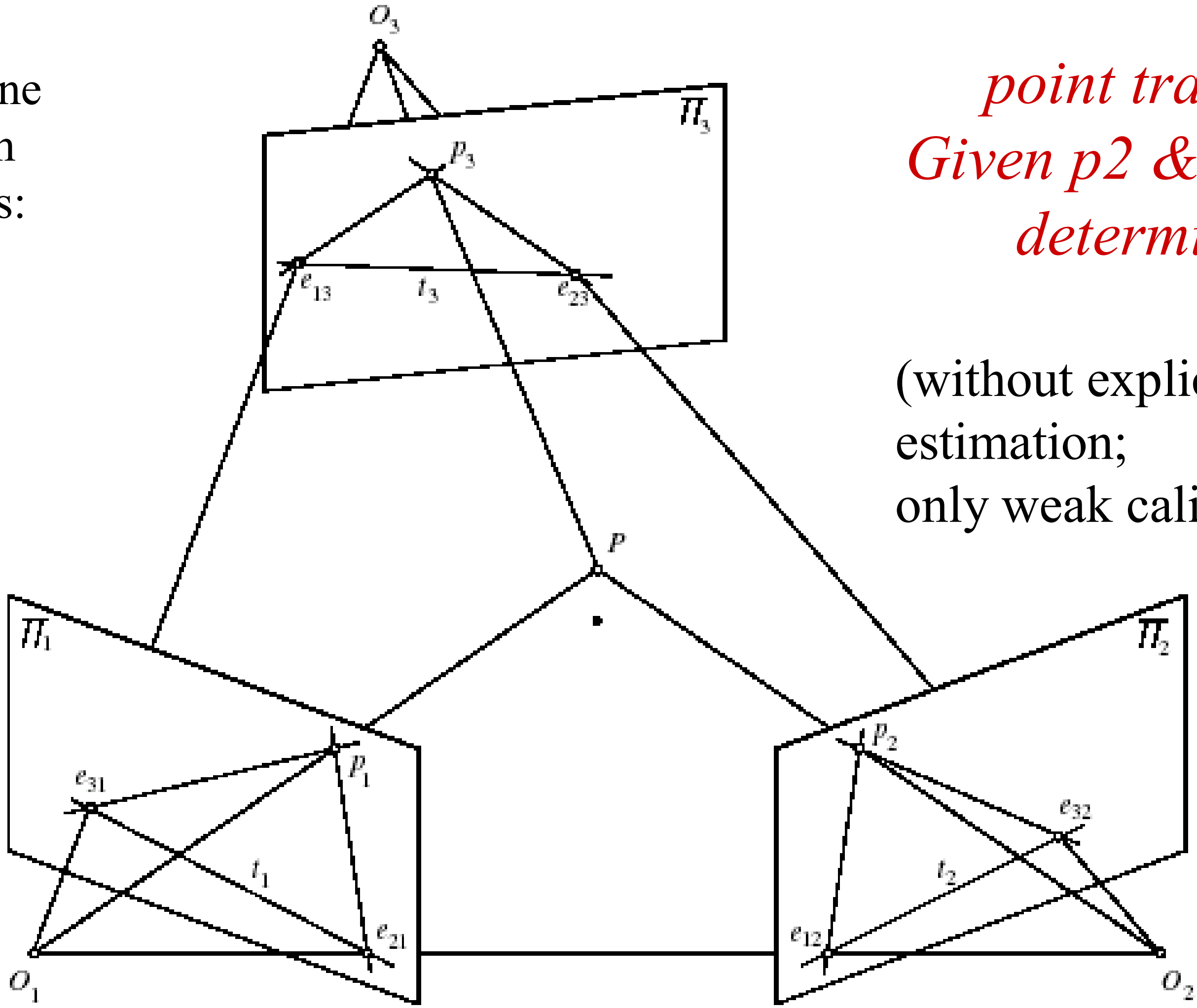
$$\begin{pmatrix} u_1u_1' & u_1v_1' & u_1 & v_1u_1' & v_1v_1' & v_1 & u_1' & v_1' \\ u_2u_2' & u_2v_2' & u_2 & v_2u_2' & v_2v_2' & v_2 & u_2' & v_2' \\ u_3u_3' & u_3v_3' & u_3 & v_3u_3' & v_3v_3' & v_3 & u_3' & v_3' \\ u_4u_4' & u_4v_4' & u_4 & v_4u_4' & v_4v_4' & v_4 & u_4' & v_4' \\ u_5u_5' & u_5v_5' & u_5 & v_5u_5' & v_5v_5' & v_5 & u_5' & v_5' \\ u_6u_6' & u_6v_6' & u_6 & v_6u_6' & v_6v_6' & v_6 & u_6' & v_6' \\ u_7u_7' & u_7v_7' & u_7 & v_7u_7' & v_7v_7' & v_7 & u_7' & v_7' \\ u_8u_8' & u_8v_8' & u_8 & v_8u_8' & v_8v_8' & v_8 & u_8' & v_8' \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \end{pmatrix} = - \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Invert and solve for $\mathcal{F}$.

(Use more points if available; find least-squares solution to minimize $\sum_{i=1}^{n} (\boldsymbol{p}_i^T \mathcal{F} \boldsymbol{p}_i')^2$)

Trinocular epipolar geometry

Trifocal plane formed from trifocal lines:



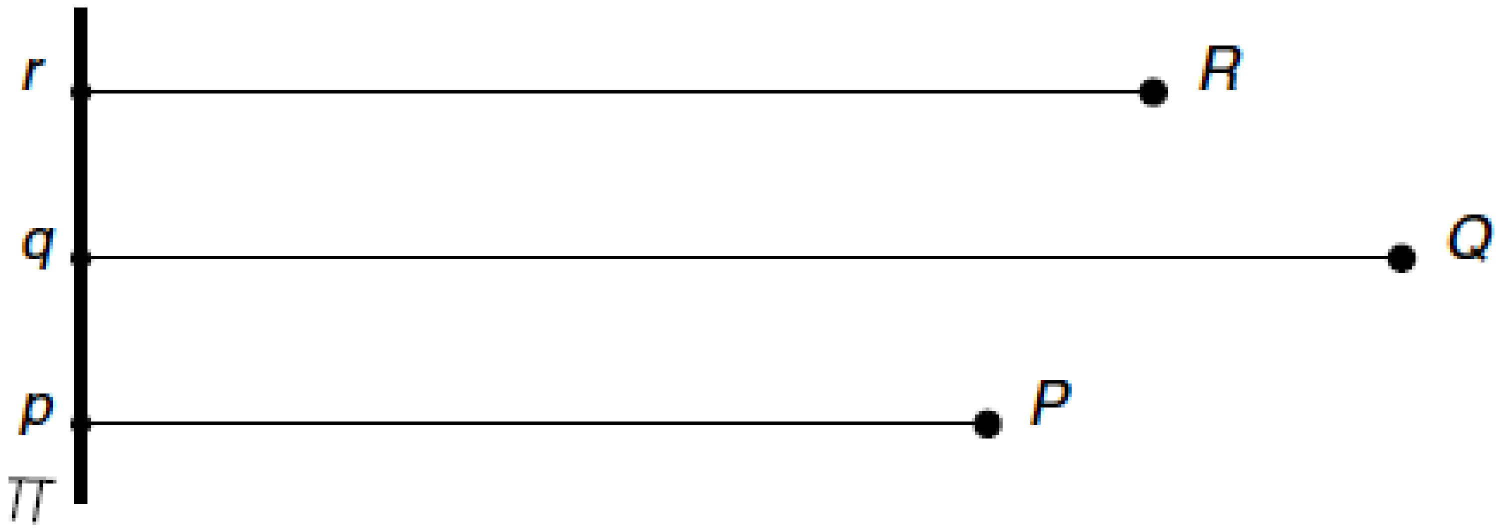
point transfer: Given p2 & p3, p1 is determined!

(without explicit depth estimation; only weak calibration)

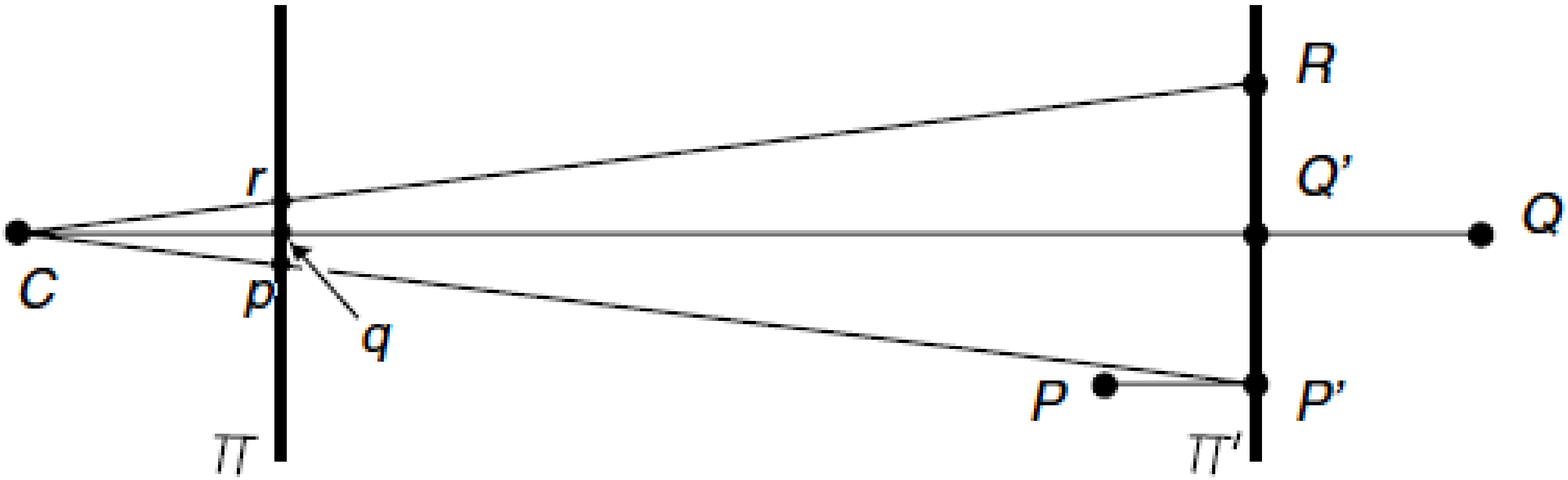
Today

Affine SFM
- Geometric Approach
- Algebraic Approach
- Tomasi/Kanade Factorization

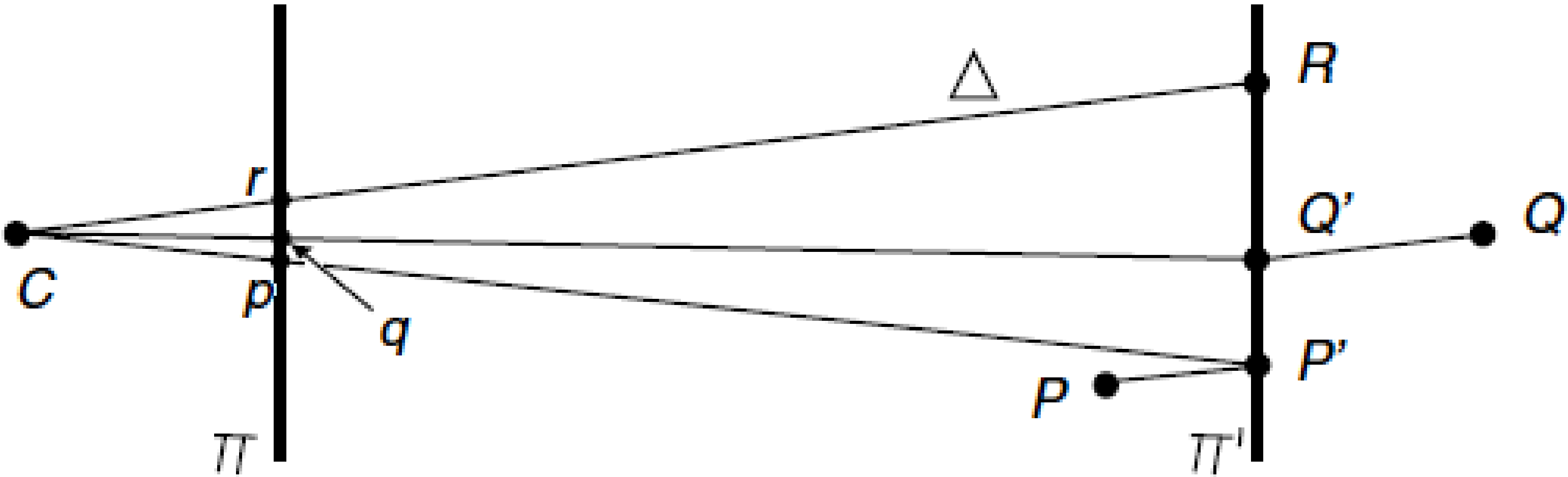
Orthographic Projection



Weak Perspective Projection



Paraperspective Projection



"Affine geometry is, roughly speaking, what is left after all ability to measure lengths, areas, angles, etc. has been removed from Euclidean geometry. The concept of parallelism remains, however, as well as the ability to measure the ratio of distances between collinear points."

[Snapper and Troyer, 1989]

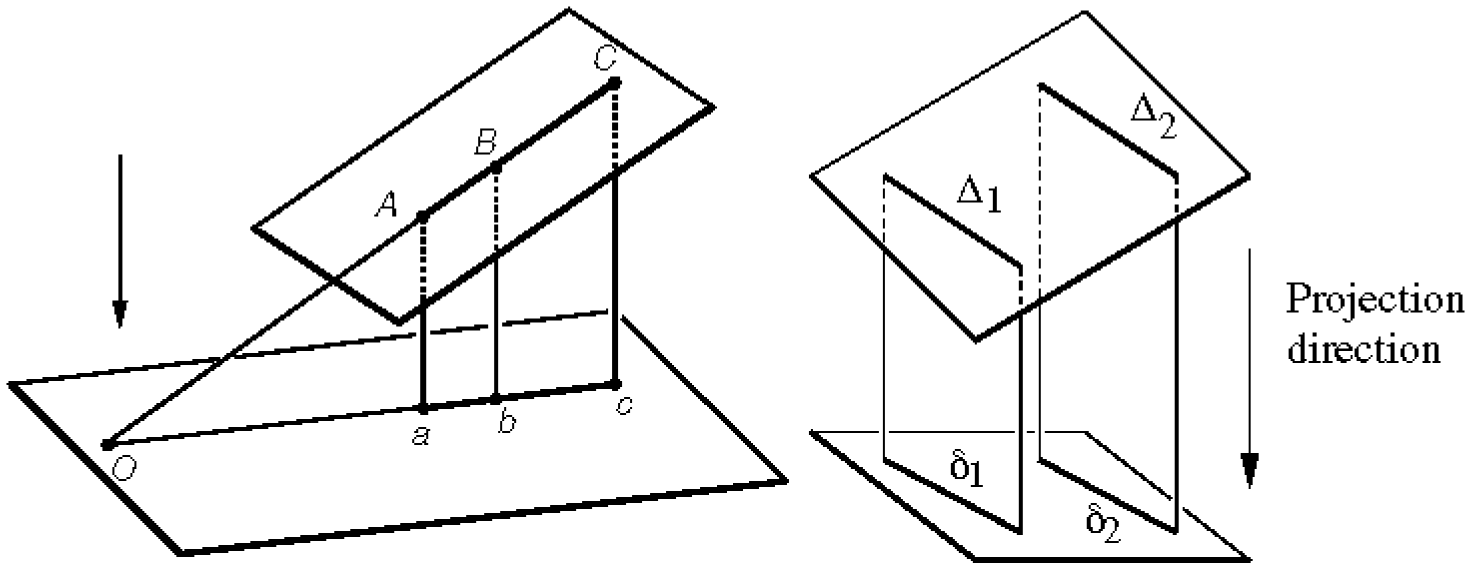
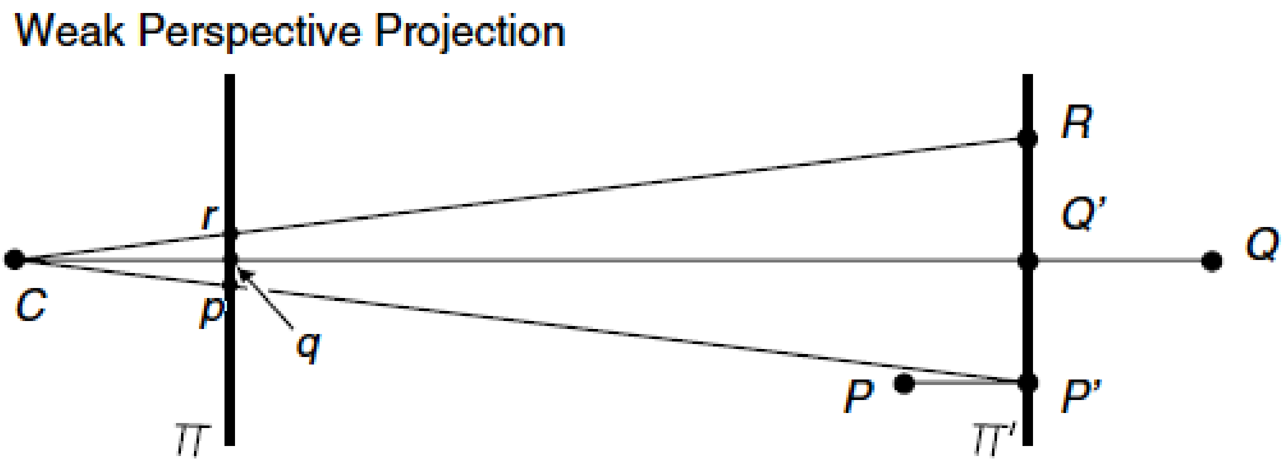


FIGURE 13.2: Parallel projection preserves: (left) the ratio of signed distances between collinear points and (right) the parallelism of lines.

Affine projection matrix

$$\boldsymbol{p}_{ij} = \mathcal{M}_i \begin{pmatrix} \boldsymbol{P}_j \\ 1 \end{pmatrix} = \mathcal{A}_i \boldsymbol{P}_j + \boldsymbol{b}_i$$



Tracked feature j in camera i: $\boldsymbol{p}_{ij}$

$$\boldsymbol{p}_{ij} = \mathcal{M}_i \begin{pmatrix} \boldsymbol{P}_j \\ 1 \end{pmatrix} = \mathcal{A}_i \boldsymbol{P}_j + \boldsymbol{b}_i$$

Affine structure from motion is the problem of estimating

m 2 × 4 matrices

$$\mathcal{M}_i = \begin{pmatrix} \mathcal{A}_i & \boldsymbol{b}_i \end{pmatrix}$$

and the ***n*** positions P_j

from the ***mn*** image correspondences p_{ij}

$$\boldsymbol{p}_{ij} = \mathcal{M}_i \begin{pmatrix} \boldsymbol{P}_j \\ 1 \end{pmatrix} = \mathcal{A}_i \boldsymbol{P}_j + \boldsymbol{b}_i$$

This equation provides 2mn constraints on the 8m+3n unknown coefficients defining the matrices M_i and the point positions P_j.

Fortunately, 2mn is greater than 8m+3n for large enough values of m and n…

But, the solution is ambiguous…

If M_i and P_j are solutions to

$$\boldsymbol{p}_{ij} = \mathcal{M}_i \begin{pmatrix} \boldsymbol{P}_j \\ 1 \end{pmatrix} = \mathcal{A}_i \boldsymbol{P}_j + \boldsymbol{b}_i$$

then so are M'_i and P'_j, where

$$\mathcal{M}'_i = \mathcal{M}_i \mathcal{Q} \quad \text{and} \quad \begin{pmatrix} \boldsymbol{P}'_j \\ 1 \end{pmatrix} = \mathcal{Q}^{-1} \begin{pmatrix} \boldsymbol{P}_j \\ 1 \end{pmatrix}$$

and Q is an arbitrary affine transformation matrix, that is,

$$\mathcal{Q} = \begin{pmatrix} \mathcal{C} & \boldsymbol{d} \\ \boldsymbol{0}^T & 1 \end{pmatrix}$$

where C is a non-singular 3 × 3 matrix and d is a vector in R3. In other words, ***any solution of the affine structure-from-motion problem can only defined up to an affine transformation ambiguity.***

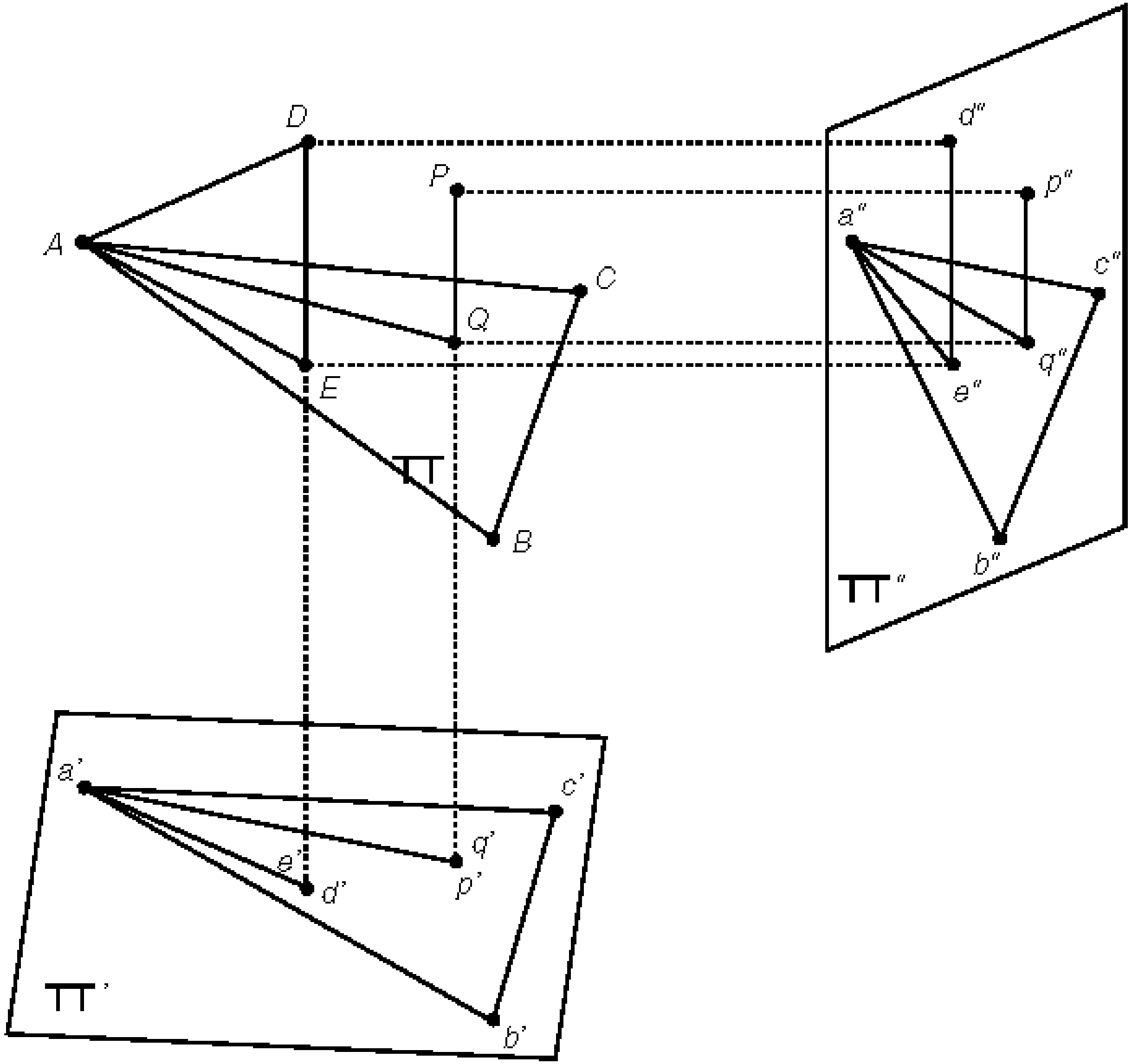
Affine Structure from Motion

- Two views
  – Geometric Approach: infer affine shape (then recover affine projection matricies if needed)
  – Algebraic Approach: estimate projection matricies (then determine position of scene points)
- Sequence
  – Factorization Approach

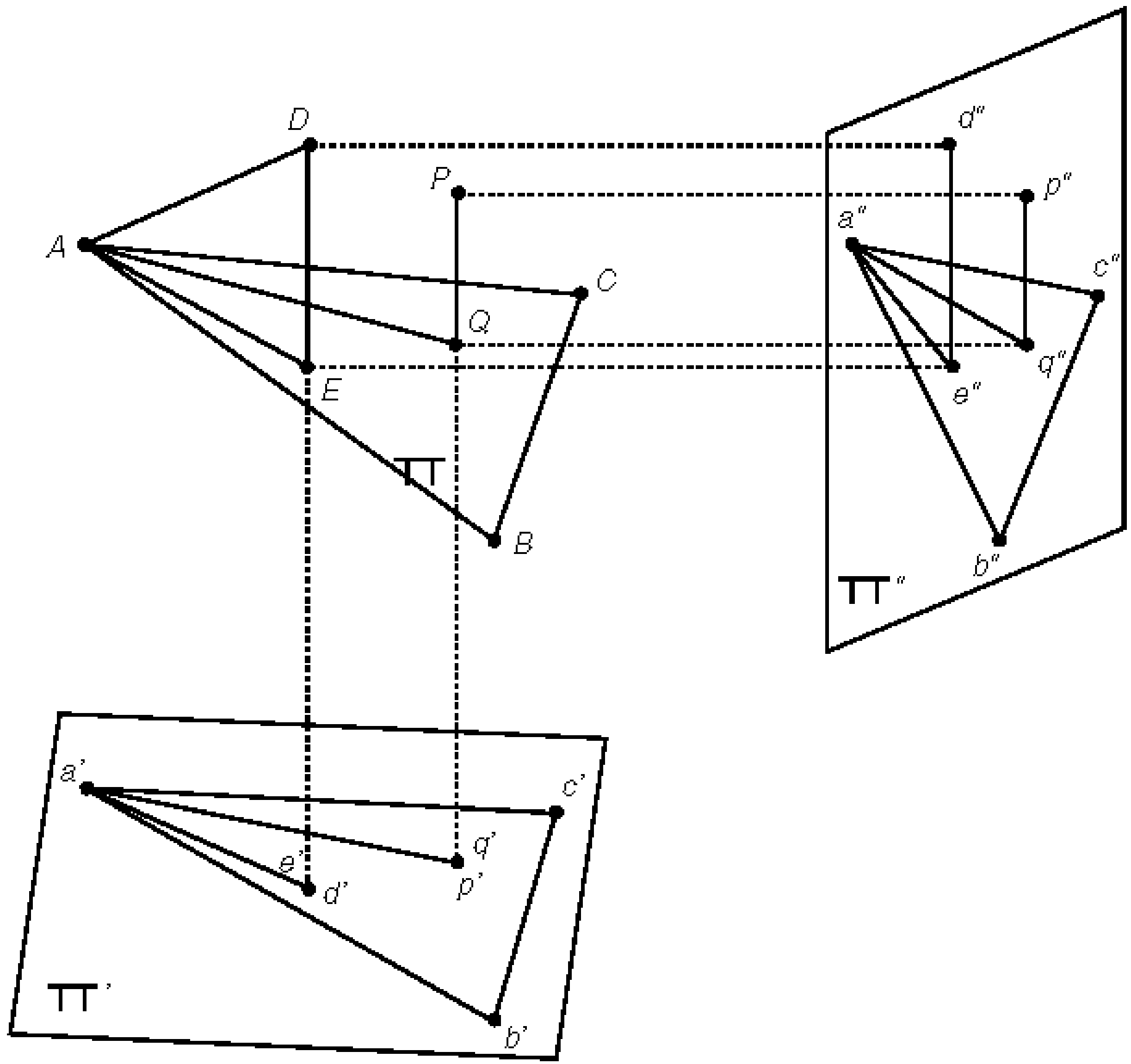
Affine Structure from Motion Theorem

Two affine views of four non co-planar points are sufficient to compute the affine coordinate of any other point P.

[Koenderink and Van Doorn, 1990]

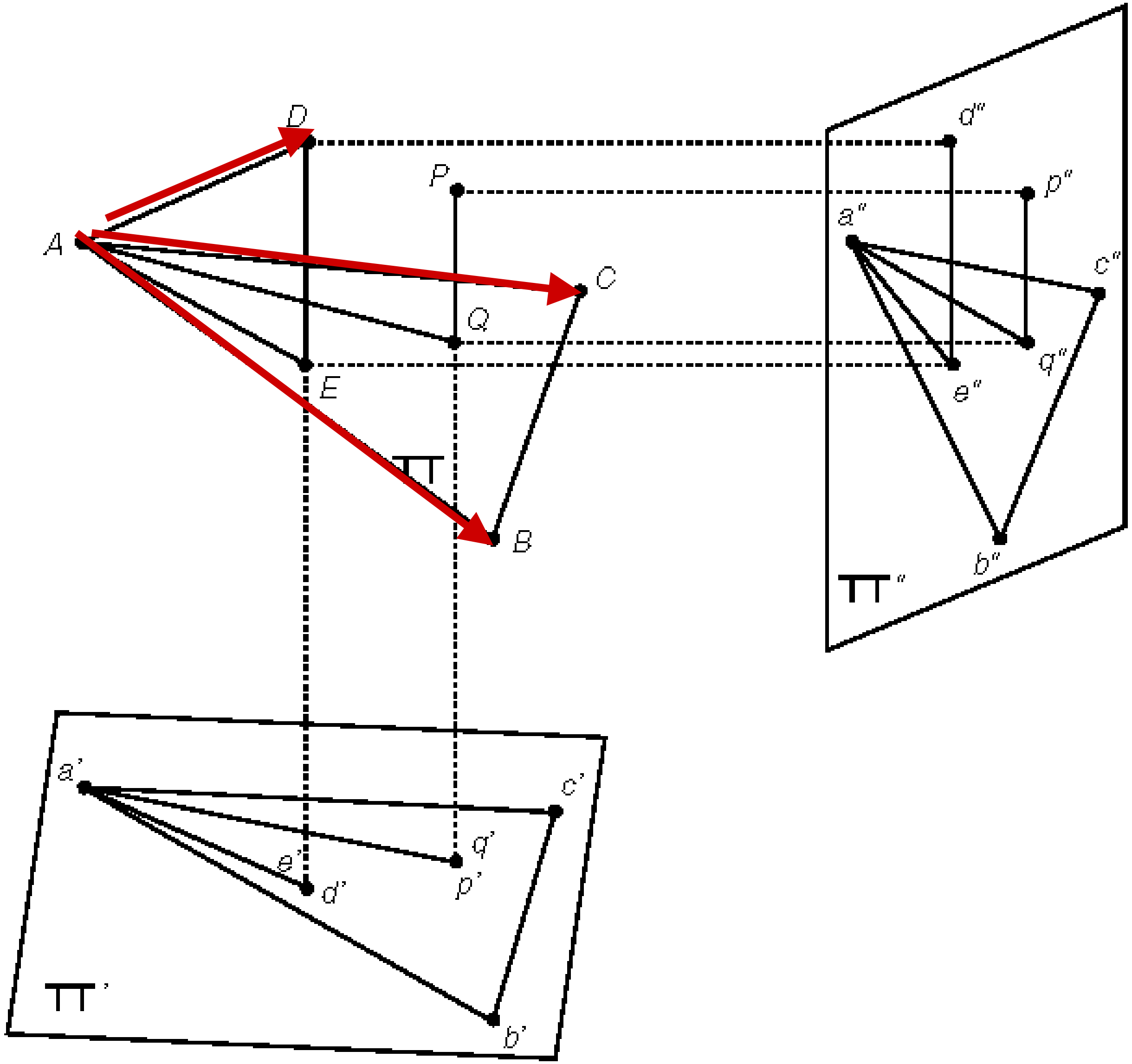

D
P
A
C
Q
E
B
TT
d"
p"
a"
c"
q"
e"
b"
TT"
a'
c'
q'
e'
d'
p'
TT'
b'

Two affine views of four points are sufficient to compute the affine coordinate of any other point P…



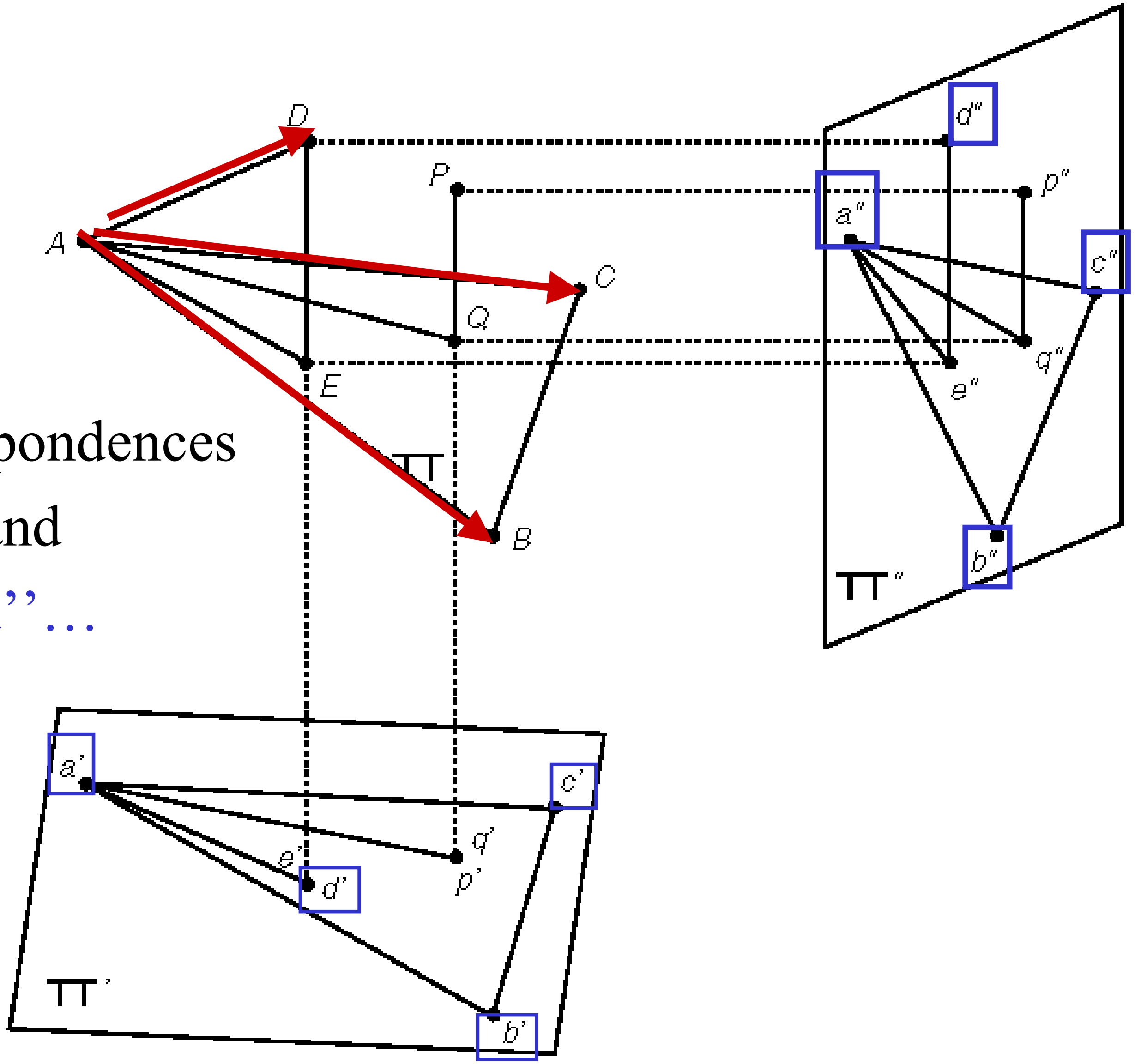
Given Affine Basis (A,B,C,D)

(e.g., A=(0,0,0), B=(0,0,1), C=(0,1,0), D=(1,0,0))



Given Affine Basis (A,B,C,D)

(e.g., A=(0,0,0), B=(0,0,1), C=(0,1,0), D=(1,0,0))



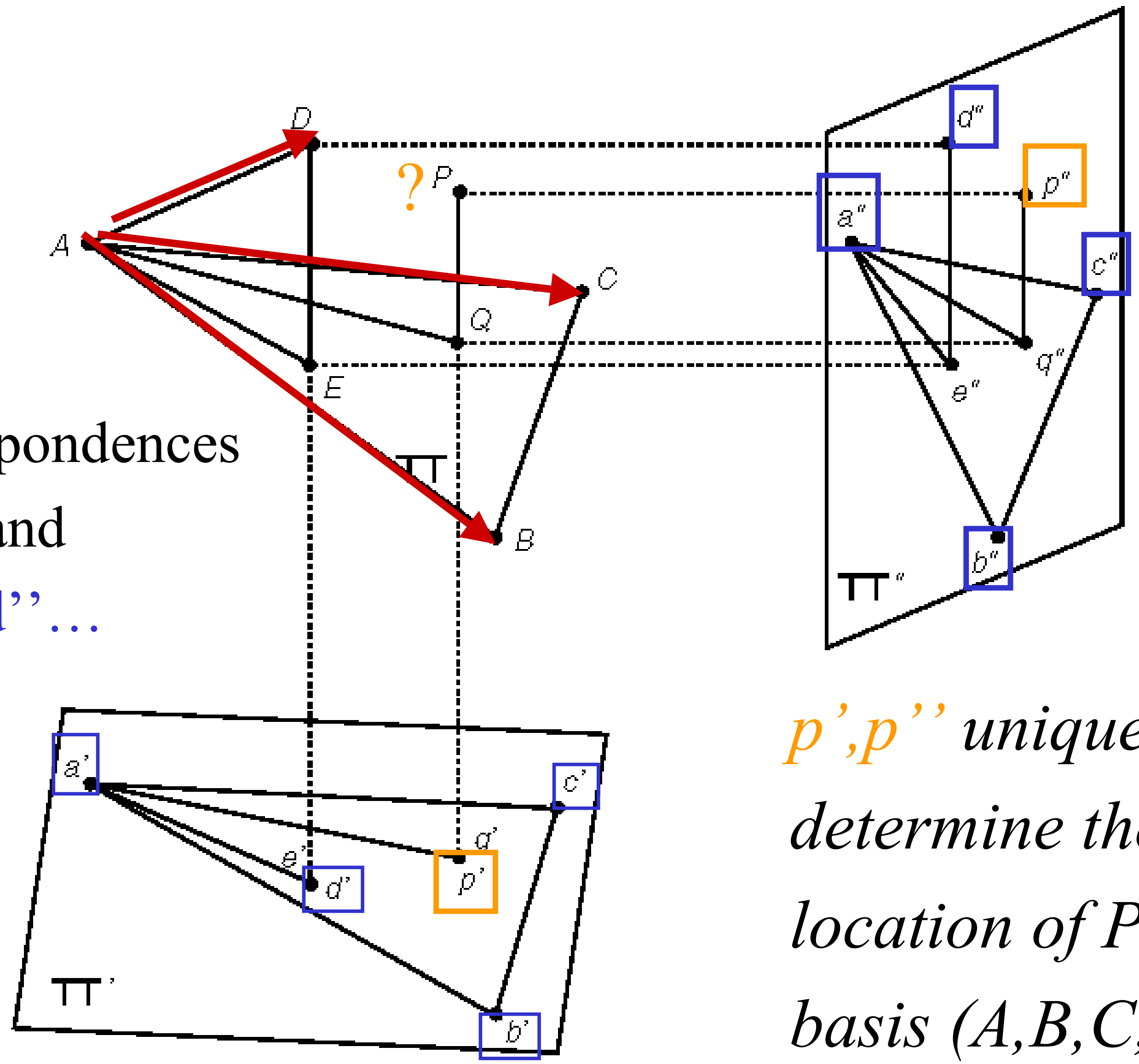
And correspondences

a’,b’,c’,d’ and

a’’,b’’,c’’,d’’…

Given Affine Basis (A,B,C,D)

(e.g., A=(0,0,0), B=(0,0,1), C=(0,1,0), D=(1,0,0))



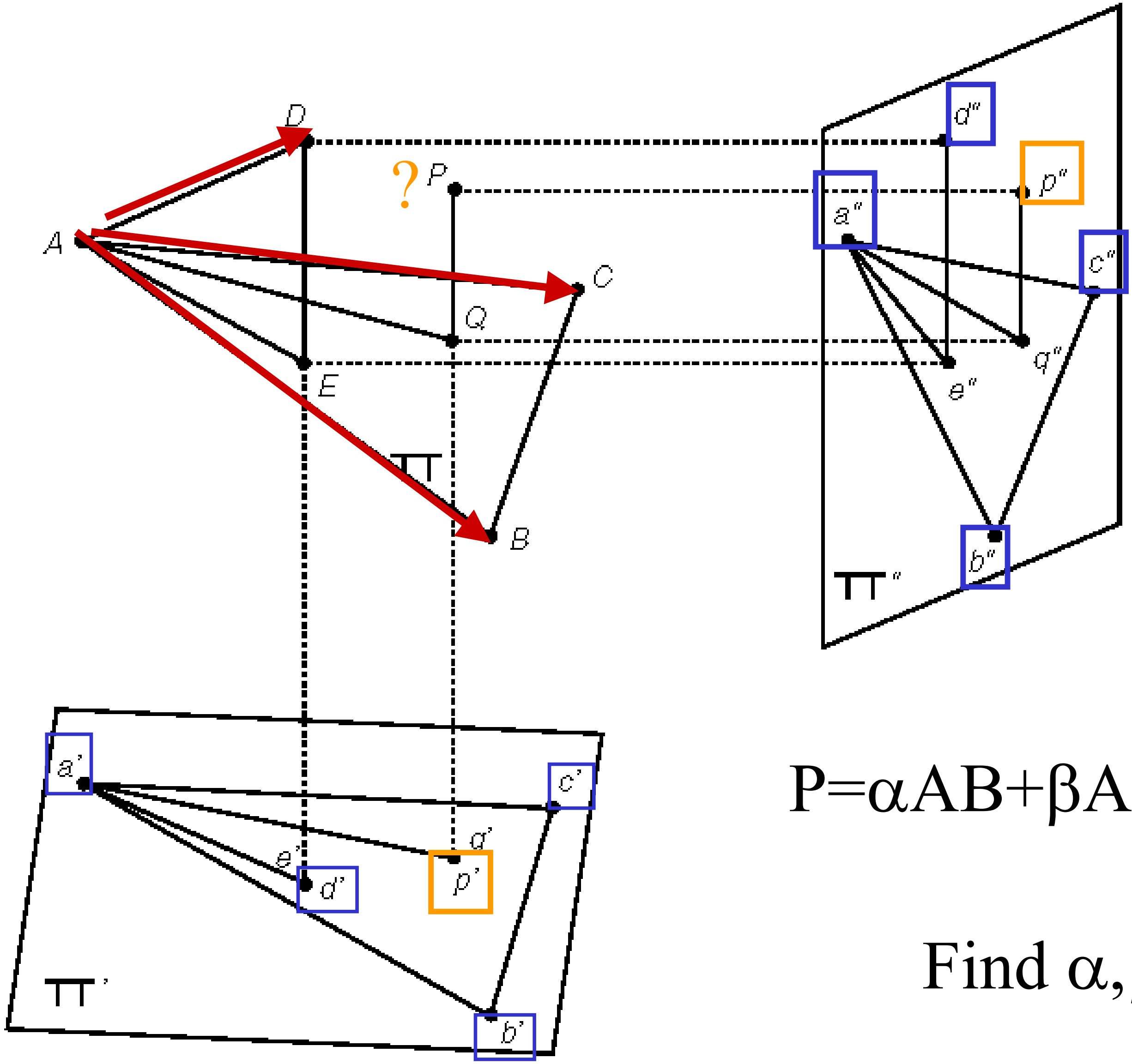
And correspondences

a',b',c',d' and

a'',b'',c'',d''…

p',p'' uniquely determine the location of P in the basis (A,B,C,D)!

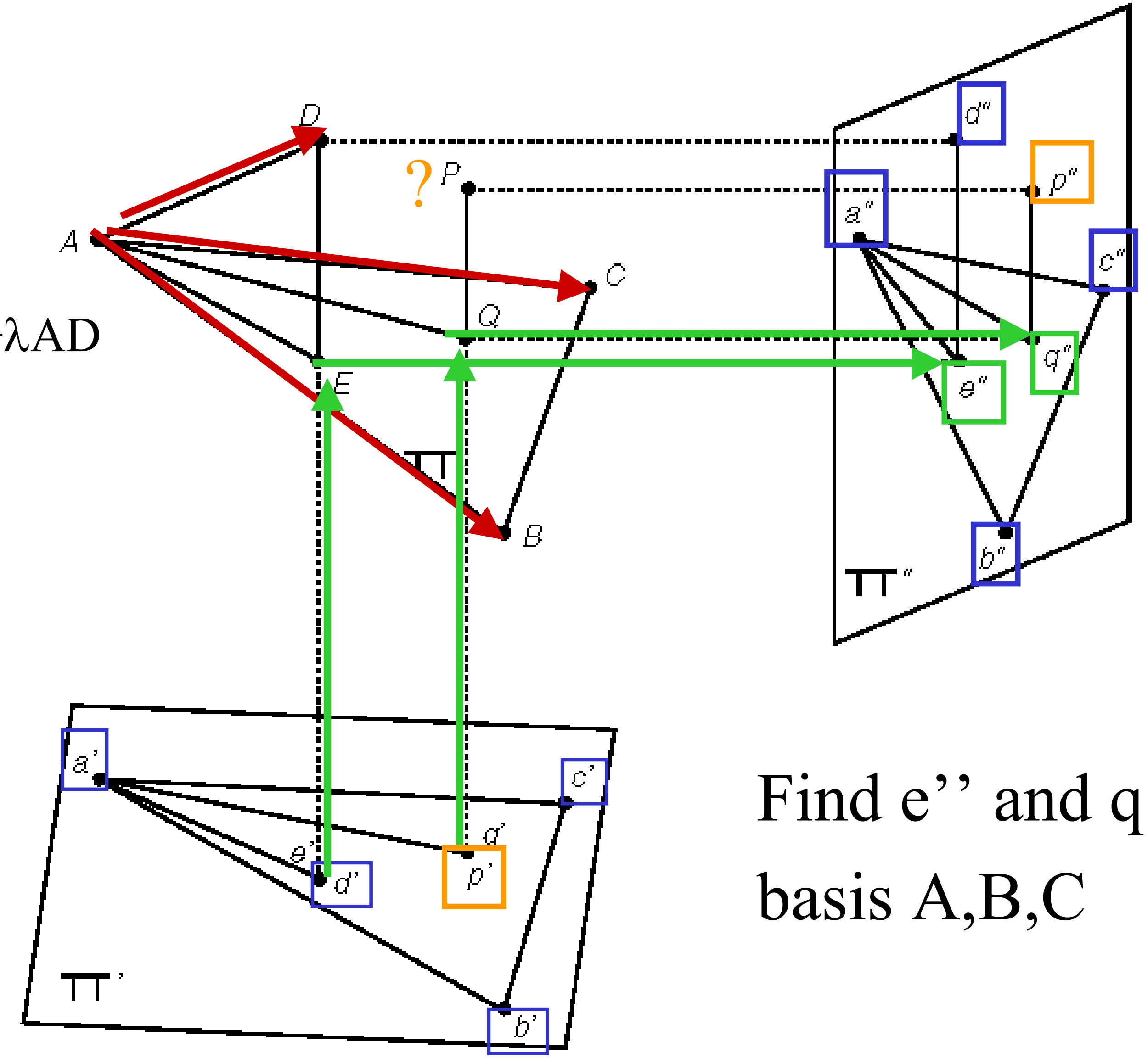
p’,p’’ uniquely determine the location of P in the basis (A,B,C,D)…



$P = \alpha AB + \beta AC + \lambda AD$

Find α, β, λ ?

p’,p’’ uniquely determine the location of P in the basis (A,B,C,D)…

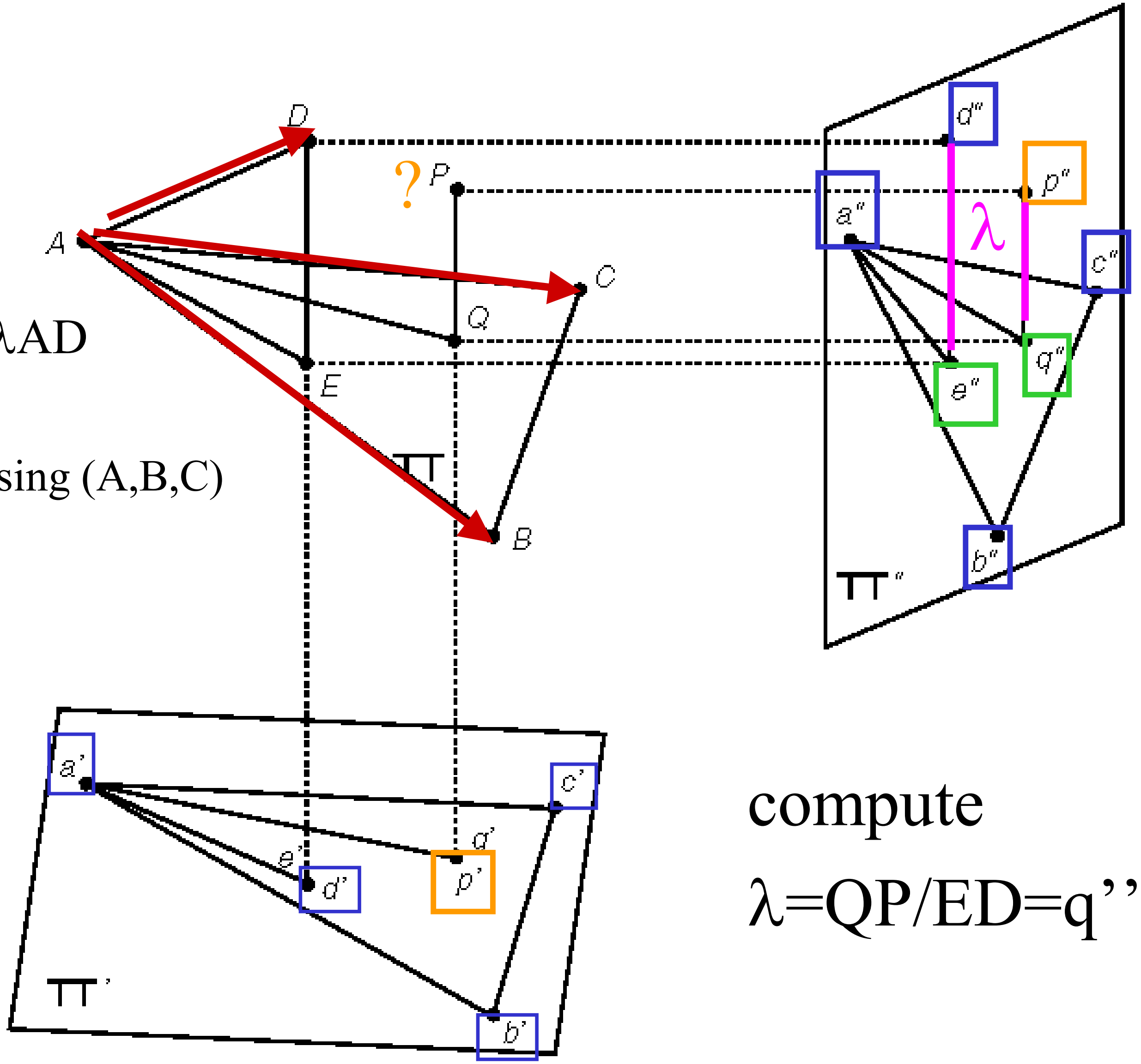


P=αAB+βAC+λAD

Find α,β,λ ?

Find e’’ and q’’ using basis A,B,C

p’,p’’ uniquely determine the location of P in the basis (A,B,C,D)…



P=αAB+βAC+λAD

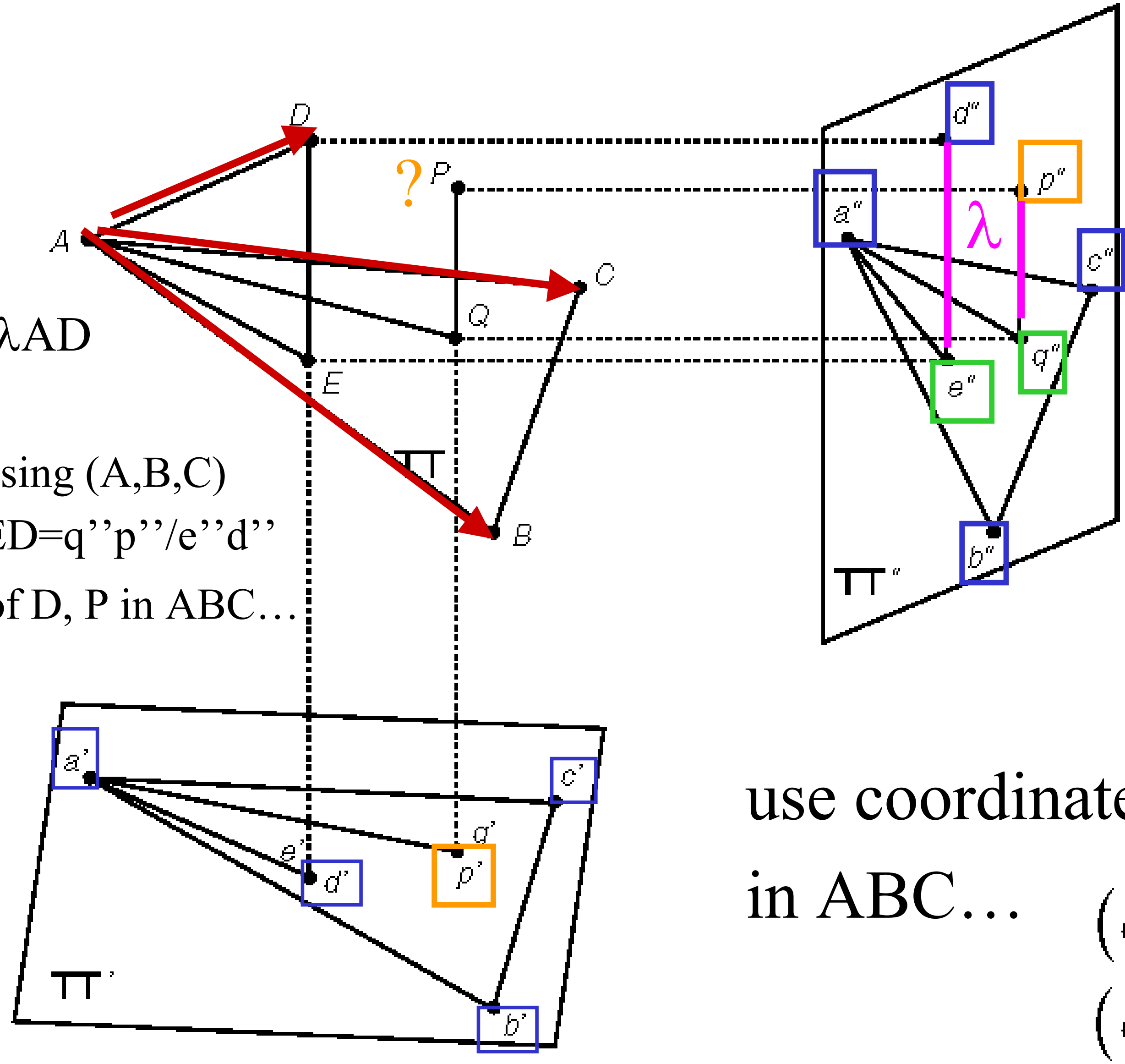
Find α,β,λ ?

Find e’’ and q’’ using (A,B,C)

compute

λ=QP/ED=q’’p’’/e’’d’’

p’,p’’ uniquely determine the location of P in the basis (A,B,C,D)…



P=αAB+βAC+λAD

Find α,β,λ ?

Find e’’ and q’’ using (A,B,C)

Compute λ=QP/ED=q’’p’’/e’’d’’

Use coordinates of D, P in ABC…

use coordinates of D, P in ABC…

$$(\alpha_{d'}, \beta_{d'})$$

$$(\alpha_{p'}, \beta_{p'})$$

p’,p’’ uniquely determine the location of P in the basis (A,B,C,D)…

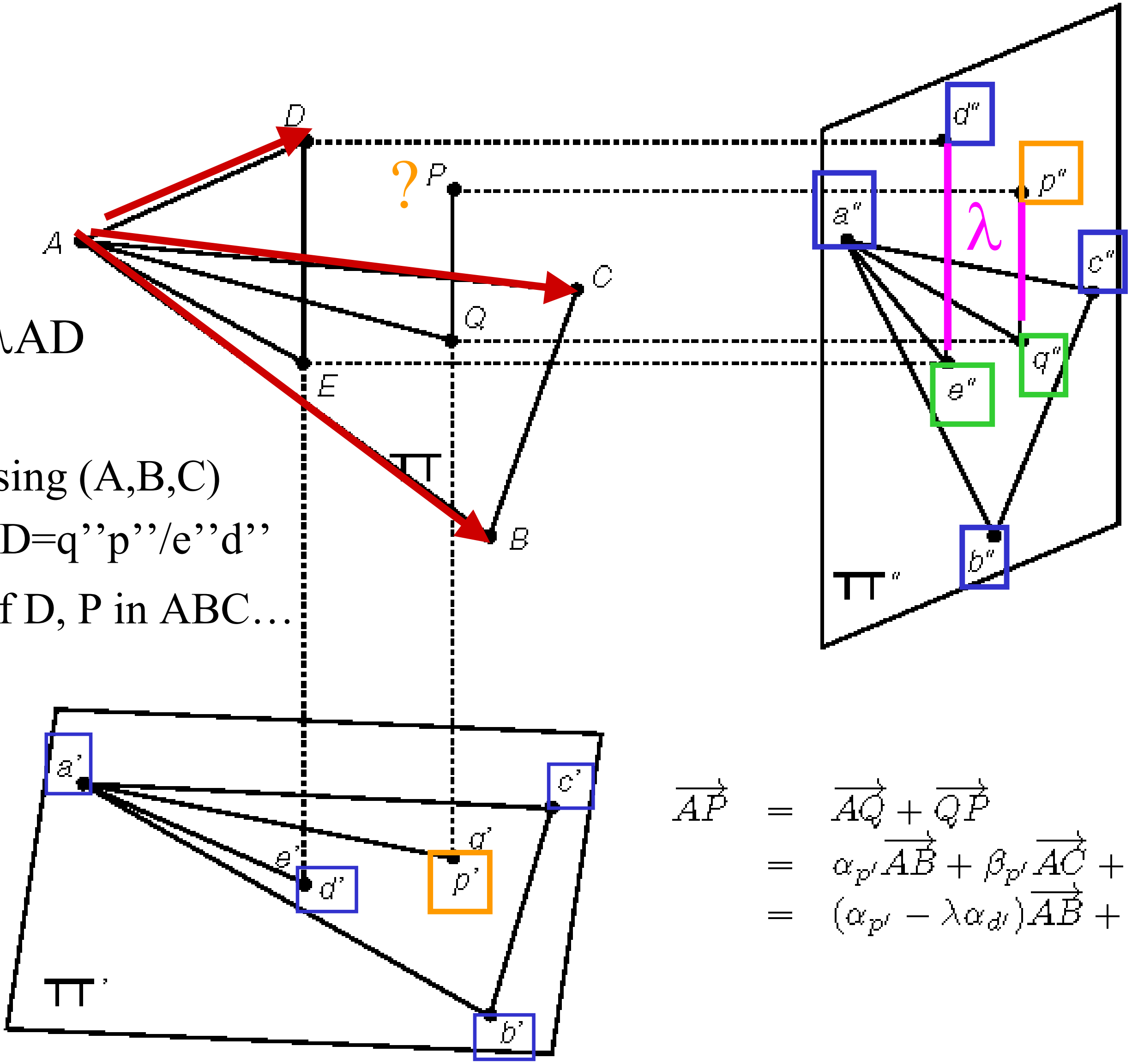
P=αAB+βAC+λAD

Find α,β,λ ?

Find e’’ and q’’ using (A,B,C)

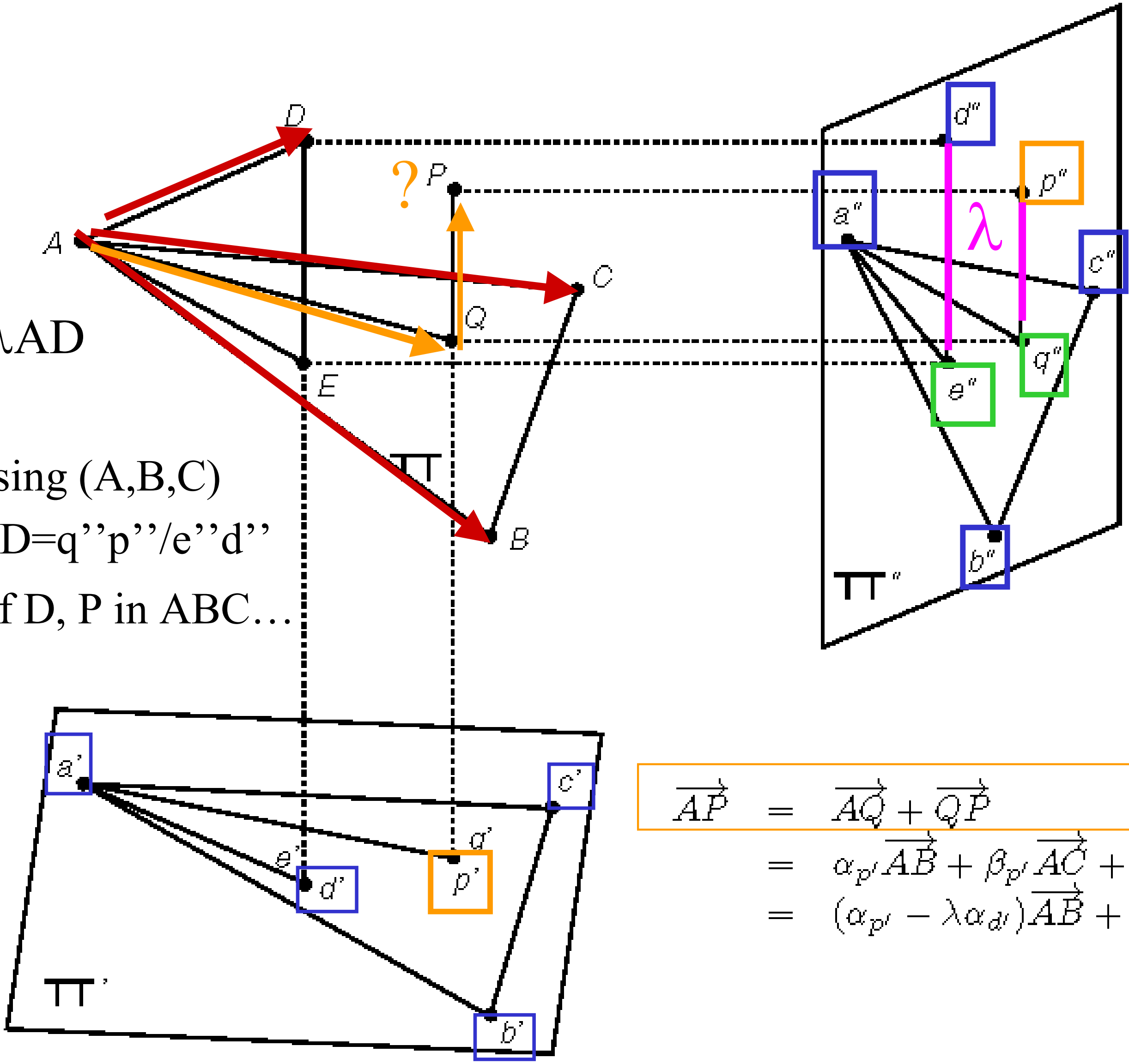
Compute λ=QP/ED=q’’p’’/e’’d’’

Use coordinates of D, P in ABC…



$$
\begin{aligned}
\overrightarrow{AP} &= \overrightarrow{AQ} + \overrightarrow{QP} \\
&= \alpha_{p'}\overrightarrow{AB} + \beta_{p'}\overrightarrow{AC} + \lambda\overrightarrow{ED} \\
&= (\alpha_{p'} - \lambda\alpha_{d'})\overrightarrow{AB} + (\beta_{p'} - \lambda\beta_{d'})\overrightarrow{AC} \\
&\quad + \lambda\overrightarrow{AD}.
\end{aligned}
$$

p’,p’’ uniquely determine the location of P in the basis (A,B,C,D)…



$P = \alpha AB + \beta AC + \lambda AD$

Find α,β,λ ?

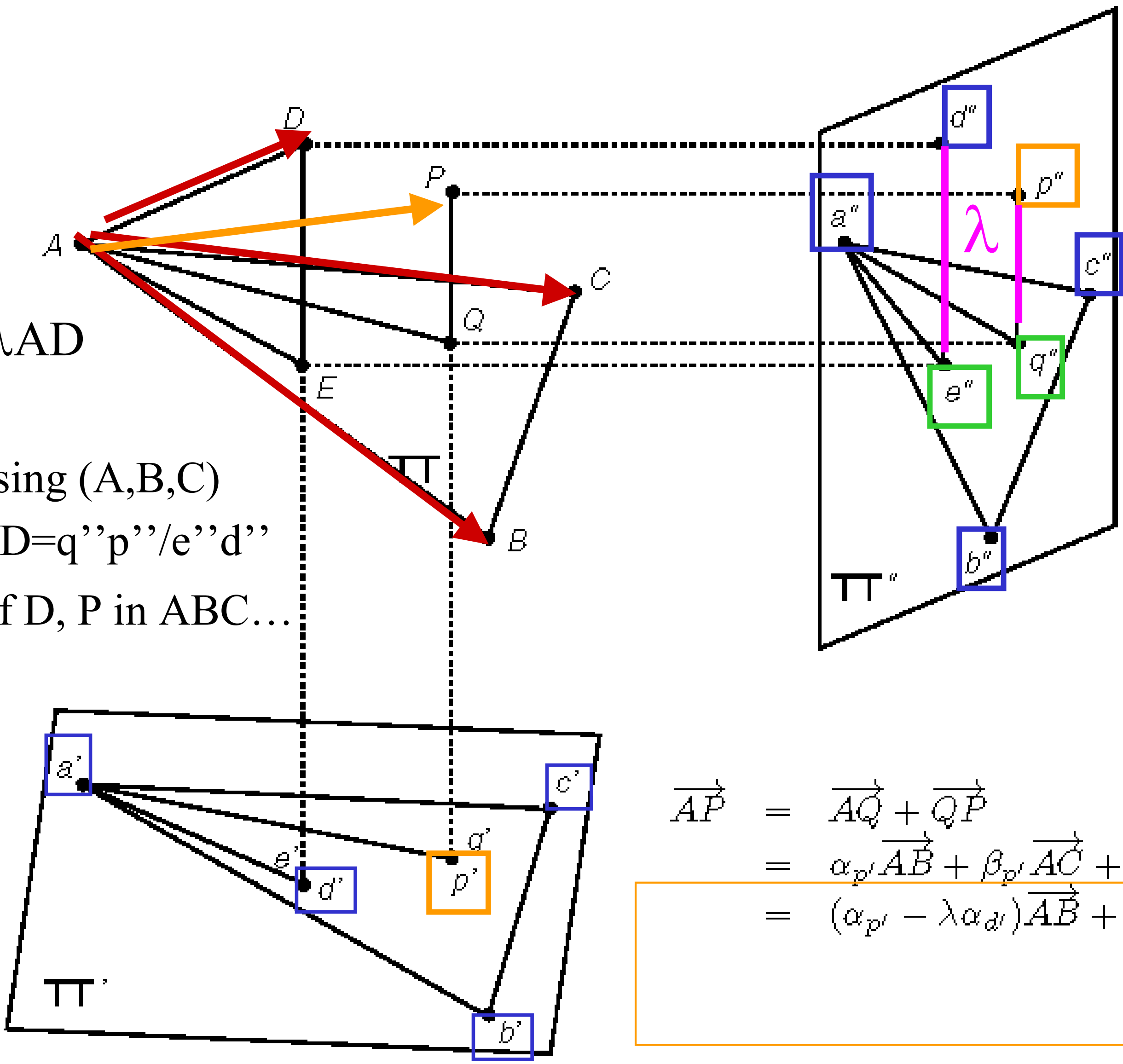
Find e’’ and q’’ using (A,B,C)

Compute $\lambda = QP/ED = q''p''/e''d''$

Use coordinates of D, P in ABC…

$$\begin{aligned}
\overrightarrow{AP} &= \overrightarrow{AQ} + \overrightarrow{QP} \\
&= \alpha_{p'}\overrightarrow{AB} + \beta_{p'}\overrightarrow{AC} + \lambda\overrightarrow{ED} \\
&= (\alpha_{p'} - \lambda\alpha_{d'})\overrightarrow{AB} + (\beta_{p'} - \lambda\beta_{d'})\overrightarrow{AC} \\
&\qquad + \lambda\overrightarrow{AD}.
\end{aligned}$$

p’,p’’ uniquely determine the location of P in the basis (A,B,C,D)…



P=αAB+βAC+λAD

Find α,β,λ ?

Find e’’ and q’’ using (A,B,C)

Compute λ=QP/ED=q’’p’’/e’’d’’

Use coordinates of D, P in ABC…

$$\begin{aligned}\overrightarrow{AP} &= \overrightarrow{AQ} + \overrightarrow{QP} \\ &= \alpha_{p'}\overrightarrow{AB} + \beta_{p'}\overrightarrow{AC} + \lambda\overrightarrow{ED} \\ &= (\alpha_{p'} - \lambda\alpha_{d'})\overrightarrow{AB} + (\beta_{p'} - \lambda\beta_{d'})\overrightarrow{AC} \\ &\quad + \lambda\overrightarrow{AD}.\end{aligned}$$

Geometric Approach

p’,p’’ uniquely determined the location of P in the basis (A,B,C,D)

AP was expressed using weighted combination of AB, AC, AD

Weights were determined by a’,a’’,b’,b’’,c’,c’’,d’,d’’,p’,p’’.

Affine Structure from Motion

- Two views
  - Geometric Approach: infer affine shape (then recover affine projection matricies if needed)
  - Algebraic Approach: estimate projection matricies (then determine position of scene points)
- Sequence
  - Factorization Approach

Algebraic approach

3-d P satisfies two affine views:

$$\begin{aligned} \boldsymbol{p} &= \mathcal{A}\boldsymbol{P} + \boldsymbol{b}, \\ \boldsymbol{p}' &= \mathcal{A}'\boldsymbol{P} + \boldsymbol{b}', \end{aligned}$$

$$\begin{pmatrix} \mathcal{A} & \boldsymbol{p} - \boldsymbol{b} \\ \mathcal{A}' & \boldsymbol{p}' - \boldsymbol{b}' \end{pmatrix} \begin{pmatrix} \boldsymbol{P} \\ -1 \end{pmatrix} = \boldsymbol{0}.$$

$$\operatorname{Det} \begin{pmatrix} \mathcal{A} & \boldsymbol{p} - \boldsymbol{b} \\ \mathcal{A}' & \boldsymbol{p}' - \boldsymbol{b}' \end{pmatrix} = 0$$

$$\mathrm{Det}\begin{pmatrix} \mathcal{A} & \boldsymbol{p}-\boldsymbol{b} \\ \mathcal{A}' & \boldsymbol{p}'-\boldsymbol{b}' \end{pmatrix} = 0$$

But any affine transform of A is equally good…

$$\mathrm{Det}\begin{pmatrix} \mathcal{A}\mathcal{C} & \boldsymbol{p}-\mathcal{A}\boldsymbol{d}-\boldsymbol{b} \\ \mathcal{A}'\mathcal{C} & \boldsymbol{p}'-\mathcal{A}'\boldsymbol{d}-\boldsymbol{b}' \end{pmatrix} = 0$$

for any affine transform

$$\mathcal{Q} = \begin{pmatrix} \mathcal{C} & \boldsymbol{d} \\ \mathbf{0}^T & 1 \end{pmatrix}$$

$$\mathrm{Det}\begin{pmatrix} \mathcal{A}\mathcal{C} & \boldsymbol{p} - \mathcal{A}\boldsymbol{d} - \boldsymbol{b} \\ \mathcal{A}'\mathcal{C} & \boldsymbol{p}' - \mathcal{A}'\boldsymbol{d} - \boldsymbol{b}' \end{pmatrix} = 0 \qquad \mathcal{Q} = \begin{pmatrix} \mathcal{C} & \boldsymbol{d} \\ \mathbf{0}^T & 1 \end{pmatrix}$$

Let’s pick a special C, d…

$$\mathcal{C} = \mathcal{S}^{-1} \qquad \mathcal{S} = \begin{pmatrix} \boldsymbol{a}_1^T \\ \boldsymbol{a}_2^T \\ \boldsymbol{a}_1'^T \end{pmatrix} \qquad \boldsymbol{r} = \begin{pmatrix} b_1 \\ b_2 \\ b_1' \end{pmatrix}$$

$$\boldsymbol{d} = -\mathcal{S}^{-1}\boldsymbol{r}.$$

which is equivalent to choosing cannonical affine projection matrices

$$\tilde{\mathcal{M}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \qquad \tilde{\mathcal{M}}' = \begin{pmatrix} 0 & 0 & 1 & 0 \\ a & b & c & d \end{pmatrix}$$

and our determinant becomes very simple:

$$\mathrm{Det}\begin{pmatrix} 1 & 0 & 0 & u \\ 0 & 1 & 0 & v \\ 0 & 0 & 1 & u' \\ a & b & c & v'-d \end{pmatrix} = au - bv + cu' + v' - d = 0$$

a,b,c,d can be estimated using least squares with a sufficient number of points. Then P can be recovered with:

$$\begin{pmatrix} 1 & 0 & 0 & u \\ 0 & 1 & 0 & v \\ 0 & 0 & 1 & u' \\ a & b & c & v'-d \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{P}} \\ -1 \end{pmatrix} = 0$$

Affine Structure from Motion

- Two views
  - Geometric Approach: infer affine shape (then recover affine projection matricies if needed)
  - Algebraic Approach: estimate projection matricies (then determine position of scene points)
- Sequence
  - Factorization Approach

Factorization Approach

Consider a sequence of affine cameras….

$$\boldsymbol{p}_i = \mathcal{M}_i \begin{pmatrix} \boldsymbol{P} \\ 1 \end{pmatrix} = \mathcal{A}_i \boldsymbol{P} + \boldsymbol{b}_i$$

Stack affine projection equations:

$$\boldsymbol{q} = \boldsymbol{r} + \mathcal{A}\boldsymbol{P}$$

$$\boldsymbol{q} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{p}_1 \\ \dots \\ \boldsymbol{p}_m \end{pmatrix}, \quad \boldsymbol{r} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{b}_1 \\ \dots \\ \boldsymbol{b}_m \end{pmatrix} \quad \text{and} \quad \mathcal{A} \stackrel{\text{def}}{=} \begin{pmatrix} \mathcal{A}_1 \\ \dots \\ \mathcal{A}_m \end{pmatrix}$$

$$\boldsymbol{q} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{p}_1 \\ \dots \\ \boldsymbol{p}_m \end{pmatrix}, \quad \boldsymbol{r} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{b}_1 \\ \dots \\ \boldsymbol{b}_m \end{pmatrix} \quad \text{and} \quad \mathcal{A} \stackrel{\text{def}}{=} \begin{pmatrix} \mathcal{A}_1 \\ \dots \\ \mathcal{A}_m \end{pmatrix}$$

Form the (2m+1)n data matrix where each column is the observed data from one point:

$$\mathcal{D} = \begin{pmatrix} \boldsymbol{q}_1 & \dots & \boldsymbol{q}_n \\ 1 & \dots & 1 \end{pmatrix}$$

$$\boldsymbol{q} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{p}_1 \\ \dots \\ \boldsymbol{p}_m \end{pmatrix}, \quad \boldsymbol{r} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{b}_1 \\ \dots \\ \boldsymbol{b}_m \end{pmatrix} \quad \text{and} \quad \mathcal{A} \stackrel{\text{def}}{=} \begin{pmatrix} \mathcal{A}_1 \\ \dots \\ \mathcal{A}_m \end{pmatrix}$$

Form the (2m+1)n data matrix where each column is the observed data from one point:

Since

$$\mathcal{D} = \begin{pmatrix} \boldsymbol{q}_1 & \dots & \boldsymbol{q}_n \\ 1 & \dots & 1 \end{pmatrix}$$

then

$$\boldsymbol{q} = \boldsymbol{r} + \mathcal{A}\boldsymbol{P}$$

With an appropriate choice of origin (e.g., first point, centriod)

$$\boldsymbol{p}_i = \mathcal{A}_i \boldsymbol{P} \qquad\qquad \boldsymbol{q} = \mathcal{A}\boldsymbol{P}$$

and the data matrix becomes:

$$\mathcal{D} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{q}_1 & \dots & \boldsymbol{q}_n \end{pmatrix} = \mathcal{A}\mathcal{P}$$

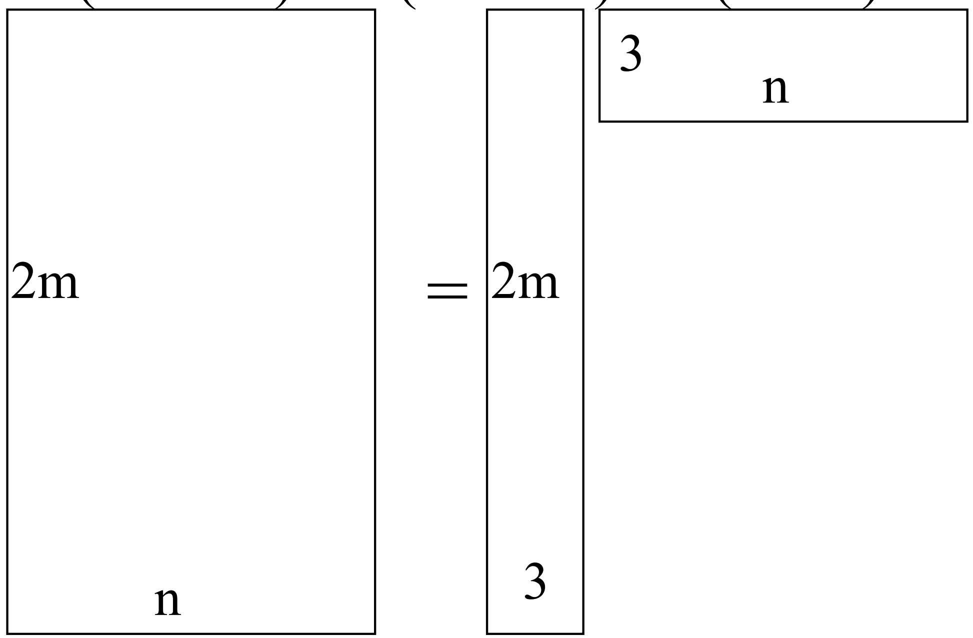
$$\mathcal{P} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{P}_1 & \dots & \boldsymbol{P}_n \end{pmatrix}.$$

Rank of Object-relative Data Matrix

$$D = A \quad P$$

Data-Matrix = Affine-Motions x 3-d-Points

$$(2m \times n) = (2m \times 3) \times (3 \times n)$$



D is now rank 3

Factorization algorithm

Given a data matrix,

find Motion (A) and Shape (P) matrices that generate that data…

Tomasi and Kanade Factorization algorithm (1992):

Use Singular Value Decomposition to factor D into appropriately sized A and P.

SVD

Technique: Singular Value Decomposition Let $\mathcal{A}$ be an $m \times n$ matrix, with $m \geq n$, then $\mathcal{A}$ can always be written as

$$\mathcal{A} = \mathcal{U}\mathcal{W}\mathcal{V}^T,$$

where:

- $\mathcal{U}$ is an $m \times n$ column-orthogonal matrix, i.e., $\mathcal{U}^T\mathcal{U} = \mathrm{Id}_m$,
- $\mathcal{W}$ is a diagonal matrix whose diagonal entries w_i $(i = 1, \dots, n)$ are the singular values of $\mathcal{A}$ with $w_1 \geq w_2 \geq \dots \geq w_n \geq 0$,
- and $\mathcal{V}$ is an $n \times n$ orthogonal matrix, i.e., $\mathcal{V}^T\mathcal{V} = \mathcal{V}\mathcal{V}^T = \mathrm{Id}_n$.

The SVD of a matrix can also be used to characterize matrices that are rank-deficient: suppose that $\mathcal{A}$ has rank $p < n$, then the matrices $\mathcal{U}$, $\mathcal{W}$, and $\mathcal{V}$ can be written as

$$\mathcal{U} = \begin{bmatrix} \mathcal{U}_p & \mathcal{U}_{n-p} \end{bmatrix} \quad \mathcal{W} = \begin{bmatrix} \mathcal{W}_p & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \mathcal{V}^T = \begin{bmatrix} \mathcal{V}_p^T \\ \mathcal{V}_{n-p}^T \end{bmatrix},$$

Factorization algorithm

1. Compute the singular value decomposition $\mathcal{D} = \mathcal{U}\mathcal{W}\mathcal{V}^T$.

2. Construct the matrices $\mathcal{U}_3$, $\mathcal{V}_3$, and $\mathcal{W}_3$ formed by the three leftmost columns of the matrices $\mathcal{U}$ and $\mathcal{V}$, and the corresponding 3×3 sub-matrix of $\mathcal{W}$.

3. Define

$$\mathcal{A}_0 = \mathcal{U}_3 \quad \text{and} \quad \mathcal{P}_0 = \mathcal{W}_3\mathcal{V}_3^T;$$

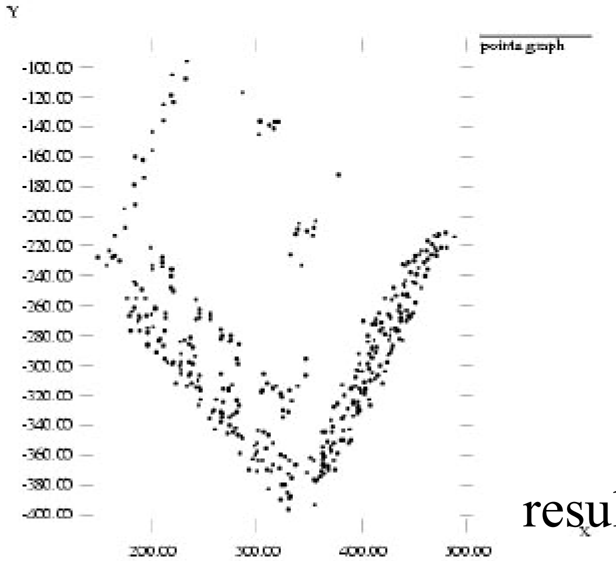
the $2m \times 3$ matrix $\mathcal{A}_0$ is an estimate of the camera motion, and the $3 \times n$ matrix $\mathcal{P}_0$ is an estimate of the scene structure.

Factorization algorithm

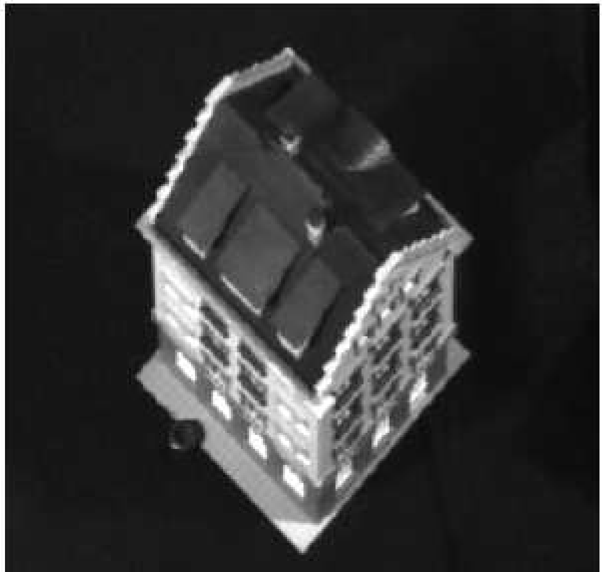
1

60

Input



result



comparision

Factorization algorithm

Can perform *Euclidean upgrade* to estimate metric quantities...

- Of all the family of affine solutions, find the one that obeys calibration constraints.

Euclidean upgrade

Lets recover Euclidean structure from affine structure, under orthographic projection:

Add constraints on rows a,b of ***A:***

$$\boldsymbol{a} \cdot \boldsymbol{b} = 0 \quad \text{and} \quad |\boldsymbol{a}|^2 = |\boldsymbol{b}|^2 = 1.$$

Recall, if M_i and P_j are solutions to

$$\boldsymbol{p}_{ij} = \mathcal{M}_i \begin{pmatrix} \boldsymbol{P}_j \\ 1 \end{pmatrix} = \mathcal{A}_i \boldsymbol{P}_j + \boldsymbol{b}_i$$

then so are M'_i and P'_j, where

$$\mathcal{M}'_i = \mathcal{M}_i \mathcal{Q} \quad \text{and} \quad \begin{pmatrix} \boldsymbol{P}'_j \\ 1 \end{pmatrix} = \mathcal{Q}^{-1} \begin{pmatrix} \boldsymbol{P}_j \\ 1 \end{pmatrix}$$

and Q is an arbitrary affine transformation matrix, that is,

$$\mathcal{Q} = \begin{pmatrix} \mathcal{C} & \boldsymbol{d} \\ \boldsymbol{0}^T & 1 \end{pmatrix}$$

where C is a non-singular 3 × 3 matrix and d is a vector in R3.

Search for Q which satisfies constraint on previous slide…

Euclidean upgrade

Orthographic camera ; constraints on rows a,b of ***A:***

$$\boldsymbol{a} \cdot \boldsymbol{b} = 0 \quad \text{and} \quad |\boldsymbol{a}|^2 = |\boldsymbol{b}|^2 = 1.$$

$$\hat{\mathcal{M}} = \mathcal{M}\mathcal{Q} \text{ and } \hat{\mathcal{P}} = \mathcal{Q}^{-1}\mathcal{P}.$$

so

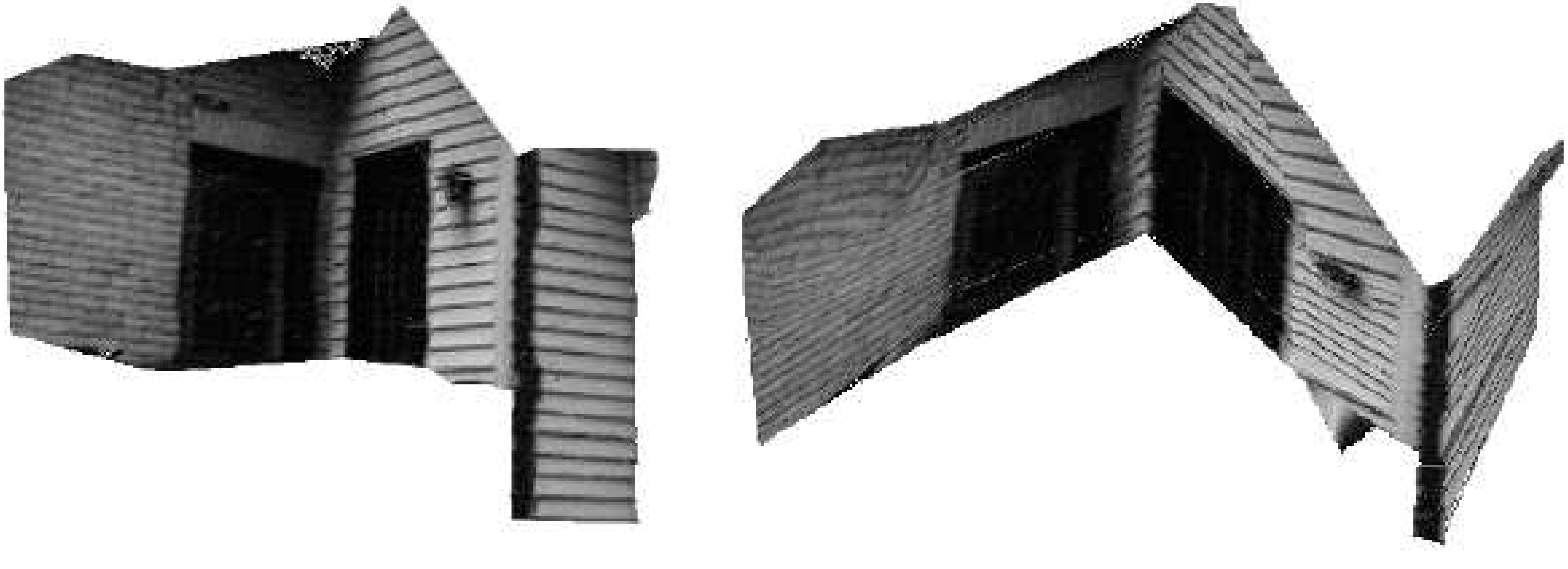
$$\begin{aligned} \boldsymbol{a}_i^T \mathcal{Q}\mathcal{Q}^T \boldsymbol{b}_i &= 0, \\ \boldsymbol{a}_i^T \mathcal{Q}\mathcal{Q}^T \boldsymbol{a}_i &= 1, \\ \boldsymbol{b}_i^T \mathcal{Q}\mathcal{Q}^T \boldsymbol{b}_i &= 1, \end{aligned}$$

but we can assume

$$\hat{\mathcal{M}}_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

Solve for $\boldsymbol{M}_i$ with nonlinear least squares (or via Choelsky decomp.)

Euclidean upgrade



Factorization algorithm

Extensions to basic algorithm:

– sparse data
– multiple motions
– projective cameras (later)

Multiple motions

With multiple motions

$$\mathcal{D} = \begin{pmatrix} \boldsymbol{p}_{11} & \cdots & \boldsymbol{p}_{1n} \\ \cdots & \cdots & \cdots \\ \boldsymbol{p}_{m1} & \cdots & \boldsymbol{p}_{mn} \\ 1 & \dots & 1 \end{pmatrix}.$$

has rank *4k*

Multiple motions

With multiple motions

for $i = 1, \ldots, k$, a rank-4 data matrix

$$\mathcal{D}^{(i)} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{p}_{11}^{(i)} & \cdots & \boldsymbol{p}_{1n_i}^{(i)} \\ \cdots & \cdots & \cdots \\ \boldsymbol{p}_{m1}^{(i)} & \cdots & \boldsymbol{p}_{mn_i}^{(i)} \end{pmatrix},$$

$$\mathcal{D}^{(i)} = \mathcal{M}^{(i)}\mathcal{P}^{(i)}$$

$$\mathcal{M}^{(i)} \stackrel{\text{def}}{=} \begin{pmatrix} \mathcal{M}_1^{(i)} & \boldsymbol{o}_1^{(i)} \\ \cdots & \cdots \\ \mathcal{M}_m^{(i)} & \boldsymbol{o}_m^{(i)} \end{pmatrix} \quad \text{and} \quad \mathcal{P}^{(i)} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{P}_1^{(i)} & \cdots & \boldsymbol{P}_{n_i}^{(i)} \\ 1 & \cdots & 1 \end{pmatrix}.$$

Let us define the $2m \times n$ composite data matrix

$$\mathcal{D} \stackrel{\text{def}}{=} (\mathcal{D}^{(1)}\mathcal{D}^{(2)} \ldots \mathcal{D}^{(k)}),$$

as well as the composite $2m \times 4k$ (motion) and $4k \times n$ (structure) matrices

$$\mathcal{M} \stackrel{\text{def}}{=} (\mathcal{M}^{(1)}\mathcal{M}^{(2)} \ldots \mathcal{M}^{(k)}) \quad \text{and} \quad \mathcal{P} \stackrel{\text{def}}{=} \begin{pmatrix} \mathcal{P}^{(1)} & 0 & \ldots & 0 & 0 \\ 0 & \mathcal{P}^{(2)} & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & 0 & \mathcal{P}^{(k)} \end{pmatrix}.$$

With this notation, we have

$$\mathcal{D} = \mathcal{M}\mathcal{P},$$

which confirms, of course, that $\mathcal{D}$ has rank $4k$ (or less).

Multiple motions

With multiple motions

$$\mathcal{D} = \begin{pmatrix} \boldsymbol{p}_{11} & \cdots & \boldsymbol{p}_{1n} \\ \cdots & \cdots & \cdots \\ \boldsymbol{p}_{m1} & \cdots & \boldsymbol{p}_{mn} \\ 1 & \ldots & 1 \end{pmatrix}.$$

has rank *4k*

Affine Structure from Motion

- Two views
  - Geometric Approach: infer affine shape (then recover affine projection matricies if needed)
  - Algebraic Approach: estimate projection matricies (then determine position of scene points)
- Sequence
  - Factorization Approach

[Most Figures from Forsythe and Ponce]

Today

Affine SFM

- Geometric Approach
- Algebraic Approach
- Tomasi/Kanade Factorization