**2.5 The projective geometry of 1D**

: Homogeneous coordinates of point x on line  : ideal point of a line

Projective transformation of a line   🡪 3 DOF

🡪 A projective transformation of a line may be determined from three corresponding points

선의 사영 변환은 3개의 대응점으로부터 결정될 수 있다.

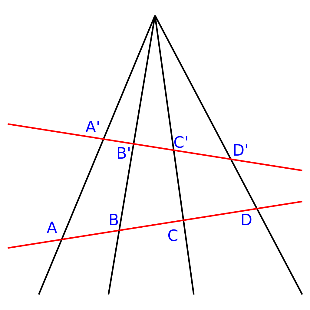
**The cross ratio**





The cross ratio is invariant under projective transformation of the line in

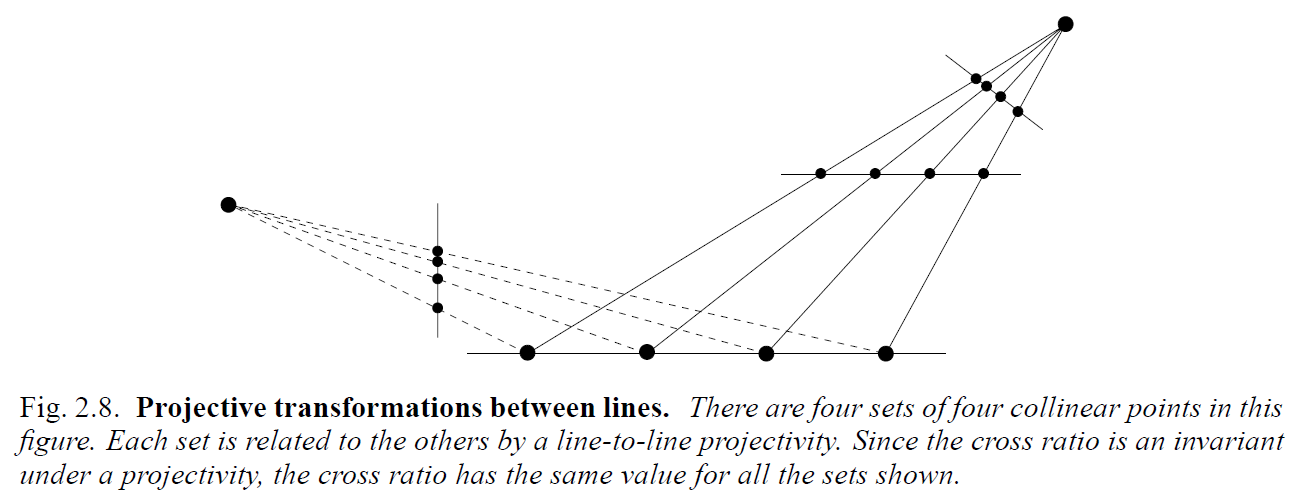




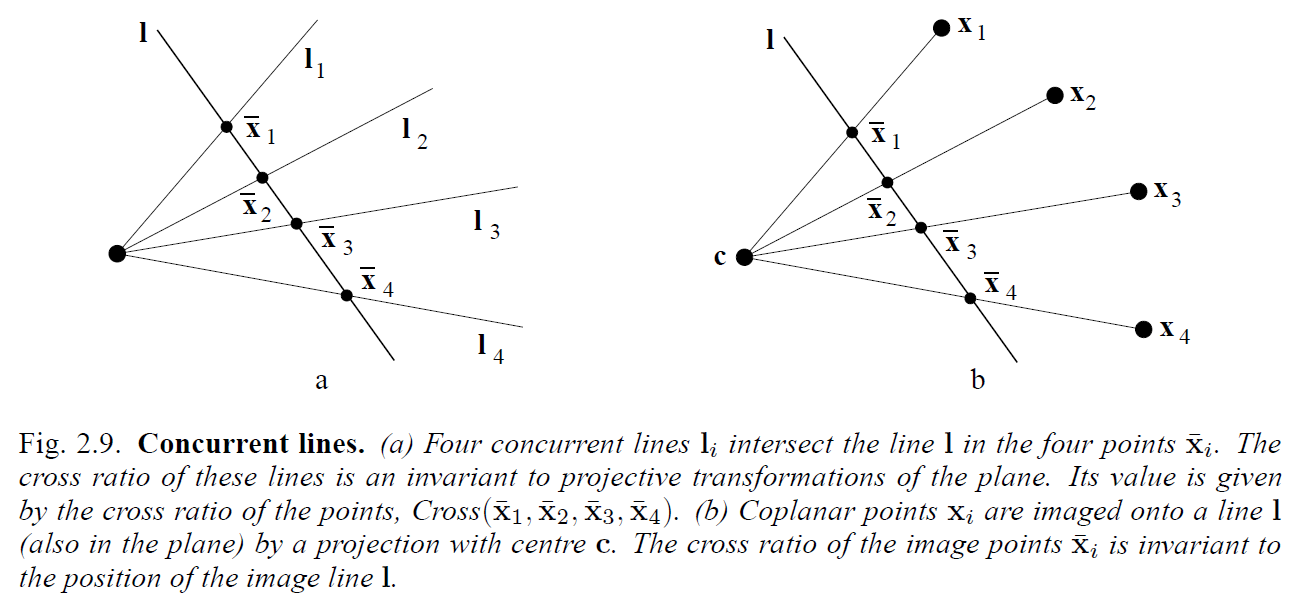
* Properties:   
  - Defines coordinates along a 1D projective line  
  - Independent of the homogeneous representation of x  
  - Valid for ideal points  
  - Invariant under homographies (Projective transformation)

Projective transformation between lines with equivalent cross ratio

1D projective transformation is induced on any line in the plane

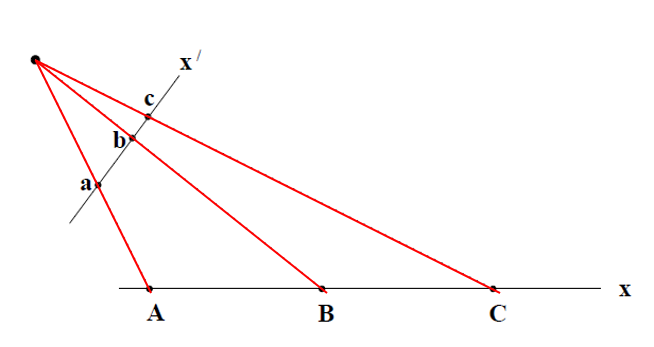


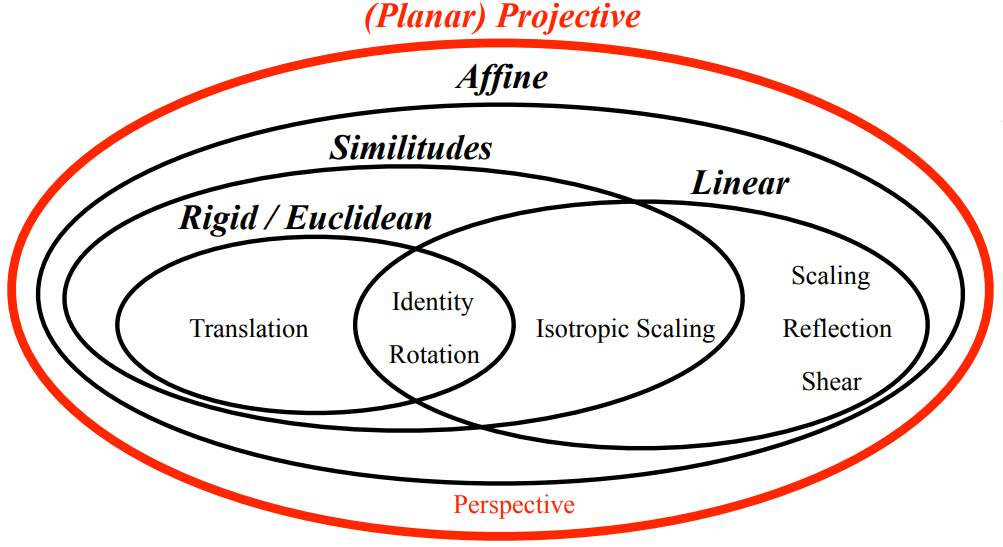
**Concurrent lines; dual to collinear points on a line** 공점선 : 공통인 한 점을 지나는 직선들



|  |  |  |
| --- | --- | --- |
| Screen%20Shot%202015-09-21%20at%209.57.27%20PM.png |  | Screen%20Shot%202015-09-21%20at%209.59.47%20PM.png |
|  |  |  |

Perspectivity property : concurrent



****

**2.7 Recovery of affine and metric properties from images**

|  |  |
| --- | --- |
| Review) **Geometry properties [[1]](#footnote-1)**   * Affine properties (line at infinity) distorted by Hp * Parallelism * Parallel length ratios * Metric properties (circular points) distorted by Ha * Angles * Length ratios | → Recover the original shape  Hp : Projective transform  Ha : Affine transform |

Projective transform된 영상에서 왜곡된 affine properties와 metric properties를 복원하면 왜곡되기 전 원영상의 모양을 복원할 수 있다.

\*\* 이해 안 되는 부분

The projective distortion in the perspective image of a plane is removed by specifying 4 points (8 DOF)

However, only 4 DOF (not 8) is required to specified in order to determine metric properties

These 4 DOF are given physical substance by associating them with geometric objects:

* The line at infinity  (2 DOF) and the two circular points (2 DOF) on 

**2.7.1 The line at infinity**

line at infinity: ideal point로 연결된 직선

* Euclidean Plane : 무한이 먼 곳 🡪 표현 x, 평행선(affine properties) 보존
* Projective transformation : ideal point, line at infinity 🡪 실제 점, 직선으로 매핑 🡺 평행성 잃음  
  🡺 affine properties 왜곡

Under a projective transformation ideal points may be mapped to finite points and consequently  is mapped to a finite line. If the transformation is an affinity, then  is not mapped to a finite line, but remais at infinity.

For an affine transformation line at infinity maps onto line at infinity

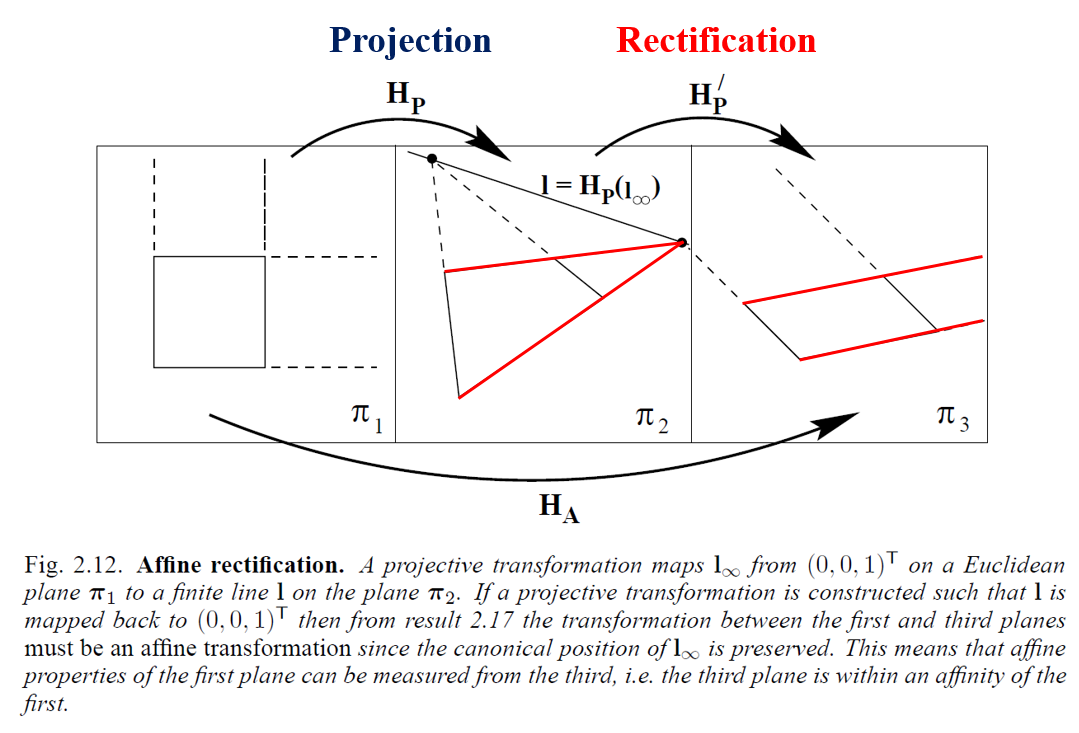


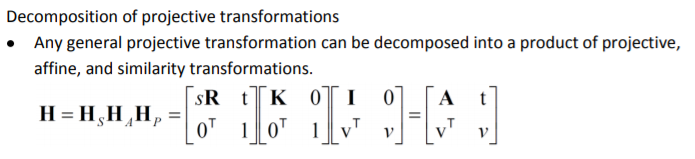
**The line at infinity  is a fixed line under the projective transformation H if and only if H is an affinity (not projectivivity)**

Note : **** is not fixed pointwise (not the same point)

A point on line at infinity is mapped to another point on the line at infinity, not necessarily the same point

Euclidean Plane에서 ideal point가 무한에서 먼 곳에서 만나기 때문에 표현 불가능  
similarity, affine 변환에도 ideal point 이동할 뿐 그 canonical position에 존재하기 때문에 line at infinity 또한 canonical position에 존재 🡺 평행선은 여전히 평행하여 affine properties 보존





Projection

Mapping of ideal point Mapping of line at infinity

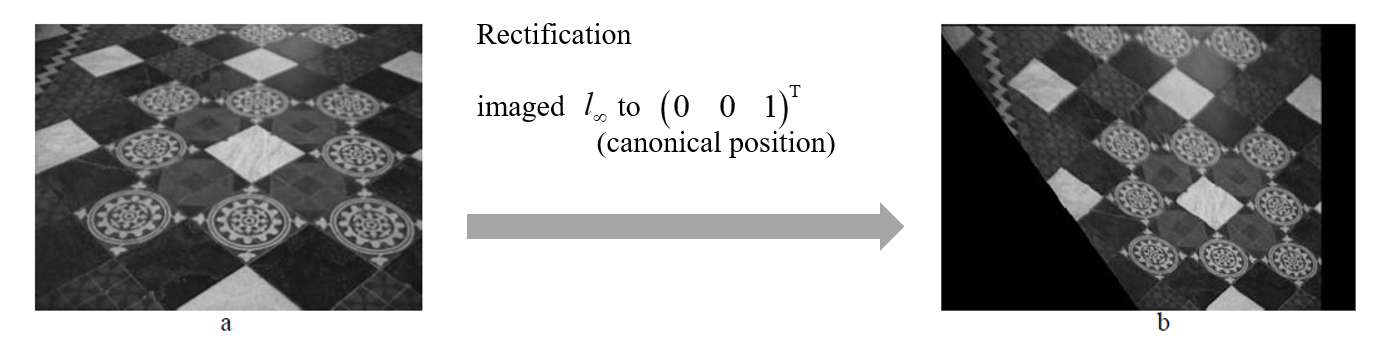
**2.7.2 Recovery of affine properties from images**

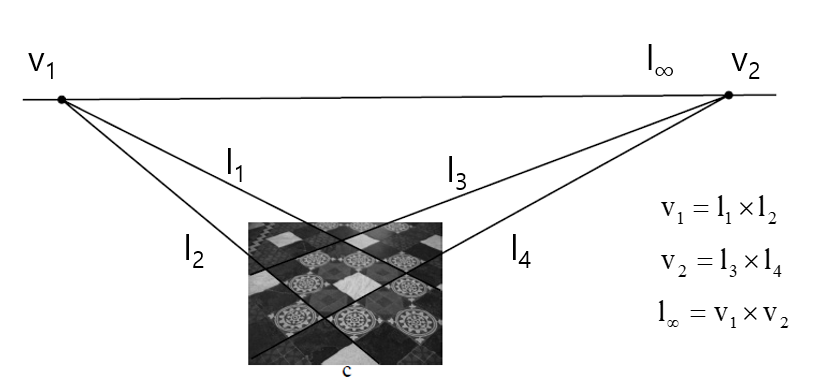
Example 2.18 Affine rectification

Moving to canonical position



평행한 직선에 다시 평행성을 복원하여 affine properties 복원 🡪 실직선으로 매핑된 를 (0,0,1)t로 이동 🡺 의 정보로 만들어진  곱하기



Metric properties(angles, length ratios)는 복원되지 않아 원본과 같지 않다.

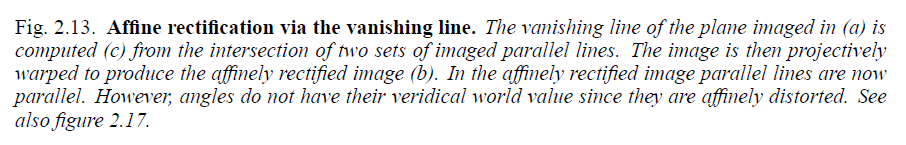
평행한 직선 2개 외적

🡪 vanishing point

2개의 vanishing point 외적

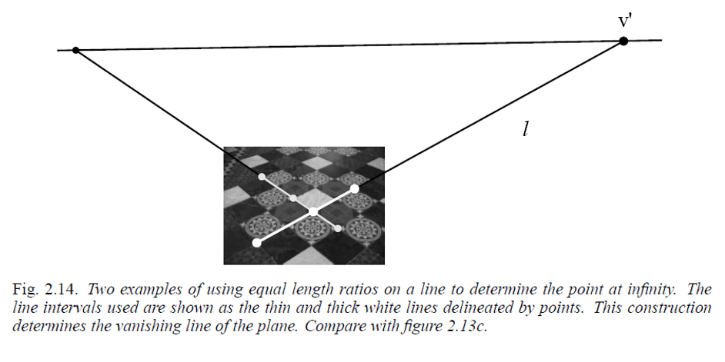
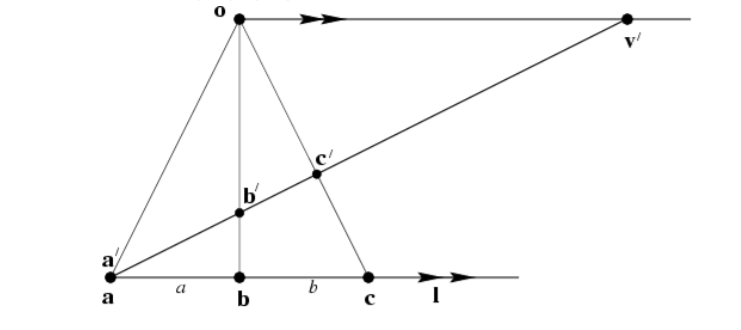
🡪 vanishing line

vanishing line 🡪 Hp’



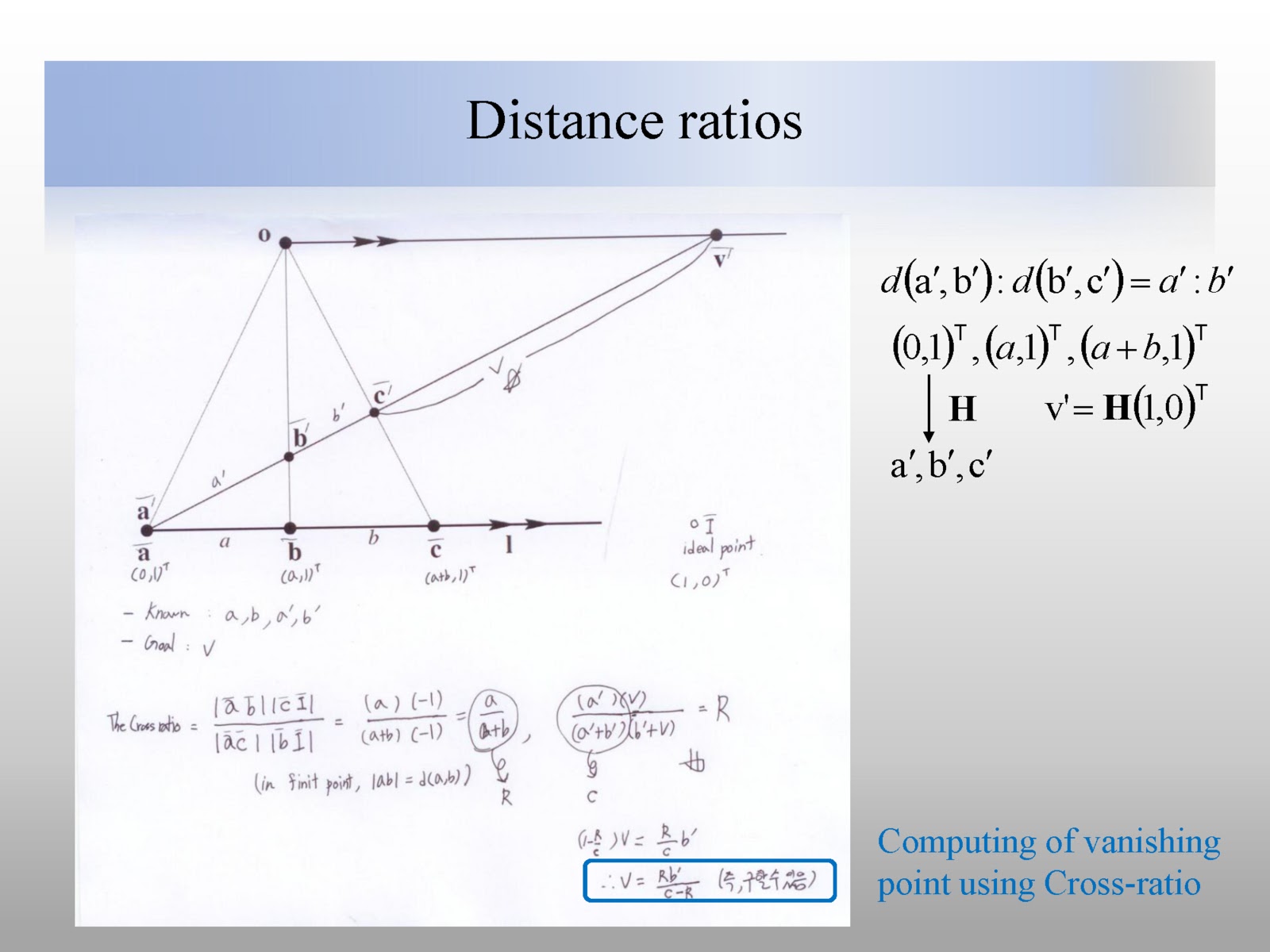
World plane

* Ideal point 🡪 vanishing point
* Line at infinity 🡪 vanishin line

Example 2.19 Computing a vanishing point from a length ratio – using invariant cross ratio

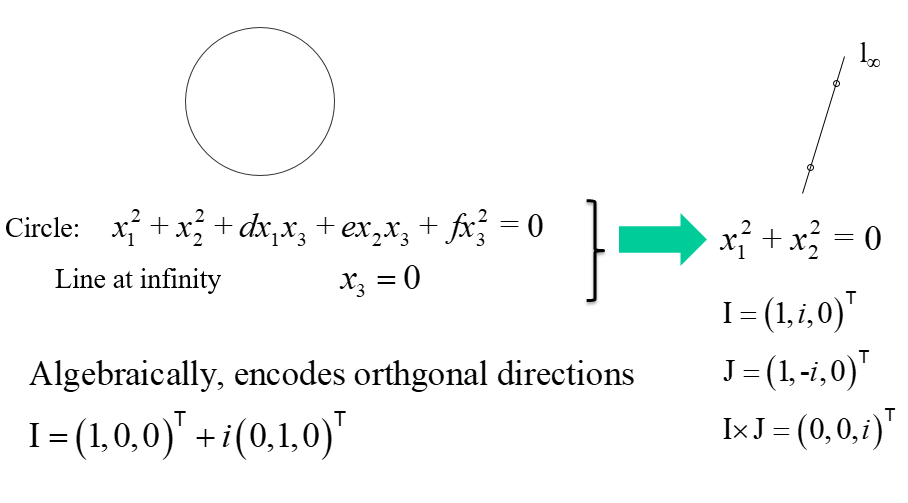


Example 2.20 Geometric construction of vanishing points from a length ratio



**2.7.3 The circular points and their dual – Metric rectification**

Two points on ** 🡪** Every circle intersets **** at circular points (absolute points)



circular points (absolute points)

* I, J : canonical coordinates를 사용하며 a pair of complex conjugate ideal points
* similarity transform에서 불변하는 점



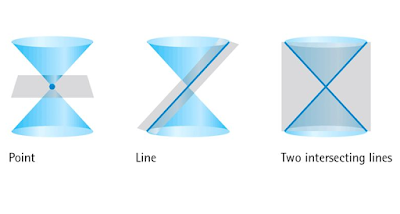
* **The circular points, I, J are fixed points under the projective transformation H iff H is a similarity.**similarity transform에 불변함

Identifying circular points allows recovery of similarity properties i.e. angles, ratios of lengths

**The conic dual to the circular points**



* **The dual conic  is fixed under the projective transformation H iff H is a similarity**.  
  두 circular points로 표현되는 degenerated dual conic은 similarity transform에 불변  
  line at infinity는 the dual cunic의 null vector (circular point가 line at infinity 위의 점임을 생각)

reminder) degenerated line conic  

Note) **** has 4 DOF (3x3 homogeneous; symmetric, determinant is zero)  
 is the null vector

**2.7.4 Angles on the projective plane**

< Angles >

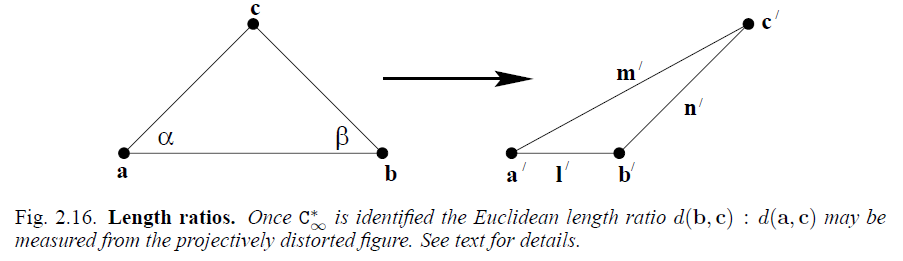


Euclidean Geometry :  not invariant under projective transformation

Projective Geometry :  invariant under projective transformation

* **Once** ** is identified in the projective plane, the Euclidean angles may be measured by (2.22)**
* **Lines l and m are orthogonal if**  **(orthogonal)**

< Length ratios >





affine transform 🡪 angles 변형됨

 이므로 분자가 0이면 두 직선 직교(orthogonal) 🡪 transformed dual conic 구함  
🡺 transformed conic dual to circular points (on the projective plane)를 identify 해서 (2.22) 식에 적용하여 원래 각도를 계산할 수 있다. (각도 복원 ~ 선분의 길이비 복원)

**2.7.5 Recovery of metric properties from images**

**Metric rectification using **

* Once the conic  is identified on the projective plane then projective distortion may be rectified up to a similarity K : 2x2 v : 2x1

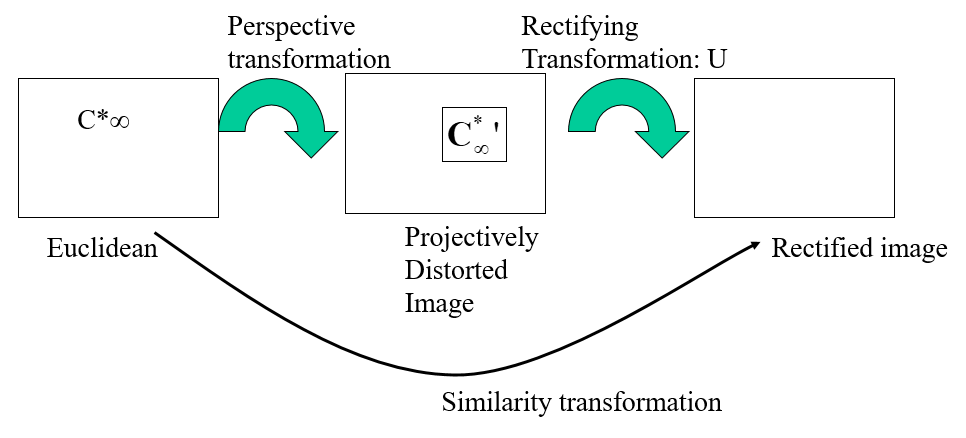
similarity transform 불변, projective(v)와 affine(K) 변환 요소만 남는다.

Rectifying transformation from SVD

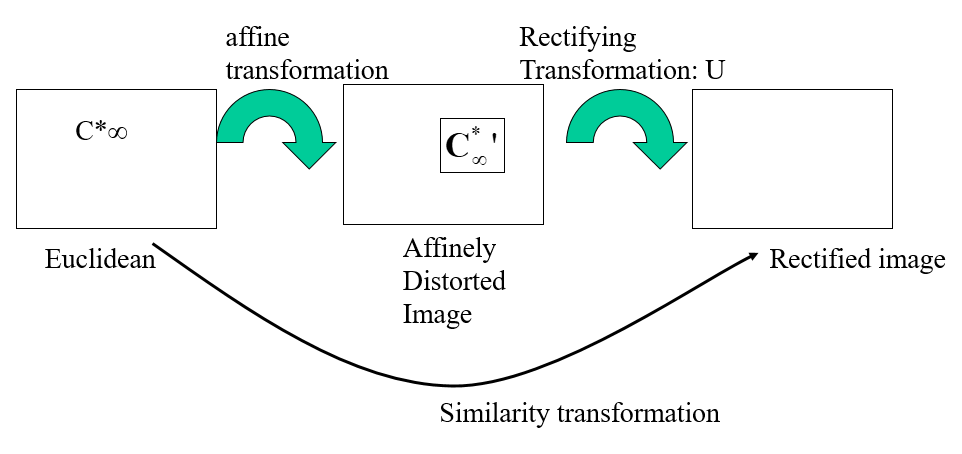


SVD decompose to get U

* Recovering up to similarity from Projective



* Recovering up to similarity from Affine



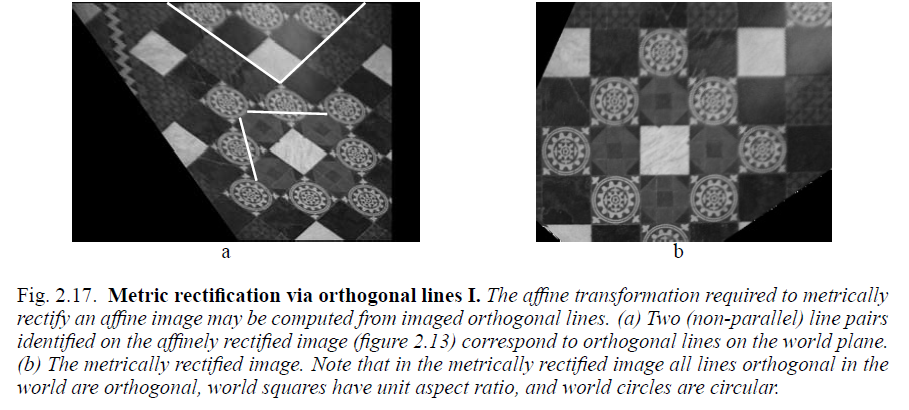
**Metric rectification I : Affine rectiffication**

if  l, m are othogonal

affine rectification 경우 Hp 변환이 제거된 것이므로 v = 0으로 K만 고려

K 2x2 🡪 3DOF : 직교하는 직선 2개를 사용해서 transformed dual conic  구할 수 있다.

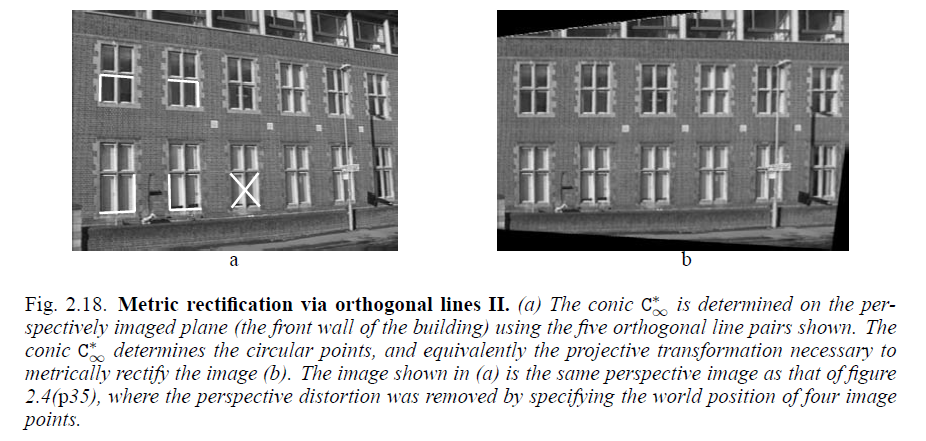


**Metric rectification II : Projective rectification (not affine rectification)**





5 DOF : 직교하는 직선 5개를 이용하여 사용해서 transformed dual conic  구할 수 있다.



**Stratification**

1. projective
2. affine distortions were removed

transformed dual conic  를 구한다는 것은 transform을 측정한 것과 같고 원 영상에서의 각도와 길이 비율을 구하여 metric properties를 복원한 것이다.

conic C

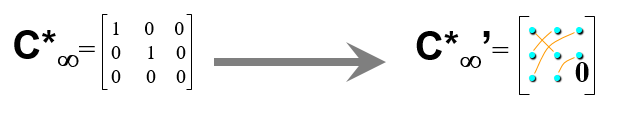


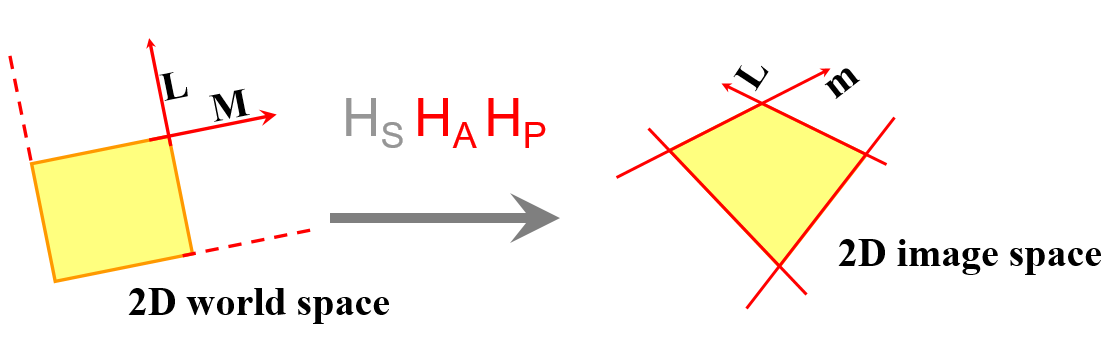
degenerate conic  

gives angle  between transformed lines l’, m; 

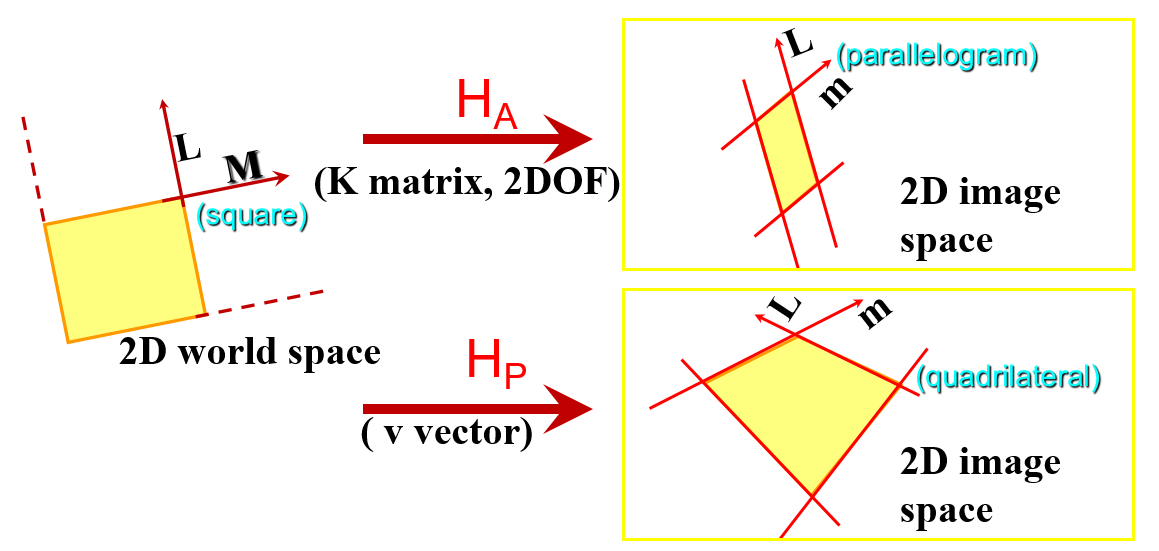
Metric rectification

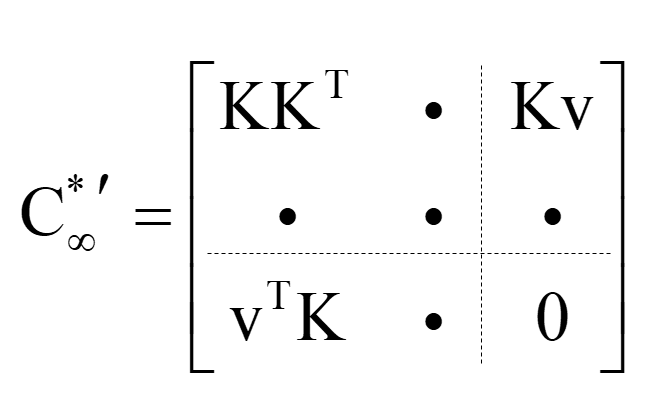
* Transformed has 4 DOF
* convert  to   matrix
* find , then find   from it...









world space transformed 

* we know 2 world space line l, m ; 
* also true for transformed l, m,  ; 
* K 2x2 symmetric (affine part : 2DOF)
* v 2x1 vector (projective part : 2DOF)

metric rectification

* method 1 : v=0, solve for K

solve for s by SVD : find input null space (Ax = 0)

* method 2 : Rearrange, solve for full  then get  using SVD



choose 5  line pairs  🡺 





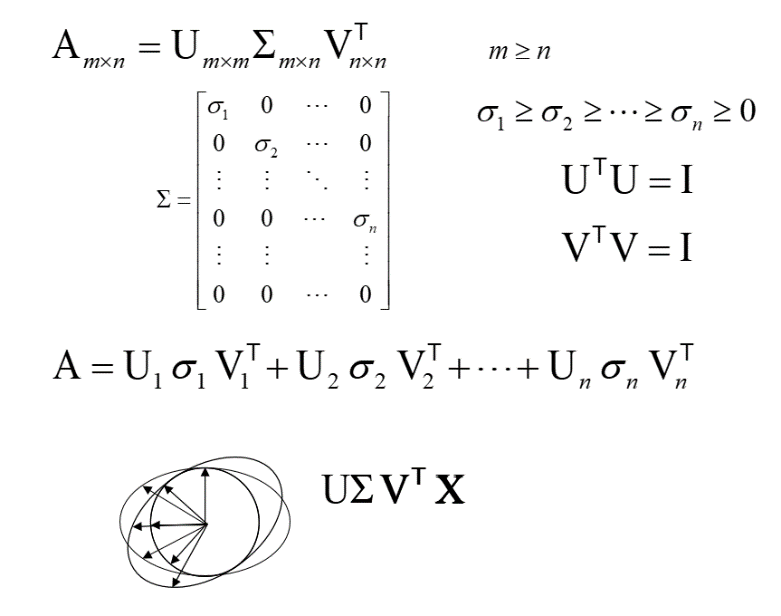
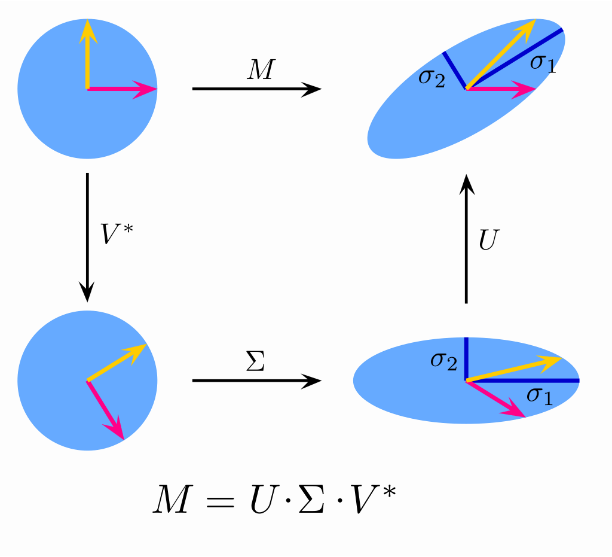
stack for all 5 line pairs

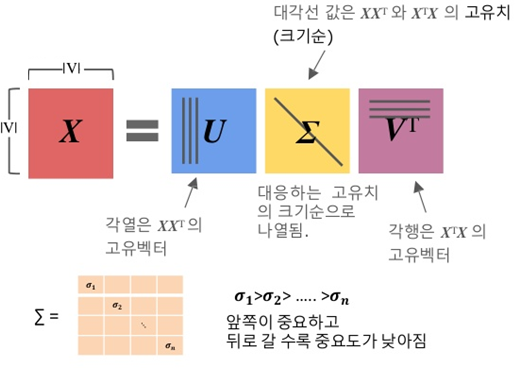


solve for a,b,c,d,e,f, with SVD (null space; Ax=0)

Extract  from  using SVD

SVD 설명 [[2]](#footnote-2) [[3]](#footnote-3)

선형변형 후에도 크기는 변하지만 여전히 직교하는 벡터의 집합 – 여러 layer로 쪼갤 수 있다.



1. http://scribbleonit.blogspot.com/2013/01/multiple-view-geometry-3.html [↑](#footnote-ref-1)
2. https://wikidocs.net/25820 [↑](#footnote-ref-2)
3. https://rfriend.tistory.com/185 [↑](#footnote-ref-3)