Image Transformations and Multiple View Geometry

4 Oct 2018

CSE473/573

Image Transformations

image filtering: change ***range*** of image

$$g(x) = T(f(x))$$

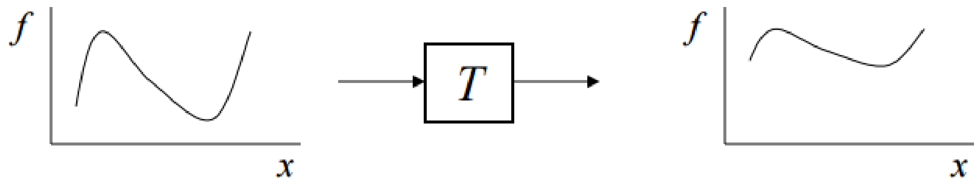


image warping: change ***domain*** of image

$$g(x) = f(T(x))$$

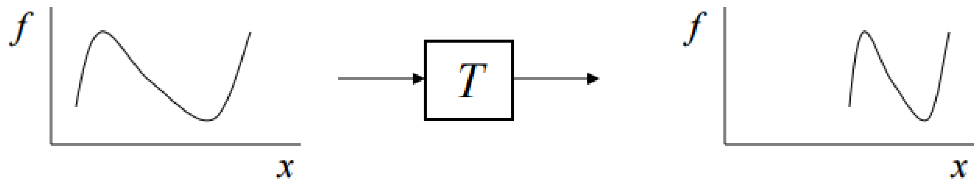


Image Transformations

image filtering: change ***range*** of image

$$g(x) = T(f(x))$$

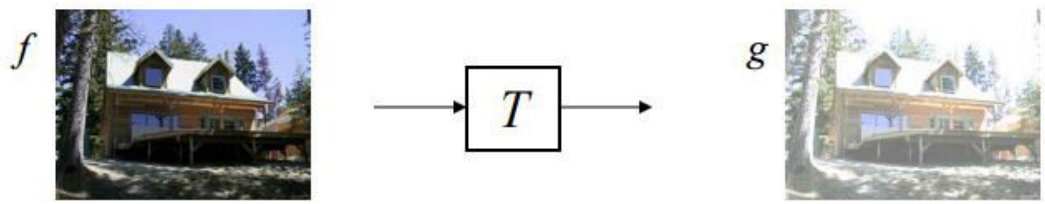
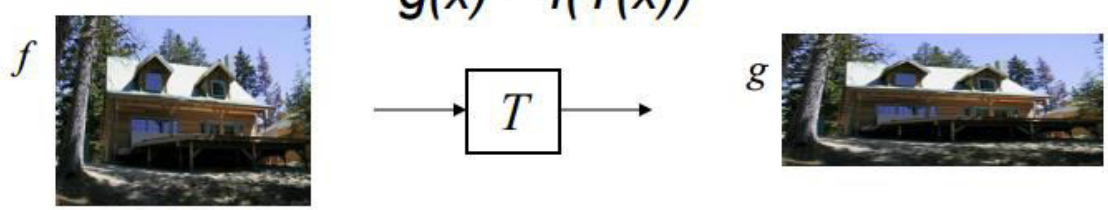


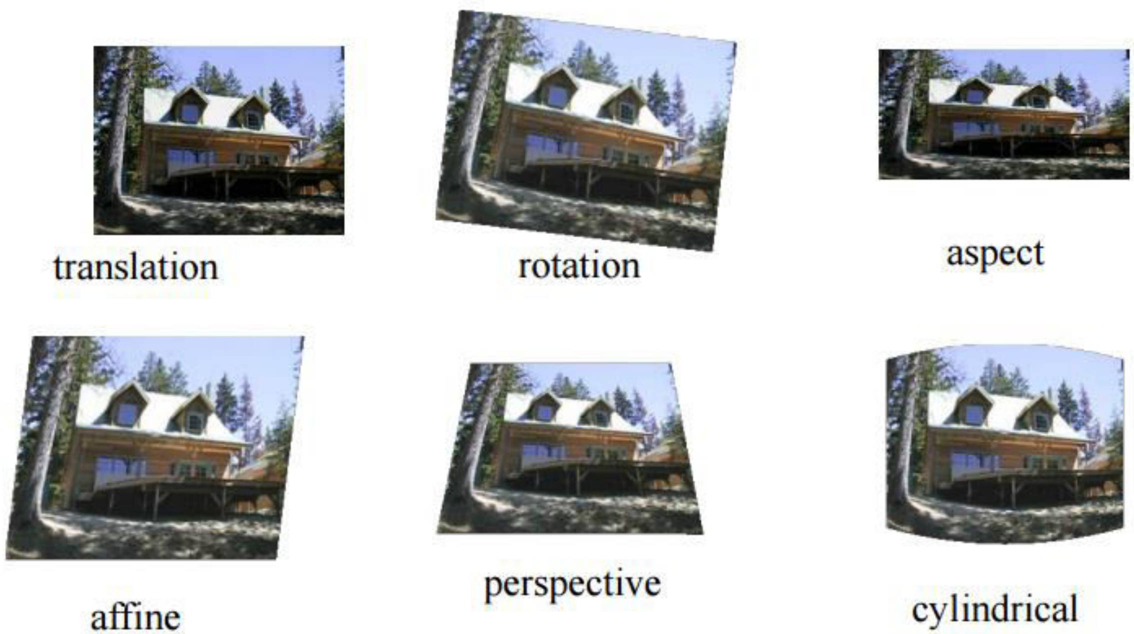
image warping: change ***domain*** of image

$$g(x) = f(T(x))$$

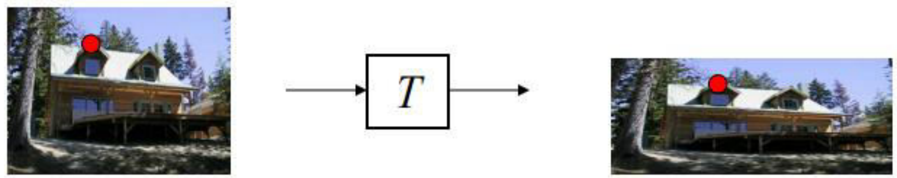


Parametric Warping

Examples of parametric warps:



Parametric (Global) Warping



p = (x,y) **p'** = (x',y')

Transformation T is a coordinate-changing machine:

$$p' = T(p)$$

What does it mean that T is global?

- Is the same for any point p
- can be described by just a few numbers (parameters)

Let's represent a linear T as a matrix:

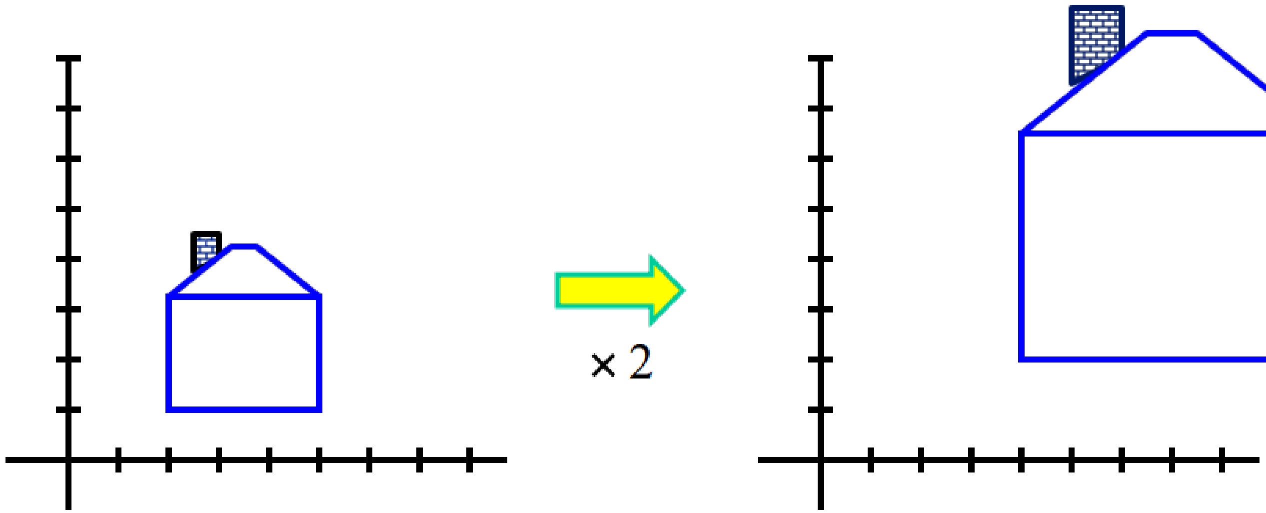
$$p' = \mathbf{M}p$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{M} \begin{bmatrix} x \\ y \end{bmatrix}$$

Scaling

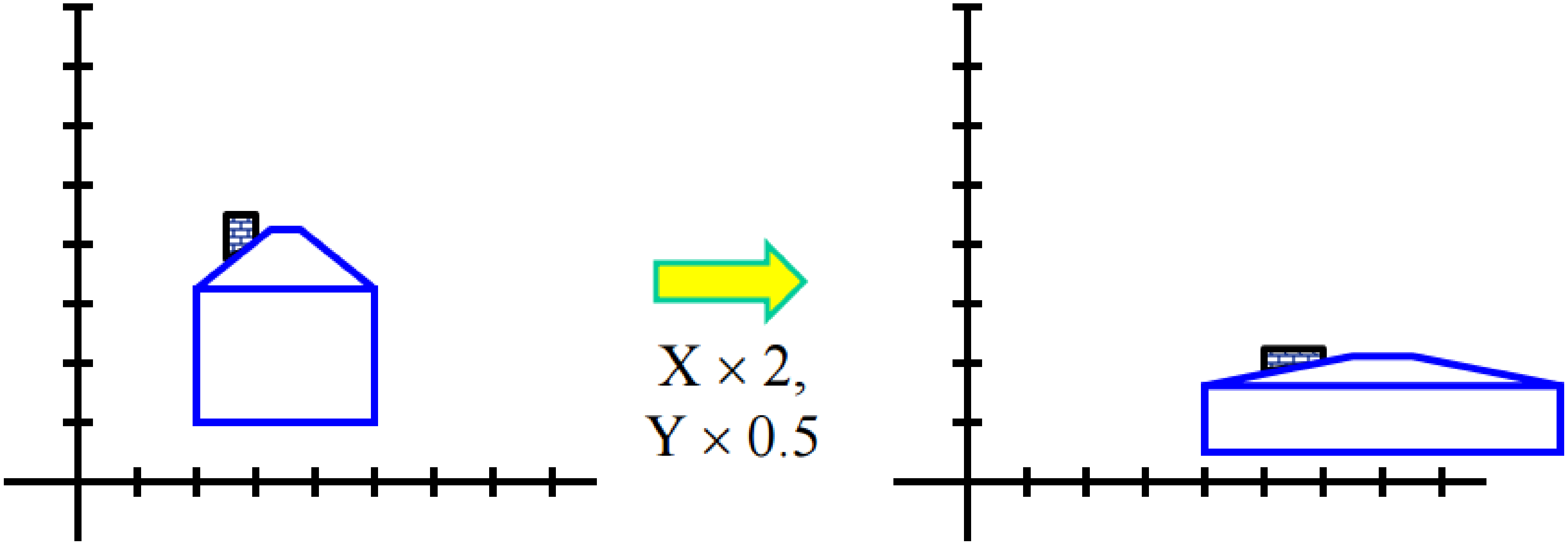
Scaling a coordinate means multiplying each of its components by a scalar

Uniform scaling means this scalar is the same for all components:



Scaling

Non-uniform scaling: different scalars per component:



Scaling

Scaling operation:

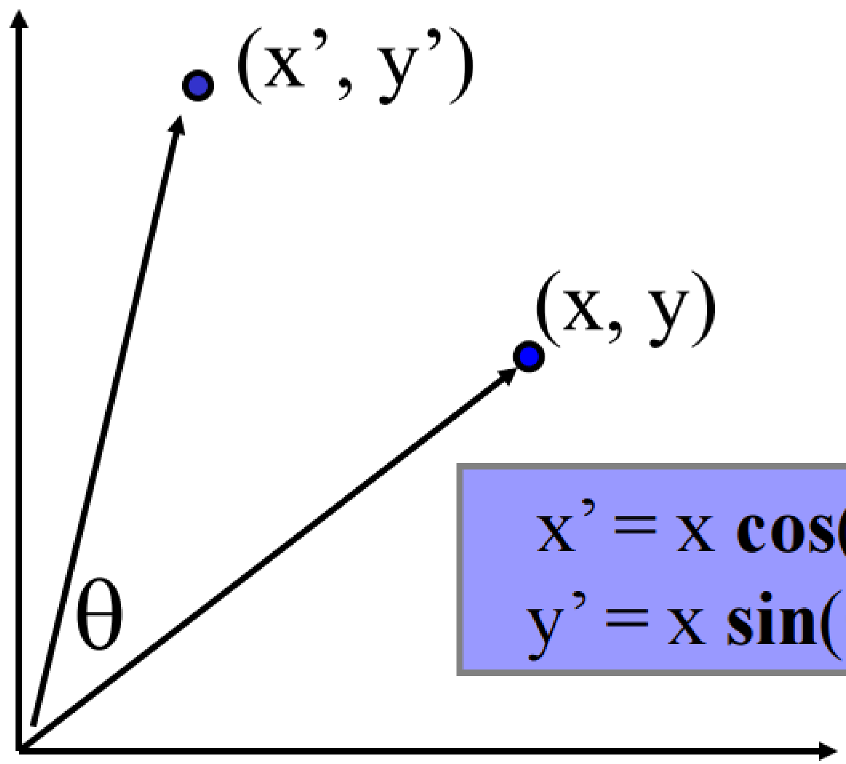
$$x' = ax$$

$$y' = by$$

Or, in matrix form:

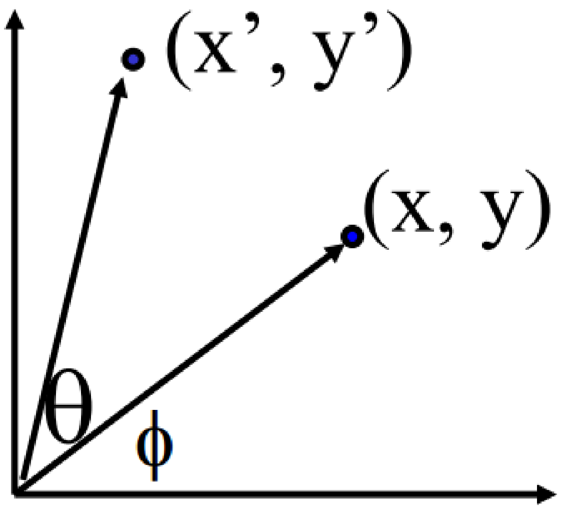
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}}_{\textit{scaling matrix } S} \begin{bmatrix} x \\ y \end{bmatrix}$$

2D - Rotation



$$x' = x\ \mathbf{cos}(\theta)\ \text{-}\ y\ \mathbf{sin}(\theta)$$
$$y' = x\ \mathbf{sin}(\theta) + y\ \mathbf{cos}(\theta)$$

2D - Rotation



$x = r \cos(\phi)$
$y = r \sin(\phi)$
$x' = r \cos(\phi + \theta)$
$y' = r \sin(\phi + \theta)$

Trig Identity…

$x' = r \cos(\phi) \cos(\theta) - r \sin(\phi) \sin(\theta)$
$y' = r \sin(\phi) \cos(\theta) + r \cos(\phi) \sin(\theta)$

Substitute…

$x' = x \cos(\theta) - y \sin(\theta)$
$y' = x \sin(\theta) + y \cos(\theta)$

2D - Rotation

This is easy to capture in matrix form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}}_{\mathbf{R}} \begin{bmatrix} x \\ y \end{bmatrix}$$

Even though sin(θ) and cos(θ) are nonlinear functions of θ,

- ***x' is a linear combination of x and y***
- ***y' is a linear combination of x and y***

What is the inverse transformation?

- Rotation by –θ
- For rotation matrices $\mathbf{R}^{-1} = \mathbf{R}^{T}$

2D - Rotation

This is easy to capture in matrix form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}}_{\mathbf{R}} \begin{bmatrix} x \\ y \end{bmatrix}$$

Even though sin(θ) and cos(θ) are nonlinear functions of θ,

- ***x' is a linear combination of x and y***
- ***y' is a linear combination of x and y***

What is the inverse transformation?

- Rotation by –θ
- For rotation matrices $\mathbf{R}^{-1} = \mathbf{R}^{T}$

2x2 Matrices

What types of transformations can be represented with a 2x2 matrix?

2D Mirror about Y axis?

$$x' = -x$$
$$y' = y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

2D Mirror over (0,0)?

$$x' = -x$$
$$y' = -y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

2x2 Matrices

What types of transformations can be represented with a 2x2 matrix?

2D Translation?

$$x' = x + t_x$$
$$y' = y + t_y$$

NO!

Only linear 2D transformations can be represented with a 2x2 matrix

2x2 Matrices

Linear transformations are combinations of …

- Scale,
- Rotation,
- Shear, and
- Mirror

Properties of linear transformations:

- Origin maps to origin
- Lines map to lines
- Parallel lines remain parallel
- Ratios are preserved
- Closed under composition

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} \begin{bmatrix} i & j \\ k & l \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Homogeneous Co-ordinates

Q: How can we represent translation as a 3x3 matrix?

$$x' = x + t_x$$
$$y' = y + t_y$$

Homogeneous Co-ordinates

Homogeneous coordinates

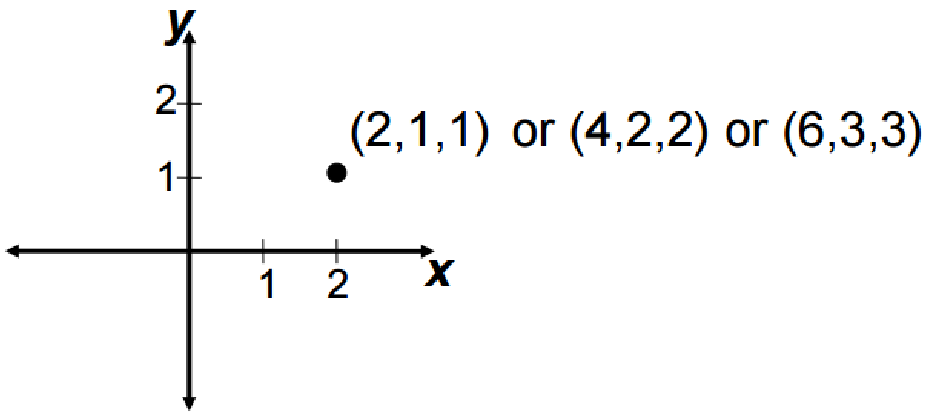
- represent coordinates in 2 dimensions with a 3-vector

$$\begin{bmatrix} x \\ y \end{bmatrix} \xrightarrow{\text{homogeneous coords}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Homogeneous Co-ordinates

Add a 3rd coordinate to every 2D point

- (x, y, w) represents a point at location (x/w, y/w)
- (x, y, 0) represents a point at infinity
- (0, 0, 0) is not allowed



Convenient coordinate system to represent many useful transformations

Homogeneous Co-ordinates

Q: How can we represent translation as a 3x3 matrix?

$$x' = x + t_x$$

$$y' = y + t_y$$

A: Using the rightmost column:

$$\mathbf{T}ranslation = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Basic 2D Transformations

Basic 2D transformations as 3x3 matrices

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Translate

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{s}_x & 0 & 0 \\ 0 & \boldsymbol{s}_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ 1 \end{bmatrix}$$

Scale

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\Theta & -\sin\Theta & 0 \\ \sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Rotate

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & \boldsymbol{sh}_x & 0 \\ \boldsymbol{sh}_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ 1 \end{bmatrix}$$

Shear

Matrix Composition

Transformations can be combined by matrix multiplication

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \left(\begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\Theta & -\sin\Theta & 0 \\ \sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} sx & 0 & 0 \\ 0 & sy & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

$$\mathbf{p'} \quad = \quad T(t_x,t_y) \qquad R(\Theta) \qquad S(s_x,s_y) \qquad \mathbf{p}$$

Affine Transformation

$$\begin{bmatrix} x' \\ y' \\ w \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

Affine transformations are combinations of …

- Linear transformations, and
- Translations

Properties of affine transformations:

- Origin does not necessarily map to origin
- Lines map to lines
- Parallel lines remain parallel
- Ratios are preserved
- Closed under composition
- Models change of basis

Will the last coordinate *w* always be 1?

Projective Transformation

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

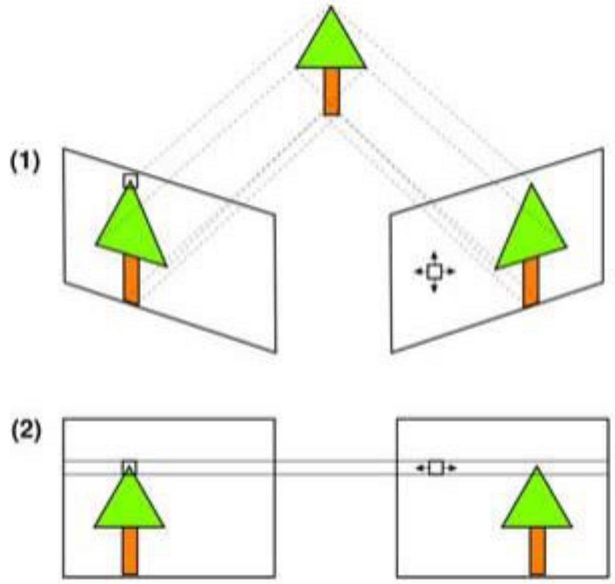
Projective transformations ...

- Affine transformations, and
- Projective warps

Properties of projective transformations:

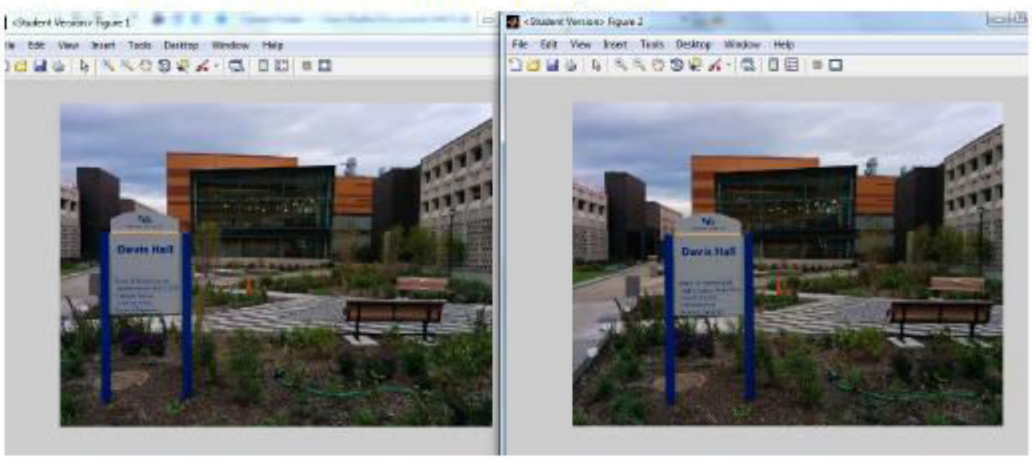
- Origin does not necessarily map to origin
- Lines map to lines
- Parallel lines do not necessarily remain parallel
- Ratios are not preserved
- Closed under composition
- Models change of basis

Image Rectification for Stereo



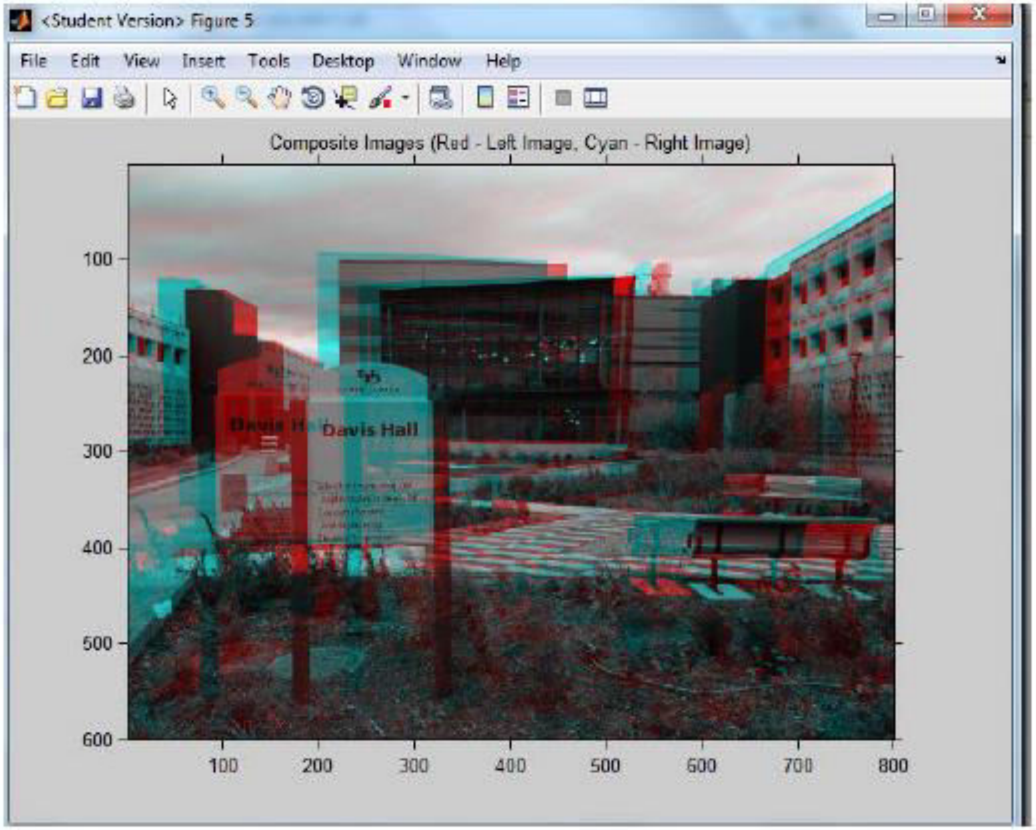
Stereo Rectifictation

- Rectification is the process of transforming stereo images, such that the corresponding points have the same row coordinates in the two images.
- It is a useful procedure in stereo vision, as the 2-D stereo correspondence problem is reduced to a 1-D problem
- Let’s see the rectification pipeline when we have are two images of the same scene taken from a camera from different viewpoints

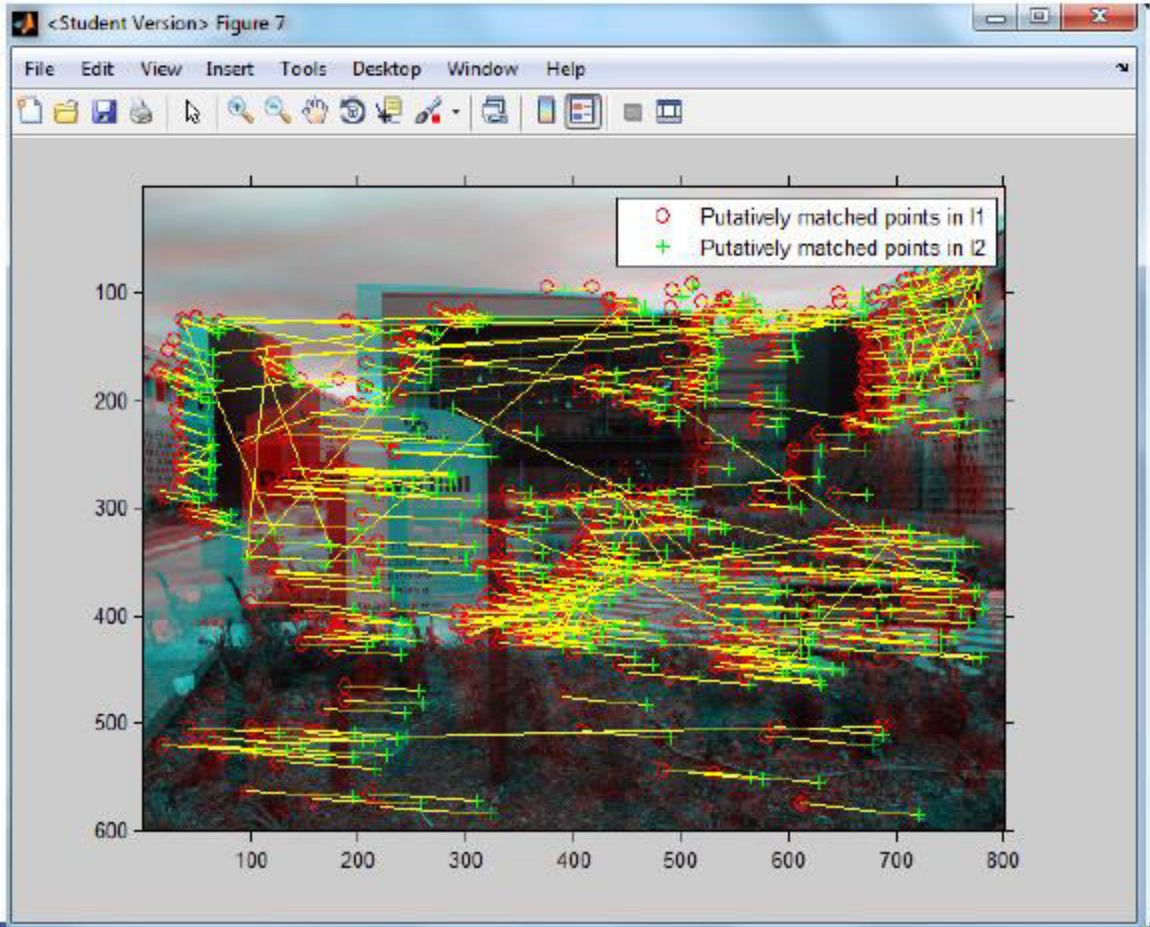


Stereo Input Images

Superposing the two input images on each other and compositing

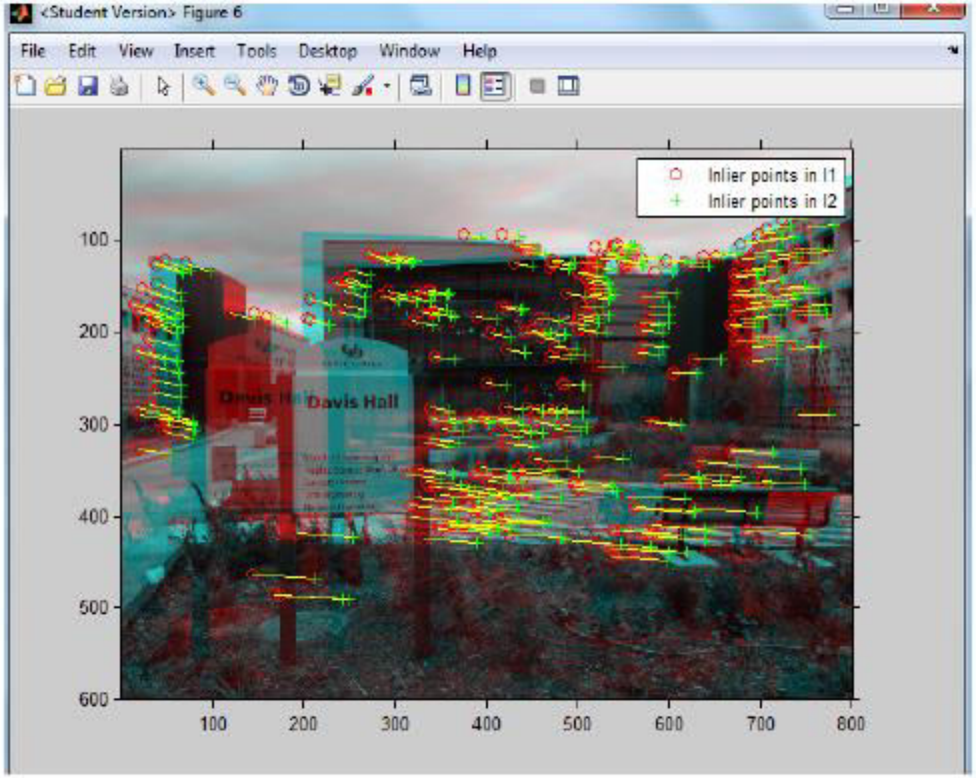


Stereo Input Images



Stereo Input Images

We can impose geometric constraints while applying RANSAC for eliminating outliers

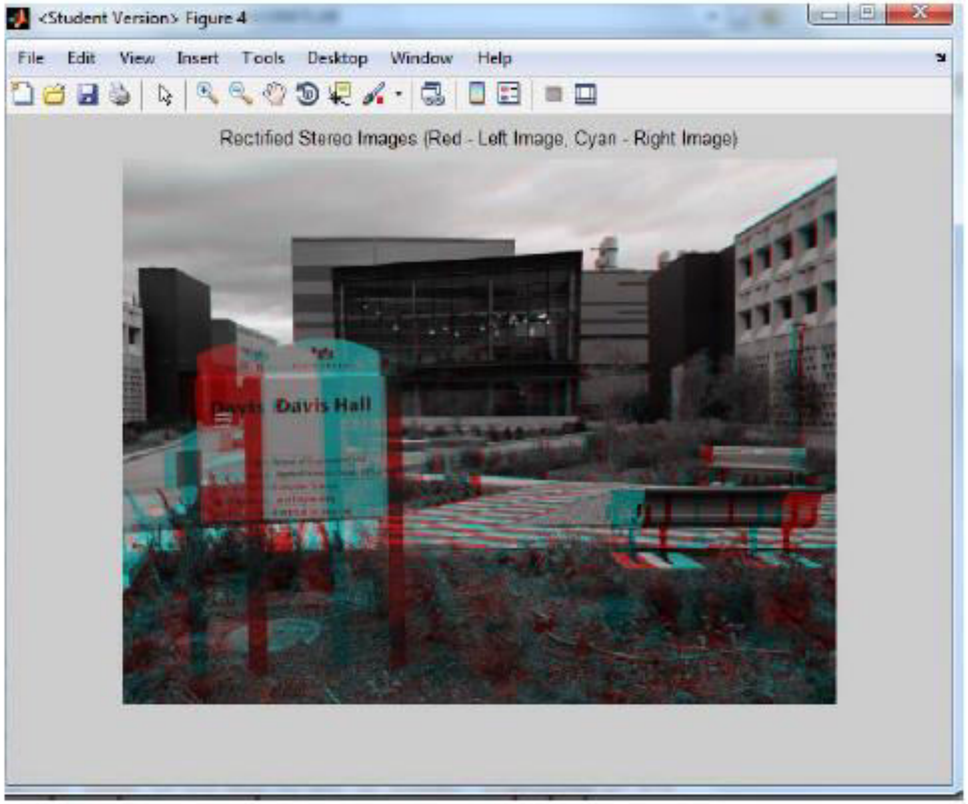


Stereo Input Images

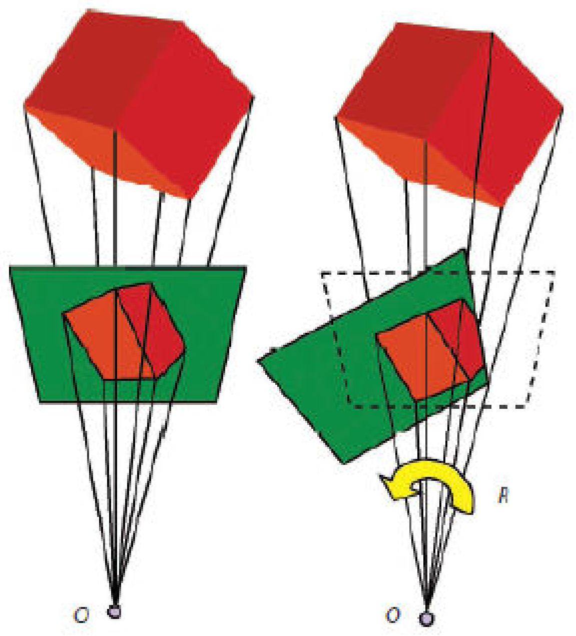
```
fMatrix = estimateFundamentalMatrix( matchedPtsOut.Location,
matchedPtsIn.Location);
```

```
EDU>> fMatrix

fMatrix =

   -0.0000    0.0001   -0.0188
   -0.0001   -0.0000   -0.0340
    0.0178    0.0317    0.9986
```

Rectified Stereo Input Images



Pure Rotational Model of Camera - Homography



$$R_{12} = R_z(\alpha)R_y(\beta)R_x(\gamma)$$

$$R_z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R_y(\beta) = \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix}$$

$$R_z(\gamma) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{pmatrix}.$$

α,β,γ are angle changes across roll, pitch and yaw

$$\mathrm{P}_i = \mathrm{K}_i\,[\mathrm{R}_i|0]$$

$$\mathbf{x} = \mathrm{K}_i\,[\mathrm{R}_i|0] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathrm{K}_i\mathrm{R}_i \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \mathrm{K}_i\mathrm{R}_i\mathbf{X}$$

Pure Rotational Model of Camera - Homography

Suppose we have two images of a scene captured from a rotating camera

point ‘x_1’ in Image1 is related to the world point ‘X’ by the equation

$x_1 = KR_1X$ which implies $X = R_1^{-1}K^{-1} x_1$ as $B^{-1} A^{-1} = (A B)^{-1}$.

point ‘x_2’ in Image2 is related to the world point ‘X’ by the equation

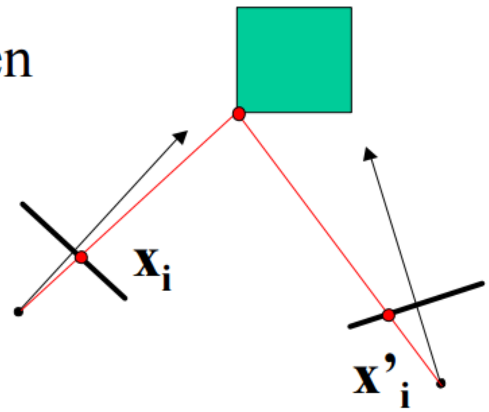
$x_2 = KR_2X = KR_2R_1^{-1}K^{-1} * x_1$

Hence the points in both the images are related to each other by a transformation of Homography ‘H’

$x_2 = H x_1$ Where $H = KR_2R_1^{-1}K^{-1}$

Fundamental Matrix

- Projective geometry between two views
- Independent of scene structure
- Depends only on the cameras' internal parameters and relative pose of cameras
- Fundamental matrix **F** encapsulates this geometry



$$\boxed{\mathbf{x'}_i^{T}\,\mathbf{F}\,\mathbf{x}_i = 0}$$

for any pair of corresponding points $\mathbf{x}_i$ and $\mathbf{x'}_i$ in the 2 images

Fundamental Matrix Song!

https://www.youtube.com/watch?v=DgGV3l82NTk

Preliminaries

- Linear Independence
- Rank of a Matrix
- Matrix Norm
- Singular Value Decomposition
- Vector Cross product to Matrix Multiplication
- RANSAC

Linear dependence

A finite subset of n vectors, $\mathbf{v}_1$, $\mathbf{v}_2$, ..., $\mathbf{v}_n$, from the vector space V, is ***linearly dependent*** if and only if there exists a set of n scalars, a_1, a_2, ..., a_n, not all zero, such that

$$a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + \cdots + a_n\mathbf{v}_n = \mathbf{0}.$$

Rank of a Matrix

- The **column rank** of a matrix A is the maximum number of linearly independent column vectors of A.
- The **row rank** of a matrix A is the maximum number of linearly independent row vectors of A.

Rank of a Matrix - Example

$$A = \begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix} R_2 \rightarrow 2r_1 + r_2$$

Rank is 2

Singular Value Decomposition (SVD)

Theorem: Any *m* by *n* matrix A, for which $m \geq n$,can be written as

$$A = O_1 \Sigma O_2$$

mxn mxn nxn nxn

Σ is diagonal

O_1 ,O_2 are orthogonal

$O_1^T O_1 = O_2^T \ O_2 = I$

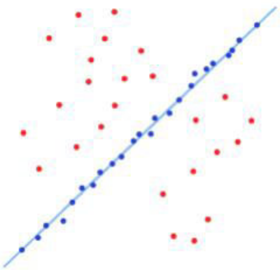
Vector Cross-product to Matrix-Vector Multiplication

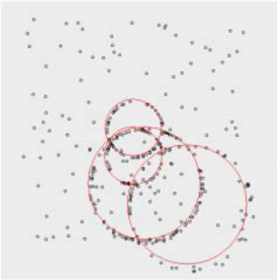
$$A \times B = \begin{bmatrix} i & j & k \\ A_x & A_y & A_z \\ B_x & B_y & B_z \end{bmatrix} = \begin{bmatrix} -A_z B_y + A_y B_z & A_z B_x - A_x B_z & -B_x A_y + B_y A_x \end{bmatrix}$$

$$A \times B = S.B = \begin{bmatrix} 0 & -A_z & A_y \\ A_z & 0 & -A_x \\ -A_y & A_x & 0 \end{bmatrix} \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} = \begin{bmatrix} -A_z B_y + A_y B_z & A_z B_x - A_x B_z & -B_x A_y + B_y A_x \end{bmatrix}$$

RANSAC

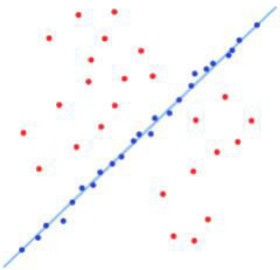
- Random Sample Consensus
- Used for Parametric Matching/Model Fitting
- Applications:

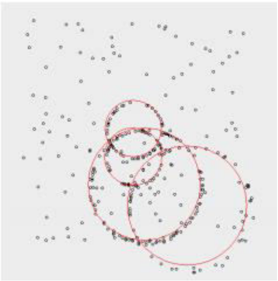




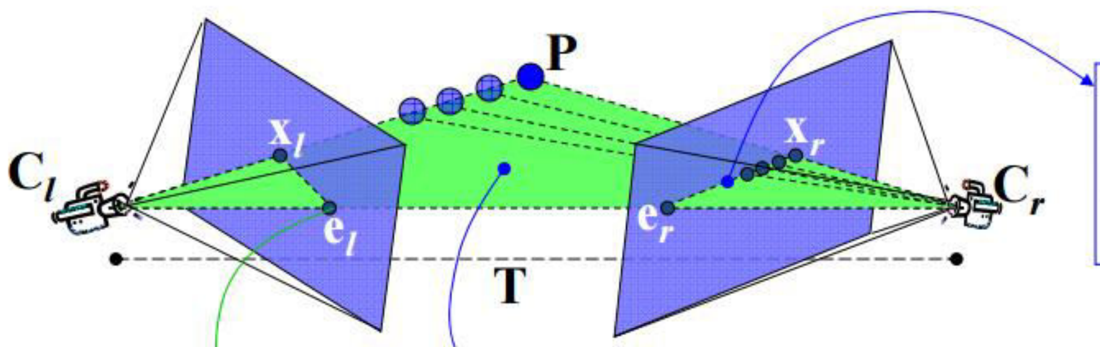
RANSAC

- Random Sample Consensus
- Used for Parametric Matching/Model Fitting
- Applications:





Epi-polar Geometry

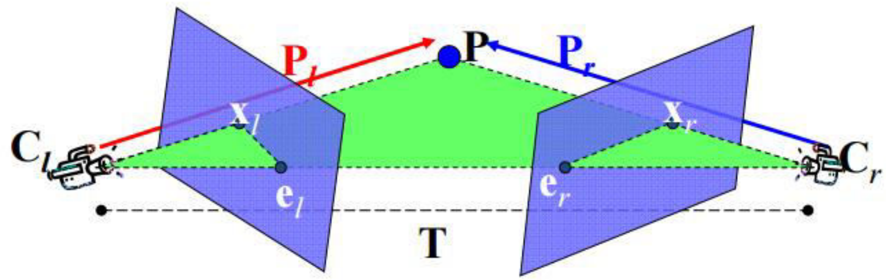


Epipolar line: set of world points that project to the same point in left image, when projected to right image forms a line.

Epipolar plane: plane defined by the camera centers and world point.

Epipole: intersection of image plane with line connecting camera centers. Image of a left camera center in the right, and vice versa.

Co-Planarity Constraint



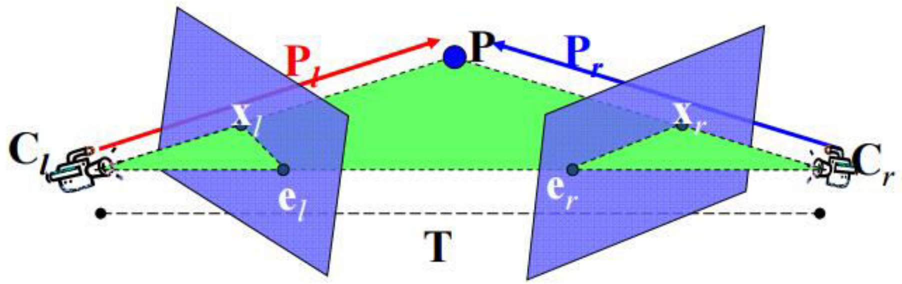
Coplanarity constraint between vectors ($\mathbf{P}_l$-$\mathbf{T}$), $\mathbf{T}$, $\mathbf{P}_l$.

$$\left.\begin{aligned}(\mathbf{P}_l - \mathbf{T})^{\mathrm{T}}\mathbf{T} \times \mathbf{P}_l = 0 \\ \mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T})\end{aligned}\right\} \quad \mathbf{P}_r^{\mathrm{T}}\mathbf{R}\mathbf{T} \times \mathbf{P}_l = 0$$

$$\mathbf{R}^T\mathbf{P}_r = (\mathbf{P}_l - \mathbf{T})$$

$$\mathbf{P}_\mathbf{r}^T\mathbf{R} = (\mathbf{P}_l - \mathbf{T})^T$$

Essential Matrix - E



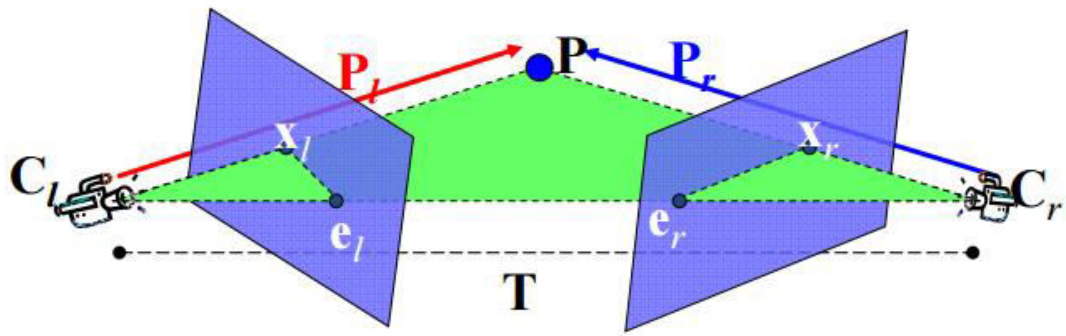
$$\left.\mathbf{P}_r^{T}\mathbf{R}\mathbf{T}\times\mathbf{P}_l = 0 \quad\right\} \quad \mathbf{P}_r^{T}\mathbf{R}\,\boxed{\mathbf{S}}\,\mathbf{P}_l = 0 \quad \Longrightarrow \quad \boxed{\mathbf{P}_r^{T} E \mathbf{P}_l = 0}$$

essential matrix

$$\mathbf{S} = \begin{pmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{pmatrix}$$

$$E = \mathbf{R}\ \mathbf{S}$$

Fundamental Matrix - F



Apply Camera model

$$M_l^{-1}\mathbf{x}_l = \mathbf{P}_l$$

$$M_r^{-1}\mathbf{x}_r = \mathbf{P}_r$$

$$\mathbf{x}_r^{T} M_r^{-T} = \mathbf{P}_r^{T}$$

$$\left.\begin{array}{l} \mathbf{x}_l = M_l \mathbf{P}_l \\ \mathbf{x}_r = M_r \mathbf{P}_r \\ \mathbf{P}_r^{\mathrm{T}} E \mathbf{P}_l = 0 \end{array}\right\}$$

$$\mathbf{x}_r^{\mathrm{T}} M_r^{-\mathrm{T}} E M_l^{-1} \mathbf{x}_l = 0$$

$$\mathbf{x}_r^{\mathrm{T}}\left(M_r^{-\mathrm{T}} E M_l^{-1}\right)\mathbf{x}_l = 0$$

$$\boxed{\mathbf{x}_r^{\mathrm{T}} F \mathbf{x}_l = 0}$$

fundamental matrix

Fundamental and Essential Matrices

Stereo Images have both rotation and translation of camera

the fundamental matrix 'F' is a 3×3 matrix which relates corresponding points x and x^1 in stereo images.

$$\mathbf{x}'^{\top}\mathbf{F}\mathbf{x} = 0.$$

It captures the essence of Epipolar constraint in the Stereo images.

Essential Matrix

$$\mathbf{E} = \mathbf{K}'^{\top}\ \mathbf{F}\ \mathbf{K}$$

Where K and K^1 are the Intrinsic parameters of the cameras capturing x and x^1 respectively

http://en.wikipedia.org/wiki/Eight-point_algorithm

Fundamental Matrix

- Fundamental matrix captures the relationship between the corresponding points in two views.

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}^T \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} = 0,$$

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}^T \begin{bmatrix} f_{11}x' + f_{12}y_i' + f_{13} \\ f_{21}x' + f_{22}y_i' + f_{23} \\ f_{31}x' + f_{32}y_i' + f_{33} \end{bmatrix} = 0,$$

Fundamental Matrix

You need at-least 8 point correspondences to solve for this

$$x_i(f_{11}x' + f_{12}y_i' + f_{13}) + y_i(f_{21}x' + f_{22}y_i' + f_{23}) + (f_{31}x' + f_{32}y_i' + f_{33}) = 0$$

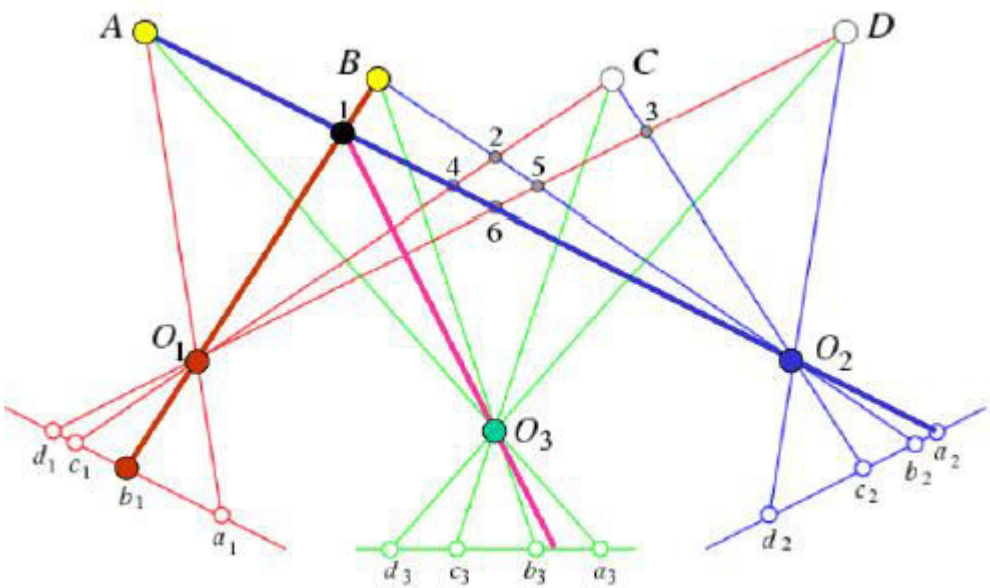
$$x_i x' f_{11} + x_i y_i' f_{12} + x_i f_{13} + y_i x' f_{21} + x' y_i' f_{22} + y_i' f_{23} + x' f_{31} + y_i' f_{32} + f_{33} = 0$$

One equation for one point correspondence

$$Mf = \begin{bmatrix} x_1'x_1 & x_1'y_1 & x_1' & y_1'x_1 & y_1'y_1 & y_1' & x_1 & y_1 & 1 \\ x_2'x_2 & x_2'y_2 & x_2' & y_2'x_2 & y_2'y_2 & y_2' & x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n'x_n & x_n'y_n & x_1' & y_n'x_n & y_n'y_n & y_n' & x_n & y_n & 1 \end{bmatrix} f = 0$$

M is 9 by n matrix $\quad f = [f_{11} \quad f_{12} \quad f_{13} \quad f_{21} \quad f_{22} \quad f_{23} \quad f_{31} \quad f_{32} \quad f_{33}]$

Beyond Two Views



Third View can be used for verification

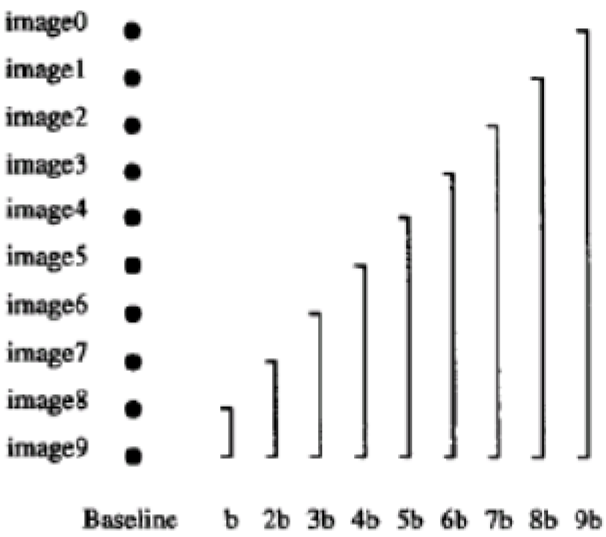
Multi-View Video in Dynamic Scenes

Multi-View Reconstruction

Generic problem formulation: given several images of the same object or scene, compute a representation of its 3D shape

Multi-View Reconstruction

Pick a reference image, and slide the corresponding window along the corresponding epipolar lines of all other images using other images



Remember? disparity

Where B is baseline, f is $d = \frac{Bf}{Z}$
and Z is the depth

This equation indicates that for the same depth the disparity is proportional to the baseline

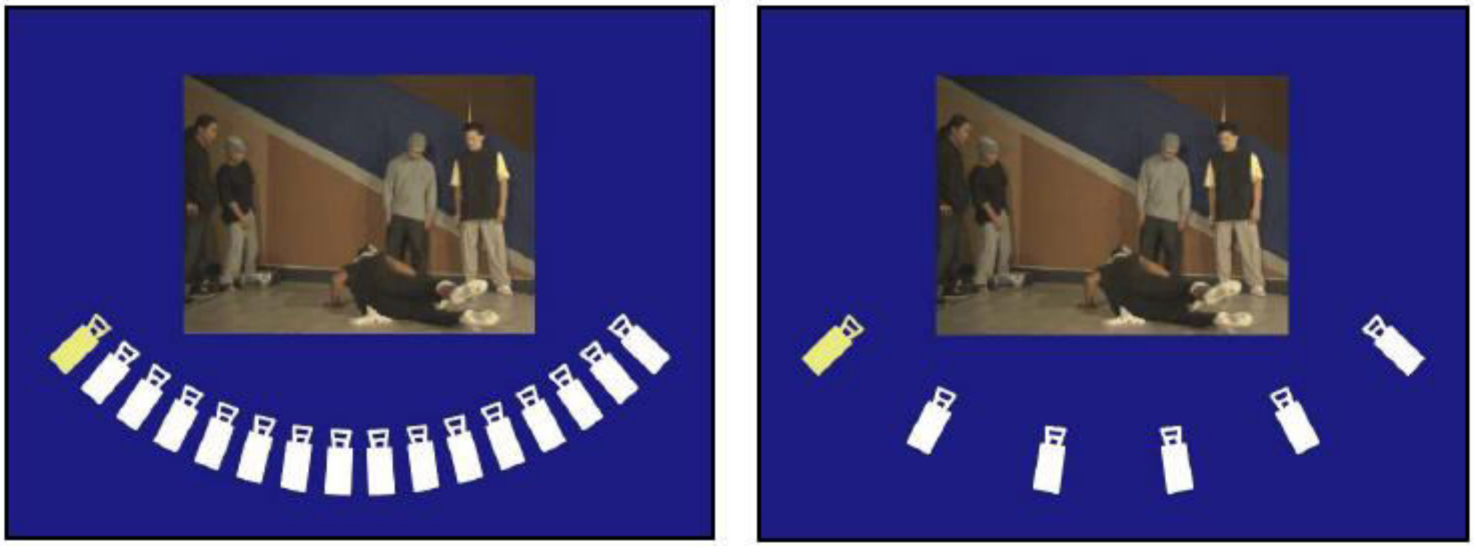
M. Okutomi and T. Kanade, "A Multiple-Baseline Stereo System," IEEE Trans. on Pattern Analysis and Machine Intelligence, 15(4):353-363 (1993)

Feature Matching to Dense Stereo

1. Extract features
2. Get a sparse set of initial matches
3. Iteratively expand matches to nearby locations Iteratively expand matches to nearby locations
4. Use visibility constraints to filter out false matches
5. Perform surface reconstruction

Yasutaka Furukawa and Jean Ponce, [Accurate, Dense, and Robust Multi-View Stereopsis](), CVPR 2007.

View Synthesis

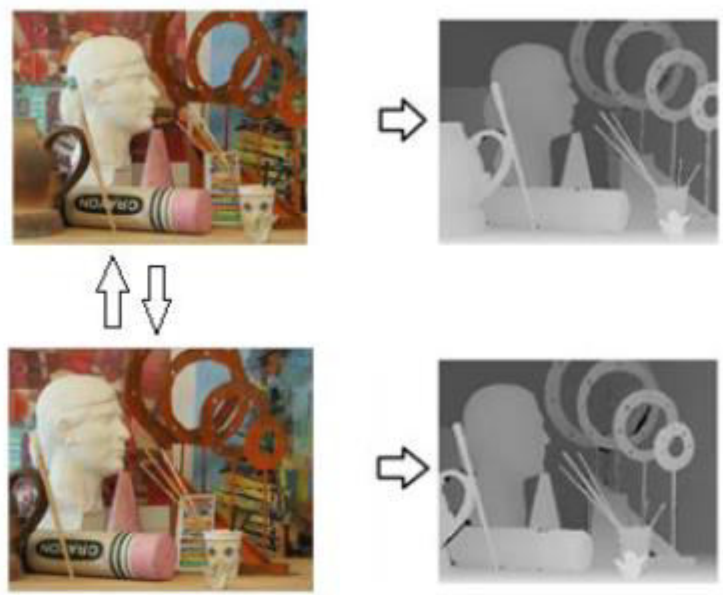


Is it possible to synthesize views from the locations where the cameras are removed? i.e Can we synthesize view from a virtual camera

View Synthesis

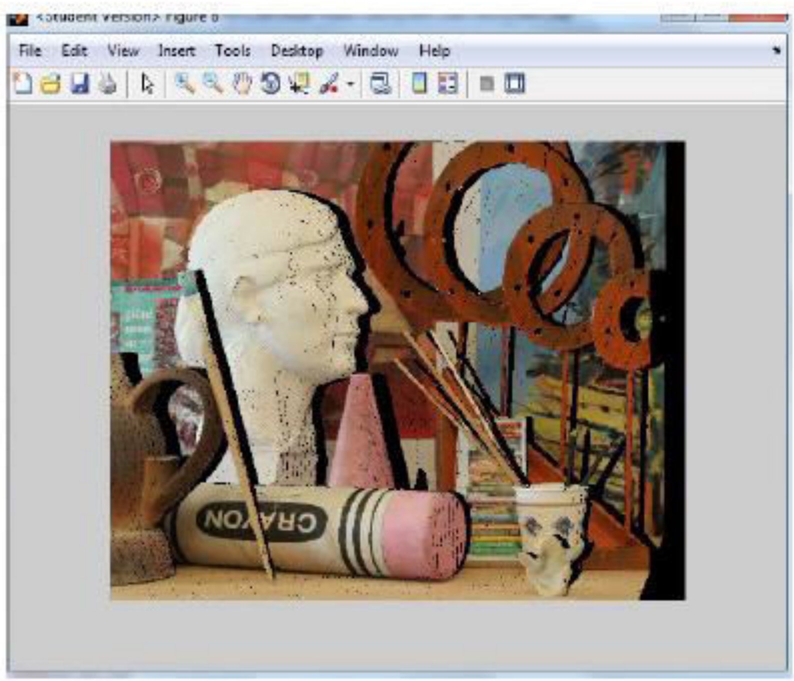
Problem: Synthesize virtual view of the scene at the mid point of line joining Stereo camera centers.

Given stereo images, find Stereo correspondence and disparity estimates between them.



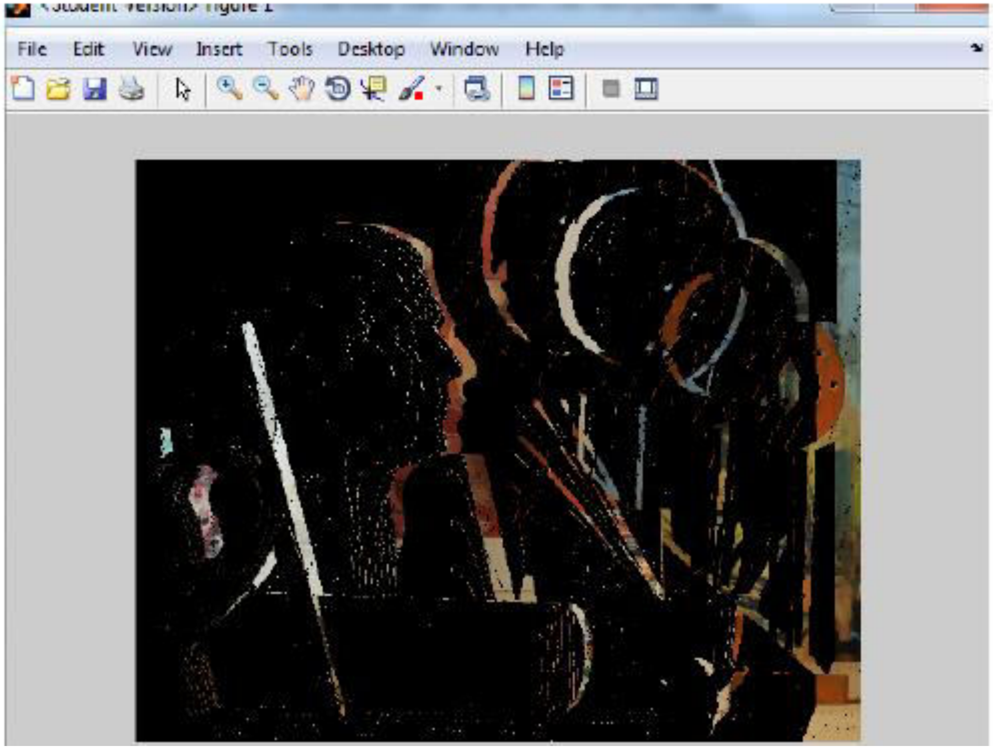
View Synthesis - Basics

Use one of the images and its disparity map to render a view at virtual camera location. By shifting pixels with half the disparity value



View Synthesis - Basics

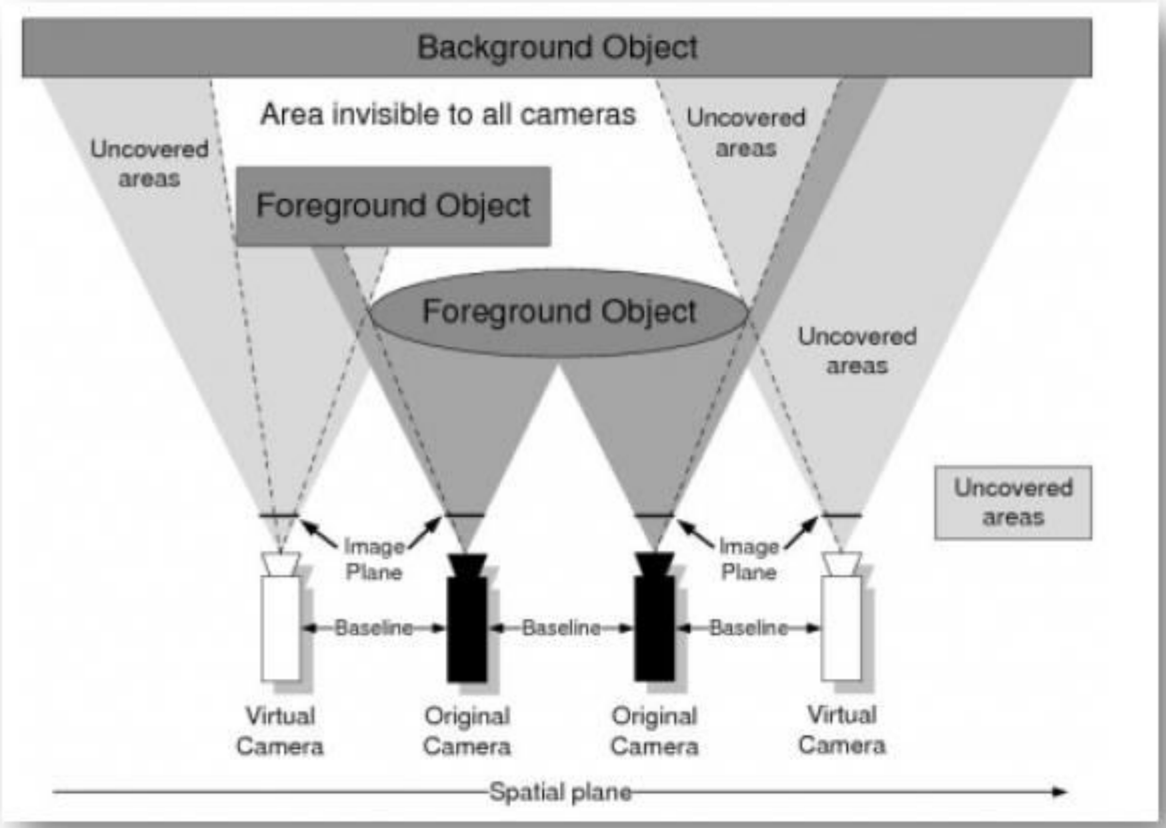
Use the information from other image to fill in the holes, by shifting the pixels by half the disparity



View Synthesis - Basics

Putting both together, we have the intermediary view. We still have holes. Why??

View Synthesis – Problem of Holes



View Synthesis – Problem of Color Variation at Boundaries