Stanford Vision & Learning Lab

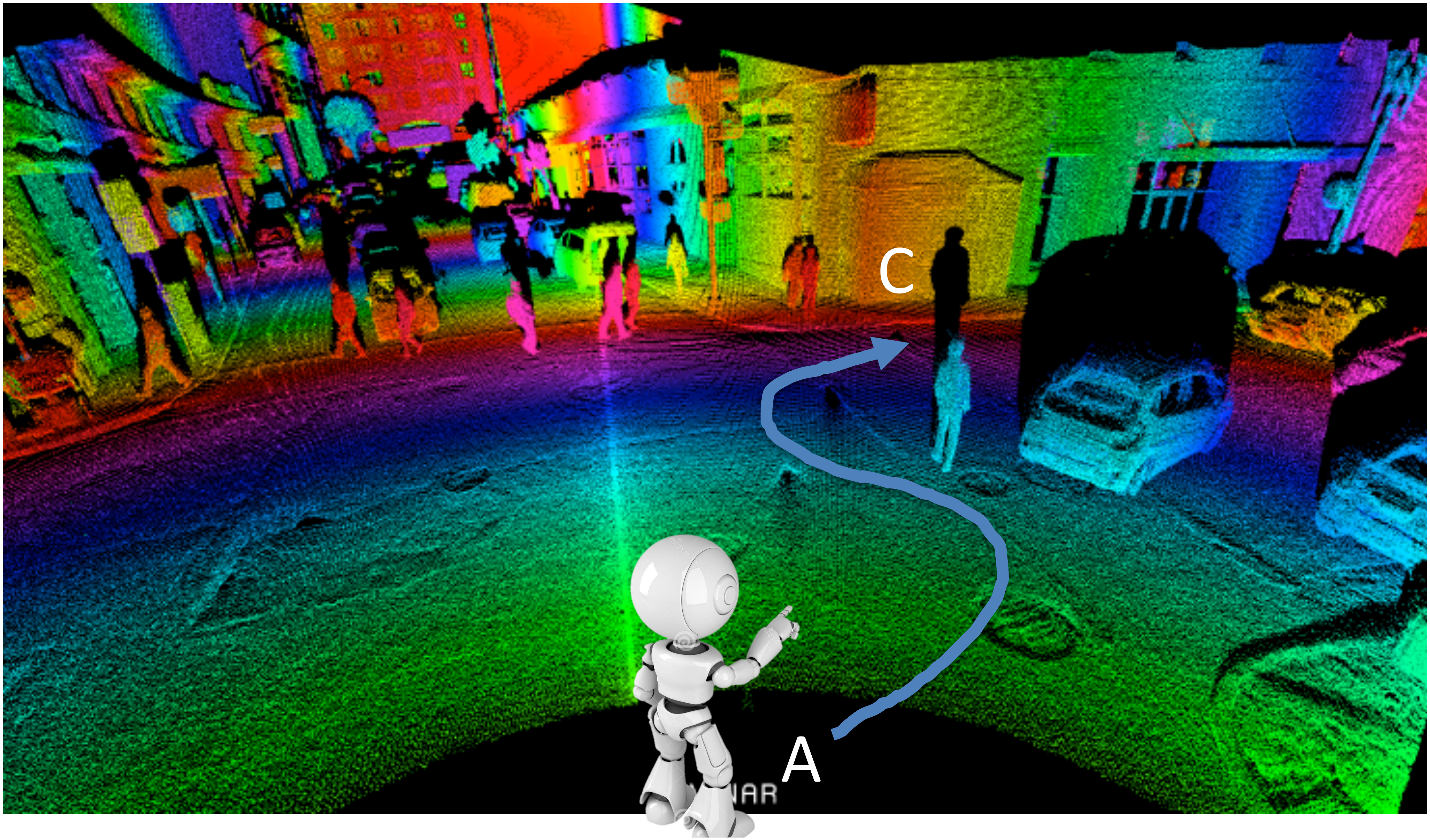
3D Scene Understanding

Silvio Savarese

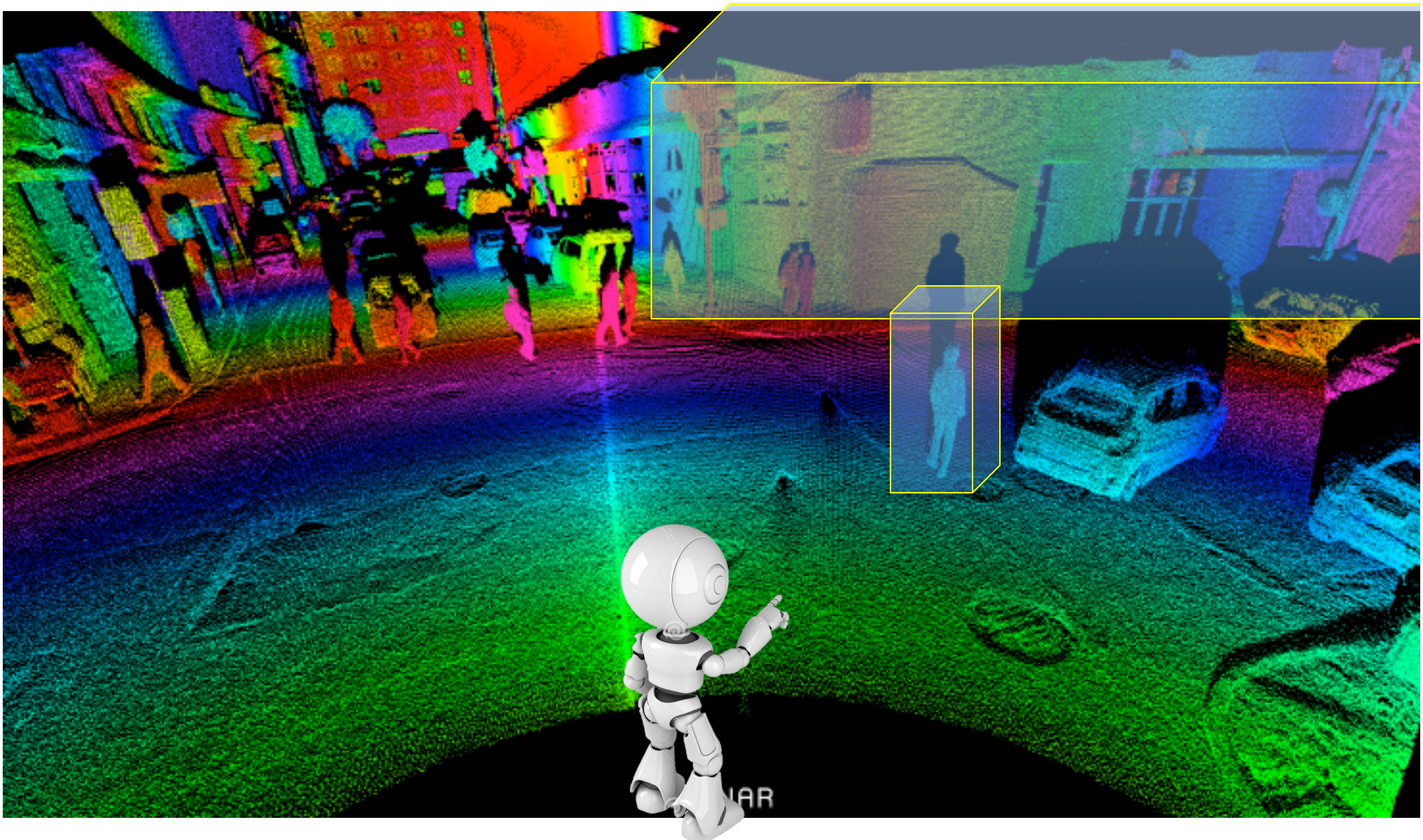
Computer Vision, CoS 429
Princeton University
November 21st 2017

3D scene understanding

Is this about where?



Is this sufficient?



Is this about what?

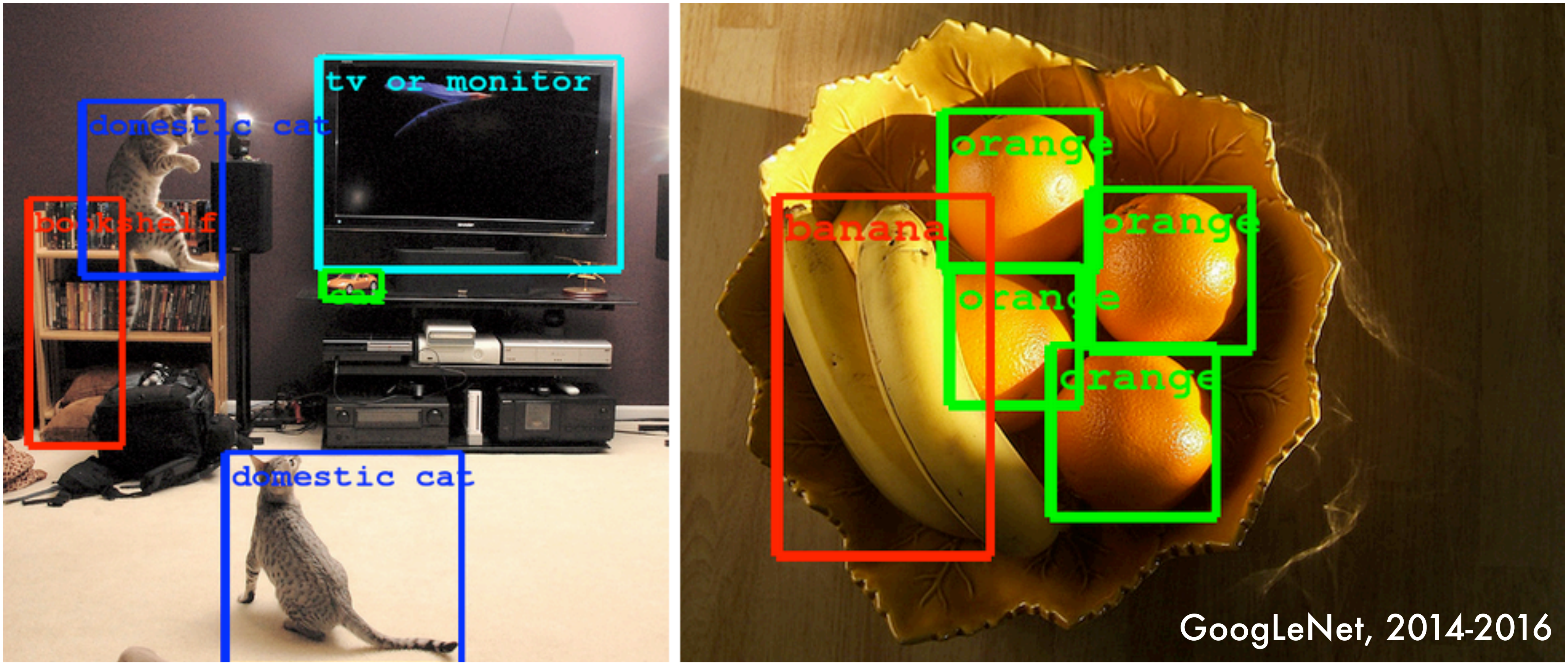
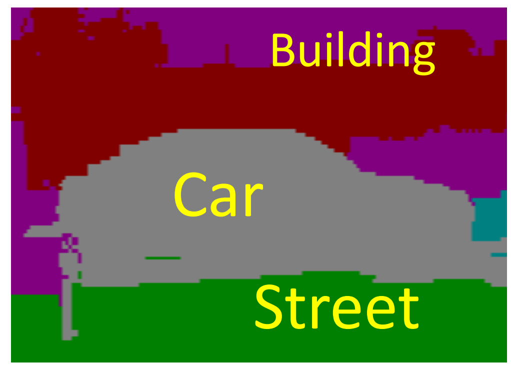
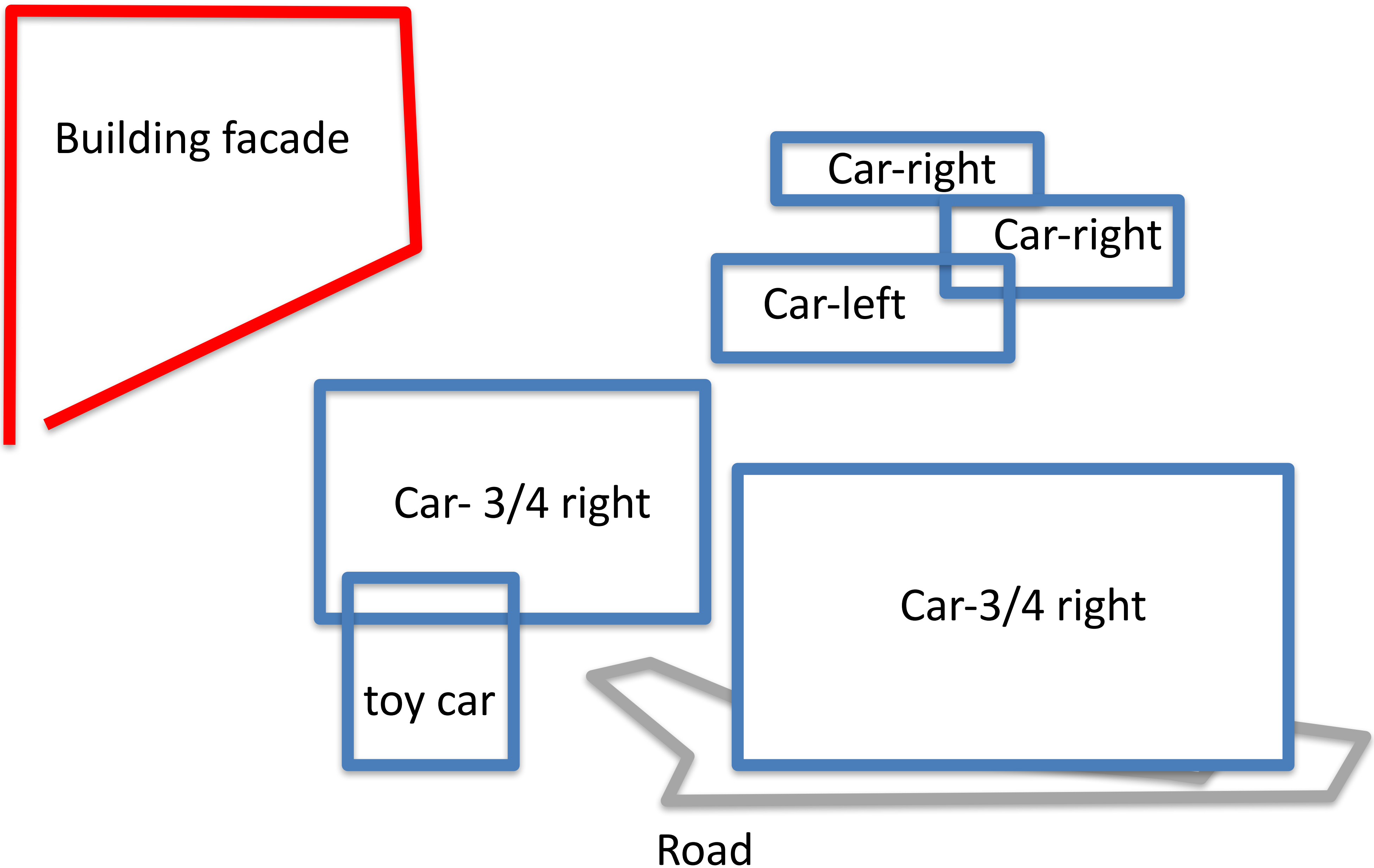


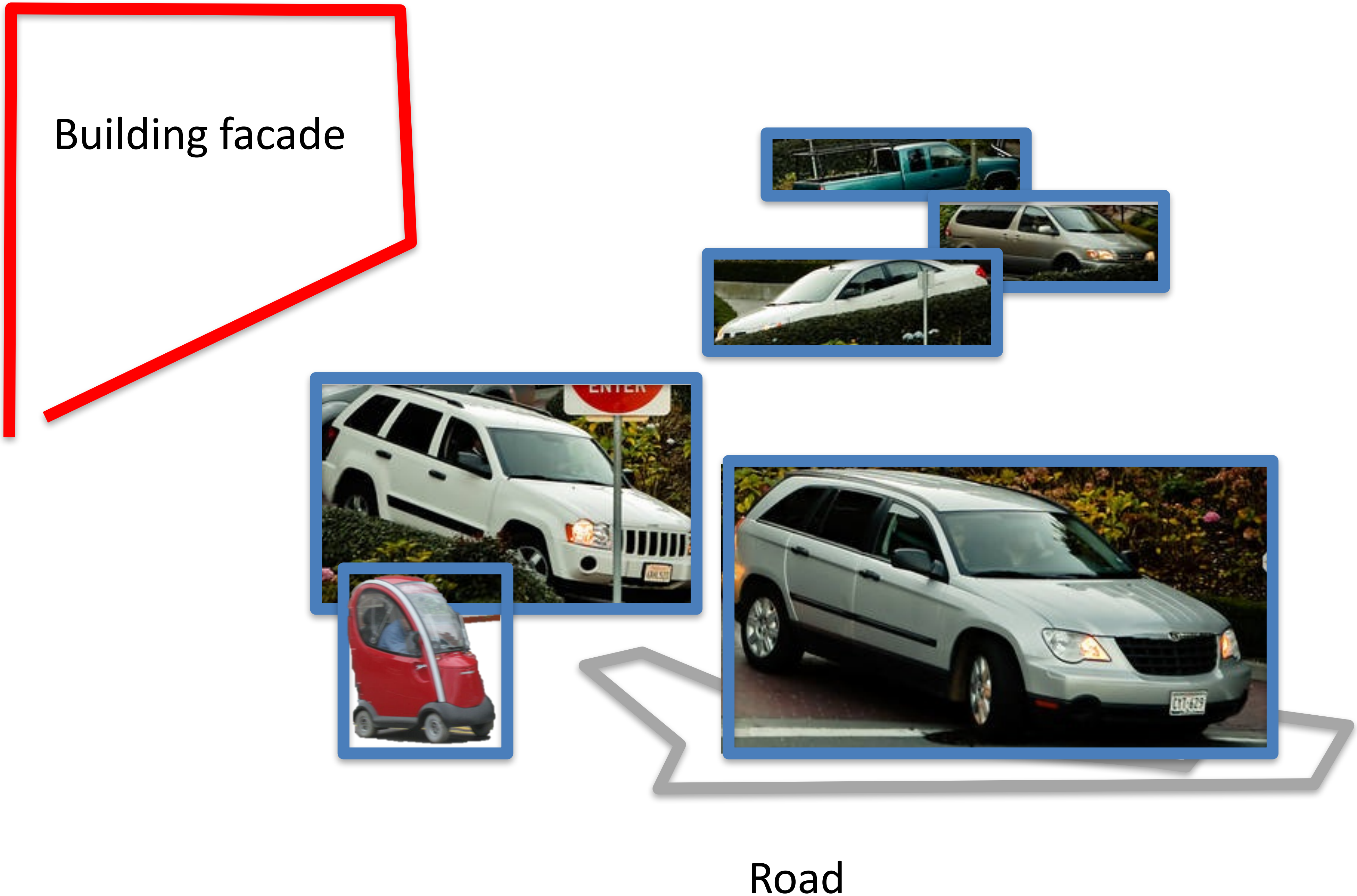
Image-to-labels paradigm

image

labels




Building facade
Car-right
Car-right
Car-left
Car- 3/4 right
Car-3/4 right
toy car
Road


Building facade
Road


Lombard Street, San Francisco (2)
DO NOT
ENTER
WorldOnaBike.com
(c) Harry Kikstra, WorldOnaBike.co

3D reconstruction from images

- The SFM problem
- Affine SFM
- Perspective SFM
- Bundle Adjustment

3D Scene Understanding

- Motivation
- Single view 3D scene understanding
- Multi-views 3D scene understanding

3D reconstruction from images

- The SFM problem
- Affine SFM
- Perspective SFM
- Bundle Adjustment

3D Scene Understanding

- Motivation
- Single view 3D scene understanding
- Multi-views 3D scene understanding

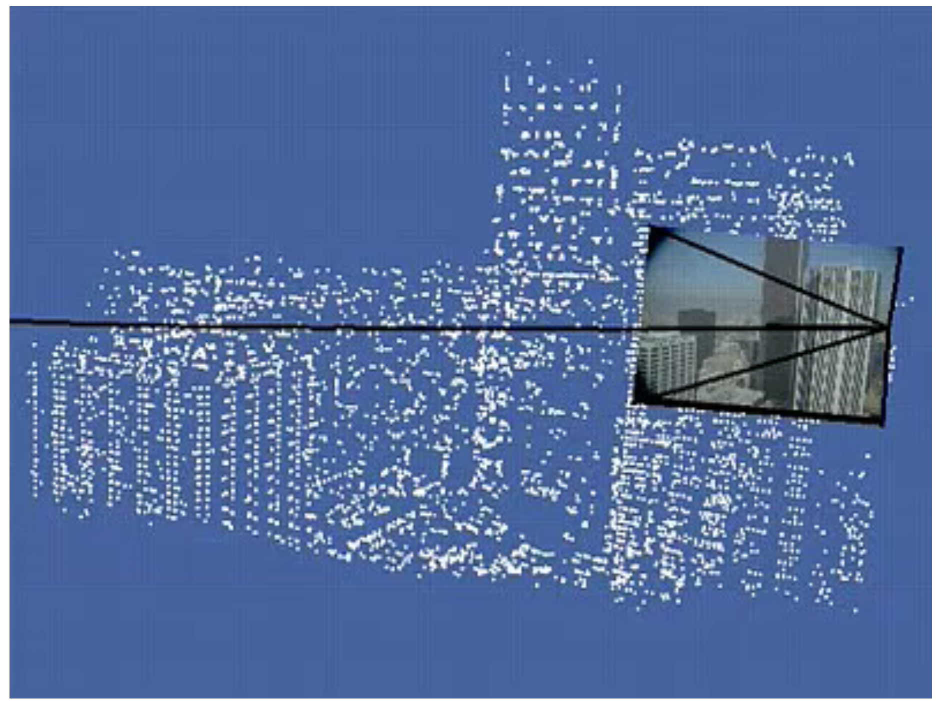
3D reconstruction from images

- The SFM problem
- Affine SFM
- Perspective SFM
- Bundle Adjustment

3D Scene Understanding

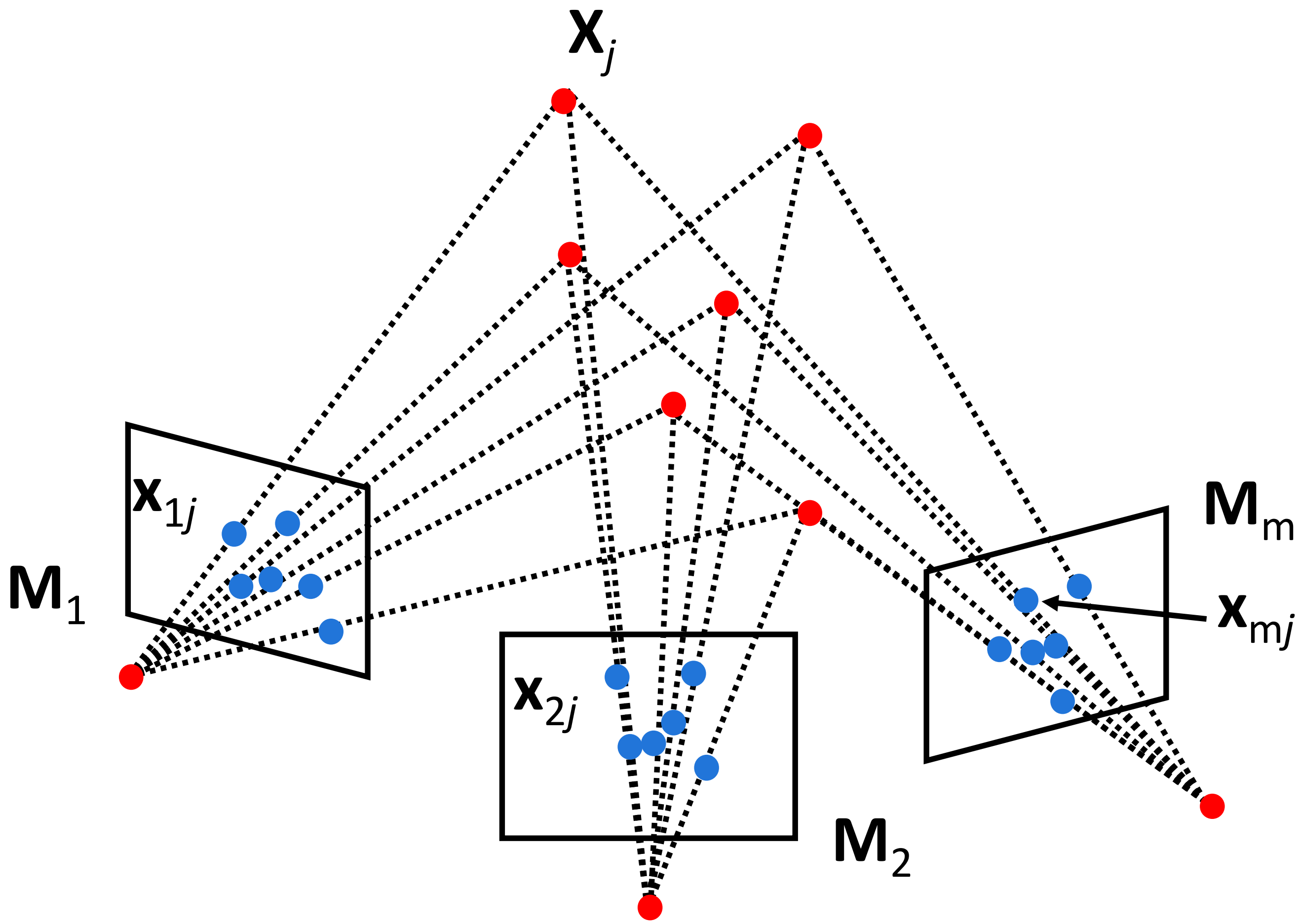
- Motivation
- Single view 3D scene understanding
- Multi-views 3D scene understanding

Structure from motion problem



Courtesy of Oxford **Visual Geometry Group**

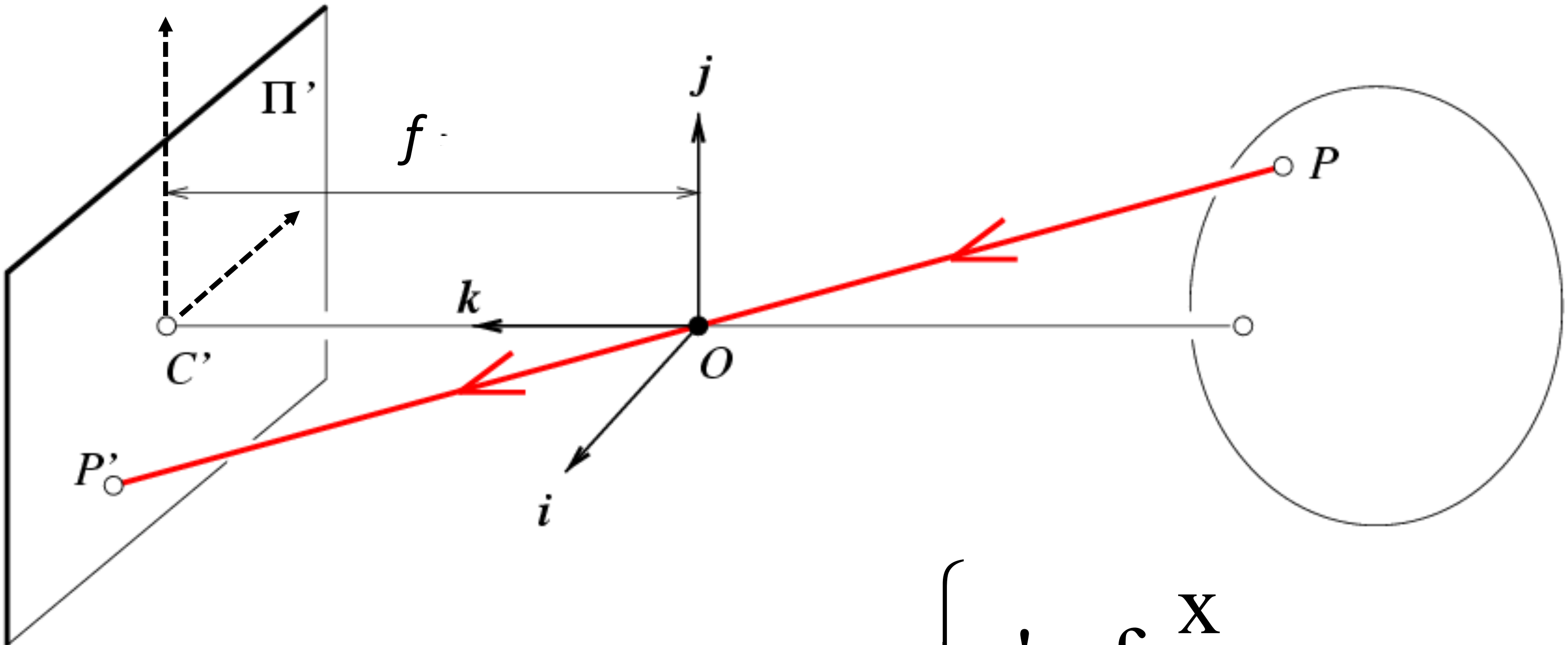
Structure from motion problem



Given *m* images of *n* fixed 3D points

- $\mathbf{x}_{ij} = \mathbf{M}_i \mathbf{X}_j, \quad i = 1, \dots, m, \quad j = 1, \dots, n$

Pinhole camera

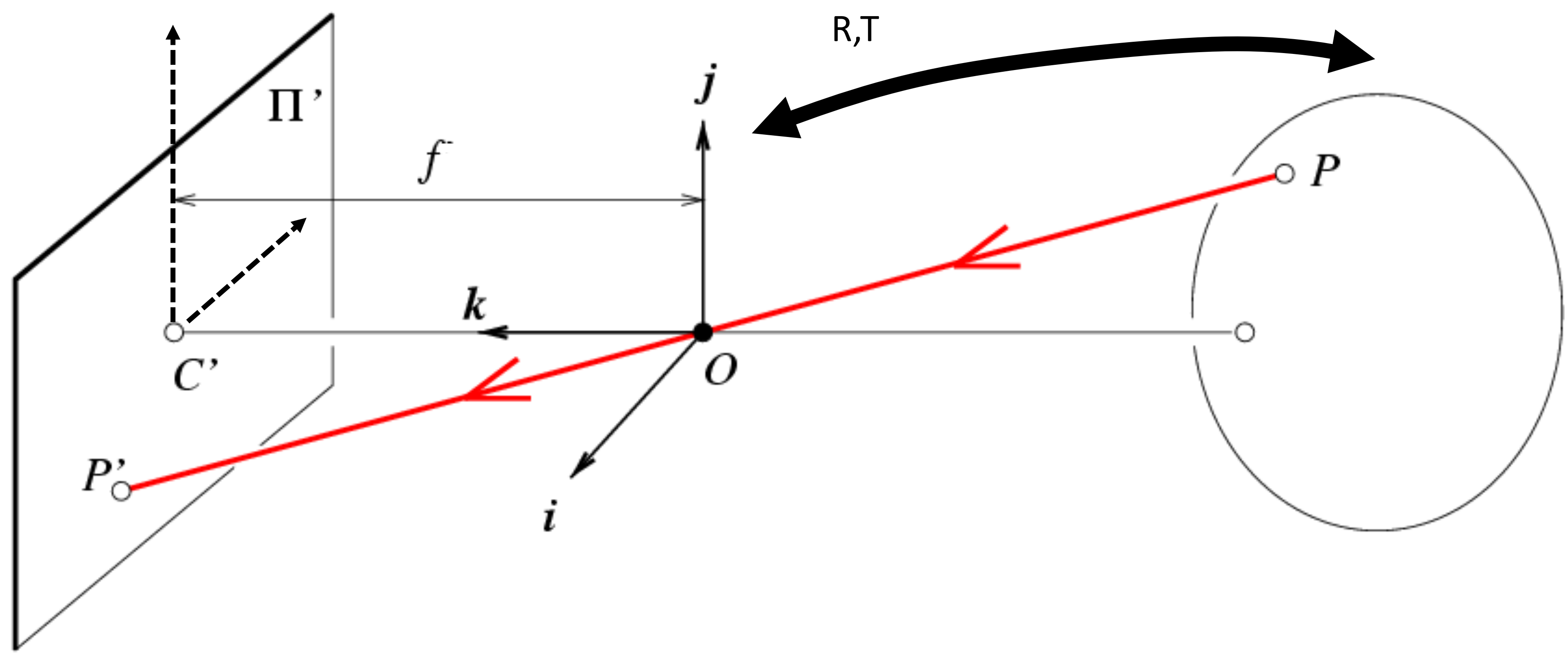


$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow P' = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

$$\begin{cases} x' = f\,\dfrac{x}{z} \\ y' = f\,\dfrac{y}{z} \end{cases}$$

Derived using similar triangles

Projective camera

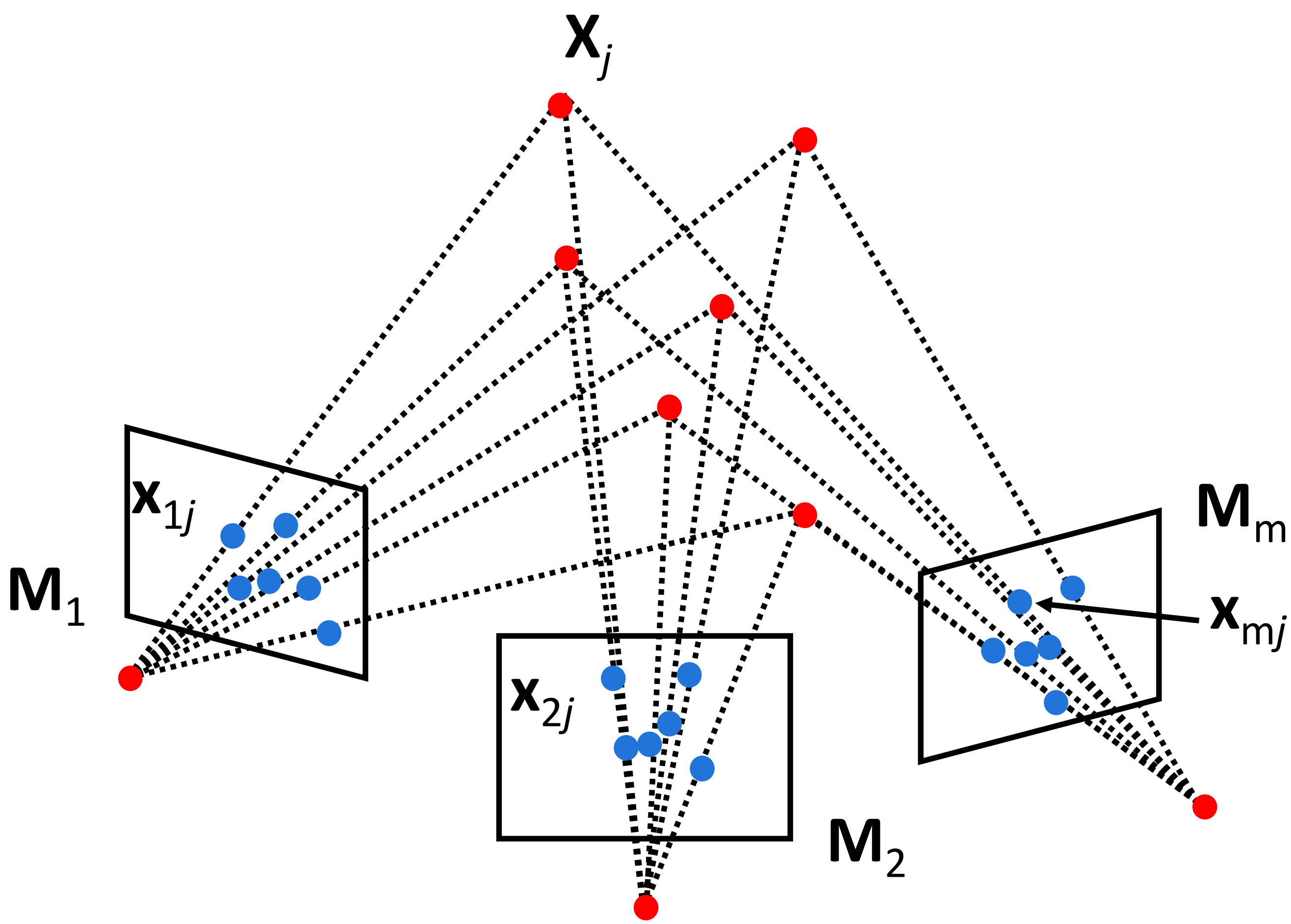


$$P'_{3\times1} = M\,P_w = K_{3\times3}\begin{bmatrix} R & T \end{bmatrix}_{3\times4} P_{w\,4\times1} \qquad M = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

Projective cameras

- Parallel lines are projected as converging lines!
- Distant objects look small!

Structure from motion problem



From the mxn observations $\mathbf{x}_{ij}$, estimate:

- m projection matrices $\mathbf{M}_i$ — motion
- n 3D points $\mathbf{X}_j$ — structure

3D reconstruction from images

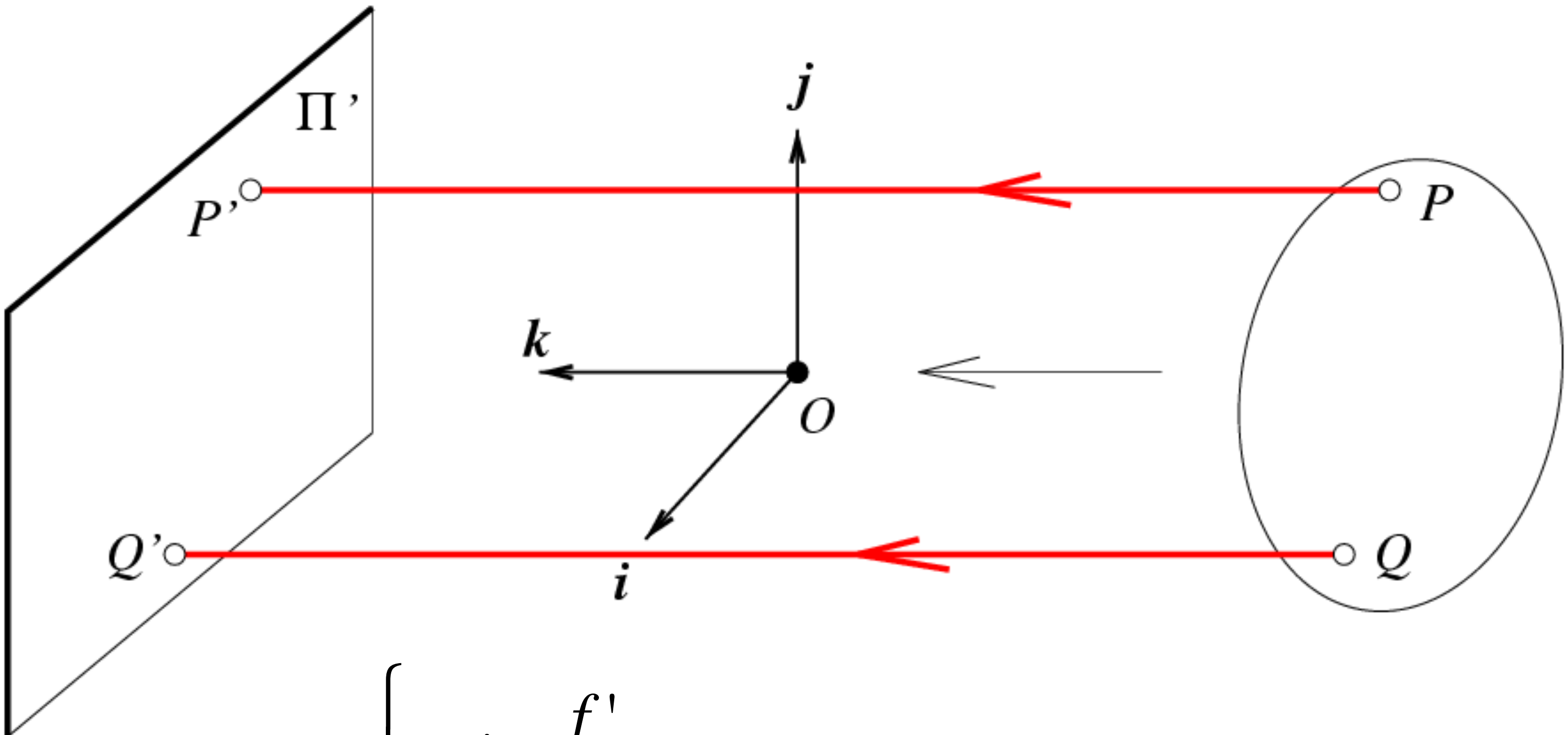
- The SFM problem
- Affine SFM
- Perspective SFM
- Bundle Adjustment

3D Scene Understanding

- Motivation
- Single view 3D scene understanding
- Multi-views 3D scene understanding

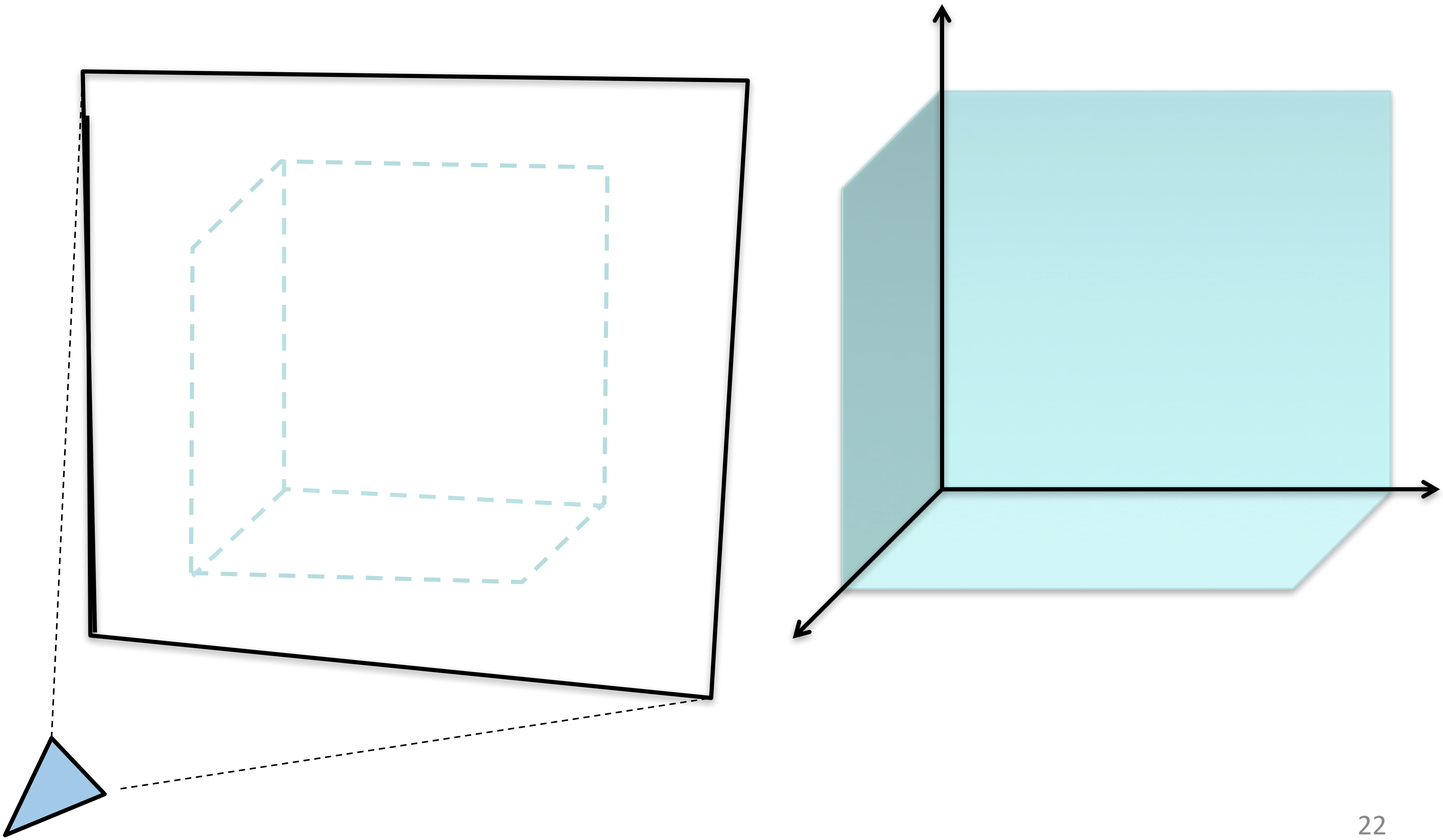
Orthographic (affine) projection

Distance from center of projection to image plane is infinite

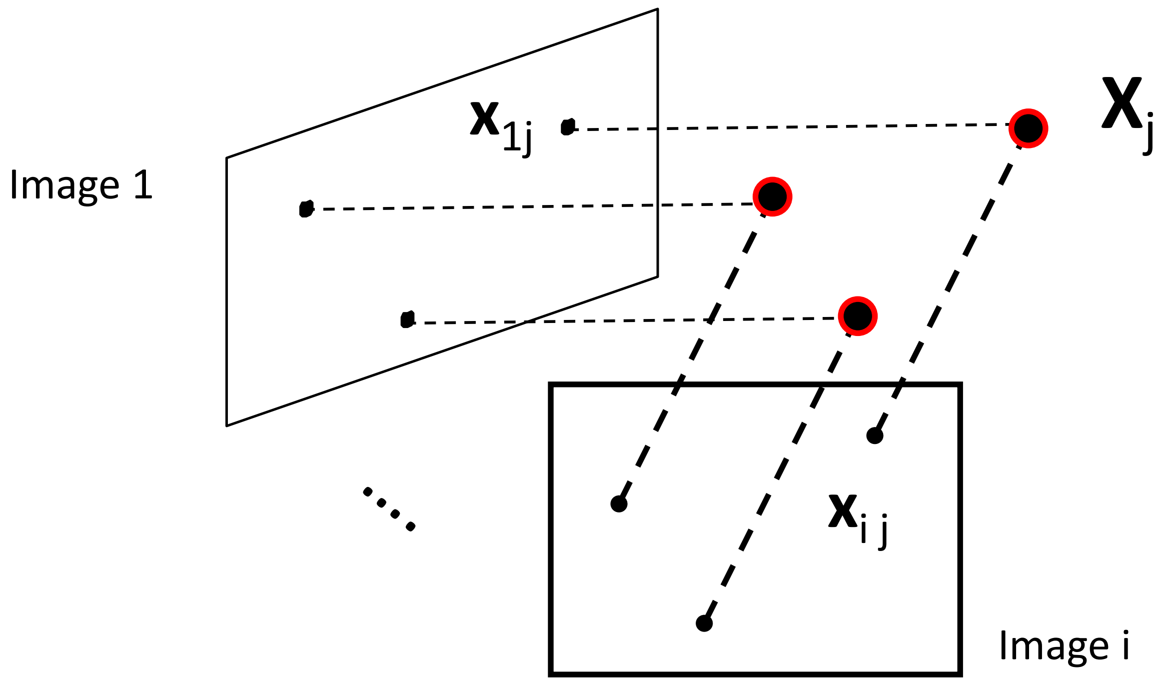


$$\begin{cases} x' = \dfrac{f'}{z} x \\ \\ y' = \dfrac{f'}{z} y \end{cases} \quad \rightarrow \quad \begin{cases} x' = x \\ y' = y \end{cases}$$

Projection of a cube with affine cameras



Affine cameras



For the affine case (in Euclidean space)

$$\underset{2\times1}{\mathbf{x}_{ij}} = \underset{2\times3}{\mathbf{A}_i}\,\underset{3\times1}{\mathbf{X}_j} + \underset{2\times1}{\mathbf{b}_i} \qquad \text{[Eq. 4]}$$

The Affine Structure-from-Motion Problem

Given *m* images of *n* fixed points $\mathbf{X}_j$ we can write

$$\mathbf{x}_{ij} = \mathbf{A}_i \mathbf{X}_j + \mathbf{b}_i \qquad \text{for } i = 1, \ldots, \boxed{m} \text{ and } j = 1, \ldots, \boxed{n}$$

N. of cameras N. of points

Problem: estimate m matrices A_i, m matrices b_i and the n positions $\mathbf{X}_j$ from the m×n observations $\mathbf{x}_{ij}$.

How many equations and how many unknown?

$2m \times n$ equations in $8m + 3n - 9$ unknowns

A factorization method –
Tomasi & Kanade algorithm

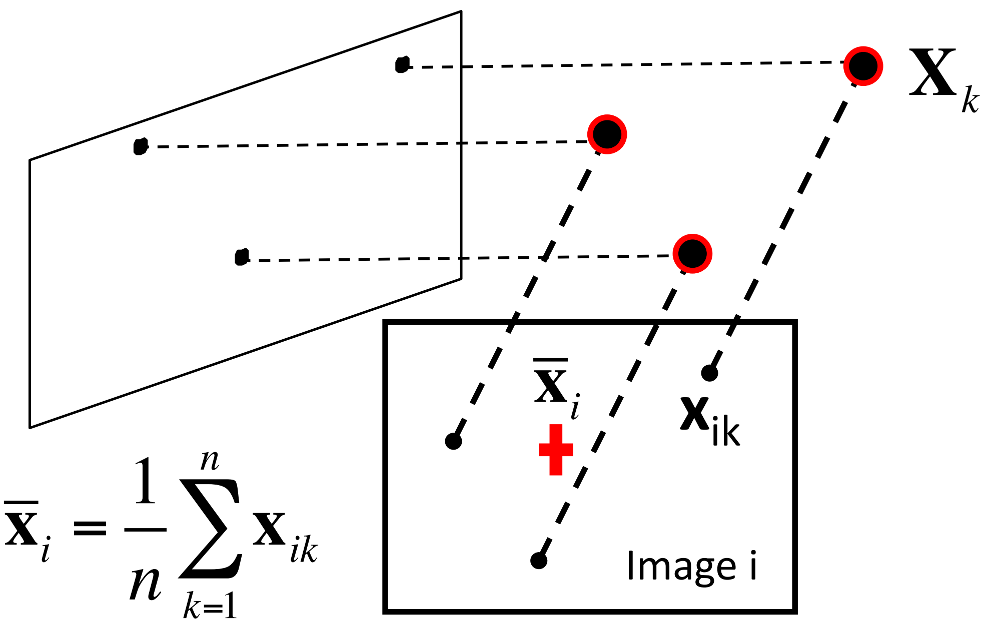
C. Tomasi and T. KanadeShape and motion from image streams under orthography: A factorization method. *IJCV*, 9(2):137-154, November 1992.

- Data centering
- Factorization

A factorization method - Centering the data

Centering: subtract the centroid of the image points

[Eq. 6] $$\hat{\mathbf{x}}_{ij} = \mathbf{x}_{ij} - \underbrace{\boxed{\frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_{ik}}}_{\bar{\mathbf{x}}_i}$$



[Eq. 5] $$\bar{\mathbf{x}}_i = \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_{ik}$$

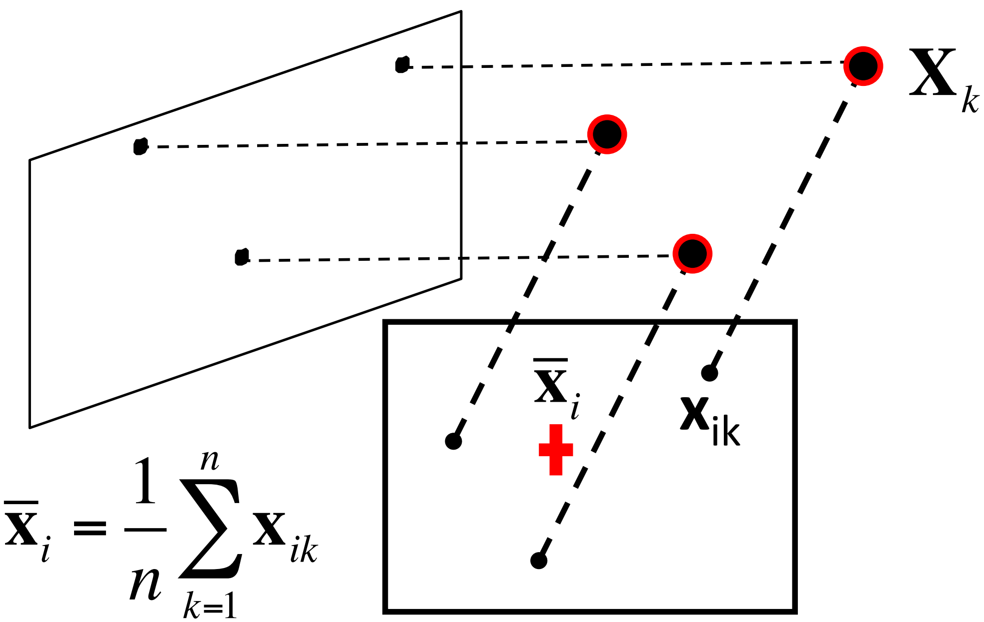
A factorization method - Centering the data

Centering: subtract the centroid of the image points

[Eq. 6] $$\hat{\mathbf{x}}_{ij} = \mathbf{x}_{ij} - \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_{ik} = \mathbf{A}_i\mathbf{X}_j + \mathbf{b}_i - \frac{1}{n}\sum_{k=1}^{n}\mathbf{A}_i\mathbf{X}_k - \frac{1}{n}\sum_{k=1}^{n}\mathbf{b}_i$$

$$\mathbf{x}_{ik} = \mathbf{A}_i\mathbf{X}_k + \mathbf{b}_i$$

[Eq. 4]



[Eq. 5] $$\bar{\mathbf{x}}_i = \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_{ik}$$

A factorization method - Centering the data

Centering: subtract the centroid of the image points

[Eq. 6]

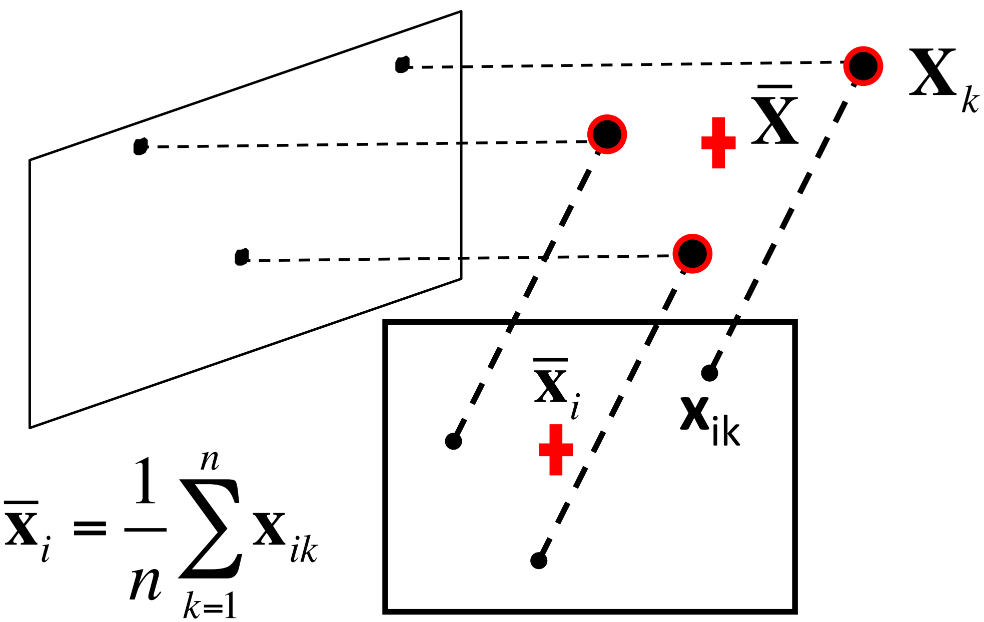
$$\hat{\mathbf{x}}_{ij} = \mathbf{x}_{ij} - \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_{ik} = \mathbf{A}_i\mathbf{X}_j + \mathbf{b}_i - \frac{1}{n}\sum_{k=1}^{n}\mathbf{A}_i\mathbf{X}_k - \frac{1}{n}\sum_{k=1}^{n}\mathbf{b}_i$$

$$= \mathbf{A}_i\left(\mathbf{X}_j - \frac{1}{n}\sum_{k=1}^{n}\mathbf{X}_k\right) = \mathbf{A}_i\left(\mathbf{X}_j - \bar{\mathbf{X}}\right)$$

$$= \mathbf{A}_i\hat{\mathbf{X}}_j$$ [Eq. 8]

$$\mathbf{x}_{ik} = \mathbf{A}_i\mathbf{X}_k + \mathbf{b}_i$$

[Eq. 4]



$$\bar{\mathbf{x}}_i = \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_{ik}$$

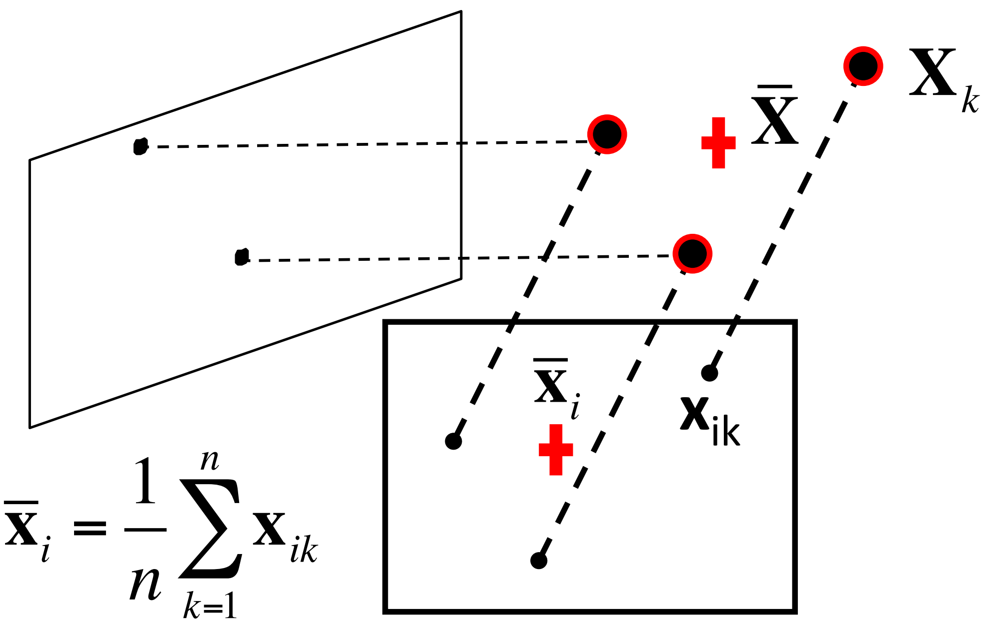
$$\bar{\mathbf{X}} = \frac{1}{n}\sum_{k=1}^{n}\mathbf{X}_k$$ [Eq. 7]

Centroid of 3D points

A factorization method - Centering the data

Thus, after centering, each **normalize**d observed point is related to the 3D point by

$$\hat{\mathbf{x}}_{ij} = \mathbf{A}_i \hat{\mathbf{X}}_j \quad \text{[Eq. 8]}$$



$$\bar{\mathbf{x}}_i = \frac{1}{n}\sum_{k=1}^{n} \mathbf{x}_{ik}$$

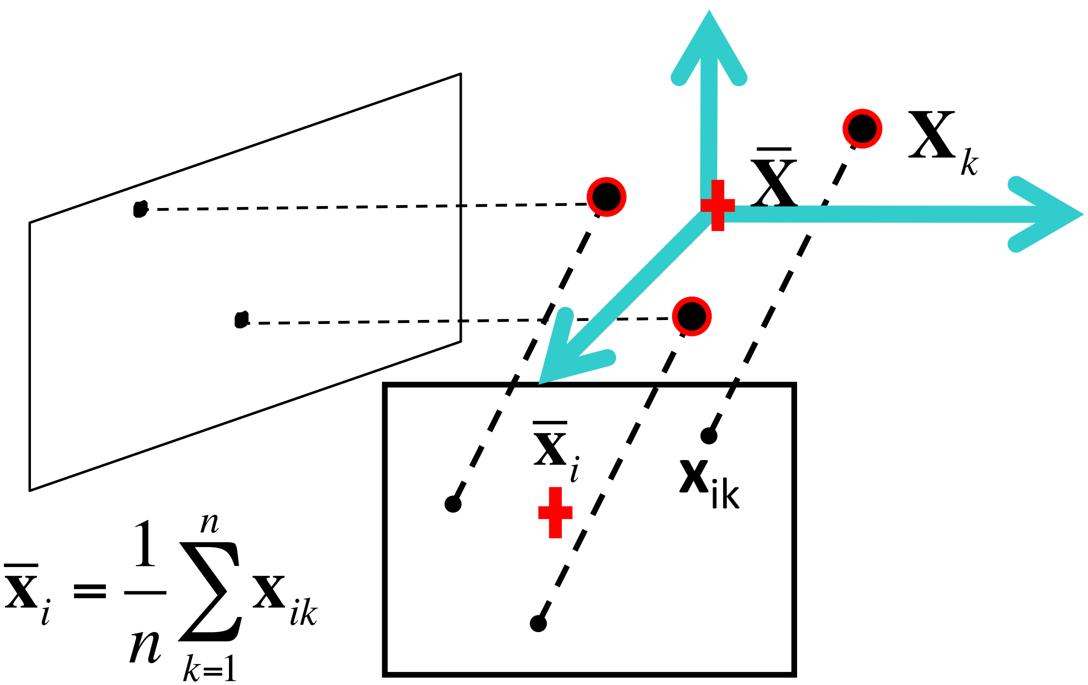
$$\bar{\mathbf{X}} = \frac{1}{n}\sum_{k=1}^{n} \mathbf{X}_k \quad \text{[Eq. 7]}$$

Centroid of 3D points

A factorization method - Centering the data

If the centroid of points in 3D = center of the world reference system

$$\hat{\mathbf{x}}_{ij} = \mathbf{A}_i \hat{\mathbf{X}}_j = \mathbf{A}_i \mathbf{X}_j$$ [Eq. 9]



$$\bar{\mathbf{x}}_i = \frac{1}{n}\sum_{k=1}^{n} \mathbf{x}_{ik}$$

$$\bar{\mathbf{X}} = \frac{1}{n}\sum_{k=1}^{n} \mathbf{X}_k$$ [Eq. 7]

Centroid of 3D points

A factorization method - factorization

Let's create a $2m \times n$ data (measurement) matrix:

$$\mathbf{D} = \begin{bmatrix} \hat{\mathbf{x}}_{11} & \hat{\mathbf{x}}_{12} & \cdots & \hat{\mathbf{x}}_{1n} \\ \hat{\mathbf{x}}_{21} & \hat{\mathbf{x}}_{22} & \cdots & \hat{\mathbf{x}}_{2n} \\ & & \ddots & \\ \hat{\mathbf{x}}_{m1} & \hat{\mathbf{x}}_{m2} & \cdots & \hat{\mathbf{x}}_{mn} \end{bmatrix}$$

cameras ($2m$)

points (n)

Each $\hat{\mathbf{x}}_{ij}$ entry is a 2x1 vector!

A factorization method - factorization

Let's create a 2m × n data (measurement) matrix:

$$\mathbf{D} = \begin{bmatrix} \hat{\mathbf{x}}_{11} & \hat{\mathbf{x}}_{12} & \cdots & \hat{\mathbf{x}}_{1n} \\ \hat{\mathbf{x}}_{21} & \hat{\mathbf{x}}_{22} & \cdots & \hat{\mathbf{x}}_{2n} \\ & & \ddots & \\ \hat{\mathbf{x}}_{m1} & \hat{\mathbf{x}}_{m2} & \cdots & \hat{\mathbf{x}}_{mn} \end{bmatrix} = \boxed{\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_m \end{bmatrix}} \boxed{\begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \cdots & \mathbf{X}_n \end{bmatrix}}$$

(2*m* × n)

cameras (2*m* × 3)

M

points (3 × *n*)

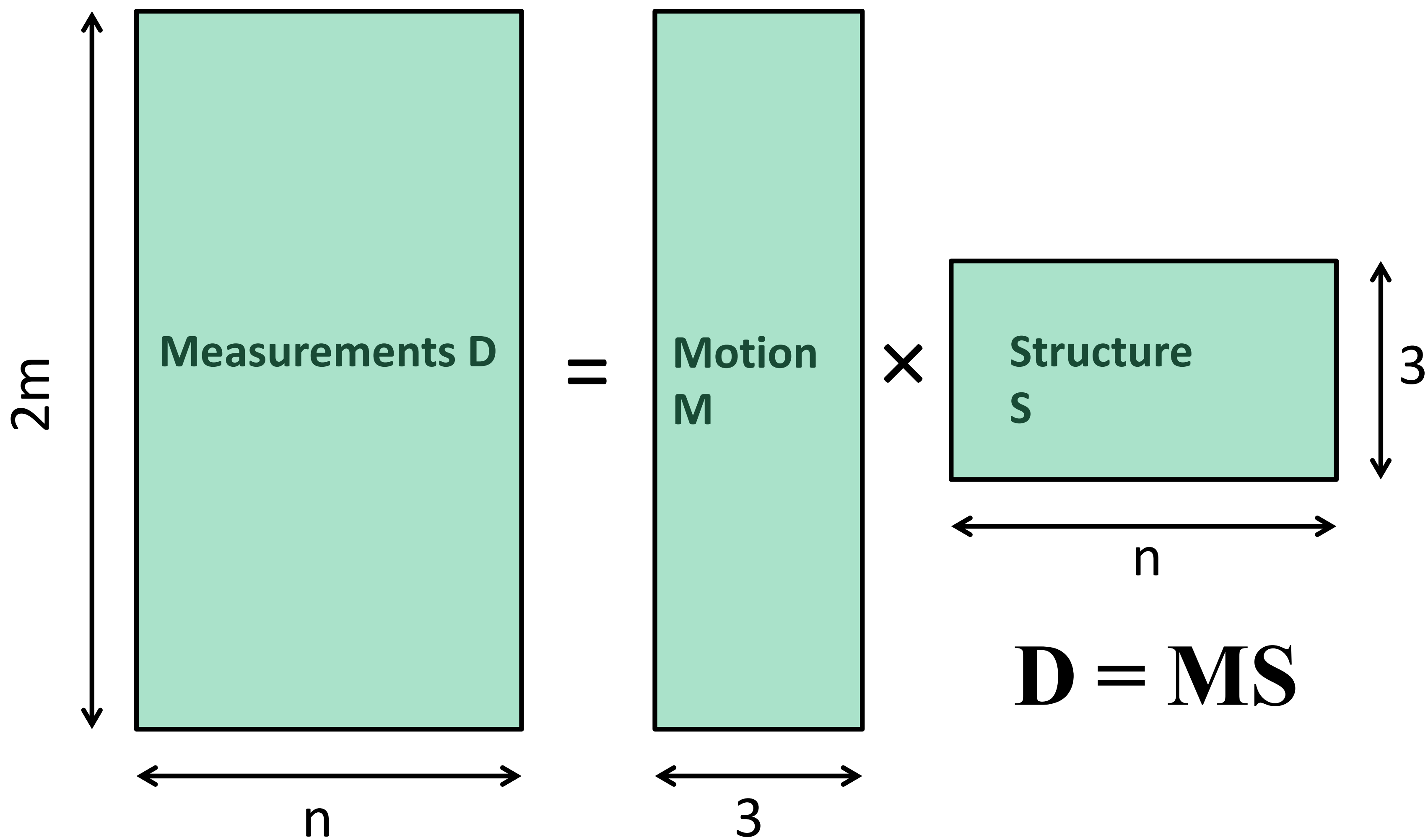
S

[Eq. 10]

Each $\hat{\mathbf{x}}_{ij}$ entry is a 2x1 vector!
A$_i$ is 2x3 and **X**$_j$ is 3x1

The measurement matrix **D = M S** has rank 3
(it's a product of a 2mx3 matrix and 3xn matrix)

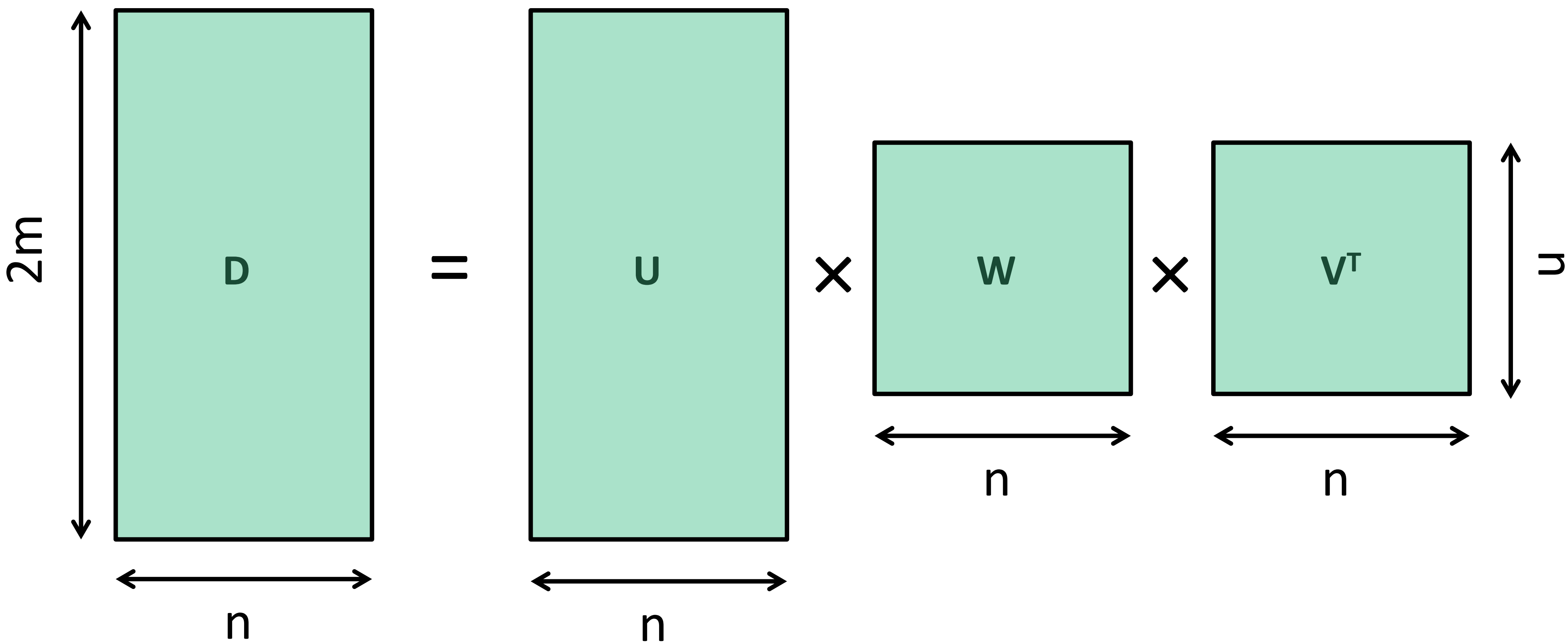
Factorizing the Measurement Matrix



$$\mathbf{D} = \mathbf{MS}$$

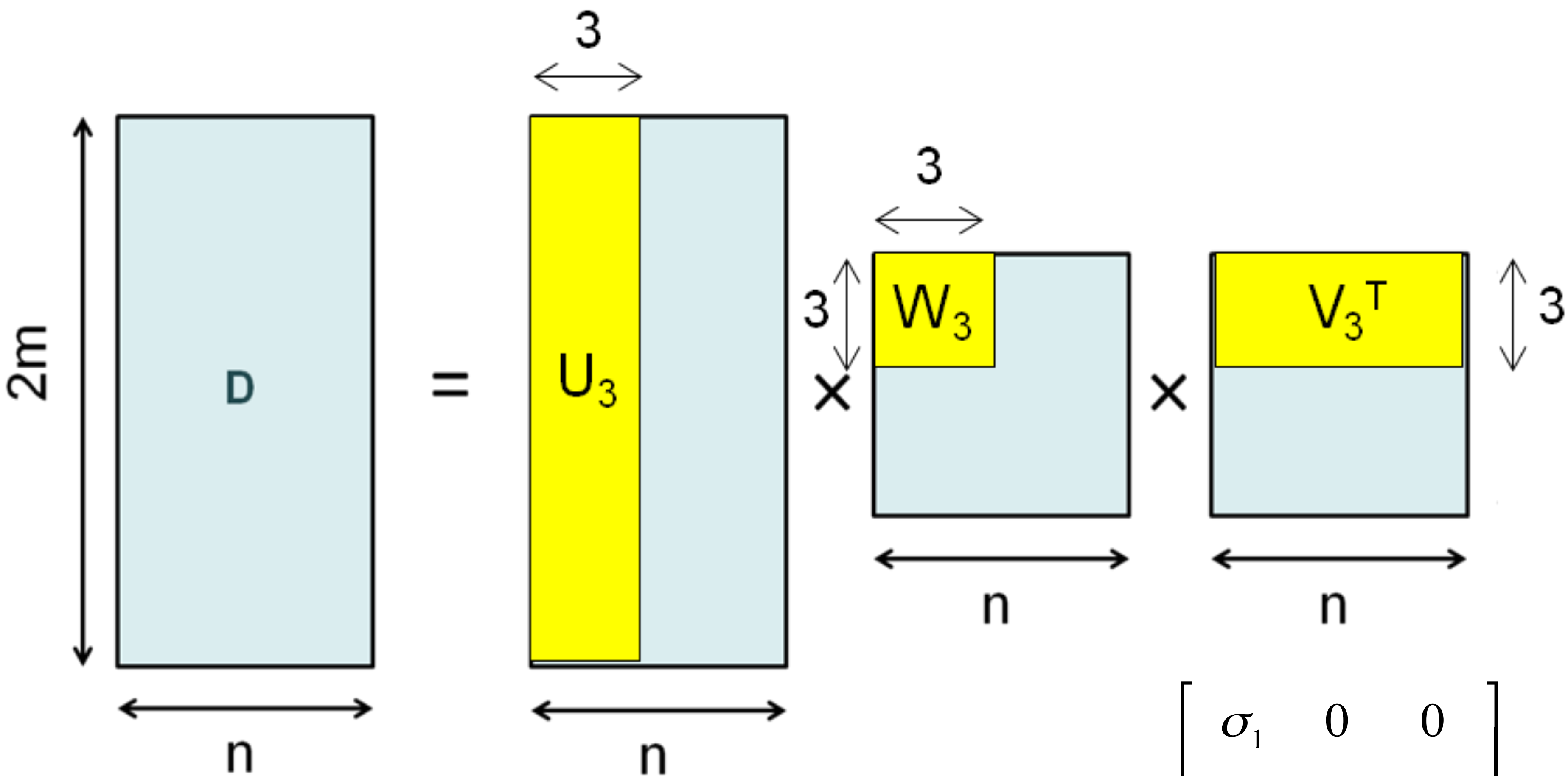
Factorizing the Measurement Matrix

- How to factorize D? By computing the Singular value decomposition of D!



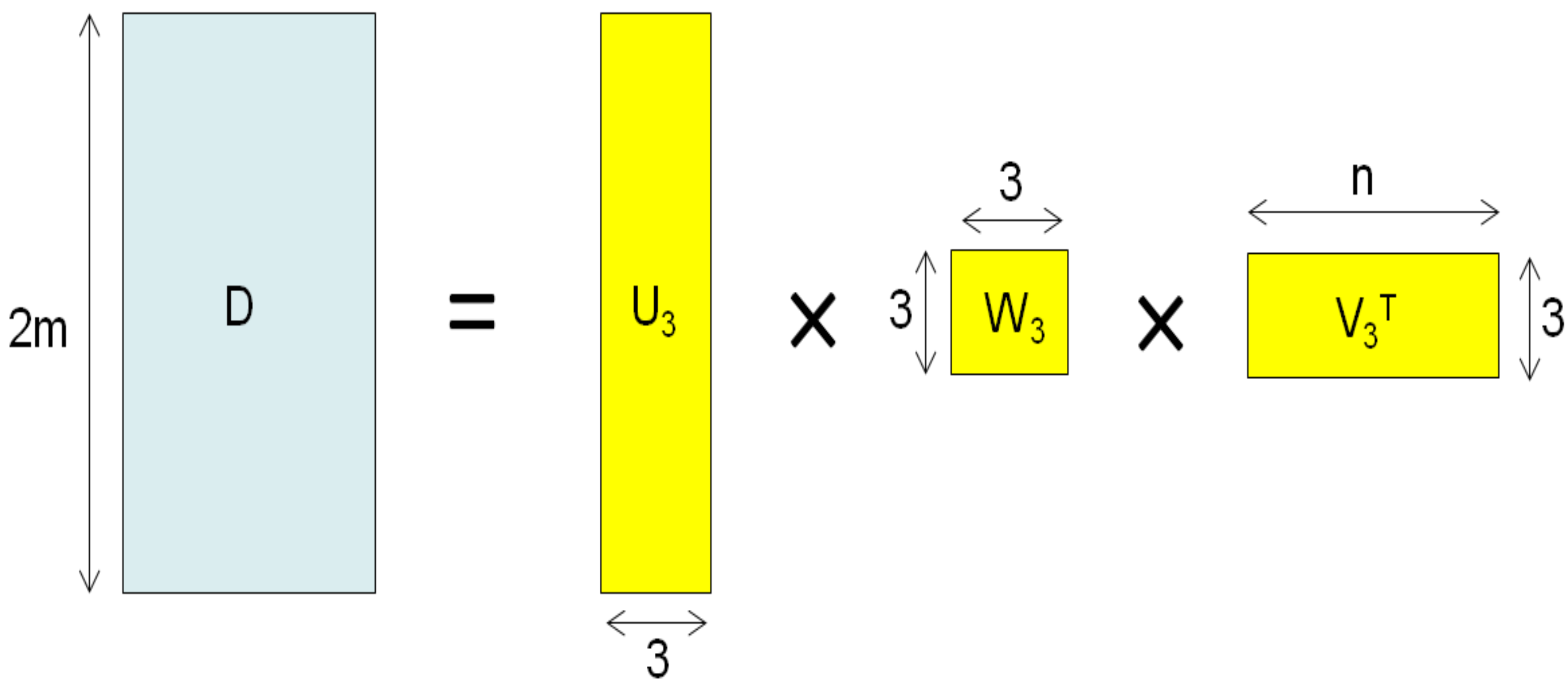
Factorizing the Measurement Matrix

Since rank (D)=3, there are only 3 non-zero singular values σ_1 , σ_2 and σ_3

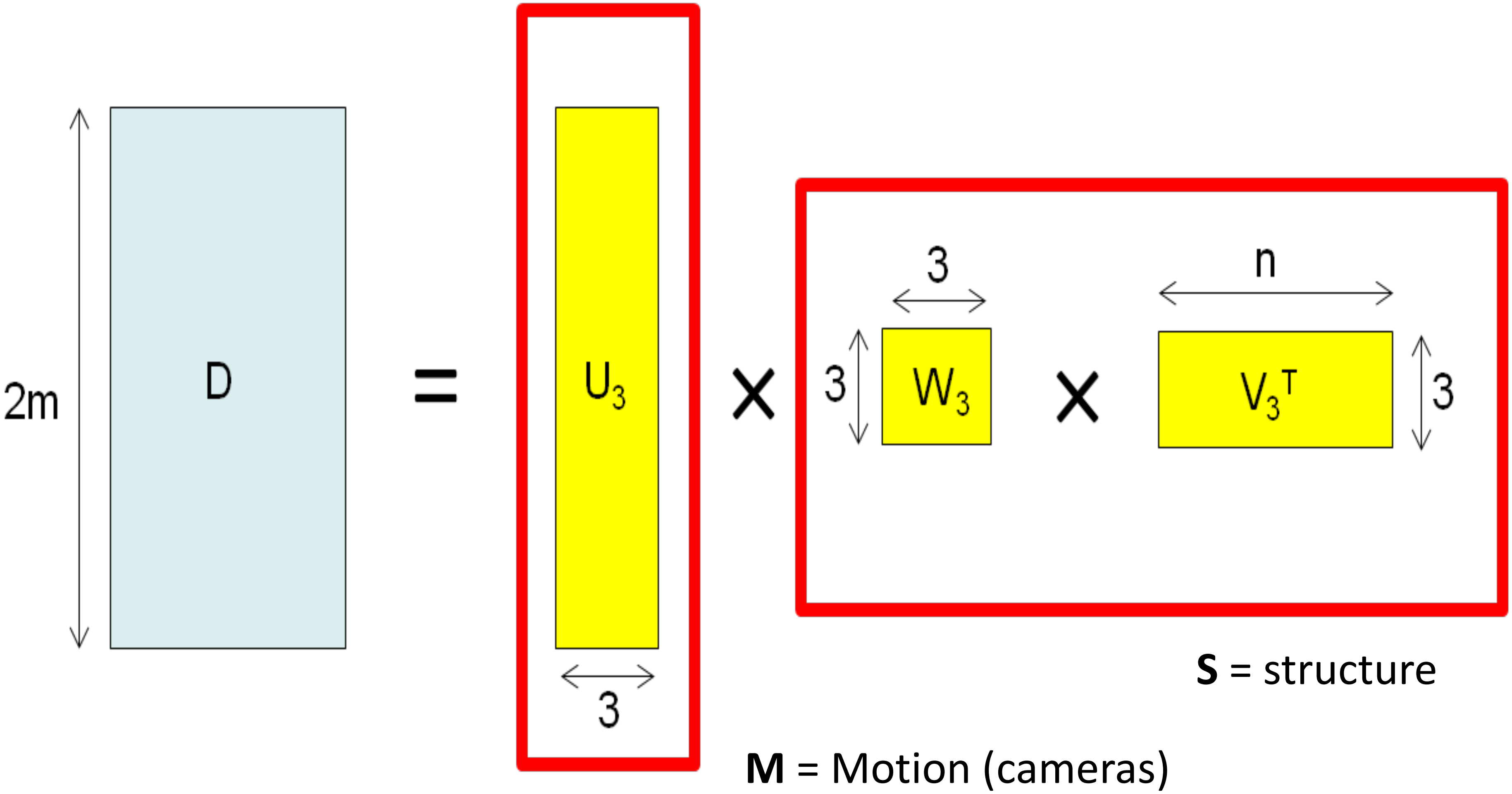


Where $W_3 = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix}$ [Eq. 11]

Factorizing the Measurement Matrix



Factorizing the Measurement Matrix



$$\mathbf{D} = \mathbf{U}_3\, \mathbf{W}_3\, \mathbf{V}_3^T = \mathbf{U}_3\, (\mathbf{W}_3\, \mathbf{V}_3^T) = \mathbf{M}\, \mathbf{S} \qquad \text{[Eq. 12]}$$

Factorizing the Measurement Matrix

$$\mathbf{D} = \mathbf{U}_3\ \mathbf{W}_3\ \mathbf{V}_3{}^{\mathrm{T}} = \mathbf{U}_3\ (\mathbf{W}_3\ \mathbf{V}_3{}^{\mathrm{T}}) = \mathbf{M}\ \mathbf{S}$$ [Eq. 12]

What is the issue here? **D** has rank>3 because of:

- measurement noise
- affine approximation

Theorem: When $\mathbf{D}$ has a rank greater than 3, $\mathbf{U}_3\mathbf{W}_3\mathbf{V}_3^T$ is the best possible rank- 3 approximation of **D** in the sense of the Frobenius norm.

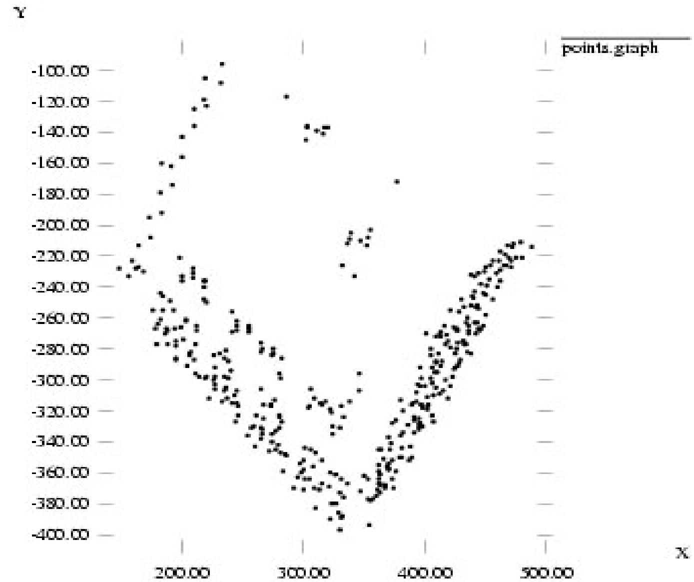
$$\mathbf{D} = \mathbf{U}_3\mathbf{W}_3\mathbf{V}_3^T \qquad \begin{cases} \mathbf{M} \approx \mathbf{U}_3 \\ \mathbf{S} \approx \mathbf{W}_3\mathbf{V}_3^T \end{cases}$$

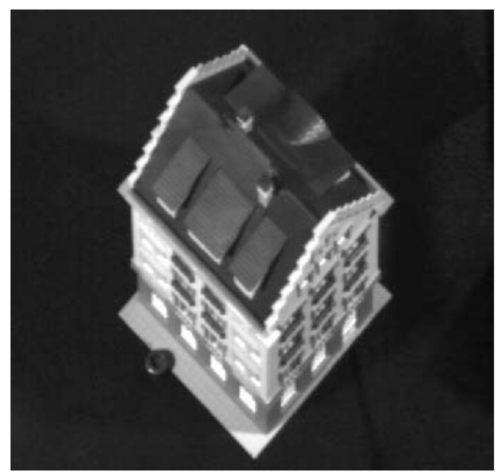
$$\boxed{\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} = \sqrt{\sum_{i=1}^{\min\{m,n\}}\sigma_i^2}}$$

Reconstruction results

1

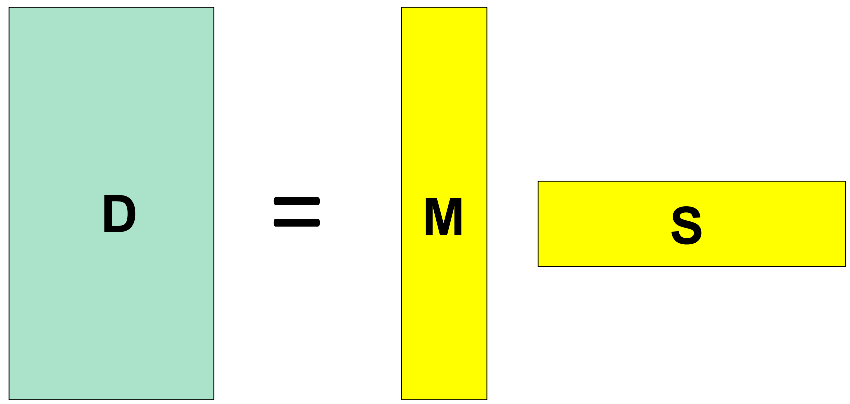
120



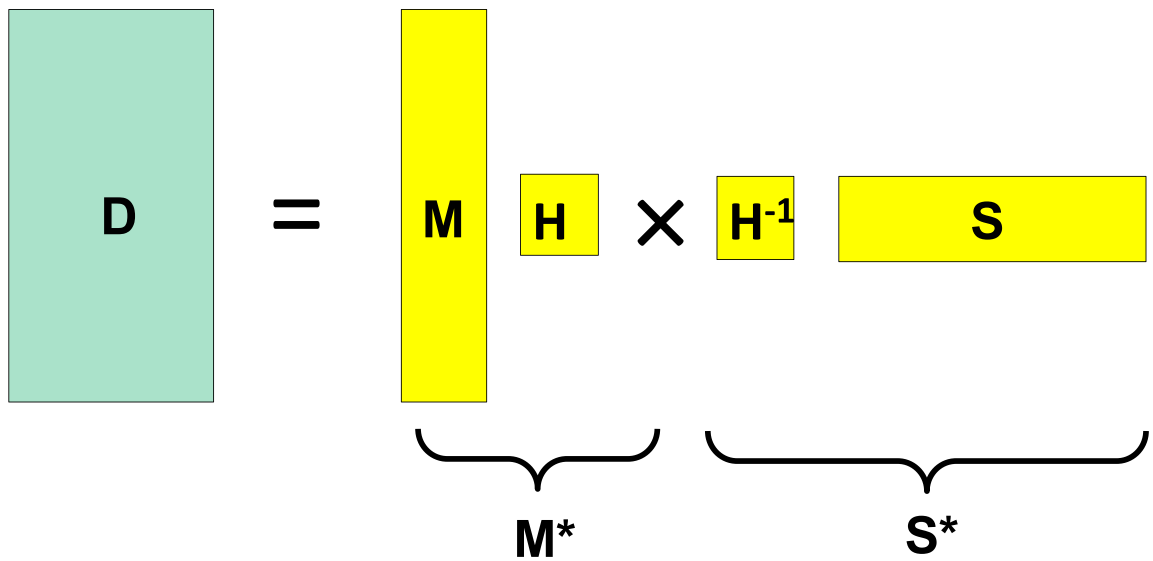


C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *IJCV*, 9(2):137-154, November 1992.

Affine Ambiguity



Affine Ambiguity



- The decomposition is not unique. We get the same **D** by applying the transformations:

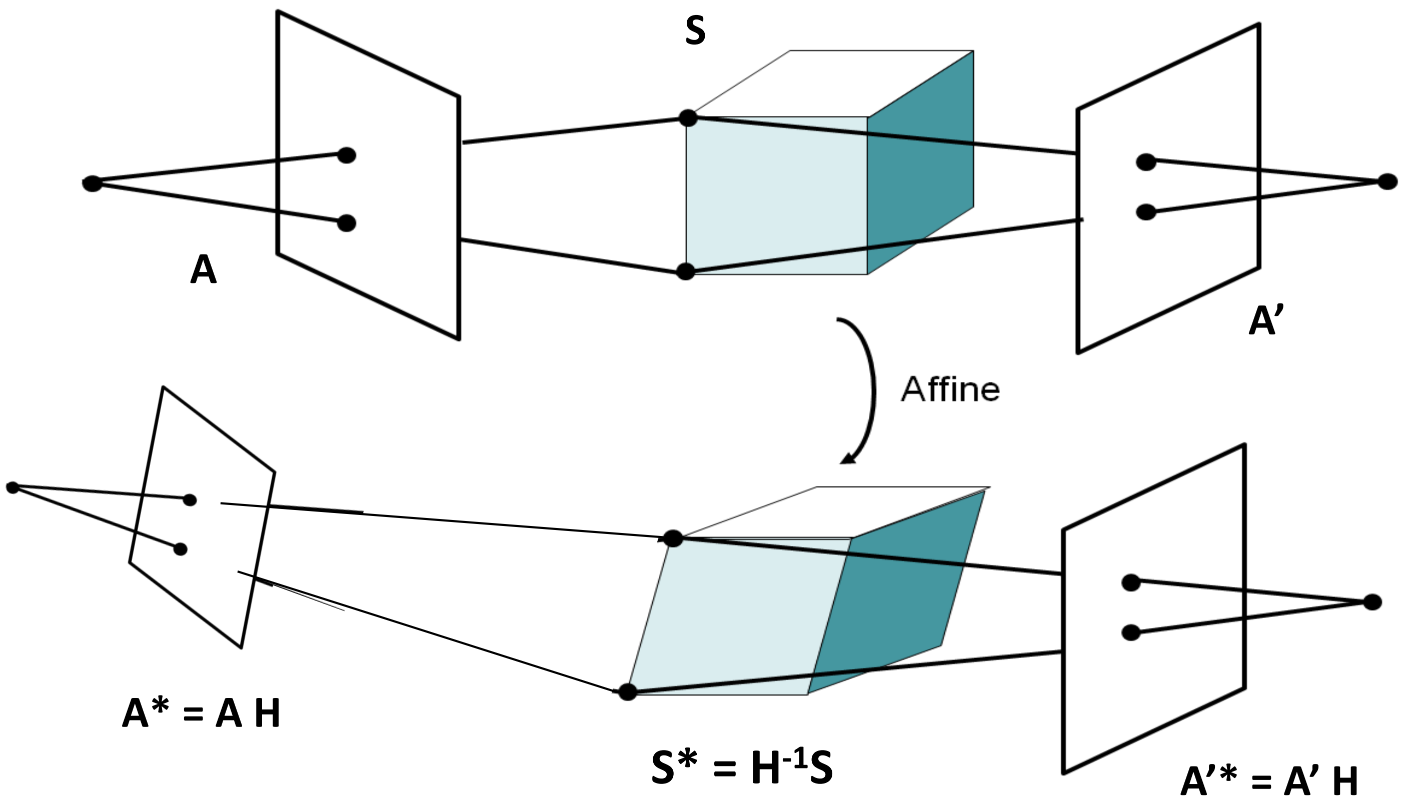
$$\mathbf{M^* = M\,H}$$

$$\mathbf{S^* = H^{-1}S}$$

where **H** is an arbitrary 3x3 matrix describing an affine transformation

- Additional constraints must be enforced to resolve this ambiguity

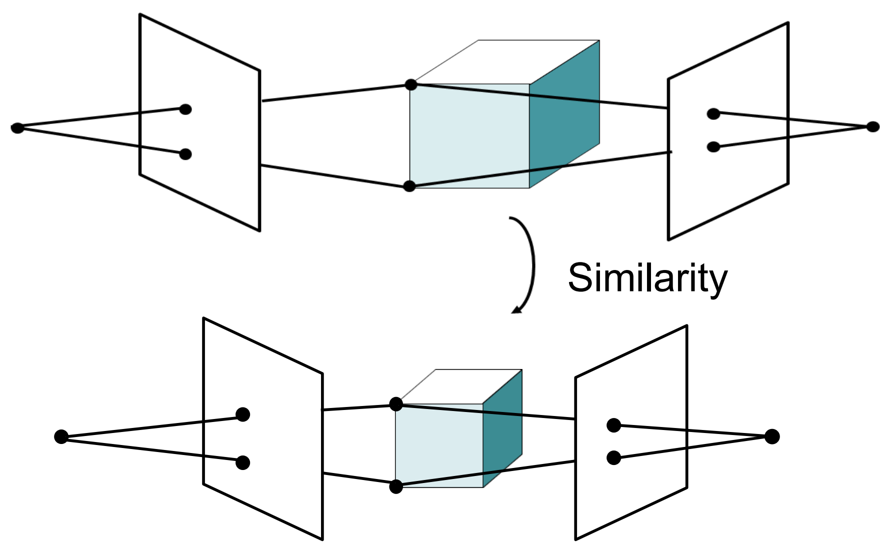
Affine Ambiguity



Similarity Ambiguity

- The scene is determined by the images only up a similarity transformation (rotation, translation and scaling)

- This is called **metric reconstruction**



- The ambiguity exists even for (intrinsically) calibrated cameras
- For calibrated cameras, the similarity ambiguity is the only ambiguity

[Longuet-Higgins '81]

Similarity Ambiguity

- It is impossible, based on the images alone, to estimate the absolute scale of the scene

Limitations

- Factorization methods assume all points are visible. Untrue when:
    - occlusions occur
    - failure in establishing correspondences

- Affine approximation is often too crude when:
    - objects are close to camera

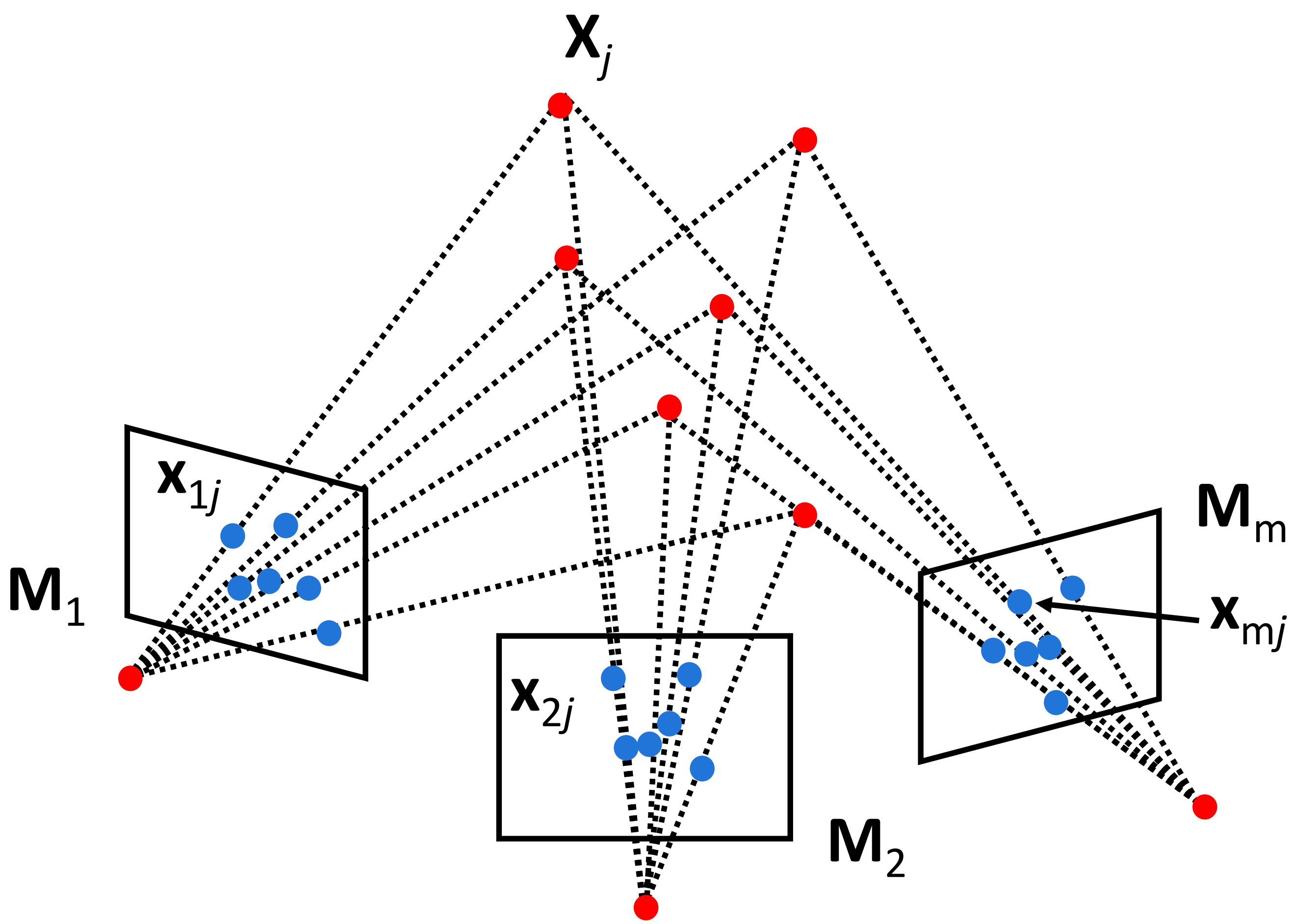
3D reconstruction from images

- The SFM problem
- Affine SFM
- Perspective SFM
- Bundle Adjustment

3D Scene Understanding

- Motivation
- Single view 3D scene understanding
- Multi-views 3D scene understanding

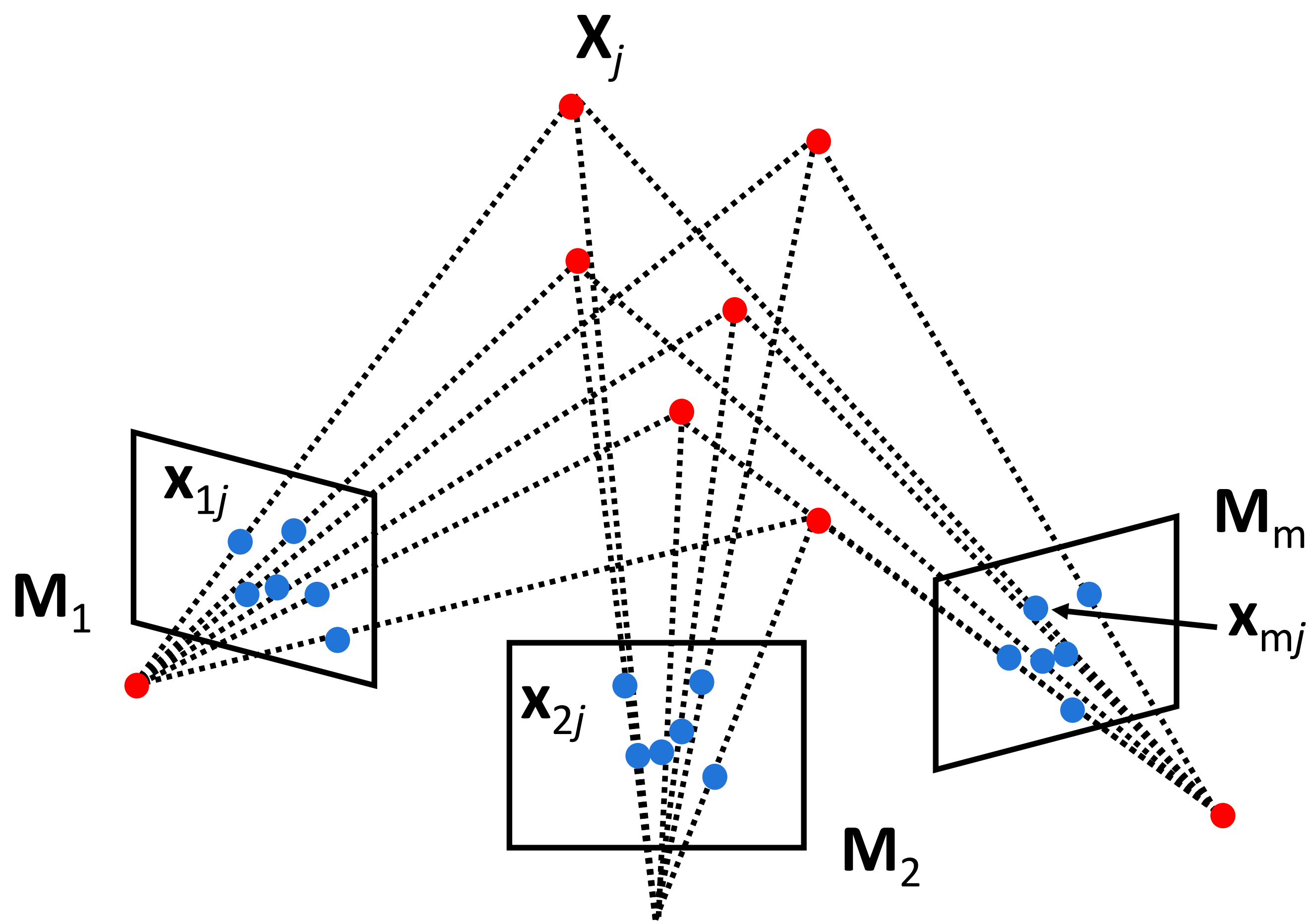
Structure from motion problem



From the mxn observations $\mathbf{x}_{ij}$, estimate:

- m projection matrices $\mathbf{M}_i$ = motion
- n 3D points $\mathbf{X}_j$ = structure

Structure from motion problem



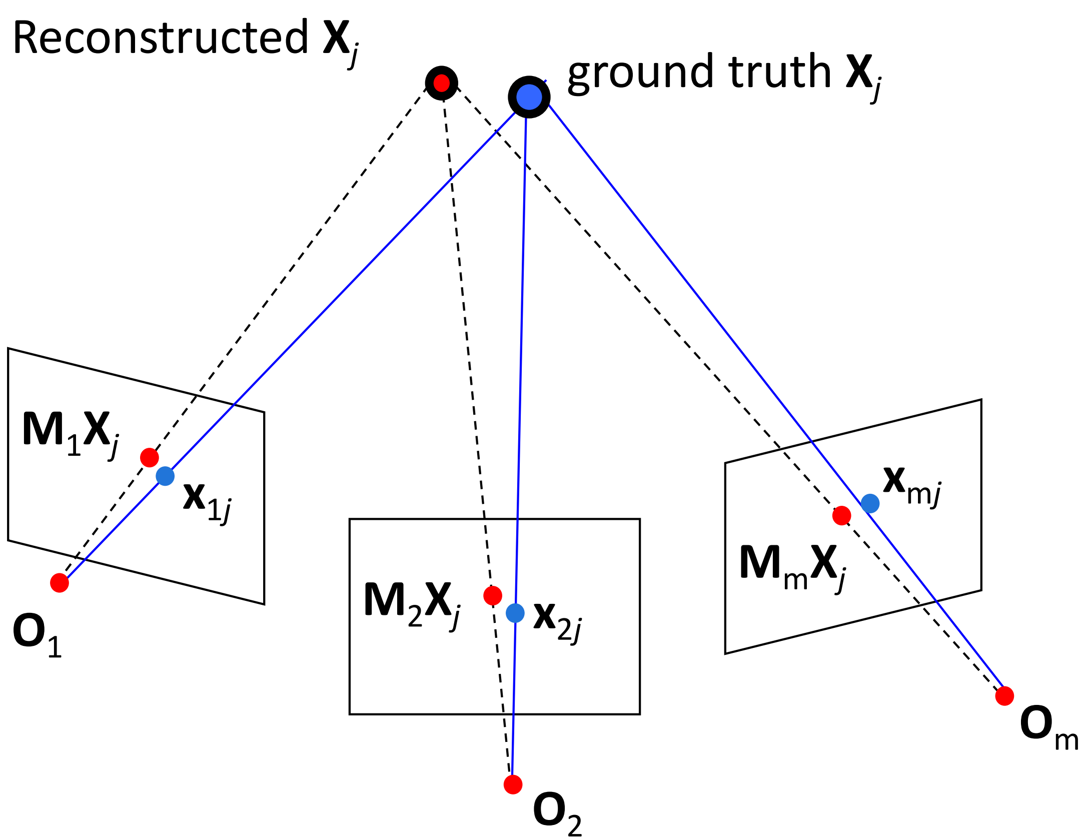
m cameras $M_1 \ldots M_m$

$$
\mathrm{M}_{\mathrm{i}} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & 1 \end{bmatrix}
$$

Bundle adjustment

- Non-linear method for refining structure and motion
- Minimizes re-projection error

$$E(M, \mathbf{X}) = \sum_{i=1}^{m} \sum_{j=1}^{n} D\left(\mathbf{x}_{ij}, M_i \mathbf{X}_j\right)^2$$



General Calibration Problem

$$E(M,\mathbf{X}) = \sum_{i=1}^{m}\sum_{j=1}^{n} D\left(\mathbf{x}_{ij}, M_i\mathbf{X}_j\right)^2$$

parameters

measurements

D is the nonlinear mapping

- Newton Method
- Levenberg-Marquardt Algorithm

• Iterative, starts from initial solution
• May be slow if initial solution far from real solution
• Estimated solution may be function of the initial solution
• Newton requires the computation of J, H
• Levenberg-Marquardt doesn't require the computation of H

Bundle adjustment

- Advantages
  - Handle large number of views
  - Handle missing data
- Limitations
  - Large minimization problem (parameters grow with number of views)
  - Requires good initial condition

- Used as the final step of SFM (i.e., after the factorization or algebraic approach)
- Factorization or algebraic approaches provide a initial solution for optimization problem

3D reconstruction from multiple views

Snavely et al., 06-08

3D reconstruction from multiple views

Snavely et al., 06-08

3D reconstruction from images

- The SFM problem
- Affine SFM
- Perspective SFM
- Bundle Adjustment

3D Scene Understanding

- Motivation
- Single view 3D scene understanding
- Multi-views 3D scene understanding

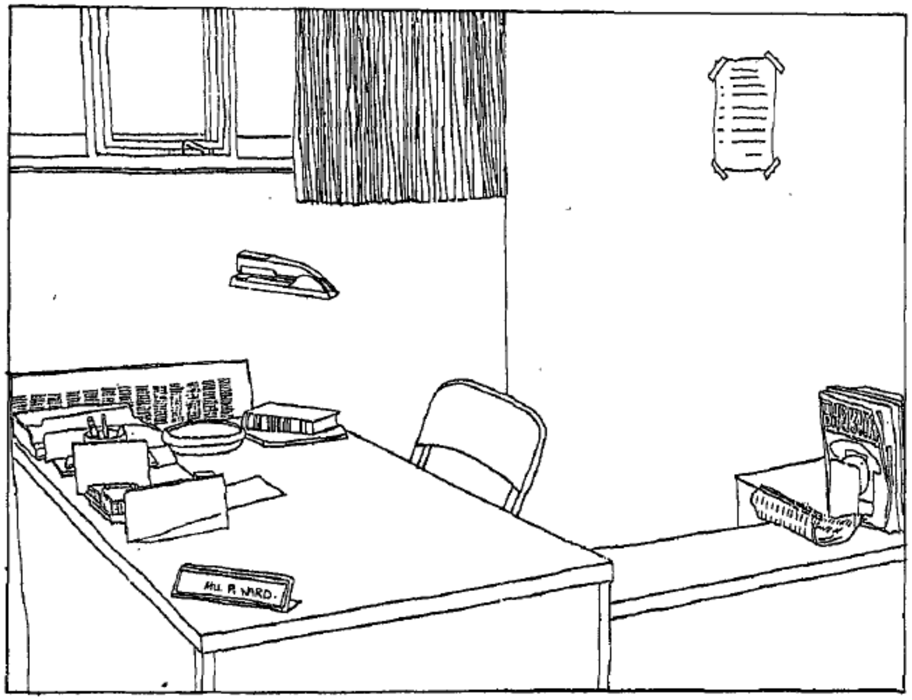

Lombard Street, San Francisco (2)
DO NOT
ENTER
WorldOnaBike.com
(c) Harry Kikstra, WorldOnaBike.co

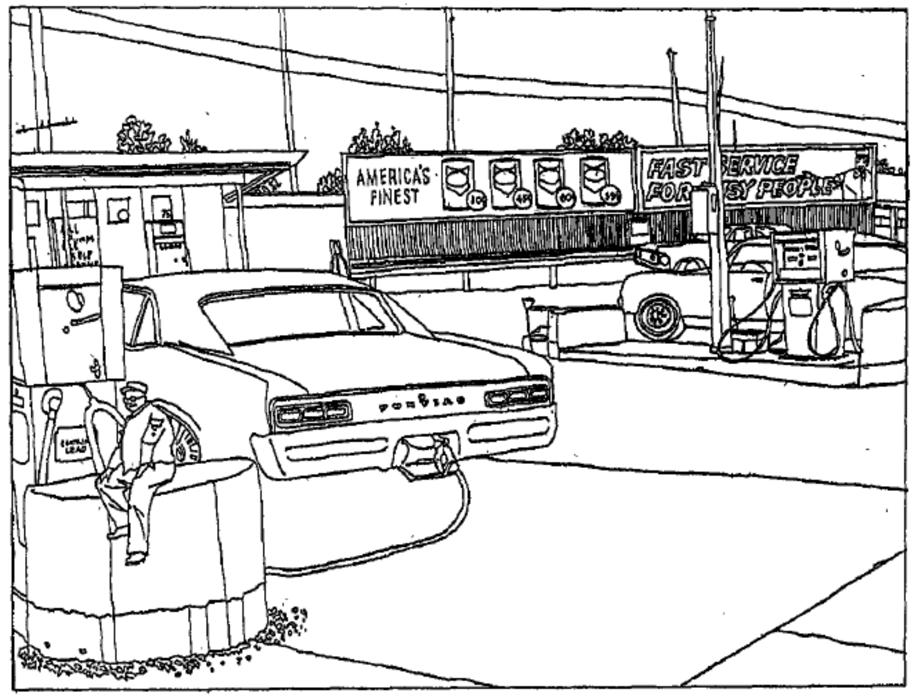
Why is this important?

Cherries or watermelon?

Cherries or watermelon?

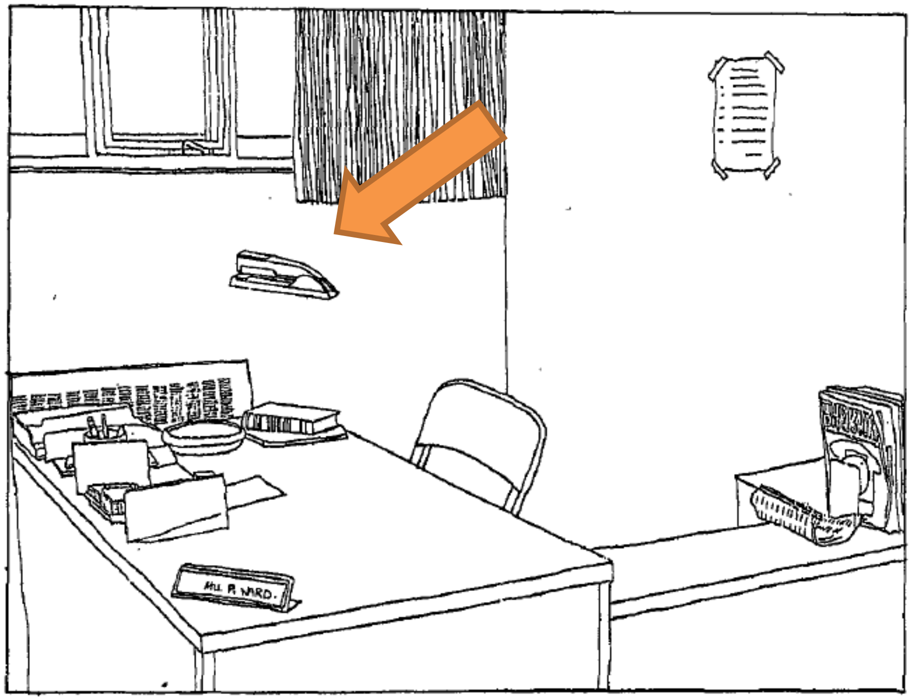
Humans perceive the world in 3D!

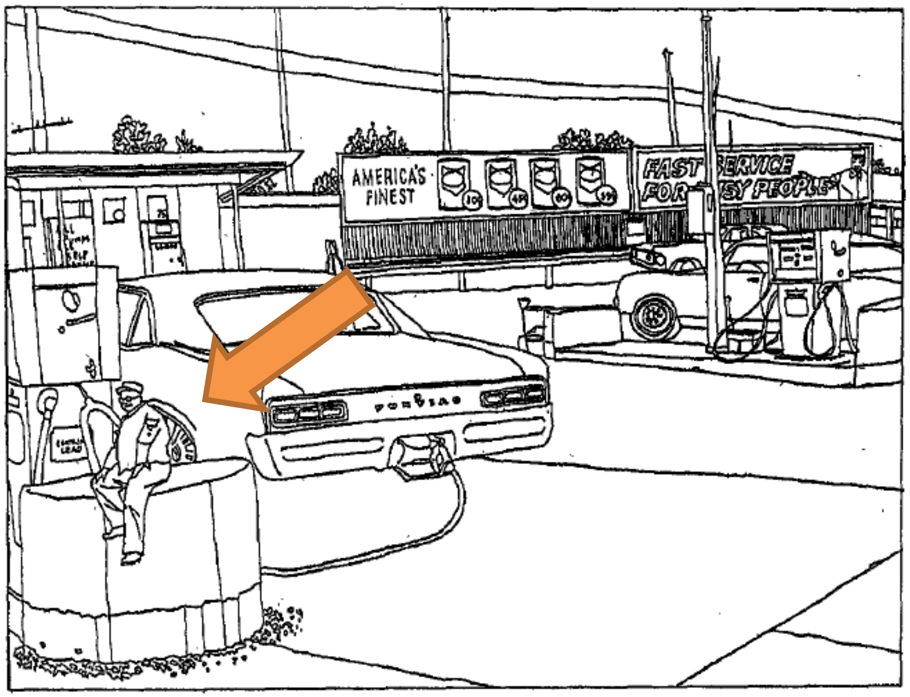




Biederman, Mezzanotte and Rabinowitz, 1982

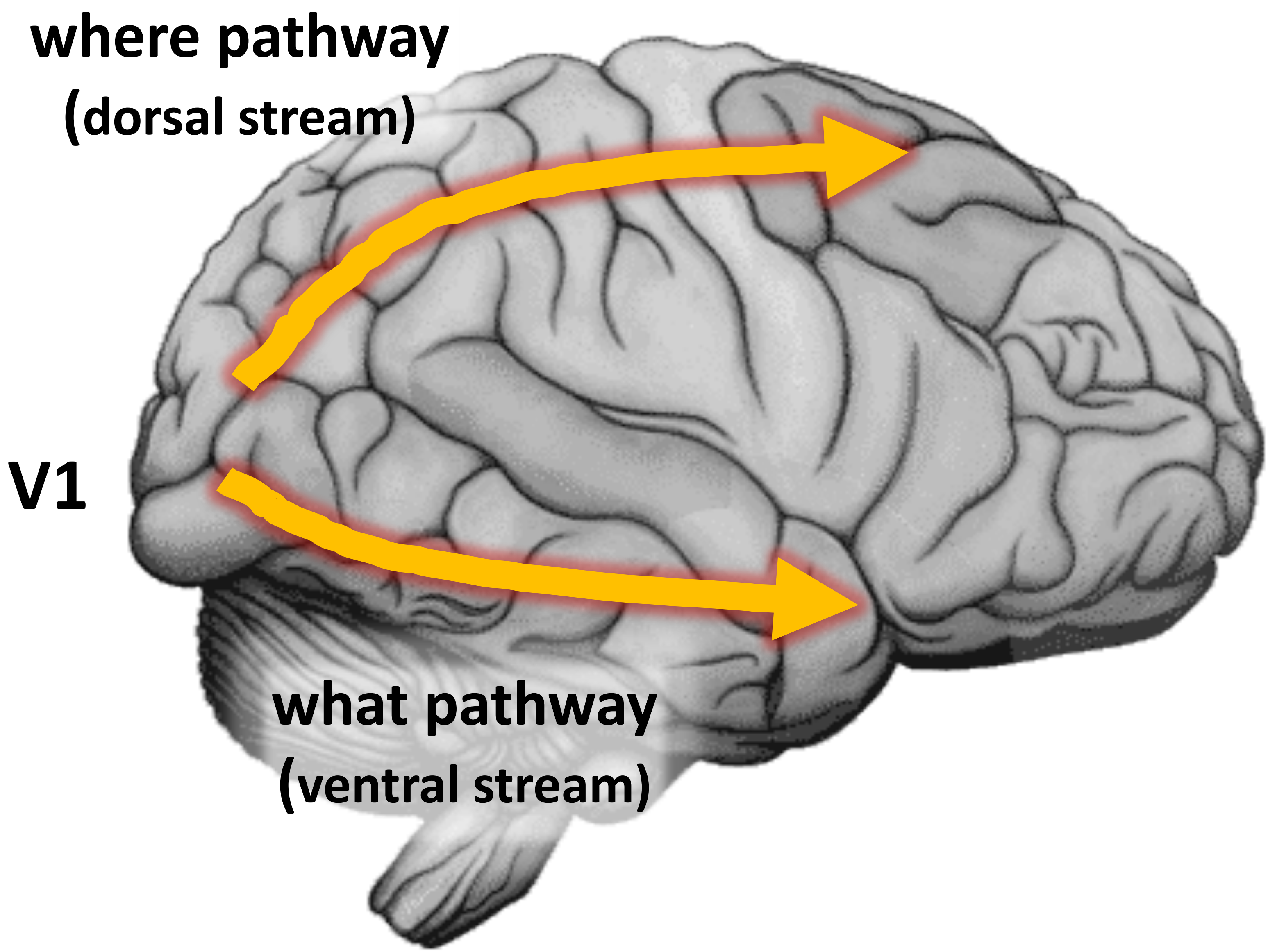
Humans perceive the world in 3D!





Biederman, Mezzanotte and Rabinowitz, 1982

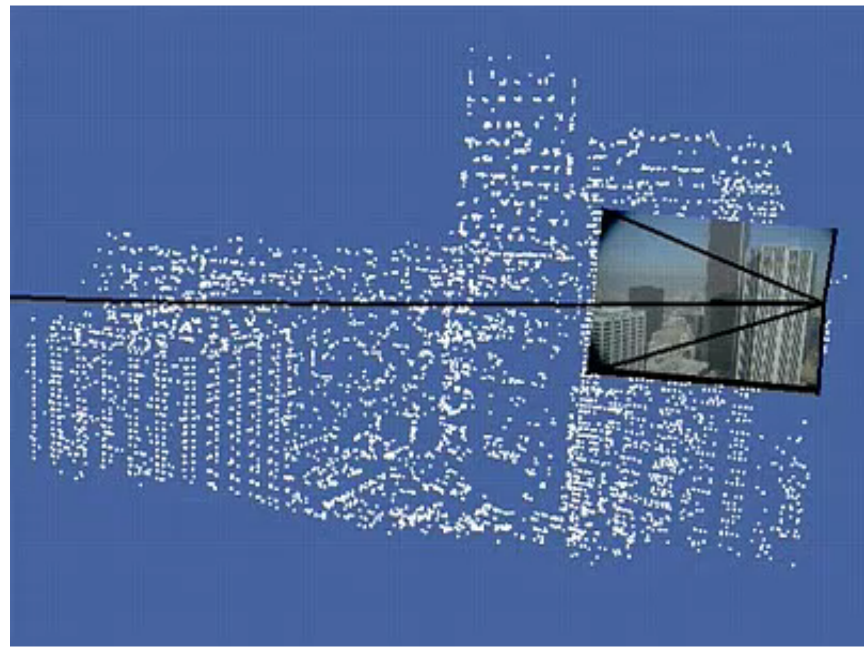
Humans perceive the world in 3D!



Representing the 3D space

Representing the 3D space

- 3D point clouds (2D features are associated to 3D points)



Courtesy of Oxford Visual Geometry Group

3D points clouds are built from SFM or SLAM

Fitzgibbon & Zisserman, 98
Triggs et al., 99
Pollefeys et al., 99
Kutulakos & Seitz, 99

Lucas & Kanade, 81
Chen & Medioni, 92
Debevec et al., 96
Levoy & Hanrahan, 96

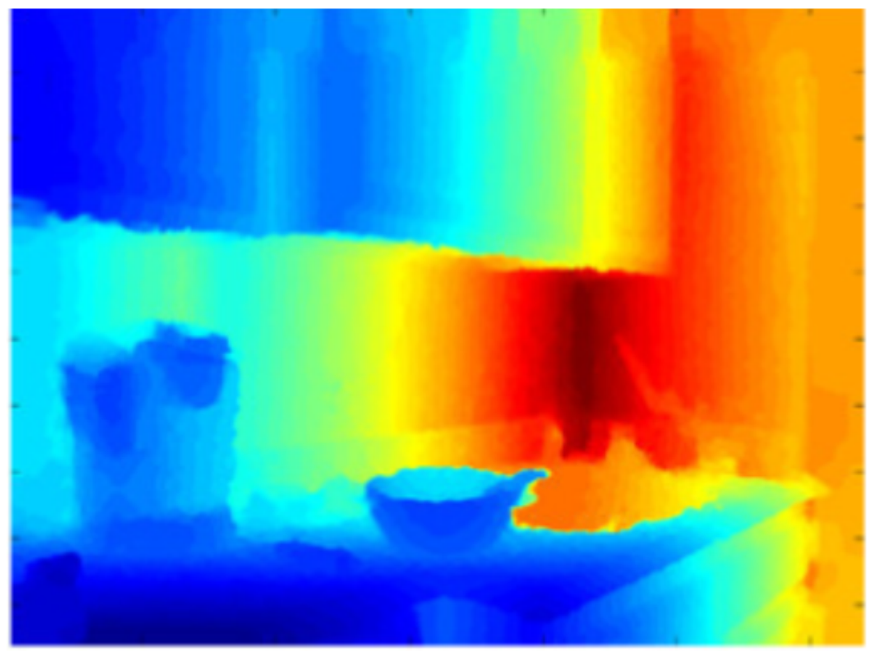
Levoy et al., 00
Hartley & Zisserman, 00
Dellaert et al., 00
Rusinkiewic et al., 02
Nistér, 04
Brown & Lowe, 04

Schindler et al., 04
Lourakis & Argyros, 04
Colombo et al., 05
Savarese et al., IJCV 05
Savarese et al., IJCV 06
Saxena et al., 07-09

Snavely et al., 06-08
Schindler et al., 08
Agarwal et al., 09
Frahm et al., 10
Golparvar-Fard, et al. JAEI 10
Pandey et al. IFAC , 2010
Pandey et al. ICRA 2011

Representing the 3D space

- Retinotopics (each 2D pixel is associated to a depth value)
  - Depth maps (from Stereo, D-RGB, etc....)

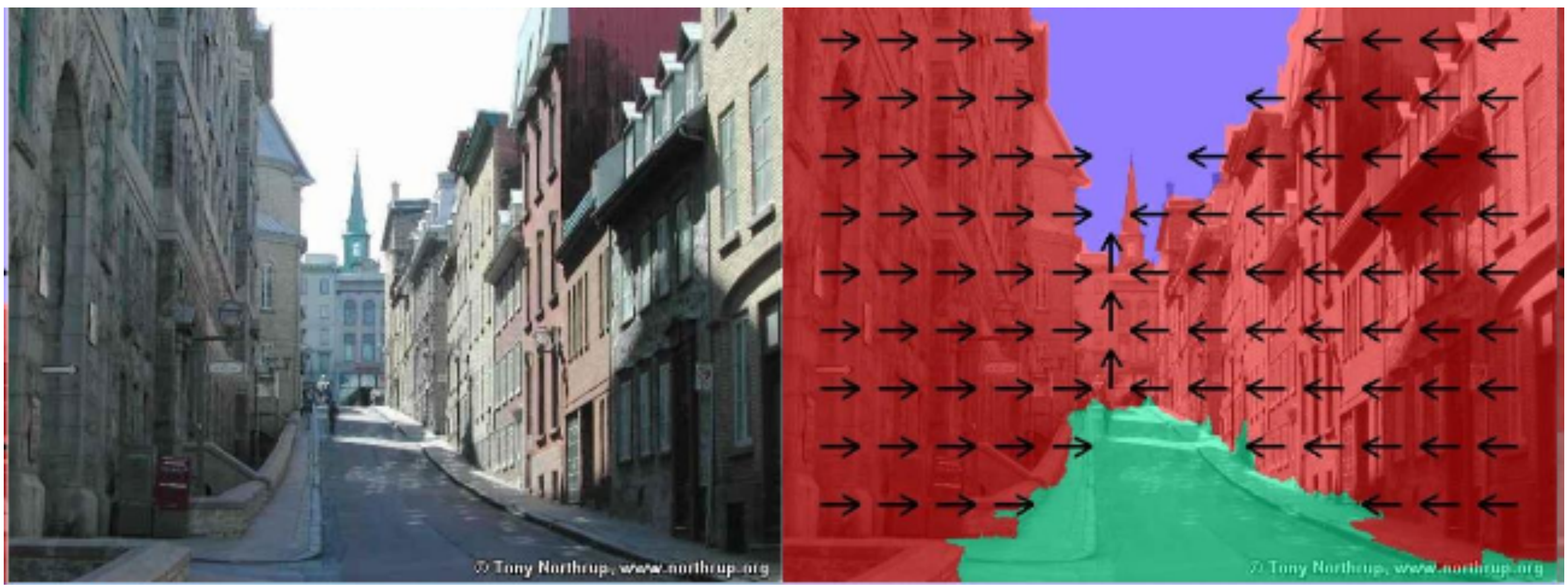


From X. Ren et al., CVPR 11, UW-dataset

Representing the 3D space

- Retinotopics (each 2D pixel is associated to a 3D property)
    - Depth maps (from Stereo, D-RGB, etc….)
    - Orientation maps (from single view)

D Hoiem, AA Efros, M Hebert , 2007

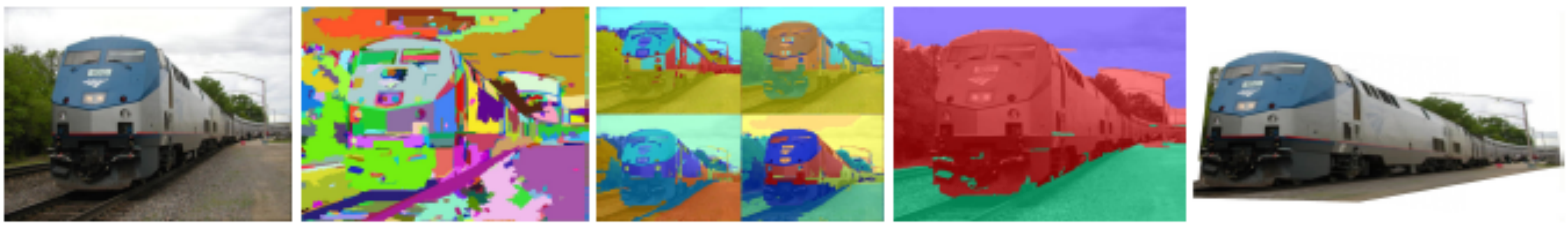


Hoiem et al. 05

Representing the 3D space

- Retinotopics (each 2D pixel is associated to a depth value)
  - Depth maps (from Stereo, D-RGB, etc….)
  - Orientation maps (from single view)

D Hoiem, AA Efros, M Hebert , 2007



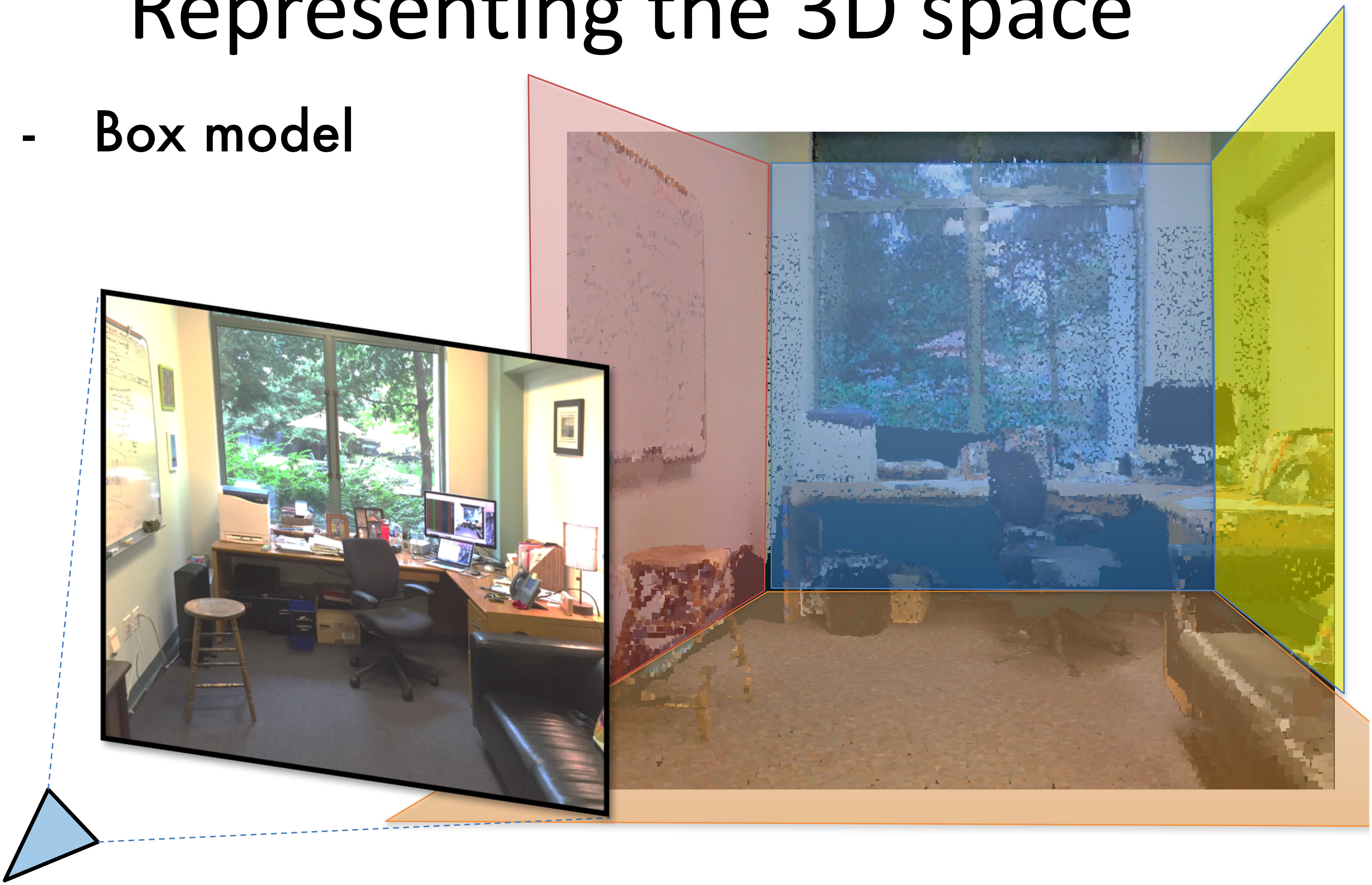
Hoiem et al. 05

Representing the 3D space

- Box model

Representing the 3D space

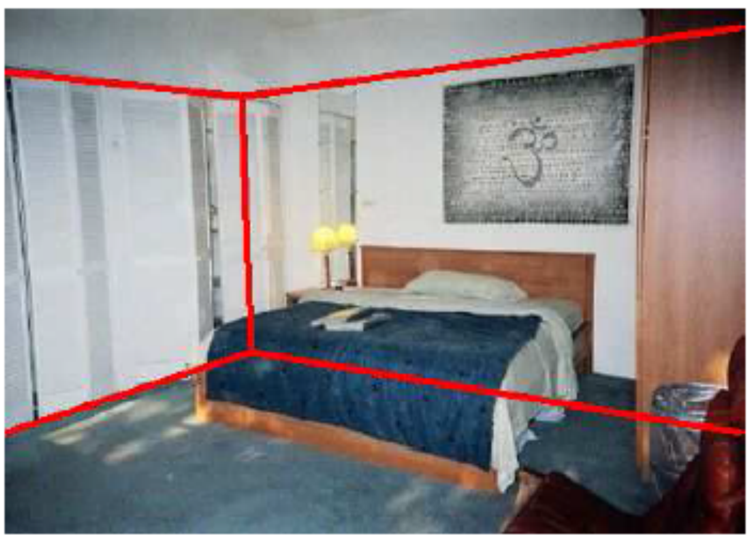
- Box model



Representing the 3D space

- Box model

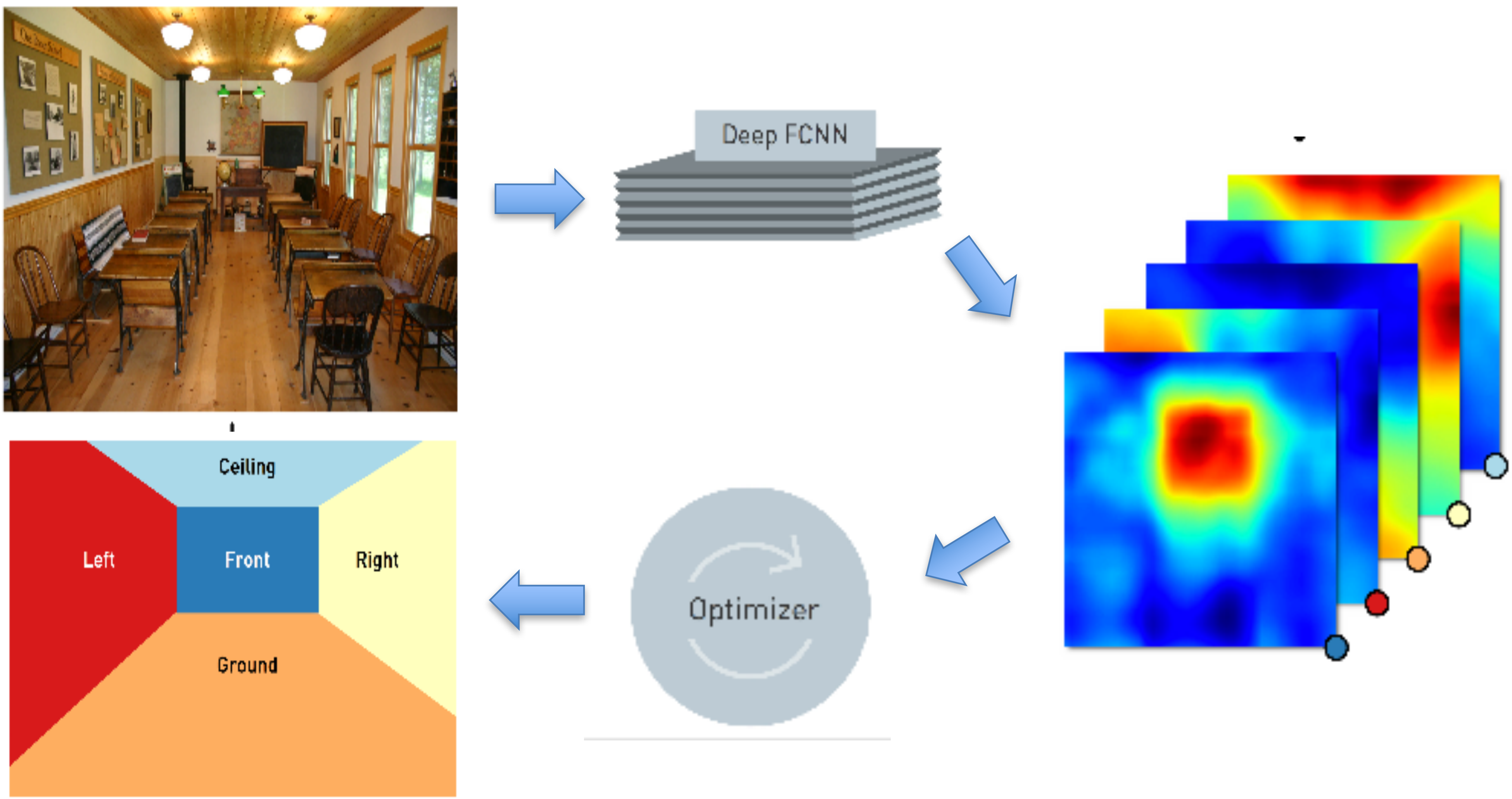
- Hoiem et al. 06-10
- Saxena et al. 06-09
- Gould et al. 09
- Hedau et al. 09
- Bao, et al. CVPR 2010
- Choi et al., 2013
- Lee et al. 09,10
- Gupta et al. 10, 11
- Koppula et al. 11
- Guo & Hoiem 12
- Del Pero et al., 12
- Schwing & Urtasun, 12



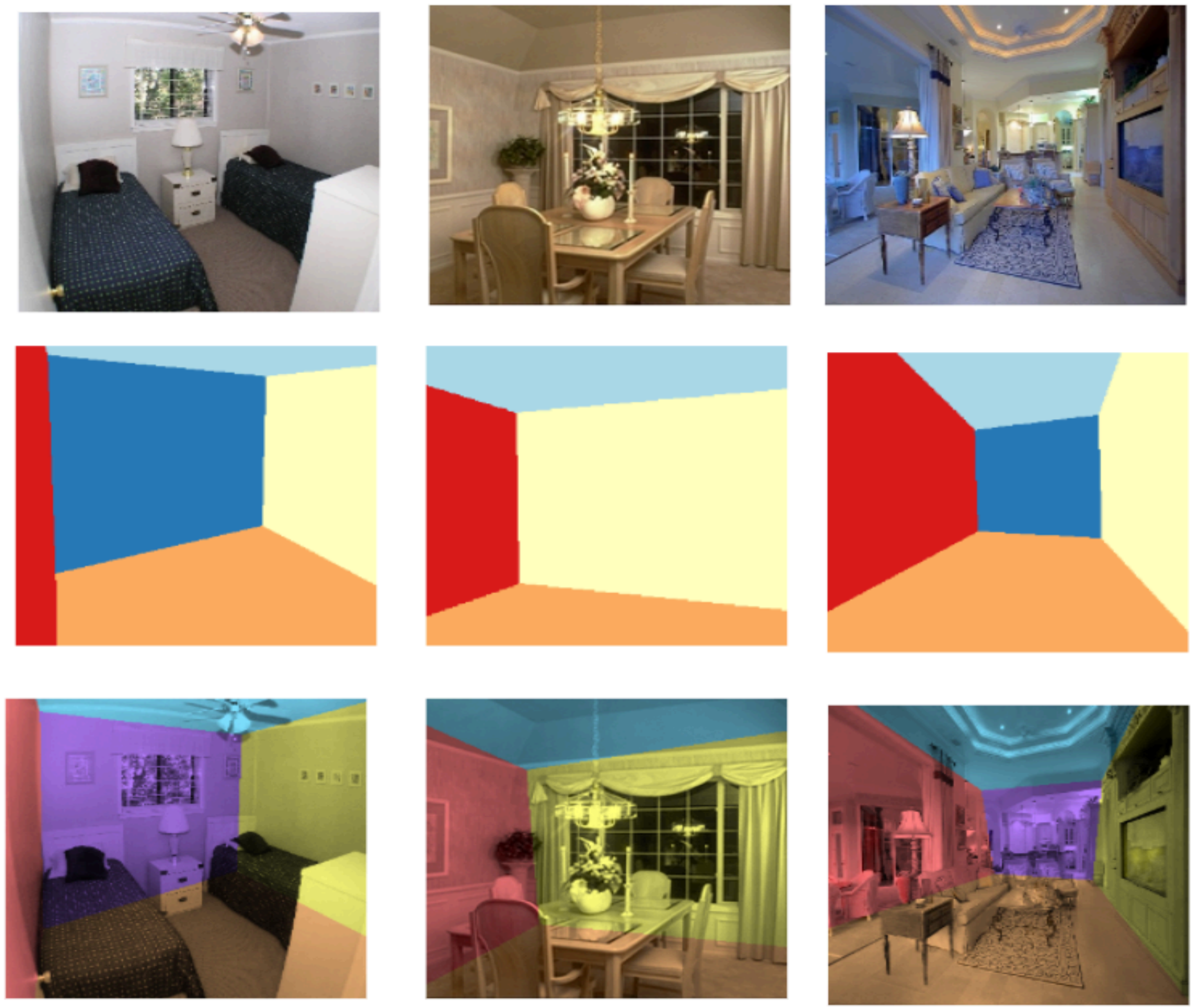
Hedau et al. 09

Learning a box model using CNNs

Dasgupta, Chen, Fang, et al. CVPR 2016



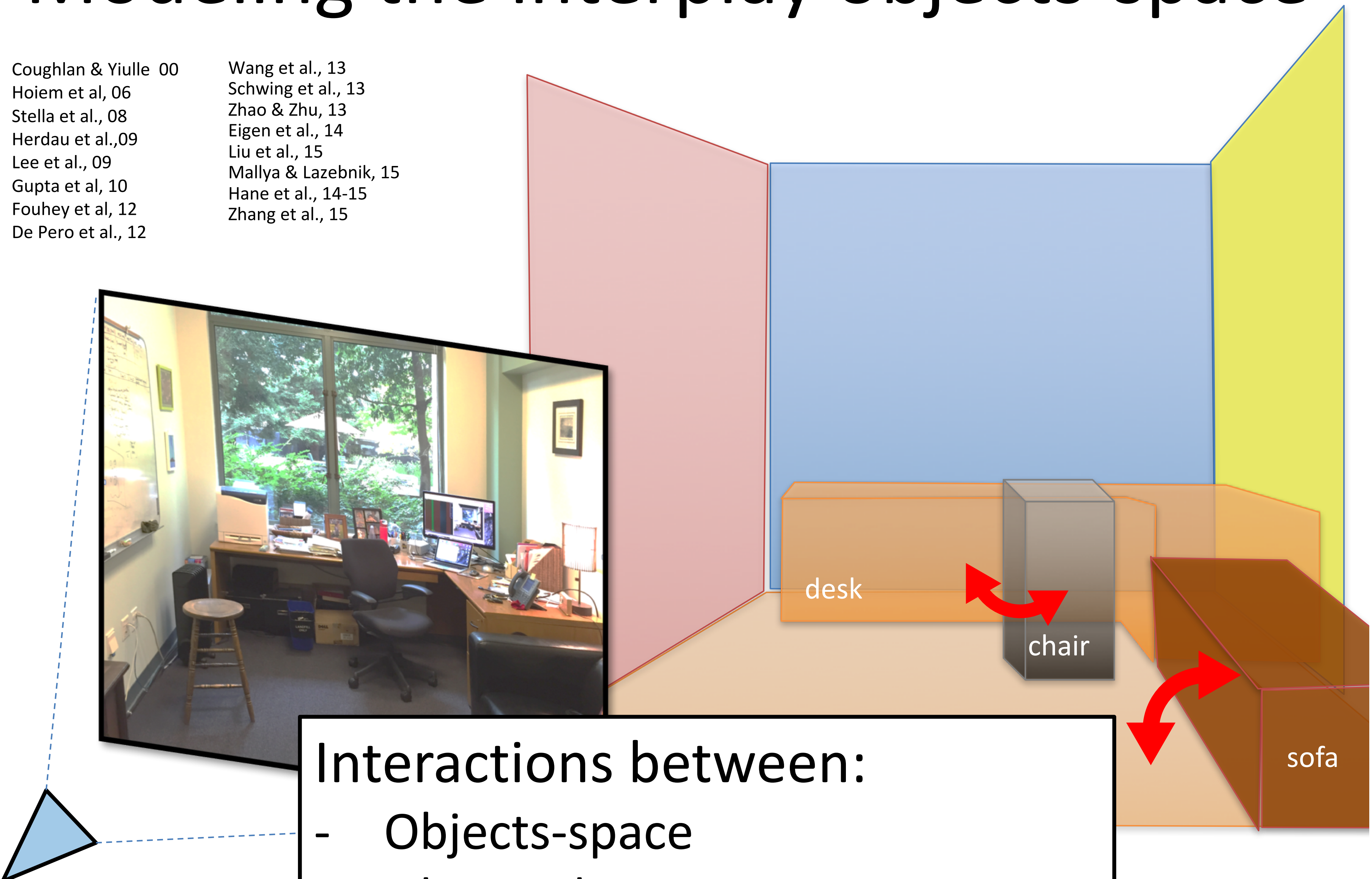
Some results



Modeling the interplay objects-space

Coughlan & Yiulle 00
Hoiem et al, 06
Stella et al., 08
Herdau et al.,09
Lee et al., 09
Gupta et al, 10
Fouhey et al, 12
De Pero et al., 12

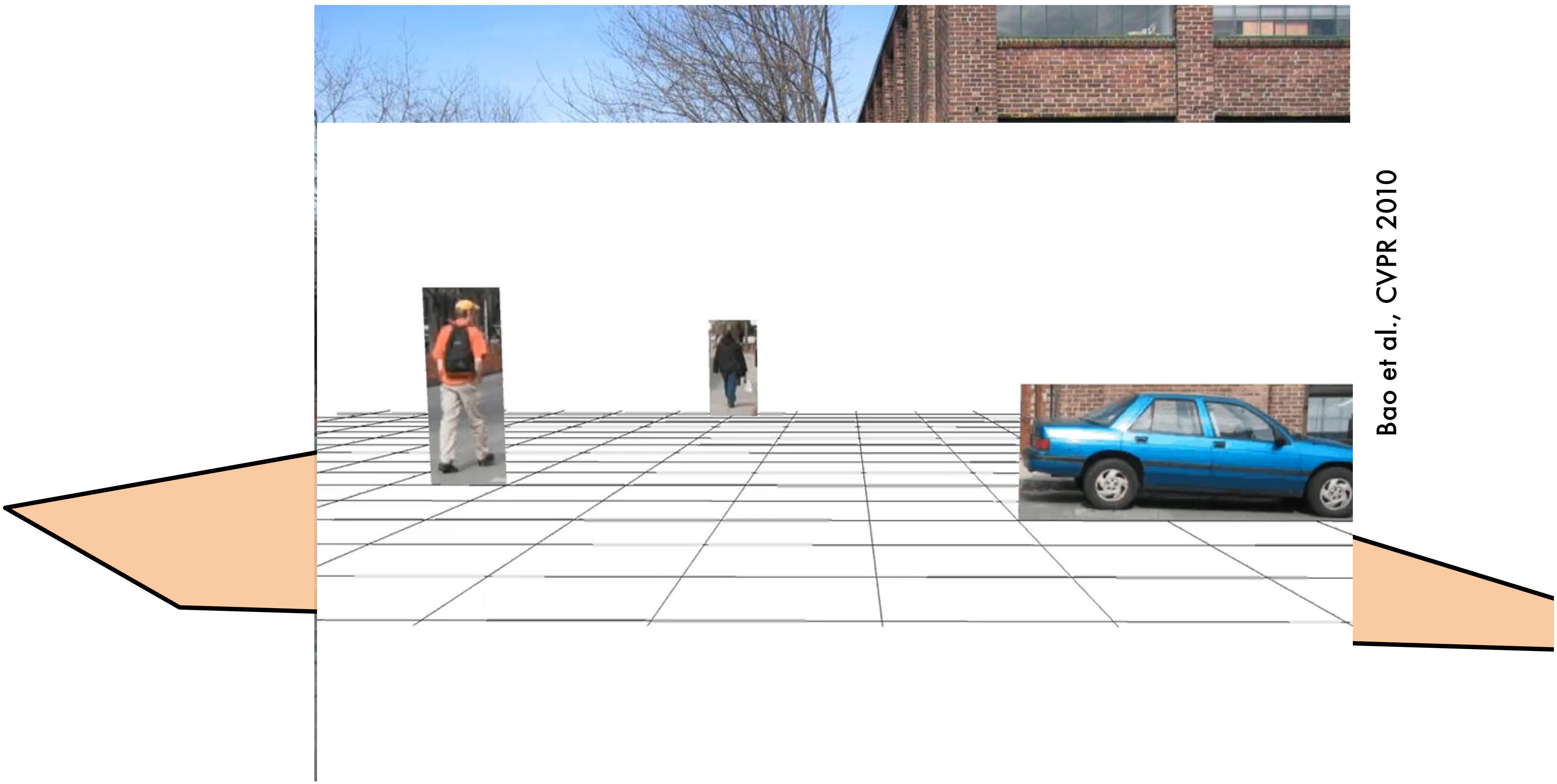
Wang et al., 13
Schwing et al., 13
Zhao & Zhu, 13
Eigen et al., 14
Liu et al., 15
Mallya & Lazebnik, 15
Hane et al., 14-15
Zhang et al., 15



Interactions between:
- Objects-space
- Object-object

Ground plane-objects

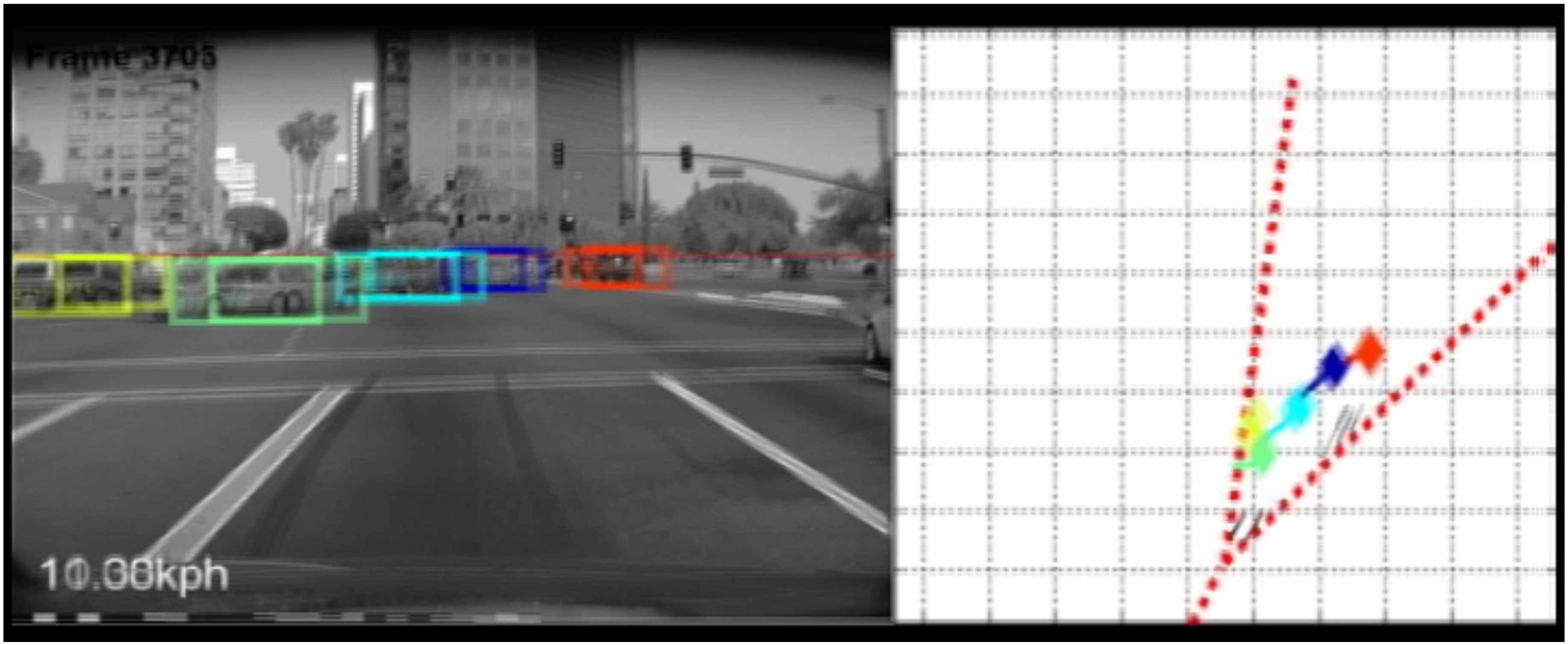
Space: ground plane
Objects: 3D pose + scale
Camera: weak perspective



Bao et al., CVPR 2010

Ground plane-objects

Choi et al., 2011

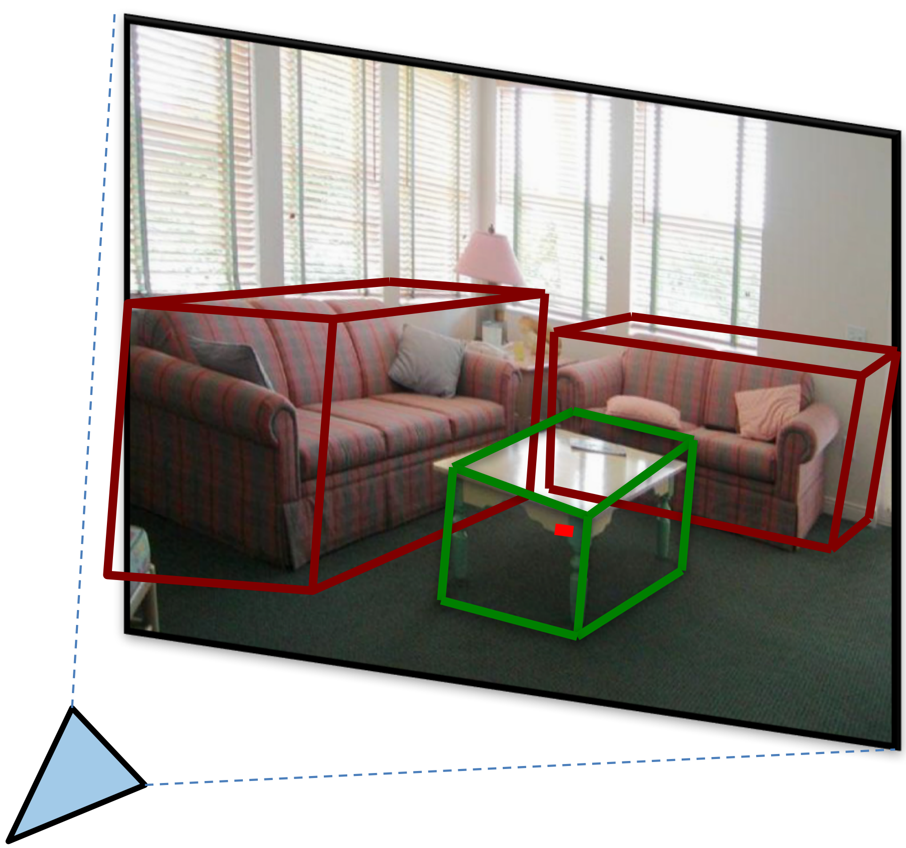


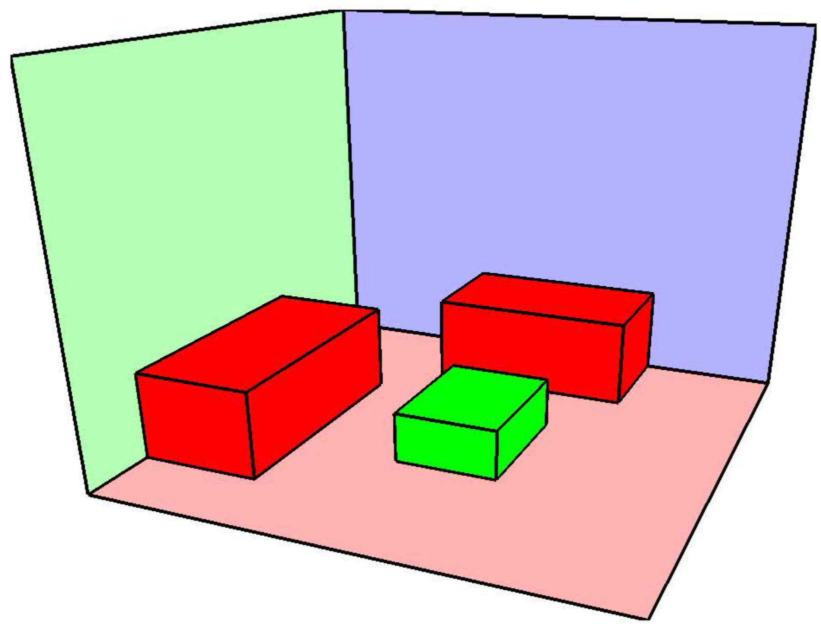
- Monocular cameras
- Un-calibrated cameras
- Arbitrary motion

3D Geometric Phrases

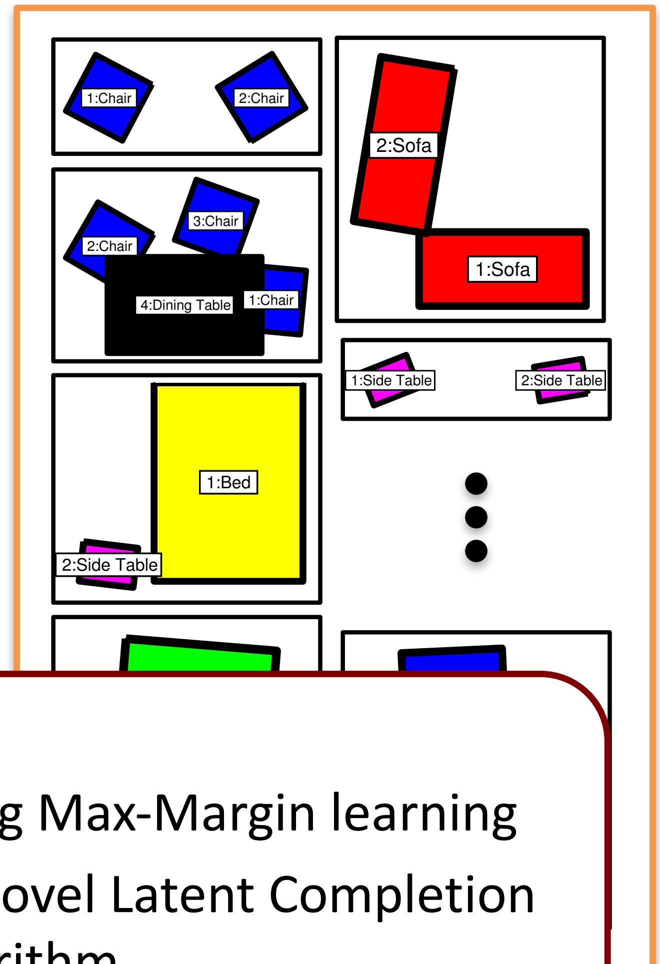
Choi et al, CVPR 13 , IJCV 15

Space: Box model
Objects: 3D pose + scale
Camera: Full perspective





3D Geometric Phrases

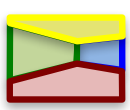


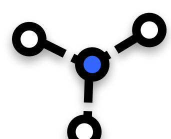
- **w/o annotations**
- **Compact**
- **View-invariant**

Using Max-Margin learning w/ novel Latent Completion algorithm

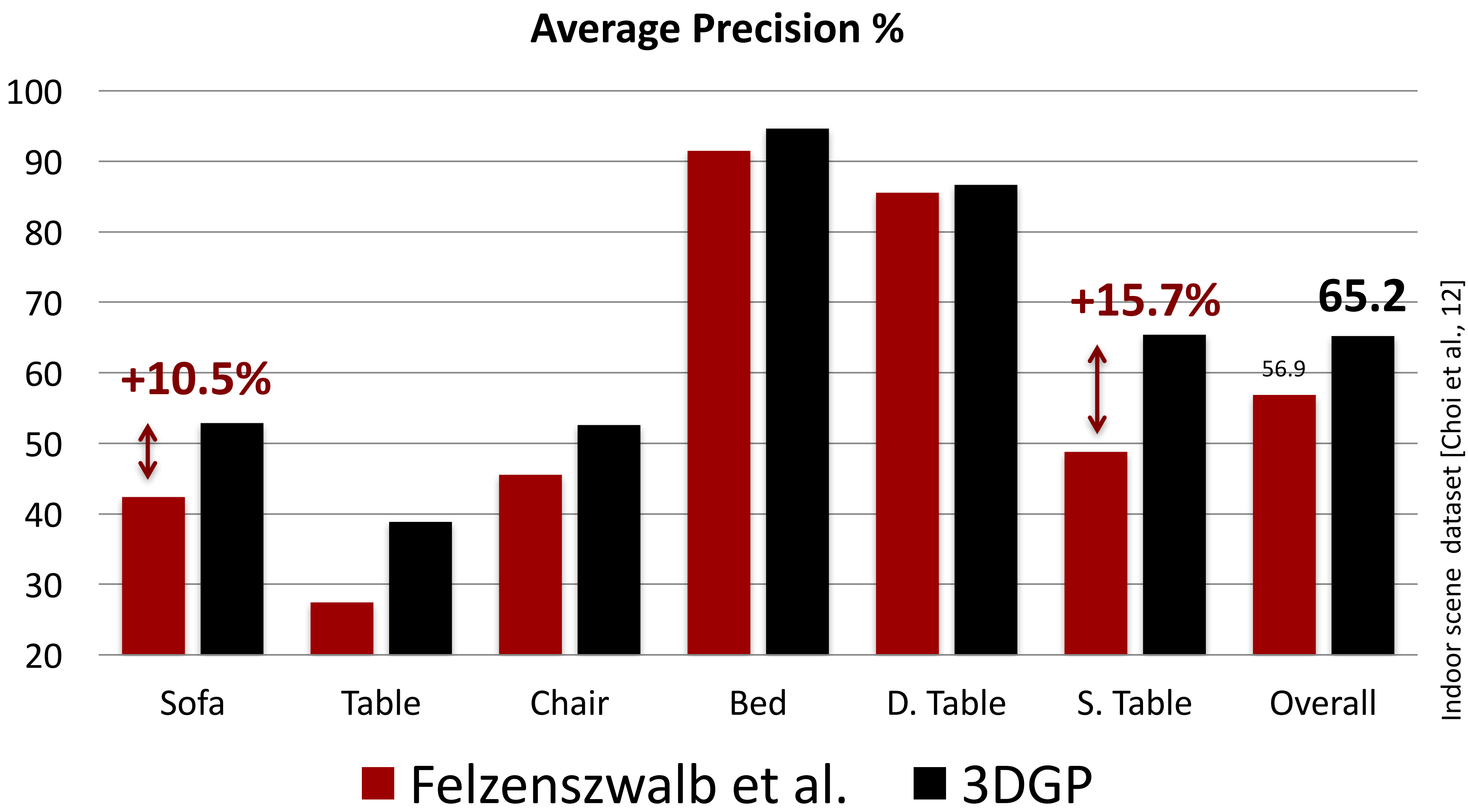
Scene understanding results

Sofa, Coffee Table, Chair, Bed, Dining Table, Side Table

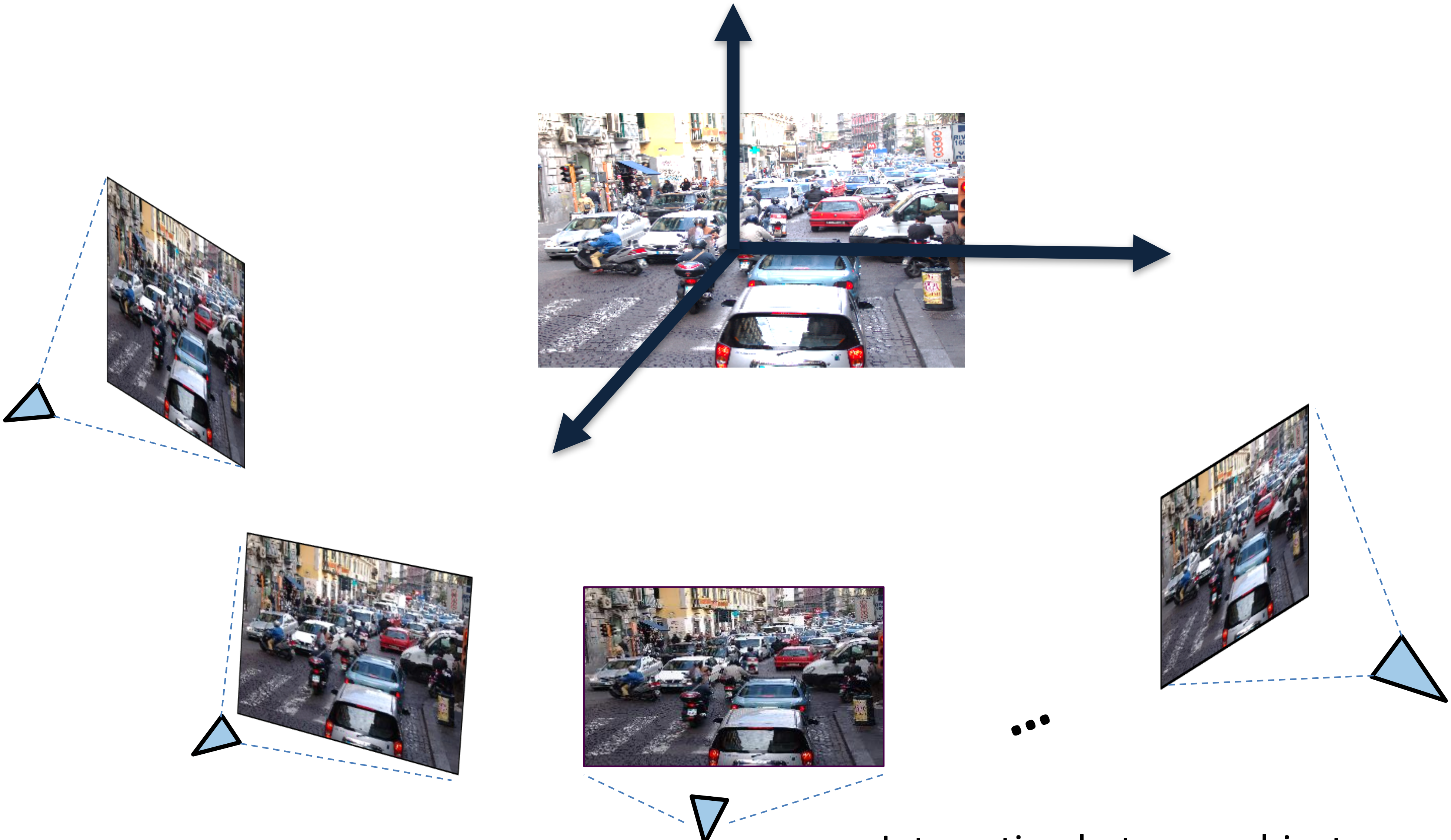
 Estimated Layout

 3D Geometric Phrases

Results: Object Detection

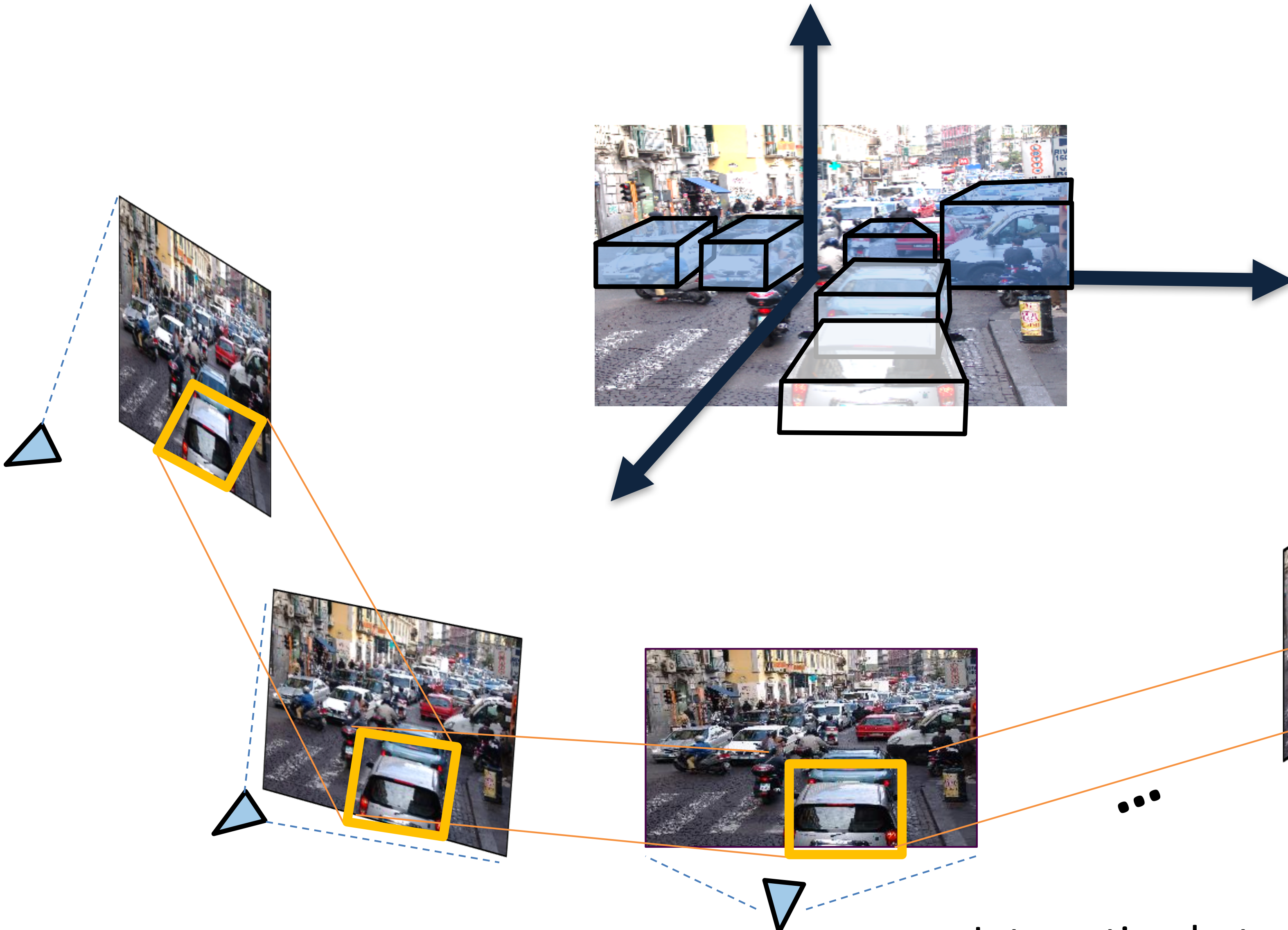


Modeling relationships of objects across views



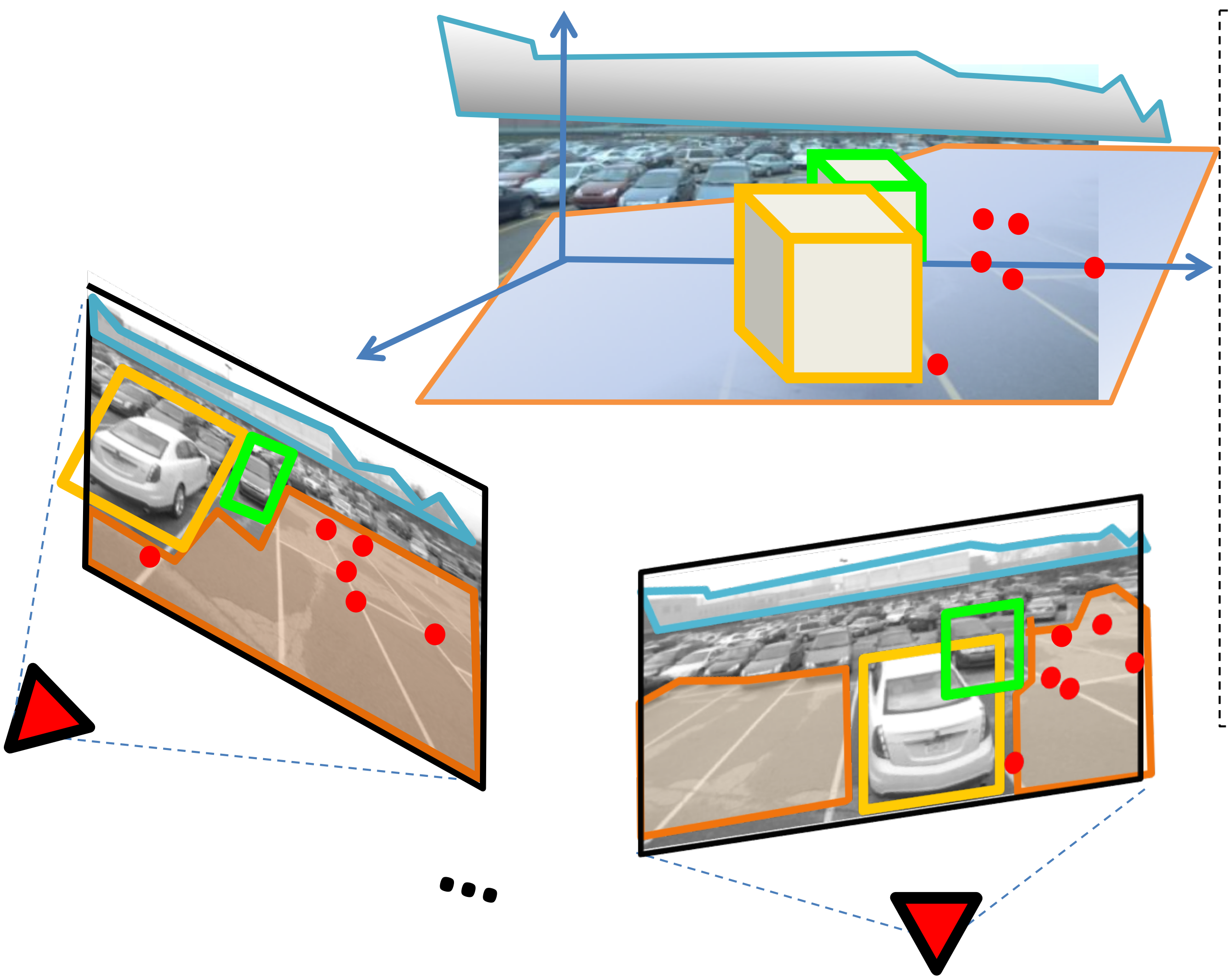
- Interaction between object-space
- Interaction among objects
- **Transfer semantics across views**

Modeling relationships of objects across views



- Interaction between object-space
- Interaction among objects
- **Transfer semantics across views**

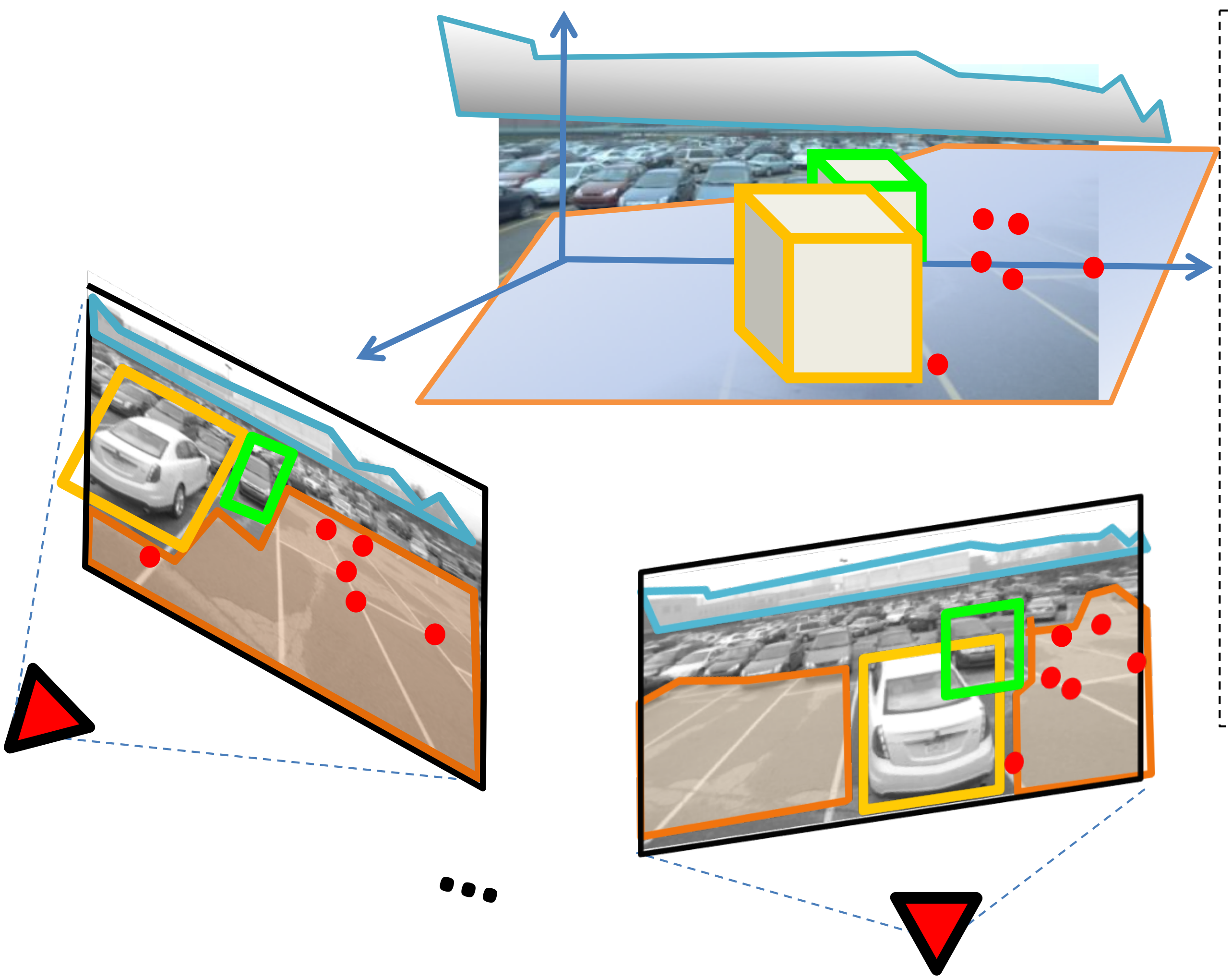
Semantic structure from motion



- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)
- Model Parameters:
  - Q = 3D points
  - O = 3D objects
  - B = 3D regions
  - **C** = cam. prm. K, R, T

Semantic structure from motion

$$\{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}\} = \arg \max_{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}} \Psi(\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}; \mathbf{I})$$

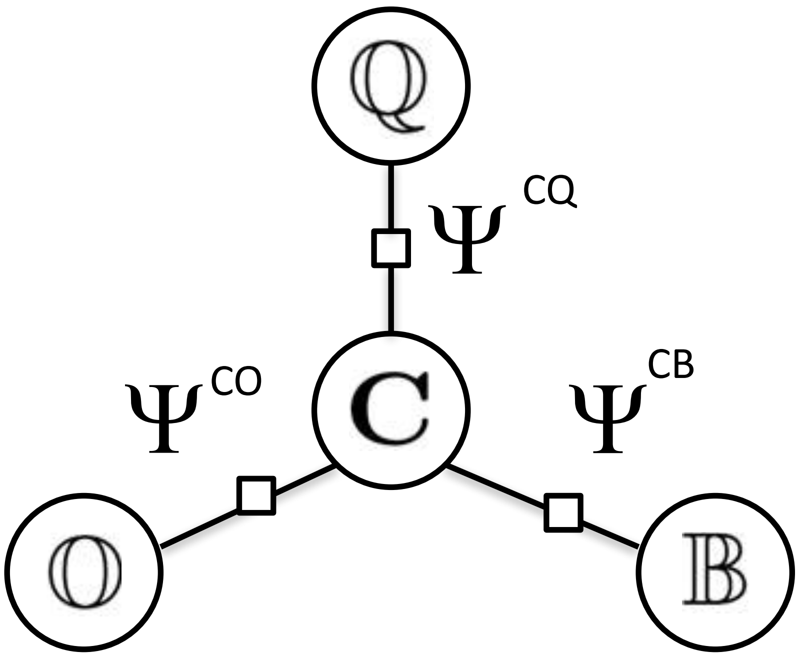


- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)
- Model Parameters:
  - $\mathbb{Q}$ = 3D points
  - $\mathbb{O}$ = 3D objects
  - $\mathbb{B}$ = 3D regions
  - $\mathbf{C}$ = cam. prm. K, R, T

Semantic structure from motion

$$\{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}\} = \arg \max_{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}} \prod_s \Psi_s^{CQ} \prod_t \Psi_t^{CO} \prod_r \Psi_r^{CB}$$

Factor graph

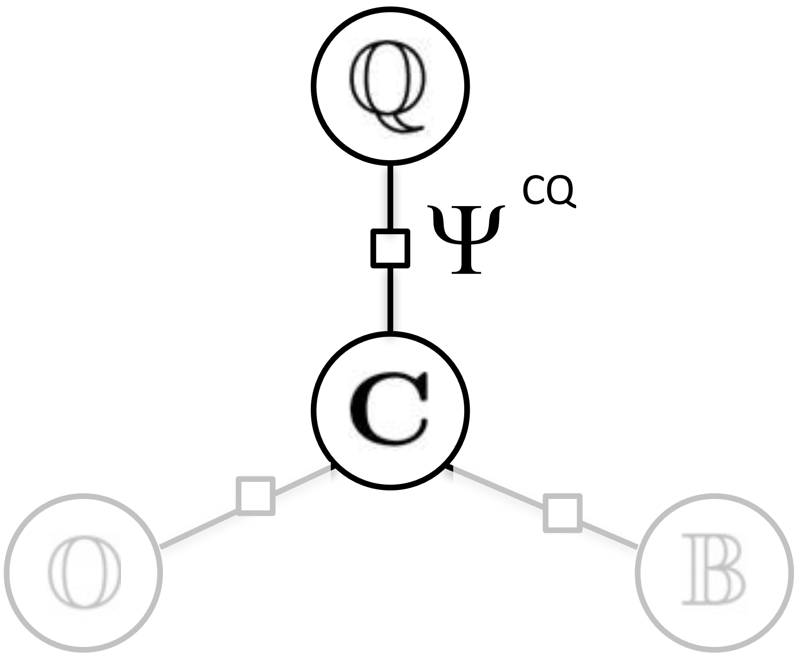


- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)

- Model Parameters:
  - Q = 3D points
  - O = 3D objects
  - B = 3D regions
  - **C** = cam. prm. K, R, T

SSFM: point-level compatibility

$$\{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}\} = \arg\max_{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}} \boxed{\prod_s \Psi_s^{CQ}} \prod_t \Psi_t^{CO} \prod_r \Psi_r^{CB}$$

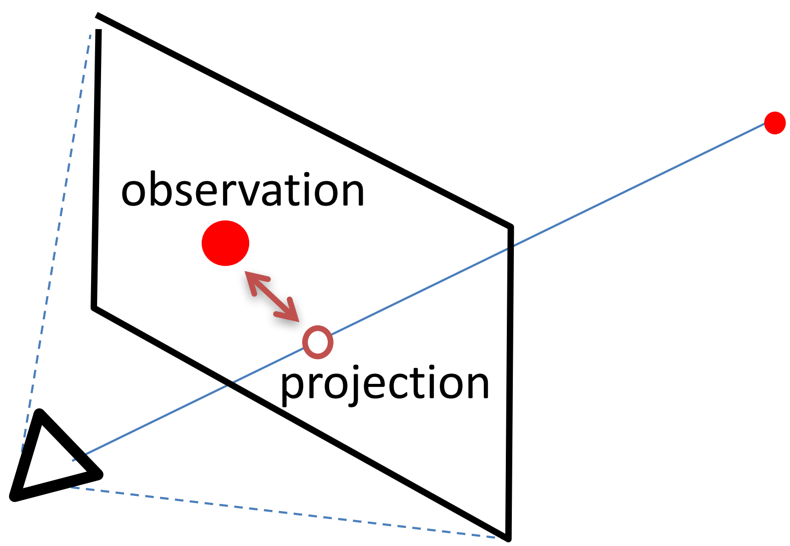


- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)

- Model Parameters:
  - Q = 3D points
  - O = 3D objects
  - B = 3D regions
  - **C** = cam. prm. K, R, T

SSFM: point-level compatibility

$$\{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}\} = \arg\max_{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}} \boxed{\prod_s \Psi_s^{CQ}} \prod_t \Psi_t^{CO} \prod_r \Psi_r^{CB}$$



- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)
- Model Parameters:
  - Q = 3D points
  - O = 3D objects
  - B = 3D regions
  - **C** = cam. prm. K, R, T

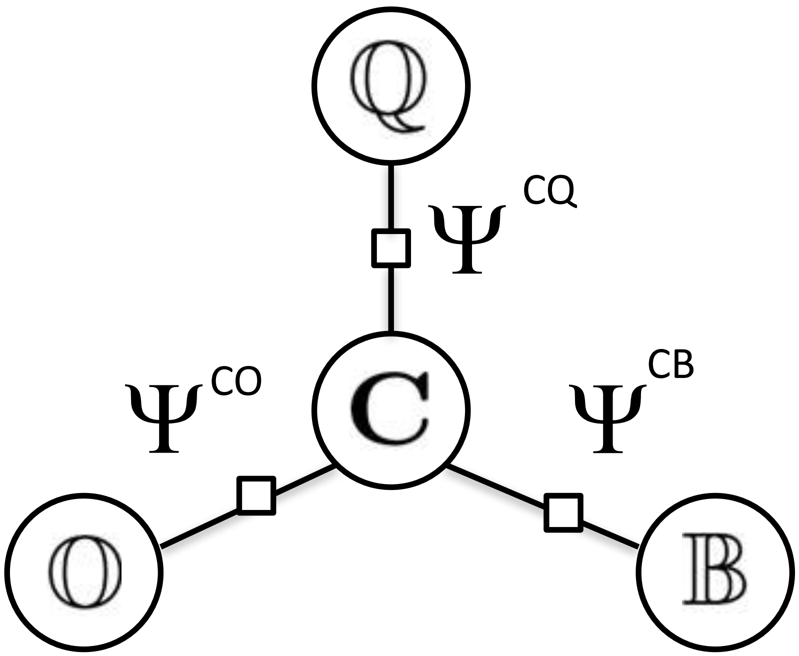
Point re-projection error

$$\prod_s \Psi_s^{CQ} \propto \prod_i^{N_Q} \prod_k^{N_k} \exp(-(q_i^k - q_{u_i^k}^k)^2/\sigma_q)$$

- Tomasi & Kanade '92
- Triggs et al '99
- Soatto & Perona 99
- Hartley & Zisserman 00
- Dellaert et al. 00
- Pollefeys & V. Gool 02
- Nister 04
- Brown & Lowe 07
- Snavely et al. 08

SSFM: Object-level compatibility

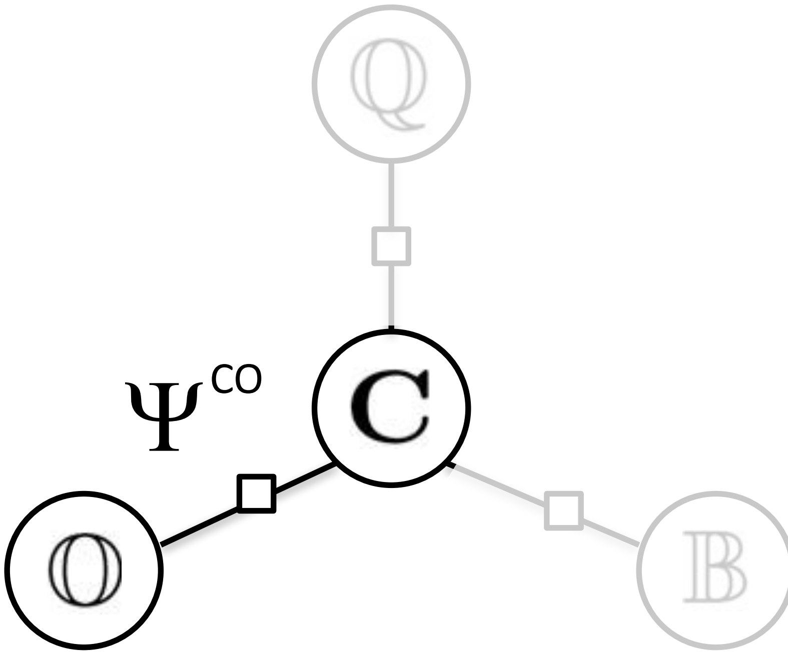
$$\{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}\} = \arg\max_{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}} \prod_s \Psi_s^{CQ} \boxed{\prod_t \Psi_t^{CO} \prod_r \Psi_r^{CB}}$$



- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)
- Model Parameters:
  - Q = 3D points
  - O = 3D objects
  - B = 3D regions
  - **C** = cam. prm. K, R, T

SSFM: Object-level compatibility

$$\{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}\} = \arg \max_{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}} \prod_s \Psi_s^{CQ} \boxed{\prod_t \Psi_t^{CO}} \prod_r \Psi_r^{CB}$$

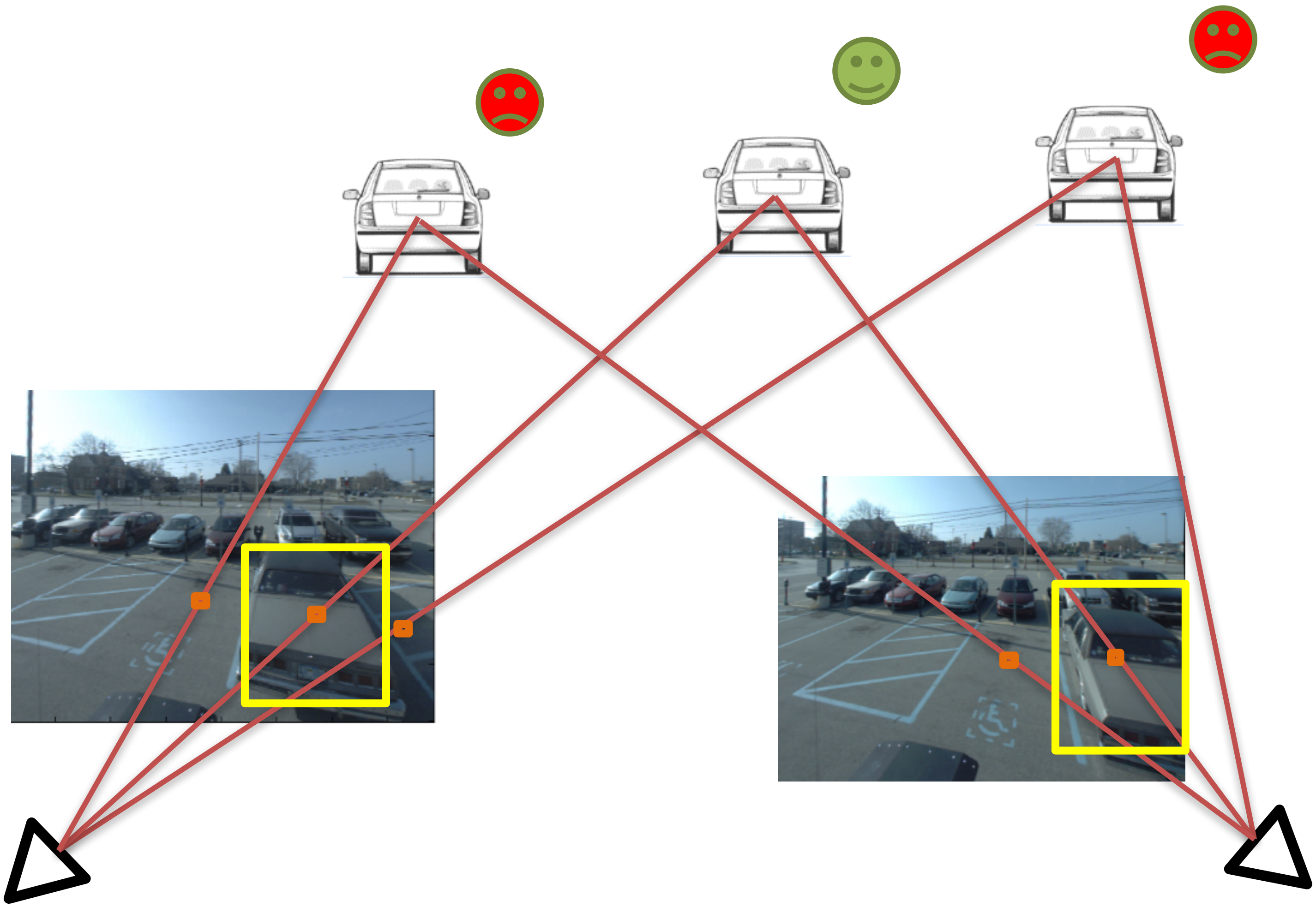


- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)
- Model Parameters:
  - $\mathbb{Q}$ = 3D points
  - $\mathbb{O}$ = 3D objects
  - $\mathbb{B}$ = 3D regions
  - $\mathbf{C}$ = cam. prm. K, R, T

Object "re-projection" error

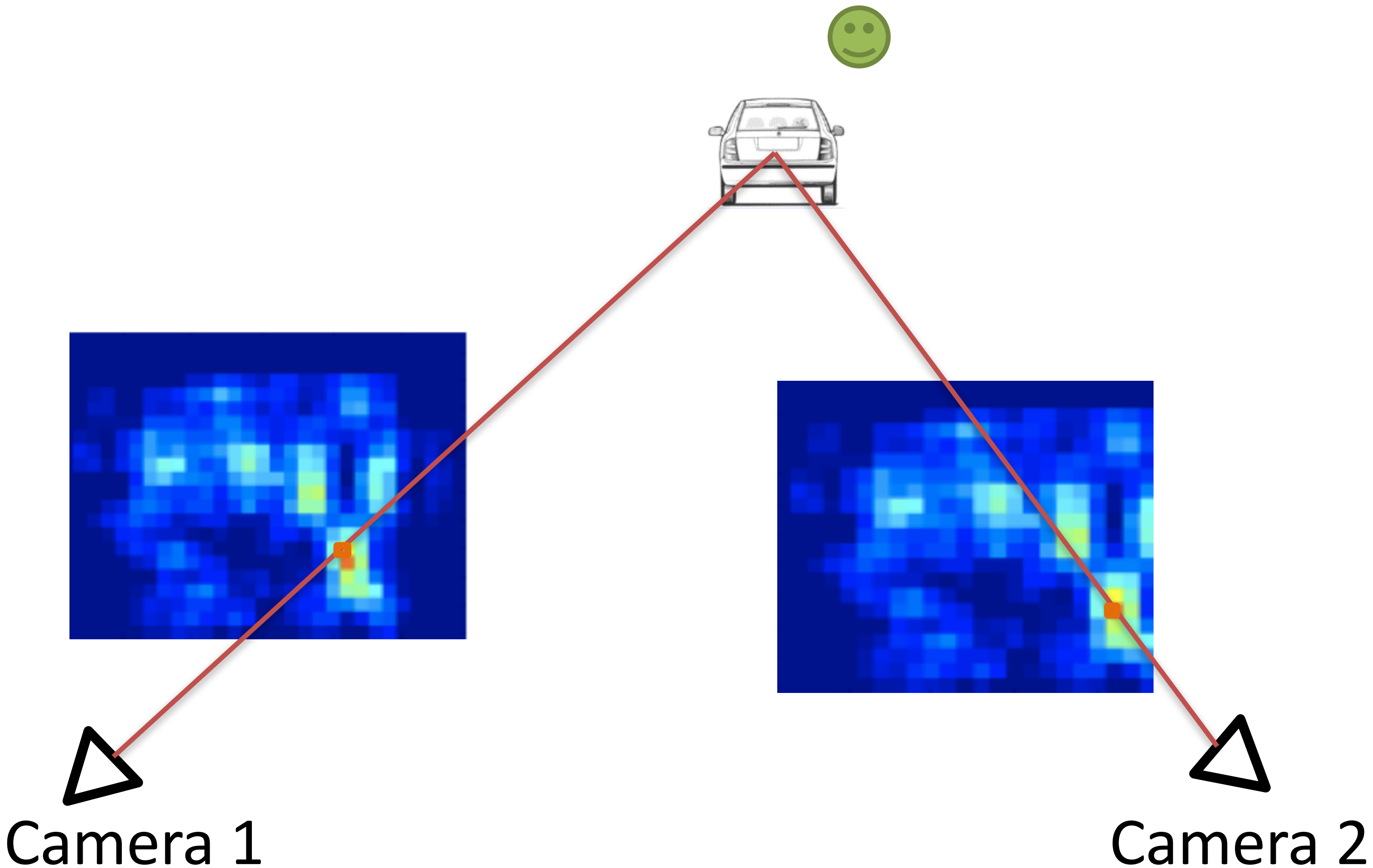
$$\Psi_t^{CO} \propto \prod_t^{N_t} (1 - \prod_k^{N_k} (1 - \Pr(o | O_t, C^k)))$$

SSFM: Object-level compatibility



- Agreement with measurements is computed using position, pose and scale

SSFM: Object-level compatibility

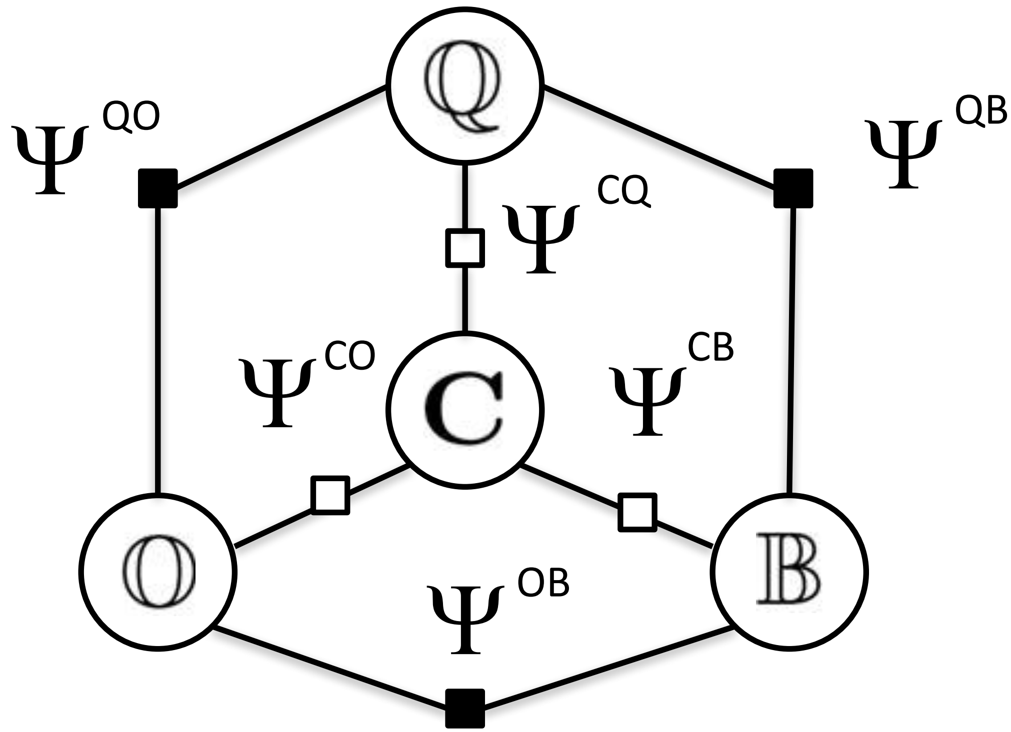


- Agreement with measurements is computed using position, pose and scale

SSFM with interactions

Bao, Bagra, Chao, Savarese
CVPR 2012

$$\{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}\} = \arg\max_{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}} \prod_s \Psi_s^{CQ} \prod_t \Psi_t^{CO} \prod_r \Psi_r^{CB} \prod_{t,s} \Psi_{t,s}^{OQ} \prod_{t,r} \Psi_{t,r}^{OB} \prod_{r,s} \Psi_{r,s}^{BQ}$$



- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)

- Model Parameters:
  - Q = 3D points
  - O = 3D objects
  - B = 3D regions
  - C = cam. prm. K, R, T

- Interactions of points, regions and objects across views
- Interactions among object-regions-points

SSFM with interactions

$$\{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}\} = \arg \max_{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}} \prod_s \Psi_s^{CQ} \prod_t \Psi_t^{CO} \prod_r \Psi_r^{CB} \prod_{t,s} \Psi_{t,s}^{OQ} \boxed{\prod_{t,r} \Psi_{t,r}^{OB}} \prod_{r,s} \Psi_{r,s}^{BQ}$$

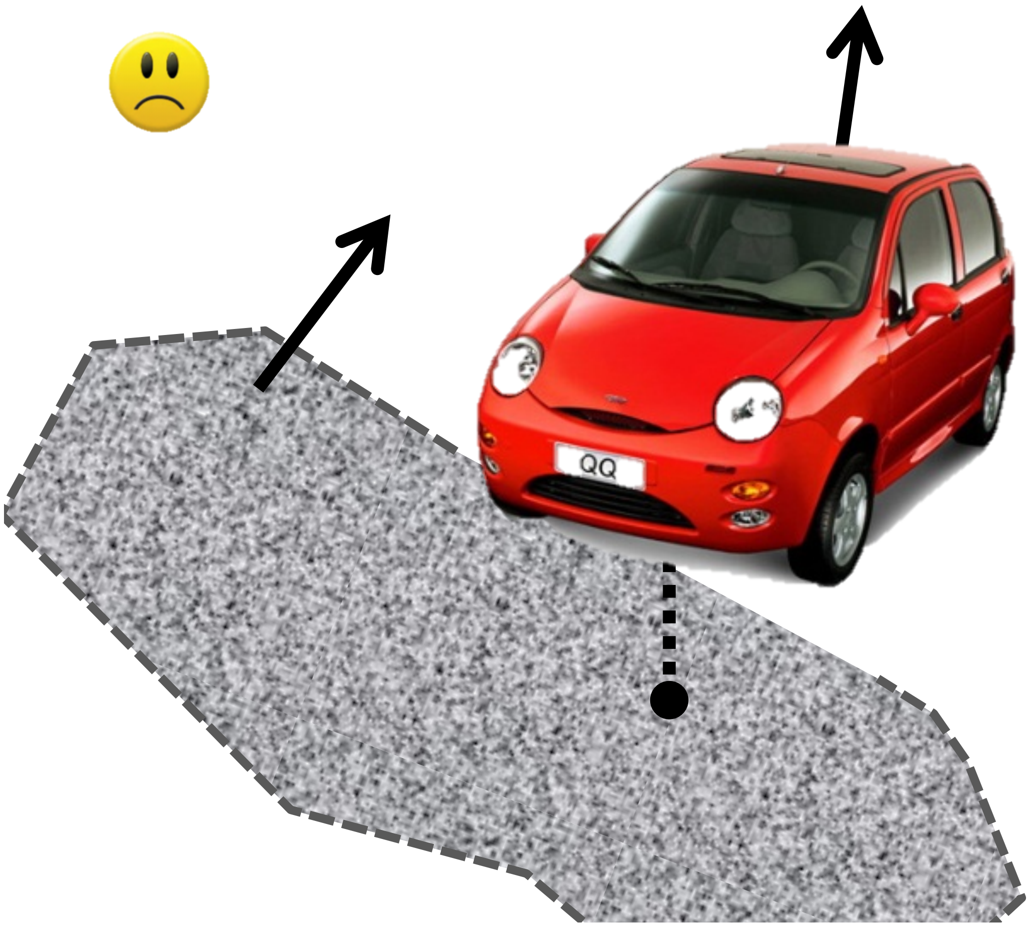
Object-Region Interactions:

- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)
- Model Parameters:
  - Q = 3D points
  - O = 3D objects
  - B = 3D regions
  - **C** = cam. prm. K, R, T

SSFM with interactions

$$\{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}\} = \arg\max_{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}} \prod_s \Psi_s^{CQ} \prod_t \Psi_t^{CO} \prod_r \Psi_r^{CB} \prod_{t,s} \Psi_{t,s}^{OQ} \boxed{\prod_{t,r} \Psi_{t,r}^{OB}} \prod_{r,s} \Psi_{r,s}^{BQ}$$

Object-Region Interactions:



- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)

- Model Parameters:
  - $\mathbb{Q}$ = 3D points
  - $\mathbb{O}$ = 3D objects
  - $\mathbb{B}$ = 3D regions
  - $\mathbf{C}$ = cam. prm. K, R, T

SSFM with interactions

$$\{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}\} = \arg \max_{\mathbb{Q},\mathbb{O},\mathbb{B},\mathbf{C}} \prod_{s} \Psi_{s}^{CQ} \prod_{t} \Psi_{t}^{CO} \prod_{r} \Psi_{r}^{CB} \boxed{\prod_{t,s} \Psi_{t,s}^{OQ}} \prod_{t,r} \Psi_{t,r}^{OB} \prod_{r,s} \Psi_{r,s}^{BQ}$$

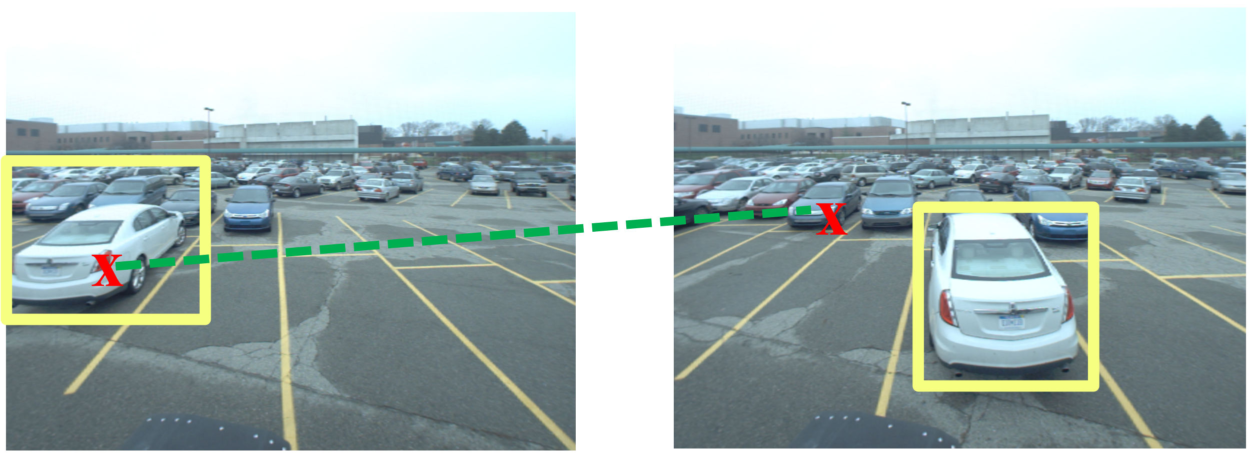
Object-point Interactions:

- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)
- Model Parameters:
  - Q = 3D points
  - O = 3D objects
  - B = 3D regions
  - $\mathbf{C}$ = cam. prm. K, R, T

SSFM with interactions

$$\{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}\} = \arg \max_{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}} \prod_{s} \Psi_{s}^{CQ} \prod_{t} \Psi_{t}^{CO} \prod_{r} \Psi_{r}^{CB} \boxed{\prod_{t,s} \Psi_{t,s}^{OQ}} \prod_{t,r} \Psi_{t,r}^{OB} \prod_{r,s} \Psi_{r,s}^{BQ}$$

Object-point Interactions:



- Measurements I
  - Points (x,y,scale)
  - Objects (x,y, scale, pose)
  - Regions (x,y, pose)

- Model Parameters:
  - Q = 3D points
  - O = 3D objects
  - B = 3D regions
  - **C** = cam. prm. K, R, T

Solving the SSFM problem

$$\{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}\} = \arg \max_{\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}} \Psi(\mathbb{Q}, \mathbb{O}, \mathbb{B}, \mathbf{C}; \mathbf{I})$$

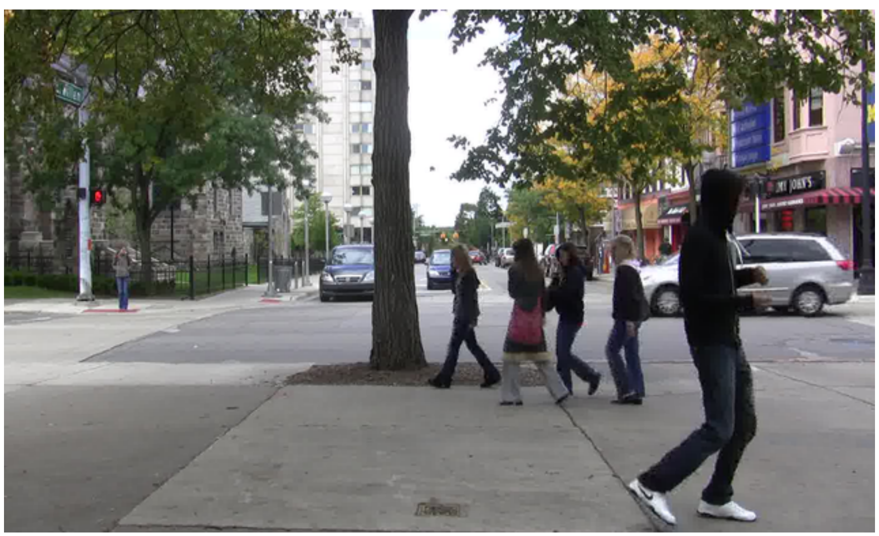
- Modified Reversible Jump Markov Chain Monte Carlo (RJ-MCMC) sampling algorithm [Dellaert et al., 2000]
- Initialization of the cameras, objects, and points are critical for the sampling
- Initialize configuration of cameras using:
    - SFM
    - consistency of object/region properties across views

Results

FORD CAMPUS dataset [Pandey et al., 09]

Input images

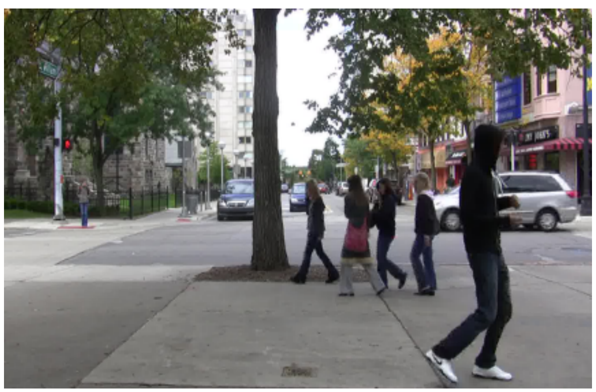
⋮

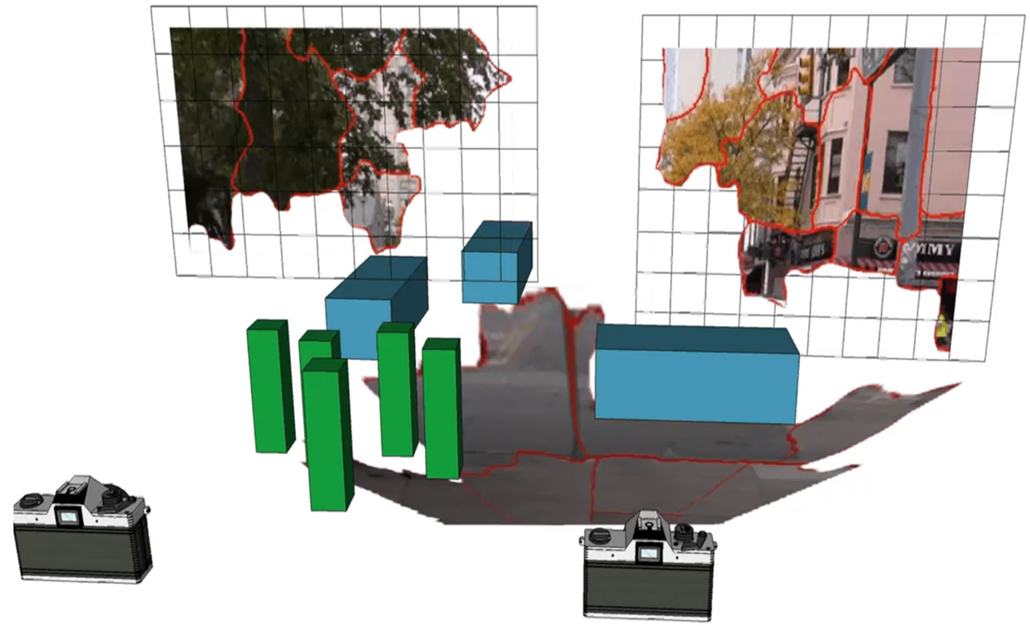


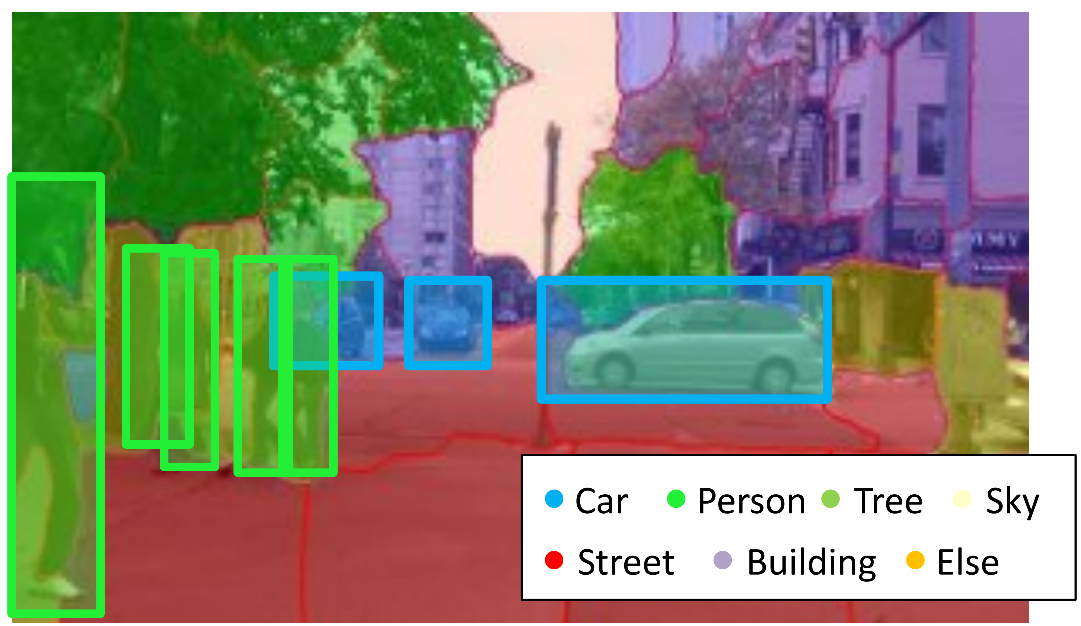
- Wide baseline
- Background clutter
- Limited visibility
- Un-calibrated cameras

Results

Input images

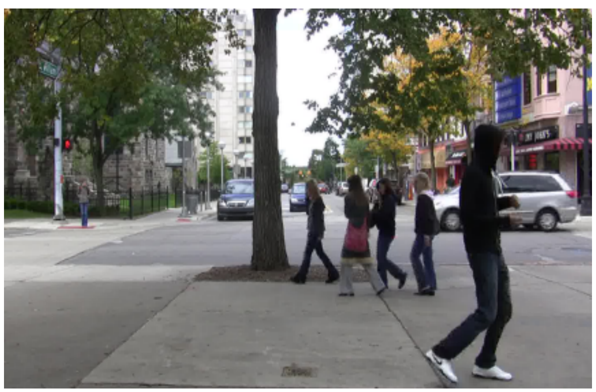


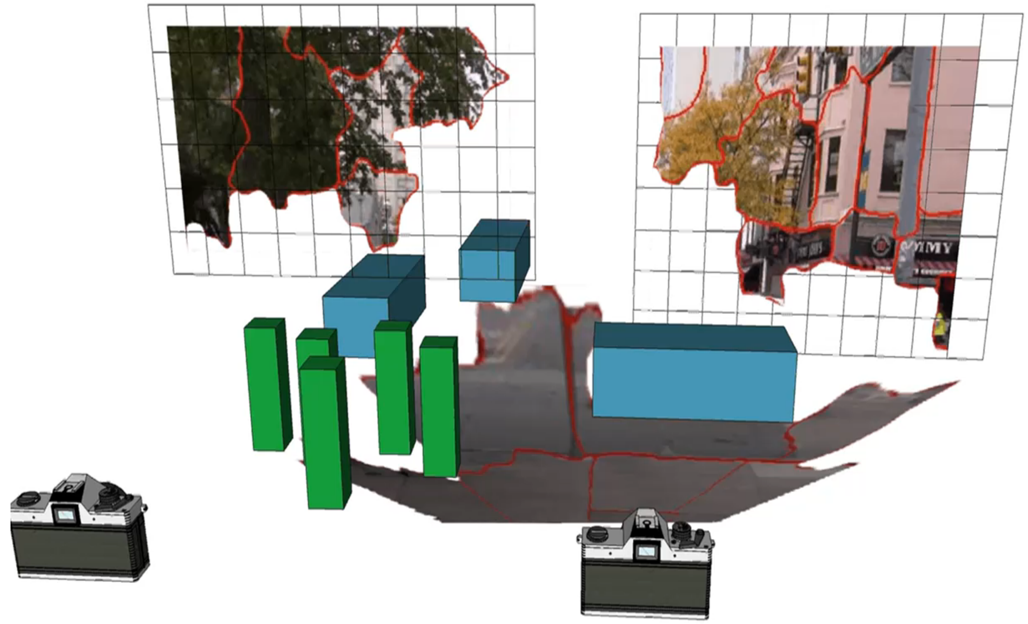


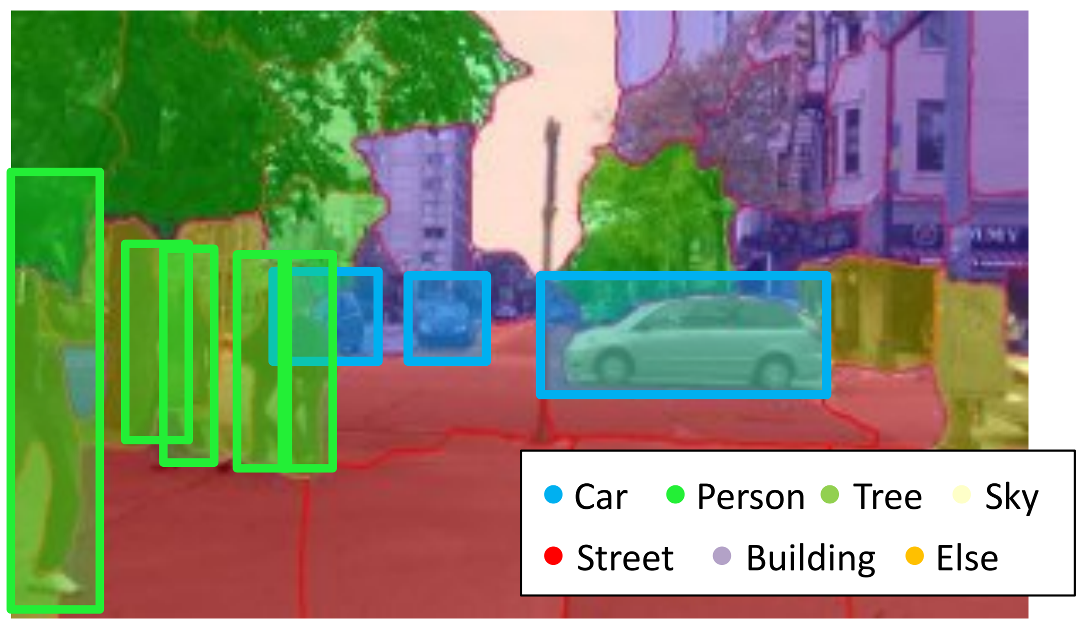


Results

Input images



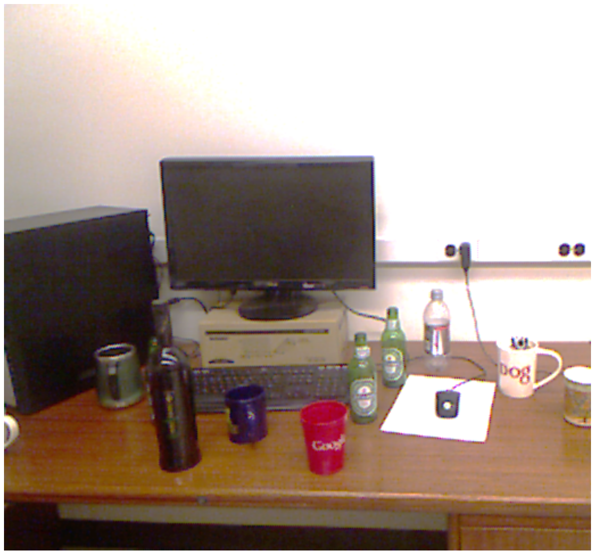


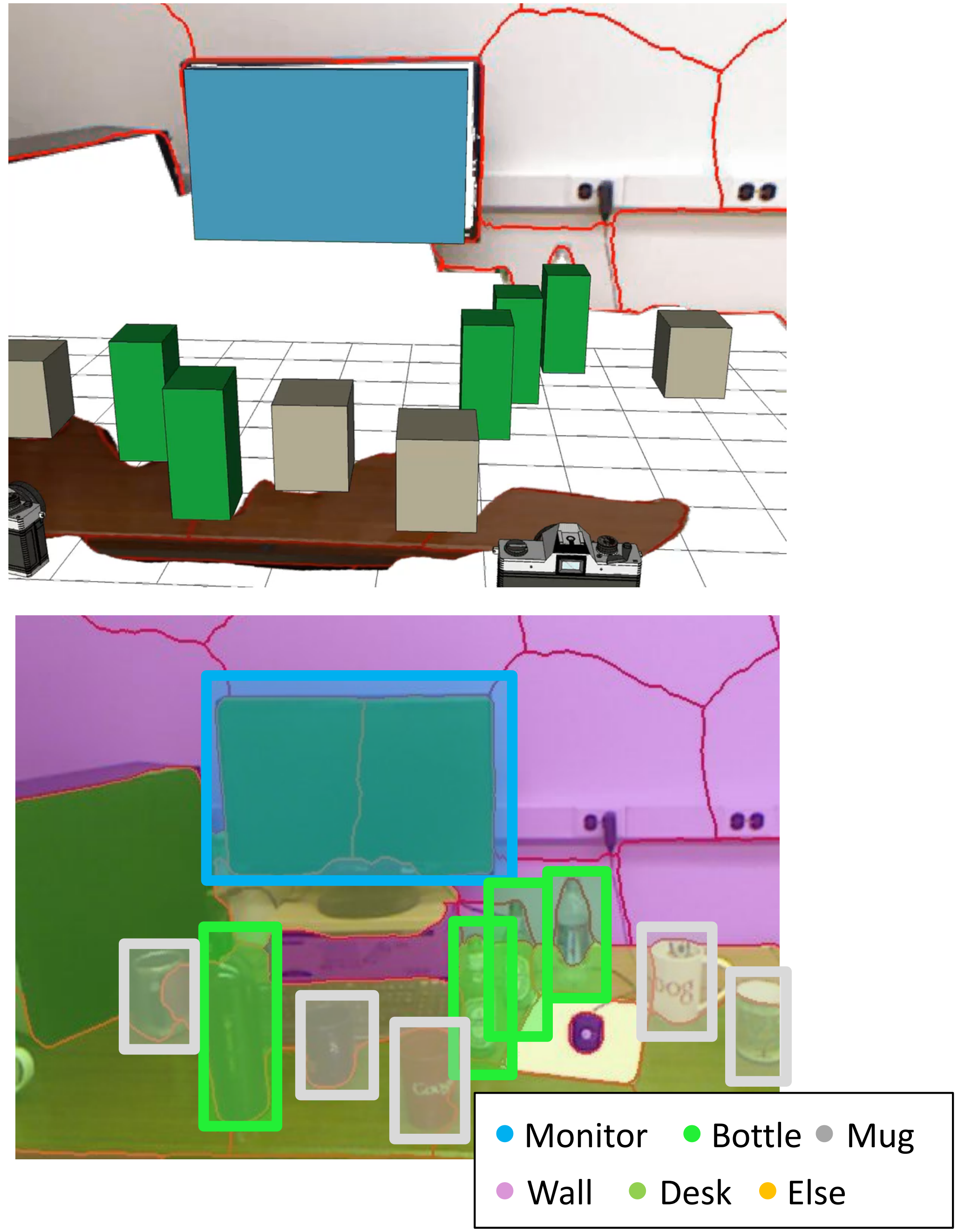


Results

Input images

From the office dataset [Bao et al., 11]

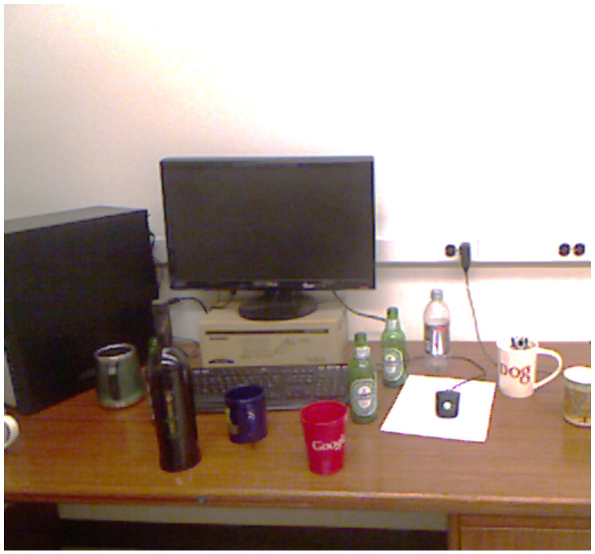


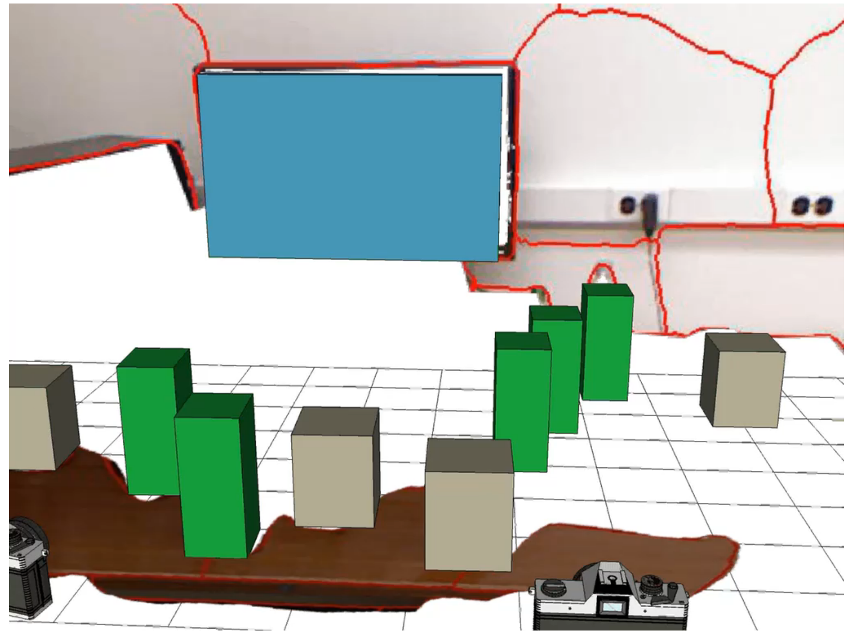


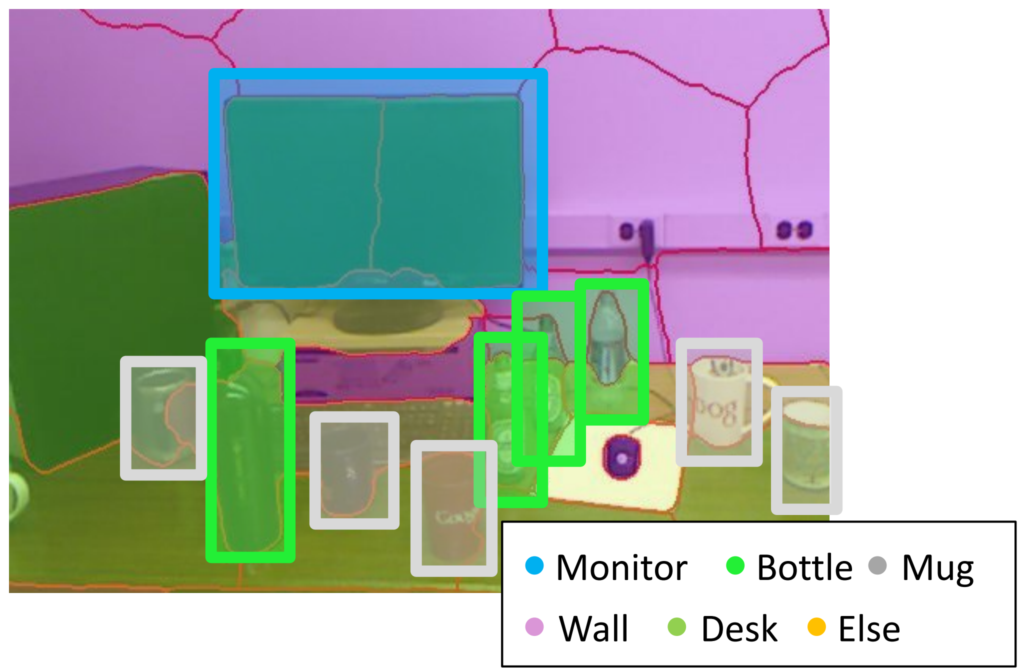
Results

Input images

From the office dataset [Bao et al., 11]



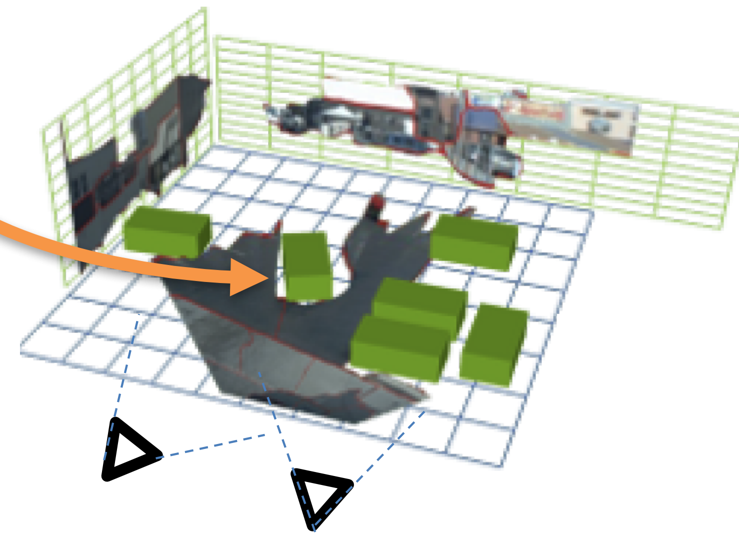




Results

Average precision in localizing objects in the 3D space

	Hoiem et al. 2011	SSFM no int.	SSFM
FORD CAMPUS	21.4%	32.7%	**43.1%**
OFFICE	15.5%	20.2%	**21.6%**



Average precision in detecting objects in the 2D image

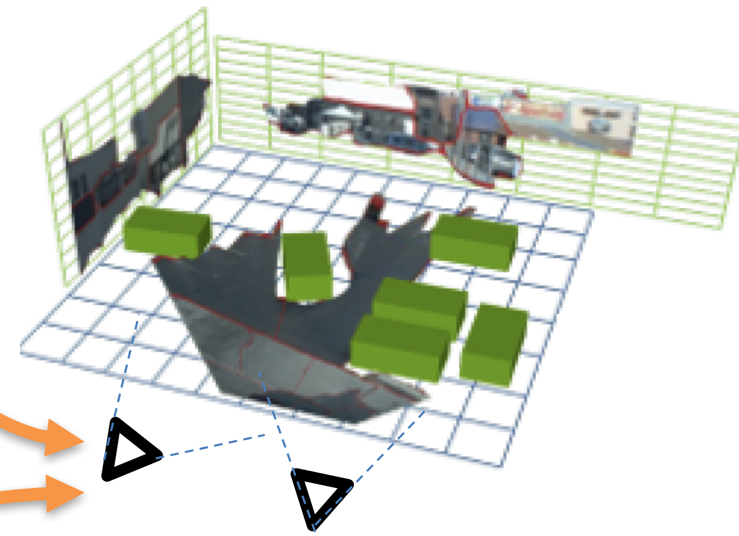
DPM [1]	SSFM 2 views no int.	SSFM 2 views	SSFM 4 views
54.5%	61.3%	62.8%	**66.5%**

FORD CAMPUS dataset [Pandey et al., 09]
Office dataset [Bao et al., 11]

[1] Felzenszwalb et al. 2008

Results

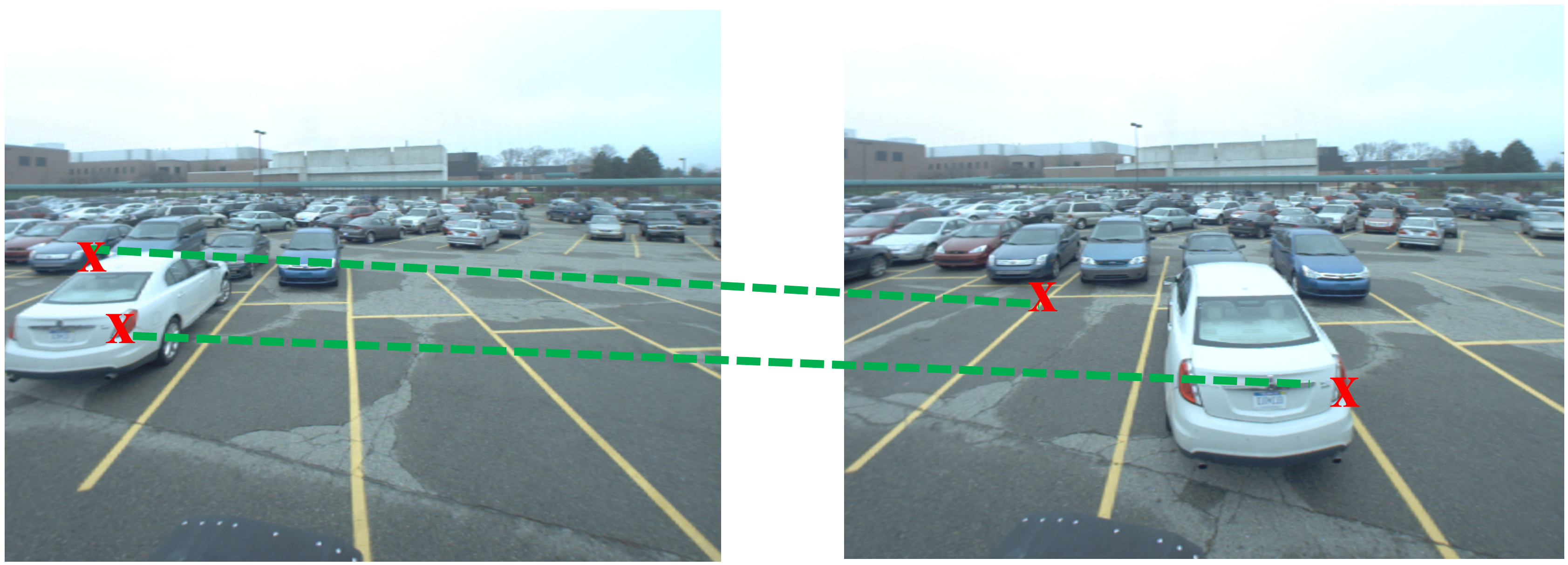
	Camera translation error		
	SFM Snavely et al., 08	**SSFM** no int.	**SSFM**
FORD CAMPUS	26.5°	19.9°	**12.1°**
OFFICE	8.5°	4.7°	**4.2°**
STREET	27.1°	17.6°	**11.4°**

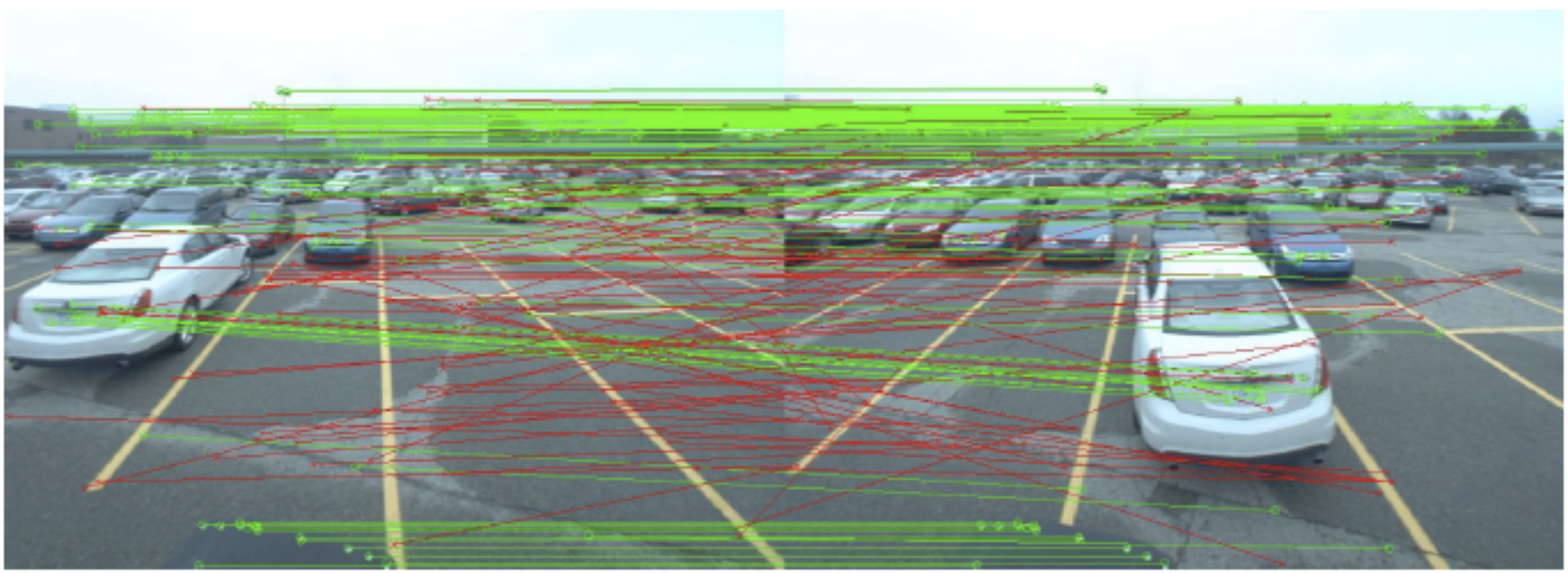


Camera rotation error		
SFM Snavely et al., 08	**SSFM** no int.	**SSFM**
<1°	<1°	**<1**
9.6°	4.2°	**3.5°**
21.1°	3.1°	**3.0°**

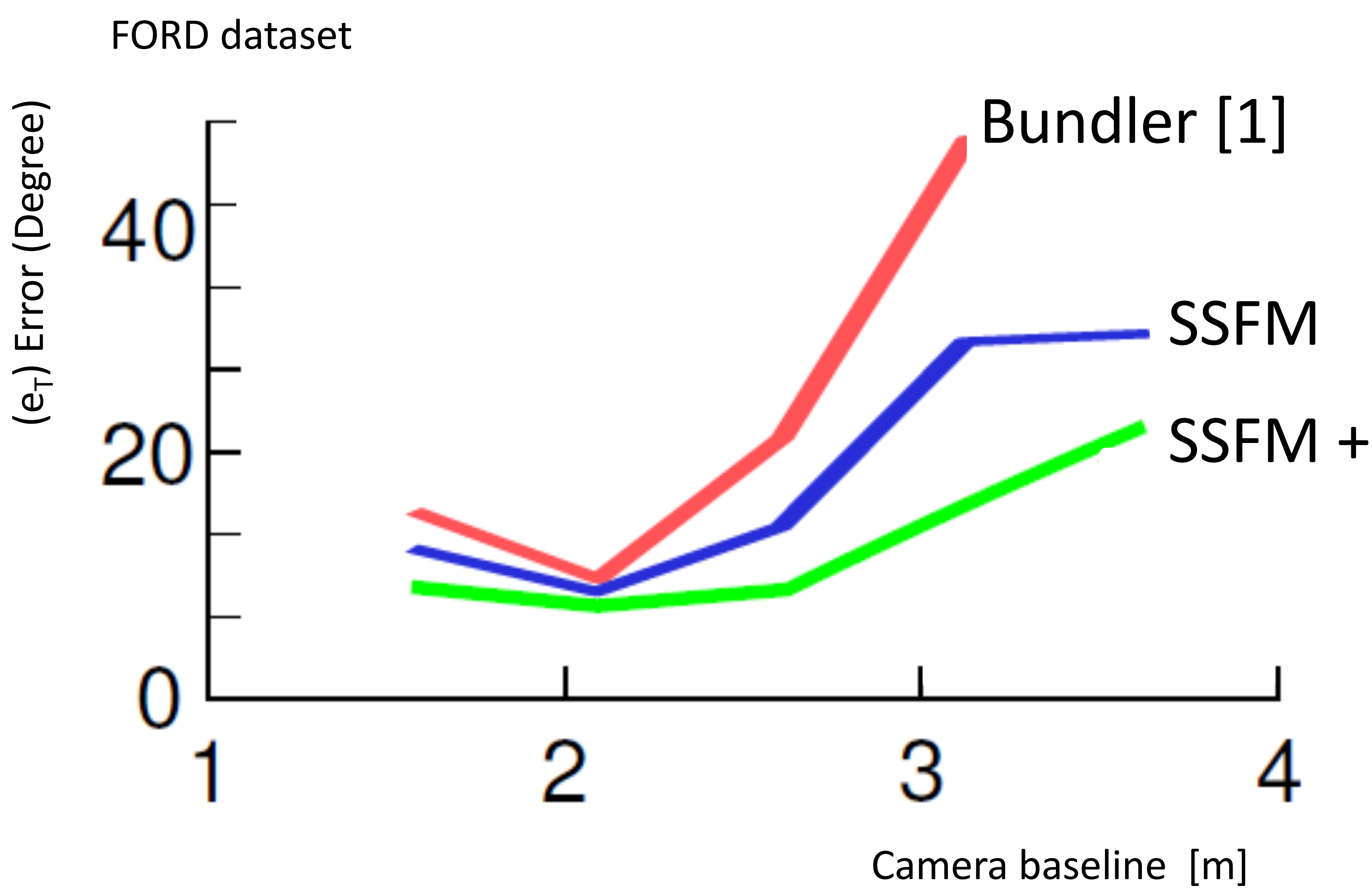
FORD CAMPUS dataset [Pandey et al., 09]
Office dataset [Bao et al., 11]
Street dataset [Bao et al., 11]

Wide-baseline feature correspondence

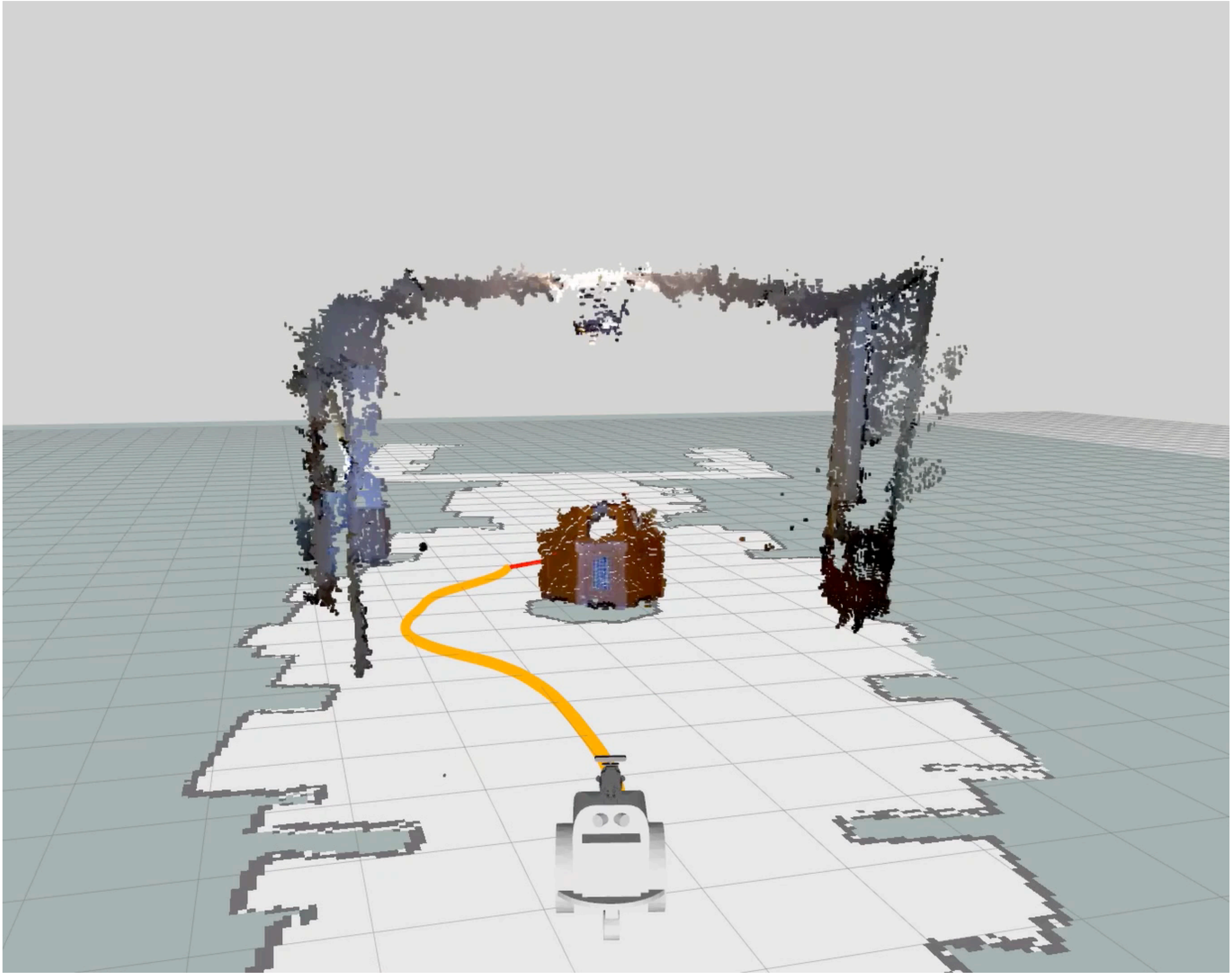




Camera Pose Estimation v.s. Base Line Width



SSFM Source code available!
Please visit: http://www.eecs.umich.edu/vision/research.html



3D reconstruction from images

- The SFM problem
- Affine SFM
- Perspective SFM
- Bundle Adjustment

3D Scene Understanding

- Motivation
- Single view 3D scene understanding
- Multi-views 3D scene understanding