ECE 470: Introduction to Robotics
Lecture 28

Liangjing Yang

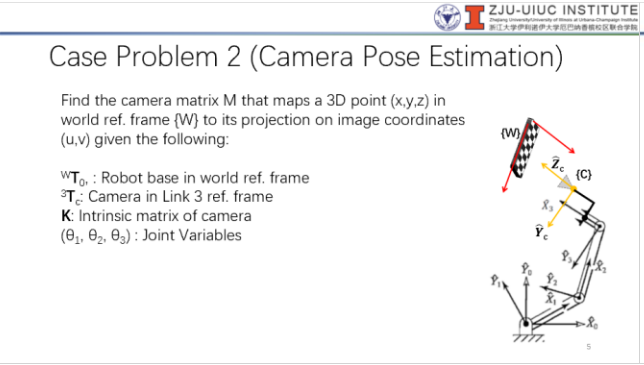
Assistant Professor, ZJU-UIUC Institute

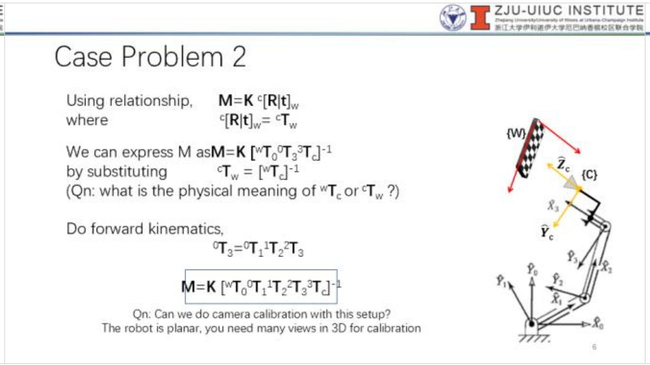
liangjingyang@intl.zju.edu.cn

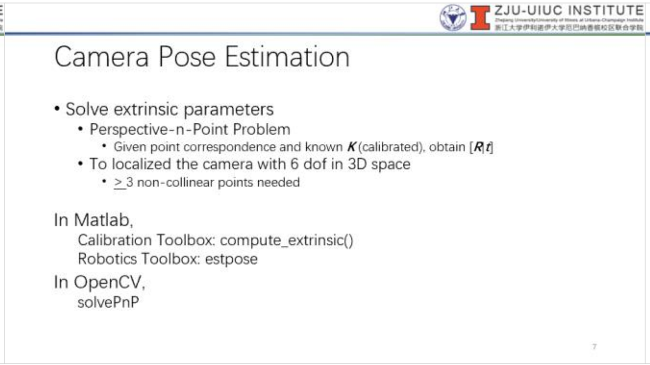
Wechat ID: Liangjing_Yang

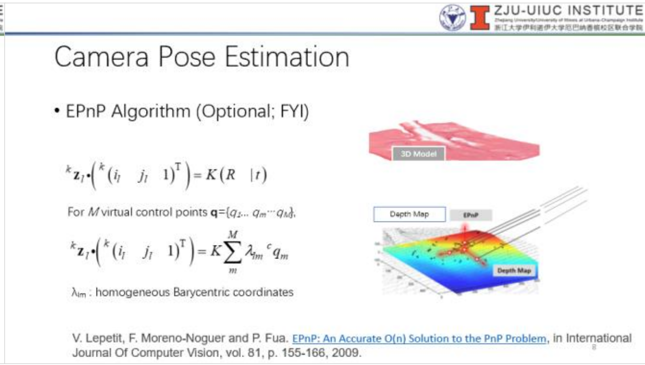
We discussed camera model and calibration last lecture……

After calibration, how do we estimate the pose of the camera? Or the pose of our robot?





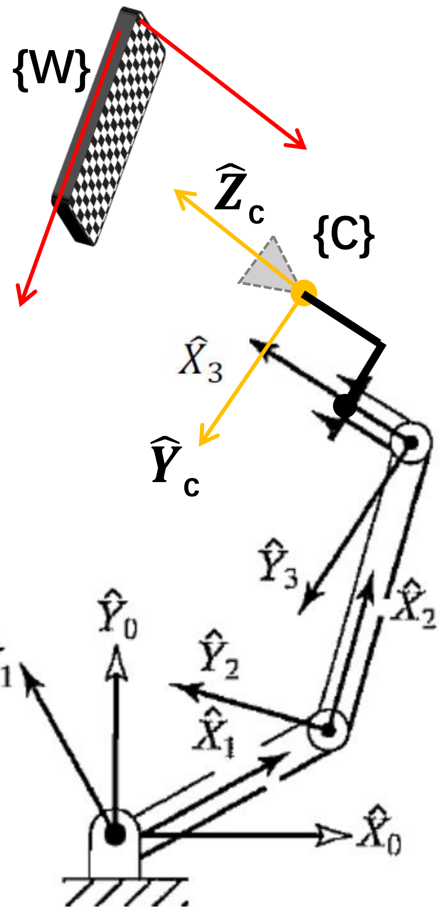




Case Problem 2 (Camera Pose Estimation)

Find the camera matrix M that maps a 3D point (x,y,z) in world ref. frame {W} to its projection on image coordinates (u,v) given the following:

${}^{W}\mathbf{T}_{0}$, : Robot base in world ref. frame
${}^{3}\mathbf{T}_{c}$: Camera in Link 3 ref. frame
K: Intrinsic matrix of camera
$(\theta_1, \theta_2, \theta_3)$: Joint Variables



Case Problem 2

Using relationship, $\mathbf{M}=\mathbf{K}\ {}^{c}[\mathbf{R}|\mathbf{t}]_{w}$
where ${}^{c}[\mathbf{R}|\mathbf{t}]_{w}= {}^{c}\mathbf{T}_{w}$

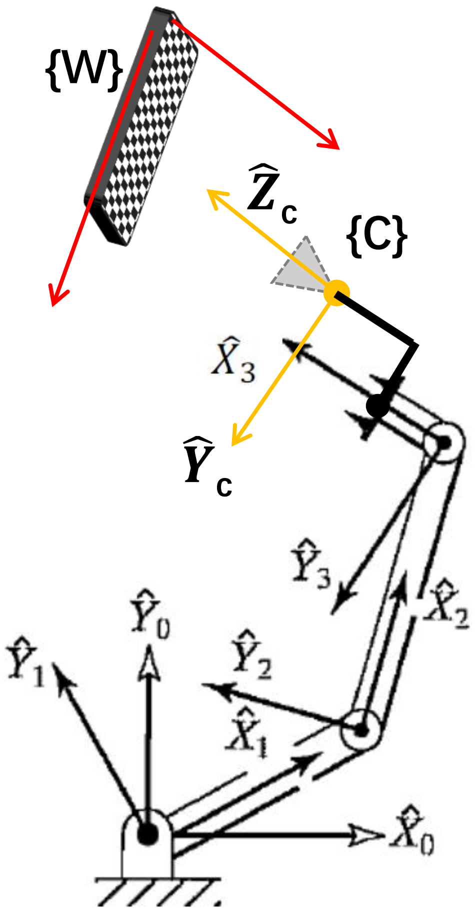
We can express M as $\mathbf{M}=\mathbf{K}\ [{}^{w}\mathbf{T}_{0}{}^{0}\mathbf{T}_{3}{}^{3}\mathbf{T}_{c}]^{-1}$
by substituting ${}^{c}\mathbf{T}_{w} = [{}^{w}\mathbf{T}_{c}]^{-1}$
(Qn: what is the physical meaning of ${}^{w}\mathbf{T}_{c}$ or ${}^{c}\mathbf{T}_{w}$?)

Do forward kinematics,

$${}^{0}\mathbf{T}_{3}={}^{0}\mathbf{T}_{1}{}^{1}\mathbf{T}_{2}{}^{2}\mathbf{T}_{3}$$

$$\boxed{\mathbf{M}=\mathbf{K}\ [{}^{w}\mathbf{T}_{0}{}^{0}\mathbf{T}_{1}{}^{1}\mathbf{T}_{2}{}^{2}\mathbf{T}_{3}{}^{3}\mathbf{T}_{c}]^{-1}}$$

Qn: Can we do camera calibration with this setup?
The robot is planar, you need many views in 3D for calibration



Camera Pose Estimation

- Solve extrinsic parameters
  - Perspective-n-Point Problem
    - Given point correspondence and known $\boldsymbol{K}$ (calibrated), obtain $[\boldsymbol{R}|\boldsymbol{t}]$
  - To localized the camera with 6 dof in 3D space
    - $\geq$ 3 non-collinear points needed

In Matlab,

Calibration Toolbox: compute_extrinsic()

Robotics Toolbox: estpose

In OpenCV,

solvePnP

Camera Pose Estimation

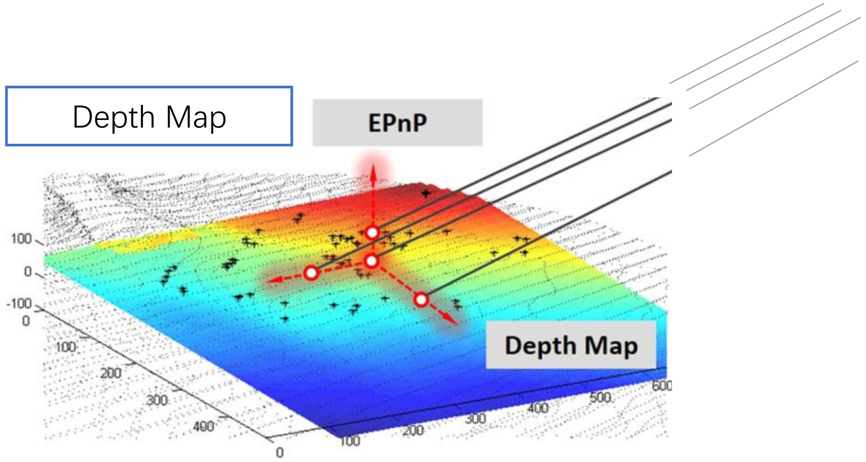
- EPnP Algorithm (Optional; FYI)

$${}^{k}\mathbf{z}_l \bullet \left({}^{k}\left(i_l \quad j_l \quad 1\right)^{\mathrm{T}} \right) = K\left(R \quad | \, t\right)$$

For M virtual control points $\mathbf{q}=\{q_1 \ldots q_m \cdots q_M\}$,

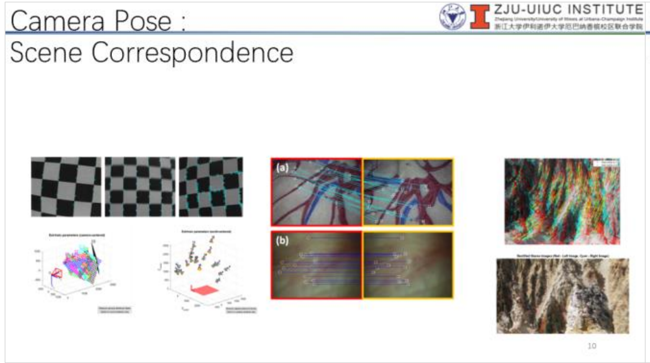
$${}^{k}\mathbf{z}_l \bullet \left({}^{k}\left(i_l \quad j_l \quad 1\right)^{\mathrm{T}} \right) = K \sum_{m}^{M} \lambda_{lm} \, {}^{c}q_m$$

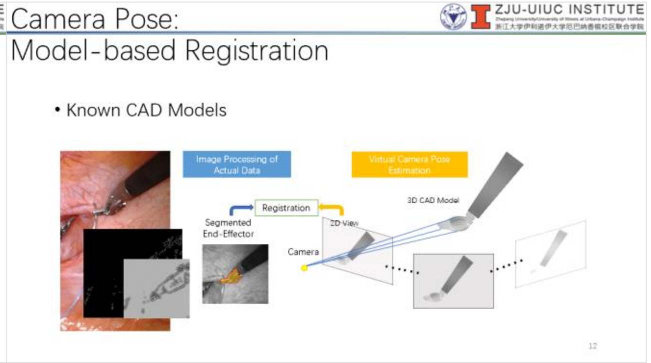
λ_{lm} : homogeneous Barycentric coordinates

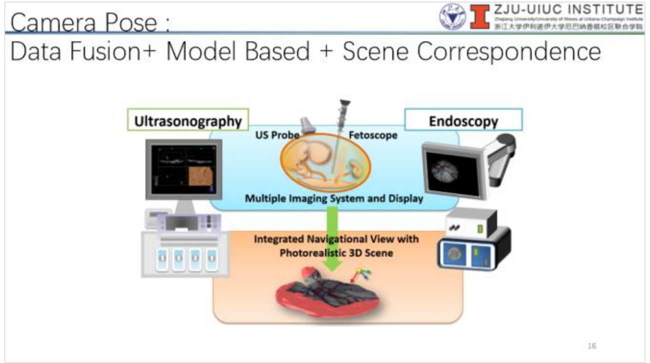


V. Lepetit, F. Moreno-Noguer and P. Fua. [EPnP: An Accurate O(n) Solution to the PnP Problem](), in International Journal Of Computer Vision, vol. 81, p. 155-166, 2009.

After calibration, how do we estimate the pose of the camera?
Or the pose of our robot?
How about spatial representation of the surrounding?

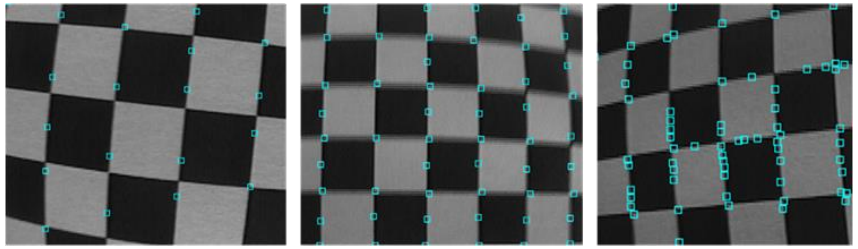


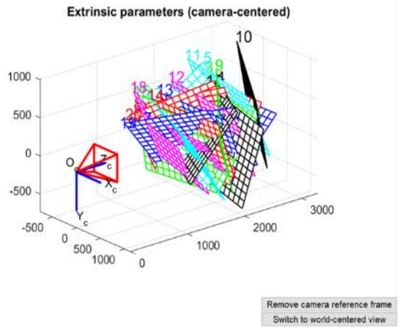


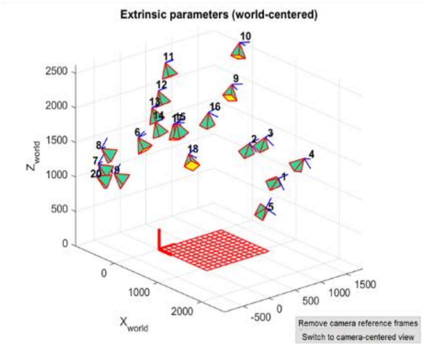


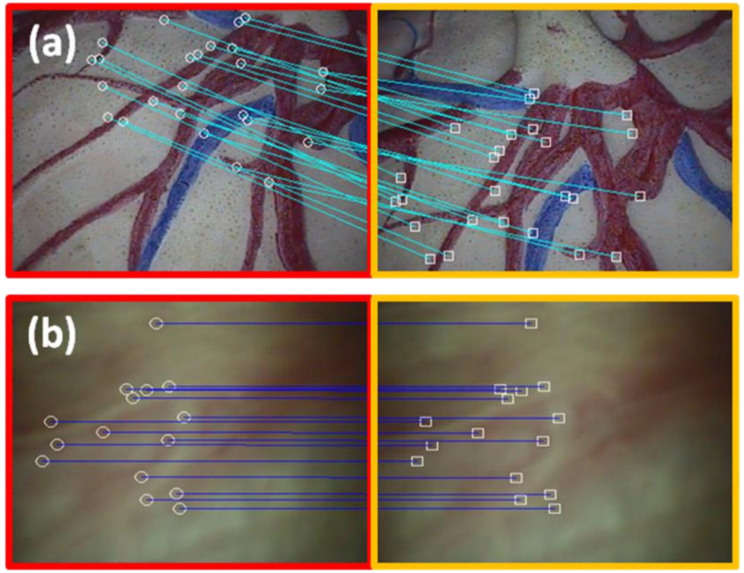
Camera Pose :

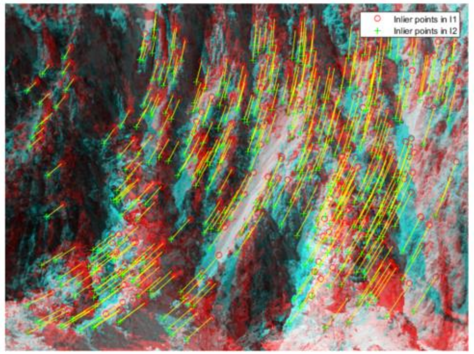
Scene Correspondence



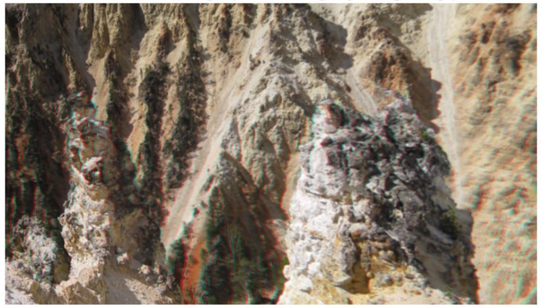






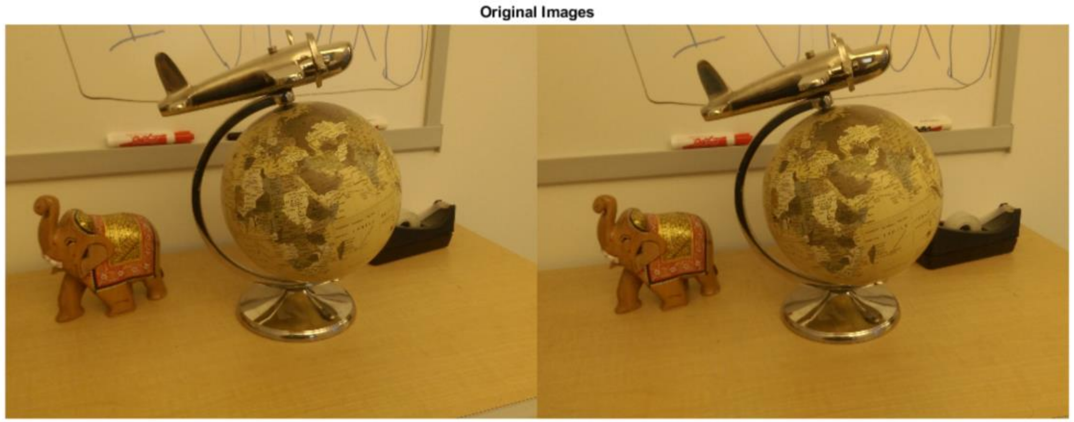


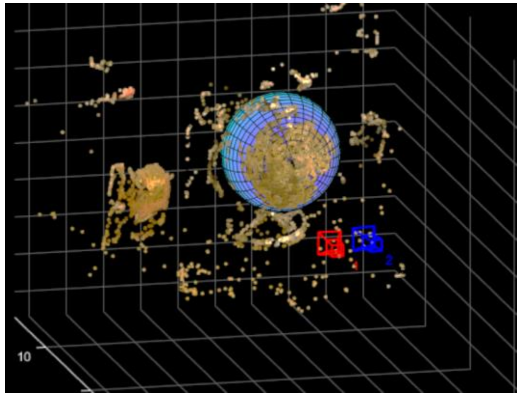
Rectified Stereo Images (Red - Left Image, Cyan - Right Image)



Camera Pose: Model-based Registration

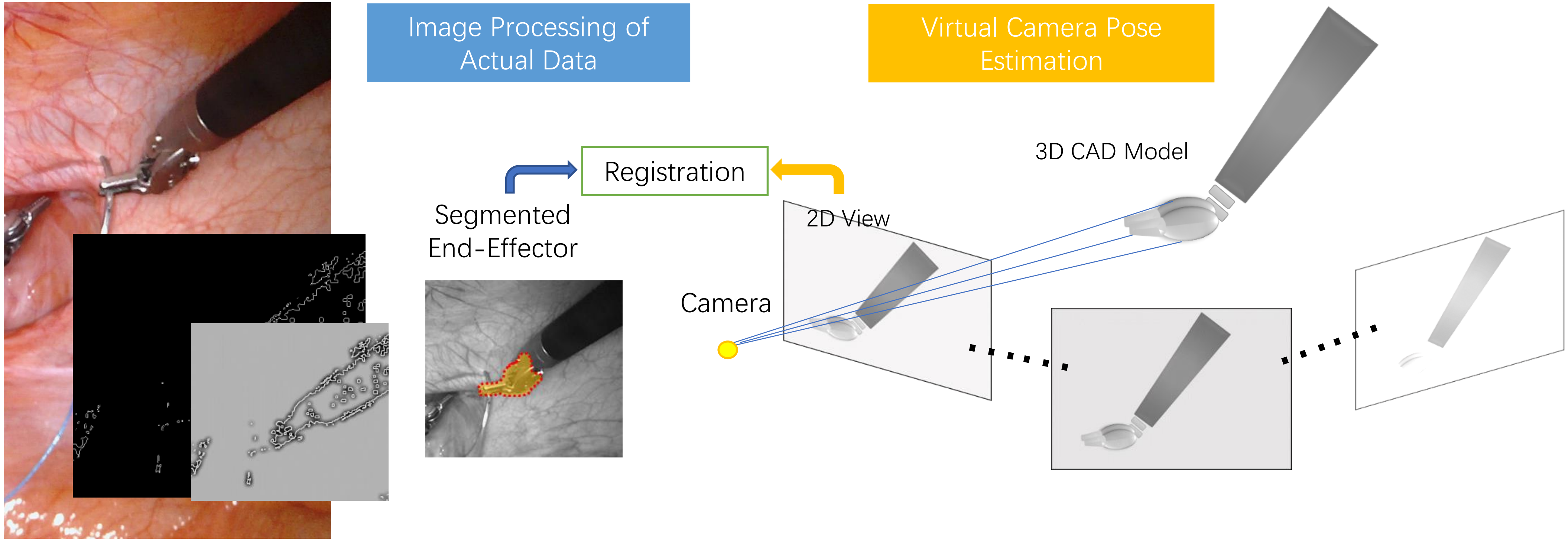
- Solve extrinsic parameters with known models in scene
  - Primitives: Ellipsoid, Cuboid etc.





Camera Pose: Model-based Registration

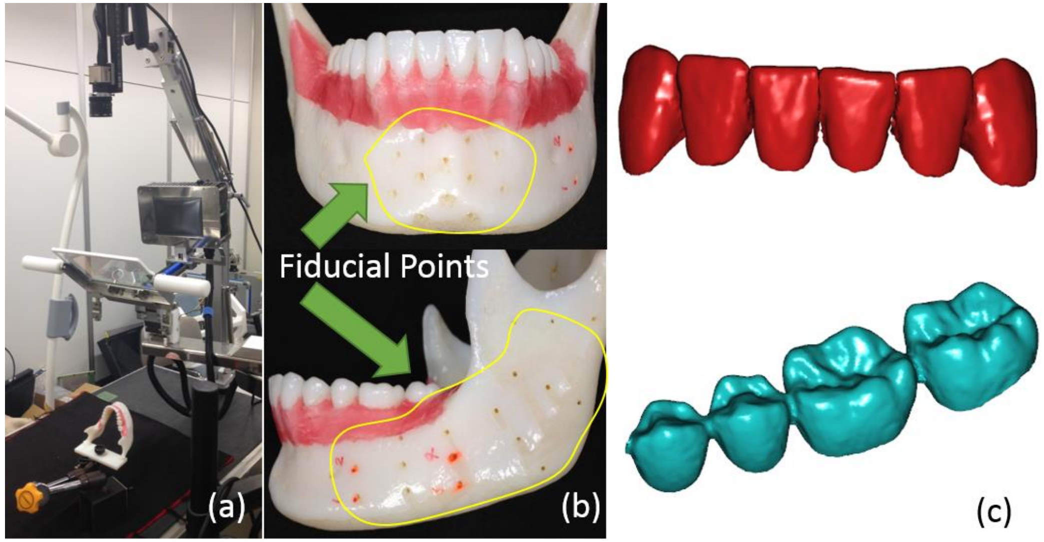
- Known CAD Models



Camera Pose:

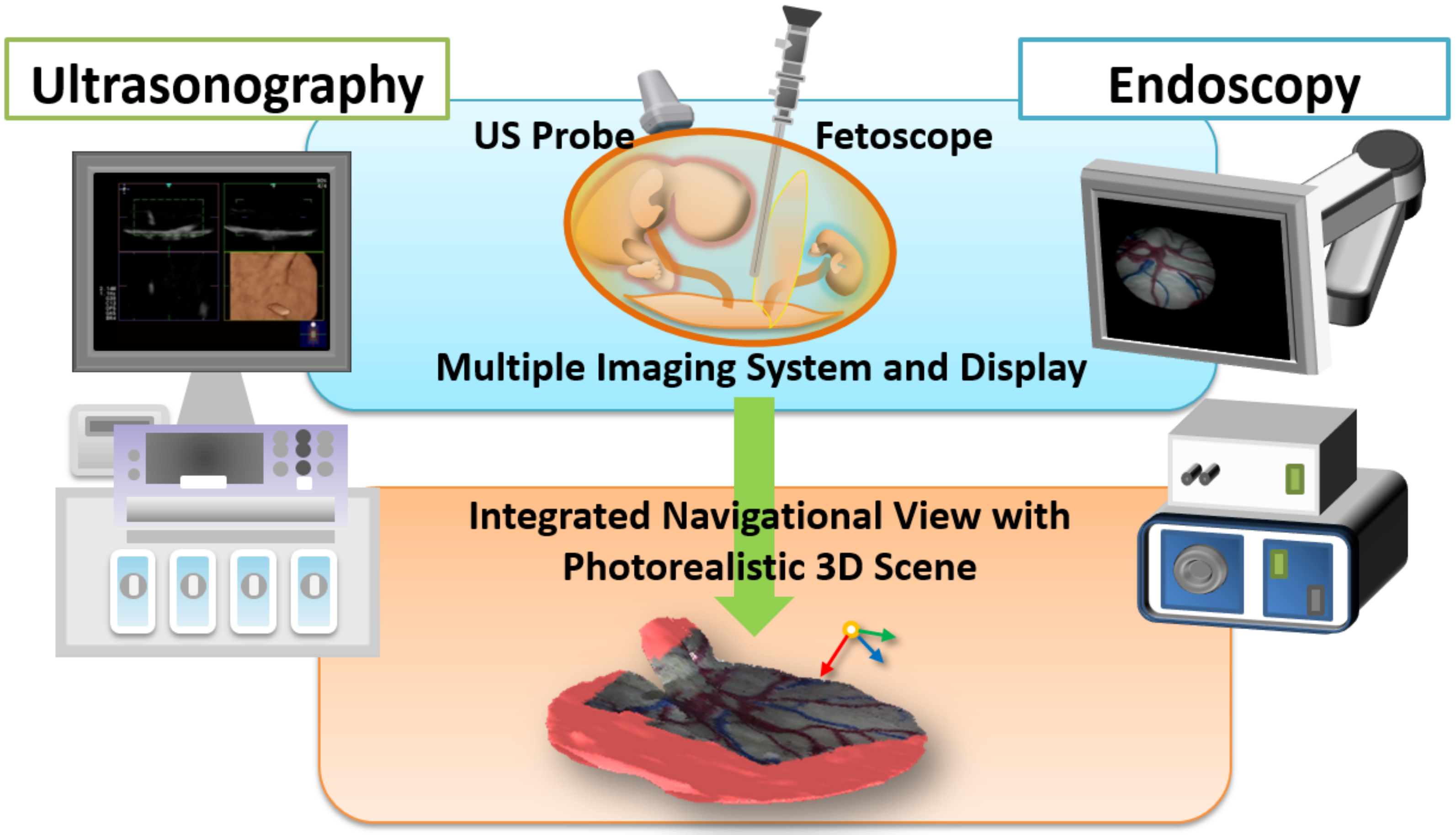
Model-based Registration

- Pre-scanned model



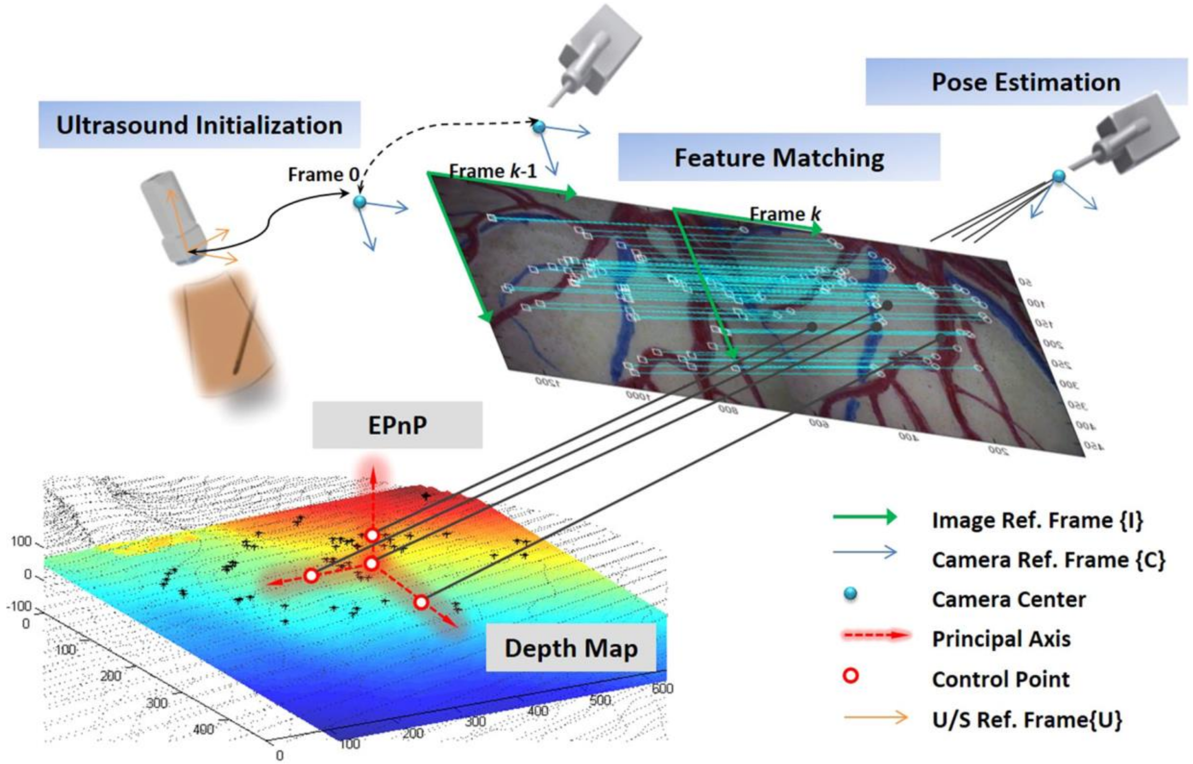
Camera Pose :

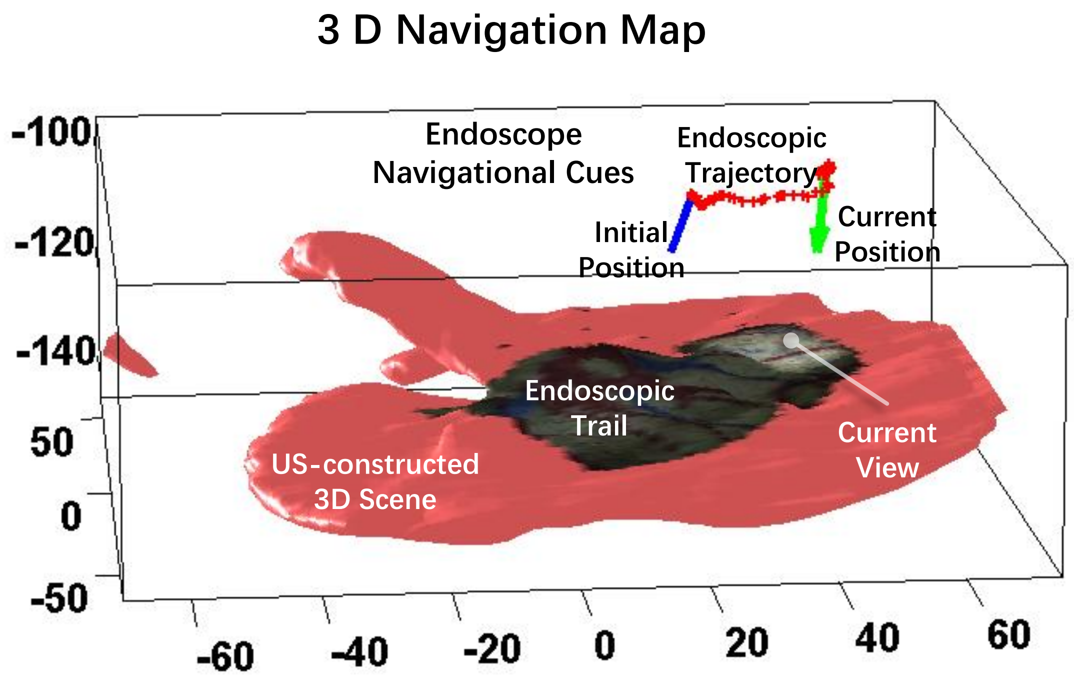
Data Fusion+ Model Based + Scene Correspondence



Camera Pose :

Data Fusion+ Model Based + Scene Correspondence



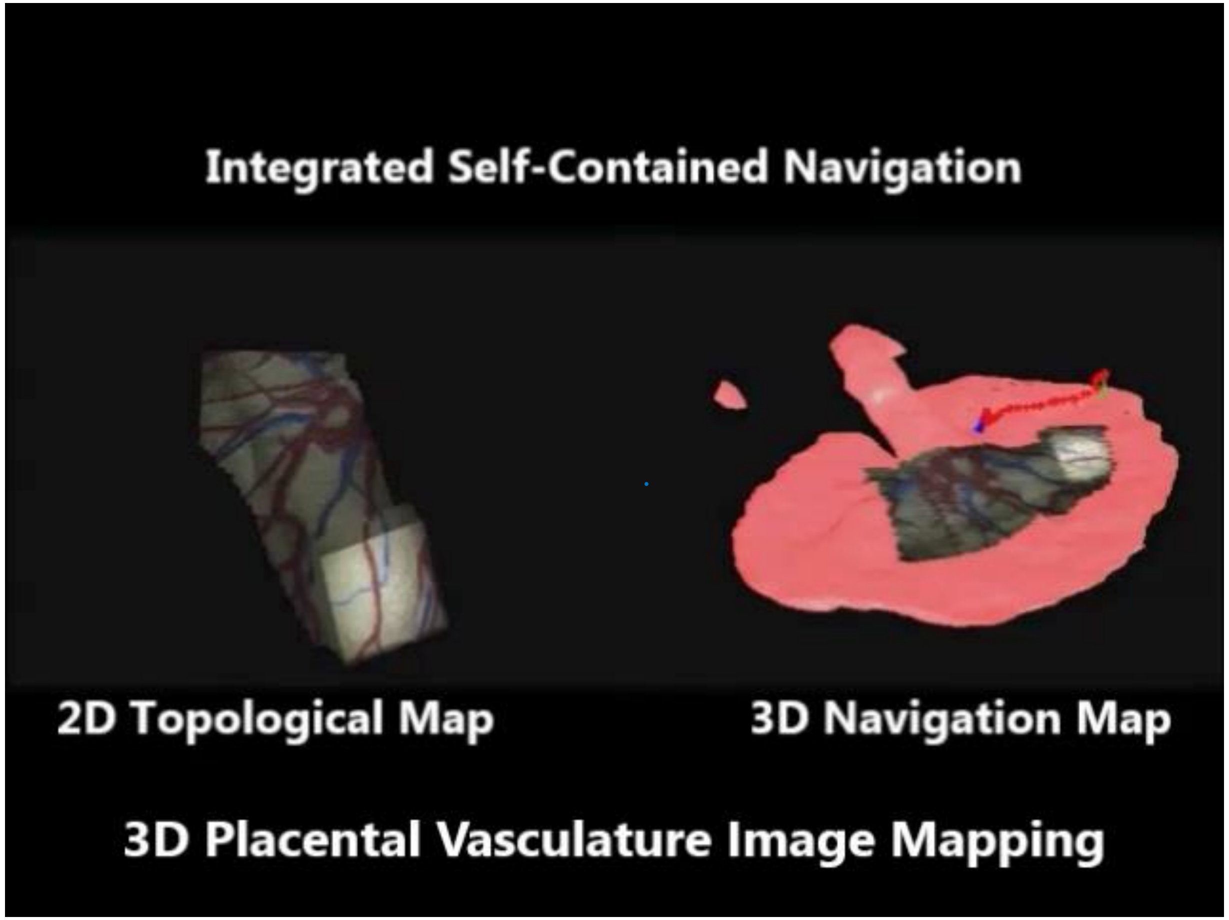


"Self-contained image mapping of placental vasculature in 3D ultrasound-guided fetoscopy", Yang et al., 2015

Camera Pose :

ZJU-UIUC INSTITUTE
Zhejiang University/University of Illinois at Urbana-Champaign Institute
浙江大学伊利诺伊大学厄巴纳香槟校区联合学院

Data Fusion+ Model Based + Scene Correspondence



After calibration, how do we estimate the pose of the camera?
Or the pose of our robot?
How about spatial representation of the surrounding?

Describing the pose of camera is mathematically equivalent to

describing the surrounding spatial information

<u>A Problem of Spatial Representation and 3D Vision</u>

3D Vision

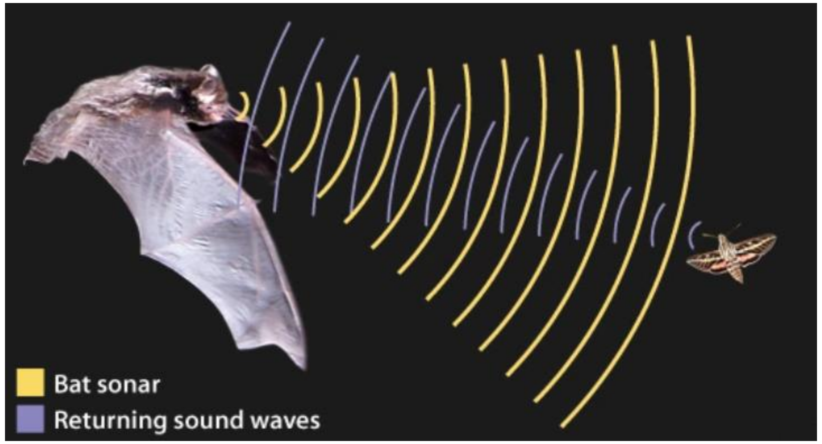
ECE 470 Introduction to Robotics

3D Vision: Depth Perception

- Many species are capable of perceiving distance

Stereopsis

https://unsplash.com/photos/OjQgsR1oyEw



Echolocation

https://neuwritesd.org/2016/05/05/hello-darkness-my-old-friend-how-echolocation-lets-bats-rule-the-night/

Defocusing

https://webvision.med.utah.edu/2012/01/jumping-spiders-use-image-defocusing-for-depth-perception/

3D Vision: Depth Perception

- Understanding 3D Vision allows us to equip our robots with 3D perception

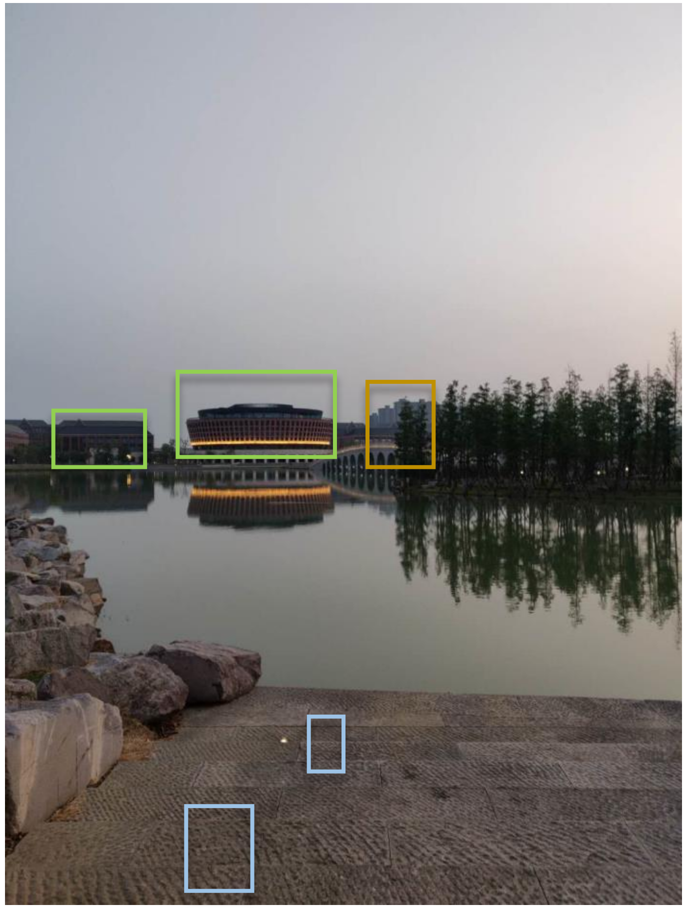
How do we, human, perceive the three-dimensional world?

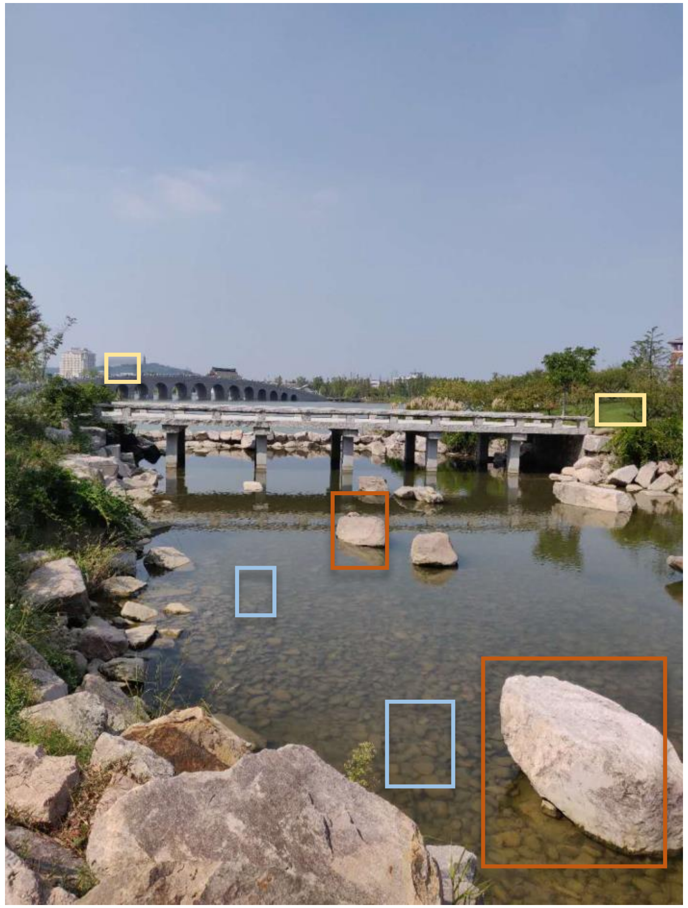
Visual Cues for Depth Perception

- Occlusion
- Relative size
- Texture Density
- Shading
- Binocular Disparity
- Motion perspective

Demo of 3D Visual Cue

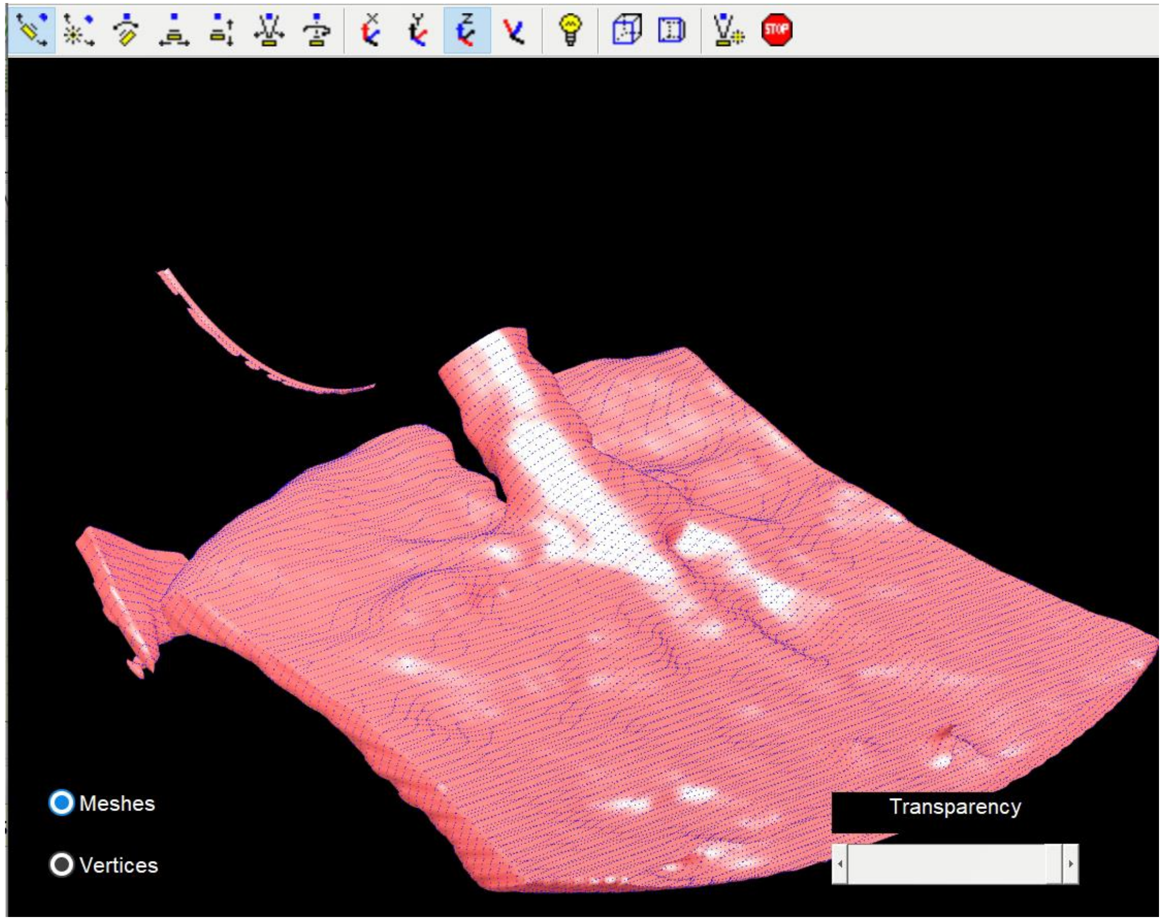
- Occlusion
- Relative size
- Texture Density
- Shading
- Binocular Disparity
- Motion perspective





Demo of 3D Visual Cue

- Occlusion
- Relative size
- Texture Density
- Shading
- Binocular Disparity
- Motion perspective



3D and Depth Recovery

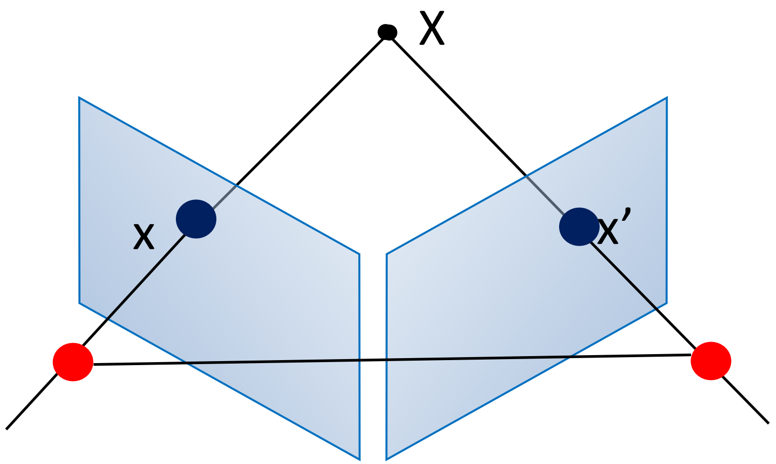
- Stereo-Triangulation
- Structure-from-Motion
- Structure-from-Shading
- Active Depth Sensing
- Defocusing
- ……

Stereo-Vision

ECE 470 Introduction to Robotics

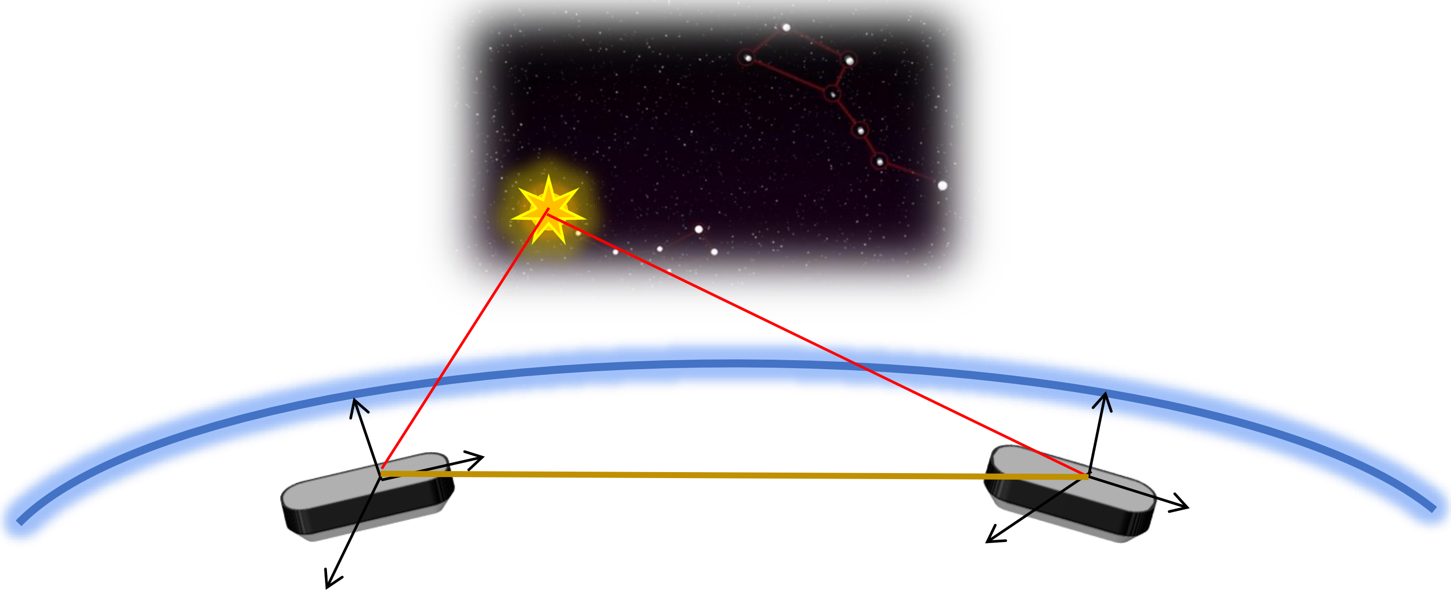
Depth Recovery

- Triangulation

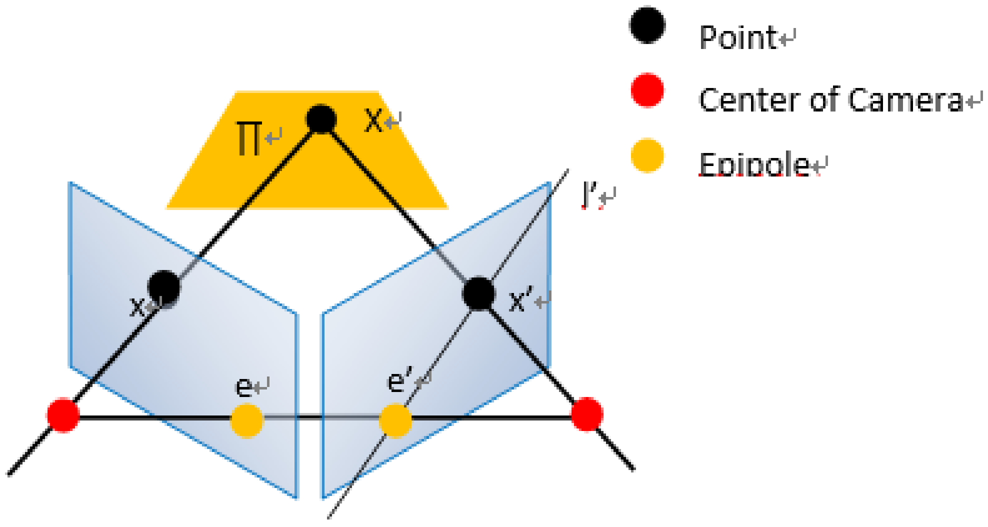


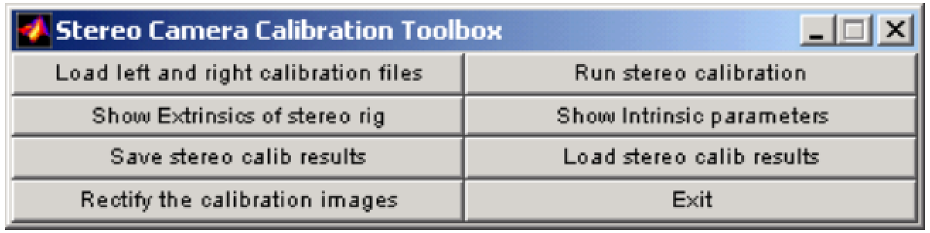
Analogous to Localization in 3D Space

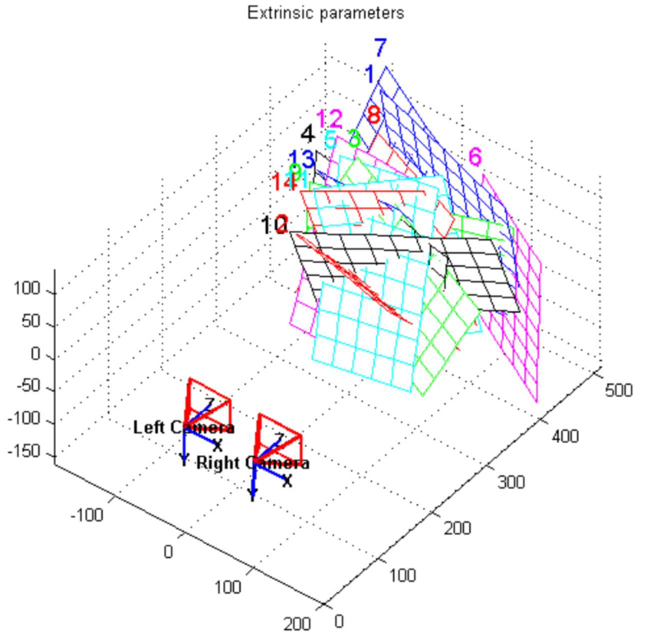
- Triangulation

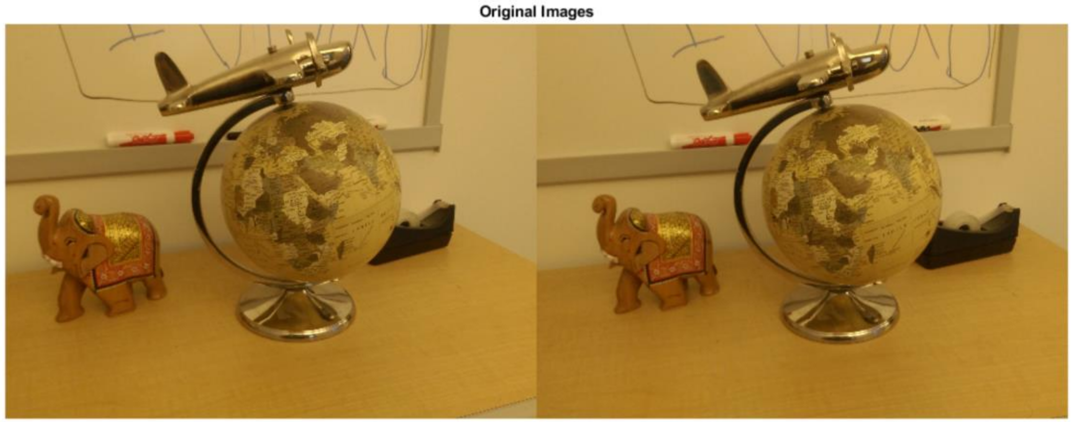


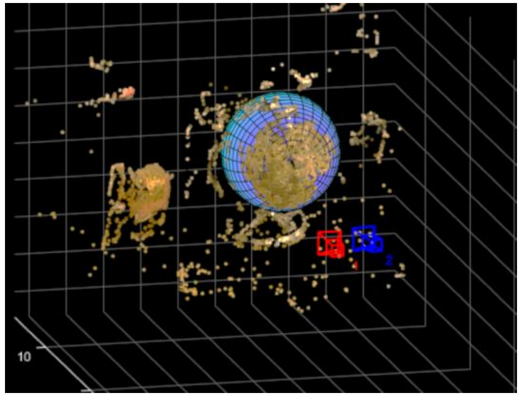
Stereo Vision







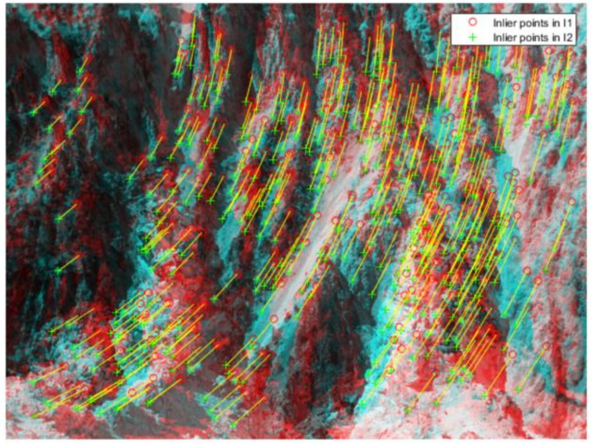




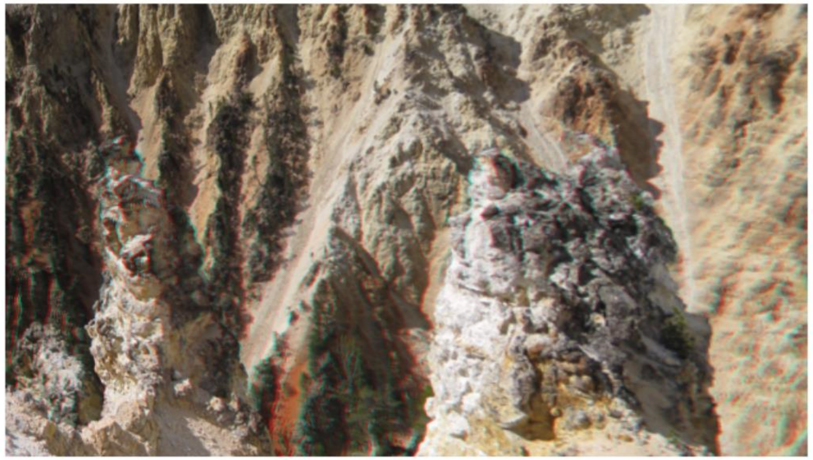
Stereo rectification

- Aim: **transform and align images** such that corresponding points appear on the same rows in both images

I1 (left); I2 (right)



Rectified Stereo Images (Red - Left Image, Cyan - Right Image)

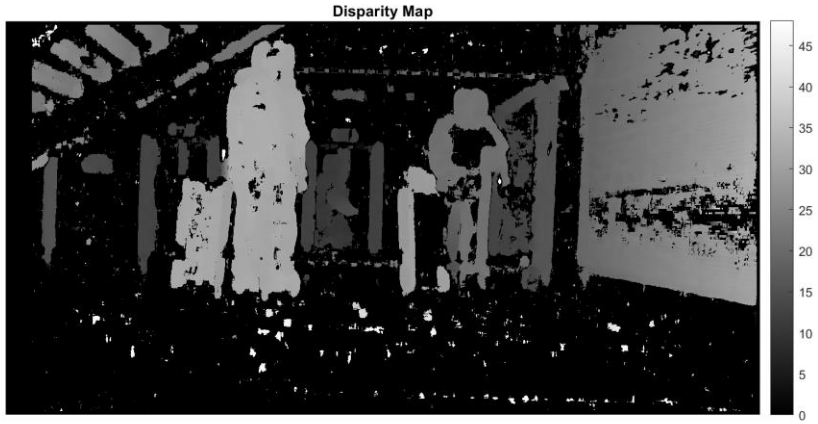


https://www.mathworks.com/help/vision/examples/uncalibrated-stereo-image-rectification.html

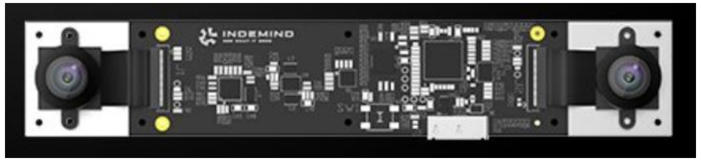
Disparity Map

Left Image

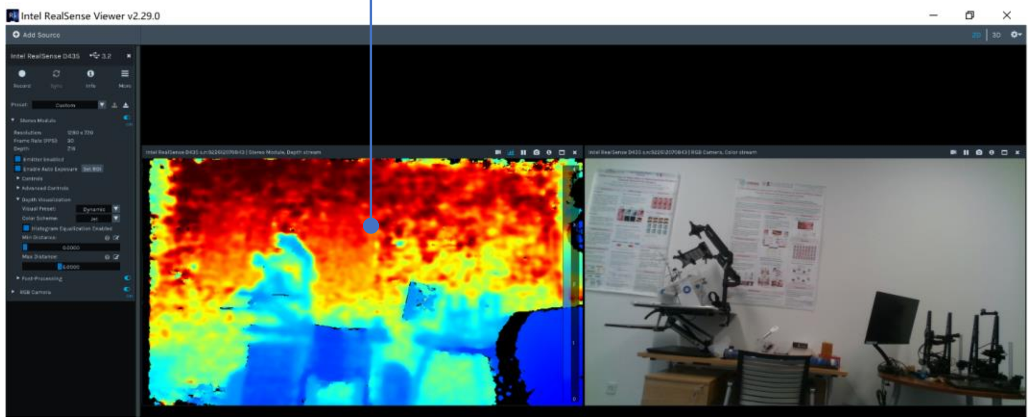
Right Image



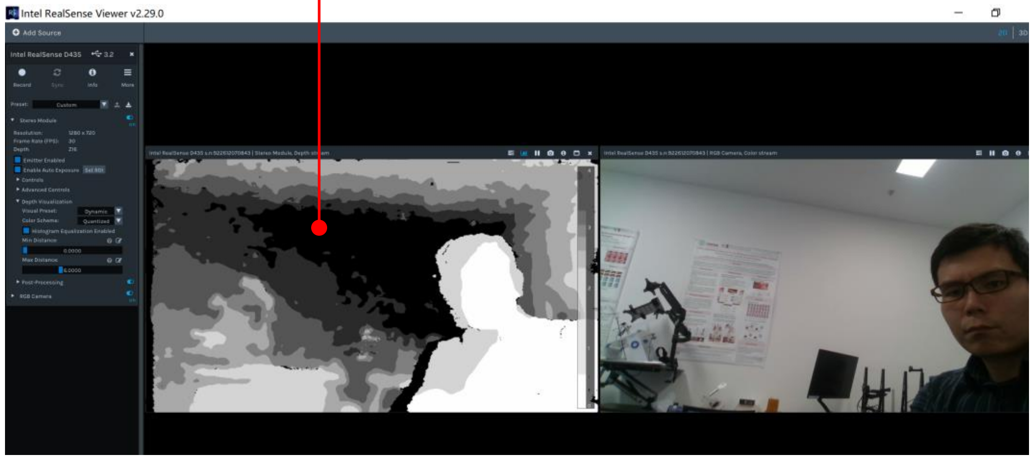
Active Depth Sensing



Depth Map



Depth Map



Structure from Motion

ECE 470 Introduction to Robotics

3D Perception

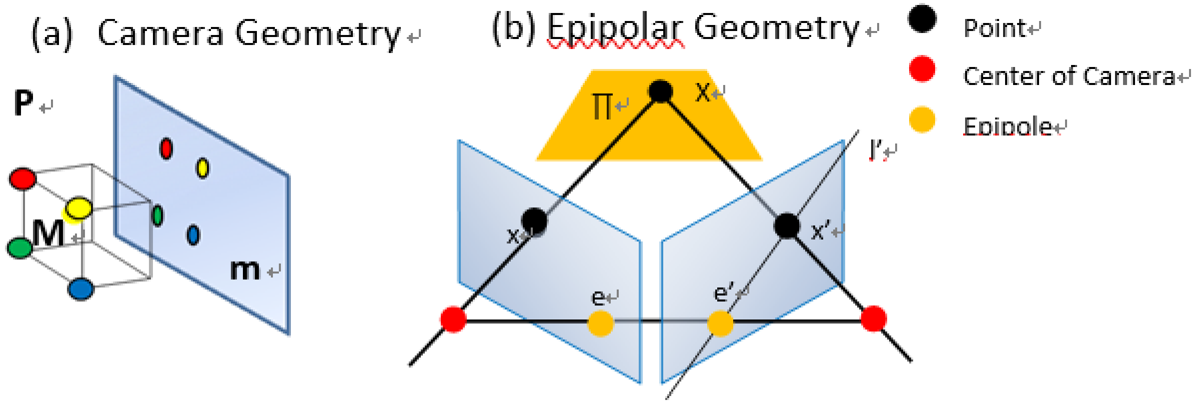
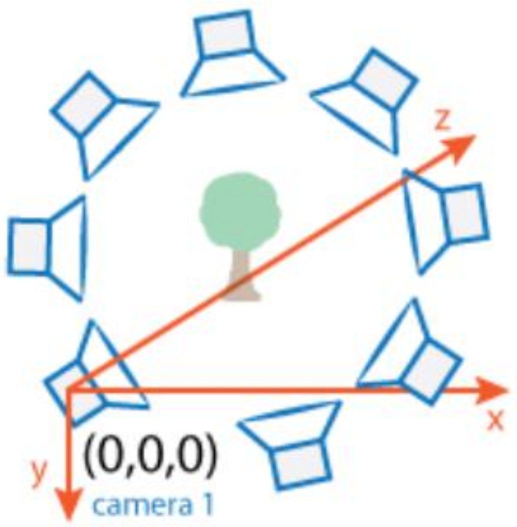
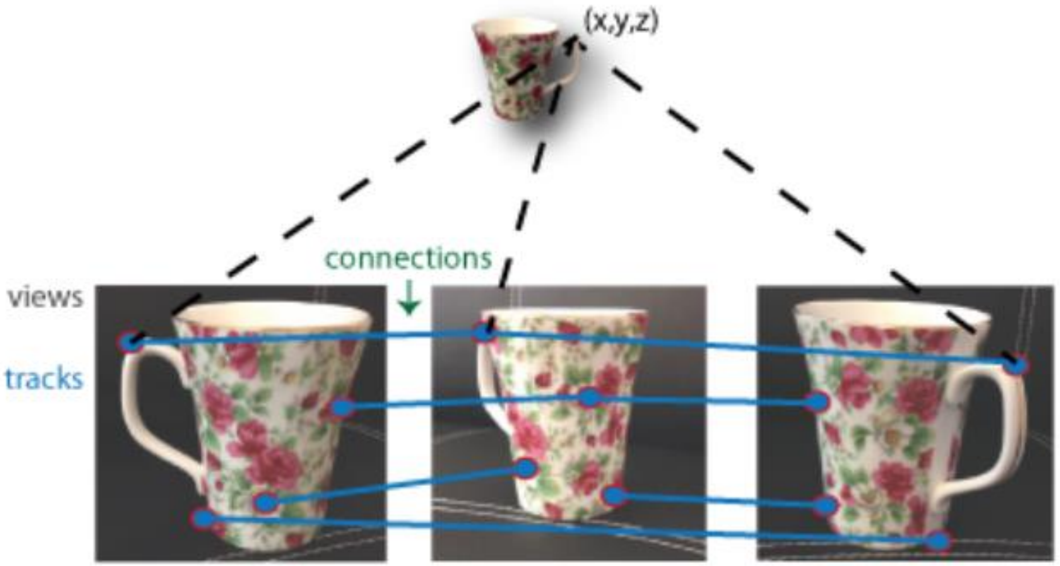


Figure 1 (a) Camera geometry (b) Epipolar geometry

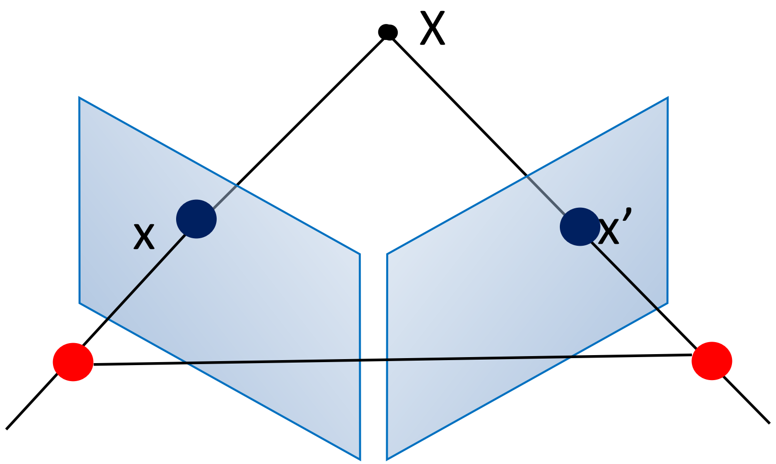
Structure from Motion

- Multiple View Geometry

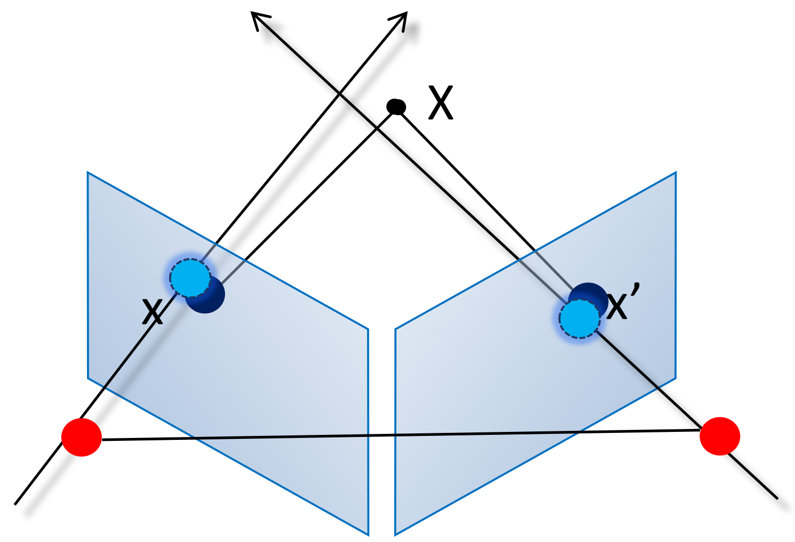




Multiple View Geometry



Multiple View Geometry



Vision-Based Control

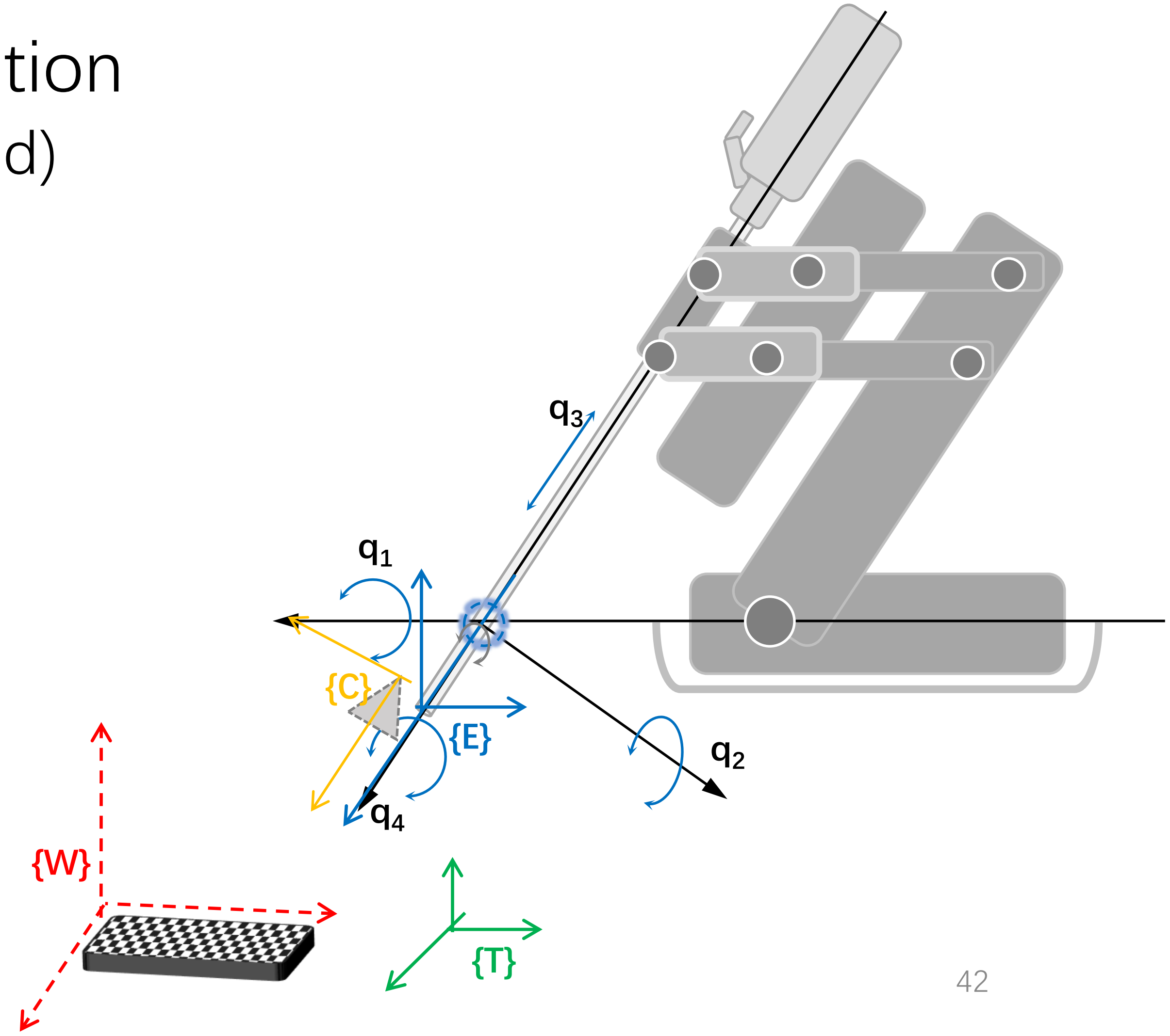
ECE 470 Introduction to Robotics

Concept of Vision-based Control

- Using extracted features from images to control the end-effector for desired task
  - Camera as a sensor for the servo mechanism
  - A.K.A. Visual Servoing

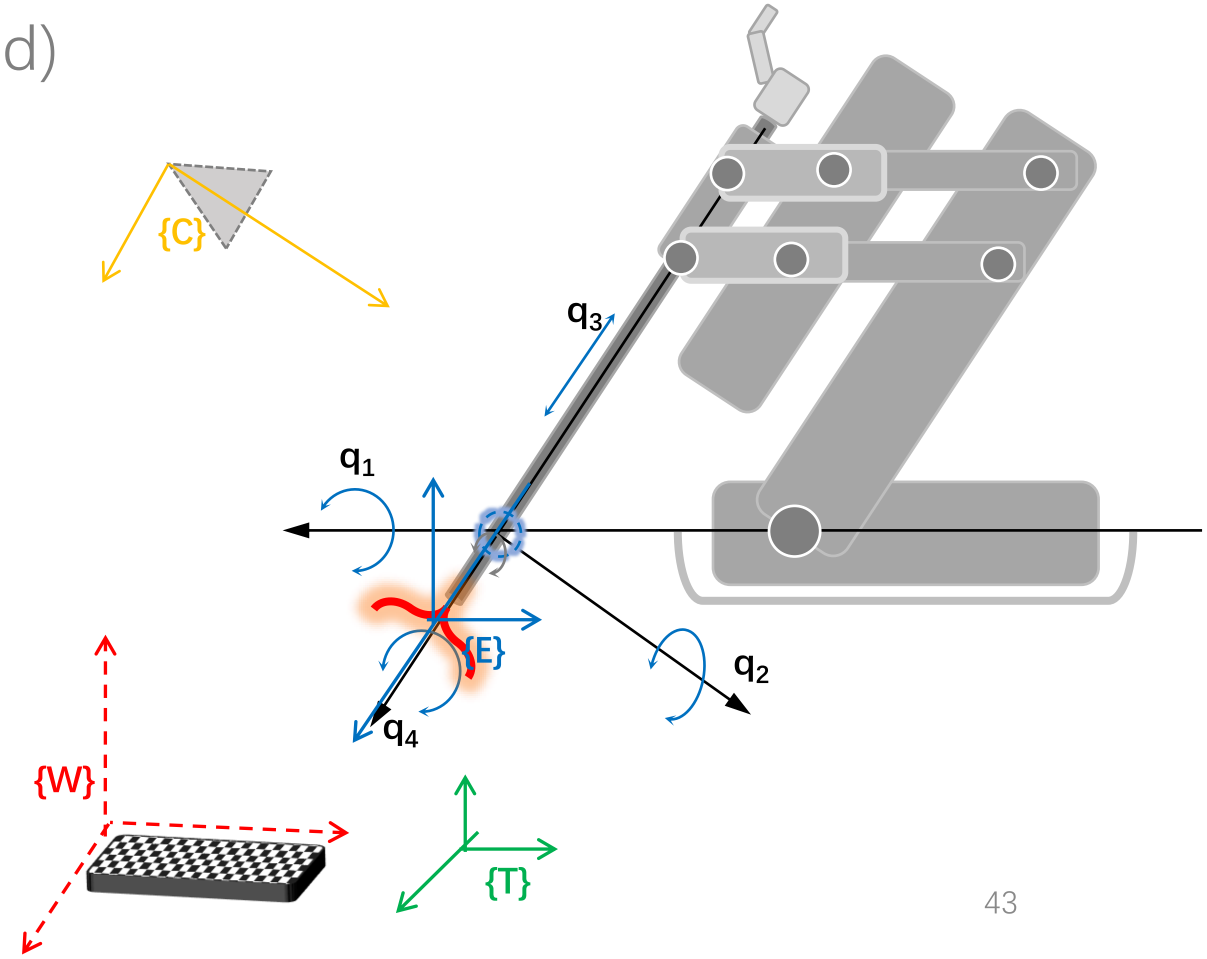
Concept of Vision-based Control

- Types of Robot-Camera Configuration
  - End-point Closed-loop (Eye-on-hand)
    - Camera move with robot/end-effector
  - End-point Open-loop
    - Camera usually stationary



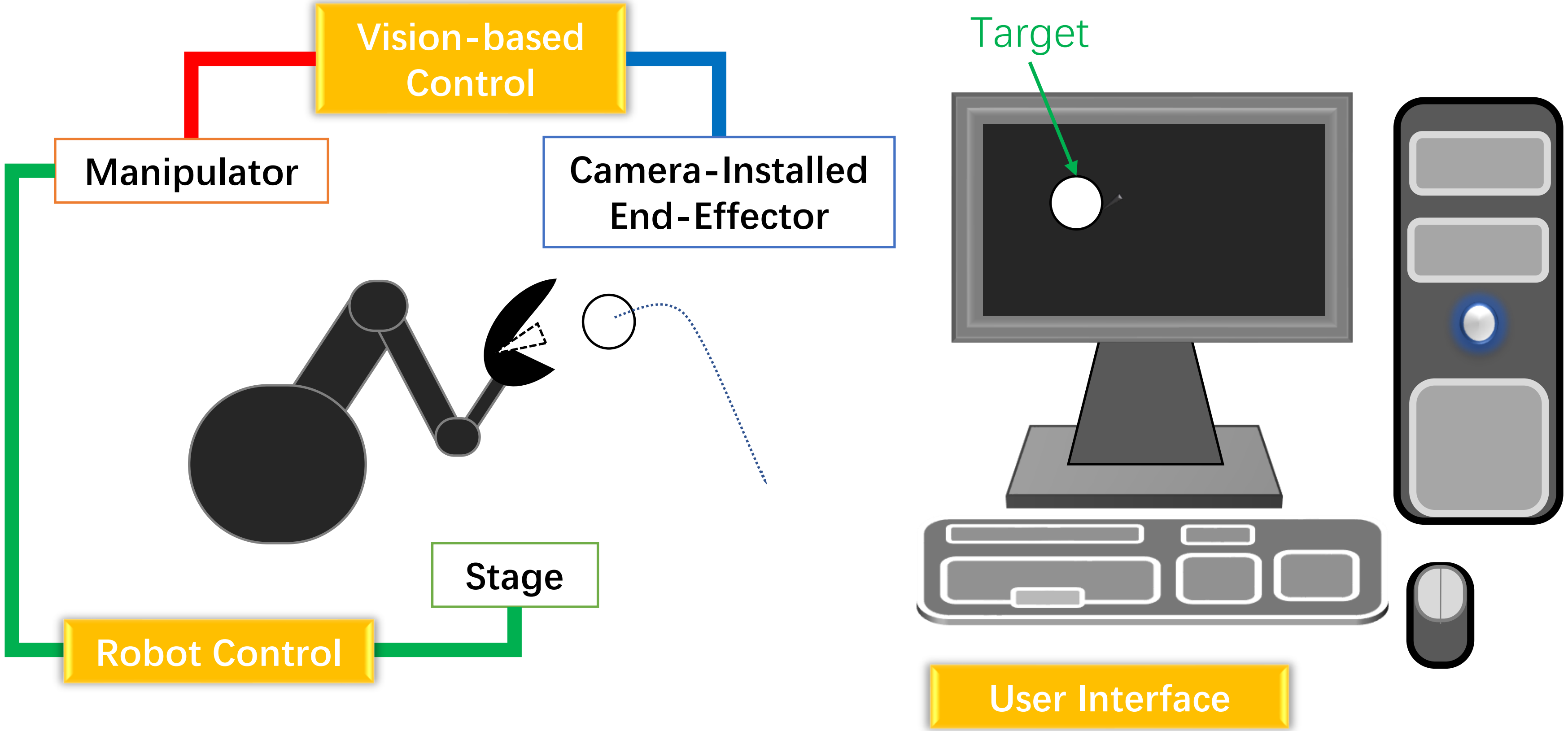
Concept of Vision-based Control

- Types of Robot-Camera Configuration
  - End-point Closed-loop (Eye-on-hand)
    - Camera move with robot/end-effector
  - End-point Open-loop
    - Camera usually stationary



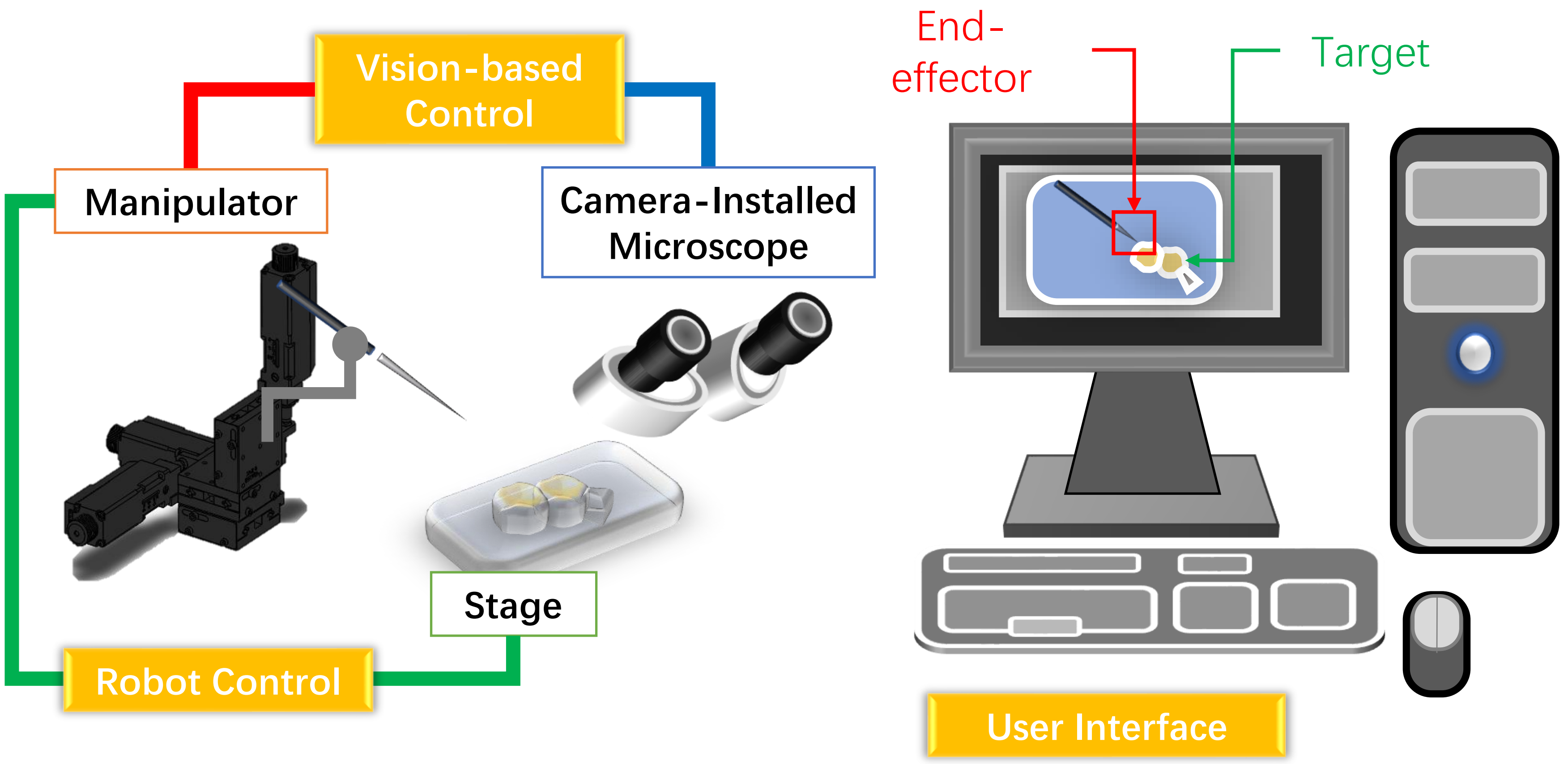
Concept of Vision-based Control

- Example of End-point Closed-loop



Concept of Vision-based Control

- Example of End-point Open-loop



Concept of Vision-based Control

- Types of Vision-based Control
  - Pose-based Visual Servo
  - Image-based Visual Servo

Pose-based Visual Servo

- Control set-point in task space
- Estimate robot pose using extracted features
- Precise calibration required

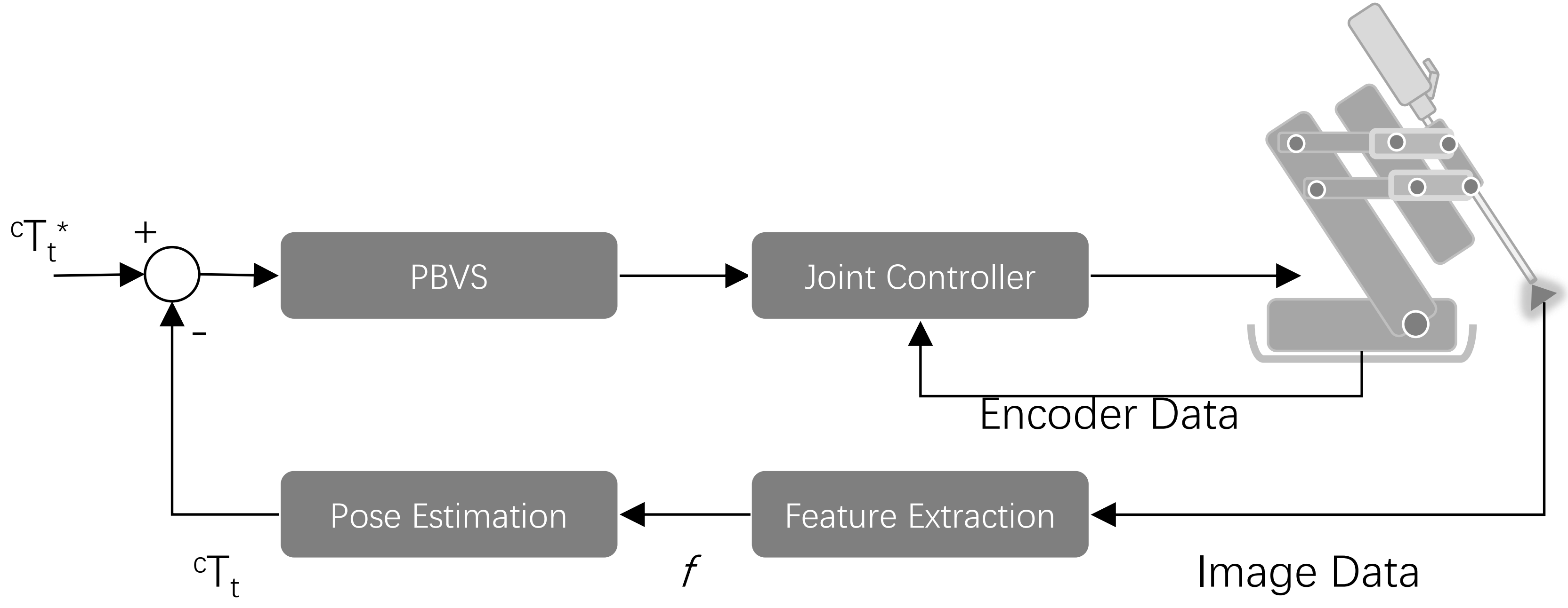


Image-based Visual Servo

- Command in image feature space
- Directly use extracted features from image as feedback
- No pose estimation required
- Usually do not require much calibration

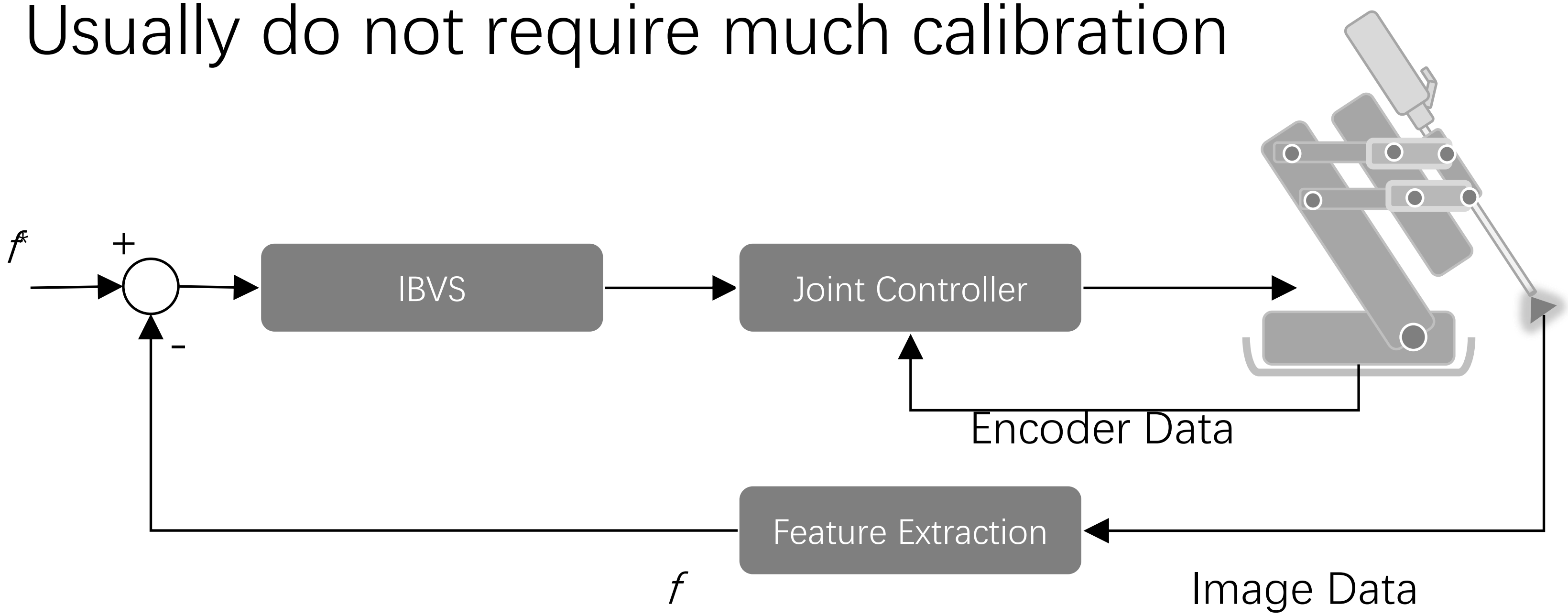
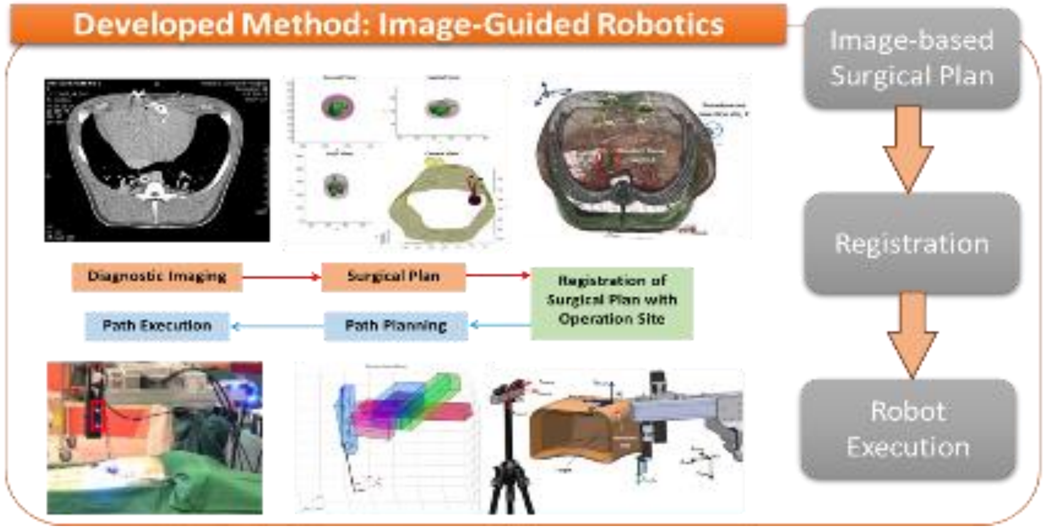


Image-Guided Robotics

Image-Guided Surgical Planning & Assistive Robotic System



Ultrasound Image-Guided Collaborative Robotic System