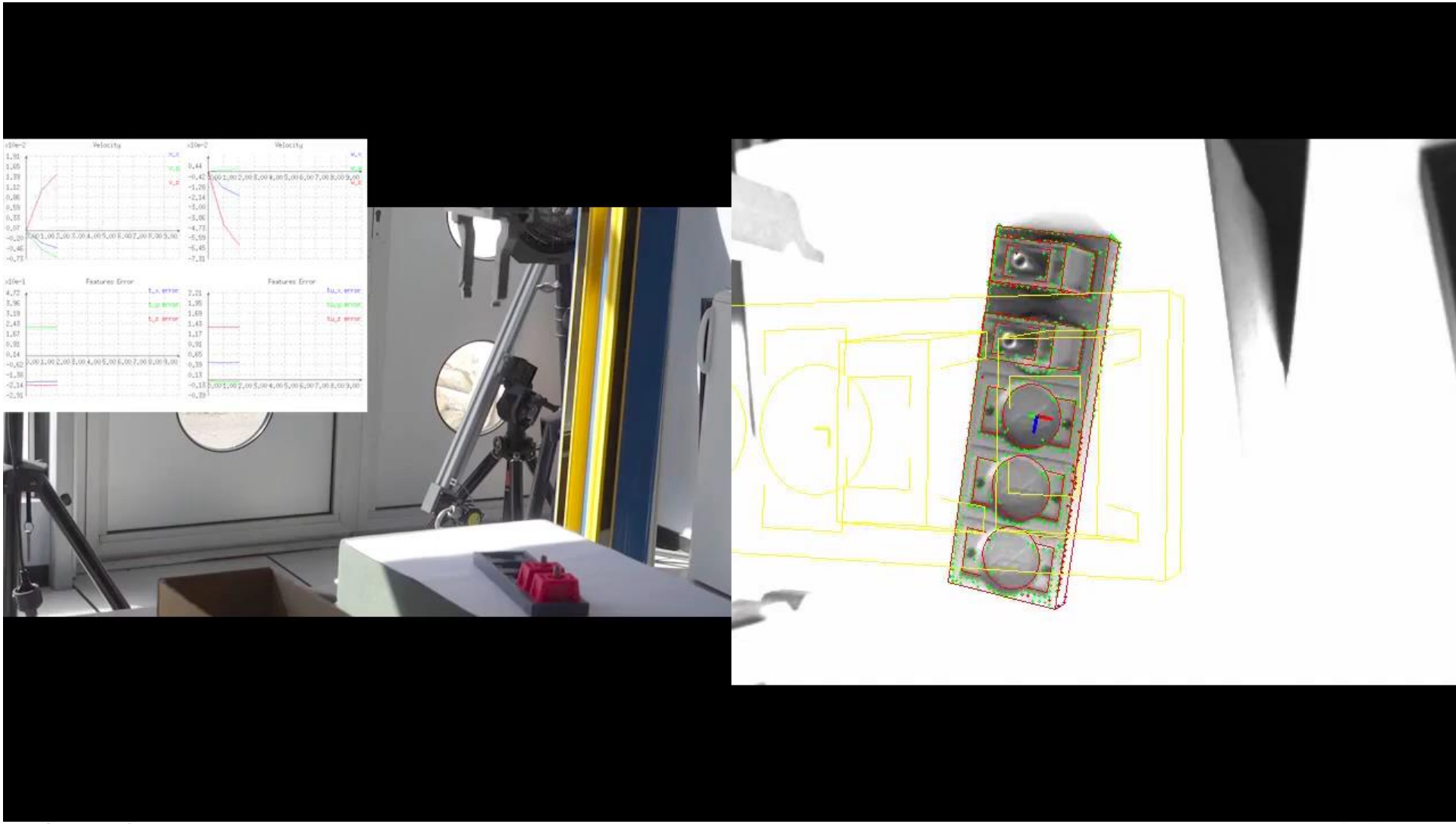
Lecture 06
Calibration & Servoing

Song Chaoyang

Assistant Professor

Department of Mechanical and Energy Engineering

songcy@sustech.edu.cn



AncoraSIR.com

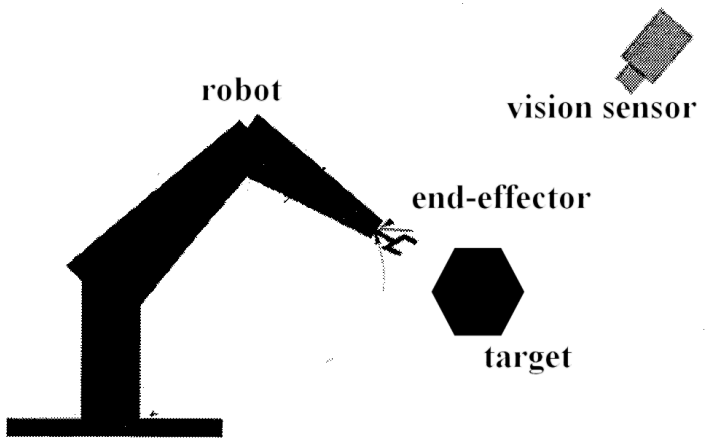
Southern University of Science and Technology

What can Computer Vision do for Robotics

- Accurate Robot-Object Positioning
- Keeping Relative Position under Movement

Visual Servoing

- Visualization / Teaching / Telerobotics
- Performing measurements
- Object Recognition
- Registration



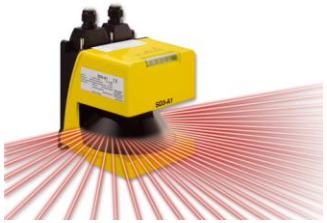
AncoraSIR.com

Vision Sensors

- Single Perspective Camera

- Multiple Perspective Cameras (e.g. Stereo Camera Pair)

- Laser Scanner



- Omnidirectional Camera

- Structured Light Sensor

AncoraSIR.com

Vision-based control

Or visual servo control

- A vision-based control system involves continuous measurement of the target and the robot using vision to create a feedback signal.
- Moves the robot arm until the visually observed error between the robot and the target is zero.

Advantage	Disadvantage
Continuous measurement and feedback provides great robustness with respect to any errors in the system (precision of the robot and camera).	Practical complexities: Camera on the end of the robot interfering with the task, camera unable to focus, or the target obscured by the gripper.

Vision-based control is quite different to taking an image, determining where the target is and then reaching for it.

Close loop VS. Open loop

AncoraSIR.com

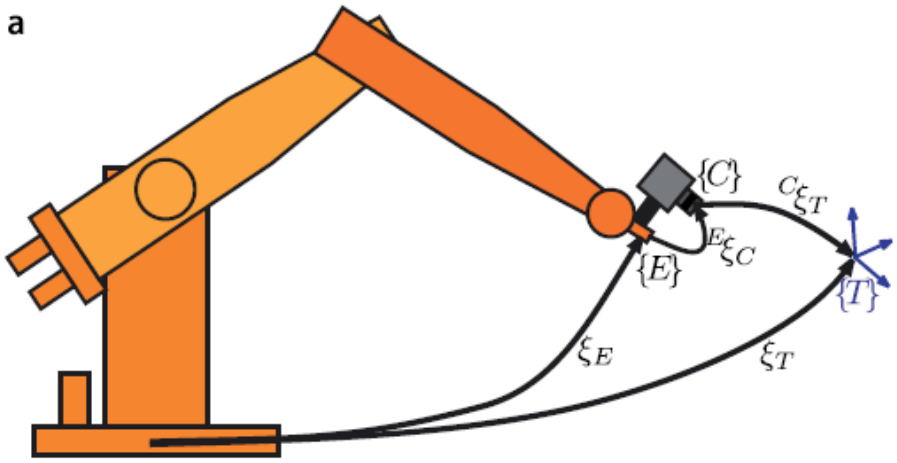
SUSTech
Southern University of Science and Technology

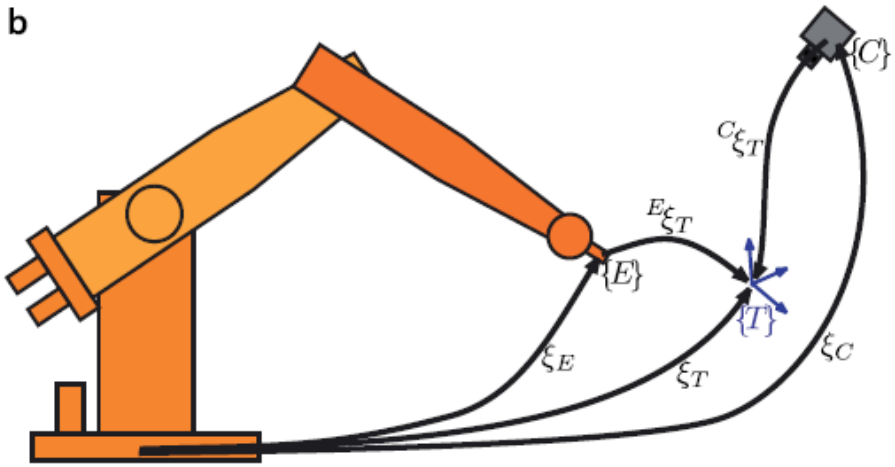
Camera Configuration

EOL and ECL control

- **ECL: endpoint closed-loop; only the target is observed by the camera**

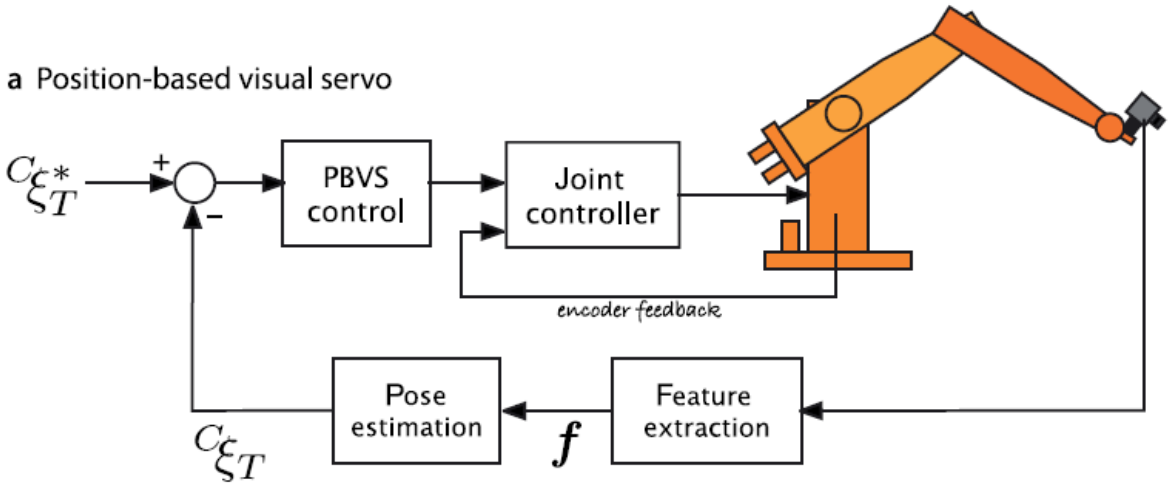
- **EOL: endpoint open-loop; target as well as end-effector are observed by the camera**



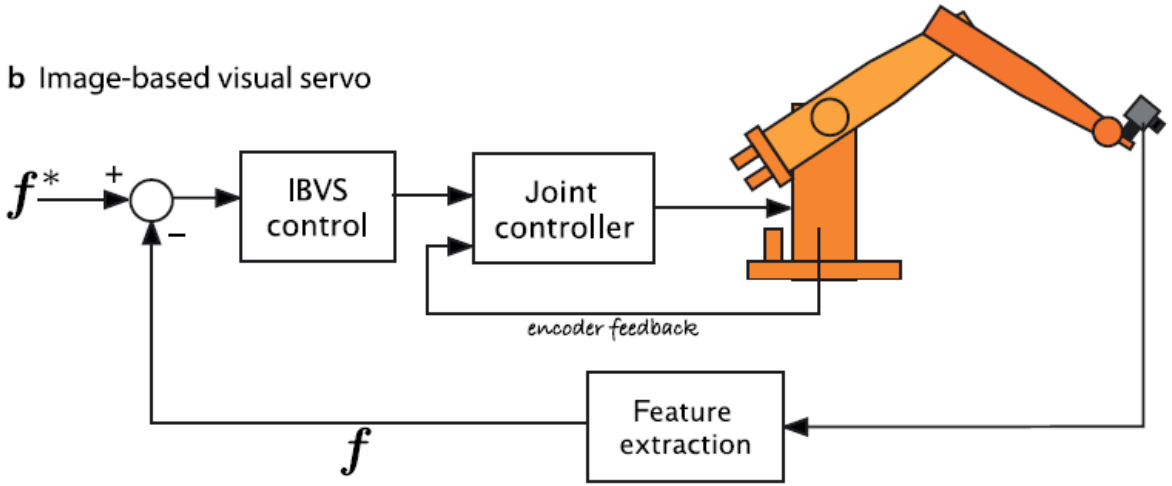


AncoraSIR.com

Two different approaches



- Uses images, calibrated camera and known geometry model of the target to determine the pose of the target with respect to the camera.
- Control is performed in task space SE(3).

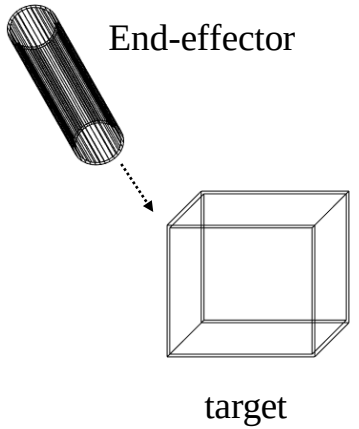


- Uses the image feature directly omitting the pose estimation step.
- Control is performed in image coordinate space R^2

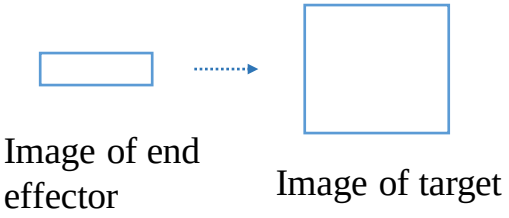
AncoraSIR.com

Position-based and Image Based control

- Position based:
  - Alignment in target coordinate system
  - The 3D structure of the target is rconstructed
  - The end-effector is tracked
  - Sensitive to calibration errors
  - Sensitive to reconstruction errors



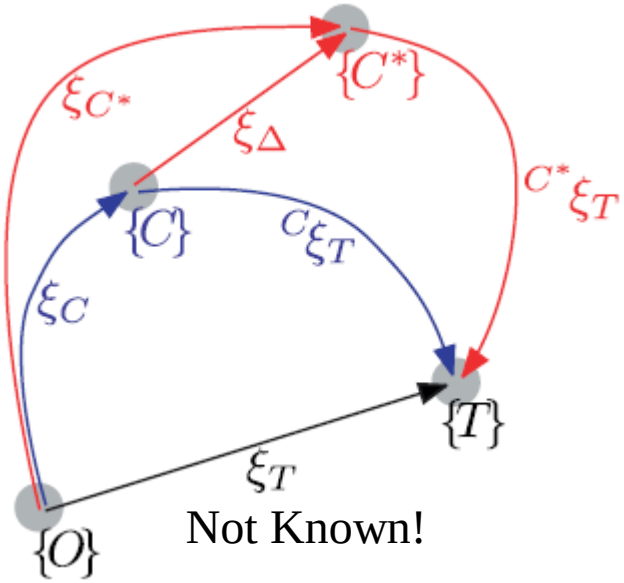
- Image based:
  - Alignment in image coordinates
  - No explicit reconstruction necessary
  - Insensitive to calibration errors
  - Only special problems solvable
  - Depends on initial pose
  - Depends on selected features



AncoraSIR.com

Position-Based Visual Servoing

Frames



$$\xi_\Delta = {}^{C}\xi_T \ominus {}^{C^*}\hat{\xi}_T$$

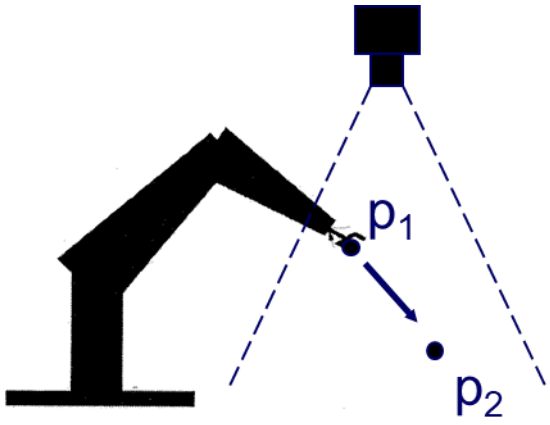
- Frame {*C*} is the current camera pose
- Frame {*C**} is the desired camera pose
- Frame {*T*} is the target

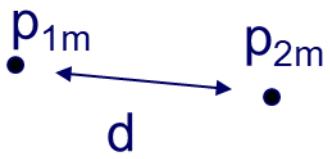
- Steps:
  1. Estimate camera pose ${}^{C}\xi_T$
  2. Compute of error between current pose and target pose ${}^{C}\xi_T \ominus {}^{C^*}\hat{\xi}_T$
  3. Determine movement of robot ξ_Δ

AncoraSIR.com

Example

Position based point alignment



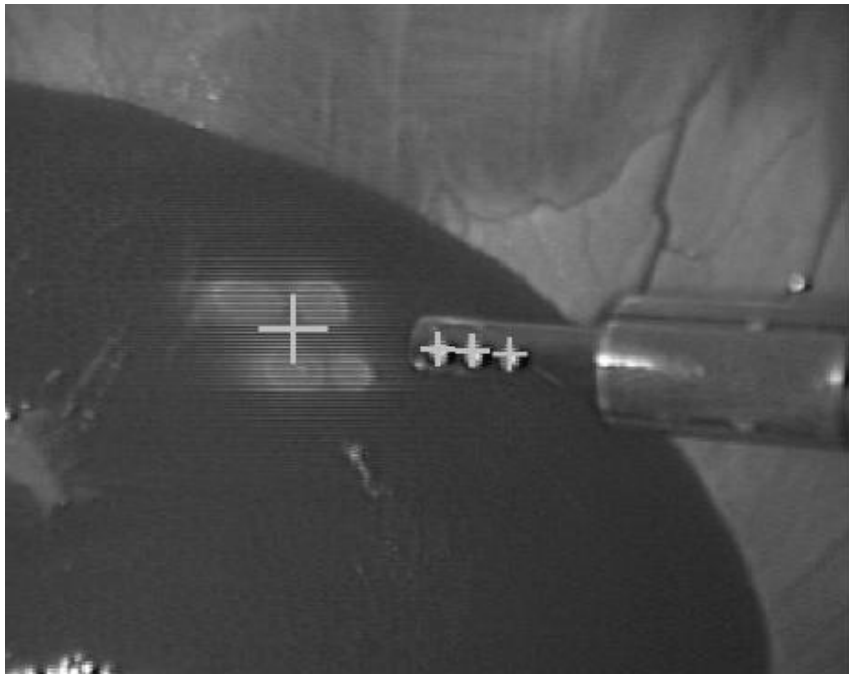


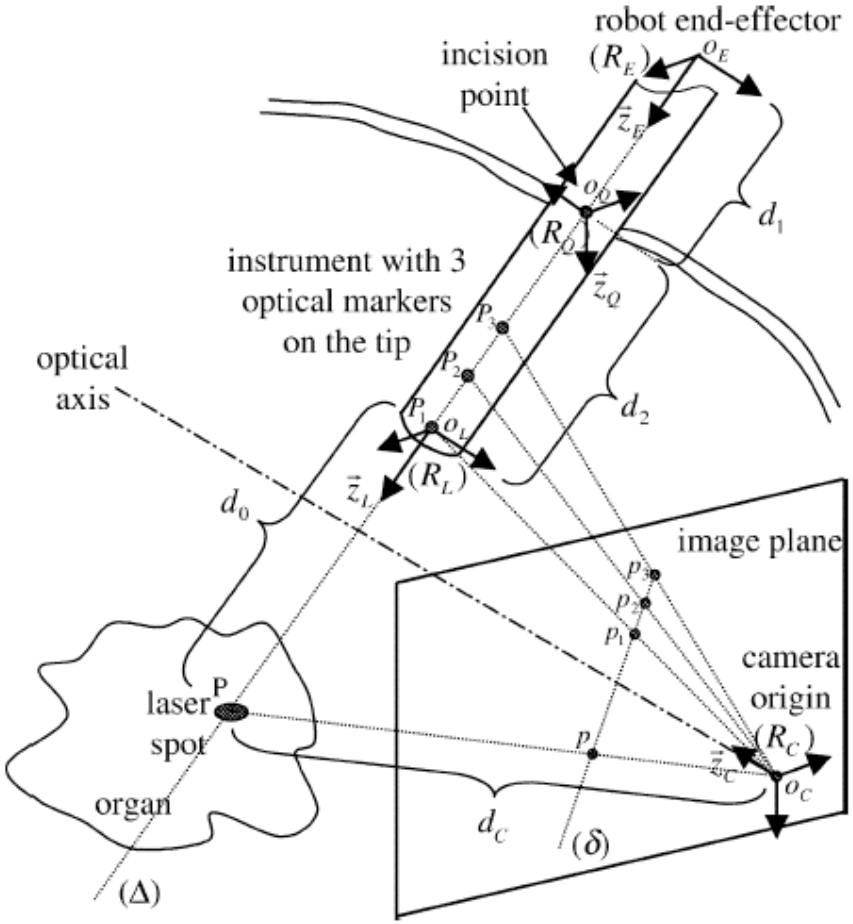
- Goal: bring e to 0 by moving p_1
  - $\mathbf{e = |p_{2m} - p_{1m}|}$
  - $\mathbf{u = k*(p_{2m} - p_{1m})}$
- p_{xm} is subject to the following measurement errors: sensor position, sensor calibration, sensor measurement error
- p_{xm} is independent of the following errors: end effector position, target position

AncoraSIR.com

Example

Image base visual servoing



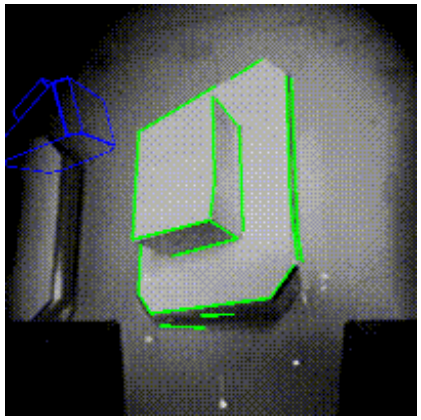
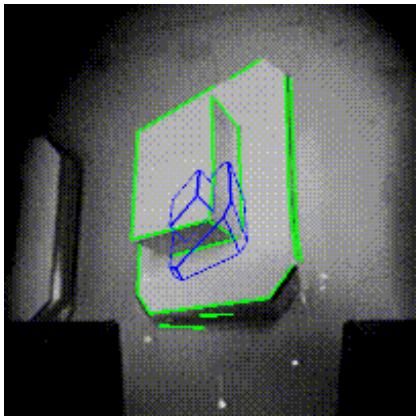
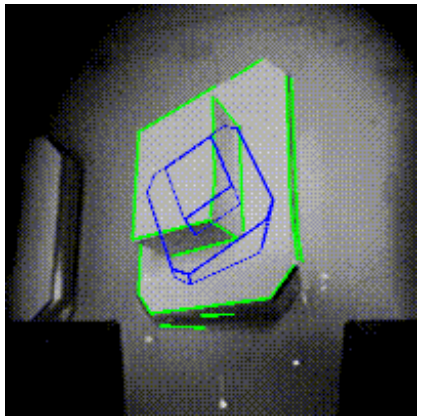
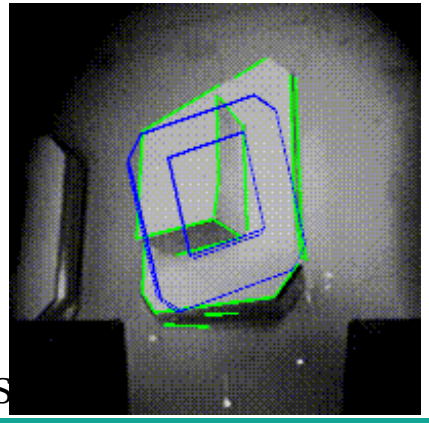
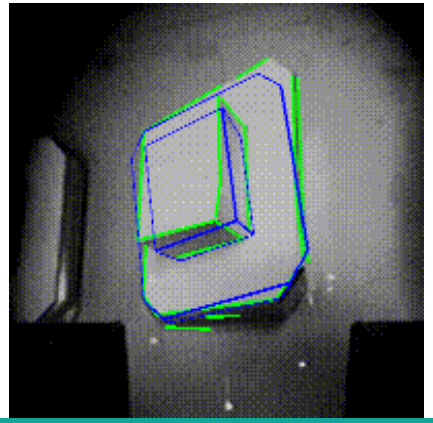
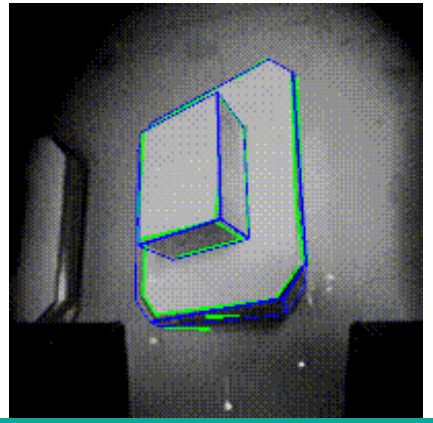


AncoraSIR.com

Example

Image base visual servoing

- Registration of CAD models to scene features:

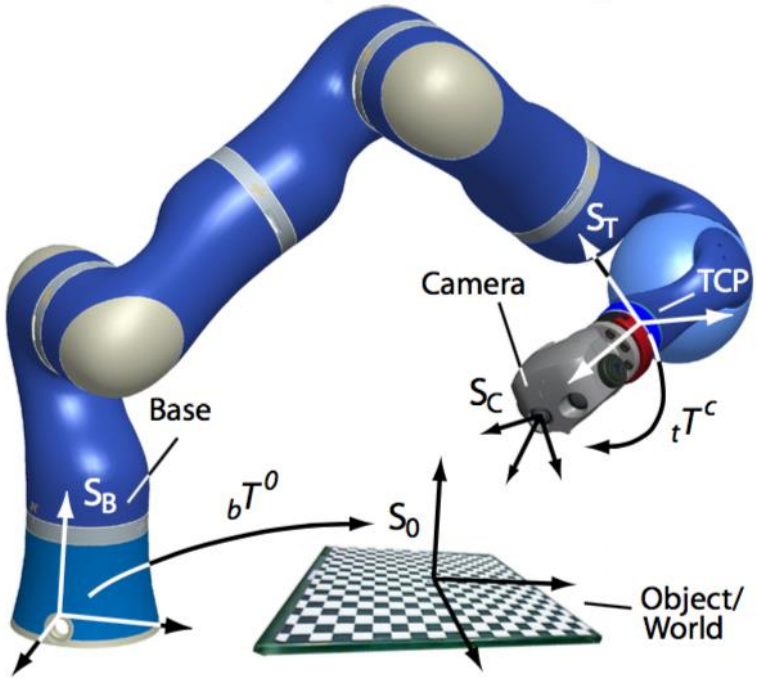







AncoraS

Before Visual Servoing

Hand Eye Calibration needed for position based visual servoing

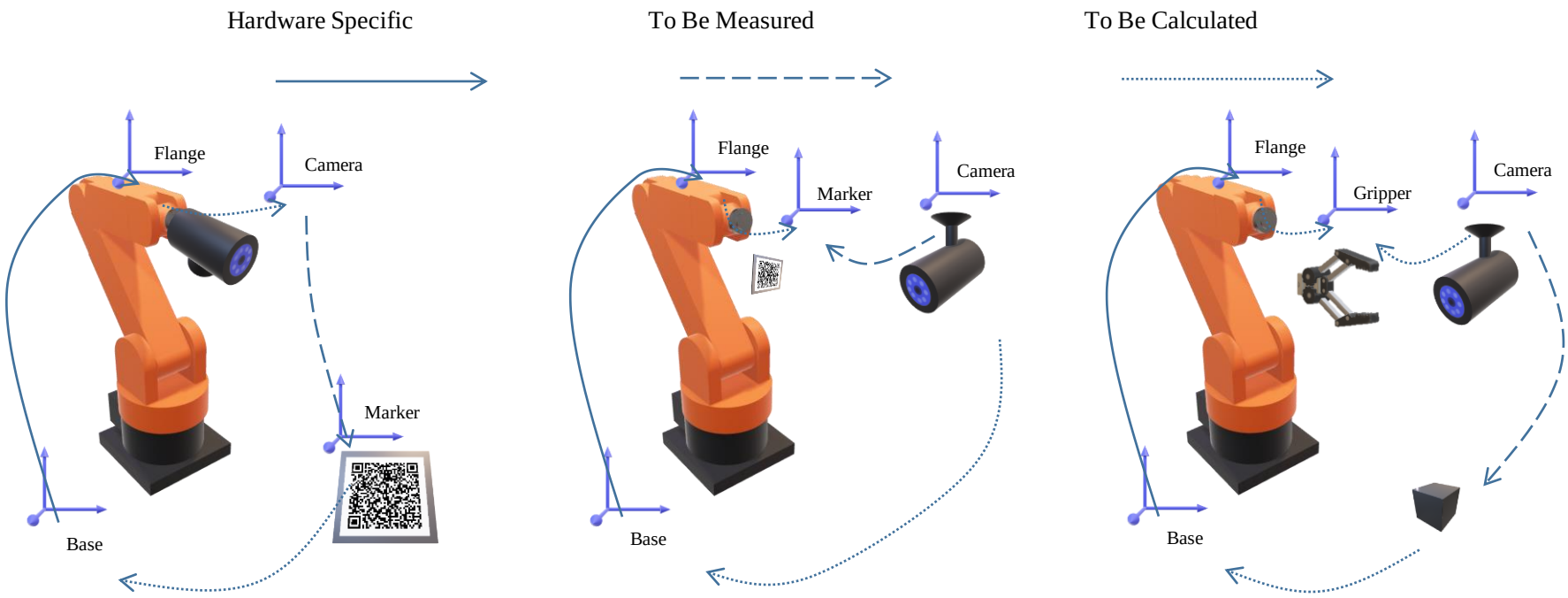
AncoraSIR.com



SUSTech
Southern University of Science and Technology

Hand Eye Calibration

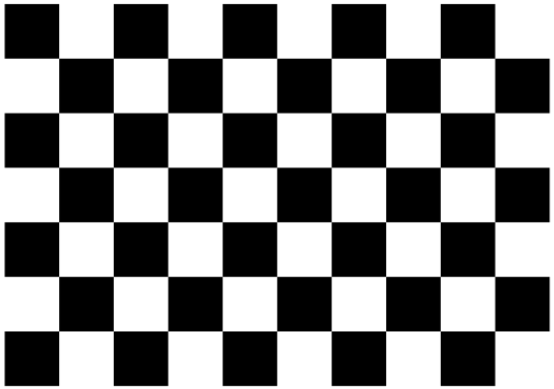
Problem types



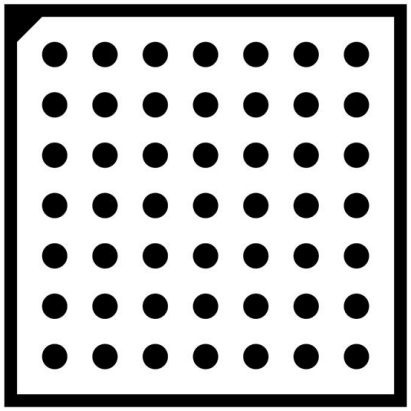
(a) Eye-in-Hand (b) Eye-on-Base (c) Learning-based

AncoraSIR.com

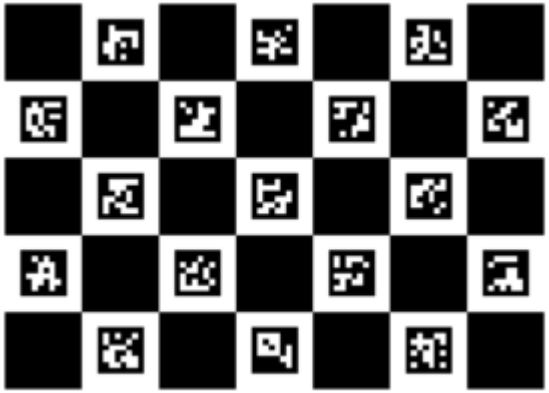
Calibration Patterns



Chessboard



Dot marker



Aruco marker

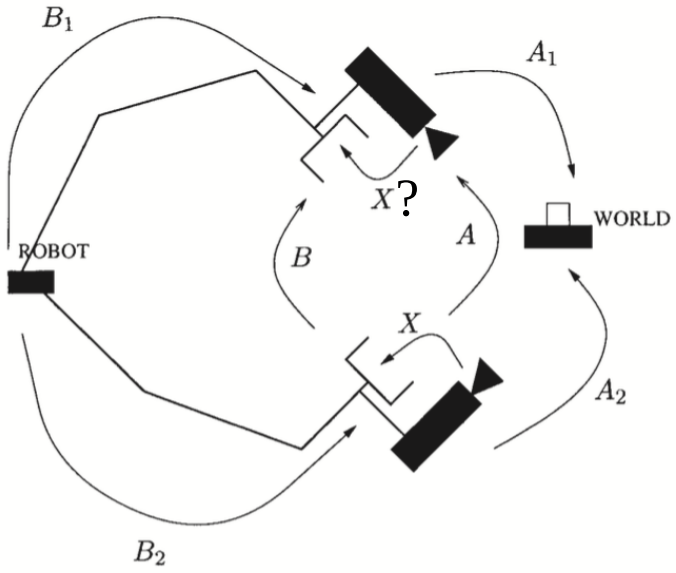
AncoraSIR.com

SUSTech
Southern University of Science and Technology

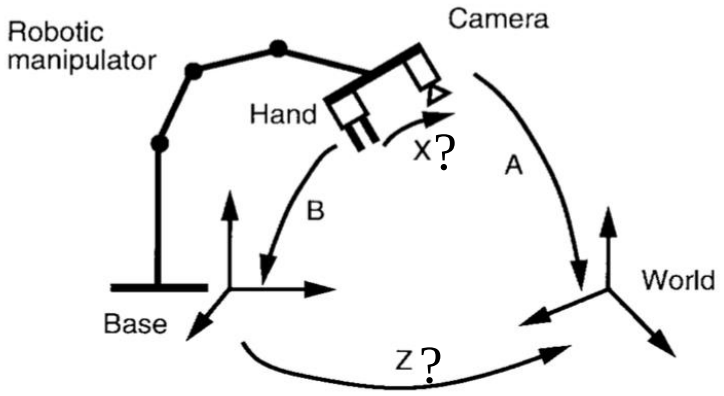
Two Common Solutions

Formulation

- Move the hand and observe/perceive the movement of the eye:
  AX = XB



- Simultaneous estimation of the hand-eye transformation and the pose of the robot in the world:
  AX = ZB



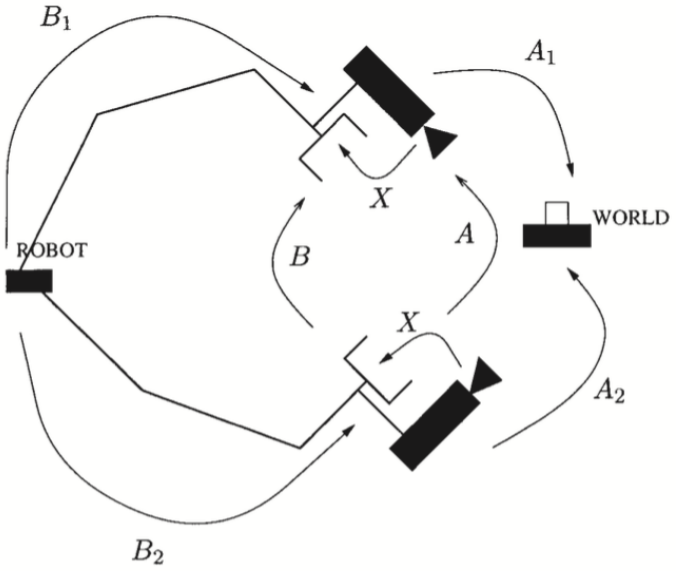
AncoraSIR.com

AX = XB

- **A**, **X**, **B** are each 4x4 spatial transform matrices

- Move an arm between multiple positions and record each
  - each camera position relative to a target
  - arm position relative to its base

- The transforms in the equation are for each time i and i+1:
  - $\mathbf{A}_i$ is between each camera pose
  - $\mathbf{B}_i$ is between each arm pose
  - **X** is the transform between the camera and arm being measured and doesn't change over time



AncoraSIR.com

AX = XB

Solutions

- Determining first the Rotation then the Translation:
  - Tsai: A new technique for fully autonomous and efficient 3D robotics hand/eye calibration

$$\begin{pmatrix} \mathbf{R_A} & \mathbf{t_A} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R_X} & \mathbf{t_X} \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R_X} & \mathbf{t_X} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R_B} & \mathbf{t_B} \\ 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{R_A R_X} & \mathbf{R_A t_X} + \mathbf{t_A} \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R_X R_B} & \mathbf{R_X t_B} + \mathbf{t_X} \\ 0 & 1 \end{pmatrix},$$

$$\mathbf{R_A R_X} = \mathbf{R_X R_B},$$

$$\mathbf{R_A t_X} + \mathbf{t_A} = \mathbf{R_X t_B} + \mathbf{t_X}.$$

Reference:
https://blog.csdn.net/Sandy_WYM_/article/details/83996479

AncoraSIR.com

SUSTech
Southern University of Science and Technology

AX = XB

Solutions

- Determine Rotation and Translation simultaneously
  - K. Daniilidis: Hand-eye calibration using dual quaternions

- Iterative solutions: based on minimizing ||AX – XB|| with Levenberg-Marquardt optimization algorithm.

- Solutions to AX = ZB can be categorized similar to AX = XB, for details refer to “An Overview of Robot-Sensor Calibration Methods for Evaluation of Perception Systems”

AncoraSIR.com

Open Source softwares

Implementation

- Matlab: Camera Calibration Toolbox
- Camodocal: https://github.com/hengli/camodocal
  - K. Daniilidis 1999 Hand-Eye Calibration Using Dual Quaternions (Tested and working)
  - Chessboard camera calibration
  - Stereo Camera Calibration
- Easy_handeye: https://github.com/IFL-CAMP/easy_handeye
- VISP: https://github.com/lagadic/vision_visp/tree/master/visp_hand2eye_calibration

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Problems with previous methods

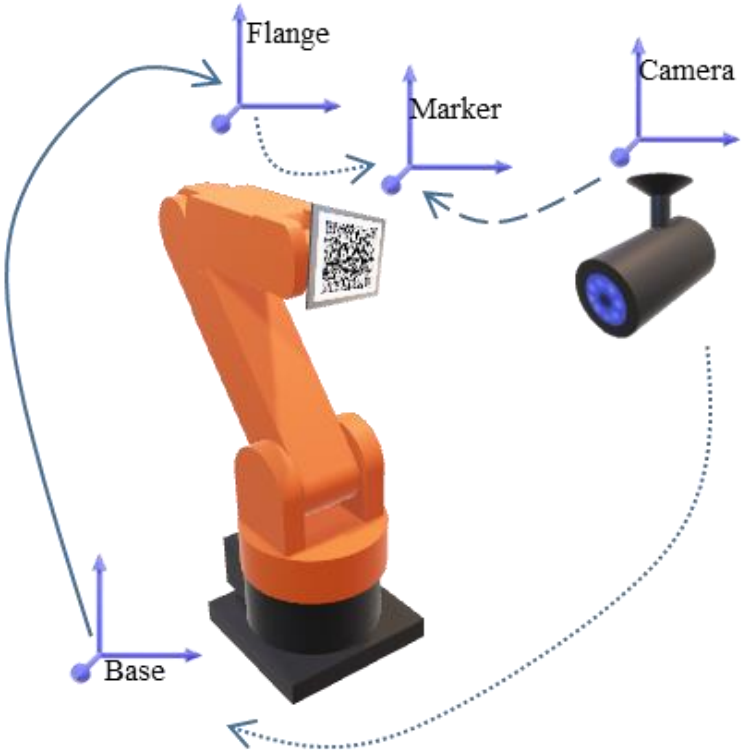
- Highly sensitive to the image data collected.
- Usually need to collect tens of images data and repeated for a few times for a reasonable result.
- Difficult to identify reasons for calibration errors.
- Universal but difficult to obtain accurate results.
- Can the problem be simplified?

  Yes!

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Flange based calibration

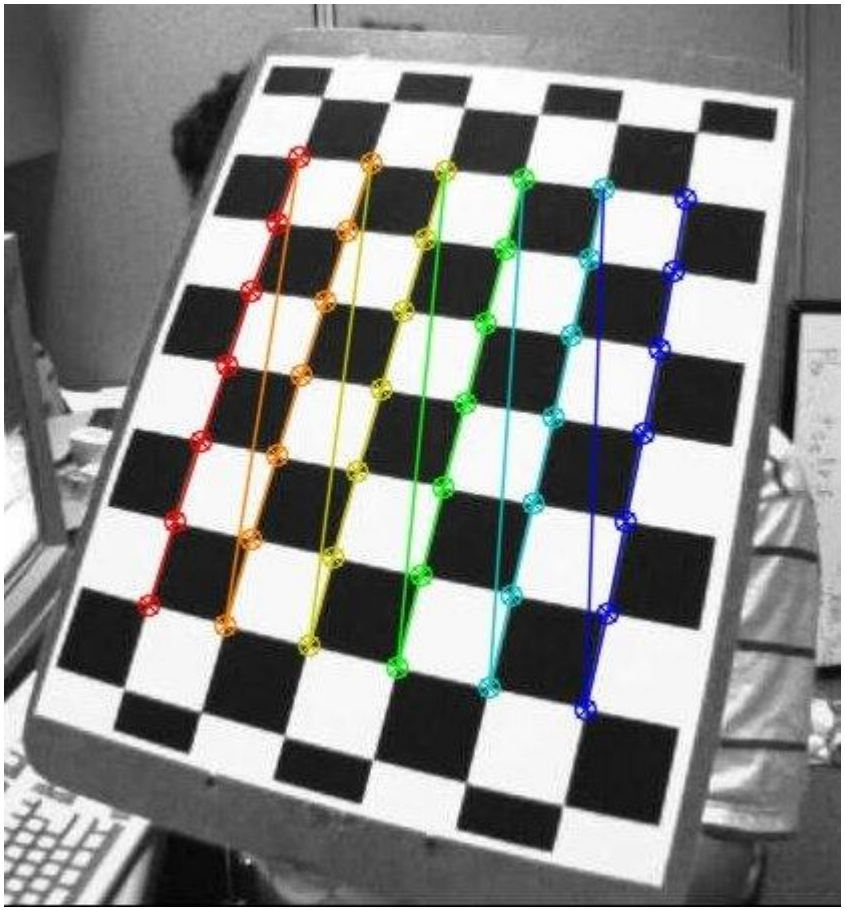


(b) Eye-on-Base

- Put marker (chessboard or Aruco) on the flange so that the center of the marker is coincident with the tool0 point (center of the tool-mounting flange)

AncoraSIR.com

Flange based calibration



Steps:

1. Detect the position of the calibration board with reference to camera.
2. Record the position of the tool0 point with reference to robot base.
3. Repeat 1 and 2 > 4 times to collect two sets of points ${}^{Cam}\{p_i\}$ and ${}^{Base}\{p_i\}$, i=1,2,…,n, n>=4

AncoraSIR.com

Hand eye transformation estimation

Least-Squares

- Given two sets of points $^{Cam}\{p_i\}$ and $^{Base}\{p'_i\}$, i=1,2,…,n,

  $p'_i = R * p_i + T + N_i$, where R and T is the rotation and translation from robot base to camera coordinate.

- Least-squares solution of R, T:

$$p' \triangleq \frac{1}{N}\sum_{i=1}^{N} p'_i \qquad p \triangleq \frac{1}{N}\sum_{i=1}^{N} p_i. \qquad q_i \triangleq p_i - p \qquad H \triangleq \sum_{i=1}^{N} q_i q_i'^t$$

$$q'_i \triangleq p'_i - p'$$

$$\Rightarrow \begin{cases} H = U\Lambda V^t \\ R = VU^t \\ \hat{T} = p' - \hat{R}p \end{cases}$$

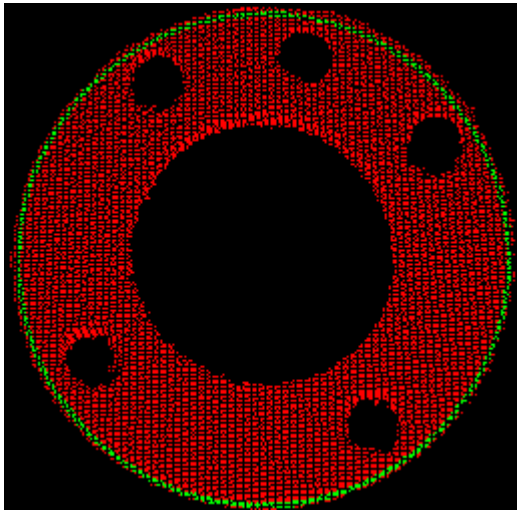
AncoraSIR.com

Hand eye calibration for 3D camera

Flanged based

- Tool-mounting flanges of robots have ISO standard and are manufactured with high precision.

- Use the point cloud of the tool-mounting flange as the calibration marker.



- Detect circle features of point cloud

AncoraSIR.com

Homework

- Allocate three robot to the students:

  I. Aubo-i5 with suction cup

  II. Cobotta with two finger gripper

  III. Franka with two finger gripper: need to install a real-time kernel and requires a PC with high network performance. Follow the steps carefully in https://frankaemika.github.io/docs/installation.html

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Homework

- IMPORTANAT:
- All robot operation must supervised by the instructors.
- If any doubt, consult the instructor first before moving the robot arm.
- Always have one hand on top of the emergency stop button when moving the robot arm.

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Homework

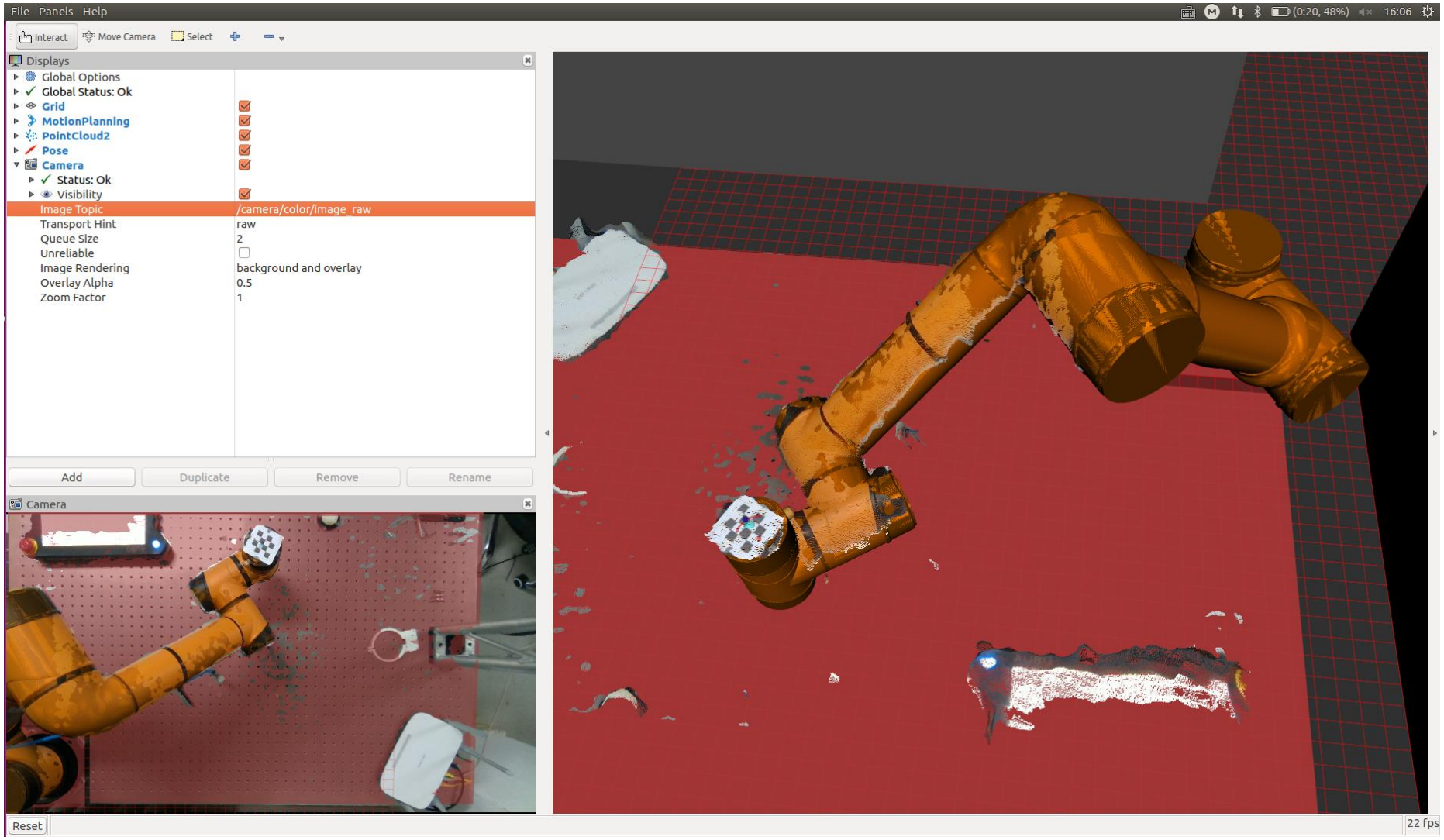
- Code: https://github.com/ancorasir/BionicDL-CobotLearning-Project2.git
  - Perform hand eye calibration using flanged-based method
  - Move gripper/suction cup to the detected chess pieces.
  - Use Realsense python package to control the camera:

    sudo pip install pyrealsense
- ROS driver for Cobotta: https://github.com/NERanger/denso_robot_ros
- ROS driver for Franka:
  - https://github.com/frankaemika/franka_ros.git
  - https://github.com/ros-planning/panda_moveit_config.git

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Good calibration result



Thank you!

Prof. Song Chaoyang

- Dr. Wan Fang (sophie.fwan@hotmail.com)

AncoraSIR.com