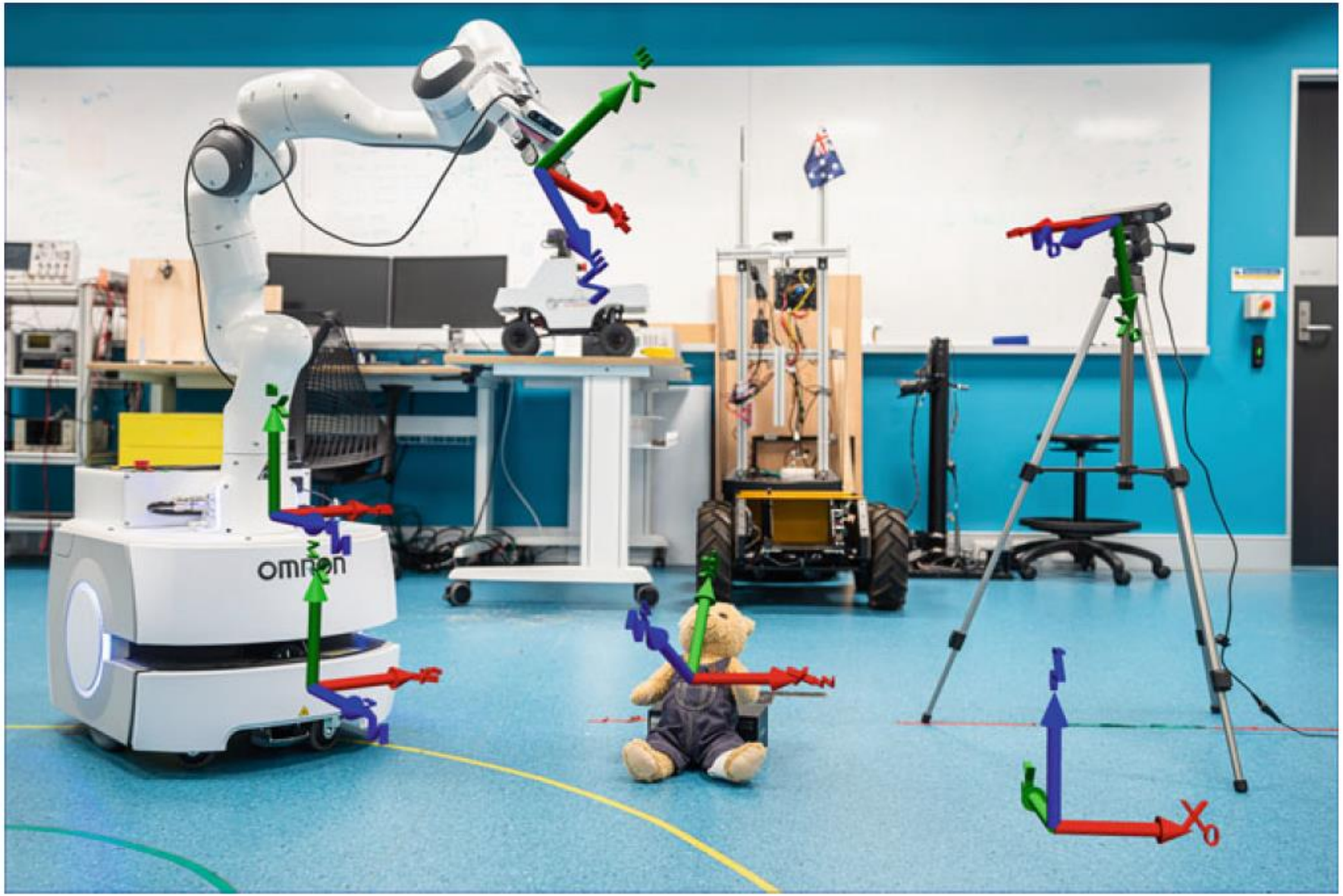
카메라 좌표계
Coordinate Frames

심주용
숙명여자대학교
기계시스템학부

Coordinate Frames



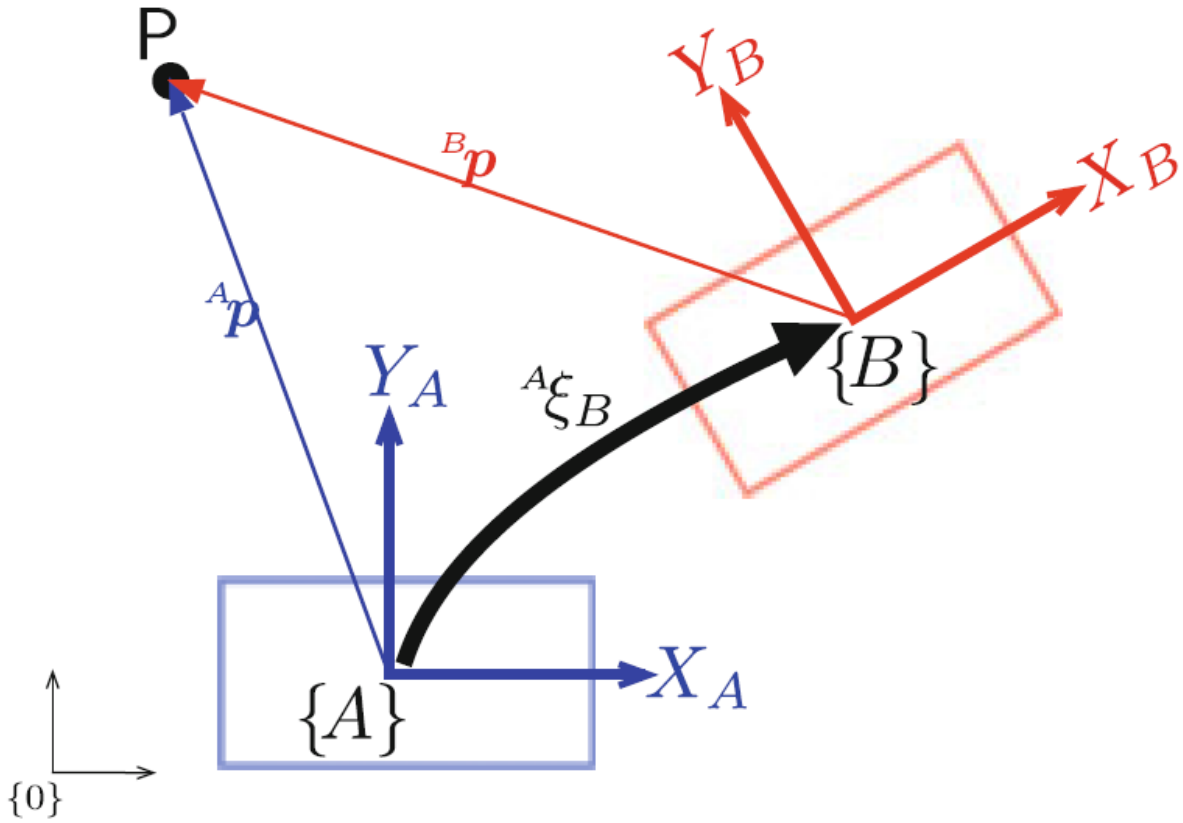
Peter Corke - Robotics, Vision and Control_ Fundamental Algorithms in Python, 3rd Edition. 146-Springer (2023)

Coordinate Frames

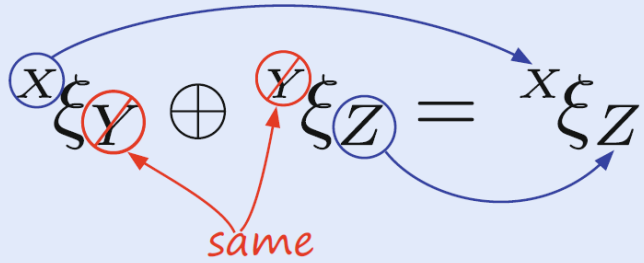
Point P can be described by coordinate vectors expressed in frame {A} or {B}

$$^{A}\boldsymbol{p} = {}^{A}\boldsymbol{\xi}_{B} \cdot {}^{B}\boldsymbol{p}$$

$^{A}\xi_{B}$: the pose of {B} relative to {A}

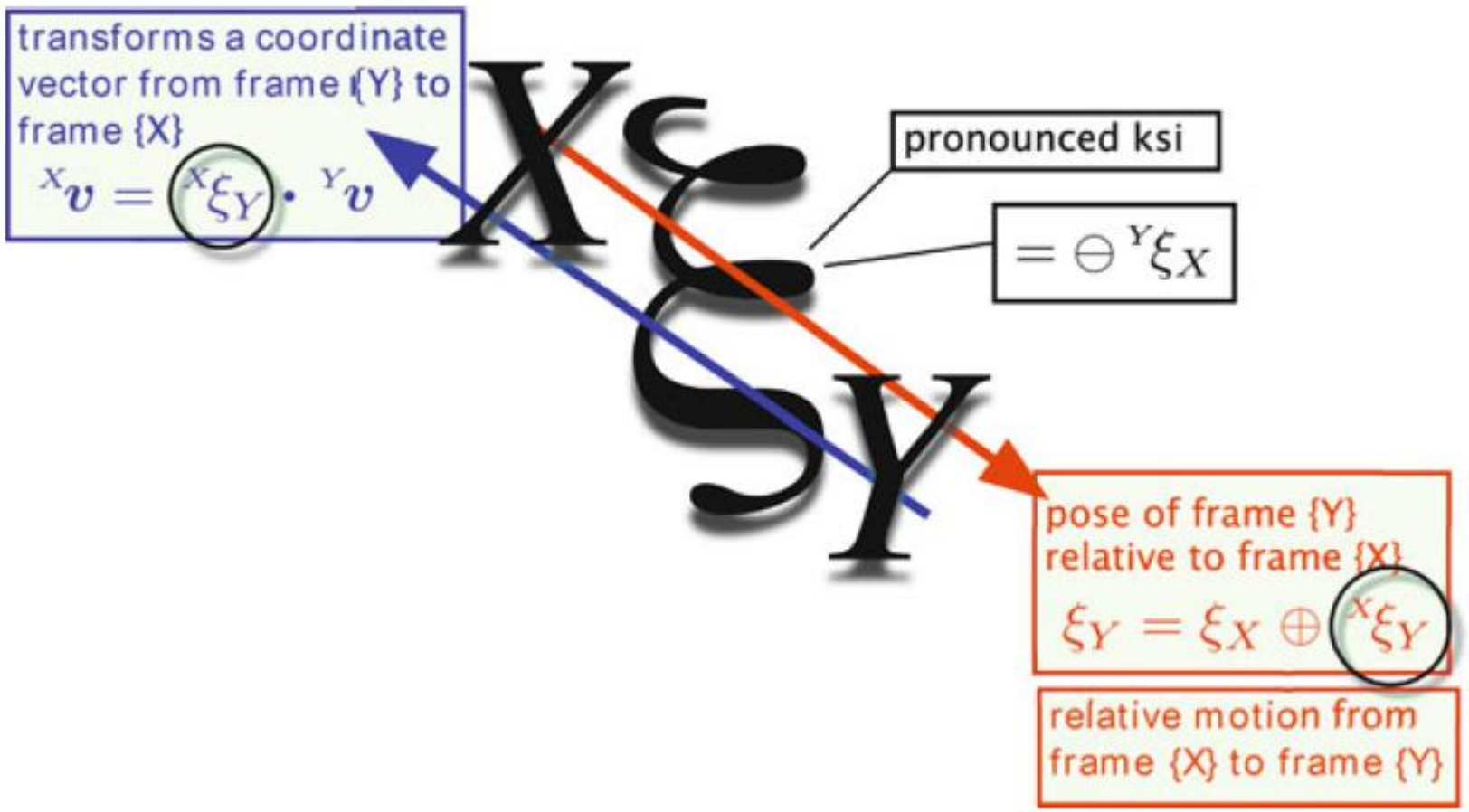


$$^{X}\boldsymbol{\xi}_{Z} = {}^{X}\boldsymbol{\xi}_{Y} \oplus {}^{Y}\boldsymbol{\xi}_{Z}$$



Peter Corke - Robotics, Vision and Control_ Fundamental Algorithms in Python, 3rd Edition. 146-Springer (2023)

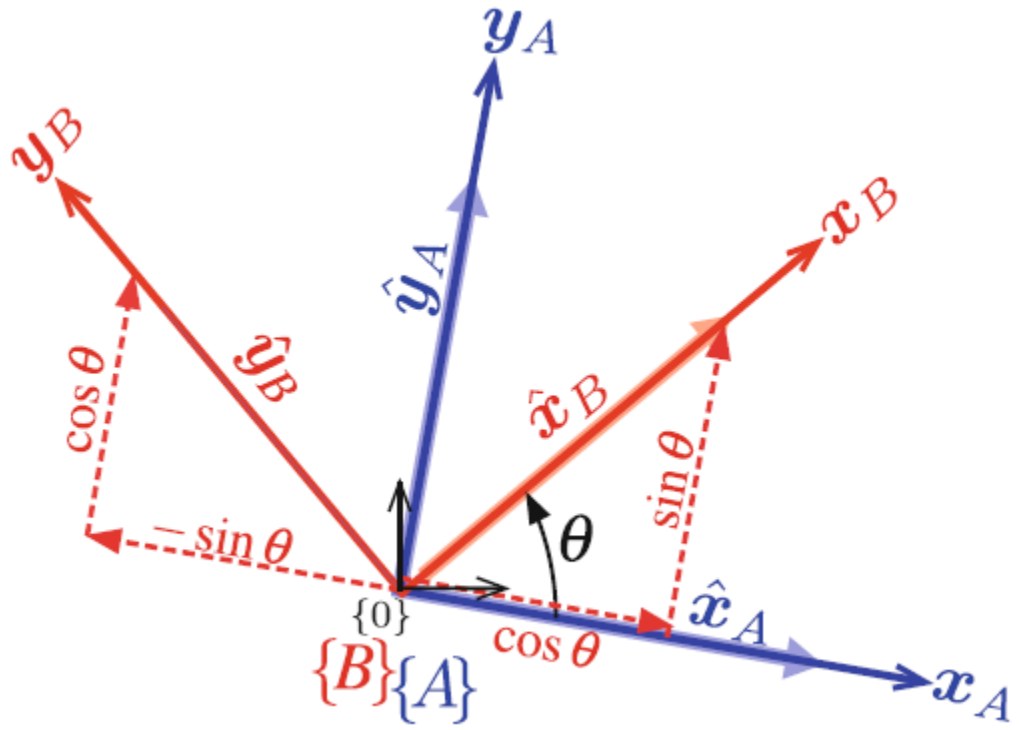
Relative Poses



Peter Corke - Robotics, Vision and Control_ Fundamental Algorithms in Python, 3rd Edition. 146-Springer (2023)

2D Rotation Matrix

${}^{A}R_{B}(\theta)$ transforms a coordinate vector from frame {B} to {A}.

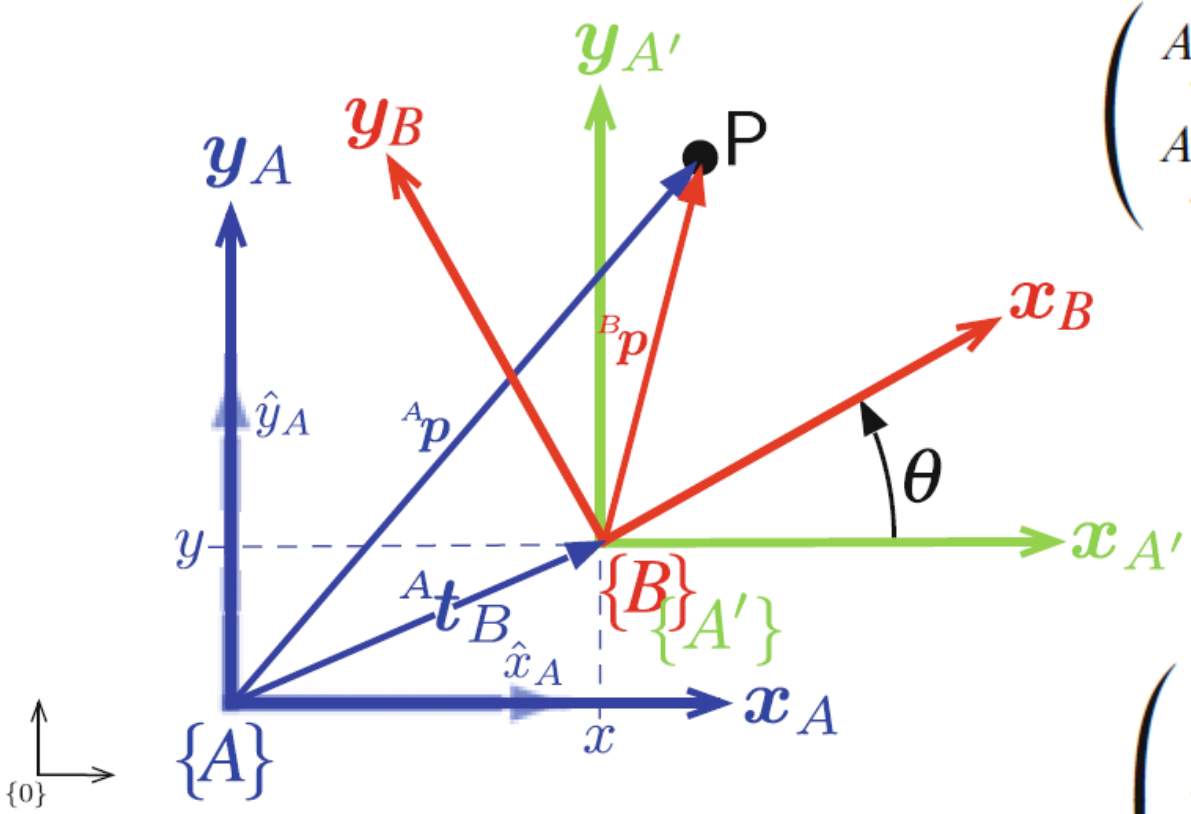


$$
{}^{A}\mathbf{R}_{B}(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}
$$

$$
\begin{pmatrix} {}^{A}p_x \\ {}^{A}p_y \end{pmatrix} = {}^{A}\mathbf{R}_{B}(\theta) \begin{pmatrix} {}^{B}p_x \\ {}^{B}p_y \end{pmatrix}
$$

2D Homogeneous Transformation Matrix

Rotated and translated coordinate frames



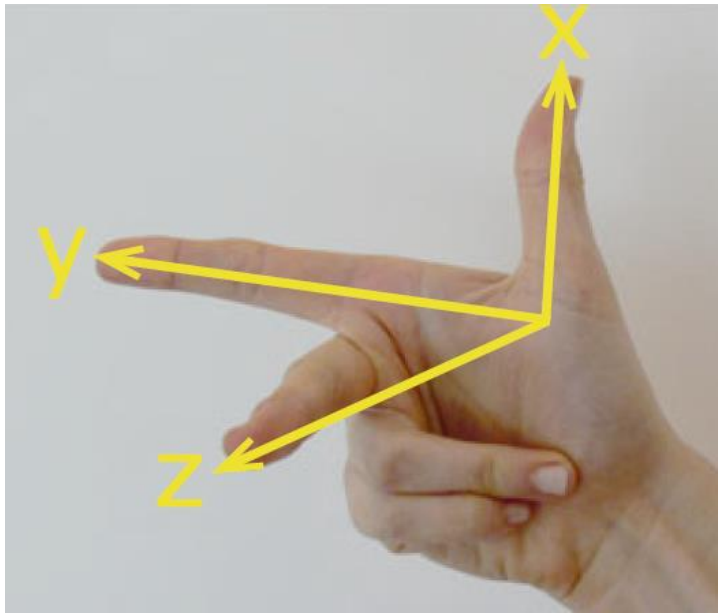
$$\begin{pmatrix} {}^Ax \\ {}^Ay \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} {}^Bx \\ {}^By \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix}$$

Homogenous Coordinate

$$\begin{pmatrix} {}^Ax \\ {}^Ay \\ 1 \end{pmatrix} = \begin{pmatrix} {}^A\mathbf{R}_B(\theta) & {}^A\boldsymbol{t}_B \\ \mathbf{0}_{1\times 2} & 1 \end{pmatrix} \begin{pmatrix} {}^Bx \\ {}^By \\ 1 \end{pmatrix}$$

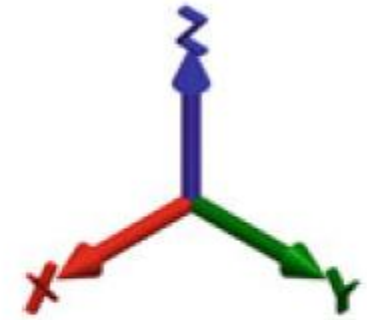
3D Coordinate Frames

Right-hand rule



$$\boldsymbol{p} = x\hat{\boldsymbol{x}} + y\hat{\boldsymbol{y}} + z\hat{\boldsymbol{z}}$$

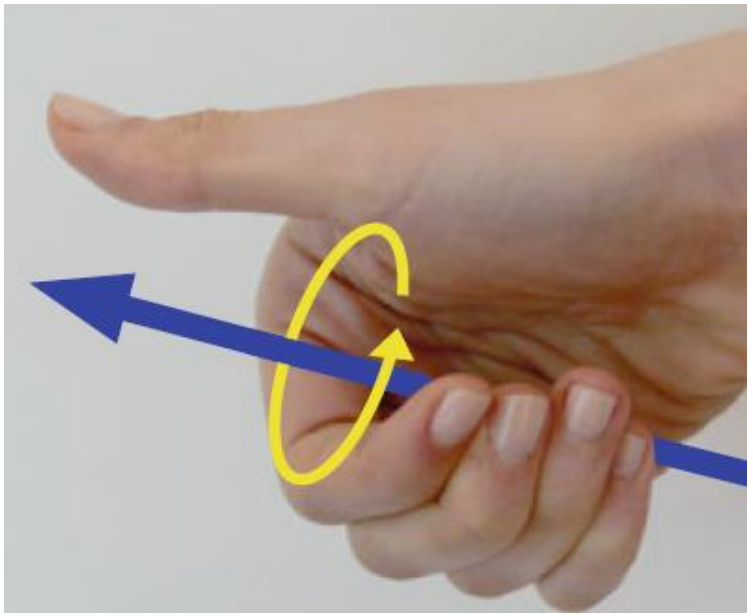
Rotation of Frames



a Original

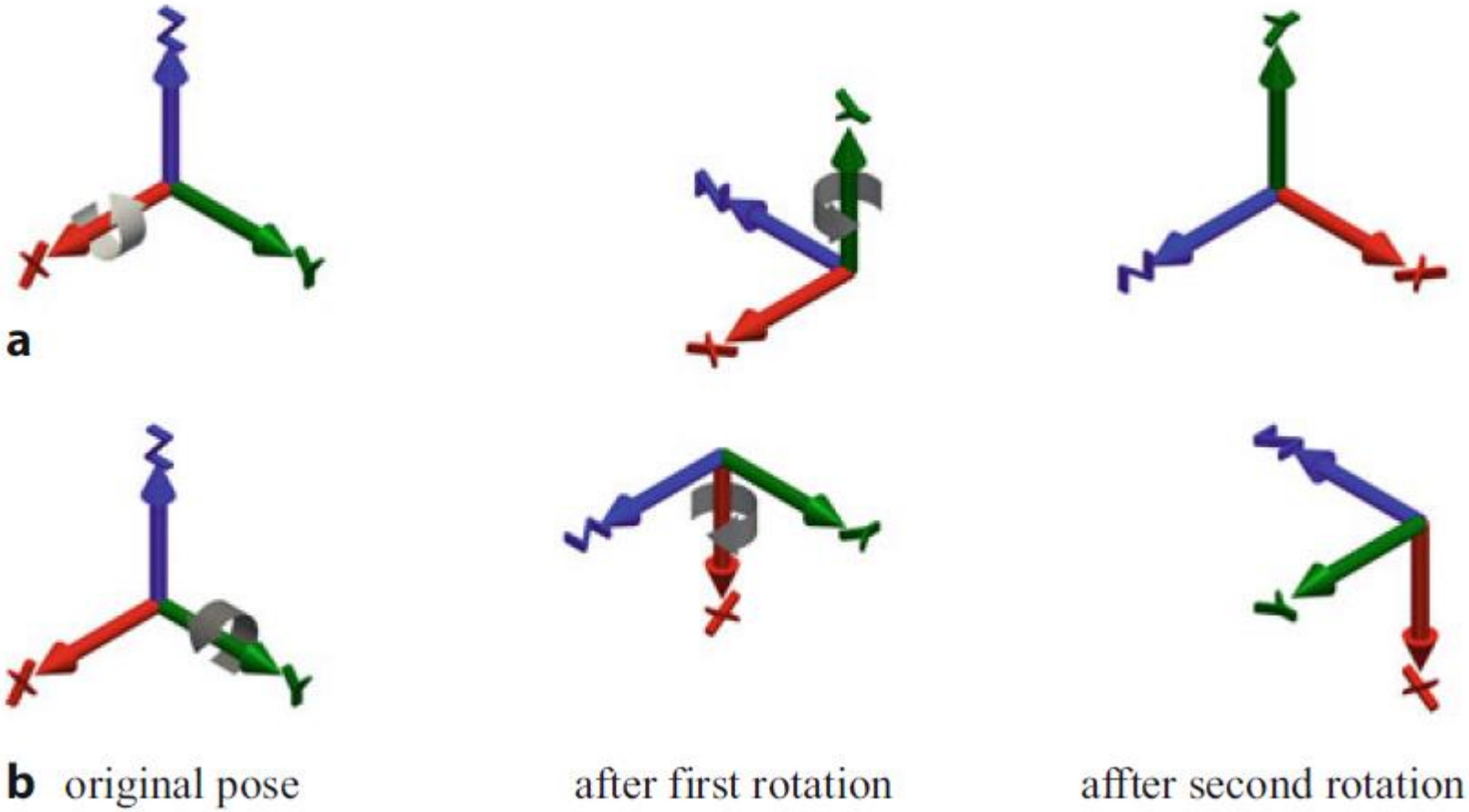
b $\frac{\pi}{2}$ about x-axis

Rotation About a Vector



3D Coordinate Frames

Noncommutativity of rotation



a

b original pose

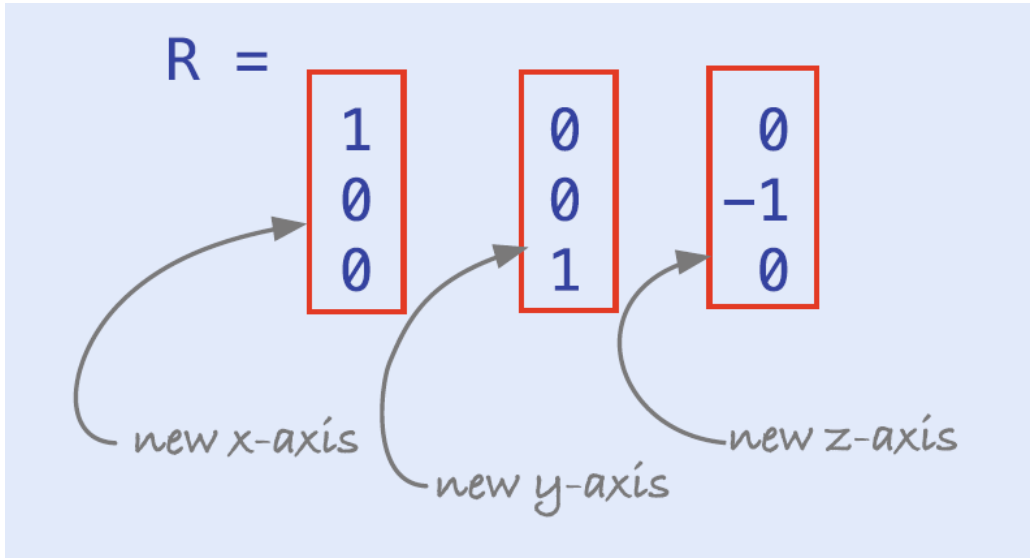
after first rotation

affter second rotation

3D Rotation Matrix

Frame B to Frame A

$$\begin{pmatrix} {}^{A}p_x \\ {}^{A}p_y \\ {}^{A}p_z \end{pmatrix} = {}^{A}\mathbf{R}_B \begin{pmatrix} {}^{B}p_x \\ {}^{B}p_y \\ {}^{B}p_z \end{pmatrix}$$



$$\mathbf{R}_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix}$$

$$\mathbf{R}_y(\theta) = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix}$$

$$\mathbf{R}_z(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Three-Angle Representations

Euler Angles

It is an ambiguous term: XYX, XZX, YXY, YZY, ZXZ, or ZYZ.
(e.g., mechanical dynamics: ZYZ)

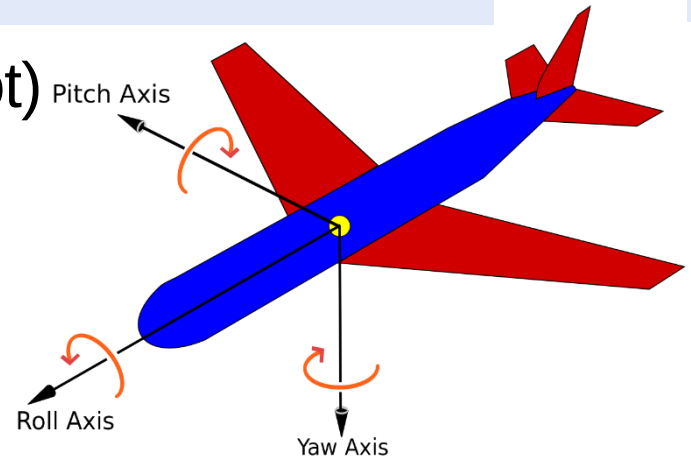
$$\mathbf{R}(\phi,\theta,\psi) = \mathbf{R}_z(\phi)\,\mathbf{R}_y(\theta)\,\mathbf{R}_z(\psi)$$

Cardan Angles

Roll (α)-pitch(β)-yaw(γ) angle ambiguity: **XYZ**, XZY, YZX, YXZ, ZXY, or **ZYX**

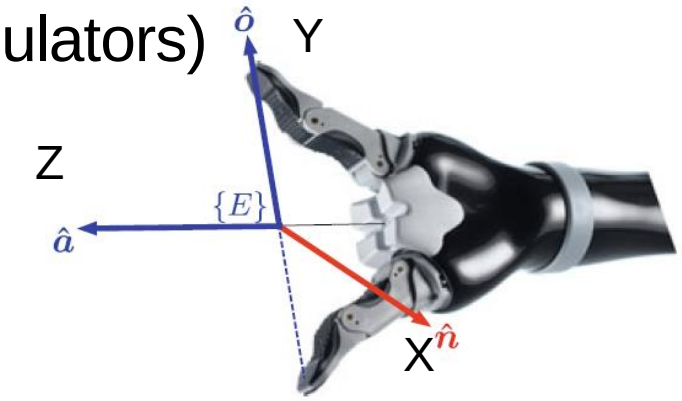
$$\mathbf{R}(\alpha,\beta,\gamma) = \mathbf{R}_z(\gamma)\,\mathbf{R}_y(\beta)\,\mathbf{R}_x(\alpha)$$

ZYX (Mobile Robot)



$$\mathbf{R}(\alpha,\beta,\gamma) = \mathbf{R}_x(\gamma)\,\mathbf{R}_y(\beta)\,\mathbf{R}_z(\alpha)$$

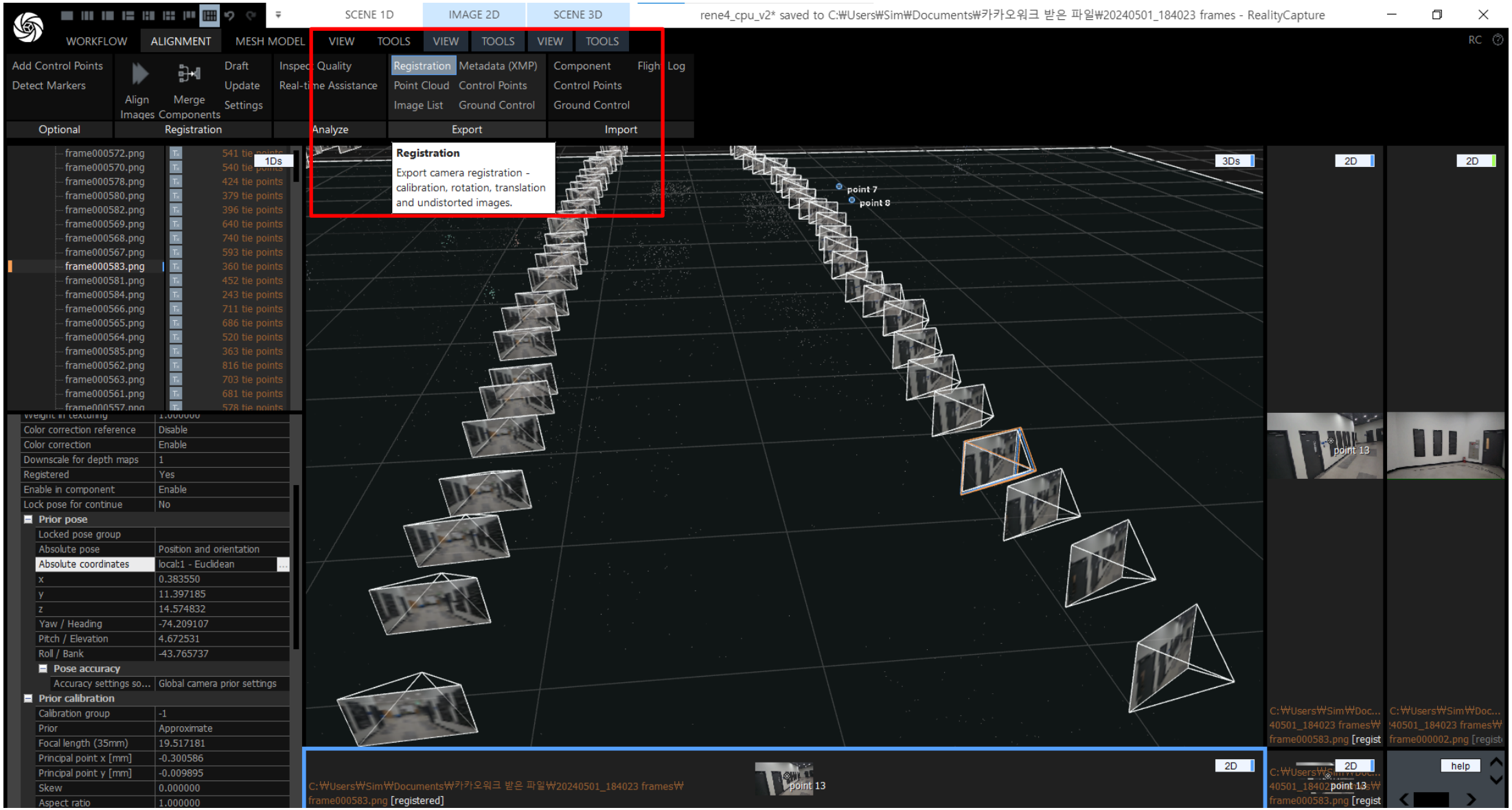
XYZ (Robot manipulators)



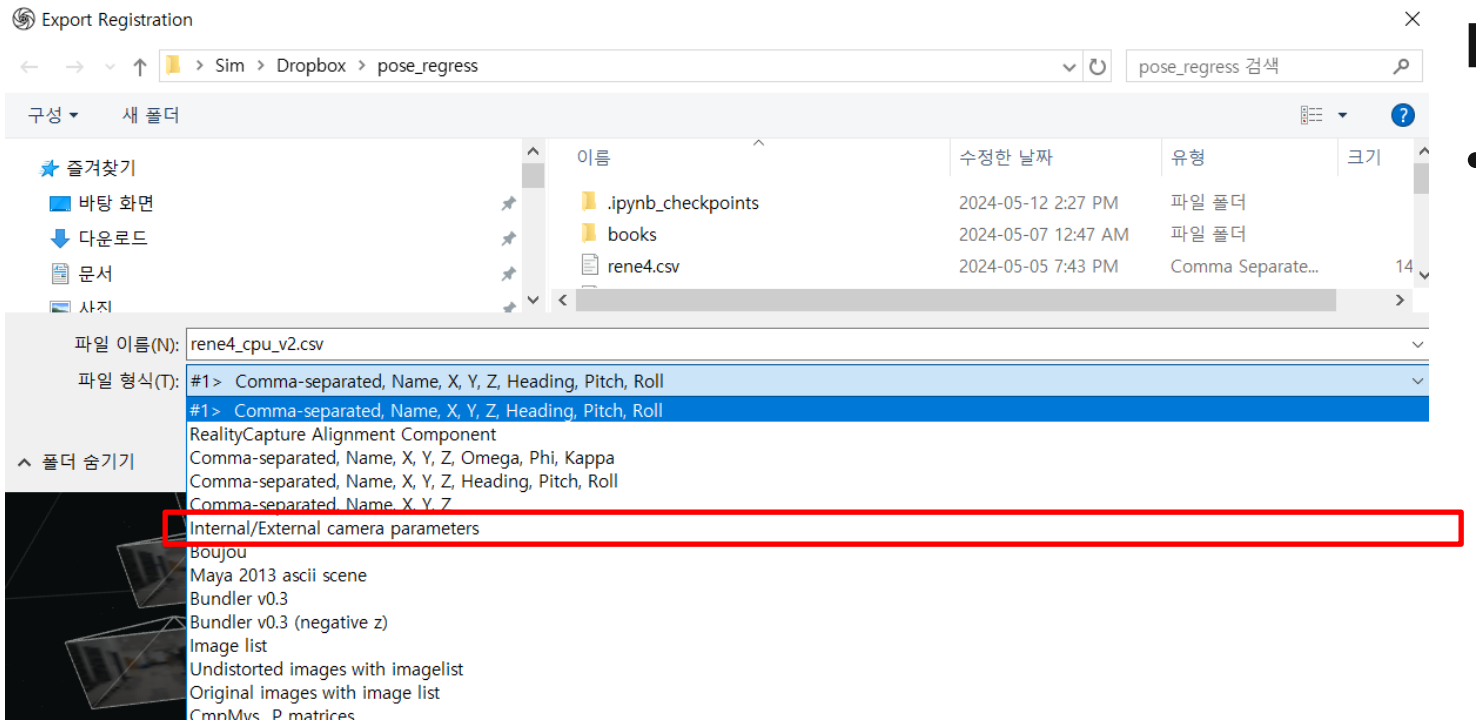
Reality Capture:
Camera Registration Export

심주용
숙명여자대학교
기계시스템학부

Camera Pose Export

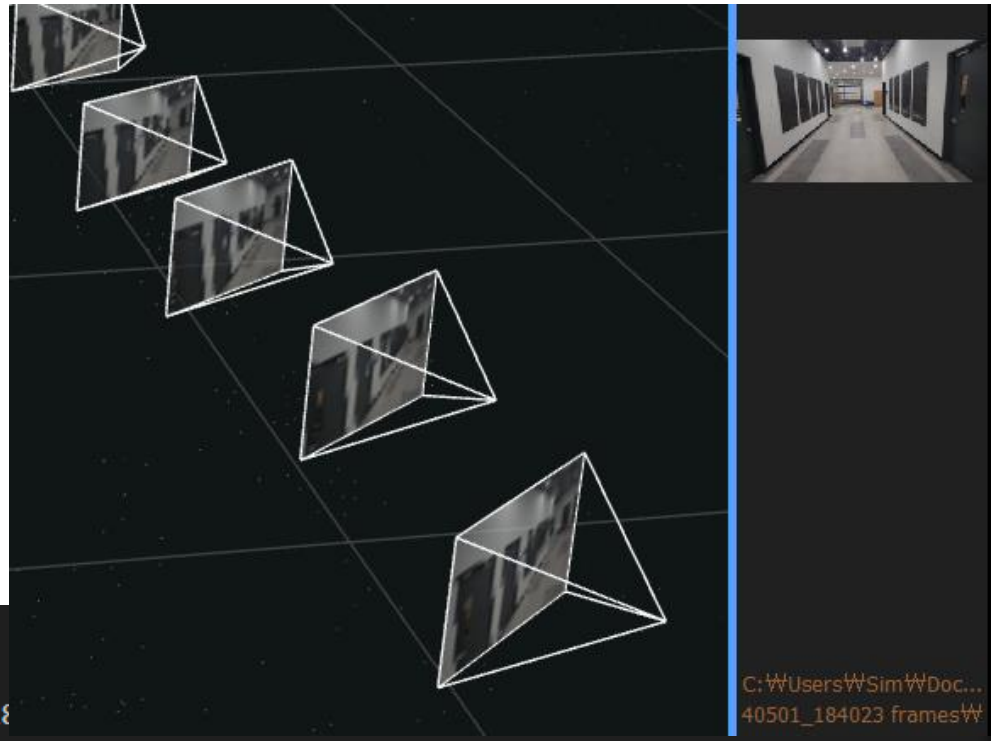


Camera Pose Export



Heading, Pitch, Roll

- Need to convert degree to radian



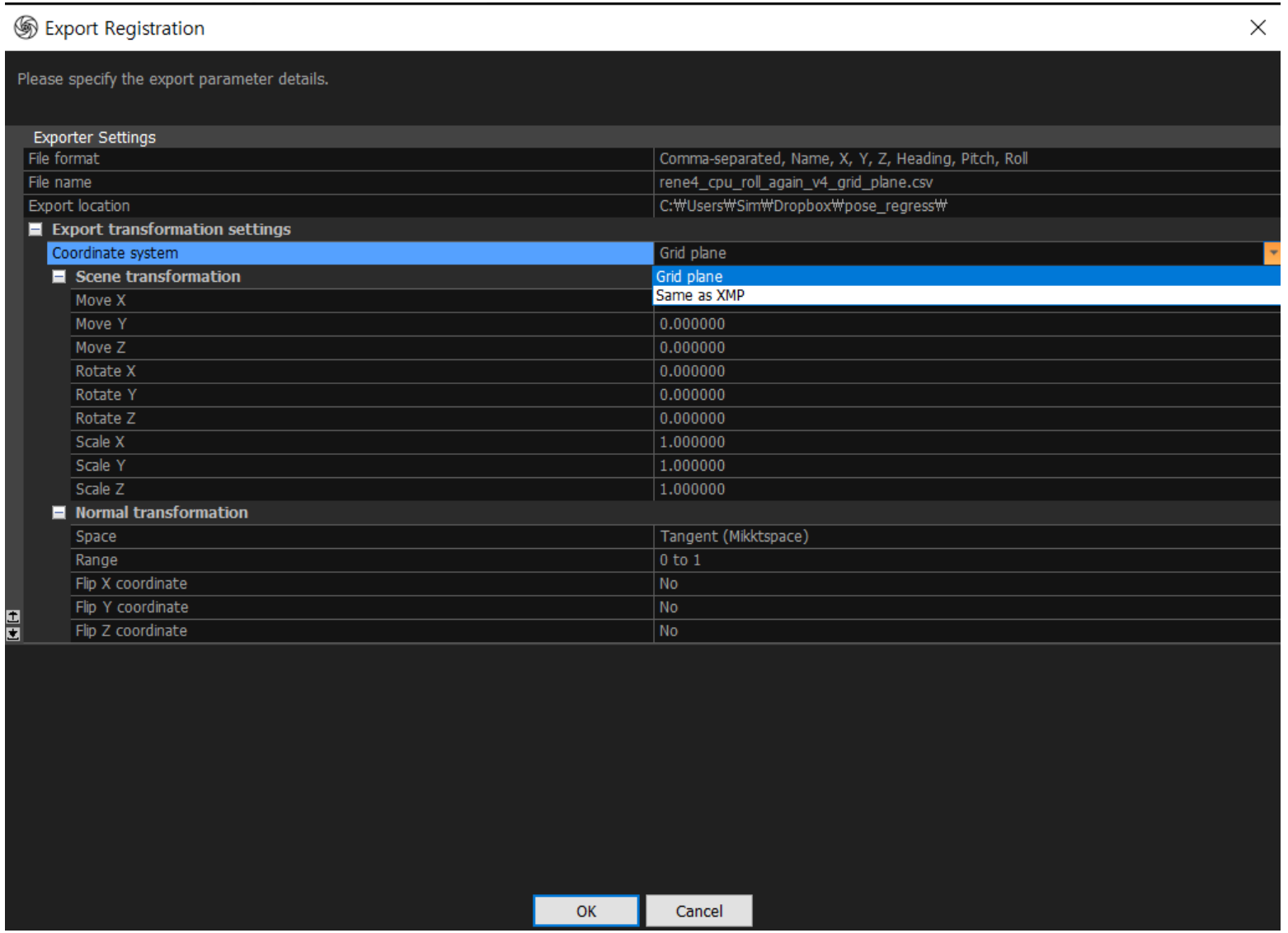
C: > Users > Sim > Dropbox > pose_regress > rene4_cpu_int_ext.csv

```
1  #name,x,y,alt,heading,pitch,roll,f,px,py,k1,k2,k3,k4,t1,t2
2  frame000073.png,8.242849089577009,28.82153654769631,1.284793488706929,-82.6622
3  frame000074.png,8.315626630247712,28.85726568726199,1.288443275987522,-86.19818085054767,84.09898026873439,-0.6112592333128565,17.50929
4  frame000075.png,8.312852163184834,28.91612268480087,1.283564208941917,-90.04488021130334,84.07744165974337,0.1633952638090864,17.537489
5  frame000076.png,8.305590132259928,28.99511588273064,1.294929081835999,-93.95549829531235,83.93654827736646,-0.3707139301559295,17.56345
6  frame000077.png,8.274867646137592,29.04837894455181,1.272630691041517,-99.49061741927936,83.97289255022996,-1.417861592209114,17.566207
7  frame000078.png,8.280084540254842,29.10011674068884,1.269980754305722,-104.659575574821,83.64214773414518,-2.720630270808134,17.5906871
8  frame000079.png,8.278448578202875,29.15957692675736,1.263355990255013,-105.4539430841437,83.31040610656544,-0.3568720927559521,17.50542
```

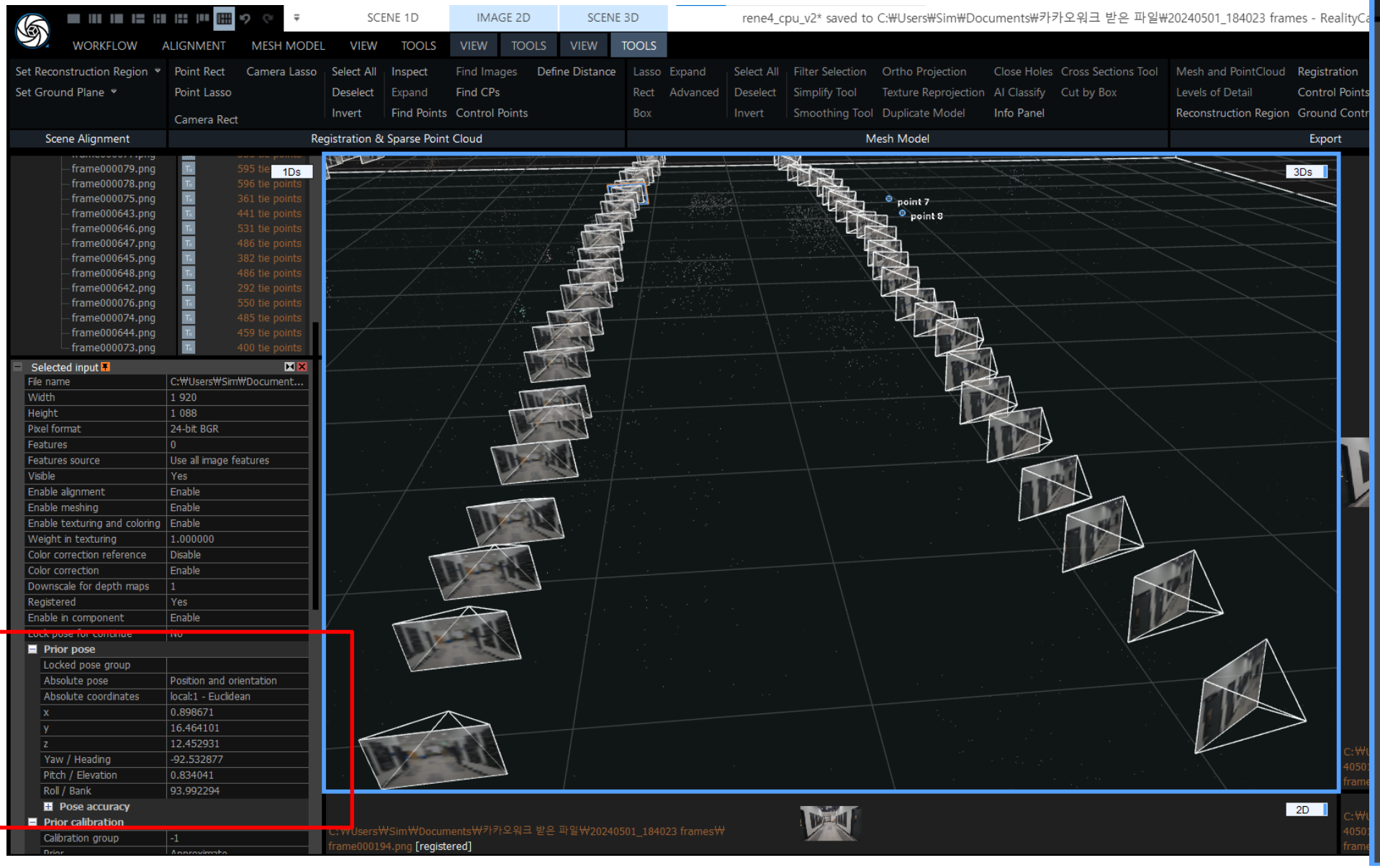
Camera Pose Export

Coordinate System

- Set Ground Plane 기준
- 독립적인 XMP 파일 기준



참고


SCENE 1D
IMAGE 2D
SCENE 3D
WORKFLOW
ALIGNMENT
MESH MODEL
VIEW
TOOLS
Scene Alignment
Registration & Sparse Point Cloud
Mesh Model
Export
point 7
point 8
Selected input
Prior pose
Prior calibration
Pose accuracy
Absolute coordinates
local:1 - Euclidean
x
0.465066
y
11.939553
z
13.226693
Yaw / Heading
-93.694961
Pitch / Elevation
0.376243
Roll / Bank
96.501750
Accuracy settings so...
Global camera prior settings
Calibration group
-1
Prior
Approximate
Focal length (35mm)
17.889311
Principal point x [mm]
0.207206
Principal point y [mm]
-0.003440
Skew
0.000000
Aspect ratio
1.000000
Prior lens distortion
Lens group
Camera model
Brown3
Radial 1
-0.006907
Radial 2
0.047186
Radial 3
-0.018298
Relative position uncertainty
Uncertainty X
0.000082
Uncertainty Y
0.000109
Uncertainty Z
0.000080
Units
Meter
Registration calibration
Registration lens distortion

Project Milestone

1. 기말 프로젝트 마일스톤 발표: ~5/27 (월) 16:00~17:00

- 발표: 4분 이내 + 2분 질의응답
- 데이터 획득 내용 및 학습 파이프 라인 구축 내용 설명
- 초기 모델 학습시킨 결과 포함

2. 기말 프로젝트 최종 발표: 6/19 (수) 16:00~17:50 예정

- 발표: 12분 이내 + 3분 질의응답
- 프로젝트 내용 및 결과 웹페이지 게시 (google sites, github pages, notion 등 이용)