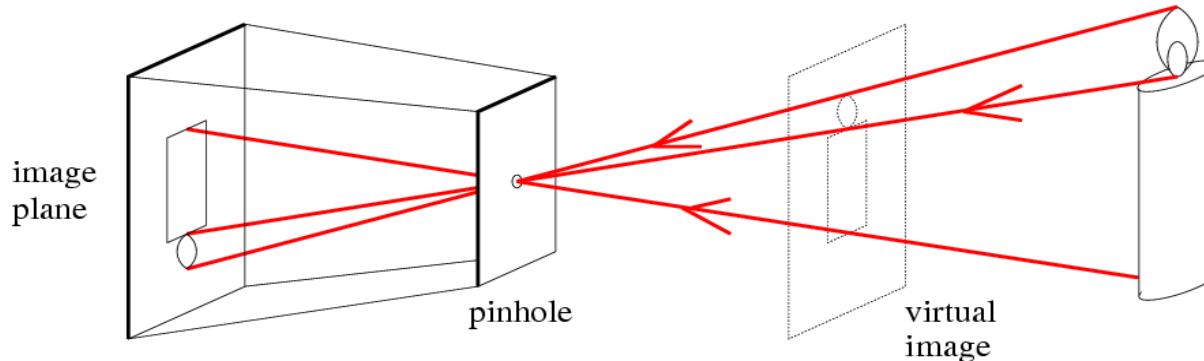


Lecture 1.2

Pose in 2D and 3D

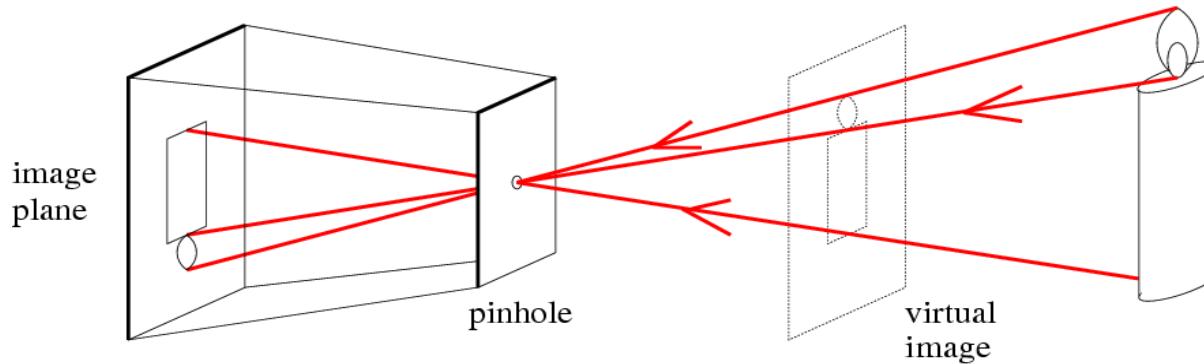
Thomas Opsahl

Motivation



- For the pinhole camera, the correspondence between observed 3D points in the world and 2D points in the captured image is given by straight lines through a common point (pinhole)
- This correspondence can be described by a mathematical model known as “*the perspective camera model*” or “*the pinhole camera model*”
- This model can be used to describe the imaging geometry of many modern cameras, hence it plays a central part in computer vision

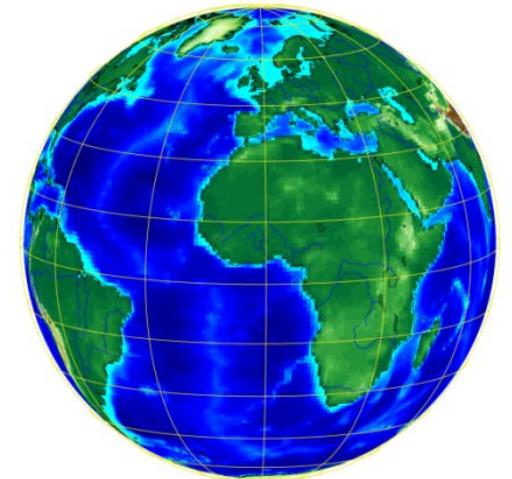
Motivation



- Before we can study the perspective camera model in detail, we need to expand our mathematical toolbox
- We need to be able to mathematically describe the position and orientation of the camera relative to the world coordinate frame
- Also we need to get familiar with some basic elements of projective geometry, since this will make it MUCH easier to describe and work with the perspective camera model

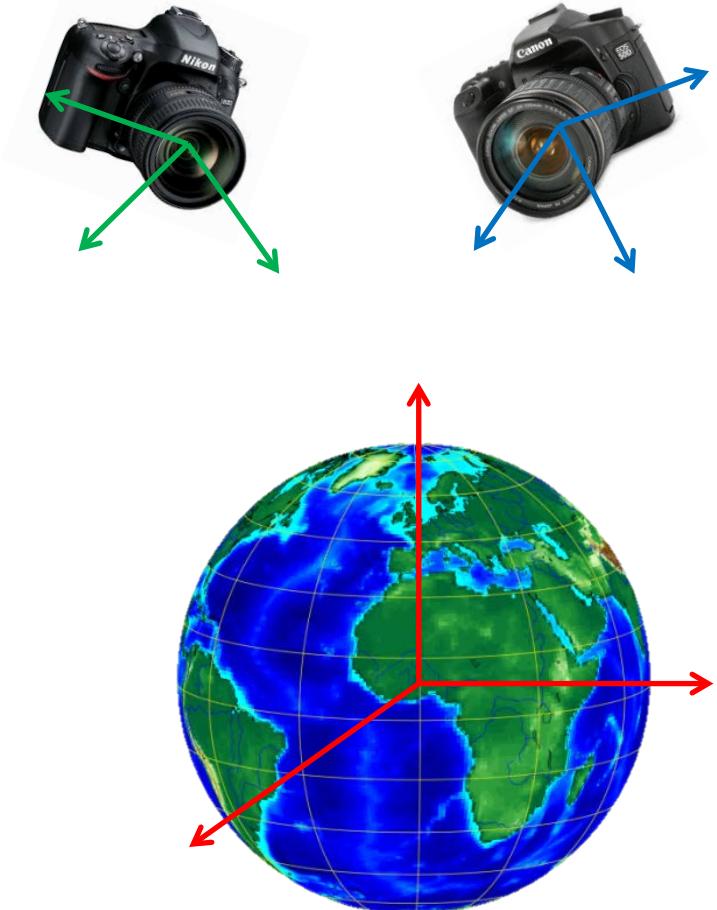
Introduction

- In computer vision we are often interested in the geometrical aspects of imaging
 - Points in the world \leftrightarrow pixels in an image
 - Pixels in image 1 \leftrightarrow pixels in image 2
- In order to express and study geometrical problems related to imaging, we first need to know how to describe the position and orientation of objects
- Position and orientation is together known as *pose*

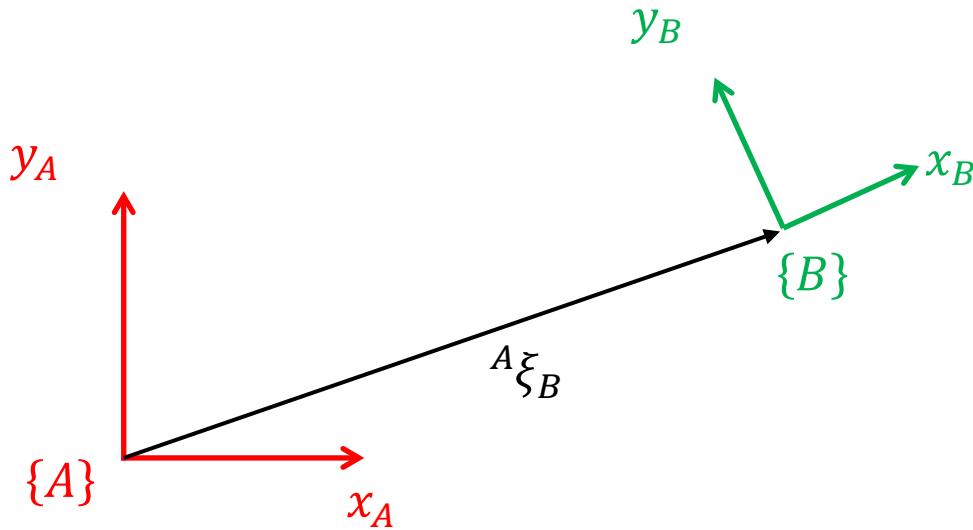


Introduction

- By representing all relevant objects by coordinate frames, it is possible to numerically represent the pose of one object relative to another
- In the following we will look at pose
 - General properties
 - Representation in 2D
 - Representation in 3D

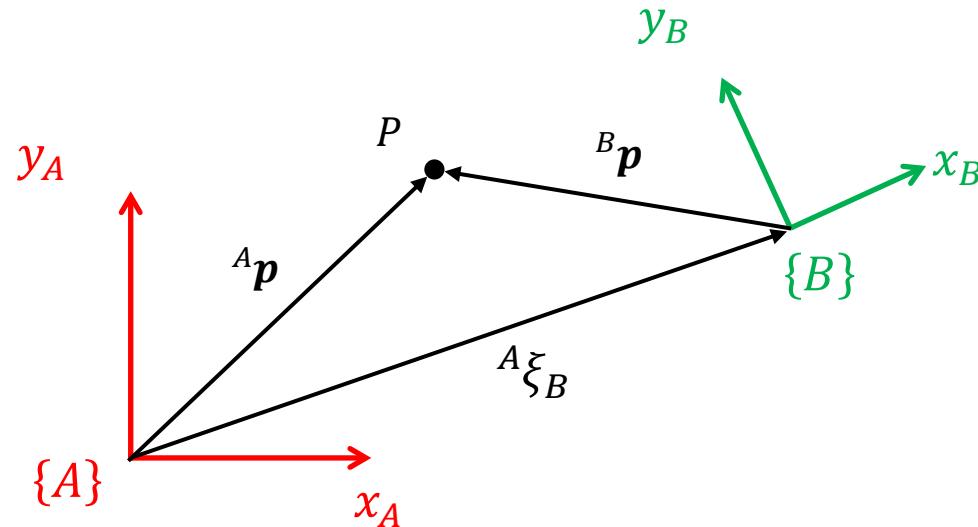


General properties



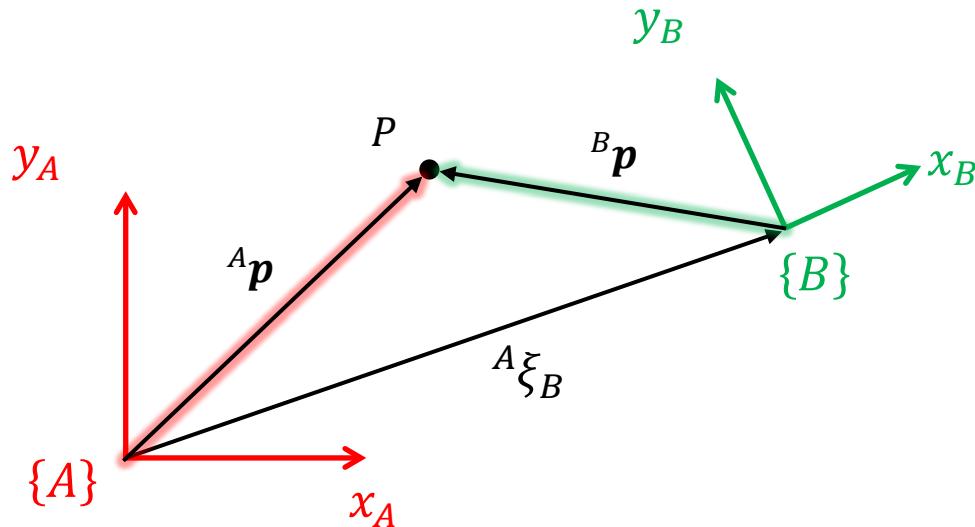
- Let us denote by ${}^A\xi_B$ the pose of frame $\{B\}$ relative to frame $\{A\}$
- We can think of ${}^A\xi_B$ as the translation and rotation required in order to make $\{A\}$ coincide with $\{B\}$

General properties



- A point P can be described with respect to either frame
- These descriptions are related by the pose
- Formally we write this as ${}^A\boldsymbol{p} = {}^A\xi_B \cdot {}^B\boldsymbol{p}$

General properties



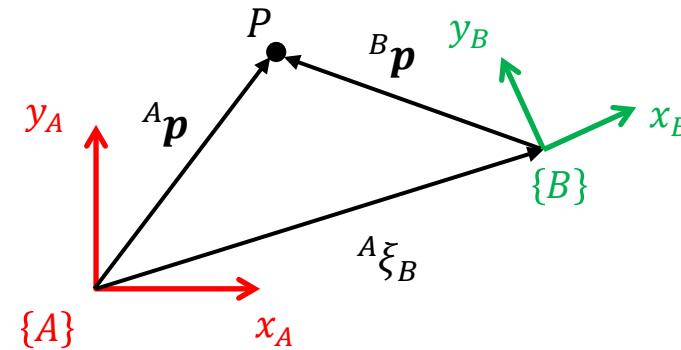
Note that ${}^A\boldsymbol{p}$ and ${}^B\boldsymbol{p}$ are different vectors!!!

- A point P can be described with respect to either frame
- These descriptions are related by the pose
- Formally we write this as ${}^A\boldsymbol{p} = {}^A\xi_B \cdot {}^B\boldsymbol{p}$

General properties

- Action on points

$${}^A\boldsymbol{p} = {}^A\xi_B \cdot {}^B\boldsymbol{p}$$



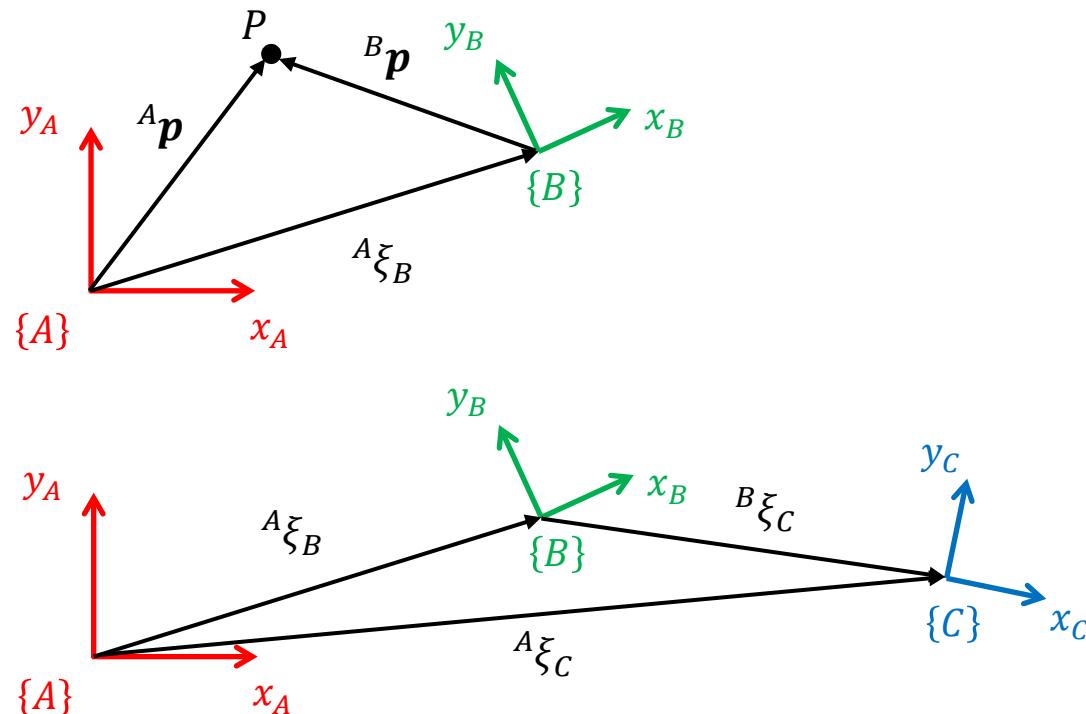
General properties

- Action on points

$${}^A\boldsymbol{p} = {}^A\xi_B \cdot {}^B\boldsymbol{p}$$

- Composition

$${}^A\xi_C = {}^A\xi_B \oplus {}^B\xi_C$$



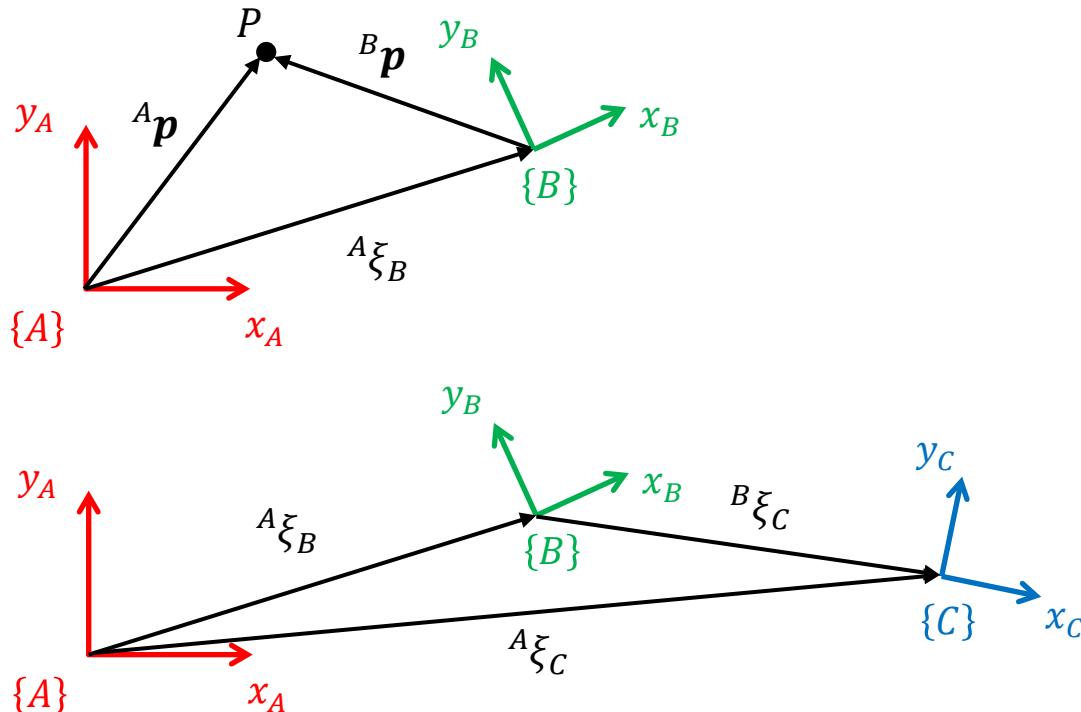
General properties

- Action on points

$${}^A\boldsymbol{p} = {}^A\xi_B \cdot {}^B\boldsymbol{p}$$

- Composition

$${}^A\xi_C = {}^A\xi_B \oplus {}^B\xi_C$$



What about ${}^A\xi_B \oplus {}^B\xi_A$?

General properties

- Action on points

$${}^A\boldsymbol{p} = {}^A\xi_B \cdot {}^B\boldsymbol{p}$$

- Composition

$${}^A\xi_C = {}^A\xi_B \oplus {}^B\xi_C$$

- Inverse

$$\ominus {}^A\xi_B = {}^B\xi_A$$

- Neutral element

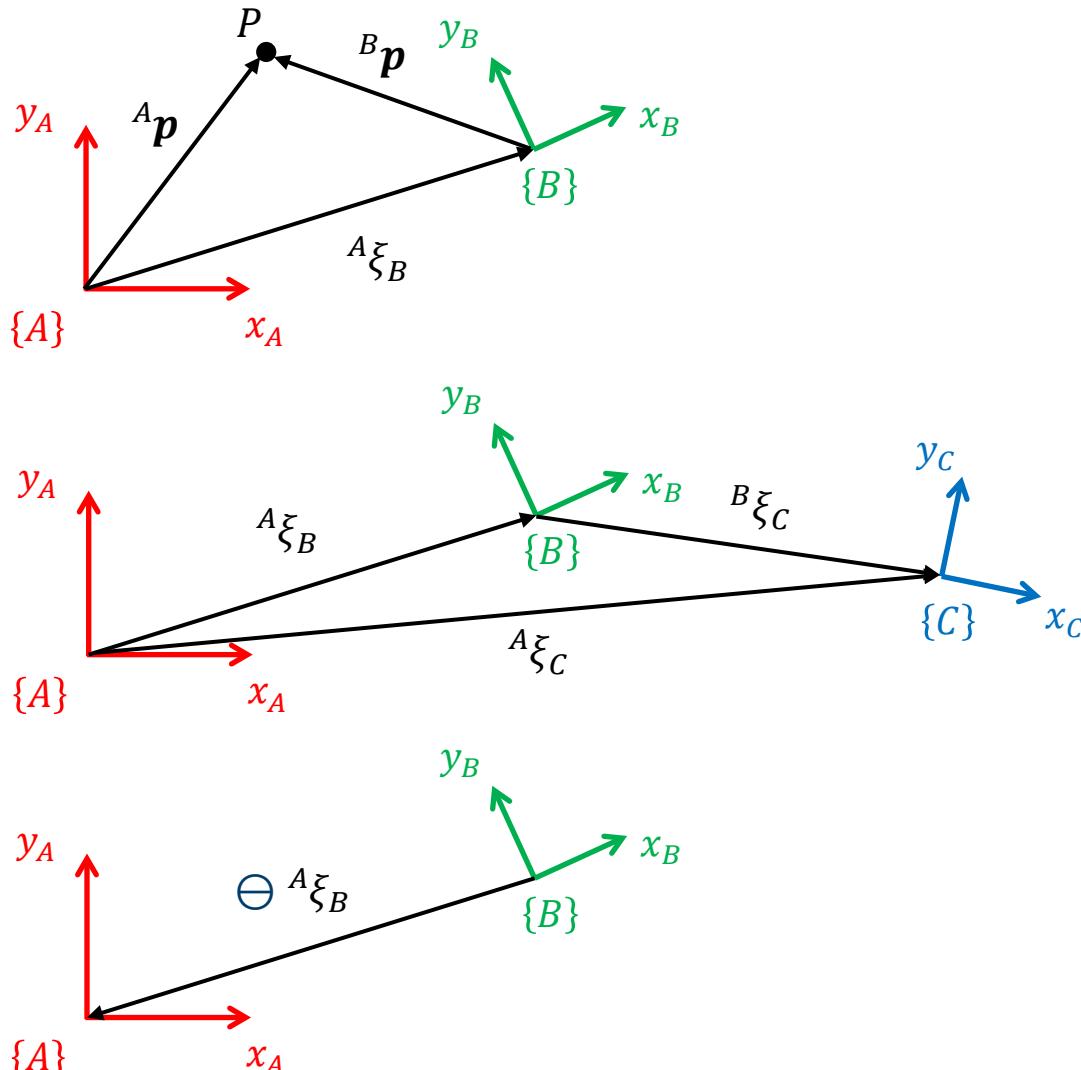
$$0 = {}^A\xi_A$$

$$0 \oplus \xi = \xi$$

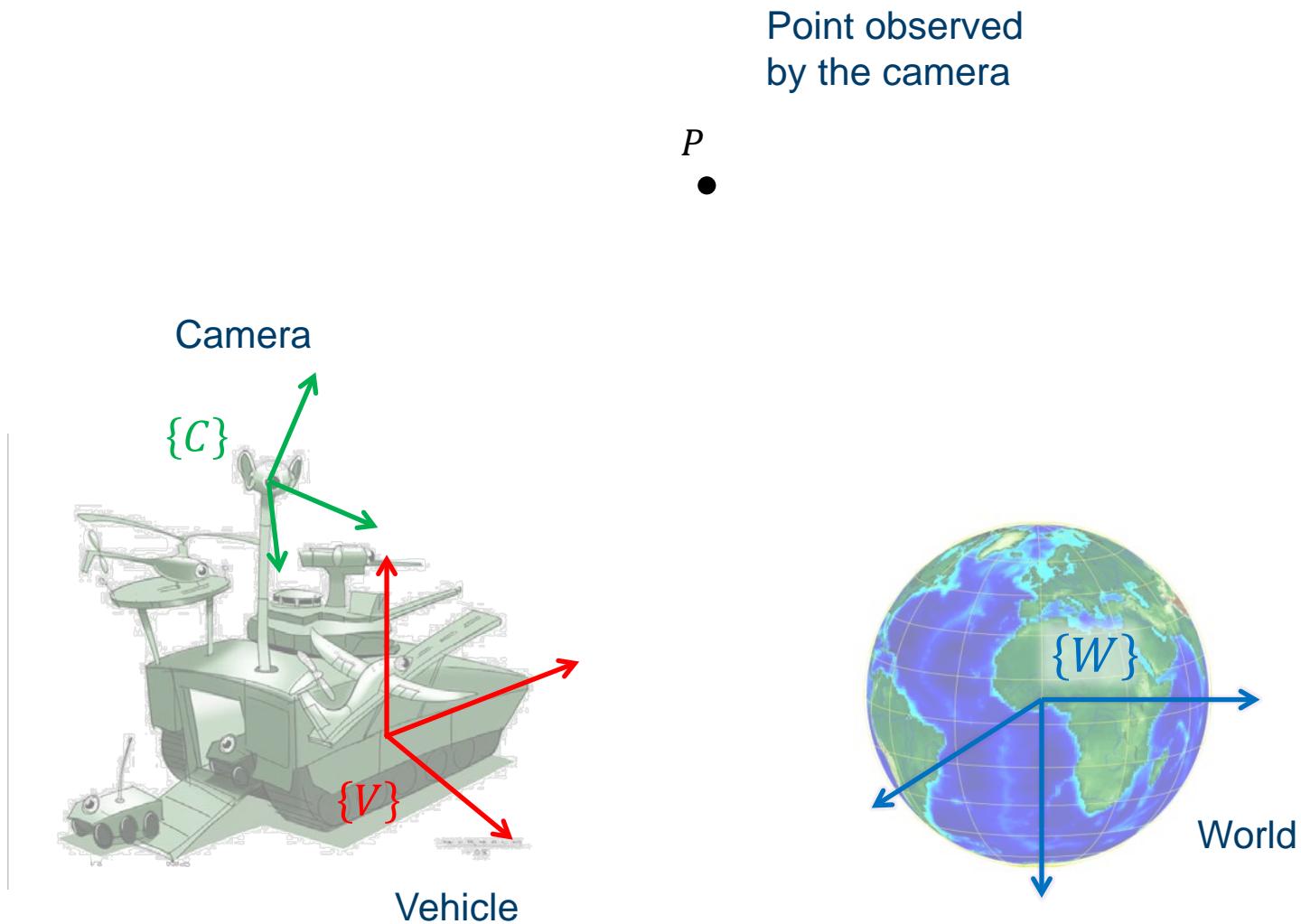
$$\xi \ominus 0 = \xi$$

$$\xi \ominus \xi = 0$$

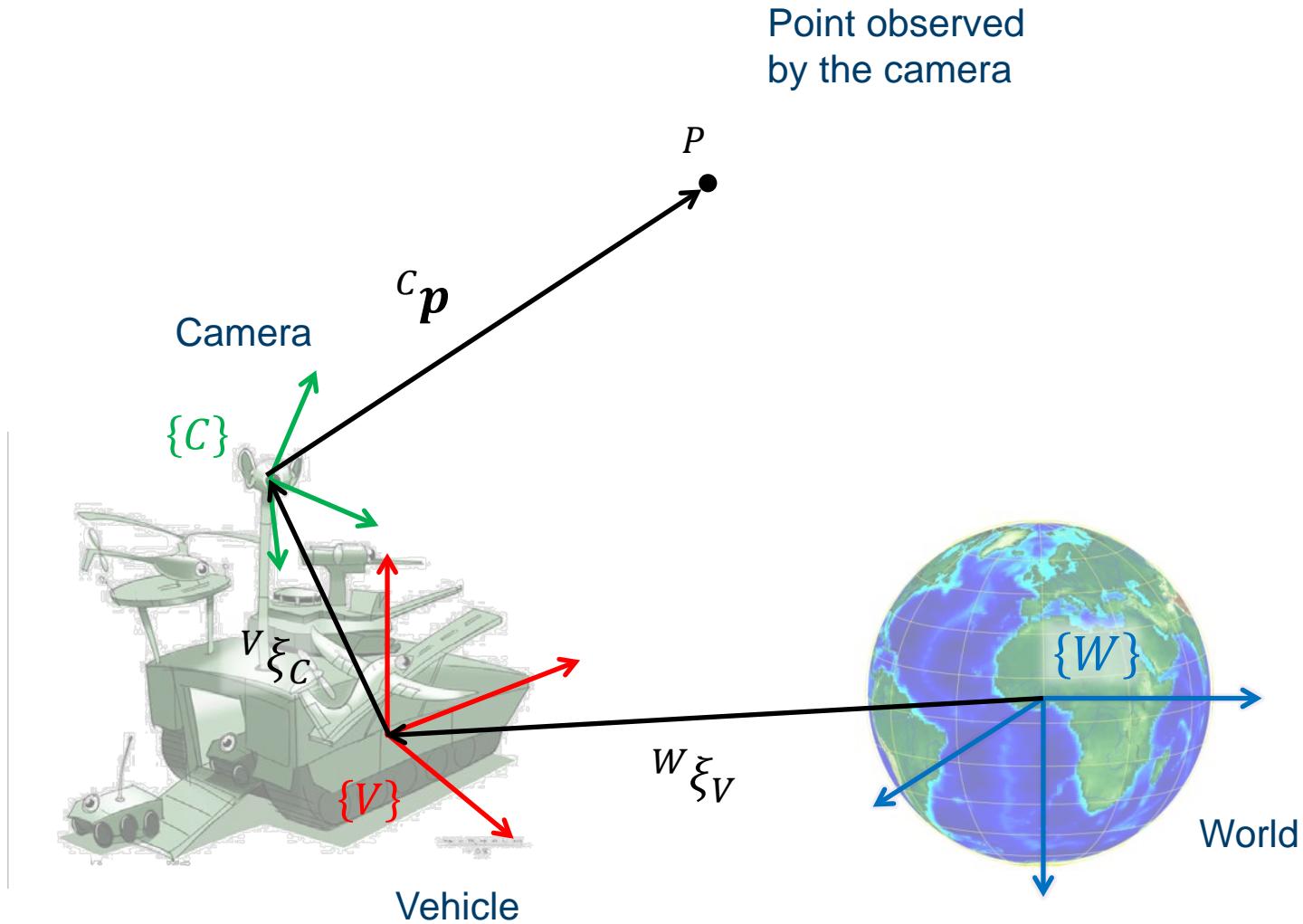
$$\ominus \xi \oplus \xi = 0$$



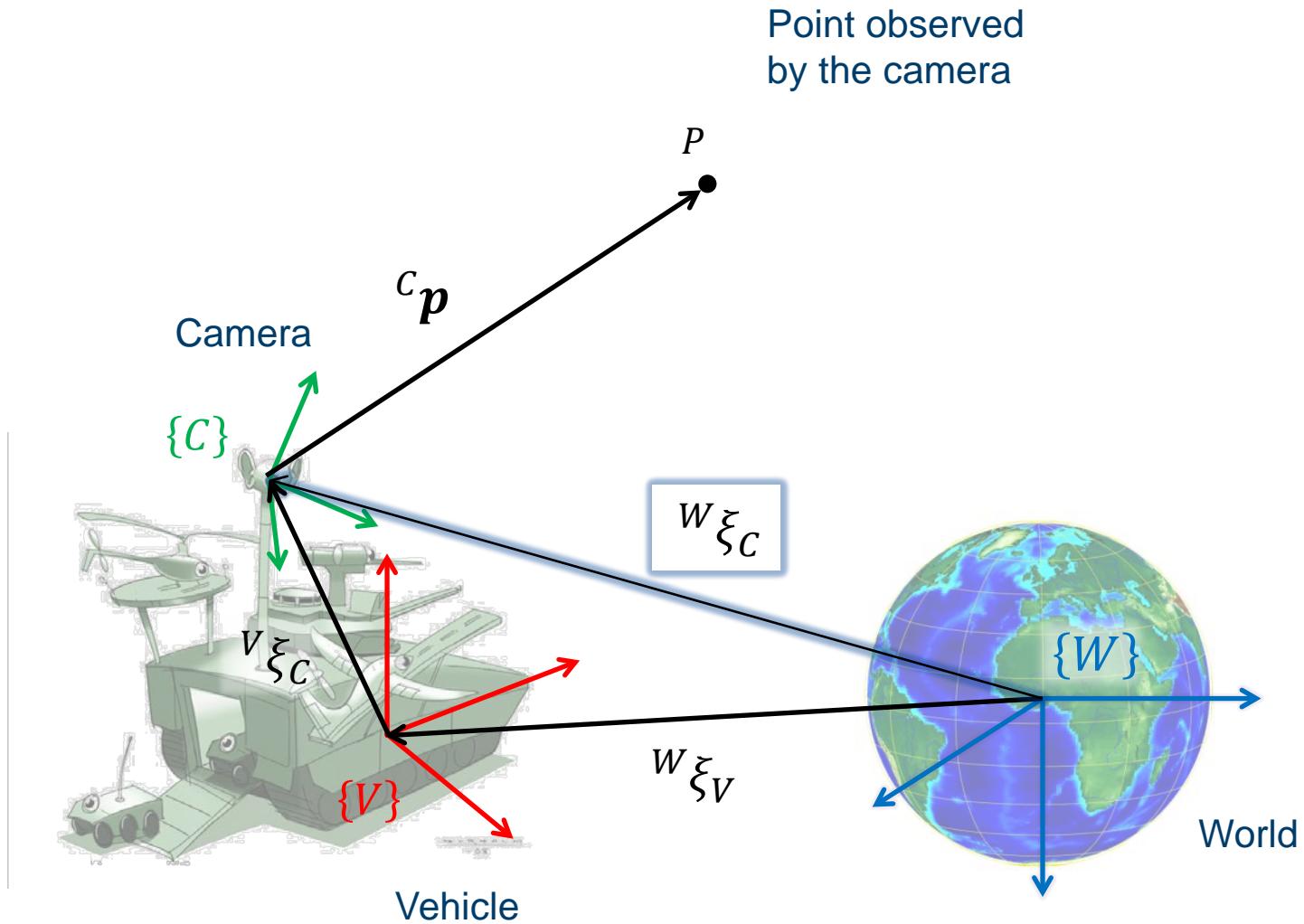
Example



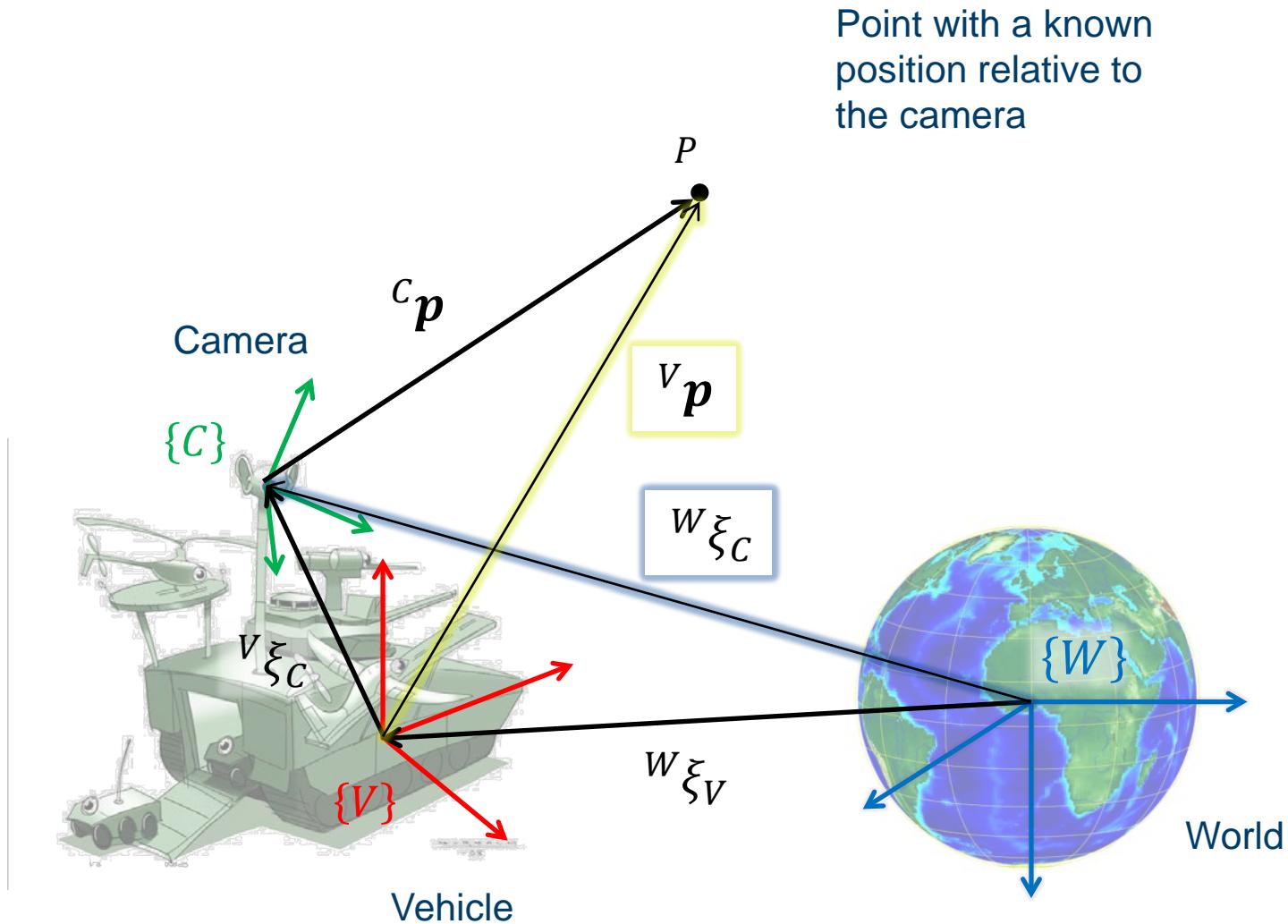
Example



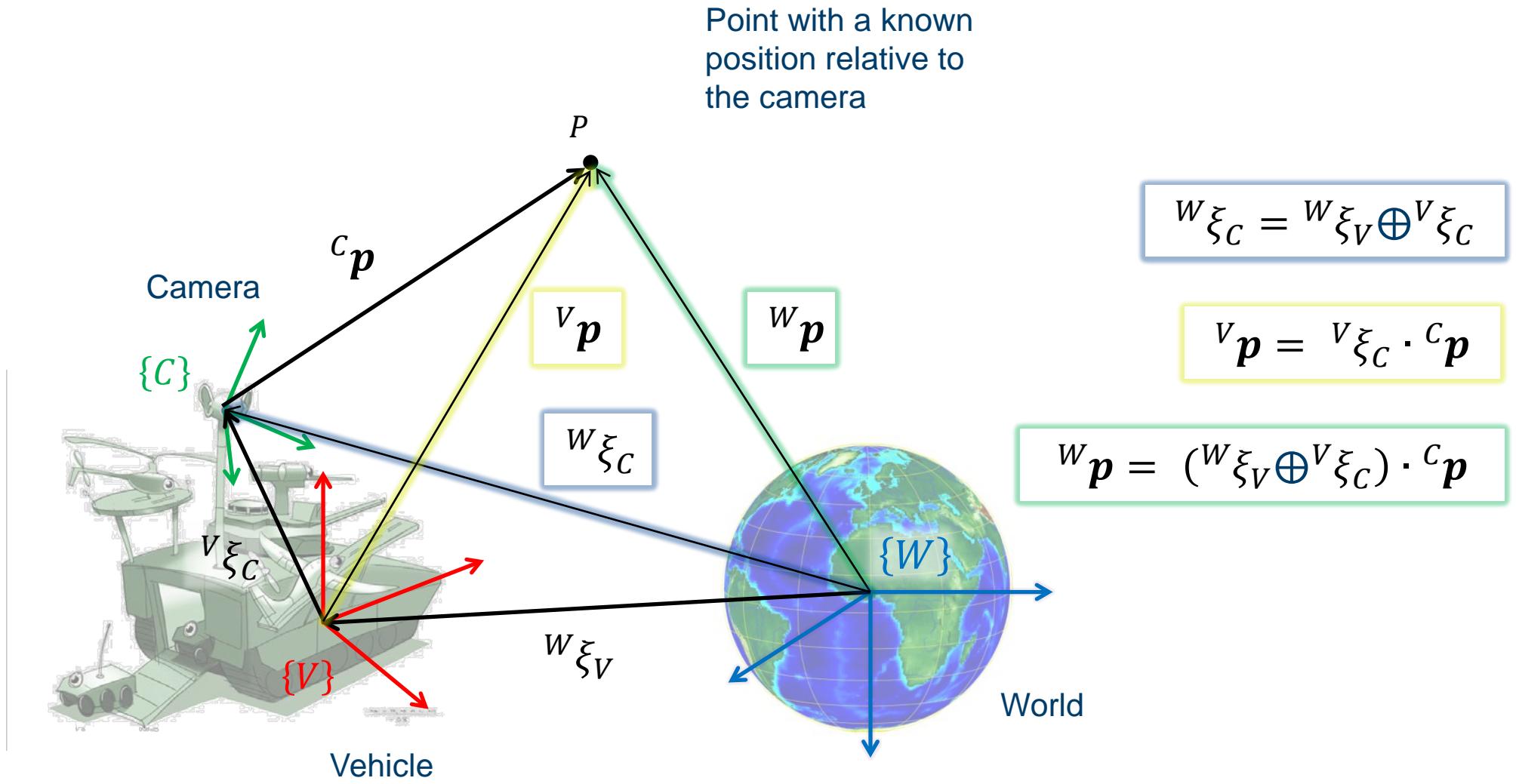
Example



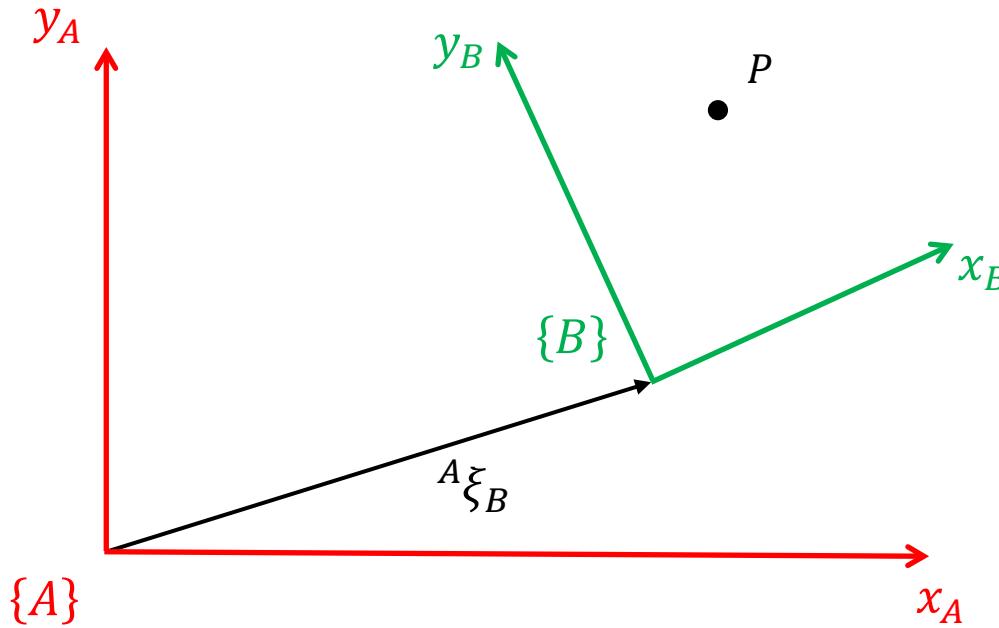
Example



Example

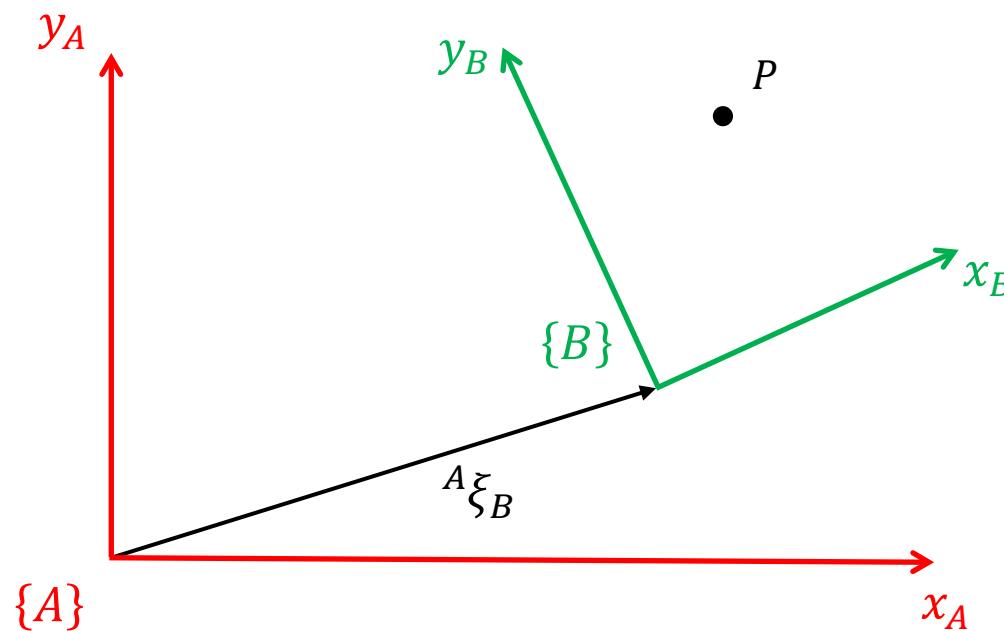


Investigating pose in 2D



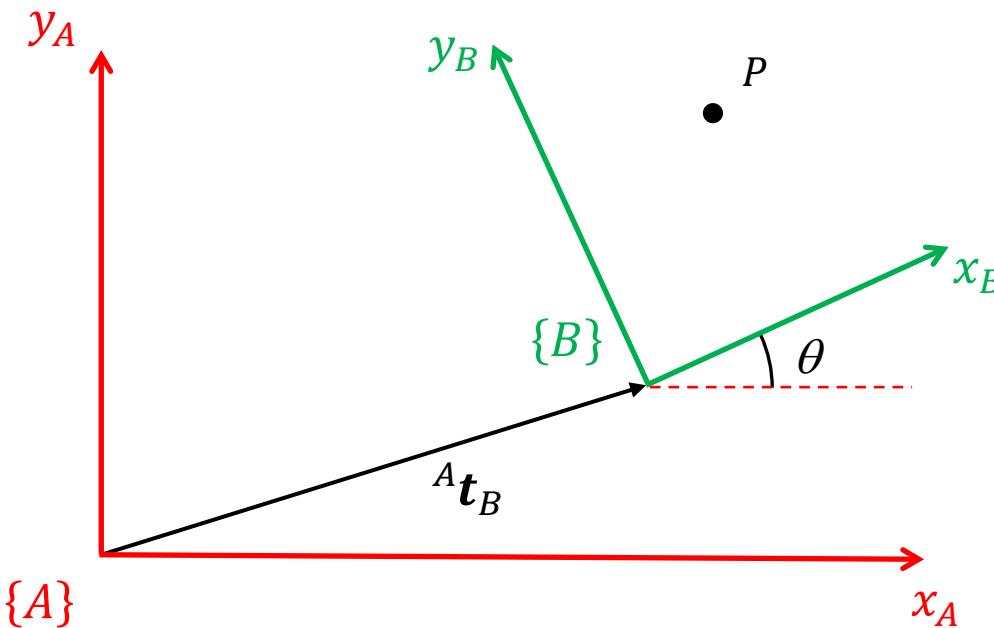
- Given two 2D frames, $\{A\}$ and $\{B\}$, how can we represent the pose ${}^A\xi_B$?
- We need a numerical representation in order to compute ${}^A\mathbf{p} = {}^A\xi_B \cdot {}^B\mathbf{p}$

Investigating pose in 2D



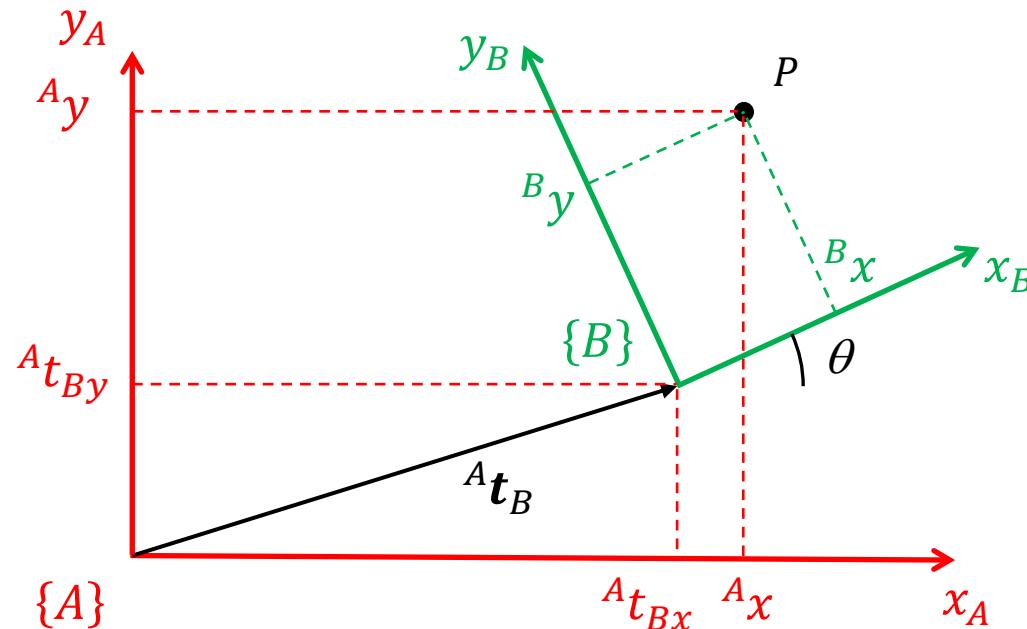
- Recall that we can think of ${}^A\xi_B$ as the translation and rotation required in order to make $\{A\}$ coincide with $\{B\}$

Investigating pose in 2D



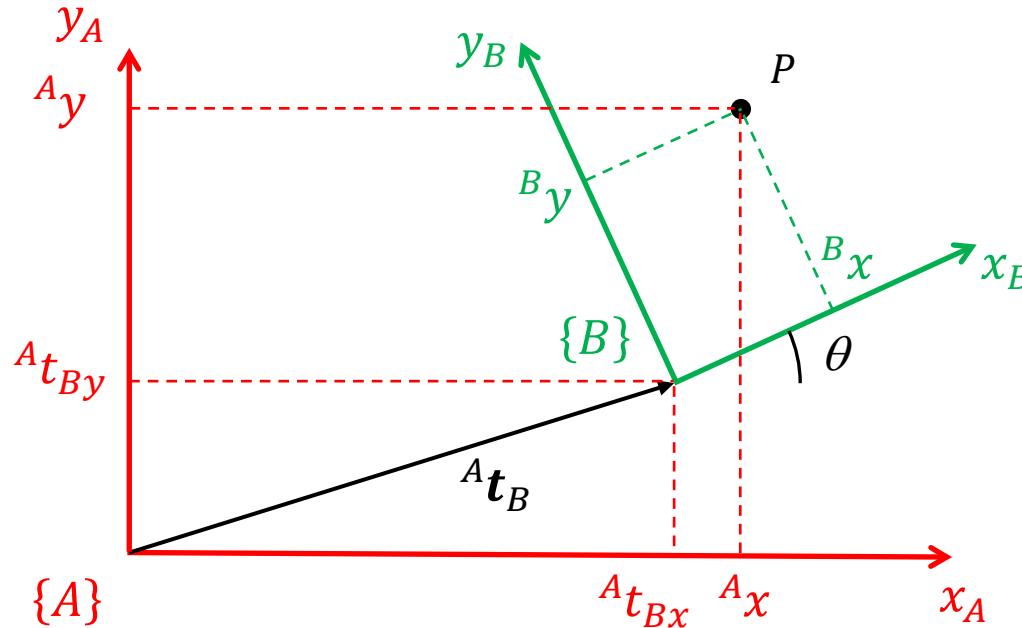
- Recall that we can think of ${}^A \xi_B$ as the translation and rotation required in order to make $\{A\}$ coincide with $\{B\}$
- To coincide with $\{B\}$, $\{A\}$ must undergo a translation ${}^A t_B$ and a rotation by an angle θ

Investigating pose in 2D



- Let ${}^A p = [{}^A x, {}^A y]^T$, ${}^B p = [{}^B x, {}^B y]^T$ and ${}^A t_B = [{}^A t_{Bx}, {}^A t_{By}]^T$

Investigating pose in 2D

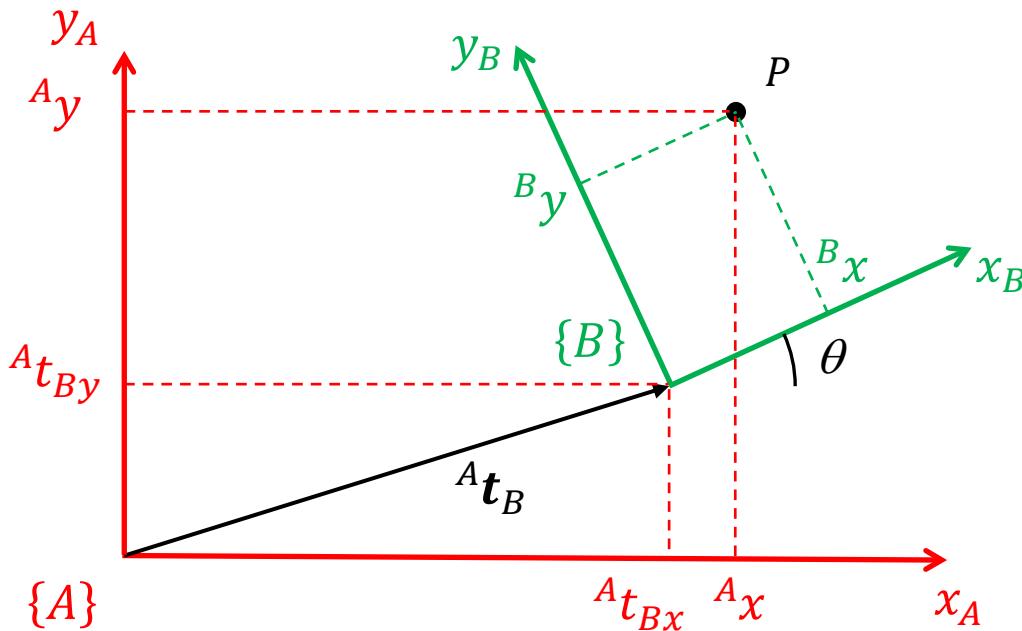


- Let ${}^A \mathbf{p} = [{}^A x, {}^A y]^T$, ${}^B \mathbf{p} = [{}^B x, {}^B y]^T$ and ${}^A \mathbf{t}_B = [{}^A t_{Bx}, {}^A t_{By}]^T$
- From the figure we can see that

$${}^A x = {}^A t_{Bx} + {}^B x \cos \theta - {}^B y \sin \theta$$

$${}^A y = {}^A t_{By} + {}^B x \sin \theta + {}^B y \cos \theta$$

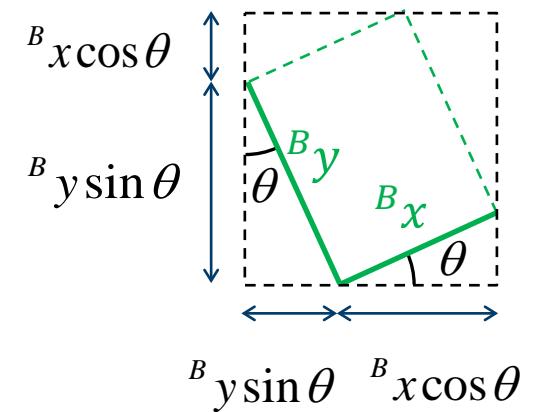
Investigating pose in 2D



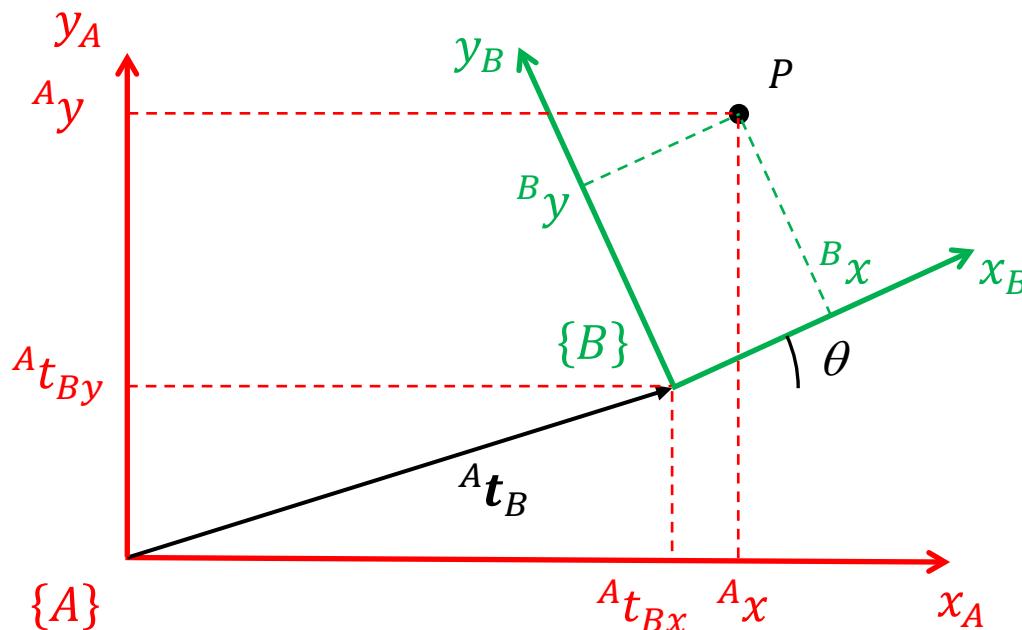
- Let ${}^A\mathbf{p} = [{}^A\mathbf{x}, {}^A\mathbf{y}]^T$, ${}^B\mathbf{p} = [{}^B\mathbf{x}, {}^B\mathbf{y}]^T$ and ${}^A\mathbf{t}_B = [{}^A\mathbf{t}_{Bx}, {}^A\mathbf{t}_{By}]^T$
- From the figure we can see that

$${}^A\mathbf{x} = {}^A\mathbf{t}_{Bx} + {}^B\mathbf{x} \cos \theta - {}^B\mathbf{y} \sin \theta$$

$${}^A\mathbf{y} = {}^A\mathbf{t}_{By} + {}^B\mathbf{x} \sin \theta + {}^B\mathbf{y} \cos \theta$$



Investigating pose in 2D



- In matrix form

$$\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix} = \begin{bmatrix} {}^A t_{Bx} \\ {}^A t_{By} \end{bmatrix} + \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix}$$
$${}^A \mathbf{p} = {}^A \mathbf{t}_B + {}^A R_B {}^B \mathbf{p}$$

Investigating pose in 2D

$$\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix} = \begin{bmatrix} {}^A t_{Bx} \\ {}^A t_{By} \end{bmatrix} + \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix}$$

$${}^A p = {}^A t_B + {}^A R_B {}^B p$$

- Can we represent the pose ${}^A \xi_B$ by the pair $({}^A R_B, {}^A t_B)$?

Investigating pose in 2D

$$\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix} = \begin{bmatrix} {}^A t_{Bx} \\ {}^A t_{By} \end{bmatrix} + \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix}$$

$${}^A p = {}^A t_B + {}^A R_B {}^B p$$

- Can we represent the pose ${}^A \xi_B$ by the pair $({}^A R_B, {}^A t_B)$?

$$\begin{aligned} {}^A p &= {}^A \xi_B \cdot {}^B p &\mapsto & {}^A p = {}^A R_B {}^B p + {}^A t_B \\ {}^A \xi_C &= {}^A \xi_B \oplus {}^B \xi_C &\mapsto & \left({}^A R_C, {}^A t_C \right) = \left({}^A R_B {}^B R_C, {}^A R_B {}^B t_C + {}^A t_B \right) \\ \ominus {}^A \xi_B &&\mapsto & \left({}^A R_C^T, -{}^A R_C^T {}^A t_C \right) \end{aligned}$$

Investigating pose in 2D

$$\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix} = \begin{bmatrix} {}^A t_{Bx} \\ {}^A t_{By} \end{bmatrix} + \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix}$$

$${}^A p = {}^A t_B + {}^A R_B {}^B p$$

- Can we represent the pose ${}^A \xi_B$ by the pair $({}^A R_B, {}^A t_B)$?

$$\begin{aligned} {}^A p &= {}^A \xi_B \cdot {}^B p &\mapsto & {}^A p = {}^A R_B {}^B p + {}^A t_B \\ {}^A \xi_C &= {}^A \xi_B \oplus {}^B \xi_C &\mapsto & \left({}^A R_C, {}^A t_C \right) = \left({}^A R_B {}^B R_C, {}^A R_B {}^B t_C + {}^A t_B \right) \\ \ominus {}^A \xi_B &&\mapsto & \left({}^A R_C^T, -{}^A R_C^T {}^A t_C \right) \end{aligned}$$

- Yes, but there is a better option!

Investigating pose in 2D

- Observe the following equivalence

$$\begin{aligned}\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix} &= {}^A R_B \begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix} + {}^A t_B &\Leftrightarrow&& \begin{bmatrix} {}^A x \\ {}^A y \\ 1 \end{bmatrix} &= \begin{bmatrix} {}^A R_B & {}^A t_B \\ \boldsymbol{0}_{1 \times 2} & 1 \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \\ 1 \end{bmatrix} \\ {}^A p &= {}^A R_B {}^B p + {}^A t_B &\Leftrightarrow&& {}^A \tilde{p} &= {}^A T_B {}^B \tilde{p}\end{aligned}$$

Investigating pose in 2D

- Observe the following equivalence

$$\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix} = {}^A R_B \begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix} + {}^A t_B \quad \Leftrightarrow \quad \begin{bmatrix} {}^A x \\ {}^A y \\ 1 \end{bmatrix} = \begin{bmatrix} {}^A R_B & {}^A t_B \\ \mathbf{0}_{1 \times 2} & 1 \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \\ 1 \end{bmatrix}$$
$${}^A p = {}^A R_B {}^B p + {}^A t_B \quad \Leftrightarrow \quad {}^A \tilde{p} = {}^A T_B {}^B \tilde{p}$$

- Can we represent the pose ${}^A \xi_B$ by the matrix ${}^A T_B$?

Investigating pose in 2D

- Observe the following equivalence

$$\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix} = {}^A R_B \begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix} + {}^A t_B \quad \Leftrightarrow \quad \begin{bmatrix} {}^A x \\ {}^A y \\ 1 \end{bmatrix} = \begin{bmatrix} {}^A R_B & {}^A t_B \\ \mathbf{0}_{1 \times 2} & 1 \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \\ 1 \end{bmatrix}$$
$${}^A p = {}^A R_B {}^B p + {}^A t_B \quad \Leftrightarrow \quad {}^A \tilde{p} = {}^A T_B {}^B \tilde{p}$$

- Can we represent the pose ${}^A \xi_B$ by the matrix ${}^A T_B$?

$$\begin{aligned} {}^A p &= {}^A \xi_B \cdot {}^B p &\mapsto {}^A \tilde{p} &= {}^A T_B {}^B \tilde{p} \\ {}^A \xi_C &= {}^A \xi_B \oplus {}^B \xi_C &\mapsto {}^A T_C &= {}^A T_B {}^B T_C \\ \ominus {}^A \xi_B &&\mapsto & {}^A T_B^{-1} \end{aligned}$$

- Yes, and the algebraic properties are nice!

Investigating pose in 2D

- But...

$$\underbrace{\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix}}_{{}^A \tilde{p}} = {}^A R_B \underbrace{\begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix}}_{{}^B \tilde{p}} + {}^A \boldsymbol{t}_B \quad \Leftrightarrow \quad \underbrace{\begin{bmatrix} {}^A x \\ {}^A y \\ 1 \end{bmatrix}}_{{}^A \tilde{p}} = \underbrace{\begin{bmatrix} {}^A R_B & {}^A \boldsymbol{t}_B \\ \boldsymbol{0}_{1 \times 2} & 1 \end{bmatrix}}_{{}^A T_B} \underbrace{\begin{bmatrix} {}^B x \\ {}^B y \\ 1 \end{bmatrix}}_{{}^B \tilde{p}}$$

- We are describing points in the plane with 3 coordinates despite that they only have 2 degrees of freedom...
- The non linear transformation ${}^B p \mapsto {}^A p$ then becomes a linear transformation ${}^B \tilde{p} \mapsto {}^A \tilde{p}$
- What is going on?

Investigating pose in 2D

- We have “discovered” some basic constructions from projective geometry

$$\begin{bmatrix} {}^A x \\ {}^A y \\ {}^A p \end{bmatrix} = {}^A R_B \begin{bmatrix} {}^B x \\ {}^B y \\ {}^B p \end{bmatrix} + {}^A t_B$$

↔

$$\begin{bmatrix} {}^A x \\ {}^A y \\ 1 \\ {}^A \tilde{p} \end{bmatrix} = \underbrace{\begin{bmatrix} {}^A R_B & {}^A t_B \\ \boldsymbol{\theta}_{1 \times 2} & 1 \end{bmatrix}}_{{}^A T_B} \begin{bmatrix} {}^B x \\ {}^B y \\ 1 \\ {}^B \tilde{p} \end{bmatrix}$$

Investigating pose in 2D

- We have “discovered” some basic constructions from projective geometry

Non-linear transformation
of the Euclidean plane \mathbb{R}^2

Points in the Euclidean
plane are described by
Cartesian coordinates

$$\boxed{\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix} = {}^A R_B \begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix} + {}^A t_B}$$

\Updownarrow

$$\begin{bmatrix} {}^A x \\ {}^A y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} {}^A R_B & {}^A t_B \\ \boldsymbol{\theta}_{1 \times 2} & 1 \end{bmatrix}}_{{}^A T_B} \begin{bmatrix} {}^B x \\ {}^B y \\ 1 \end{bmatrix}$$

${}^A \tilde{p}$ ${}^B \tilde{p}$

Investigating pose in 2D

- We have “discovered” some basic constructions from projective geometry

Non-linear transformation
of the Euclidean plane \mathbb{R}^2

Points in the Euclidean
plane are described by
Cartesian coordinates

$$\begin{bmatrix} {}^A x \\ {}^A y \end{bmatrix}_{\substack{{}^A p}} = {}^A R_B \begin{bmatrix} {}^B x \\ {}^B y \end{bmatrix}_{\substack{{}^B p}} + {}^A t_B$$

↔

$$\begin{bmatrix} {}^A x \\ {}^A y \\ 1 \end{bmatrix}_{\substack{{}^A \tilde{p}}} = \begin{bmatrix} {}^A R_B & {}^A t_B \\ \boldsymbol{0}_{1 \times 2} & 1 \end{bmatrix}_{\substack{{}^A T_B}} \begin{bmatrix} {}^B x \\ {}^B y \\ 1 \end{bmatrix}_{\substack{{}^B \tilde{p}}}$$

Linear transformation
of the projective plane \mathbb{P}^2

Points in the projective plane
are described by homogeneous
coordinates

This means that they are only
unique up to scale, i.e.
 $(x, y, 1) = (\lambda x, \lambda y, \lambda) \forall \lambda \in \mathbb{R} \setminus \{0\}$

The matrix representing the
projective transformation is also
homogeneous, i.e.

$${}^A T_B = \lambda {}^A T_B \forall \lambda \in \mathbb{R} \setminus \{0\}_{37}$$

Investigating pose in 2D

- Euclidean geometry

- ${}^A\xi_B \mapsto ({}^A R_B, {}^A t_B)$

- Complicated algebra

$$\begin{aligned} {}^A p &= {}^A \xi_B \cdot {}^B p &\mapsto & {}^A p = {}^A R_B {}^B p + {}^A t_B \\ {}^A \xi_C &= {}^A \xi_B \oplus {}^B \xi_C &\mapsto & \left({}^A R_C, {}^A t_C \right) = \left({}^A R_B {}^B R_C, {}^A R_B {}^B t_C + {}^A t_B \right) \\ \ominus {}^A \xi_B &&\mapsto & \left({}^A R_C^T, -{}^A R_C {}^T {}^A t_C \right) \end{aligned}$$

- Projective geometry

- ${}^A \xi_B \mapsto {}^A T_B = \begin{bmatrix} {}^A R_B & {}^A t_B \\ \mathbf{0} & 1 \end{bmatrix}$

- Simple algebra

$$\begin{aligned} {}^A p &= {}^A \xi_B \cdot {}^B p &\mapsto & {}^A \tilde{p} = {}^A T_B {}^B \tilde{p} \\ {}^A \xi_C &= {}^A \xi_B \oplus {}^B \xi_C &\mapsto & {}^A T_C = {}^A T_B {}^B T_C \\ \ominus {}^A \xi_B &&\mapsto & {}^A T_B^{-1} \end{aligned}$$

- Many problems in computer vision are easier to express and solve if we choose to think of points and transformations in terms of projective geometry
 - Algebra and computations become simpler

Representing pose in 2D

- The pose of $\{B\}$ relative to $\{A\}$ can be represented by a homogeneous transformation ${}^A T_B \in SE(2)$

$${}^A \xi_B \mapsto {}^A T_B = \begin{bmatrix} {}^A R_B & {}^A t_B \\ \boldsymbol{\theta} & 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & {}^A t_{Bx} \\ \sin \theta & \cos \theta & {}^A t_{By} \\ 0 & 0 & 1 \end{bmatrix}$$

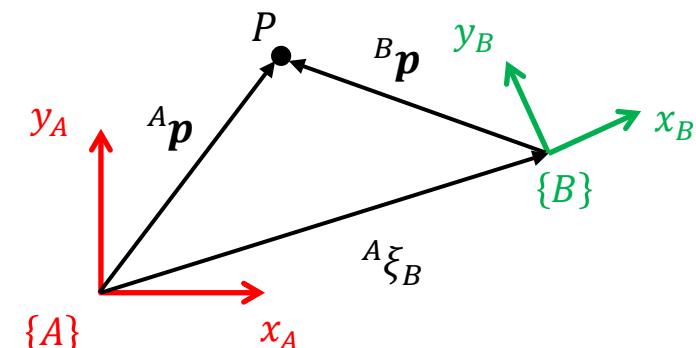
- Properties

$$\begin{aligned} {}^A p &= {}^A \xi_B \cdot {}^B p & \mapsto & {}^A \tilde{p} = {}^A T_B {}^B \tilde{p} \\ {}^A \xi_C &= {}^A \xi_B \oplus {}^B \xi_C & \mapsto & {}^A T_C = {}^A T_B {}^B T_C \\ \ominus {}^A \xi_B & & \mapsto & {}^A T_B^{-1} \end{aligned}$$

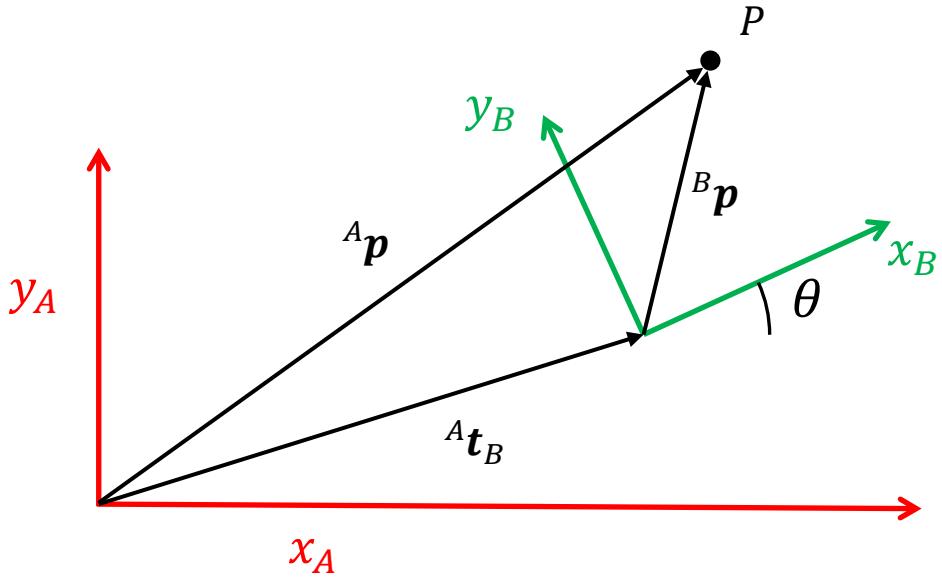
- Points are represented in homogeneous coordinates

$${}^A \tilde{p} = {}^A T_B {}^B \tilde{p}$$

$$\begin{bmatrix} {}^A x \\ {}^A y \\ 1 \end{bmatrix} = \begin{bmatrix} {}^A R_B & {}^A t_B \\ \boldsymbol{\theta}_{1 \times 2} & 1 \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \\ 1 \end{bmatrix}$$



Example



- Let ${}^A\mathbf{t}_B = [4,1]^T$, ${}^B\mathbf{p} = [2,3]^T$ and $\theta = 27^\circ$
- Determine the pose of frame {B} relative to {A}, i.e. ${}^A T_B$
- Determine the coordinates of P in {A}, i.e. ${}^A\mathbf{p}$

Example

- From the previous slides we know that

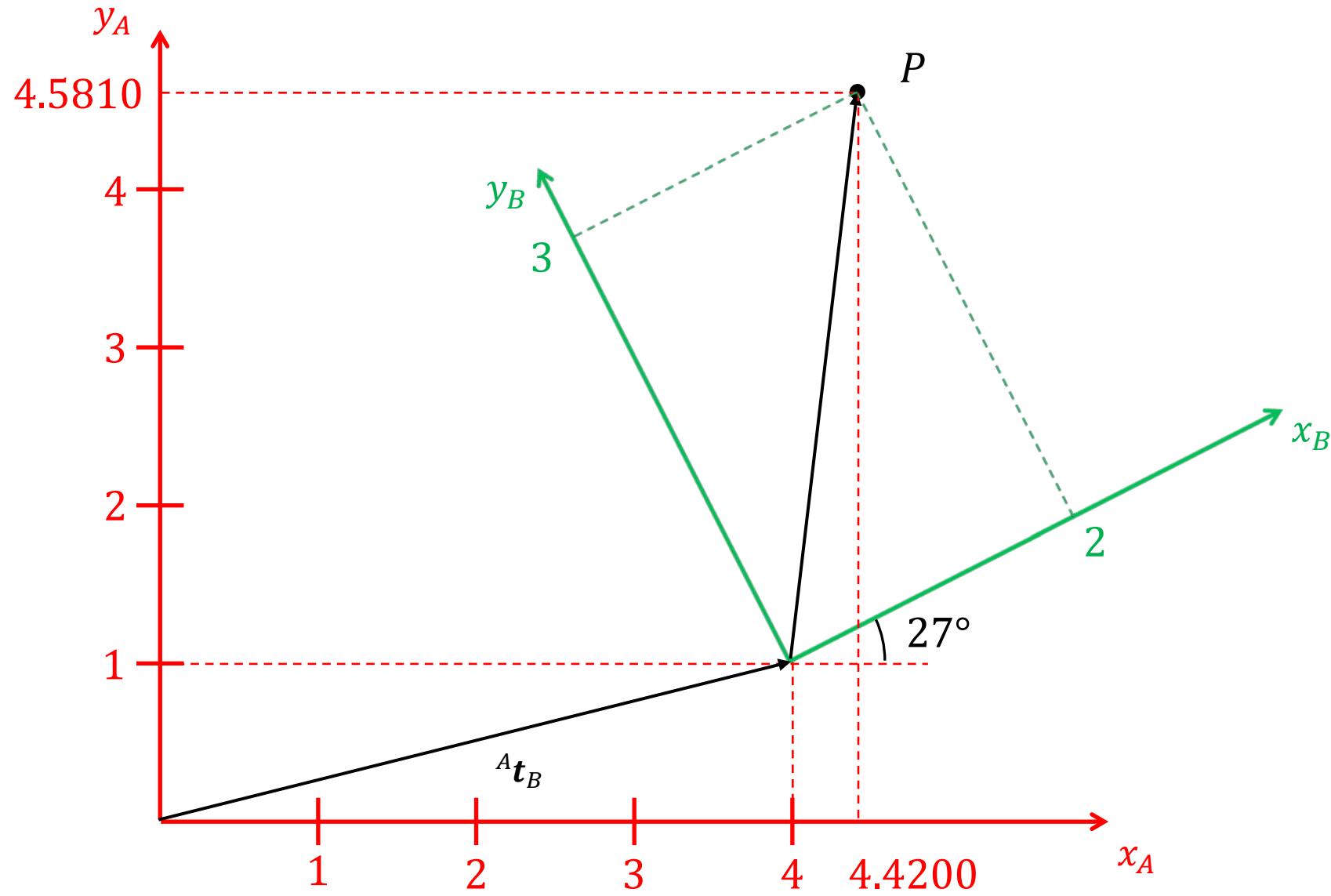
$${}^A T_B = \begin{bmatrix} \cos \theta & -\sin \theta & {}^A t_{Bx} \\ \sin \theta & \cos \theta & {}^A t_{By} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos 27^\circ & -\sin 27^\circ & 4 \\ \sin 27^\circ & \cos 27^\circ & 1 \\ 0 & 0 & 1 \end{bmatrix} \approx \begin{bmatrix} 0.8910 & -0.4540 & 4 \\ 0.4540 & 0.8910 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

- This allows us to compute ${}^A \tilde{\mathbf{p}}$

$$\begin{aligned} {}^A \tilde{\mathbf{p}} &= {}^A T_B {}^B \tilde{\mathbf{p}} \\ \begin{bmatrix} {}^A x \\ {}^A y \\ 1 \end{bmatrix} &= \begin{bmatrix} 0.8910 & -0.4540 & 4 \\ 0.4540 & 0.8910 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 4.4200 \\ 4.5810 \\ 1 \end{bmatrix} \Rightarrow {}^A \mathbf{p} &= [4.4200 \quad 4.5810]^T \end{aligned}$$

- This can also be verified by drawing

Example



Example

```
%% Example: Visualize {A}, {B} and P in coordinates of {A}
robotics_path = 'G:\MATLAB\BIBLIOTEKER\PeterCork_Robotics\robot-9.10\rvctools';
addpath(genpath(robotics_path));

%Pose of {A} relative to {A}
t_AA = [0;0];
theta_AA = 0;
T_AA = se2(t_AA(1),t_AA(2),theta_AA*pi/180);

%Pose of {B} relative to {A}
t_AB = [4;1];
theta_AB = 27;
T_AB = se2(t_AB(1),t_AB(2),theta_AB*pi/180);

%Point P relative to {B}
P_B = [2;3];

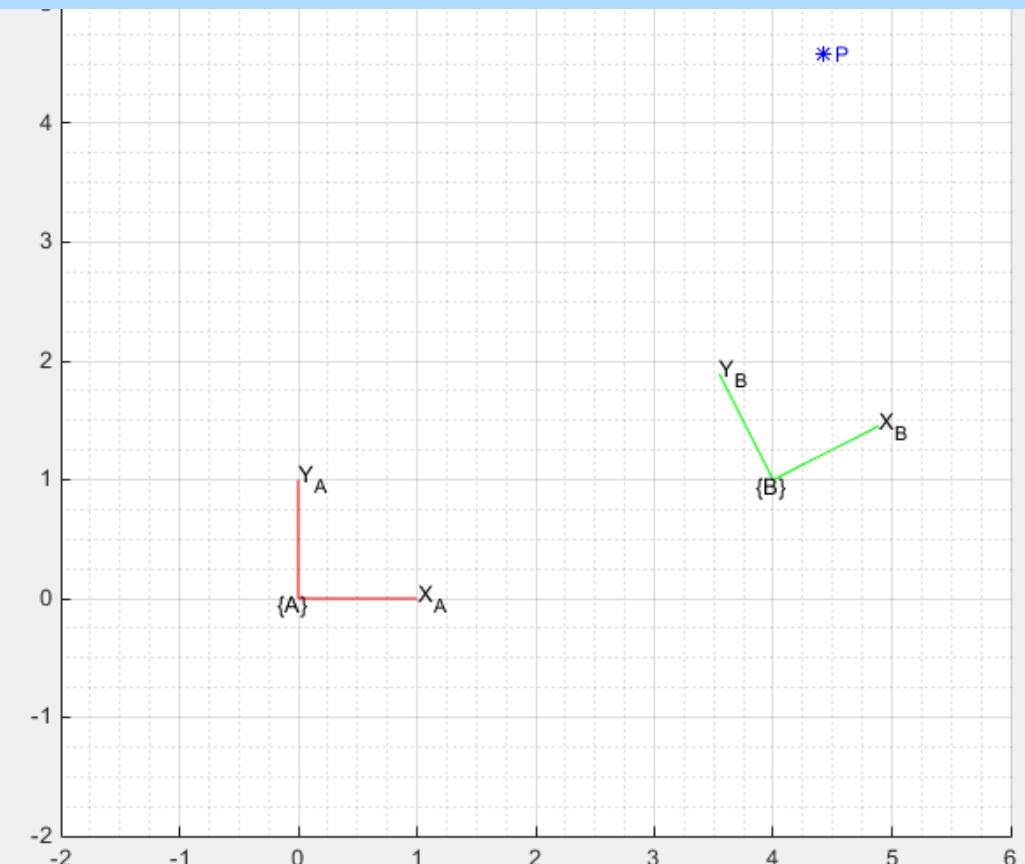
%Transform point P to {A} using homogeneous coordinates
hP_B = e2h(P_B);      % e2h: Changes representation Euclidean --> homogeneous
hP_A = T_AB*hP_B;      % Transformation by multiplication
P_A = h2e(hP_A);       % h2e: Changes representation homogeneous --> Euclidean

%Visualize {A}, {B} and P relative to {A}
figure(1);
clf
axis equal
grid on
grid minor
axis([-2, 6, -2, 6]);
hold on
trplot2(T_AA, 'frame', 'A', 'color', 'r')
trplot2(T_AB, 'frame', 'B', 'color', 'g')
plot_point(P_A, '*b','printf',{' P',P_A}, 'textcolor', 'b')
```

You can visualize this example in matlab using the toolboxes created by Peter Cork

- Robotics Toolbox
- Machine Vision Toolbox

www.petercorke.com/Toolboxes.html



Representing pose in 3D

- The pose of $\{B\}$ relative to $\{A\}$ can be represented by a homogeneous transformation ${}^A T_B \in SE(3)$

$${}^A \xi_B \mapsto {}^A T_B = \begin{bmatrix} {}^A R_B & {}^A \mathbf{t}_B \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & {}^A t_{Bx} \\ r_{21} & r_{22} & r_{23} & {}^A t_{By} \\ r_{31} & r_{32} & r_{33} & {}^A t_{Bz} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

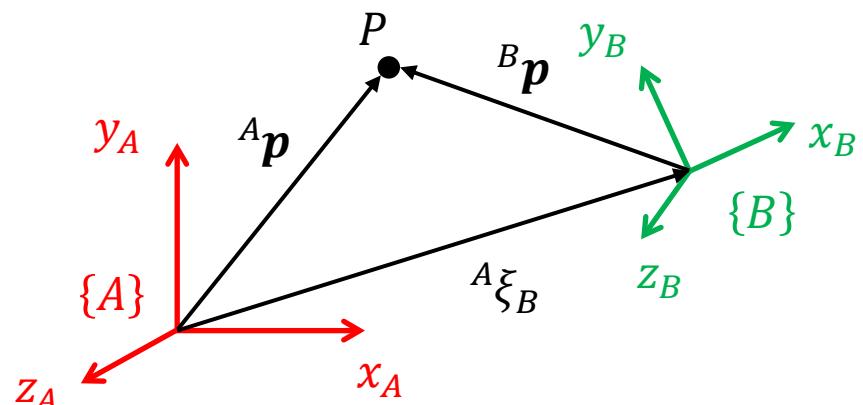
- Properties

$$\begin{aligned} {}^A p &= {}^A \xi_B \cdot {}^B p & \mapsto & {}^A \tilde{p} = {}^A T_B {}^B \tilde{p} \\ {}^A \xi_C &= {}^A \xi_B \oplus {}^B \xi_C & \mapsto & {}^A T_C = {}^A T_B {}^B T_C \\ \ominus {}^A \xi_B & & \mapsto & {}^A T_B^{-1} \end{aligned}$$

- Points are represented in homogeneous coordinates

$${}^A \tilde{p} = {}^A T_B {}^B \tilde{p}$$

$$\begin{bmatrix} {}^A x \\ {}^A y \\ {}^A z \\ 1 \end{bmatrix} = \begin{bmatrix} {}^A R_B & {}^A \mathbf{t}_B \\ \mathbf{0}_{1 \times 2} & 1 \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \\ {}^B z \\ 1 \end{bmatrix}$$



Representing pose in 3D

- The main difference between 3D and 2D is that rotation is far less intuitive in 3D
Also there are several different representations of rotation in 3D
 - Orthonormal rotation matrix $R \in SO(3)$
 - Euler angles $(\theta_1, \theta_2, \theta_3)$
 - Angle-axis (θ, \mathbf{e}) or just $\boldsymbol{\theta} = \theta \mathbf{e}$
 - Unit quaternions $q = r + xi + yj + zk$
- Hence there are several ways to represent pose
 - Rotation matrix and translation vector $({}^A R_B, {}^A \mathbf{t}_B)$
 - Homogeneous transformation ${}^A T_B$
 - Euler angles and translation vector $(\theta_1, \theta_2, \theta_3, {}^A \mathbf{t}_B)$
 - Angle-axis and translation vector $(\theta, \mathbf{e}, {}^A \mathbf{t}_B)$
 - Unit quaternion and translation vector $({}^A q_B, {}^A \mathbf{t}_B)$

Representing rotation in 3D

- Orientation of $\{B\}$ relative to $\{A\}$
 - How $\{B\}$ should rotate to coincide with $\{A\}$

- Orthonormal rotation matrix ${}^A R_B \in SO(3)$

$${}^A R_B = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

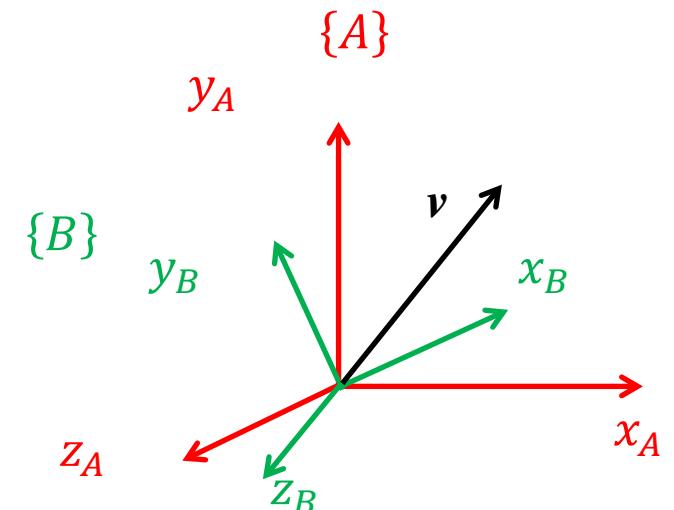
- Properties

$$\det({}^A R_B) = 1$$

$${}^A R_B^{-1} = {}^A R_B^T$$

$${}^A R_C = {}^A R_B {}^B R_C$$

$${}^A \mathbf{v} = {}^A R_B {}^B \mathbf{v}$$



- Nice algebraic properties
- Computations are simple
- Provides little intuitive understanding

Representing rotation in 3D

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}$$

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

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

- Euler angles $(\theta_1, \theta_2, \theta_3)$ describes rotation in terms of basic rotations about 3 coordinate axis that must be specified
 - 12 compositions: $xyx, xzx, yxy, yzy, zxz, zyz, xyz, xzy, yzx, yxz, zxy, zyx$
 - E.g. $R_{xyz} = R_x(\theta_1)R_y(\theta_2)R_z(\theta_3)$

- Suffers from singularities
 - $\theta_2 = \pm\frac{\pi}{2}$ for sequences without repetition
 - $\theta_2 \in \{0, \pi\}$ for sequences with repetition
- It is possible to determine the Euler angles that corresponds to a given rotation matrix

$$\theta_y = \text{atan2}\left(r_{13}, \sqrt{r_{11}^2 + r_{12}^2}\right) \quad -\frac{\pi}{2} < \theta_y < \frac{\pi}{2}$$

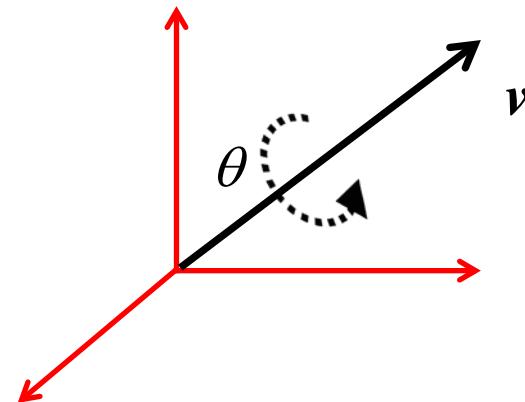
$$R_{xyz} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \Rightarrow \theta_x = \text{atan2}\left(-\frac{r_{23}}{\cos \theta_y}, \frac{r_{33}}{\cos \theta_y}\right) \quad -\pi < \theta_x < \pi$$

$$\theta_z = \text{atan2}\left(-\frac{r_{12}}{\cos \theta_y}, \frac{r_{11}}{\cos \theta_y}\right) \quad -\pi < \theta_z < \pi$$

Representing rotation in 3D

- An angle-axis pair (θ, ν) describes rotation in terms of an angle and an axis of revolution
- Euler's rotation theorem states that any sequence of rotations in 3 dimensions is equivalent to a single rotation θ about an axis ν
- The rotation matrix R corresponding to an angle-axis pair (θ, ν) can be found by Rodrigues' rotation formula

$$R = I_3 + \sin \theta [\nu]_x + (1 - \cos \theta)(\nu \nu^T - I_3)$$



Where $[\nu]_x$ is the matrix representation of the cross product

$$[\nu]_x = \begin{bmatrix} 0 & -\nu_3 & \nu_2 \\ \nu_3 & 0 & -\nu_1 \\ -\nu_2 & \nu_1 & 0 \end{bmatrix}$$

- The angle-axis pair (θ, ν) corresponding to a rotation matrix R can be found from the eigenvalues and eigenvectors of R

Representing rotation in 3D

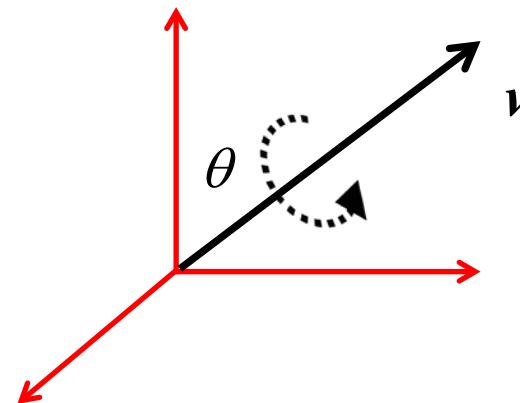
- Unit quaternions are commonly used to work with rotation/orientation in 3D
- A quaternion $q \in \mathbb{H}$ is a number, with 1 real term and 3 imaginary terms

$$q = q_0 + q_1i + q_2j + q_3k = q_0 + \mathbf{v}$$

$$i^2 = j^2 = k = ijk = -1$$

- Conjugate quaternion
- A unit quaternion satisfy

$$\|q\| = \sqrt{q^*q} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1$$



$$q_0 = \cos\left(\frac{\theta}{2}\right)$$
$$\mathbf{v} = \sin\left(\frac{\theta}{2}\right)\mathbf{e}$$

- Quaternion multiplication

$$\begin{aligned} qp &= (q_0 + \mathbf{v})(p_0 + \mathbf{w}) \\ &= \underbrace{q_0 p_0 - \mathbf{v} \cdot \mathbf{w}}_{\in \mathbb{R}} + \underbrace{q_0 \mathbf{w} + p_0 \mathbf{v} + \mathbf{v} \times \mathbf{w}}_{\in \mathbb{R}^3} \end{aligned}$$

- Composing two unit quaternions takes 16 multiplications and 12 additions, while composing two rotation matrixes takes 27 multiplications and 18 additions

Representing rotation in 3D

- The unit quaternion corresponding to the rotation matrix R is given by

$$q_0 = \frac{1}{2} \sqrt{1 + r_{11} + r_{22} + r_{33}}$$

$$q_1 = \frac{r_{32} - r_{23}}{4q_0}$$

$$q_2 = \frac{r_{13} - r_{31}}{4q_0}$$

$$q_3 = \frac{r_{21} - r_{12}}{4q_0}$$

- The rotation matrix corresponding to the unit quaternion $q = q_0 + q_1i + q_2j + q_3k$ is given by

$$R = \begin{bmatrix} 1 - 2q_2^2 - 2q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & 1 - 2q_1^2 - 2q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 1 - 2q_1^2 - 2q_2^2 \end{bmatrix}$$

- SLERP – Spherical linear interpolation
An algorithm for determining a rotation partway between two given rotations

Useful for visualizing rotations

Summary

- Pose
 - General properties
- Representing pose in 2D
 - Homogeneous transformations
- Representing pose in 3D
 - Homogeneous transformations
 - Other alternatives
- Representing rotation in 3D
 - Rotation matrix
 - Euler angles
 - Angle-axis
 - Unit quaternion
- Additional information
 - Szeliski 2.1.1, 2.1.2, 2.1.4