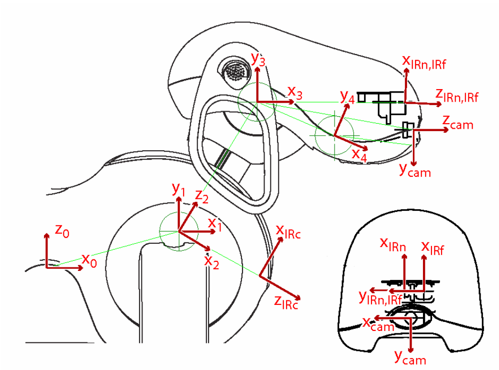
Coordinate Transformations & Forward Kinematics



CSE398-011 Tekkotsu
31 Jan 06

Supporting References

- The Tekkotsu Tutorial

  http://www.cs.cmu.edu/~dst/Tekkotsu/Tutorial/forwardkin.shtml

Background

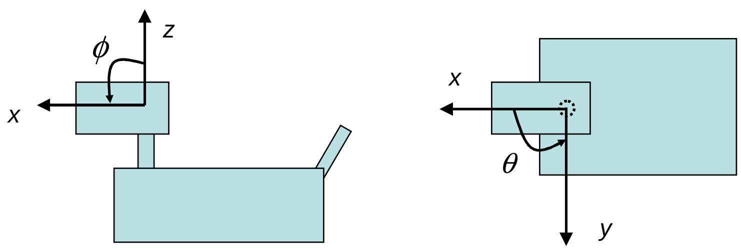
- Ultimately, each Aibo will need to estimate the relative position of items of interest (the ball, the goal, teammates, etc.) from a camera image
- Let us assume that using its camera alone, the Aibo is able to infer the relative distance to the ball
  - How could it do this???
- The Aibo's head has 2 degrees of freedom (ignoring the lower neck joint for now)
  - It can pan left and right (yaw)
  - It can tilt up and down (pitch)

Today's Question

- The Aibo sees the soccer ball in the center of the camera image:
  - The ball is estimated to be 1 meter away
  - It's head is panned to the left 45 degrees
  - It's head is nodded down 30 degrees
- Q: Where is the ball?
- A: Relative to what?

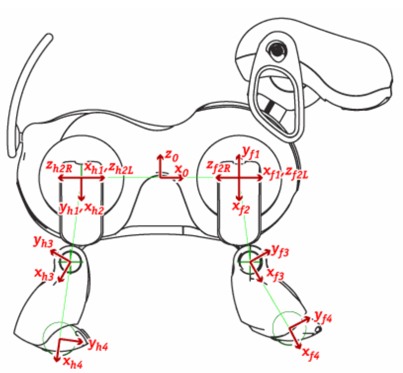
Defining a Reference Frame

- Prior to any position estimation, the coordinate frame that we are measuring relative to must be defined
- Q1: Where is the ball?
- A1: Relative to what?
- Q2: OK… Where is the ball relative to the reference frame below?



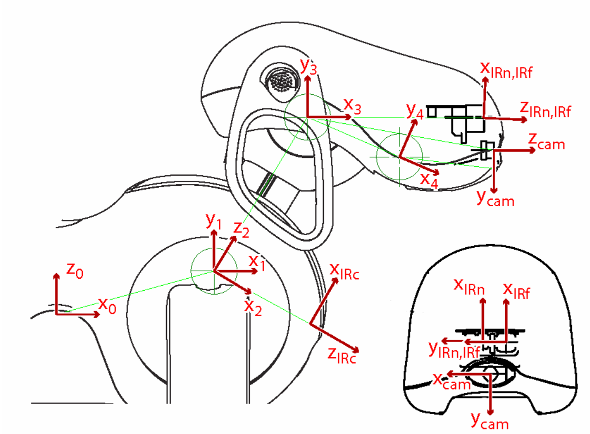
Defining a Reference Frame (cont'd)

- A2: [1 + a nose, 0, 0]^T
- OK…, maybe that wasn't such a good choice
- Let's try a body fixed frame



Defining a Reference Frame (cont'd)

- Or better still, why not define several relevant coordinate frames



Coordinate Transformations

- A *Coordinate Transformation* relates the position vector of any point in coordinate frame *2* to the same point in coordinate frame *1*
- We are interested in *Rigid Transformations* which reflect the relative position and orientation of 1 coordinate frame with respect to another

$${}^{1}\vec{x} = {}^{1}R_{2}\,{}^{2}\vec{x} + {}^{1}\vec{t}_{2}$$

- Here ${}^{i}x$ denotes the position vector for point x as viewed from coordinate frame i, ${}^{i}R_{j}$ denotes a rotation matrix which describes the rotation necessary to align the axis of coordinate frame i to j, and ${}^{i}t_{j}$ the translation from the origin of frame *i to j*
- The rigid transformation necessary to align coordinate frame *i* with *j* has the *opposite* effect of translating points from frame *j* to frame *i*

Rotation Matrices

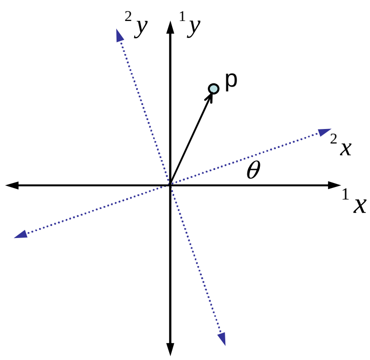
- A *rotation matrix R* rotates position vectors in reference frame 2 (*F2*) to position vectors in *F1*
- In two dimensions

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

where θ corresponds to the relative difference in orientation of *F2* with respect to *F1*

- In this definition, *R* transforms a position vector in *F2* to how the corresponding position vector would "appear" in frame *F1*.

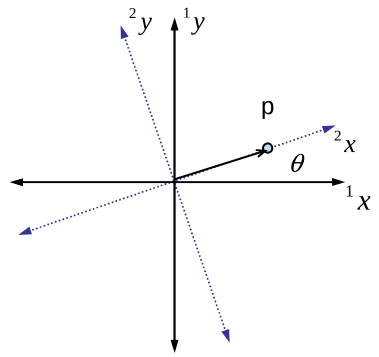
Coordinate Transformation from Pure Rotation



$${}^{1}p = {}^{1}R_{2}\,{}^{2}p$$

$${}^{1}p = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} {}^{2}p$$

Coordinate Transformation from Pure Rotation Example



Example: Let's say that a point in frame F2 is $[1,0]^T$, and θ=30º. What are the point's coordinates in frame F1?

$$^1p = {^1R_2}\, {^2p}$$

$$^1p = \begin{bmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{3}}{2} \\ \frac{1}{2} \end{bmatrix}$$

Some Properties of Rotation Matrices

NOTE: Order is important, as ABC ≠ ACB in general

- det(R) = 1, $R^T=R^{-1}$
- R=I denotes no rotation
- The product of 2 rotation matrices is a rotation matrix
- In three dimensions:

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \quad R_y = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix}$$

$$R_z = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

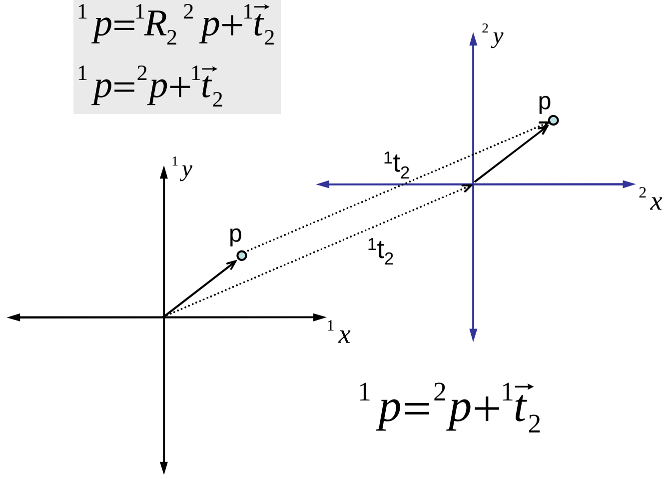
where *xyz* rotations correspond to roll, pitch, and yaw

Pure Translations

- A pure translation corresponds to a transformation when *R=I*. The equation for a rigid transformation then reduces to

$$^{1}p = {}^{1}R_{2}\,{}^{2}p + {}^{1}\vec{t}_{2}$$

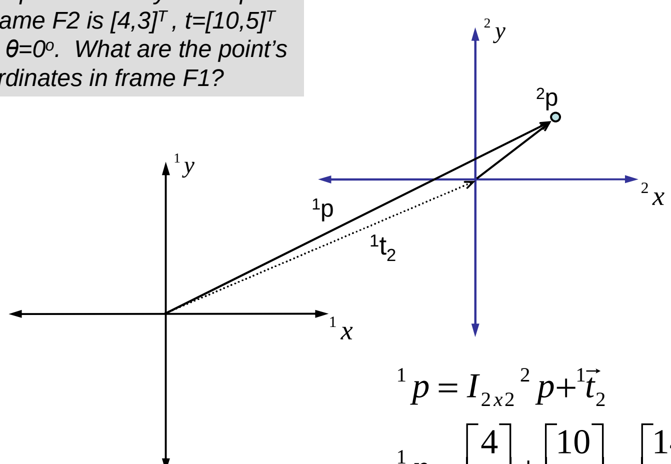
$$^{1}p = {}^{2}p + {}^{1}\vec{t}_{2}$$



$$^{1}p = {}^{2}p + {}^{1}\vec{t}_{2}$$

Pure Translations

Example: Let's say that a point in frame F2 is $[4,3]^T$, $t=[10,5]^T$ and $\theta=0^o$. What are the point's coordinates in frame F1?

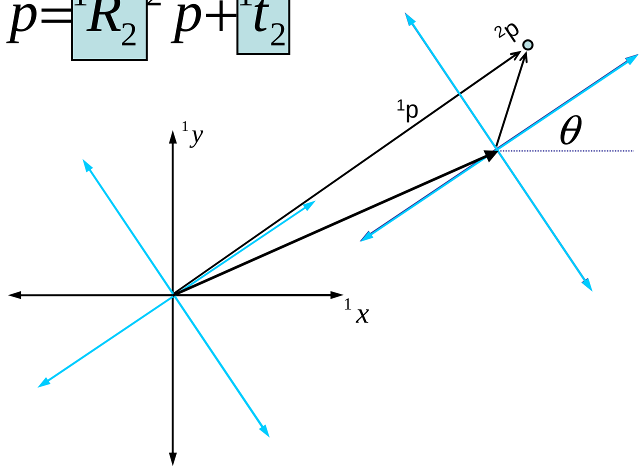


$$^{1}p = I_{2x2}\,{}^{2}p + {}^{1}\vec{t}_{2}$$

$$^{1}p = \begin{bmatrix} 4 \\ 3 \end{bmatrix} + \begin{bmatrix} 10 \\ 5 \end{bmatrix} = \begin{bmatrix} 14 \\ 8 \end{bmatrix}$$

General Rigid Transformation

$$ {}^{1}p = {}^{1}R_{2}\,{}^{2}p + {}^{1}\vec{t}_{2} $$



Homogeneous Coordinates

- There is no nice way to represent translations using a 2x2 matrix in R^2 (or 3x3 matrix in R^3)
- *Homogeneous Coordinates* provide a convenient means for representing and composing multiple rigid transformations
- The dimension of each coordinate is increased by 1:

$$ \begin{bmatrix} x \\ y \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ \lambda \end{bmatrix} \qquad \begin{bmatrix} x \\ y \\ z \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ z \\ \lambda \end{bmatrix} $$

Homogeneous Coordinates

- There is no nice way to represent translations using a 2x2 matrix in R^2 (or 3x3 matrix in R^3)
- *Homogeneous Coordinates* provide a convenient means for representing and composing multiple rigid transformations
- The dimension of each coordinate is increased by 1:

$$\begin{bmatrix} x \\ y \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad \begin{bmatrix} x \\ y \\ z \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

- Last coordinate normally "normalized" to 1

Homogeneous Coordinates (cont'd)

- An important property of homogenous coordinates is that two are equivalent if they are a scalar multiple of one another

$$\begin{bmatrix} \lambda x \\ \lambda y \\ \lambda \end{bmatrix} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- HC also provides a convenient representation for points at infinity

$$p = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}$$

Homogeneous Transformation Matrices

- With this representation, we can compose a rigid transformation into a single, linear transformation matrix.
- In two dimensions, the representation is as follows:

$$\begin{bmatrix} {}^1x \\ {}^1y \\ 1 \end{bmatrix} = \begin{bmatrix} \overbrace{\begin{matrix}\cos\theta & -\sin\theta \\ \sin\theta & \cos\theta\end{matrix}}^{{}^1R_2} & \overbrace{\begin{matrix}{}^1tx_2 \\ {}^1ty_2\end{matrix}}^{{}^1t_2} \\ \underbrace{\begin{matrix}0 & 0\end{matrix} \quad 1}_{\text{constant}} \end{bmatrix} \begin{bmatrix} {}^2x \\ {}^2y \\ 1 \end{bmatrix} \qquad {}^1p = {}^1A_2\,{}^2p$$

- Note that this is a rotation followed by a translation, which is NOT the same as the reverse!

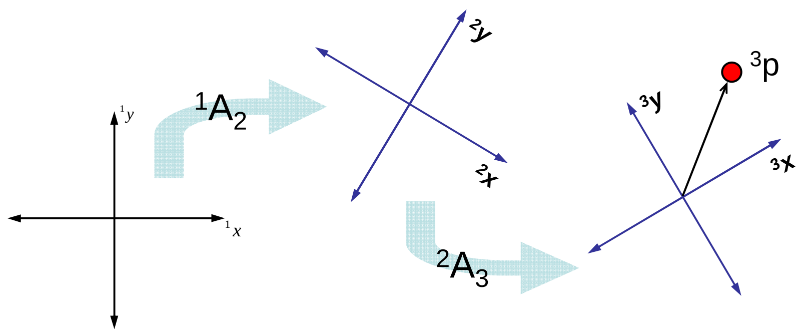
Homogeneous Transformation Matrices in Three Dimensions

- The representation readily extends to three dimensions

$$\begin{bmatrix} {}^1x \\ {}^1y \\ {}^1z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{11} & r_{13} & {}^1tx_2 \\ r_{21} & r_{22} & r_{23} & {}^1ty_2 \\ r_{31} & r_{32} & r_{33} & {}^1tz_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^2x \\ {}^2y \\ {}^2z \\ 1 \end{bmatrix} \qquad {}^1p = {}^1A_2\,{}^2p$$

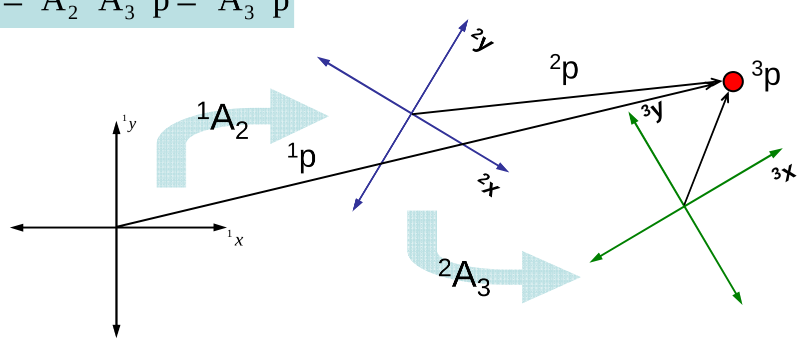
Composing Homogenous Transformations

- Perhaps the strongest point for the homogeneous transformation representation is the ability to compose multiple transformations across multiple frames
- Suppose we would like to estimate the position of a point that has seen coordinate transformations across 2 frames



Composing Homogenous Transformations

$$^1p = {}^1A_2\,{}^2A_3\,{}^3p = {}^1A_3\,{}^3p$$



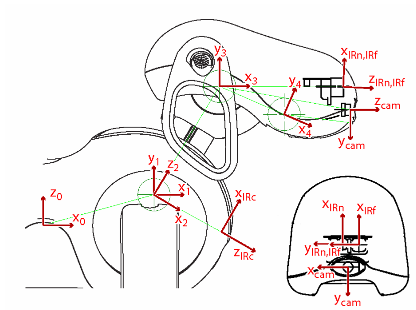
This generalizes for *n* frames

$$^1p = {}^1A_2 \ldots {}^{n-1}A_n\,{}^np = {}^1A_n\,{}^np$$

NOTE: The transformations are done LOCAL to the current frame

Ball Detection Example

- The Aibo sees the soccer ball in the center of the camera image:
  - The ball is estimated to be 1 meter away
  - It's head is panned to the left 45 degrees
  - It's head is nodded down 30 degrees



ERS-7 Head

	Δx	Δy	Δz
1. - $tilt_0$	67.5	0	19.5
2. - pan_1	0	0	0
3. - nod_2	0	0	80
4. - jaw_3	40	-17.5	0
cam. - $camera_3$	81.06	-14.6	0
IRn. - $NearIR_3$	76.9	*1.917*	*2.795*
IRf. - $FarIR_3$	76.9	*1.052*	*-8.047*
IRc. - $ChestIR_0$	109.136	-3.384	0

$x_3 \angle x_4 = -23.6294°$

Summary

- Points are defined with respect to a specific coordinate frame
- Often, it is convenient to measure a point with respect to one frame (*e.g.* an object's position in the sensor frame), but it must be transformed to another frame for other reasons (*e.g.* navigational convenience)
- Coordinate transformations provide this mechanism
- The transformation necessary to align coordinate frame F1 with frame F2 is also the same transformation necessary to convert points from frame F2 to frame F1
- Homogeneous coordinates provide a convenient means for representing and composing rigid transformations