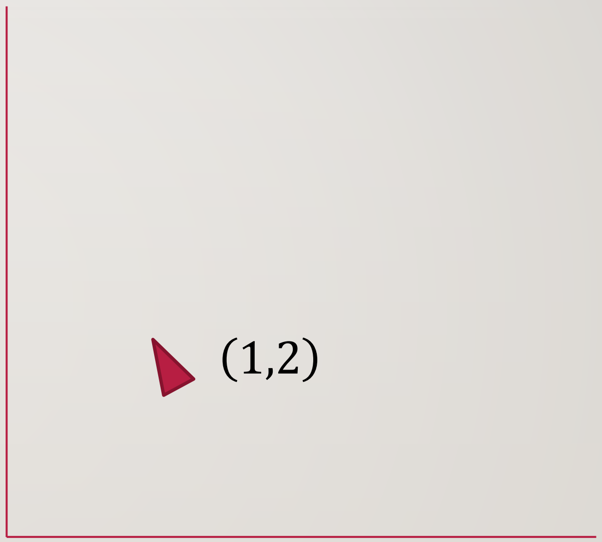
ROBOTICS

COORDINATE FRAMES AND TRANSFORMATIONS

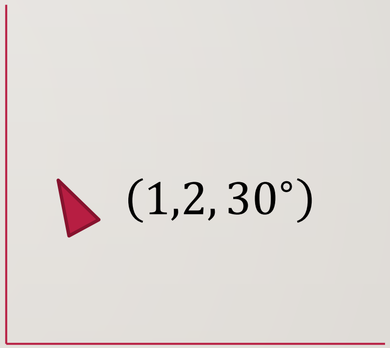
WHAT? WHY?

- How do we specify position?
  - In 2D? - x, y
  - What is this relative to?



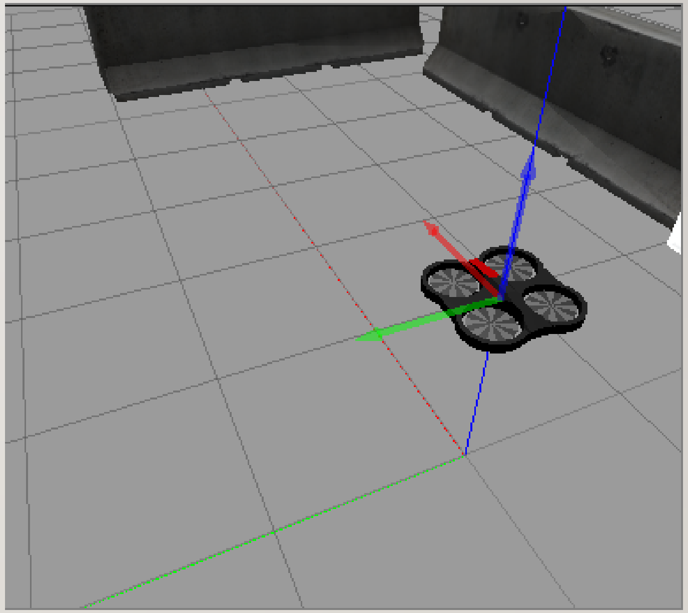
WHAT? WHY?

- How do we specify position?
  - In 2D? - x, y
  - What is this relative to?
- Orientation?
  - θ



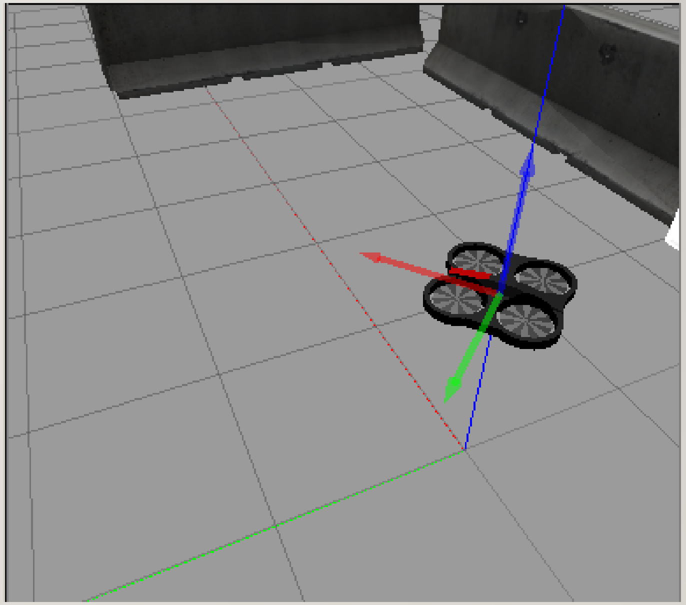
IS THIS ALWAYS FINE?

- AR Drone example
  - x, y, z
  - If you're applying some force along x, what would happen?
  - When doing PID, this was very useful because you could figure out where you wanted to be in the world, find an error in each axis, and apply some control



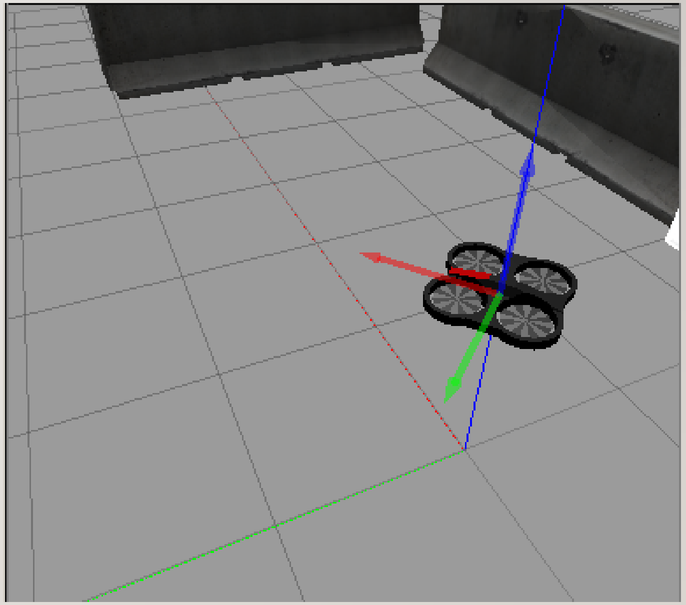
IS THIS ALWAYS FINE?

- AR Drone example
  - x, y, z
  - If you're applying some force along x, what would happen?
  - Wait a minute… There are two x axes now
  - Which one is the one that matters?



IS THIS ALWAYS FINE?

- AR Drone example
  - x, y, z
  - If you're applying some force along x, what would happen?
  - Wait a minute… There are two x axes now
  - Which one is the one that matters?
    - Target and source are specified relative to the world
    - Control is specified relative to the robot

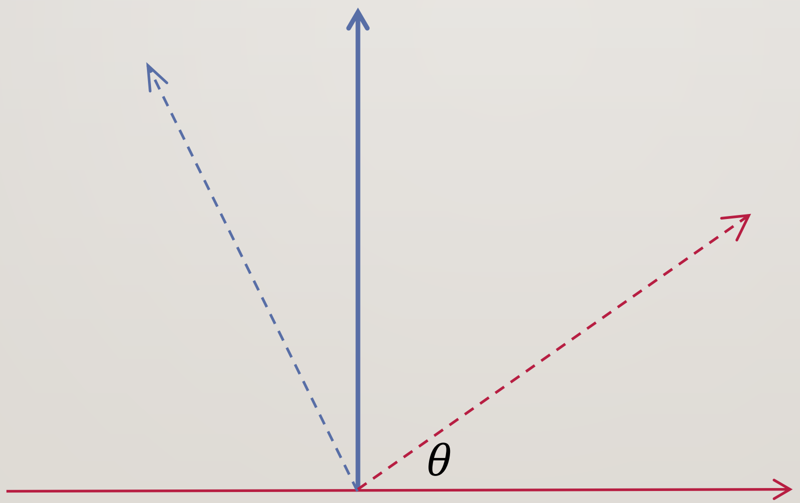


TRANSFORMATIONS!

- We need a way of converting between these reference frames
- The problem here is that we have one reference frame that's rotated relative to the other one

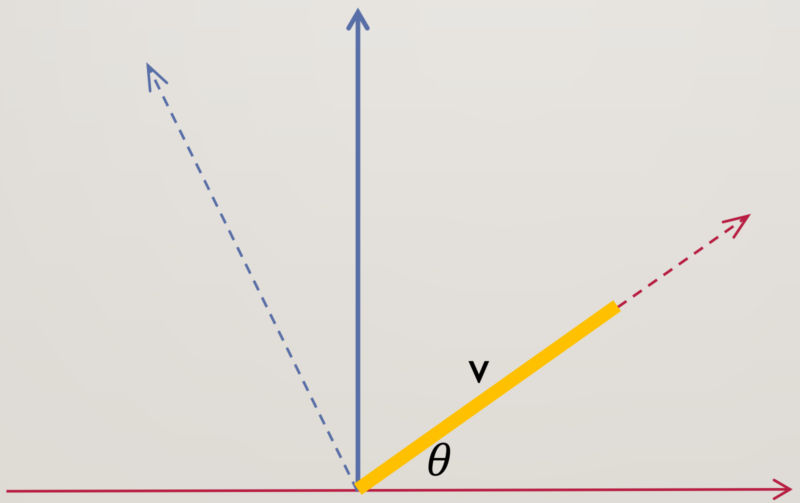
TRANSFORMATIONS!

- We need a way of converting between these reference frames
- The problem here is that we have one reference frame that's rotated relative to the other one



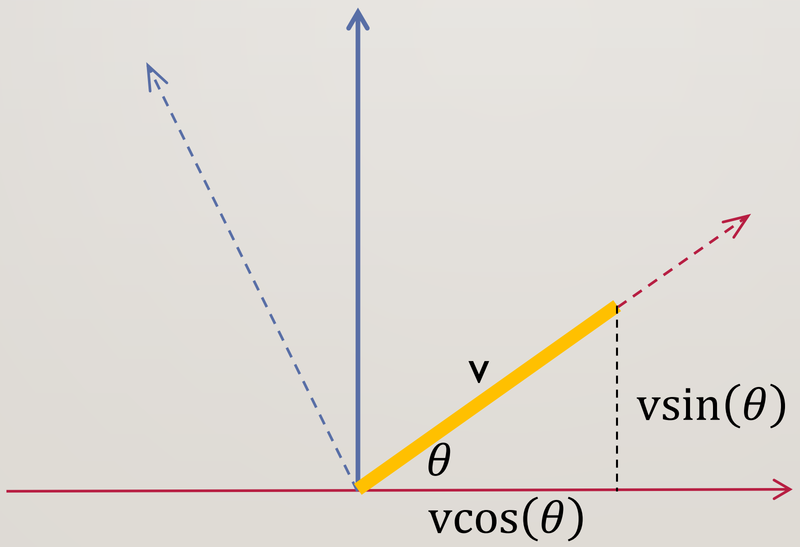
TRANSFORMATIONS!

- If I send the drone v units forward along the x axis, what will that do its position?
- Let's say it's currently at $(0,0)$



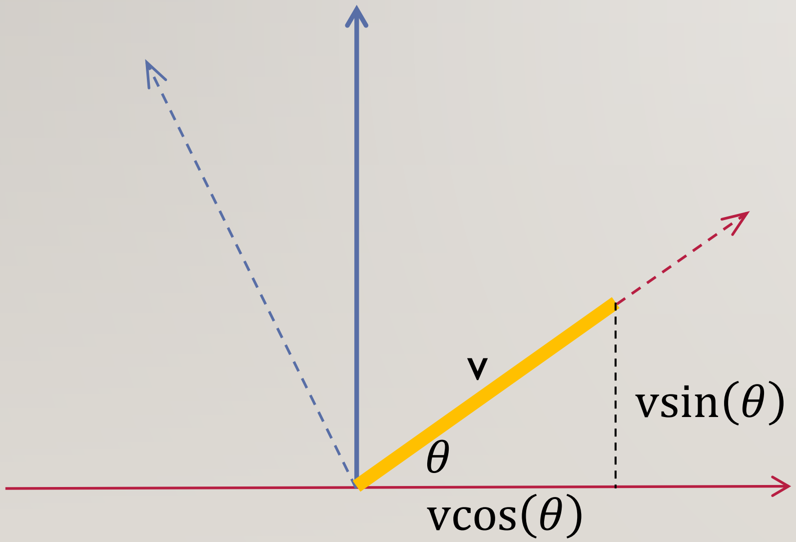
TRANSFORMATIONS!

- If I send the drone v units forward along the x axis, what will that do its position?
- Let's say it's currently at $(0,0)$



TRANSFORMATIONS!

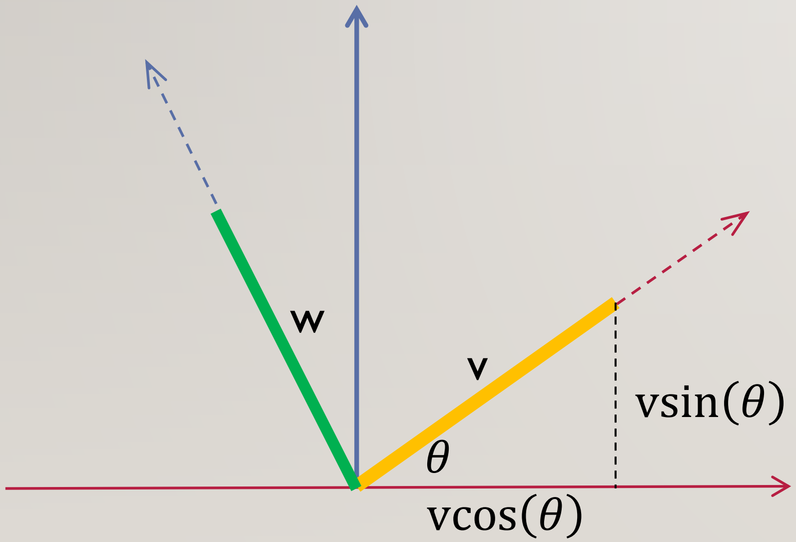
- The position of the robot (p) is now different in the two axes - World (0) and Robot (1)
- $p^1 = (v, 0)$
- $p^0 = \left(vcos(\theta), vsin(\theta)\right)$



- A v change in x^1 results in a $vcos(\theta)$ change in x^0
- A v change in x^1 results in a $vsin(\theta)$ change in y^0

TRANSFORMATIONS!

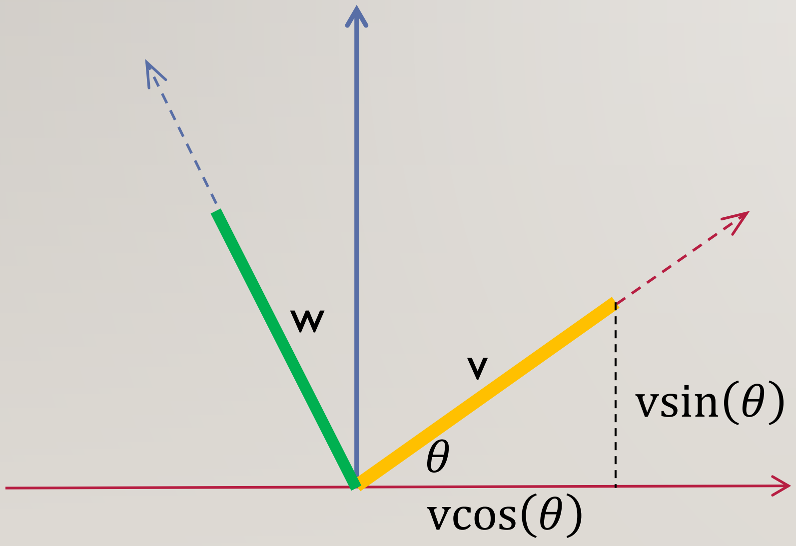
- What happens with an w change in y^1?



- A v change in x^1 results in a $vcos(\theta)$ change in x^0
- A v change in x^1 results in a $vsin(\theta)$ change in y^0

TRANSFORMATIONS!

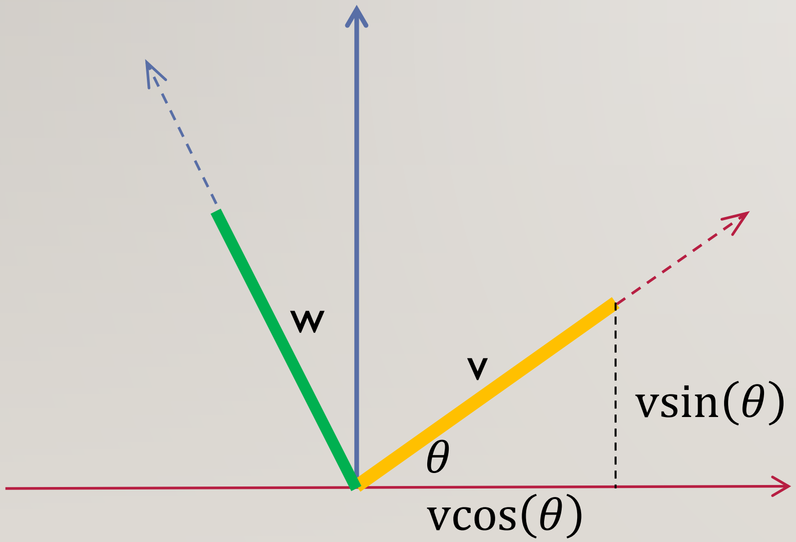
- What happens with an w change in y^1?



- A v change in x^1 results in a $vcos(\theta)$ change in x^0
- A v change in x^1 results in a $vsin(\theta)$ change in y^0
- A w change in y^1results in a $-wsin(\theta)$change in x^1
- A w change in y^1results in a $wcos(\theta)$change in y^1

TRANSFORMATIONS!

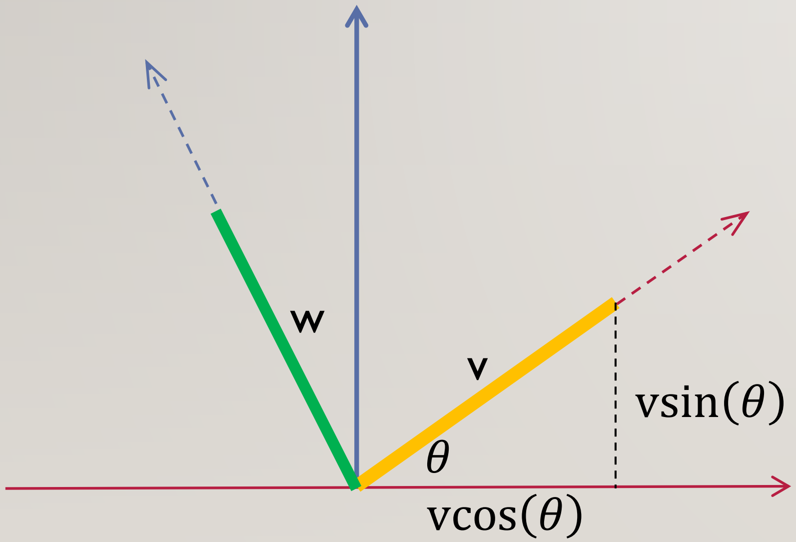
- $x^1 = v, y^1 = w$
- $x^0 = vcos(\theta) - wsin(\theta)$
- $y^0 = vsin(\theta) + wcos(\theta)$



- So we can restate this as:
- $x^0 = x^1cos(\theta) - y^1sin(\theta)$
- $y^0 = x^1sin(\theta) + y^1cos(\theta)$

TRANSFORMATIONS!

- $x^0 = x^1 cos(\theta) - y^1 sin(\theta)$
- $y^0 = x^1 sin(\theta) + y^1 cos(\theta)$



- And we can restate this as:
- $\begin{bmatrix} x^0 \\ y^0 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x^1 \\ y^1 \end{bmatrix}$

TRANSFORMATIONS!

- $\begin{bmatrix} x^0 \\ y^0 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x^1 \\ y^1 \end{bmatrix}$
- We call this the rotation matrix, denoted by R_1^0
- $p^0 = R_1^0 p^1$
- So R_1^0 transforms from reference frame 1 to reference frame 0

TRANSFORMATIONS!

- $\begin{bmatrix} x^0 \\ y^0 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x^1 \\ y^1 \end{bmatrix}$
- $p^0 = R_1^0 p^1$
- If frame 1 is rotated by $90°$ relative to frame 0, and $p^1$is $\begin{bmatrix} 2 \\ 1 \end{bmatrix}$ find p^0

3D ROTATIONS

- 3D rotations are performed around a particular axis
- How would we rotate around the z-axis?
- Picture the z-axis as a handle you're rotating the world around
- This would look just like the rotation we've been doing, so it's like rotating in 2D, for x and y
- If we're rotating around z, what transformation would be needed between $z^1 and\ z^0$?

ROTATION AROUND Z

- $p^0 = R_1^0 p^1$
- $x^0 = x^1 cos(\theta) - y^1 sin(\theta)$
- $y^0 = x^1 sin(\theta) + y^1 cos(\theta)$
- $z^0 = z^1$
- $\begin{bmatrix} x^0 \\ y^0 \\ z^0 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^1 \\ y^1 \\ z^1 \end{bmatrix}$

OTHER ROTATIONS

- Rotation matrices for x and y are similar, just using the other planes

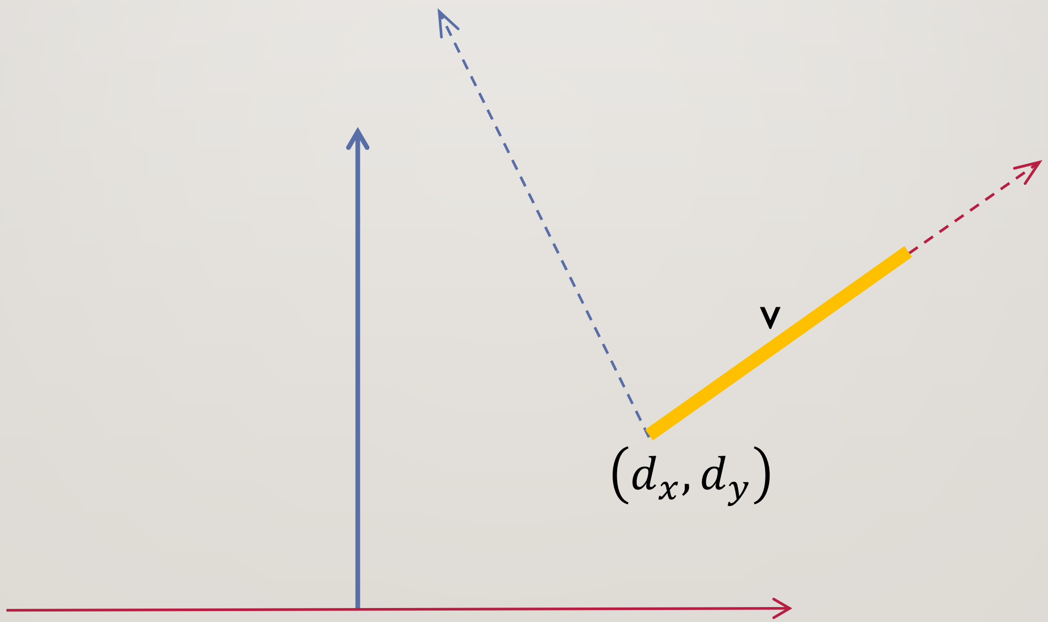
- $R_{\{x,\theta\}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix}$

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

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

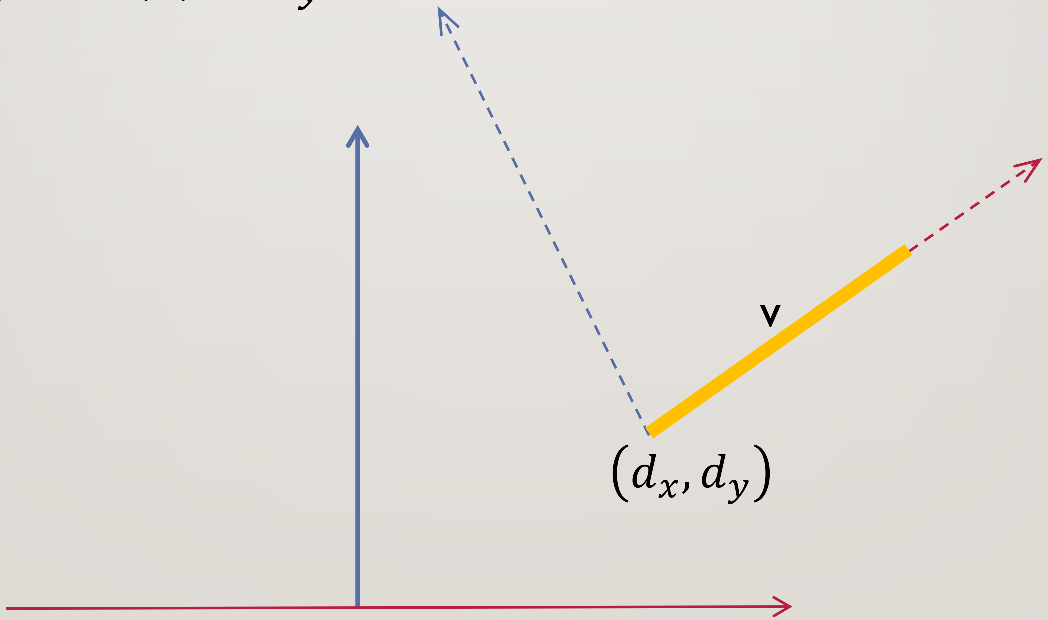
WHAT ABOUT POSITION?

- What do we do when the axes are not at the same origin? Let's say the drone has moved on a bit and rotated as well



WHAT ABOUT POSITION?

- $x^0 = x^1 cos(\theta) - y^1 sin(\theta) + d_x$
- $y^0 = x^1 sin(\theta) + y^1 cos(\theta) + d_y$



WHAT ABOUT POSITION?

- $x^0 = x^1 cos(\theta) - y^1 sin(\theta) + d_x$
- $y^0 = x^1 sin(\theta) + y^1 cos(\theta) + d_y$
- $\begin{bmatrix} x^0 \\ y^0 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & d_x \\ \sin(\theta) & \cos(\theta) & d_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^1 \\ y^1 \\ 1 \end{bmatrix}$
- $\begin{bmatrix} p^0 \\ 1 \end{bmatrix} = T_1^0 \begin{bmatrix} p^1 \\ 1 \end{bmatrix}$

HOMOGENOUS TRANSFORMATION IN 3D

- Reference frame 1 is rotated by θ degrees around z and displaced by $\left(d_x,\ d_y, d_z\right)$

- $\begin{bmatrix} x^0 \\ y^0 \\ z^0 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 & d_x \\ \sin(\theta) & \cos(\theta) & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^1 \\ y^1 \\ z^1 \\ 1 \end{bmatrix}$

- $\begin{bmatrix} p^0 \\ 1 \end{bmatrix} = T_1^0 \begin{bmatrix} p^1 \\ 1 \end{bmatrix}$

- You can change the relevant portion of the transformation matrix if you're rotating around x or y instead

STACKING TRANSFORMATIONS

- Reference frame 1 is rotated by θ degrees around z and displaced by $(d_x,\ d_y, d_z)$ relative to reference frame 0 : construct T_1^0
- Reference frame 2 is rotated by α degrees around x and displaced by $(q_x,\ q_y, q_z)$ relative to reference frame 1 : construct T_2^1
- $p^0 = T_1^0 p^1$
- $p^1 = T_2^1 p^2$
- $p^0 = T_1^0 T_2^1 p^2$
- $T_2^0 = T_1^0 T_2^1$
- $p^0 = T_2^0 p^2$