General Transformation Matrices
Definitions

Unit Vector

Vector of magnitude  = 1

Orthogonal Vectors

Dot product = 0

90 degree angle  between them
System vs. Frame

Coordinate System
3 unit orthogonal vectors

Coordinate Frame
3 unit orthogonal vectors
Position

(a) (b)
Coordinate Frames

As we have seen, transformations can be made wrt a fixed coordinate frame or wrt the coordinate frame that is moving with the object.

In general, we can view any **rigid body** transformation as moving a coordinate frame (translations and rotations).

That new coordinate frame (described in terms of the old frame) makes up the transformation matrix.
Example

Rot(y, 90), Trans(z, a) Rot(x, 90)Trans(y, a) = 

\[
\begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 \\
-1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & a \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & a \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
= 

\begin{bmatrix}
0 & 1 & 0 & 2a \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
x_n \\
y_n \\
z_n \\
\end{bmatrix}
\begin{bmatrix}
0 & 0 & 1 \\
1 & 0 & 0 \\
0 & 1 & 0 \\
2a & -1 & 0 \\
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
z \\
\end{bmatrix}
\]
Example

Rot(y, 90), Trans(z, a) Rot(x, 90)Trans(y, a) =

\[
\begin{bmatrix}
0 & 1 & 0 & 2a \\
0 & 0 & -1 & 0 \\
-1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

book calls this \( M^T \)

Describe new coordinate frame (in cols) in terms of the old
Rotation Matrix Properties

Rotation component is a unit orthogonal matrix
  All rows or cols are unit length
  All rows are orthogonal, all cols are orthogonal

*Any* unit orthogonal matrix is a valid rotation matrix!!
*Any* unit orthogonal matrix describes a frame orientation
Using Unit Orthogonal Matrices...

If I know

- Where I want the object to go (translation)
- And how I want it oriented (description of the unit orthogonal axes)

I can write the transformation matrix in one step by filling the *columns* appropriately.
General Transformation Matrix

\[
\begin{bmatrix}
0 & 1 & 0 & 2a \\
0 & 0 & -1 & 0 \\
-1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

Rotation part (unit orthogonal)  Translation Part
Unit Orthogonal Matrices - LookAt Revisited

\[
\text{LookAt}(\text{eyex}, \text{eyey}, \text{eyez}, \text{atx}, \text{aty}, \text{atz}, \text{upx}, \text{upy}, \text{upz})
\]
LookAt Revisited

Remember LookAt requires **eye (or from), at, up**

How can you use the concepts from today to build a transformation matrix for the camera (or, more specifically, a transformation matrix for the world!)

Build a frame for the ‘imaginary’ camera

Invert it and apply to the world!
**Change of Frames**

Often need to take a point in one frame and get it’s representation in another frame

Object to World

World to Eye/Camera

If you have M, you change frames by inverting it \((M)^{-1}\)

\[ P_{\text{world}} = M \ P_{\text{local}} \]  // transformation from a point in new to the old .... or from local to the world

So, to change the frame of a point from world to local,

\[ P_{\text{local}} = M^{-1} \ P_{\text{world}} \]
I want the robot to touch the doorbell

What if I know the position of the doorbell in the world (reference)

And I know the frame \((M_{\text{hand}})\) for the hand

\[
P_{\text{ref}} = M_{\text{hand}} P_{\text{hand}}
\]

Then I can compute the position of doorbell wrt the hand (and therefore how much to move the hand!) by applying the inverse transform of the hand

\[
P_{\text{hand}} = M_{\text{hand}}^{-1} P_{\text{ref}}
\]
More useful properties…

The inverse of a rotation matrix is its transpose.

If I know the orientation of an object in space…and I want to align it with the references axes, I simply invert the unit orthogonal rotation matrix.

I fill the rows with the appropriate vectors (by transpose).

To invert the entire frame (rot + trans part), must compute the full inverse:

\[(TR)^{-1} = R^{-1}T^{-1}\]