Reference frame conversions

Laurens Valk

December 26, 2017

Contents

1 From Pixels to Meters **1**
1.1 Transformation Matrices 1
1.2 Scaling Factor 3

2 Local Robot Frames **4**
2.1 Label Orientation 4

3 Relative Agent Positions and Orientations **4**

1 From Pixels to Meters

1.1 Transformation Matrices

The symbol $\tilde{r}$ expresses vectors in units of pixels. Any other vector r has units of meters.

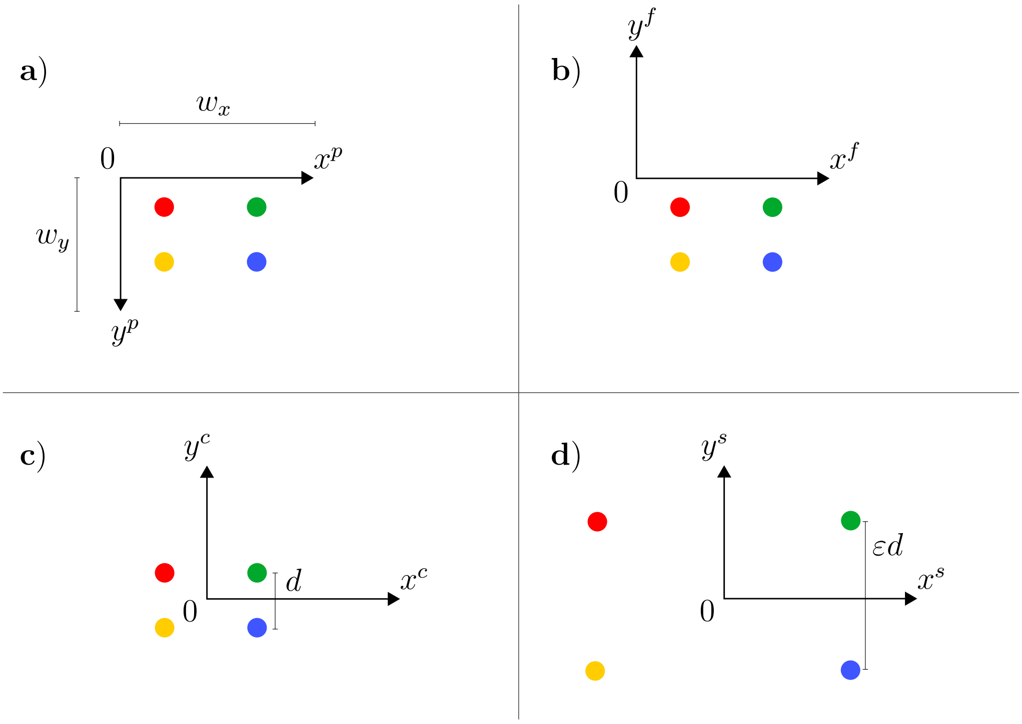

Figure 1: Four coordinates: a) in the camera picture, b) with flipped y axis, c) with the origin at the center of the picture, d) scaled to physical distances.

$$\begin{bmatrix} \tilde{r}^f \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{r}^p \\ 1 \end{bmatrix} = H_p^f \begin{bmatrix} \tilde{r}^p \\ 1 \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} \tilde{r}^c \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\frac{1}{2}\tilde{w}_x \\ 0 & 1 & \frac{1}{2}\tilde{w}_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{r}^f \\ 1 \end{bmatrix} = H_f^c \begin{bmatrix} \tilde{r}^f \\ 1 \end{bmatrix} \tag{2}$$

$$\begin{bmatrix} r^s \\ 1 \end{bmatrix} = \begin{bmatrix} \varepsilon & 0 & 0 \\ 0 & \varepsilon & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{r}^c \\ 1 \end{bmatrix} = H_c^s \begin{bmatrix} \tilde{r}^c \\ 1 \end{bmatrix} \tag{3}$$

$$\begin{bmatrix} r^s \\ 1 \end{bmatrix} = H_c^s H_f^c H_p^f \begin{bmatrix} \tilde{r}^p \\ 1 \end{bmatrix} \tag{4}$$

The overall transformation from pixels to the real world representation is

$$\begin{bmatrix} r^s \\ 1 \end{bmatrix} = H_p^s \begin{bmatrix} \tilde{r}^p \\ 1 \end{bmatrix} \tag{5}$$

where

$$H_p^s = H_c^s H_f^c H_p^f = \begin{bmatrix} \varepsilon & 0 & -\frac{1}{2}\varepsilon\tilde{w}_x \\ 0 & -\varepsilon & \frac{1}{2}\varepsilon\tilde{w}_y \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

1.2 Scaling Factor

The distance $d_{\text{top/center}}$ in Figure 2 is known, constant, and equivalent for each agent:

$$d_{\text{top/center}} \equiv ||r_{\text{top},i} - r_{\text{center},i}|| \quad \forall\, i \tag{7}$$

Per agent, we can estimate the scaling factor ϵ_i as:

$$\varepsilon_i = \frac{d_{\text{top/center}}}{||\tilde{r}_{\text{top},i} - \tilde{r}_{\text{center},i}||} \quad \text{m/pixel} \tag{8}$$

A more accurate estimate for the overall scaling factor is:

$$\varepsilon = \frac{1}{N}\sum_{i=1}^{N} \varepsilon_i \tag{9}$$

However, because ε is constant it may be more desirable to calculate it once and use it as a constant parameter. If so, we may divide the arena width by the picture with in pixels:

$$\varepsilon = \frac{d_{\text{field}}}{\tilde{w}_x} \quad \text{m/pixel} \tag{10}$$

2 Local Robot Frames

2.1 Label Orientation

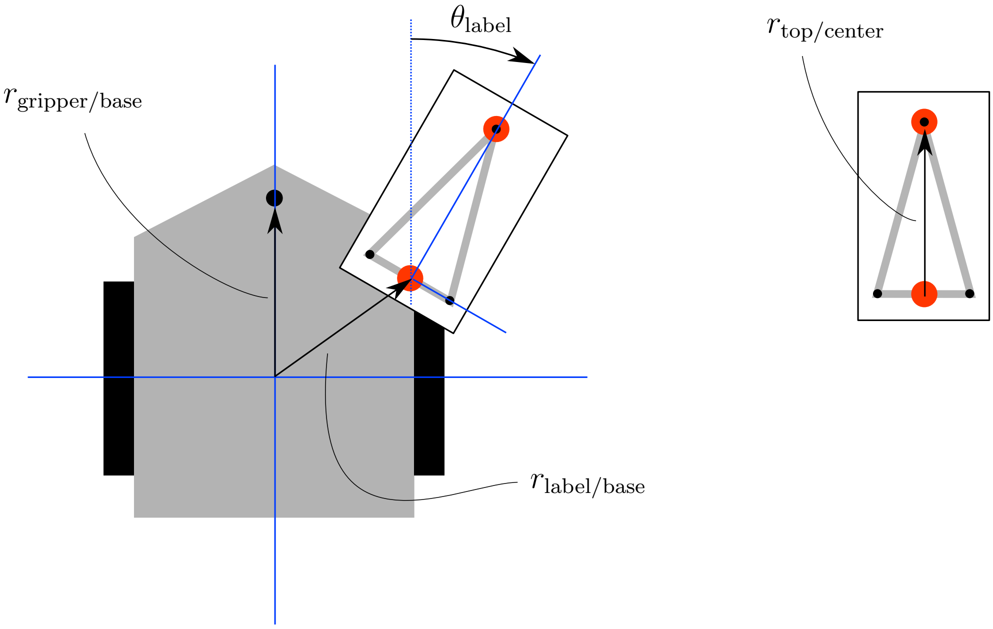

Figure 2: Local vehicle frame definitions

fix θ_{label} to fixed value to avoid computing sine/cos explicitly to get rotation matrix.

Define homogeneous transformation matrices between all steps to account for both displacement and rotation

3 Relative Agent Positions and Orientations