

The origin of MATLAB's coordinate system is at the top-left corner of the image. The positive x -axis goes horizontally right and the positive y -axis goes vertically down from the origin of MATLAB's coordinate system. This behaviour is quite similar to the provided image as it also has its positive axes going horizontally right and vertically down. Hence, a translation and scaling operation is needed to convert from MATLAB's coordinate system to the coordinate system of the graph. It also means the x and y transformations can be calculated independently.

Let's assume the vertical labelled line in graph is $x = m$ and the horizontal labelled line is $y = n$. Now, our points of interest are the end labels of each labelled line and the point of their intersection. The known coordinates of these points are shown in the table 1.

Point	Graph (x, y) coordinates	MATLAB (x, y) coordinates
1	$(m, 640)$	$(143, 1594)$
2	(m, n)	$(143, 1520)$
3	$(m, 540)$	$(143, 32)$
4	$(-10, n)$	$(411, 1520)$
5	$(10, n)$	$(730, 1520)$

Table 1: Comparison of graph and MATLAB coordinates for points of interest

Now, assume that we apply a scaling with the scaling factor s_x and s_y and then a shift with translation factor t_x and t_y . If (g_x, g_y) are the graph's coordinates and (m_x, m_y) are MATLAB's coordinates then they can be converted as follows:

$$\begin{aligned} g_x &= m_x \cdot s_x + t_x \\ g_y &= m_y \cdot s_y + t_y \end{aligned} \tag{1}$$

Note that the order of scaling and shifting doesn't matter as $(m + t) \cdot s \equiv m \cdot s + \underbrace{t \cdot s}_t$. Now to find s_x, s_y, t_x, t_y from 1, we can rewrite the data as a matrix

$$\begin{bmatrix} & 1594 & 1 \\ & 32 & 1 \\ 411 & 1 & \\ 730 & 1 & \end{bmatrix} \begin{bmatrix} s_x \\ t_x \\ s_y \\ t_y \end{bmatrix} = \begin{bmatrix} 640 \\ 540 \\ -10 \\ 10 \end{bmatrix} \tag{2}$$