# COMP116 – Work Sheet Two

## Associated Module Learning Outcome

> Ability to solve problems involving the outcome of matrix-vector products as might arise in standard transformations

### Vectors & Matrices

**Q1:** There are a number of ways of defining the "size" of an $n$-vector. These include, often regarded as the standard approach, what is called the **Euclidean** distance (also known as the $L_2$-norm) which is denoted $\|\mathbf{x}\|$ and, as we shall use, $\|\mathbf{x}\|_2$. For the $n$-vector, $\mathbf{x} = < x_1, x_2, \ldots, x_n >$, as described in the lectures (and on page 67 of the course textbook)

$$\| < x_1, x_2, \ldots, x_n > \|_2 \; = \; \sqrt{\sum_{k=1}^{n} |x_k|^2}$$

The *positive* square root being used. Although, strictly speaking the $|x_i|$ qualification is not needed as the operation of squaring renders this redundant, i.e $(-x_i)^2 = (x_i)^2 = |x_i|^2$, it is, however, useful to include this as it helps with describing generalizations.

Another widely used, in CS contexts, definition of "vector size" is the so-called *Manhattan distance* (also called the $L_1$-norm) which is denoted $\|\mathbf{x}\|_1$. This often arises in robot-motion control contexts in which the robot cannot turn about through arbitrary angles and is only able to move in one of four ways: continue in the direction being travelled; turn around and move in the opposite direction that had been used; turn $90^\circ$ right; turn $90^\circ$ left.

The Manhattan distance of the $n$-vector $\mathbf{x} = < x_1, x_2, \ldots, x_n >$ is

$$\| < x_1, x_2, \ldots, x_n > \|_1 \; = \; \sum_{k=1}^{n} |x_k|$$

One way of thinking of this is as moving in an $n$-dimensional grid with movement limited to being able to reach an immediately adjacent, i.e. connected by a "grid line" point from the current point.

The first collection of questions concerns differences between these two measures of distance.

a. Suppose that we have 4-vectors, $\mathbf{x} = < -2, 3, -5, 8 >$ and $\mathbf{y} = < 5, -3, 7, 10 >$. What is the value of the following quantities?

1. $\|\mathbf{x}\|_2$.
2. $\|\mathbf{x}\|_1$.
3. $\|\mathbf{y}\|_2$.
4. $\|\mathbf{y}\|_1$.

b. Similarly what are the values of the following quantities?

1. $\|\mathbf{x}\|_2 + \|\mathbf{y}\|_2$.
2. $\|\mathbf{x}\|_1 + \|\mathbf{y}\|_1$.
3. $\|\mathbf{x} + \mathbf{y}\|_2$.
4. $\|\mathbf{x} + \mathbf{y}\|_1$.

c. What relationship is suggested by your answer to (b1) compared with your answer to (b3). Similarly when comparing your answer to (b2) with that of (b4)?

d. Is the behaviour you notice in answering (c) indicative of a more general relationship between $\|\mathbf{x}\|_2+\|\mathbf{y}\|_2$ and $\|\mathbf{x}+\mathbf{y}\|_2$. Similarly between $\|\mathbf{x}\|_1 + \|\mathbf{y}\|_1$ and $\|\mathbf{x} + \mathbf{y}\|_1$.

e. What does your answer to (d) allow you to deduce (if anything) about combining **three** or more 4-vectors, i.e. $\|\mathbf{x}\|_2$, $\|\mathbf{y}\|_2$, $\|\mathbf{z}\|_2$ and $\|\mathbf{x} + \mathbf{y} + \mathbf{z}\|_2$? Is the same true if the Manhattan distance ($\|\mathbf{x}\|_1$) is used?

f. Suppose one tries to compute $\|\mathbf{x}\|_1$ for the 3-vector $\mathbf{x} =< x_1, x_2, x_3 >$ using a **matrix-vector** product $\mathbf{M} \cdot \mathbf{x}^\top$ with $\mathbf{M}$ the $1 \times 3$ matrix

$$\mathbf{M} \;=\; (1,\ 1,\ 1)$$

What important feature of the Manhattan distance is such an attempt **failing** to consider?

If $\mathbf{M}$ was replaced by the $8 \times 3$ matrix

$$\mathbf{P} \;=\; \begin{pmatrix} -1 & -1 & -1 \\ -1 & -1 & 1 \\ -1 & 1 & -1 \\ -1 & 1 & 1 \\ 1 & -1 & -1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \\ 1 & 1 & 1 \end{pmatrix}$$

How might $\mathbf{P} \cdot \mathbf{x}^\top$ (which is an $8 \times 1$-matrix) be used to find $\|\mathbf{x}\|_1$?

**2-D Graphics Transformations and Matrices**

**Q2** This question deals with a very basic 2-D vector-graphic game that involves moving a circular puck around a playing area 100 units wide and 100 units high. The general configuration is shown in Figure 1.

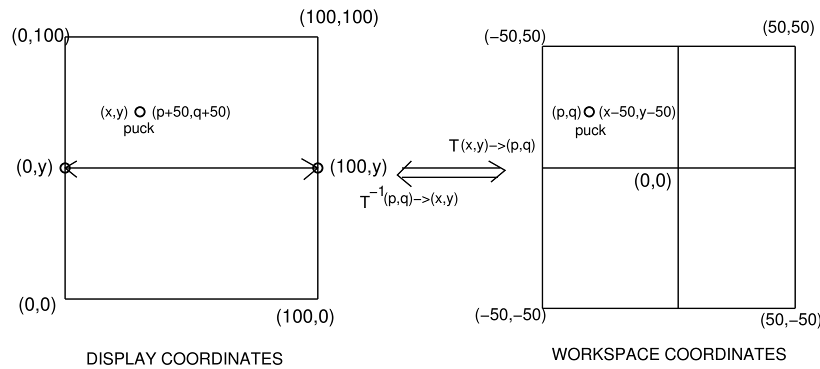


Figure 1: Simple Game Environment

Standard reference systems for graphical images will often use $x$ (Horizonal/width) values between $0$ and some maximum $W$ and, similarly $y$ (Vertical/height) values between $0$ and some maximum $H$.

This can, however, prove rather awkward as a framework for realizing effects (for example if integers less than $0$ could result). In such cases an animation effect might be thought of as carried out in three stages:

S1. Map the coordinate $(x, y)$ to be manipulated within the display system to a coordinate $(p, q)$ in another coordinate reference scheme.

S2. Compute the coordinates $(p', q')$ resulting by applying some transformation, $\mathbf{M}$ to $(p, q)$.

S3. Map the coordinate $(p', q')$ back to the corresponding coordinate $(x', y')$ in the display system.

In Figure 1 effects in the Display System are realized by mapping to the coordinate system in which $x$ values range between $-50$ and $50$; similarly $y$ values between $-50$ and $50$.

a. What are the **two** $3\times3$-matrices, $\mathbf{M}_{d-to-w}$ and $\mathbf{M}_{w-to-d}$ that should be used to translate the homogenous coordinate $< x, y, 1 >$ in the Display setting to the homogenous coordinate $< x - 50, y - 50, 1 >$ in Workspace system and *vice-versa*? That is for which

$$\begin{array}{rcl} \mathbf{M}_{d-to-w} \cdot < x, y, 1 >^\top & = & < x - 50, y - 50, 1 >^\top \\ \mathbf{M}_{w-to-d} \cdot < p, q, 1 >^\top & = & < p + 50, q - 50, 1 >^\top \end{array}$$

b. Consider the point $(0, y)$ (Display coordinates). Describe the animation effect (on the Display) of applying

$$\mathbf{M}_{w-to-d} \cdot \mathbf{F}_1 \cdot \mathbf{M}_{d-to-w} \cdot < 0, y, 1 >^\top$$

where $\mathbf{F}_1$ is the $3 \times 3$ matrix,

$$\mathbf{F}_1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 100 \\ 0 & 0 & 1 \end{pmatrix}$$

(for instance the effect could be displayed by drawing a line on the display joining $(x, y)$ to the new position).

c. Now suppose we introduce 2 new $3 \times 3$ matrices $\mathbf{F}_2$ and $\mathbf{F}_3$ with

$$\mathbf{F}_2 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad ; \quad \mathbf{F}_3 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & -100 \\ 0 & 0 & 1 \end{pmatrix}$$

Let

$$\begin{array}{rcl} \mathbf{T}_1 & = & \mathbf{M}_{w-to-d} \cdot \mathbf{F}_1 \cdot \mathbf{M}_{d-to-w} \\ \mathbf{T}_2 & = & \mathbf{M}_{w-to-d} \cdot \mathbf{F}_2 \cdot \mathbf{M}_{d-to-w} \\ \mathbf{T}_3 & = & \mathbf{M}_{w-to-d} \cdot \mathbf{F}_3 \cdot \mathbf{M}_{d-to-w} \end{array}$$

1. Describe the effect displayed after animating

$$\begin{array}{ll} \mathbf{T}_1 \cdot < 0, y, 1 >^\top & \text{then} \\ \mathbf{T}_2 \cdot \mathbf{T}_1 \cdot < 0, y, 1 >^\top & \text{then} \\ \mathbf{T}_3 \cdot \mathbf{T}_2 \cdot \mathbf{T}_1 \cdot < 0, y, 1 >^\top & \text{then} \\ \mathbf{T}_2 \cdot \mathbf{T}_3 \cdot \mathbf{T}_2 \cdot \mathbf{T}_1 \cdot < 0, y, 1 >^\top & \end{array}$$

2. Suppose, instead of applying these in the order described, the ordering was

$$\begin{array}{ll} \mathbf{T}_2 \cdot < 0, y, 1 >^\top & \text{then} \\ \mathbf{T}_3 \cdot \mathbf{T}_2 \cdot < 0, y, 1 >^\top & \text{then} \\ \mathbf{T}_2 \cdot \mathbf{T}_3 \cdot \mathbf{T}_2 \cdot < 0, y, 1 >^\top & \text{then} \\ \mathbf{T}_1 \cdot \mathbf{T}_2 \cdot \mathbf{T}_3 \cdot \mathbf{T}_2 \cdot < 0, y, 1 >^\top & \end{array}$$

   What would be the resulting effect?