1 Introduction to Shader Programs

1.1 Sending Data from a C++ Program to a Shader Program

There are three ways to send data to a shader program.

1. **Per-Vertex Vertex Attribute:** Here, a vertex attribute is a data value associated with a particular vertex. These are stored in the VBO, and generally different values for each vertex.
2. **Generic Vertex Attribute:** These are the same values for all vertices in a single draw command (e.g. `glDrawArrays` or `glDrawElements`). These can change between calls, though.
3. **Uniform Variables:** Essentially the same as generic vertex attributes.

Note that the values in (1) and (2), vertex attributes, can be accessed only by the vertex shader. However, uniform values, like (3), can be accessed by both the vertex and fragment shader.

1.1.1 Per-Vertex Attributes

We make use of the `glBindBuffer` and `glBufferData` commands to load vertex attributes into a VBO. We also use `glVertexAttribPointer` to tell the VAO the information about what vertex attributes are in the VBO, and where they are. We can also use `glEnableVertexAttribArray` to enable the data from the VBO.

1.1.2 Generic Attributes

We use `glDisableVertexAttribArray` and `glVertexAttrib3f`[1] to set a generic attribute's value.

1.1.3 Uniform Variables

For matrices, we use `glUniformMatrixfv` to set the values in a 4×4 matrix in column order. We use `GetUniformLocation` to get the "location number" of the uniform variable.

[1] `3f` means three floats.

2 Graphics Pipeline, Linear & Affine Transformations in $\mathbb{R}^2$

The graphics pipelines, in the simplest form, has the following series of transformations which are applied to points or objects to place them on the screen for viewing. Let $\mathbf{x}$ be a point in $\mathbb{R}^2$ or $\mathbb{R}^3$, and we denote this as a *local coordinate*.

- Generally, we start by transforming this by using a model matrix M; that is, $M\mathbf{x}$ in *world coordinates*.
- Then, to view this from some particular point of view, we take some matrix V and apply it to $M\mathbf{x}$ like so:
$$V(M\mathbf{x}) = (VM)\mathbf{x}.$$
These are known as the *view coordinates*, and we denote V as the *view matrix*.
- Finally, there is a projection matrix P, which can be applied like so:
$$PVM\mathbf{x}.$$
This gives us the *screen coordinates*.
- As a final step, there is the *perspective division*, which we'll learn about later.

Note that M, V, and R are generally 4×4 matrices.

Suppose that we're modeling a ferris wheel, and suppose it has a chair centered at the origin. Let $\mathbf{x}$ be some point of the chair.

- We might use $M\mathbf{x}$ to move the chair to the appropriate position (world coordinates) on the ferris wheel, for example as the ferris wheel is moving.
- Then, $VM\mathbf{x}$ gives the position of the ferris wheel relative to the viewer, and describes where the chair is relative to the viewer.
- $PVM\mathbf{x}$ would map the point to some pixel on the screen as we're rendering the image.

2.1 Linear Transformations

We begin by talking about linear transformations in $\mathbb{R}^2$. Points in $\mathbb{R}^2$ are given by xy-coordinates. Let us denote a point as $\mathbf{x} = \langle x_1, x_2 \rangle = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$. Points are generally written as column vectors.

Definition 2.1: Transformation

A **transformation** is a mapping (or function)
$$A : \mathbb{R}^2 \mapsto \mathbb{R}^2$$
A transformation A is **linear** if the following two things hold:

1. $A(\alpha\mathbf{x}) = \alpha A(\mathbf{x})$ for all $\alpha \in \mathbb{R}$ and $\mathbf{x} \in \mathbb{R}^2$.
2. $A(\mathbf{x} + \mathbf{y}) = A(\mathbf{x}) + A(\mathbf{y})$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$.

Theorem 2.1

If A is linear, then $A(\mathbf{0}) = \mathbf{0}$ where $\mathbf{0} = \langle 0, 0 \rangle$.

Proof. $A(\mathbf{0}) = A(0\mathbf{x}) = 0A(\mathbf{x}) = 0$. □

(Example.) Let
$$A(\langle x, y\rangle) = \langle -y, x\rangle.$$
This transformation is a rotation of 90 degrees ($\pi/2$ radians) counter-clockwise around the origin. Note that we will name this transformation as R_{90° or $R_{\frac{\pi}{2}}$ depending on whether we're working with degrees or radians.

(Example.) Let
$$A(\langle x, y\rangle) = \langle x + y, y\rangle.$$
This is also linear.

(Example.) Let
$$A(\langle x, y\rangle) = \langle x, y\rangle.$$
This is known as the identity function, and is often named as I.

(Example.) Let
$$A(\langle x, y\rangle) = \mathbf{0}.$$
This is known as the zero transformation.

2.2 Affine Transformations

Recall that a translation is a mapping which moves points around without changing its size or shape or orientation; it only changes its position.

Definition 2.2: Translation

A **translation** is a transformation
$$A : \mathbb{R}^2 \mapsto \mathbb{R}^2$$
such that
$$A(\mathbf{x}) = \mathbf{x} + \mathbf{u}$$
for some fixed point $\mathbf{u} \in \mathbb{R}^2$. This is called $T_{\mathbf{u}}$. So, $T_{\mathbf{u}}(\mathbf{x}) = \mathbf{x} + \mathbf{u}$.

Remark: For $\mathbf{u} = \mathbf{0}$, $T_{\mathbf{u}} = T_{\mathbf{0}} = I$ is the identity transformation. However, for $\mathbf{u} \neq \mathbf{0}$, $T_{\mathbf{u}}$ is not linear since, for instance, $T_{\mathbf{u}}(\mathbf{0}) = \mathbf{0} + \mathbf{u} = \mathbf{u} \neq \mathbf{0}$.

Definition 2.3: Affine Transformation

An **affine transformation** A is a translation of the form
$$A(\mathbf{x}) = B(\mathbf{x}) + \mathbf{u}$$
where B is a linear transformation and $\mathbf{u} \in \mathbb{R}^2$.

Remark: Translations are affine transformations. In fact, every linear transformation is an affine transformation because we can take $\mathbf{u} = \mathbf{0}$.

Definition 2.4: Composition

If A and B are transformations, then their **composition** $A \circ B$ is defined by

$$(A \circ B)(\mathbf{x}) = A(B(\mathbf{x})).$$

In particular, an affine transformation has the form

$$T_{\mathbf{u}} \circ B,$$

where B is a linear transformation and $\mathbf{u} \in \mathbb{R}^2$. This is because $T_{\mathbf{u}} \circ B = T_{\mathbf{u}}(B(\mathbf{x})) = B(\mathbf{x}) + \mathbf{u}$.

2.3 Visualizing Linear & Affine Transformations

We'll now visualize actions of a transformation on an "F"-shape. Figure (1) below shows this "F"-shape.

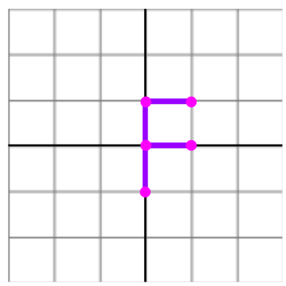

Figure 1

We assume that the grid lengths are unit lengths.

(Example.) Consider the transformation

$$A : \langle x, y \rangle \mapsto \langle -y, x \rangle.$$

Using the "F"-shape above, this transformation maps it to:

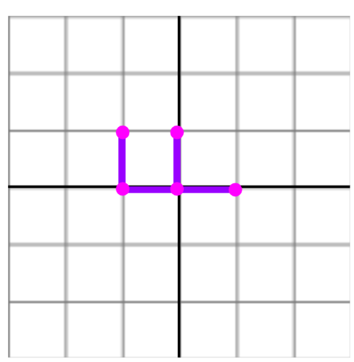

Figure 2

(Example.) Consider the *shearing* transformation

$$A : \langle x, y\rangle \mapsto \langle x + y, y\rangle.$$

Using the "F"-shape above, this transformation maps it to:

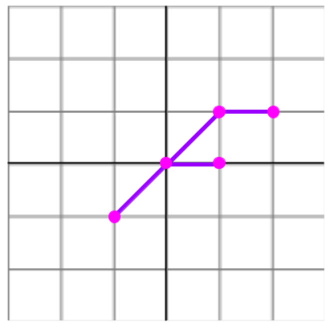

Figure 3

(Example.) Consider the *scaling* transformation[a]

$$A : \langle x, y\rangle \mapsto \langle 2x, \frac{1}{2}y\rangle \qquad S_{\langle 2,\frac{1}{2}\rangle}.$$

Using the "F"-shape above, this transformation maps it to:

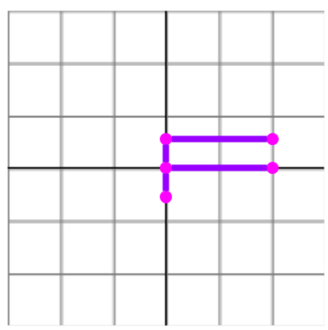

Figure 4

[a] If α and β are scalars that scale the x- and y-component, respectively, then $S_{\alpha,\beta}(\langle x, y\rangle) = \langle \alpha x, \beta y\rangle$.

(Example.) Consider the following affine transformation

$$A(\mathbf{x}) = (T_{\langle 2,1\rangle} \circ R_{90^\circ})(\mathbf{x}) = A(\mathbf{x}) = R_{90^\circ}(\mathbf{x}) + \langle 2, 1\rangle.$$

The idea is that, for $\langle x_0, x_1\rangle \in \mathbb{R}^2$, we have

$$A(\langle x_0, x_1\rangle) = \langle -x_1, x_0\rangle + \langle 2, 1\rangle = \langle -x_1 + 2, x_0 + 1\rangle.$$

Using the "F"-shape above, this transformation maps it to:

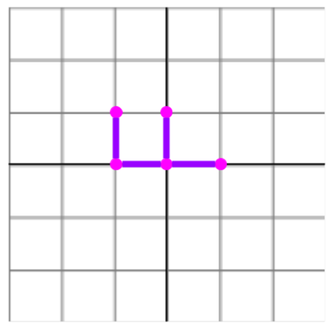

Figure 6

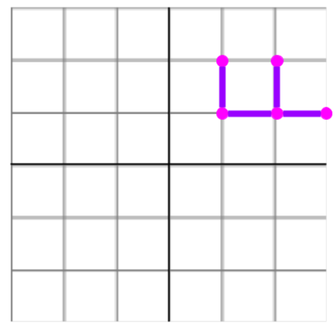

Figure 7

Here, figure (6) shows the rotation by 90 degrees counter-clockwise, while figure (7) shows the translation.

2.4 Matrix Representation & Rotations

In this section, assume that:

- $\mathbb{R}^2$ is the 2-dimensional vector space.
- Points $\mathbf{x}$ and $\mathbf{y}$ are column vectors $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \langle x_1, x_2 \rangle$.
- The linear transformation is defined by the mapping $A : \mathbb{R}^2 \mapsto \mathbb{R}^2$.
- The standard basis $\mathbf{i}$ and $\mathbf{j}$ is given by

$$\mathbf{i} = \langle 1, 0 \rangle \qquad \mathbf{j} = \langle 0, 1 \rangle.$$

- Let $\mathbf{u} = A(\mathbf{i})$ and $\mathbf{v} = A(\mathbf{j})$, where

$$\mathbf{u} = \langle u_1, u_2 \rangle \qquad \mathbf{v} = \langle v_1, v_2 \rangle.$$

Proposition. *Let* $M = \begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \end{bmatrix}$. *This represents* A *in that*

$$M\mathbf{x} = A(\mathbf{x}).$$

Proof. First, note that

$$M\mathbf{x} = \begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} u_1x_1 + v_1x_2 \\ u_2x_1 + v_2x_2 \end{bmatrix}.$$

On the other hand,

$$\begin{aligned} A(\mathbf{x}) &= A(x_1\mathbf{i} + x_2\mathbf{j}) \\ &= A(x_1\mathbf{i}) + A(x_2\mathbf{j}) && \text{By linearity.} \\ &= x_1A(\mathbf{i}) + x_2A(\mathbf{j}) && \text{By linearity.} \\ &= x_1\mathbf{u} + x_2\mathbf{v} \\ &= x_1\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + x_2\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \\ &= \begin{bmatrix} u_1x_1 + v_1x_2 \\ u_2x_1 + v_2x_2 \end{bmatrix}. \end{aligned}$$

Thus, we are done. □

Remark: The moral of the story is that we look at the action of the transformation on the standard basis vectors, which gives us values **u** and **v**; the columns of the matrix M are equal to the images (the values) of A on the vectors **i** and **j**.

(Example.) Consider the linear transformation A that takes the "F"-shape from Figure (1) and transforms it to the following "F"-shape:

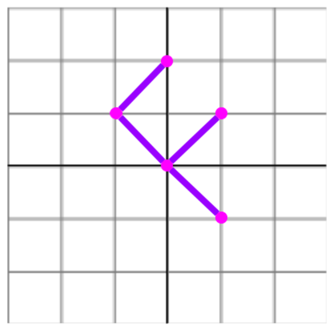

Figure 8

We want to find the matrix representing A. To do so, we consider the image of the standard basis vectors $\mathbf{i} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\mathbf{j} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. So:

Old Point (Figure 1)	**Transformed Point, i.e. Image** (Figure 8)	**Vector Form**
$\mathbf{i} = \langle 1, 0 \rangle$	$\langle 1, 1 \rangle$	$\mathbf{u} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$
$\mathbf{j} = \langle 0, 1 \rangle$	$\langle -1, 1 \rangle$	$\mathbf{v} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$

Thus, the matrix representing A is $\begin{bmatrix} \mathbf{u} & \mathbf{v} \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$.

(Example.) Let R_θ be the transformation by rotating counter-clockwise (CCW) around the origin[a] by angle θ. We want the matrix representation of R_θ. So, we just need to see what R_θ does to the standard basis vectors.

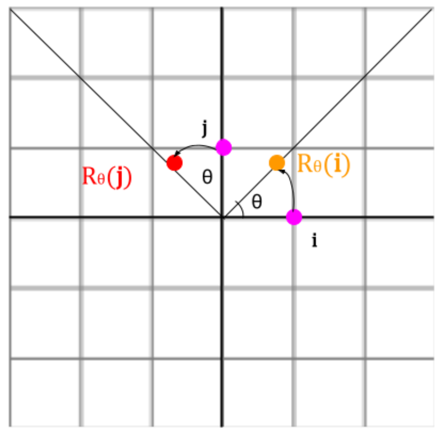

Figure 9

Here, we have

$$R_\theta(\mathbf{i}) = \langle \cos\theta, \sin\theta \rangle \qquad R_\theta(\mathbf{j}) = \langle -\sin\theta, \cos\theta \rangle.$$

This gives us the matrix

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$

What does this mean for representing affine transformations?

[a]Think of it as if you have a pin which is on the origin. Then, you're rotating the figure about the origin, the pin.

(Example.) Consider the *affine* transformation A that takes the "F"-shape from Figure (1) and transforms it to the following "F"-shape:

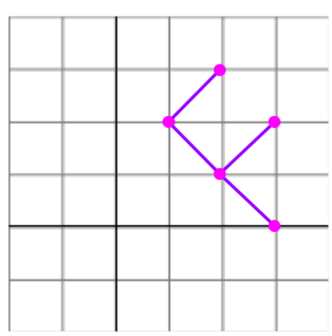

Figure 10

In the previous examples under this section, the origin of the transformed "F"-shape was still on the origin. So, the vector image was always given by $\mathbf{u} = A(\mathbf{i}) - \mathbf{i} = \langle i_0 - 0, i_1 - 0\rangle$ and $\mathbf{v} = A(\mathbf{j}) - \mathbf{j} = \langle j_0 - 0, j_1 - 0\rangle$. This is not the case here.

We note that the $\mathbf{i}$ vector from the original "F"-shape (that is, from $\langle 0,0\rangle$ to $\langle 1,0\rangle$) has been mapped to the image $A(\mathbf{i})$ vector (that is, from $\langle 2,1\rangle$ to $\langle 3,2\rangle$). We see that $A(\mathbf{i})$ is thus the vector $\langle 3-2, 2-1\rangle = \langle 1,1\rangle$. Note that we just took the difference of the two components of the points in the image.

Likewise, the $\mathbf{j}$ vector from the original "F"-shape (that is, from $\langle 0,0\rangle$ to $\langle 0,1\rangle$) has been mapped to the image $A(\mathbf{j})$ vector (that is, from $\langle 2,1\rangle$ to $\langle 1,2\rangle$). We see that $A(\mathbf{j})$ is thus the vector $\langle 1-2, 2-1\rangle = \langle -1,1\rangle$.

We also note that there is a translation of $\langle 2,1\rangle$. Therefore, $A(\mathbf{x})$ is given by

$$A(\mathbf{x}) = \overbrace{\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}^{\text{Linear part.}} + \underbrace{\begin{bmatrix} 2 \\ 1 \end{bmatrix}}_{\text{Translation.}} .$$