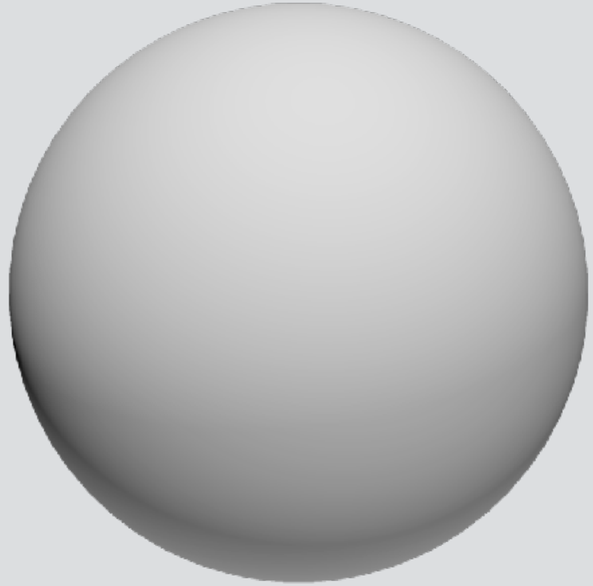
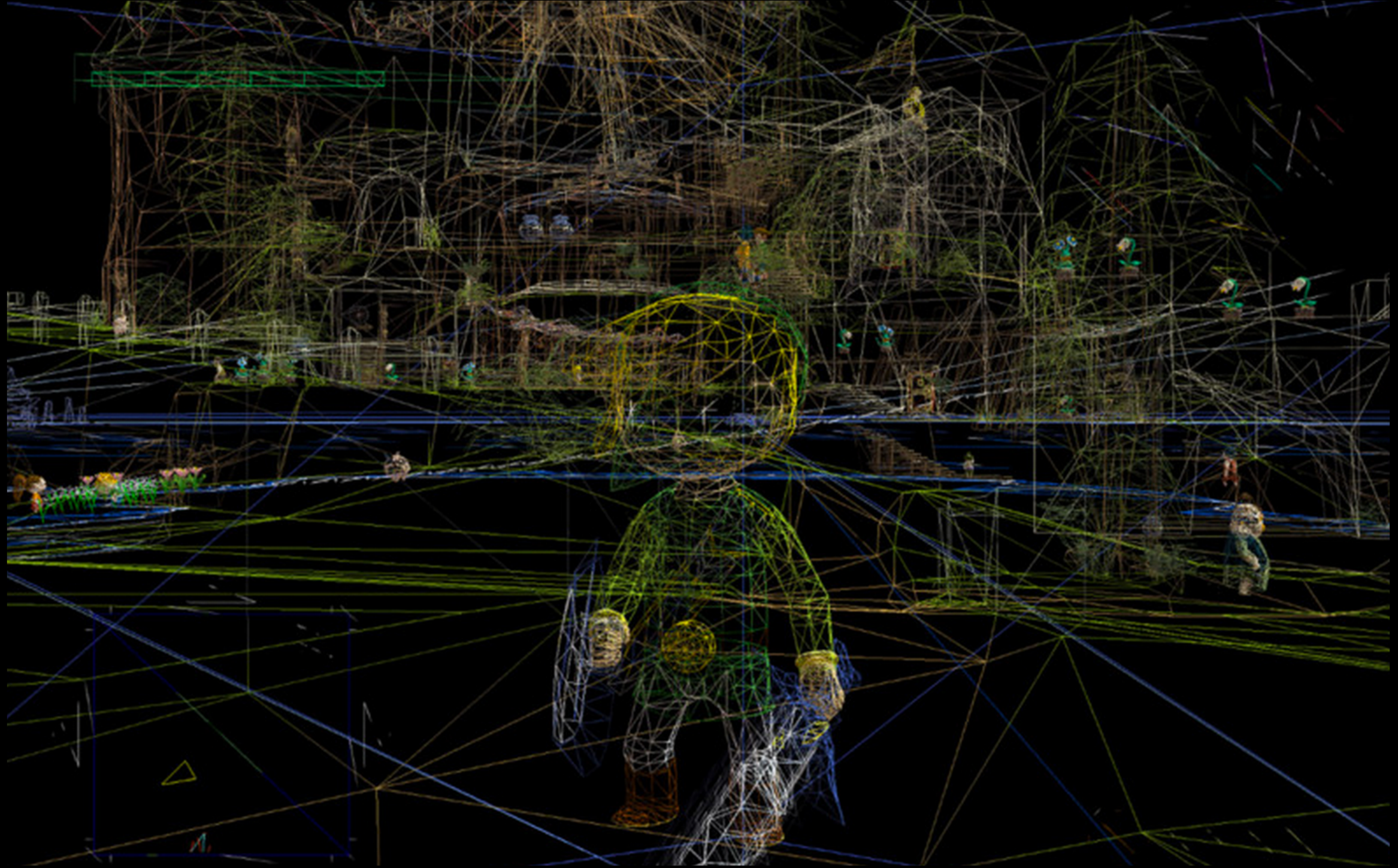
Graphics Foundations

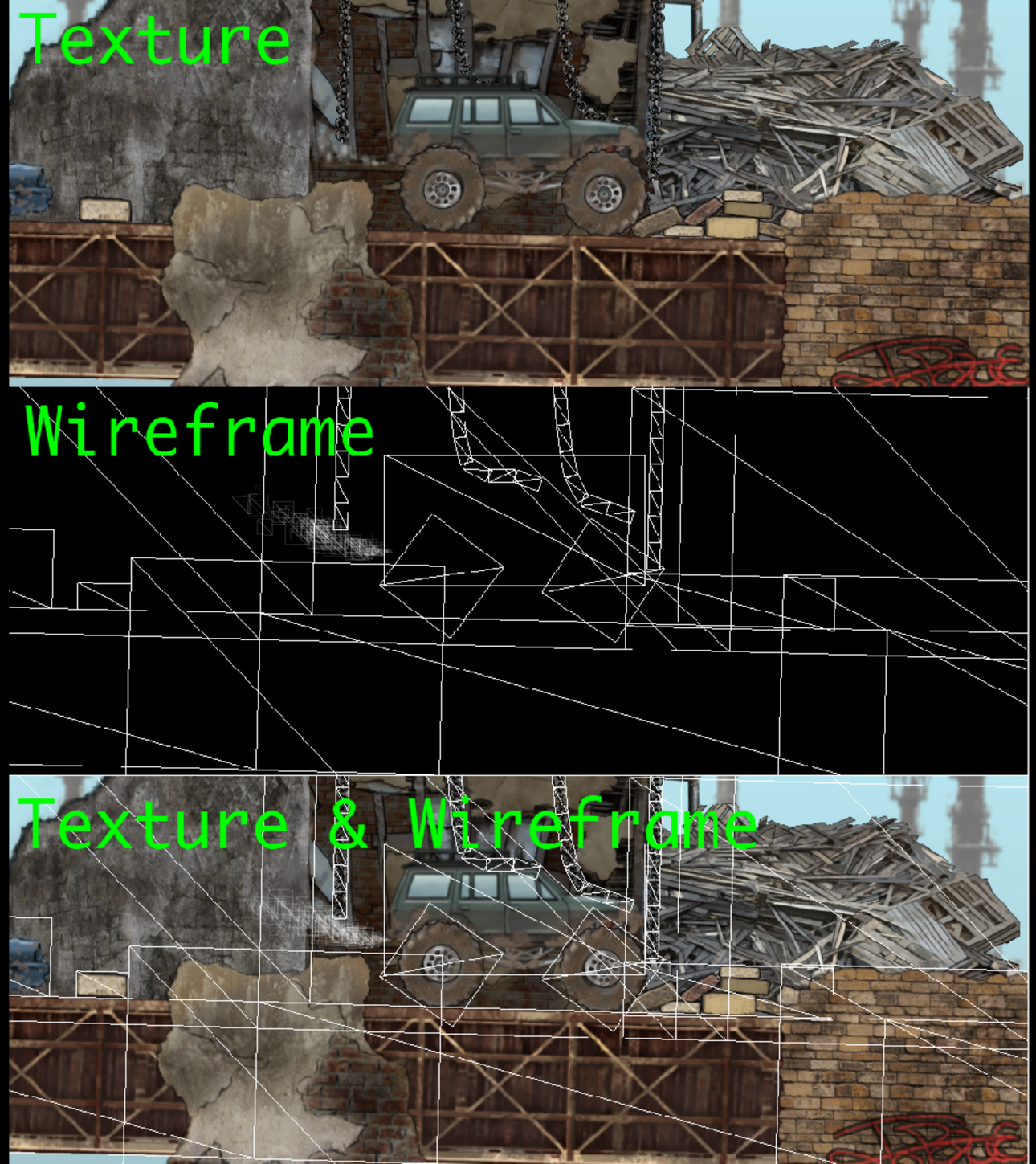


CS 3113

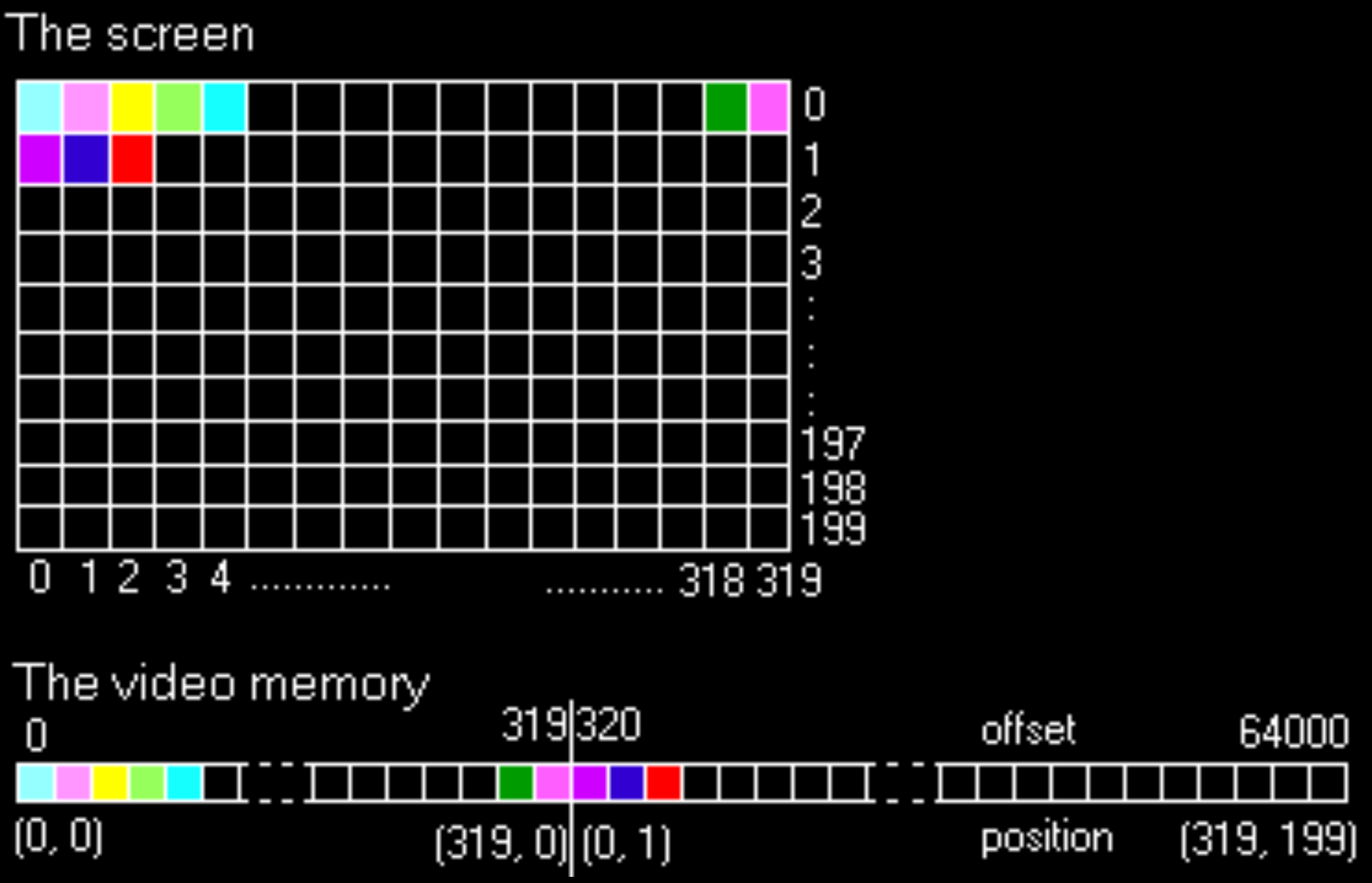
Graphics in games


Defend
Put Away
72
FREE
4168

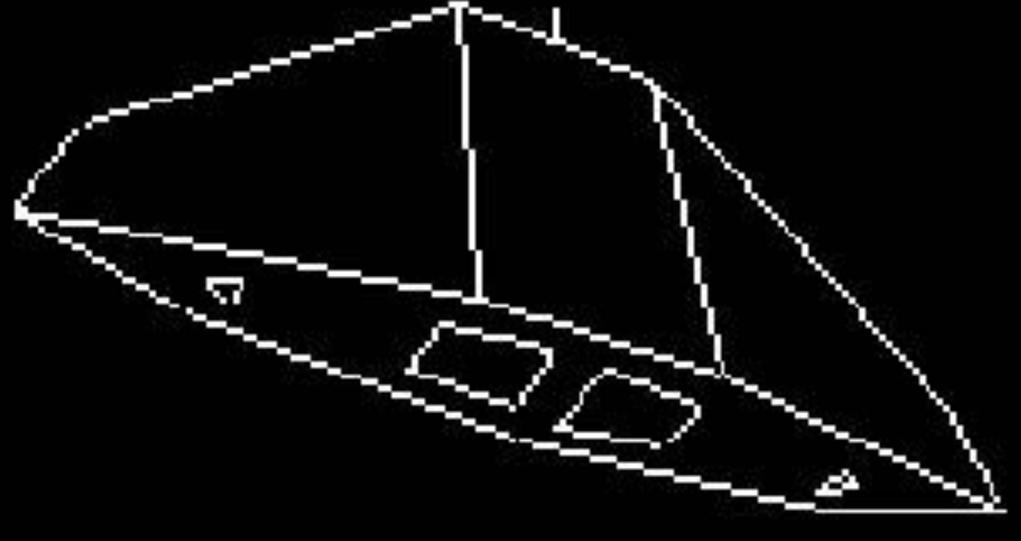



Texture
Wireframe
Texture & Wireframe

Why triangles?


The screen
0
1
2
3
197
198
199
0 1 2 3 4 .... 318 319
The video memory
0
319
320
offset
64000
(0, 0)
(319, 0)
(0, 1)
position
(319, 199)

Software rendering



Load New Commander (Y/N)?

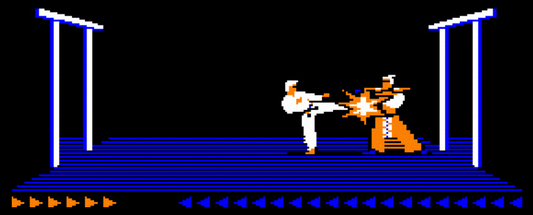

SPD: 0
VSI: 01.78
THR:90%
HDG:0
ALT:83

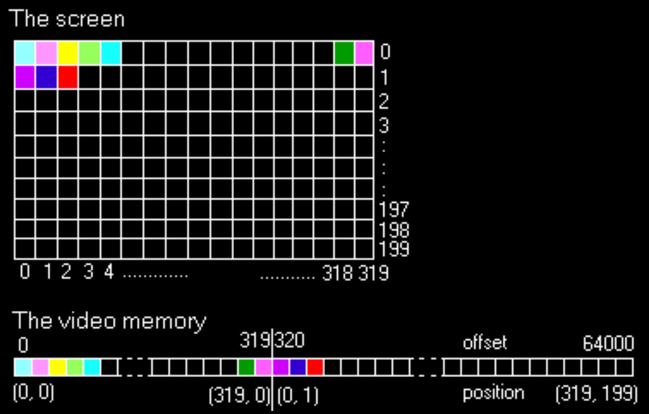

93
75
13
0
78
91
13


LEVEL UP
CHAR
QUESTS
MAP
MENU
1
2
3
4
5
6
7
8
RED STORM
TOTAL KILLS : 19
RESISTS : LIGHTNING
IMMUNE : MAGIC
INV
SPELLS

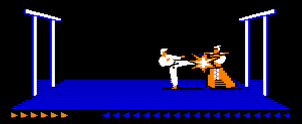
Software Rendering

RAM

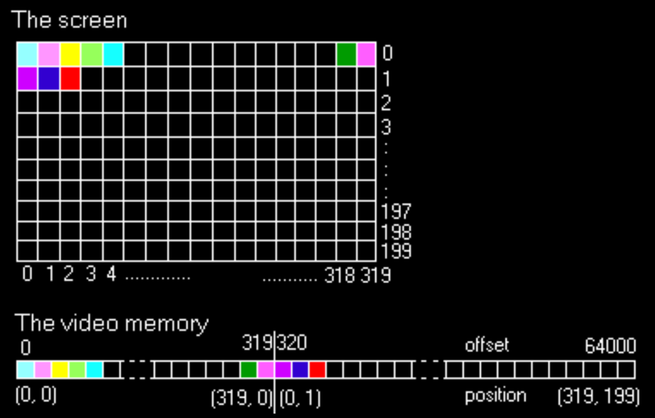




Video card



Software Rendering



2560 * 1600 = 4,096,000 pixels

~12 MB == 4 * sizeof(War and Peace)

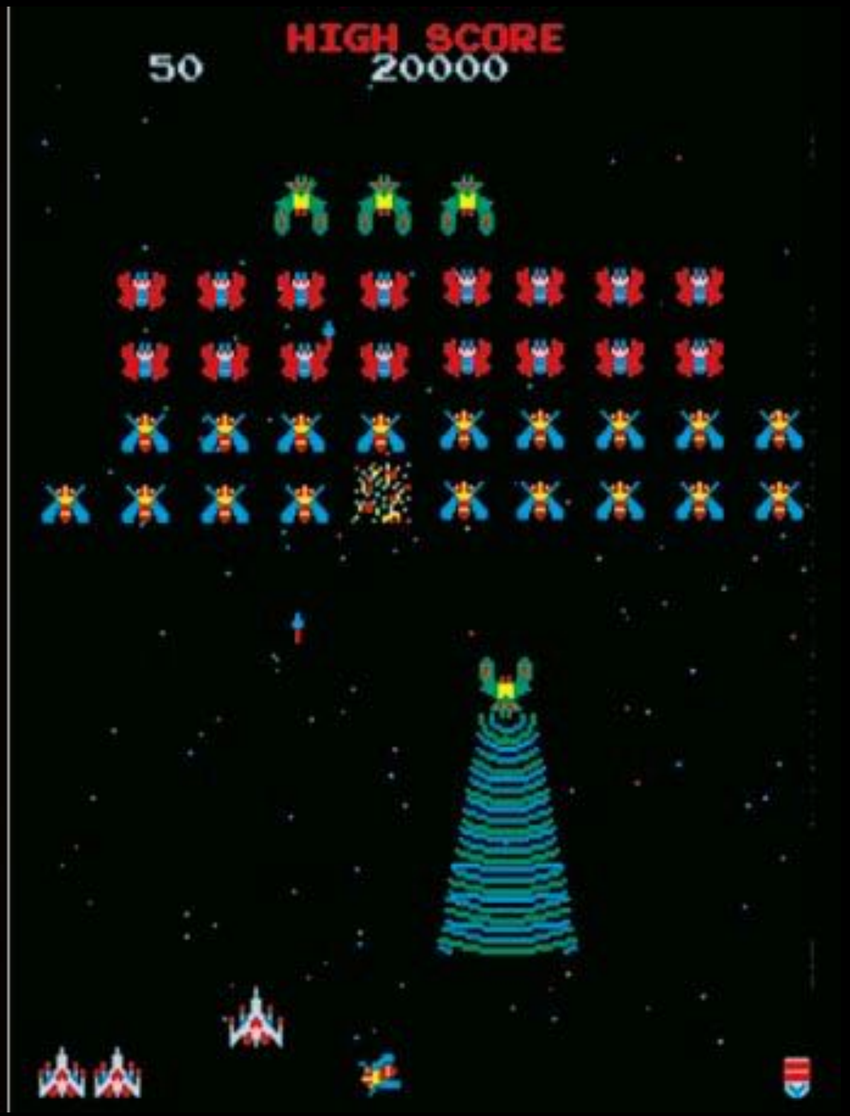
60 times a second

Hardware rendering


Galaga
DOUBLE DONKEY KONG
Nintendo
MS. PAC-MAN
MIDWAY
Bally
Home Theater of Arcades


HIGH SCORE
1050
1050


1UP
340
HIGH SCORE
340


HIGH SCORE
50
20000


Nintendo
ENTERTAINMENT SYSTEM™
POWER
RESET
Nintendo

MARIO
000000

×00

WORLD
1-1

TIME

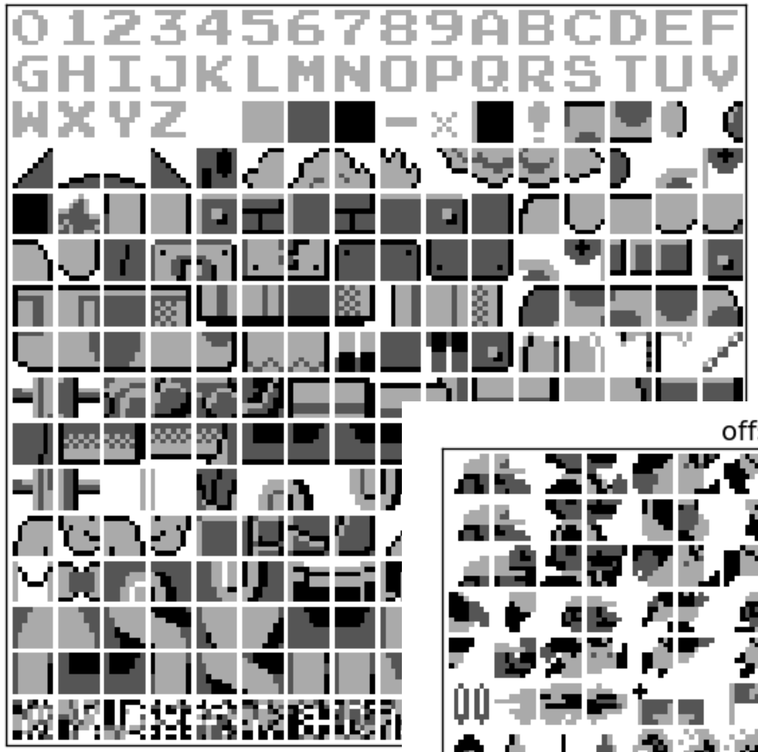
SUPER MARIO BROS.

1 PLAYER GAME

2 PLAYER GAME

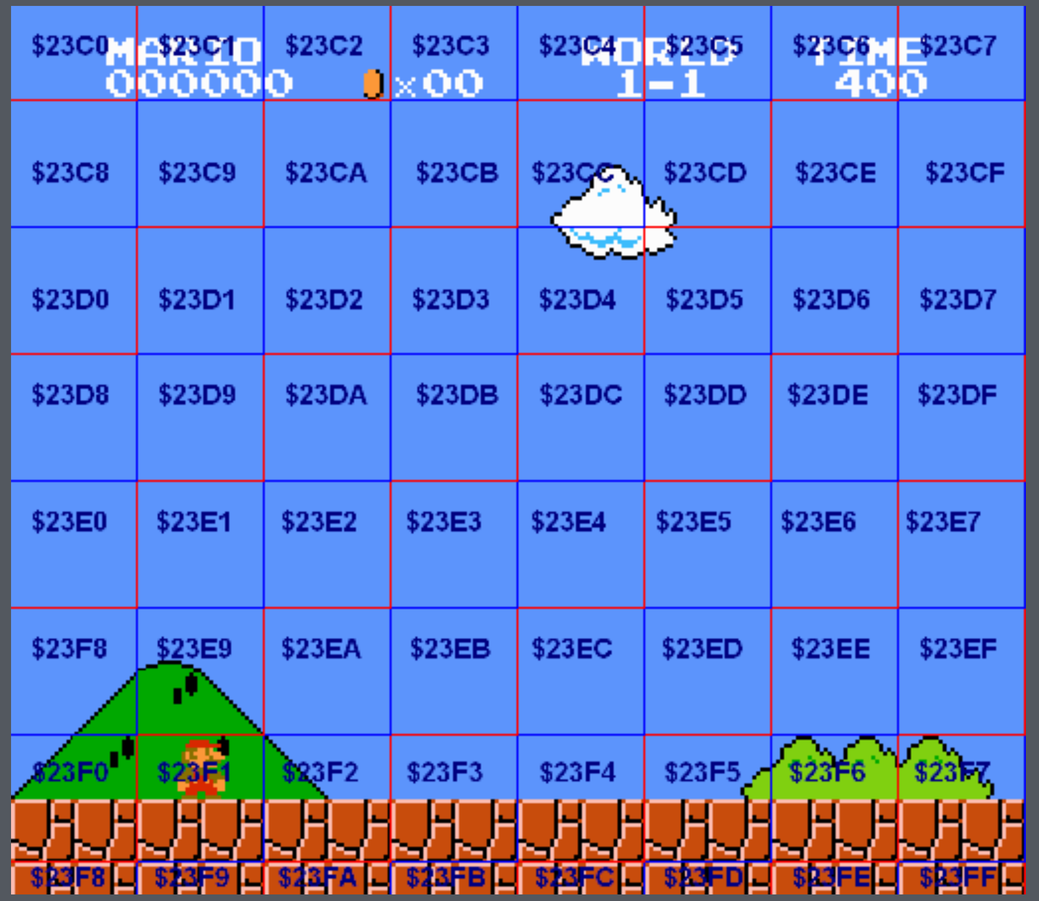
TOP- 000000

offset = 2305

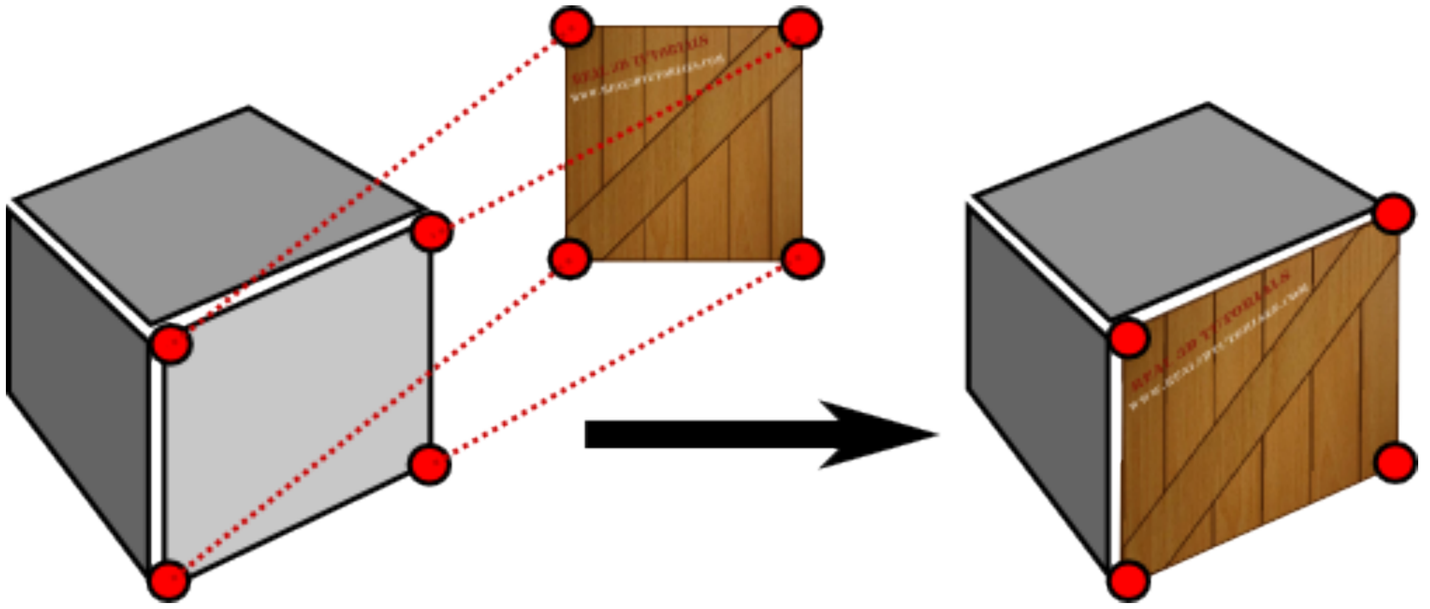


offset = 2049

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F




DAYTONA
USA
RACE LEADER
RACE LEA
50
2ND/2
50
1ST/2




12MB
BLASTER
Hardware XTRA AWARD
PC-Games 05/98
Hardware XTRA AWARD
PC-Action 05/98
PC Player
PC-Player 05/98
FASZINIERENDE SPIELE
4
FASZINIERENDE SPIELE
Voodoo 2
Get the Magic of Speed!
Basiert auf dem neuen Voodoo2 Graphics™ Chipsatz von 3Dfx Interactive™
Ausgestattet mit 12 MB Hochleistungsspeicher für stärkste Leistung
Bis zu 50 Milliarden Operationen und 3 Millionen Triangles pro Sekunde
Bis zu dreimal schnellere 3D-Verarbeitung als beim ursprünglichen Voodoo Graphics™-Chipsatz!
Arbeitet mit Ihrer vorhandenen Grafikkarte zusammen und bietet Ihnen das schnellste 3D-Spiel aller Zeiten
Enthält 4 topaktuelle Spiele
INCOMING
actua soccer 2
ULTIMATE RACE PRO
CREATIVE
WWW.SOUNDBLASTER.COM


20
0
0
11
108

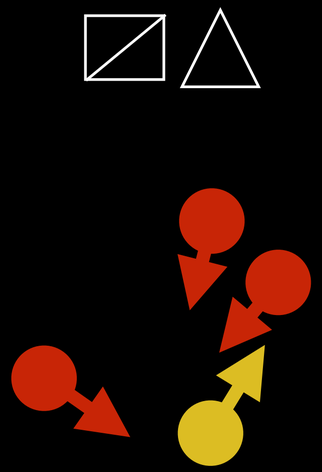

PARK
ONE W

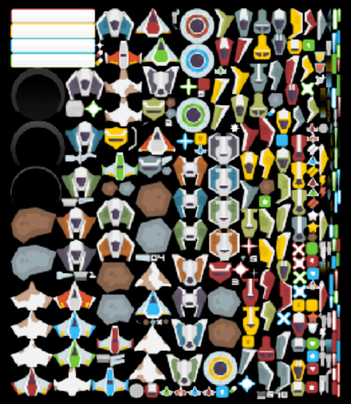
Hardware Rendering

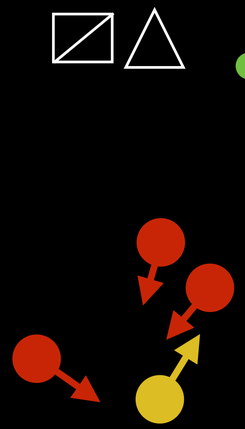
Hardware Rendering

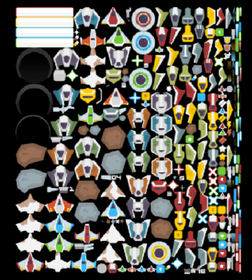
RAM

Video card



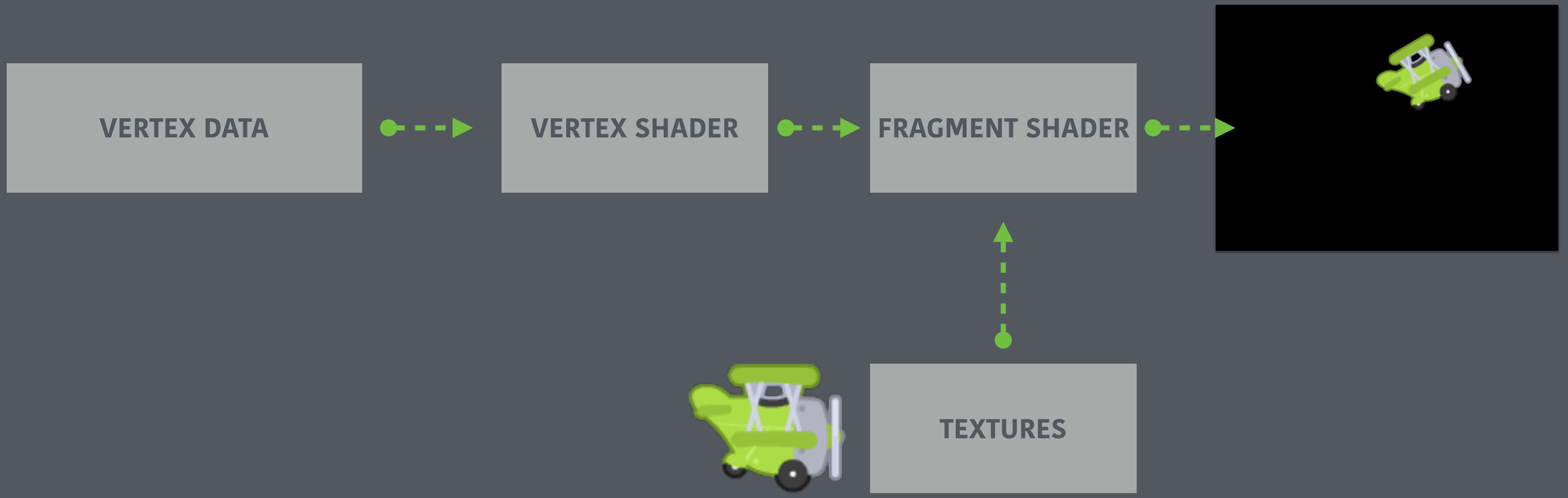




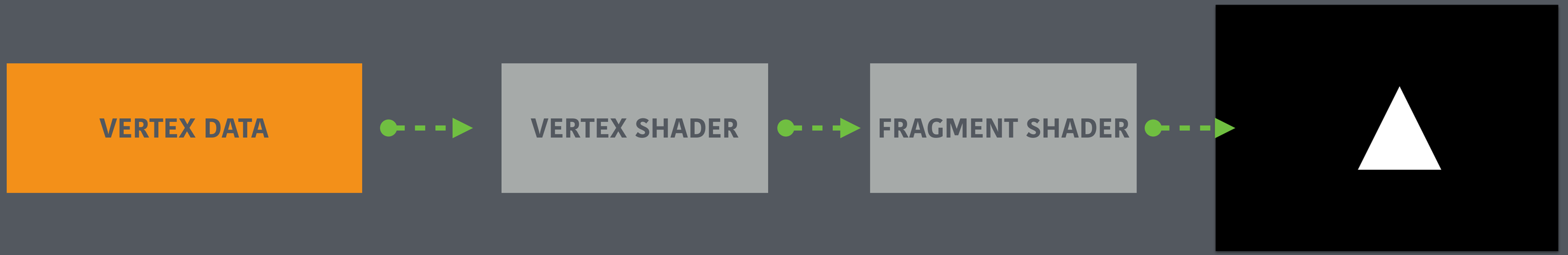


The GPU pipeline

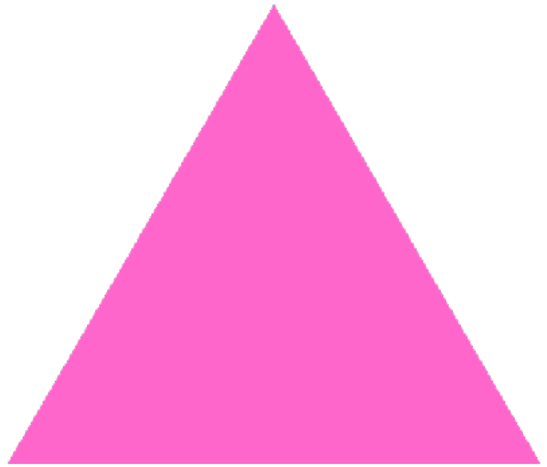
The **GPU** pipeline

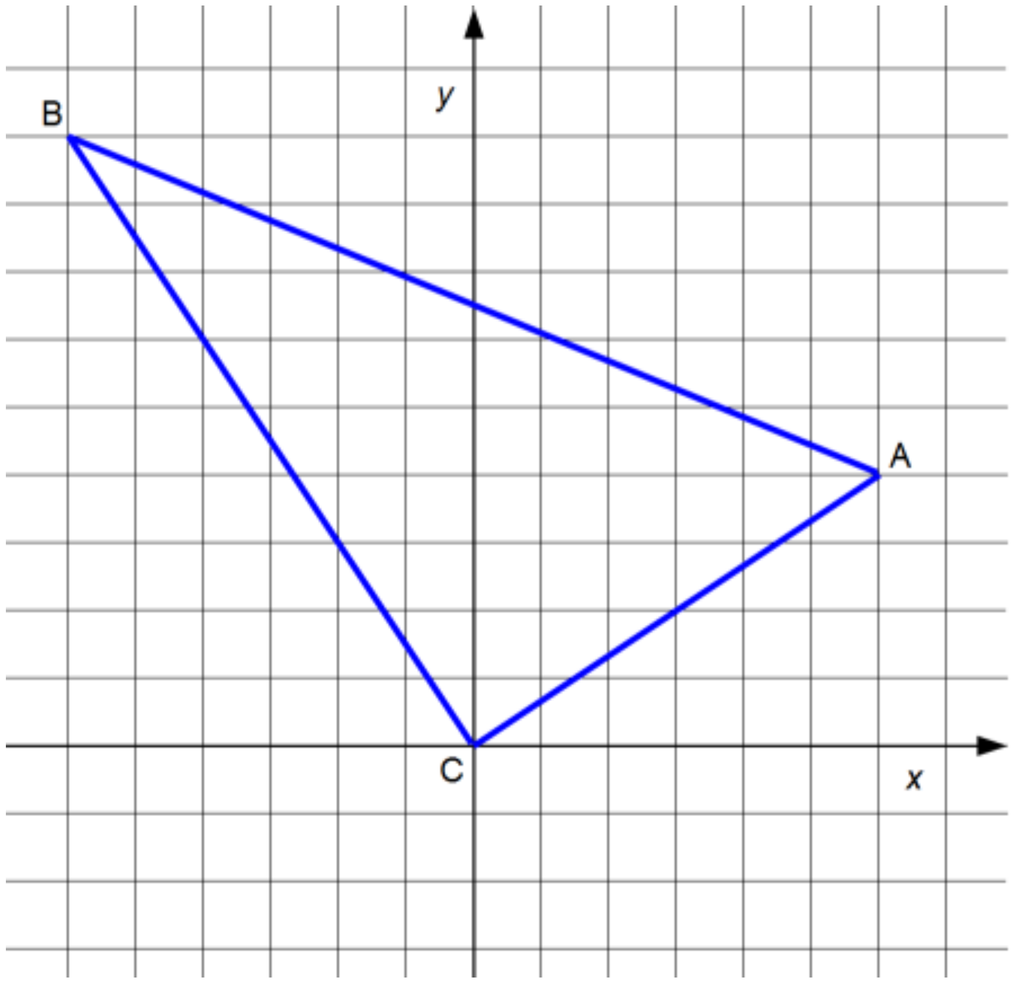


The **GPU** pipeline

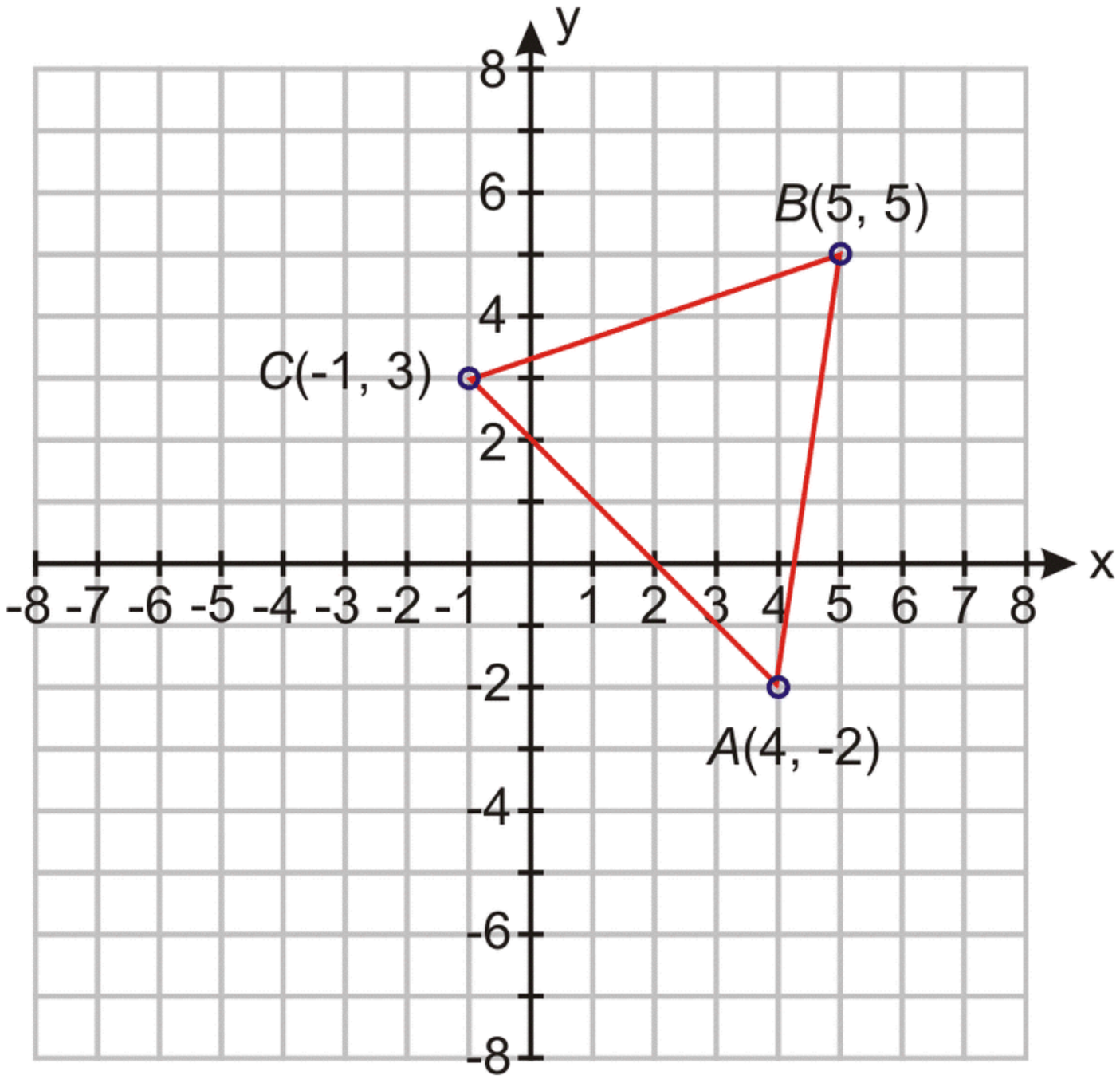


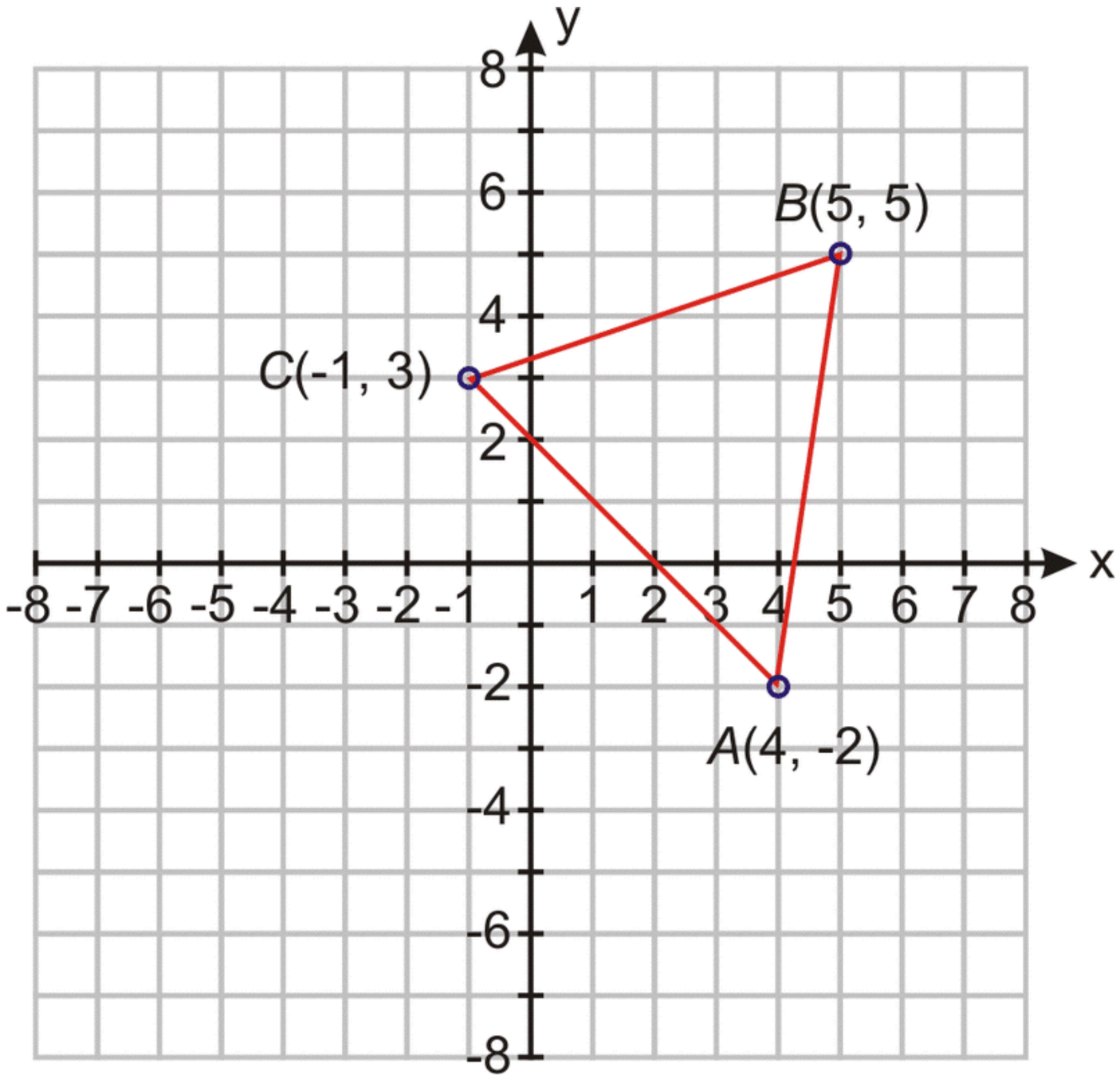
Vertex data




B
y
A
C
x

A polygon is defined by
points in space called vertices


y
x
8
6
4
2
-2
-4
-6
-8
-8 -7 -6 -5 -4 -3 -2 -1
1 2 3 4 5 6 7 8
B(5, 5)
C(-1, 3)
A(4, -2)

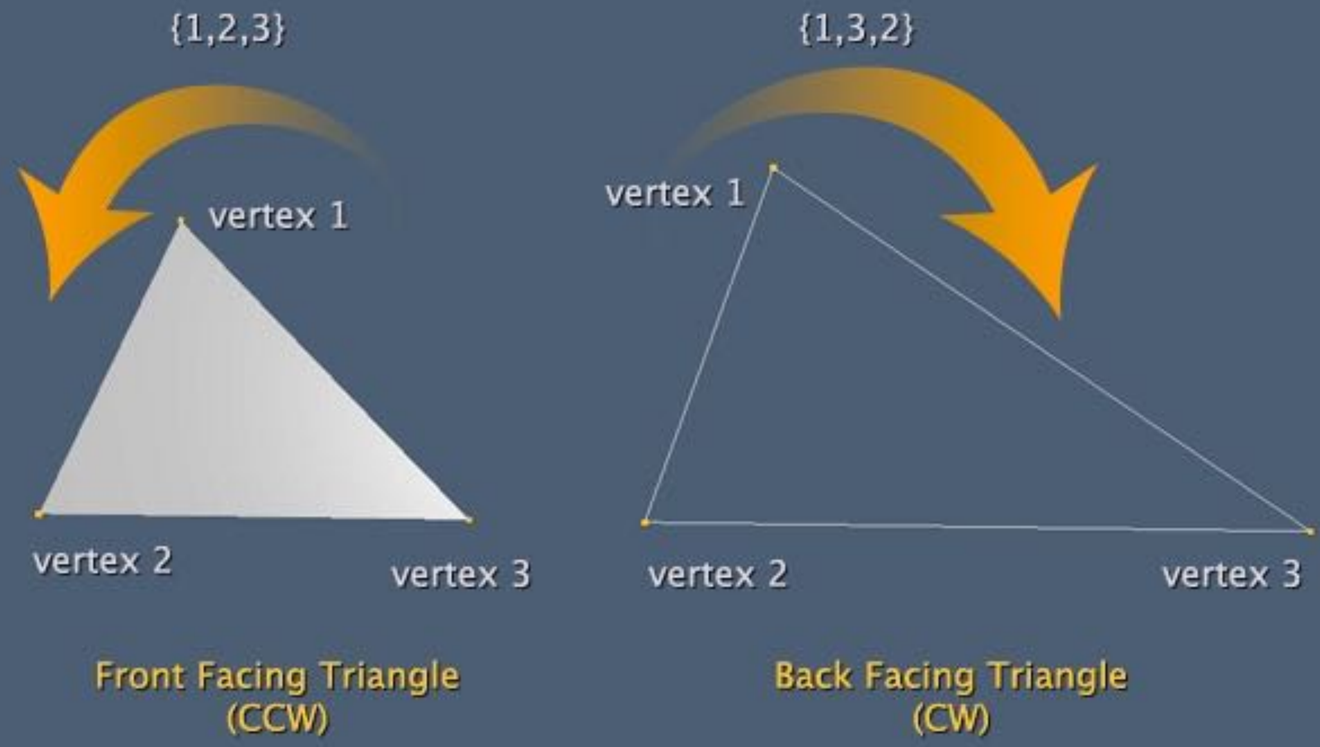


A: (4,−2)

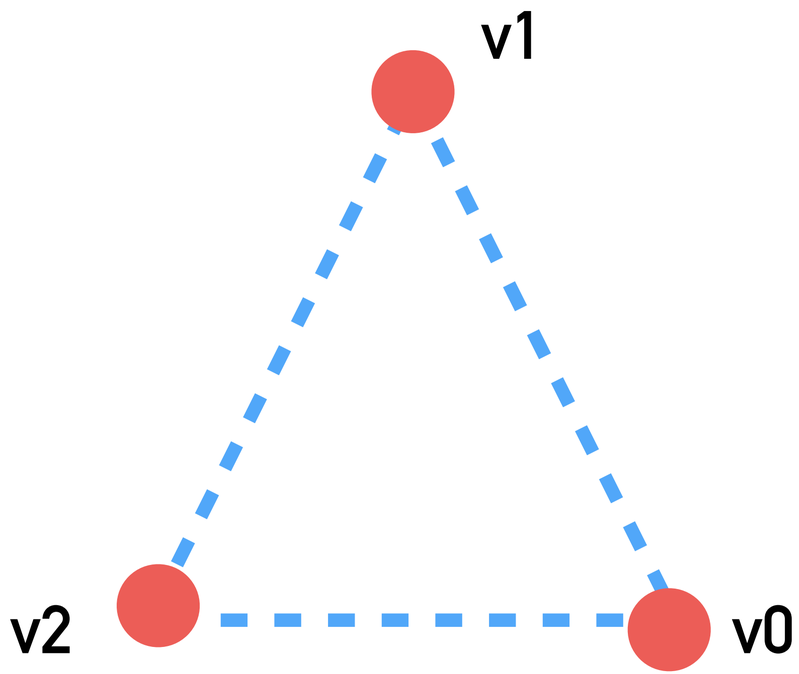
B: (5,5)

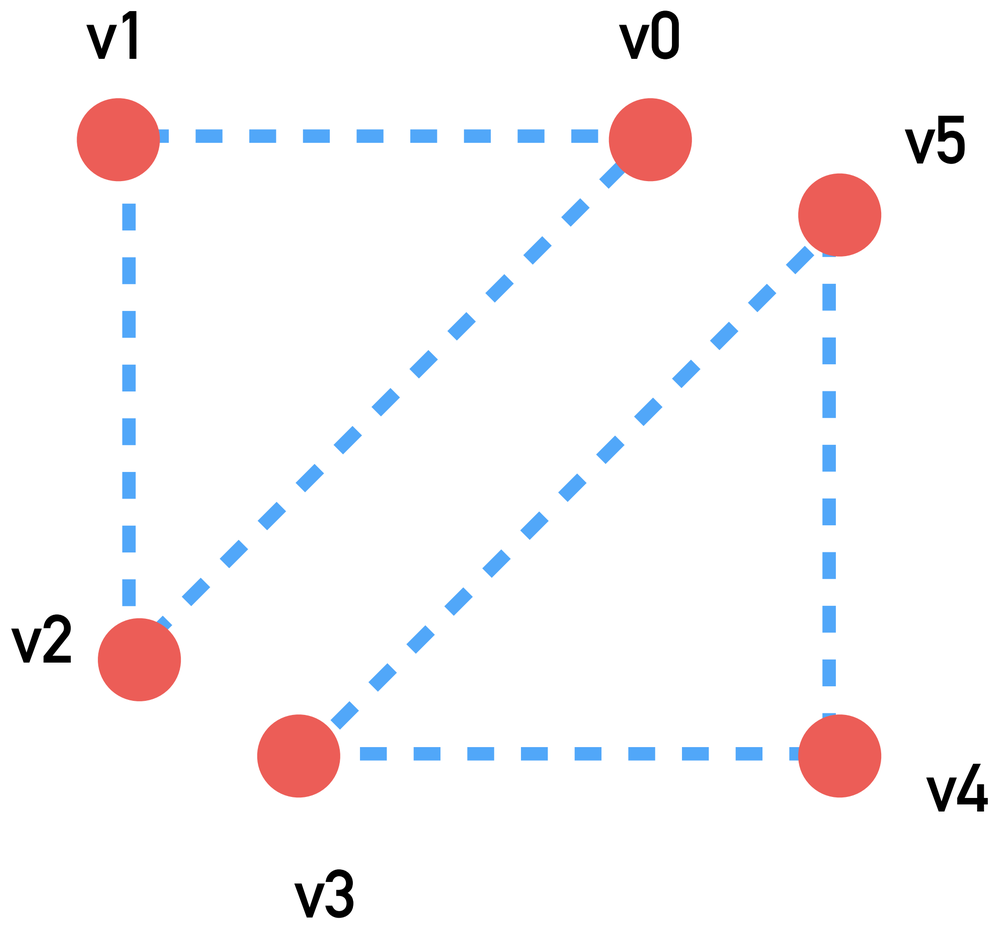
C: (−1, 3)

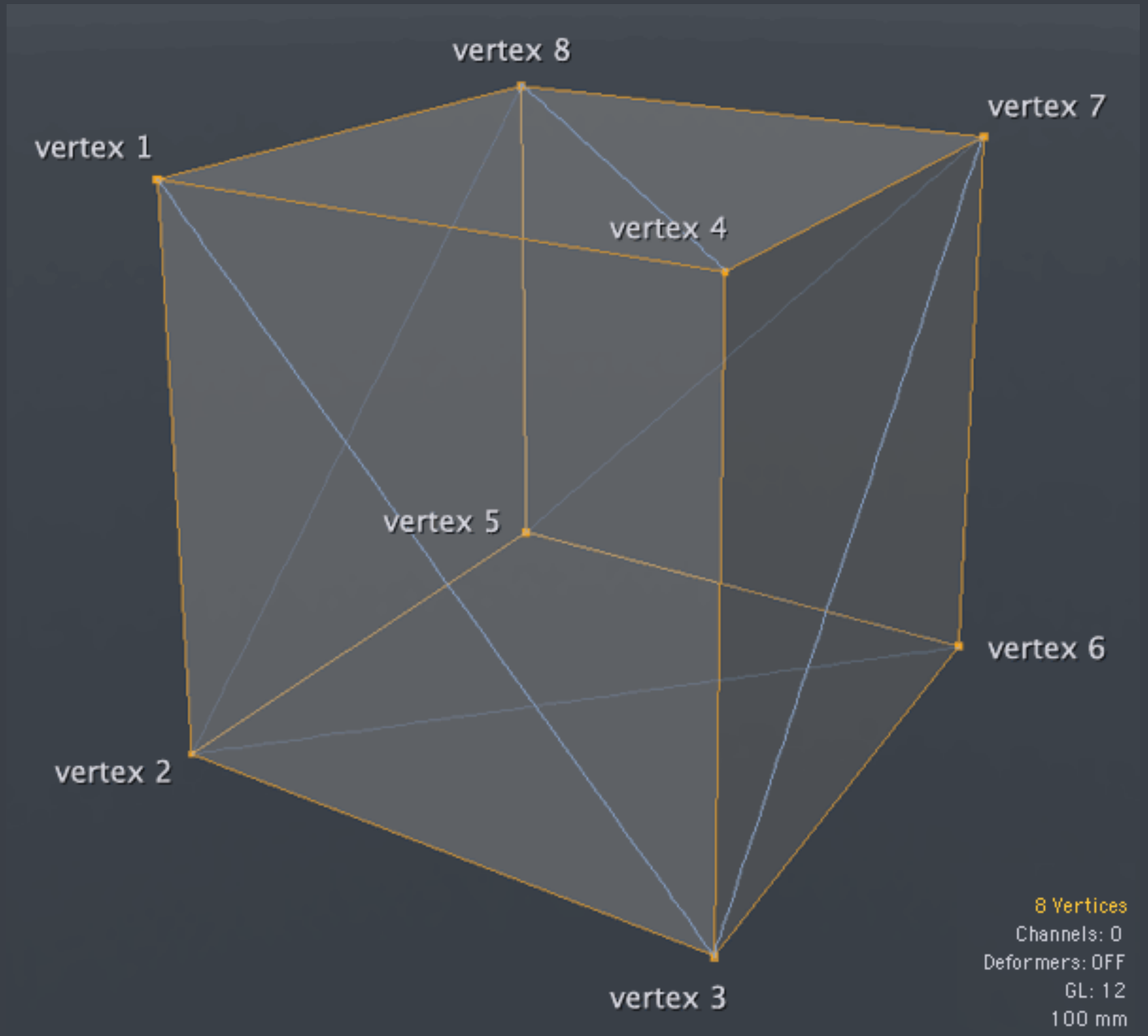
Polygons are **one-sided** and the side is defined by the **order of the vertices**



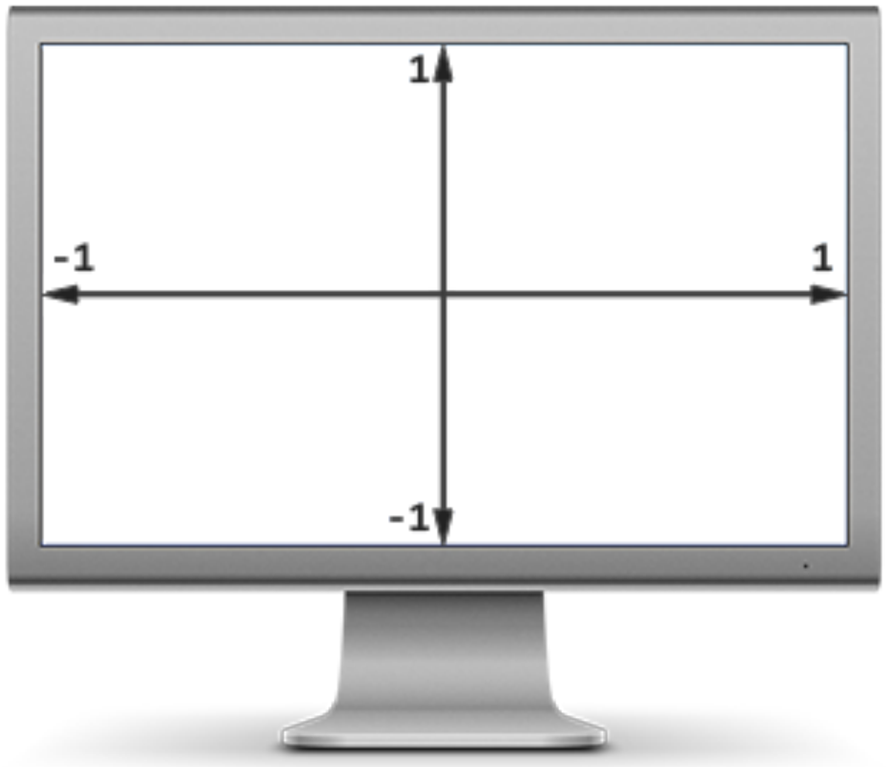
Counter-clockwise order!


v1
v2
v0

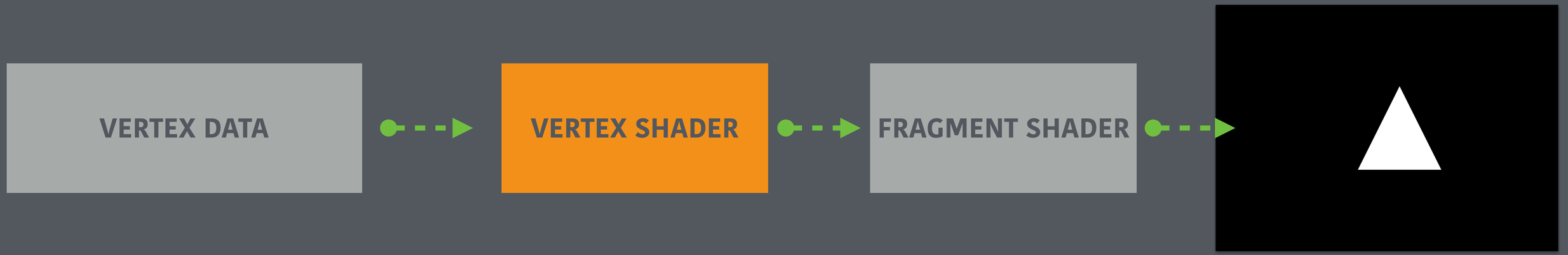

v1
v0
v5
v2
v4
v3


vertex 8
vertex 7
vertex 1
vertex 4
vertex 5
vertex 6
vertex 2
vertex 3
8 Vertices
Channels: 0
Deformers: OFF
GL: 12
100 mm

Normalized Device Coordinates



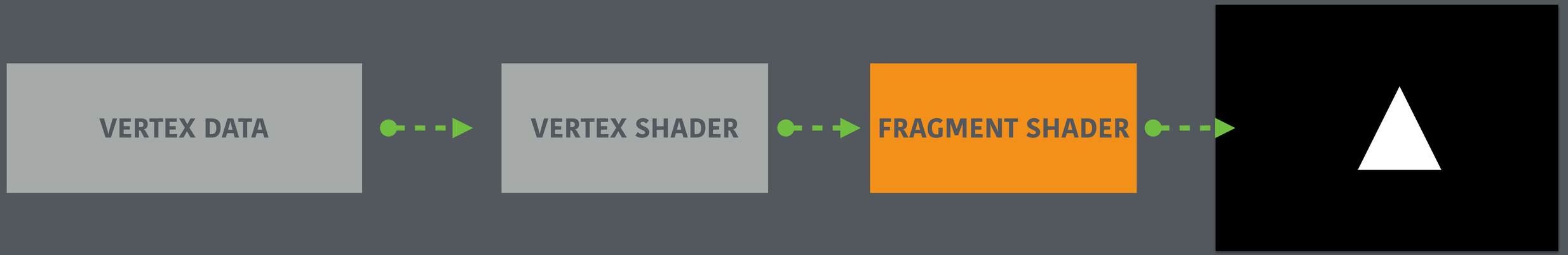
The **GPU** pipeline



The vertex shader

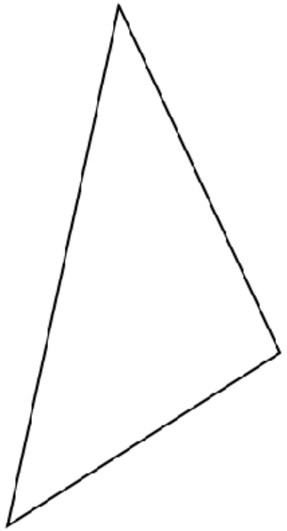
A program that transforms the attributes
(such as position, color or others)
of every vertex passed to the GPU.

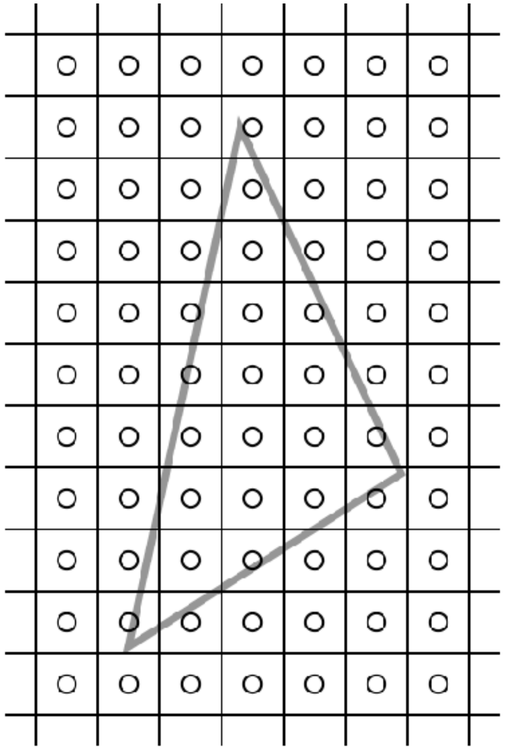
The **GPU** pipeline

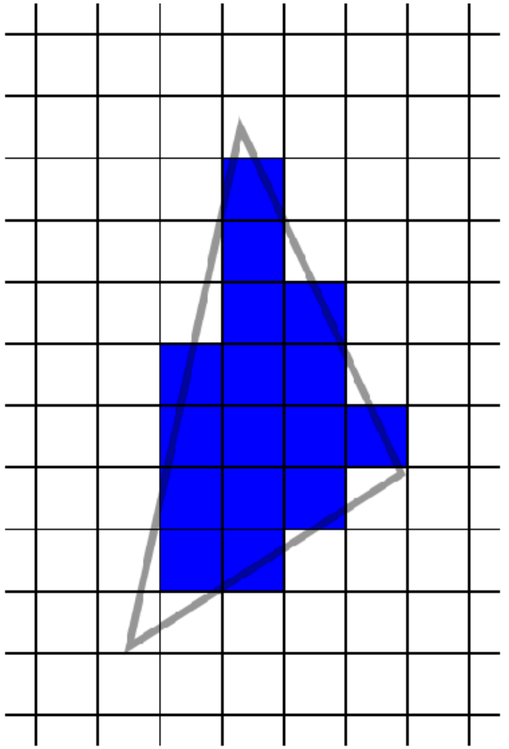


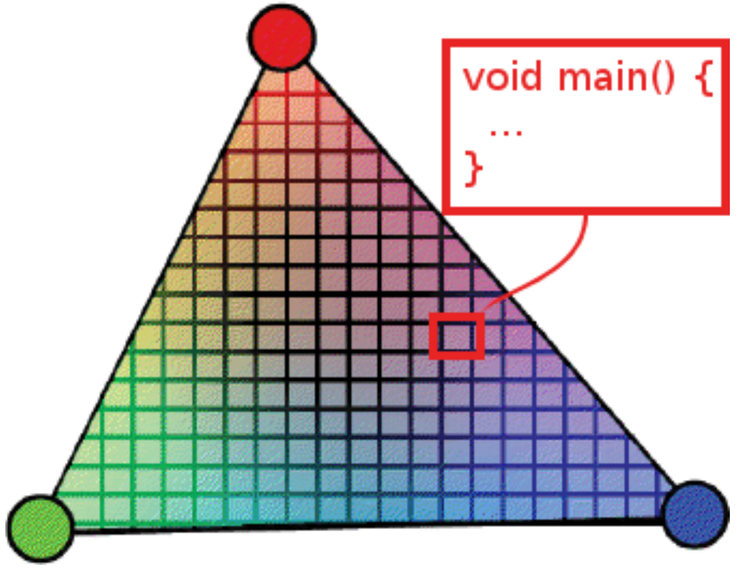
The fragment shader

A **program** that returns **the color of each pixel** when geometry is rasterized on the **GPU**.

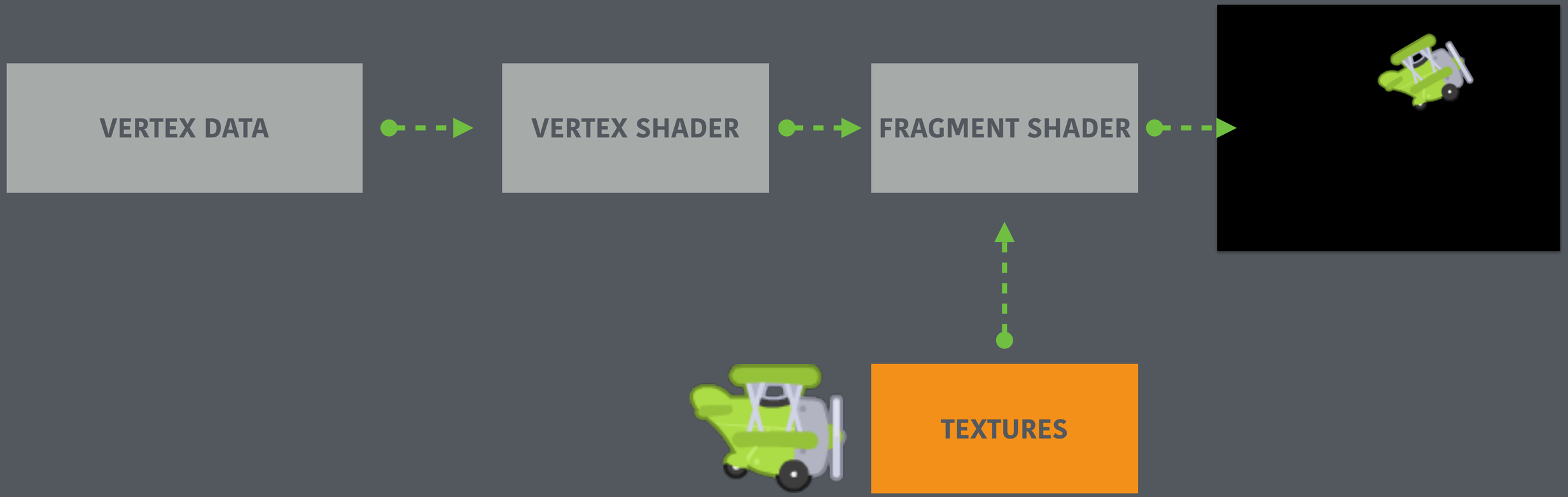




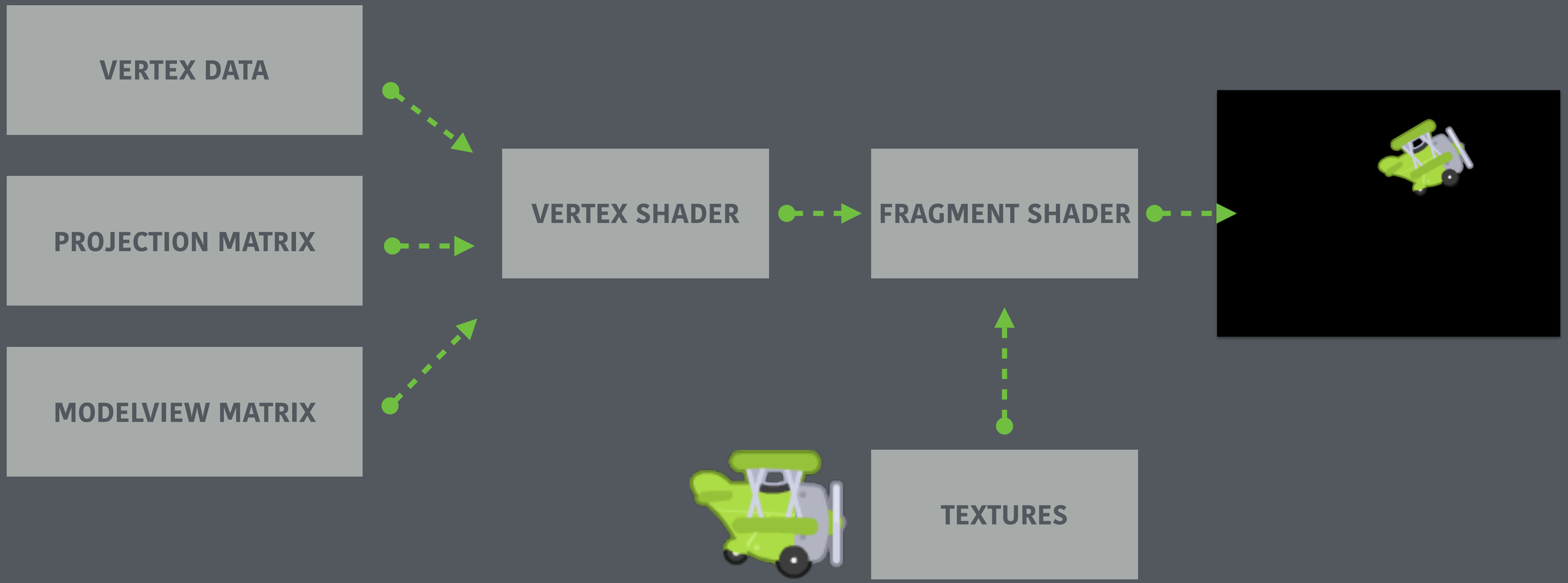



void main() {
...
}

The **GPU** pipeline



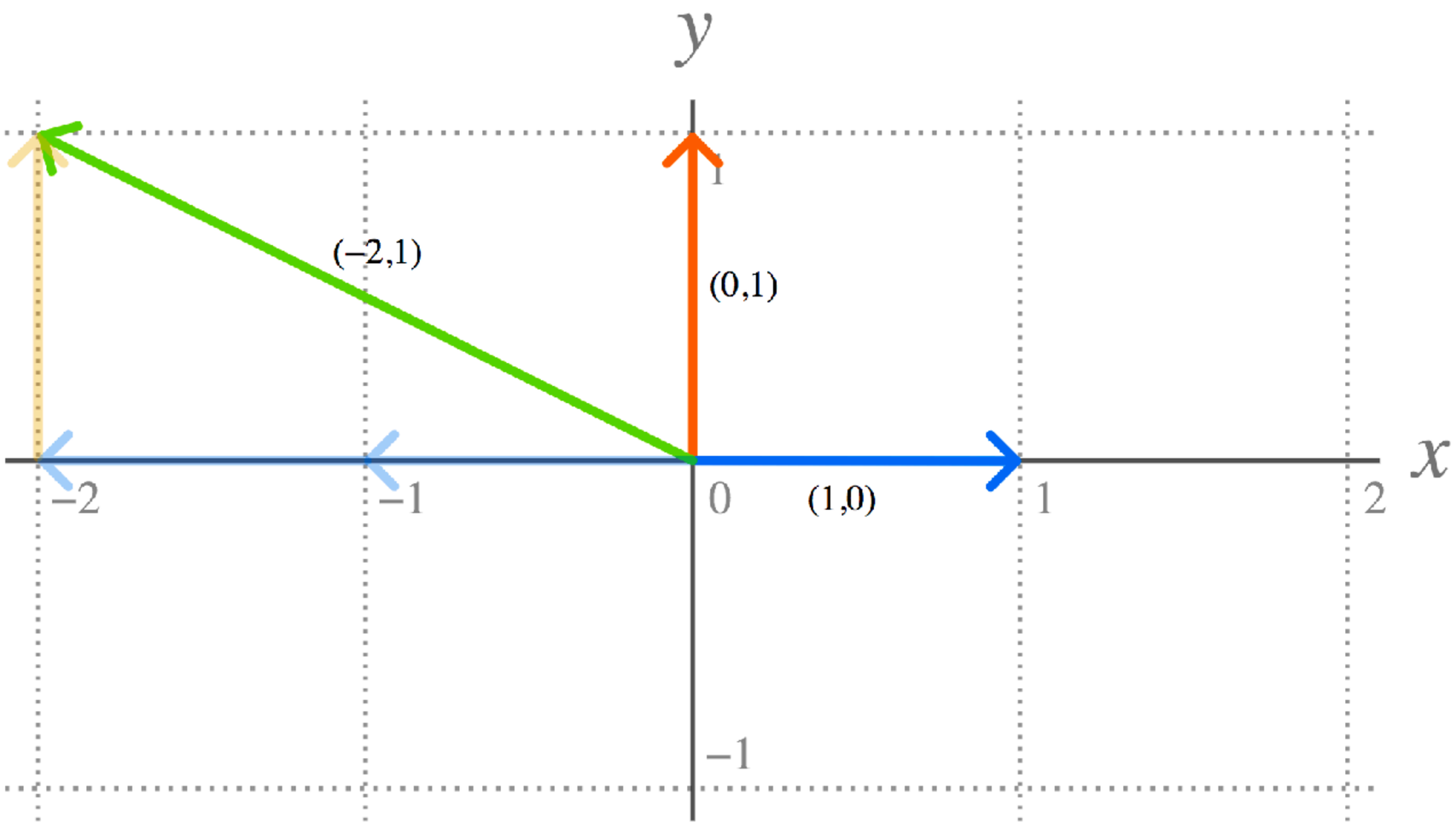
The **GPU** pipeline

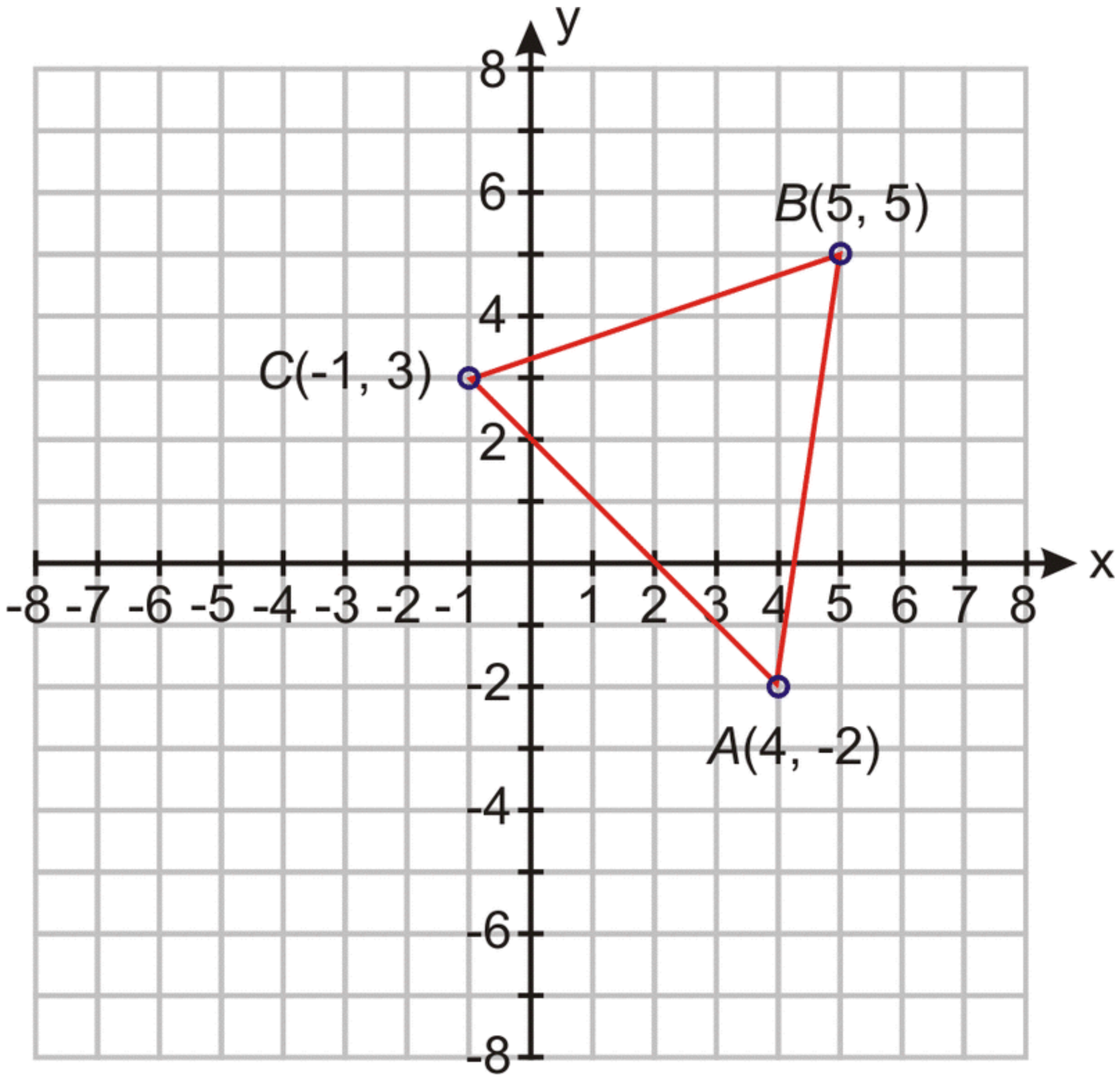


Transformation matrices

Vectors.

Basis vectors.



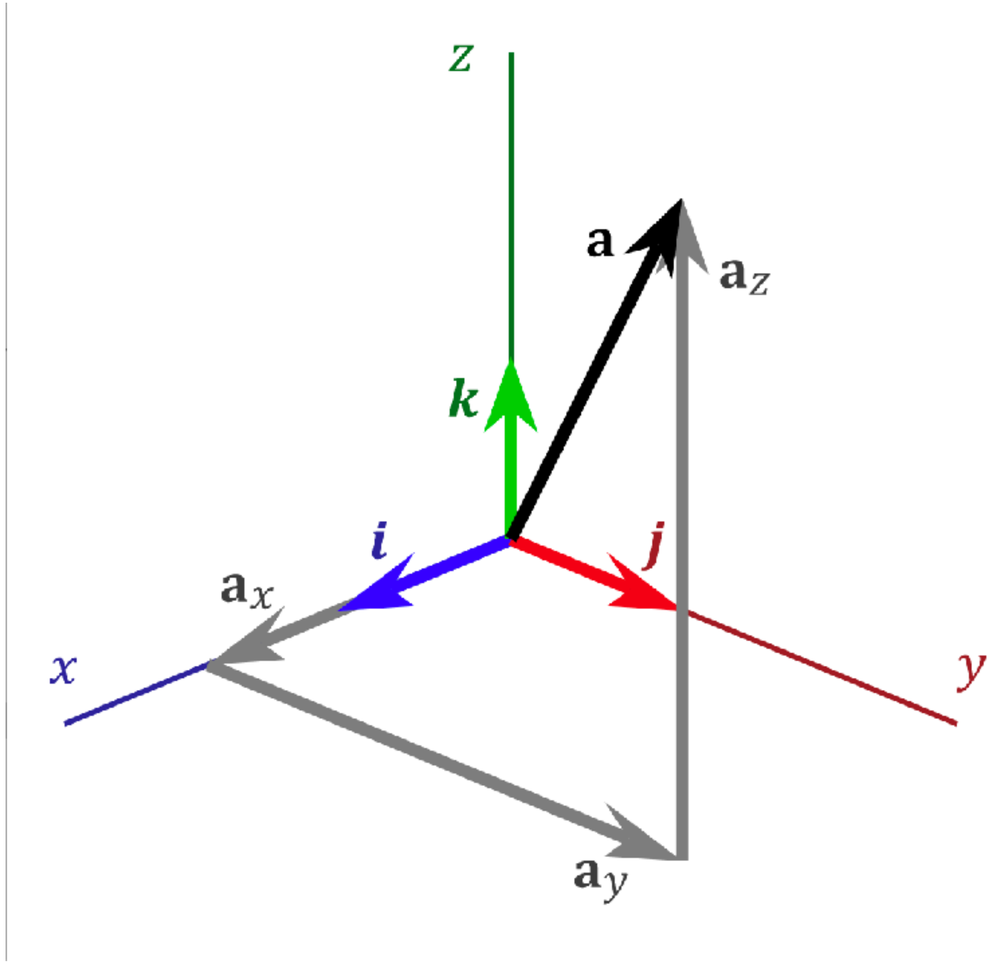


A: (4,−2)

B: (5,5)

C: (−1, 3)

Basis vectors.



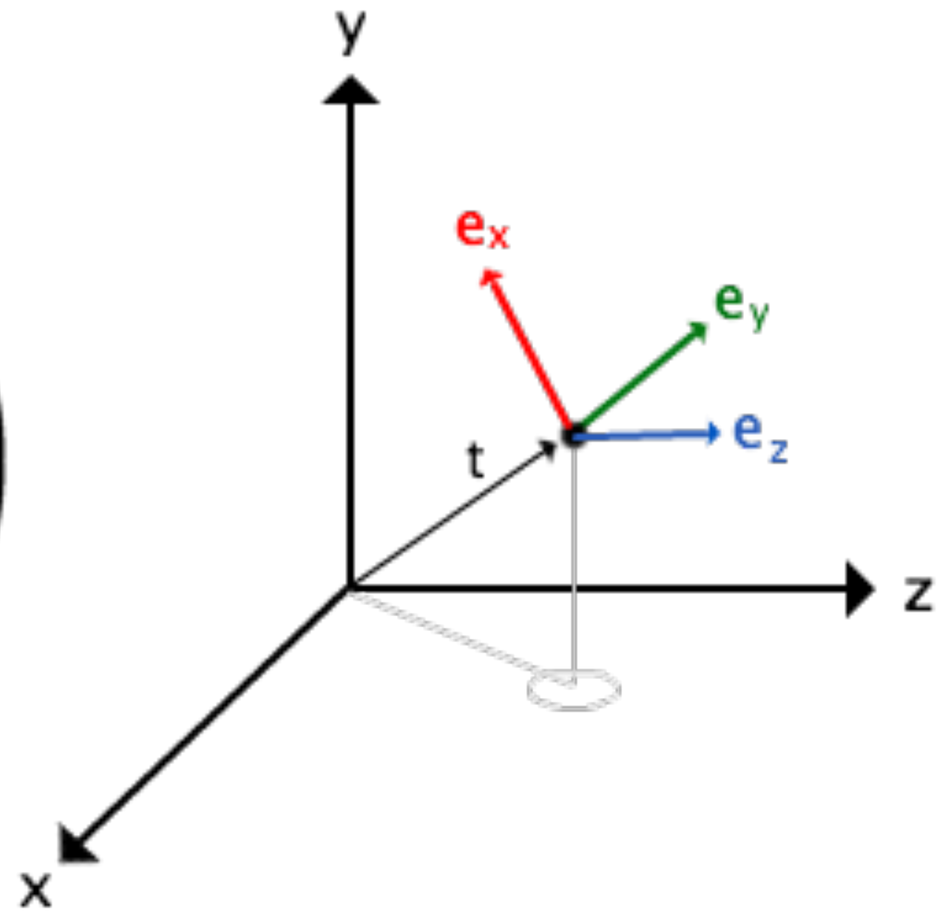
$$\vec{v} = \sum_i c_i \vec{b}_i .$$

$$\vec{v} = \sum_i c_i \vec{b}_i = \begin{bmatrix} \vec{b}_1 & \vec{b}_2 & \vec{b}_3 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} .$$

Homogeneous coordinates.

Transformation matrix.

$$\mathcal{T} = \begin{pmatrix} e_x & e_y & e_z & t \\ & & & \\ & & & \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



ABCD is the linear part of the affine transformation matrix.

$$\begin{bmatrix} A & B & ? \\ C & D & ? \\ ? & ? & ? \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

3D identity matrix and 3D position in homogeneous coordinates.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Transformations.

Identity

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} 1X + 0Y + 1x0 \\ 0X + 1Y + 1x0 \\ 0X + 0Y + 1x1 \end{bmatrix}$$

Translation

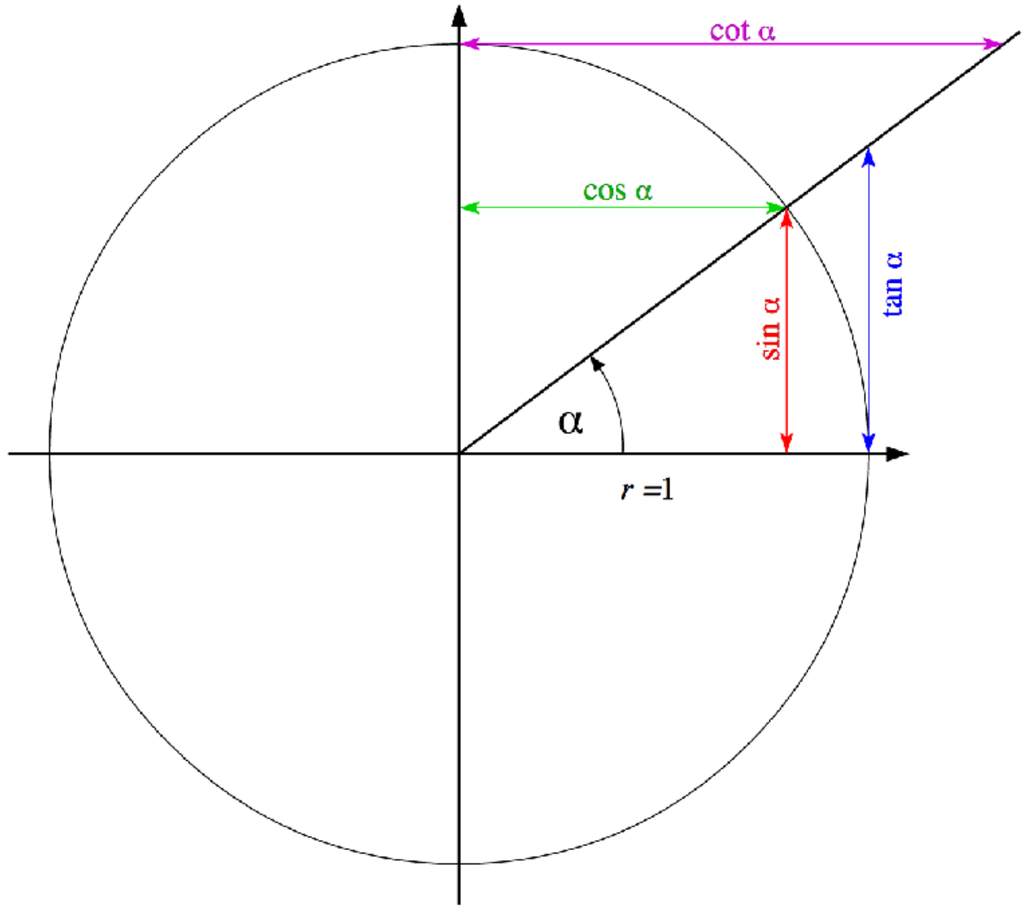
Translate

$$\begin{bmatrix} 1 & 0 & Tx \\ 0 & 1 & Ty \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} 1X + 0Y + 1Tx \\ 0X + 1Y + 1Ty \\ 0X + 0Y + 1x1 \end{bmatrix}$$

Rotation

Rotation

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} \cos\theta X + -\sin\theta Y \\ \sin\theta X + \cos\theta Y \end{bmatrix}$$



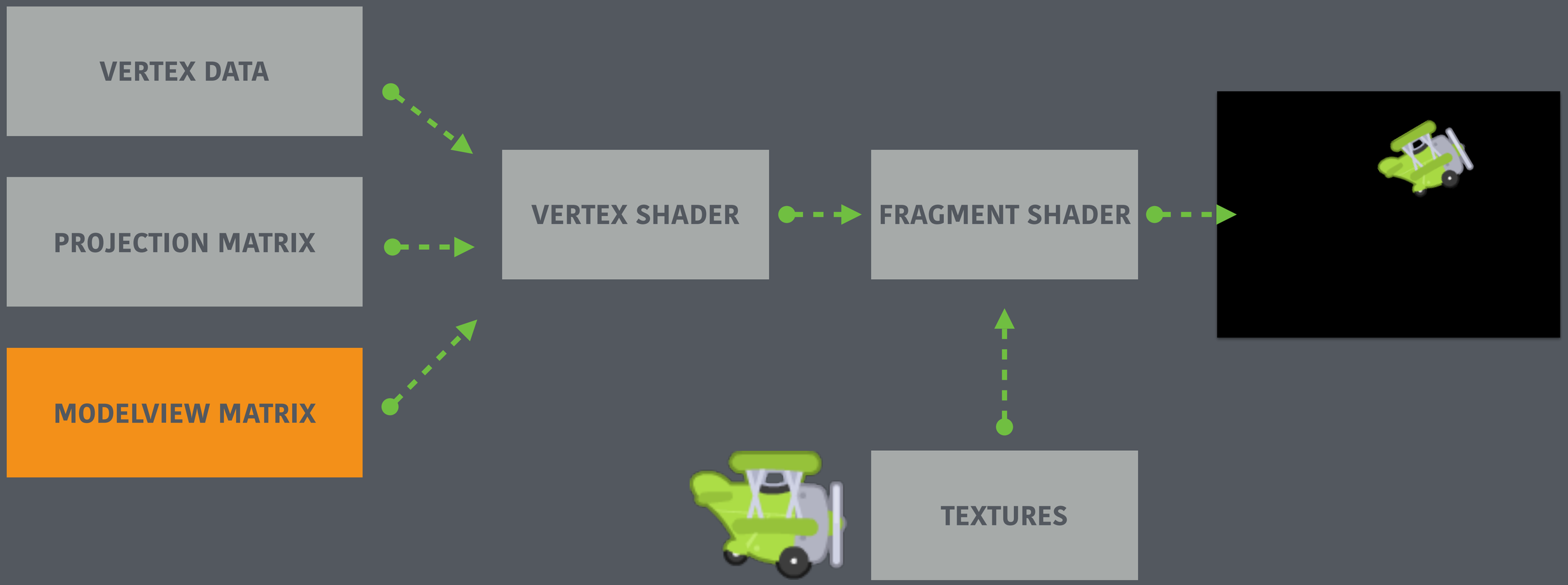
$$\begin{bmatrix} \cos\theta X + -\sin\theta Y \\ \sin\theta X + \cos\theta Y \end{bmatrix}$$

3D identity matrix and 3D position in homogeneous coordinates.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Transformation matrices

The **GPU** pipeline



All transformations in 3D

X-Rotation in 3D

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Z-Rotation in 3D

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Scale in 3D

$$\begin{bmatrix} Sx & 0 & 0 & 0 \\ 0 & Sy & 0 & 0 \\ 0 & 0 & Sz & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

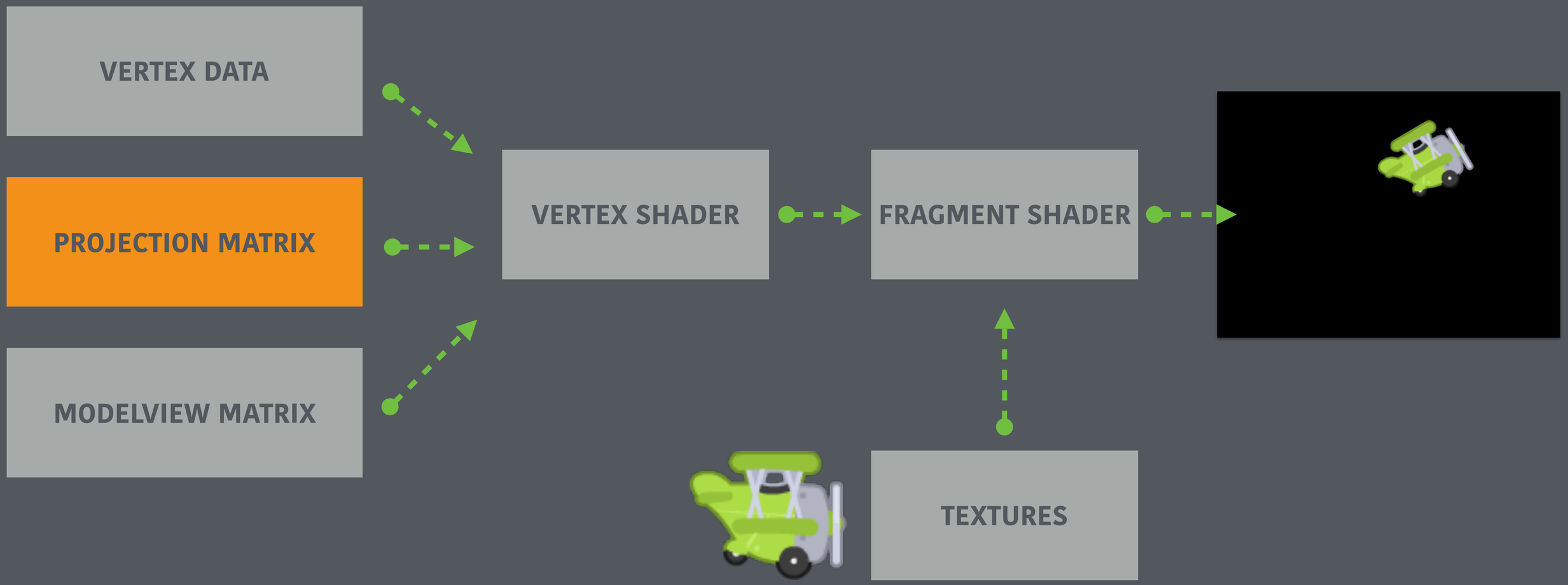
Y-Rotation in 3D

$$\begin{bmatrix} \cos\phi & 0 & \sin\phi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\phi & 0 & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

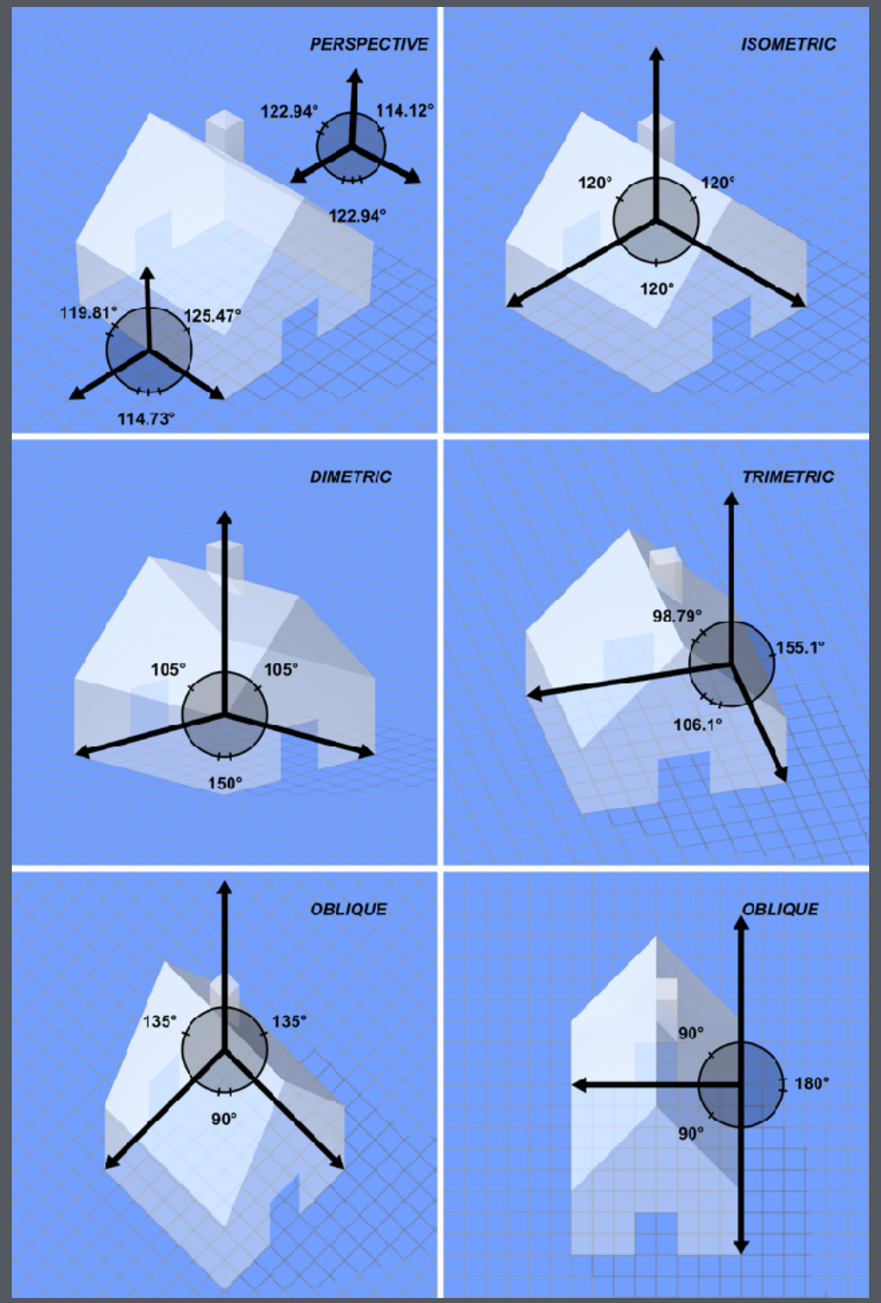
Translation in 3D

$$\begin{bmatrix} 1 & 0 & 0 & Tx \\ 0 & 1 & 0 & Ty \\ 0 & 0 & 1 & Tz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

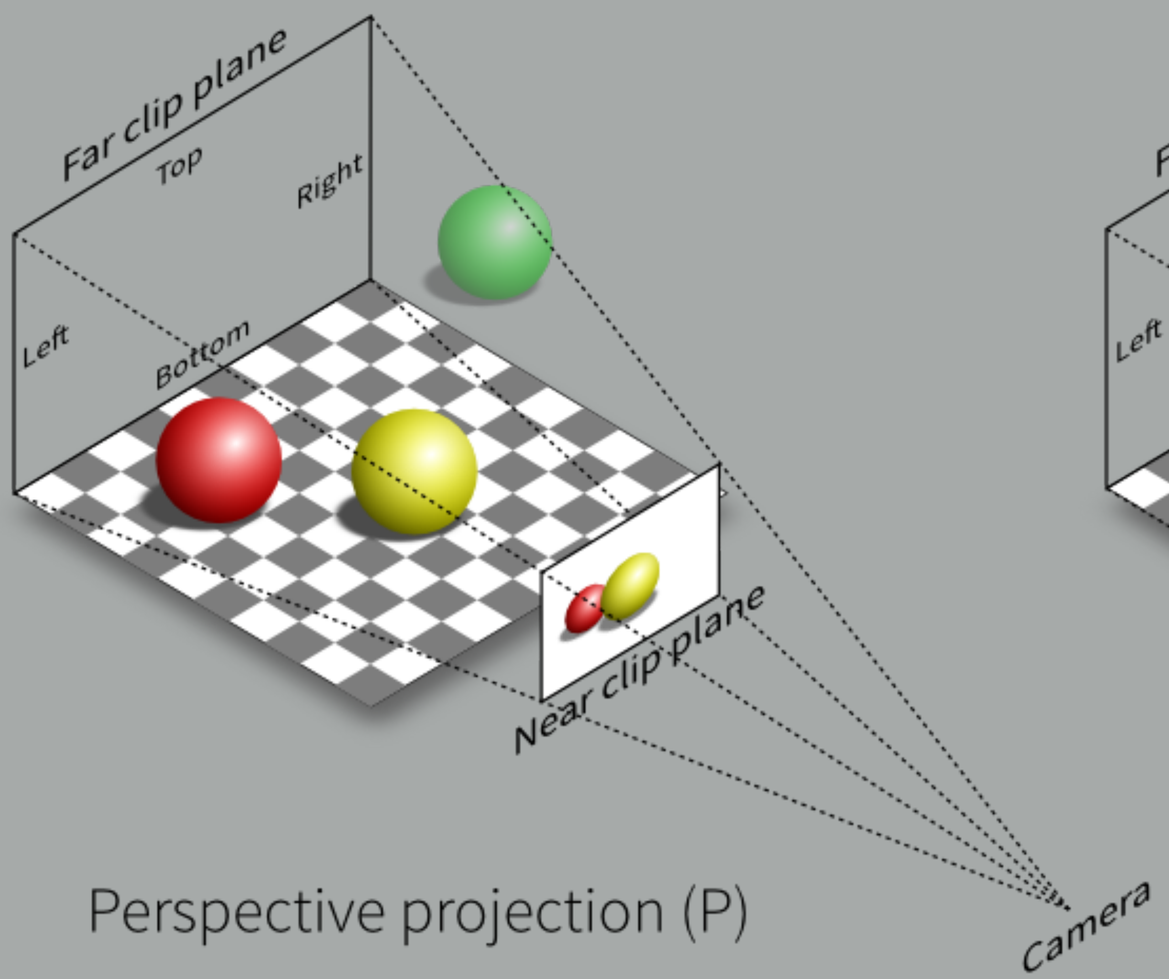
The **GPU** pipeline



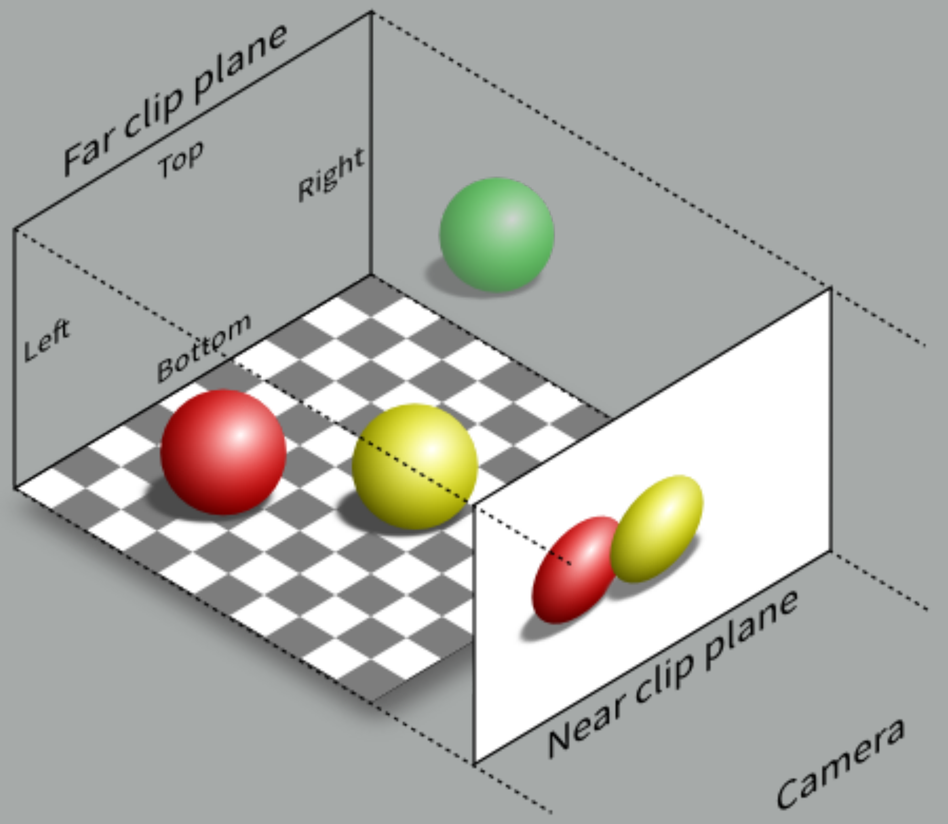
A means of representing a **three-dimensional** object in **two-dimensions**.



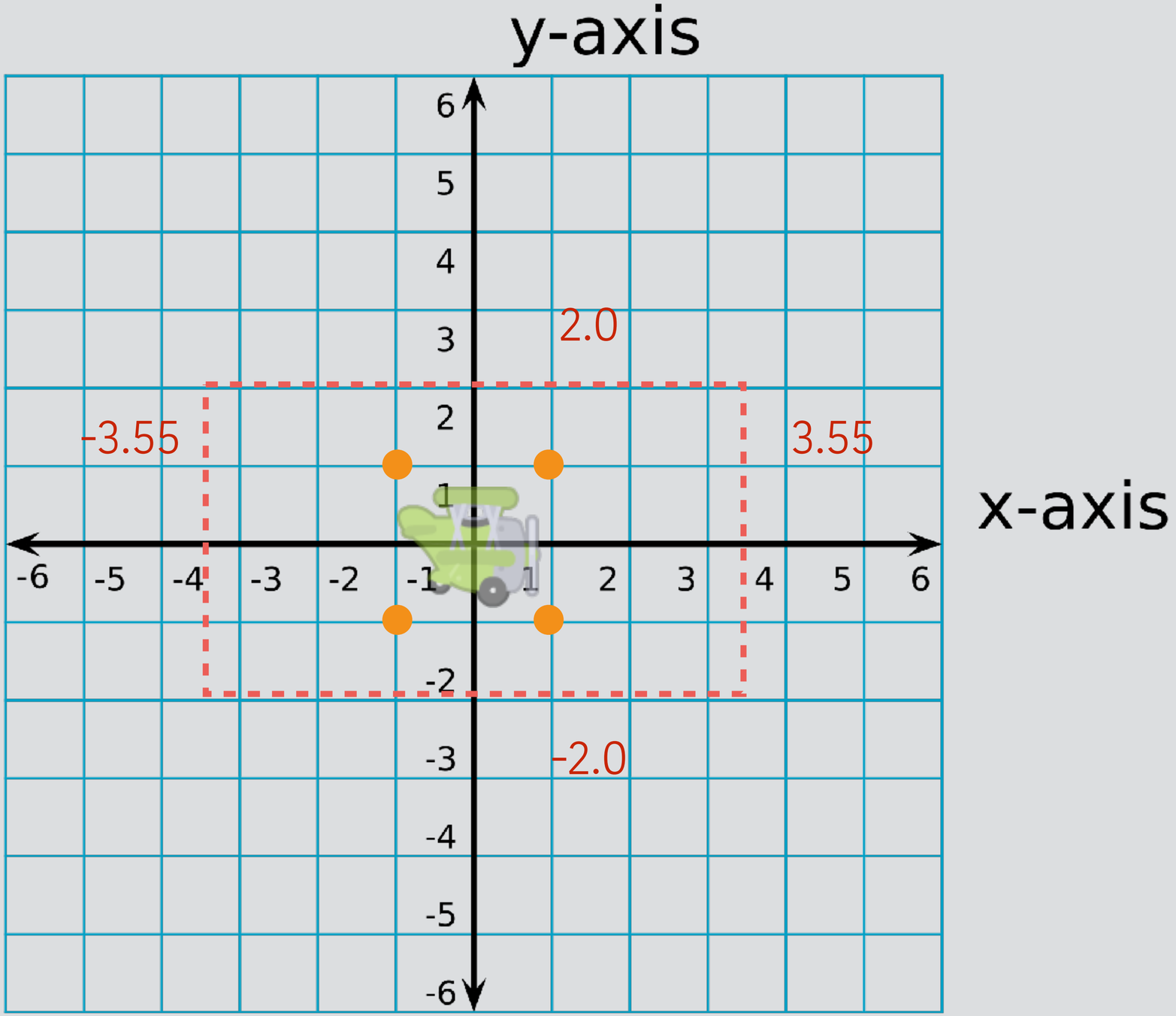
Perspective vs. **Orthographic** projection

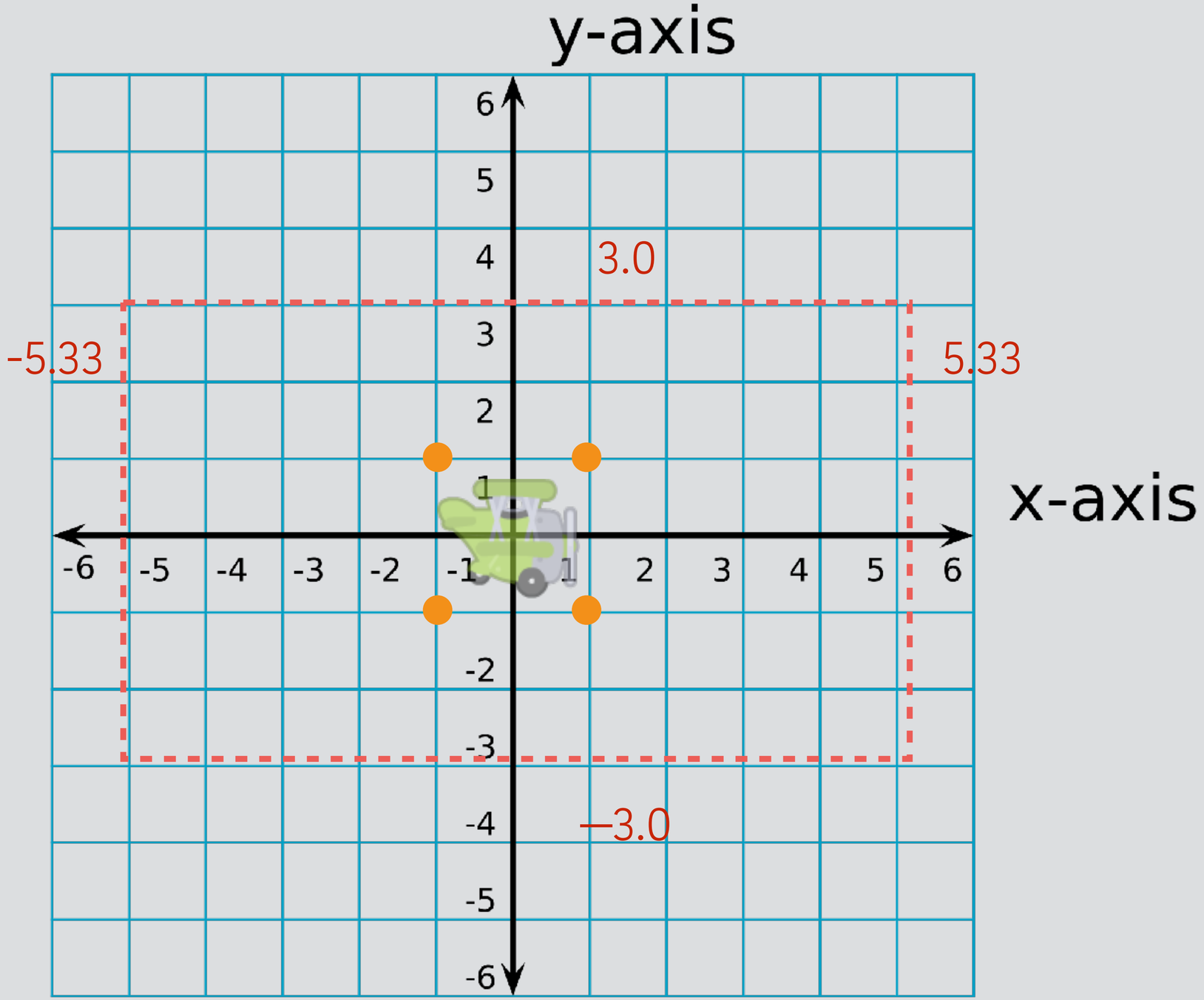


Perspective projection (P)



Orthographic projection (O)

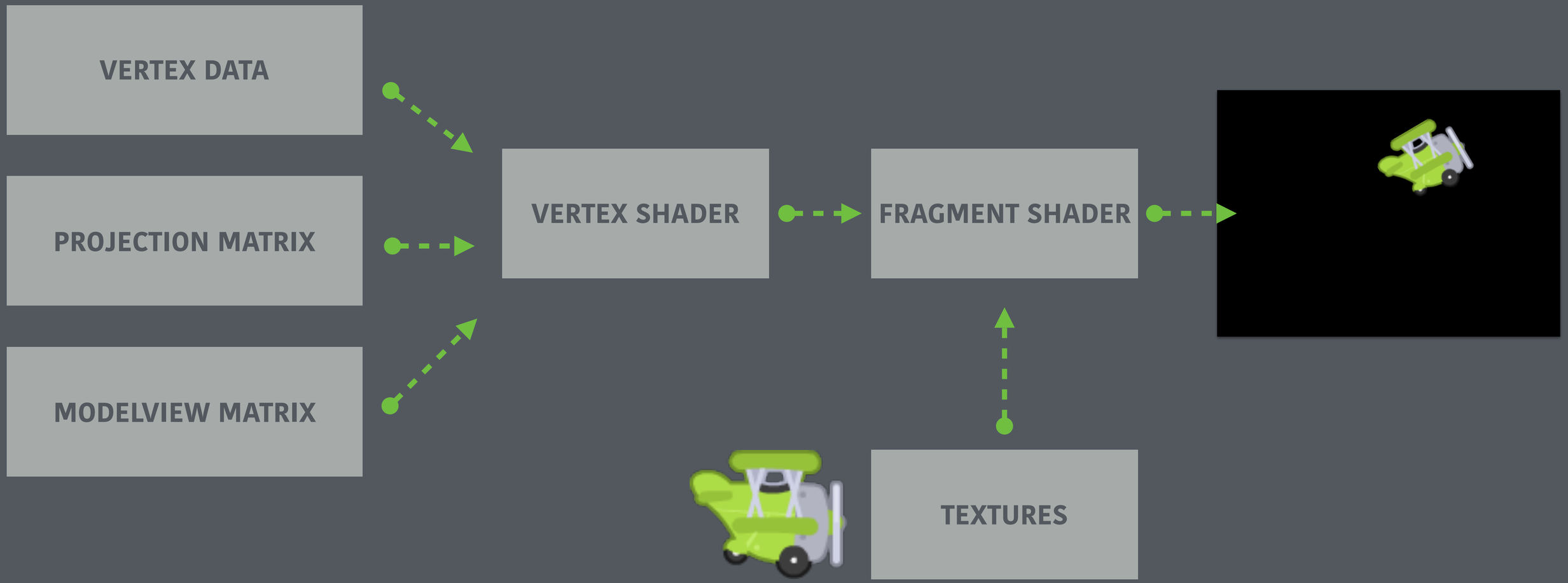

y-axis
x-axis
2.0
-2.0
-3.55
3.55
6
5
4
3
2
1
-1
-2
-3
-4
-5
-6
-6
-5
-4
-3
-2
-1
1
2
3
4
5
6


y-axis
x-axis
3.0
-3.0
-5.33
5.33
6
5
4
3
2
1
-1
-2
-3
-4
-5
-6
-6
-5
-4
-3
-2
-1
1
2
3
4
5
6

Orthographic projection transformation marix.

$$\begin{bmatrix} \frac{2}{\text{right}-\text{left}} & 0 & 0 & -\frac{\text{right}+\text{left}}{\text{right}-\text{left}} \\ 0 & \frac{2}{\text{top}-\text{bottom}} & 0 & -\frac{\text{top}+\text{bottom}}{\text{top}-\text{bottom}} \\ 0 & 0 & \frac{-2}{\text{far}-\text{near}} & -\frac{\text{far}+\text{near}}{\text{far}-\text{near}} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The **GPU** pipeline



Recap of what we need to render.

1. Vertex coordinate data that defines our triangles.

2. Modelview matrix that defines the transformation of the triangles relative to origin.

3. Projection matrix that defines how the triangles will be projected into the Normalized Device Coordinates.

4. Appropriate vertex and fragment shaders.