University of Toronto
CSC418 Computer Graphics

CSC418 A2 PartA

XIARUI HUANG

1001117695

1. Transformation

a. A 2D affine transformation which can be expressed as below
$M = \begin{bmatrix} A & b \\ 0 & 1 \end{bmatrix}$ $where\ A\ is\ a\ 2X2\ matrix\ and\ b\ is\ vector$.

According to the information provided in the question which specified by its effect on three noncolinear points, we assume there exists three points $p_1(x_1, y_1),\ p_2(x_2, y_2)\ and\ p_3(x_3, y_3)$. Meanwhile, it exists three points $p_1{}'(x_1{}', y_1{}'),\ p_2{}'(x_2{}', y_2{}')$ and $p_3{}'(x_3{}', y_3{}')$ which are the points transformed by M and correspond with p_1,p_2, p_3.

Now we convert M to the following form $M = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix}$

By establishing the transformation for p_1,p_2, p_3, we can get the following equations.

$$p_1 \to p_1' = \begin{bmatrix} x_1{}' \\ y_1{}' \\ 1 \end{bmatrix} = M \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$
$$= \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$
$$= \begin{bmatrix} ax_1 + by_1 + c \\ dx_1 + ey_1 + f \\ 1 \end{bmatrix}$$

Therefore, it is easily to get the relationship between the transformed point and the original point as below:
1) $x_1' = ax_1 + by_1 + c$
2) $y_1' = dx_1 + ey_1 + f$

Like the above process, we can derive the equation for $p_2 \to p_2'$ and $p_3 \to p_3'$ as follow:

3) $x'_2 = ax_2 + by_2 + c$
4) $y'_2 = dx_2 + ey_2 + f$
5) $x'_3 = ax_3 + by_3 + c$
6) $y'_3 = dx_3 + ey_3 + f$

To be fully mapped, the six parameters in M should be calculated from the above six relations. Meanwhile, according to noncolinear points property of 2D Affine Matrix we mentioned at the beginning, the condition is that taking six points (3 pairs of original point and the transformed point) and ensure the six relationships are not identical.

b. 2D Homography Matrix can be expressed as below:

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix}$$

To determine 8 unknown parameters for H, it need 8 points which means it need to have 4 pairs (the original point and the transformed point).

2D Similarity Matrix can be treated as the composition of translation, uniform scale and rotation.

$$M = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} S & 0 & 0 \\ 0 & S & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \mathrm{S}\cos(\theta) & -\mathrm{S}\sin(\theta) & t_x \\ \mathrm{S}\sin(\theta) & \mathrm{S}\cos(\theta) & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \mathrm{a} & \mathrm{b} & c \\ -\mathrm{b} & \mathrm{a} & d \\ 0 & 0 & 1 \end{bmatrix}$$

As you can see from the above matrix, it need at least 4 points (2 pairs) to determine the four unknown parameters a, b, c, d.

c. Are the centroid (average of the three points) and ortho-center (intersection point of the altitudes) of a triangle affine invariant?

we assume there exists six points $p_1(x_1, y_1)$, $p_2(x_2, y_2), p_3(x_3, y_3)$, $p_1'(x_1', y_1')$, $p_2'(x_2', y_2')$ and $p_3'(x_3', y_3')$, which are the same points we defined in Q1a).

1)First, we can easily get the centroid of the three points as

$$p_c = \left(\frac{x_1+x_2+x_3}{3},\ \frac{y_1+y_2+y_3}{3}\right)$$

If p_c is a triangle affine invariant, which means p_c' is also the centroid of the transformed one, which we can get the equation from (1) - (6) relations in Q1 a). Then p_c' can be expressed as follow.

$$p_c' = (\frac{x_1' + x_2' + x_3'}{3}, \qquad \frac{y_1' + y_2' + y_3'}{3})$$

$$= \left(\begin{array}{c} \frac{ax_1+by_1+c+ax_2+by_2+c+ax_3+by_3+c}{3}, \\ \frac{dx_1+ey_1+f+dx_2+ey_2+f+dx_3+ey_3+f}{3} \end{array}\right)$$

$$= \left(\frac{a(x_1+x_2+x_3)+b(y_1+y_2+y_3)}{3} + c,\ \frac{d(x_1+x_2+x_3)+e(y_1+y_2+y_3)}{3} + f\right) \qquad (1)$$

In order to test whether the centroid is a variant of triangle affine invariant, we multiply with p_c to see whether the output p_{test} is equal to p_c'.

$$p_{test} = \begin{bmatrix} x_{test} \\ y_{test} \\ 1 \end{bmatrix} = M \begin{bmatrix} \frac{x_1 + x_2 + x_3}{3} \\ \frac{y_1 + y_2 + y_3}{3} \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{x_1+x_2+x_3}{3} \\ \frac{y_1+y_2+y_3}{3} \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} a\frac{x_1+x_2+x_3}{3} + b\frac{y_1+y_2+y_3}{3} + c \\ d\frac{x_1+x_2+x_3}{3} + e\frac{y_1+y_2+y_3}{3} + f \\ 1 \end{bmatrix} \quad (2)$$

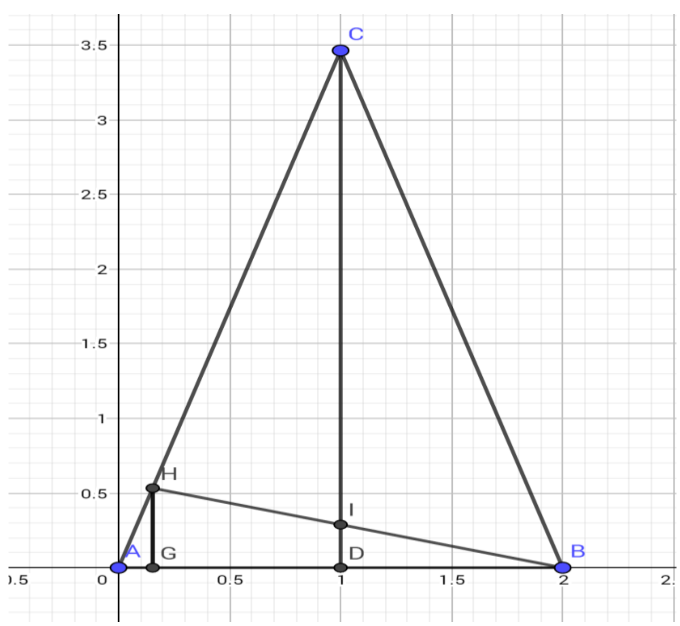
By comparing (1) and (2), we can find that $p'_c = p_{test}$, which means centroid is a triangle affine invariant.

2)

ortho-center is not a triangle affine invariant after transformation. Here is a counterexample which can prove the statement. We assume there are three points $a_1(0,0), a_2\left(1,\sqrt{3}\right), a_3\ (2,0)$, which is simple equilateral triangle, and its ortho-center can be easily found at $p_{oc}\left(1,\frac{\sqrt{3}}{3}\right)$ by triangle geometry.

Assume $M\ = \begin{bmatrix} \mathrm{a} & \mathrm{b} & c \\ d & \mathrm{e} & f \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and we can get the transformed points of a_1, a_2, a_3 by using the equations in Q1 a).

$$a'_1 = (0,0)\ \ a'_2 = \left(1,2\sqrt{3}\right)\ \ a'_3 = (2,0)\ \ p'_{oc} = (1,\frac{2\sqrt{3}}{3})$$



The above graph is the transformed triangle, where I is the ortho-center of this triangle.

By calculating the position of point I, we apply triangle geometry in this graph to find the angle ∠HAG and by using this angle to calculate the length of HG. Since $\Delta HGB \ and \ IDB \ are \ similar \ triangle,$ therefore it is easy to calculate the length of ID.

By the calculation, we can get ID=0.24425.

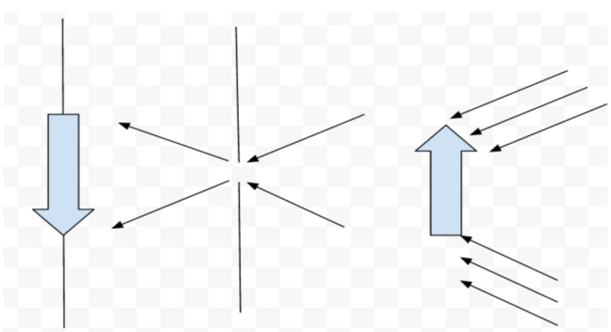
Therefore, we can get the orthocenter of this triangle is (1,0.24425), which is not equal to $p'_{oc} = (1, \frac{2\sqrt{3}}{3})$.

Therefore, orthocenter is not a triangle affine invariant.

2. Viewing and Projection

a) Why is the image formed in a pin hole camera inverted? (no more than a few sentences)

As you can see from the following figure that light contains a property that it travels in straight line at the same medium. Therefore, the light pass through the pin-hole still travels in a straight line and form its inverted image on the screen.



b) Given a 3D camera position c, a point along the viewing direction at the centre of the screen p, and a vector parallel to the vertical axis of the screen $\vec{u}$, compute the world to camera transformation matrix.

First, we define the 3D camera position c at $P_c(x,y,z)$, our reference point P_{ref} at $(0,0,0)$ as the world origin, vector $\vec{u}$ is same as $V_{up}(0,1,0)$, a vector w is at the opposite of the viewing direction, and a vector $\vec{\imath}$ is the cross product of vector $\vec{u}$ and vector $\vec{w}$. Meanwhile, a vector $\vec{q}$ is defined as the cross product of vector $\vec{\imath}$ and vector $\vec{w}$.

Therefore, we can define the camera to world transformation matrix as follow:

$$M = [\vec{\imath} \quad \vec{q} \quad \vec{w} \quad P_c] \tag{1}$$

Now, the following steps is going to find the value of vectors for $\vec{w}, \vec{\imath}, \vec{q}$:

Vector $\vec{w}$:

$\vec{w} = \frac{P_c - P_{ref}}{|P_c - P_{ref}|}$ (where denominator is the amplitude of the vector)

$$= \frac{(x,y,z)-(0,0,0)}{\sqrt{x^2+y^2+z^2}}$$

$$= \left(\frac{x}{\sqrt{x^2+y^2+z^2}}, \frac{y}{\sqrt{x^2+y^2+z^2}}, \frac{z}{\sqrt{x^2+y^2+z^2}}\right)$$

Vector $\vec{\imath}$:

$$\vec{\imath} = \frac{V_{up} \times \vec{w}}{|V_{up} \times \vec{w}|}$$

$$V_{up} \times \vec{w} = \begin{matrix} i & j & k \\ 0 & 1 & 0 \\ w_x & w_y & w_z \end{matrix} = w_z\,\vec{\imath} + 0\vec{\jmath} - w_x\,\vec{k} = (w_z, 0, -w_x)$$

$$|V_{up} \times \vec{w}| = \sqrt{w_z^2 + w_x^2} = \sqrt{\frac{x^2+z^2}{x^2+y^2+z^2}}$$

$$\vec{\imath} = \frac{V_{up} \times \vec{w}}{|V_{up} \times \vec{w}|} = \left(\frac{w_z}{|\vec{w} \times V_{up}|}, 0, \frac{-w_x}{|\vec{w} \times V_{up}|}\right) = \left(\frac{z}{\sqrt{x^2+z^2}}, 0, -\frac{x}{\sqrt{x^2+z^2}}\right)$$

Vector $\vec{q}$:

$$\vec{q} = \vec{w} \times \vec{\imath}$$

$$= \begin{matrix} i & j & k \\ w_x & w_y & w_z \\ i_x & 0 & i_z \end{matrix} = w_y i_z \vec{i} + (w_z i_x - w_x i_z)\vec{j} - w_y i_x \vec{k}$$

$$= (w_y i_z, w_z i_z - w_x i_z, -w_y i_x)$$

$$= \left(\frac{-xy}{(\sqrt{x^2+y^2+z^2})(\sqrt{x^2+z^2})}, \frac{x^2+z^2}{(\sqrt{x^2+y^2+z^2})(\sqrt{x^2+z^2})}, \frac{-yz}{(\sqrt{x^2+y^2+z^2})(\sqrt{x^2+z^2})}\right)$$

Then the camera to world matrix can be substituted by the value we calculated above.

$$M_{camera\_to\_world} = [\vec{i} \quad \vec{q} \quad \vec{w} \quad P_c]$$

$$= \begin{bmatrix} i_x & q_x & w_x & x \\ i_y & q_y & w_y & y \\ i_z & q_z & w_z & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{z}{\sqrt{x^2+z^2}} & \frac{-xy}{(\sqrt{x^2+y^2+z^2})(\sqrt{x^2+z^2})} & \frac{x}{\sqrt{x^2+y^2+z^2}} & x \\ 0 & \frac{x^2+z^2}{(\sqrt{x^2+y^2+z^2})(\sqrt{x^2+z^2})} & \frac{y}{\sqrt{x^2+y^2+z^2}} & y \\ -\frac{x}{\sqrt{x^2+z^2}} & \frac{-yz}{(\sqrt{x^2+y^2+z^2})(\sqrt{x^2+z^2})} & \frac{z}{\sqrt{x^2+y^2+z^2}} & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

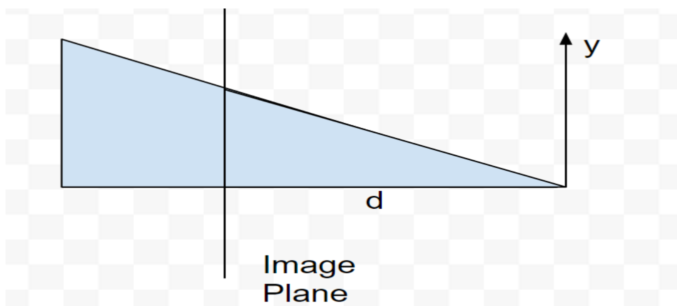
Therefore, we can easily get $M_{world\_to\_camer} = M_{camera\_to\_world}{}^T$:

$M_{world\_to\_camer} =$

$$\begin{bmatrix} \frac{z}{\sqrt{x^2 + z^2}} & \frac{-xy}{(\sqrt{x^2 + y^2 + z^2})(\sqrt{x^2 + z^2})} & \frac{x}{\sqrt{x^2 + y^2 + z^2}} & x \\ 0 & \frac{x^2 + z^2}{(\sqrt{x^2 + y^2 + z^2})(\sqrt{x^2 + z^2})} & \frac{y}{\sqrt{x^2 + y^2 + z^2}} & y \\ -\frac{x}{\sqrt{x^2 + z^2}} & \frac{-yz}{(\sqrt{x^2 + y^2 + z^2})(\sqrt{x^2 + z^2})} & \frac{z}{\sqrt{x^2 + y^2 + z^2}} & z \\ 0 & 0 & 0 & 1 \end{bmatrix}^T$$

$$= \begin{bmatrix} \frac{z}{\sqrt{x^2+z^2}} & 0 & -\frac{x}{\sqrt{x^2+z^2}} & 0 \\ \frac{-xy}{(\sqrt{x^2+y^2+z^2})(\sqrt{x^2+z^2})} & \frac{x^2+z^2}{(\sqrt{x^2+y^2+z^2})(\sqrt{x^2+z^2})} & \frac{-yz}{(\sqrt{x^2+y^2+z^2})(\sqrt{x^2+z^2})} & 0 \\ \frac{x}{\sqrt{x^2+y^2+z^2}} & \frac{y}{\sqrt{x^2+y^2+z^2}} & \frac{z}{\sqrt{x^2+y^2+z^2}} & 0 \\ x & y & z & 1 \end{bmatrix}$$

3) Given a 3D camera at the origin viewing along the z-axis, and a projection plane (screen) located at distance d, what is the 2D (x,y) point that is the perspective projection of a 3D point p = (px, py, pz) ?



We assume there exist a 3D point p = (p_X, p_y, p_z) and it will project as some point called $p' = (p_X', p_y', d)$ at the image plane.

According to the property of similar triangle, we can calculate the length ratio $r = \frac{d}{p_z}$. Moreover, p_X' *can be derived as* $= rP_x = \frac{dP_x}{p_z}$ and p_y' *as* $rP_y = \frac{dP_y}{p_z}$.

Therefore, 2D $(x, y) = (P_x', P_y') = \left(\frac{dP_x}{p_z}, \frac{dP_y}{p_z}\right)$

4) Under what conditions will a family of lines parallel to the vector v = (vx, vy, vz) remain parallel after this perspective projection?

A family of lines can be regarded as lots of planes which parallel to the vector, therefore, we assume there exists a vector $\vec{w} = c\vec{v} = (cv_x, cv_y, cv_z)$ on the plane, where c is arbitrary number.

Like the above process, if $\vec{w}$ want to preserve parallel after the perspective projection, it need to keep the property of parallel $\vec{w} = c\vec{v}$.

Therefore, $\vec{w}' = c\vec{v'} = \left(\frac{dcv_x}{cv_z}, \frac{dcv_y}{cv_z}, d\right) = \left(\frac{dv_x}{v_z}, \frac{dv_y}{v_z}, d\right)$

5) When this condition is not met, do all lines converge at a single 2D point? If so, which point? If not, provide counterexample?

If d = nv_z was not meet, all lines will converge at a single 2D point.

The vector line equation related to the vector $\vec{v}$ can be expressed as $\vec{r_1} = \vec{a} + \lambda\vec{v}$, where $\vec{r}$ is a position vector, $\vec{a}$ can be the vector where some point at parallel line travelling along with. Similar, we can get another line which parallel to the vector, $\vec{r_2} = \vec{b} + \lambda\vec{v}$

Therefore, we can get the perspective projection of $\vec{r_1}$ and $\vec{r_2}$ as follow.

$$\vec{r}_1{}'(\lambda) = (\frac{d(a_x + \lambda v_x)}{a_z + \lambda v_z}, \frac{d(a_y + \lambda v_y)}{a_z + \lambda v_z})$$

$$\vec{r}_2{}'(\lambda) = (\frac{d(b_x + \lambda v_x)}{b_z + \lambda v_z}, \frac{d(b_y + \lambda v_y)}{b_z + \lambda v_z})$$

When λ is close to infinity, the effect from a and b is much less, therefore it will converge at point $p = (\frac{dv_x}{v_z}, \frac{dv_y}{v_z})$

3. Surfaces

The tangent plane of a surface at a point is defined so that it contains all tangent vectors. In this exercise, you will verify that a specific tangent vector is contained in the tangent plane. Let the surface be a torus in 3D (Figure 1) defined by the implicit equation: $f(x, y, z) = (R - \sqrt{x^2 + y^2})^2 + z^2 - r^2 = 0 \; where \; R > r$.

a) Give a surface normal at point p = (x, y, z), using the surface implicit equation.

$$Suface\ normal = \vec{N} = \nabla f(x, y, z)$$

$$= (\frac{\partial f(x,y,z)}{\partial x}, \frac{\partial f(x,y,z)}{\partial y}, \frac{\partial f(x,y,z)}{\partial z}) \qquad (1)$$

$$f_X = \frac{\partial f(x,y,z)}{\partial x} = \frac{d}{dx}(R^2 - 2R\sqrt{x^2 + y^2} + x^2 + y^2 + z^2 - r^2)$$

$$= -\frac{2Rx}{\sqrt{x^2+y^2}} + 2x$$

$$f_y = \frac{\partial f(x,y,z)}{\partial y} = \frac{d}{dy}(R^2 - 2R\sqrt{x^2 + y^2} + x^2 + y^2 + z^2 - r^2)$$

$$= -\frac{2Ry}{\sqrt{x^2+y^2}} + 2y$$

$$f_z = \frac{\partial f(x,y,z)}{\partial z} = \frac{d}{dz}\left(R^2 - 2R\sqrt{x^2 + y^2} + x^2 + y^2 + z^2 - r^2\right)$$

$$= 2z$$

Then equation (1) can be derived as follow:

$$\vec{N} = (-\frac{2Rx_0}{\sqrt{x_0^2+y_0^2}} + 2x_0\, , -\frac{2Ry_0}{\sqrt{x_0^2+y_0^2}} + 2y_0, 2z_0) \text{ at } p \; = \; (x_0, y_0, z_0)$$

b) Give an implicit equation for the tangent plane at p.

According to the fact of the gradient vector or normal vector that $\nabla f(x, y, z)$ which is orthogonal to the tangent plane at $p\ (x_0\ , y_0, z_0)$, so the tangent plane to the surface given by $f(x, y, z)\ at$ p has the equation as follow:

$$f_x(x - x_0) + f_y(y - y_0) + f_z(z - z_0) = 0 \qquad (2)$$

Where $f_X = -\frac{2Rx_0}{\sqrt{x_0^2+y_0^2}} + 2x_0$, $f_y = -\frac{2Ry_0}{\sqrt{x_0^2+y_0^2}} + 2y_0$, and$f_z \; = 2z_0$ at p.

So equation (2) can be written as:

$$(-\frac{2Rx_0}{\sqrt{x_0^2+y_0^2}}+2x_0)\,(x-x_0)-(\frac{2Ry_0}{\sqrt{x_0^2+y_0^2}}+2y_0)(y-y_0)+2z_0(z-z_0)=0$$

c)Show that the parametric curve q(λ) = (R cos λ, R sin λ, r) lies on the surface.

We can simply substitute the point into the surface function $f(x,y,z)=(R-\sqrt{x^2+y^2})^2+z^2-r^2$ to check whether the point is on the surface if the value =0.

$$f(\mathrm{R}\cos\lambda,\mathrm{R}\sin\lambda,\mathrm{r})=(R-\sqrt{(\mathrm{R}\cos\lambda)^2+(\mathrm{R}\sin\lambda)^2})^2+r^2-r^2$$
$$=0$$

Therefore, the parametric curve lies on the surface.

d) Find a tangent vector of q(λ) as a function of λ.

$\vec{\mathrm{T}}=$ Tangent vector of q(λ) $=\nabla\mathrm{q}(\lambda)$

$$=\frac{dq}{d\lambda}=\left(\frac{dq_x}{d\lambda},\frac{dq_y}{d\lambda},\frac{dq_z}{d\lambda}\right)=(-Rsin\lambda,\mathrm{R}\cos\lambda,0)$$

e) Show this tangent vector at q(λ) to lie on the implicit equation of the tangent plane.

To solve this question, what we need to do is taking the tangent vector $(-Rsin\lambda,\mathrm{R}\cos\lambda,0)$ and substituting in the implicit equation of the tangent plane at $\mathrm{q}(\lambda)=(R\ cos\ \lambda,R\ sin\ \lambda,r)$, which is equation (2) shown in question (b):

$$(-\frac{2Rx_0}{\sqrt{x_0^2+y_0^2}}+2x_0)\,(x-x_0)-(\frac{2Ry_0}{\sqrt{x_0^2+y_0^2}}+2y_0)(y-y_0)+2z_0(z-z_0)=0$$

At this question, $(x_0,\ y_0,z_0)=(R\ cos\ \lambda,R\ sin\ \lambda,r)$, $(x-x_0),(y-y_0)\ and\ (z-z_0)$ are equal to $-Rsin\lambda,\mathrm{R}\cos\lambda,0$ respectively.

Substituting the point into the above equation, which will become as follow:

$$Since\ \left(-\frac{2Rx_0}{\sqrt{x_0^2+y_0^2}}+2x_0\right)=\left(-\frac{2R^2cos\ \lambda}{\sqrt{(Rcos\ \lambda)^2+(R\ sin\ \lambda)^2}}+2R\ cos\ \lambda\right)$$

$$=-\frac{2R^2cos\ \lambda}{R}+2R\ cos\ \lambda$$

$$=0$$

Therefore

$$\left(-\frac{2Rx_0}{\sqrt{x_0^2+y_0^2}}+2x_0\right)(-Rsin\lambda)=0$$

Similarly,

$$-\left(\frac{2Ry_0}{\sqrt{x_0^2+y_0^2}}+2y_0\right)=-\left(\frac{2R^2sin\ \lambda}{\sqrt{(Rcos\ \lambda)^2+(R\ sin\ \lambda)^2}}+2\ R\ sin\ \lambda\right)$$

$$=-\frac{2R^2sin\ \lambda}{R}+2R\sin\lambda$$

$$=0$$

Thus,

$$-\left(\frac{2Ry_0}{\sqrt{x_0^2+y_0^2}}+2y_0\right)(\mathrm{Rcos}\lambda)=0$$

Finally, we can simply get the following z value:

$$2z_0(z-z_0)=0$$

In conclusion, tangent vector at q(λ) lies on the implicit equation of the tangent plane that the final answer result in 0 at the right-hand side of equation.

4. Visibility

a) Does a hierarchical axis-aligned bounding box (AABB) tree of the seven oaks depend on the location and orientation of the viewing frustum?

Yes, if the object was not in the viewing frustum, it will not be considered.

b) Partition the oaks into a balanced binary tree of AABBs. Draw each bounding box and draw a schematic of the tree using the letters on each oak. At each node of the tree, sort and split the contained oaks along the longest bounding box direction. Suppose the first box is drawn as a square:

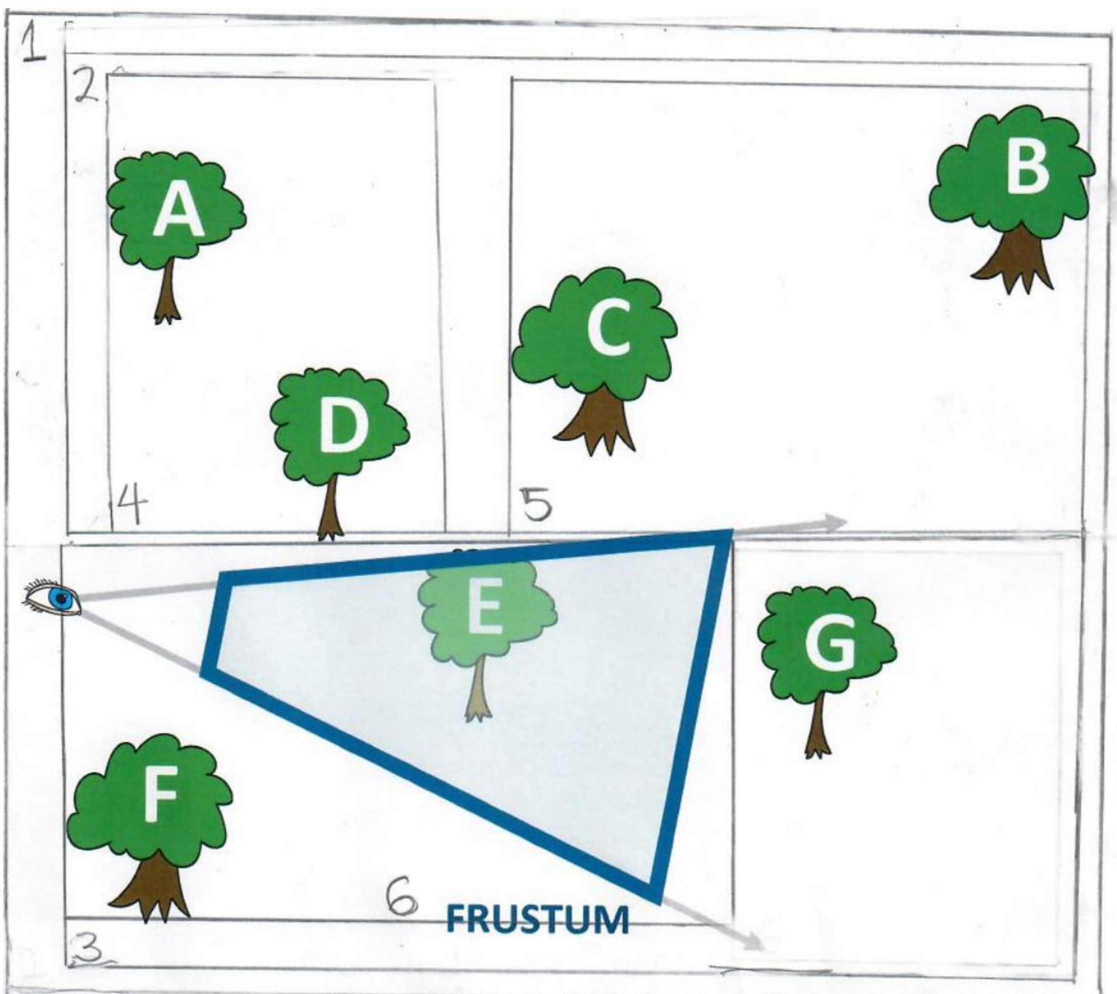


Figure 2: A 2D scene composed of seven oak trees viewed within a frustum (blue parallelogram).

The Bounding Box is shown as above.

We can partition the oaks in the AABB as above.

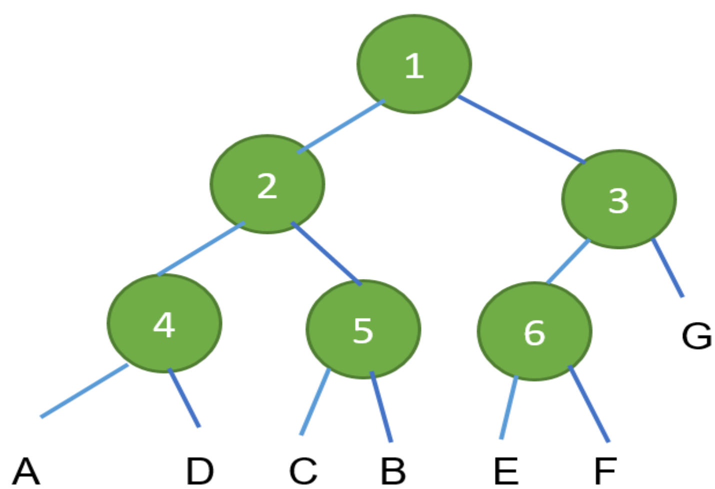
First, we use a box 1 to include 7 oaks.

Second, we use box 2 and box 3 to divide 7 oaks into ABCD and EFG.

Third, in box 2, we can partition the AD to a box 4 and BC as box 5.

Forth, we can divide EFG into EF or EG in a box 6.

The following graph is shown in balanced binary AABB tree.



c) Assuming a fast box_frustum_intersection subroutine, write a pseudocode algorithm to traverse a given AABB tree and gather all oaks that may intersect with the frustum.

In AABB tree, we stored the box information in the box node and the seven oaks information in its node.

V_frustum include the information of its shape, area, and other information which can detect whether collision occurred.

LinkList *box_frustum_intersection(AABB T, V_frustum, LinkedList *a){

    if T.current is leaf {

```
        if T.current intersect V_frustum{
        store the node into LinkedList a
        return the Linkedlist a;
        }
        }
        if V_frustum intersect T.box{
        box_frustum_intersection(T. left, V_frustum,a);
        box_frustum_intersection(T. right, V_frustum,a);
        }
}
```