

# Trabajo Realidad Aumentada Laberinto 3D

César Bragagnini - [cesarbrma91@gmail.com](mailto:cesarbrma91@gmail.com)

# 1. Matriz Rotation a Ángulos de Euler

Los ángulos de Euler constituyen un conjunto de tres coordenadas angulares que sirven para especificar la orientación de un sistema de referencia de ejes ortogonales, normalmente móvil, respecto a otro sistema de referencia de ejes ortogonales normalmente fijos. Estos ángulos  $\theta_1, \theta_2, \theta_3$  forman los ángulos respecto a los ejes X,Y,Z

```
glm::vec3 rotationMatrixToEulerAngles(Mat &R)
{
    float sy = sqrt(R.at<double>(0, 0) * R.at<double>(0,
        0) + R.at<double>(1, 0) * R.at<double>(1, 0));
    bool singular = sy < 1e-6;
    float x, y, z;
    if (!singular){
        x = atan2(R.at<double>(2, 1), R.at<double>(2, 2));
        y = atan2(-R.at<double>(2, 0), sy);
        z = atan2(R.at<double>(1, 0), R.at<double>(0, 0));
    }else{
        x = atan2(-R.at<double>(1, 2), R.at<double>(1,
            1));
        y = atan2(-R.at<double>(2, 0), sy);
        z = 0;
    }
    return glm::vec3(x, y, z);
}
```

Código para obtener los ángulos de Euler a partir de la matriz de rotación

## 2. Cuando varia la posicion en X y en Y

```
glm::vec3 velocSphere(0.0, 0.0, 0.0);

double staY = 0.0;
double staX = 3.1416;
double staZ = 0.0;
double rangePerY = 0.08;
double rangePerX = 0.15;

double vecX = 0.5;
double vecY = 0.5;

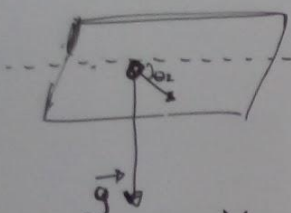
velocSphere.y = genVelocityByTheta(staX, rangePerX, anglesEuler.x, vecX, 1);
velocSphere.x = genVelocityByTheta(staY, rangePerY, anglesEuler.y, vecY, 0);
```

Considerar que existe un rango [0.0-0.15] para  $\text{abs}(\text{theta}_1)$ , donde no existe movimiento.

Considerar que existe un rango [3.0616-3.1416] para  $\text{abs}(\text{theta}_2)$ , donde no existe movimiento

```
double genVelocityByTheta(double staX, double rangePer,
double x, double vecX, int flag)
{
    double res = 0;
    if (flag) {
        if (abs(x) < staX - rangePer)
            res = vecX * (x < 0 ? -1 : 1);
    }
    else {
        if (abs(x) > staX + rangePer)
            res = vecX * (x > 0 ? 1 : -1);
    }
    return res;
}
```

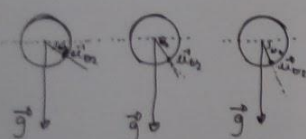
plano xy



$\text{eje } x \Rightarrow y=0$

$\vec{g}$

Cuando  $\theta_2 \neq 0 \Rightarrow V_y = \text{constante}$  (positiva o negativa)  
conservación  $V_z$   
tenemos



$90 - \theta_2$

$\vec{g}$

$\text{Proj}_{\vec{u}_{\theta_2}} \vec{g}$

$V_z = \|\text{Proj}_{\vec{u}_{\theta_2}} \vec{g}\|$

$\vec{V}_{\theta_2} = (0, V_y, \|\text{Proj}_{\vec{u}_{\theta_2}} \vec{g}\|)$

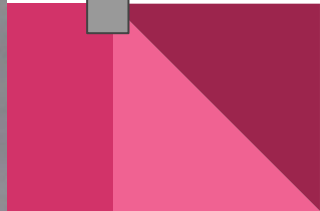
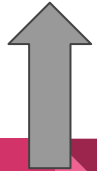
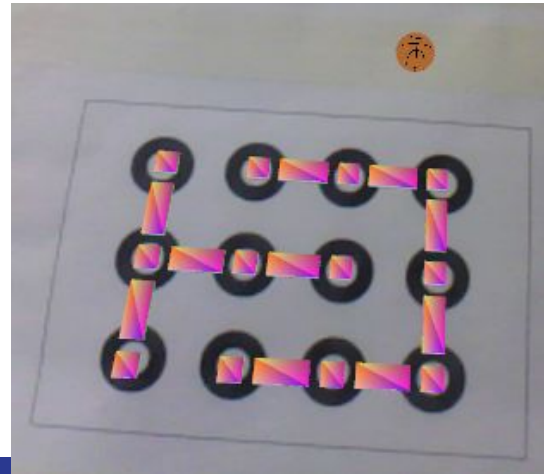
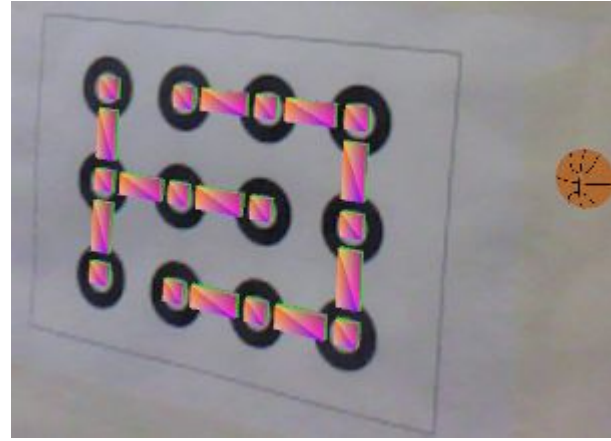
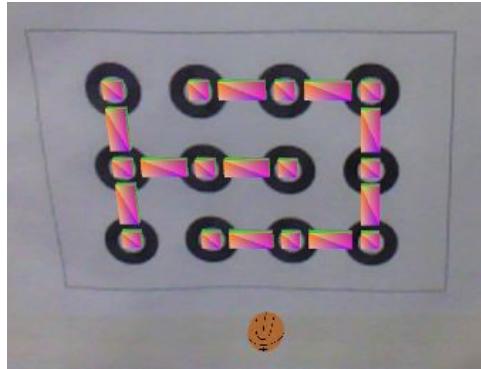
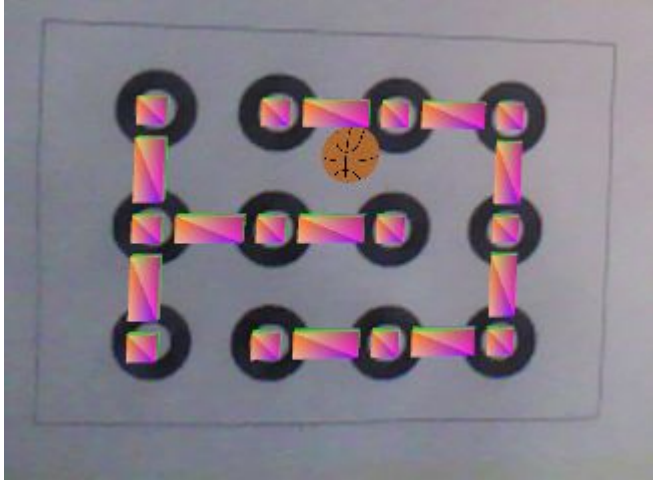
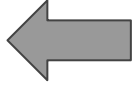
para  $\theta_1$  eje y:  $V_x = \text{constante}$

$\vec{V}_{\theta_1} = (V_x, 0, \|\text{Proj}_{\vec{u}_{\theta_1}} \vec{g}\|)$

$\vec{V}_{\text{bolita}} = \vec{V}_{\theta_1} + \vec{V}_{\theta_2}$

$\vec{V}_{\text{bolita}} = (V_x, V_y, \|\text{Proj}_{\vec{u}_{\theta_1}} \vec{g}\| + \|\text{Proj}_{\vec{u}_{\theta_2}} \vec{g}\|)$

## 4. Resultados



# Gracias



# Trabajo Realidad Aumentada Laberinto 3D

César Bragagnini - [cesarbrma91@gmail.com](mailto:cesarbrma91@gmail.com)