1.5.3 Getting coordinates of four corners

Store three-dimensional coordinates using Vector 3 type variables, variable names as intuitive as possible.

These codes already exist in scripts of asset.

```
private Vector3 TopLeft_Pl_W;
private Vector3 BottomLeft_Pl_W;
private Vector3 TopRight_Pl_W;
private Vector3 BottomRight_Pl_W;
```

The coordinates of the four corners of the recognition Image can not be extracted directly. The location information that can be extracted directly is the position of the recognition Image itself, that is, the center of the recognition Image.

Apply for public variables to store the identification Image itself, and apply for private variables to store the world coordinates of the identification Image.

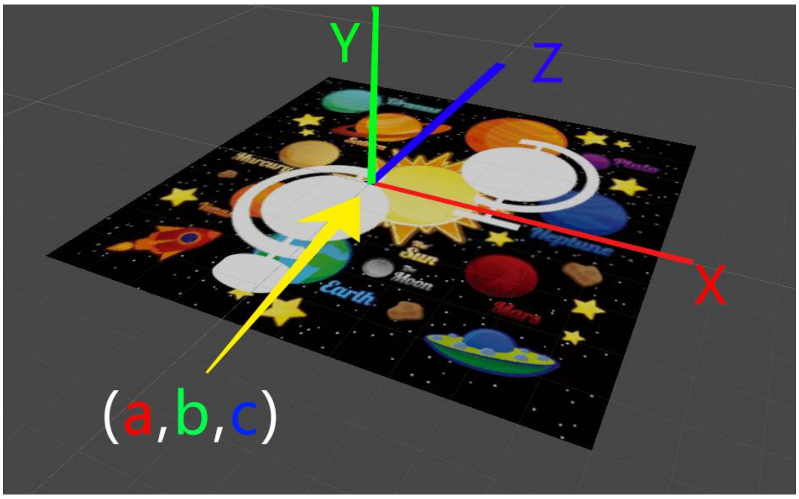
```
public GameObject Card_Track;
private Vector3 Center_Card;
```

Get the world coordinates of the recognition map:

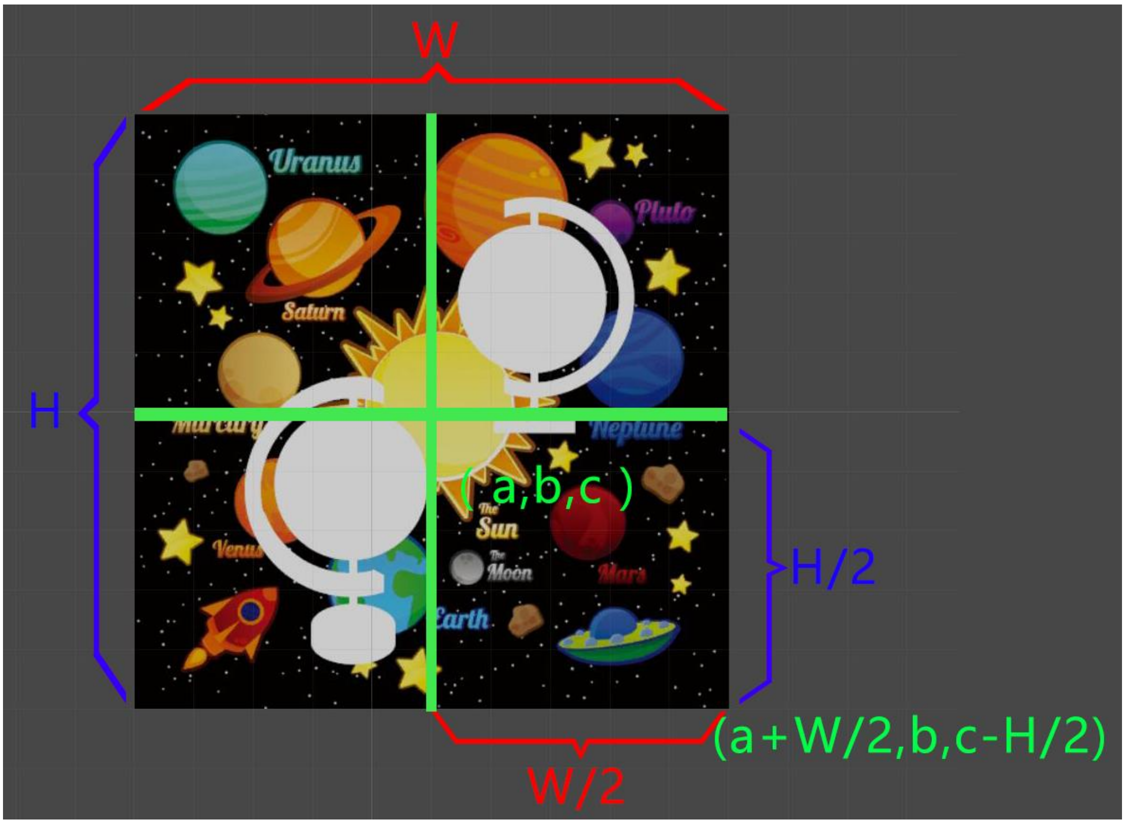
```
Center_Card = Card_Track.transform.position;
```

Since the length and width of the recognition Image are determined, the coordinates of the four corners can be obtained by the coordinates of the length and width and the center.

As shown in the figure, for example, the current world coordinates of the recognition Image are (a, b, c), the height of the recognition Image is H, and the width of the recognition Image is W.



Then the coordinates of the lower right corner are equal to (a+H/2, b, c-W/2).



By analogy, the coordinates in the upper left corner are （a-W/2,b,c+H/2）

The coordinates in the lower left corner are （a-W/2,b,c-H/2）

The coordinates in the upper right corner are （a+W/2,b,c+H/2）

As for the coordinates of the y-axis, they are always consistent with the coordinates of the recognition map, because the recognition Image itself is not inclined in the program, and the inclination in the picture is caused by the camera's perspective.

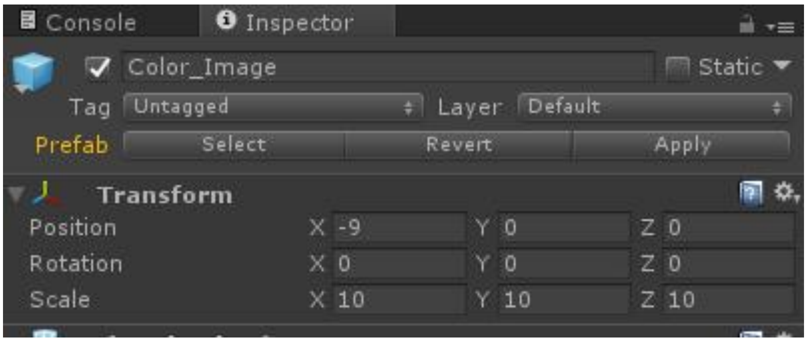
In the script, the variable is first applied to record half the width of the recognition Image.

```
private float Half_W;
private float Half_H;
```

Calculate the width and height of the Image. Getting Width of Recognition Image Use Code :

```
GetComponent<MeshFilter>().mesh.bounds.size.x;
```

Note that this code is to obtain the original width of the Image, if the recognition Image itself has been scaled, it is necessary to multiply the coefficient of scaling when using it. For example, if the recognition Image is scaled 10 times, the calculation will be multiplied by 10.



☑ Color_Image □ Static

Tag Untagged Layer Default

Transform

Position	X -9	Y 0	Z 0
Rotation	X 0	Y 0	Z 0
Scale	X 10	Y 10	Z 10

Since we need half the width and half the height, we have to multiply it by 0.5f.

```
Half_W=Card_Track.GetComponent<MeshFilter>().mesh.bounds.size.x*10*0.5f;
Half_H=Card_Track.GetComponent<MeshFilter>().mesh.bounds.size.z*10*0.5f;
```