查看NGUITools.FindCameraForLayer

public static Camera FindCameraForLayer(int layer)

{

int layerMask = 1 << layer;

Camera camera;

// 先在UICamera的list中获取，依赖于NGUI的UICamera

for (int i = 0; i < UICamera.list.size; ++i)

{

camera = UICamera.list.buffer[i].cachedCamera;

if (camera && (camera.cullingMask & layerMask) != 0) return camera;

}

// 判断是否是主相机（主相机是第一个enabled相机，并且Tag为Main Camera:注意，这里是Tag而不是名字（已证实））

camera = Camera.main;

if (camera)

{

Debug.Log("main camera is not null, camera = " + camera);

}

else

{

Debug.Log("main camera is null");

}

if (camera && (camera.cullingMask & layerMask) != 0) return camera;

// Camera.allCameras：所有enabled的Camera

Camera[] c = Camera.allCameras;

Debug.Log("c.Length = " + c.Length);

for (int i = 0, size = c.Length; i < size; i++)

{

Debug.Log("i = " + i + ", name = " + c[i].gameObject.name);

}

Camera[] c0 = GameObject.FindObjectsOfType<Camera>();

Debug.Log("c0.Length = " + c0.Length);

for (int i = 0, size = c0.Length; i < size; i++)

{

Debug.Log("i = " + i + ", name = " + c0[i].gameObject.name);

}

Camera[] c1 = (Camera[])Object.FindObjectsOfType(typeof(Camera));

Debug.Log("c1.Length = " + c1.Length);

for (int i = 0, size = c1.Length; i < size; i++)

{

Debug.Log("i = " + i + ", name = " + c1[i].gameObject.name);

}

Debug.Log("Camera.allCamerasCount = " + Camera.allCamerasCount);

Camera[] cameras = new Camera[Camera.allCamerasCount];

int camerasFound = Camera.GetAllCameras(cameras);

Debug.Log("camerasFound = " + camerasFound);

for (int i = 0; i < camerasFound; i++)

{

camera = cameras[i];

if (camera && camera.enabled && (camera.cullingMask & layerMask) != 0) return camera;

}

return null

}

 

