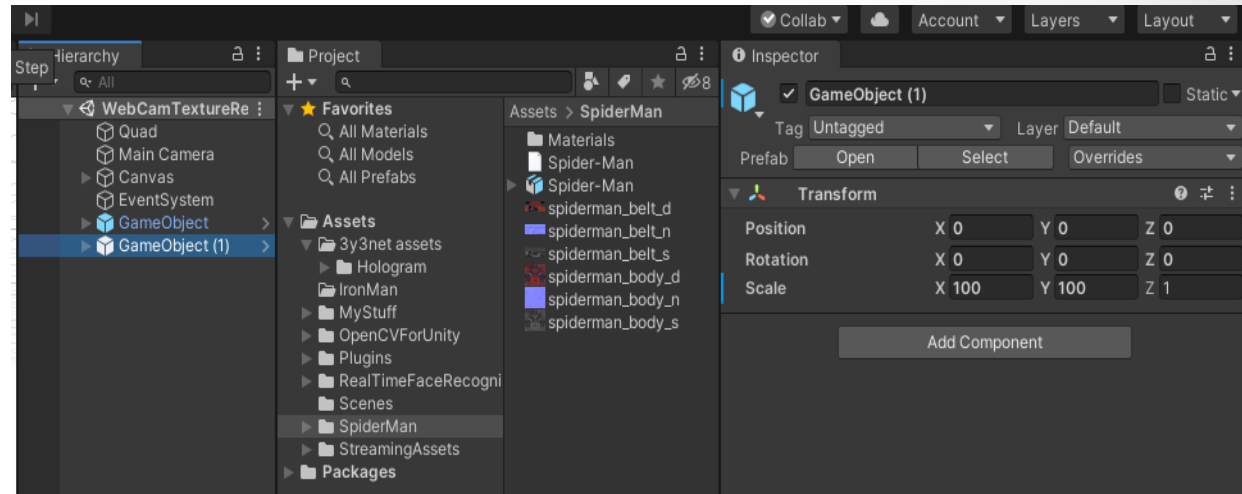
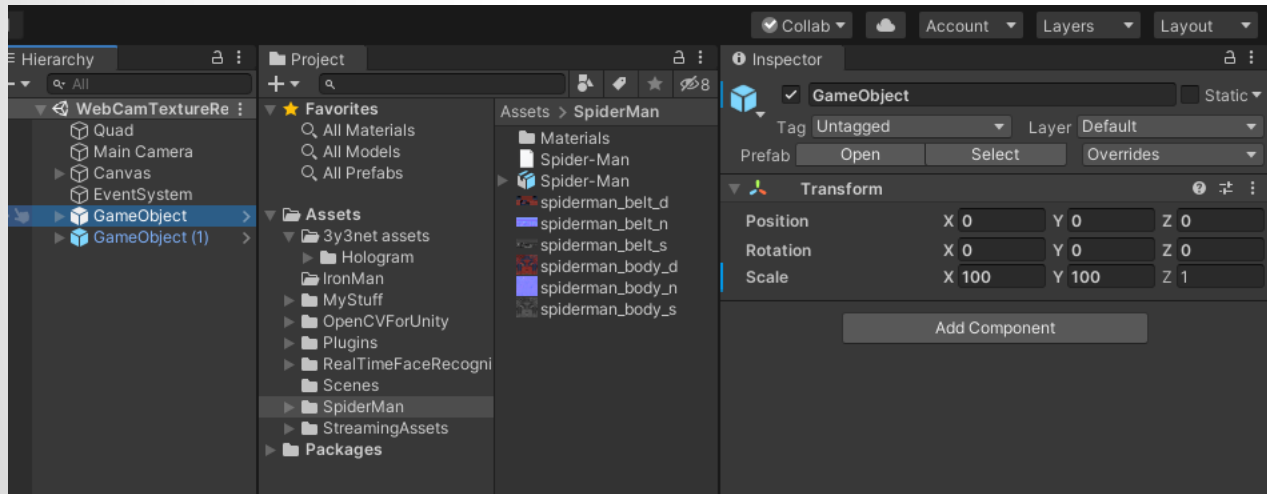


OpenCV를 활용한 AR 프로그래밍 심화

이준

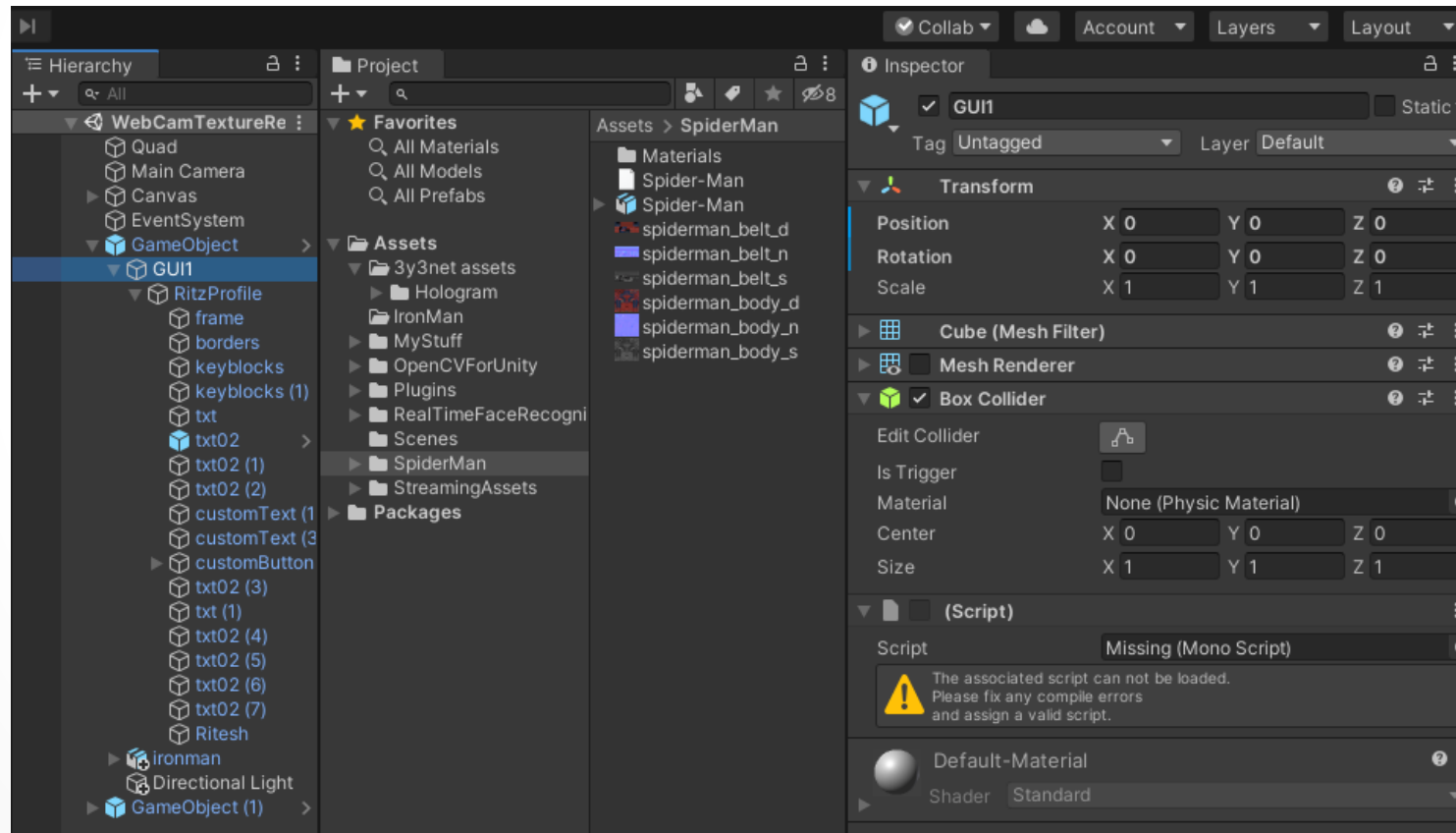
얼굴 인식 및 기억 방법 적용

- 지난시간 사용한 홀로그램 ID를 사용해보자!
 - 각각 GameObject 와 GameObject (1) (복사하여) 준비
 - X 및 Y 크기를 각각 100 으로 설정



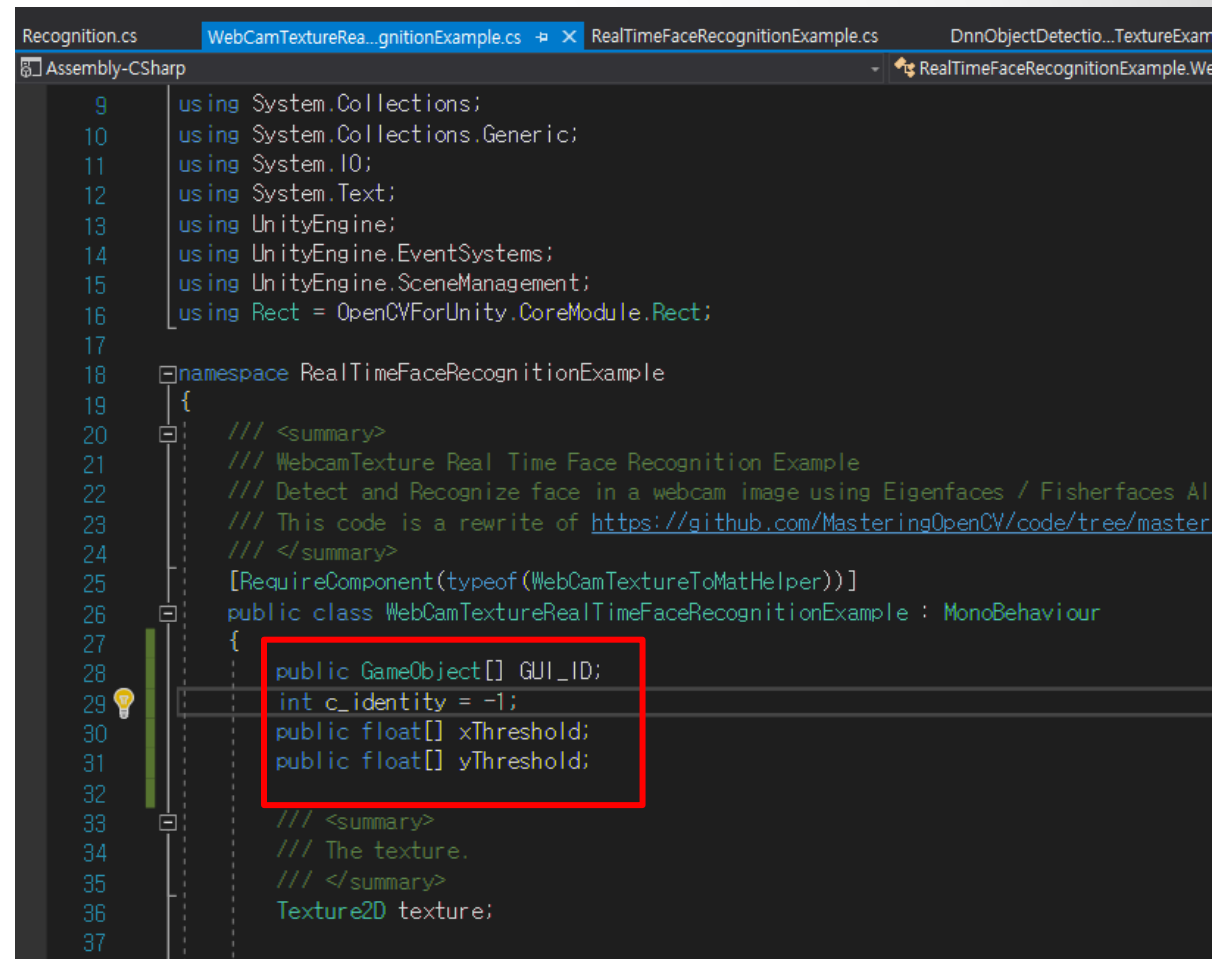
얼굴 인식 및 기억 방법 적용

- 프리팹 사용시 GUI1 은 0, 0, 0 으로 리셋반드시 해야함



얼굴 인식 및 기억 방법 적용

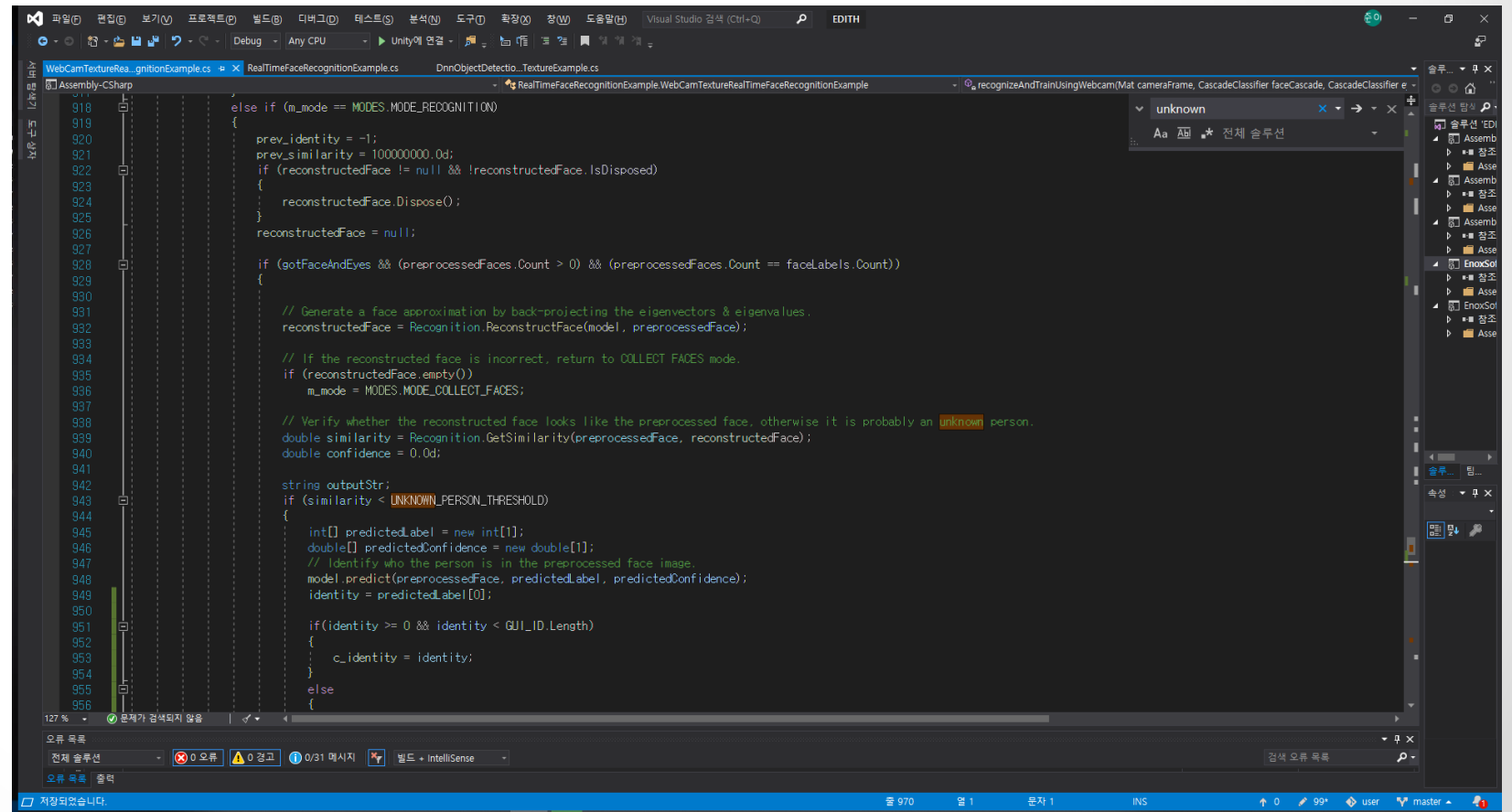
- 4개의 변수들 추가
 - GUI_ID : 배열로 GameObject들 관리
 - C_identity : 내부적으로 얼굴인식 결과확인
 - xThreshold, yThreshold : UI 를 맞춰주기 위한 배열



```
Recognition.cs WebCamTextureRealTimeFaceRecognitionExample.cs RealTimeFaceRecognitionExample.cs DnnObjectDetection...TextureExam
Assembly-CSharp RealTimeFaceRecognitionExample.W
9 using System.Collections;
10 using System.Collections.Generic;
11 using System.IO;
12 using System.Text;
13 using UnityEngine;
14 using UnityEngine.EventSystems;
15 using UnityEngine.SceneManagement;
16 using Rect = OpenCVForUnity.CoreModule.Rect;
17
18 namespace RealTimeFaceRecognitionExample
19 {
20     /// <summary>
21     /// WebcamTexture Real Time Face Recognition Example
22     /// Detect and Recognize face in a webcam image using Eigenfaces / Fisherfaces AL
23     /// This code is a rewrite of https://github.com/MasteringOpenCV/code/tree/master
24     /// </summary>
25     [RequireComponent(typeof(WebCamTextureToMatHelper))]
26     public class WebCamTextureRealTimeFaceRecognitionExample : MonoBehaviour
27     {
28         public GameObject[] GUI_ID;
29         int c_identity = -1;
30         public float[] xThreshold;
31         public float[] yThreshold;
32
33         /// <summary>
34         /// The texture.
35         /// </summary>
36         Texture2D texture;
37     }
38 }
```

얼굴 인식 및 기억 방법 적용

- recognizeAndTrainUsingWebcam 부분에서
 - 실제로 트레이닝 파일이 생성되면, 그후에는 인식을 수행



얼굴 인식

- recognizeAndTrainUsingWebcam 함수에서
- 인식 부분에서 다음과 같이 추가하기!
- Id 에 일치하는 게임 오브젝트의 위치 이동을 수행

```
if (similarity < UNKNOWN_PERSON_THRESHOLD)
{
    int[] predictedLabel = new int[1];
    double[] predictedConfidence = new double[1];
    // Identify who the person is in the preprocessed face image.
    model.predict(preprocessedFace, predictedLabel, predictedConfidence);
    identity = predictedLabel[0];
    Debug.Log("identity..." + identity);
    if (identity >= 0 && identity < GUI_ID.Length)
    {
        c_identity = identity;
    }
    else
    {
        c_identity = -1;
    }

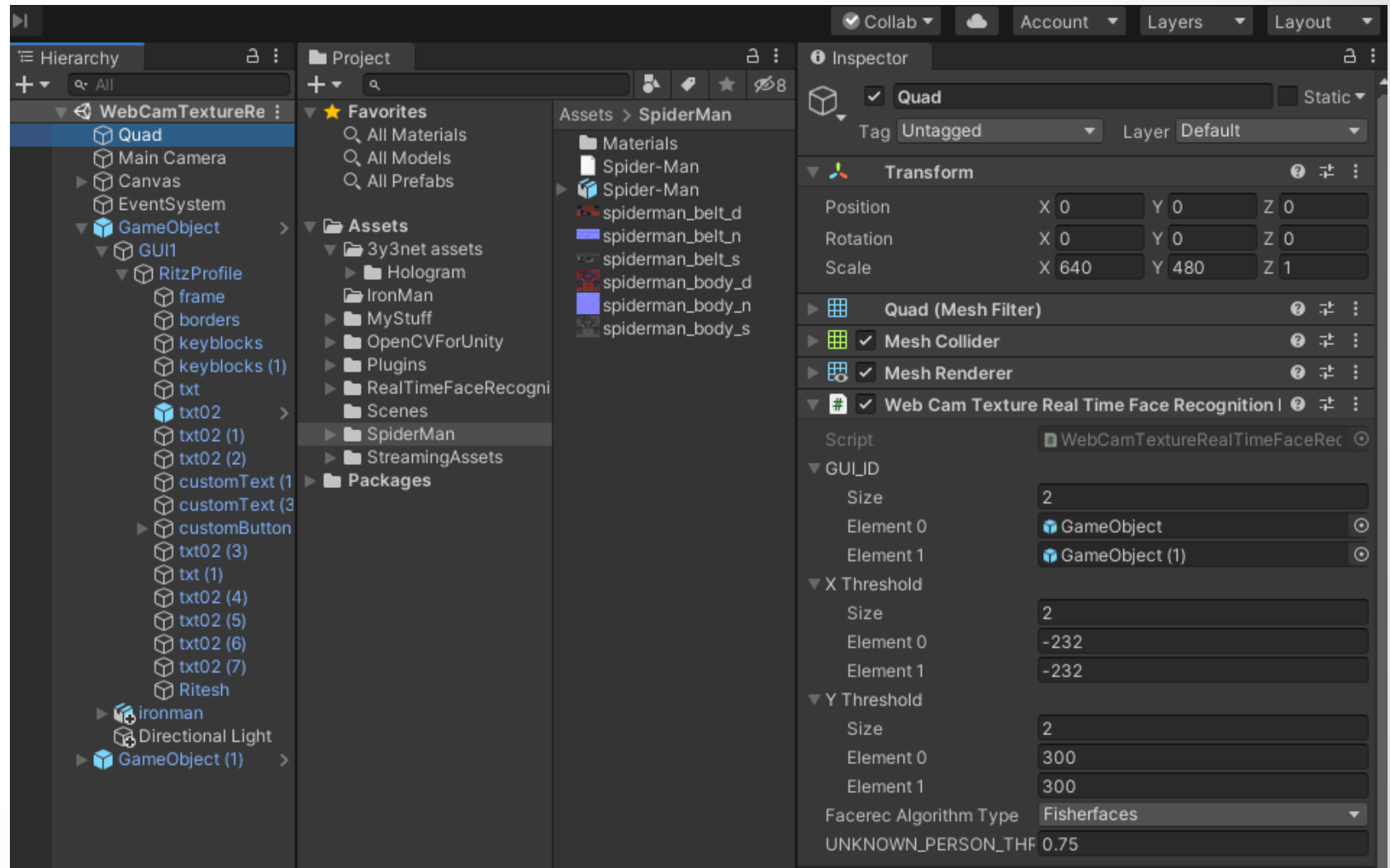
    if (c_identity != -1)
    {
        GUI_ID[c_identity].SetActive(true);
    }

    confidence = predictedConfidence[0];
    outputStr = identity.ToString();
    prev_identity = identity;

    switch (c_identity)
    {
        case 0:
            GUI_ID[0].transform.position = new Vector3(faceRect.x + xThreshold[0], -faceRect.y + yThreshold[0], 0);
            Debug.Log("Positon0:" + GUI_ID[0].transform.position);
            GUI_ID[0].SetActive(true);
            GUI_ID[1].SetActive(false);
            break;
        case 1:
            GUI_ID[1].transform.position = new Vector3(faceRect.x + xThreshold[1], -faceRect.y + yThreshold[1], 0);
            Debug.Log("Positon1:" + GUI_ID[0].transform.position);
            GUI_ID[0].SetActive(false);
            GUI_ID[1].SetActive(true);
            break;
        default:
            GUI_ID[0].SetActive(false);
            GUI_ID[1].SetActive(false);
            break;
    }
}
```

얼굴 인식 및 기억 방법 적용

- 다음과 같이 변수 설정하기

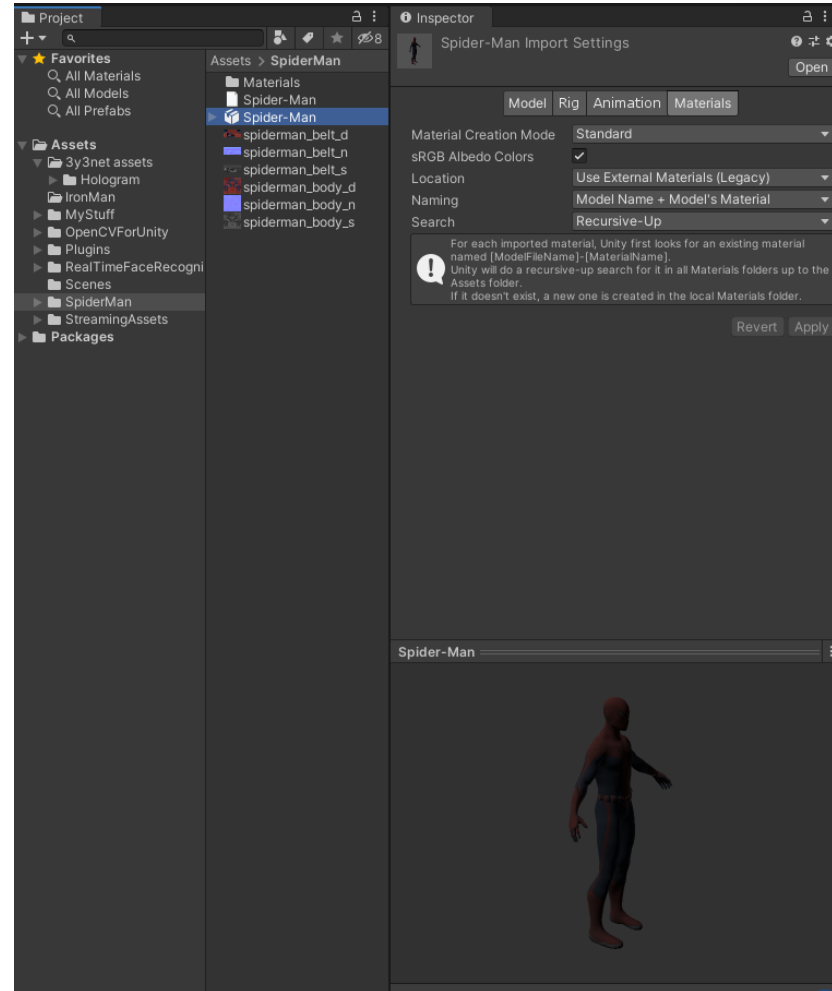
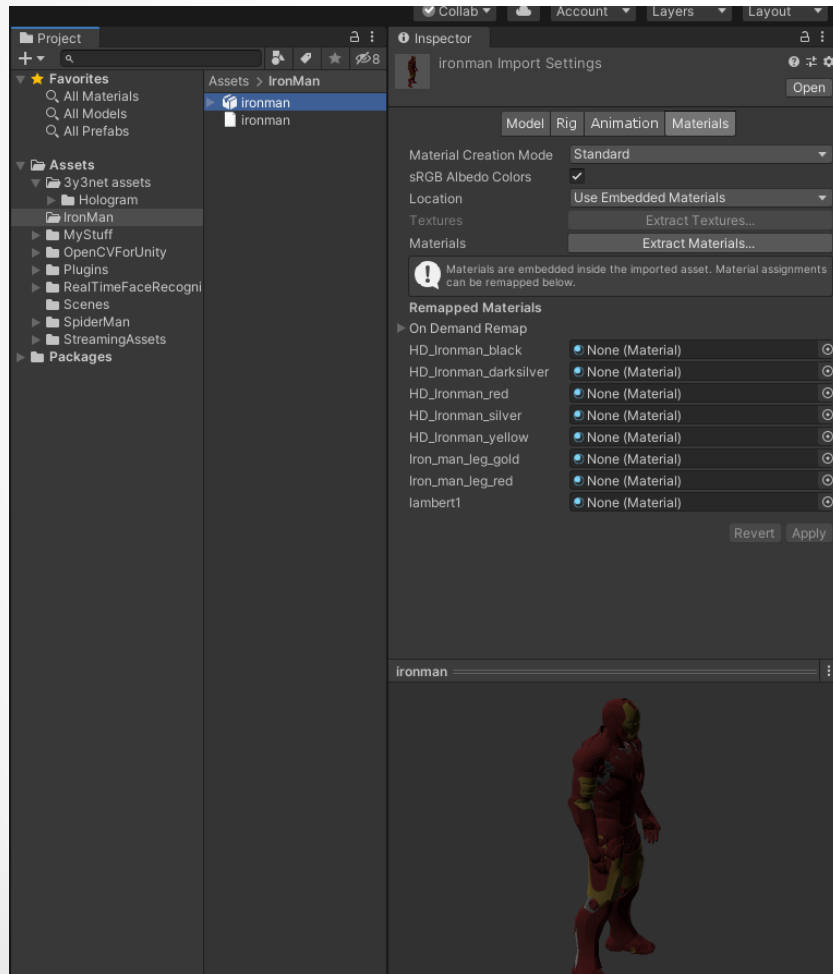


얼굴 인식 및 기억 방법 적용

- 각각 아이언맨 및 스파이더맨 모델도 같이 띄워 보기!
- 다음의 무료 모델들을 링크에서 다운 받기
- 아이언맨 :
<https://drive.google.com/file/d/1nBzSXBrQjaOc96wlBC-elf-Nby41n0Tn/view?usp=sharing>
- 스파이더맨 :
https://drive.google.com/file/d/1mxKcG1pYsGYjtlqiX-uA_Ff5RHGmQ6CE/view?usp=sharing

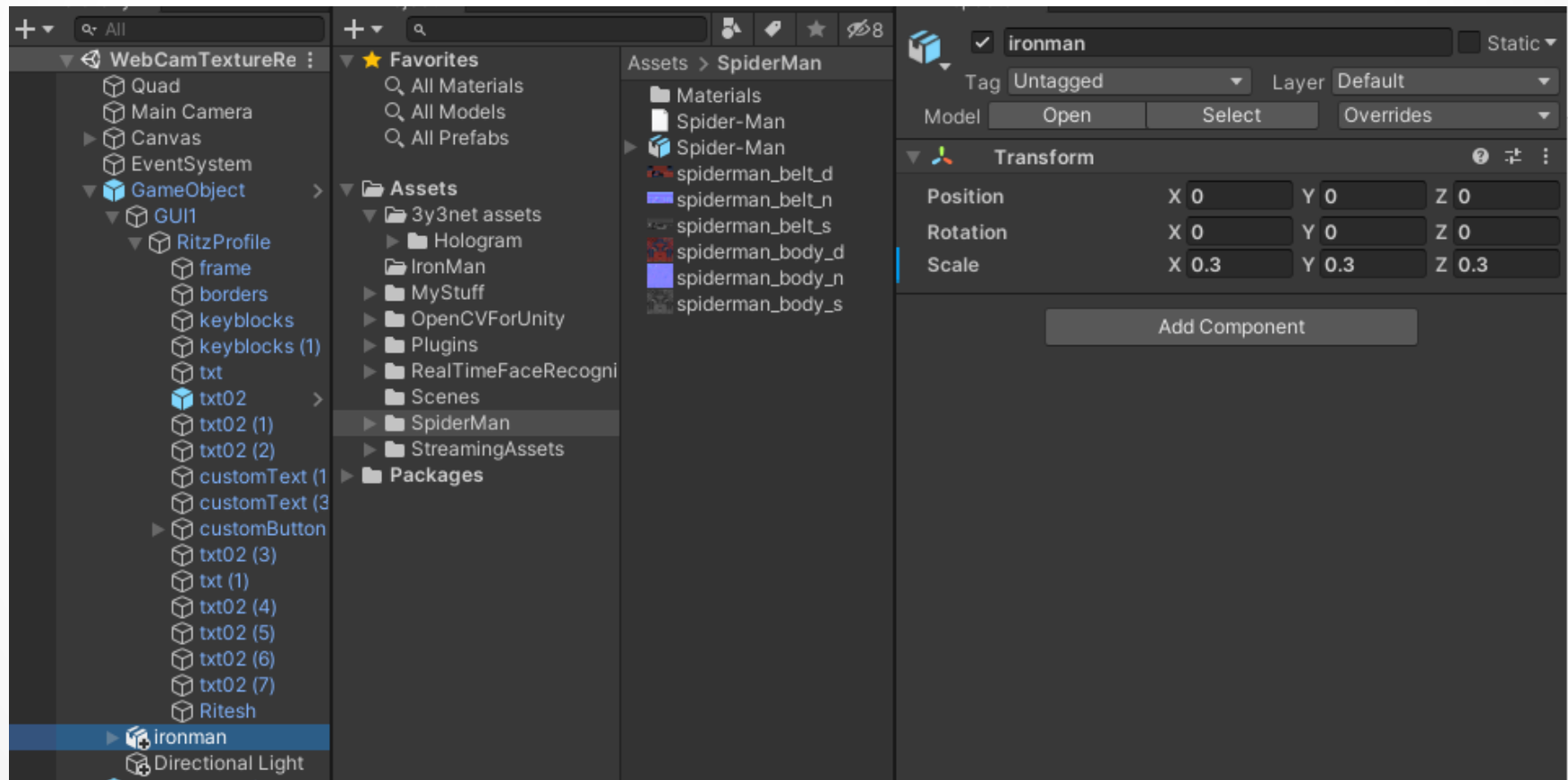
얼굴 인식 및 기억 방법 적용

- 각각 폴더를 만들고 애셋 임포트를 시키고 확인해보자



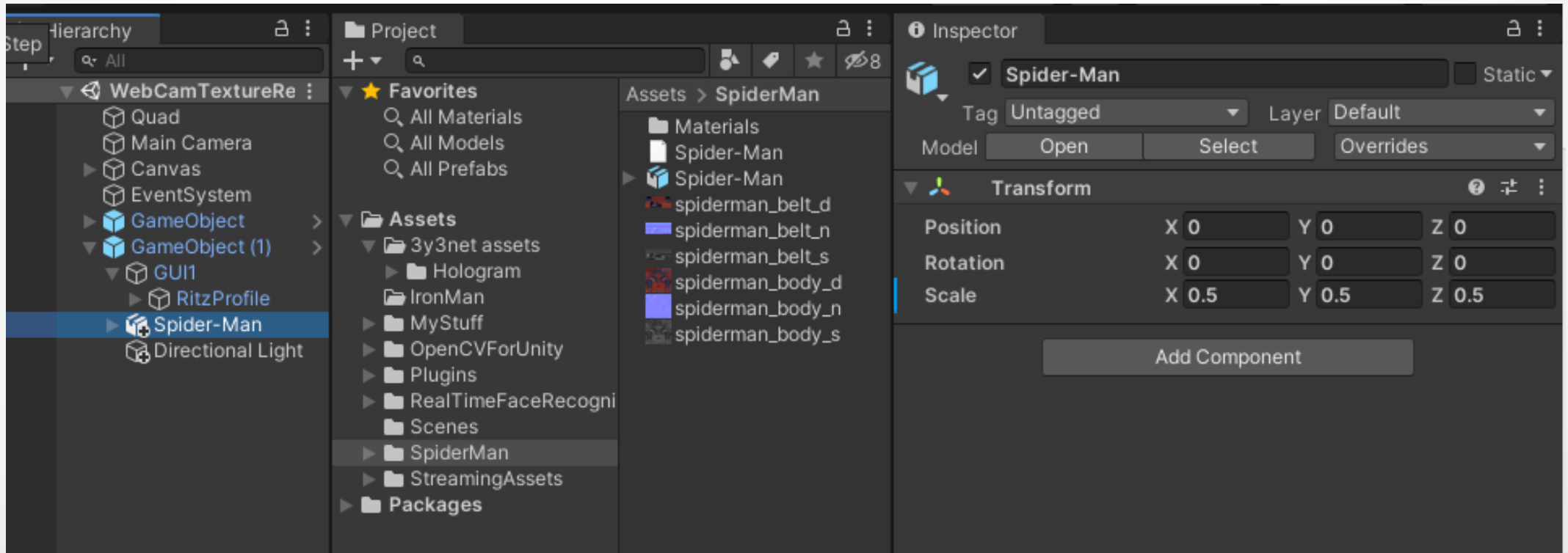
얼굴 인식 및 기억 방법 적용

- GameObject에 아이언맨 추가
 - 텍스처가 어두우므로 Directional Light도 추가하기!

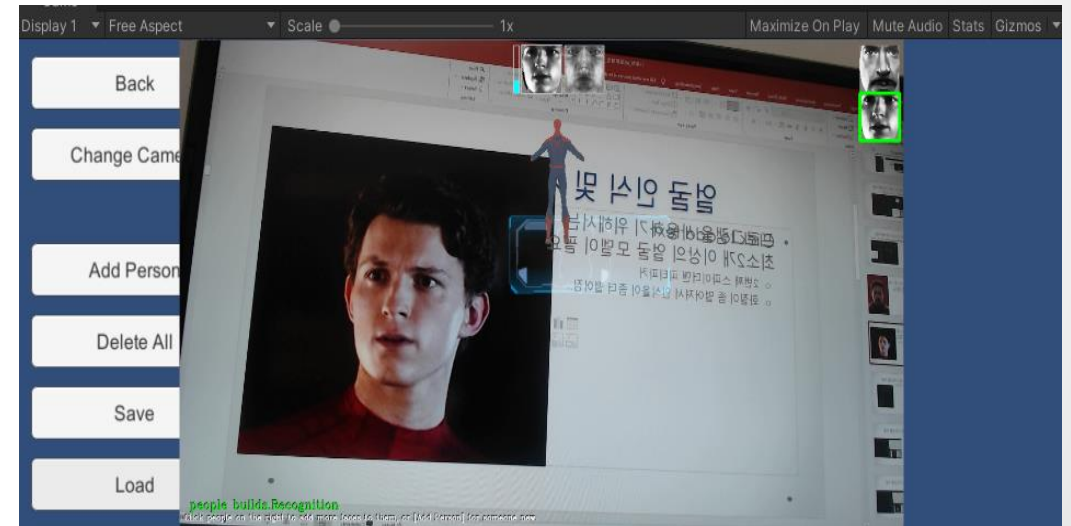
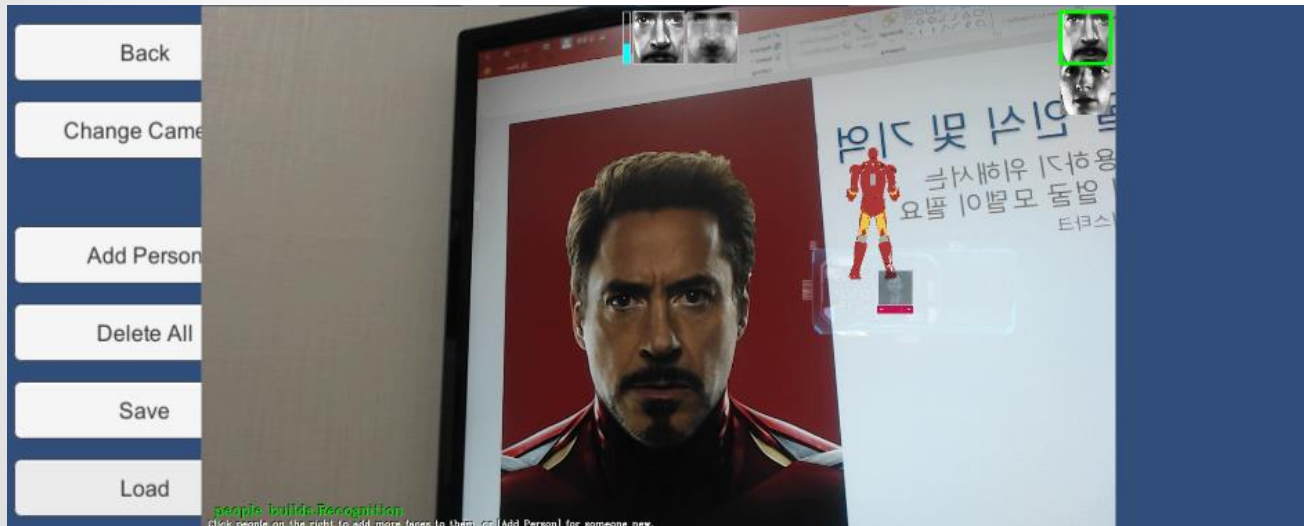


얼굴 인식 및 기억 방법 적용

- GameObject (1)에 스파이더맨 추가
 - 텍스처가 어두우므로 Directional Light도 추가하기!



얼굴 인식 및 기억 방법 적용



다양한 이미지를 인식할 수 있을까?



다양한 이미지를 인식할 수 있을까?

