

X3D Mobile
Unity X3D Viewer
Implementation (Updates)

Web3D Standardization Meeting at SIGGRAPH 2018
Vancouver, Canada

August 13, 2018

Myeong Won Lee (The University of Suwon)

Mobile VR and Sensors

- HMD (Gear VR)
 - Camera, Acceleration, Zyro, Proximity, Magnetic
- Mobile phone
 - Camera, Zyro, GPS, Gesture, Proximity, Light, RGB, Acceleration, Magnetic, Temperature, Pressure, Atmosphere



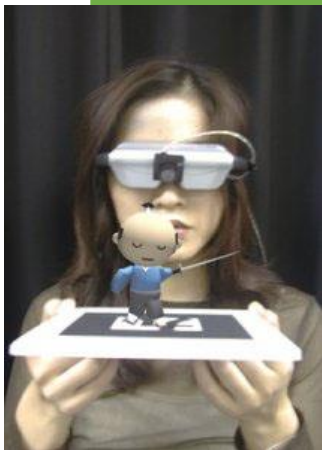
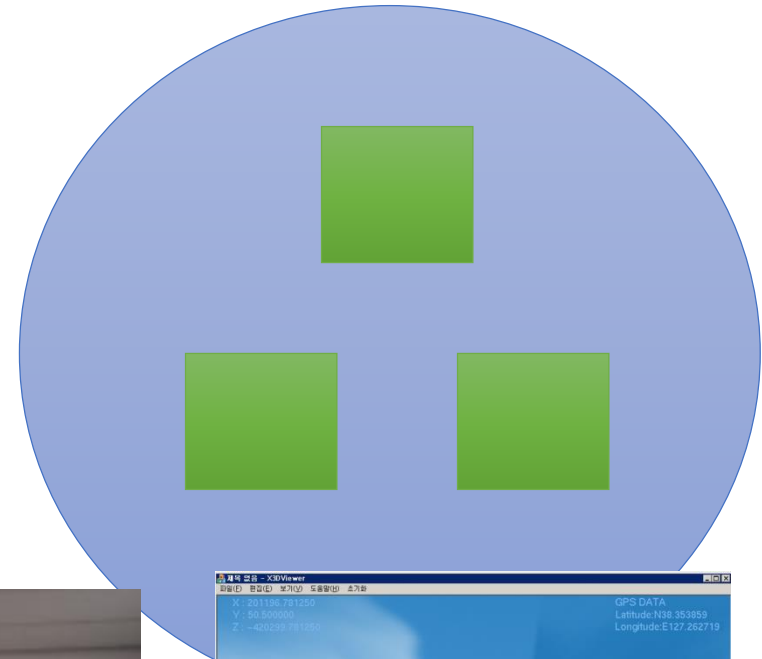
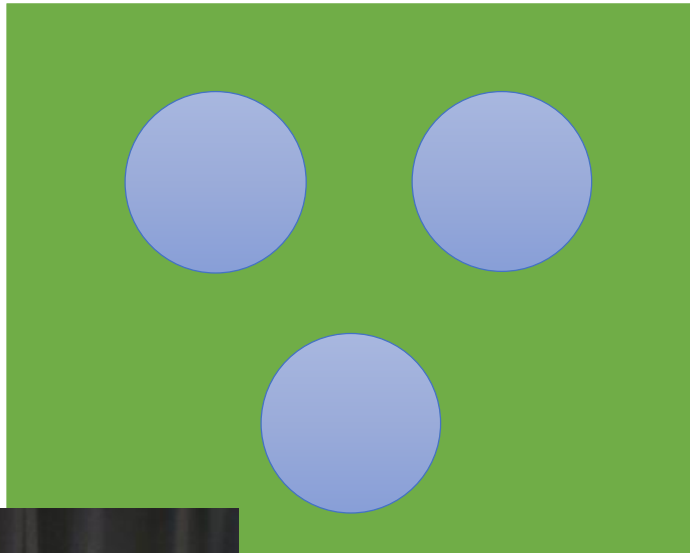
- An example of mobile travel VR
- 3D VR world
 - Geo-synchronized
 - Units specified
- Sensors
 - GPS, Camera, Acceleration, Zyro, Proximity, Light, RGB, Temperature



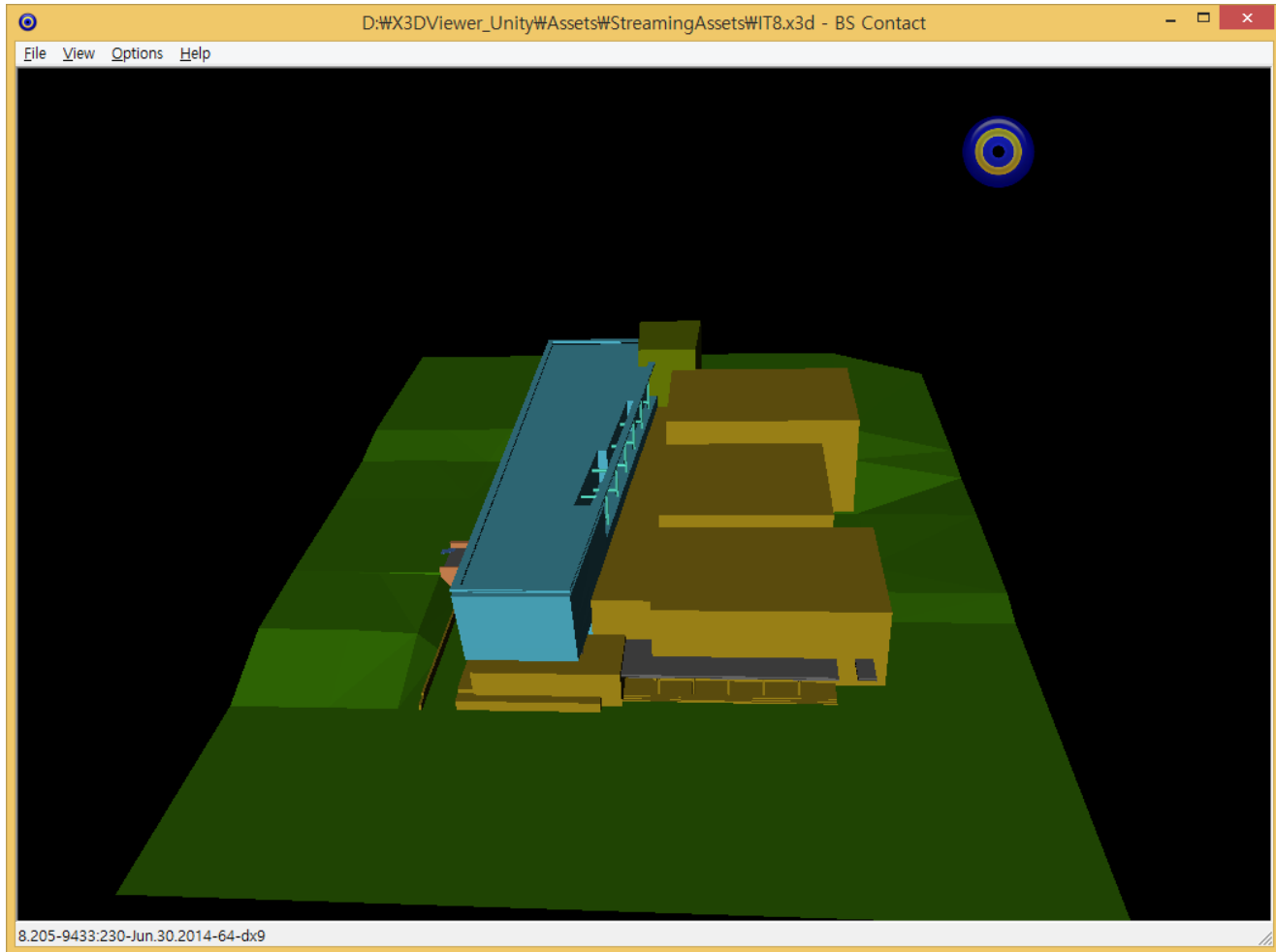
Mobile MAR

Video worlds + Augmented objects

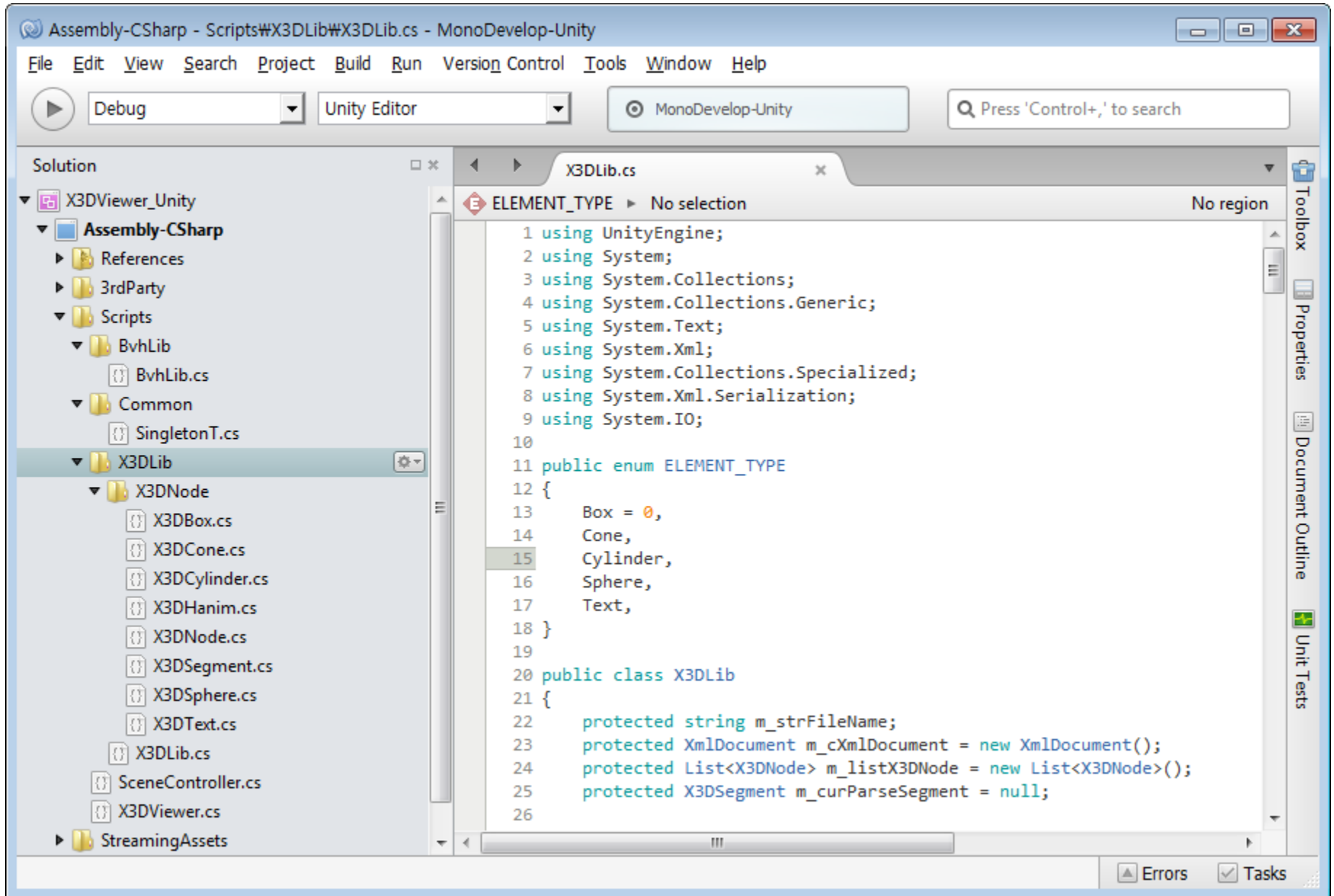
3D VR worlds + Augmented video objects



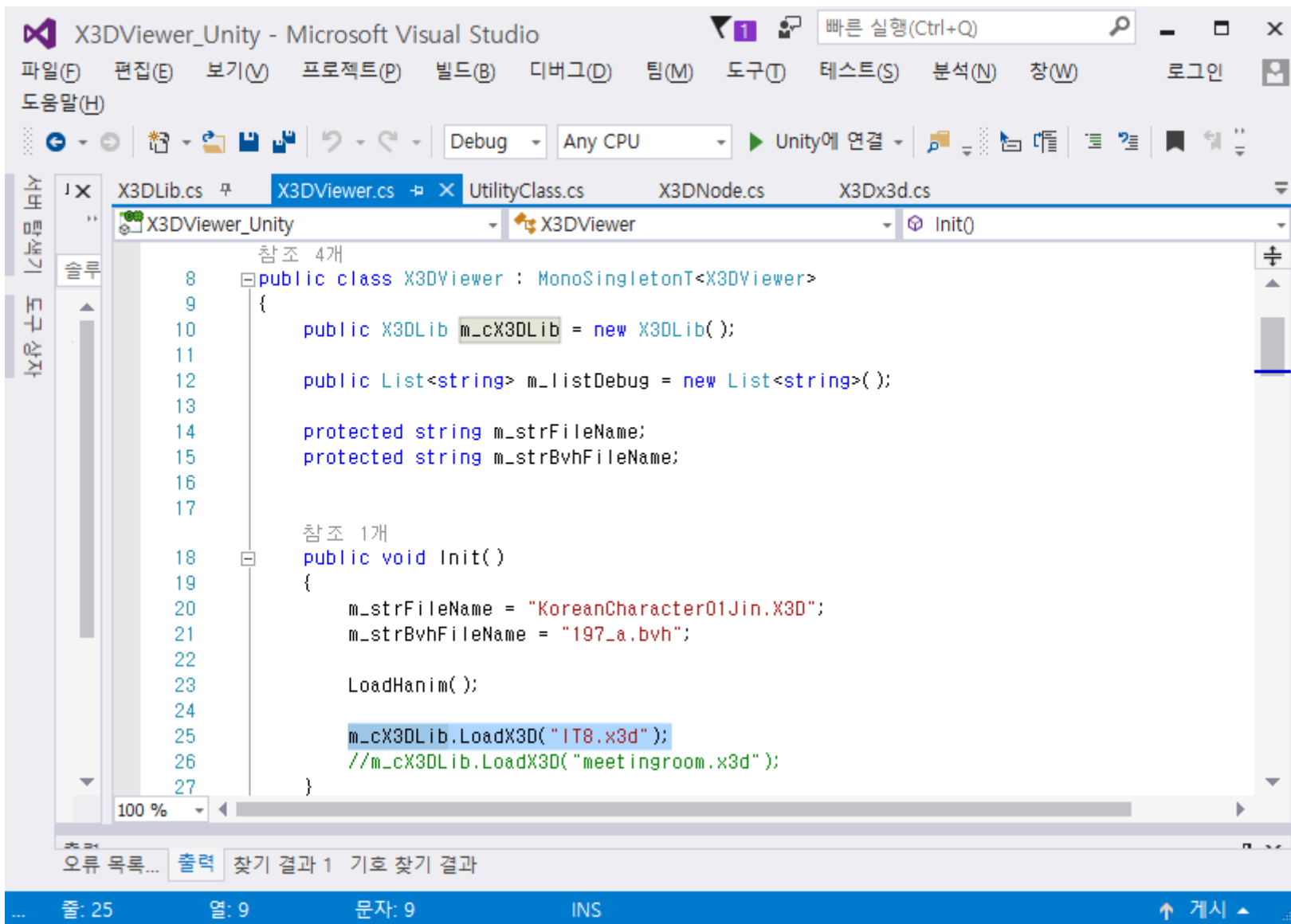
An Example X3D File



Unity H-Anim Viewer Organization

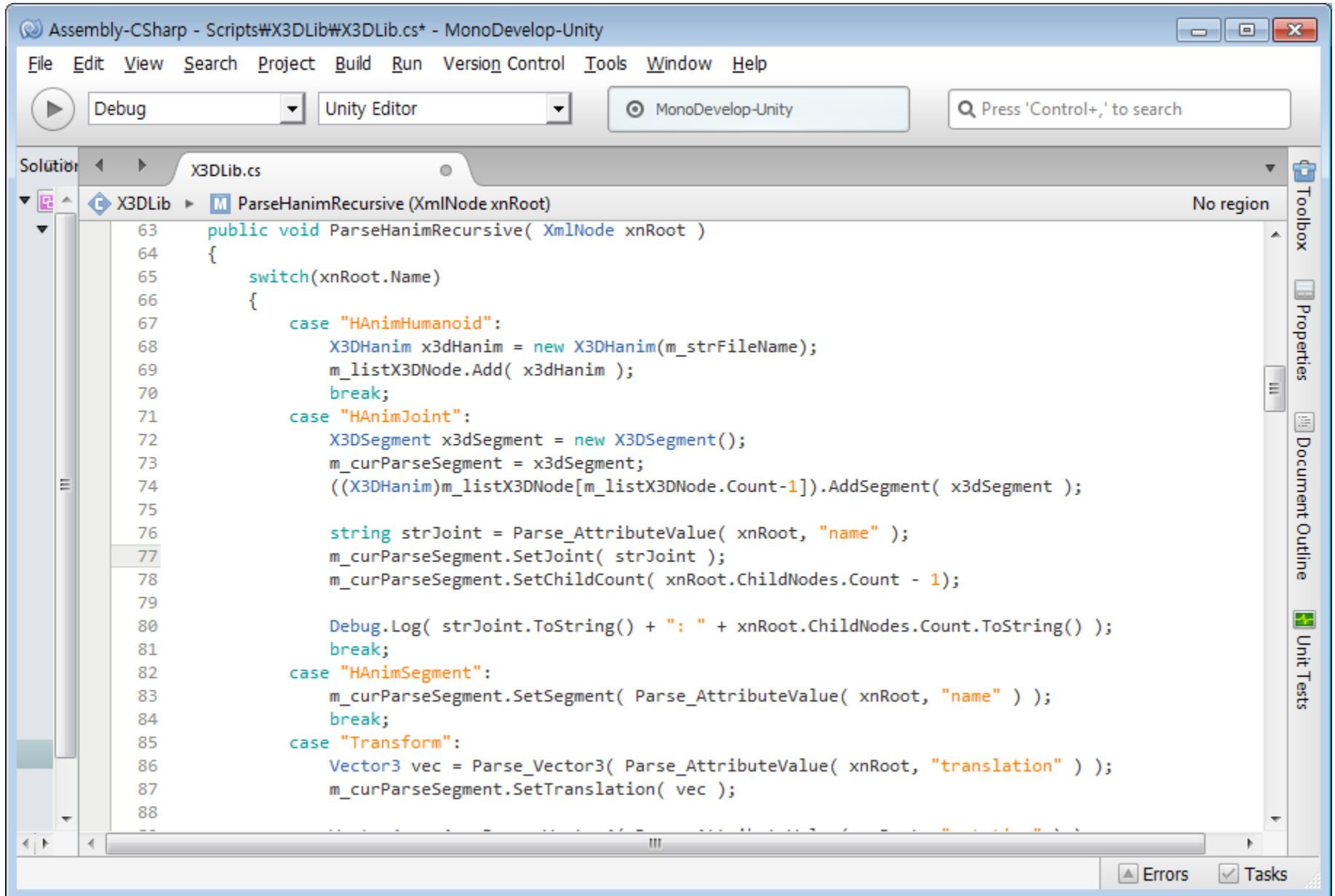


Import an X3D File into Unity



```
X3DViewer_Unity - Microsoft Visual Studio
빠른 실행(Ctrl+Q)
파일(F) 편집(E) 보기(V) 프로젝트(P) 빌드(B) 디버그(D) 팀(M) 도구(T) 테스트(S) 분석(N) 창(W) 로그인
도움말(H)
Debug Any CPU Unity에 연결
X3DLib.cs X3DViewer.cs UtilityClass.cs X3DNode.cs X3Dx3d.cs
X3DViewer_Unity X3DViewer Init()
참조 4개
8 public class X3DViewer : MonoSingletonT<X3DViewer>
9 {
10     public X3DLib m_cX3DLib = new X3DLib();
11
12     public List<string> m_listDebug = new List<string>();
13
14     protected string m_strFileName;
15     protected string m_strBvhFileName;
16
17
18     참조 1개
19     public void Init()
20     {
21         m_strFileName = "KoreanCharacter01Jin.X3D";
22         m_strBvhFileName = "197_a.bvh";
23
24         LoadHanim();
25
26         m_cX3DLib.LoadX3D("IT8.x3d");
27         //m_cX3DLib.LoadX3D("meetingroom.x3d");
28     }
29
오류 목록... 출력 찾기 결과 1 기호 찾기 결과
줄: 25 열: 9 문자: 9 INS 게시
```

X3D Parsing in Unity

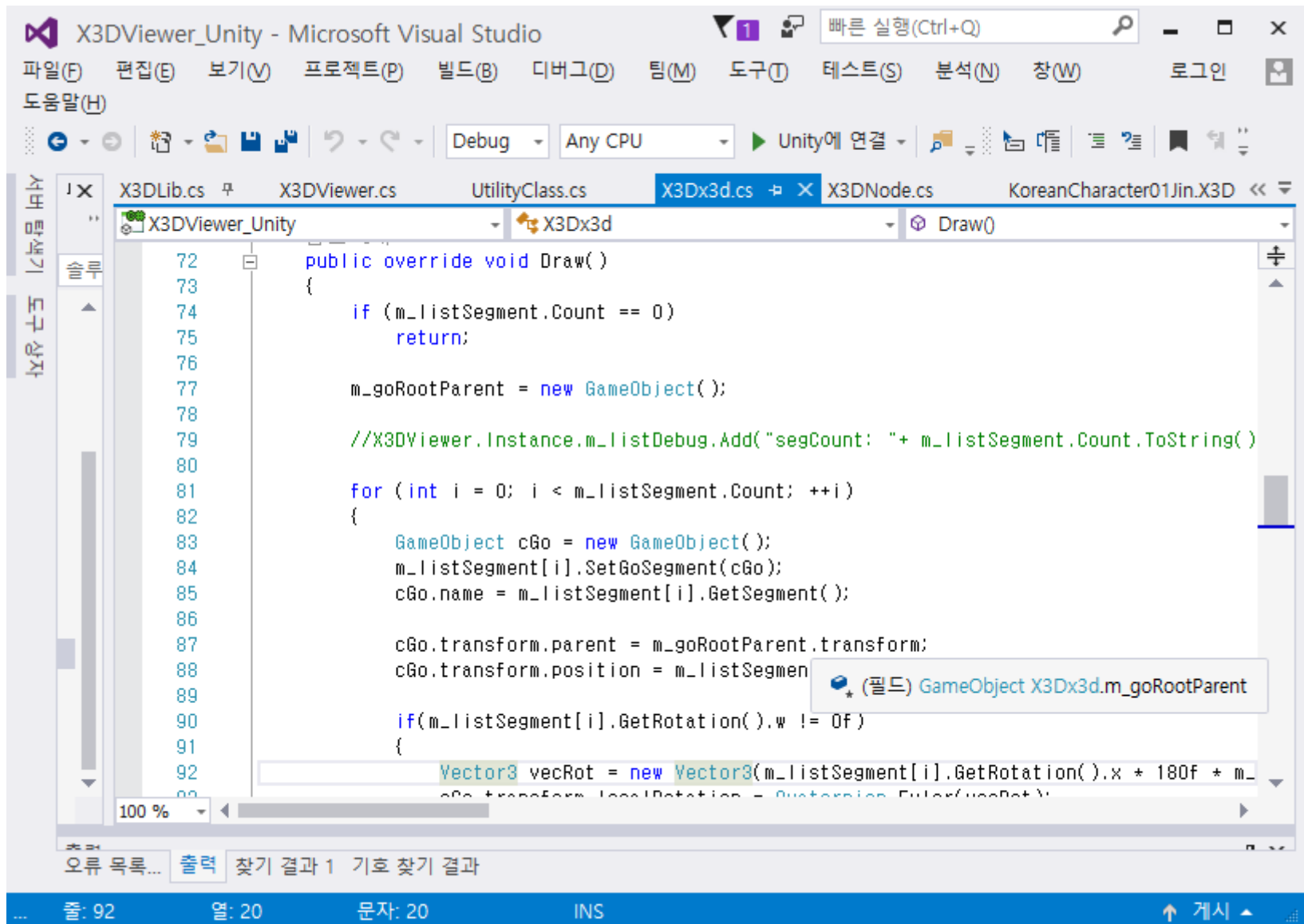


The image shows a screenshot of the MonoDevelop-Unity IDE. The main window displays the source code for `ParseHanimRecursive` in `X3DLib.cs`. The code is a recursive function that parses XML nodes representing HAnim data. It uses a `switch` statement to handle different node types: `HAnimHumanoid`, `HAnimJoint`, `HAnimSegment`, and `Transform`. The `HAnimJoint` case includes a `Debug.Log` statement to output the joint name and the number of child nodes.

```
Assembly-CSharp - Scripts\X3DLib\X3DLib.cs* - MonoDevelop-Unity
File Edit View Search Project Build Run Version Control Tools Window Help
Debug Unity Editor MonoDevelop-Unity Press 'Control+', to search
Solution X3DLib.cs
X3DLib ParseHanimRecursive (XmlNode xnRoot) No region
63 public void ParseHanimRecursive( XmlNode xnRoot )
64 {
65     switch(xnRoot.Name)
66     {
67         case "HAnimHumanoid":
68             X3DHanim x3dHanim = new X3DHanim(m_strFileName);
69             m_listX3DNode.Add( x3dHanim );
70             break;
71         case "HAnimJoint":
72             X3DSegment x3dSegment = new X3DSegment();
73             m_curParseSegment = x3dSegment;
74             ((X3DHanim)m_listX3DNode[m_listX3DNode.Count-1]).AddSegment( x3dSegment );
75
76             string strJoint = Parse_AttributeValue( xnRoot, "name" );
77             m_curParseSegment.SetJoint( strJoint );
78             m_curParseSegment.SetChildCount( xnRoot.ChildNodes.Count - 1);
79
80             Debug.Log( strJoint.ToString() + ": " + xnRoot.ChildNodes.Count.ToString() );
81             break;
82         case "HAnimSegment":
83             m_curParseSegment.SetSegment( Parse_AttributeValue( xnRoot, "name" ) );
84             break;
85         case "Transform":
86             Vector3 vec = Parse_Vector3( Parse_AttributeValue( xnRoot, "translation" ) );
87             m_curParseSegment.SetTranslation( vec );
88     }

```

Draw an X3D File

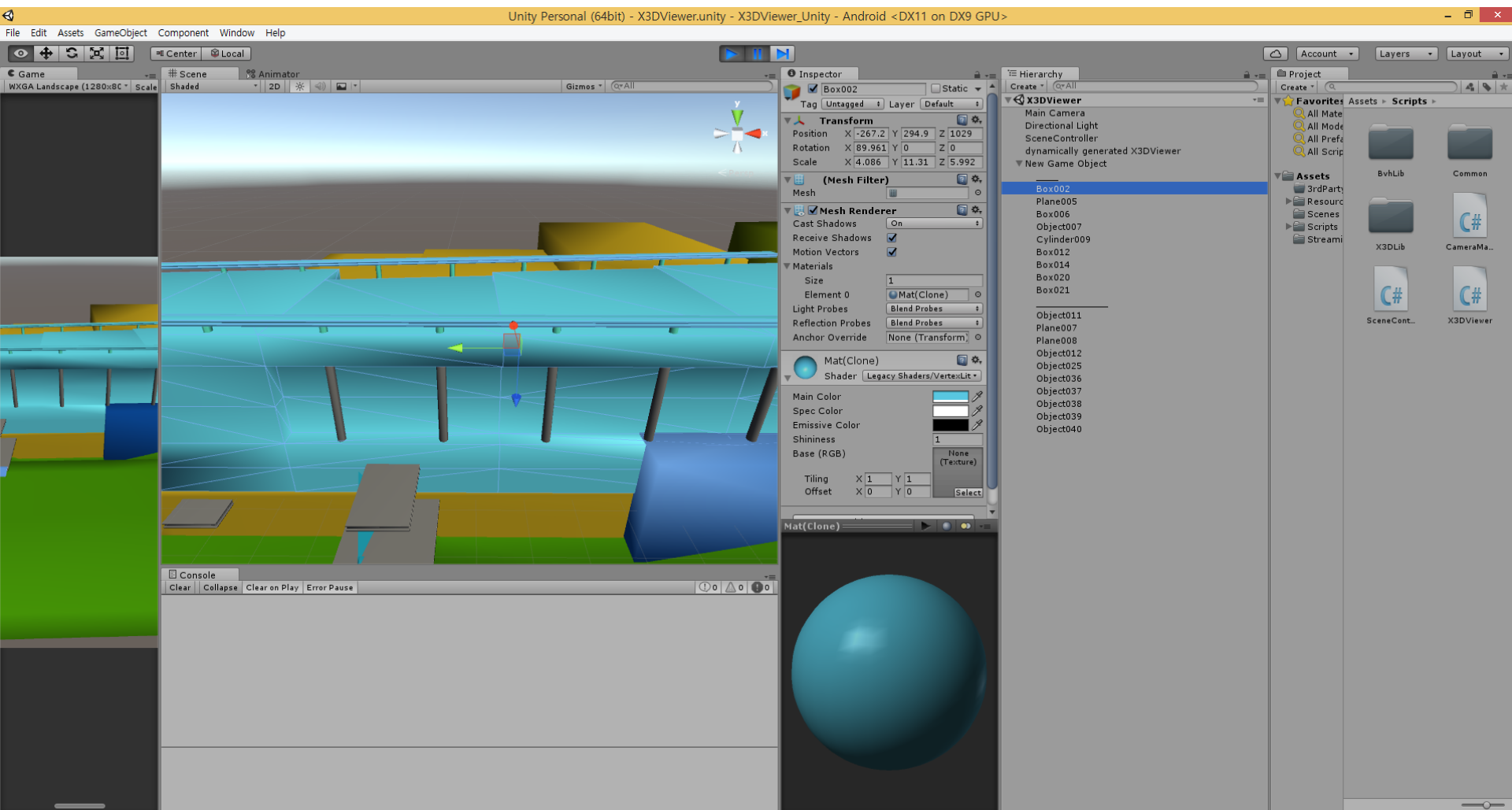


```
X3DViewer_Unity - Microsoft Visual Studio
빠른 실행(Ctrl+Q)
파일(F) 편집(E) 보기(V) 프로젝트(P) 빌드(B) 디버그(D) 팀(M) 도구(T) 테스트(S) 분석(N) 창(W) 로그인
Debug Any CPU Unity에 연결
X3DLib.cs X3DViewer.cs UtilityClass.cs X3Dx3d.cs X3DNode.cs KoreanCharacter01Jin.X3D
X3DViewer_Unity X3Dx3d Draw()
72 public override void Draw()
73 {
74     if (m_listSegment.Count == 0)
75         return;
76
77     m_goRootParent = new GameObject();
78
79     //X3DViewer.Instance.m_listDebug.Add("segCount: " + m_listSegment.Count.ToString())
80
81     for (int i = 0; i < m_listSegment.Count; ++i)
82     {
83         GameObject cGo = new GameObject();
84         m_listSegment[i].SetGoSegment(cGo);
85         cGo.name = m_listSegment[i].GetSegment();
86
87         cGo.transform.parent = m_goRootParent.transform;
88         cGo.transform.position = m_listSegment[i].GetPosition();
89
90         if(m_listSegment[i].GetRotation().w != 0f)
91         {
92             Vector3 vecRot = new Vector3(m_listSegment[i].GetRotation().x * 180f + m_listSegment[i].GetRotation().y * 180f, m_listSegment[i].GetRotation().z * 180f, 0f);
93             cGo.transform.localRotation = Quaternion.Euler(vecRot);
94         }
95     }
96 }
```

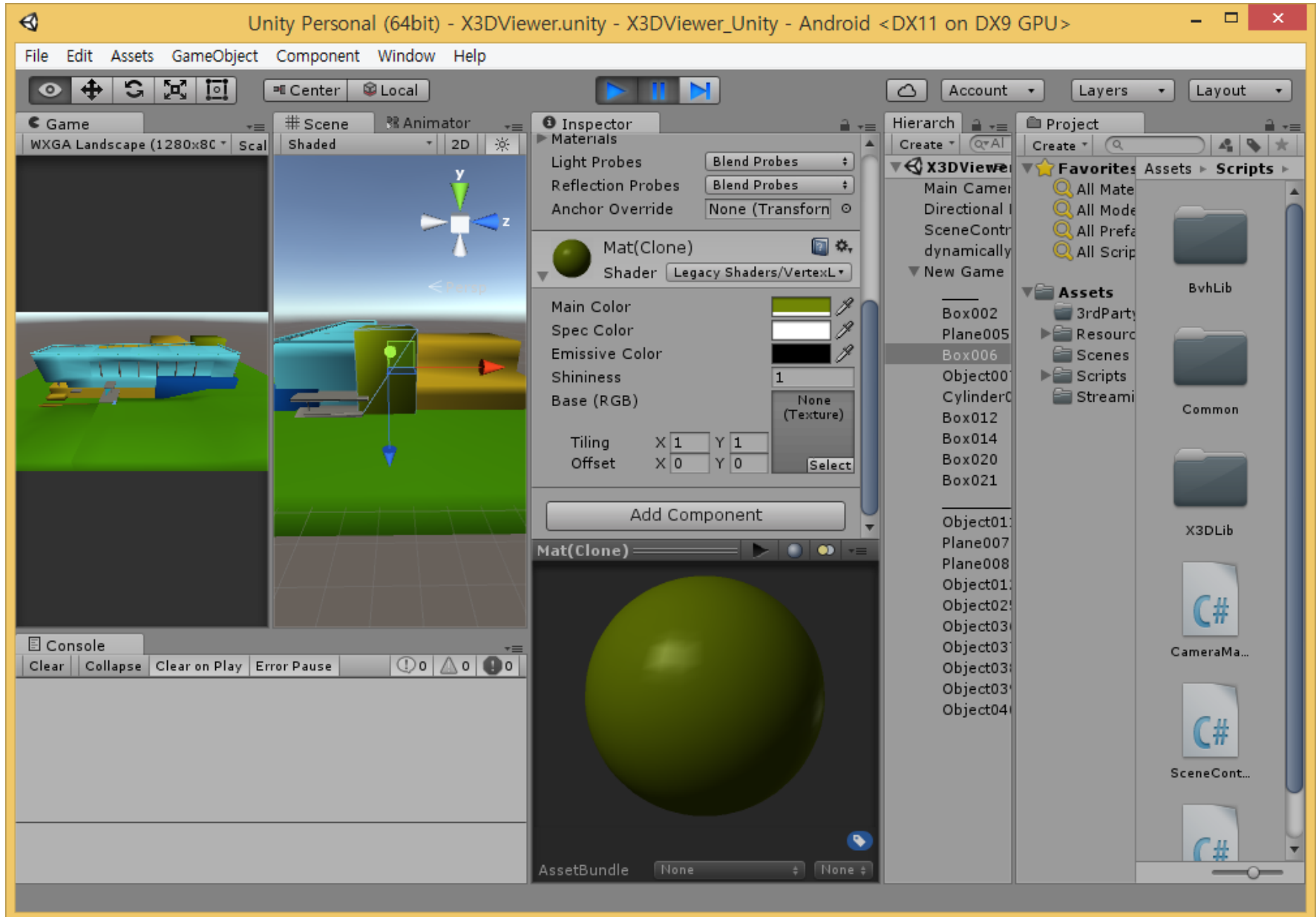
오류 목록... 출력 찾기 결과 1 기호 찾기 결과

줄: 92 열: 20 문자: 20 INS 게시

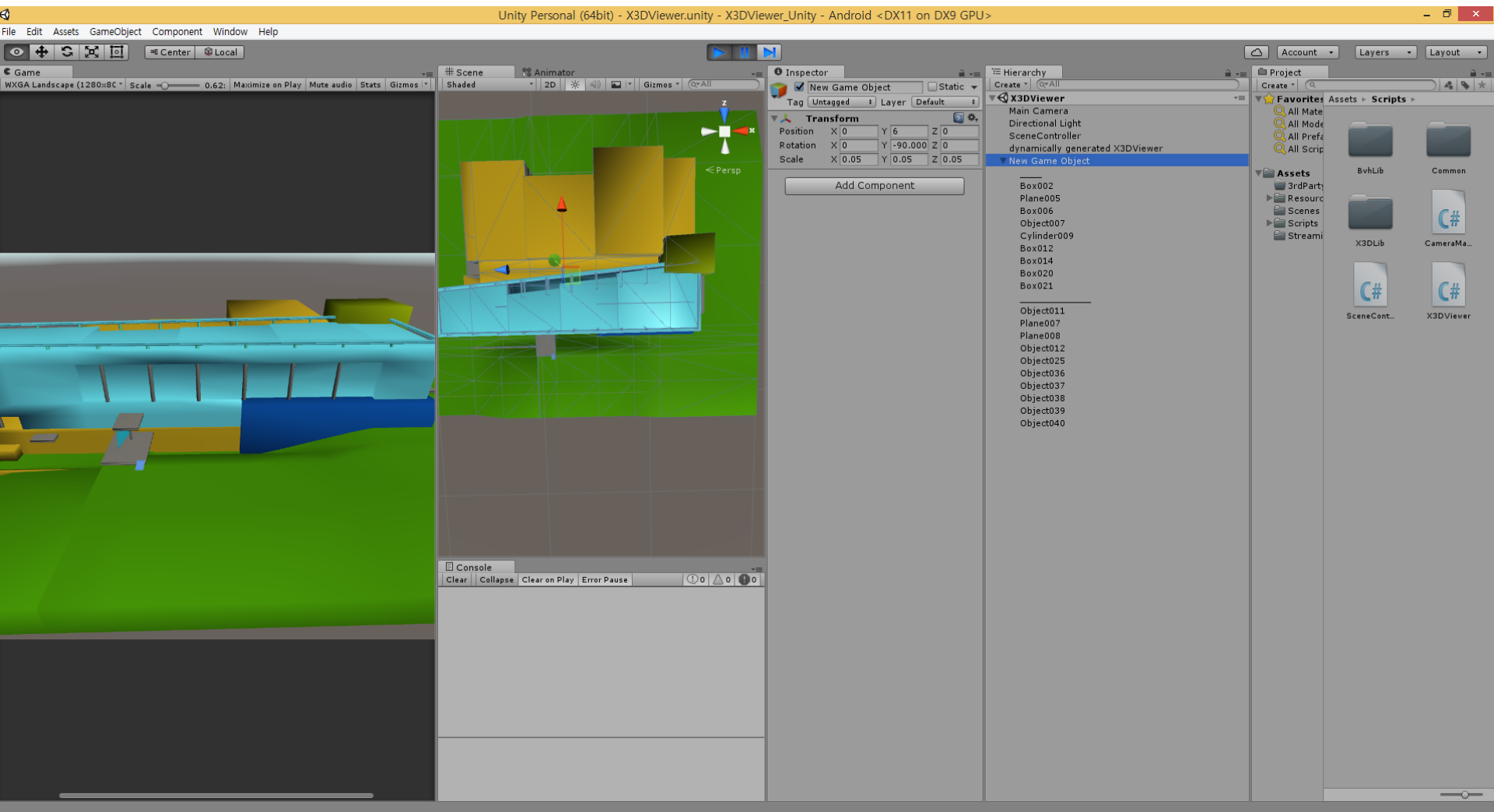
Unity X3D Scene Generation



Unity X3D Scene Generation



Unity X3D Scene Generation



Unity X3D Texture Mapping

```
public IEnumerator CoroutineLoadImage()  
{  
    string strFileName = m_listSegment[0].GetTextureName();  
  
    WWW www = UtilityClass.LoadFile(strFileName);  
  
    yield return www;  
  
    m_texImage = www.texture;  
}
```

Unity X3D Texture Mapping

```
public override void Draw()
{
    if (m_listSegment.Count == 0)
        return;

    m_goRootParent = new GameObject();
    m_goRootParent.name = m_strFileName;
    List<GameObject> listParent = new List<GameObject>();

    //X3DViewer.Instance.m_listDebug.Add("segCount: "+ m_listSegment.Count.ToString());

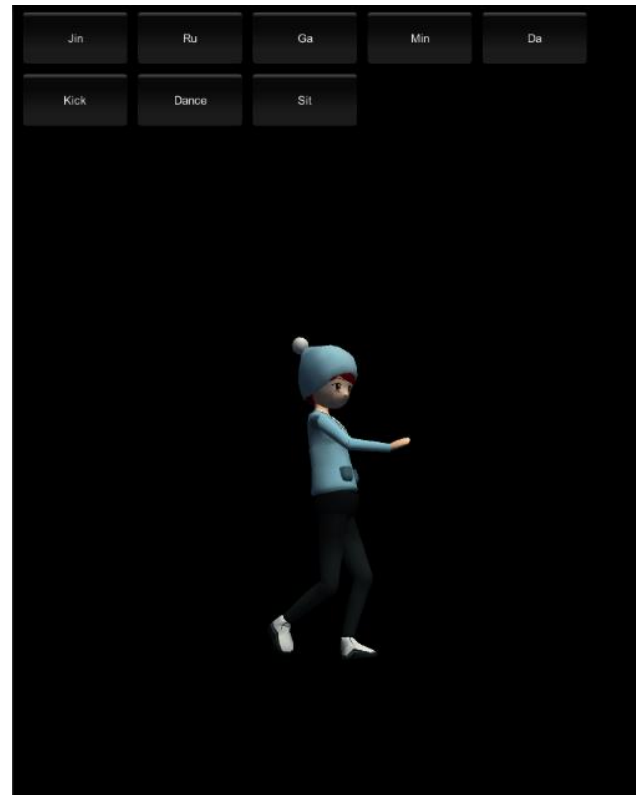
    for (int i = 0; i < m_listSegment.Count; ++i)
    {
        GameObject cGo = new GameObject();
        m_listSegment[i].SetGoSegment(cGo);
        cGo.name = m_listSegment[i].GetSegment();

        if (listParent.Count > 0)
        {
            cGo.transform.parent = listParent[listParent.Count - 1].transform;
            listParent.RemoveAt(listParent.Count - 1);
        }
        else
        {
            cGo.transform.parent = m_goRootParent.transform;
        }

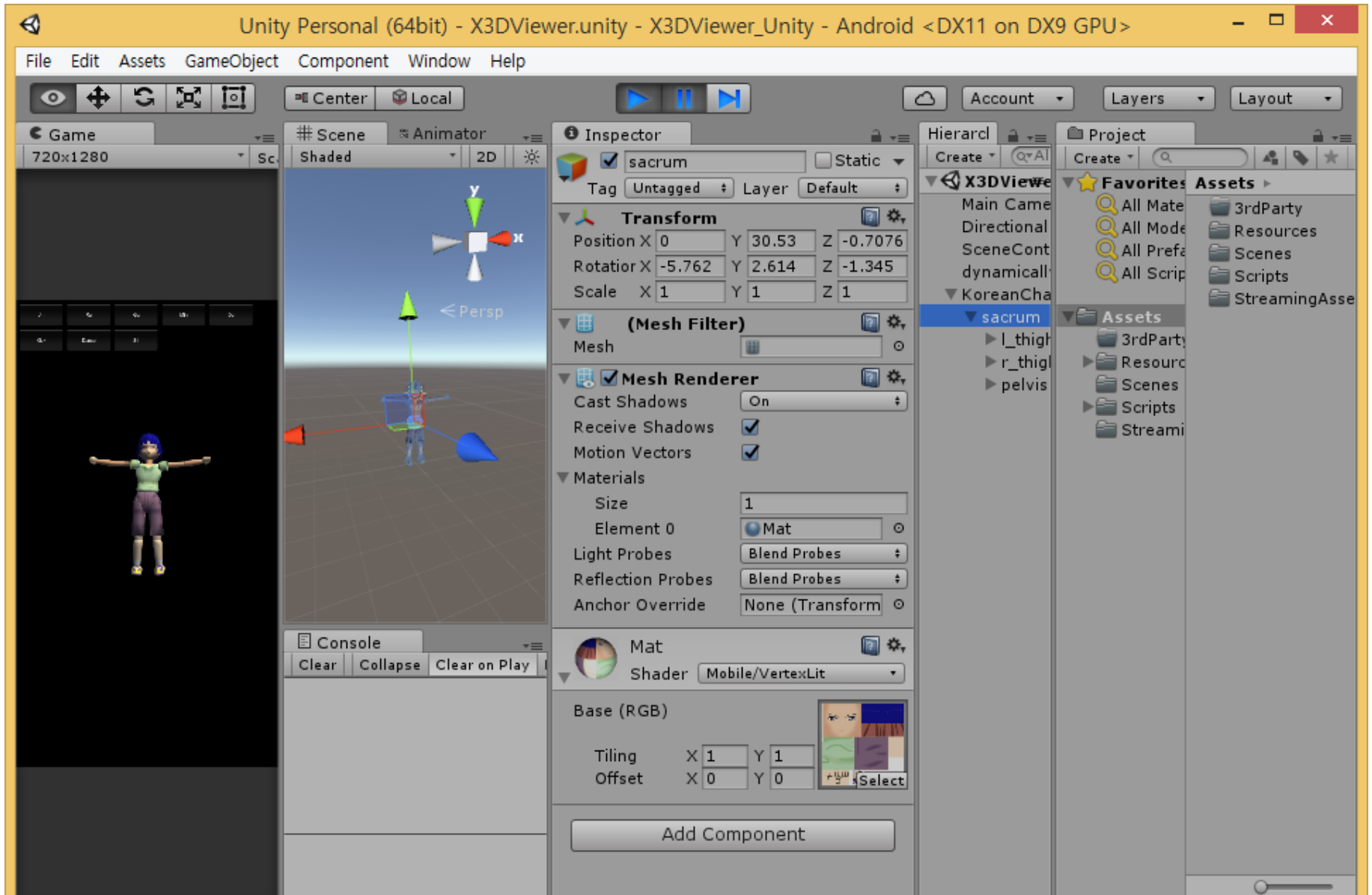
        cGo.transform.position = m_listSegment[i].GetTranslation();
        cGo.transform.localRotation = Quaternion.Euler(m_listSegment[i].GetRotation());
        //cGo.transform.lossyScale = m_listSegment[i].GetScale();

        for (int j = 0; j < m_listSegment[i].GetChildCount(); ++j)
        {
            listParent.Add(cGo);
        }
    }
}
```

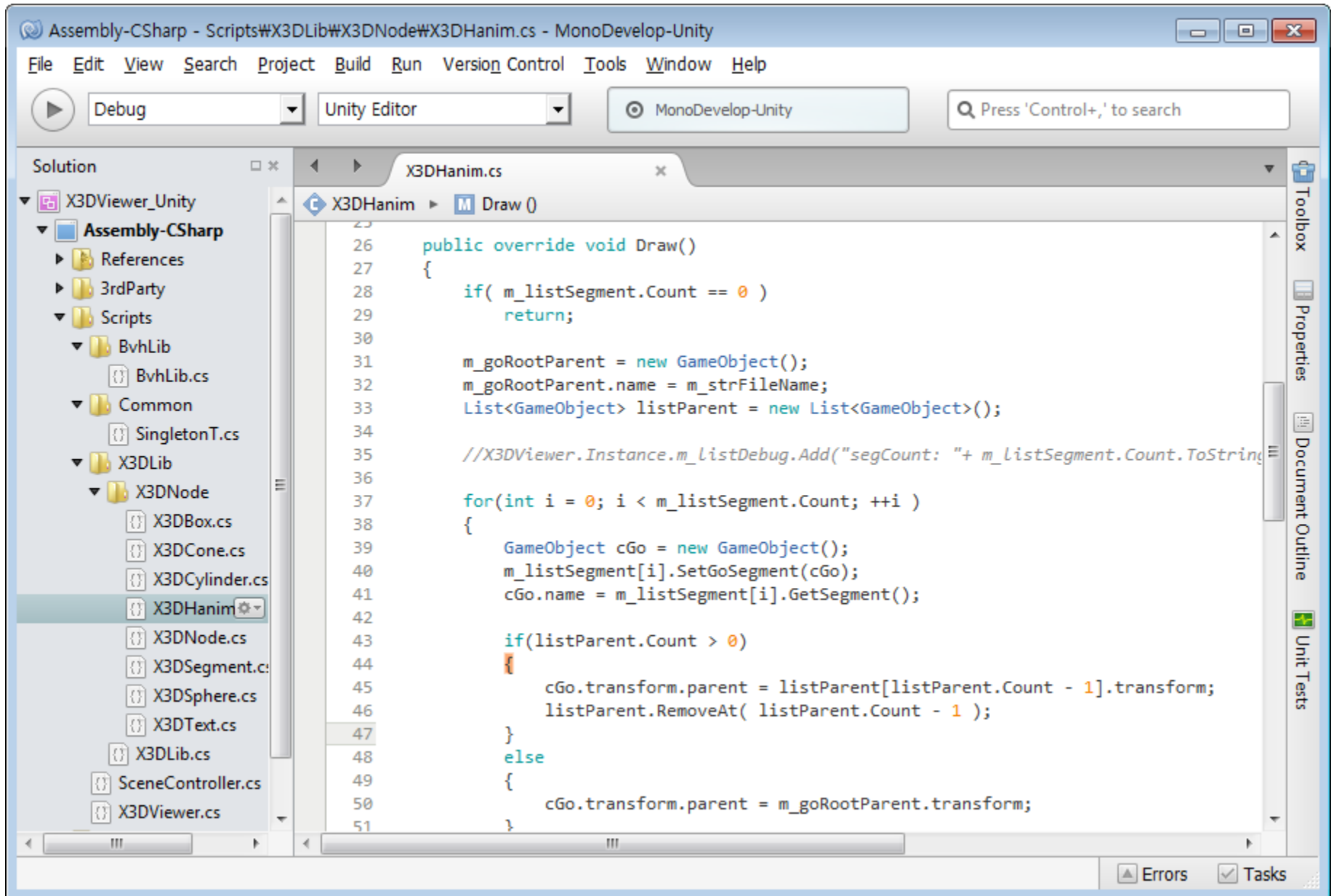
Results of Unity X3D Texture Mapping



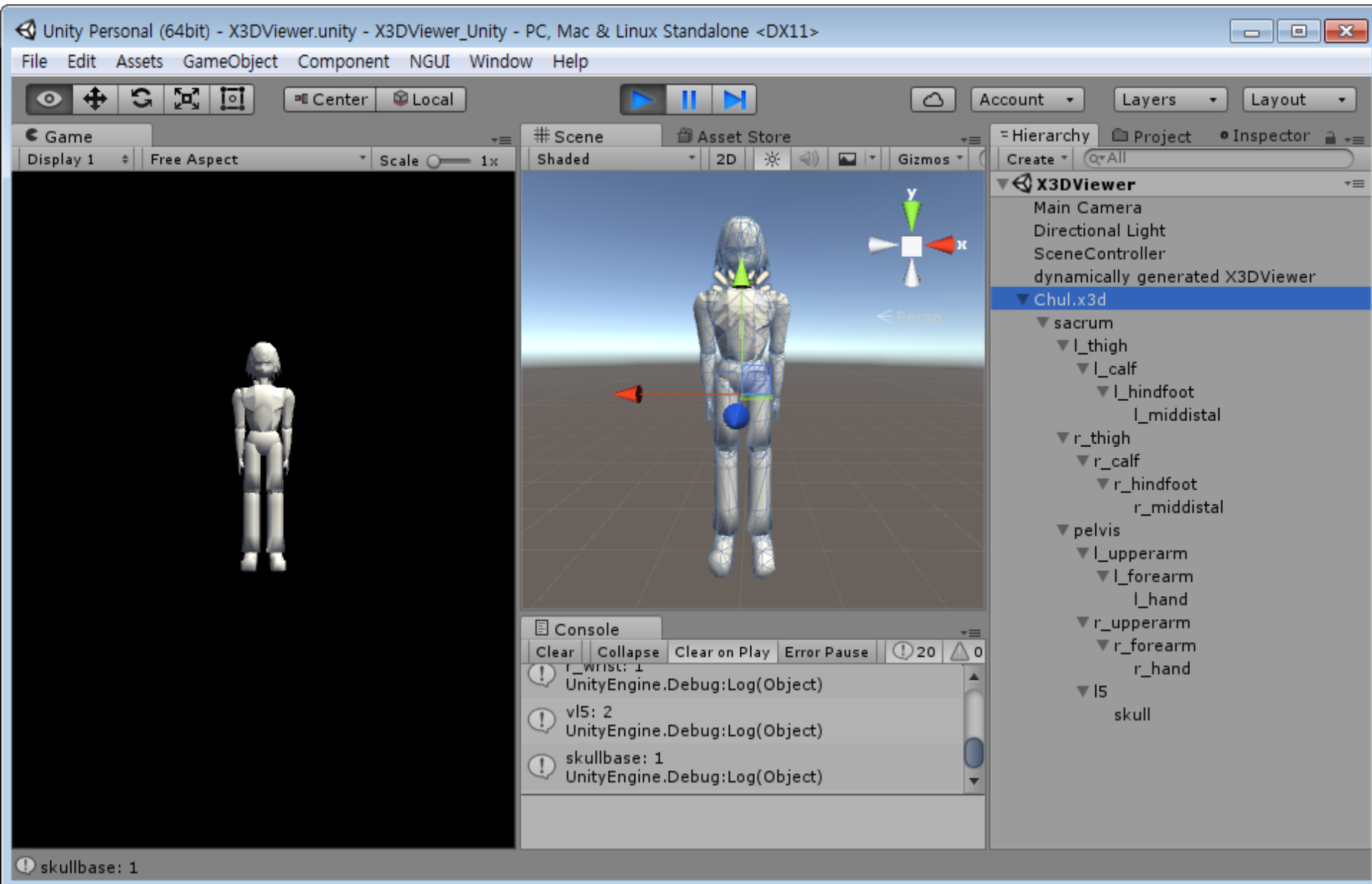
Results of Unity X3D Texture Mapping



Draw an H-Anim Character



Load the H-Anim Character



Load BVH Mocap Data

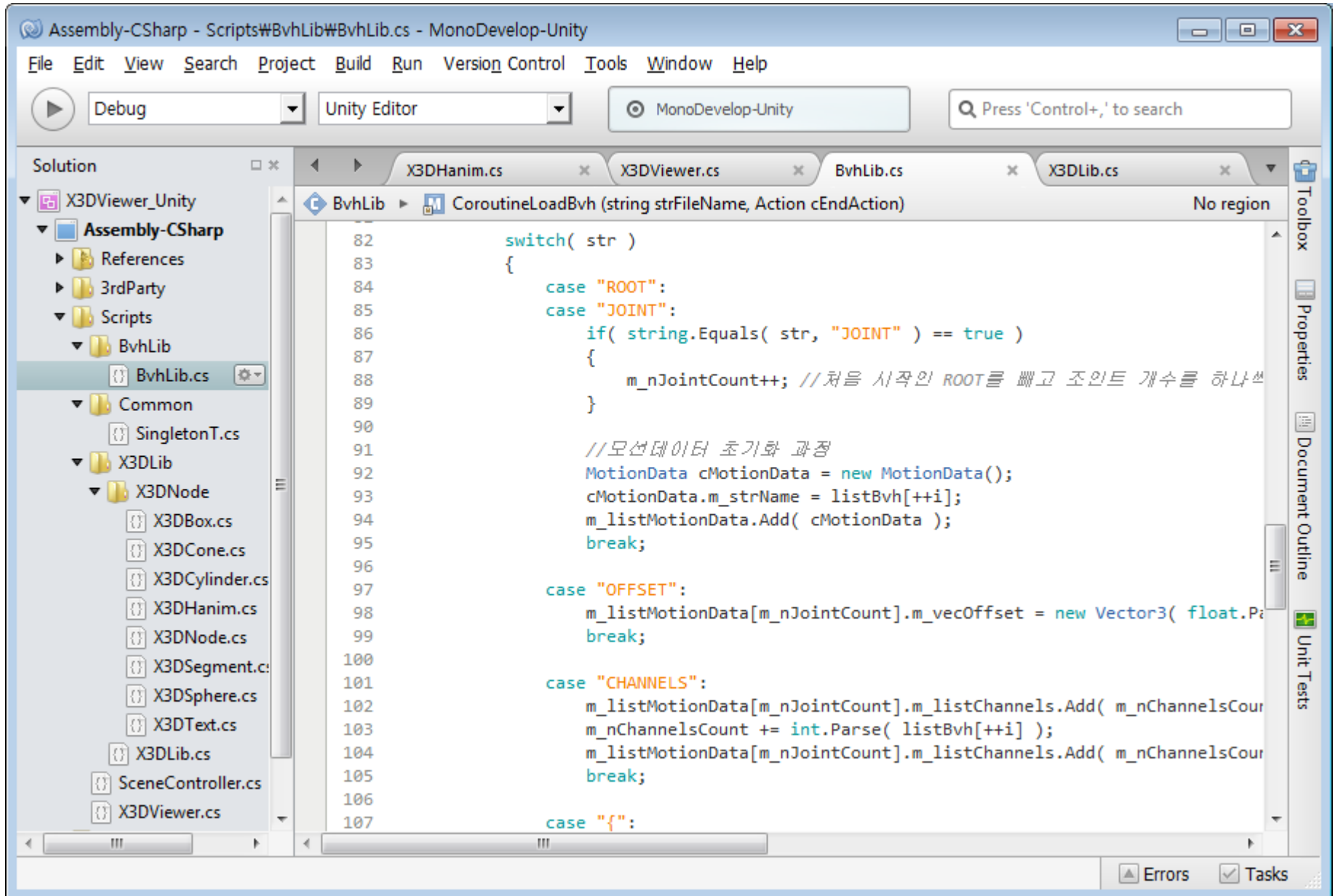
The image shows a screenshot of the MonoDevelop-Unity IDE. The main window displays the C# source code for `BvhLib.cs`. The code defines a `LoadBvh` method and an `IEnumerator` `CoroutineLoadBvh` for loading BVH data. The code is as follows:

```
38 public void LoadBvh( string strFileName, Action cEndAction )
39 {
40     X3DViewer.Instance.StartCoroutine( CoroutineLoadBvh( strFileName, cEndAction ) )
41 }
42
43 IEnumerator CoroutineLoadBvh( string strFileName, Action cEndAction )
44 {
45     Debug.Log( strFileName );
46     m_strFileName = strFileName;
47
48     string strPath = string.Empty;
49     // 플랫폼 별 로드 |
50 #if ( UNITY_EDITOR || UNITY_STANDALONE_WIN )
51     strPath += ("file:///");
52     strPath += (Application.streamingAssetsPath + "/" + strFileName);
53 #elif UNITY_ANDROID
54     strPath = "jar:file://" + Application.dataPath + "!/assets/" + strFileName;
55 #endif
56
57     WWW www = new WWW( strPath );
58
59     yield return www;
60
61     ParseBvhRecursive( www.text );
62
63     cEndAction();
```

The Solution Explorer on the left shows the project structure:

- X3DViewer_Unity
 - Assembly-CSharp
 - References
 - 3rdParty
 - Scripts
 - BvhLib
 - BvhLib.cs
 - Common
 - SingletonT.cs
 - X3DLib
 - X3DNode
 - X3DBox.cs
 - X3DCone.cs
 - X3DCylinder.cs
 - X3DHanim.cs
 - X3DNode.cs
 - X3DSegment.cs
 - X3DSphere.cs
 - X3DText.cs
 - X3DLib.cs
 - SceneController.cs
 - X3DViewer.cs

Parse BVH Mocap Data

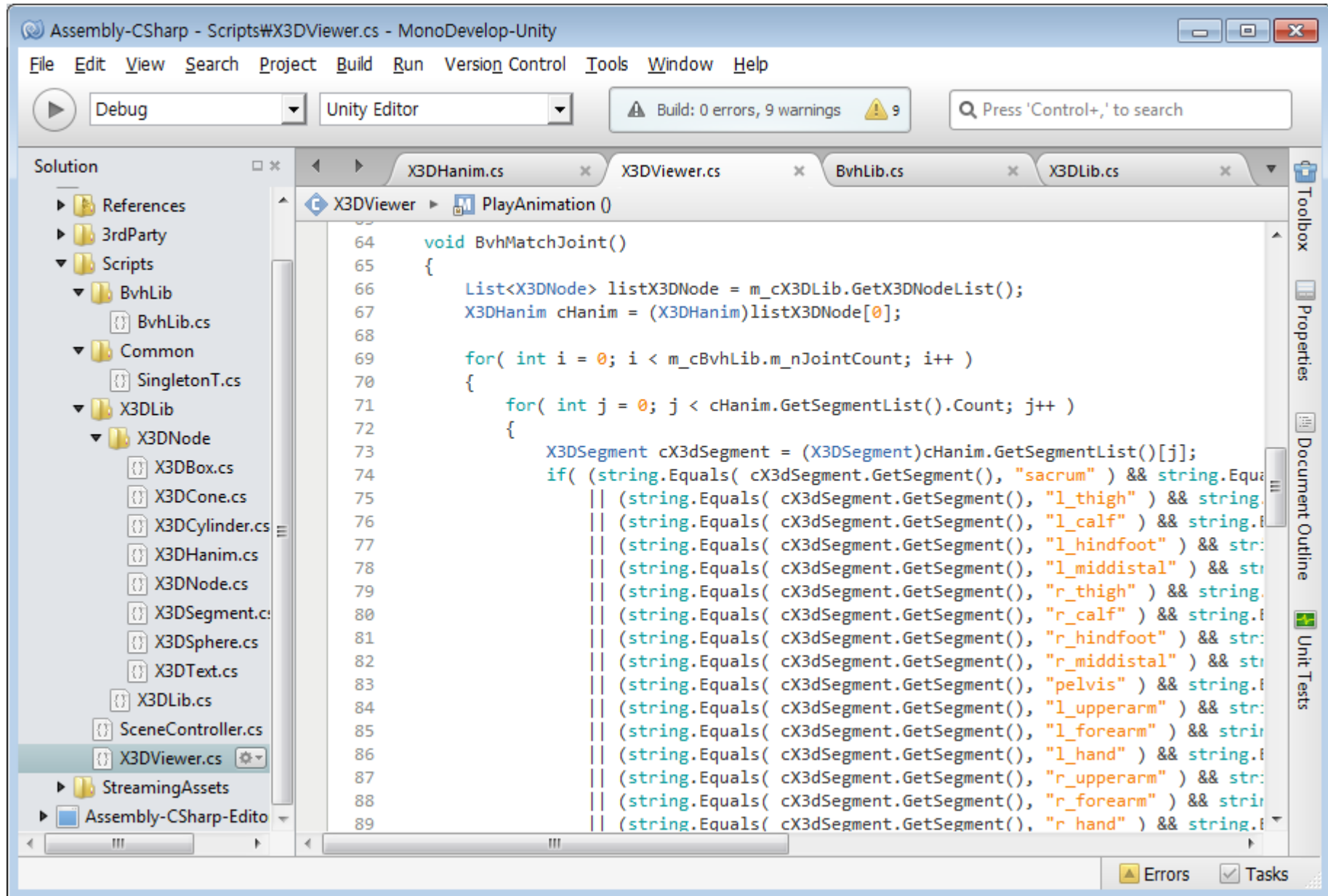


The image shows a screenshot of the MonoDevelop-Unity IDE. The main window displays a C# script named `BvhLib.cs` with the following code:

```
82         switch( str )
83         {
84             case "ROOT":
85             case "JOINT":
86                 if( string.Equals( str, "JOINT" ) == true )
87                 {
88                     m_nJointCount++; //처음 시작인 ROOT를 빼고 조인트 개수를 하나씩
89                 }
90
91                 //모션데이터 초기화 과정
92                 MotionData cMotionData = new MotionData();
93                 cMotionData.m_strName = listBvh[++i];
94                 m_listMotionData.Add( cMotionData );
95                 break;
96
97             case "OFFSET":
98                 m_listMotionData[m_nJointCount].m_vecOffset = new Vector3( float.Parse( listBvh[++i] ),
99                                     float.Parse( listBvh[++i] ),
100                                    float.Parse( listBvh[++i] ) );
101                 break;
102
103             case "CHANNELS":
104                 m_listMotionData[m_nJointCount].m_listChannels.Add( m_nChannelsCount );
105                 m_nChannelsCount += int.Parse( listBvh[++i] );
106                 m_listMotionData[m_nJointCount].m_listChannels.Add( m_nChannelsCount );
107                 break;
108
109             case "{":
110                 m_nChannelsCount = 0;
111                 break;
112         }
```

The IDE interface includes a menu bar (File, Edit, View, Search, Project, Build, Run, Version Control, Tools, Window, Help), a toolbar with a play button and a dropdown menu set to 'Debug', and a search bar. The Solution Explorer on the left shows a project named 'X3DViewer_Unity' with folders for 'References', '3rdParty', and 'Scripts'. The 'Scripts' folder contains a sub-folder 'BvhLib' with the file 'BvhLib.cs' selected. The right sidebar contains panels for 'Toolbox', 'Properties', 'Document Outline', and 'Unit Tests'. The status bar at the bottom shows 'Errors' and 'Tasks'.

Joint Mapping of BVH to H-Anim



The image shows a screenshot of the MonoDevelop-Unity IDE. The title bar reads "Assembly-CSharp - Scripts#X3DViewer.cs - MonoDevelop-Unity". The menu bar includes "File", "Edit", "View", "Search", "Project", "Build", "Run", "Version Control", "Tools", "Window", and "Help". The toolbar shows a play button, a dropdown menu set to "Debug", another dropdown menu set to "Unity Editor", a build status indicator showing "Build: 0 errors, 9 warnings", and a search bar with the text "Press 'Control+', to search".

The Solution Explorer on the left shows a project structure with folders "References", "3rdParty", "Scripts", "BvhLib", "Common", "X3DLib", and "StreamingAssets". Under "X3DLib", there is a sub-folder "X3DNode" containing several .cs files, with "X3DViewer.cs" selected.

The main editor window displays the code for the `BvhMatchJoint()` method in `X3DViewer.cs`. The code is as follows:

```
64 void BvhMatchJoint()
65 {
66     List<X3DNode> listX3DNode = m_cX3DLib.GetX3DNodeList();
67     X3DHanim cHanim = (X3DHanim)listX3DNode[0];
68
69     for( int i = 0; i < m_cBvhLib.m_nJointCount; i++ )
70     {
71         for( int j = 0; j < cHanim.GetSegmentList().Count; j++ )
72         {
73             X3DSegment cX3dSegment = (X3DSegment)cHanim.GetSegmentList()[j];
74             if( (string.Equals( cX3dSegment.GetSegment(), "sacrum" ) && string.Equals( cX3dSegment.GetSegment(), "l_thigh" ) && string.Equals( cX3dSegment.GetSegment(), "l_calf" ) && string.Equals( cX3dSegment.GetSegment(), "l_hindfoot" ) && string.Equals( cX3dSegment.GetSegment(), "l_middistal" ) && string.Equals( cX3dSegment.GetSegment(), "r_thigh" ) && string.Equals( cX3dSegment.GetSegment(), "r_calf" ) && string.Equals( cX3dSegment.GetSegment(), "r_hindfoot" ) && string.Equals( cX3dSegment.GetSegment(), "r_middistal" ) && string.Equals( cX3dSegment.GetSegment(), "pelvis" ) && string.Equals( cX3dSegment.GetSegment(), "l_upperarm" ) && string.Equals( cX3dSegment.GetSegment(), "l_forearm" ) && string.Equals( cX3dSegment.GetSegment(), "l_hand" ) && string.Equals( cX3dSegment.GetSegment(), "r_upperarm" ) && string.Equals( cX3dSegment.GetSegment(), "r_forearm" ) && string.Equals( cX3dSegment.GetSegment(), "r hand" ) && string.Equals(
```

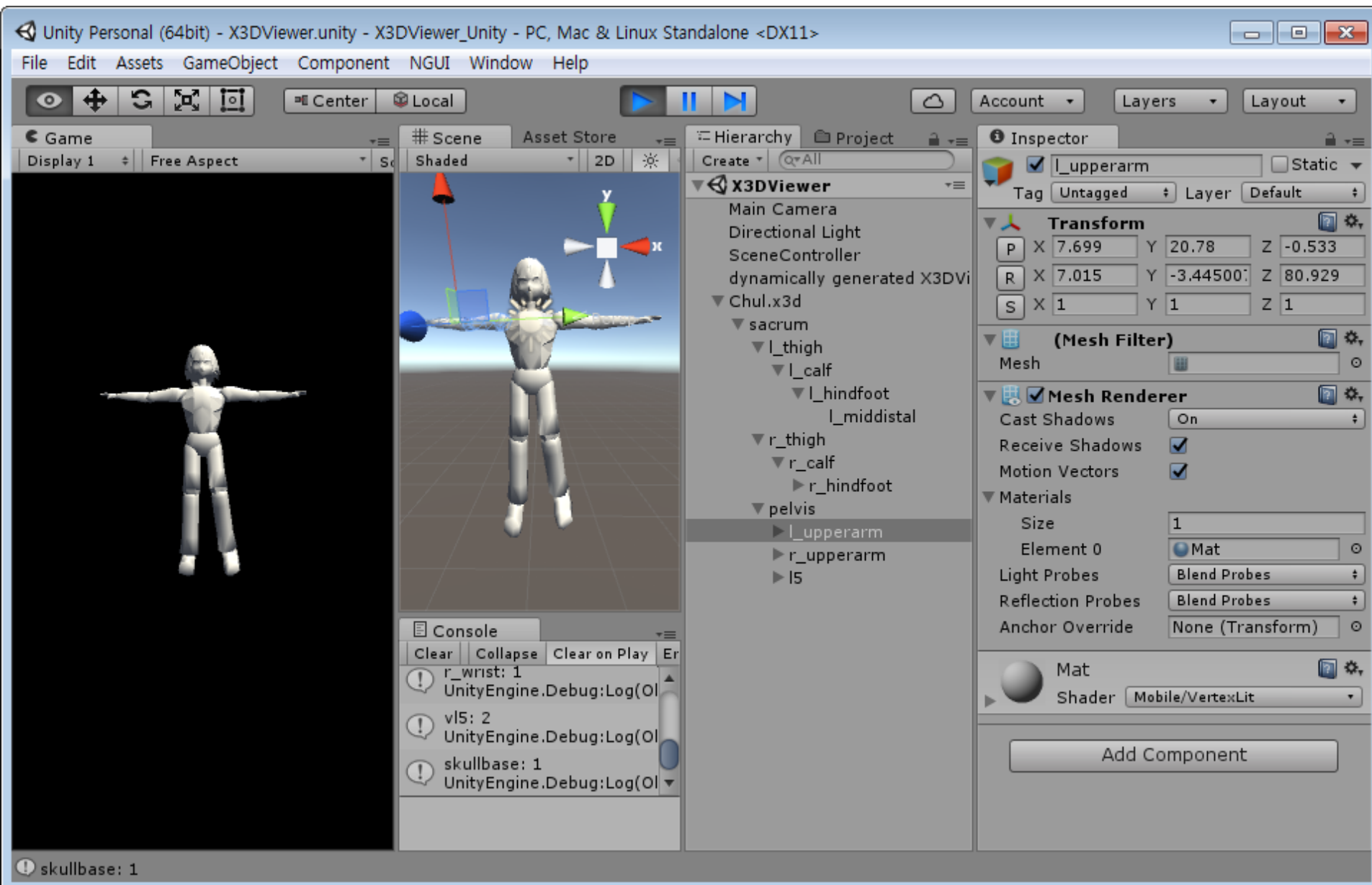
BVH Mocap Animation

The image shows a screenshot of the MonoDevelop-Unity IDE. The window title is "Assembly-CSharp - Scripts#X3DViewer.cs* - MonoDevelop-Unity". The menu bar includes File, Edit, View, Search, Project, Build, Run, Version Control, Tools, Window, and Help. The toolbar shows a play button, a dropdown menu set to "Debug", and another dropdown menu set to "Unity Editor". The main editor area displays the code for the "PlayAnimation()" method in "X3DViewer.cs".

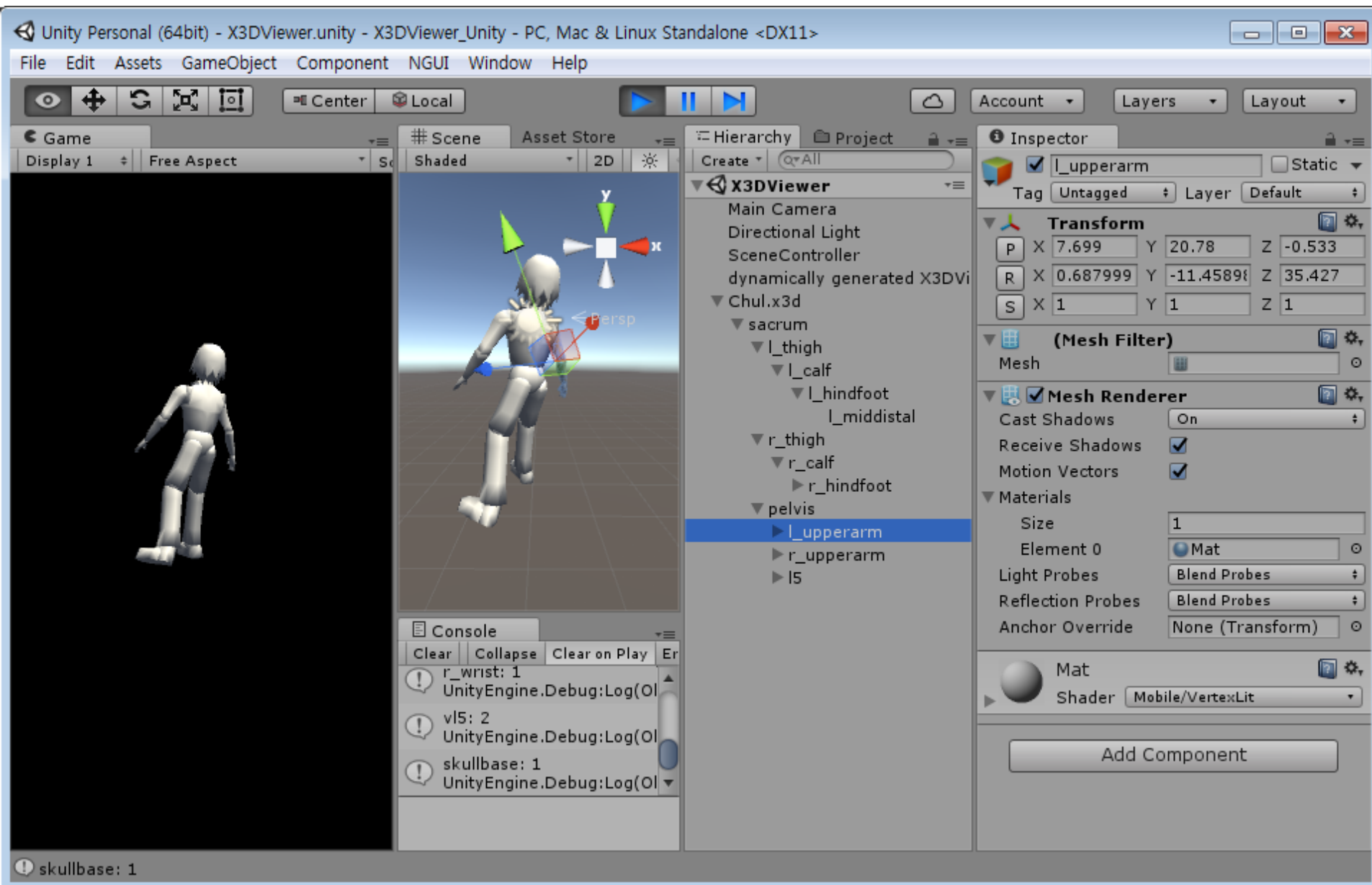
```
100 private void PlayAnimation()
101 {
102     m_nAnimationFrame++;
103     if( m_nAnimationFrame >= m_cBvhLib.m_cFramesData.m_nFrames )
104     {
105         m_nAnimationFrame = 0;
106     }
107     List<X3DNode> listX3DNode = m_cX3DLib.GetX3DNodeList();
108     X3DHanim cHanim = (X3DHanim)listX3DNode[0];
109
110     for( int i = 0; i < m_cBvhLib.m_nJointCount; ++ i )
111     {
112         BvhLib.MotionData cMotionData = m_cBvhLib.m_listMotionData[i];
113         BvhLib.FramesData cFrameData = m_cBvhLib.m_cFramesData;
114         int nMatchHanim = m_cBvhLib.m_listMotionData[i].m_nMatchHanim;
115         List<X3DSegment> listSegment = cHanim.GetSegmentList();
116
117         if(listSegment.Count <= nMatchHanim)
118         {
119             continue;
120         }
121
122         X3DSegment cX3dSegment = listSegment[nMatchHanim];
123         if( cMotionData.m_listChannels[1] - cMotionData.m_listChannels[0] == 5 )
124         {
125             float fX = cFrameData.m_listFramesRotation[m_nAnimationFrame][cMotionD
```

The Solution Explorer on the left shows a project structure with folders for References, 3rdParty, Scripts, BvhLib, Common, X3DLib, and StreamingAssets. The X3DLib folder contains several .cs files, including X3DViewer.cs which is currently selected. The right sidebar contains the Toolbox, Properties, Document Outline, and Unit Tests panels.

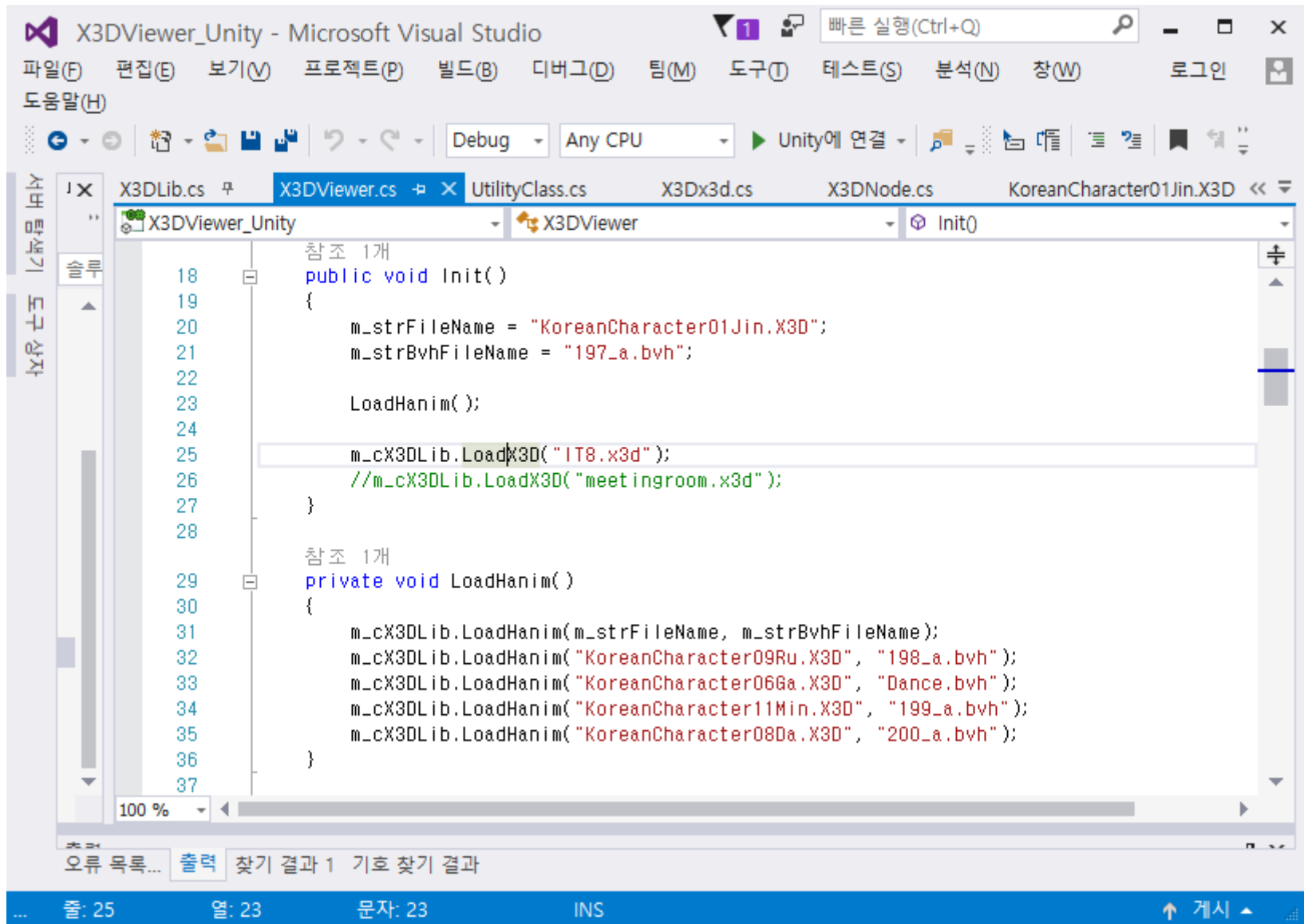
Display Unity BVH Mocap Animation (1)



Display Unity BVH Mocap Animation (2)



Load Multiple H-Anim Characters

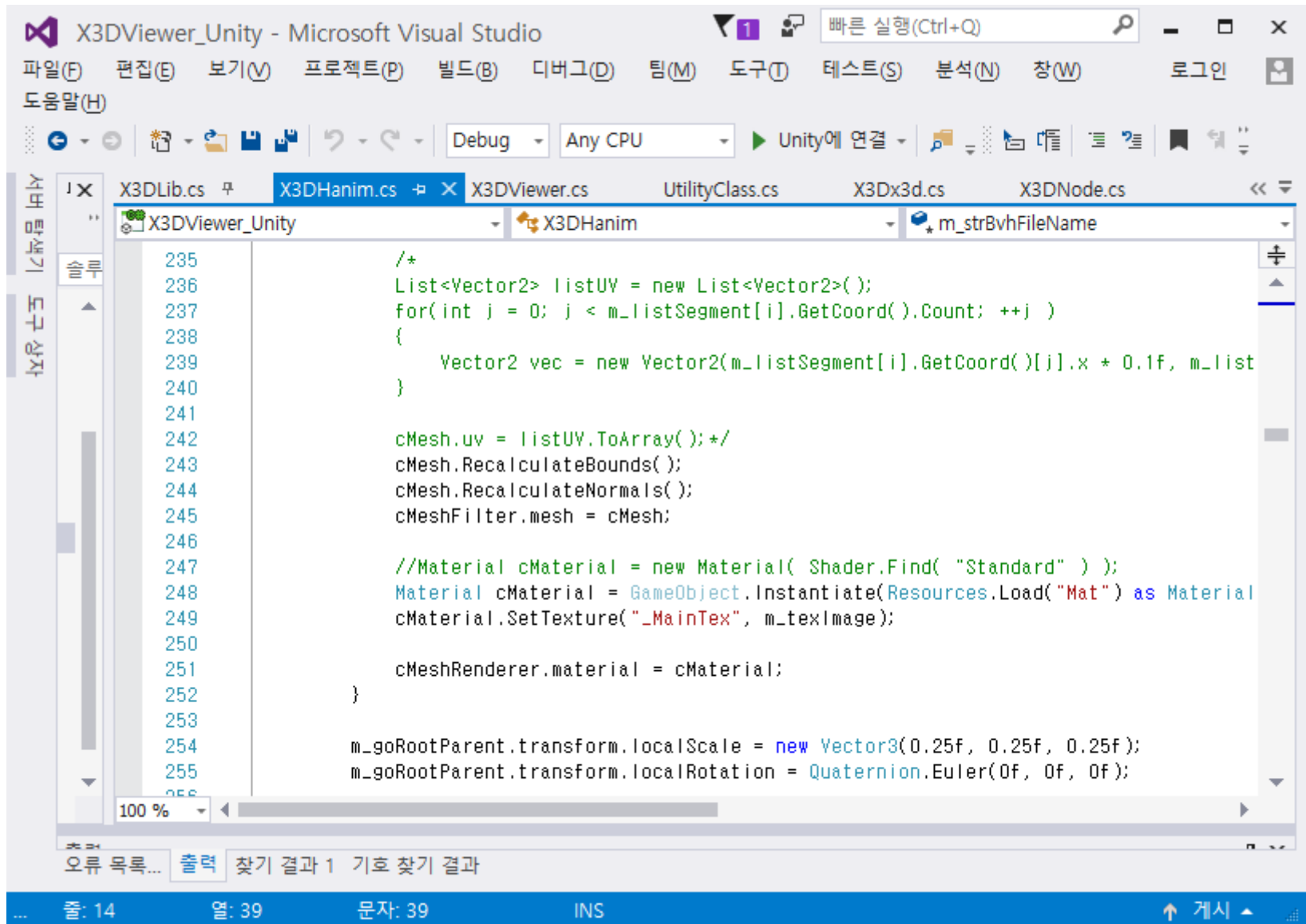


The screenshot shows the Visual Studio IDE with the following details:

- Window Title: X3DViewer_Unity - Microsoft Visual Studio
- Menu Bar: 파일(F), 편집(E), 보기(V), 프로젝트(P), 빌드(B), 디버그(D), 팀(M), 도구(T), 테스트(S), 분석(N), 창(W), 로그인
- Toolbar: Includes icons for file operations and a dropdown menu showing 'Debug' and 'Any CPU'. A button labeled 'Unity에 연결' is also visible.
- File Explorer: Shows the project structure with 'X3DViewer' selected under 'X3DViewer_Unity'. The file 'X3DViewer.cs' is open.
- Code Editor: Displays the C# code for the 'Init()' and 'LoadHanim()' methods. The 'Init()' method calls 'LoadX3D()' for 'IT8.x3d' and 'meetingroom.x3d'. The 'LoadHanim()' method calls 'LoadHanim()' for five different character files.

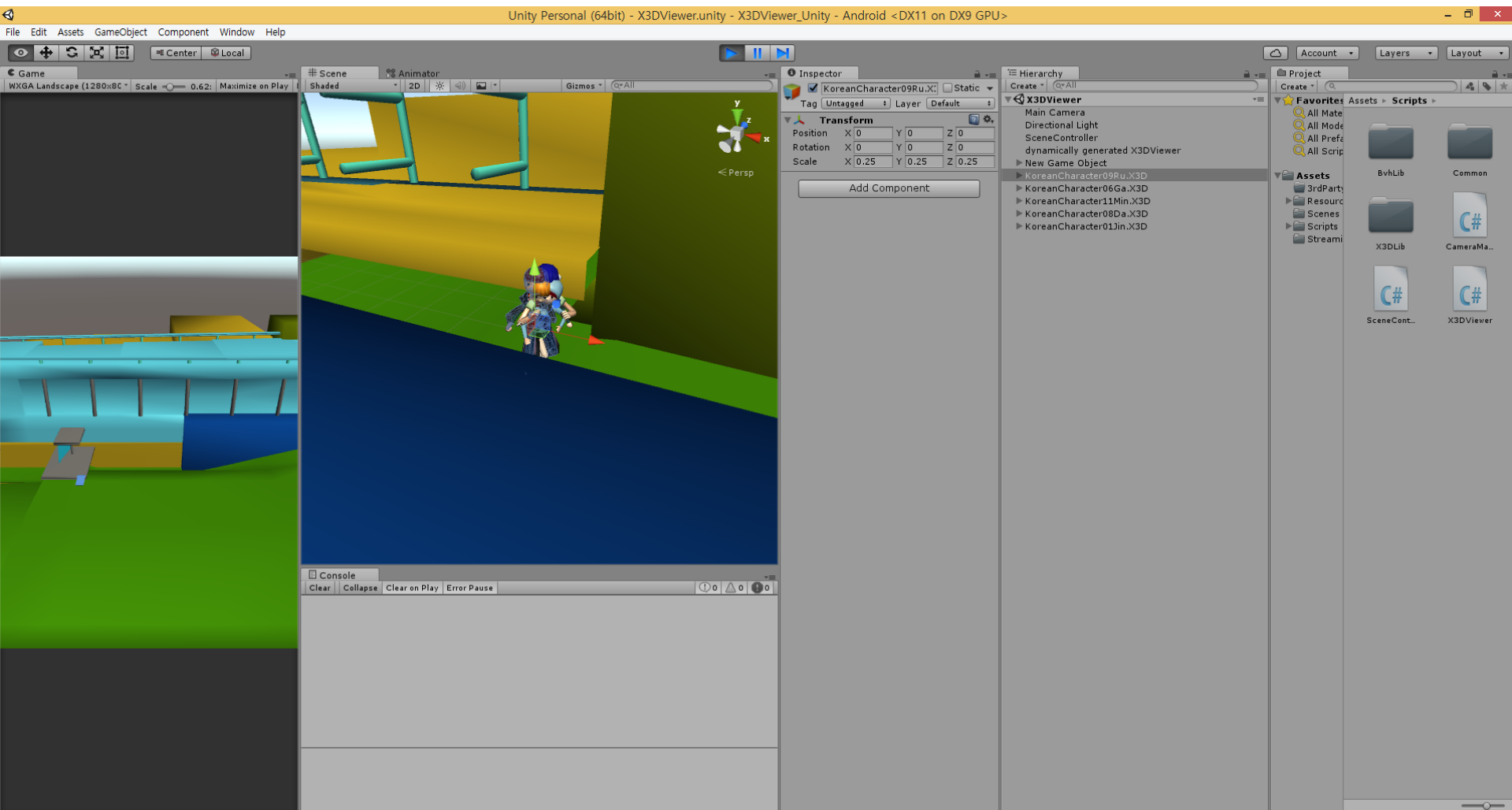
```
18     참조 1개
19     public void Init()
20     {
21         m_strFileName = "KoreanCharacter01Jin.X3D";
22         m_strBvhFileName = "197_a.bvh";
23
24         LoadHanim();
25
26         m_cX3DLib.LoadX3D("IT8.x3d");
27         //m_cX3DLib.LoadX3D("meetingroom.x3d");
28     }
29     참조 1개
30     private void LoadHanim()
31     {
32         m_cX3DLib.LoadHanim(m_strFileName, m_strBvhFileName);
33         m_cX3DLib.LoadHanim("KoreanCharacter09Ru.X3D", "198_a.bvh");
34         m_cX3DLib.LoadHanim("KoreanCharacter06Ga.X3D", "Dance.bvh");
35         m_cX3DLib.LoadHanim("KoreanCharacter11Min.X3D", "199_a.bvh");
36         m_cX3DLib.LoadHanim("KoreanCharacter08Da.X3D", "200_a.bvh");
37     }
```


Transform Texture and Size

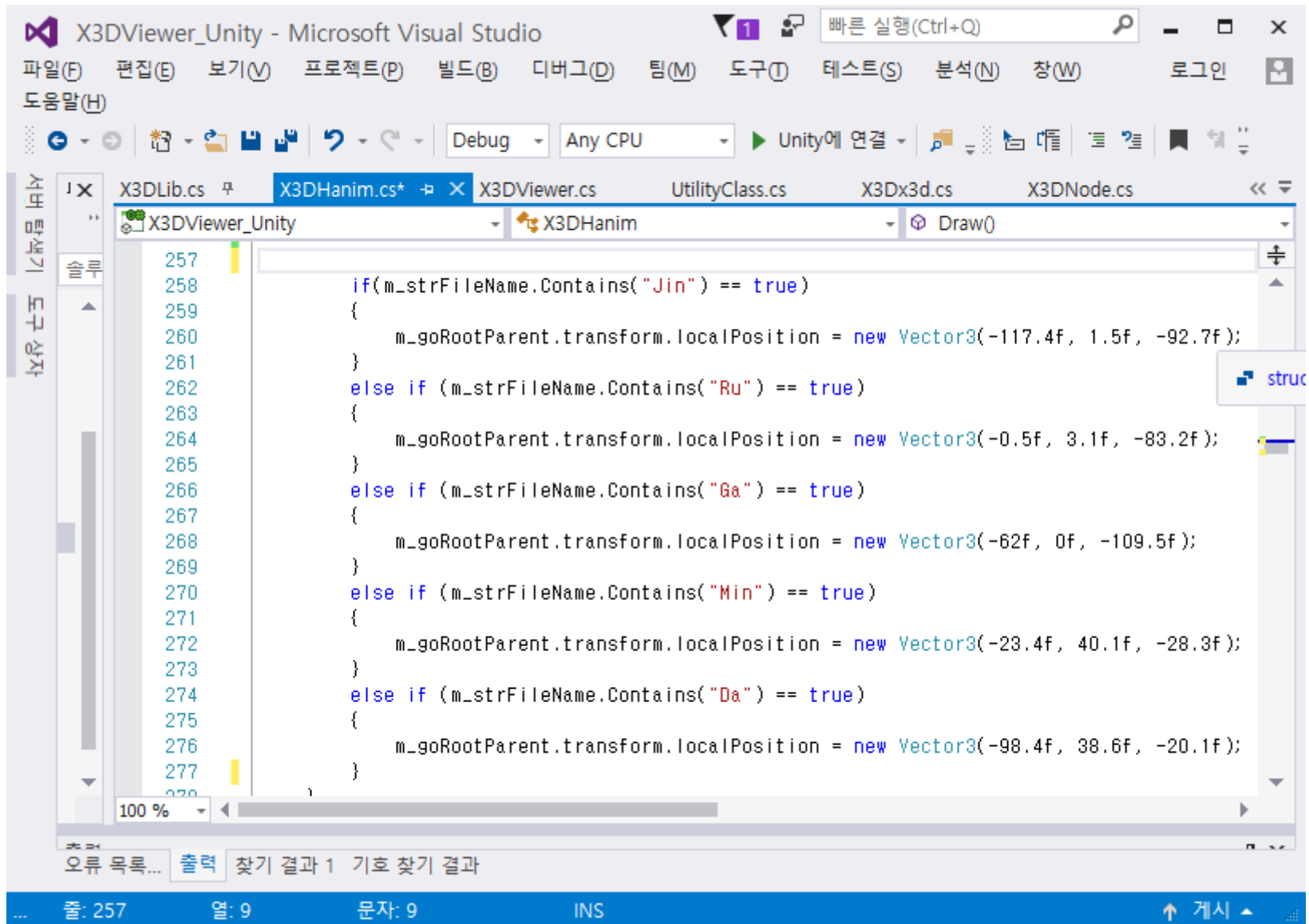


```
X3DViewer_Unity - Microsoft Visual Studio
빠른 실행(Ctrl+Q)
파일(F) 편집(E) 보기(V) 프로젝트(P) 빌드(B) 디버그(D) 팀(M) 도구(T) 테스트(S) 분석(N) 창(W) 로그인
도움말(H)
Debug Any CPU Unity에 연결
X3DLib.cs X3DHanim.cs X3DViewer.cs UtilityClass.cs X3Dx3d.cs X3DNode.cs
X3DViewer_Unity X3DHanim m_strBvhFileName
235 /*
236 List<Vector2> listUV = new List<Vector2>( );
237 for(int j = 0; j < m_listSegment[i].GetCoord().Count; ++j )
238 {
239     Vector2 vec = new Vector2(m_listSegment[i].GetCoord()[j].x + 0.1f, m_list
240 }
241
242 cMesh.uv = listUV.ToArray();*/
243 cMesh.RecalculateBounds( );
244 cMesh.RecalculateNormals( );
245 cMeshFilter.mesh = cMesh;
246
247 //Material cMaterial = new Material( Shader.Find( "Standard" ) );
248 Material cMaterial = GameObject.Instantiate(Resources.Load("Mat") as Material
249 cMaterial.SetTexture("_MainTex", m_texImage);
250
251 cMeshRenderer.material = cMaterial;
252 }
253
254 m_goRootParent.transform.localScale = new Vector3(0.25f, 0.25f, 0.25f);
255 m_goRootParent.transform.localRotation = Quaternion.Euler(0f, 0f, 0f);
256
100 %
오류 목록... 출력 찾기 결과 1 기호 찾기 결과
줄: 14 열: 39 문자: 39 INS 게시
```

Load Another H-Anim Character



Adjust H-Anim Coordinates

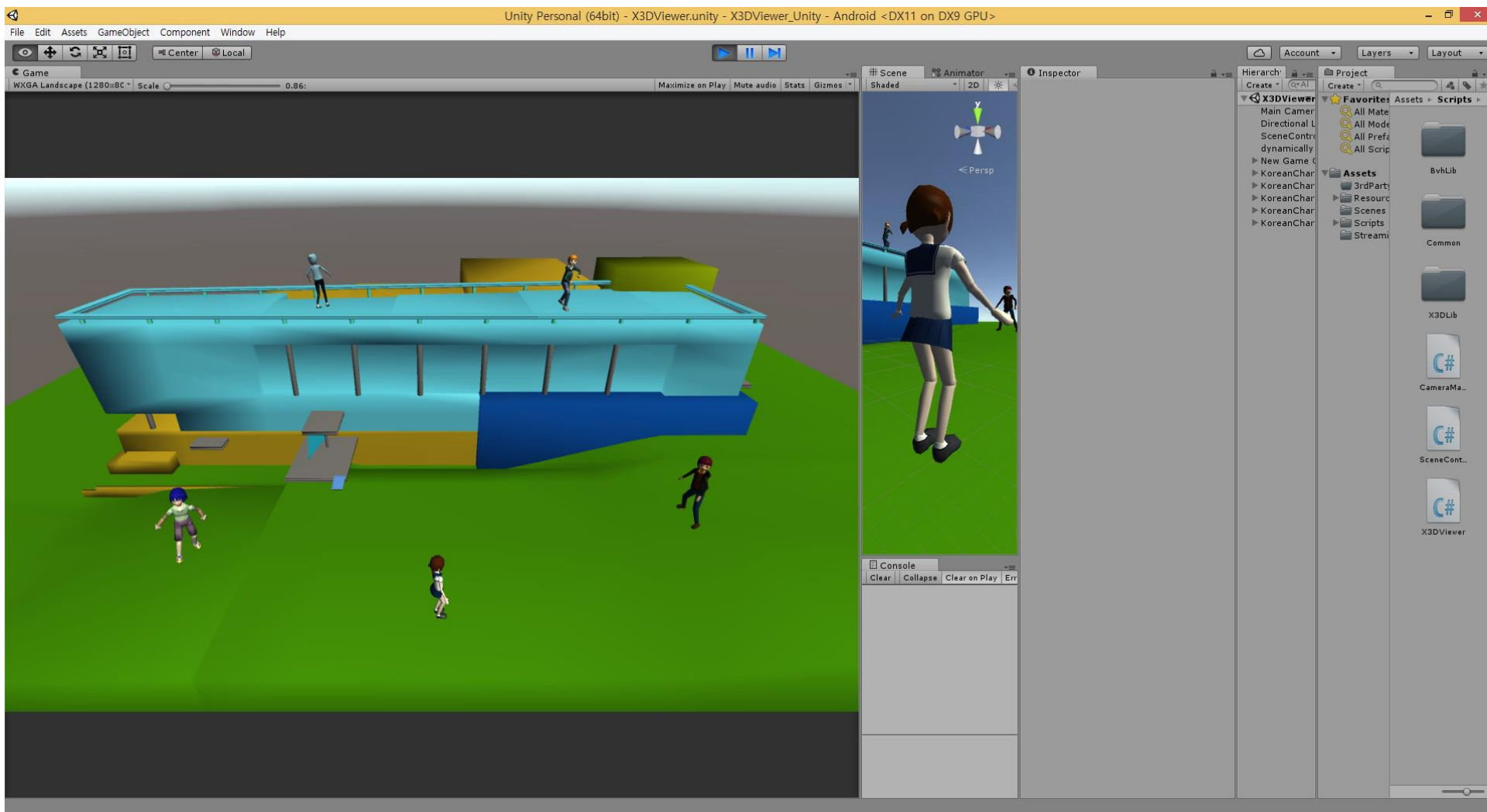


The screenshot shows the Visual Studio IDE with the following details:

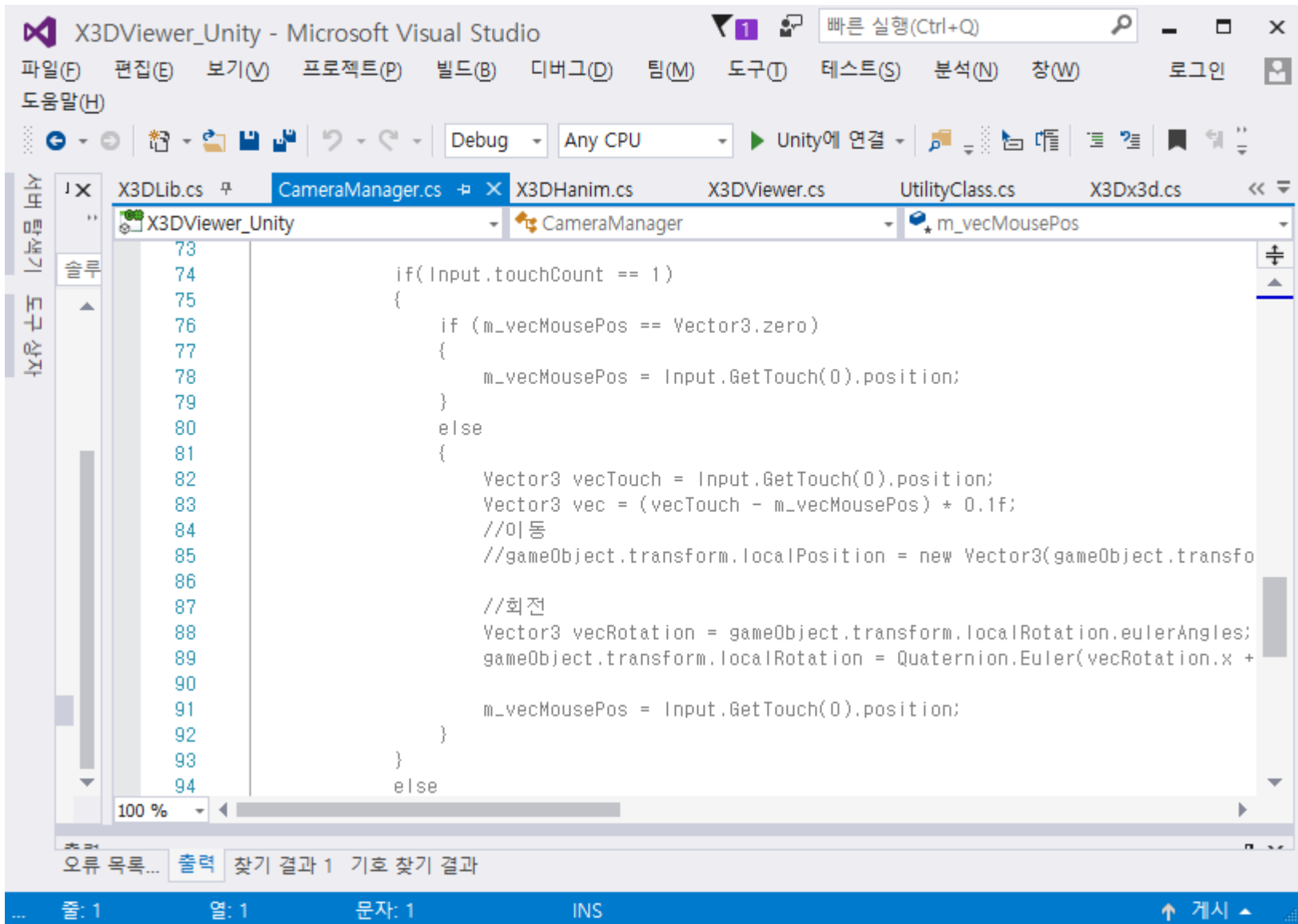
- Title Bar:** X3DViewer_Unity - Microsoft Visual Studio
- Menu Bar:** 파일(F) 편집(E) 보기(V) 프로젝트(P) 빌드(B) 디버그(D) 팀(M) 도구(T) 테스트(S) 분석(N) 창(W) 로그인
- Toolbar:** Includes buttons for file operations, a 'Debug' dropdown, 'Any CPU' target, and a 'Unity에 연결' button.
- File Explorer:** Shows a project structure with folders 'X3DViewer_Unity' and 'X3DHanim', and files 'X3DViewer.cs', 'UtilityClass.cs', 'X3Dx3d.cs', and 'X3DNode.cs'. The 'Draw()' method in 'X3DHanim.cs' is selected.
- Code Editor:** Contains the following C# code:

```
257  
258     if(m_strFileName.Contains("Jin") == true)  
259     {  
260         m_goRootParent.transform.localPosition = new Vector3(-117.4f, 1.5f, -92.7f);  
261     }  
262     else if (m_strFileName.Contains("Ru") == true)  
263     {  
264         m_goRootParent.transform.localPosition = new Vector3(-0.5f, 3.1f, -83.2f);  
265     }  
266     else if (m_strFileName.Contains("Ga") == true)  
267     {  
268         m_goRootParent.transform.localPosition = new Vector3(-62f, 0f, -109.5f);  
269     }  
270     else if (m_strFileName.Contains("Min") == true)  
271     {  
272         m_goRootParent.transform.localPosition = new Vector3(-23.4f, 40.1f, -28.3f);  
273     }  
274     else if (m_strFileName.Contains("Da") == true)  
275     {  
276         m_goRootParent.transform.localPosition = new Vector3(-98.4f, 38.6f, -20.1f);  
277     }  
278 }
```
- Bottom Bar:** Shows '줄: 257', '열: 9', '문자: 9', 'INS', and '게시'.

Results of Loading Multiple H-Anim Characters



Rotate X3D Camera

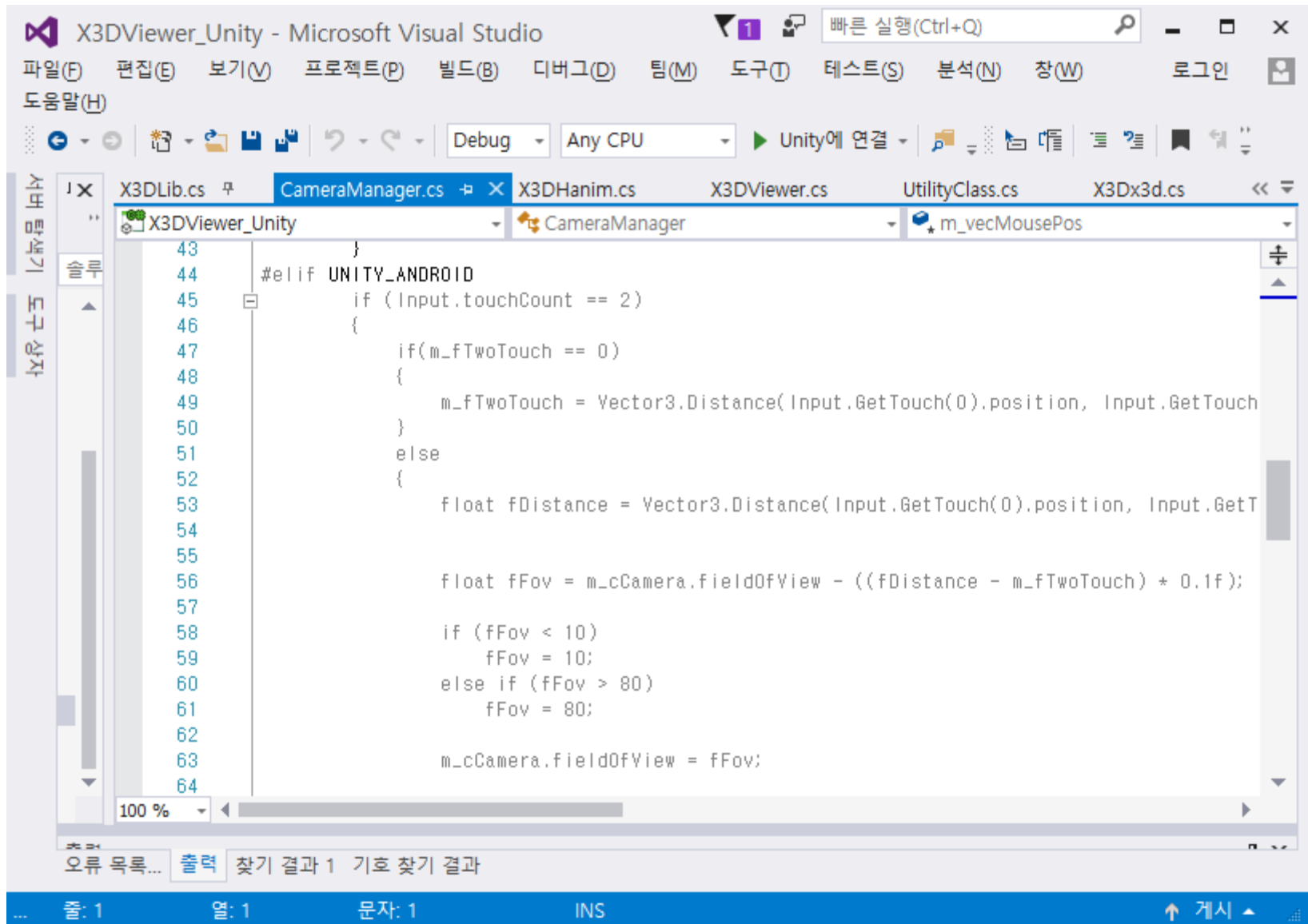


```
X3DViewer_Unity - Microsoft Visual Studio
빠른 실행(Ctrl+Q)
파일(F) 편집(E) 보기(V) 프로젝트(P) 빌드(B) 디버그(D) 팀(M) 도구(T) 테스트(S) 분석(N) 창(W) 로그인
도움말(H)
Debug Any CPU Unity에 연결
X3DLib.cs CameraManager.cs X3DAnim.cs X3DViewer.cs UtilityClass.cs X3Dx3d.cs
X3DViewer_Unity CameraManager m_vecMousePos
73
74     if(Input.touchCount == 1)
75     {
76         if (m_vecMousePos == Vector3.zero)
77         {
78             m_vecMousePos = Input.GetTouch(0).position;
79         }
80     }
81     else
82     {
83         Vector3 vecTouch = Input.GetTouch(0).position;
84         Vector3 vec = (vecTouch - m_vecMousePos) * 0.1f;
85         //이동
86         //gameObject.transform.localPosition = new Vector3(gameObject.transfo
87
88         //회전
89         Vector3 vecRotation = gameObject.transform.localRotation.eulerAngles;
90         gameObject.transform.localRotation = Quaternion.Euler(vecRotation.x +
91
92         m_vecMousePos = Input.GetTouch(0).position;
93     }
94     }
95     else
```

오류 목록... 출력 찾기 결과 1 기호 찾기 결과

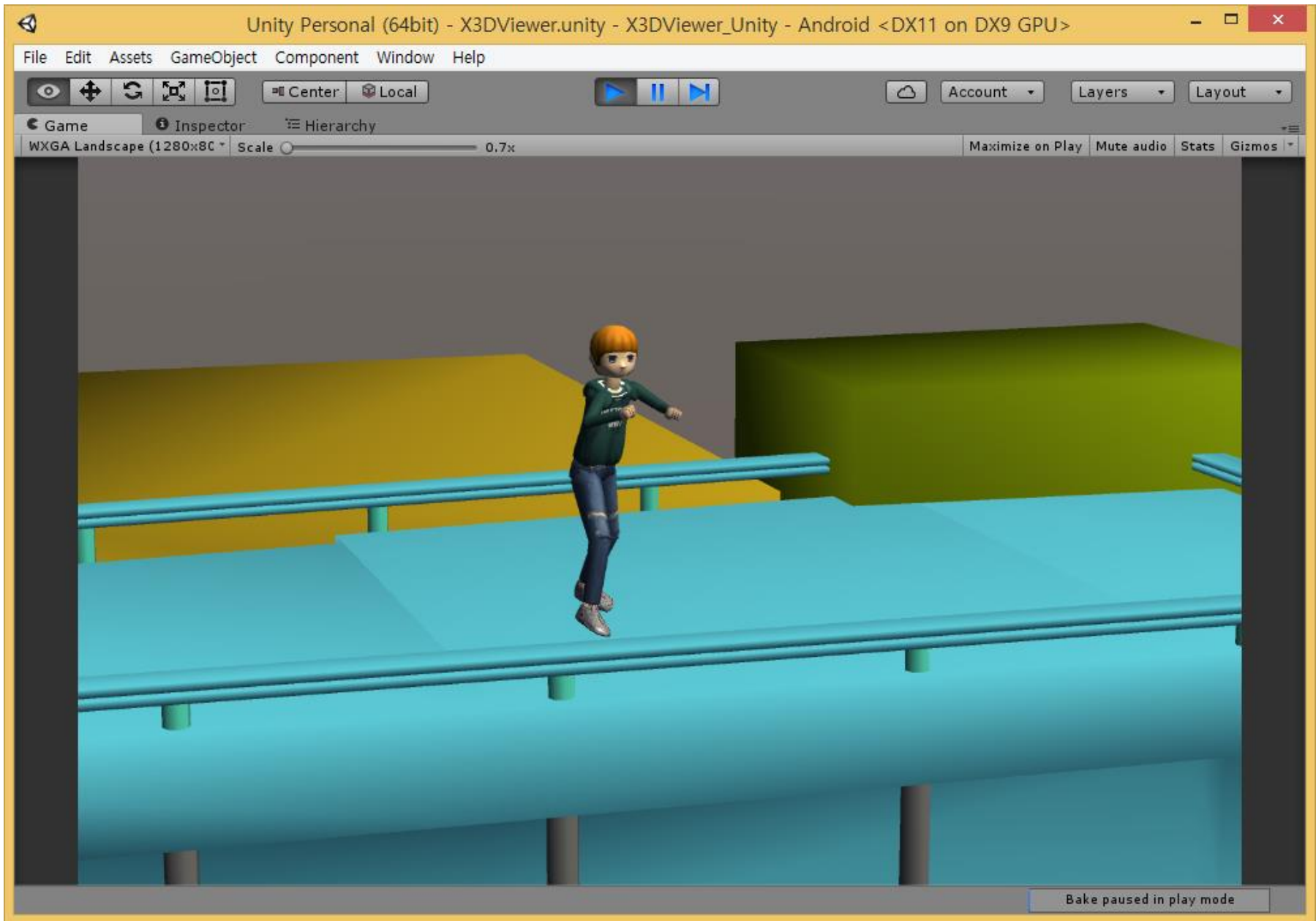
줄: 1 열: 1 문자: 1 INS 게시

Adjust X3D Camera FOV

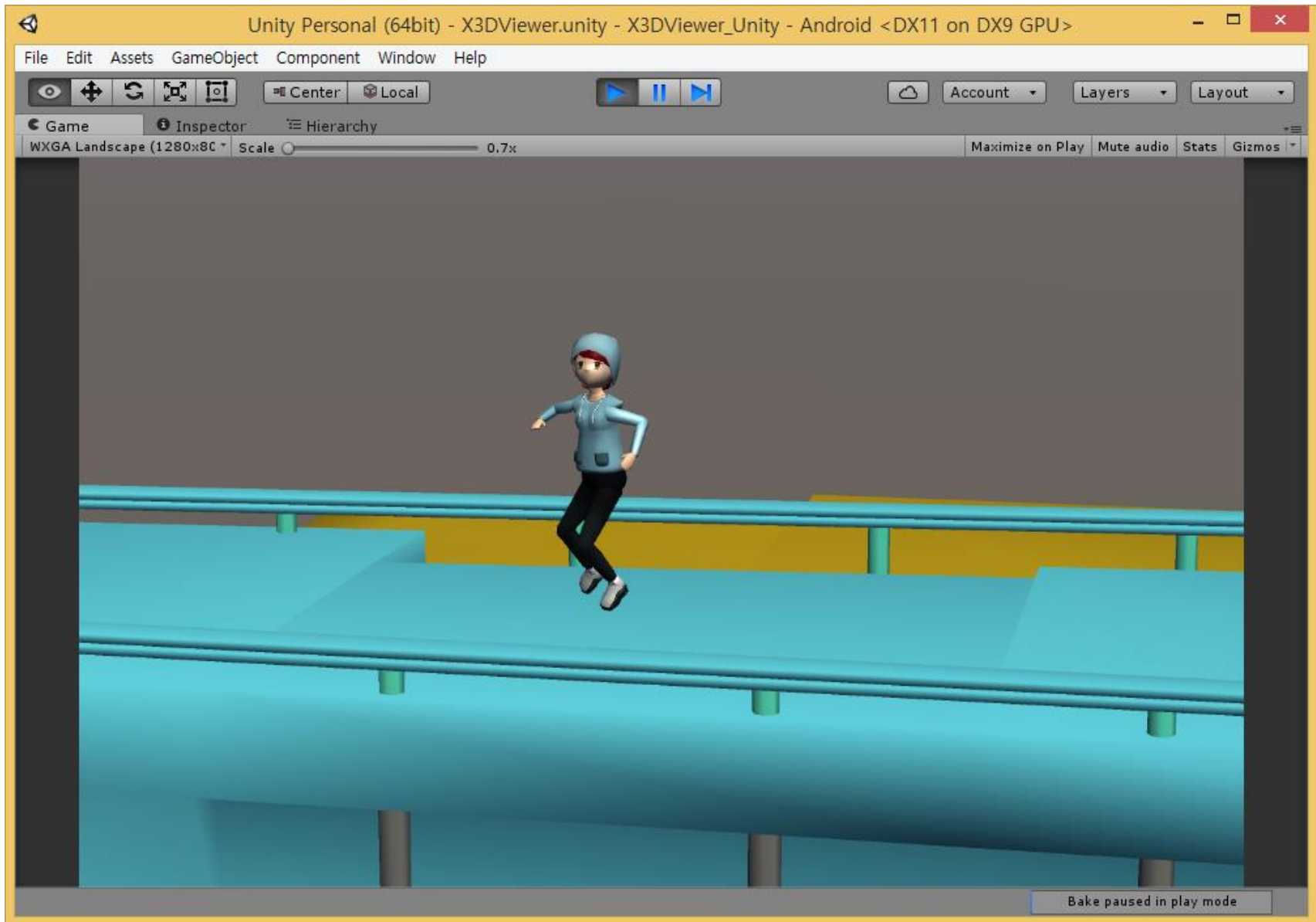


```
X3DViewer_Unity - Microsoft Visual Studio
빠른 실행(Ctrl+Q)
파일(F) 편집(E) 보기(V) 프로젝트(P) 빌드(B) 디버그(D) 팀(M) 도구(T) 테스트(S) 분석(N) 창(W) 로그인
다음말(H)
Debug Any CPU Unity에 연결
X3DLib.cs CameraManager.cs X3DHanim.cs X3DViewer.cs UtilityClass.cs X3Dx3d.cs
X3DViewer_Unity CameraManager m_vecMousePos
43 }
44 #elif UNITY_ANDROID
45     if (Input.touchCount == 2)
46     {
47         if(m_fTwoTouch == 0)
48         {
49             m_fTwoTouch = Vector3.Distance(Input.GetTouch(0).position, Input.GetTouch
50         }
51     else
52     {
53         float fDistance = Vector3.Distance(Input.GetTouch(0).position, Input.GetT
54
55
56         float fFov = m_cCamera.fieldOfView - ((fDistance - m_fTwoTouch) * 0.1f);
57
58         if (fFov < 10)
59             fFov = 10;
60         else if (fFov > 80)
61             fFov = 80;
62
63         m_cCamera.fieldOfView = fFov;
64
100 %
오류 목록... 출력 찾기 결과 1 기호 찾기 결과
줄: 1 열: 1 문자: 1 INS ↑ 게시
```

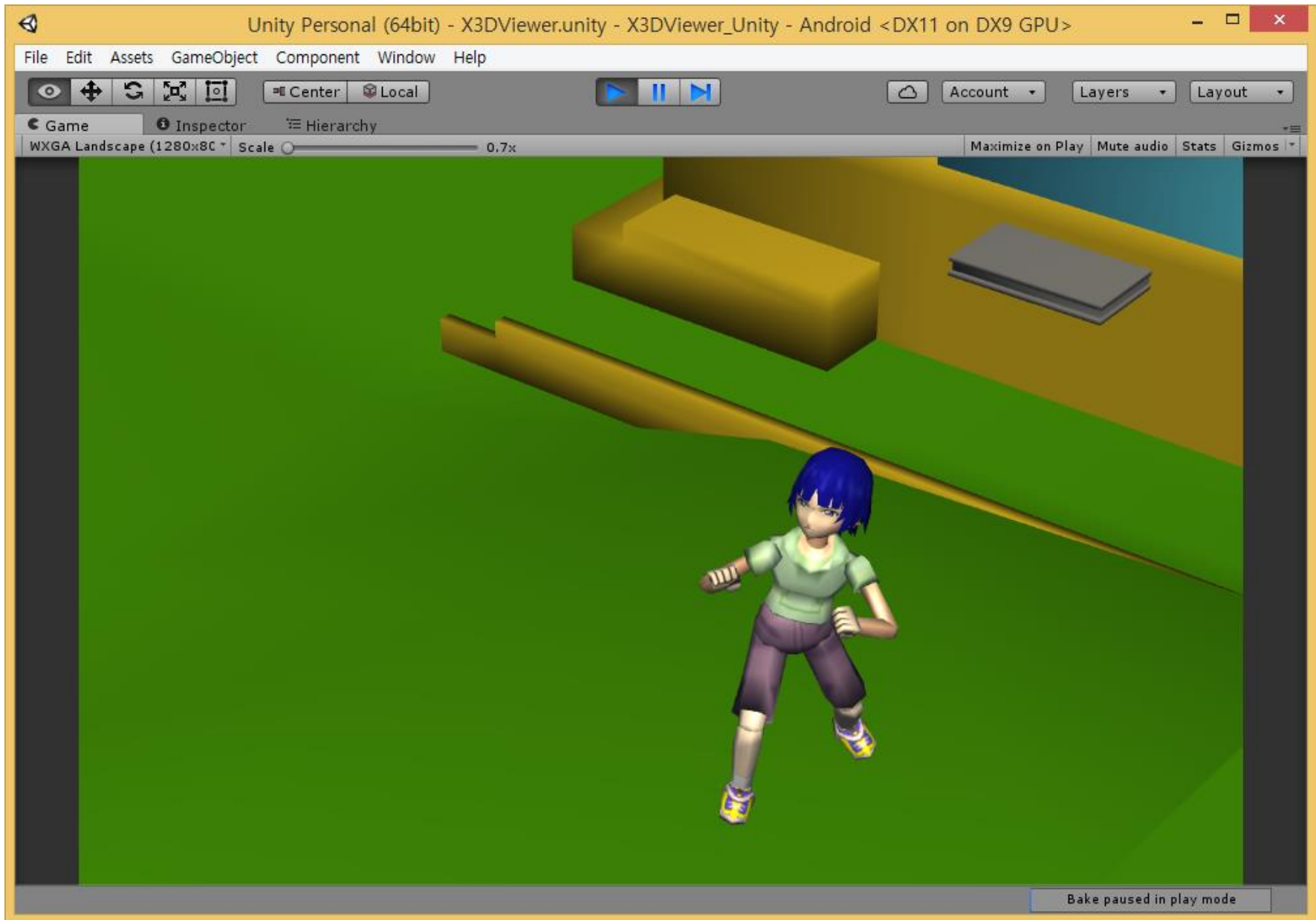
Screen Touch for Camera Rotate and Zoom (1)



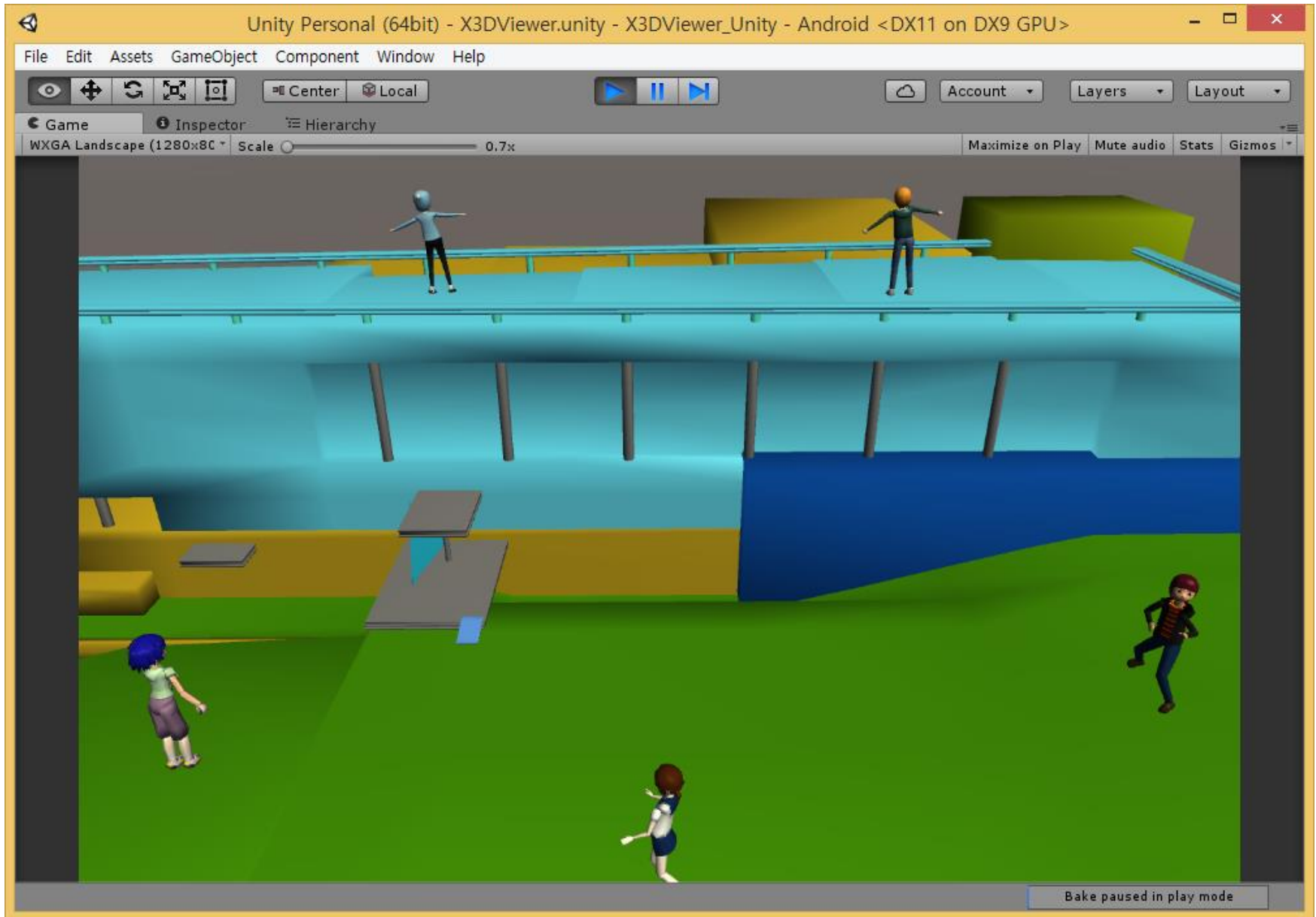
Screen Touch for Camera Rotate and Zoom (2)



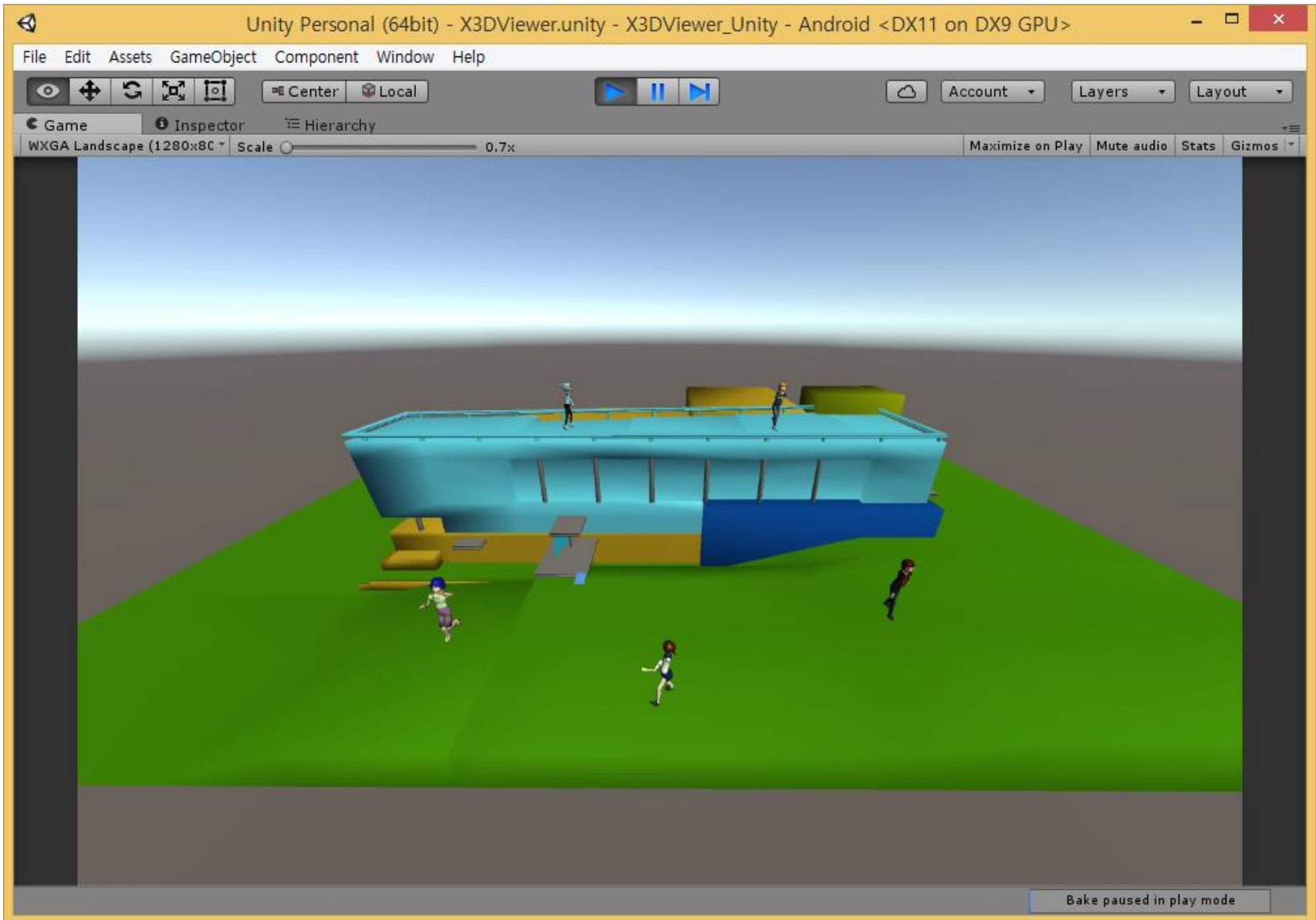
Screen Touch for Camera Rotate and Zoom (3)



Screen Touch for Camera Rotate and Zoom (4)



Screen Touch for Camera Rotate and Zoom (5)



Unity X3D Implementation

**H-Anim LOA1, LOA2, LOA3, and LOA4
Character Animation Mobile**

Prepare H-Anim characters



```
<HAnimJoint DEF="hanim_humanoid_root" center="0.000000  
<HAnimSegment DEF="hanim_sacrum" name="sacrum" >  
<Transform translation="0.000000 30.530001 -0.707600"  
<Shape>  
<Appearance>  
<Material diffuseColor="0.588000 0.588000 0.588000"/>  
<ImageTexture DEF="Jin_LOA1TextureAtlas" url="Jin.png"  
</Appearance>  
<IndexedFaceSet creaseAngle="3.14159" coordIndex="0, 1  
texCoordIndex="0, 1, 2, -1, 0, 2, 3, -1, 0, 3, 4, -1,  
<Coordinate point="0.0000 10.7900 0.1424, 0.0000 10.00  
<TextureCoordinate point="0.6211 0.5754,0.7851 0.5720,  
</IndexedFaceSet>  
</Shape>  
</Transform>  
</HAnimSegment>  
  
<HAnimJoint DEF="hanim_sacroiliac" center="0.000000 35  
<HAnimSegment DEF="hanim_pelvis" name="pelvis" >  
<Transform translation="0.000000 35.799999 -0.707600"  
</Transform>
```

Prepare motion capture data

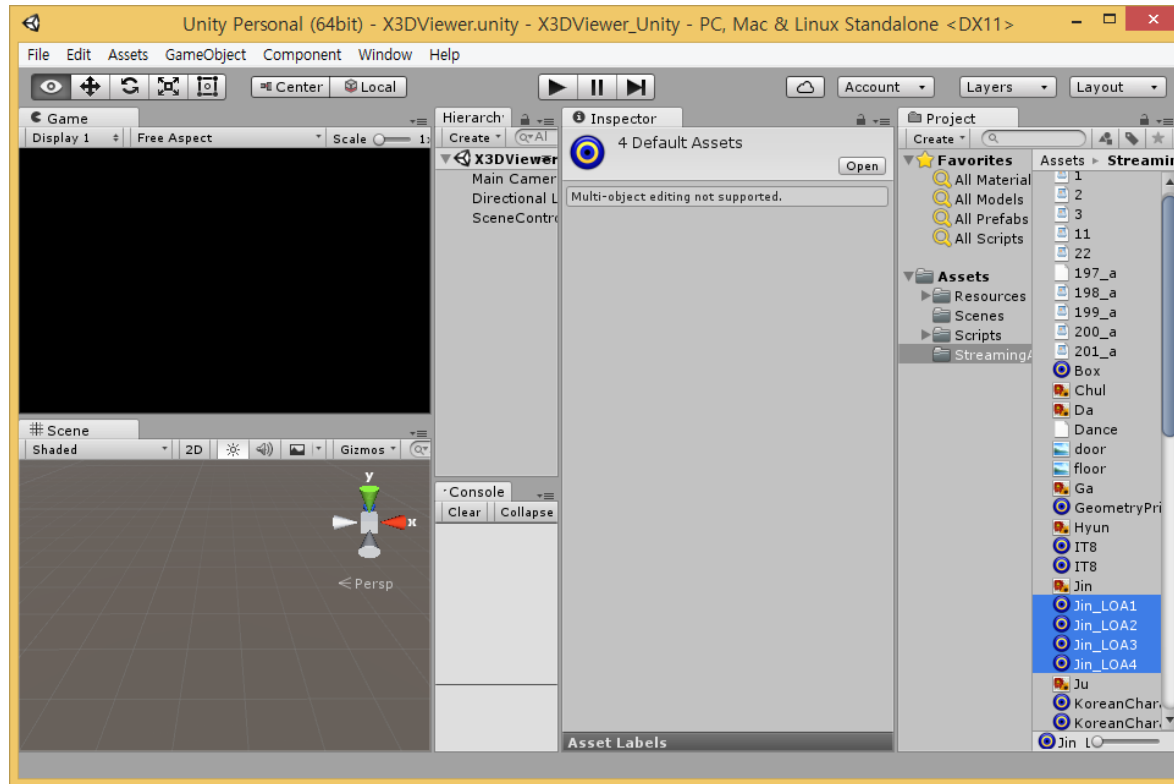
- Example: Kick BVH motion capture file

```
HIERARCHY
ROOT Hips
{
  OFFSET 0.000000 0.000000 0.000000
  CHANNELS 6 Xposition Yposition Zposition Zrotation
  JOINT Chest
  {
    OFFSET 0.000000 5.613096 0.000000
    CHANNELS 3 Zrotation Xrotation Yrotation
    JOINT LeftCollar
    {
      OFFSET 0.003804 10.354579 1.025227
      CHANNELS 3 Zrotation Xrotation Yrotation
      JOINT LeftShoulder
      {
        OFFSET 3.922637 0.000000 0.000000
        CHANNELS 3 Zrotation Xrotation Yrotation
        JOINT LeftElbow
        {
          OFFSET 0.000000 -10.663884 0.000000
          CHANNELS 3 Zrotation Xrotation Yrotation
          JOINT LeftWrist
          {
            OFFSET 0.000000 -7.995827 0.000000
            CHANNELS 3 Zrotation Xrotation Yrotation
```

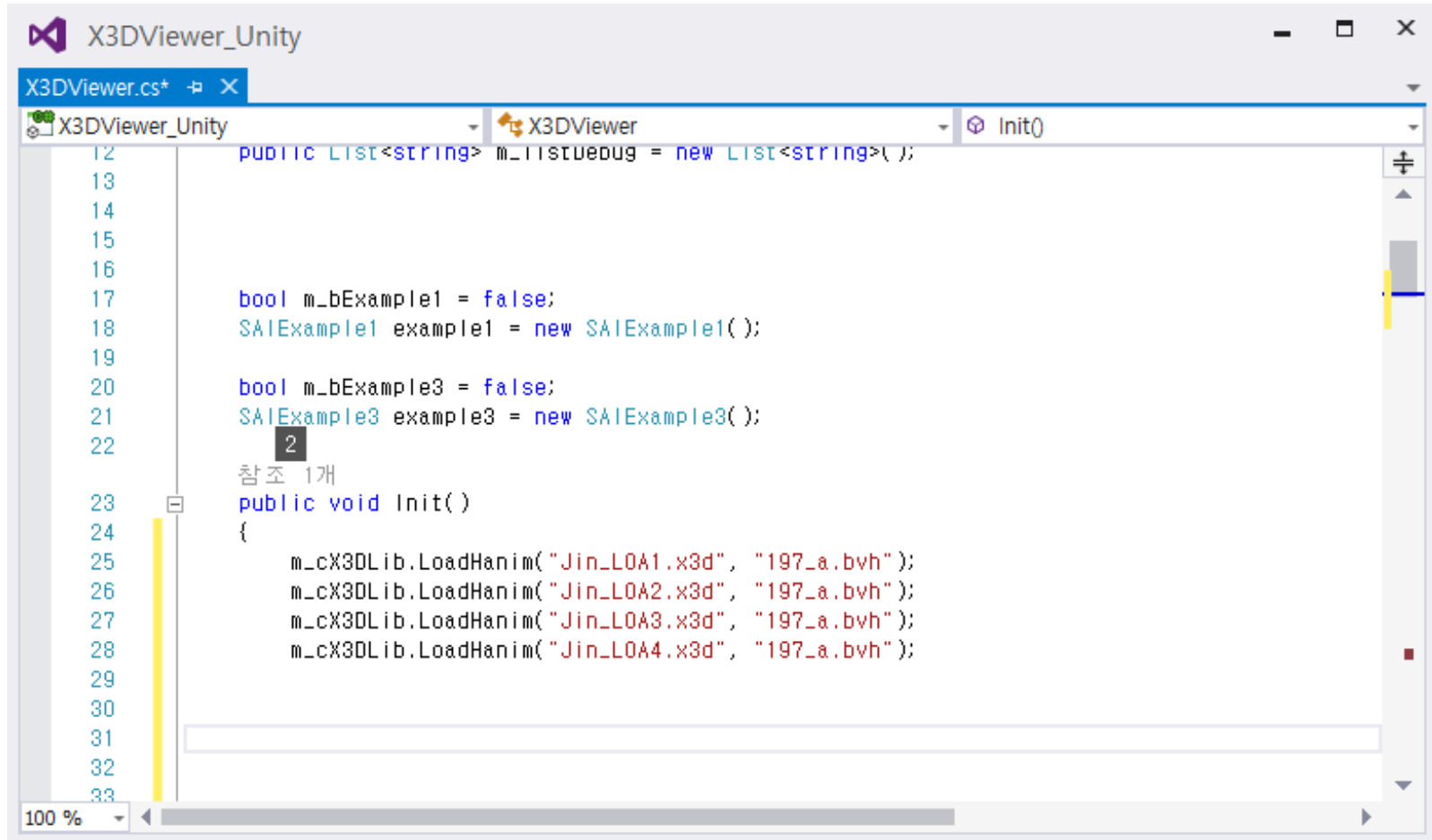
```
MOTION
Frames: 482
Frame Time: 0.016667
1.662 31.427 60.304 -1.249 -4.859 -3.582 4.463
1.659 31.427 60.307 -1.268 -4.835 -3.588 4.487
1.657 31.428 60.310 -1.287 -4.811 -3.594 4.512
1.654 31.428 60.313 -1.306 -4.787 -3.599 4.536
1.652 31.428 60.316 -1.324 -4.764 -3.605 4.560
1.649 31.428 60.319 -1.343 -4.740 -3.611 4.584
1.647 31.428 60.322 -1.362 -4.716 -3.616 4.609
1.645 31.428 60.324 -1.381 -4.693 -3.622 4.633
1.642 31.428 60.327 -1.400 -4.669 -3.628 4.657
1.640 31.428 60.330 -1.419 -4.646 -3.634 4.682
1.637 31.428 60.333 -1.438 -4.622 -3.639 4.706
1.635 31.428 60.336 -1.457 -4.599 -3.645 4.730
1.633 31.428 60.339 -1.476 -4.575 -3.651 4.755
1.630 31.428 60.342 -1.495 -4.552 -3.656 4.779
1.628 31.428 60.345 -1.514 -4.528 -3.660 4.804
1.625 31.428 60.348 -1.533 -4.505 -3.665 4.829
1.623 31.428 60.351 -1.552 -4.483 -3.671 4.853
1.621 31.428 60.354 -1.571 -4.462 -3.679 4.877
```

Read H-Anim characters

- H-Anim LOA1, LOA2, LOA3, and LOA4 characters

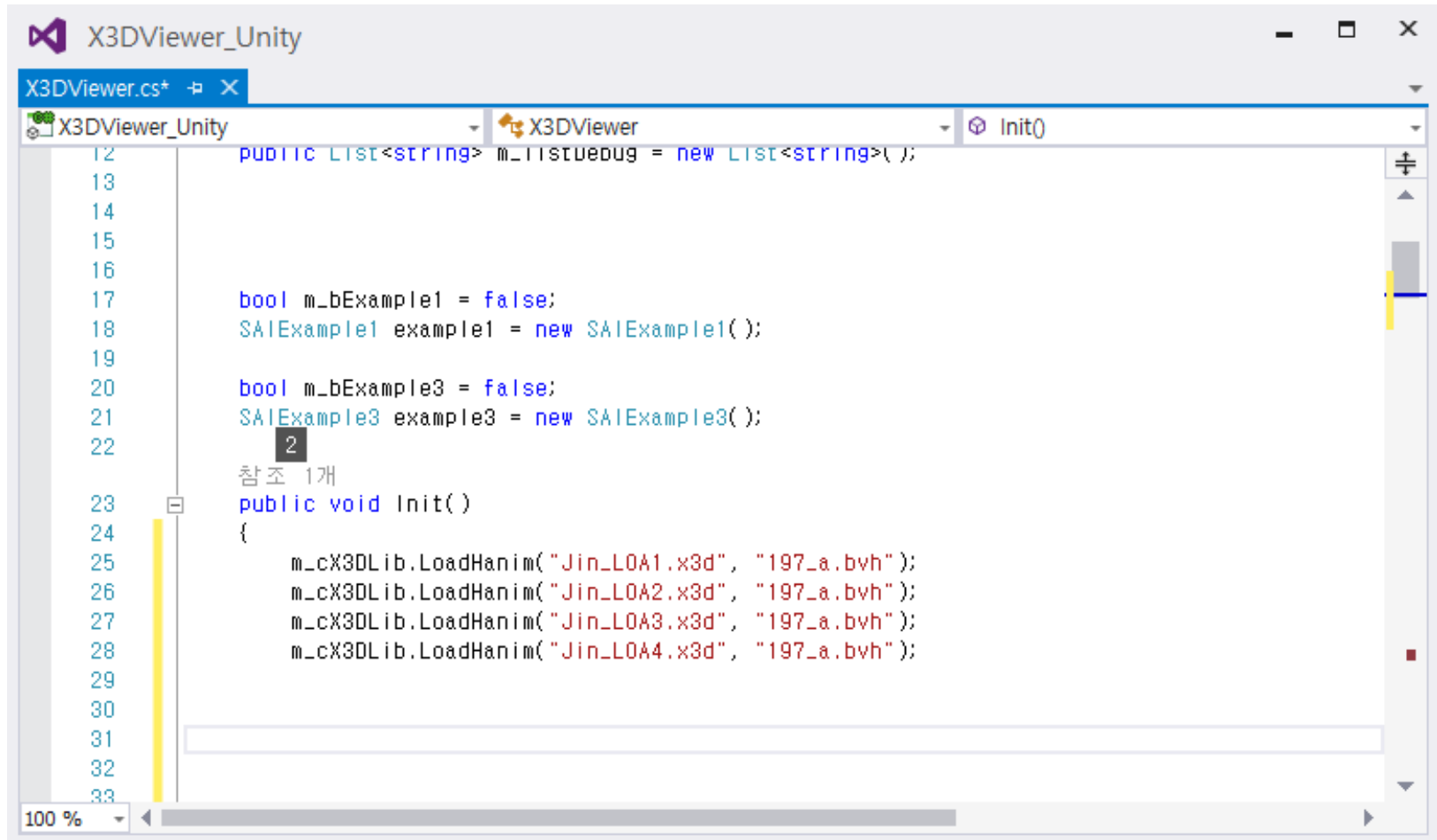


Load H-Anim characters and BVH mocap



```
X3DViewer_Unity
X3DViewer.cs* X X
X3DViewer_Unity X3DViewer Init()
12 public List<string> m_mListDebug = new List<string>();
13
14
15
16
17 bool m_bExample1 = false;
18 SAExample1 example1 = new SAExample1();
19
20 bool m_bExample3 = false;
21 SAExample3 example3 = new SAExample3();
22
23 참조 1개
24 public void Init()
25 {
26     m_cX3DLib.LoadHanim("Jin_LOA1.x3d", "197_a.bvh");
27     m_cX3DLib.LoadHanim("Jin_LOA2.x3d", "197_a.bvh");
28     m_cX3DLib.LoadHanim("Jin_LOA3.x3d", "197_a.bvh");
29     m_cX3DLib.LoadHanim("Jin_LOA4.x3d", "197_a.bvh");
30
31
32
33
100 %
```


Parsing H-Anim characters



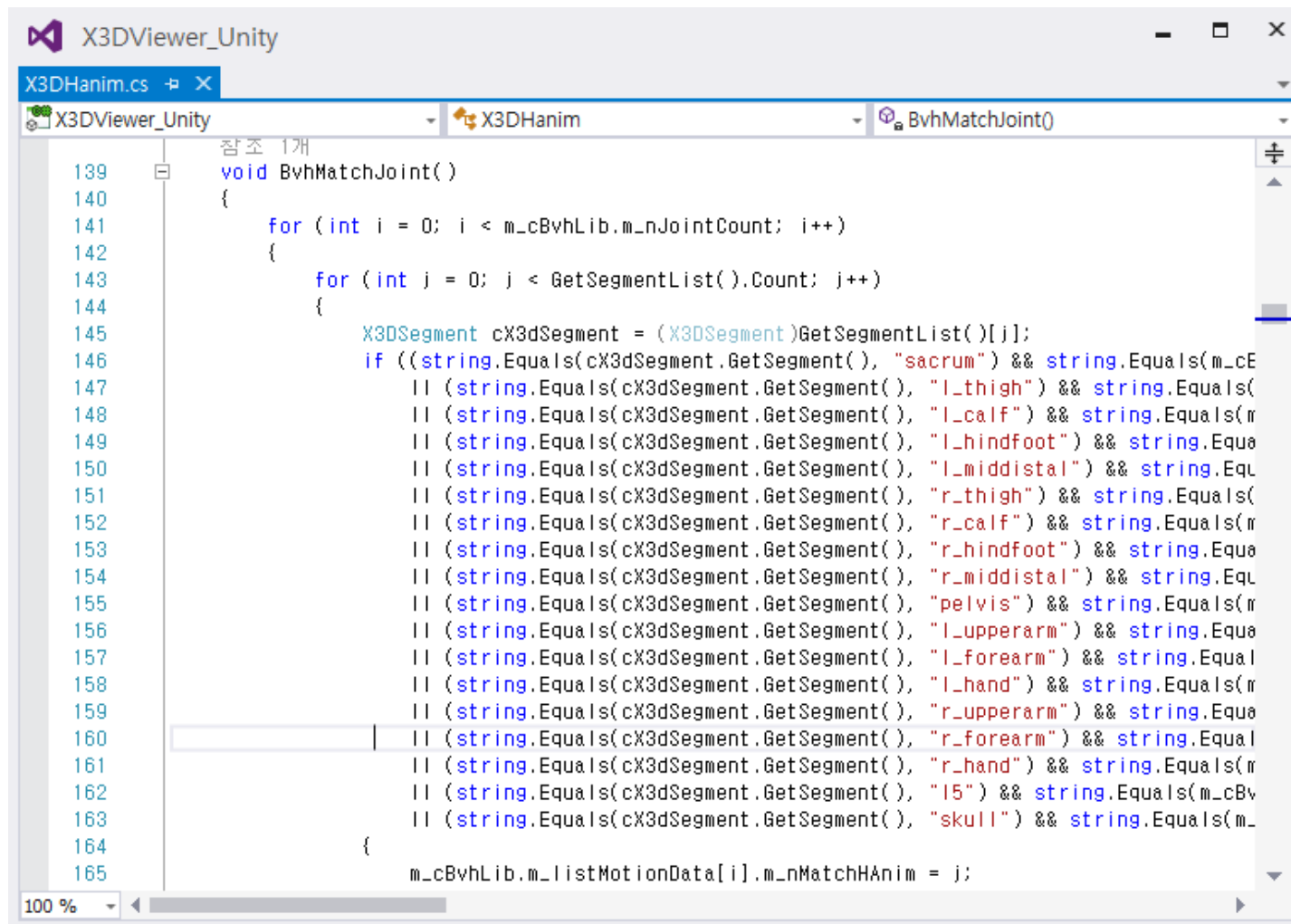
```
X3DViewer_Unity
X3DViewer.cs* X X
X3DViewer_Unity X3DViewer Init()
12 public List<string> m_ListDebug = new List<string>();
13
14
15
16
17 bool m_bExample1 = false;
18 SAExample1 example1 = new SAExample1();
19
20 bool m_bExample3 = false;
21 SAExample3 example3 = new SAExample3();
22
23 참조 1개
24 public void Init()
25 {
26     m_cX3DLib.LoadHanim("Jin_LOA1.x3d", "197_a.bvh");
27     m_cX3DLib.LoadHanim("Jin_LOA2.x3d", "197_a.bvh");
28     m_cX3DLib.LoadHanim("Jin_LOA3.x3d", "197_a.bvh");
29     m_cX3DLib.LoadHanim("Jin_LOA4.x3d", "197_a.bvh");
30
31
32
33
100 %
```

Parsing BVH mocap data

```
X3DViewer_Unity
BvhLib.cs
X3DViewer_Unity
BvhLib
ParseBvhRecursive(string strBvh)
47
참조 1개
48 public IEnumerator CoroutineLoadBvh( string strFileName )
49 {
50     m_strFileName = X3DViewer_Unity
51
52     WWW www = Utility
53
54     yield return www;
55
56     ParseBvhRecursive
57 }
58
참조 1개
100 %
```

```
X3DViewer_Unity
BvhLib.cs
X3DViewer_Unity
BvhLib
ParseBvhRecursive(string strBvh)
58
참조 1개
59 public void ParseBvhRecursive( string strBvh )
60 {
61     string[] arrBvh = strBvh.Split();
62     List<string> listBvh = new List<string>();
63     for( int i = 0; i < arrBvh.Length; ++i )
64     {
65         if(string.Equals("", arrBvh[i]) == false)
66         {
67             listBvh.Add( arrBvh[i] );
68         }
69     }
70
71     for( int i = 0; i < listBvh.Count; ++i )
72     {
73         string str = listBvh[i];
74
75         switch( str )
76         {
77             case "ROOT":
78             case "JOINT":
79                 m_nJointCount++;
80
81                 //모션데이터 초기화 과정
82                 MotionData cMotionData = new MotionData();
83                 cMotionData.m_strName = listBvh[++i];
84                 m_listMotionData.Add( cMotionData );
85                 break;
86         }
87     }
88 }
100 %
```

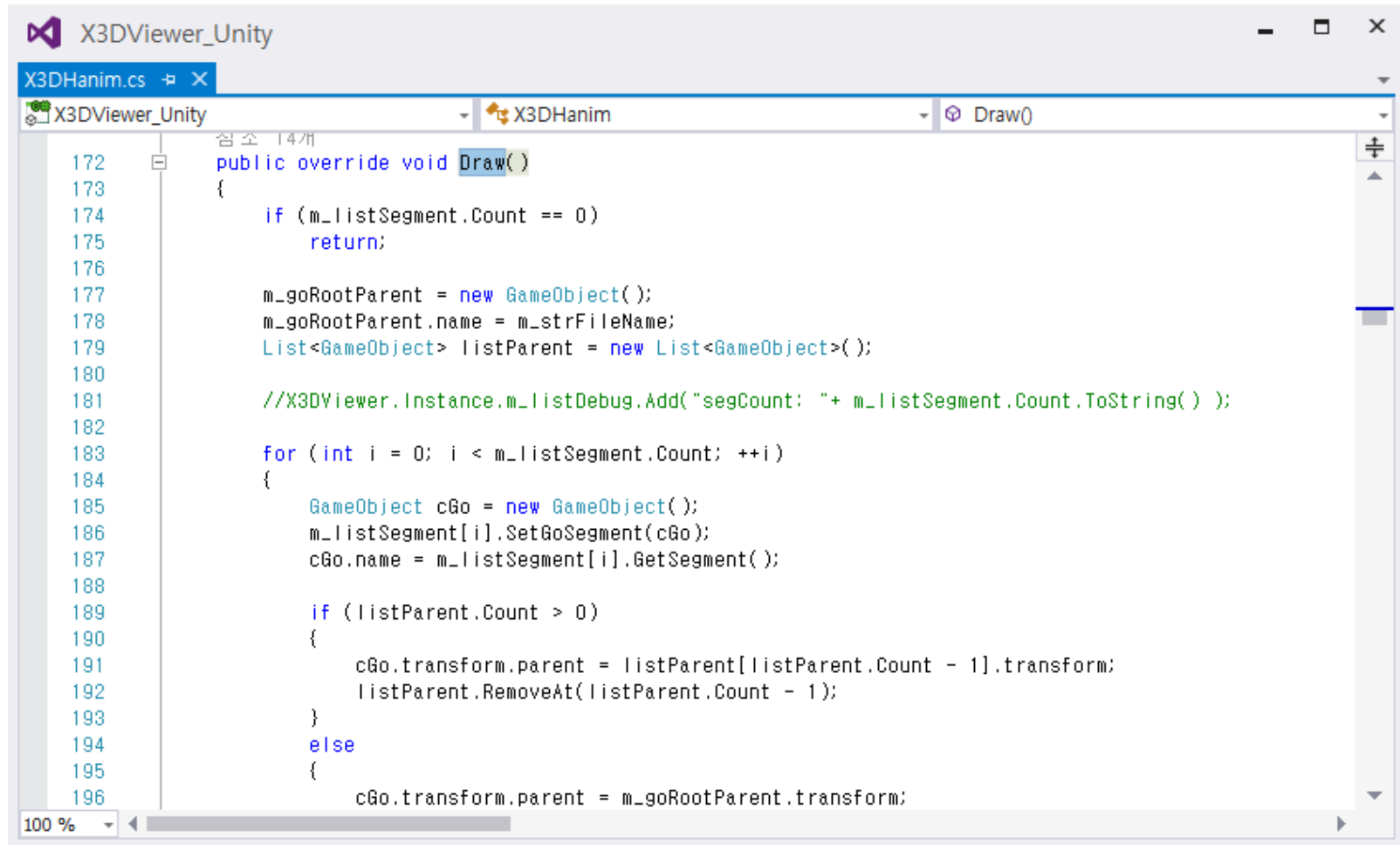
Joint mapping between H-Anim characters and BVH mocap



```
X3DViewer_Unity
X3DHanim.cs
X3DViewer_Unity
X3DHanim
BvhMatchJoint()

참조 1개
139 void BvhMatchJoint()
140 {
141     for (int i = 0; i < m_cBvhLib.m_nJointCount; i++)
142     {
143         for (int j = 0; j < GetSegmentList().Count; j++)
144         {
145             X3DSegment cX3dSegment = (X3DSegment)GetSegmentList()[j];
146             if ((string.Equals(cX3dSegment.GetSegment(), "sacrum") && string.Equals(m_cE
147                 || (string.Equals(cX3dSegment.GetSegment(), "l_thigh") && string.Equals(
148                 || (string.Equals(cX3dSegment.GetSegment(), "l_calf") && string.Equals(r
149                 || (string.Equals(cX3dSegment.GetSegment(), "l_hindfoot") && string.Equa
150                 || (string.Equals(cX3dSegment.GetSegment(), "l_middistal") && string.Eql
151                 || (string.Equals(cX3dSegment.GetSegment(), "r_thigh") && string.Equals(
152                 || (string.Equals(cX3dSegment.GetSegment(), "r_calf") && string.Equals(r
153                 || (string.Equals(cX3dSegment.GetSegment(), "r_hindfoot") && string.Equa
154                 || (string.Equals(cX3dSegment.GetSegment(), "r_middistal") && string.Eql
155                 || (string.Equals(cX3dSegment.GetSegment(), "pelvis") && string.Equals(r
156                 || (string.Equals(cX3dSegment.GetSegment(), "l_upperarm") && string.Equa
157                 || (string.Equals(cX3dSegment.GetSegment(), "l_forearm") && string.Equal
158                 || (string.Equals(cX3dSegment.GetSegment(), "l_hand") && string.Equals(r
159                 || (string.Equals(cX3dSegment.GetSegment(), "r_upperarm") && string.Equa
160                 || (string.Equals(cX3dSegment.GetSegment(), "r_forearm") && string.Equal
161                 || (string.Equals(cX3dSegment.GetSegment(), "r_hand") && string.Equals(r
162                 || (string.Equals(cX3dSegment.GetSegment(), "l5") && string.Equals(m_cBv
163                 || (string.Equals(cX3dSegment.GetSegment(), "skull") && string.Equals(m_
164             {
165                 m_cBvhLib.m_listMotionData[i].m_nMatchHAnim = j;
```

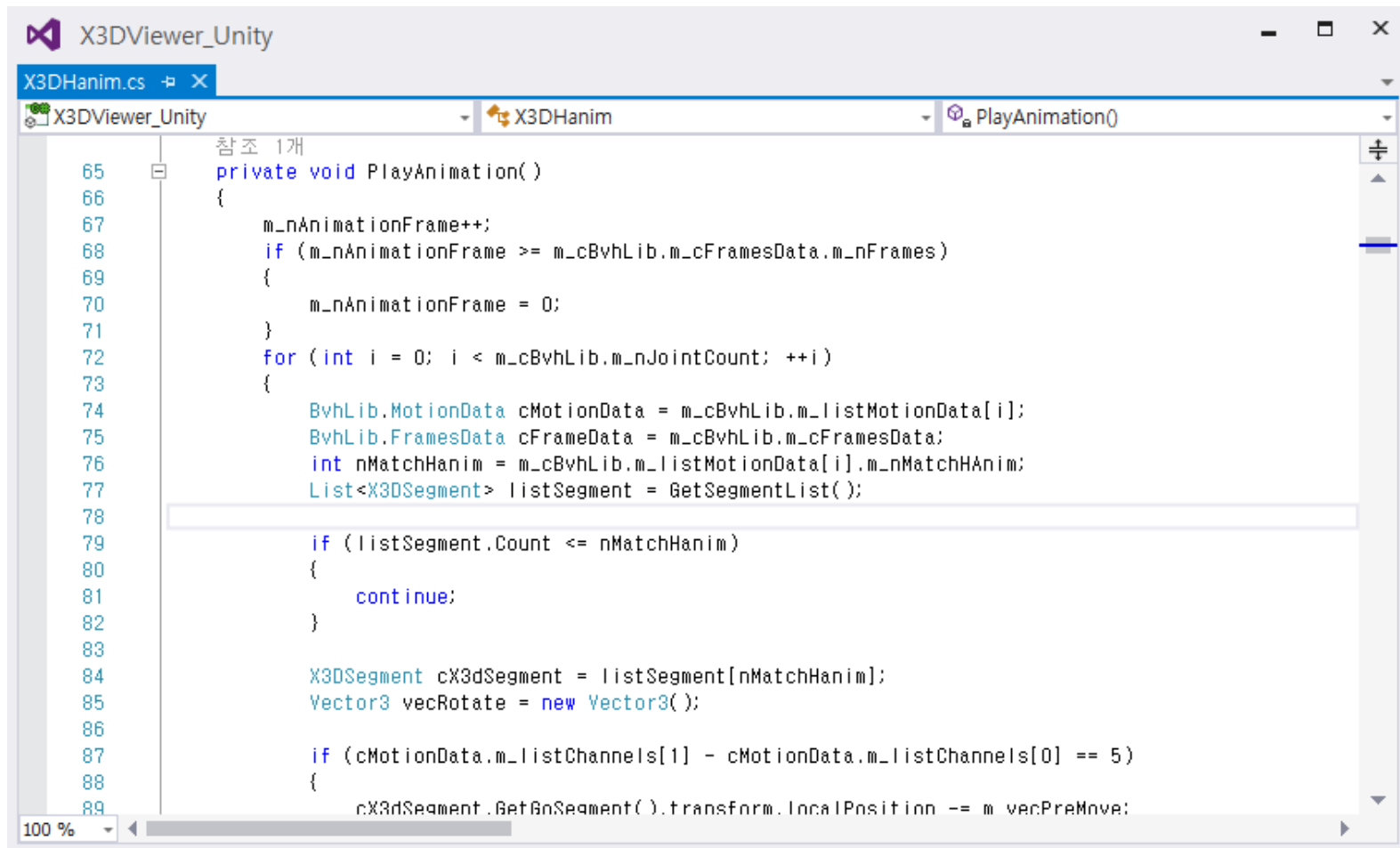
Draw H-Anim characters



The screenshot shows a Visual Studio window titled "X3DViewer_Unity" with a tab for "X3DHanim.cs". The code editor displays the following C# code for the `Draw()` method:

```
172     public override void Draw()
173     {
174         if (m_listSegment.Count == 0)
175             return;
176
177         m_goRootParent = new GameObject();
178         m_goRootParent.name = m_strFileName;
179         List<GameObject> listParent = new List<GameObject>();
180
181         //X3DViewer.Instance.m_listDebug.Add("segCount: " + m_listSegment.Count.ToString());
182
183         for (int i = 0; i < m_listSegment.Count; ++i)
184         {
185             GameObject cGo = new GameObject();
186             m_listSegment[i].SetGoSegment(cGo);
187             cGo.name = m_listSegment[i].GetSegment();
188
189             if (listParent.Count > 0)
190             {
191                 cGo.transform.parent = listParent[listParent.Count - 1].transform;
192                 listParent.RemoveAt(listParent.Count - 1);
193             }
194             else
195             {
196                 cGo.transform.parent = m_goRootParent.transform;
```

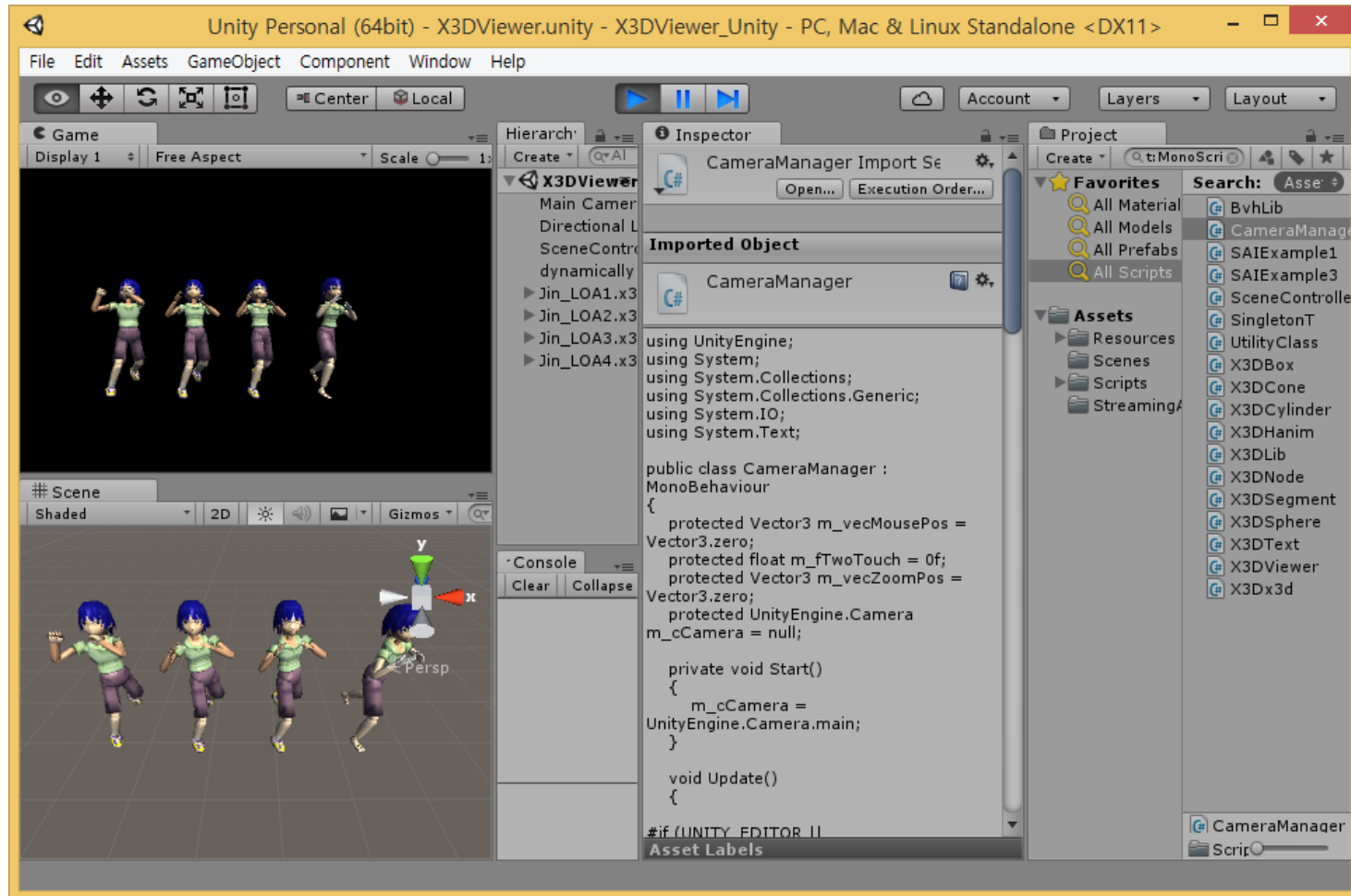
Generate H-Anim character motion



```
X3DViewer_Unity
X3DHanim.cs
X3DHanim
PlayAnimation()

참조 1개
65 private void PlayAnimation()
66 {
67     m_nAnimationFrame++;
68     if (m_nAnimationFrame >= m_cBvhLib.m_cFramesData.m_nFrames)
69     {
70         m_nAnimationFrame = 0;
71     }
72     for (int i = 0; i < m_cBvhLib.m_nJointCount; ++i)
73     {
74         BvhLib.MotionData cMotionData = m_cBvhLib.m_listMotionData[i];
75         BvhLib.FramesData cFrameData = m_cBvhLib.m_cFramesData;
76         int nMatchHanim = m_cBvhLib.m_listMotionData[i].m_nMatchHanim;
77         List<X3DSegment> listSegment = GetSegmentList();
78
79         if (listSegment.Count <= nMatchHanim)
80         {
81             continue;
82         }
83
84         X3DSegment cX3dSegment = listSegment[nMatchHanim];
85         Vector3 vecRotate = new Vector3();
86
87         if (cMotionData.m_listChannels[1] - cMotionData.m_listChannels[0] == 5)
88         {
89             cX3dSegment.GetBnSegment().transform.localPosition -= m_vecPreMove;
```

Results on the Unity X3D viewer



Results on the Unity X3D viewer

- H-Anim LOA1, LOA2, LOA3, and LOA4 characters



Conclusions

- Goal
 - X3D based Mobile VR
 - X3D based mobile 3D data representation and exchange
 - Representation of mobile sensors, their functions, and interaction with copied real worlds
 - Physical sensor nodes and device interfaces
- Unity X3D Mobile Implementation
 - Unity X3D Viewer

Acknowledgement:

This research was partially supported by the MSIT(Ministry of Science, ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2018-2016-0-00312) supervised by the IITP(Institute for Information & communications Technology Promotion)