

VRコミュニケーションライブラリについて

2018年6月13日

NTTドコモ
移動機開発部
第二イノベーション推進担当

■ VRコミュニケーションライブラリについて


ドコモで実装した3Dアバターを制御するプログラムをUnity用ライブラリ化したものです。
HTC VIVE及びVIVEトラッカー使用を前提に、下記機能を提供します。

- ・アバター制御ライブラリ
- ・モーションキャプチャライブラリ（3点/6点トラッキング、及びリープモーションの活用）
- ・瞬き/リップシンクライブラリ（瞬き、音声によるリップシンク）
- ・身体キャリブレーションライブラリ（個人ごとの手足の長さに合わせてアバターの自動調整）

作成するアプリに、3Dアバターを用いたよりリッチなコミュニケーション環境を実装することを可能とします。

■ 本資料について

提供する「DocomoAvatar.unitypackage」の導入説明資料です。
アイデアソン及びハッカソンにおいては、サンプルシーン・サンプルアバター・そこで使用しているPrefab・プログラムを本資料と併せてご提供します。



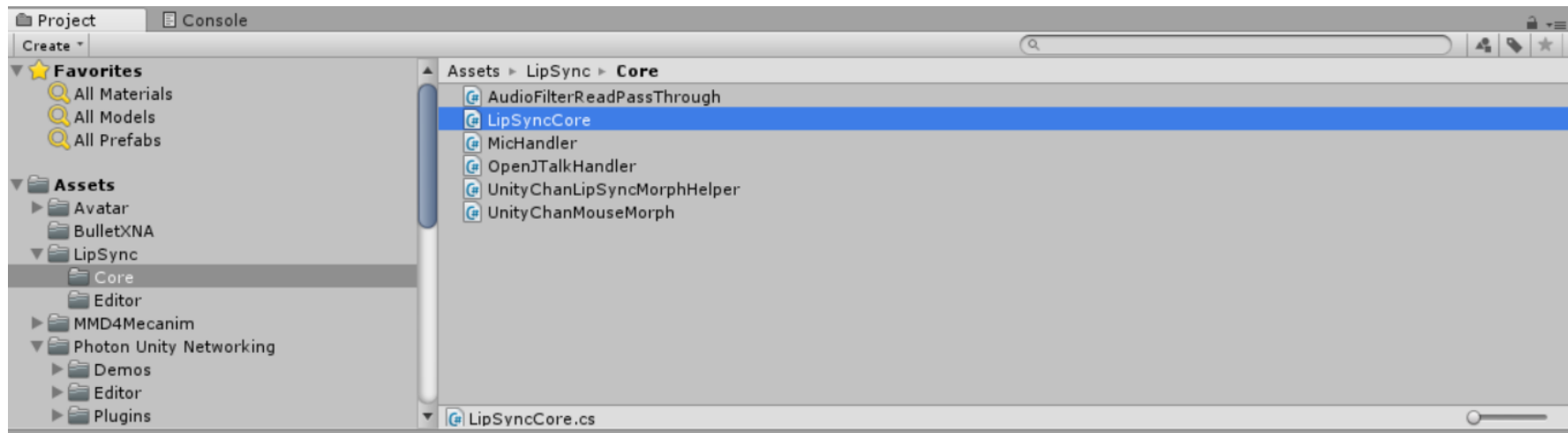
サンプルシーンの実行

事前導入アセット

「DocomoAvatar.unitypackage」をインポートする前に下記のAssetをインポートしてください

- SteamVR Plugin
(<https://assetstore.unity.com/packages/templates/systems/steamvr-plugin-32647>)
- Photon Unity Networking Free
(<https://assetstore.unity.com/packages/tools/network/photon-unity-networking-free-1786>)
- Photon Voice
(<https://assetstore.unity.com/packages/tools/audio/photon-voice-45848>)
- ~~FinalIK~~
(<https://assetstore.unity.com/packages/tools/animation/final-ik-14290>)
※不要化対応中
- MMD4Mecanim
(http://stereoarts.jp/MMD4Mecanim_Beta_20180304.zip)
- MMD4MFaceBlink
(<https://github.com/udasan/unity-mmd4mface>)
- LipSync-Plugin
(<https://github.com/hecomi/MMD4Mecanim-LipSync-Plugin>)
※Asset/LipSyncをインポートしてください。

LipSyncを修正する



LipSyncCore.csをPhotonVoiceの音で動作するように変更します。

同梱の「AssetのLipSyncのCoreに置き換える」フォルダのLipSyncCore.csの内容をインポートしてきたLipSyncCore.csにマージしてください。

サンプルSceneの使い方（1）

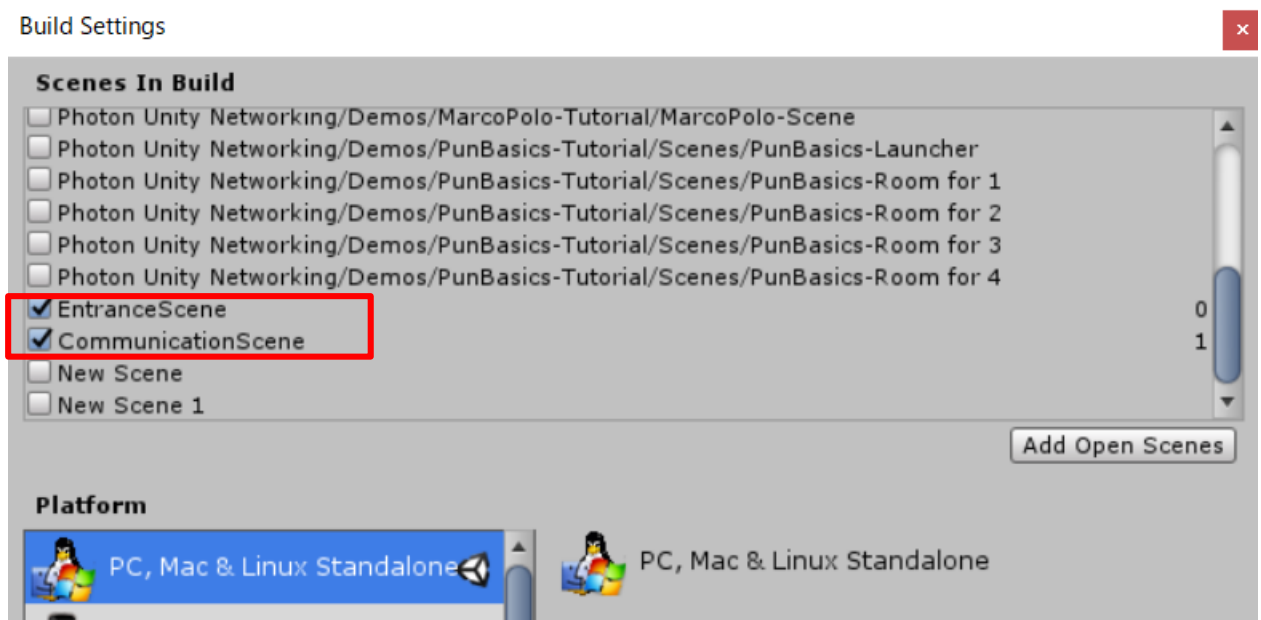
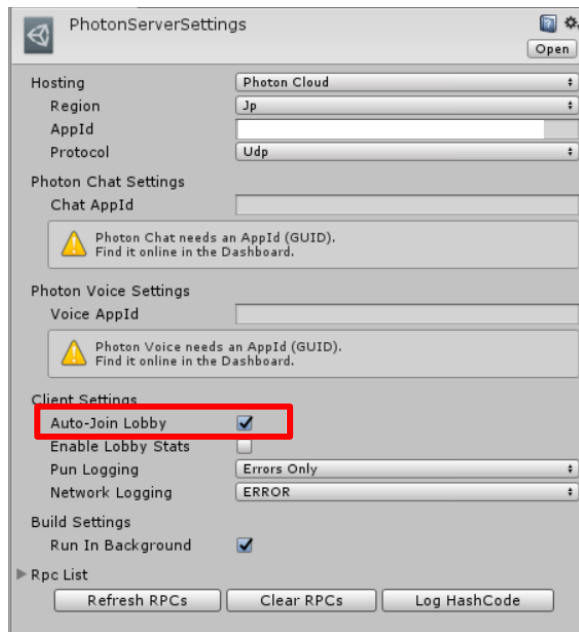
ドコモアバターパッケージに梱包されているシーンから、空間共有を実行できます。

「EntranceScene」 アバターと席の選択を行うシーン

「CommunicationScene」 アバターになって動きを共有するシーン

サンプルシーンを実行する際には「EntranceScene」と「CommunicationScene」を両方Photonのビルド対象にしてから実行するようにしてください。

また、Photonの設定は「AutoJoinLobby」にしておいてください。



サンプルSceneの使い方（2）

EntranceSceneを実行してください。

Photonに接続できれば、緑のパネルが表示されます。
VIVEコントローラのトリガーでアバターと席を選択して開始ボタンを押してください
CommunicationSceneに自動的に遷移します。

CommunicationSceneに入ると3点トラッキングが開始されます。
サンプルではVIVEトラッカーに右手・左手が追従し、VIVE HMDに頭が追従します。

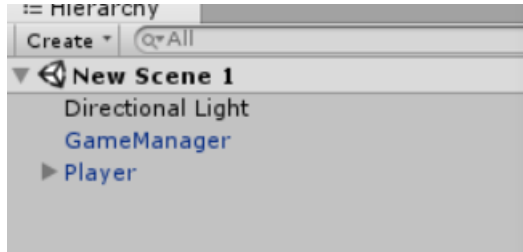
VIVEトラッカーは最初にペアリングしたものが右手、次にペアリングしたものが左手に対応します。
VIVEトラッカーは電源ランプが左側に来るようにつけてください。

ランプが左側になるようにつける



アバター同期の取り込み方

SynchronizeAvatar配下の「Player」と「GameManager」をSceneに追加してください。



GameManagerでアバターを生成しています。
EntranceSceneのようにアバターを選んだ場合、GameManager_StartGameで
インスタンス化していますので、アバターを増やす場合は種類を追加してください。

```
// 選択した席に従ってPlayerの位置を設定する
GameObject player = GameObject.Find("Player");
switch(seat_no) {
    case 1:
        player.transform.position = chair1.transform.position - new Vector3(0.0f, 0.225f, 0.0f);
        break;
    case 2:
        player.transform.position = chair2.transform.position - new Vector3(0.0f, 0.225f, 0.0f);
        break;
    default:
        break;
}

// 選択したアバター番号に従ってアバターを生成
switch (player_no) {
    case 1:
        gameplayer = PhotonNetwork.Instantiate("Man", player.transform.position, Quaternion.Euler(new
        break;
    case 2:
        gameplayer = PhotonNetwork.Instantiate("Woman", player.transform.position, Quaternion.Euler(ne
        break;
    default:
        break;
}
```


パッケージに含まれるディレクトリの説明

- **Avatar**
サンプルのシーンで利用しているアバターデータ
- **Resources**
アバター選択画面で利用している空間共有と直接関係ないファイル
- **SynchronizeAvatar**
空間共有で利用しているファイル