

## 1 2. 製作したソフトウェア

### 1.1 ソフトウェアの使用方法

#### 1.1.1 環境のセットアップ

本製品の使用には以下のハードウェア環境が必要となる。

- Windows10(64-bit) machine
- CUDA 対応 GPU

加えて以下のソフトウェア環境が必要となる。

- Python3.X (Anaconda3 の最新バージョンに対応したものを推奨)
- Anaconda3
- Unity editor
- CUDA toolkit v9.0
- CUDNN 7.0.5
- Openpose v1.4.0
- Lifting from the Deep

同梱のバッチファイルを利用することで環境構成負担は大幅に軽減されている。紙面の都合上，実行環境のセットアップに関するの詳細は <https://github.com/syspro5/iwamoto> を参照のこと。

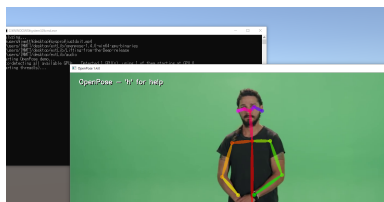
#### 1.1.2 実行方法

1. unity エディタを起動し，本製品のプロジェクトファイルを開く。
2. 解析したい動画のパスを指定する。
3. Analyze ボタンを押すと解析が開始され，結果ファイルが出力される
4. 解析終了後 Animate ボタンを押すことで字幕とモデルの動きの動画が出力される。

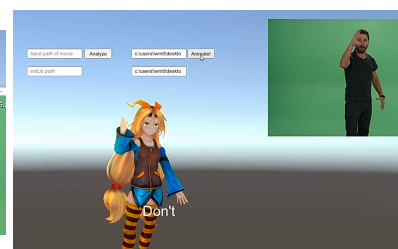
実際の実行の様子は同梱されたビデオファイルを参照のこと。



解析したい動画のパス指定



処理中の画面



実行結果

図 1: ミーティング内容の比較

## 1.2 ソフトウェアの機能・仕様

- 深度情報なしに骨格推定を行い、モデルに動きを反映させる一連の処理を画一化。
- 音声解析により字幕描画も実装。
- 推定可能人数は実装時点では1人
- 現時点では Unity editor 上での実行となる

## 1.3 ソフトウェアの設計・製作過程

### 1.3.1 設計

処理のフローチャートを以下の図2に示す。動画から音声などを抽出し、要素ごとに分けて処理を行っている。通常 Unity から直接 python の仮想環境へのアクセスはできないため、今回の実装ではシェルスクリプトを介して anaconda の仮想環境にアクセスする手法をとっている。仮想環境をセットアップするスクリプトも同梱しているため、利用者が一々必要なライブラリをインストールする必要はない。また、専用の仮想環境を作成することで、利用者の python 環境を改変することなしに複数のライブラリの利用が可能となることも利点となる。

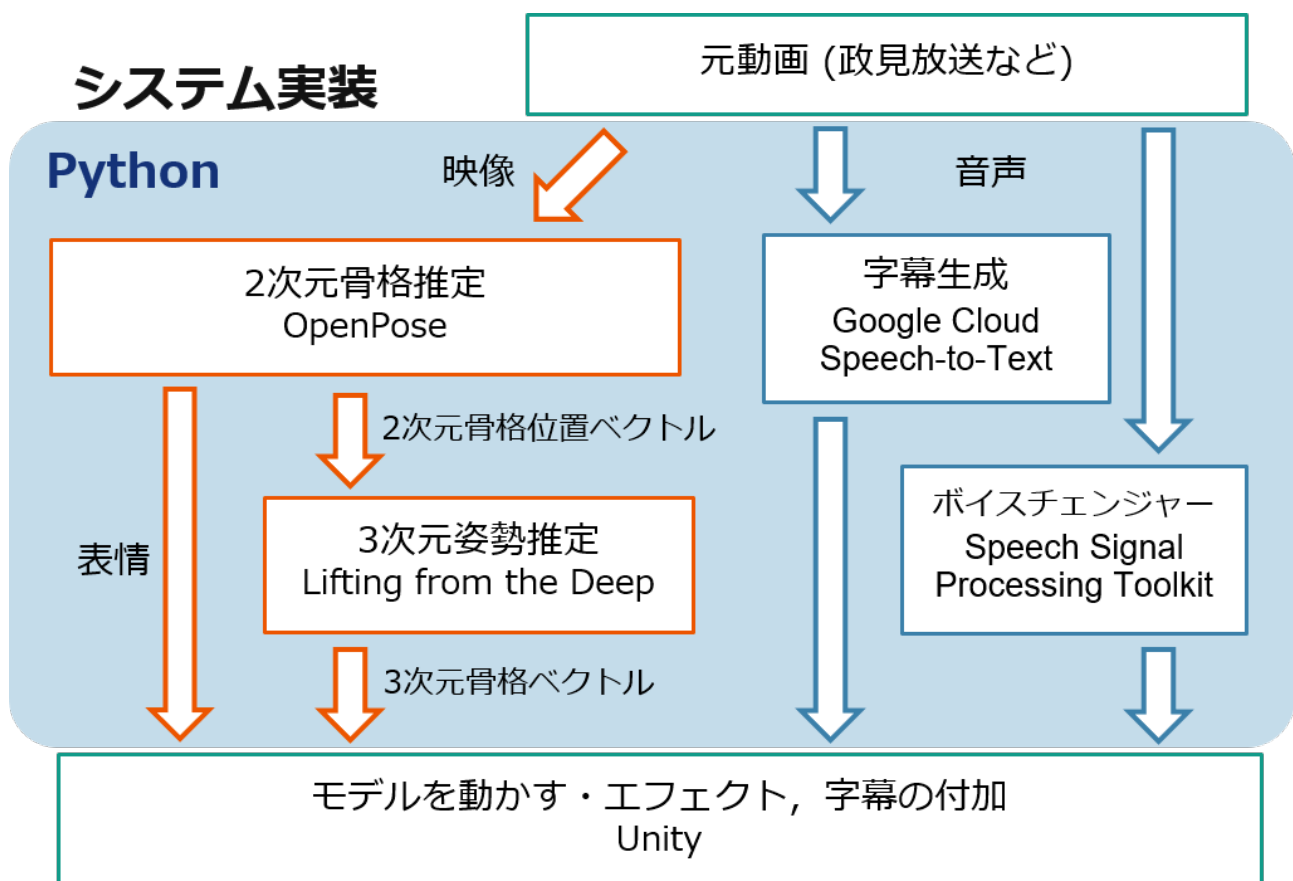
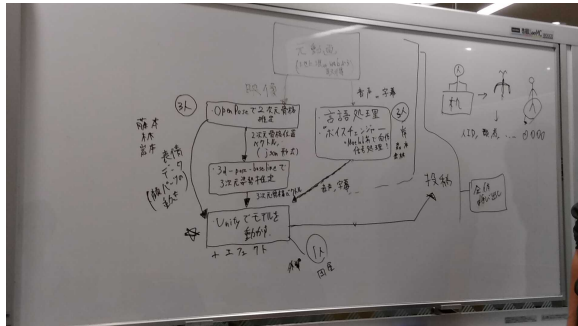


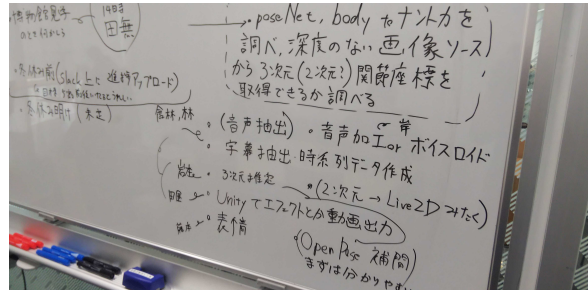
図 2: 処理の概要図

### 1.3.2 製作過程

10月時点と12月時点でのミーティング事項の比較を図3に示す。10月時点では定まっていなかった「具体的に何を使用するか」が、12月時点では明確に定まった。



10月時点



12月時点

図3: ミーティング内容の比較