体の動きに応じた特殊攻撃アニメーション投影システムの開発

久保市聡[†1]　　大橋裕太郎[†1]

プロジェクションマッピングを用いた美しい写真撮影、特にチームラボのようなインスタ映えするコンテンツが増加しているが、特定の姿勢推定を利用した映像反応コンテンツは少ない。本研究では、MediaPipe によるポーズ推定と TensorFlow を用いて体の動きを認識し、Unity でアニメーションを投影するプロジェクションマッピング環境を構築した。AI に詳しくない人でも容易に実現できるよう一部 GUI を実装した。このシステムを用いて、3 つの作品を制作した。

Development of special attack animation system based on body movements

SO KUBOICHI[†1]　YUTARO OHASHI[†2]

While there has been an increase in beautiful photography using projection mapping, especially for installations such as Team Lab's, there is a dearth of video reaction content that uses specific pose estimation. In this study, we built a projection mapping environment that recognizes body movements using pose estimation with MediaPipe and TensorFlow, and projects animations using Unity We implemented a partial GUI so that people who are not familiar with AI can easily implement the system. Using this system, we produced three works.

1. はじめに

1.1. 背景と動機

最新技術を用いたデジタルアートを用いたミュージアムやイベントが増えており、人々の関心を集めている。東京お台場では 2018 年 6 月、日本初のデジタルアート・ミュージアムである「MORI Building DIGITAL ART MUSEUM：EPSON TeamLabBorderless（森ビルデジタルアートミュージアム：エプソン チームラボボーダレス）」が開館し、開館から 1 年で約 230 万人の来場者数を記録した [1]。同様に、VTuber(バーチャル YouTuber)も最新技術を活用したエンターテイメントの一例として注目されている。VTuber は、モーションキャプチャ機器と CG 技術を組み合わせることでリアルタイムに仮想のキャラクターを操作し、視聴者とインタラクティブなコミュニケーションを取ることができる。さらに VTuber コンテンツの作成のため、2023 年に Sony が Mocopi [2]というモーションキャプチャ機器を発売した。Mocopi は、Bluetooth 接続の小型トラッカーを使用して、リアルタイムで動きを仮想キャラクターに反映させる技術で、VTuber の活動をさらに進化させている。そこで、本研究ではプロジェクションマッピングを用いたデジタルアートに姿勢推定技術を導入することで、特定のポーズに応じたインタラクティブな映像反応を実現する新しい体験を提案する。このアプローチにより、従来のデジタルアートが持つ視覚的な美しさに加え、鑑賞者の動きに反応する要素を加えることでより没入感のある体験を提供し、デジタルアートの可能性を広げることを目指す。

1.2. 目的

本研究の目的は、ポーズ推定技術を活用した新たなインタラクティブ体験を創出し、これを通じてアートの創作活動を刺激することである。具体的には、この技術を用いることで他のアーティストやデザイナーが自身の作品にインタラクティブ要素を組み込みやすくなるよう、支援ツールを開発する。以下の 2 つの目的・目標を設定している。

- 人のポーズをリアルタイムで認識し、その情報を用いてデジタルアート作品にインタラクティブな反応をさせるシステムの開発（以下、「本システム」とする）
- 本システムのコンテンツ開発補助ツールの作成

このアプローチにより、ツール利用者が、観賞者が直接作品に影響を与えるような体験を制作することができる。

2. 関連研究

2.1. 動的プロジェクションマッピング技術

末吉の研究では、動的な植物に対してプロジェクションマッピングを行い、メディアアートの作品を制作している [3]。この研究では、風や手による動きによって変形する植物に対し、自動で投影映像の位置合わせを行うシステムを開発している。このシステムは、カメラで撮影した情報を元に画像処理を施して行う投影対象の認識や、その情報を用いた動的なプロジェクションマッピングの位置合わせを実現している [3]。

2.2. カメラを用いた姿勢推定技術

飯野の研究では、Google が提供する Mediapipe [4]という機械学習を用いた骨格推定を行うことができるツールを用

†1 芝浦工業大学大学院
Shibaura Institute of Technology

いた。全身にキーポイント(landmark)を打ち、情報化する Mediapipe Pose を用いて必要な座標を抽出し、クラシックギターのより良い演奏姿勢について解析している。キーポイントを抽出し、演算を行なってベクトルなどの特定の値とすることで動きの変化を数字で追いやすくするという工夫がされている [5]。

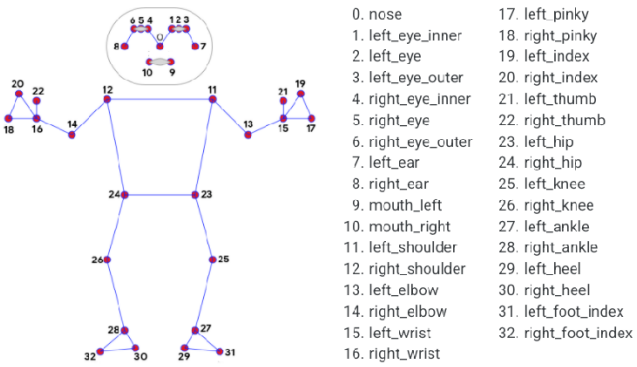

図 1 Mediapipe の Pose の 33 点の landmark [6]

Figure 1 33-point keypoints of Mediapipe's Pose

2.3. 芸術や娯楽におけるインタラクティブ技術.

株式会社ネイキッドが 2023 年 10 月に法隆寺でデジタルアートを活用した体験型の空間演出を行なった [7]。このプロジェクトでは観客が提灯を持ち、光の輪の中に立つことで観客も光の演出の中に入ることができるような演出があり、観客がインタラクティブにアート作品の一部になることができる。

3. システム設計

3.1. 概要

本システムは、Mediapipe Pose を用いてカメラから事前にユーザーの骨格情報を取得し、その情報を元にリアルタイムでポーズ認識を行う。その後、リアルタイムで骨格情報を取得し、ユーザーが指定したポーズをとっているかの識別を行い、識別結果を Unity に送信し、CG エフェクトを生成してプロジェクターから映像を投影する。

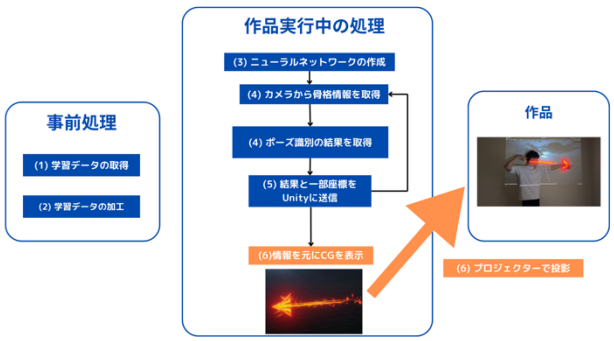

図 2 システムの概要図

Figure 2 System overview

ユーザーは、自分が姿勢を作ることでエフェクトを出しているような体験ができる。

3.2. システム構成

本システムは以下の要素から構成される。

(1) 学習データの取得

Mediapipe Pose を使用して、ユーザーはカメラからユーザーの骨格情報を取得することができる。

(2) 学習データの加工

取得したデータを加工し、ニューラルネットワークの学習用データセットを構築する。顔、腕などの向きを重視した特徴量を示すことができるベクトルデータとして変換する。この操作には識別の精度を向上させる狙いがある。

(3) ポーズ識別用ニューラルネットワークの作成

ベクトルデータを入力、指定したポーズをとっているかの分類を出力とするニューラルネットワークを作成する。

(4) ポーズ識別

リアルタイムで取得した骨格情報をニューラルネットワークに入力し、指定したポーズをとっているかの識別を行う。

(5) データの送信

識別結果と体の座標情報を、UDP 通信を使用して Unity に送信する。リアルタイムでデータを送信することで、スムーズなインタラクションを実現できる。

(6) アニメーション投影

Unity でデータを解析し、CG エフェクトを生成してプロジェクターで投影します。ユーザーのポーズに応じてエフェクトが体の適切な位置に表示される。

4. システム実装

4.1. システム実装の概要

（3 章で述べた）本システムの具体的な実装について詳細に説明する。本システムは同様の作品を筆者以外も簡単に作成できるように一部 GUI を作成している。GUI は全て Python のライブラリである flet を用いて作成した。

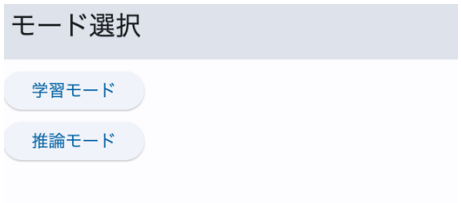

図 3 モード選択画面

Figure 3 Mode selection screen

4.2. 学習データの取得

ユーザーがポーズを取った際の Mediapipe Pose のランドマークデータを 0 番から 33 番まで順に CSV 形式で保存する。

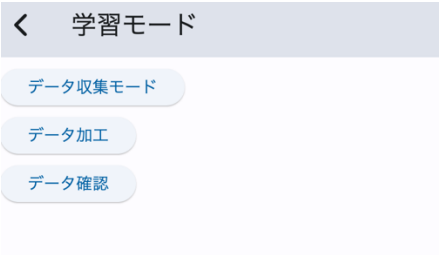

図 4　学習モードの機能選択画面

Figure 4 Learning mode function selection screen

図 5 データ収集モード画面

Figure 5 Data Acquisition Mode Screen

図 3 で学習モードを選択し、その後に表示される図 4 でデータ収集モードを選択することでデータ保存ができる画面へ遷移する。図 5 の画面で、識別したいモーションの数、これから記録するモーションの番号、起動してから自動で保存する際の撮影のインターバルと撮影回数を入力し、最後に保存する csv の名前を記入する。最後に「カメラを記録して保存を始める」ボタンを押すと Mediapipe Pose が起動され、骨格推定と、GUI で指定したデータの保存が始まる。

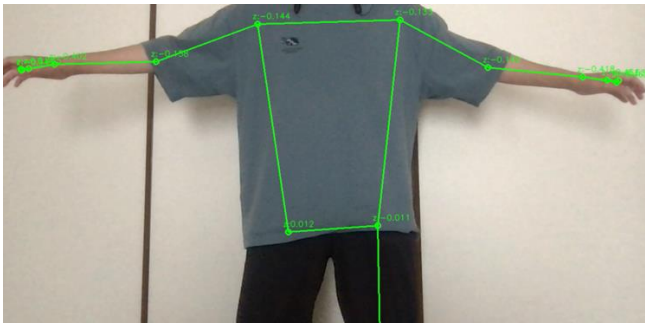

図 6 Mediapipe Pose の実行中の様子

Figure 6 Mediapipe Pose in action

この他、保存したデータが正しいポーズをとっているかを目視で確認できる機能も実装した。図 3 の「データ確認」ボタンを押すことでデータの確認をする画面へと遷移する。図 7 は確認したいデータを指定する画面である。確認したいデータが入っている csv ファイルと、登録したデータの番号を入力する。その後データの確認ボタンを押すことで、図 8 のような画面が表示される。ここでは Mediapipe Pose で定義された番号に合わせてプロットされ、保存されたポーズが視覚的に確認できる。

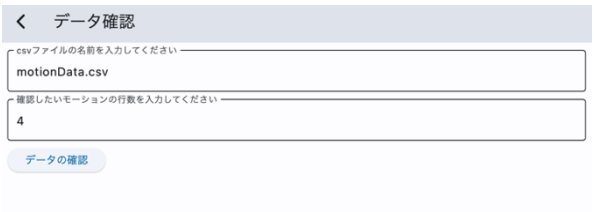

図 7　データ確認の設定画面

Figure 7 Data Confirmation Setup Screen

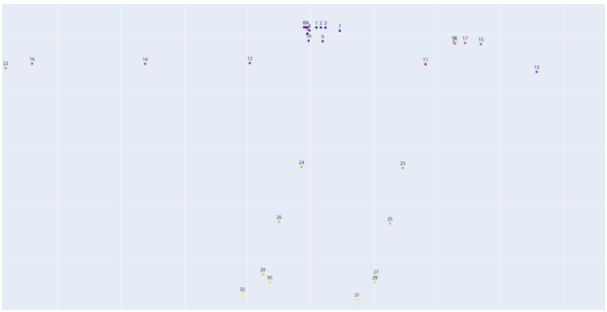

図 8　保存されたポーズデータの表示例

Figure 8 Example of saved pose data screen

4.3.　学習データの加工

収集したデータから、識別したいポーズの中で特徴的な向きをとる部分をユーザーが指定することで、ベクトルの情報に加工した新たな CSV ファイルが保存される。

図 3 で「データ加工」ボタンを押すことで図 8 の画面に遷移する。ここでは(1)で保存した csv ファイルを加工前のファイルとして入力し、出力の csv ファイルの名前を入力する。次に識別に使いたいベクトルの数を入力する。すると具体的なベクトルの入力フォームが指定した数表示されるため、それを入力する。最後に「ベクトルを元にした csv の生成」ボタンを押すことでベクトルデータに変換された csv ファイルが生成され、推論に必要な学習データが生成できる。

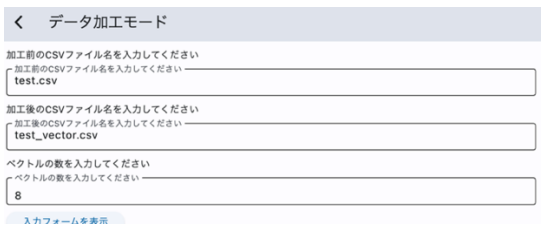

図 9 ベクトルデータへの変換画面 1

Figure 9 Conversion screen1 to vector data

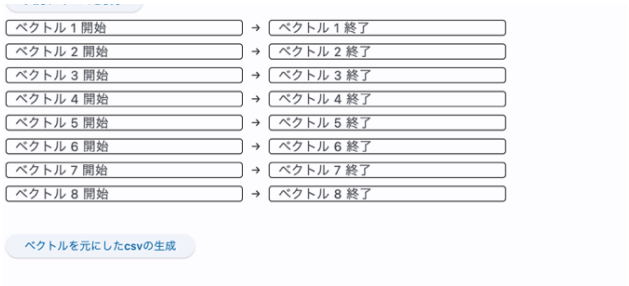

図 10　ベクトルデータへの変換画面 2

Figure 10 Conversion screen2 to vector data

4.4.　ニューラルネットワークの作成

ポーズ推定のためのニューラルネットワークは Python で利用できるライブラリである TensorFlow を使用して構築した。ベクトルデータを入力として、以下の構造のニューラルネットワークを設計した。入力層はベクトルデータで、中間層には 64 ノードと 32 ノードの ReLU 関数を使用した 2 層、出力層は各ポーズの識別結果を出力するものとした。

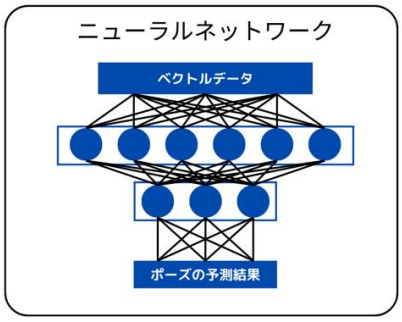

図 11　ポーズ推定のニューラルネットワーク概要図

Figure 11 Overview of neural network for pose estimation

4.5.　リアルタイムのポーズ識別

Mediapipe Pose を Python で実行すると図 11 のようなニューラルネットワークを生成して、その後すぐにカメラが起動し、リアルタイムで骨格推定が行われる。Mediapipe Pose の処理が行われる毎フレームの結果を 4.4 で作成したニューラルネットワークに代入する。

4.6.　データの送信

4.5 で出力された演算結果に加え､送信したい部位の位置を、UDP 通信を用いて送信する。図 3 で「推論モード」ボタンを押すことで図 12 の画面へ遷移する。図 11 の画面では、まず 4.3 で生成するベクトルデータの csv、Unity に送信する際に必要な IP アドレス、ポート番号、送信したい部位を示すキーポイントの番号を入力する。キーポイントの番号は図 1 で示したものである。その後に「推論を開始する」ボタンを押すと Mediapipe Pose が起動し、4.4、4,5 から 4.6 までの処理が一気に行われる。

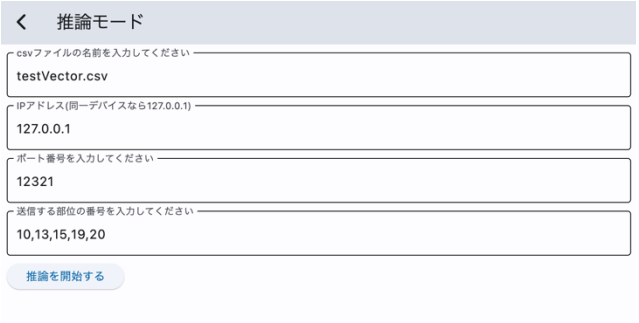

図 12 推論モード画面

Figure 12 Inference mode screen

4.7.　アニメーション投影

本システムにおけるアニメーション投影は UDP 通信を通じて行われる。UDP 通信では、文字列を送信する形式を採用しており、具体的には「AI の演算結果」 ＋ 「(指定した部位の座標)」という形でデータを送信している。この形式では、ポーズ識別の演算結果の後に指定した部位の座標が続き、それぞれの数値は「,,,」で区切られている。このデータを Unity で扱いやすい形に変換するプロセスは以下の通りである。

まず、Unity のエディター上で識別したいポーズの数を入力する。するとその要素数の float 型の配列が作成され、正しく送信されるとその配列にポーズ識別の演算結果が格納される。その後、指定した部位の数を入力する。全て正しく入力された場合、送信されたそれぞれの部位の座標は Unity の Transform 型の position に代入される。このようにして、AI の演算結果と各部位の座標データを Unity で利用しやすい形に解凍することができる。解凍したデータは、C#を用いて様々なエフェクトの管理に利用することができる。例えば、指定したポーズに応じて特定のエフェクトを表示することや、リアルタイムでポーズの変化に対応するダイナミックなアニメーションを生成することが可能である。また、プロジェクターやカメラの位置によって、投影する際の体の位置と映像の座標の対応関係が変化するため、さまざまなオフセット値を設けている。

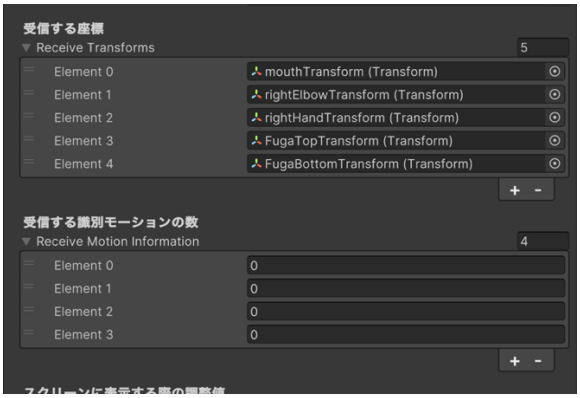

図 13 Unity のエディター上の設定の様子

Figure 13 A look at the settings on the Unity editor

5. 作品

5.1. 使用機器、投影環境

この章で紹介する作品に使用した機器と、環境について説明する。

骨格推定をするために映像を撮影するカメラや演算、Unity でのリアルタイムエフェクト生成は全て一台の MacbookPro 2019 で行った。

映像を投影するプロジェクターは Anker Nebula Capsule II を使用した。配置は図 14 の通りである。

図 14 システムの体験環境

Figure 14 System Experience Environment

5.2. 今回の作品の設定

今回、ポーズ推定の入力となるベクトルは両腕、両手、頭と肩の位置関係、上半身の向きを重視できるように設定した。

今回の作品のために識別するポーズは 4.3 節で述べる 3 作品に対応するポーズと、それ以外のポーズの 4 種類とした。それ以外のポーズには、棒立ちをしているポーズやジャンプをしたり、両手を広げたりなど、3 作品のポーズではない様々なポーズを登録した。

投影する際は、Unity 側で高さや左右の感度を合わせ、実際の体の位置と、Unity からプロジェクターを通して表示される位置を調整できるような offset 値、ポーズ推定の結果の値の閾値を調整するような機能を Unity で作成した。

5.3. 3 作品の開発

(1) 炎の矢

右手の人差し指と左手の人差し指を胸の高さまで上げて右手を後ろに引いて左手を前に出すポーズを取ることで、図 15 のように、左手に矢の先端、右手に矢尻がくるように炎を纏った矢が投影される。ここで利用している体の座標は両手の人差し指である。この距離に応じて矢の先端を表現しているオブジェクトのサイズもリアルタイムで変わる。

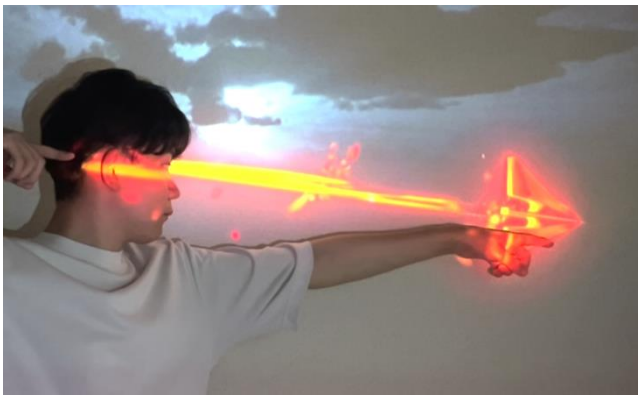

図 15 「炎の矢」の投影の様子

Figure 15 Projection of "Arrow of Fire”

(2) 氷の息

右手を口の近くに添えて少しかがむことで、図 16 のように口から氷をまとった息を吹いているようなエフェクトが投影される。ここで氷の向きは手首から人差し指に向かうベクトルの方向としている。

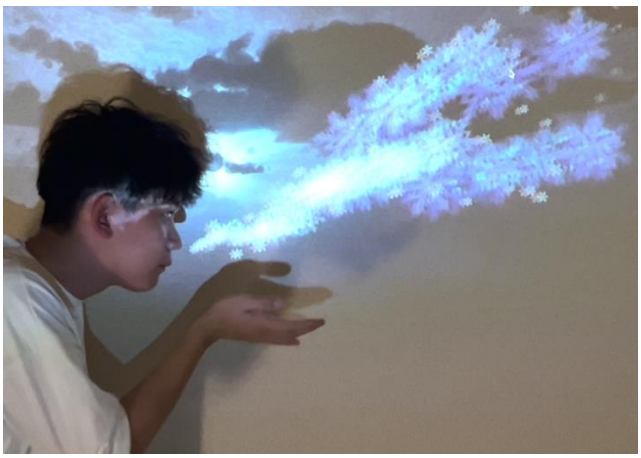

図 16 「氷の息」の投影の様子

Figure 16 Projection of "Breath of Ice”

(3) 両手からビーム

両手を前に伸ばして手をあわせることで、図 17 のように両手から赤いビームを出しているようなエフェクトが投影される。ここでのビームの向きは、右手の肘から手首に向かうベクトルの方向としている。

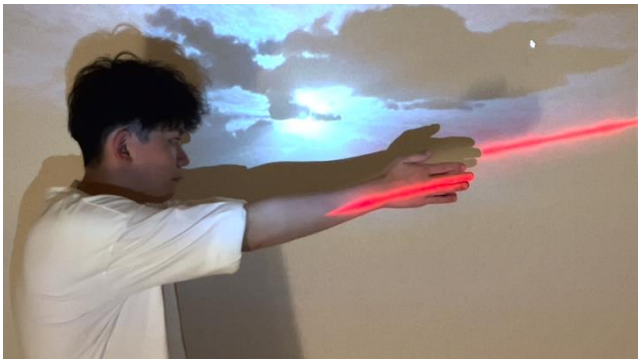

図 17 「両手からビーム」の投影の様子

Figure 17 Projection of "Beam from Both Hands”

6. 骨格識別の性能評価

6.1. 概要

第 5 章で紹介した作品を作成するために作成したデータセットを用いて、開発した骨格推定システムについて評価を行った。具体的には、Mediapipe のランドマークを全て入力とした演算と、4.2 節で説明したベクトルデータを入力とした演算で、識別の精度について比較した。選定したベクトルが今回の作品におけるポーズの識別にとって適切だったかどうかを評価した。

6.2. 評価手法と指標

4.4 節で説明したものと同じ構造のニューラルネットワークを作成する。今回の検証では、Mediapipe Pose のランドマークの値がそのまま入ったデータセット、それを 4.3 節のように加工したデータセット、それぞれ 278 データ用意した。まず、このうちの 8 割を、乱数を用いてランダムに選び、学習データとしてニューラルネットワークの学習に使用する。作成したニューラルネットワークが、残りの 2 割のデータを予測し、その値を用いて評価を行う。

今回は精度を用いる。精度はモデルが正しい予測をした割合であり、0 から 1 の範囲で、1 に近づくほど精度が良いと評価する。この実験を 5 回ずつ行い、2 つのデータセットの精度の差を比較する。

6.3. 結果と考察

Mediapipe Pose のランドマークを用いた場合の平均精度は 0.95714 であった。一方、ベクトルデータを用いた場合の平均精度は 0.9696 であり、わずかに高い結果となった。

どちらも高い精度を示したことから、作成したニューラルネットワークのモデルは今回の作品のポーズ推定を良いレベルで行っていると考える。また、Mediapipe Pose のランドマークをそのまま用いた場合、入力に使われる数字の数は 99 であるが、ベクトルデータは 39 であった。こちらで必要なデータを考えるという手間はかかるが、演算数を減らし、精度が落ちることがないということがわかったため、ベクトルデータへの変換は有効であると言える。

7. まとめと今後の展望

本研究では、体の動きに応じた特殊攻撃アニメーション投影システムの開発を行った。特定の姿勢推定と体の動きの認識には、Mediapipe と TensorFlow を使用し、アニメーションの投影には Unity を用いた。AI に詳しくないユーザーでも容易に使用できるように GUI を実装し、プロジェクションマッピング環境を構築した。本研究では、体の動きに応じた特殊攻撃アニメーション投影システムの開発に成功し、3 作品を作成した。

これらの作品に対して、ポーズの識別の性能評価を行った結果、Mediapipe のランドマークを用いた場合の平均精度は 0.95714 であり、ベクトルデータを用いた場合の平均精度は 0.9696 であることが確認された。ベクトルデータの使用により、入力データの数を減少させつつ、精度が向上することがわかった。

今回は筆者自身のデータのみを用いてシステムを構築したため、他のユーザーのデータを用いた際の精度については検証が不十分である。将来的には、異なるユーザーのデータを収集し、システムの汎用性と精度を確認することが重要である。

ポーズ推定にはニューラルネットワークを用いたが、他の手法（例えば、決定木やサポートベクターマシン）も検討する価値がある。これにより、ポーズ推定の精度やリアルタイム性をさらに向上させる可能性がある。

また、カメラを用いたモーションキャプチャは、プロジェクションマッピングの映える非常に暗い環境ではうまく動作しないことが判明した。光学式モーションキャプチャや LiDAR などのセンサーを使用することで、非常に暗い環境下でもこのシステムが動作し、さらに幅広い写真映えを狙ったコンテンツを作成できると考える。また、Unity 部分での処理の工夫をすることにより、よりインタラクティブなゲーム性を持たせることも考えられる。

本研究では、写真を撮ることができるような映えるコンテンツを目指したが、今後はよりインタラクティブ性を重視したコンテンツの開発も検討する。例えば、ユーザーの動きに応じてゲームのようなフィードバックを提供することで、さらなる没入感を与えることができる。

最後に、本研究の成果を基に、より多様な体験を提供するためのシステムの改良と、ユーザーのフィードバックを取り入れたコンテンツの開発を進めていくことが今後の課題である。

参考文献

1) 増子美穂: 没入型デジタルアートと芸術体験についての一考察, 観光学研究 第 19 号(2020)
2) Mocopi 公式ウェブサイト
https://www.sony.net/Products/mocopi-dev/jp/
3) 末吉知樹: 葉や花を対象とした動的プロジェクションマッピングの自動生成 九州大学学術情報リポジトリ (2022)
4) Mediapipe ソフトウェア情報
https://ai.google.dev/edge/mediapipe/solutions/guide?hl=ja
5) 飯野 健弘: 骨格認識を用いたクラシックギター演奏時における姿勢の評価手法の提案, JAIST 学術研究成果 リポジトリ (2023)
6) Mediapipe GitHub リポジトリ
https://github.com/google-ai-edge/mediapipe/blob/master/docs/solutions/pose.md
7) 株式会社ネイキッド: 世界遺産・法隆寺でネイキッドのデジタルアートを開催
https://prtimes.jp/main/html/rd/p/000000931.000008210.html