

Title	カメラ画像を用いた体幹トレーニングの姿勢支援手法の提案
Author(s)	綿谷, 惇史
Citation	
Issue Date	2019-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/15832
Rights	
Description	Supervisor:宮田 一乗, 先端科学技術研究科, 修士 (知識科学)

修士論文

カメラ画像を用いた
体幹トレーニングの姿勢支援手法の提案

1710236

綿谷 惇史

主指導教員 宮田 一乗
審査委員主査 宮田 一乗
審査委員 永井 由佳里
西本 一志
由井蘭 隆也

北陸先端科学技術大学院大学

先端科学技術研究科 [知識科学]

平成 31 年 2 月

目次

第1章 はじめに	1
1.1 研究背景	1
1.2 研究目的	2
1.3 本論文の構成	2
第2章 関連研究	3
2.1 2次元姿勢推定	3
2.2 3次元姿勢推定	3
2.3 深度カメラを用いたトレーニング支援	4
2.4 複数のカメラを用いたトレーニング支援	5
2.5 本研究の位置づけ	6
第3章 画像からの3次元モデルの生成	7
3.1 手法	7
3.1.1 OpenPose	7
3.1.2 SMPL: A Skinned Multi-Person Linear Model	10
3.1.3 HMR: Human Mesh Recovery	13
3.2 3次元モデル生成	14
第4章 姿勢支援手法	17
4.1 システム概要	17
4.2 レンダリング条件	19
4.3 視覚的フィードバックの生成	19
4.4 レンダリングの高速化	27
4.5 実装環境	30
4.6 User Interface	31
第5章 実験	38
5.1 実験概要	38
5.2 実験環境	43
5.3 実験結果	44
5.4 考察	51
第6章 おわりに	51
6.1 まとめ	51
6.2 今後の展望	51
謝辞	51
参考文献	51

目次

図 2.1 見本動作の提示 [20]	4
図 2.2 フィードバック映像例 [21]	5
図 2.3 HMD に表示する支援映像 [22]	5
図 3.1 multi-stage CNN [16]	7
図 3.2 Part Confidence Map [16]	8
図 3.3 Part Affinity Fields [16]	8
図 3.4 Body Model による推定関節数の違い	9
図 3.5 出力の例 (Body Model : COCO)	9
図 3.6 SMPL	11
図 3.7 SMPL 各パラメータによる変化および男性モデル	13
図 3.8 デモコード実行結果	13
図 3.9 バウンディングボックスの例	14
図 3.10 入出力画像	15
図 3.11 人物番号画像	15
図 3.12 バウンディングボックスに沿って正規化した画像 (単位[pixel])	16
図 4.1 システムの流れ	18
図 4.2 身体の各部位とメッシュ頂点对応付けに用いた画像	21
図 4.3 重ね合わせた画像	21
図 4.4 画像座標の始点と終点	22
図 4.5 各部位に対応したメッシュ頂点番号	24
図 4.6 メッシュ頂点番号を用いたマーカ表示	25
図 4.7 (11)式を用いたマーカ表示	26
図 4.8 トレーニング前のウィンドウ	27
図 4.9 トレーニング中のウィンドウ	27
図 4.10 レンダリング速度比較に用いた画像	28
図 4.11 レンダリング速度比較結果 (左 : OpenDR, 右 : PyOpenGL)	29
図 4.12 起動画面	32
図 4.13 ファイル選択画面	32
図 4.14 ファイル選択完了画面	33
図 4.15 3次元モデル表示画面	33
図 4.16 視点変更	34
図 4.17 トレーニング画面	35
図 4.18 視点の設定例	35
図 4.19 目標姿勢表示画面	35

図 4.20 各表示ウィンドウ	36
図 4.21 トレーニング中の画面	37
図 5.1 UI および提示例（手法 1）	38
図 5.2 UI および提示例（手法 2）	39
図 5.3 体幹トレーニング 1	39
図 5.4 体幹トレーニング 2	40
図 5.5 体幹トレーニング 3	40
図 5.6 体幹トレーニング 4	40
図 5.7 アンケート用紙	42
図 5.8 実験風景の例	43
図 5.9 首より上のメッシュ頂点	44
図 5.10 被験者 A	49
図 5.11 被験者 B	49
図 5.12 被験者 C	49
図 5.13 被験者 D	50
図 5.14 被験者 E	50
図 5.15 被験者 F	50
図 5.16 被験者 G	51
図 5.17 被験者 H	51

表目次

表 1 SMPL サンプルコードレンダリング条件.....	10
表 2 各人物の最大値, 最小値, 距離.....	15
表 3 レンダリング条件.....	19
表 4 身体の各部位の画像座標とメッシュ頂点番号.....	23
表 5 レンダリング速度 (OpenDR) [s].....	30
表 6 レンダリング速度 (PyOpenGL) [s].....	30
表 7 被験者における手法と体幹トレーニングの組み合わせ.....	41
表 8 実験機材.....	43
表 9 被験者 A.....	45
表 10 被験者 B.....	45
表 11 被験者 C.....	46
表 12 被験者 D.....	46
表 13 被験者 E.....	47
表 14 被験者 F.....	47
表 15 被験者 G.....	48
表 16 被験者 H.....	48
表 17 アンケート結果.....	51
表 18 RMSE (初期値と最小値の差) [%].....	51
表 19 RMSE (最小値に達する時間) [s].....	51
表 20 アンケート結果 (平均値).....	51

第1章

はじめに

本章では、はじめに研究の背景について述べる。つづいて、本研究の目的を説明し、最後に、論文の構成について記述する。

1.1 研究背景

近年、深層学習を用いて、ひとつの画像から姿勢を推定する研究 [4-7]が多く行われている。本研究では、深層学習を利用して姿勢を推定する研究 [5] [7]を応用し、カメラ画像を用いた体幹トレーニングの姿勢支援の提案を行う。

体幹とは、身体の四肢と頭部を除いた重量約 48%を占める非常に大きな部位であり、四肢間の運動連結やバランスに関して重要な役割を果たしている。そのため、体幹筋の働きを重視したトレーニング方法をスポーツ選手が行っており、その重要性が一般の方にも注目されるようになってきている。そして、この体幹を鍛える体幹トレーニングは、Worldwide Survey of Fitness Trends のフィットネス流行予測において、2007 年から 2010 年まで、トップ 5 にランクインしており、2010 年以降もずっとランクインしている [1]。このことから、体幹トレーニングが世界的に注目されていると言える。

この体幹トレーニングは一般的に個人で行うことができ、体幹筋を中心とした全身の筋肉を鍛えることができる実用的かつ簡単なトレーニング手法である。そして、その効果を最大限に発揮するには、正しい姿勢を保つことが極めて重要である [2]。正しくない姿勢で行うトレーニングは怪我につながる可能性がある。しかしながら、個人がトレーニング中の姿勢を把握するのは難しい。そのため、鏡やカメラで撮影した映像を通じて姿勢を把握する方法があるが、把握した姿勢が正しいかどうかを判断することも難しい。このことから、体幹トレーニングでの姿勢支援は有効であると言える。

トレーニング支援手法として、姿勢推定を行い、ユーザにフィードバックし、正しい動作、姿勢でのトレーニングを促す研究 [9-11]がある。それらの姿勢推定には、深度カメラや複数台のカメラを必要とするため、コストがかかる。また、ユーザへの視覚的フィードバックが、1 視点および骨格情報のみを示している 2 次元ボーン画像であるため、姿勢の把握が難しい。そこで、本研究では、単一 RGB カメラのみを用いて、カメラ画像から姿勢推定を行う。そして、推定結果に基づいて 3 次元モデルを生成し、2 視点でユーザへ視覚的フィードバックをす

ることで、体幹トレーニングの姿勢支援を行う。2 視点である理由は、3 次元空間の情報を複数視点で提供することでユーザの空間把握を促進する研究 [3] があり、姿勢の把握にも複数視点で行うことで、把握しやすさが向上するのではないかと考えたからである。しかしながら、視点数が多すぎると情報過多になってしまう可能性があるため 2 視点とする。

1.2 研究目的

本研究の目的は、単一 RGB カメラのみを用いた姿勢支援システムの提案である。ユーザが姿勢を把握しやすい視覚的フィードバックをすることで、姿勢の補正を容易に行えるシステムを目指す。画像から姿勢推定を行い、トレーニングの目標姿勢と現在の姿勢の 3 次元モデルを生成し、重畳表示する。さらに、目標姿勢と現在の姿勢の違いを把握しやすくするために、身体の各部位 10 箇所（両手および肘、肩、膝、足首）にマーカを表示し、誤差に応じて、4 段階に色変化させる。これらの情報を 2 視点でユーザへ視覚的フィードバックをすることで、姿勢補正の支援を行う。このシステムを利用することにより、個人でトレーニングをする際、トレーナーのいるジムに行かなくても正しい姿勢でトレーニングができる。

1.3 本論文の構成

本論文は、全章で構成する。第 2 章では、関連研究について述べ、本研究の位置づけを明らかにする。第 3 章では、本研究で提案する姿勢支援に用いる画像から 3 次元モデルの生成について説明し、第 4 章では、提案する姿勢支援システム、第 5 章では、評価実験の結果と考察を示す。最後に第 6 章で本研究を総括し、今後の課題について述べる。

第2章

関連研究

本章では、はじめに、単一の画像から 2 次元および 3 次元の姿勢推定を行っている研究について説明する。次に、トレーニングの姿勢支援を行っている研究を深度カメラ、複数台のカメラに分けて、説明する。最後に、本研究の位置づけを述べる。

2.1 2次元姿勢推定

Newell ら [4]は、**Stacked Hourglass** という新しいネットワークを提案した。このネットワークは、画像中の特徴をすべてのスケールで処理し、身体に関する関係性を捉えるために連結され、ボトムアップとトップダウン処理を行うことで、性能を向上させている。

Cao ら [5]は、**multi-stage CNN** と呼ばれる手法を提案した。この手法はボトムアップ処理で行う。入力画像から身体の各部位が存在する位置と身体の各部位間のつながり得る可能性を表す 2D ベクトルを算出する。その後、同じ人物の部位を組み合わせることで、高精度で体の姿勢を推定できている。

2.2 3次元姿勢推定

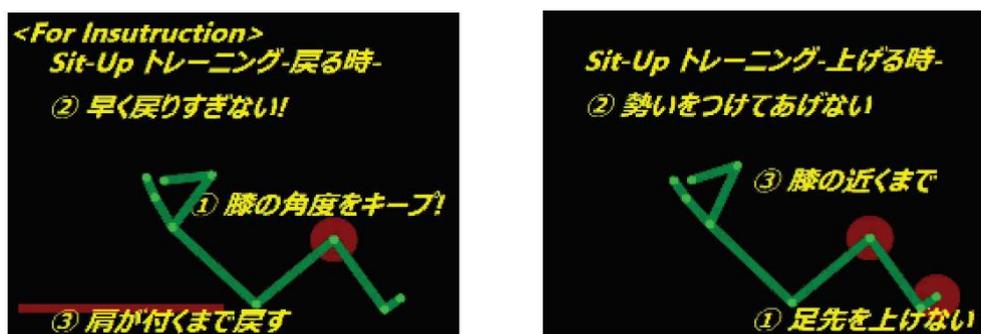
Rogez ら [6]は 2 次元および 3 次元の姿勢推定のための **end-to-end** アーキテクチャを提案した。この手法は、画像ごとにいくつかのポーズ生成し、それを採点することであり、ポーズを生成する生成器とポーズを採点する分類器、ポーズをより **ground-truth** に近づけるための回帰子で構成されている。

Kanazawa ら [7]は人間の 3 次元メッシュを推定する **end-to-end** のフレームワークを提案した。この手法は、**GANs** [8]の手法を用いて、不自然な関節の曲がり方をしない角度制限や、細すぎる体など不自然な体型にしない制限などを学習することで、より自然な 3 次元モデルを生成することを可能にしている。

2.3 深度カメラを用いたトレーニング支援

高久ら [9]は筋力トレーニングの中でも比較的簡易で認知度の高い腹筋運動（シットアップ）を対象に支援を行っている。この研究は、深度情報とカラー画

像を同時に取得できる Kinect を用いてユーザの筋力トレーニングの動作を検出する。そして、検出したトレーニング動作を速度、角度の観点から支援を行っている。また、ユーザのモチベーションを高めるため、トレーニングによる効果やゲーム性を付加している。見本動作の提示には図 2.1 に示すように骨格情報のみを示しているボーン画像を用いており、付加情報として、推定消費カロリーを表示している。



(a) 上体を戻す時 (b) 上体を上げる時

図 2.1 見本動作の提示 [9]

また、岡本ら [10]は陸上競技者を対象としたハードルまたぎ練習の支援システムを提案している。この研究では、深度カメラである Kinect で取得した深度情報をもとにユーザの関節位置を推定し、ユーザの姿勢を取得する。そして、取得した姿勢をもとに適切かどうかを判断し、その結果からフィードバック情報を提示する。そのフィードバック情報は図 2.2 のように撮影している映像に重畳表示させ、文字やマーカなどで情報を示している。このシステムでは、マーカは正しい姿勢、誤った姿勢、訓練上目標となっている位置をそれぞれ指定している色で示しており、文字は事前に支援システムのプロトタイプを作成し、体験した陸上競技者のアンケート調査から数単語に限定して表示している。

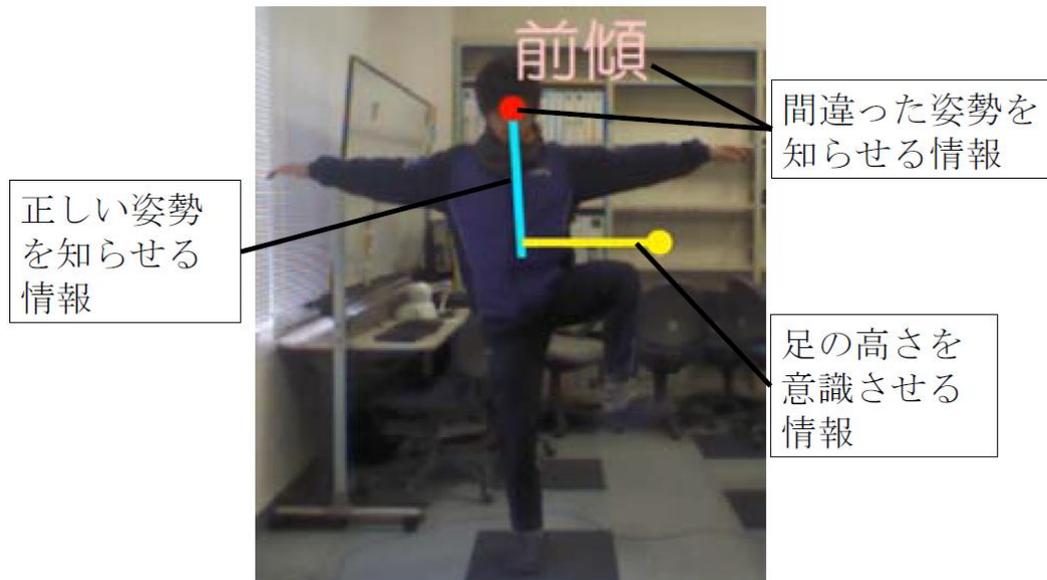


図 2.2 フィードバック映像例 [10]

2.4 複数のカメラを用いたトレーニング支援

松村ら [11]は、2 台の Playstation®Eye を用いて作成した 2 眼カメラを使って、弓道の練習を対象としたフォーム改善システムを開発した。この研究では、図 2.3 のように事前にお手本動画を取得し、半透明の動画にする。さらに、リアルタイムで撮影した映像と半透明にしたお手本動画を重畳した映像を Head Mounted Display に表示する。

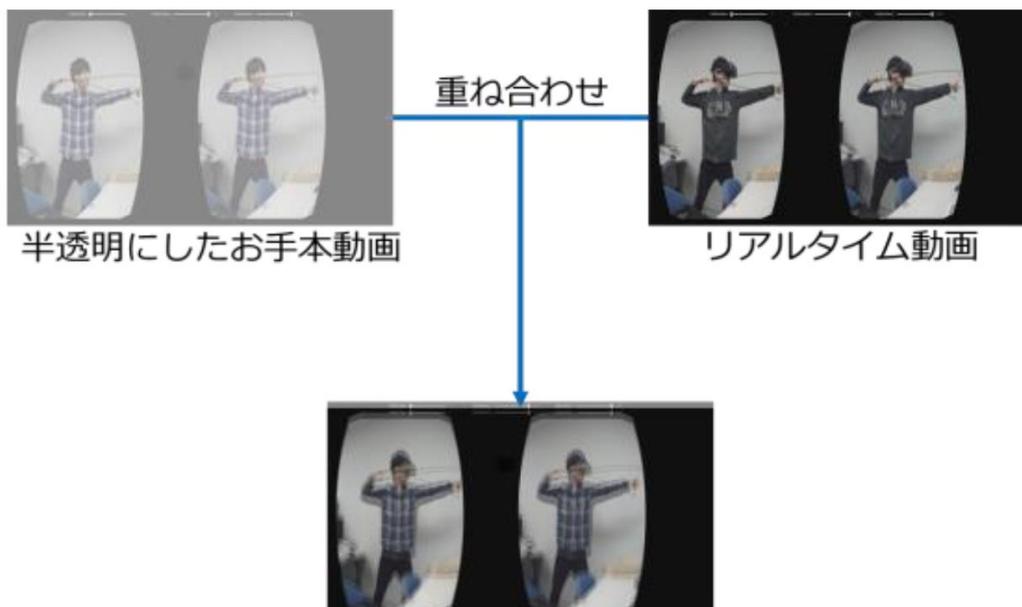


図 2.3 HMD に表示する支援映像 [11]

2.3 本研究の位置づけ

以上で述べたトレーニング支援の関連研究では、深度カメラや複数台のカメラを用いてトレーニング支援を行っている。しかしながら、それらはコストがかかるため、本研究では、汎用性が高い単一の RGB カメラのみを用いて支援を行う。提案するシステムは、事前にトレーニング画像からトレーニング姿勢を推定する。この手法の利点は、参考書やインターネットなどの画像でも、トレーニングの姿勢を取得できることである。すなわち、多種目への応用が容易にできる。

トレーニング中の姿勢のフィードバックに関して、トレーニング支援の関連研究では、ボーン画像や撮影した映像に注釈、お手本動画と撮影した映像の重畳などであり、ユーザの視点が固定されている。ユーザが見たい視点からの映像を提供することにより、姿勢を把握しやすくなるのではと考え、本研究では、カメラ画像から姿勢推定を行い、3次元モデルを生成し、ユーザが見たい視点で見ることができる映像を提供することとした。

第3章

画像からの3次元モデルの生成

本章では、姿勢支援に用いる画像から3次元モデルの生成する手法について説明する。はじめに、本研究で用いる手法について述べる。次に、本システムの画像から3次元モデルの生成の流れについて説明する。

3.1 手法

本節では、本システムで用いた画像から3次元モデルを生成する手法であるOpenPose, SMPL, HMRについて述べる。本システムでOpenPose, SMPL, HMRはそれぞれ、バウンディングボックスの取得, 3次元モデルの生成, 姿勢推定に用いた。これらの詳しい説明は3.2節で述べる。

3.1.1 OpenPose¹

ZCaoら[5]は、図3.1のようなmulti-stage CNNと呼ばれる手法を提案した。この手法は、身体の一部が存在する位置をヒートマップとして表すPart Confidence Map (図3.2)を予測するネットワークと、身体の一部間のつながり得る可能性を表すベクトルマップPart Affinity Fields (PAFs) (図3.3)を予測するネットワークで構成されており、最初の4ステージはPAFsを予測し、最後の2ステージはPart Confidence Mapを予測する。それぞれの予測結果は入力データと連結して、次のステージに入力する。

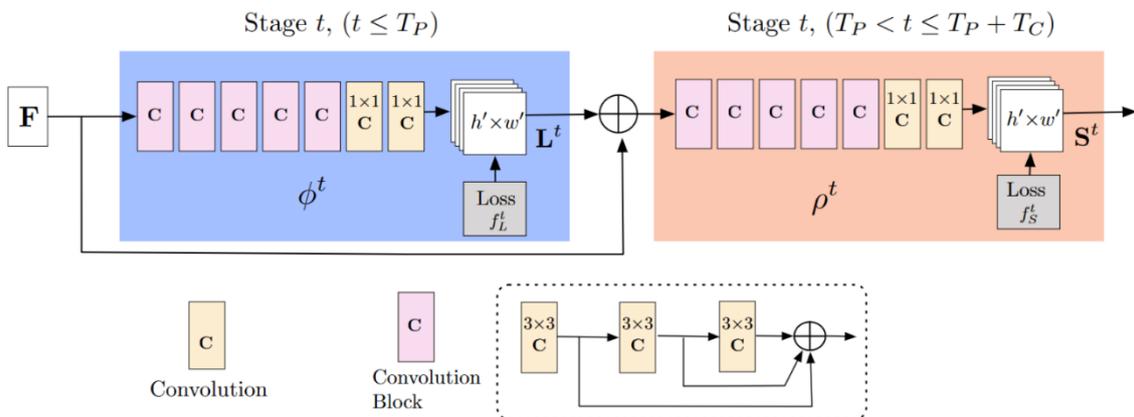


図 3.1 multi-stage CNN [5]

¹ Openpose (Github) <https://github.com/CMU-Perceptual-Computing-Lab/openpose>



図 3.2 Part Confidence Map [5]

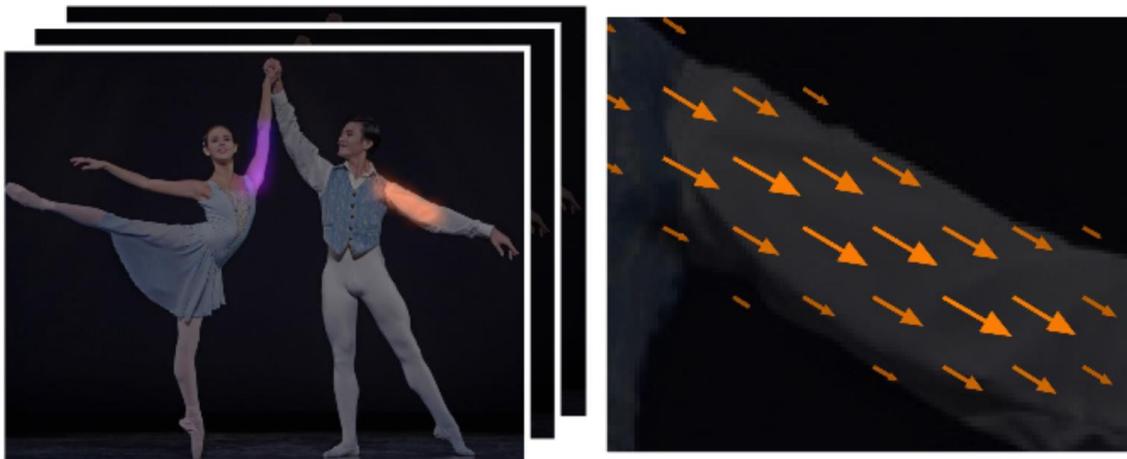


図 3.3 Part Affinity Fields [5]

学習データには、MPII データセット [12], COCO データセット [13], COCO データセットに足部の 6 箇所をキーポイントを追加したデータセットを用いている。それらのデータで学習したモデルは、コード¹内で **Body Model** と定義されていて、それぞれ、MPI, COCO, BODY_25 となっており、推定可能な関節の数が異なる。図 3.4 に、それぞれの **Body Model** を用いて、推定を行った結果を示す。また、出力として、各関節位置座標 ($x[\text{pixel}]$, $y[\text{pixel}]$) と各関節における信頼度(0~1)を得ることができる。図 3.5 に COCO を用いて取得した出力の例を示す。

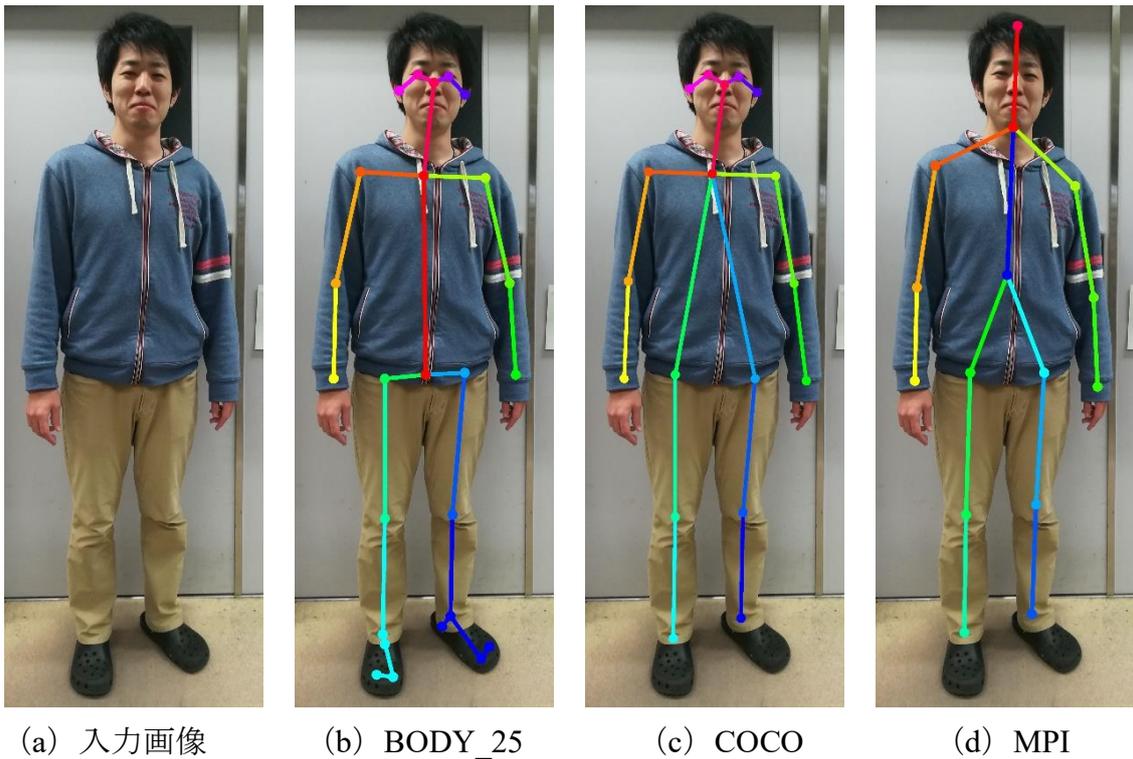


図 3.4 Body Model による推定関節数の違い

```

%YAML:1.0
pose_0: !!opencv-matrix
  rows: 1
  cols: 18
  dt: "3f"
  data: [ 6.97268311e+02, 4.00156586e+02, 8.91805887e-01,
          6.35905334e+02, 8.51399780e+02, 8.99787903e-01, 3.18099518e+02,
          8.40593628e+02, 7.82836318e-01, 2.15614288e+02, 1.39440479e+03,
          7.77973890e-01, 1.94957718e+02, 1.88633789e+03, 8.37646008e-01,
          9.53485535e+02, 8.61557434e+02, 8.89910996e-01, 1.04577344e+03,
          1.40456995e+03, 7.54416466e-01, 1.10717834e+03, 1.89655835e+03,
          7.79823899e-01, 4.51153595e+02, 1.86562817e+03, 6.43704236e-01,
          4.51069946e+02, 2.58352271e+03, 8.13172281e-01, 4.41271942e+02,
          3.19800732e+03, 7.97447503e-01, 8.51483459e+02, 1.88619080e+03,
          6.57688558e-01, 7.89653503e+02, 2.56259009e+03, 8.11709166e-01,
          7.79692810e+02, 3.09560571e+03, 8.43494296e-01, 6.15984680e+02,
          3.48568268e+02, 9.14498866e-01, 7.69344238e+02, 3.58800964e+02,
          9.42902446e-01, 5.23106140e+02, 4.20549866e+02, 9.00994301e-01,
          8.60920227e+02, 4.41179413e+02, 9.65280890e-01 ]

```

図 3.5 出力の例 (Body Model : COCO)

3.1.2 SMPL: A Skinned Multi-Person Linear Model²

SMPL とは, Loper ら [14]が提案した人間の様々な姿勢や体形を表現することができる人体モデルである. SMPL は, $N = 6,980$ の頂点をもつ三角メッシュ $M(\theta, \beta) \in \vec{R}^{3 \times N}$ で表現する. Pose $\theta \in \vec{R}^{3 \times K + 3}$ は 3 次元の回転であり, 関節 $K = 23$ の軸角度と全体の回転で表す. すなわち, θ は $3 \times K + 3 = 3 \times 23 + 3 = 72$ 個のパラメータを有している. Shape $\beta \in \vec{R}^{10}$ は, 主成分分析を用いて 10 次元で表された形状空間のパラメータである. 図 3.6 に頂点, メッシュ, θ , β を代入した SMPL を示す. また, 図 3.7 にサンプルコード³の SMPL の各パラメータを変え, レンダリングした結果を示す. レンダリングには OpenDR⁴ [15]を使用しており, 表 1 のサンプルコード通りのレンダリング条件で行った.

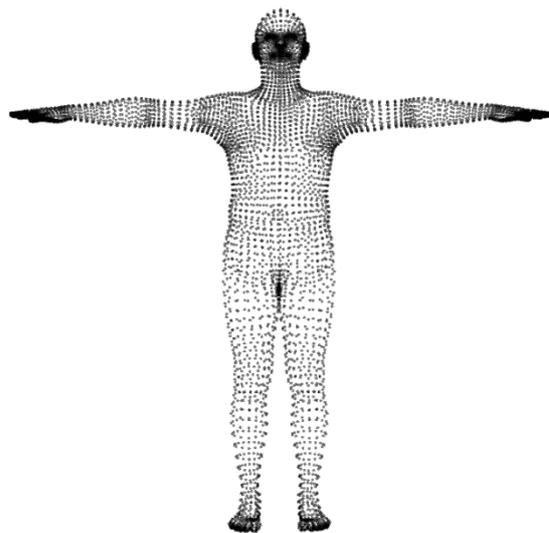
表 1 SMPL サンプルコードレンダリング条件

画像サイズ (横, 縦)	光学中心(c_x, c_y)	焦点距離 f
640, 480	320, 240	320

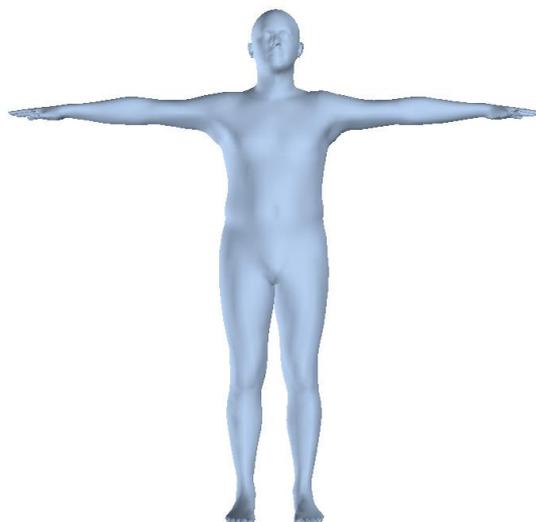
² SMPL (Project Page) <http://smpl.is.tue.mpg.de/>

³ SMPL (Project Page)²→Downloads→SMPL for Python Users→smpl→smpl_webuser→hello_world→render_smpl.py

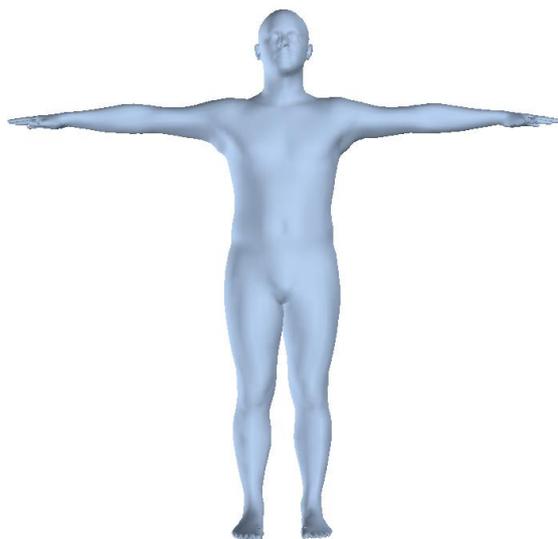
⁴ OpenDR (Github) <https://github.com/mattloper/opendr/wiki>



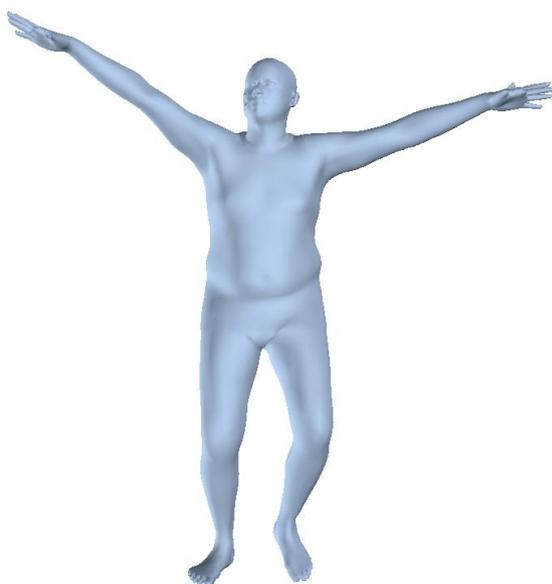
(a) $N = 6,980$ の頂点



(b) $M(0,0)$

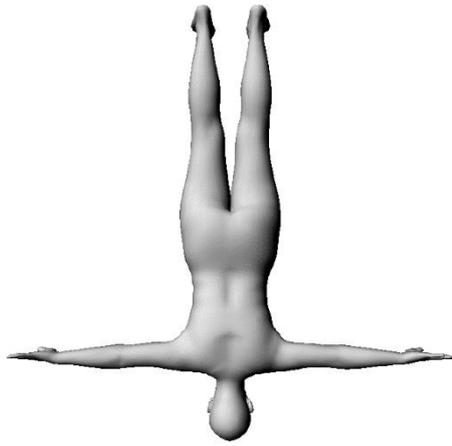


(c) $M(0,\beta)$

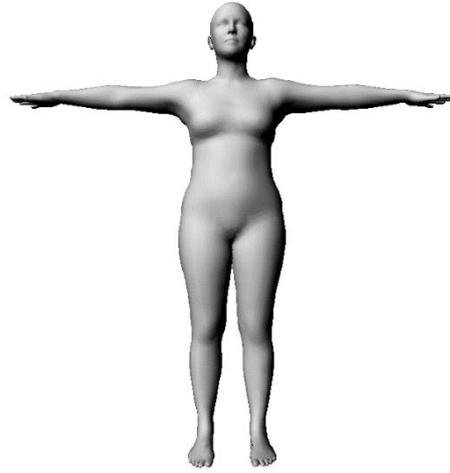


(d) $M(\theta,\beta)$

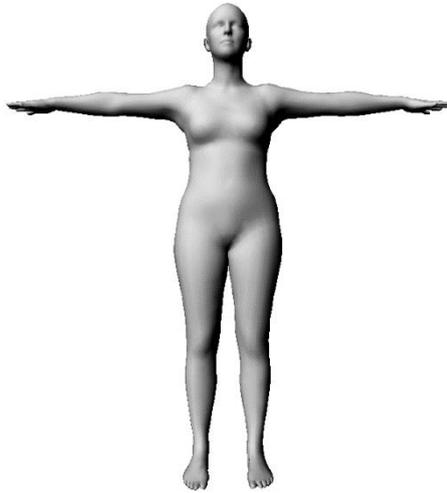
図 3.6 SMPL



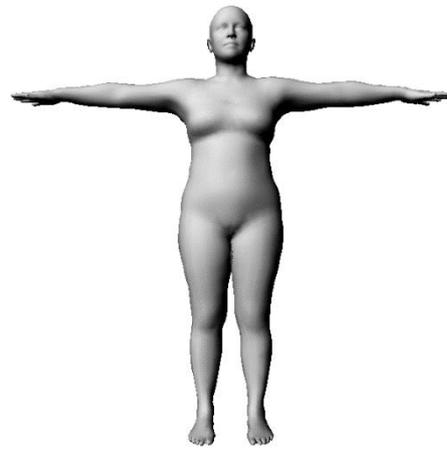
(a) $\theta_n = 0$ ($n = 1, \dots, 72$)



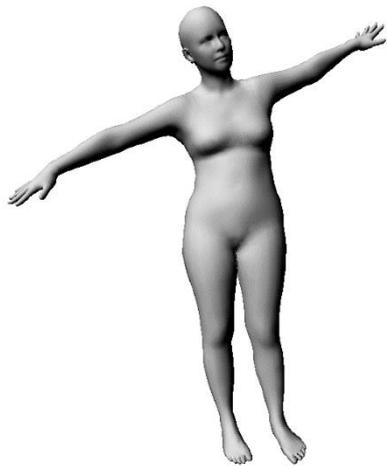
(b) $\theta_n = \begin{cases} \pi & (n = 1) \\ 0 & (\text{otherwise}) \end{cases}$



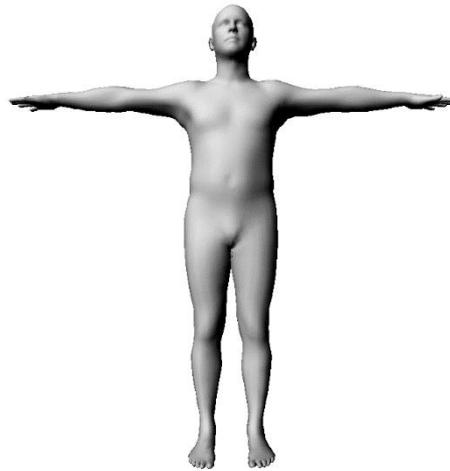
(c) $\beta_n = 1$ ($n = 1, \dots, 10$)



(d) $\beta_n = -1$ ($n = 1, \dots, 10$)



(e) $\theta_n = 0.08$ ($n = 1, \dots, 72$)



(f) 男性モデル

図 3.7 SMPL 各パラメータによる変化および男性モデル

3.1.3 HMR: Human Mesh Recovery⁵

HMR は, Kanazawa ら [7]が提案した, 単一の RGB 画像から人体モデルを再構成する手法である. 画像から 3D 回帰モジュールによって推測されたパラメータ (カメラの回転, 移動, スケールの 3 つと SMPL のパラメータである Pose θ , Shape β) で生成された人体モデル (SMPL) を不自然な関節の曲がり方をしない角度制限や, 細すぎる体など不自然な体型にしない制限などを学習した識別器によって自然な人体モデルであるかを判別する. 自然なモデルであると判断されたモデルは出力され, 違う判断された場合は, もう一度 3D 回帰モジュールによってパラメータを推測する. レンダリングには SMPL と同様に OpenDR を用いている. Github⁶では, 訓練が行えるコードがあるが, 本研究では, 配布されている事前に訓練された学習モデルを用いる.

図 3.8 にデモコード⁷を実行した結果を示す.

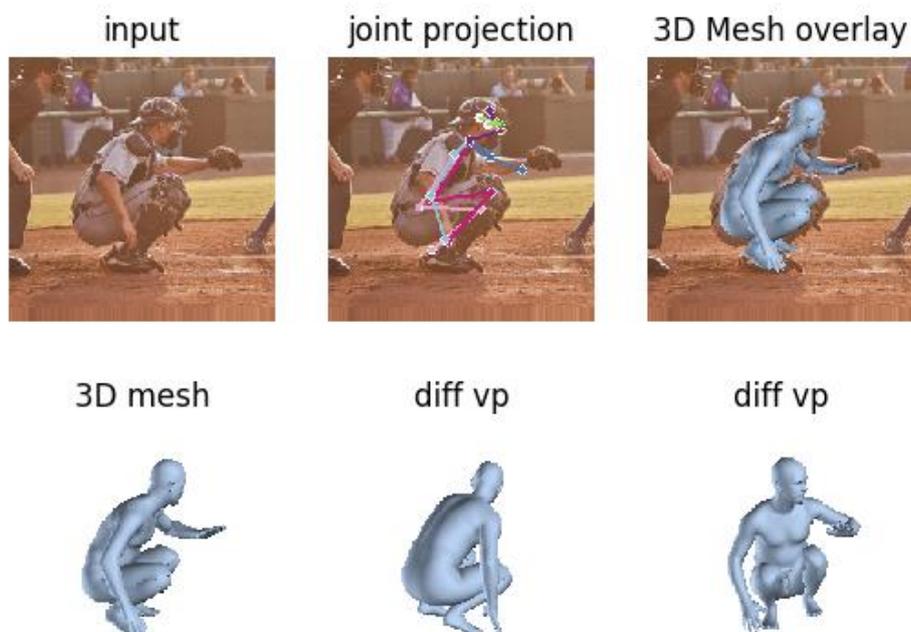


図 3.8 デモコード実行結果

⁵ HMR (Project Page) <https://akanazawa.github.io/hmr/>

⁶ HMR (Github) <https://github.com/akanazawa/hmr>

⁷ HMR (Github) <https://github.com/akanazawa/hmr/blob/master/demo.py>

3.2.3 次元モデル生成

本システムでは、画像に対して姿勢推定を行う前に、画像の正規化を行う。正規化を行う理由は、姿勢推定で用いる HMR の学習データが以下の2つの条件に統一されているためである。

- 1)人物のバウンディングボックスの対角がおよそ 150pixel
- 2)縦横それぞれ 224pixel

ここで、バウンディングボックスとは、対象とする物体を含む矩形であり、図 3.9 に示すように本研究では、人を含む矩形である。

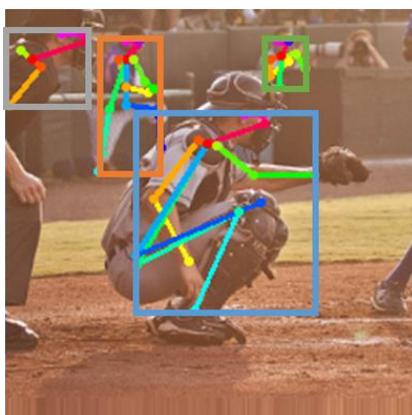


図 3.9 バウンディングボックスの例

正規化を行う際、人物のバウンディングボックスを用いる。バウンディングボックスの取得には、OpenPose で得られる各関節の位置座標を用いる。信頼度 0 の点を除いた関節の位置座標 (x, y) のそれぞれに対し最大値と最小値を求めることで、バウンディングボックスを得る。ここで、画像内に複数人いる場合、このバウンディングボックスの対角距離が一番大きい人物（画像内で一番大きく映っている人）を姿勢推定の対象とする。

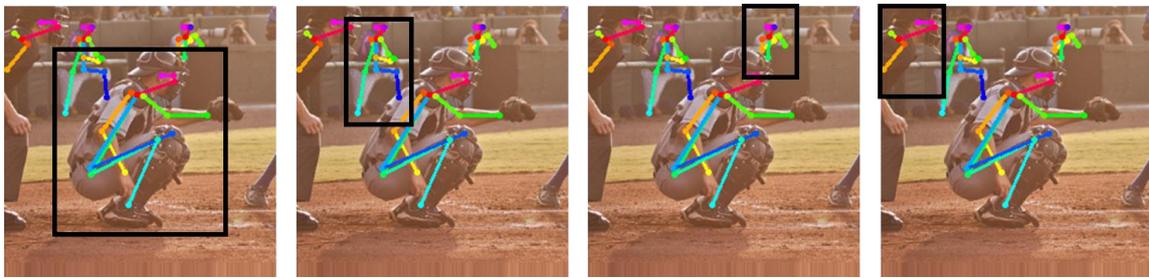
例として、図 3.10 の入力画像を使い、姿勢推定の対象人物を求めてみる。図 3.11 に各人物番号、表 2 に各人物の画像座標における最大値 (x_{max}, y_{max}) および最小値 (x_{min}, y_{min}) 、その差分の距離であるバウンディングボックスの対角距離 d を示す。

この処理において、Openpose の Body Model には、COCO を用いる。COCO より BODY_25 の方が、推定できる関節数が多いが処理は重く、また、得る情報はバウンディングボックスを得るために使用のみであり、推定可能な関節数よりも処理速度を重視するため、COCO を用いた。



(a) 入力画像 [13] (b) Openpose 出力画像

図 3.10 入出力画像



(a) 1 (b) 2 (c) 3 (d) 4

図 3.11 人物番号画像

表 2 各人物の最大値, 最小値, 距離

	1	2	3	4
x_{max}, y_{max}	167, 164	84, 88	162, 42	44, 52
x_{min}, y_{min}	70, 57	50, 17	143, 17	2, 13
d	144	79	32	57

表 2 で求めたバウンディングボックスの対角距離 d が, およそ 150pixel になるように縦横比を保ちながら, 画像を拡大および縮小する. 次に, 縦横が 224pixel になるように上下左右の端, 一列の画素列で補間することで, 正規化を行う. 図 3.12 にバウンディングボックスに沿って正規化した画像を示す. 左が入力画像で, 右が正規化した画像である.

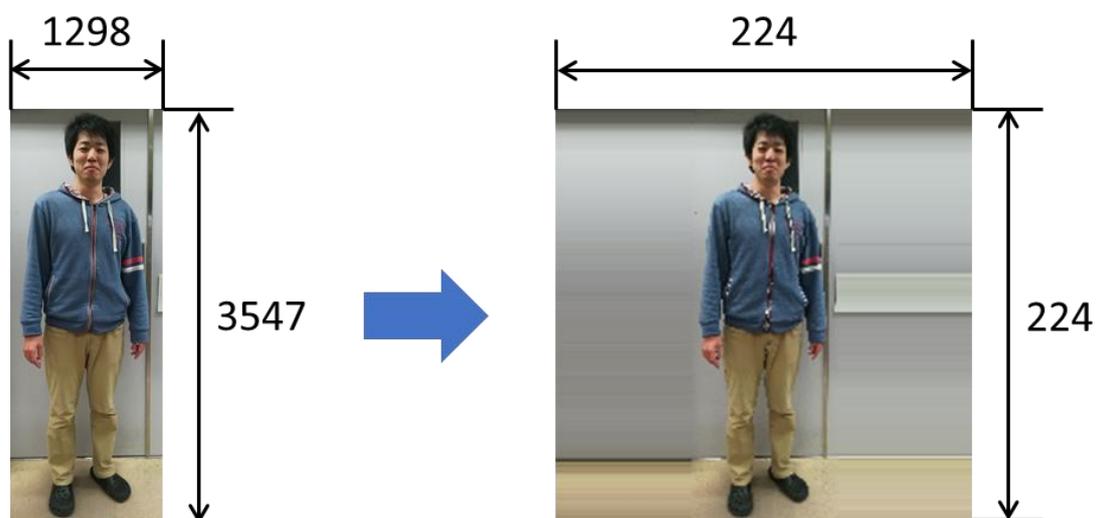


図 3.12 バウンディングボックスに沿って正規化した画像（単位[**pixel**]

正規化した画像に対して、**HMR** を用いて姿勢推定を行う。後述するユーザへの視覚的フィードバックとして、目標姿勢と現在の姿勢を重畳表示するため、3次元モデルのスケールをあわせる必要がある。そのため、各関節の軸角度および全体の回転を表す **Pose θ** のみを **SMPL** に代入し、3次元モデルを生成する。

第4章

姿勢支援手法

本章では、提案する姿勢支援手法の概要や手法、UI、実装環境について説明する。はじめに、本システムの全体図を示し、概要を述べる。次に、本システムのレンダリング条件、視覚的フィードバックの生成、レンダリングの高速化、実装について説明し、最後に作成したユーザインターフェース (UI) について記述する。

4.1 システム概要

本システム全体の流れを図 4.1 に示す。本システム全体の流れとして、トレーニング前のトレーニング画像、トレーニング中のカメラ画像を入力として 3 次元モデルを生成する処理は同様であるが、以降の処理が異なる。トレーニング前は視点と 3 次元モデルの保存、トレーニング中は、視覚的フィードバックの生成を行う。以降、トレーニング前、トレーニング中の処理について詳述する。

まず、トレーニング前の処理について説明する。ユーザは、はじめに行いたいトレーニングの画像を入力する。システムは、入力した画像に対して 3 次元モデルを生成する。そして、ユーザは生成された 3 次元モデルの姿勢が分かりやすい視点の設定を行う。その視点と生成した 3 次元モデルは、トレーニング中の処理で使用するため、保存する。

次に、トレーニング中の処理について説明する。トレーニング中は、Web カメラからユーザのトレーニング画像を取得し、トレーニング前の処理と同様に、3 次元モデルの生成を行う。そして、ユーザへの視覚的フィードバックとして、生成された現在姿勢の 3 次元モデルと事前に保存した目標姿勢の 3 次元モデルを重畳表示する。また、目標姿勢と現在の姿勢の違いを分かりやすくするために、身体の各部位 10 箇所（両手および肘，肩，膝，足首）にマーカを表示し、目標姿勢との誤差に応じて、色を 4 段階に変化させ、姿勢の補正を促す。ユーザはこれらの提示された情報をもとに姿勢を目標姿勢へと近づけていく。

以上の流れで、本システムはユーザへの姿勢支援を行い、正しい姿勢でのトレーニングを促す。

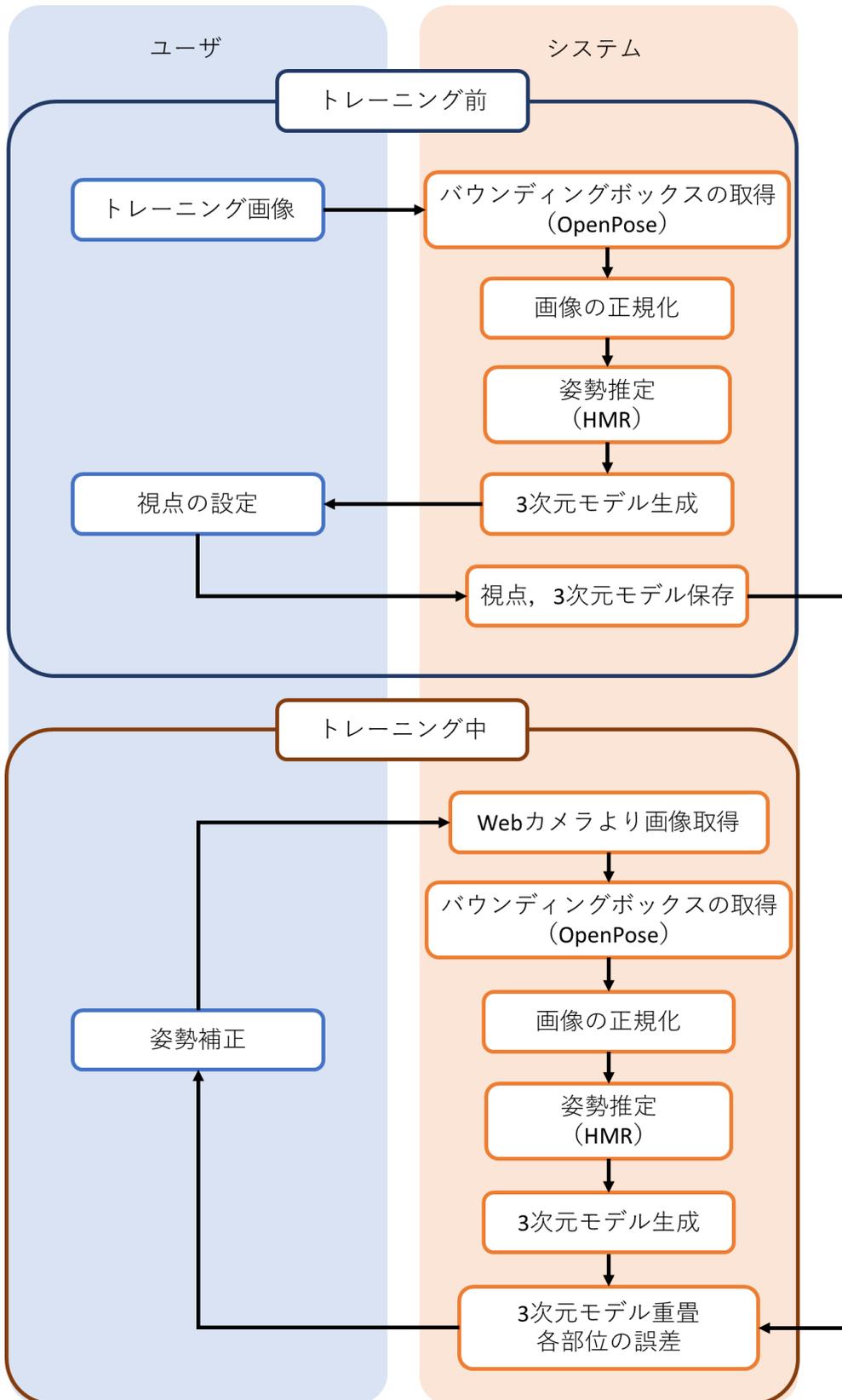


図 4.1 システムの流れ

4.2 レンダリング条件

本システムのレンダリング条件は、後述する UI の表示領域内に直立した 3 次元モデルが収まるように、トレーニング前、トレーニング中、それぞれ表 3 の値を用いる。

表 3 レンダリング条件

	画像サイズ (横, 縦)	光学中心(c_x, c_y)	焦点距離 f
トレーニング前	665, 650	332.5, 325	500
トレーニング中	870, 930	435, 465	750

4.3 視覚的フィードバックの生成

ユーザへの視覚的フィードバックとして、目標姿勢と現在姿勢の 3 次元モデルを重畳させた画像と身体の各部位 10 箇所 (両手および肘, 肩, 膝, 足首) にマーカを表示し、目標姿勢との誤差に応じて、その色を 4 段階に変化させる。

まず、目標姿勢と現在姿勢の 3 次元モデルを重畳させた画像は、トレーニング前の処理で保存された目標姿勢の 3 次元モデルに対して、表 3 に示すトレーニング中のレンダリング条件であらかじめレンダリングし、その画像を保存しておく。そして、現在姿勢の 3 次元モデルを同条件でレンダリング後、保存済みの目標姿勢のレンダリング画像と重畳する。2 つの 3 次元モデルの原点は、腰部分で一致しているため、位置合わせは行わない。

次に、身体の各部位 10 箇所 (両手および肘, 肩, 膝, 足首) の誤差を示すマーカの表示手法について述べる。

はじめに、マーカの表示位置を両手および肘, 肩, 膝, 足首の 10 箇所に選んだ理由として、末端の部位は体幹部分よりも誤差が大きくなりやすく、姿勢を補正するには重要な部分であると考えたためである。さらに、体幹部分の姿勢提示として股関節および肩の表示を考慮した。股関節は 3 次元モデルの原点が腰部分であることから、大きな誤差が出にくく、末端 (足首, 膝) の表示で十分と考え、本システムの姿勢支援には適切でないと考えた。肩に関しては、原点である腰部分から離れていることや、肘, 手だけの表示では上半身の姿勢提示に不十分であると考え、肩を表示することとした。

次に、各部位のマーカの表示位置を求める際に、用いる式を説明する。レンダリングに用いている OpenDR は、透視投影モデルであり、座標変換行列は式(1)で表される。

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \mathbf{K} [\mathbf{R} | \mathbf{t}] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1)$$

ここで, f は焦点距離, c_x, c_y は光学中心, \mathbf{R} は回転行列, \mathbf{t} は平行移動行列である. さらに, レンズ歪みは,

$$X'' = (1 + k_1 r^2 + k_2 r^4 + k_3 r^6)X' + 2p_1 X'Y' + p_2(r^2 + 2X'^2) \quad (2)$$

$$Y'' = (1 + k_1 r^2 + k_2 r^4 + k_3 r^6)Y' + p_1(r^2 + 2X'^2) + 2p_2 X'Y' \quad (3)$$

と表され, X', Y' はそれぞれ $X' = X/Z$, $Y' = Y/Z$ である.

本システムでは, レンダリングの条件を

$$\mathbf{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{t} = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}, \quad k_1 = k_2 = k_3 = p_1 = p_2 = 0 \text{ とし, } f, c_x, c_y \text{ は, 表 3 のよ}$$

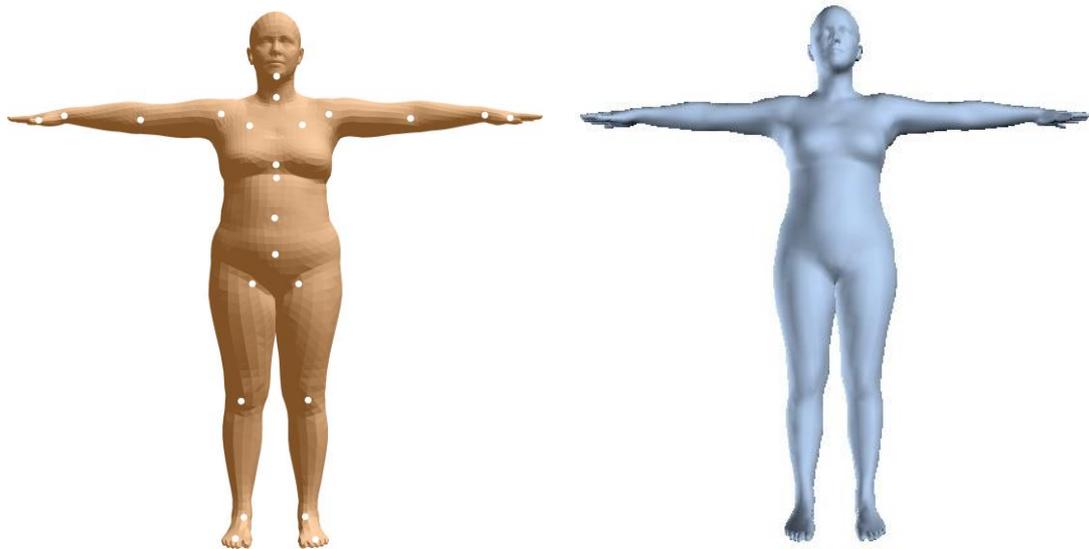
うに場面に応じて変更する. Z 軸方向に平行移動しているのは, 原点に人体モデルの腰部分が設定されているので, 平行移動しなければ近すぎて, 人体モデル全体を把握することができないためである. よって, 式(1)は,

$$\begin{aligned} \begin{bmatrix} x_{image} \\ y_{image} \\ 1 \end{bmatrix} &= \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} X_{model} \\ Y_{model} \\ Z_{model} \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} f & 0 & c_x & 2c_x \\ 0 & f & c_y & 2c_y \\ 0 & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} X_{model} \\ Y_{model} \\ Z_{model} \\ 1 \end{bmatrix} = \begin{bmatrix} fX_{model} + c_x Z_{model} + 2c_x \\ fY_{model} + c_y Z_{model} + 2c_y \\ Z_{model} + 2 \end{bmatrix} \end{aligned} \quad (4)$$

となり, 左辺に合わせるために, 右辺を $Z_{model} + 2$ で割ると, 式(5)となる.

$$\begin{bmatrix} x_{image} \\ y_{image} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{fX_{model}}{Z_{model} + 2} + c_x \\ \frac{fY_{model}}{Z_{model} + 2} + c_y \\ 1 \end{bmatrix} \quad (5)$$

各部位のマーカの表示位置は, あらかじめ身体の各部位とメッシュ頂点とを対応付けておき, 対応するメッシュ頂点の 3 次元モデル空間座標から画像座標に変換することで求める. 身体の各部位とメッシュ頂点の対応付けは, まず, 図 4.2 (a) に示す関節位置を記した画像と図 4.2 (b) に示す表 3 トレーニング前の条件下で出力した SMPL の画像を図 4.3 に示すように重ね合わせ, 後述する手法で各位置の画像座標を取得する. その後, 式(5)に 表 3 トレーニング前の条件を代入した式(6)を用いて, 3 次元モデル空間座標に変換し, 該当するメッシュ頂点を取得する.



(a) 関節位置を記した SMPL ([14] Figure 3 (b)) (b) 現環境で出力した SMPL
 図 4.2 身体の各部位とメッシュ頂点对応付けに用いた画像



図 4.3 重ね合わせた画像

$$\begin{bmatrix} x_{image} \\ y_{image} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{500 \times X_{model}}{Z_{model} + 2} + 332.5 \\ \frac{500 \times Y_{model}}{Z_{model} + 2} + 325 \\ 1 \end{bmatrix} \quad (6)$$

各部位の画像座標一点だけでは対象のメッシュ頂点が得られなかった。そこで、対応するメッシュ頂点を得られるように、ある幅を持った領域内を探索する。

検索対象の領域は図 4.4 の赤枠とし，探索時の条件は，図 4.4 に示すように検索対象領域の左上（図中オレンジ枠）を始点，右下（図中青枠）を終点とする．始点 (x_s, y_s) ，終点 (x_e, y_e) の大小関係は式(7)のようになっており，対応するメッシュ頂点の座標は式(7)の範囲内（式(8)）となる．よって，式(8)に式(6)を代入すると式(9)となり，メッシュ頂点は式(9)を満たす．

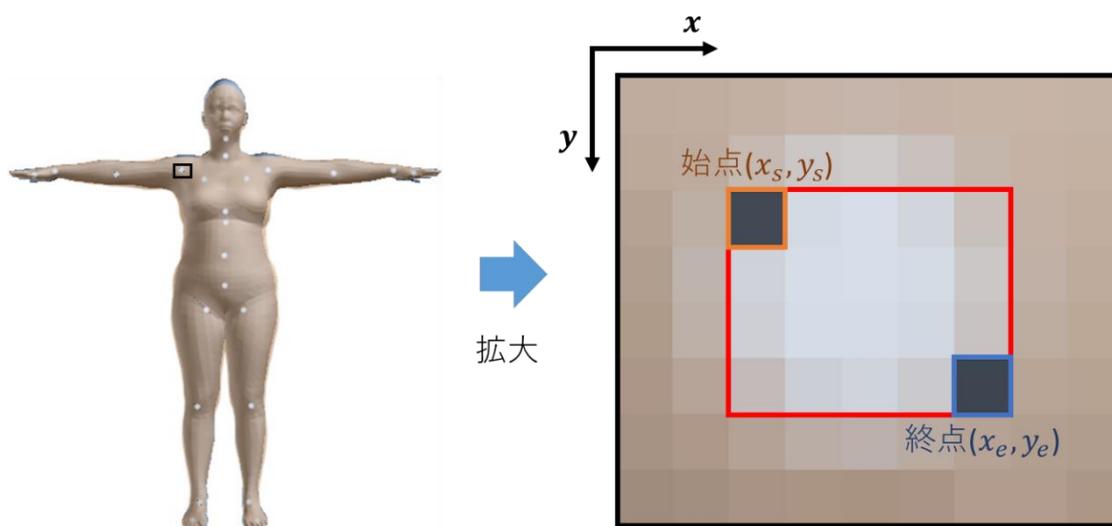


図 4.4 画像座標の始点と終点

$$\begin{cases} x_s < x_e \\ y_s < y_e \end{cases} \quad (7)$$

$$\begin{cases} X_e < X_{model} < X_e \\ Y_s < Y_{model} < Y_e \end{cases} \quad (8)$$

$$\begin{cases} \frac{x_s - 332.5}{500} (Z_{model} + 2) < X_{model} < \frac{x_e - 332.5}{500} (Z_{model} + 2) \\ \frac{y_s - 332.5}{500} (Z_{model} + 2) < Y_{model} < \frac{y_e - 332.5}{500} (Z_{model} + 2) \end{cases} \quad (9)$$

表 4 に本手法を用いて取得した身体の各部位の画像座標（始点，終点）とメッシュ頂点を示す．各部位に対応するメッシュ頂点は，視点によって表示位置がずれるのを防ぐために，前面と背面で 1 点ずつ取得し，表示にはその 2 点の平均値を用いる．

表 4 身体の各部位の画像座標とメッシュ頂点番号

	画像座標		メッシュ頂点番号
	始点(x_s, y_s)[pixel]	終点(x_e, y_e)[pixel]	
左手	480,179	484,181	2107,2109
左肘	424,181	427,185	1644,1699
左肩	363,177	367,180	1887,1898
左膝	347,397	351,402	1047,1065
左足首	247,487	350,491	3386,3400
右手	167,179	169,181	5527,5573
右肘	223,181	226,185	5133,5168
右肩	283,177	287,180	5313,5359
右膝	297,397	303,402	4541,4551
右足首	300,487	303,491	6787,6800

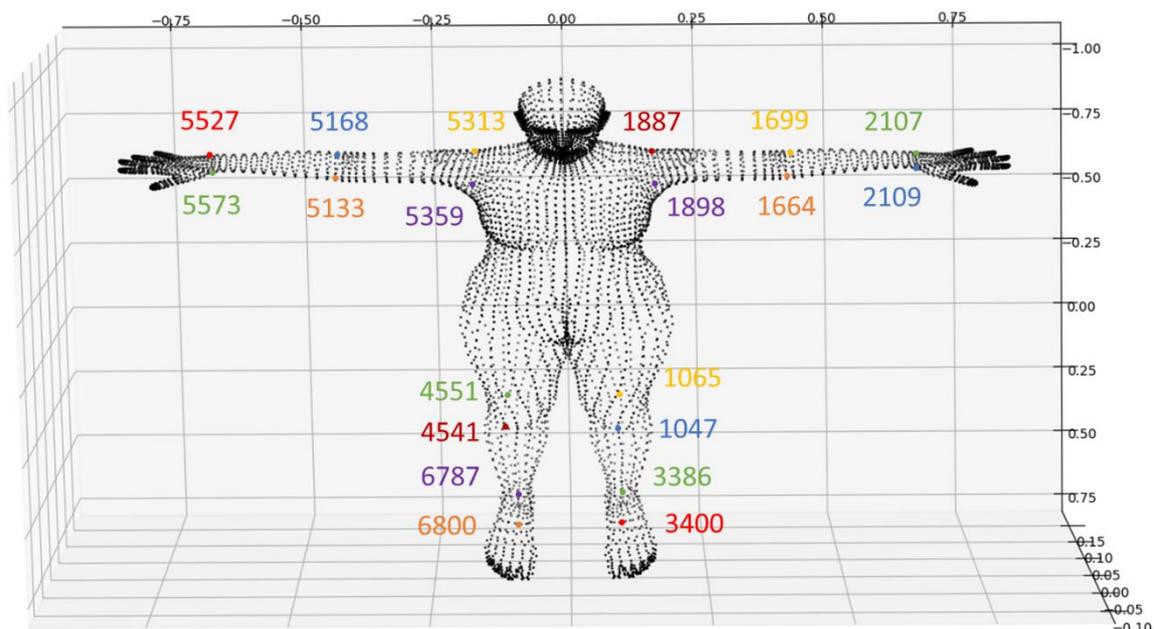


図 4.5 各部位に対応したメッシュ頂点番号

表 4 に示したメッシュ頂点番号を用いて、現在姿勢の 3 次元モデルから身体
の各部位に対応した 3 次元モデル空間におけるメッシュ座標を取得する。そし
て、表 3 に示すトレーニング中のレンダリング条件を式(5)に代入して得られる
式(10)を用いて、表示位置となる画像座標を取得する。図 4.6 にメッシュ頂点番
号を用いてマーカを表示した結果を示す。

$$\begin{bmatrix} x_{image} \\ y_{image} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{750 \times X_{model}}{Z_{model} + 2} + 435 \\ \frac{750 \times Y_{model}}{Z_{model} + 2} + 465 \\ 1 \end{bmatrix} \quad (10)$$

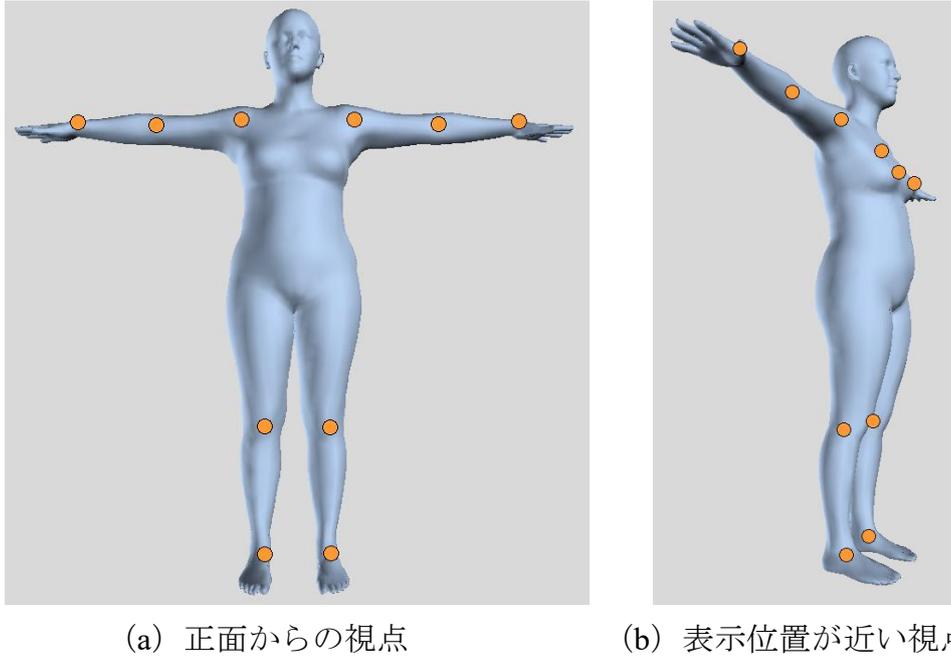
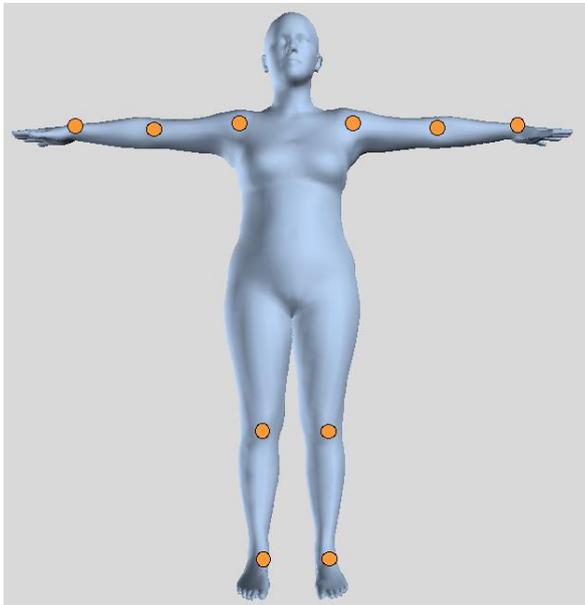


図 4.6 メッシュ頂点番号を用いたマーカ表示

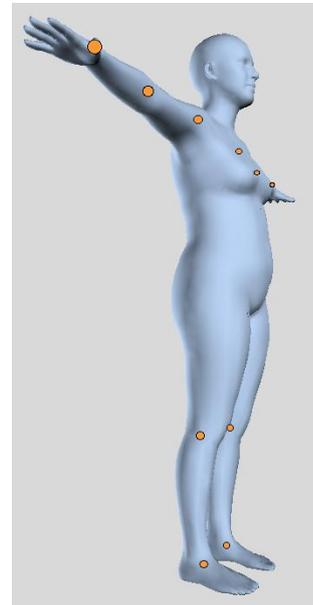
取得した表示位置にマーカを同一の大きさで表示すると、表示する部位が奥と手前で近くなっている状況下で、どちらがどちらの部位を示しているのか分かりづらくなってしまいう可能性がある。そこで、身体各部位に対応した3次元モデル空間におけるメッシュ座標のZ座標値 (Z_{model}) を用いて、マーカの大きさを変えることにより、視認性を向上させる。Z軸は奥方向に正であるため、一番手前(近く)にある点は最小値 Z_{min} を取る。そこで、最小値を基準に式(11)を用いて、マーカの大きさを変更する。定数 k は基準となるマーカの大きさで変更する。本研究では、 $k = 8$ とした。

$$\begin{cases} P_{xs} = x_{image} - k \left(\frac{Z_{min}}{Z_{model}} \right)^2 \\ P_{xe} = x_{image} - k \left(\frac{Z_{min}}{Z_{model}} \right)^2 \\ P_{ys} = y_{image} + k \left(\frac{Z_{min}}{Z_{model}} \right)^2 \\ P_{ye} = y_{image} + k \left(\frac{Z_{min}}{Z_{model}} \right)^2 \end{cases} \quad (11)$$

本システムでは，始点(P_{xs}, P_{ys})と終点(P_{xe}, P_{ye})を指定してマーカを表示する．
 図 4.7 に式(11)を用いてマーカを表示した結果を示す．



(a) 正面からの視点



(b) 表示位置に近い視点

図 4.7 (11)式を用いたマーカ表示

現在姿勢と目標姿勢の誤差による色変化には，マーカの表示位置を求めるのに用いたメッシュ頂点座標の平均値を現在姿勢，目標姿勢とでそれぞれ求め，その2つの平均値間距離 d_{err} を用いる．条件は以下の通りである．

$$Color = \begin{cases} red & (if\ 0.5 \leq d_{err}) \\ orange & (if\ 0.25 \leq d_{err} < 0.5) \\ yellow & (if\ 0.1 \leq d_{err} < 0.25) \\ greenyellow & (if\ d_{err} < 0.1) \end{cases}$$

ユーザは，目標姿勢と現在姿勢の重畳した 3 次元モデルと各部位の誤差を示したマーカをもとに姿勢補正を行う．

4.4 レンダリングの高速化

プロトタイプとして構築したシステムを図 4.8, 図 4.9 に示す.

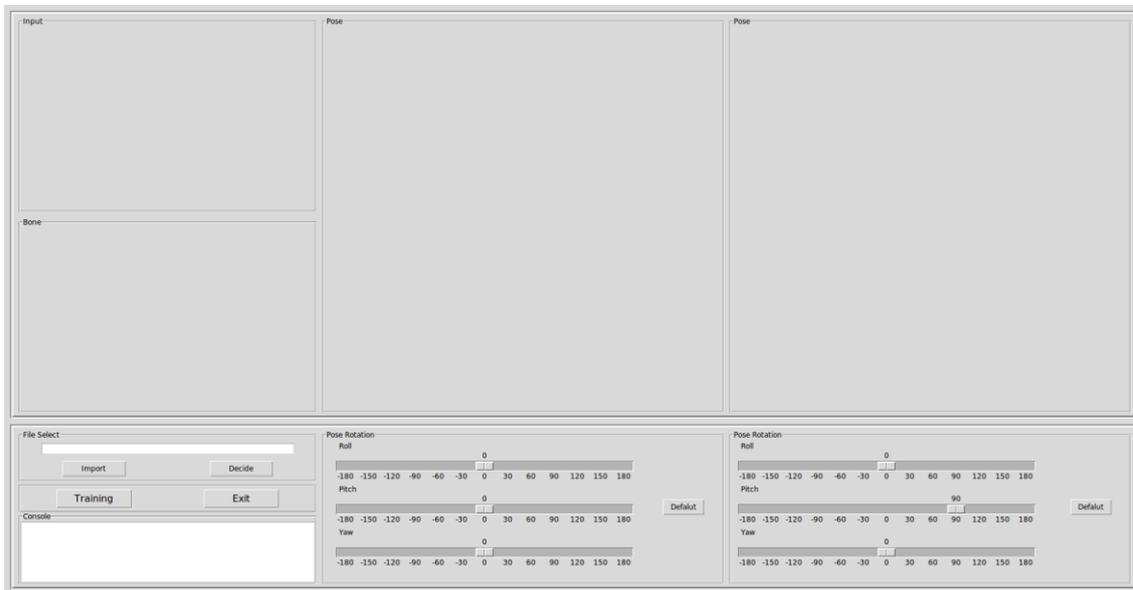


図 4.8 トレーニング前のウィンドウ

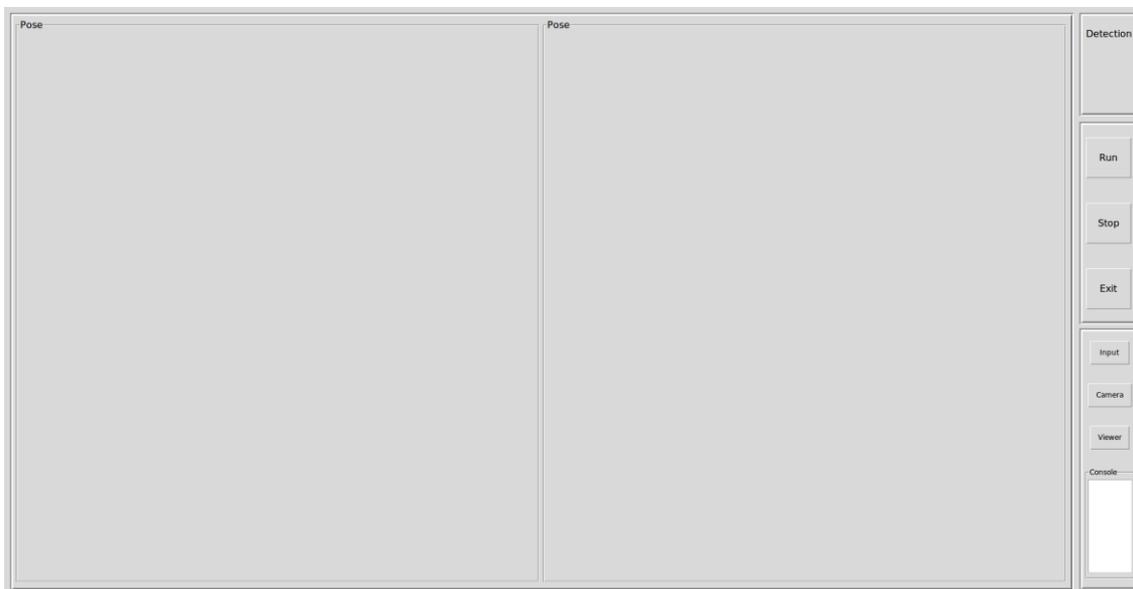
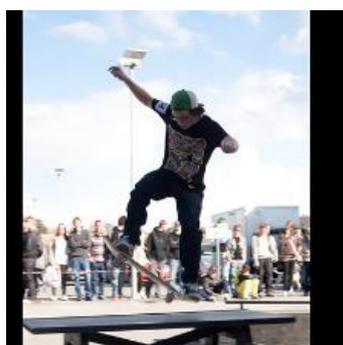


図 4.9 トレーニング中のウィンドウ

このシステムでは、視覚的フィードバックの更新速度が 5 秒程度に一度と遅く、姿勢の支援が困難であると考え、システムの高速化を検討した。本システムで用いている HMR の Github (Issues)⁸では、ボトルネックは OpenDR によるレンダリングであると述べている。このことから、レンダリングの高速化を図った。レンダリングの高速化にあたり、Python での実装を考え、OpenGL⁹の Python API である PyOpenGL¹⁰を用いる。レンダリングの速度比較に用いた画像を図 4.10 に示す。また、レンダリング結果および時間を図 4.11 に示す。



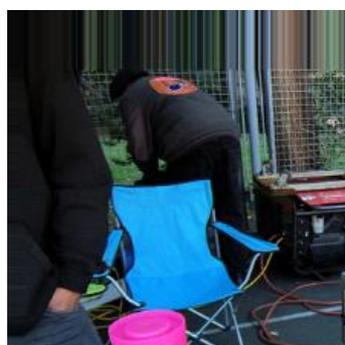
(a) 画像 1



(b) 画像 2



(c) 画像 3



(d) 画像 4



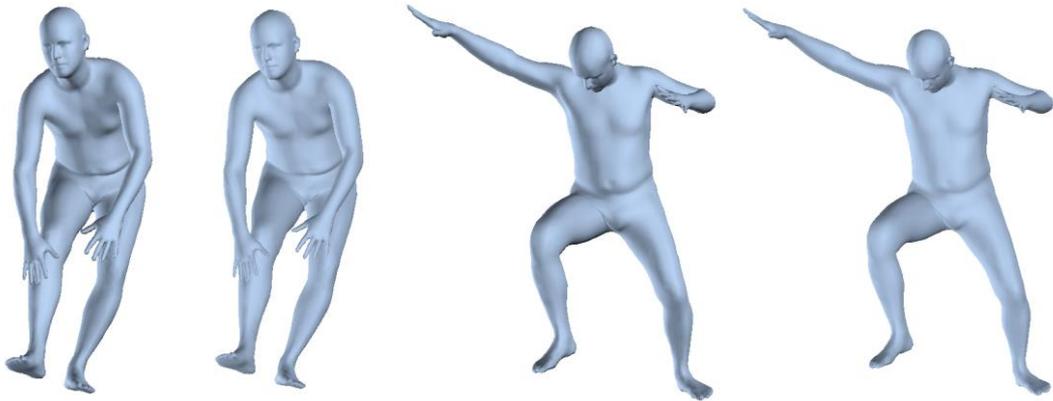
(e) 画像 5

図 4.10 レンダリング速度比較に用いた画像 [13]

⁸ HMR (Github Issues) <https://github.com/akanazawa/hmr/issues/53>

⁹ OpenGL <https://www.opengl.org/>

¹⁰ PyOpenGL <http://pyopengl.sourceforge.net/index.html>



(a) 画像 1

(b) 画像 2



(c) 画像 3

(d) 画像 4



(e) 画像 5

図 4.11 レンダリング速度比較結果 (左 : OpenDR, 右 : PyOpenGL)

表 5 レンダリング速度 (OpenDR) [s]

	画像 1	画像 2	画像 3	画像 4	画像 5
1 回目	0.321	0.314	0.313	0.316	0.318
2 回目	0.319	0.317	0.319	0.321	0.314
3 回目	0.323	0.314	0.333	0.313	0.315
平均値	0.321	0.315	0.322	0.317	0.316

表 6 レンダリング速度 (PyOpenGL) [s]

	画像 1	画像 2	画像 3	画像 4	画像 5
1 回目	0.0580	0.0557	0.0467	0.0477	0.0452
2 回目	0.0562	0.0468	0.0594	0.0463	0.0584
3 回目	0.0561	0.0571	0.0485	0.0533	0.0484
平均値	0.0568	0.0532	0.0513	0.0491	0.0507

表 5, 表 6 に示した結果から, OpenDR よりも PyOpenGL の方が, レンダリング速度が約 6 倍速いことが分かる. よって, 本システムを PyOpenGL で実装することとした. ただし, OpenDR を用いて構築したプロトタイプの視覚的フィードバックの更新速度と表 5 のレンダリング速度を見比べると更新速度の方がかなり遅い. このことから, UI の処理にも時間がかかっている可能性があるため, 実装時には, なるべく複雑な機能は搭載せず, 簡単な機能のみを搭載することを心掛けることとした.

4.5 実装環境

本システムを実装するにあたり, 以下の環境で行った.

OS : Ubuntu 16.04.5 LTS (Xenial Xerus)

CPU : Intel® Core™ i5-4690 CPU@3.50GHz

GPU : Geforce GTX 1060 6GB

RAM : 16GB

CUDA : 8.0.61

cuDNN : 6.0.21

Tools : Pyenv, Pyenv-virtualenv

4.6 User Interface

操作手順と共に UI について説明する。

まず、システムを起動すると図 4.12 のような画面を表示する。



図 4.12 起動画面

はじめに、ユーザは左下の **Import** を押し、トレーニングしたい画像を選択する。**Import** を押し、図 4.13 のような画面を表示し、画像ファイルを選ぶ。ファイルタイプは `.jpg`, `.png`, `.jpeg` のみを表示する。

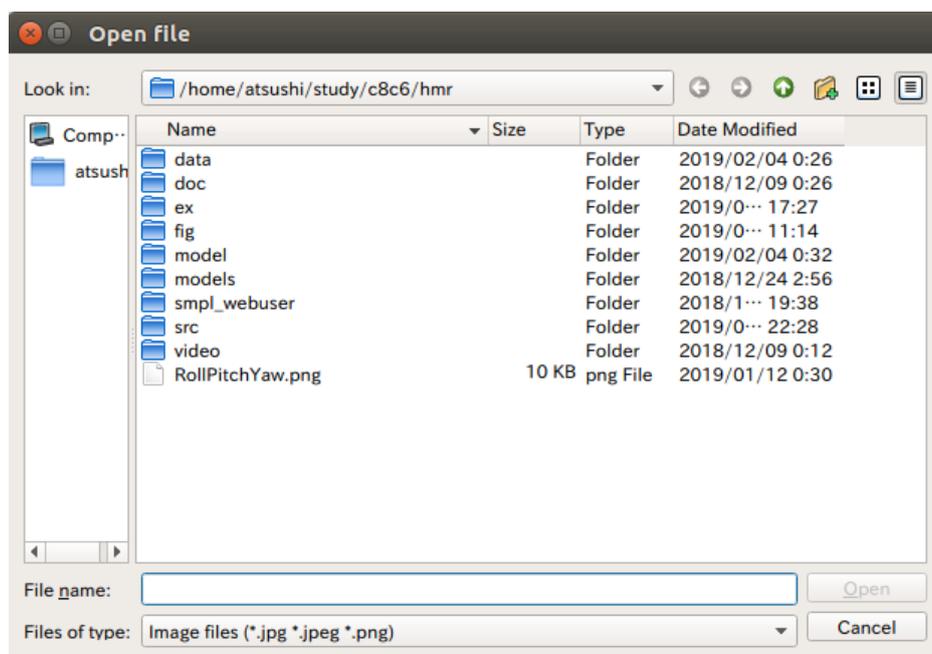


図 4.13 ファイル選択画面

画像を選択すると、図 4.14 のように左上の **Input** に選択した画像を示す。



図 4.14 ファイル選択完了画面

表示された画像で良ければ, **Decide** を押す. 違う画像が良ければ, もう一度 **Import** を押し, ファイルを選択しなおす. **Decide** を押すと姿勢推定を行い, 図 4.15 のように画面右側の **Pose** に 3 次元モデル, 画面左側の **Bone** に骨格情報を示すボーン画像を表示する.

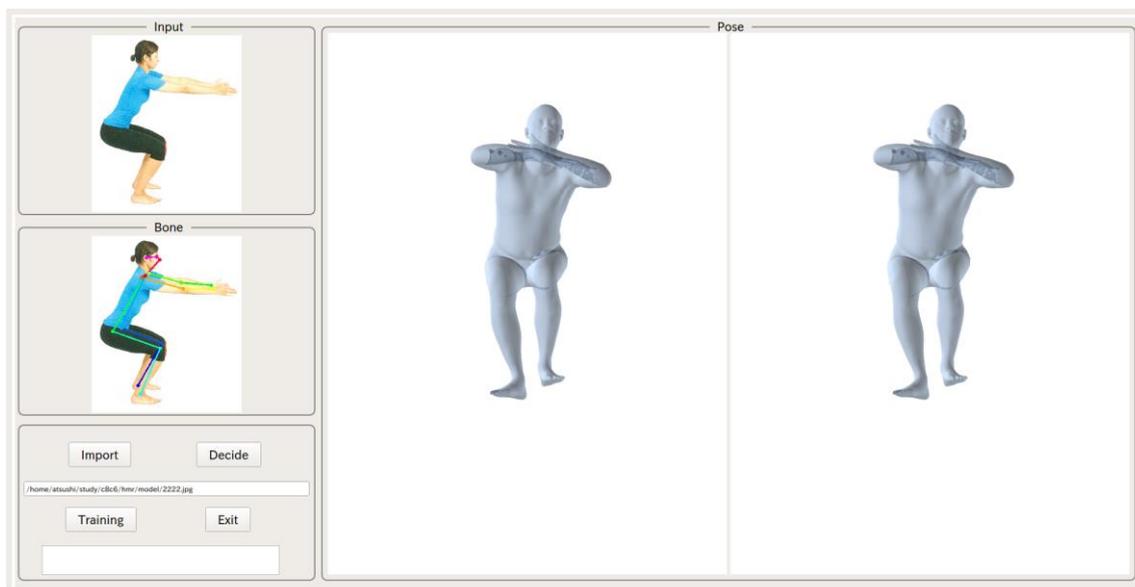


図 4.15 3次元モデル表示画面

次に、表示されている3次元モデルをマウスでドラッグすると視点が回転する。ユーザはマウス操作を行い、3次元モデルの姿勢が分かりやすい視点を設定する。例として、視点を変更した画面を図4.16に示す。

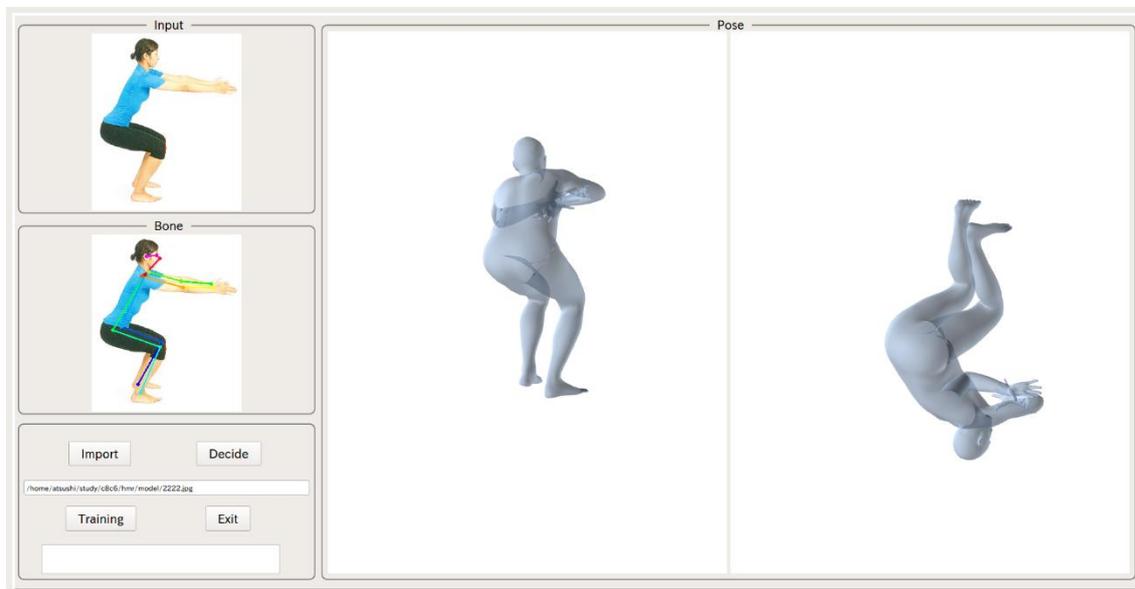


図 4.16 視点変更

ユーザが視点の設定をし終わり、**Training** を押すと、図4.17のようなトレーニング画面に切り替わる。



図 4.17 トレーニング画面

Show を押すと、図 4.16 の時にユーザが設定した視点で目標姿勢の 3 次元モデルを表示する。例として、ユーザが設定した視点を図 4.18, Show を押して表示される画面を図 4.19 に示す。

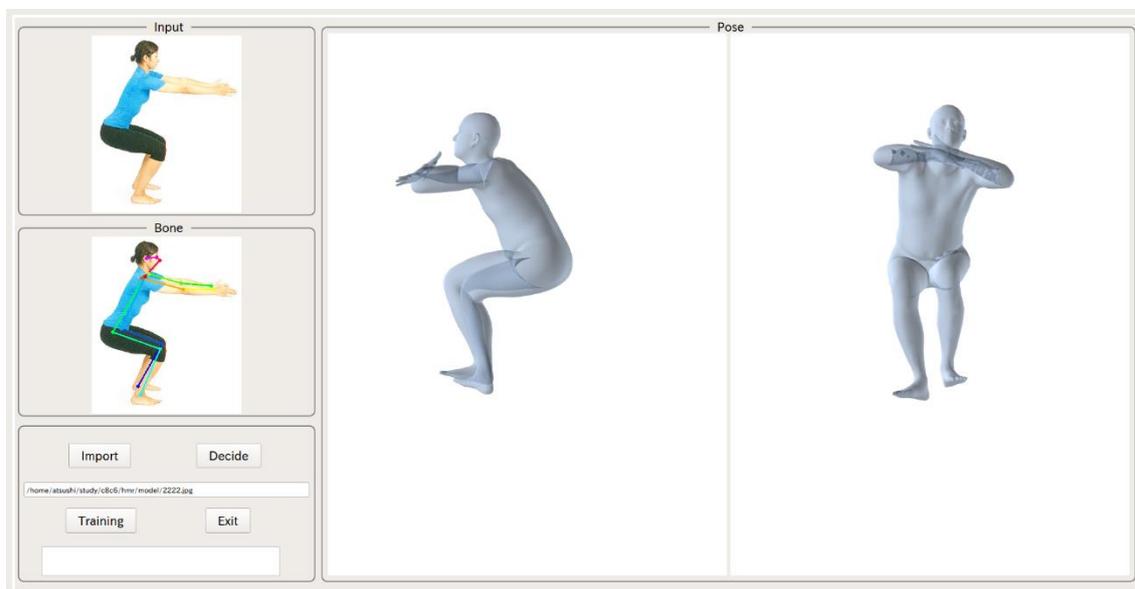


図 4.18 視点の設定例

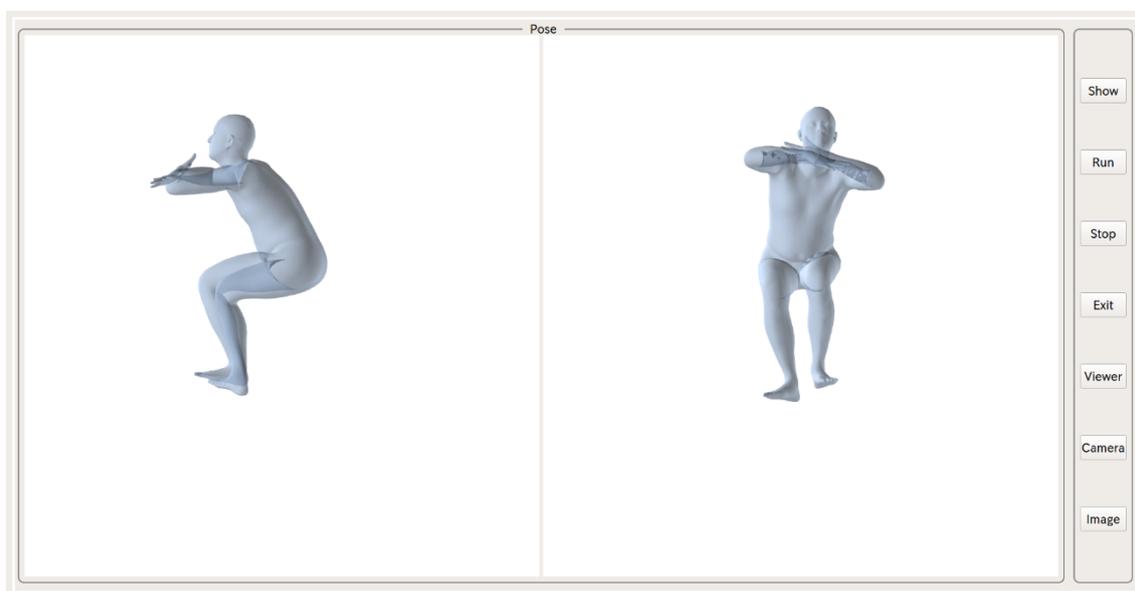
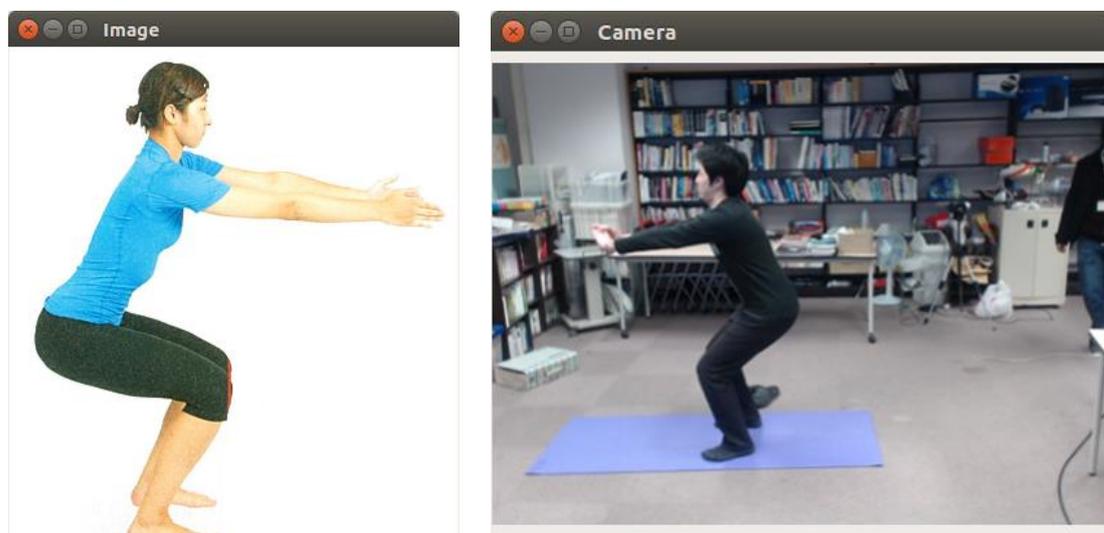


図 4.19 目標姿勢表示画面

Image と Camera をそれぞれ押すと，図 4.20 のような入力画像やカメラ画像のウィンドウを表示する．このウィンドウは常に先程のトレーニング画面より前に表示する．さらに，表示されている画像は，ウィンドウサイズに合わせて自動リサイズするため，ユーザが見やすい位置，大きさに変更できる．



(a) 入力画像 [16]

(b) カメラ画像

図 4.20 各表示ウィンドウ

ウィンドウを配置し終わり，Run を押すと図 4.21 のよう目標姿勢と現在姿勢の3次元モデルと身体の一部の誤差に応じて，色変化するマーカを表示する．ユーザは，この画面を見ながら姿勢の補正を行う．

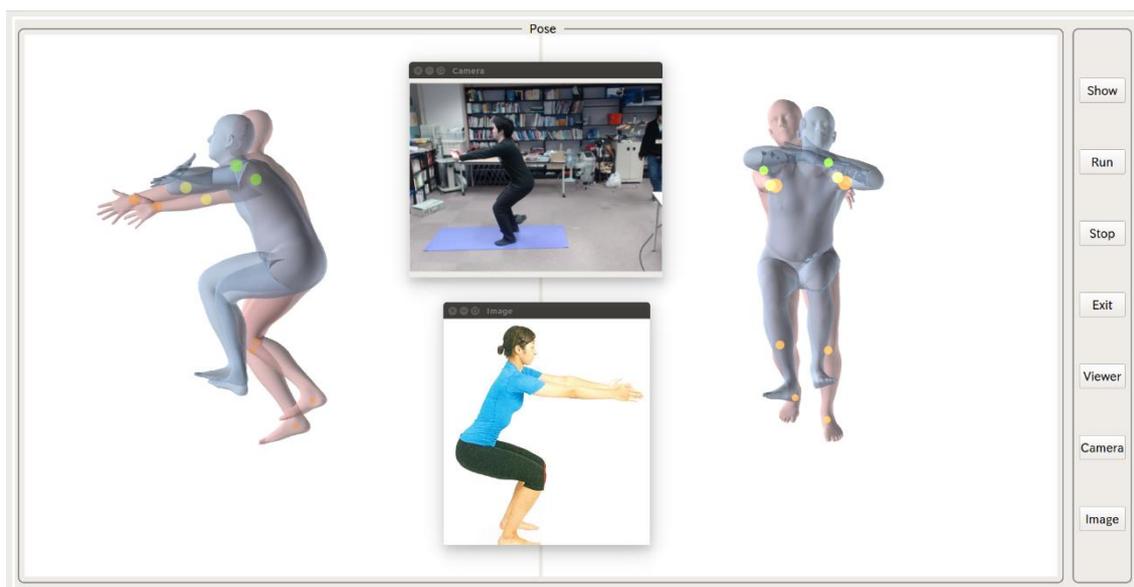


図 4.21 トレーニング中の画面

トレーニングが終了したら、**Stop** を押し、その後に **Exit** を押すとシステムは終了する。また、**Viewer** を押すと、最初に表示された画面に戻ることができる。

第5章

実験

本章では、提案手法の有用性を検証するための実験について説明する。はじめに、実験概要、実験環境について述べる。そして、結果を示し、考察を行う。

5.1 実験概要

本研究で提案した手法の有用性を検証するために評価実験を行う。関連研究 [9] でフィードバック情報として用いられている 2 次元ボーンを 1 視点で提示する手法（手法 1）と、本研究で提案した 3 次元モデルを 2 視点から提示する手法（手法 2）を比較する。手法ごとに異なる推定手法を用いると提示する目標姿勢に誤差が生じる可能性があるため、本研究で用いた推定手法を利用して、手法 1 を実装した。提示する関節数は、OpenPose [5] で用いられている COCO データセット [13] を参考に 18 箇所（両耳、両目、鼻、首、両肩、両肘、両手、両腰、両膝、両足首）とする。手法 1、手法 2 の UI および提示例を図 5.1、図 5.2 に示す。

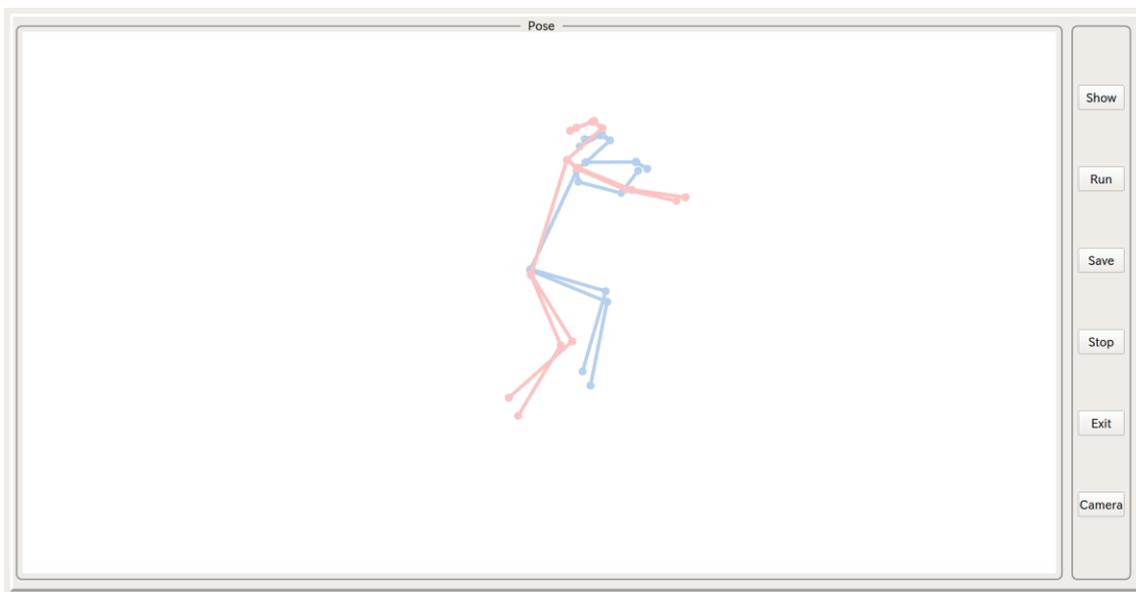
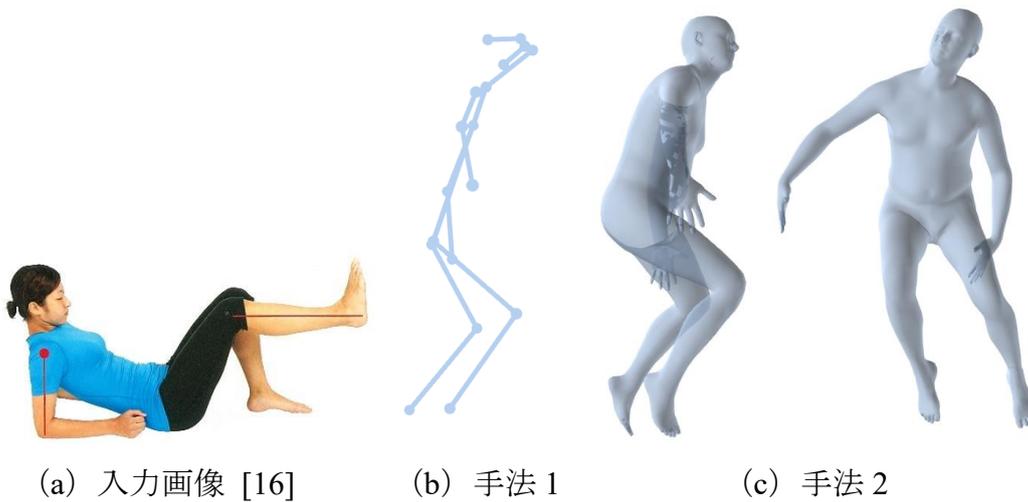


図 5.1 UI および提示例（手法 1）



図 5.2 UI および提示例（手法 2）

実験の対象とする体幹トレーニングは図 5.3 (a) ～図 5.6 (a) に示す 4 つであり、体位による影響を考え、立位，側臥位，仰臥位で行うトレーニングを参考文献 [16] から選定した．対象とする体幹トレーニングそれぞれにおいて，手法 1，手法 2 で提示するボーンおよび 3 次元モデル画像を図 5.3 (b)，(c) ～図 5.6 (b)，(c) に示す．



(a) 入力画像 [16]

(b) 手法 1

(c) 手法 2

図 5.3 体幹トレーニング 1

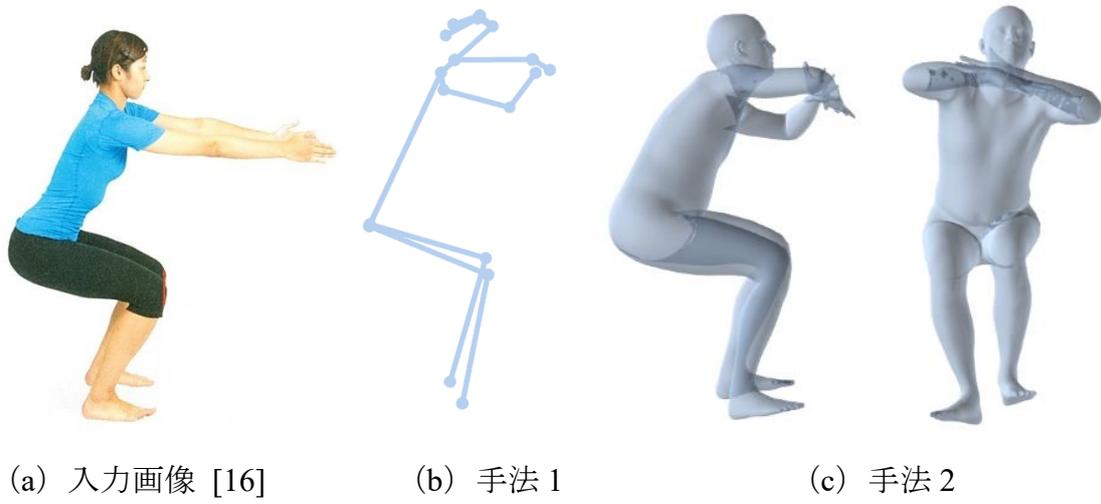


図 5.4 体幹トレーニング 2

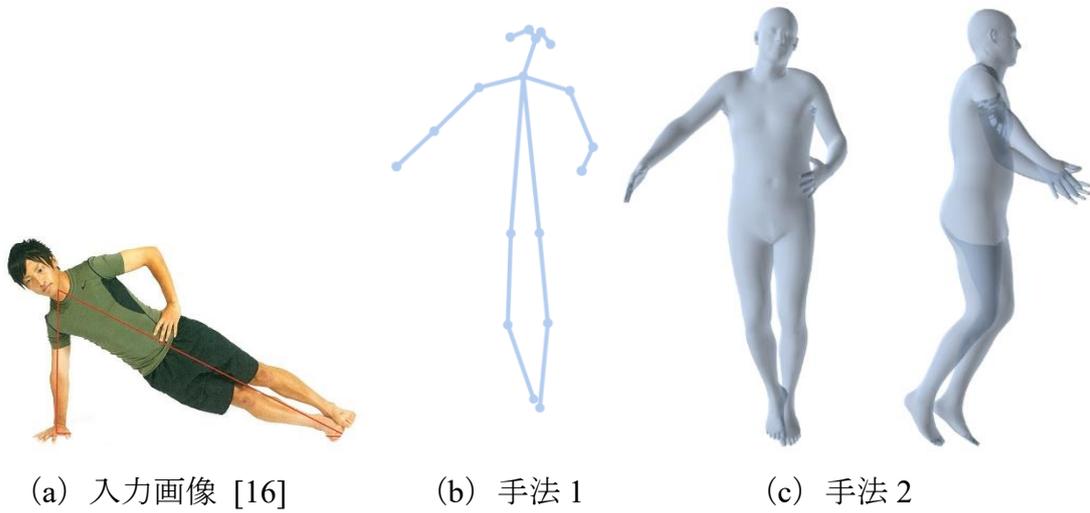


図 5.5 体幹トレーニング 3

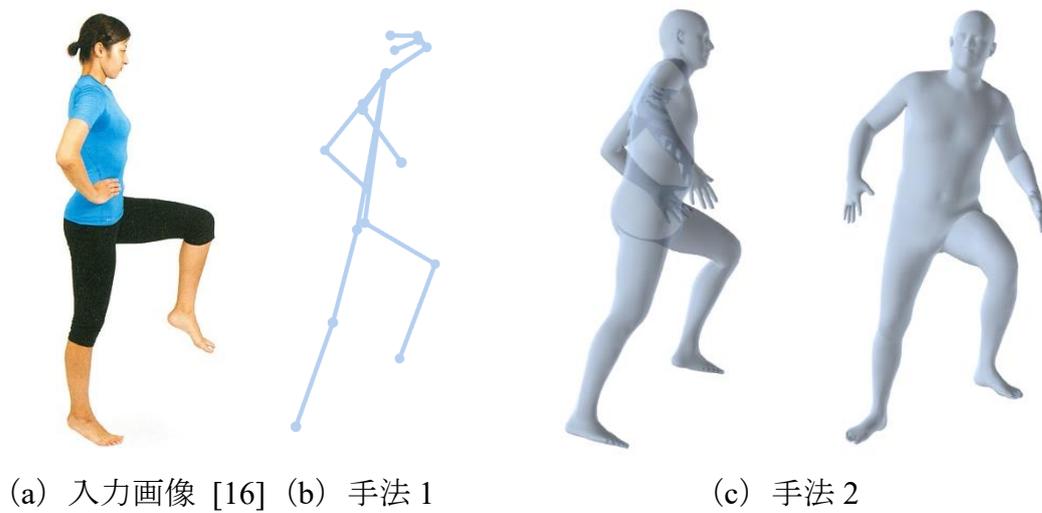


図 5.6 体幹トレーニング 4

本実験の被験者は、本学の男子学生 8 名（日本人学生：5 名，中国人留学生：3 名，年齢：24.2±1.05）である。手法 1, 2 と体幹トレーニングの組み合わせは、実施する順番による影響を考慮し、表 7 のようにする。体幹トレーニングはそれぞれ 25 秒程度行い、各フレームにおけるカメラ画像およびトレーニング姿勢を保存する。また、実験終了後にアンケート（図 5.7）を実施し、姿勢の把握しやすさや本システムで付加した視覚的フィードバックについて調査する。

表 7 被験者における手法と体幹トレーニングの組み合わせ
(手法番号 - 体幹トレーニング番号)

	A	B	D	C	E	F	G	H
1 回目	2-1	2-3	1-4	1-2	2-1	2-3	1-4	1-2
2 回目	2-2	2-4	1-3	1-1	2-2	2-4	1-3	1-1
3 回目	1-3	1-1	2-2	2-4	1-3	1-1	2-2	2-4
4 回目	1-4	1-2	2-1	2-3	1-4	1-2	2-1	2-3

アンケート

氏名

年齢

1. これまでに体幹トレーニングをしたことはありますか.

はい

いいえ

2. どちらの条件が姿勢を把握しやすかったですか.

1視点ボーン画像

2視点3次元モデル

3. 2視点になることで、姿勢の把握がしやすくなりましたか

5 4 3 2 1
しやすい ある程度 全く

4. 視点が変えられることで、姿勢の把握がしやすくなりましたか.

5 4 3 2 1
しやすい ある程度 全く

5. 3次元モデルになることで、姿勢の把握がしやすくなりましたか.

5 4 3 2 1
しやすい ある程度 全く

6. 色マーカーは姿勢の補正に役立ちましたか

5 4 3 2 1
役立った ある程度 全く

7. マーカーの大きさが近いほど大きく、遠いほど小さくなることに気が付きましたか

気が付いた

気が付かなかった

7-1. 大きさの変更で見やすくなりましたか (7の回答:気が付いた方のみ)

5 4 3 2 1
見やすい ある程度 全く

8. システムが提示したモデルとあなたが思っている姿勢はどの程度一致しましたか.

5 4 3 2 1
思った通り ある程度 全く

図 5.7 アンケート用紙

5.2 実験環境

評価実験を行うにあたり，表 8 に示す機材を用いる．システムの実装環境は 4.5 節に示した実装環境で行う．実験の様子を図 5.8 に示す．

表 8 実験機材

液晶テレビ	メーカー	TOSHIBA
	型番	55A2
	画面サイズ	55V 型
	解像度	1920×1080
Web カメラ	メーカー	Logicool®
	型番	c920r
	設定解像度	640×480

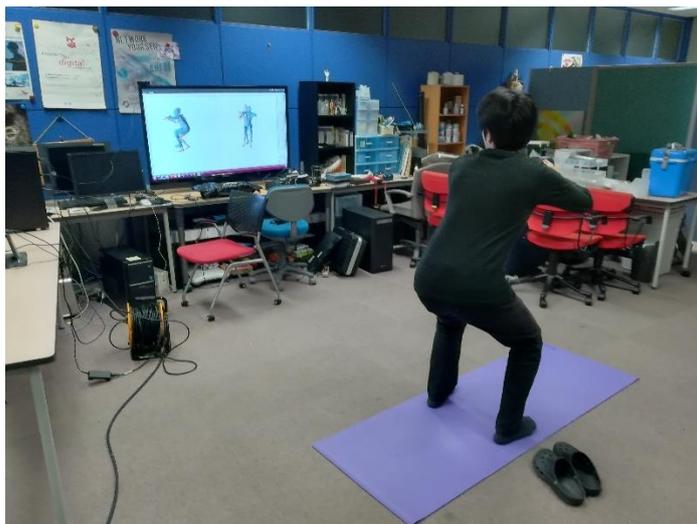


図 5.8 実験風景の例

5.3 実験結果

実験では、各フレームにおける 3 次元モデルの頂点座標を取得する。評価の指標として、平均平方二乗誤差 (RMSE: Root Mean Square Error) を用いる。RMSE は式(12)で表される。

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\vec{E}_i - \vec{M}_i)^2} \quad (12)$$

本実験では、それぞれ n は 6,980, \vec{E}_i は被験者におけるトレーニング姿勢の 3 次元モデル座標ベクトル $\vec{E}_i = (x_{e_i}, y_{e_i}, z_{e_i})$ である。 \vec{M}_i は目標姿勢の 3 次元モデル座標ベクトル $\vec{M}_i = (x_{m_i}, y_{m_i}, z_{m_i})$ である。

図 5.8 に示すように被験者はテレビ画面を見ながら姿勢の補正を行う。そのため、顔の付近では目標姿勢の 3 次元モデルと被験者のトレーニング姿勢が一致しない。そこで、首より上のメッシュ頂点 (図 5.9) を除いて、RMSE を求めることとした。

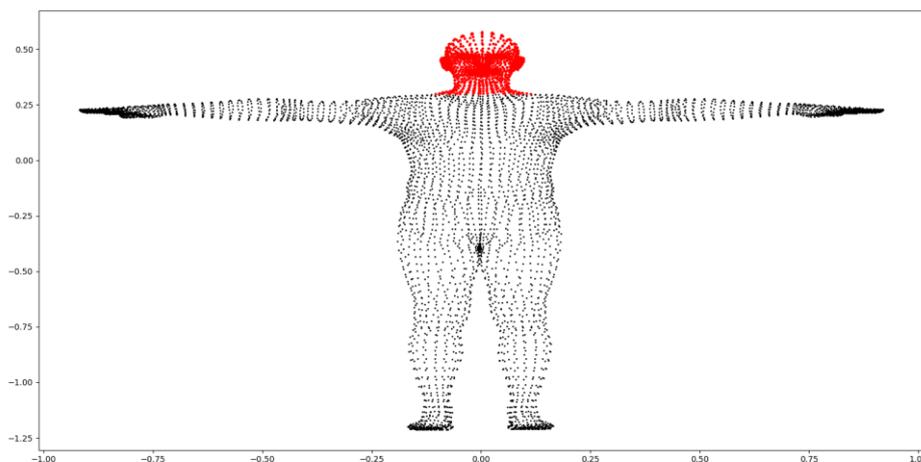


図 5.9 首より上のメッシュ頂点

表 9 被験者 A

時間[s]	RMSE			
	2-1	2-2	1-3	1-4
1.68±0.038	0.31640	0.20311	0.29916	0.15543
3.95±0.060	0.30650	0.23864	0.33810	0.14655
6.35±0.15	0.30387	0.26263	0.31333	0.15036
8.62±0.14	0.29869	0.22834	0.30517	0.15326
10.9±0.15	0.29900	0.17773	0.31068	0.15510
13.1±0.16	0.29361	0.25048	0.30159	0.16912
15.4±0.18	0.28563	0.24418	0.30964	0.17033
17.7±0.19	0.25006	0.23630	0.27272	0.16035
20.0±0.19	0.23222	0.24349	0.27928	0.15896
22.3±0.16	0.21917	0.24796	0.32331	0.16170

表 10 被験者 B

時間[s]	RMSE			
	2-3	2-4	1-1	1-2
1.72±0.012	0.15627	0.17570	0.27692	0.41645
4.07±0.035	0.20957	0.16334	0.29285	0.42594
6.39±0.059	0.22111	0.15854	0.29230	0.48125
8.72±0.072	0.19096	0.17388	0.30612	0.50641
11.0±0.069	0.21876	0.17175	0.31915	0.35412
13.4±0.066	0.21032	0.15463	0.37227	0.34701
15.7±0.078	0.20400	0.16896	0.38294	0.36770
18.0±0.11	0.20273	0.16324	0.39728	0.33130
20.3±0.13	0.22080	0.18758	0.38372	0.42536
22.6±0.13	0.36989	0.21324	0.31326	0.43794

表 11 被験者 C

時間[s]	RMSE			
	1-4	1-3	2-2	2-1
1.73±0.023	0.17986	0.20502	0.21171	0.42891
4.05±0.032	0.17980	0.20475	0.22954	0.40174
6.37±0.039	0.17068	0.20308	0.14454	0.40679
8.69±0.043	0.14006	0.17579	0.14758	0.40000
11.1±0.059	0.14479	0.21284	0.16786	0.39701
13.4±0.046	0.14755	0.15796	0.17196	0.39283
15.7±0.057	0.13758	0.11128	0.18211	0.19550
18.1±0.078	0.14085	0.13986	0.15872	0.36070
20.4±0.086	0.13817	0.11977	0.18624	0.40403
22.7±0.10	0.14404	0.12489	0.15826	0.40675

表 12 被験者 D

時間[s]	RMSE			
	1-2	1-1	2-4	2-3
1.69±0.052	0.31808	0.38770	0.16363	0.080085
4.00±0.067	0.32960	0.39356	0.16651	0.19919
6.29±0.11	0.30757	0.33074	0.15051	0.21554
8.58±0.12	0.29099	0.36011	0.17430	0.20005
10.9±0.013	0.34370	0.33487	0.16600	0.12430
13.2±0.15	0.31924	0.35502	0.12329	0.13241
15.5±0.16	0.29390	0.33753	0.11494	0.12891
17.8±0.19	0.34895	0.29161	0.14389	0.16194
20.1±0.22	0.34163	0.29770	0.12899	0.17862
22.4±0.23	0.34158	0.35786	0.13918	0.19190

表 13 被験者 E

時間[s]	RMSE			
	2-1	2-2	1-3	1-4
1.68±0.011	0.20263	0.22231	0.25105	0.12406
3.95±0.035	0.19845	0.21861	0.15576	0.16748
6.21±0.076	0.15704	0.18713	0.21472	0.12619
8.47±0.099	0.12478	0.19118	0.23908	0.14258
10.7±0.12	0.13431	0.15488	0.18857	0.13180
13.0±0.13	0.15447	0.13220	0.25012	0.11110
15.3±0.14	0.15644	0.15971	0.16896	0.11109
17.6±0.17	0.16143	0.15425	0.16785	0.083112
19.9±0.16	0.18140	0.16855	0.17077	0.085050
22.1±0.17	0.15789	0.16471	0.17715	0.090327

表 14 被験者 F

時間[s]	RMSE			
	2-3	2-4	1-1	1-2
1.70±0.0071	0.22279	0.17260	0.35208	0.26044
4.00±0.024	0.24189	0.17792	0.32935	0.23560
6.28±0.034	0.24863	0.19557	0.30979	0.26292
8.57±0.061	0.17983	0.20283	0.29566	0.25209
10.8±0.78	0.14641	0.21519	0.24820	0.24515
13.1±0.11	0.09653	0.22007	0.28140	0.22729
15.4±0.14	0.15254	0.21498	0.28594	0.26292
17.7±0.16	0.22773	0.21772	0.28906	0.25500
20.0±0.20	0.15751	0.20870	0.20858	0.25894
22.2±0.19	0.13776	0.19915	0.29821	0.26469

表 15 被験者 G

時間[s]	RMSE			
	1-4	1-3	2-2	2-1
1.67±0.039	0.18170	0.30664	0.32469	0.46038
4.07±0.20	0.19932	0.36852	0.26781	0.28980
6.46±0.25	0.17229	0.30215	0.28168	0.35825
8.75±0.25	0.17660	0.25589	0.25532	0.36253
11.0±0.23	0.16385	0.21452	0.22704	0.36066
13.3±0.23	0.29033	0.23350	0.22220	0.44569
15.6±0.25	0.18052	0.26078	0.26086	0.36536
17.8±0.24	0.28886	0.23232	0.25650	0.33886
20.1±0.24	0.31776	0.21947	0.26091	0.33707
22.4±0.25	0.14650	0.21774	0.24793	0.31977

表 16 被験者 H

時間[s]	RMSE			
	1-2	1-1	2-4	2-3
1.68±0.030	0.29576	0.34625	0.16793	0.29347
4.00±0.026	0.36980	0.26603	0.13870	0.27656
6.21±0.038	0.36070	0.25627	0.14362	0.27274
8.48±0.070	0.32641	0.26948	0.13731	0.27743
10.8±0.087	0.34614	0.25988	0.14542	0.27591
13.0±0.085	0.33472	0.25949	0.13870	0.28487
15.3±0.083	0.37685	0.28300	0.15213	0.25878
17.6±0.076	0.31120	0.28887	0.16711	0.29027
19.9±0.082	0.35034	0.28894	0.13576	0.26501
22.2±0.11	0.31041	0.28443	0.12905	0.28816

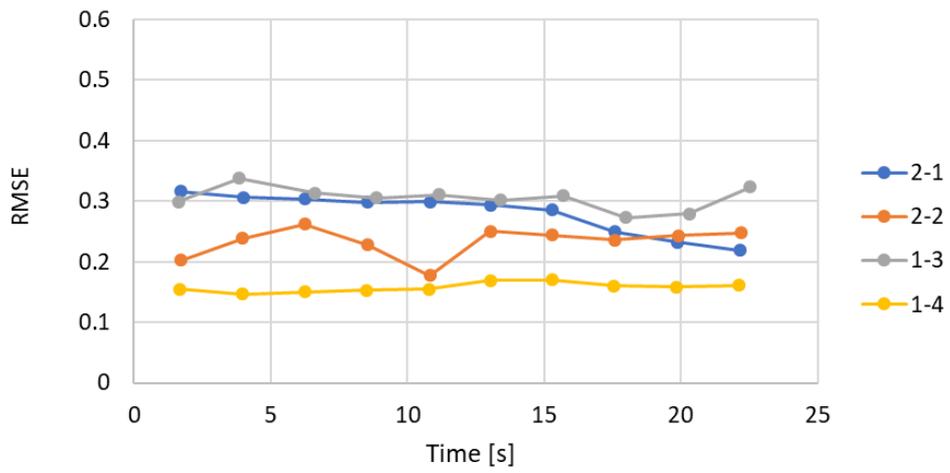


図 5.10 被験者 A

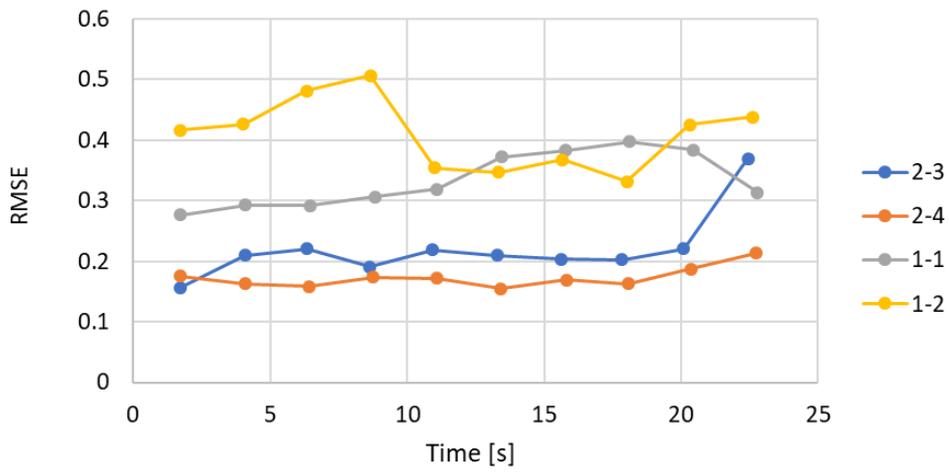


図 5.11 被験者 B

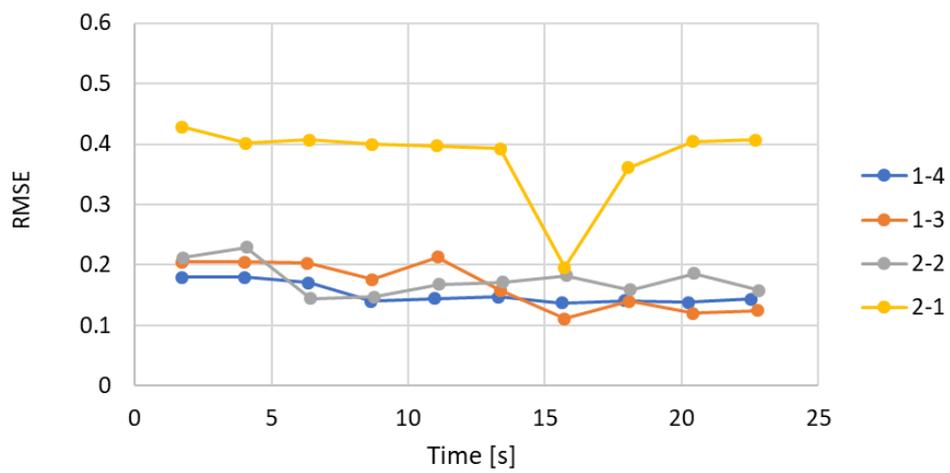


図 5.12 被験者 C

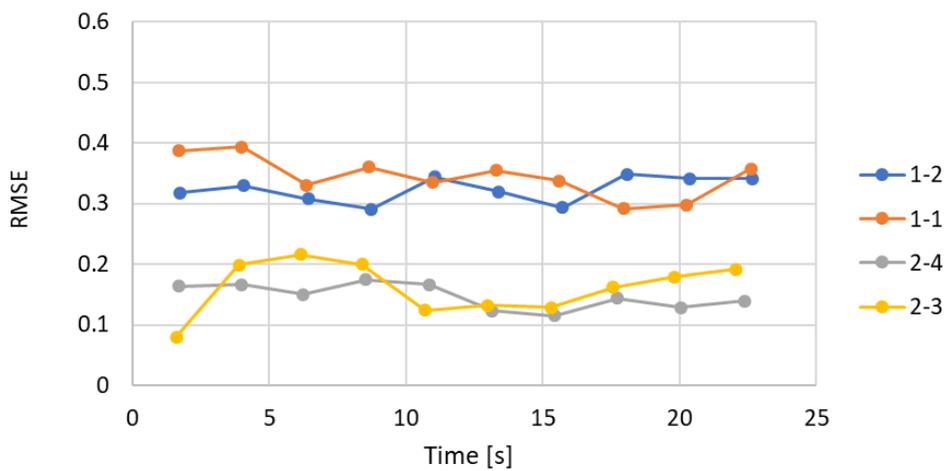


図 5.13 被験者 D

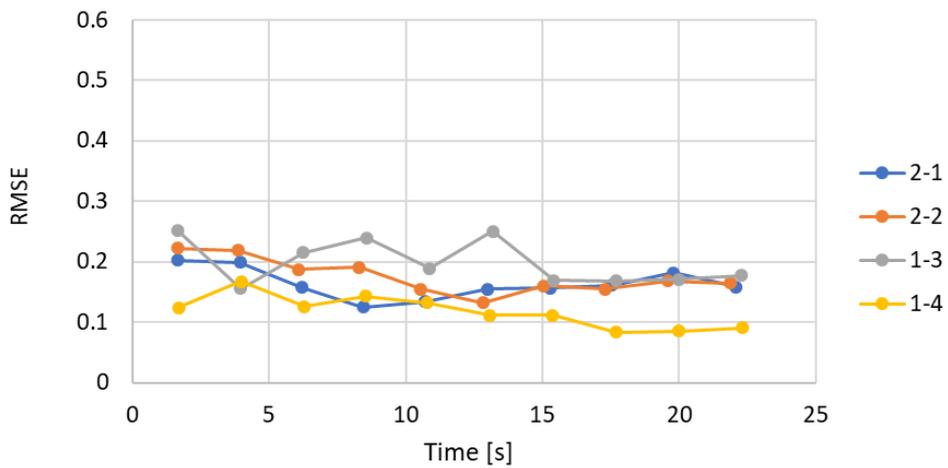


図 5.14 被験者 E

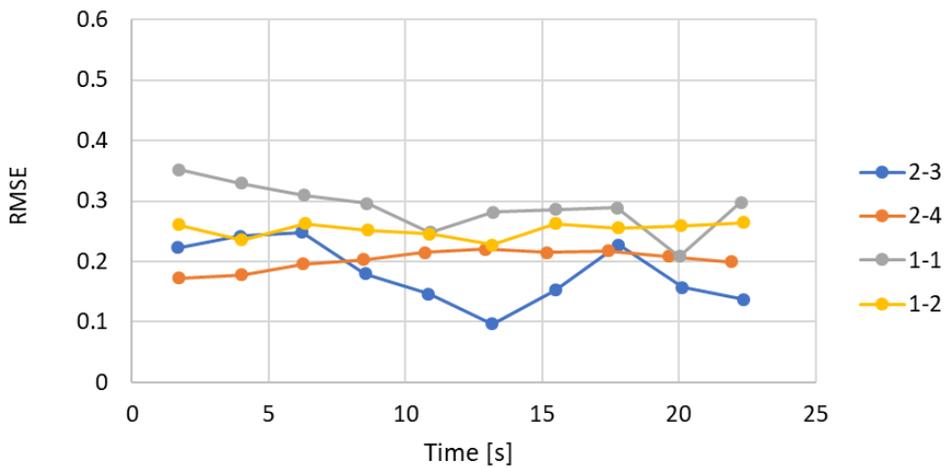


図 5.15 被験者 F

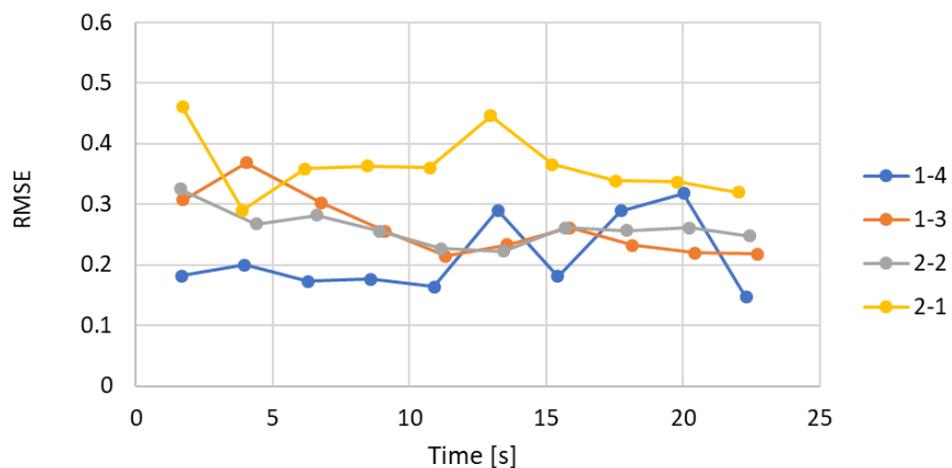


図 5.16 被験者 G

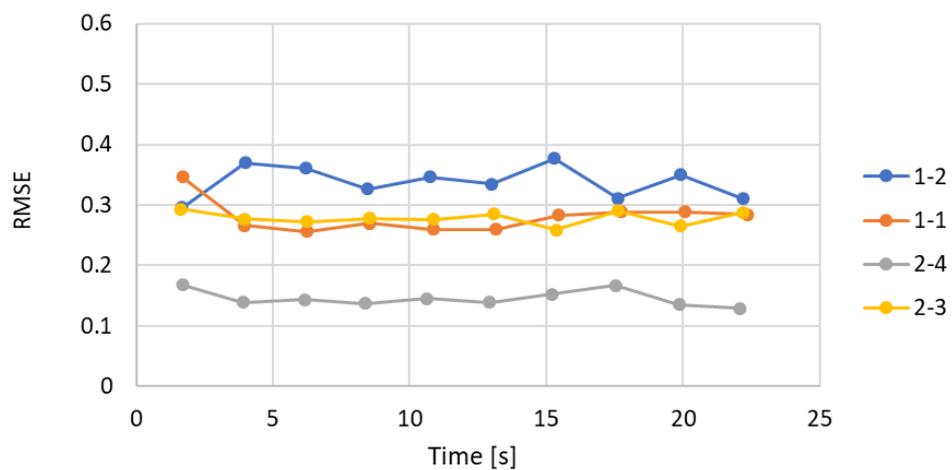


図 5.17 被験者 H

表 17 アンケート結果

被験者	設問								
	1	2	3	4	5	6	7	7-1	8
A	○	2	3	4	4	2	×	-	4
B	×	2	3	4	5	2	×	-	3
C	○	2	2	4	5	4	×	-	4
D	○	2	5	5	5	5	○	5	5
E	×	2	5	5	5	5	×	-	5
F	○	2	5	3	5	4	×	-	3
G	○	2	4	5	4	3	×	-	3
H	×	2	3	5	5	3	×	-	4

5.4 考察

はじめに、実験結果について考察をする。体幹トレーニング 3, 4 は手法を問わず、RMSE が小さい。すなわち、容易なトレーニング姿勢であると言える。RMSE が急激に減少した直後の RMSE 値は大きくなっていることが多い。これは視覚フィードバックの更新速度が 2 秒程度であるため、フィードバックすぐに返ってこないことが起因していると考えられる。

姿勢補正への影響を調べるために、表 18 に RMSE の最小値から初期値を引いた値を初期値で割った割合およびその平均値、表 19 に最小値に達する時間およびその平均値を示す。ここで、時間の平均値を求める際に 0 は除いた。表 18 より、手法 1 よりも手法 2 の方が 7%程度大きいことが分かる。このことから、手法 2 の方が姿勢補正への影響が大きいと推測する。また、表 19 より、最小値に達する時間の平均値には大きな違いがないと考えられる。

表 18 RMSE (初期値と最小値の差) [%]

手法 1				手法 2			
A 1-3	A 1-4	B 1-1	B 1-2	A 2-1	A 2-2	B 2-3	B 2-4
9.69	6.06	0	25.70	44.36	14.28	0	13.62
C 1-4	C 1-3	D 1-2	D 1-1	C 2-2	C 2-1	D 2-4	D 2-3
30.73	84.24	130.81	0	46.47	119.40	68.80	14.58
E 1-3	E 1-4	F 1-1	F 1-2	E 2-1	E 2-2	F 2-3	F 2-4
42.36	0	61.17	49.26	9.31	32.95	62.39	68.16
G 1-4	G 1-3	H 1-2	H 1-1	G 2-2	G 2-1	H 2-4	H 2-3
24.02	42.95	0	35.11	46.13	58.86	30.13	13.41
平均値				平均値			
33.88				40.18			

表 19 RMSE（最小値に達する時間）[s]

手法 1				手法 2			
A 1-3	A 1-4	B 1-1	B 1-2	A 2-1	A 2-2	B 2-3	B 2-4
16.38	2.31	0	16.33	20.47	9.1	0	11.7
C 1-4	C 1-3	D 1-2	D 1-1	C 2-2	C 2-1	D 2-4	D 2-3
13.89	14.01	11.48	0	4.66	13.99	18.32	11.48
E 1-3	E 1-4	F 1-1	F 1-2	E 2-1	E 2-2	F 2-3	F 2-4
13.72	0	2.29	16.01	6.99	16.26	6.78	11.17
G 1-4	G 1-3	H 1-2	H 1-1	G 2-2	G 2-1	H 2-4	H 2-3
20.65	9.62	0	4.54	11.83	2.19	20.35	13.72
平均値				平均値			
11.77				11.93			

次に、アンケート結果について考察する。表 20 にアンケート結果の平均値を示す。被験者の全員が 2 視点 3 次元モデルを姿勢の把握がしやすいと回答していることから、本研究で提案した手法は有効であると推測する。

アンケート結果の平均値が最も低かった回答は設問 6 の“色マーカは姿勢の補正に役立ちましたか”であり、被験者 8 名のうち、2 名が“2”と回答している。さらに、設問 7 の“マーカの大きさが近いほど大きく、遠いほど小さくなることに気が付きましたか”では、1 名を除く 7 名が“気が付かなかった”と回答している。このことから、被験者は実験中マーカなどを確認する余裕はなく、提案手法では 3 次元モデルが提示するため、モデルの姿勢を真似することで姿勢の補正を行っていると考えられる。すなわち、何度も繰り返し本システムを利用し慣れることで、マーカなどを見る余裕も生じ、より効率的な姿勢の補正が行えるのではないかと推測する。

設問 3 の“2 視点になることで、姿勢の把握がしやすくなりましたか”について、設問 6 に次いで結果が低かった。2 視点になることで姿勢の把握がしやすくなるのではないかと考え実装したが、本実験ではよい結果を得ることは出来なかった。設問 4, 5 の結果は高かったため、視点を変更できることや 3 次元モデルを用いてフィードバックを行うことは姿勢の把握しやすさを向上するのに、有効な手法であると考えられる。

表 20 アンケート結果（平均値）

設問					
3	4	5	6	7-1	8
3.75	4.38	4.75	3.5	5（回答者 1 人）	4

第6章

おわりに

本章では、本論文の結論、今後の展望について述べる。

6.1 まとめ

本研究では、カメラ画像を用いた体幹トレーニング支援手法の提案およびシステムを構築した。はじめに、深層学習を利用した研究を用いて、カメラ画像から人体3次元モデルを生成した。次に、その生成した3次元モデルを用いて、トレーニングの目標姿勢と現在の姿勢の違いを把握しやすくするための視覚的フィードバックの生成およびレンダリングの高速化を行い、それらを利用して体幹トレーニングの支援システムを構築した。

関連研究でフィードバック情報として用いられている2次元ボーンを1視点で提示する手法と本研究で提案した3次元モデルを2視点から提示する手法を本システムで実装し、比較する評価実験を行った。目標姿勢の3次元モデルと推定したトレーニング姿勢の3次元モデルの平方平均二乗誤差RMSEを用いて比較を行ったが、大きな違いは見られなかった。しかしながら、同時に実施したアンケート調査によって、提案した手法の有効性があると考えられる。

6.2 今後の展望

本システムの問題点として、視覚的フィードバックの更新速度の遅さが挙げられる。現状では約2秒に一回更新される。実際に利用する際に、フィードバックがすぐに返ってこないため、その時点での目標姿勢との誤差が分からず修正してしまい、目標姿勢から遠ざかってしまう可能性がある。そのため、処理の高速化が望まれる。また、図5.3(c)～図5.6(c)より、トレーニング画像を入力として、生成した目標姿勢の3次元モデルは入力画像と一致しているとは言い難い。より良い支援を行うには、推定精度の向上も望まれる。

そして、本システムを利用する際、必ずディスプレイを見なければならないという問題もある。この問題に対しては、HMD (Head Mounted Display) を利用することで解決できると考える。

謝辞

本研究を進めるにあたり、指導教員をはじめ、様々な方々にご指導、ご協力頂きました。

様々な場面において、お世話になった指導教員の宮田一乗教授に深く感謝いたします。はじめに、M1の11月頃ゼミに長い間参加しておらず、メールを戴きました。その節は、ご心配をおかけして申し訳ございませんでした。研究に関して、適切な助言を下さったおかげで本論文を書き上げることができたと思っております。

研究のご相談にのって下さり、その折々で適切な助言をして下さった謝浩然助教に深く感謝いたします。また、研究室内の情報環境を整えていただいたことで、これまで以上にスケジュールを把握、先生方にご相談しやすくなりました。重ねて感謝申し上げます。

現在、金沢工業大学で講師をしておられる浦正広先生には、昨年度、宮田研究室の助教として、就職活動中の履歴書のご添削や研究計画書のご相談にのっていただき、数多くの助言を戴きました。深く感謝いたします。

副指導教員として、中間発表の際、的確なコメントをしていただき、もう一度考え直す機会を戴きました西本一志教授に深く感謝いたします。考え直すことで、様々な視点から考えることができ、研究の助けとなりました。

また、急な計画にも関わらず、本研究の評価実験に快く受けて頂いた学生の皆様に深く感謝いたします。

宮田研究室の皆様には、大変お世話になりました。皆様のおかげでJAISTでの2年間は楽しいものとなりました。

I would like to express my thanks to Ms. Angjoo Kanazawa who is first author of research basis for my thesis. I could facilitate my research because she promptly replied my question regarding her research.

参考文献

- [1] T. R. Walter, “WORLDWIDE SURVEY OF FITNESS TRENDS FOR 2018: The CREP Edition,” ACSM's Health & Fitness Journal, Volume 21 - Issue 6 pages 10-19, 2017.
- [2] J. Stephenson , A. M. Swank, “Core Training: Designing a Program for Anyone,” Strength and Conditioning Journal, Volume 26, Number6, pages 34-37, 2004.
- [3] 樽川香澄, 井上智雄, 岡田謙一, サッカーの戦略会議を支援する複数視点を用いた協調作業空間, 情報処理学会, Vol.1 No.1 pages 19-26, 2013.
- [4] A. Newell, K. Yang , J. Deng, Stacked Hourglass Networks for Human Pose Estimation, ECCV, pages 483-499, 2016.
- [5] Z. Cao, G. Hidalgo, T. Simon , Y. Sheikh, “OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields,” arXiv preprint arXiv:1812.08008, 2018.
- [6] G. Rogez, P. Weinzaepfel , C. Schmid, LCR-Net: Localization-Classification-Regression for Human Pose, Computer Vision Pattern Recognition, pages 1216-1224, 2017.
- [7] A. Kanazawa, M. J. Black, D. W. Jacobs , J. Malik, “End-to-end Recovery of Human Shape and Pose,” Computer Vision and Pattern Recognition (CVPR), 2018.
- [8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville , Y. Bengio, “Generative Adversarial Nets,” Advances in Neural Information Processing Systems, pages 2672-2680, 2014.
- [9] 高久大輔, 中島克人, “Kinect を用いた筋力トレーニング支援システム,” 情報処理学会第 77 回全国大会, pages 2-437--2-438, 2015.
- [10] 岡本勝, 磯村智将, 松原行宏, “姿勢推定手法を活用したリアルタイム運動訓練支援環境,” 第 30 回人工知能学会全国大会, 2016.
- [11] 松村海沙, 小池崇文, “Head Mounted Display を用いた三人称視点によるフォーム改善システム,” 情報処理学会第 78 回全国大会, pages 4-357--4-358, 2016.
- [12] M. Andriluka, L. Pishchulin, P. Gehler , B. Schiele, “2D Human Pose Estimation: New Benchmark and State of the Art Analysis,” Computer Vision and Pattern Recognition (CVPR), pages 3686-3693, 2014.

- [13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár , L. C. Zitnick, “Microsoft COCO: Common Objects in Context,” ECCV, 2014.
- [14] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll , M. J. Blasck, “SMPL: A Skinned Multi-Person Linear Model,” ACM Trans. Graphics (Proc. SIGGRAPH Asia), 34(6) pages 248:1--248:16, 16, Oct. 2015.
- [15] M. M. Lopper , M. J. Black, “OpenDR: An Approximate Differentiable Renderer,” In Computer Vision - ECCV 2014, Springer, Heidelberg, D. Fleet, T. Pajdla, B. Schiele, and T.Tuyelaars, Eds., vol. 8695 of Lecture Notes in Computer Science, 2014.
- [16] 木場克己, 体幹力を上げるコアトレーニング, 成美堂出版, 2012.