オクルージョン領域をリアルタイムに復元するハンドトラッキングシステムの開発

竹内康太[†1]　川上童夢[†1]　岡夏樹[†2]　田中一晶[†1]

概要：機械学習技術の進歩により，RGB カメラの映像のみでも高い精度の姿勢推定が可能になってきた．しかしながら，そのような姿勢推定モデルでは，他の物体によって推定対象となる部分が遮蔽されると，その推定精度が低下するか，推定対象の追従が途切れてしまう問題が生じる．特に手の姿勢推定においては作業中に物体で手が隠れてしまうと，この問題の影響が顕著に表れる．この問題に対し，動画修復技術によって手を遮蔽する物体を除去し，物体で隠蔽されていない手の映像を復元することで，既存の姿勢推定モデルをそのまま使用しても安定して手の追従及び姿勢推定が可能になると考えた．本研究では，手を遮蔽する物体の領域をリアルタイムに自動的に特定し，その領域の手映像を復元するシステムを開発した．さらに，手映像の復元に特化した動画修復モデルの構築も行った．この提案システムを用いて遮蔽物を除去し，手映像を復元した場合，遮蔽物を除去する前の映像と比較して，手の追従精度および姿勢推定精度の両方で改善が見られ，遮蔽物のない状態に近い姿勢推定を行えることを示した．

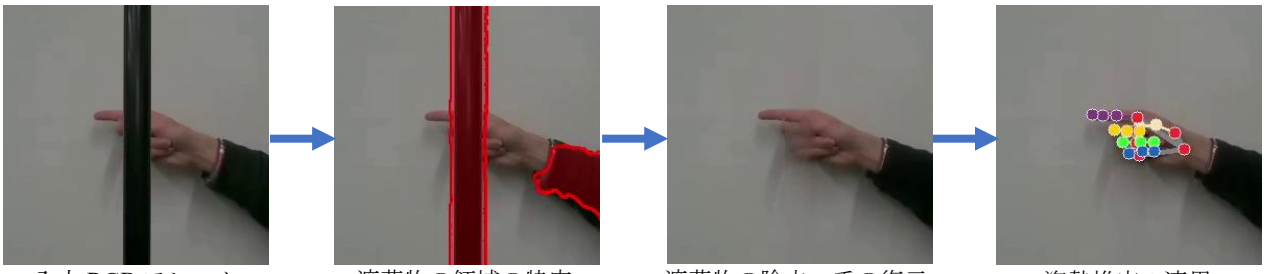


図 1 提案システムの概略図

1. 緒言

手の姿勢推定技術は，機械学習技術の進歩により，精度と手軽さが向上している。AR・VR 空間でのアバタの操作[1]，ジェスチャによる非接触操作[2]，遠隔でのロボットハンドの直感的な操作[3]，自動車の運転安全支援[4]など，人とコンピュータのインタラクションにおいて幅広く活用されている．

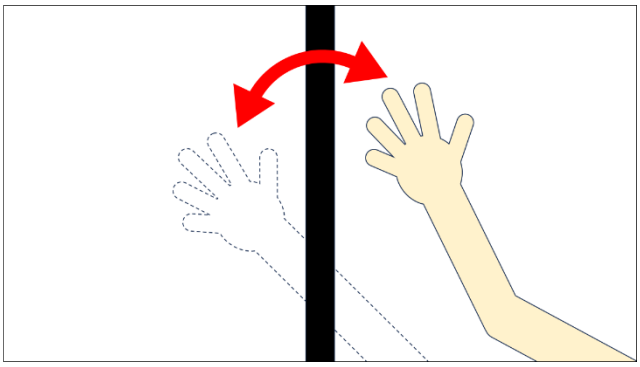
姿勢推定の手法として，センサを備えた専用のデバイスを使用して計測する方法[5][6][7][8]や，反射マーカを取り付けて複数のカメラで追従する方法[9]の他，カメラ画像または映像のみを用いた方法[10][11]がある．センサやマーカを使用する姿勢推定システムでは，高い精度の推定が可能である一方で，専用のハードウェアが高価である，デバイスやマーカの着脱に手間がかかる，自然な触覚が得られない，といった課題がある．

これに対し，カメラ映像のみを用いた手の指定推定技術[10][11]では，Web カメラ等の安価なデバイスだけで素手で姿勢推定を行うことが可能であり，より自然なインタラクションを可能にする．しかしながら，オクルージョンが発生すると（推定対象が他の物体によって遮蔽されると）姿勢推定の精度が低下するか，または推定対象の追従が途切れてしまい，システムのスムーズな操作が妨げられるという問題が発生する．

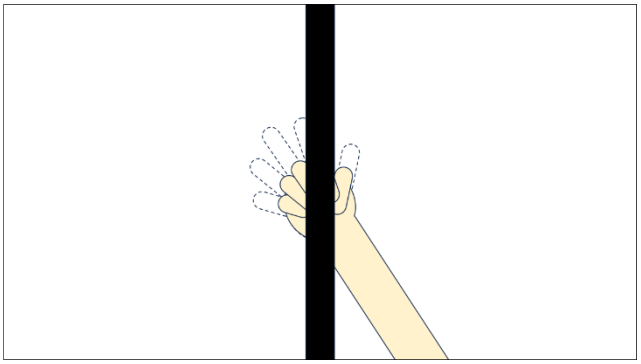
この問題に対し，本研究では，カメラ映像による手の姿勢推定において，手を遮蔽する物体の領域をリアルタイムで自動的に特定し，その領域の手映像を復元するシステムを提案する．このシステムの概略図を図 1 に示す．これにより，オクルージョンが発生してもカメラ映像のみを用いた姿勢推定モデルをそのまま使用することを可能にし，自然かつ途切れのないインタラクションを可能にする．

システムのうち，手の遮蔽物の領域特定部分の実装には，領域分類技術を用いた．手を遮蔽する物体を除去し，隠れている手を復元するために，動画修復モデル DSTT[12]を高

†1 京都工芸繊維大学
†2 宮崎産業経営大学

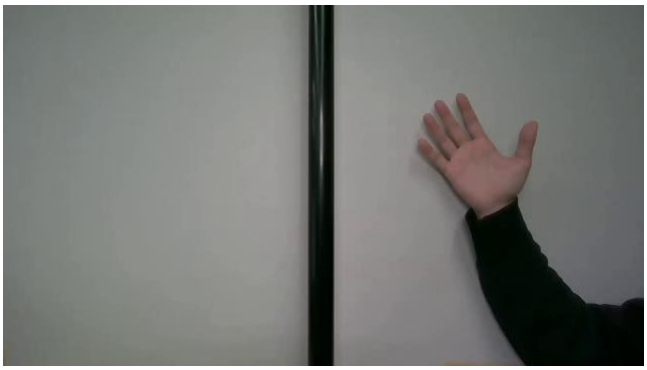


(a) 位置変更タスク

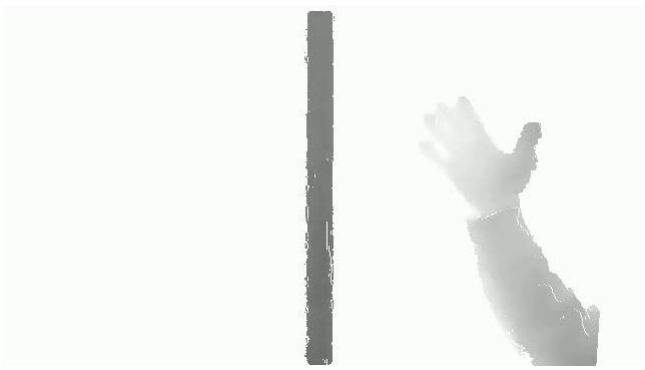


(b) 姿勢変更タスク

図 3 タスクのイメージ

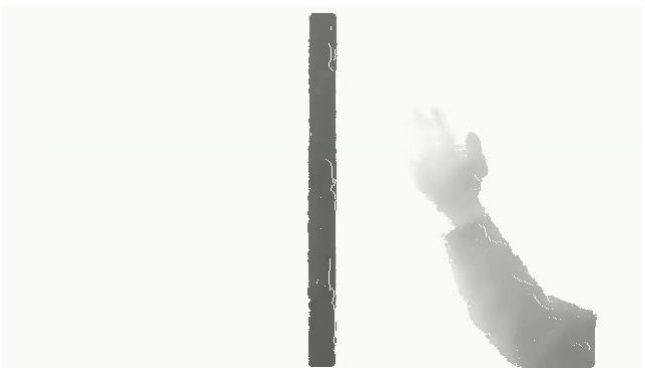


(a) RGB（実際のポール）



(b) 深度（実際のポール）

(c) RGB（合成ポール）



(d) 深度（合成ポール）

図 2 撮影された RGBD 映像のフレーム

速化し，リアルタイム処理を可能にしたモデル[13]を用いた．さらに，このモデルを手のデータセットについて学習することで，手の復元に特化したモデルを作成した．オクルージョンに頑健な手の姿勢推定モデルも研究されている[14][15]が，本手法は，システムを遮蔽物の領域分類，遮蔽物の除去，姿勢推定に分けてモジュール化することにより，既存のモデルを活用できる他，モジュールごとに性能の改善を図ることが可能である．

第 2 章では，実験のために収集した手の映像データについて説明し，第 3 章では，本研究で構築したシステムついて説明する．第 4 章で，提案手法の評価のための指標を示し，第 5 章に実験内容とその結果を示す．実験の結果について第 6 章で考察する．最後に，第 7 章で本研究の成果と今後の展望を述べる．

2. 手の映像のデータ収集

京都工芸繊維大学の学生 20 名（男性 18 名，女性 2 名）を被験者とし，手の姿勢推定において，ポールによってオクルージョンが発生する状況を想定した 2 種類のタスクの RGBD 映像を撮影した．正解データとして，ポールによって手が遮蔽されていない映像を得るために，実際にポールを用いた映像に加え，ポールのない状態の映像も撮影し，後付けでポールを合成して加えた映像を作成した．

2.1 タスク

位置変更タスクと姿勢変更タスクの 2 種類のタスクを設定した．

位置変更タスク：ある手の姿勢を取りながら，手をポール

の後ろで往復させる. それを複数の手の姿勢について行う.
姿勢変更タスク： ポールで手が隠れている状態で手の姿勢を変える.
それぞれのタスクのイメージを図 3 に示す.

さまざまな手の姿勢の映像を収集するために，ひらがな 46 種類，アルファベット 26 種類の指文字のうち，重複するものを除いた 44 種類を用いた．20 人の被験者に 1 人当たり 10 種類の文字を用い，偏りがないよう，44 種類の文字のうち 24 種類は 5 回ずつ，20 種類は 4 回ずつ使用されるように割り当てた．さらに，被験者ごとに文字の順番をランダムに並べ替えた.

RGBD 映像の撮影には, Intel RealSense Depth Camera D455 を用いた. 撮影された RGBD 映像のフレームを図 2 に，撮影に使用した環境を図 4 に示す.

図 4 撮影環境

3. 提案システムの構成

提案システムは, 遮蔽物の領域分類, 遮蔽領域（手映像）の復元，手の姿勢推定の適用の 3 つの主要なコンポーネントから構成される．以下，各コンポーネントの処理の詳細と，システム全体の高速化のための実装上の工夫について説明する.

3.1 遮蔽物の領域分類

遮蔽物の領域分類は，手の領域の特定，手を遮蔽する物体の領域の特定の 2 段階で構成される．以下，それぞれの処理を説明する.

初期フレームにおいて，手が他の物体によって遮蔽されていない状態で, RGB フレームに対して手の姿勢推定モデル MediaPipe Hands[10]を適用することにより手関節点位置を得る．それを領域分類モデル MobileSAM[16]にプロンプトとして与えることにより, 最初の手のマスクを生成する. このマスクを用いて，動画物体領域分類モデル XMem++[17]を初期化する．後続のフレームについては，XMem++に RGB フレームのみを入力することにより手のマスクを得る. これにより, XMem++の記憶機構によって，オクルージョンが発生する場合においても，手のマスクの追跡を維持することができる.

以下の 3 つを掛け合わせることにより，遮蔽物を指す点の集合を得る.

- 深度マスク（図 5 (a)）: 深度画像のうち，手のマスク内の深度値の平均よりも値が小さい領域を取得することにより得られる,（カメラから見て）手よりも前にある領域
- 周囲マスク（図 5 (b)）: 手のマスクに対し，モルフォロジー変換を用いることで得た手の周囲の領域
- グリッドポイント（図 5 (c)）: 画像の縦横それぞれ 30px おきにグリッド上に取った点

得られた点を MobileSAM にプロンプトとして与えることで，遮蔽物のマスクが得られる．得られた点と遮蔽物のマスクを図 5（d）に示す.

3.2 遮蔽領域（手映像）の復元

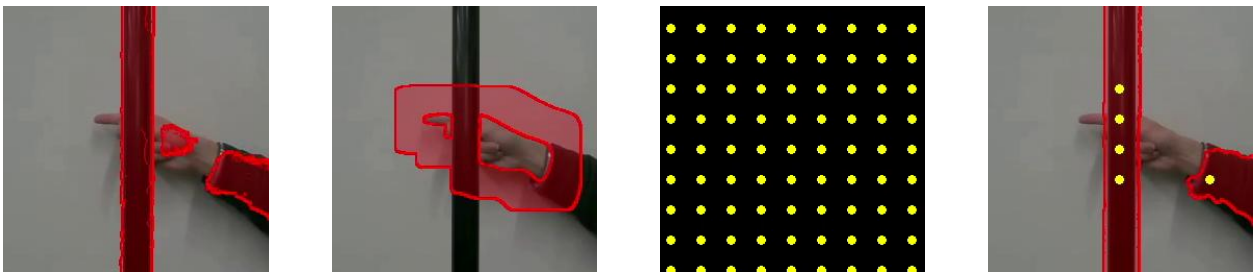
RGB フレームと生成された遮蔽物のマスクを動画修復モデルの入力として与えることで，遮蔽物を除去し，隠れている手を復元する. 動画修復モデルには， DSTT[12]をリアルタイム処理可能にし，記憶機構を導入することにより高速化したモデル[13]を使用した.

3.3 手の姿勢推定の適用

遮蔽物を除去した映像に対し，MediaPipe Hands[10]を適用し，手の姿勢推定を行う．これにより，オクルージョンが発生している状況下でも，あたかも手が遮蔽されていないかのように姿勢推定を行うことが可能となる.

3.4 手映像のための動画修復モデルの学習

作成した手のデータセットのうち，ポールのない状態で撮影した映像の，被験者 20 名中 10 名分のデータを，3〜5 秒程度に分割し 447 件のデータとした上で，データ拡張と背景合成を行った．それを元々の DSTT の学習に使用され



(a) 深度マスク　(b) 周囲マスク　(c) グリッドポイント　(d) 得られた点と遮蔽物のマスク

図 5 遮蔽物のマスク獲得の過程

ている YouTube-VOS データセットに追加し再学習した HanDSTT モデルを作成した．

データ拡張は，輝度，彩度，手の大きさ，手の回転の 4 種類に関して変換を行った．パラメータは以下の通りである．

- 輝度: 1 倍，1.5 倍，2 倍の 3 通り
- 彩度: 0.5 倍，1 倍，1.5 倍の 3 通り
- 大きさ: 1 倍，1.25 倍，1.5 倍の 3 通り
- 回転: 0°，180°の 2 通り

これにより，$3 \times 3 \times 3 \times 2 = 54$通りのデータ拡張が行われ，の$54 \times 447 = 24138$件のデータが得られた．

合成する背景には，屋内背景データセット[18]の 15620 種類を用い，24138 件のデータに 1〜2 回使用されるように割り当てた．データ拡張が行われない組み合わせ（輝度 1 倍，彩度 1 倍，大きさ 1 倍，回転 0°）のみ，背景を合成しない，元々の背景のデータも用いた．データの数はデータ拡張と合わせて 24585 件，YouTube-VOS の 3471 件データと合わせて 28056 件となった．

学習に使用されるマスクは，元々の DSTT ではランダムな位置に生成されるが，HanDSTT では，手の復元の精度を高めるために，手にマスクが被りやすいようにマスクの生成位置を制御した．具体的には，追加した手のデータセットそれぞれに対し MediaPipe Hands を用いて手の姿勢推定を行い，動画内で手が動く範囲を矩形として取得し，マスクが生成される範囲をこの範囲に限定した．YouTube-VOS データセットに対しては，元々の DSTT のマスク生成方法を用いた．

元々の DSTT の学習設定に従い，Adam オプティマイザ[19]をパラメータ$\beta_1 = 0.9$, $\beta_2 = 0.999$で用い，損失関数には，穴領域と有効領域それぞれの L1 再構成損失，および敵対的損失を組み合わせたものを使用した．具体的には以下の式で表される．

$$L = L_{hole} + L_{valid} + 0.01L_{adv}$$

ここで，L_{hole}は穴領域の再構成損失，L_{valid}は有効領域の再構成損失，L_{adv}は敵対的損失を表す．学習率 0.0001，バッチサイズ 8 で 100 万回の学習を行い， loss が 0.173 で最小となった学習回数 937,500 回の時点のものを結果として用いた．

3.5 ハイパーパラメータの設定

リアルタイムの DSTT では，修復対象の最新フレームに加えて，近傍の連続する数フレームと，近傍以外の広い範囲（大域）のフレームを参照して修復に利用する．ハイパーパラメータとして，それぞれの参照するフレームの数を指定できる．参照するフレームの数が多いほど，高い精度の修復が期待される．本研究では，精度とリアルタイム性のトレードオフを考慮し，近傍 10 フレーム，大域 5 フレームとした．

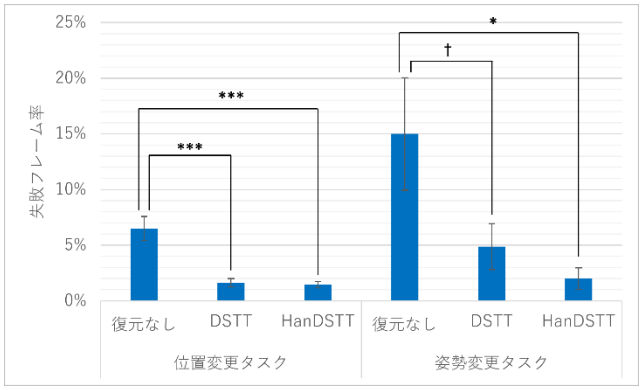
MediaPipe Hands[10]では，手の検出と追従に対してそれぞれ 0.0〜1.0 の確信度スコアが算出され，それらに対してしきい値が設けられる．しきい値を下回る場合は推定失敗とみなされ，推定結果は出力されない．パラメータを小さくすれば，推定精度が悪くなることを許しつつ，姿勢推定が継続しやすくなり，逆に大きくすれば，姿勢推定が途切れやすくなる．本研究では，いずれのパラメータも規定値である 0.5 で用いた．

処理する映像の解像度は，横 640px×縦 360px とした．元々の DSTT の入出力サイズは横 432px×縦 240px であるが，入力と出力のテンソルサイズを横 640px×縦 360px に合うよう変更し，チェックポイントから読み込んだ重みのサイズもそれに合わせて変形させた．

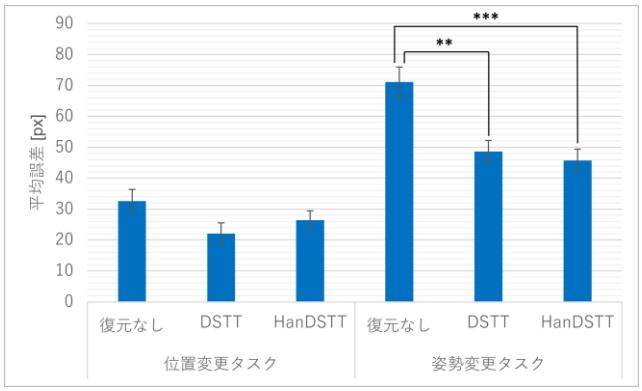
3.6 深度推定・領域分類・動画修復モデルの高速化

機械学習ライブラリの多くでは，標準で推論に 32bit 精度の浮動小数点が用いられるが，精度を削減することにより，推論を高速化する，量子化と呼ばれる手法[20]がある．リアルタイム性の確保のため，深度推定，領域分類，動画修復の全てにおいて，16bit 浮動小数点を用いて推論を行った．この結果，各コンポーネントにおいて，32bit の場合と比較して 1.5 倍程度高速された． 深度推定は，量子化に加え，NVIDIA TensorRT による最適化により，さらなる高速化を行った．これにより，量子化前と比較して，1.8 倍程度高速化された．

Intel Core i9-14900KF CPU，及び 2 基の NVIDIA GeForce RTX 4090 GPU 搭載したマシンで提案システムを実行した．計算量に関してボトルネックとなる動画修復に一方の GPU を占有させ，残りの深度推定と領域分類は全てもう一方の GPU で処理した．その結果，深度推定，領域分類，動



(a) 失敗フレーム率



(b) 平均誤差

図 6 復元なし条件，DSTT 条件，HanDSTT 条件の比較（† $p < 0.1$, ∗ $p < 0.05$, ∗∗ $p < 0.01$, ∗∗∗ $p < 0.001$）

画修復の全てが 30fps 以上で動作し，リアルタイムで使用するために十分なフレームレートを達成した．

4. 評価指標

手の姿勢推定の精度を評価するための指標として，以下の数式で表される，失敗フレーム率と平均誤差の 2 つを用いた．

$$\text{失敗フレーム率} = \frac{\text{姿勢推定が途切れたフレーム数}}{\text{オクルージョンが発生したフレーム数}}$$

$$\text{平均誤差} = \frac{\text{推定された手関節点位置の正解データとのユークリッド距離の和}}{\text{オクルージョンが発生したフレーム数}}$$

ここで手関節点位置の正解データは，ポールを合成する前の映像に対して姿勢推定を行うことで得られたものを指す．

MediaPipe Hands は x 座標と y 座標をそれぞれ入力映像の縦横のピクセル数について正規化して出力するが，入力映像の縦横のピクセル数が異なる場合，結果の中で x 座標と y 座標の誤差の重みに偏りが出てしまう．そのため MediaPipe Hands の出力に入力画像の縦横のピクセル数を掛け合わせることで正規化を元に戻し，ピクセル単位で誤差を求めた．また，MedaiPipe Hands では x 座標と y 座標に加えて手首を基準点とした相対的な深度を示す z 座標を出力するが， x 座標と y 座標とは基準点と単位が異なるため，ここでは z 座標は用いないこととした．

5. 実験

提案手法の有効性の評価のために，以下の 3 条件の映像に対して姿勢推定を行い，失敗フレーム率と平均誤差を算出し比較した．

復元なし条件：手の遮蔽物を除去していない映像
DSTT 条件：動画修復モデル DSTT で遮蔽物を除去した映像
HanDSTT 条件：手の復元に特化した動画修復モデル HanDSTT で遮蔽物を除去した映像

評価のためのデータセットには，作成した手のデータセットから，被験者 20 名のうち HanDSTT の学習に使った 10 名分を除いた残りの 10 名分のデータを用いた．

復元なし条件，DSTT 条件，HanDSTT 条件の 3 条件の映像に対して姿勢推定を行い，失敗フレーム率と平均誤差を算出した．3 条件を水準とした復元要因に関して一元配置分散分析をタスク･指標の組み合わせそれぞれに対し行い，有意差が見られた場合には Tukey’s HSD 検定による多重比較を行った．その結果を図 6 に示す．ただし，エラーバーは標準誤差を示す．

失敗フレーム率（図 6（a））について，位置変更タスクにおいては，復元要因の主効果が有意であり（$F(2,27) = 18.160, p < .001$），復元なし条件よりも DSTT 条件の方が小さく（$p < .001$），復元なし条件よりも HanDSTT 条件の方が小さかった（$p < .001$）．姿勢変更タスクにおいては，復元要因の主効果が有意であり（$F(2,27) = 4.553, p < .05$），復元なし条件よりも DSTT 条件の方が小さく（$p = .083$）．復元なし条件よりも HanDSTT 条件の方が小さかった（$p < .05$）．平均誤差（図 6（b））について，姿勢変更タスクにおいてのみ，復元要因の主効果が有意であり（$F(2,27) = 11.820, p < .001$），復元なし条件よりも DSTT 条件の方が小さく（$p < .001$），復元なし条件よりも HanDSTT 条件の方が小さかった（$p < .001$）．位置変更タスクでは，復元要因の主効果は有意でなかった（$F(2,27) = 2.336, p = 0.116$）．いずれの場合においても，DSTT 条件と HanDSTT 条件の間に有意な差は認められなかった．

6. 考察

領域分類技術により遮蔽物の領域を特定し，動画修復技術により除去することで，失敗フレーム率・平均誤差の両指標が減少する結果が得られ，手の姿勢推定精度を向上させることができた．失敗フレーム率は，復元なしと比較して，平均 70%程度の大幅な改善が得られたが，平均誤差では平均 30%程度の改善に留まった．しかしながら，姿勢推定に失敗している間は平均誤差を算出していないため，失敗フレーム率の小さい復元動画に不利な指標であることを考慮すれば，平均誤差も十分な改善が得られたと言える．

手映像を追加して学習した動画修復モデル HanDSTT を

用いても，元々の DSTT を用いた場合と比較して，失敗フレーム率と平均誤差の指標において統計的に優位な差は見られなかった．その原因として，DSTT が事前に物体の形状や特徴についての知識を学習するのではなく，与えられた映像の中で空間的・時間的一貫性を利用して欠損部分を補完する方法を学習しているために，手映像に特化した学習が有効に機能しなかった可能性が考えられる．

7. 結言

本研究では，遮蔽物の領域を自動的に特定し，動画修復技術によって遮蔽物を除去，隠れている手を復元するシステムにより，オクルージョン発生下での手の姿勢推定精度を改善する手法を提案した．実験の結果，提案手法を用いることで，遮蔽物を除去しない場合と比較して，姿勢推定の失敗フレーム率を 80%程度，平均誤差を 30%程度改善することができた．メモリ機構を導入した動画修復モデルの使用や，量子化などの工夫によって，深度推定，領域分類，動画修復の全ての処理を 30fps 以上で実行することができ，システムをリアルタイムに実行することが可能であることを示した．

本研究の成果により，AR・VR 空間でのアバタの操作，ロボットハンドの遠隔操作，ジェスチャ認識等の手の姿勢推定に関するアプリケーションにおいて，オクルージョンを意識することなく，インタラクションを継続することが可能になる．本研究の提案手法は，手の姿勢推定に限らず，人体の姿勢推定や物体追跡等の他のコンピュータビジョンアプリケーションにおけるオクルージョン問題を解決することが期待できる．

謝辞　本研究は，JSPS 科研費 JP22K12126 の支援を受けた．

参考文献

[1] VR・MR・AR 用センサーとしてモーションキャプチャを活用，株式会社スパイス，https://mocap.jp/vr/（参照 2024-01-27）．
[2] C. Yang, Yujeong Jang, J. Beh, D. Han and H. Ko, "Gesture recognition using depth-based hand tracking for contactless controller application," 2012 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 2012, pp. 297-298, doi: 10.1109/ICCE.2012.6161876.
[3] M. Tomida and K. Hoshino, “Wearable device for high-speed hand pose estimation with a ultrasmall camera,” Journal of Robotics and Mechatronics, vol.27, no.2, pp.167–173, 2015.
[4] 日本電気株式会社，姿勢推定技術による運行安全支援，車外・車室内状況見守りソリューション，https://jpn.nec.com/manufacture/jidousya/img/mimamori/pose_estimation_leaf.pdf（参照 2024-01-27）．
[5] K. Arimatsu and H. Mori, “Evaluation of machine learning techniques for hand pose estimation on handheld device with proximity sensor,” Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 2020.
[6] COBRA GLOVES for Vicon, Optitrack & Vive, AiQ-Synertial Ltd, https://www.synertial.com/cobra-gloves-1（参照 2024-01-27）．
[7] MANUS, OptiTrack, https://www.optitrack.jp/products/finger/finger01.html（参照 2024-02-04）．
[8] StretchSense, OptiTrack, https://www.optitrack.jp/products/finger/finger02.html（参照 2024-02-04）．
[9] OptiTrack モーションキャプチャ ，OptiTrack，https://www.optitrack.jp（参照 2024-01-27）．
[10] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, “Mediapipe hands: On-device real-time hand tracking,” 2020.
[11] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, “Openpose: Realtime multiperson 2d pose estimation using part affinity fields,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.43, no.1, pp.172–186, 2021.
[12] Liu, Rui, et al. "Decoupled spatial-temporal transformer for video inpainting." arXiv preprint arXiv:2104.06637 (2021).
[13] Thiry, Guillaume, et al. "Towards Online Real-Time Memory-based Video Inpainting Transformers." arXiv preprint arXiv:2403.16161 (2024).
[14] F. Mueller, D. Mehta, O. Sotnychenko, S. Sridhar, D. Casas, and C. Theobalt, “Realtime hand tracking under occlusion from an egocentric rgb-d sensor,” 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, Oct. 2017.
[15] H. Xu, T. Wang, X. Tang and C. -W. Fu, "H2ONet: Hand-Occlusion-and-Orientation-Aware Network for Real-Time 3D Hand Mesh Reconstruction," 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 2023, pp. 17048-17058, doi: 10.1109/CVPR52729.2023.01635.
[16] Zhang, Chaoning, et al. "Faster segment anything: Towards lightweight sam for mobile applications." arXiv preprint arXiv:2306.14289 (2023).
[17] Bekuzarov, Maksym, et al. "Xmem++: Production-level video segmentation from few annotated frames." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023.
[18] A. Quattoni, and A.Torralba. Recognizing Indoor Scenes. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.
[19] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In 3rd International Conference on Learning Representations, ICLR, 2015.
[20] Nagel, Markus, et al. "A white paper on neural network quantization." arXiv preprint arXiv:2106.08295 (2021).