

人物姿勢推定における関節寄与度推定に基づく 観測偏りに頑健なイベントカメラ駆動遅延補償

大武 一平^{1,a)} 北野 和哉¹ 櫛田 貴弘² 藤村 友貴¹
前島 謙宣³ 久保 尋之⁴ 船富 卓哉⁵ 向川 康博¹

概要

オンラインでの人物姿勢推定は推論時間により入力画像の取得時刻と姿勢の出力時刻に差が生じ、姿勢出力のずれが生じる。これを補償するイベントカメラを用いた姿勢更新技術は、イベント位置の単純参照では視点上で重なるイベントの発生源でない関節まで誤って更新し、また一様でない照明に起因するイベント観測の偏りが更新を不安定にする。本研究では各イベントの寄与関節を推定して更新先を選別し巻き込みを抑制する。さらに人物・関節別の重み付き集約に基づき撮影環境に応じた更新量を回帰し、イベントの欠落や偏りを更新量推定に織込む。実験を通じ、以上が巻き込みと観測偏りの双方に頑健である事を確認した。

1. はじめに

近年、ダンスエンタテインメントなどで人物動作に同期した映像エフェクトや投影演出をリアルタイムに重畳する応用が広がっている。このような演出では、従来のフレームベース人物姿勢推定 [1], [4], [5], [11] をオンラインで用いると、特に速い動作で演出に時間遅れが生じる。これは、推定処理時間に起因する遅延により、姿勢推定器への入力取得時刻の姿勢と出力時刻の姿勢がずれるためである。

これに対し、輝度変化を低遅延かつ高時間分解能で出力するイベントカメラ [2], [6], [7] を用い、推論時間中に得られるイベントから姿勢推定出力を更新して遅延を補償する枠組みが提案されている [8]。これは、イベントカメラから人物姿勢を直接推定する系 [3], [10], [12], [13] とは異なり、既存の姿勢推定器の推論遅延を前提として、その出力を遅延補償することに特化している。しかしダンス動作では、セルフ・人物間オクルージョンが頻発し、イベントの発生位置を単純に関節へ反映する更新 [8] では、関節対応

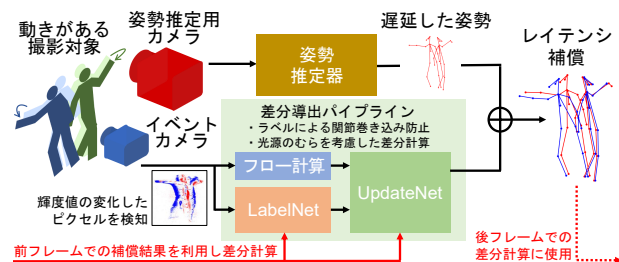


図 1 提案手法の概要。姿勢推定器から得られる遅延姿勢に対し、イベントカメラが出力する輝度変化イベントを用いてフローを計算し、LabelNet で関節寄与度を付与したうえで、UpdateNet が関節ごとの更新量を推定することでレイテンシを補償する。

の混同によりイベント発生源とは異なる関節を更新してしまう巻き込みが生じやすい。また、照明や反射といった撮影環境依存の要因でイベント発生が不均一となるため、人間とカメラの相対位置が変化するとイベントの出力に変化が生じ、安定した更新量の導出が難しい。

そこで本研究では図 1 に示すように、イベントが各関節の運動にどの程度寄与するかを推定しつつ、観測の不均一性を考慮して関節の更新量を推定する 2 段階の学習ベース更新を遅延補償手法に導入する。まず各イベントへ関節寄与度を付与し、それに基づく関節別・グローバルな重み付き集約により更新特徴を抽出する。次に、集約した更新特徴とイベント頻度を表す信頼度から更新量を回帰し、動きに伴うイベント出力が十分に得られていない場合でも安定に遅延補償を行う。さらに推論時には直前フレームの更新後姿勢を相対位置の基準に用いることで、姿勢推定器の処理と並行し差分導出を進め、遅延増大を避ける。

2. 人物姿勢推定のイベントカメラ駆動遅延補償

2.1 姿勢推定の処理時間による出力のずれ

時刻 t における現実の姿勢が $\mathcal{J}_t$ と表現される動作シーケンスにおいて、任意のフレームベースカメラおよび姿勢推定器を用いてリアルタイムに推論を繰り返す系を考える。 n 回目の姿勢推定開始時の時刻を S_n 、推定完了時の時刻を G_n とする。また、姿勢推定器が推論した姿勢を $\mathbf{J}_n$ とする。このとき、推論姿勢 $\mathbf{J}_n$ は時刻 S_n における現実の

¹ 奈良先端科学技術大学院大学

² 立命館大学

³ 株式会社オー・エル・エム・デジタル

⁴ 千葉大学

⁵ 京都大学

^{a)} otake.ippei.oj2@is.naist.jp

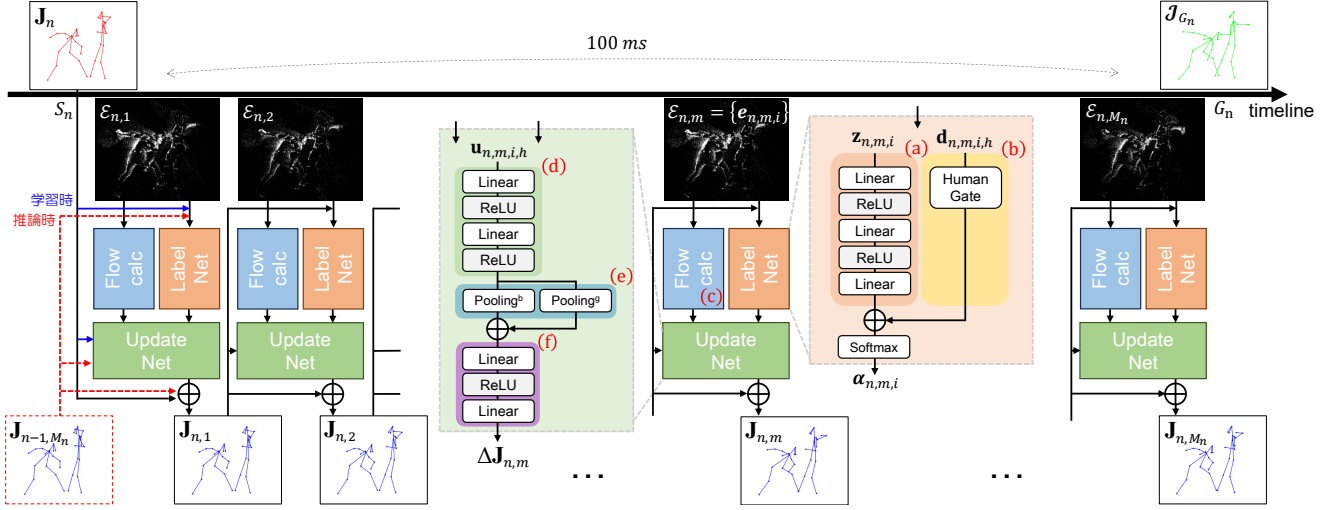


図 2 LabelNet および UpdateNet を使用した姿勢更新による遅延補償の概要.

姿勢 $\mathcal{J}_{S_n}$ を推定対象とするが、これを得られる時刻は G_n であり、このときの現実の姿勢は $\mathcal{J}_{G_n}$ である。

2.2 問題設定

イベントカメラ駆動遅延補償の問題設定は、時刻 G_n において任意の姿勢推定器が推論を完了した姿勢 $\mathbf{J}_n$ と、区間 $[S_n, G_n]$ に取得されるイベント $\mathbf{e}$ を入力として、時刻 G_n における本来の現実の姿勢 $\mathcal{J}_{G_n}$ に近い遅延補償姿勢 $\hat{\mathbf{J}}_n$ を出力することである [8]。

入力として扱うイベント $\mathbf{e}$ は、イベントカメラが輝度値の変化を検知した際に得られ、輝度変化があった位置座標 $\mathbf{x}$ 、正負の極性 p 、タイムスタンプ t にて構成される。また入出力姿勢および教師姿勢として用いる現実の姿勢 $\mathbf{J}, \mathcal{J}_{G_n} \in \mathbb{R}^{2HB}$ は 2 次元関節位置 $\mathbf{j}_k, \mathbf{j}_{G_n,k} \in \mathbb{R}^2$ を複数人に渡りまとめたものである。ここで、人物の通し番号を $h = 1, \dots, H$ 、人物あたりの関節の通し番号を $b = 1, \dots, B$ と定義し、全関節の通し番号は $k = 1, \dots, HB$ となる。ただし、人数が減少し関節数が縮退する場合は、総関節が HB となるようにゼロ埋めが追加されている。

3. 提案手法

3.1 多段階の更新による遅延補償

提案手法は、区間 $[S_n, G_n]$ に取得される多量のイベント $\mathbf{e}$ を固定数 I 個ごとに区切ることで M_n 個のイベント群を作成する。このイベント群をイベントバッファ

$$\mathcal{E}_{n,m} = \{\mathbf{e}_{n,m,i}\}_{i=1}^I, \quad m \in \{1, \dots, M_n\} \quad (1)$$

と表す。ここで、 i はイベントバッファ中の通し番号である。 $\mathbf{J}_n$ に対して、これらイベントバッファを用いて M_n 回の更新を行うことで、徐々に姿勢を $\mathcal{J}_{G_n}$ へ近づける。 $\mathbf{J}_n$ から m 回更新した姿勢を $\mathbf{J}_{n,m}$ とすれば、更新は

$$\mathbf{J}_{n,m} = \mathbf{J}_{n,m-1} + \Delta \mathbf{J}_{n,m} \quad (2)$$

と表現され、姿勢推定器からの直接の出力は $\mathbf{J}_n = \mathbf{J}_{n,0}$ 、遅延補償後の最終出力は $\hat{\mathbf{J}}_n = \mathbf{J}_{n,M_n}$ となる。ここで、 $\Delta \mathbf{J}_{n,m}$ は更新に用いる姿勢差分であり、更新手法の主眼は各イベントバッファ $\mathcal{E}$ から姿勢差分 $\Delta \mathbf{J}$ を導出することである。

以下に任意のイベントバッファ $\mathcal{E}$ および更新前姿勢 $\mathbf{J}$ から姿勢差分 $\Delta \mathbf{J}$ を導出する手順の詳細を示す。

3.1.1 LabelNet による関節寄与度推定

イベント位置の単純参照では、セルフオクルージョンや人物間近接により誤った関節の更新が生じやすい。そこで LabelNet では、各イベントの寄与関節を推定して誤った関節への影響を防ぐ。LabelNet は、イベントバッファ $\mathcal{E}$ および姿勢 $\mathbf{J}$ を入力として、定義された HB 個の全関節に対しての寄与度 $\alpha_i \in \mathbb{R}^{HB}$ をイベントごとに出力する。

関節ロジットの導出 (図 2 中 (a)) LabelNet の入力 $\mathbf{z}_i$ は、イベントのうちタイムスタンプをバッファ区間で min-max 正規化した $\mathbf{e}'_i = (\mathbf{x}_i, p_i, t_i^{\text{norm}})$ と、更新前の姿勢 $\mathbf{J}$ に対する相対位置座標を結合して

$$\mathbf{z}_i = \left[\mathbf{e}'_i, \{\mathbf{x}_i - \mathbf{j}_k\}_{k=1}^{HB} \right] \in \mathbb{R}^{(4+2HB)} \quad (3)$$

と構成する。 $\mathbf{z}_i$ を LabelNet の MLP $g_\theta: \mathbb{R}^{4+2HB} \rightarrow \mathbb{R}^{HB}$ に入力し、全関節に対応する関節ロジット $\mathbf{s}_i$ を得る。

$$\mathbf{s}_i = g_\theta(\mathbf{z}_i) \in \mathbb{R}^{HB} \quad (4)$$

人物ゲーティング (図 2 中 (b)) 複数人シーンにおいて人数減少に明示的に対応し、また人物間での関節対応の混同をより強固に回避するため、関節ロジットへ加算する人物ゲートを導入する。まず、イベント位置と人物 h の中心との距離 $\|\mathbf{d}_{i,h}\|_2$ から、人物ゲート確率

$$g_{i,h} = \frac{\exp(-\|\mathbf{d}_{i,h}\|_2)}{\sum_{h'} \exp(-\|\mathbf{d}_{i,h'}\|_2) + \varepsilon} \quad (5)$$

を求める。ここで、姿勢推定結果が人数 H よりも縮退している場合、該当するゲート強度をゼロ化する。人物ゲー

トを加算した補正後ロジット s'_i に温度 τ 付き softmax を適用して、関節寄与度 α_i を得る。

3.1.2 イベントからの optical flow 計算 (図 2 中 (c))

更新量の直接的な手がかりとして、イベントバッファ $\mathcal{E}$ から optical flow 群

$$\mathcal{F} = \{\mathbf{f}_i \in \mathbb{R}^4\}_{i=1}^I \quad (6)$$

を対応づけ手法により計算する [9]。ここで個別のフローは、始点と速度のベクトルにより $\mathbf{f}_i = (\mathbf{x}_i, \Delta \mathbf{x}_i)^\top$ と表現される。対応づけ手法により得られるフロー数は入力イベント数より少なくなる。これは、一部のイベントを始点として他イベントへ対応づけを行い、対応づけられたイベント群に対し代表的な 1 つのフローが与えられるためである。ここではその代表フローを対応づけられた各イベントへ割り当て直し、各イベントが始点となるフロー群として扱う。

3.1.3 UpdateNet による姿勢更新量の推定

UpdateNet は、前述したイベントフロー $\mathcal{F}$ と LabelNet が出力する関節寄与度 α_i を人物ごとに分解した $\alpha_{i,h}$ を用いて、各人物 h の姿勢差分 $\Delta \mathbf{J}_h \in \mathbb{R}^B$ を推定する。これにより、動きに伴ったイベント出力が十分に得られない撮影環境でも、その環境を加味した更新器を設計する。

重み付きイベントフローの埋め込み (図 2 中 (d)) イベントバッファ $\mathcal{E}$ に対し、人物 h ごとに、入力として関節重み付きイベントフロー $\mathbf{u}_{i,h} \in \mathbb{R}^{(6+B)}$ を

$$\mathbf{u}_{i,h} = [\mathbf{d}_{i,h}, t_i^{\text{norm}}, p_i, \Delta \mathbf{x}_i, \alpha_{i,h}] \quad (7)$$

と構成し、人物間で共有の MLP ϕ で埋め込み $\mathbf{e}_{i,h}$ を得る。

$$\mathbf{e}_{i,h} = \phi(\mathbf{u}_{i,h}) \in \mathbb{R}^{256} \quad (8)$$

関節別およびグローバルな重み付き pooling (図 2 中 (e)) イベントごとの特徴から各関節に対応する局所的な更新特徴を抽出するとともに、人物全体の動きや観測状況を表す大域的な文脈も併せて保持する。関節寄与度を重みとして人物 h の関節 b ごとの代表特徴

$$\mathbf{v}_{h,b} = \frac{\sum_i \alpha_{i,h,b} \mathbf{e}_{i,h}}{\sum_i \alpha_{i,h,b} + \varepsilon} \quad (9)$$

を集約する。また、全関節の総和を用いてグローバル特徴

$$\mathbf{v}_{h,g} = \frac{\sum_i \left(\sum_{b=1}^B \alpha_{i,h,b} \right) \mathbf{e}_{i,h}}{\sum_i \sum_{b=1}^B \alpha_{i,h,b} + \varepsilon} \in \mathbb{R}^{256} \quad (10)$$

を集約し、人物ごとにブロードキャストして用いる。

信頼度スカラー 観測量が少ない状況では入力特徴だけから更新量を安定に決定しづらいため、そのイベント頻度を関節ごとの信頼度 $c_{h,b}$ と人物全体の信頼度 $c_{h,g}$ として回帰ヘッドへ与える。 $c_{h,b}$ は人物 h の関節 b に I 個中どれだけイベントが割り当てられたかを表し、 $c_{h,g}$ は人物 h 全体にどれだけイベントが割り当てられたかを表す。

姿勢差分への回帰 (図 2 中 (f)) 人物 h ごとに、関節別特徴、グローバル特徴、および信頼度を結合した

$$\mathbf{h}_h = \left[\{\mathbf{v}_{h,b}\}_{b=1}^B, \mathbf{v}_{h,g}, \{c_{h,b}\}_{b=1}^B, c_{h,g} \right] \quad (11)$$

を人物共有ヘッドへ入力し、各人の姿勢差分 $\Delta \mathbf{J}_h$ を得る。

3.2 損失関数

最終更新後姿勢 $\mathbf{J}_{n,M_n}$ が教師姿勢 $\mathcal{J}_{G_n}$ に近づくよう、以下の損失で学習を行う。

$$\mathcal{L} = \sum_n \sum_{k=1}^{HB} \|\mathbf{j}_{G_n,k} - \mathbf{j}_{n,M_n,k}\|_1 + \lambda_{\text{ent}} \sum_n \sum_{m=1}^{M_n} \sum_{i=1}^{Q_{n,m}} \mathcal{H}(\alpha_{n,m,i}) \quad (12)$$

ここで $\mathcal{H}(\cdot)$ は LabelNet 出力に対するエントロピー項であり、極端に固定的な割当など不適切な退化の防止に用いる。

3.3 推論時における遅延増大の回避

LabelNet/UpdateNet で用いる相対位置を姿勢推定器が出力する本来の入力姿勢 $\mathbf{J}_n$ から構成しようとする、 $\mathbf{J}_n$ の出力を待ってから実行する必要がある、更新処理が推論パイプラインの末尾に追加され遅延が増大する。そこで図 2 赤線の様に、推論時には相対位置の基準となる姿勢として、前フレームでの遅延補償後姿勢 $\mathbf{J}_{n-1,M_{n-1}}$ を用いることで、姿勢推定に先行シラベリングと差分導出を進められる。

4. 実験

データセット モーションキャプチャによる関節位置計測とイベントカメラを時間・空間の両面で同期させ 30 fps で収録したデータセットを用いて評価した。全 7 シーケンスはいずれも異なるダンス内容であり、総フレーム数 16,888, $H = 2$ 人までの人物増減を含む多様な状況をカバーする。**学習** 学習・検証は、7 シーケンス中 6 シーケンスをそれぞれ 8:2 に分割し、イベントは $I = 2,000$ 個で区切り更新を行った。なお姿勢推定器による遅延は 100 ms とし [8], 3 フレーム先の姿勢の推論に相当する。学習は GeForce RTX 4080 上で実行し、損失関数の重みは $\lambda_{\text{ent}} = 0.5$ を用いた。**評価手順** 評価は学習に含めないテストシーケンス 1 つを用いて実施した。評価指標として、平均関節誤差である MPJPE、閾値に対する正解率 PCK を用いた。PCK については、胴体長で正規化した代表長さを閾値として用いた。

5. 結果

5.1 定性評価

図 3 に定性結果を示す。提案手法が赤で示した入力姿勢を段階的に更新し緑の正解姿勢へ補正する様子が確認できる。また上下で異なるフレームに対しての遅延補償を示しているが、上段は合計で 90 回の更新が行われているのに対し、下段は半分の 45 回となっている。また、図 4 に LabelNet による寄与関節推定結果を示す。2 人物間で腕が

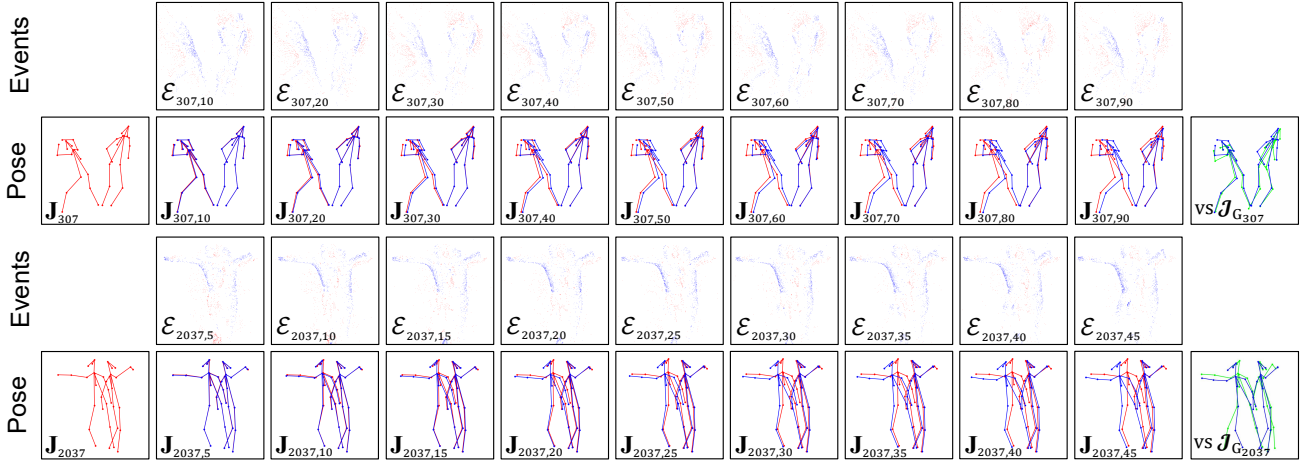


図 3 更新による姿勢推移. 上から, イベント, 入力姿勢 (赤), 更新結果 (青), 正解姿勢 (緑).

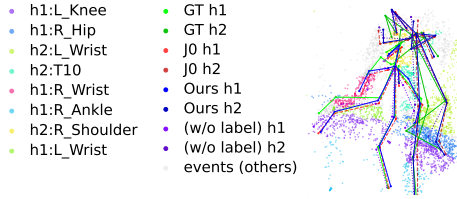


図 4 LabelNet による寄与関節推定結果の例.

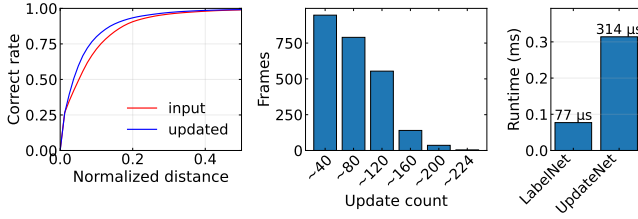


図 5 入力姿勢 (赤) と更新後姿勢 (青) について PCK を走査した正解率曲線 (左), 各フレームで実行された更新の回数 (中央), 提案手法の 1 更新あたり平均処理時間 (右).

胴体に交差するようなシーンであるが, LabelNet により寄与関節が限定されているため人物 1 の左腕などで顕著に誤った関節への更新が抑制されている.

5.2 定量評価

正解率による評価 図 5 左は, PCK を走査した正解率曲線である. 提案手法は入力に対して全域で正解率が高く, 推論遅延による姿勢のズレが更新により緩和されている.

更新回数の評価 図 5 中央は, フレームごとの更新回数の分布である. 更新回数は一定ではなく, 動きの大小に応じてイベントバッファ数が変化し適応的に変化している.

処理時間の評価 図 5 右側に, 1 更新あたりの平均処理時間を示す. 提案手法を構成する 2 つのネットワークによる遅延補償は, 図 5 中央に示した今回の最大更新回数である 224 回を考慮しても想定遅延である 100 ms より大幅に小さく, 姿勢推定完了前に実行可能な構成である.

単純な位置参照手法との評価 表 1 はシーケンス全域での平均誤差の比較である. 単純な更新 [8] による補償精度の低

表 1 シーケンス全域での平均 MPJPE [px] ($\downarrow$) の比較.

input	16.43
Otake et al. (2024)[8]	16.28
Ours w/o UpdateNet	15.52
Ours w/o LabelNet	24.45
Ours	12.97

さから, データセットの撮影環境が極めて弱い光源であることがわかる. 提案手法は誤差を 2/3 程に減少させている.

5.3 Ablation study

提案した 2 段階のネットワークの有効性を検証した結果を表 1 に示す. LabelNet のみでも関節寄与度の推定により誤った関節への更新の抑制により精度が改善する一方で, 関節別・人物別の情報を統合して更新量を安定に回帰する機構が不足し, その改善幅は限定的である. 一方 UpdateNet のみの場合, 寄与関節が曖昧なまま集約されることで, セルフオクルージョンや人物間近接に起因する対応混同が顕在化し, 誤更新が増えて精度が大きく低下する.

6. 結論

本研究では, オンライン人物姿勢推定における遅延を補償するイベントカメラを用いた頑健な姿勢更新手法を提案した. 従来の単純なイベント参照に基づく更新では, 関節の巻き込みや照明条件による観測偏りにより, 安定な遅延補償が困難であった. これに対し本研究では, 各イベントの関節寄与度を推定する LabelNet と, その寄与度に基づいて更新量を回帰する UpdateNet からなる更新を導入し, 巻き込みと観測偏りの双方に頑健な補償を実現した. 実験の結果, 提案手法は従来手法に比べて LabelNet と UpdateNet の併用が有効的に精度を改善することを確認した. また処理時間計測により, 姿勢推定中に差分導出が可能であり遅延の増加無しでの補償が可能であることを示した.

謝辞 本研究の一部は JSPS 科研費 JP25KJ1827, JST BOOST JPMJBS2423 の支援を受けた.

参考文献

- [1] Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T. L., Zhang, F. and Grundmann, M.: BlazePose: On-device Real-time Body Pose tracking, *ArXiv*, Vol. abs/2006.10204, No. -, pp. 1–4 (2020).
- [2] Brandli, C., Berner, R., Yang, M., Liu, S.-C. and Delbruck, T.: A $240 \times 180 \times 130$ dB 3 μ s Latency Global Shutter Spatiotemporal Vision Sensor, *IEEE Journal of Solid-State Circuits*, Vol. 49, No. 10, pp. 2333–2341 (2014).
- [3] Calabrese, E., Taverni, G., Easthope, C. A., Skriabine, S., Corradi, F., Longinotti, L., Eng, K. and Delbruck, T.: DHP19: Dynamic Vision Sensor 3D Human Pose Dataset, *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1695–1704 (2019).
- [4] Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E. and Sheikh, Y.: OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 43, No. 01, pp. 172–186 (2021).
- [5] Fang, H.-S., Li, J., Tang, H., Xu, C., Zhu, H., Xiu, Y., Li, Y.-L. and Lu, C.: AlphaPose: Whole-Body Regional Multi-Person Pose Estimation and Tracking in Real-Time, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 45, No. 6, p. 7157–7173 (2023).
- [6] Gallego, G., Delbrück, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A. J., Conradt, J., Daniilidis, K. and Scaramuzza, D.: Event-Based Vision: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 44, No. 1, pp. 154–180 (2022).
- [7] Lichtsteiner, P., Posch, C. and Delbruck, T.: A $128 \times 128 \times 120$ dB 15 μ s Latency Asynchronous Temporal Contrast Vision Sensor, *IEEE Journal of Solid-State Circuits*, Vol. 43, No. 2, pp. 566–576 (2008).
- [8] Otake, I., Kitano, K., Kushida, T., Kubo, H., Maejima, A., Fujimura, Y., Funatomi, T. and Mukaigawa, Y.: Updating Human Pose Estimation using Event-based Camera to Improve Its Accuracy, *ACM SIGGRAPH 2023 Posters*, SIGGRAPH '23, New York, NY, USA, Association for Computing Machinery (2023).
- [9] Shiba, S., Aoki, Y. and Gallego, G.: Fast Event-Based Optical Flow Estimation by Triplet Matching, *IEEE Signal Processing Letters*, Vol. PP, pp. 1–5 (2023).
- [10] Xu, L., Xu, W., Golyanik, V., Habermann, M., Fang, L. and Theobalt, C.: EventCap: Monocular 3D Capture of High-Speed Human Motions Using an Event Camera, *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4967–4977 (2020).
- [11] Xu, Y., Zhang, J., Zhang, Q. and Tao, D.: ViTPose: Simple Vision Transformer Baselines for Human Pose Estimation (2022).
- [12] Zou, S., Guo, C., Zuo, X., Wang, S., Wang, P., Hu, X., Chen, S., Gong, M. and Cheng, L.: EventHPE: Event-based 3D Human Pose and Shape Estimation (2021).
- [13] Zou, S., Mu, Y., Zuo, X., Wang, S. and Cheng, L.: Event-based Human Pose Tracking by Spiking Spatiotemporal Transformer, *arXiv*, Vol. abs/2303.09681 (2023).