

フローヒストグラムを用いたRecurrent Neural Networkによる 自己移動量推定

阿知破 千浩[†], 福井 宏[†], 山下 隆義[†], 藤吉 弘亘[†]

[†]: 中部大学, {yukihiro@mprg.cs., fhio@mprg.cs., takayoshi@isc., fujiyoshi@isc.}@chubu.ac.jp

概要: 自動運転支援において, 自車の位置や運動を把握することは重要である. Convolutional-Recurrent Neural Network(C-RNN)は, 画像から自己運動の識別が可能である. 一方で, C-RNNは詳細な自己移動量の推定ができないという問題がある. そこで本研究では, C-RNNを自己移動量推定に拡張する. その際, シーンに頑健な推定が可能となる入力方法としてフローヒストグラムを用いる. 評価実験の結果, 従来のRNNより高精度に自己移動量の推定が可能であることを確認した.

1. はじめに

自己移動量推定は, 自動運転支援技術やナビゲーション技術の研究において重要なタスクである. その研究の一環として, 車載映像を用いた自己移動量推定がある. 一般的なナビゲーション等に用いられているGlobal Positioning System(GPS)は, トンネルなどで自己位置を取得できない. また, ジャイロセンサは, 車両の停止中にハンドルのわずかな動きを取得してしまう. そのため, 画像のように全体のコンテキストが分かる情報を利用した自己移動量推定が必要とされている. 画像による自己移動量推定と他のセンサを組み合わせることで, トンネルなどの自己位置をセンサでは取得できない環境での精度向上に繋げることができる.

自動車分野における自己移動量推定の関連研究として, Dynamic Time Warping [1], Longest Common Subsequence [2], Levenshtein Distance on Trajectories [3], パーティクルフィルタを用いた運動推定 [4]などがある. これらの手法は, 車両の軌道をストリングマッチングによって求めている. センサ値のみを計算によって求めているため, 道路状況に応じて臨機応変に対応する事ができない問題がある.

Convolutional-Recurrent Neural Network(C-RNN)を用いて動画画像から自己運動識別を行う手法[5][6]では, 連続する2フレームの画像から算出したオプティカルフローをDeep Convolutional Neural Network(DCNN)により特徴抽出し, Recurrent Neural

Network(RNN)[7]により前進, 後退, 右左折の運動を識別している. オプティカルフローの可視画像を入力としているため, 方向ベクトルを用いて生成した可視画像にノイズが生じる問題がある. また, 運動方向の識別のみ行っており, 車両のトラッキングに必要な自己移動量の推定をできない問題がある.

そこで本研究では, オプティカルフローの方向ベクトルから, フローヒストグラムを算出することでノイズに頑健な特徴を入力する. これにより, 各フレームの移動方向と移動量の分布を学習することが可能となる. また, 画像を分割し, 分割領域ごとにフローヒストグラムを算出する. これにより, 分割領域ごとに移動方向と移動量を計算する事ができ, 高精度な移動量推定が可能となる. 分割方法の検討として, 勾配方向ヒストグラムを用いる Scale Invariant Feature Transform (SIFT)[8]と Gradient Location and Orientation(GLOH)のグリッド分割方法を参考とする.

2. 関連研究

本研究に関連する自己移動量推定手法および, 本研究でベースとするC-RNNについて以降で説明する.

2.1. 自己移動量推定

自己移動量推定は, 自動車の自己位置推定に限らず様々な分野で研究されている. Velocity ObstaclesとMotion Predictionを用いて与えられる軌道から運動モデルを学習し, 物体の挙動を推測する手法 [10]やEcho State Networkを用いたロボットの動きの

推定[11]などがある。自動車を対象とした手法では、Dynamic Time Warping, Longest Common Subsequence, Levenshtein Distance on Trajectories, パーティクルフィルタを用いた運動推定などがある。これらの手法は、車両の軌道をストリングマッチングに基づくマッチング方法によって求めている。しかし、センサ値のみを計算によって求めているため、道路状況に応じて臨機応変に対応する事ができない問題がある。

2.2. 画像を用いた自己移動量推定

画像を用いた自己運動識別の手法の一つにC-RNNを用いた手法がある。C-RNNは、DCNNによる特徴抽出とRNNによる時系列対応により、動画像などの高精度な時系列パターンを学習することができるニューラルネットワークである。RNNにDCNNに畳み込み層とプーリング層を組み合わせることで、動画像から識別に有効な特徴量を抽出し、運動方向の識別性能を向上させている。C-RNNによる運動方向の識別の流れを図1に示す。

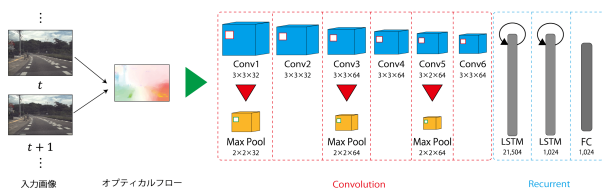


図 1 C-RNN の流れ

C-RNNの構造は、DCNNの畳み込み層とプーリング層、RNNの中間層と出力層を配置した構造となっている。ここで、RNNの中間層にはLong Short Term Memory(LSTM)[12]、C-RNNの入力データにはオプ

ティカルフローの可視画像を用いている。オプティカルフローの可視画像をDCNNにより特徴抽出し、RNNにより前進、後退、右左折の運動を識別する。この手法では、オプティカルフローの可視画像を入力としているため、停止時やシーンに変化が少ない場合、可視画像にノイズが生じることがある。また、運動方向の識別を対象としており、詳細な自己移動量の推定に対応していない。

3. 提案手法

本研究では、フローヒストグラムを用いた RNN による自己移動量推定手法を提案する。図2に提案手法の流れを示す。

3.1. フローヒストグラムの作成

従来のC-RNNの問題点を解決するため、オプティカルフローの方向ベクトルからフローヒストグラムを算出する。これにより、各フレームの移動方向と移動量の分布を学習する事が可能となる。オプティカルフローの方式には、密なオプティカルフローを算出するTV-L1を用いる。また、画像を分割し、分割領域ごとにフローヒストグラムを算出する。これにより、分割領域ごとに移動方向と移動量を計算する事ができ、高精度な移動量推定が可能となる。

画像の分割法は、SIFTとGLOHを参考にする。SIFTとGLOHは、図3のような正方形のグリッドと周辺領域を対数極座標に変換した分割方法を用いている。SIFTを参考にした分割方法は、図4(a)のように縦横7分割の49分割する。また、GLOHを参考にした分割方法は、図4(b)のように48分割する。

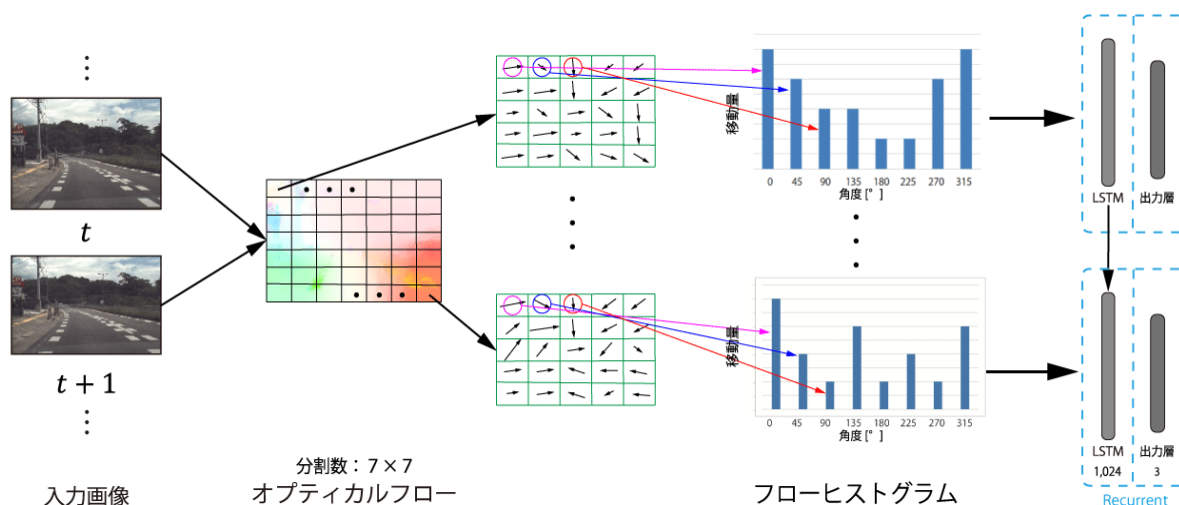


図 2 提案手法の流れ

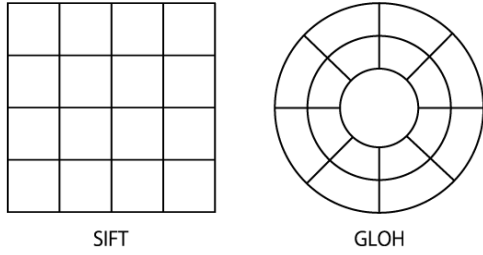
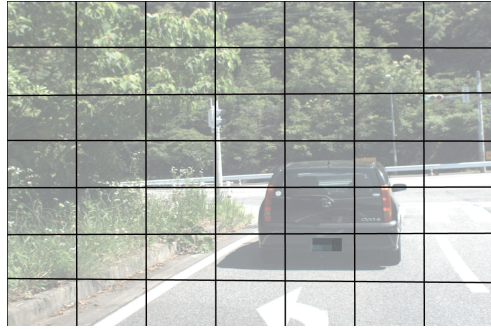
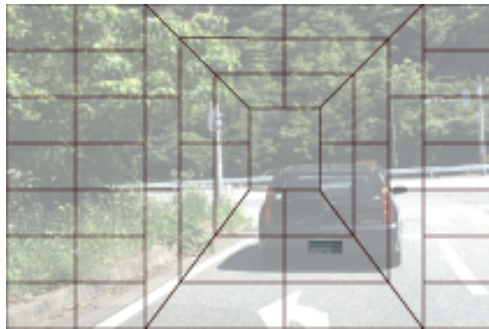


図 3 SIFT と GLOH のグリッド分割



(a)SIFT を参考にした分割方法



(b)GLOH を参考にした分割方法

図 4 分割方法の比較

3.2. RNN による学習および推定

RNNの入力には、前節の方法で算出したフローヒストグラムを用いる。RNNはLSTM1層、出力層1層からネットワークを構成する。入力フローヒストグラムの1次元ベクトル、出力はYaw, Pitch, 移動距離の推定値である。LSTMのユニット数は1,024である。オイラー角のYaw, Pitchは、物体の姿勢表現に使われる角度であり、物体の中心を原点とした x, y, z 軸を中心に回転する角度がRoll, Pitch, Yaw角から算出できる。カメラの設置位置は車室内のバックミラー付近とし、前方を撮影する位置に設置する。

4. 評価実験

提案手法の有効性を調査するため、評価実験により推定精度の比較をする。

4.1. データセット

車載カメラによる一般道の走行動画を使用する。使用した走行動画の例を図5に示す。使用するデータセットは、同じ一般道を2周走行した動画像から構成される。このデータを用いて、既知環境における評価および未知環境における評価の2種類の実験を行う。既知環境では、1周目の14,370フレームを学習データとし、2周目の13,245フレームを評価データとする。未知環境では、1周目の前半6,000フレームと2周目の前半6,000フレームを学習データとし、1周目の後半8,383フレームを評価データとする。



図 5 走行動画の例

4.2. 分割方法の比較

210×140の画像を、SIFTを参考にした格子状に分割する方法とGLOHを参考にした重要な領域を大きく分割する方法の2種類を比較する。分割方法の比較を表1に示す。

表 1 各分割方法の推定誤差の比較

環境	分割方法	Yaw[°]	Pitch[°]	移動距離[m]
既知	SIFT ベース	0.0561	0.0496	0.1257
	GLOH ベース	1.1045	0.8913	0.5101
未知	SIFT ベース	0.0679	0.0468	0.1558
	GLOH ベース	0.2532	0.2717	0.2413

表1から、どの環境においてもSIFTベースはGLOHベースよりも推定誤差が小さい。分割領域を均一にすることが移動量推定には最適だといえる。

4.3. 分割数の評価

前節の実験より、SIFTベースの分割方法の方が良い精度であることがわかった。次に、分割数を変更し、最適な分割数を選出する。210×140の画像を用いるため、1×1、2×2、5×5、7×7の4種類のブロック分割方法を比較する。データセットには、既知環境を用いる。推定誤差の比較を表2に示す。

表 2 分割数による推定誤差の比較

分割数	Yaw[°]	Pitch[°]	移動距離[m]
1×1	0.1353	0.0751	0.1706
2×2	0.0734	0.0604	0.1496
5×5	0.0722	0.0553	0.1248
7×7	0.0561	0.0496	0.1257

表2より、分割数7×7が1×1に比べて、Yawは約0.0792°、Pitchは約0.0255°の誤差が削減され、精度が最も良い。移動距離は、分割数5×5が1×1に比べて約0.0458mの誤差を削減して最も精度が良い。また、分割数7×7と5×5の移動距離の推定誤差の差は約0.0009mと非常に小さいため、Yaw、Pitchの精度が良い7×7が最も有効であるといえる。

4.4. ネットワーク構造の評価

自己移動量推定に有効なネットワークの構造を評価する。使用するネットワークを表3に示す。学習時のミニバッチサイズは3、エポック数は100とする。データセットは既知環境、分割方法は、7×7のSIFTベースを用いる。各ネットワークの推定精度の比較を表4に示す。

表 3 ネットワーク構造

	モデル 1	モデル 2	モデル 3
入力	17,640	17,640	17,640
全結合層 1	-	4,096	4,096
LSTM1	-	-	4,096
LSTM2	1,024	1,024	1,024
出力	3	3	3

表4からYawの推定誤差はモデル1が約0.0561°と最も低い。Pitchにおいては、モデル2がモデル1の推定誤差より約0.0039°小さい。移動距離においては、モデル1の推定誤差が約0.1257mと最も誤差が小さい。しかし、どのモデルも推定誤差に大きな差がない

ため、ネットワークの最小構成であるモデル1が最も有効であるといえる。

表 4 各ネットワークの推定誤差

ネットワーク	Yaw[°]	Pitch[°]	移動距離[m]
モデル1	0.0561	0.0496	0.1257
モデル2	0.0651	0.0457	0.1592
モデル3	0.0664	0.0468	0.1511

4.5. 従来手法との比較

前節の実験結果より、210×140の画像を7×7のブロックに分割して、フローヒストグラムを作成する。RNNに入力するフローヒストグラムは、17,640次元である。従来手法として、畳み込み層6層、LSTM2層、全結合1層の計8層のC-RNNを用いる。動画撮影時にセンサから取得された値を教師信号とし、教師信号と推定値の平均絶対誤差をもとに学習する。

C-RNNと提案手法の推定誤差の比較を表5、分散の比較を表6に示す。表5より、提案手法はC-RNNと比較して、既知環境において、Yawは約0.0029°、移動距離は、約0.0096mの誤差を削減することができる。また、表6より、未知環境において、Yawは約0.4022°、Pitchは約0.0022°の誤差を削減できている。

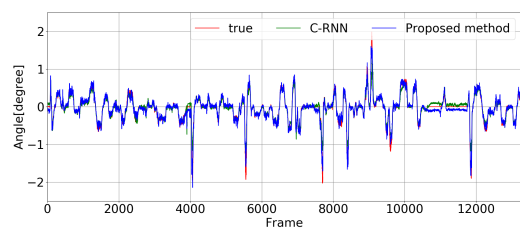
表 5 自己移動量推定の誤差の比較

環境	モデル	Yaw[°]	Pitch[°]	移動距離[m]
既知	C-RNN	0.0590	0.0450	0.1353
	提案手法	0.0561	0.0496	0.1257
未知	C-RNN	0.4701	0.0490	0.1477
	提案手法	0.0679	0.0468	0.1558

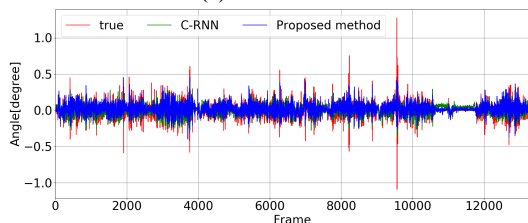
表 6 自己移動量推定の分散の比較

環境	モデル	Yaw[°]	Pitch[°]	移動距離[m]
既知	C-RNN	0.0794	0.0049	0.0911
	提案手法	0.1126	0.0045	0.1025
未知	C-RNN	0.0888	0.0041	0.0574
	提案手法	0.1037	0.0051	0.0801

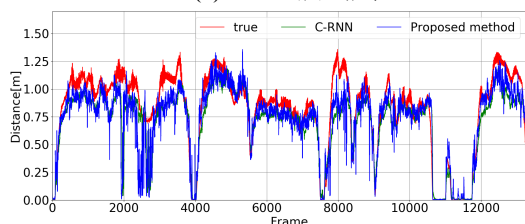
既知環境における各フレームの推定結果を図6、未知環境における各フレームの推定結果を図7に示す。グラフの赤線が正解値、緑線がC-RNNによる推定結果、青線が提案手法による推定結果である。図6(a)から、提案手法は既知環境において、正解に近



(a)Yaw の推定結果

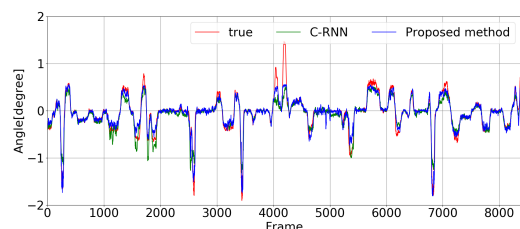


(b)Pitch の推定結果

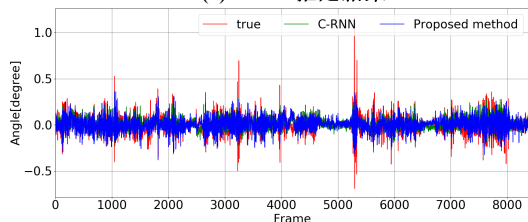


(c)移動距離の推定結果

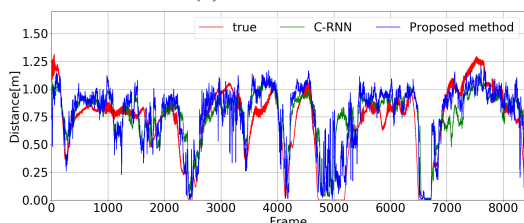
図 6 既知環境の推定結果



(a)Yaw の推定結果



(b)Pitch の推定結果



(c)移動距離の推定結果

図 7 未知環境の推定結果

いYaw値を推定できている。特に、11,000フレームあたりでは、提案手法がC-RNNよりも正解に近い値を推定している。図6(b)から、Pitchも同様に正解に近い

値を推定できている。一方、図6(c)のように、移動距離は正解に近い値が推定できているものの、移動距離が大きい場合に誤差が大きい。図7(a)から、提案手法は4,000フレームあたりの車が右折するシーンにおいて、C-RNNに比べて正解に近い推定ができている。また、図7(b)から、Pitchは既知環境同様に、正解に近い値が推定できている。一方、図7(c)のように、移動距離は正解との差が大きく、推定値が大きく振動しており、精度改善には至らなかった。

推定結果の成功例を図8、失敗例を図9に示す。画像左側の矢印がYaw角を表し、推定値が正の値の時は右方向、負の時は左方向を示す。画像中央の矢印がPitch角を表し、正の値の時は上方向、負の値の時は下方向を示す。画像右側の四角形は、移動距離を示している。また、正解値が赤、C-RNNの推測値が緑、提案手法の推測値が青色である。

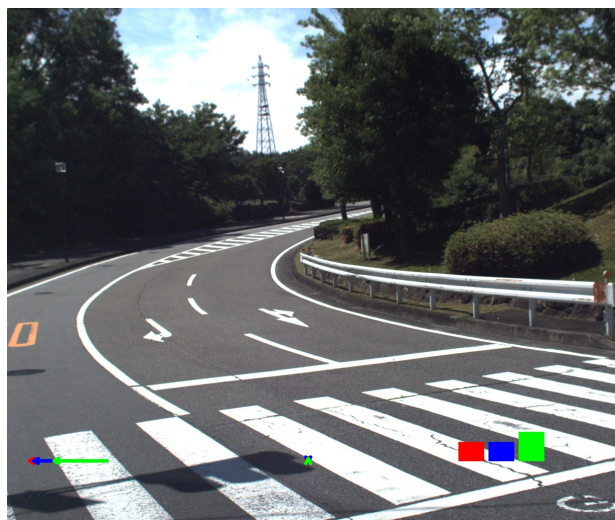


図 8 成功例



図 9 失敗例

図8のように交差点など大きく曲がるようなシーンでは、画像左のYawの矢印を見ると、C-RNNの推定結果に比べて提案手法が正解値に近い値を出力できているのがわかる。これは、提案手法によって急な動きによるノイズを抑制できたためだと考える。しかし、図9のような停止しているシーンでは、車両が前を横切ると移動距離が誤った推定結果となる。これは分割したフローヒストグラムによって、周囲のフローの変化により、自車が動いていると誤って推定したことが原因と考えられる。本手法はフローヒストグラムのみを入力しているため、自車が停止中または走行中かがわかる情報が必要と考える。

評価の結果より、既知環境ではC-RNNと同精度、未知環境ではYawの推定精度を大きく向上させることができた。特に大きく右左折するフレームにおいて、提案手法はC-RNNに比べて推定精度が向上していたことから、急な動きなどに頑健であると言える。

5. おわりに

本稿では、フローヒストグラムを用いた自己移動量推定手法を提案した。既知環境ではYawで約 0.0029° 、移動距離で約0.0096m、未知環境ではYawで約 0.4022° 、Pitchで約0.0022mの推定誤差を削減した。今後の課題として、移動距離の推定精度の更なる向上が挙げられる。

参考文献

- [1] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series", in Proc. of KDD, Seattle, Washington, Jul. 1994, pp. 359–370.
- [2] M. Vlachos, G. Kollios, and D. Gunopulos, "Elastic translation invariant matching of trajectories", M. Learn., vol. 58, no. 2-3, pp. 301–334, '05.
- [3] M. Hahn, L. Krüger, and C. Wöhler, "3d action recognition and longterm prediction of human motion.", in ICVS, ser. Lecture Notes in Computer Science, A. Gasteratos, M. Vincze, and J. K. Tsotsos, Eds., vol. 5008. Springer, 2008, pp. 23–32.
- [4] C. Hermes and C. Wohler, "Long-term vehicle motion prediction", Intelligent Vehicles Symposium, 2009 IEEE, pp. 652–657, 2009.
- [5] Z.Zuo, B.Shuai, G.Wang, X.Liu, X.Wang, B.Wang, Y.Chen, "Convolutional Recurrent Neural Networks: Learning Spatial Dependencies for Image Representation", In CVPR, 2015.
- [6] 神谷龍司, 川口俊樹, 福井宏, 石井育規, 小塚和紀, 羽川令子, 築澤宗太郎, 山下隆義, 山内悠嗣, 藤吉弘

亘, "Convolutional-Recurrent Neural Network による自己運動識別", 画像センシングシンポジウム, 2016.

- [7] J. Elman, "Finding Structure in Time", Cognitive Science 14 (2), 179–211, 1990.
- [8] D. Lowe, "Distinctive image features from scale-invariant keypoints", Int'l. J. Computer Vision, 60, 2, 99, 91–110, 2004.
- [9] Krystian Mikolajczyk and Cordelia Schmid, "GLOH "A performance evaluation of local descriptor", IEEE tran. On Pattern Analysis and Machine Intelligence, 1615–1630, 2005.
- [10] F. Large, D. A. Vasquez Govea, T. Fraichard, and C. Laugier, "Avoiding cars and pedestrians using velocity obstacles and motion prediction", in Proc. of the IEEE Intelligent Vehicle Symp., June 2004.
- [11] S. Hellbach, S. Strauss, J. Eggert, E. K. erner, and H.-M. Gross, "Echo state networks for online prediction of movement data - comparing investigations. " in ICANN (1), ser. Lecture Notes in Comput. Sci., vol. 5163. Springer, 2008, pp. 710–719.
- [12] S.Hochreiter, "LONG SHORT-TERM MEMOR", Neural Computation 9(8), 1735–1780, 1997.

阿知破千浩:現在中部大学大学院修士課程在学中, 画像を用いた自己移動量推定の研究に従事.

福井宏:2016年中部大学大学院博士前期課程修了, 現在同大学大学院博士後期課程在学中, 画像を用いた物体認識の研究に従事.

山下隆義:2002年奈良先端科学技術大学大学院大学博士前期課程修了. 2002年オムロン株式会社入社, 2009年中部大学大学院博士後期課程修了(社会人ドクター), 2014年中部大学講師, 2017年より同大学准教授. 人の理解に向けた動画像処理, パターン認識・機械学習の研究に従事.

藤吉弘亘:1997年中部大学大学院博士後期課程修了. 1997～2000年米カーネギーメロン大学ロボット工学研究所 Postdoctoral Fellow. 2000年中部大学講師. 2004年より同大学教授. 2005～2006年米カーネギーメロン大学ロボット工学研究所客員研究員, 計算機視覚, 動画像処理, パターン認識・理解の研究に従事.