

# 線形方程式の適応的選択による オプティカルフロー推定の高速化

五十嵐 博昭<sup>1,a)</sup> 中野 学<sup>1</sup>

**概要:** オプティカルフロー推定の高速化手法を提案する。オプティカルフロー推定手法の代表である勾配法では線形方程式を立ててオプティカルフローを導出する。勾配法には線形方程式を解く計算量が多い課題がある。本論文ではフロー推定においてフローの動径成分と偏角成分の収束速度が異なることに着目し、動径成分のみを導出する計算量の少ない線形方程式を新たに導入する。従来の直交座標系の線形方程式と動径成分を導出する線形方程式をフローの収束状況に応じて適応的に切り替えることで、オプティカルフローの精度を維持しつつ線形方程式の計算量を削減する手法を提案する。提案手法を HornShunck 法に適用することで 3.00 倍の高速化を確認した。

## 1. はじめに

オプティカルフローは動画像における動きベクトル場を表し、物体検出等で幅広く利用されている。オプティカルフロー推定の代表的な手法が勾配法 [1] である。勾配法は各画素値の時間及び空間勾配を利用してフローを推定する。勾配法では勾配に関する線形方程式を解くことでフローを計算するが、線形方程式を解く演算量が大きい課題がある。

本稿では、フローの収束状況に応じて解く線形方程式を切り替えることでオプティカルフロー推定を高速化する手法を提案する。提案手法ではフローを極座標表現し、動径成分のみを導出する線形方程式を利用する。動径成分を導出する線形方程式と直交座標で導出する線形方程式とをフローの収束状況に応じて適応的に切り替える。動径成分を導出する線形方程式は従来の直交座標系の線形方程式と比較して解く計算量が半分になるため、収束にかかる演算量を削減できる。本稿では、提案手法を HornShunck 法 [1] に適用しその効果を報告する。

## 2. 従来手法

### 2.1 勾配法

勾配法は古典的なオプティカルフロー推定手法である HornShunck 法 [1] で導入された手法である。勾配法では、画素の時空間勾配を用いた線形方程式からオプティカルフローを導出する。線形方程式はテイラー展開によって線形化され、最終的に線形方程式を解くことでフローが求まる。具体的には、フロー  $h = (u, v)$  を用いたエネルギー関

数を用い、エネルギー最小となるフロー  $h$  を求める。以下に HornShunck 法におけるエネルギー関数を示す。

$$E(u, v) = E_{data}(u, v) + \alpha E_{smooth}(u, v) \quad (1)$$

多くのオプティカルフロー推定手法は式 1 に示したエネルギー関数を改善することでフローの精度を向上している。

よりロバストなオプティカルフロー推定手法として Brox 法がある [2]。Brox 法ではエネルギー関数にロバスト関数を導入すことに加え、画像を Warping しつつフローの差分更新をすることで高精度なフロー推定が可能である。さらに高精度なオプティカルフロー推定手法として DeepFlow [3] がある。DeepFlow では前処理として DeepMatching と呼ばれる手法を用い、フレーム間でおおまかなマッチングを取る。オプティカルフロー推定ではデータ項とスムース項に加え、DeepMatching の結果をマッチング項として加味する。これにより非剛体のフローやより大きなフローを正確に推定できる。

多くの従来手法 [1], [2], [3] は勾配法を用いるため処理の流れは同じである。勾配法によるオプティカルフロー推定の流れを図 1a に示す。図 1a において前処理は画像処理等の線形方程式とは直接関係ない処理を表す。直交座標線形方程式作成では線形方程式を導出する。  $2N \times 2N$  線形方程式ソルバでは PCG (Preconditioned Conjugate Gradient) 法や SOR (Successive Over-Relaxation) 法で線形方程式を解く。ここで画素数を  $N$  とすると、解くべき線形方程式の大きさは  $N(\text{画素数}) \times 2(u, v \text{ の } 2 \text{ 成分}) = 2N$  となる。従って線形方程式を解くのに必要な計算量は  $O(2N)$  となる。勾配法では線形方程式を解く演算量が大きい課題があり、多くの従来手法共通の課題である。

<sup>1</sup> 日本電気株式会社  
NEC Corporation

<sup>a)</sup> h-igarashi@hf.jp.nec.com

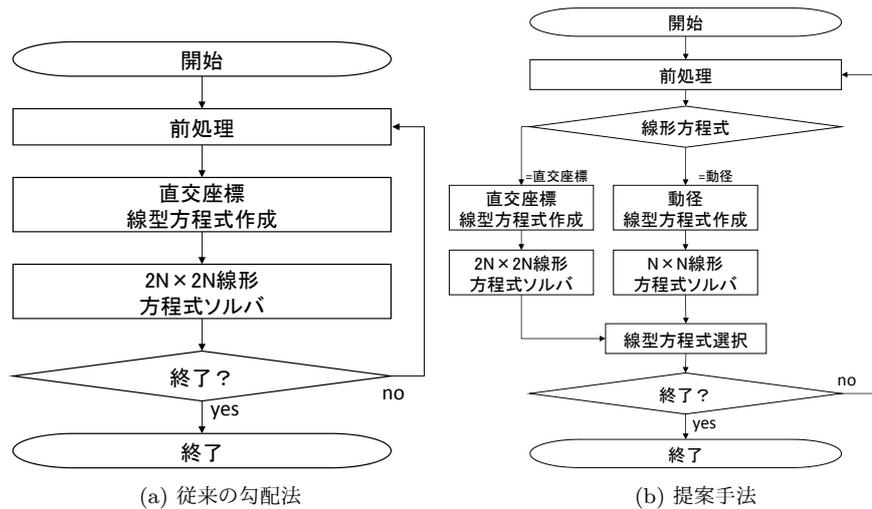


図 1: 従来の勾配法と提案手法の処理の流れ

## 2.2 極座標空間における勾配法

より複雑なオプティカルフローを推定できる手法として極座標空間においてオプティカルフローを推定する手法 [4] がある。[4] では従来の直交座標空間におけるオプティカルフロー  $(u, v)$  を  $(\cos \theta, \sin \theta, r)$  の極座標空間における 3 変数で表現する。尚、 $\theta$  は偏角を、 $r$  は動径を表す。極座標空間では以下の式が成り立つ。

$$\cos^2 \theta + \sin^2 \theta = 1 \quad (2)$$

式 2 が成り立つことを利用すると、極座標空間におけるエネルギー関数は以下となる。

$$\begin{aligned} E(\cos \theta, \sin \theta, r) &= E_{data}(\cos \theta, \sin \theta, r) + \alpha_{\theta} E_{smooth}(\cos \theta, \sin \theta) \\ &+ \alpha_r E_{smooth}(r) + \beta E_{pythagoras}(\cos \theta, \sin \theta) \end{aligned} \quad (3)$$

$E_{pythagoras}$  は式 2 が成り立つことを示す拘束項である。

直交座標空間の勾配法と同様に式 3 は線形方程式に変形できる。[4] の実験結果より、流体運動から生じる鏡面上のオプティカルフロー推定などの複雑なフロー推定で、直交座標空間における勾配法よりも高い精度を実現できる。一方で、未知数が  $2N$  から  $3N$  に増えるため、線形方程式を解く際にかかる計算量は直交座標でのオプティカルフロー推定と比較して増加する。

## 3. 提案手法

### 3.1 従来の勾配法の課題

勾配法では反復法が用いられるため、反復回数を重ねるに連れてフローの精度は徐々に収束する。図 2 に MATLAB 上で HornSchunck 法 [5] を用いて Middlebury Dataset [6] における Dimetrodon のオプティカルフローを推定したときの反復回数と相対誤差の関係を示す。フローの誤差として

出力フローと正解フローの距離を示す Endpoint Error の平均 (AEPE) と、出力フローと正解フローのなす角を示す Angular Error の平均 (AAE) を用い、初期反復解の精度を 1 として正規化した。また、フロー推定において 5 階層のガウシアンピラミッドによって解像度を階層的に変化させ、各階層で 20 反復した。図 2 より、各階層で Angular Error は Endpoint Error よりも早い段階で収束していることがわかる。従って、オプティカルフロー推定においてフローの動径成分と偏角成分は収束速度に関して異なる傾向を示すと推測される。一方で従来手法はフローの動径成分と偏角成分を分けて導出することができないため、収束速度の違いを考慮できていない。

### 3.2 線形方程式の適応的選択

オプティカルフロー推定においてフローの偏角成分は動径成分よりも早く収束するため、提案手法では偏角成分が十分に収束したら動径成分に着目しフローを推定する。具体的にはフローのうち動径成分のみを導出する線形方程式と従来手法と同様に直交座標で導出する線形方程式を用い、二つの線形方程式を切り替えつつフローを推定する。表 1

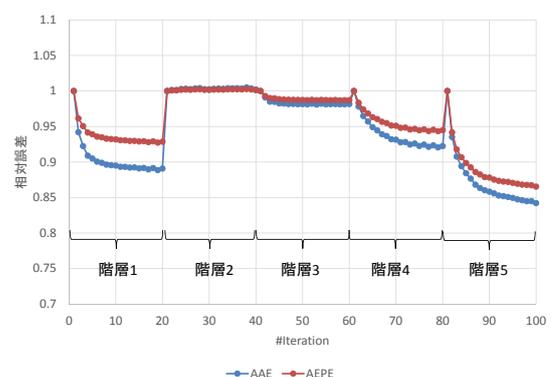


図 2: HornSchunck 法における反復回数と相対誤差

表 1: 各手法のエネルギー関数と計算量の関係

手法	エネルギー関数	計算量
HS 法 [1]	$E(u, v) = E_{data}(u, v) + \alpha E_{smooth}(u, v)$	$\mathcal{O}(2N)$
極座標 [4]	$E(\cos \theta, \sin \theta, r) = E_{data}(\cos \theta, \sin \theta, r) + \alpha_r E_{smooth}(r) + \alpha_\theta E_{smooth}(\cos \theta, \sin \theta) + \beta E_{pythagoras}(\cos \theta, \sin \theta)$	$\mathcal{O}(3N)$
提案手法	$E(u, v) = E_{data}(u, v) + \alpha E_{smooth}(u, v)$ $E_r(r) = E_{data}(r) + \alpha_r E_{smooth}(r)$	$\mathcal{O}(2N)$ or $\mathcal{O}(N)$

に各手法のエネルギー関数と計算量の関係を示す。ここで画素数を  $N$  とすると、動径  $r$  または偏角  $\theta$  成分の一方を解く線形方程式の大きさは  $N \times N$  となり、必要な計算量は  $\mathcal{O}(N)$  となる。

図 1b に提案手法のオプティカルフロー推定の流れを示す。図 1b より、提案手法では直交座標、動径の二種類の線形方程式を用い、次反復で使用する線形方程式を毎反復選択する。線形方程式は出力フローの偏角成分の収束状況に応じて適応的に切り替える。提案手法ではフローの偏角成分が十分に収束した場合に、計算量の小さい動径成分のみを求める線形方程式に切り替えることでフロー全体の収束にかかる演算量を削減できる。

Algorithm1 に線形方程式選択の擬似コードを示す。  $TH_\theta$  は閾値を表す。ここで、提案手法では動径成分を導出する線形方程式において近似を用いるため、  $\delta\theta$  が十分に小さくないと求解できない。したがって、偏角の変位平均  $\Delta\theta_{avg}$  が閾値  $TH_\theta$  よりも大きい場合には従来の直交座標線形方程式を用いる。偏角の変位平均  $\Delta\theta_{avg}$  が十分に小さくなると、使用する線形方程式を動径方程式に切り替えることで動径成分の収束を促す。反復開始時は従来の直交座標方程式を解くため  $\mathcal{O}(2N)$  の計算量がかかる。反復を重ねると動径方程式に切り替わるため計算量が  $\mathcal{O}(N)$  に削減でき高速化できる。

### 3.3 提案手法における線形方程式の導出法

本項では HornShunck 法 [1] をベースに、極座標空間において動径または偏角のいずれかの成分を導出する線形方程式の立て方を説明する。  $h + dh = (u + du, v + dv)$  を極座標表現し、加法定理によって一次近似を用い、二次の項を微小量として無視すると以下となる。

#### Algorithm 1 線形方程式選択

```

Ensure: next_eq = 次反復の線形方程式
 $\Delta\theta_{avg} \leftarrow avg(\Delta\theta)$ 
if  $\Delta\theta_{avg} > TH_\theta$  then
    next_eq  $\leftarrow$  直交座標
else
    next_eq  $\leftarrow$  動径
end if
    
```

$$h + dh = \begin{bmatrix} (r_i + \Delta r_i) \cos(\theta_i + \Delta\theta_i) \\ (r_i + \Delta r_i) \sin(\theta_i + \Delta\theta_i) \end{bmatrix} \sim \underbrace{\begin{bmatrix} u_i \\ v_i \end{bmatrix}}_h + \underbrace{\begin{bmatrix} \cos \theta_i & -v_i \\ \sin \theta_i & u_i \end{bmatrix} \begin{bmatrix} \Delta r_i \\ \Delta \theta_i \end{bmatrix}}_{dh} \quad (4)$$

式 4 より  $dh$  は以下の式となる。

$$dh = \begin{bmatrix} \cos \theta_i & -v_i \\ \sin \theta_i & u_i \end{bmatrix} \begin{bmatrix} \Delta r_i \\ \Delta \theta_i \end{bmatrix} = \underbrace{\begin{bmatrix} \cos \theta_i \\ \sin \theta_i \end{bmatrix}}_\Theta \Delta r_i + \underbrace{\begin{bmatrix} -v_i \\ u_i \end{bmatrix}}_W \Delta \theta_i \quad (5)$$

式 5 を HornShunck 法に適用するとエネルギー関数は以下となる。

$$\begin{aligned} E(\Delta r, \Delta \theta) &= |I_t + I_x \Delta u + I_y \Delta v|^2 \\ &\quad + \alpha_r (|D_x(r + \Delta r)|^2 + |D_y(r + \Delta r)|^2) \\ &\quad + \alpha_\theta (|D_x(\theta + \Delta \theta)|^2 + |D_y(\theta + \Delta \theta)|^2) \\ &= \left| \underbrace{\begin{bmatrix} I_x & I_y \end{bmatrix}}_A \underbrace{\begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix}}_{\Theta \Delta r + W \Delta \theta} + \underbrace{I_t}_b \right|^2 \\ &\quad + \alpha_r \underbrace{\left| \begin{bmatrix} D_x \\ D_y \end{bmatrix} \right|}_{L} (r + \Delta r)^2 + \alpha_\theta \left| \begin{bmatrix} D_x \\ D_y \end{bmatrix} \right| (\theta + \Delta \theta)^2 \\ &= |A(\Theta \Delta r + W \Delta \theta) + b|^2 \\ &\quad + \alpha_r |L(r + \Delta r)|^2 + \alpha_\theta |L(\theta + \Delta \theta)|^2 \quad (6) \end{aligned}$$

動径を導出する場合に関して本稿では IRLS 法 [7] による解法を説明する。動径のみ変化する場合、式 6 を  $\Delta r$  で偏微分した結果が 0 になる。従って動径を導出する方程式は以下の式で表される。

$$\underbrace{(\Theta^T A^T A \Theta + \alpha_r L^T L)}_{A_r} \underbrace{\Delta r}_{x_r} = -\underbrace{(\Theta^T A^T b + \Theta^T A^T A W \Delta \theta + \alpha_r L^T L r)}_{b_r} \quad (7)$$

式 7 より  $A_r x_r = b_r$  を解くことで、フローのうち動径成分を導出できる。尚、式 7 において  $\Delta \theta$  は定数とし、0 または前反復の値を用いる。

## 4. 実装評価

本章では 3 章で説明した提案手法を計算機実験した結果を示す。

提案手法を HornShunck 法 [5] に適用し、MATLAB を用いて実装評価した。尚、提案手法では線形方程式を切り替える閾値として  $TH_\theta = 0.17$  を用いた。また、線形方程式ソルバとして Jacobi-preconditioner 付き PCG 法を用

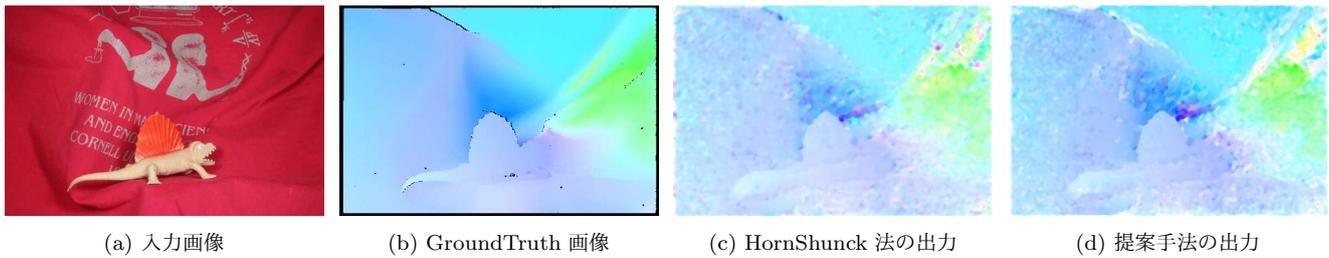


図 3: 入出力画像と推定フロー画像

表 2: HornSchunck 法と提案手法の精度比較

	HornSchunck 法	提案手法
Endpoint Error	0.35	0.36
Angular Error	7.17	6.88

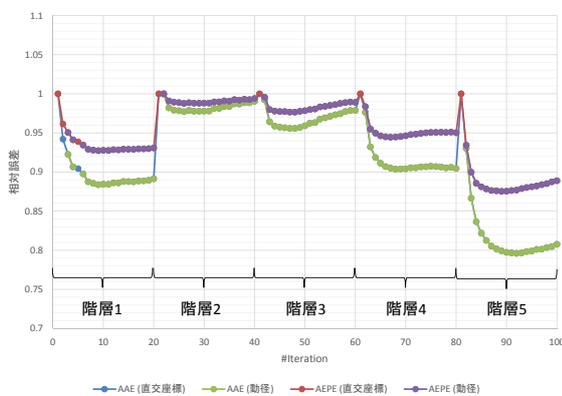


図 4: 提案手法における反復回数と相対誤差

いた。

図 3 に入出力画像と, HornSchunck 法と提案手法による推定フローを可視化した画像を示す。また, 表 2 に HornSchunck 法と提案手法の出力フローの精度の関係を示す。表 2 より提案手法は従来手法と同精度である。

図 4 に提案手法における反復回数と相対誤差の関係を示す。図 4 より提案手法では各階層において 1 反復目に直交座標でフローを推定する線形方程式を用い, 以降の反復処理では動径を求める線形方程式を用いていることがわかる。図 5 に HornSchunck 法と提案手法における実行時間の比較を示す。図 5 より提案手法は従来手法と比較して線型

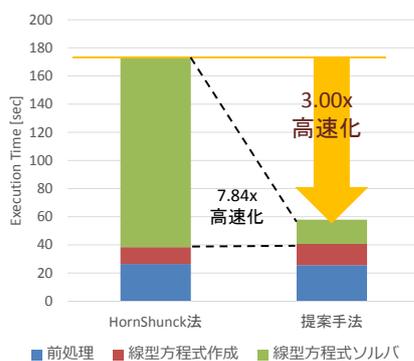


図 5: 実行時間の比較

方程式ソルバの 7.84 倍高速化が実現されており, 全体で 3.00 倍の高速化を達成した。

## 5. おわりに

本稿では, 適応的な線形方程式選択によるオプティカルフロー推定の高速化手法を説明した。提案手法では, オプティカルフローの動径と内角成分のうち, 内角成分が先に収束することに着目し, 内角成分が十分に収束した場合には動径を求める線形方程式を使用することで線形方程式の計算量を削減する。提案手法を HornSchunck 法に適用したところ, 同精度で 3.00 倍の高速化を確認した。

本稿では提案手法を HornSchunck 法に適用した例を説明したが, 本提案手法はその他の手法にも適用可能である。また, 本稿では動径成分のみを導出する線形方程式を導入したが, さらに偏角のみを導出する線形方程式を組み合わせることでより適応的にフローを推定でき, さらなる高速化が期待できる。

## 参考文献

- [1] B. K. P. Horn and B. G. Schunck, "Determining optical flow", *Artificial Intelligence*, vol. 17, pp. 185-203, 1981.
- [2] T. Brox, A. Bruhn, N. Papenberger, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping", in *Proc. European Conference on Computer Vision (ECCV)*, Springer, LNCS, vol. 3024, pp. 25-36, May, 2004.
- [3] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deep flow: large displacement optical flow with deep matching", in *Proc. IEEE International Conference on Computer Vision*, pp.1385-1392, Dec. 2013.
- [4] Y. Adato, T. Zickler, and O. Ben-Shahar, "A polar representation of motion and implications for optical flow", in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.
- [5] F. Zheng, "Optical flow estimation with Horn-Schunck method" (online), 入手先 <http://fzheng.me/2015/03/25/optical-flow/>, Mar. 2015.
- [6] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow", in *Proc. International Conference on Computer Vision*, 2007.
- [7] C. Liu, "Introduction to dense optical flow", in *Dense Image Correspondences for Computer Vision*, Springer, pp. 3-14, 2016.