

令和 2 年度
修士学位論文

時間相関カメラと深層学習を用いた オプティカルフロー推定に関する研究

Study on optical flow estimation using
correlation image sensor and deep learning

1235069 橋本 悠衣里

指導教員 栗原 徹

2021 年 3 月 9 日

高知工科大学大学院 工学研究科 基盤工学専攻
情報学コース

要 旨

時間相関カメラと深層学習を用いた オプティカルフロー推定に関する研究

橋本 悠衣里

オプティカルフローとは、3次元空間での運動を画像上に射影した光学的な動きベクトルである。オプティカルフローの推定は、コンピュータビジョンの分野で難しい課題の一つであり、これまで精力的に研究され続けている分野の一つでもある。また、時間相関カメラは、各画素に入射する光強度信号と参照信号との時間相関値を出力し、運動により生じる輝度変化を複素数として記録することが可能である。

そこで本研究では、時間相関カメラからの出力を入力とする FlowNetS ベースの深層学習ネットワークを提案する。本研究では、FlowNetS ベースの深層学習ネットワークに1枚の相関画像を入力としたパターンと、強度画像と相関画像の2枚を入力としたパターンの大きく2つの深層学習ネットワークを構築し、それぞれのパターンでチャンネル数を増減させた計8パターンのモデルを比較する。そして、新しいデータセットを用いて実験を行い、学習済みである FlowNetS との比較を行う。

生成したデータセットを用いて、大きく2種類の提案ネットワークと学習済みモデルとの比較を行った結果、学習済みモデルと同様のチャンネル数であるモデル以降の全てのモデルで、提案ネットワークの方が高精度であることが確認でき、強度画像と相関画像の有効性を示した。

キーワード オプティカルフロー，時間相関カメラ，深層学習，FlowNetS

Abstract

Study on optical flow estimation using correlation image sensor and deep learning

Yui HASHIMOTO

Optical flow is an optical motion vector that projects motion in three-dimensional space onto an image. Estimating optical flow is one of the difficult tasks in the field of computer vision, and one of the fields that has been energetically studied so far. In addition, the time correlation camera can output the time correlation value between the light intensity signal incident on each pixel and the reference signal, and record the change in brightness caused by motion as a complex number.

Therefore, in this research, we propose a FlowNetS-based deep learning network that takes the output from the time-correlated camera as input. In this study, we constructed two deep learning networks, one with one correlated image input to the FlowNetS-based deep learning network and the other with two input images, an intensity image and a correlated image. We also compare a total of eight patterns of models in which the number of channels is increased or decreased in each network. Then, an experiment is performed using the new data set, and a comparison is made with the trained FlowNetS.

As a result of comparing the two types of proposed networks and the trained model using the generated data set, the proposed network is more accurate in all the models after the model with the same number of channels as the trained model. It was confirmed that the correlation image was effective.

key words Optical Floow, Correlation Image Sensor, Deep Neural Network,
FlowNetS

目次

第 1 章	はじめに	1
1.1	背景と目的	1
1.2	本論文の構成	2
第 2 章	関連技術	3
2.1	オプティカルフロー	3
2.1.1	ブロックマッチング法	3
2.1.2	勾配法	4
2.2	時間相関イメージセンサ	5
2.3	Convolutional Neural Network(CNN)	6
第 3 章	関連研究	8
3.1	FlowNetS/FlowNetC	8
3.2	時間相関カメラを用いた研究	11
第 4 章	提案手法	12
4.1	ネットワークの概要	12
4.2	入力画像	12
4.3	ネットワーク構造	13
4.4	拡張部分	14
4.5	ロス関数	15
4.6	データセット	16
4.7	学習方法	19
第 5 章	実験・考察	20

目次

5.1	実験	20
5.2	実験結果	21
5.3	推定したオプティカルフローの誤差	24
第 6 章	まとめ	28
	謝辞	29
	参考文献	30

目次

2.1	ブロックマッチング法によるオプティカルフロー検出	4
2.2	時間相関イメージセンサ	6
2.3	CNN の構造	7
3.1	FlowNetS/FlowNetC のネットワーク構造 [5]	9
3.2	拡張部分 [5]	9
3.3	比較手法 EpicFlow と FlowNet の出力結果 [5]	11
4.1	提案法のネットワーク構造	13
4.2	拡張部分	15
4.3	データセット生成に用いた画像群	16
4.4	生成した画像例	18
5.1	validation での全エポックの平均 EPE のグラフ	21
5.2	評価により推定されたオプティカルフローと正解ラベル	23
5.3	誤差の大きさをグレイスケール画像として可視化	25
5.4	誤差の方向をオプティカルフロー画像として可視化	26

表目次

4.1	各モデルのチャンネル数	14
5.1	各ネットワークのモデル名	20
5.2	平均 EPE	22
5.3	評価における平均 EPE	22

第 1 章

はじめに

本章では，本研究の背景と目的について述べ，本論文の構成について説明する．

1.1 背景と目的

オプティカルフロー [1] とは，3 次元空間での運動を画像上に射影した光学的な動きベクトルである．これまで，3 次元環境推定やジェスチャー認識・車両追跡などに幅広く応用されており，精力的に研究されている分野の 1 つである．近年では，深層学習を用いたオプティカルフロー推定が盛んに研究されている．

また，時間相関カメラ [2] という特殊なカメラを用いてオプティカルフロー推定を行っている研究も存在する．時間相関カメラは，一般的なカメラとは異なり各画素に入射する光強度信号と参照信号との 1 フレーム時間相関を出力するセンサである．例として時間相関カメラは，運動により生じる輝度変化を複素数として記録することが可能となっている．

そこで我々は，時間相関カメラを深層学習と組み合わせることで従来手法よりも高精度にオプティカルフロー推定を行うことができるのではないかと考え，1 枚の相関画像を深層学習ネットワークの入力としたオプティカルフロー推定を提案してきた．

本研究ではさらなる推定精度の向上を目的として，深層学習ネットワークに 1 枚の相関画像を入力としたパターンと強度画像と相関画像の 2 枚を入力としたパターンの大きく 2 つのネットワークを構築する．そして，新しいデータセットを用いて実験を行い，学習済みモデルとの比較を行う．

1.2 本論文の構成

本節では，本論文の構成について述べる．

第 2 章では，関連技術としてオプティカルフロー，時間相関イメージセンサ，CNN について記述し，第 3 章では関連研究について述べる．第 4 章では，提案手法としてオプティカルフローを推定するために構築したネットワークやデータセットの生成方法，学習方法について述べる．第 5 章では，提案手法を用いて実験を行い，比較手法も含めた結果を示した後考察を述べる．第 6 章では，結論として本論文をまとめを述べる．

第 2 章

関連技術

本章では，本研究に関する諸技術について述べる．

2.1 オプティカルフロー

3次元空間上での運動を画像平面上に射影した光学的な動きベクトルをオプティカルフローという．オプティカルフローは，3次元環境推定やジェスチャー認識、物体追跡など様々な分野で応用されている．オプティカルフロー検出を行う代表的な手法として，ブロックマッチング法と勾配法があげられる．

2.1.1 ブロックマッチング法

ブロックマッチング法は，テンプレートマッチングによる類似領域探索を行いオプティカルフローを検出する手法である．

図 1 のような時間的に連続したフレームにおいて，画像 A のある位置 a の画素が，画像 B のどの位置に移動したのかを探索する．そこで， a の位置を中心とした小領域をテンプレートとして用い，画像 A のある位置 a と同じ位置である画像 B の位置 b を中心として，周辺範囲の探索を行う．探索を行った結果，SAD/SSD が最も小さかった位置の中心画素を位置 a の対応点 a' と決定し，画像 A において位置 a から a' へのベクトルがオプティカルフローとなる．この処理を，画像 A のすべての画素に対して行う．この手法は，決定された範囲の探索を行うため移動量の大きい対象でもオプティカルフローを検出することが可能であるが，決定された範囲を全て探索することから処理時間も長い．

2.1 オプティカルフロー

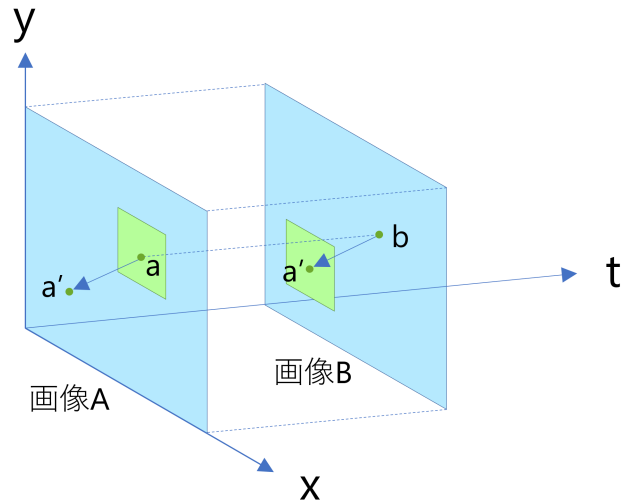


図 2.1: ブロックマッチング法によるオプティカルフロー検出

2.1.2 勾配法

勾配法は，下記 3 つを前提条件として，時空間微分を利用することでオプティカルフローを検出する手法である．

- 連続したフレームにおいて対象物の移動量が微小であること
- 移動前後の明るさが不変であること
- 画素値が滑らかに変化している

時刻 t における画像上の画素 (x, y) を $I(x, y, t)$ として，時間 Δt のその画素の移動量を $(\Delta x, \Delta y)$ と仮定する．このとき，移動前後の明るさが不変であるという前提条件より，式 (2.1) が成り立つ．

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \quad (2.1)$$

式 (2.1) の右辺に対してテイラー展開を行い，高次の項を無視すると式 (2.2) が得られる．

$$I(x, y, t) \doteq I(x, y, t) + \frac{\partial I}{\partial x} \Delta x + \frac{\partial I}{\partial y} \Delta y + \frac{\partial I}{\partial t} \Delta t \quad (2.2)$$

2.2 時間相関イメージセンサ

式 (2.2) を Δt で割り、整理をすると式 (2.3) と表せる.

$$\frac{\partial I}{\partial x} \frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y} \frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t} = 0 \quad (2.3)$$

連続したフレームにおいて対象物の移動量が微小であるという前提条件より, $\Delta t \rightarrow 0$ とすると, 式 (2.4) が成り立つ.

$$\frac{\partial I}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial I}{\partial y} \frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0 \quad (2.4)$$

ここで, オプティカルフローを $(u, v) = (\frac{\partial x}{\partial t}, \frac{\partial y}{\partial t})$ とする. 画像上の画素値の x, y 方向の偏微分をそれぞれ I_x, I_y , 画素値の時間偏微分を I_t とすると, 式 (2.5) が得られる.

$$I_x u + I_y v + I_t = 0 \quad (2.5)$$

この式 (2.5) をオプティカルフローの拘束条件式という. 式 (2.5) において, u, v は未知数でありこれらの値を一意に定めるためにはさらに条件を加える必要がある. そこで, (u, v) を求めるために Lucas-Kanade の方法などの様々な手法が提案されている.

2.2 時間相関イメージセンサ

時間相関イメージセンサは, 各画素に入射する光強度信号と参照信号との 1 フレーム時間相関を出力するセンサである [2]. 時間相関イメージセンサの外観を図 2.2 に示す.

2.3 Convolutional Neural Network(CNN)



図 2.2: 時間相関イメージセンサ

強度画像 $g_x(x, y, t)$ は、通常のカメラと同様で、対応する画素の入射光量の 1 フレーム時間積分として式 (2.6) と表される.

$$g_0(x, y, t) = \int_{-T/2}^{T/2} f(x, y, t) dt \quad (2.6)$$

入射光と参照信号との時間相関値は複素相関画像 $g_\omega(x, y, t)$ といい、参照信号として周波数 ω の複素正弦波を用いると式 (2.7) と表される.

$$g_\omega(x, y, t) = \int_{-T/2}^{T/2} f(x, y, t) e^{-j\omega t} dt \quad (2.7)$$

2.3 Convolutional Neural Network(CNN)

Convolutional Neural Network(CNN) とは、ニューラルネットワークの一種であり、Yann Lecun らによって発明されたものである [3]. 画像認識や画像分類に特化しており、図 2.3 に示すように畳み込み層とプーリング層が何段にも深く積み重なった構造となっている.

2.3 Convolutional Neural Network(CNN)

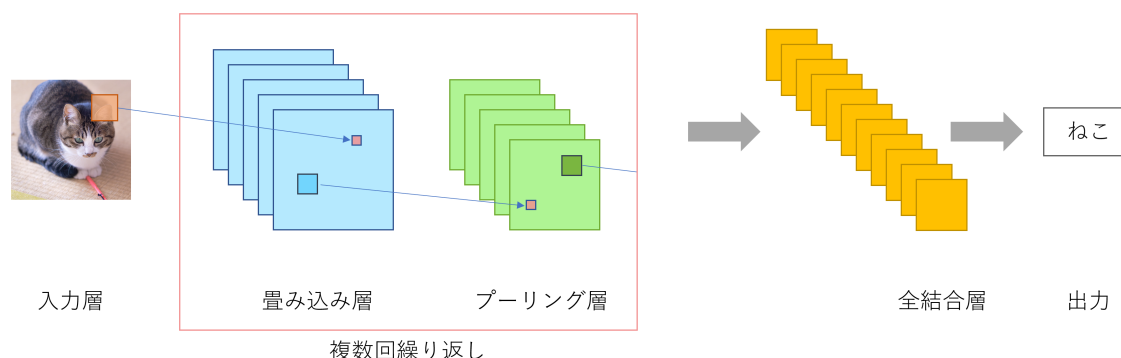


図 2.3: CNN の構造

通常のニューラルネットワークでは、画像を入力する際に 3 次元構造を分解し 1 次元データに変換して処理を行う。そのため、画像の縦方向・横方向・チャンネル数や、隣接した画素同士の空間的な情報などの画像特有の情報が失われてしまう。一方 CNN では、入力された画像に対してカーネルと呼ばれるフィルタを畳み込むことで、画像から直接特徴を抽出する。この畳み込み操作によって得られたデータを特徴マップという。このように CNN では画像の構造を維持したまま処理を行うため、画像特有の情報を失うことなく処理を行うことが可能となっている。畳み込みに用いるフィルタの重みは、学習を行う中で適切な値に更新されていく。

また、一般的に CNN は図 2.3 に示すように畳み込み層、プーリング層、全結合層から構成されるが、全結合層が存在せず全てが畳み込み層のみで構成されている FCN(Fully Convolutional Network) という物体認識に特化したネットワークも存在する [4]。

本研究では、FCN 構造のネットワークを用いている。

第 3 章

関連研究

オプティカルフローは、応用範囲が広く精力的に研究されている分野の一つである。本章では、オプティカルフロー推定における関連研究について述べる。

3.1 FlowNetS/FlowNetC

CNN を用いてオプティカルフローを推定する手法として、Dosovitskiy らが考案した FlowNetS/FlowNetC があげられる [5]。FlowNetS は、運動前後の静止画像を結合したものを入力として、畳み込み処理と逆畳み込み処理で構成されるネットワークであり、FlowNetC は、運動前後の静止画像を別々で入力し、相関層で出力された変位に対して畳み込み処理と逆畳み込み処理を行うネットワークとなっている。

通常、CNN による学習では、畳み込み処理を繰り返し行っていくことで、高レベルの大域的な特徴マップを得ることができるが、低レベルの局所的な細かい情報は失われてしまう。そのため、直接オプティカルフローを推定すると低レベルの局所的な細かい情報が失われてしまっているため、粗い推定結果になってしまうという問題がある。しかし、FlowNet は、逆畳み込み処理を行う際に、各畳み込み層で出力された特徴マップを連結することで、本来ならば失われてしまっている低レベルの局所的な細かい情報を保持したままオプティカルフローを推定することを可能にした。またこの手法は、既存のオプティカルフロー推定手法よりも処理が高速であることが特徴として挙げられる。ネットワークの構造を図 3.1, 3.2 に示す。

3.1 FlowNetS/FlowNetC

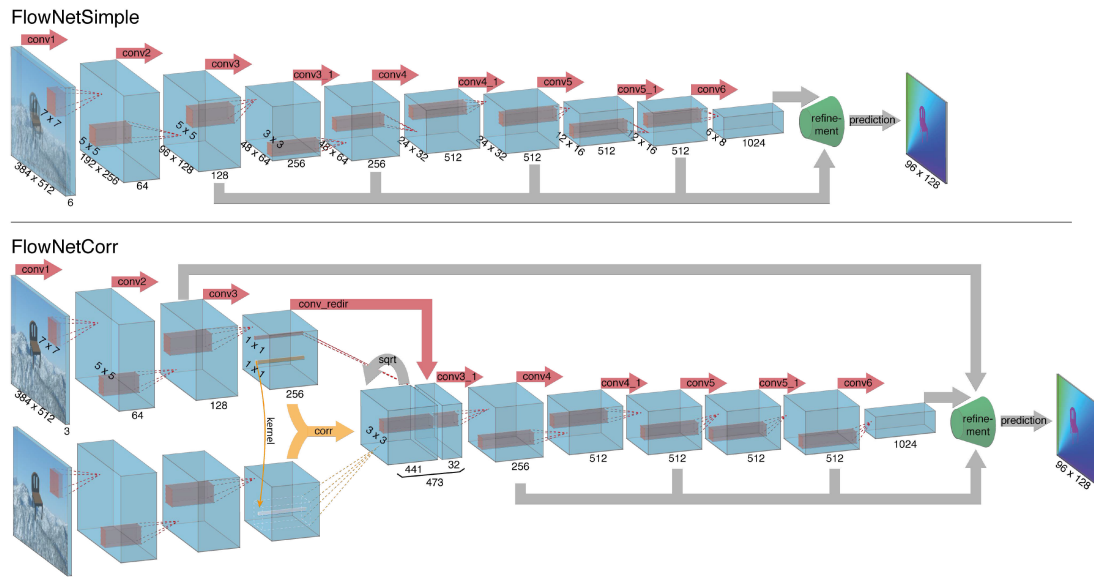


図 3.1: FlowNetS/FlowNetC のネットワーク構造 [5]

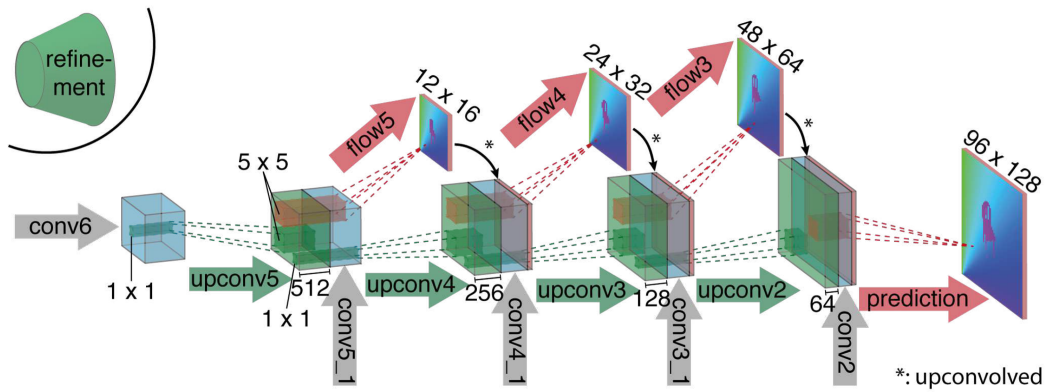


図 3.2: 拡張部分 [5]

FlowNet は、図 3.1 に示すように連続した 2 枚のカラー画像 6 チャンネルを入力としており、畳み込み処理を行う圧縮部分と、逆畳み込み処理を行う拡張部分で構成されている。また、全結合層が存在しないため、任意サイズの画像を入力することが可能である。

圧縮部分では、9 つの畳み込み層のうち 6 つの畳み込み層でストライド 2 の畳み込み処理を行っており、各層の後に ReLU 非線形を使用している。畳み込みフィルタのサイズは、最初の層では 7×7 、2 層目以降は 5×5 、4 層目以降は 3×3 といったようにネットワークが深くなるにつれて小さくなり、得られる特徴マップの数はネットワークがより深くなるに

3.1 FlowNetS/FlowNetC

つれて増加する.

拡張部分では, 図 3.2 に示すように得られた特徴マップに対して, 各畳み込み層で出力された低レベルの局所的な特徴マップと, 各層でアップサンプリングされた予測フローの 3 つを連結した特徴マップに対して逆畳み込み処理を行っている. このようにすることで, 高レベルの大域的な情報と, 低レベルの局所的な情報の両方を保持することが可能となる. 各ステップで解像度が 2 倍になり, 最終的に入力画像の $1/4$ のサイズの予測フローを出力する.

また FlowNet では, 式 (3.2) に示すような各層でアップサンプリングされた予測フロー $\hat{\mathbf{f}}_k$ の誤差と重み ω_k での総和を計算するロス関数を用いている. 各層での予測フロー $\hat{\mathbf{f}}_k$ の誤差には, 式 (3.3) に示す予測フロー $\hat{\mathbf{f}}_k$ と正解フロー \mathbf{f} 間の全てのピクセルの差の 2 乗和である End Point Error(EPE) の総和を用いている. ここで, 式 (3.2)(3.3) 中の \mathbf{f} , $\hat{\mathbf{f}}_k$ は, それぞれ式 (3.1) に示す正解フローと各層でアップサンプリングされた予測フローを表す. このとき, $v_x, v_y, \hat{v}_x, \hat{v}_y$ はフローベクトルの各要素, (i, j) は各要素の座標を表している.

$$\mathbf{f} = \begin{bmatrix} v_x(i, j) & v_y(i, j) \end{bmatrix}^T, \hat{\mathbf{f}} = \begin{bmatrix} \hat{v}_x(i, j) & \hat{v}_y(i, j) \end{bmatrix}^T \quad (3.1)$$

$$L_{EPE} = \sum_{k=1}^{v5} \omega_k * L_{EPE}(\mathbf{f}, \hat{\mathbf{f}}_k) \quad (3.2)$$

$$L_{EPE}(\mathbf{f}, \hat{\mathbf{f}}) = \sqrt{(v_x(i, j) - \hat{v}_x(i, j))^2 + (v_y(i, j) - \hat{v}_y(i, j))^2} \quad (3.3)$$

図 3.3 に FlowNet と比較手法との出力結果を示す. 図 3.3 を見るとわかるように, 比較手法である EpicFlow では推定できていなかった椅子の足や背もたれの部分が, FlowNet では FlowNetS/FlowNetC のどちらの手法でも推定できていることが確認できる.

3.2 時間相関カメラを用いた研究

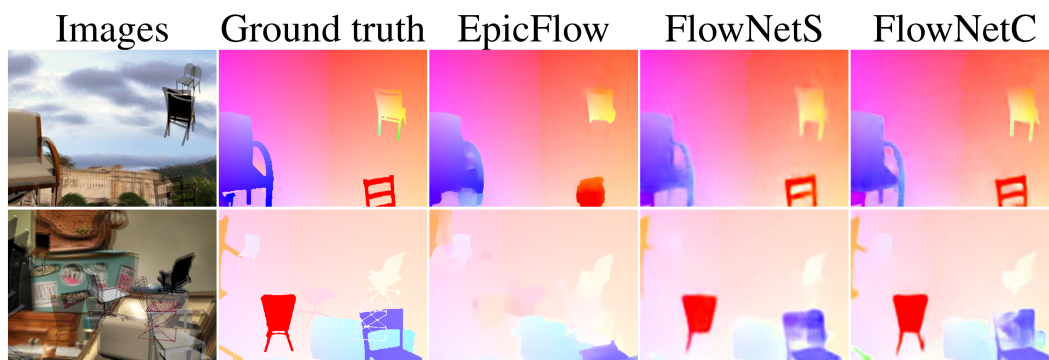


図 3.3: 比較手法 EpicFlow と FlowNet の出力結果 [5]

3.2 時間相関カメラを用いた研究

時間相関カメラを用いてオプティカルフローを推定する研究が複数提案されている。

安藤らは、時間相関カメラと加重積分法 [6] を用いたオプティカルフロー推定手法を提案した [7]。彼らは他にも、2.1.2 で述べたオプティカルフローの拘束条件式に対して、時間方向での 1 フレーム時間積分を行うことで積分形式に定式化し、オプティカルフローを推定する手法を提案している [8]。この手法は、差分近似の精度が悪化する時間軸方向の差分を除いた積分形式のオプティカルフロー恒等式を用いるため、高速移動物体を推定することができるという特徴をもつ。

また、栗原らは GPU を使用した高速オプティカルフロー推定手法を提案した [9]。さらに彼らは、時間相関カメラを用いたオプティカルフロー推定では、模様のない様な面で正しくオプティカルフローを推定することができなかったのに対して、Total Variation を用いて推定する手法を提案した [10]。

第 4 章

提案手法

本章では，時間相関カメラと深層学習を組み合わせてオプティカルフローを推定する手法について述べる．また，提案するネットワークの train/validation に用いるデータセットや，比較手法との評価を行う際に用いるデータセットの生成方法についても述べる．

4.1 ネットワークの概要

本研究では，3.1 で前述した FlowNetS をベースとして，1 枚の相関画像を入力とした深層学習ネットワークと，強度画像と相関画像の 2 枚を入力とした深層学習ネットワークの大きく 2 つのネットワークを構築する．

4.2 入力画像

2.2 で前述したように，時間相関カメラは強度画像と相関画像というそれぞれ異なる情報を持つ 2 種類の画像を出力する．

強度画像は，1 フレーム時間積分値であり，時間軸での低周波成分の情報を保持しているのに対して，相関画像は参照信号との積を積分したものであり，時間軸での参照信号で入力した周波数成分の情報を保持している．そのため，相関画像のみを入力とした場合と，本来相関画像にはない低周波成分の情報を持つ強度画像を入力に加えた場合の 2 種類のネットワークを構築する．

また，本研究では相関画像の実部・虚部をそれぞれ異なるチャンネルとして入力する．そのため，FlowNetS では運動前後のカラー画像 2 枚の 6 チャンネルを入力としていたのに対

4.3 ネットワーク構造

して、相関画像のみを入力としたネットワークでは実部・虚部の 2 チャンネルを、強度画像を加えたネットワークでは、強度画像をグレイスケールとして 3 チャンネルを入力としている。

4.3 ネットワーク構造

本研究で提案するネットワークの構造を図 4.1 に示す。

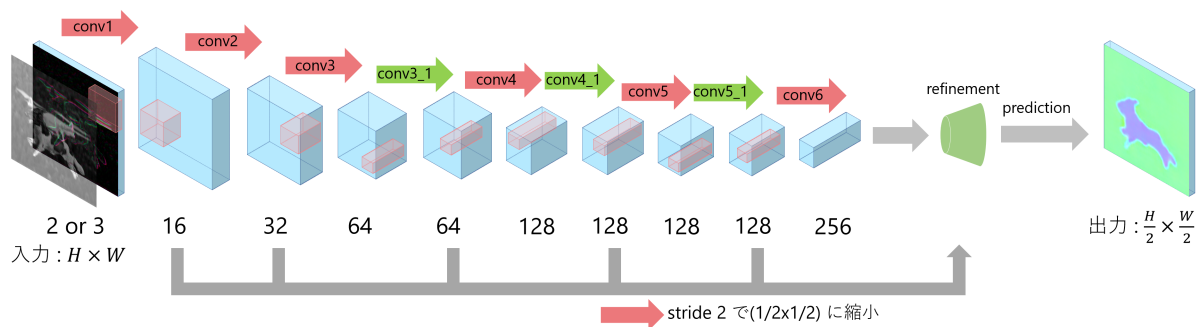


図 4.1: 提案法のネットワーク構造

提案法ネットワークは、FlowNetS をベースとして相関画像と強度画像を入力としている。しかし、FlowNetS と同様のチャンネル数で、相関画像と強度画像が持つ運動情報を捉えきれぬかは定かでない。そのため、本研究では相関画像の 2 チャンネルを入力した場合と、強度画像と相関画像の 3 チャンネルを入力とした場合のネットワークそれぞれでチャンネル数を FlowNetS の $1/4$ 倍、1 倍、2 倍、3 倍と増減させた計 8 つのモデルで実験を行う。FlowNetS と提案法モデルのチャンネル数を表 4.1 に示す。

4.4 拡張部分

表 4.1: 各モデルのチャンネル数

	FlowNetS		提案法 (相関画像 2ch, 強度画像と相関画像 3ch)							
	チャンネル数		1/4 倍		1 倍		2 倍		3 倍	
	in	out	in	out	in	out	in	out	in	out
Conv1	6	64	2	16	2	64	2	128	2	192
Conv2	64	128	16	32	64	128	128	256	192	384
Conv3	128	256	32	64	128	256	256	512	384	768
Conv3_1	256	256	64	64	256	256	512	512	768	768
Conv4	256	512	64	128	256	512	512	1024	768	1536
Conv4.1	512	512	128	128	512	512	1024	1024	1536	1536
Conv5	512	512	128	128	512	512	1024	1024	1536	1536
Conv5.1	512	512	128	128	512	512	1024	1024	1536	1536
Conv6	512	1024	128	256	512	1024	1024	2048	1536	3072
Conv6.1	1024	1024	256	256	1024	1024	2048	2048	3072	3072

4.4 拡張部分

逆畳み込み層により構成される拡張部分を図 4.2 に示す。

FlowNetS では、最終的に入力画像の $1/4$ のサイズの予測フローを出力しており、この大きさから更に解像度を上げて結果は改善しなかったと報告されている。本手法では、予備実験より、入力画像の $1/2$ サイズの予測フローの方が高精度だったことが確認できた。そのため、拡張部に層を 1 層追加し、入力画像の $1/2$ サイズの予測フローを出力するよう変更を行った。

4.5 ロス関数

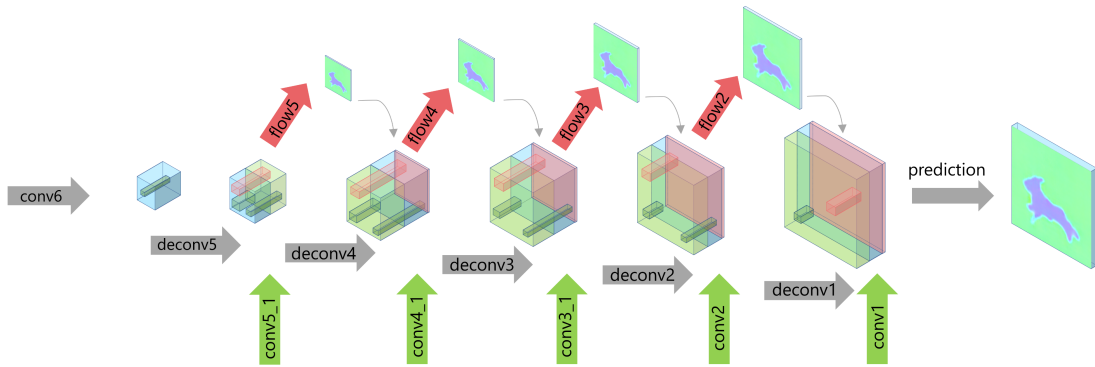


図 4.2: 拡張部分

4.5 ロス関数

3.1 で前述したように，FlowNetS では各層でサイズの異なる予測フローを出力し，それぞれで出力したロスと重み w での総和を計算するロス関数を用いている．本手法でもこのロス関数を用いた．

FlowNetS では，重みとして $w = [0.0005, 0.01, 0.02, 0.08, 0.32]$ を用いていた．本手法では予備実験として，下記に示すような各層ごとに等しい重み w_1 ，指数関数的に増減する低解像度重視の重み w_2 と高解像度重視の重み w_3 ， $r = 2$ の等比級数の和が 1 となるような低解像度重視の重み w_4 と高解像度重視の重み w_5 の 5 つの重みでロスへの影響を調査した．その結果，最も推定精度が高かった w_5 を本手法での重み w と設定した．

$$w_1 = [1/6, 1/6, 1/6, 1/6, 1/6, 1/6]$$

$$w_2 = [1/21, 2/21, 3/21, 4/21, 5/21, 6/21]$$

$$w_3 = [6/21, 5/21, 4/21, 3/21, 2/21, 1/21]$$

$$w_4 = [1/63, 2/63, 4/63, 8/63, 16/63, 32/63]$$

$$w_5 = [32/63, 16/63, 8/63, 4/63, 2/63, 1/63]$$

4.6 データセット

強度画像と相関画像は、図 4.3 のような 30 枚の静止画像から前景と後景をランダムに組み合わせることで生成した。各画像は、運動速度の上限を 32 画素/フレームとして、ランダムな方向にランダムな速度で直線もしくは回転運動をさせた。また、前景の画像はアルファチャンネルのマスク画像を使用し、各マスク画像は AfterEffect を使用して作成した。

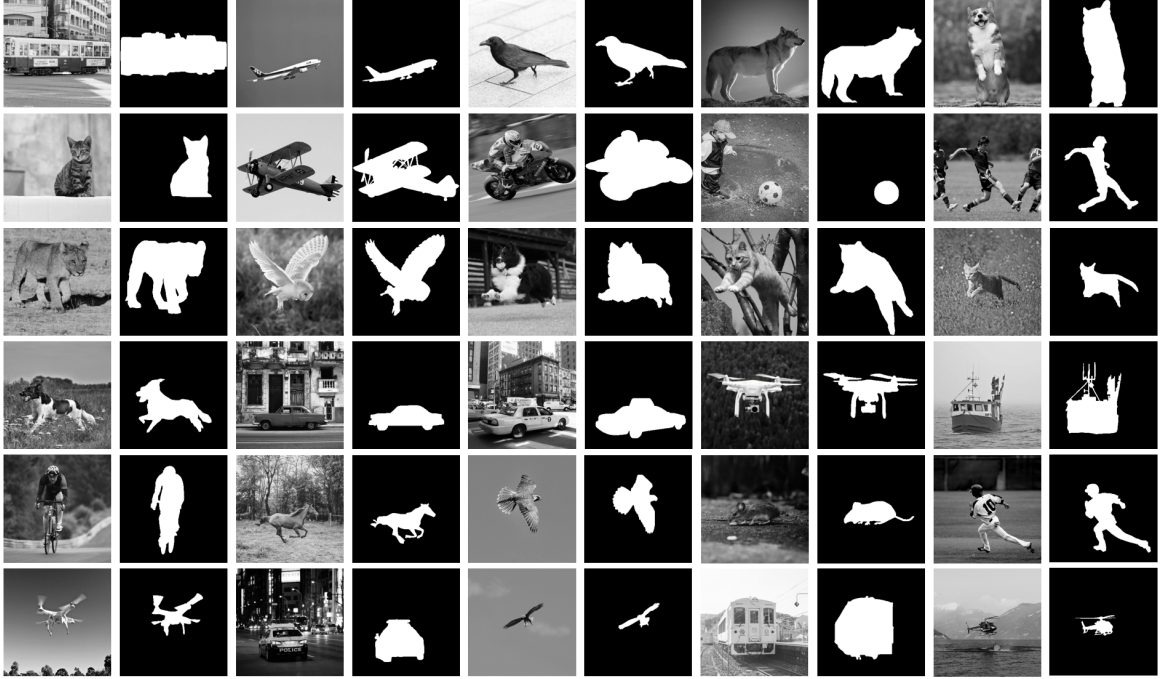


図 4.3: データセット生成に用いた画像群

シミュレーションでは、1 フレーム時間 T をサンプリング時間 Δ の K サンプルに分割した。これはつまり、 $T = \Delta K$ となる。静止画像 $f(x, y, t)$ は、等速運動を仮定し各ステップで (Δ_x, Δ_y) だけ運動させた。 $t = k\Delta$ とすると強度画像 g_0 、相関画像 g_ω は式 (4.1)(4.2) のように計算することができる。

$$g_0(x, y) = \sum_{k=0}^{K-1} f(x, y, t) \quad (4.1)$$

4.6 データセット

$$g_\omega(x, y) = \sum_{k=0}^{K-1} f(x, y, t) \exp(-j\omega_0 t) \quad (4.2)$$

データセットは、 $f_f(x, y, t)$ を前景として、 $f_b(x, y, t)$ を後景として 2 枚の画像を重ね合わせることで生成している。静止画像 $f(x, y, t)$ は、マスク画像 $m_f(x, y, t)$ を使用して式 (4.3) で得られる。このとき、前景画像 $f_f(x, y, t)$ 、後景画像 $f_b(x, y, t)$ 、マスク画像 $m_f(x, y, t)$ は、式 (4.4)-(4.6) のように表すことができる。

$$f(x, y, t) = f_f(x, y, t)m_f(x, y, t) + f_b(x, y, t)(1 - m_f(x, y, t)) \quad (4.3)$$

$$f_f(x, y, t) = f_f(x - \Delta_x^{(f)}, y - \Delta_y^{(f)}) \quad (4.4)$$

$$f_b(x, y, t) = f_b(x - \Delta_x^{(b)}, y - \Delta_y^{(b)}) \quad (4.5)$$

$$m_f(x, y, t) = m_f(x - \Delta_x^{(f)}, y - \Delta_y^{(f)}) \quad (4.6)$$

運動した画像 $f(x - \Delta_x, y - \Delta_y)$ は、直線運動と回転運動でそれぞれ異なる方法で計算を行った。このとき、前景 $f_f(x, y, t)$ と後景 $f_b(x, y, t)$ は、それぞれ異なる運動を行う。

直線運動は、元のパターン $f(x, y)$ のシフトとして考えることができる。そこで $f_f(x, y, t)$ 、 $f_b(x, y, t)$ はフーリエ変換を用いて、式 (4.7)(4.8) のように得られる。ここで、 $F(u, v) = \mathcal{F}[f(x, y)]$ であり、 u, v はそれぞれ空間周波数を表している。

$$f_f(x, y, t) = f_f(x - \Delta_x^{(f)}, y - \Delta_y^{(f)}) = \mathcal{F}^{-1}[\exp(ju\Delta_x^{(f)}) \exp(jv\Delta_y^{(f)}) F_f(u, v)] \quad (4.7)$$

$$f_b(x, y, t) = f_b(x - \Delta_x^{(b)}, y - \Delta_y^{(b)}) = \mathcal{F}^{-1}[\exp(ju\Delta_x^{(b)}) \exp(jv\Delta_y^{(b)}) F_b(u, v)] \quad (4.8)$$

回転運動は、回転の中心座標 (c_x, c_y) と回転角度 θ を前景・後景でそれぞれランダムに決定して回転処理を行う。回転運動の移動量 $(\Delta_x^{(f)}, \Delta_y^{(f)})$ 、 $(\Delta_x^{(b)}, \Delta_y^{(b)})$ は、式 (4.9)(4.10) で表される。このとき、 $(c_x^{(f)}, c_y^{(f)})$ 、 $(c_x^{(b)}, c_y^{(b)})$ 、 $R(\theta(t))^{(f)}$ 、 $R(\theta(t))^{(b)}$ はそれぞれ前景・後景の回転の中心座標、回転角度を表す。

4.6 データセット

$$\begin{pmatrix} \Delta_x^{(f)} \\ \Delta_y^{(f)} \end{pmatrix} = R(\theta(t))^{(f)} \begin{pmatrix} x - c_x^{(f)} \\ y - c_y^{(f)} \end{pmatrix} + \begin{pmatrix} c_x^{(f)} \\ c_y^{(f)} \end{pmatrix} \quad (4.9)$$

$$\begin{pmatrix} \Delta_x^{(b)} \\ \Delta_y^{(b)} \end{pmatrix} = R(\theta(t))^{(b)} \begin{pmatrix} x - c_x^{(b)} \\ y - c_y^{(b)} \end{pmatrix} + \begin{pmatrix} c_x^{(b)} \\ c_y^{(b)} \end{pmatrix} \quad (4.10)$$

このような処理を行うことで、直線・回転運動を含んだ 69,753 枚の画像を生成した。また、train/validation で用いる画像とは別に評価用データセットとして、運動前後のカラー画像と強度画像・相関画像を新たに 7,830 枚生成した。図 4.4 に生成した運動前後のカラー画像と強度画像・相関画像を示す。



(a) 運動前画像



(b) 運動後画像



(c) 強度画像



(d) 相関画像

図 4.4: 生成した画像例

4.7 学習方法

学習の際、最適化には Adam、活性化関数には ReLU を使用し、バッチサイズを 8、学習率を 0.0001、エポック数を 200 と設定した。また、学習率は 100 エポック学習後に 50 エポックごとで $1/2$ している。

さらに、本手法では FlowNetS と同様に過学習を防ぐため、学習時にネットワーク中で 50% の確率で水平/垂直反転、等確率で 0° 、 90° 、 180° 、 270° のいずれかの角度で回転、全てのデータに対して中心座標を $-32 \sim 32\text{px}$ の範囲でランダムに決定し (256×256) サイズで切り取るという計 4 種類のデータ拡張を用いている。

第 5 章

実験・考察

本章では，提案手法の有効性を示すために行った実験について述べ，実験結果を基に考察を行う．

5.1 実験

本手法の有効性を示すためにシミュレーションを行う．シミュレーションでは，4 章で前述したように入力画像とチャンネル数の異なる 8 つのモデルそれぞれで学習を行う．そのため，提案する 8 つのモデルのモデル名を表 5.1 に定義する．また，生成したデータセット 69,753 枚のうち，train に 55,870 枚，varidation に 13,883 枚を使用する．

表 5.1: 各ネットワークのモデル名

入力	1/4 倍	1 倍	2 倍	3 倍
相関画像	Model2ch_1/4	Model2ch_1	Model2ch_2	Model2ch_3
強度画像＋相関画像	Model3ch_1/4	Model3ch_1	Model3ch_2	Model3ch_3

比較手法には，FlyingChairDatasets により学習済みである FlowNetS(pre-trained) を用い [11]，4.6 で前述した評価用データセット 7,830 枚を用いて提案手法との比較を行う．

5.2 実験結果

8つのモデルで学習を行った結果, validation の全エポックの平均 EPE のグラフを図 5.1 に, validation の際に最良の結果が得られたときの train/validation の平均 EPE を表 5.2 に示す. 表 5.2 より平均 EPE は, 提案法の 8 つのモデルの中では, Model2ch_3 で最も良い精度が得られた.

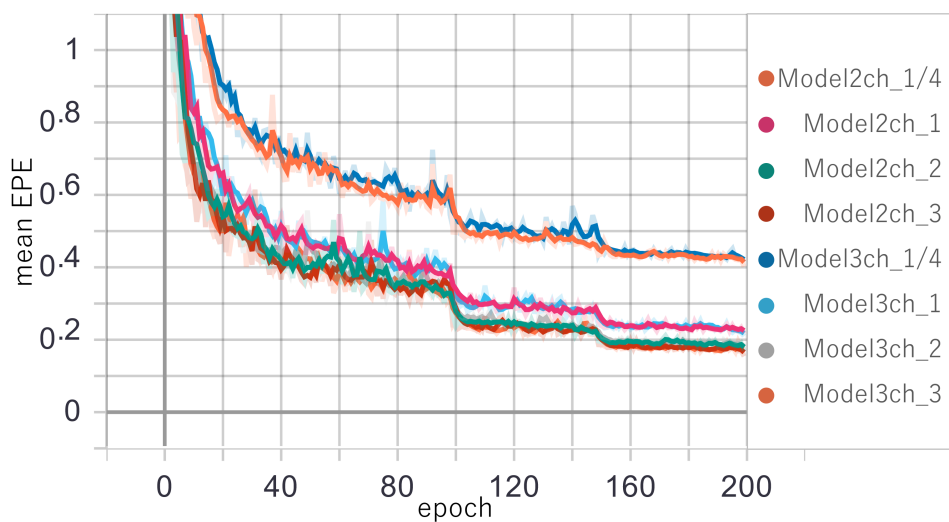


図 5.1: validation での全エポックの平均 EPE のグラフ

5.2 実験結果

表 5.2: 平均 EPE

Model	train(55870 枚)	varidation(13883 枚)
Model2ch_1/4	0.4863	0.4015
Model2ch_1	0.2506	0.2174
Model2ch_2	0.1986	0.169
Model2ch_3	0.1839	0.1542
Model3ch_1/4	0.4946	0.4138
Model3ch_1	0.2519	0.2101
Model3ch_2	0.1984	0.1737
Model3ch_3	0.2036	0.1668

また, varidation の際に最良の結果が得られたときの 8 つのモデルと pre-traind に対して, 評価用データを使用して推定されたオプティカルフローと正解ラベルを図 5.2 に, それぞれのモデルでの平均 EPE を表 5.3 に示す.

表 5.3: 評価における平均 EPE

pre-traind	Model2ch_1/4	Model2ch_1	Model2ch_2	Model2ch_3	Model3ch_1/4	Model3ch_1	Model3ch_2	Model3ch_3
0.4852	0.5042	0.3646	0.3107	0.3133	0.5254	0.3275	0.2816	0.2904

5.2 実験結果

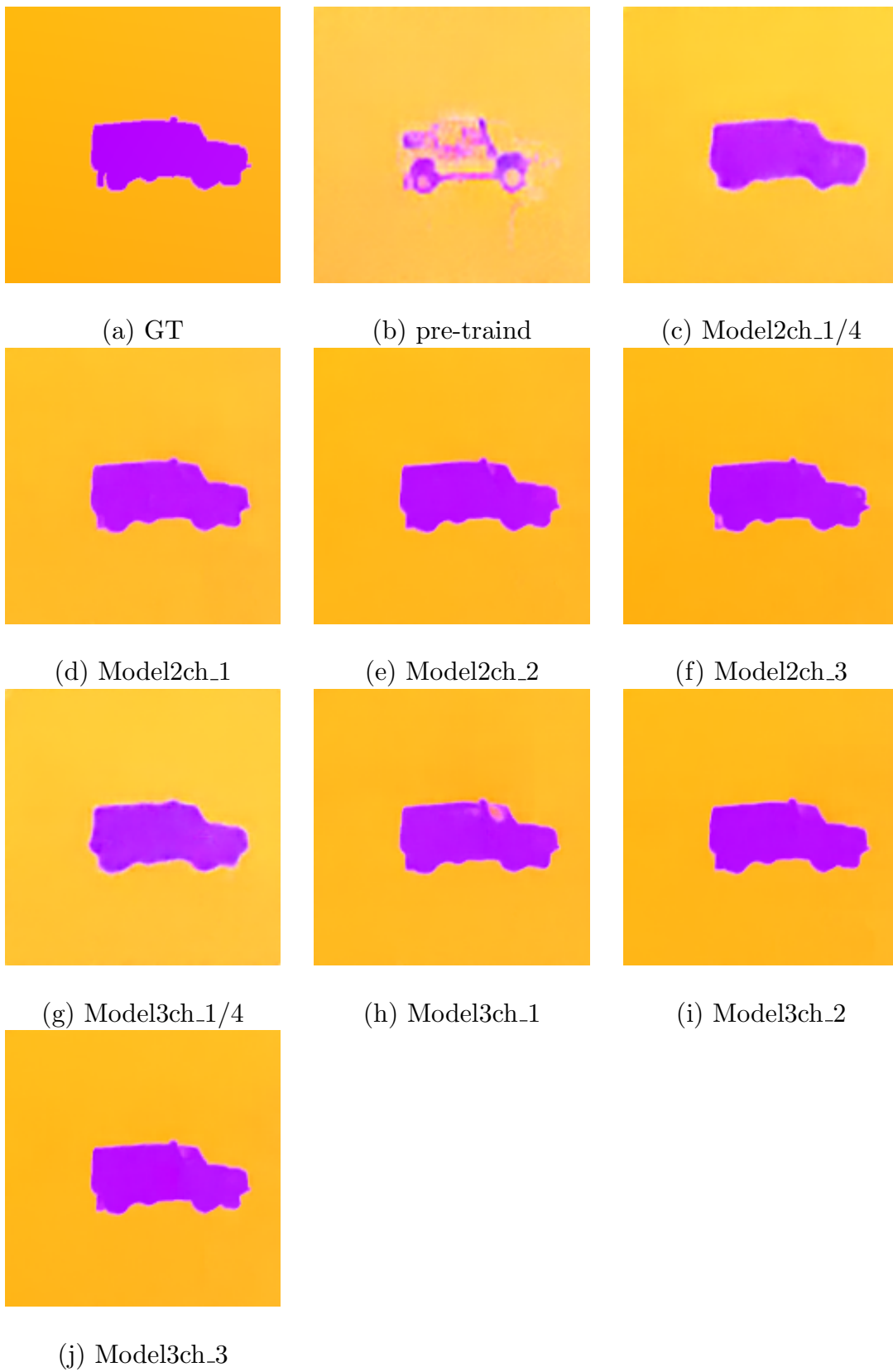


図 5.2: 評価により推定されたオプティカルフローと正解ラベル

5.3 推定したオプティカルフローの誤差

まず、pre-traind と提案モデルについての比較を行う。表 5.2, 5.3 より、チャンネル数がある程度増加させることで推定精度が高くなっていることが確認できる。また、pre-traind と同じチャンネル数である Model2ch_1, Model3ch_1 以降のモデルでは、提案モデルの方が推定精度が高いという結果になった。このことから、同じチャンネル数で学習を行った際には、運動前後のカラー画像よりも強度画像・相関画像の方が運動に対する有益な特徴を抽出することができ、さらにチャンネル数を増加させることでより多くの特徴を得ることができているといえる。

続いて、提案モデルである Model2ch と Model3ch について比較を行う。表 5.2 より、varidation 時の結果では前述したように Model2ch_3 で最も良い精度が得られた。しかし、表 5.3 より評価結果では、全体的に Model3ch の方が推定精度が高く、Model3ch_3 で最も良い精度が得られた。さらに、Model3ch_3 は、同じチャンネル数である Model2ch_3 と比較しても 10%も推定精度が向上していることが確認できる。これは、4.2 節で前述したように、相関画像とは異なる強度画像を加えたことによる影響であると考えられる。このことから、Model3ch は Model2ch と比較して、オプティカルフローを推定するにあたって汎用性の高いモデルであるということがいえる。また、表 5.3 より、チャンネル数を 3 倍増加させた Model2ch_3, Model3ch_3 では、それぞれ 2 倍増加させているモデルよりも精度が悪化していることから、オーバーフィットしていることが確認できる。これより、強度画像、相関画像が保持している有益な特徴を抽出するには、2 倍のチャンネル数で十分であると考えられる。

5.3 推定したオプティカルフローの誤差

正解ラベルと各モデルで推定されたオプティカルフローとの誤差の大きさをグレイスケール画像として可視化したものを図 5.3 に、誤差の方向をオプティカルフロー画像として可視化したものを図 5.4 に示す。図 5.3, 5.4 は、画像として可視化する際に pre-traind の最大誤差 0.33 で正規化を行っており、誤差 0 を白、誤差 0.33 以上を黒と設定したものになって

5.3 推定したオプティカルフローの誤差

いる.

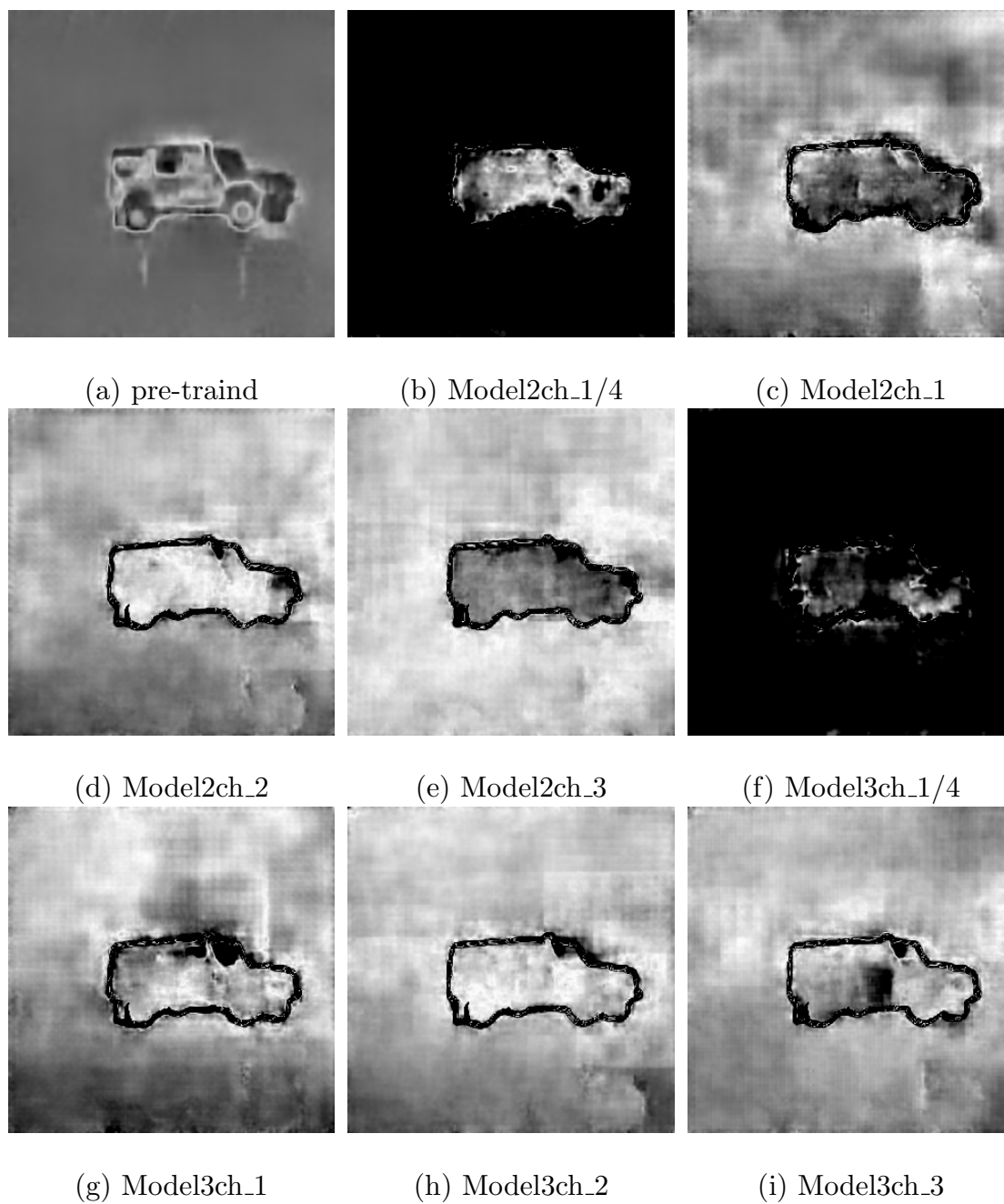


図 5.3: 誤差の大きさをグレースケール画像として可視化

5.3 推定したオプティカルフローの誤差

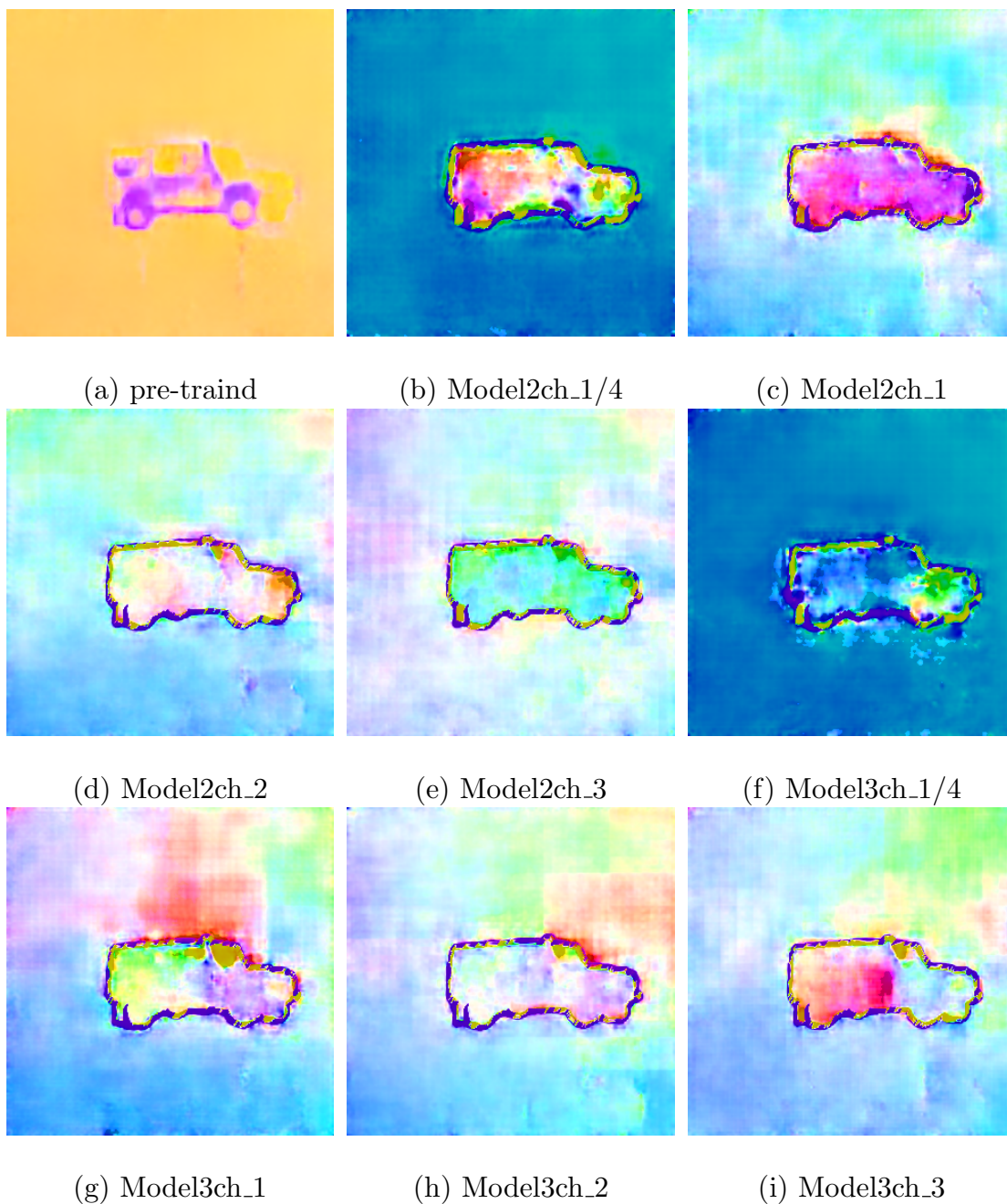


図 5.4: 誤差の方向をオプティカルフロー画像として可視化

図 5.3, 5.4 より, pre-traind は全体的に誤差が広がっていることが確認できる. 一方で, pre-traind と同様のチャンネル数である Model2ch_1, Model3ch_1 以降の提案モデルでは, 前景と後景の境界に大きな誤差が集中しており, 全体的にみると白に近く, pre-traind より

5.3 推定したオプティカルフローの誤差

も誤差が少ないことが確認できる．そのため，2 枚のカラー画像を相関画像・強度画像に変更することで，pre-traind よりも運動に対する有益な情報が得られていると考えられる．

以上の評価結果より，pre-traind と同じのチャンネル数以降の全ての提案モデルで pre-traind よりも高い推定精度を得られていることが確認できた．このことから，運動前後のカラー画像と比較して，時間相関カメラの出力である強度画像と相関画像は，運動に対する有益な情報を保持しているといえる．そのため，FlowNetS 以外のオプティカルフローを推定するその他のネットワークについても，入力画像を時間相関カメラからの出力に変更することで従来よりも高い推定精度を得ることができると考えられる．

第 6 章

まとめ

本研究では、1 枚の相関画像を入力とした場合と、強度画像と相関画像の 2 枚を入力とした場合の大きく 2 パターンの FlowNetS ベースの深層学習ネットワークを構築した。そして、新しいデータセットを用いて学習済みである FlowNetS との比較を行った。

その結果、pre-traind と同様のチャンネル数である Model2ch_1, Model3ch_1 以降の全てのモデルで pre-traind よりも高い精度を示すことが確認できた。また、Model2ch と Model3ch の結果より、1 枚の相関画像を入力とした場合と強度画像と相関画像の 2 枚を入力とした場合とでは、強度画像と相関画像の 2 枚を入力とした Model3ch_2 で最も良い精度を得ることが確認できた。

これらの結果より、時間相関カメラの出力である強度画像と相関画像には、運動に対する有益な情報が保持されていることが示された。そのため、FlowNetS 以外のオプティカルフローを推定するどのようなネットワークにおいても、入力画像を時間相関カメラからの出力に変更することで、従来よりも高精度にオプティカルフローを推定することができると期待される。

謝辞

本研究の実施の機会を与えて戴き，その遂行と論文の執筆にあたって終始，多大なるご指導およびご助言を受け賜りました高知工科大学 情報学群 栗原 徹 准教授に，心より感謝し，厚く御礼申し上げます。

並びに，本研究の副査をお引き受けくださった同学群 松崎 公紀 教授，繁桝 博昭 教授においては，適切にご指導を戴きましたことを深く御礼申し上げます。

参考文献

- [1] B.K.P. Horn, B.G. Schunck : "Determining Optical Flow", Artificial intelligence, vol. 17, pp. 185-203 (1981)
- [2] S. Ando and A. Kimachi : "Correlation image sensor: two-dimensional matched detection of amplitude-modulated light," IEEE Trans. on Electron Devices, vol. 50, no. 10, pp. 2059-2066 (Oct.2003)
- [3] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner : "Gradient-based learning applied to document recognition", Proceedings of the IEEE, 86-11, p.2278-2324, (1998)
- [4] Jonathan Long, Evan Shelhamer, Trevor Darrell, : "Fully Convolutional Networks for Semantic Segmentation," CVPR 2015, p.3431-3440, (2014)
- [5] A. Dosovitskiy et al : "FlowNet: Learning Optical Flow with Convolutional Networks," 2015 IEEE International Conference on Computer Vision (ICCV), pp. 2758-2766,(2015)
- [6] Ando, S. and Nara, T. : "An exact direct method of sinusoidal parameter estimation derived from finite fourier integral of differential equation," IEEE Transactions on Signal Processing, 57(9):3317–3329. (2009)
- [7] Ando, S., Kurihara, T., and Wei, D. : "Exact algebraic method of optical flow detection via modulated integral imaging –theoretical formulation and real-time implementation using correlation image sensor," In Proc. Int. Conf. Computer Vision Theory and Applications (VISAPP 2009), pages 480–487, (2009)
- [8] 魏 大比, マズレル ポル, 栗原 徹, 安藤 繁 : "複素正弦波変調撮像法に基づくオプティカルフロー推定," 電子情報通信学会和文論文誌 D, Vol. J90-D No.8 pp.2009-2018 (2007)
- [9] 栗原徹, 安藤繁, : "時間相関イメージセンサを用いた荷重積分法によるオプティカルフ

参考文献

- ロー検出と GPU による高速化,” 平成 25 年電気学会全国大会, pp.170–171, 3/20-22, 名大 (名古屋), (2013)
- [10] Kurihara, T. and Ando, S. : ”Tv minimization of direct algebraic method of optical flow detection via modulated integral imaging using correlation image sensor,” In 2014 International Conference on Computer Vision Theory and Applications (VISAPP), volume 3, pages 705–710, (2014)
- [11] ClementPinard, : ”FlowNetPytorch”, <https://github.com/ClementPinard/FlowNetPytorch>, 閲覧日 2020 年 1 月 11 日