

平成 28 年度自動車安全運転センター交通安全等に関する調査研究

事業用自動車の安全対策に資するもの 報告書

ドライブレコーダ記録映像に基づく危険度推定を活用した安全運転  
教育システムの研究開発

平成 29 年 5 月

九州工業大学大学院  
榎田 修一

株式会社 堀場製作所  
石倉 理有

# 目次

目次	2
第1章 序論	3
第2章 車両の検出について	5
2.1 はじめに	5
2.2 Deformable Part Model	5
2.3 Latent SVM による Deformable Part Model の構築	6
2.4 拡張 Deformable Part Model を用いた追突検出実験	10
第3章 歩行者の検出について	15
3.1 はじめに	15
3.2 人物検出に有効な手法	15
3.3 歩行者向き推定	16
3.4 実験	16
第4章 危険度推定システムについて	27
4.1 はじめに	27
4.2 車両検出に関する実験結果について	27
4.3 歩行者検出に関する実験結果について	27
4.4 非優先道からの進入判別に関する実験結果について	27
4.5 危険度推定システムの構築	27
第5章 結言	34
付録 再帰型深層学習 (Long Short Term Memory) について	35

# 第1章 序論

近年、運送事業における運行管理の義務化や、安全運転支援効果の期待から、事業用車両へのドライブレコーダの搭載が普及している。また、ドライブレコーダに備えられる記録媒体の高性能化、低価格化により、映像を記録するタイミングも、従来のトリガ（加速度センサによる急制動感知）による部分記録型から、常時記録型へと移行している。映像の撮りもちがなくなり、詳細な運転状況を把握可能になった反面、多くの事業所では、映像の洪水により、最終的には目を通さない映像も多い現状がある。そこで、一般には、常時記録の映像から危険な場面を洗い出すためには、従来のトリガに基づく危険度によることとなる。以上の方式では、事故等の発生により、安全運転指導が必要な運転手が判明した場合は、トリガにより記録された映像と、それ以外の箇所でも映像を確認することができる利点がある。一方、自身も気づかない危険な運転マナーが身についてしまった運転手については、運行管理者は従来のトリガ付近の映像のみから「要改善」の判断が必要となる。

本研究では、タクシーや配送用トラック等の事業用自動車に設置されたドライブレコーダを対象に、記録された映像と速度情報、電子地図情報等から、車体の挙動を計測しても検出することのできないヒヤリハットを自動で収集するための画像処理システムを構築することを目指して研究する。具体的には、車両同士の事故の半数を占める(1) 直進車両同士の追突、(2) 交差点での出会い頭および安全運転意識と深い関係のある非優先道からの進入時の一時停止不履行を対象に検討する。(事故類別、割合については、自動車技術会2016年春季大会フォーラム「ドライブレコーダ活用の最前線」によって発表された数値に基づいている。) また、重大な事故につながる(3) 歩行者との事故の予防についての自動解析に向けても検討する。具体的には、ドライブレコーダ記録映像からの車両の検出((1)への対応)、非優先道から優先道への進入場面の検出((2)への対応)、歩行者の検出((3)への対応)を研究する。

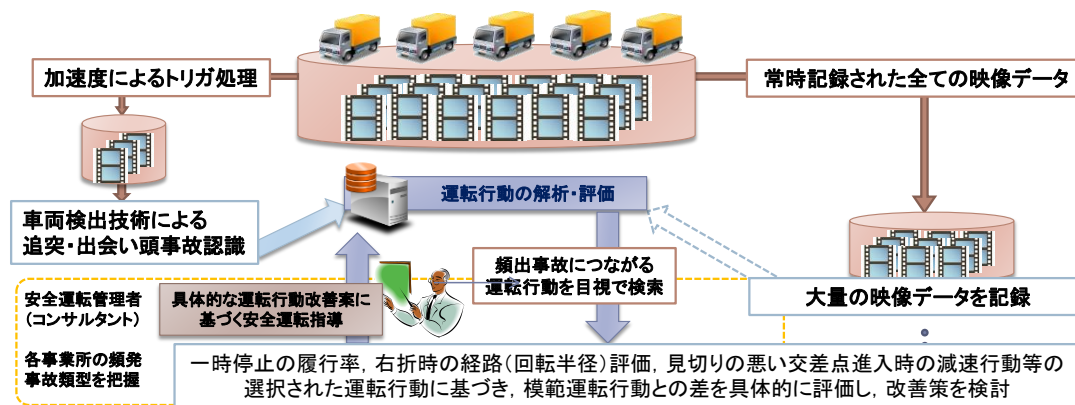


図1.1：加速度によるトリガ処理だけを備えたドラレコ映像処理システムを利用した安全運行管理の概要。大量の映像データは活用されることが無く、事故を起こした等の場合のみ目視されることが一般的である。

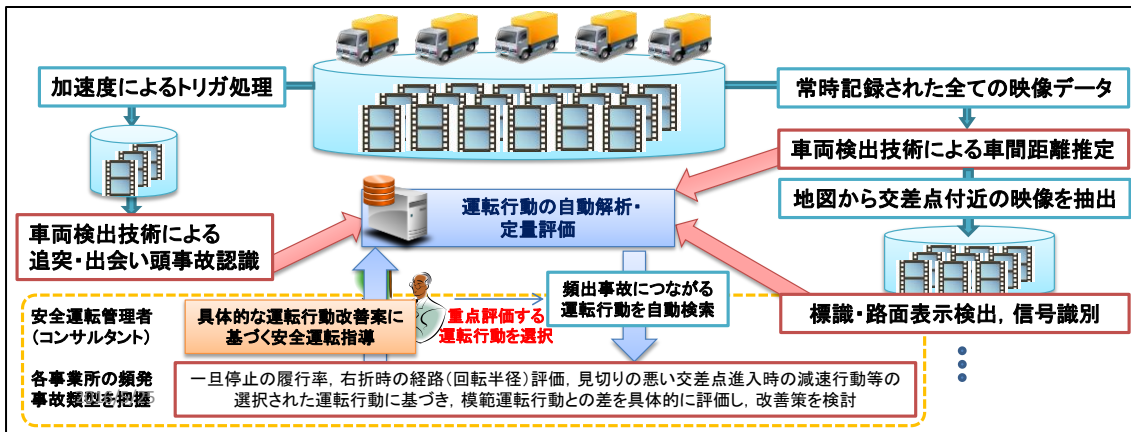


図1.2：画像の自動処理を備えたドラレコ映像処理システムを利用した安全運行管理の概要。常時記録された映像データは、運行管理者の評価視点に対応したタグが自動で貼られており、希望の情報を効率よく参照可能となる。

本報告書において第2章では、車両の検出アルゴリズムの紹介とドライブレコーダ映像への適用結果の報告を行う。第3章では、歩行者検出アルゴリズムの紹介とドライブレコーダ映像への適用結果の報告を行う。非優先道からの進入判定アルゴリズムの紹介とドライブレコーダ映像への適用結果の報告は平成27年度の報告[1]にて行った。以上の結果を踏まえ、第4章ではそれらをまとめ、開発が期待される運転マナーの発見に関する危険度推定システムの構成を示す。最後に、第5章で結言を述べる。

#### 参考文献

- [1] 榎田修一，石倉理有，“ドライブレコーダ映像からの映像情報を用いた危険度推定”，平成27年度自動車安全運転センター研究助成事業報告書，2016。

## 第 2 章 車両の検出について[8]

### 2.1 はじめに

近年、画像処理を用いたシステムの需要が高まっており、画像処理技術による物体検出手法は様々なシステムに利用されている。代表的な例として、車載カメラを用いた安全運転支援システムが挙げられる。安全運転支援システムのように事故を未然に防止するシステムでは、運転手や歩行者の安全を守るために高精度な物体検出が要求され、システムの支えとなる物体検出精度はシステム全体の安全性や信頼性に大きく影響する。また、物体検出の精度向上を目指すために、主な検出対象として考えられる人や車両に対する検出精度が問題となるが、人は様々な姿勢をとることがあり、車両は車種により形状が異なる。そのような人の姿勢変化や、車種による車両の形状変化に頑健な検出手法として、Deformable Part Model (DPM) [1][2]を用いた物体検出がある。DPM は、対象全体を捉える一つのルートフィルタと、ルートフィルタ内の特徴的な局所領域を捉える複数のパートフィルタによって構成される。DPM におけるパートフィルタは検出時に配置を変えることができ、これにより姿勢変化や形状変化に頑健な検出を実現している。DPM に関連する研究[3][4][5]は様々行われているが、本研究では DPM の最大の特徴であるパートフィルタの生成アルゴリズムに注目した。DPM のパートフィルタ生成アルゴリズムにおいて、特徴的なエッジが特定の領域に集中してしまうとパートフィルタ生成後のエッジ強度更新により、後半に生成されるパートフィルタほどエッジ情報を考慮できなくなる。そこで、生成されるパートフィルタがエッジ強度を考慮できるエッジ強度更新に基づく拡張 DPM の提案を行い、特徴的な領域を有効に活用することで検出精度の向上を目指す。

### 2.2 Deformable Part Model

Deformable Part Model (DPM)の特徴は、検出対象全体を捉える一つのルートフィルタとルートフィルタ内の特徴的な局所領域を捉える複数のパートフィルタという 2 種類のフィルタによって構成され、検出時にパートフィルタが配置を変えることにより姿勢変化や形状変化に頑健な検出を実現していることである。DPM による車両検出例を図 2.1 に示す。検出対象画像には Stanford cars Dataset を用いた。図 2.1 中の水色矩形がルートフィルタ、黄色矩形がパートフィルタを表す。

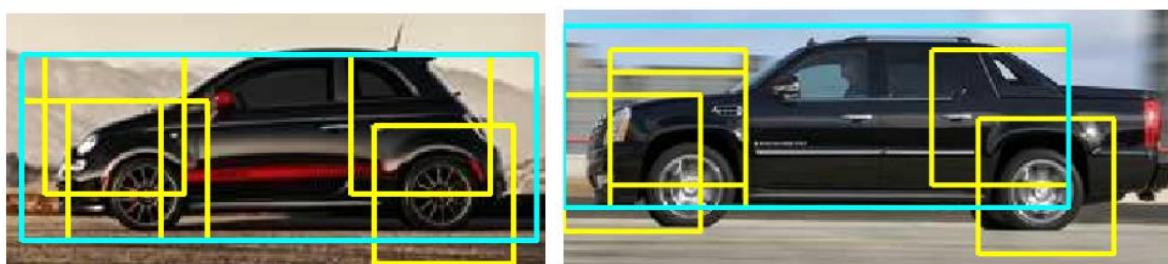


図 2.1 : Deformable Part Model を用いた車両検出例

#### 2.2.1 Deformable Part Model に基づく検出スコア

DPM では Histograms of Oriented Gradients (HOG) 特徴量が用いられている。DPM を用いた検出では、パートフィルタはルートフィルタ内の特徴的な局所領域を捉える必要があるため、パートフィルタではルートフィルタの 2 倍の解像度で HOG 特徴量を算出する。そこで、図 2.2 に示す HOG ピラミッドを定義する。HOG ピラミッドは、異なる解像度の画像と HOG 特徴量をピラミッドのように並べたもので、上層ほど低い解像度画像、下層ほど高い解像度画像を用いる。

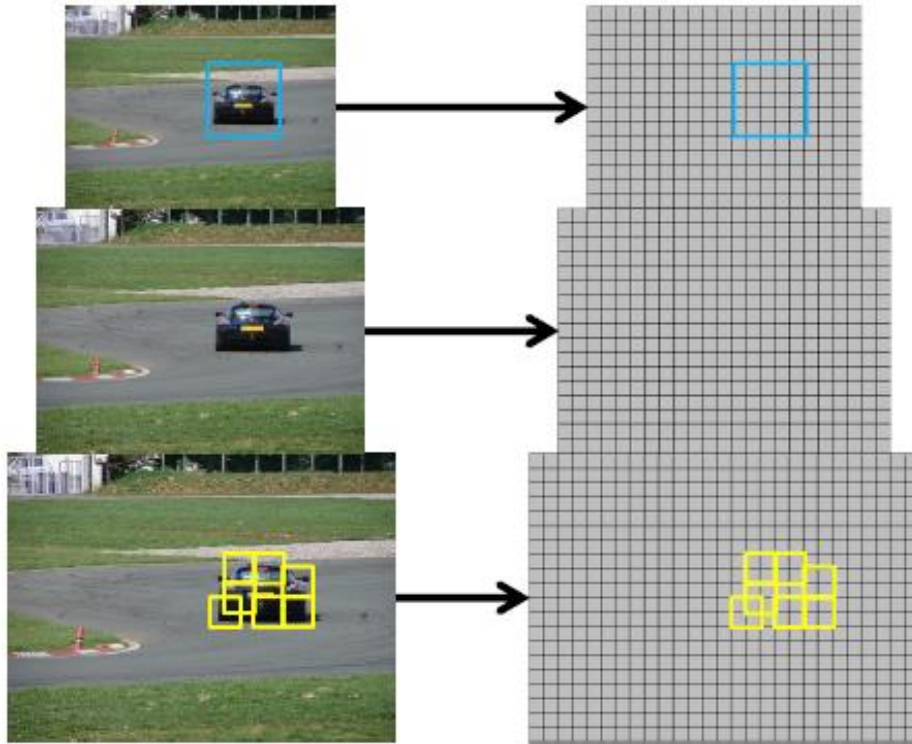


図 2.2 : HOG ピラミッド

## 2.3 Latent SVM による Deformable Part Model の構築

DPM の構築において、学習用データセットは PASCAL Visual Object Classes の VOC2007 が用いられている。各学習画像には annotation ファイルが用意されており、画像中にどのようなカテゴリの物体がどの領域に写っているかという情報が記述されている。

### 2.3.1 ルートフィルタの初期化

ルートフィルタの初期化において、bounding box の情報をもとに学習を行う。まず、学習を行うカテゴリの全ポジティブサンプルの bounding box 情報を取得する。その後、bounding box の情報から計算されたアスペクト比の統計値により、ルートフィルタのサイズを決定する。次に、潜在変数を用いずに通常の SVM によってルートフィルタの学習を行う。このとき、学習画像から bounding box の領域を切り取った画像をポジティブ画像とし、ポジティブ画像を決定されたルートフィルタのサイズにリサイズしたものを使用する。ネガティブ画像は、ネガティブサンプル画像からランダムに切り取った画像を使用する。

### 2.3.2 ルートフィルタの更新

ルートフィルタの更新では、3.1 節により決定されたルートフィルタの再学習を行う。ここではポジティブ画像のリサイズは行わず、リサイズ前のポジティブ画像を用いて学習を行う。

### 2.3.3 パートフィルタの初期化

パートフィルタの初期化は、以下の(a)～(c)のステップで行う。 $p$ 個のパートフィルタ生成時には(a)～(c)を $p$ 回繰り返す。

- (a) エッジ強度の総和が最大となる局所領域を探索

(b) (a) で選択された領域にパートフィルタを配置

(c) パートフィルタ内のエッジ強度を 0 に更新

2.3.3.2 節ではステップ(a), 2.3.3.3 節ではステップ(c) における処理手順を詳述する.

### 2.3.3.1 ルートフィルタ内のエッジ強度計算

2.3.3.2 節に示す処理の前準備として, まず学習されたルートフィルタ内のエッジ強度計算を行う. 計算結果は学習するアスペクト比ごとに配列として出力され, エッジ強度の強い領域ほど配列要素の値は大きくなる. 学習するルートフィルタのサイズが  $4 \times 11$  セルのとき, パートフィルタ生成時にはルートフィルタの 2 倍の解像度で HOG 特徴量を計算するため, 計算結果は  $8 \times 22$  セルの配列として出力される. 実際に使用された学習画像例を図 2.3, 学習によって計算された初期エネルギーのヒートマップを図 2.4 に示す.



図 2.3 : 学習画像例

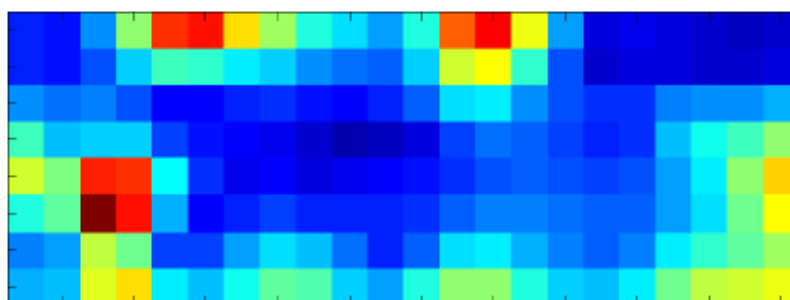


図 2.4 : 初期エネルギーのヒートマップ

エッジ強度が強い領域ほど赤に近い色で, エッジ強度が弱い領域ほど青に近い色で表示されている. 図 2.3 と図 2.4 より, タイヤや屋根付近, フロント部分など, 車両として特徴的な領域がヒートマップに反映されている.

### 2.3.3.2 パートフィルタ生成領域の選択

2.3.3.1 節に示した処理が終わると, 次はエッジ強度の総和が最大となる局所領域を探索する. パートフィルタ生成領域の選択において, エッジ強度の総和が最大となる局所領域を探索するために, ルートフィルタ内の全ての領域に  $6 \times 6$  セルのフィルタがかけられ, それぞれの領域におけるエッジ強度の総和計算が行われる. 従来手法では平均化フィルタが用いられ, フィルタ内のエッジ強度の配置によらず同じ重み付けが行われる. 提案手法では, 従来手法における平均化フィルタをガウシアンフィルタに変更し, フィルタの中心付近に位置するエッジ強度ほど大きな重み付けを行った.

### 2.3.3.3 パートフィルタ内のエッジ強度更新

2.3.3.2 節で選択された領域においては, エッジ強度の更新が行われる. 従来手法では, パートフィルタが生成された領域内のエッジ強度をすべて 0 に更新する. ルートフィルタ内のエッジ強度の座標を  $\mathbf{x}$ ,  $n$  個目のパートフィルタ生成時のルートフィルタ内のエッジ強度を  $E_n$ ,  $E_n$  内でパートフィルタが生成される領

域を $R_n$ ,  $R_n$ の中心座標を $\mathbf{p}$ とすると, 従来手法におけるエッジ強度の更新は式(5)に基づき行われる. また, 式(6)に示す関数を図 2.5 に示す.

$$E_{n+1} = W(\mathbf{x}; \mathbf{p}) \times E_n \quad (5)$$

$$W(\mathbf{x}; \mathbf{p}) = \begin{cases} 0 & \mathbf{x} \in R_n \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

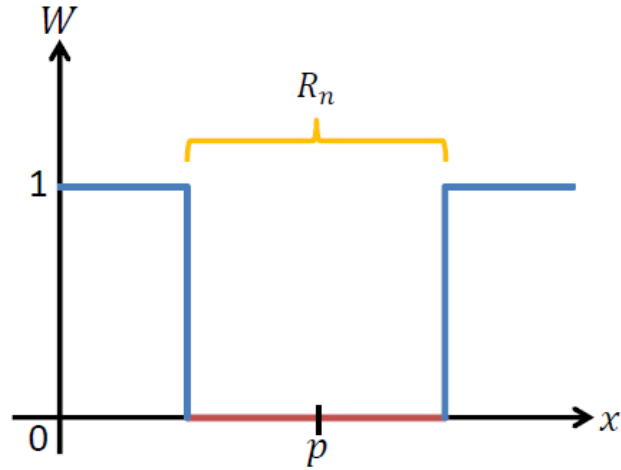


図 2.5 : DPM におけるエッジ強度更新関数

ここで,  $6 \times 7$  セルのルートフィルタを例とし, 初期エネルギーのヒートマップを図 2.6 に示す. また, 従来手法によるエッジ強度更新の様子を図 2.7 に示す.

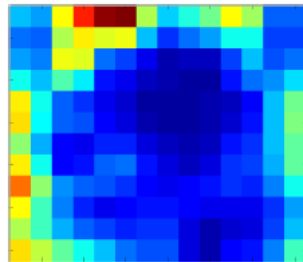


図 2.6 : 初期エッジ強度 ( $6 \times 7$  セル)

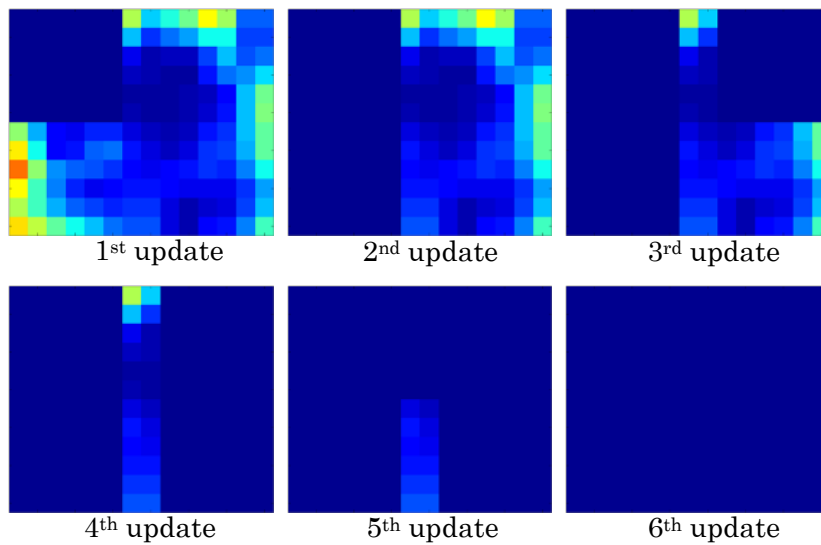


図 2.7 : エッジ強度更新の様子 (DPM)



図 2.7 より，更新 6 回目でルートフィルタ内の全ての領域でエッジ強度が 0 に更新されている．ルートフィルタ内の全ての領域でエッジ強度が 0 に更新されてしまうと，次に生成されるパートフィルタがエッジ強度を考慮できなくなってしまう．そこで，提案手法ではエッジ強度の更新をガウス分布に基づくエッジ強度の削減に変更した．提案手法におけるエッジ強度更新を式(7)に示す．また，式(8)に示す関数を図 2.8 に示す．

$$E_{n+1} = W(x; \mathbf{p}, \sigma) \times E_n \quad (7)$$

$$W(x; \mathbf{p}, \sigma) = \begin{cases} G(x; \mathbf{p}, \sigma) & x \in R_n \\ 1 & \text{otherwise} \end{cases} \quad (8)$$

$$G(x; \mathbf{p}, \sigma) = 1 - \exp\left(-\frac{(x - \mathbf{p})^2}{2\sigma^2}\right) \quad (9)$$

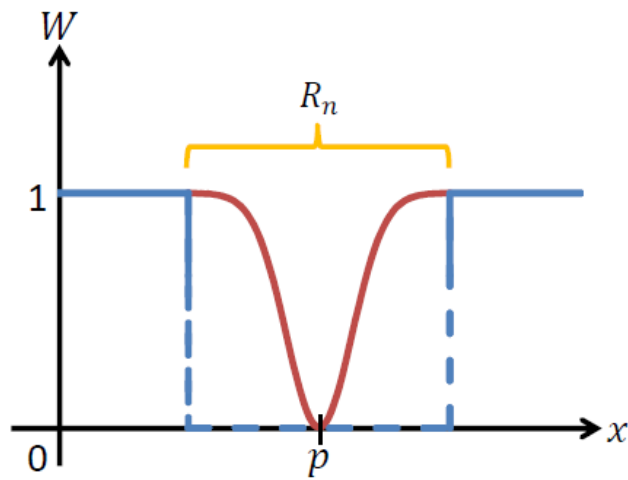


図 2.8 : 拡張 DPM におけるエッジ強度更新関数

提案手法によるエッジ強度エッジ強度更新の様子を図 2.9 に示す．

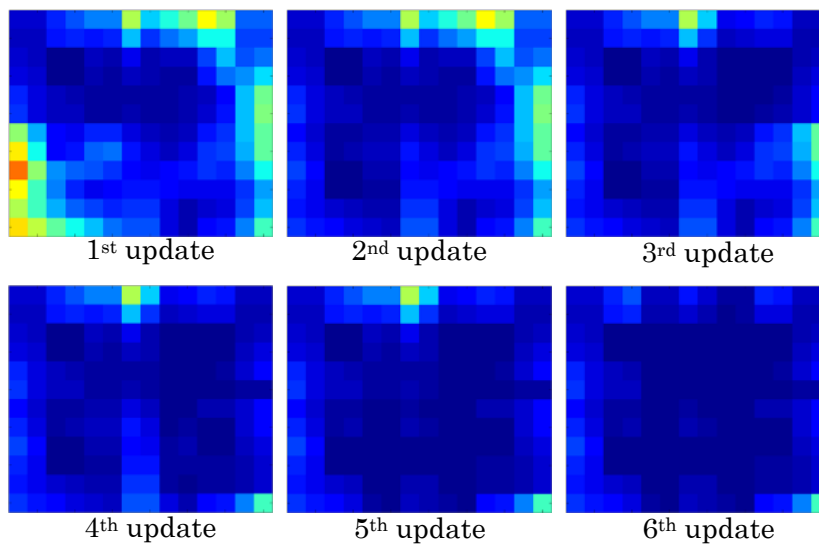


図 2.9 : エッジ強度更新の様子(拡張 DPM)

更新6回目のエッジ強度を比較すると、従来手法ではルートフィルタ内の全ての領域でエッジ強度が0に更新されているのに対し、提案手法ではエッジ強度が残っていることが確認できる。このことから、提案手法におけるエッジ強度更新により、効果的に多くのパートフィルタが配置されることが期待される。

### 2.3.4 モデルの更新

モデルの更新で学習するパラメータは、フィルタのスコアに関わる重みベクトルとパートフィルタの配置 $z$ の二つである。この二つのパラメータを同時に学習することはできないため以下の手法を用いる。

$\beta$ を固定し、最もスコアの高いパートフィルタの配置 $z_i$ を求める。

$$z_i = \operatorname{argmax}_{z \in Z(x_i)} \beta \cdot \varphi(H(x_i), z) \quad (10)$$

次に、 $z$ を固定し、固定された配置で最も高いスコアとなる $\beta$ を求める。

$$\beta^*(D) = \operatorname{argmin}_{\beta} \left\{ \lambda \|\beta\|^2 + \sum_{i=1}^n \max(0, 1 - y_i f_{\beta}(x_i)) \right\} \quad (11)$$

## 2.4 拡張 Deformable Part Model を用いた追突検出実験

本実験では、従来手法と提案手法を用いた車両検出のうち追突検出に注目し、定性評価実験を行った。拡張 DPM の定量評価実験については文献(7)において既に行われており、生成するパートフィルタ数を8個とした実験結果を図 2.10 に示す。ここで、パートフィルタのサイズを $s$ とした。また、図 2.10 中の提案手法名の設定を表 2.1 に示す。

表 2.1 : 提案手法の設定

手法名	$s$ に対する標準偏差 $\sigma$
提案手法 1	0.5s
提案手法 2	0.7s
提案手法 3	1.0s

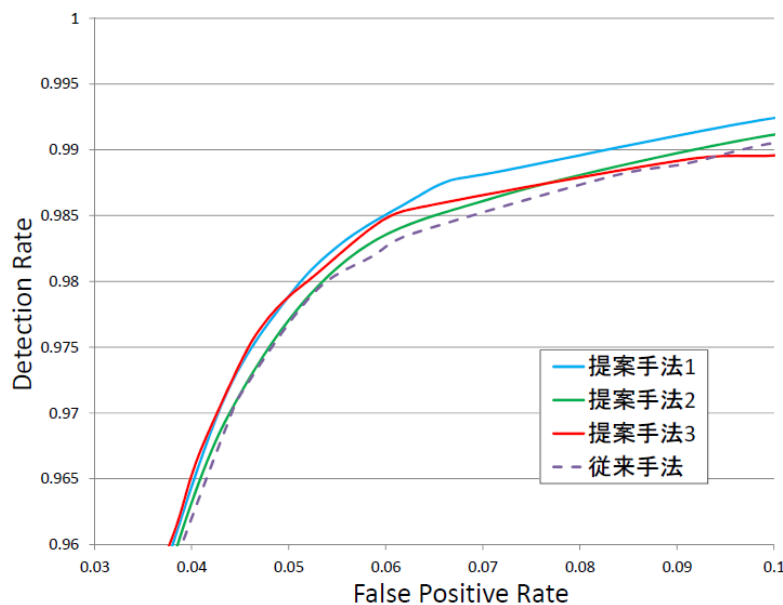


図 2.10 : 定量評価実験結果

図 2.10 より，定性評価実験に用いる標準偏差 $\sigma$ は 0.5s とした．また，アスペクト比ごとの評価を図 2.11 に示す．

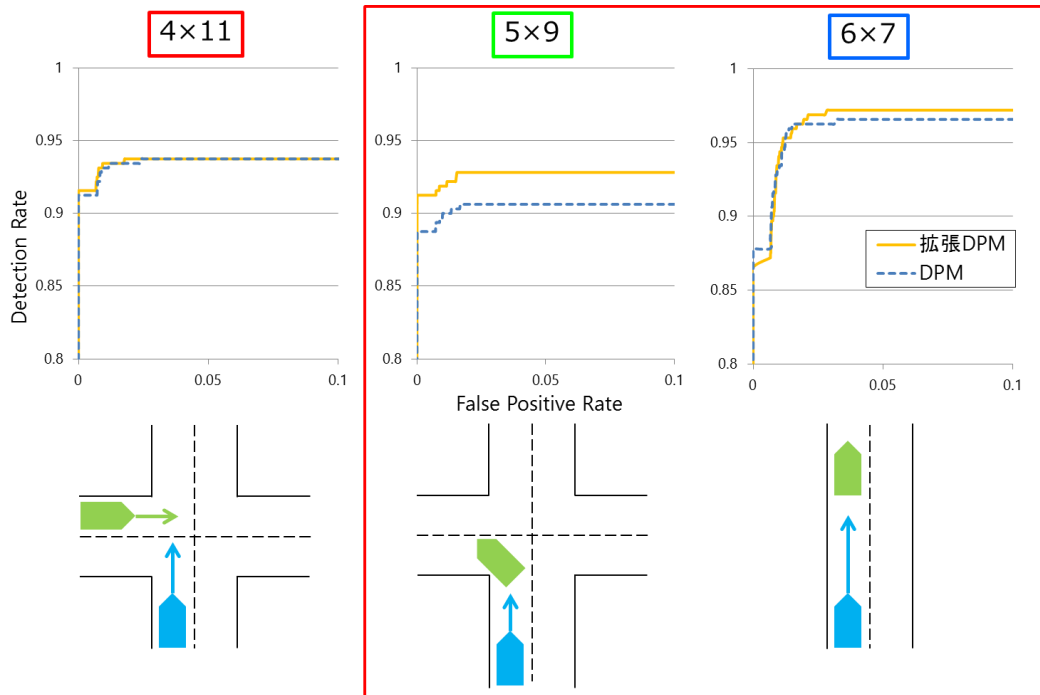


図 2.11 : アスペクト比ごとの評価結果

### 2.4.1 ドライブレコーダ記録映像における定性評価実験

定性評価実験において，農工大 TL0 より提供されたドライブレコーダ記録映像を用いた．全データのうち，ヒヤリハット類型が追突と分類されたデータのみに対して，従来手法と提案手法による検出結果の比較を行った．使用するドライブレコーダデータの検索画面を図 2.12 に示す．



図 2.12 : ドライブレコーダデータ検索画面

実験に使用するデータの詳細な分類を以下に示す。図 2.12 より、ヒヤリハットの度合いを示すヒヤリハットレベルは、高、中レベルとした。対象は車両のうち乗用車とし、ヒヤリハットの類型は追突とした。自車動作、道路形状、カメラ、地域はすべての場合とした。検索の結果、482 件のデータが条件に合致した。しかし、そのうち 1 件は後進中の記録により対象車両が確認できず、2 件は車内の映像であった。また、7 件は検出対象がトラック、もしくは軽トラックであり、19 件は対象車両との距離が遠かった。そのため、これら 29 件を除く 453 件を実験対象データとした。

#### 2.4.1.1 評価方法

評価を行ったデータに対して、ID、照明条件、従来手法と提案手法での検出と誤検出、また 2 つの手法の比較を記録する。照明条件は「逆光」、「良好」、「暗い」のいずれかを記入する。従来手法と提案手法の検出は、検出対象を検出できた場合は「可」、できなかった場合は「不」とした。誤検出は、「無」、「少」、「多」とした。また、従来手法と提案手法の検出と誤検出の比較をそれぞれ行う。提案手法による検出精度が向上すれば「○」、精度が低下すれば「×」、同程度であれば「-」と記入する。また、提案手法により誤検出が減少すれば「○」、増加すれば「×」、変化がなければ「-」を記入する。評価例を表 2.2 に示す。

表 2.2：追突検出実験の評価例

ID	照明条件	従来手法		提案手法		比較	
		検出	誤検出	検出	誤検出	検出	誤検出
36	良好	可	多	可	少	-	○
48	暗い	不	少	不	無	-	○
50	暗い	不	少	不	少	-	-
77	良好	不	少	不	少	-	-
93	暗い	不	無	不	無	-	-
94	良好	不	少	不	少	-	-
110	良好	可	少	可	少	-	-
239	良好	可	少	可	少	-	-
259	暗い	不	少	不	少	-	-

#### 2.4.1.2 追突検出実験結果

照明条件別の評価結果を表 2.3 に示す。照明条件ごとに最も検出率が高い箇所を赤色の塗りつぶしで表示している。表 2.3 中の検出率は、照明条件ごとの全データ数のうち検出が行えたデータ数の割合を示している。

表 2.3：照明条件による評価結果

照明条件	全データ	検出可		検出率(%)	
		従来手法	提案手法	従来手法	提案手法
逆光	24	13	13	54.2	54.2
良好	288	248	249	86.1	86.5
暗い	141	43	46	30.5	32.6

また、従来手法と提案手法の比較に関して、検出が「○」または「×」と評価されたデータ数と、誤検出が「○」または「×」と評価されたデータ数を表 2.4 に示し、照明条件ごとの内訳を表 2.5 に示す。

表 2.4：全データ数

照明条件	検出○	検出×	誤検出○	誤検出×
逆光	0	0	4	3
良好	3	2	35	17
暗い	3	0	20	5

表 2.5 : 照明条件ごとの内訳

検出○	検出×	誤検出○	誤検出×
6	2	59	25

次に、評価結果から検出が「○」とされたデータにおける検出結果の例を示す。従来手法による検出結果を図 2.13、提案手法による検出結果を図 2.14 とする。画像は従来手法と提案手法による検出結果を連結させたものとなっており、左側が従来手法による検出、右側が提案手法による検出となっている。対象車両の後部の検出が行えた際、青色の矩形を出力している。

#### 参考文献

- [1] P.Felzenszwalb, R.Girshick, D.McAllester, D.Ramanan, Object Detection with Discriminatively Trained Part Based Models, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.32, No.9, pp. 1627-1645, Sep. 2010.
- [2] P.Felzenszwalb, D.McAllester, D.Ramanan, A discriminatively trained, multiscale, deformable part model, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008.
- [3] Eduard Trulls, Stavros Tsogkas, Iasonas Kokkinos, Alberto Sanfeliu, Francesc Moreno-Noguer, Segmentation-aware Deformable Part Models, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 168-175, 2014.
- [4] Yicong Tian, Rahul Sukthankar, Mubarak Shah, Spatiotemporal Deformable Part Models for Action Detection, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2642-2649, 2013.
- [5] Fidler, Sanja and Sven Dickinson and Urtasun, Raquel, 3D Object Detection and Viewpoint Estimation with a Deformable 3D Cuboid Model, Advances in Neural Information Processing Systems 25, 2012, pp. 611-619.
- [6] M.Everingham, L.Van Gool, C.K.I.Williams, J.Winn, A.Zisserman, The PASCAL Visual Object Classes Challenge 2007.  
<http://host.robots.ox.ac.uk/pascal/VOC/voc2007/>  
(2016/07/28 アクセス)
- [7] 本石大記, 榎田修一, パートフィルタ生成条件に注目した拡張 Deformable Part Model の提案, 第 22 回画像センシングシンポジウム(SSII), IS3-05, 2016.
- [8] 本石大記, 榎田修一, ドライブレコーダ記録映像における拡張 Deformable Part Model を用いた追突検出, 自動車技術会秋季大会, 154(運転支援 V(ドライブレコーダ))-273, 2016.

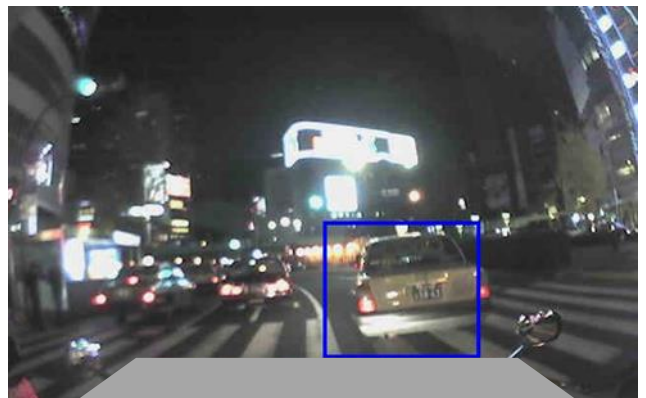
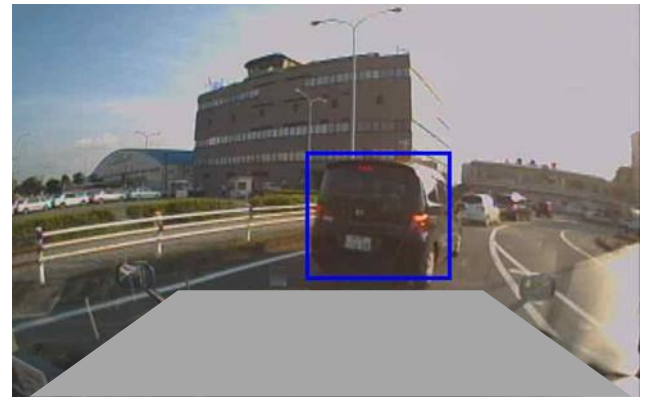
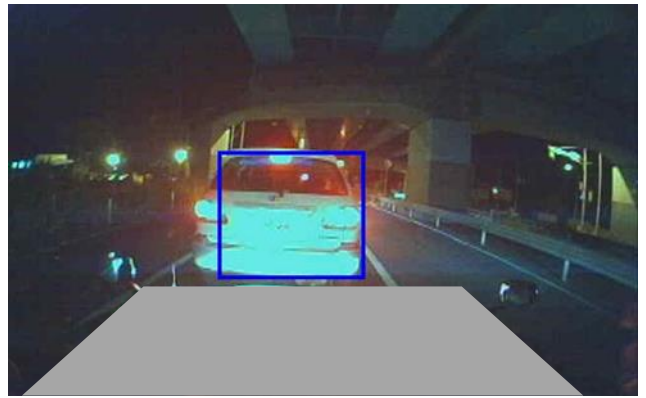


図 2.13 : DPM に基づく 追突検出

図 2.14 : 拡張 DPM に基づく 追突検出

## 第3章 歩行者の検出について[10]

### 3.1 はじめに

近年、交通事故の要因解析等に使用されるドライブレコーダが急速に普及している。しかし、ドライブレコーダにより記録された事故映像の解析は、一般的に目視によりなされている現状があり、数百に及ぶ事故状況のタグ付けは一つ一つ手作業で行われている。そのため、爆発的に増加する映像に対して対処できない現実がある。そこで、本研究では、事故映像の自動解析に向けて映像中の人物を自動検出する技術の開発を目指している。本研究では、高速かつ高精度な人物検出が可能な MRCoHOG 特徴量[1]を用いてドライブレコーダ映像から人物検出を行い、歩行者の詳細な解析の為に Deep Learning を用いた歩行者向き推定を行う。

### 3.2 人物検出に有効な手法

人物検出には、勾配分布に着目した特徴量が多数提案されており、特に Dalal らが提案した HOG 特徴量[2]は、セルと呼ばれる領域ごとに勾配強度ヒストグラムを作成し、正規化することで照明変化に頑健な人物検出を可能としている。このような勾配分布に着目した特徴量は、人物のわずかな平行移動や回転およびスケール変化に対する頑健性を向上させた。しかし、局所領域における単一の勾配分布に着目する場合、複雑な形状を捉えることができないため、人物に似た形状の物体を誤検出することが問題視されている。そこで、人物の形状をより精巧に捉えるために、局所領域におけるエッジの共起性を考慮した特徴量が提案されている。

Watanabe らは局所領域における勾配分布の 2 要素の共起性を扱う Co-occurrence Histograms of Oriented Gradients (CoHOG) 特徴量[3]を提案した。本研究で用いる MRCoHOG 特徴量は、様々なスケールのフィルタを用いて勾配を抽出し、2 点間の共起分布を生成する。図 3.1 に示すように、CoHOG 特徴量では勾配計算領域の広さが一定であるため、同じスケールの輪郭ペアしか存在しない。一方、MRCoHOG 特徴量は、注目画素と共起対象画素の勾配計算領域のスケールを別々に設定することができるため、特徴記述能力が高くなる。

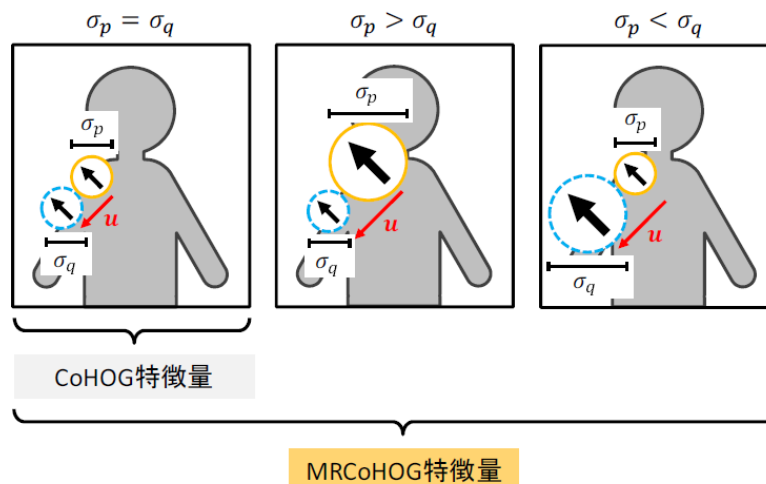


図 3.1 : CoHOG 特徴量と MRCoHOG 特徴量の概要

### 3.3 歩行者向き推定

ドライブレコーダにより記録された映像から歩行者を検出し、詳細な事故の要因を調べるために、歩行者の移動経路の解析が必要である。そのために、歩行者向き推定を行い、時系列情報と組み合わせることで、高精度な歩行者の移動経路の解析を行う。歩行者向き推定には、二値分類器の組み合わせによる歩行者向き推定に関する研究[4]のように画像特徴量を用いて向き推定する手法がある。しかし、ドライブレコーダ記録映像中の歩行者は一般的に低解像度のものが多く、画像特徴量を用いた手法では十分な性能を得ることが出来ない可能性がある。そのため、本研究では、低解像度の画像でも十分な性能を得ることができ、近年画像分類の分野で注目されている Convolutional Neural Network (CNN)を用いた歩行者向き推定を行い、従来手法との比較を行った。CNN には大規模画像分類において、高い精度を持つ Network In Network[5]を用いた。また、歩行者の向きを学習するためのデータセットとして Daimler Pedestrian Dataset を我々で歩行者の向き前後左右の4方向に分類した合計 21,000 枚のデータセットを用いて学習を行った。

### 3.4 実験

#### 3.4.1 基礎実験

人物画像のデータセットである INRIA Person Dataset を用いて、MRCoHOG の識別精度を検証した。識別器は Real AdaBoost により構築し、学習回数は 500 回に設定した。実験結果は図 3.2 に示す。図 3.2 のグラフは左上にあるほど精度が高いことを示す。図 3.2 より、MRCoHOG が全域で HOG,CoHOG を上回る結果となり、歩行者検出において十分な精度が得られると考えられる。

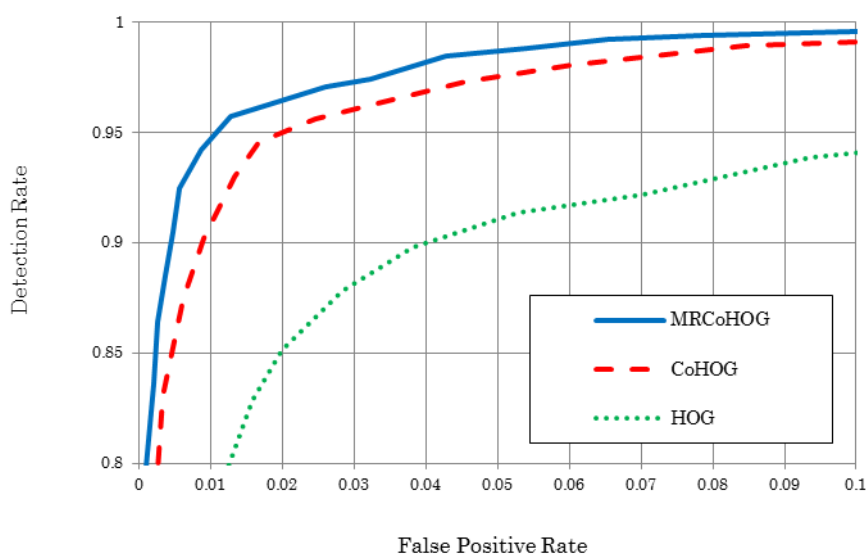


図 3.2 : 定量評価実験結果

#### 3.4.2 ドライブレコーダ記録映像からの歩行者検出

MRCoHOG 特徴量による人物検出は、ドライブレコーダにより記録された事故映像のうち、歩行者との事故映像 95 件を用いた。検出は、事前に車のボンネット付近に設定したベースラインの上で行う。先行研究[6]より、夜で暗い事や、逆光で人物のエッジが消失してしまい、検出が出来なくなる問題があった。本実験で用いたドライブレコーダ映像は、東京農工大学スマートモビリティ研究拠点の提供であり、個人を特定出来ない程度の解像度の映像を使用している。



表 3.1 : MRCoHOG 特徴量による検出率

	Detection rate
Pedestrian	72%

表 3.2 : 各行動による検出率

	crossing ahead	go forward	coming toward	stand
Detection rate	70%	70%	83%	100%

表 3.1 より，検出率は 72% となった．また，検出対象の動作別の検出率は表 3.2 より，停止中が最も高く，次に対面通行中，そして，背面通行中と横断中が最も低い結果となった．次に，未検出である動画に対し，検出対象を手動で切り出し，検出器に入力する実験を行った．これにより，ベースラインのズレによる未検出を発見し，残りの動画から，未検出の原因の解析を行う．実験の結果は以下のようになった．

表 3.3 : MRCoHOG 特徴量による検出率

	Detection rate
Pedestrian	96%

表 3.4 : 各行動における検出率

	crossing ahead	go forward	coming toward	stand
Detection rate	97%	90%	83%	100%

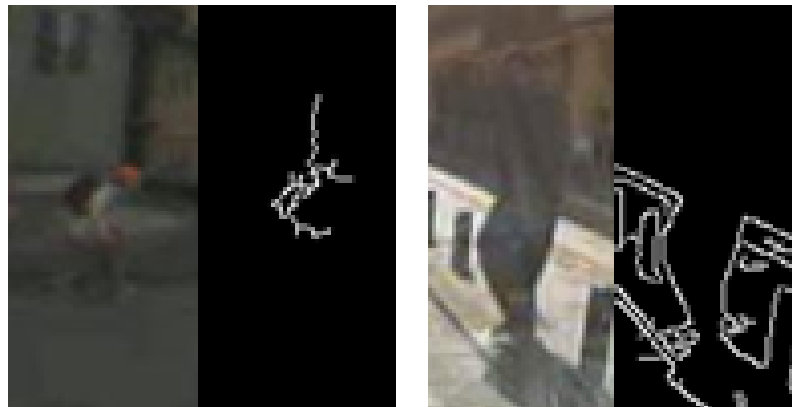


図 3.3 : 横断歩道の歩行者



図 3.4 : 背面通行中の歩行者



図 3.5 : 対面通行中の歩行者

表 3.3,3.4 より, 手動で切り出した場合, 全体で検出率は 96%となり, 特に横断中と背面通行中の検出率が向上した. 手動で切り出した場合でも, 未検出となったシーンは図 3.3,3.4,3.5 の 3つのシーンである. 図 3.3 と図 3.4 の 2つのシーンは共通して日陰によって対象の輝度差がほとんど無いため, 歩行者として検出が出来なかった. 図 3.5 の対面通行中では, 歩行者のエッジはとれているが, 特徴記述の際に, 荷物や背景のエッジの影響を受け, 結果としてスコアが低くなる原因となった. 実験の結果は表 3.5 にまとめた.

表 3.5 : 実験結果

ID	結果	対象の動作	場所	種類	手動
39	検出	停止中	路上停止中	大人	○
69	検出	停止中	路上遊戯中	子供	○
126	検出	横断中	その他	大人	○
469	検出	横断中	横断歩道	大人	○
1967	検出	横断中	その他	子供	○
1979	未検出	横断中	その他	子供	×
2053	検出	対面通行中	路側帯	大人	○
2068	未検出	横断中	その他	大人	○
2501	検出	横断中	その他	子供	○

ID	結果	対象の動作	場所	種類	手動
3131	検出	横断中	その他	大人	○
3259	未検出	横断中	横断歩道	大人	○
3486	未検出	対面通行中	車道	大人	×
3500	未検出	背面通行中	車道	大人	×
3532	未検出	横断中	その他	大人	○
3581	検出	背面通行中	車道	大人	○
3716	検出	横断中	その他	大人	○
3733	検出	対面通行中	路側帯	大人	○
4263	検出	横断中	横断歩道	大人	○
4273	未検出	横断中	その他	大人	○
4337	未検出	横断中	その他	大人	○
4478	検出	横断中	その他	大人	○
4510	未検出	横断中	その他	大人	○
5048	検出	横断中	横断歩道	大人	○
5145	検出	背面通行中	路側帯	子供	○
5159	検出	背面通行中	車道	子供	○
5170	検出	背面通行中	車道	大人	○
5311	検出	横断中	横断歩道	大人	○
5332	検出	横断中	横断歩道	大人	○
5689	未検出	背面通行中	車道	大人	○
5804	未検出	横断中	その他	大人	○
5858	検出	横断中	横断歩道	大人	○
6749	検出	横断中	その他	大人	○
6824	検出	背面通行中	車道	大人	○
7132	検出	横断中	横断歩道	大人	○
7452	未検出	横断中	その他	大人	○
7524	検出	横断中	その他	大人	○
7591	検出	横断中	横断歩道	大人	○
7616	未検出	横断中	その他	大人	○
7747	検出	横断中	横断歩道	大人	○
7939	検出	横断中	横断歩道	大人	○
8151	検出	横断中	その他	大人	○
8382	検出	横断中	その他	大人	○
8444	検出	背面通行中	車道	大人	○
8470	検出	横断中	その他	大人	○
8512	未検出	横断中	その他	大人	○
8734	検出	横断中	その他	大人	○

ID	結果	対象の動作	場所	種類	手動
8930	検出	横断中	その他	大人	○
9327	検出	横断中	横断歩道	大人	○
10431	検出	横断中	その他	大人	○
10692	検出	横断中	横断歩道	大人	○
10754	検出	横断中	その他	大人	○
11168	検出	停止中	路上作業中	大人	○
11275	検出	横断中	その他	大人	○
11459	検出	横断中	横断歩道	大人	○
11821	検出	横断中	横断歩道	大人	○
11907	検出	横断中	その他	大人	○
12060	未検出	背面通行中	車道	大人	○
12096	検出	対面通行中	車道	大人	○
12257	未検出	横断中	その他	大人	○
12401	未検出	横断中	その他	大人	○
12565	検出	横断中	横断歩道	大人	○
12686	検出	横断中	横断歩道	大人	○
12739	検出	横断中	その他	大人	○
12775	検出	横断中	横断歩道	大人	○
13383	検出	横断中	その他	大人	○
13446	未検出	横断中	横断歩道	大人	○
13475	検出	横断中	横断歩道	子供	○
13478	検出	横断中	その他	大人	○
13579	未検出	横断中	横断歩道	大人	○
13624	検出	横断中	横断歩道	大人	○
13665	検出	横断中	その他	大人	○
14455	未検出	横断中	横断歩道	大人	○
14547	検出	対面通行中	車道	大人	○
14642	未検出	横断中	その他	大人	○
14709	検出	横断中	その他	大人	○
14726	検出	横断中	その他	子供	○
14731	検出	横断中	その他	子供	○
15072	未検出	横断中	その他	子供	○
15452	検出	背面通行中	車道	大人	○
16107	未検出	横断中	その他	大人	○
16590	未検出	横断中	横断歩道	大人	○
17391	検出	横断中	横断歩道	大人	○
17641	未検出	横断中	その他	大人	○

ID	結果	対象の動作	場所	種類	手動
18194	検出	横断中	その他	子供	○
18416	未検出	横断中	その他	大人	×
18424	検出	横断中	横断歩道	大人	○
18528	検出	横断中	その他	大人	○
20689	検出	対面通行中	車道	子供	○
20789	検出	横断中	その他	大人	○
21391	検出	横断中	その他	大人	○
21397	検出	横断中	その他	大人	○
21478	検出	横断中	その他	大人	○

手動で検出を行った結果から、エッジが消失する問題や、荷物や背景のエッジによる影響以外で、検出のスコアが低くなる原因を調査した。図 3.6 より、検出対象の足が車のボンネットに隠れてしまうことで、検出のスコアが低くなり、歩行者として検出されないことを確認した。また、横断中の歩行者は図 3.7 のように多様な姿勢をとるため、MRCoHOG の様な、画像全体から勾配共起分布を計算する手法では姿勢変化が大きなシーンでの検出率が低下する事が分かった。この結果から、姿勢変動に頑健なモデルである Deformable Part Model(DPM)[7]や、DPM の構造を Deep Learning に応用した Joint Deep[8]や Deep Parts[9]を用いた検出を行う事で、検出率の向上が期待できる。



図 3.6 : ボンネットによる検出対象の足の隠れ



図 3.7 : 交差点における歩行者の姿勢変化

ドライブレコーダ記録映像より、MRCoHOG を用いた歩行者検出実験を行い、未検出である対象に対して解析を行った。表 3.6,3.7,3.8 に、先行研究と本研究を合わせた検出に影響を与える撮影環境をまとめた。表 3.6 より、エッジが消失する場合や、逆に荷物や背景のエッジがノイズになる場合で未検出が発生することが分かった。さらに、ボンネットによる足の隠れや横断中の姿勢の変化が大きく、検出のスコアが低くなる事により未検出となる場合があった。このことから、今後、姿勢変動に頑健なモデルを用いた歩行者検出を行う事でさらなる検出精度向上が期待できる。

表 3.6 : 歩行者の検出に悪影響を及ぼす原因(1)



種類	詳細	画像 (上:オリジナル画像, 下:エッジ画像)
エッジ消失	逆光や影による消失	
	背景と同化することによる消失	

表 3.7 : 歩行者の検出に悪影響を及ぼす原因(2)



種類	詳細	画像 (上:オリジナル画像, 下:エッジ画像)
ノイズ	雨粒によるエッジのぼけ	
	荷物や服装によるエッジの形状変化	

表 3.8 : 歩行者の検出に悪影響を及ぼす原因(3)

種類	画像 (上:オリジナル画像, 下:エッジ画像)
種類 隠れ	
姿勢の変化	



### 3.4.3 Deep Learning を用いた歩行者向き推定

実験には、Network In Network(NIN)と、MRCoHOG による向き推定[4]を用い、比較を行う。データセットには Daimular Pedestrian Dataset から評価用の画像 4000 枚を用い、再現率と適合率の調和平均である F 値によって評価する。さらに、定性評価を行い、NIN を用いた手法の性能を評価した。結果は以下ようになった。

表 3.9：歩行者姿勢推定結果

	F-measure
Previous method	0.73
NIN	0.80

表 3.10：各方向の適合率と再現率

	Front	Back	Left	Right
Precision	0.997	0.994	0.582	0.694
Recall	0.991	0.996	0.843	0.374

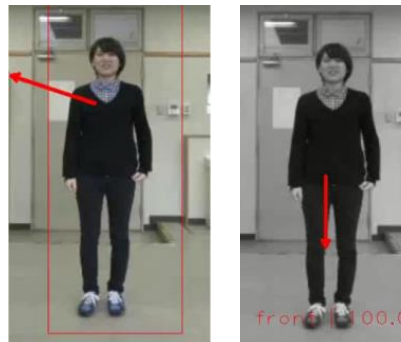


図 3.8：定性評価実験結果  
(左：従来手法，右：NIN)

表 3.9 より、従来手法と比較し、NIN を用いた手法の有効性を確認した。これは、NIN が従来の画像特徴量と比較して、高い特徴表現能力があるためであると考えられる。表 3.10 より、NIN を用いた手法は、前後の識別に特に性能を発揮することが分かった一方、左右の識別に弱いことが判明した。これは、左右の移動の場合、姿勢変動が大きいため、画像全体から特徴記述を行う現在のモデルで左右の識別の精度が低くなったことが考えられる。以上より、姿勢変動の大きな左右の移動の識別には、姿勢変化に頑健なモデルを用いることでさらなる識別精度の向上が期待できる。

## 参 考 文 献

- [1] S. Iwata and S. Enokida: "Object Detection Based on Multiresolution CoHOG", ISVC pp.427-437, 2014.
- [2] N. Dalal and B. Triggs: "Histograms of Oriented Gradients for Human Detection", Proc. IEEE Computer Vision and Pattern Recognition, vol.1, pp.886-893 (2005).
- [3] T. Watanabe, S. Ito, and K. Yokoi: "Co-occurrence Histograms of Oriented Gradients for Human Detection", Proc. Pasific-Rim Symposium on Image and Video Technology, pp.37-47 (2009).
- [4] 浦川 楓: "二値分類器の組み合わせによる歩行者向き推定に関する研究" 九州工業大学卒業論文, 2016
- [5] Min Lin, Qiang Chen, Shuicheng Yan: "Network In Network" arXiv preprint arXiv:1312.4400, 2013.
- [6] 大塚, 榎田: "ドライブレコーダー記録映像における MRCoHOG による人物検出" 智能メカトロニクスワークショップ, 2015
- [7] P. Felzenszwalb, D. Mcallester, and D. Ramanan, "A Discriminatively Trained , Multiscale , Deformable Part Model," in IEEE Conference on Computer Vision and Pattern Recognition, 2008.
- [8] (8)W. Ouyang and X. Wang: "Joint Deep Learning for Pedestrian Detection" In ICCV, 2013.
- [9] Yonglong Tian, Ping Luo, Xiaogang Wang, Xiaoou Tang: "Deep Learning Strong Parts for Pedestrian Detection" The IEEE International Conference on Computer Vision ICCV, 2015, pp. 1904-1912.
- [10] 大塚, 榎田, MRCoHOG と Deep Learning を併用したドライブレコーダー記録映像における歩行者検出, 自動車技術会秋季大会, 154(運転支援 V(ドライブレコーダ))-274, 2016.

## 第4章 危険度推定システムについて

### 4.1 はじめに

本報告書では、第2章、および第3章にて、車両の検出、歩行者の検出の自動化について述べた。本章では、まず、それぞれの実験結果を総括する。その後、本研究、および平成27年度の助成事業により開発された画像処理技術を活用し、今後、構築されることが期待される、運転マナーの発見に関する危険度推定システムの構成を示す。

### 4.2 車両検出に関する実験結果について

第2章で示した通り、本研究により、従来のDPMによる車両検出の精度を向上させることができ、ドライブレコーダ記録映像に対する処理についても、定量評価より効果を確認した。結果、夜間の車両検出、ドライブレコーダが斜めに取付けられたことによる画像の回転等にも顔減に対応することが可能となった。また、検出精度が向上した車両の姿勢について確認すると、先行車両が進行方向を同じくする時から、少し斜めになる時までの検出精度の向上が顕著であった。これは、同一車線上の先行車両検出がより高精度になったばかりでなく、先行車が右左折して向きを変えていく過程も高精度に検出が可能となったことを示している。このことにより、先行車両が右左折中に危険な状況（例えば、自転車の急接近）に気付き急停車するような場面での検出が頑健になったことを確認した。以上より、本車両検出アルゴリズムにより、さらに高精度なドライブレコーダ記録映像自動識別システムの構築が可能であると期待される。今後の課題としては、更なる計算精度の向上と、計算時間の削減が挙げられる。

### 4.3 歩行者検出に関する実験結果について

本研究にて開発しているMRCoHOGに基づく歩行者検出について、ドライブレコーダ記録映像への適用結果をまとめた。結果、INRIAデータセットなどで事前評価した性能がお概ね得られることを確認した。検出精度の低下の原因は、ドライブレコーダに備えられたカメラの性能によるところが大きく、特に、ダイナミックレンジが狭いために輝度情報がつぶれており、検出を困難としていた。それ以外にも、車両との接触を避けるため、歩行者が通常の歩行時とは異なる姿勢をしていることも多く、これらの高精度な検出には学習用画像の充実等、更なる作業が必要であることを確認した。

### 4.4 非優先道からの進入判別に関する実験結果について

平成27年度の報告により、一時停止標識の検出、もしくは、一時停止が必要な交差点の検出を深層学習により99%程度の精度で自動識別可能であることを示している。本研究においてもこれらの開発実績は十分活用すべきであると考えている。

### 4.5 危険度推定システムの構築

#### 4.5.1 開発済み画像処理モジュールの活用に関する展望

平成27年度より、ドライブレコーダの記録映像解析モジュール群を開発してきた。歩行者検出等それぞれのモジュールについては、精度が90%を上回ることを確認し、今後、これらのモジュールを組み合わせることで高精度な危険度推定システムが構築されることが期待される。具体的には、運転手が歩行者のそばを走行するとき、どの程度リスクテイクをしているのか、もしくは、一時停止が必要な交差点にて十分な安全確認がなされているのか等、運行管理者側の要望に合わせて安全システムを構築する準備が完了したと言える。今後、詳細な危険度の推定については、本画像解析結果とドライブレコーダに記録される速度、加速度情報等との統合によって実現されると考えられる。

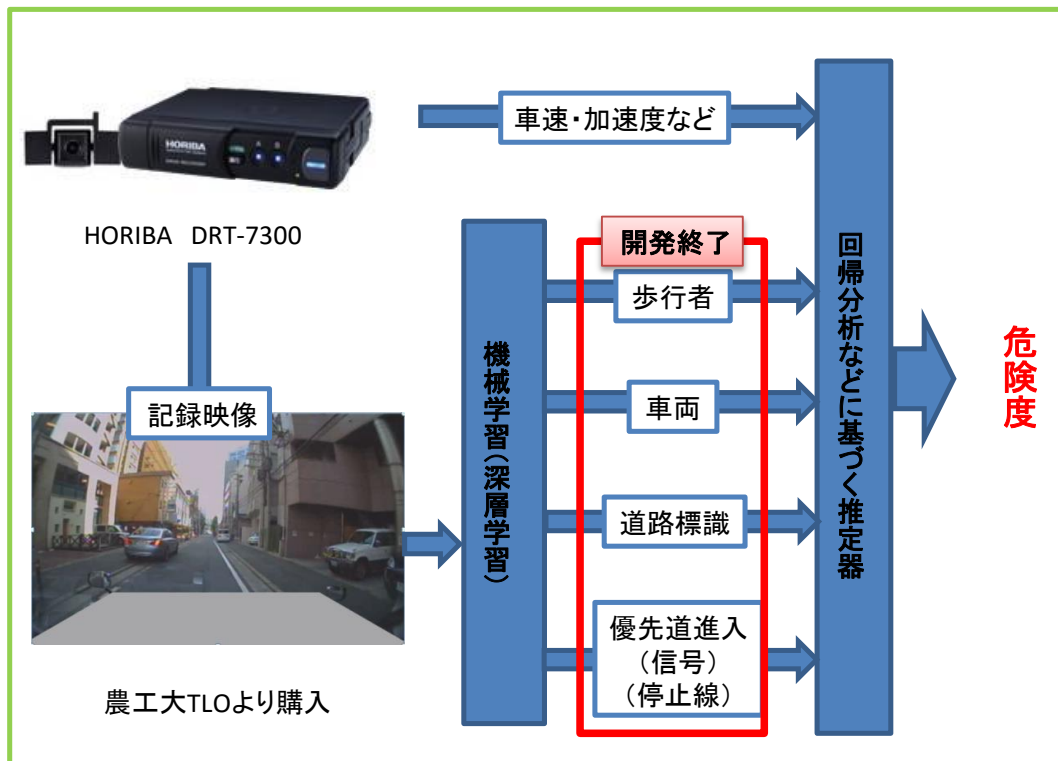


図 4.1 : 開発済み画像処理モジュールを活用した危険度推定システムの構成例

#### 4.5.2 深層学習による危険度の直接推定

ドライブレコーダ記録映像から深層学習により直接危険度を推定する基礎実験を行う。

危険度推定までの流れを図 4.2 に示す。まず、ドライブレコーダから取得した学習画像を入力し、CNN の学習を行う。CNN の学習においては、ワンショット画像を用いて行い、画像の持つ時系列情報は考慮しない学習によって CNN 学習済みモデルを構築する。次に、連続的な情報を持つ画像列である時系列学習画像を CNN 学習済みモデルへと入力し、時系列情報を持った特徴ベクトルを作成する。作成した特徴ベクトルを LSTM への入力として学習を行い、LSTM 学習済みモデルを構築する。最後に、LSTM 学習済みモデルを用いて危険度推定を行う。

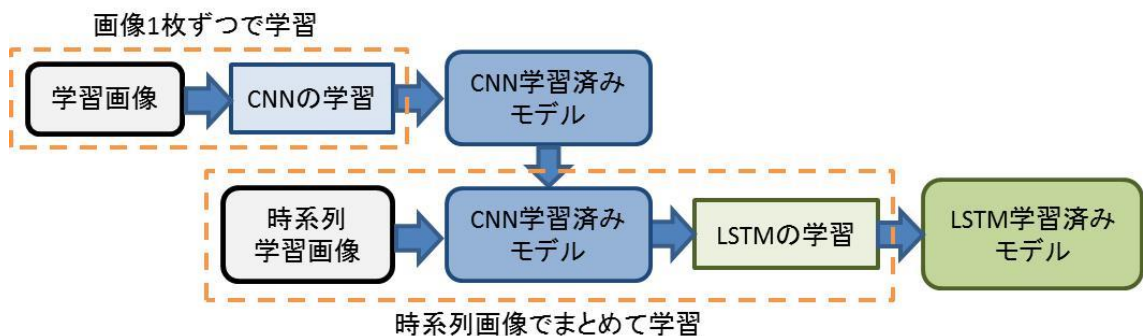


図 4.2 : 危険度推定の流れ

本研究では、危険度推定に用いる CNN として Network in Network (NIN) を用いた。従来の CNN において、一般に畳み込み層では線形分離可能なフィルタを用いる。一方 NIN では、図 4.3 に示すように小規模な多層パーセプトロン (MLP) を含むフィルタを用いる。これにより、従来の CNN よりも複雑な表現が可能となる。図 4.4 に本研究で用いた NIN の構造を示す。本研究で用いた NIN は、4 つの MLPConvolution 層と Max pooling 層、1 つの Softmax 層から構成される。また、活性化関数は ReLU 関数、プーリングサイズは  $3 \times 3$  である。パラメータの学習には、Stochastic Gradient Descent (SGD) に慣性項を追加した MomentumSGD が用いられている。

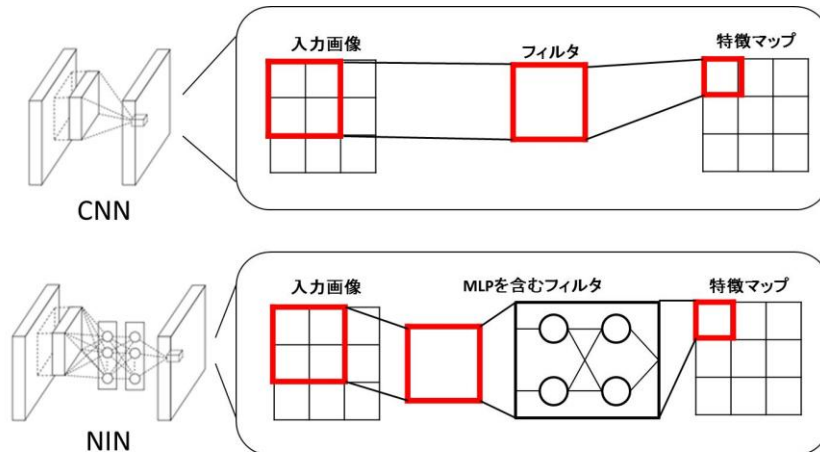


図 4.3 : CNN と NIN における畳み込み層の構造  
 (Min Lin, Qiang Chen, Shuicheng Yan: “Network In Network”  
 arXiv preprint arXiv:1312.4400, 2013. より出典)

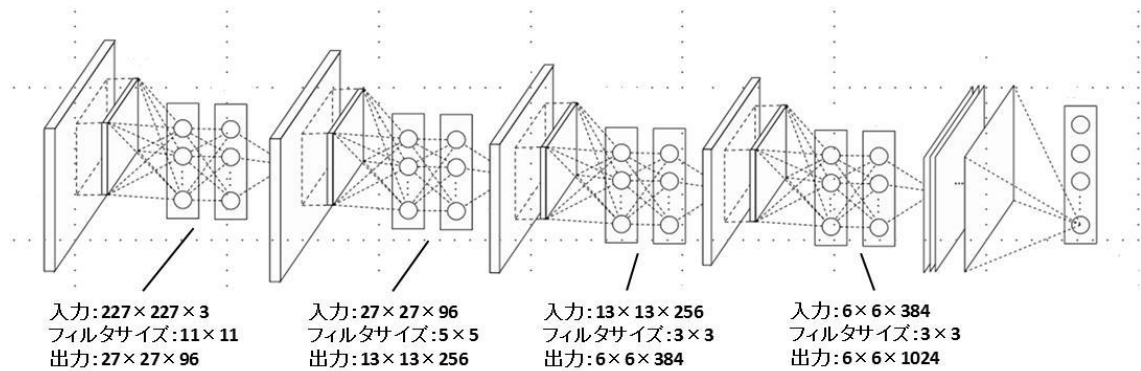


図 4.4 : NIN のネットワーク構造  
 (Min Lin, Qiang Chen, Shuicheng Yan: “Network In Network”  
 arXiv preprint arXiv:1312.4400, 2013. より出典)

図 4.5 に本研究で用いた LSTM の構造を示す。CNN によって画像から変換された特徴ベクトルを LSTM Block への入力としている。入力された特徴ベクトルは LSTM Block 内部のループ構造によって学習された後に全結合層へと接続され、最終層で分類クラスと同数のユニットへと展開後、Softmax 関数により確率へと変換され出力される。パラメータの更新には、Adaptive Moment Estimation(Adam)が用いられており、ニューラルネットワークの汎化性能を向上させるため、一部のユニットのみを用いて学習を行うドロップアウトが適用されている。

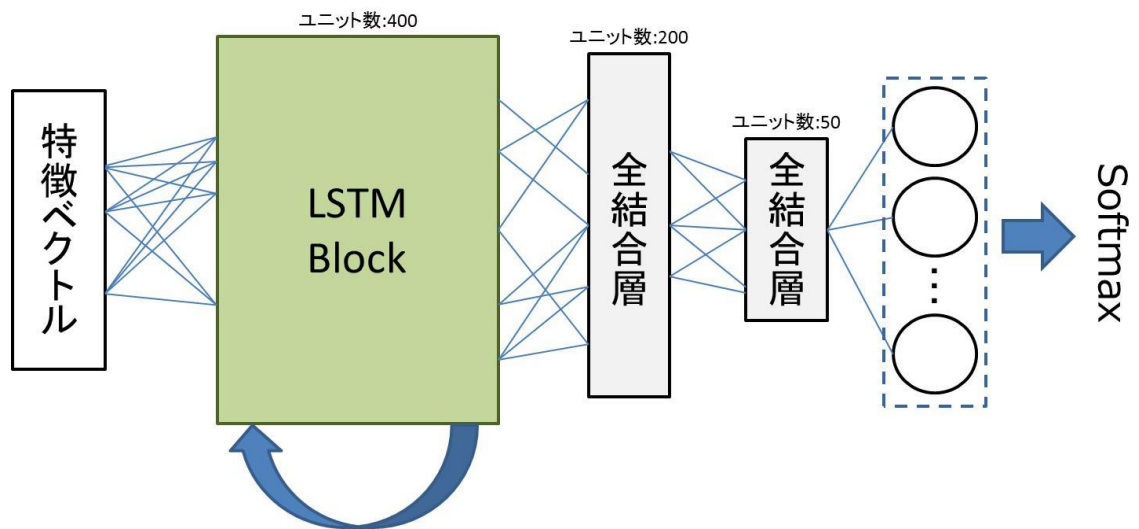


図 4.5 : LSTM の構造

本研究で学習と評価に使用したドライブレコーダ画像は、道路環境全体が写った画像である。しかし、道路環境には左側部分には歩道の道路環境や道路標識等が映り込みやすく、右側部分は対向車線の情報が映り込みやすい、正面には前方の車両や信号が映り込みやすい等、部分領域ごとの特徴がある。そこで今回は、学習に使用した歩行者ドライブレコーダ画像を事前に部分領域へと分割し、分割した各画像に対してネットワークを構築し、学習を行った。概要を図 4.6 に示す。 $n$  分割した画像に対して  $n$  個のネットワークを使用し、各部分領域において学習した NIN 学習済みモデルを構築する。構築した NIN 学習済みモデルを用いて、各部分領域ごとの画像に対する特徴ベクトルを作成し、これらを合わせて LSTM への入力として学習を行う。これによって、歩行者ドライブレコーダの特徴を事前知識として NIN に与えることが可能となり、特徴量抽出過程における処理コストの軽減や精度向上につながる。

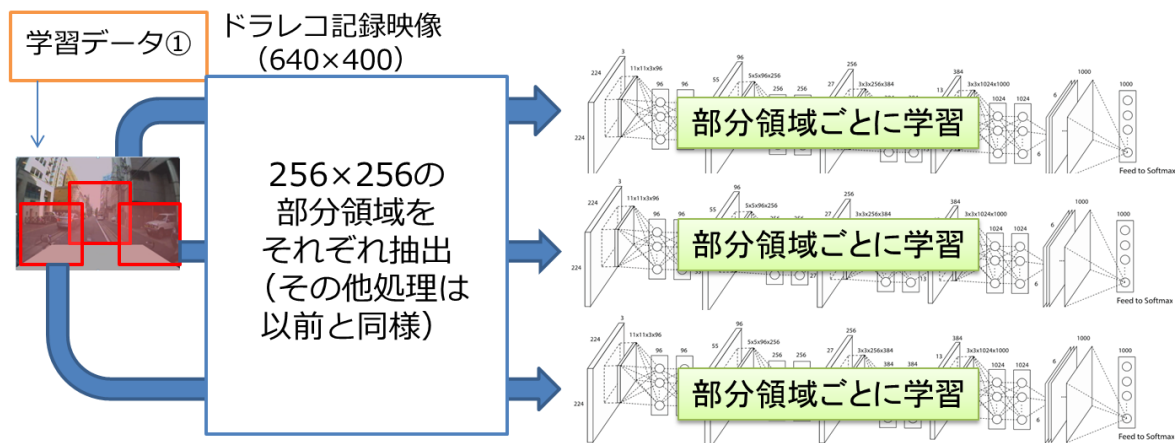


図 4.6 : 部分領域に着目した深層学習の概要

## 【危険度推定実験】

本実験では、ドライブレコーダ動画からフレーム画像を抜き出し、2状態に分類した時系列ドライブレコーダ画像データセットを作成した。危険性度の高さから分類した2状態を図4.7に示す。図4.2に示すように時系列ドライブレコーダ画像データセットは、急制動のトリガが発生する5秒前から3秒前の2秒間20フレームを危険度小とし、トリガが発生した1秒後から3秒後の2秒間20フレームを危険度大として時系列画像を作成した。抜き出したフレームに対して図4.8に示すData Augmentationを行い、1つのシーンに対して120枚の時系列画像を作成した。解像度は640×400pixelである。作成した歩行者の画像データセット計10,560枚を、学習画像10,000枚、評価画像560枚に分割した。LSTMの学習に用いられるドロップアウトの確率は50%とした。学習画像の入力方法は、部分領域に分割する手法と分割を行わない手法を比較した。

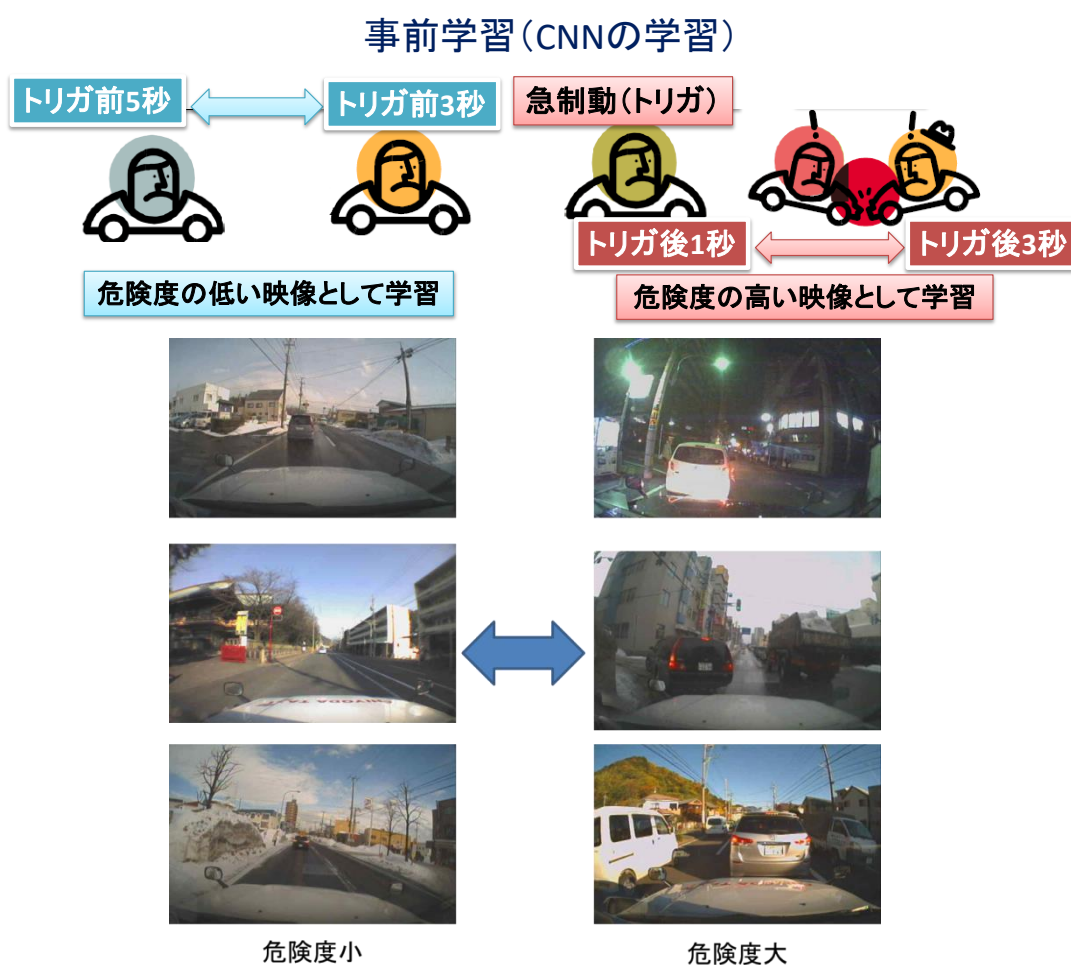


図 4.7 : ドライブレコーダ記録映像からの学習用危険映像切り出し



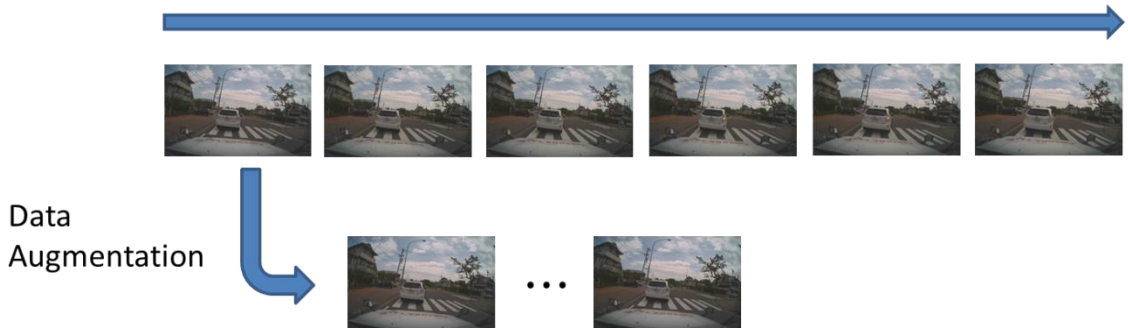


図 4.8 : 作成した時系列画像データセット

### 【実験結果】

危険度推定実験の結果を図 4.9 に示す. 図 4.9 の横軸は学習回数, 縦軸は正答率であり, 入力画像を部分領域に分割した手法を分割あり, 分割を行わずにリサイズのみを行った手法を分割なしとしている. 学習回数が 200 回を超えた辺りから正答率の上昇が収束しており, 分割ありの手法は最大で 71%, 分割なしの手法は最大で 61%となった. 実験結果より, 入力画像を部分領域に分割することで精度が向上することを確認した.

以上の結果から, 深層学習によるドライブレコーダ記録映像からの危険度直接推定の可能性が確認された. ただし, データの数が少なく, 更なるデータの収集, ラベル付けが必要であり, 非常に大きなコストが必要であることも見込まれる.

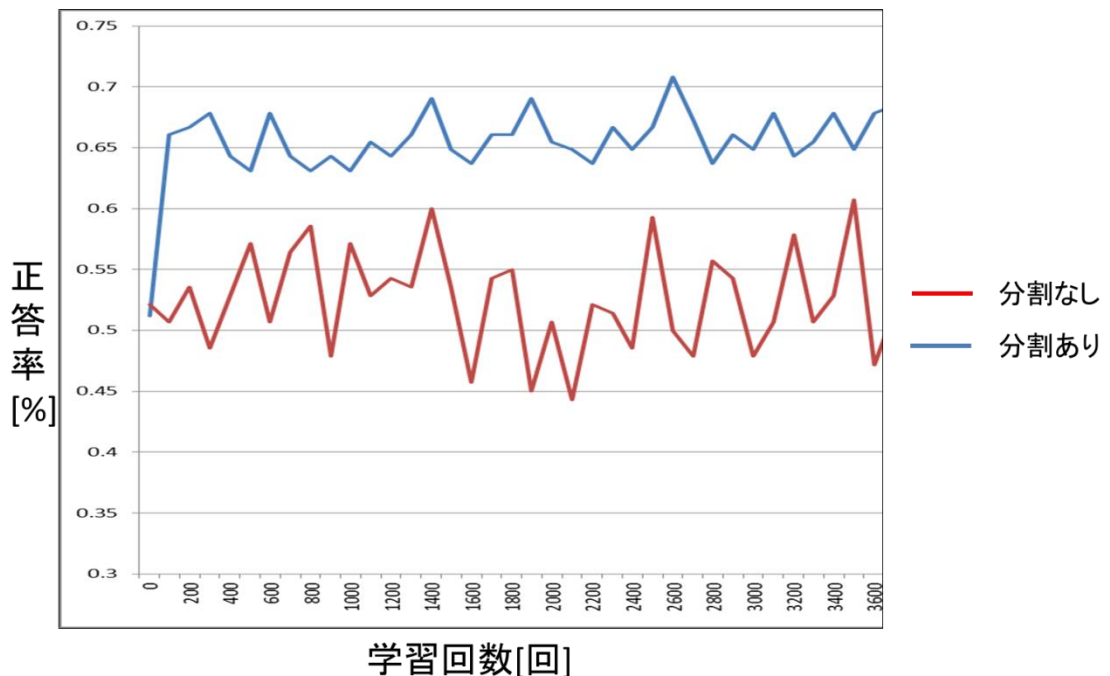


図 4.9 : LSTM を用いた危険度推定結果

## 第5章 結言

本報告書においては、ドライブレコーダ記録映像を処理することにより、従来の加速度トリガでは検出が困難な危険な場面（例えば一時停止不履行）や危険な運転マナー（歩行者のそばを走行する際のリスクテイク）等を検出することを目的とした。本研究により、ドライブレコーダ記録映像からの歩行者の検出、車両（特に先行車両）の検出、一時停止標識、および一時停止が義務付けられた交差点の検出が、画像処理により実現可能であることが示された。これらの画像処理モジュールは、検出性能が90%を超えており、最終的に構築される危険度推定システムの性能向上に大きく寄与することが期待される。また、本研究では、目的毎の画像処理モジュールを必要とせず、深層学習により映像から危険度へと直接変換可能であるかも確認した。結果、本研究に用いたドライブレコーダ記録映像データの規模が小さいため、深層学習が上手く機能せず高精度な結果は得られなかった。ただし、今後は、大規模なデータをそろえることで、煩雑なモジュール構成を必要とせず、深層学習器一つで目的とする危険度推定システムが構築される可能性もある。より大規模なデータを収集、整理して再度実験を行う必要があると考えられる。

以上の研究開発結果を受け、以後、ドラレコ製造メーカ、運送事業者と更なる緊密な連携をとり、1) 画像モジュールによる組み合わせでの危険度推定システムの試作、2) 大規模映像データの整理および深層学習の活用による危険度推定システムの試作を行うこととなった。

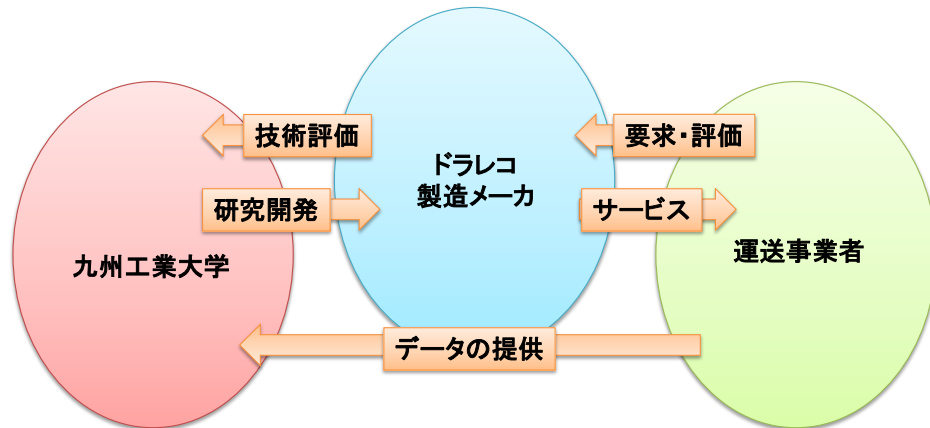


図 5.1 : 今後の研究体制の概要

## 付録 再帰型深層学習(Long Short Term Memory)

本付録では、画像認識分野に活用されている深層学習の技術について説明する。A.1 節では、画像認識において高い性能を達成したことで注目されている畳み込みニューラルネットワークについて説明する。A.2 節では、再帰型深層学習の一種で時系列情報を扱うことのできる再帰型ニューラルネットワークについて説明する。A.3 節では、再帰型ニューラルネットワークを拡張した長短期記憶ユニットについて説明する。

### A.1 畳み込みニューラルネットワーク

畳み込みニューラルネットワーク(CNN)とは、Fukushima らが提案したネオコグニトロンが基礎となっている。ネオコグニトロンの構造は、S 層(単純細胞層)と C 層(複素細胞層)を組み合わせた 2 層の神経回路を基本要素とする。S 層が入力に対して特徴的なパターンを検出する役割を担い、C 層が受容野内(S 層の小領域)にある最大値を出力する。CNN の構造は、畳み込み層とプーリング層から構成される。ネオコグニトロンにおける S 層を畳み込み層、C 層をプーリング層に置き換え、2 つの層を交互に接続した構造をしている。入力画像が与えられた際、畳み込み層(C1,C2)とプーリング層(P1,P2)で入力画像の特徴を取得し、全結合層(F)に取得した特徴を伝え、クラス識別を行う。CNN の構造を図 A.1 に示す。

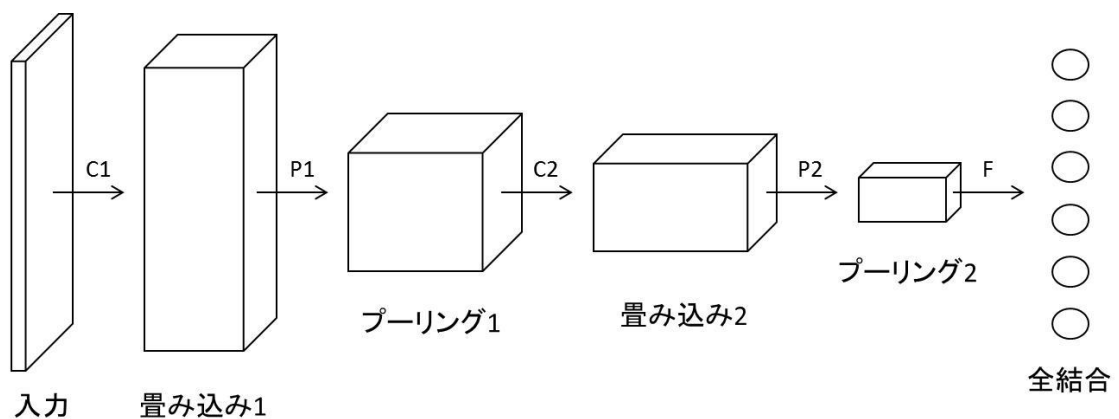


図 A.1 CNN の構造(文献 [A1] より出典)

画像認識分野における CNN において、畳み込み層は入力画像に対してフィルタを畳み込む。フィルタを畳み込む処理は、画像をぼかす処理や、エッジを強調させる処理と同様である。畳み込み層におけるフィルタの値とバイアス値は、学習によって最適化される。フィルタサイズは、ネットワークの設計時に設定され、学習によって変化しない。畳み込み層での処理の概要を図 A.2 に示す。

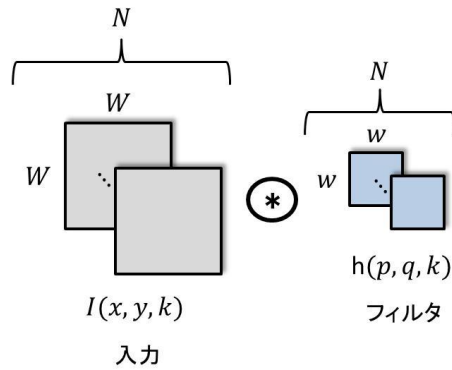


図 A.2 畳み込み処理の概要(文献 [A1] より出典)

入力を  $W \times W$  サイズの画像  $N$  枚とする。この入力を以下では  $W \times W \times N$  とする。入力層では、入力画像がグレースケールの場合  $N=1$  となり、カラー画像の場合 RGB より  $N=3$  となる。これ以降の中間層の数は直前の畳み込み層の出力(フィルタの総数)と同じである。ここで具体的な畳み込み処理を考える。入力  $W \times W$  pixel 解像度の画像  $N$  枚に  $w \times w$  サイズのフィルタを畳み込む。ここでフィルタを  $h(p, q, k)$  ( $p \in [1, \dots, w], q \in [1, \dots, w], k \in [1, \dots, N]$ ) とし、入力を  $I(x, y, k)$  ( $x \in [1, \dots, W], y \in [1, \dots, W], k \in [1, \dots, N]$ )、出力を  $f(x, y, i)$  とすると、 $i$  番目の出力結果は式 A.1 で示される。係数  $b_i$  はバイアスである。

$$f(x, y, i) = \sum_{k=1}^N \left[ \sum_{p,q=1}^w I(x+p, y+q, k) h(p, q, k) \right] + b_i \quad (\text{A.1})$$

プーリング層は畳み込み層と交互に接続されるように配置されるため、畳み込み層からの出力がプーリング層への入力となる。プーリング層では、入力された画像を粗くサンプリングすることで、画像内に現れる特徴の位置や回転等の微小なずれに不変な特徴を得ることを目的とする。プーリング層での計算内容はネットワーク設計時に決定されており、学習によってパラメータが変化することはない。通常、プーリング処理は画像が持つ情報を間引く処理である。間引く量をスライド幅  $s$  とすると、 $s = 2$  の場合、プーリング層からの出力は入力サイズの半分となる。プーリング層では、入力の一部の小領域  $P_{ij}$  に対して、この小領域内部のユニット  $(p, q) \in P_{ij}$  の出力  $y_{pq}$  を集約し 1 つの出力とする。プーリングのアルゴリズムは複数ある。各プーリングアルゴリズムを図 A.3 に示す。本研究では最大プーリング (max pooling) を用いる。最大プーリングは、式 A.2 に示すように  $P_{ij}$  に属するユニットの最大値を出力する。

$$\tilde{y}_{ijk} = \max_{(p,q \in P_{ij})} y_{pqk} \quad (\text{A.2})$$

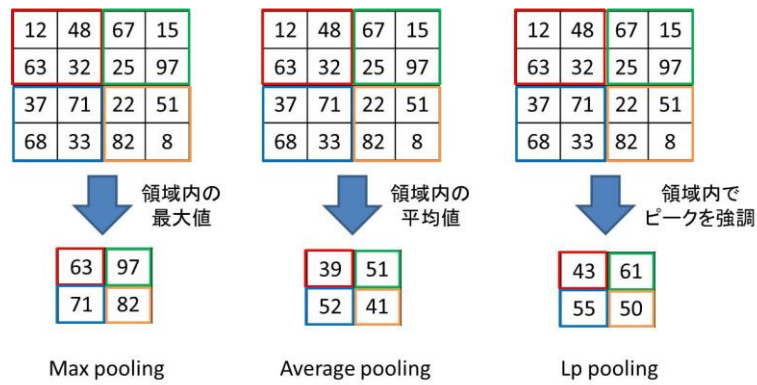


図 A.3 プーリングアルゴリズム

## A.2 再帰型深層学習

再帰型深層学習とは、動画画像や音声などの時系列データを扱うことができる手法である。再帰型深層学習の一種として再帰型ニューラルネットワーク(Recurrent Neural Network:RNN)がある。RNN は画像認識においても応用されており、ネットワークに入力された画像がどのような画像であるかを文章で説明する手法も提案されている。一般的にニューラルネットワーク(NN)は、ワンショット画像を対象に学習を行っており、時系列データに含まれる時系列情報を考慮した学習は行えない。図 A.4 に RNN の概要を示す。RNN は入力層、中間層、出力層から構成されている。入力層への入力を $x$ 、中間層からの出力を $h$ とする。RNNの入力層には、動画画像や音声などの時系列データ $x_t$ が、時刻  $t = 1$  から入力される。時刻 $t$ における RNN の中間層には、時刻 $t$ の入力層の応答値 $x_t$ と、時刻 $t - 1$ の中間層からの応答値 $h_{t-1}$ が入力される。このように、中間層がループ構造を持つことで時系列に対応した学習が可能となる。

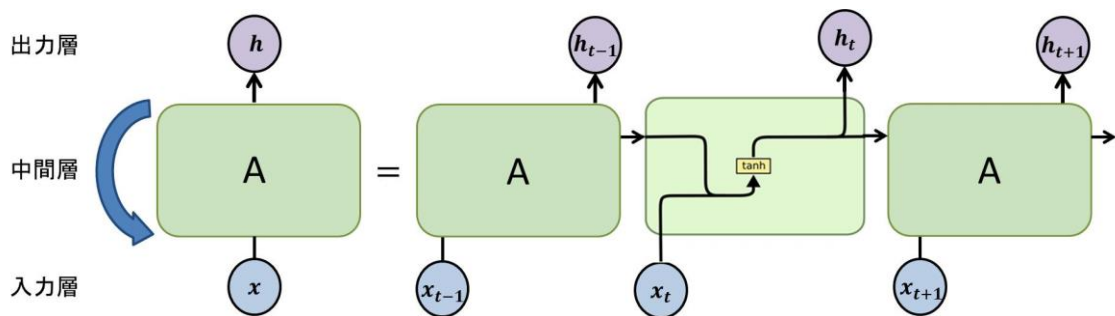


図 A.4 RNN の概要図(文献 [A2] より出典)

### A.3 長短期記憶ユニット(LSTM)

長短期記憶ユニット(LSTM)とは, RNN を拡張した時系列ディープラーニングの一種である. 図 A.5 に LSTM の概要を示す. LSTM は RNN と同様に, 動画像や音声などの時系列データに対して 1 時刻前までの入力に対する内部状態を保持しながら, 現在の入力に対応した出力を行う. RNN は, 学習の際に重みが何度も掛けられることにより学習が正しく行えなくなる勾配消失問題 [7] により, 長期的な時系列情報を必要とする学習を行えない欠点がある. LSTM では, RNN における中間層を LSTM Block と呼ばれる, メモリと 3 つのゲートを持つブロックに置き換えることで勾配消失問題に対応し, 長期的な時系列情報を考慮した学習が行える. 3 つのゲートは, 忘却ゲート, 入力ゲート, 出力ゲートである. 図 A.6 に LSTM Block 内部の 3 つのゲートを示す.

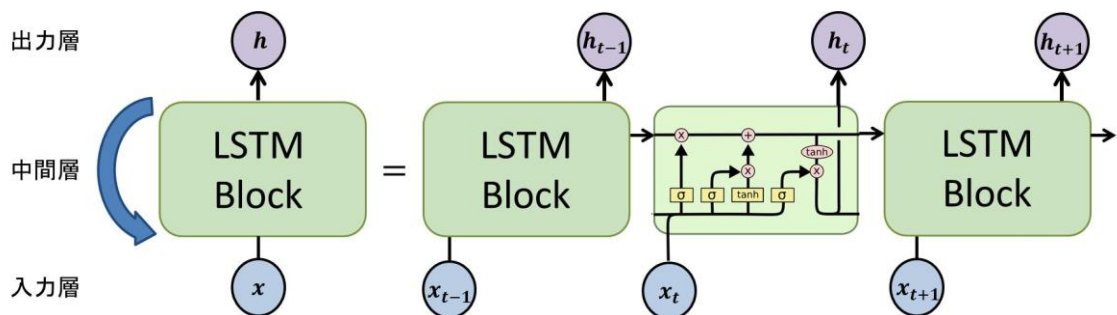


図 A.5 LSTM の概要図(文献 [A2] より出典)

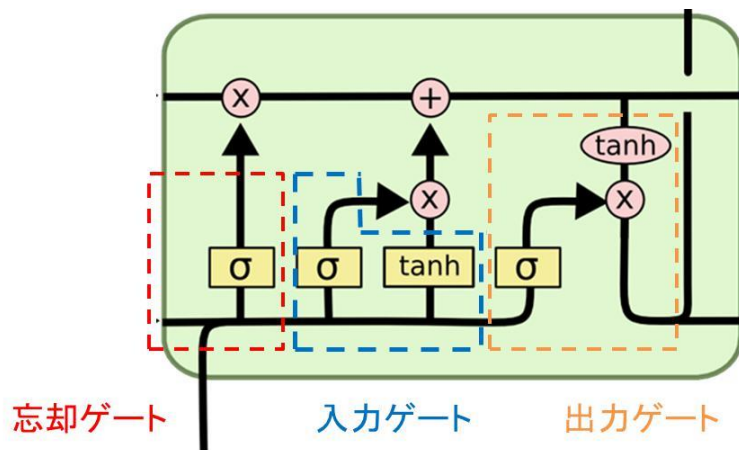


図 A.6 LSTM Block の内部構造

忘却ゲートでは、1 時刻前の内部状態からどの程度の情報を現時刻の内部状態の推定に利用するかを判断する。忘却ゲートの構造を図 A.7 に示す。RNN と同様に  $x_t$  は時刻  $t$  の入力、 $h_t$  は 1 時刻前の出力である。 $f_t$  は時刻  $t$  における忘却ゲートからの出力を表す。式 A.3 に  $f_t$  の式を示す。

$$f_t = \sigma(W_f x_t + R_f h_{t-1} + b_f) \quad (\text{A.3})$$

$W_f$ ,  $R_f$  は重み,  $b_f$  はバイアス,  $\sigma$  はシグモイド関数である。図 A.8 にグラフ, 式 A.4 にシグモイド関数を示す。式 A.4 中の  $\alpha$  はゲインと呼ばれ, シグモイド関数の曲線の傾きを変更するパラメータである。の値が大きいほど傾きが急になり, 小さいほど傾きが緩やかになる。シグモイド関数は, どのような入力に対しても必ず 0 から 1 の値を出力する。 $f_t$  は 1 時刻前の内部状態を 0 に近いほど消去し, 1 に近いほど維持する。

$$f_x = \frac{1}{1 + e^{-\alpha x}} \quad (\text{A.4})$$

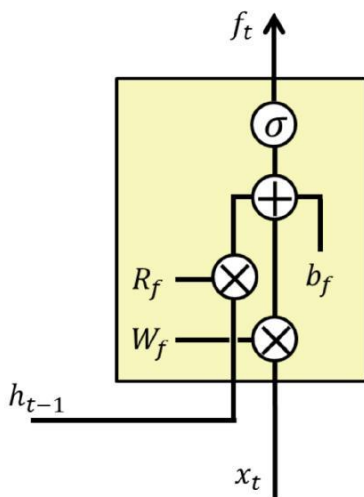


図 A.7 忘却ゲートの構造

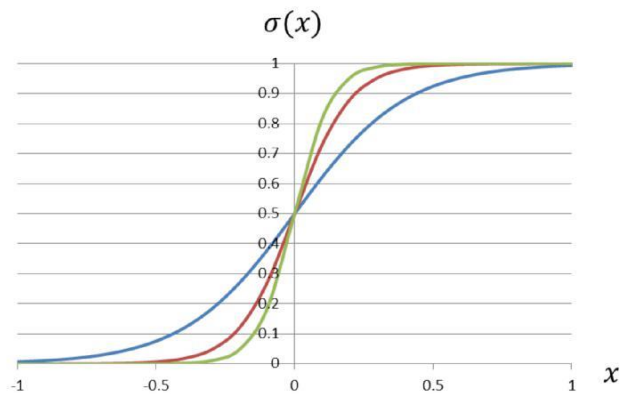


図 A.8 様々なゲインに対するシグモイド関数



入力ゲートでは、現時刻での内部状態にどの情報を加えるかを判断する. 入力ゲートの構造を図 A.9 に示す.  $i_t$  は時刻  $t$  における入力ゲートからの出力,  $\tilde{C}_t$  は時刻  $t$  における内部状態の候補となるベクトルを表す. 式 A.5, A.6 に  $i_t$  と  $\tilde{C}_t$  の式を示す.

$$i_t = \sigma(W_i x_t + R_i h_{t-1}) + b_i \quad (\text{A.5})$$

$$\tilde{C}_t = \tanh(W_c x_t + R_c h_{t-1}) + b_c \quad (\text{A.6})$$

$W_i, R_i, W_c, R_c$  は重み,  $b_i, b_c$  はバイアスである.  $\tanh$  はハイパボリックタンジェント関数と呼ばれ, 式 A.7 で表される. 図 A.10 に  $\tanh$  関数を示す.  $\tanh$  関数は, どのような入力に対して も  $-1 \sim 1$  の値を出力する. 式 A.8 にこれらを用いた内部状態の更新式を示す.  $C_t$  は時刻  $t$  の内部状態,  $C_{t-1}$  は 1 時刻前の内部状態を表す.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (\text{A.7})$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (\text{A.8})$$

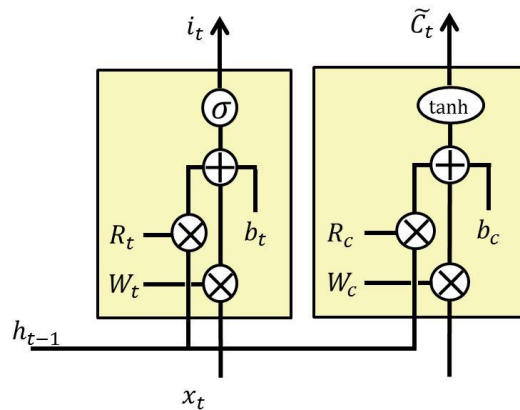


図 A.9 入力ゲートの構造

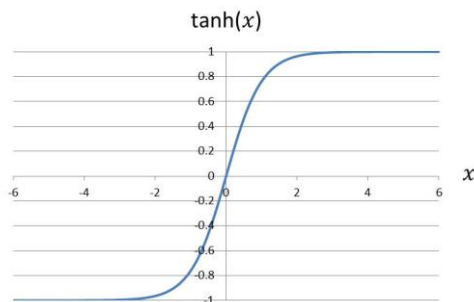


図 A.10 tanh 関数

出力ゲートでは, 入力ゲートで決定した時刻  $t$  の内部状態にフィルタリングを行うことで, 次の時刻での内部状態にどの程度反映させるかを決定する. 出力ゲートの構造を図 A.11 に示す.  $o_t$  は内部状態をどの程度反映させるかを決定するフィルタ,  $h_t$  は時刻  $t$  の出力を表す. 式 A.9 および A.10 に  $o_t$  と  $h_t$  の式を示す.  $W_o, R_o$  は重み,  $b_o$  はバイアスである.

$$o_t = \sigma(W_o x_t + R_o h_{t-1} + b_o) \quad (\text{A.9})$$

$$h_t = o_t * \tanh(C_t) \quad (\text{A.10})$$

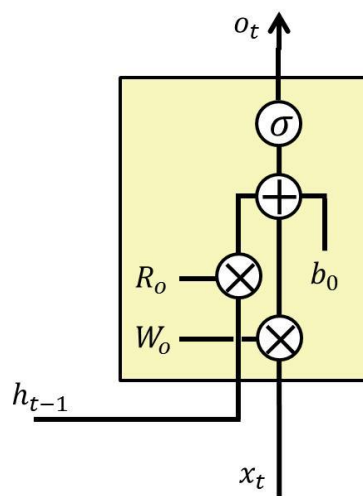


図 A.11 出力ゲートの構造

参考文献

- [A1] 岡谷貫之, "画像認識のための深層学習", 人工知能学会誌 28 巻 6 号, pp.962-974, 2013.
- [A2] Understanding LSTM Networks  
<http://colah.github.io/posts/2015-08-Understanding-LSTM/>