

修士論文

車載カメラとエッジ計算機による運転事故防止技術

Vehicle Accident Prevention Technology  
by Using In-vehicle Camera and Edge Computing

平成 31 年 01 月 31 日提出

指導教員

坂井 修一 教授

東京大学大学院 情報理工学系研究科  
電子情報学専攻

48-166436 谷合 廣紀

# 概要

本研究ではリアルタイム性と消費電力を考慮した運転事故を防止するための車載システムを目指し、その実事例として消費電力を考慮したドライバモニタリングシステムとフォークリフト運転支援のための柵検知システムを提案する。

ドライバモニタリングシステムとは、ドライバの健康状態に起因する事故を防ぐためにドライバが問題なく運転が可能かどうかをモニタリングし、運転不能状態と判断したら警告を出し、場合によってはドライバに代わり車両を安全に停止させるシステムである。

本研究ではドライバモニタリングシステムを RGB カメラと深層学習モデルによって構築した。このシステムでは、常にドライバの姿勢を推定し続けることで、運転不能状態をリアルタイムに検知する。近年の深層学習技術の発展により、ドライバの姿勢を精度よく推定することが可能となった一方、深層学習のような重い計算を常に行うことは電力に制約のある車載システムにとっては大きな負荷となる。

そこで本研究ではドライバが正常運転を維持している間、骨格検知の計算を間引くことによって消費電力の抑制を行う。このシステムは組み込み用計算機上で 7.8fps とリアルタイムに実行でき、安全性を維持しつつ消費電力を 35% 削減できた。

一方、フォークリフト運転支援のための柵検知システムとは、工場内における柵の空き状況を推定し、ドライバに伝えることで運転支援することを目的としたシステムである。工場内で作業するフォークリフトは接触事故をはじめ、転倒や積荷の転落など重大事故につながるものが少なくない。特に柵の高所に積み下ろしを行う際は、フォークリフトのリフト部分に視界が遮られ操作を間違えることで積荷が転落する事故が発生する。そのような事故は、フォークの位置合わせを自動化するなどの運転支援で防ぐことができると考えられる。本研究ではそのようなフォークリフトの運転支援システムの構築を目指し、その第一歩として柵の空き状況を車載カメラによって検知するシステムを提案・実装した。このシステムは組み込み用計算機で 5.7fps とリアルタイムに動作することが確認でき、柵領域に対する IoU は 0.829、空き状況も考慮した IoU は 0.690 となった。

# 目次

第 1 章	序論	5
1.1	本研究の背景	5
1.2	ドライバモニタリングシステム	6
1.3	フォークリフト運転支援のための柵検知システム	7
1.4	本研究の構成	7
第 2 章	関連研究	8
2.1	ドライバモニタリングシステム	8
2.2	フォークリフト運転支援のための柵検知システム	12
第 3 章	ドライバモニタリングシステム	15
3.1	提案手法	15
3.2	実験	20
3.3	評価	24
第 4 章	フォークリフト運転支援のための柵検知システム	28
4.1	柵検知における課題	29
4.2	提案手法	31
4.3	実装	32
4.4	評価	33
第 5 章	結論	36
	参考文献	38

# 目次

2.1	Microsoft COCO [1] を Kanazawa ら [2] の手法で予測した例 . . . . .	11
2.2	Cityscapes [3] を DeepLabV3+ [4] で予測した例 . . . . .	14
3.1	ドライバモニタリングシステムの全体図 . . . . .	16
3.2	ドライバの 3 次元骨格推定 . . . . .	16
3.3	Joint Relation Tensor . . . . .	17
3.4	電力モードの状態遷移図 . . . . .	19
3.5	実車実験 (fig.4.3 [5]) . . . . .	22
3.6	電力評価環境 . . . . .	23
3.7	正常運転誤検知率 . . . . .	26
3.8	異常運転検出数 . . . . .	26
3.9	電力評価 . . . . .	27
4.1	工場内の柵 . . . . .	28
4.2	工場内の柵に対するハフ変換による直線抽出 . . . . .	30
4.3	工場内の柵の積荷 . . . . .	30
4.4	撮影に使用したフォークリフトと GoPro カメラ . . . . .	32
4.5	Intersection over Union . . . . .	33
4.6	予測結果 . . . . .	35



# 表目次

3.1	JetsonTX2 の性能 . . . . .	20
3.2	通常姿勢時の運転環境 . . . . .	21
4.1	評価 . . . . .	33

# 第 1 章

## 序論

### 1.1 本研究の背景

近年、運送業界では慢性的なドライバ不足に陥っており、ドライバの長時間労働が問題となっている [6]. 長時間の運転は健康状態に影響を与え、居眠りを引き起こし結果として深刻な事故につながる可能性がある。そのようなドライバ不足を補う技術として自動運転技術が期待されているが、運転権限を計算機に持たせる完全な自動運転の実現はまだ難しい。そのため現状では、運転権限はドライバが持ちつつ、ドライバ支援システムと連携して運転事故を防止する技術が重要となっている。

そのような事故防止技術の必要性は公道を走る自動車に限らない。物流を支える長距離ドライバ同様、それらの拠点となる倉庫においても事故の危険性がある。特に工場内で作業するフォークリフトは接触事故をはじめ、転倒や転落など重大事故につながる 경우가少なくなく、フォークリフトによる事故は平成 29 年には国内で 30 件もの死亡事故が発生している [7]. フォークリフトの作業においても、コンピュータによる運転支援があれば人為的なミスによる事故を減らすことが可能である。

これらの運転事故防止技術に求められるひとつの性質はリアルタイム性である。公道を走る車や、倉庫内のフォークリフトなどに搭載される事故防止システムは、時々刻々と変化する周囲の環境に対して迅速かつ正確に判断を下す必要がある。そのようなリアルタイム性を実現するためには、計算サーバと通信することは現実的でないため、車載上のシステムで完結していることが望ましい。車載システムでは使える電力に制約があることから、省電力性も同時に求められる。これらからリアルタイムかつ省電力を意識したドライバ支援システムは重要な役割を持つ。そこで本研究では、そのようなリアルタイムかつ省電力な車載可能な事故防止技術の事例として、ドライバモニタリングシステムとフォークリフトの柵検知システムを提案する。

## 1.2 ドライバモニタリングシステム

ドライバモニタリングシステムとは画像センサや生体センサなどを用いてドライバの状態を測定するシステムである。健康の問題によるドライバの意識喪失が起きると、車両はコントロールを失い、結果として深刻な事故につながる。ドライバの体調急変は疲労や眠気とは異なり、ドライバ自身が異常状態を予測するのは困難である。ドライバモニタリングシステムによってドライバの異常を検知できれば、ドライバに休憩を促すように警告を出したり、ドライバに代わり車両を停止するなどの措置をとることができる。

ドライバの体調不良を原因とした交通事故の多くは、運転姿勢が崩れることが報告されている [8]。また、ドライバーに発作や急病が発生したときの調査報告では、その 6 割で運転姿勢の崩れが起きていることが報告されている [9]。これらの報告から、ドライバの異常を検出する方法として、運転姿勢を用いることは有効であることがわかる。

また、ドライバの不注意によっても交通事故は起こりうる。近年、スマートフォンの普及により「ながら運転」が問題になってきている。警察庁の発表によれば、2016 年にはスマートフォンなどの携帯電話を運転中に操作したことを原因に、1999 件の交通事故が発生している [10]。ハンズフリーでの使用を除けば、運転中の携帯電話の操作は、正常な運転姿勢とは異なる姿勢をとる。このことから、ドライバの運転姿勢情報は「ながら運転」の検知にも有効である。

そのため、本研究で提案するドライバモニタリングでは運転姿勢情報を利用する。近年の深層学習技術の発展により、精度よく姿勢推定をできるようになった一方、電力に制約のある車載システムにとって常に深層学習の推論計算を行うことは負担となる。ドライバモニタリングシステムの用途を考えると、運転中におけるドライバの運転姿勢のほとんどは通常姿勢であり、そのような場合、ドライバが異常な姿勢をとっていないことさえ分かれば、姿勢推定の精度や頻度が高い必要はない。そこで本研究では、Approximate Computing の考え方を導入し、ドライバの状態に合わせて適切に計算精度を変化させることで、要求品質と低電力を両立させる。具体的にはドライバが正常な運転姿勢を維持している間は、骨格検知の姿勢を間引くことで消費電力を抑制し、異常姿勢の兆候が見られたら即座に通常の計算頻度に戻し、ドライバをモニタリングする。本手法では、組み込み用計算機上でリアルタイムに動作し、さらに Approximate Computing を行わない場合と比べて、異常姿勢の検出数を維持しつつ最大で電力を 35% 削減できることを確認した。

## 1.3 フォークリフト運転支援のための柵検知システム

フォークリフト運転支援のための柵検知システムとは、工場内における柵の空き状況を推定し、ドライバに伝えることで運転支援することを目的としたシステムである。工場内で作業するフォークリフトの事故は転倒や転落など重大事故につながるものが少なくなく、特に柵の高所に積み下ろしを行う際は、フォークリフトのリフト部分に視界が遮られ操作を間違えることで積荷が転落する事故が発生する。柵の空き状況をシステムが正確に認識していれば、そのような事故を防ぐこともできる。

本研究で提案する柵検知システムはフォークリフトに RGB カメラと小型計算機を載せることで実現する。このシステムでは工場内にあるフォークリフトが積み下ろしできる柵を検知し、さらにその柵が埋まっているかどうかを同時に判定する。本研究では、一連の判定をセマンティックセグメンテーション技術を応用することでワンショットで行うことを提案する。セマンティックセグメンテーションとは自動運転分野にも応用される画像処理のタスクの 1 つであり、画像内のピクセルごとにクラス分類を行う。セマンティックセグメンテーションの特徴として、検出したい対象領域が長方形でなくても過不足なく推定することができるため、空き状況の推定に適している。さらに、本システムをフォークリフトの自動運転に応用する際にも、セマンティックセグメンテーションによる予測は自動運転研究の知見を取り入れやすいという利点がある。

本研究ではフォークリフトに搭載することを意識し、省電力な深層学習モデルを用いることで、組み込み GPU プラットフォームでリアルタイム動作を達成した。また、このシステムはフォークリフトに載せたカメラで撮影した工場の画像について評価を行った。

## 1.4 本研究の構成

本研究の構成は以下の通りである。2 章では本研究に関連する運転事故防止技術や深層学習技術について紹介を行う。3 章では、提案するドライバモニタリングシステムの詳細を述べる。続く 4 章では、提案するフォークリフト柵検知の詳細を述べる。最後に 5 章で結論を述べる。

## 第 2 章

# 関連研究

### 2.1 ドライバモニタリングシステム

ドライバモニタリングシステムとはドライバの状態を推定する手法であり，運転に影響を与え，事故を引き起こす原因となるドライバの眠気や疲労，注意散漫さを検知することを目的としている．これらの既存手法は主に，生体電気信号，運転状態，ドライバの挙動を用いるものに分類することができる [11]．

#### 2.1.1 生体電気信号

生体電気信号とは，脳波をはじめとした生物における発電現象であり，人体に測定器を直接当てることで測定する．脳波は脳の活性状態の指標として広く受け入れられており，眠気の推定に有効である．その正確さから脳波はしばしば眠気の参照基準としても用いられ，OSS(objective sleepiness scoring) [12] と呼ばれる脳波をもとに推定される 5 段階の眠気度合いは，他の眠気推定手法の正解データとして用いられることもある [13]．

脳波以外にも，筋電位や眼電位を使った運転者の状態推定に使う手法が提案されている．Yang ら [14] は脳波と筋電位を用いたベイジアンネットワークで疲労の推定を行い，より多くの信号を用いることでより正確にドライバの疲労を推論することができることを示した．

眼電位は眼球の動きによって変化する．人間は覚醒時と眠いときで，眼球運動に変化が生じることが知られているため，眼電位の測定によってドライバの眠気を推定することができる．Hu ら [15] は，眼電位から 11 個の特徴量を抽出し，サポートベクターマシンによって眠気予測を行っている．

これら生体電気信号は，ドライバの状態を直接的に反映するため精度のよい状態推定を行うことができる一方，測定するための機材はドライバへの侵襲性が高く，運転に影響を与える可能性があることが問題となる [13]．

### 2.1.2 自動車の運転挙動

ドライバの状態の変化は、車の運転状態に影響を与える。Faircloughら[16]は、睡眠不足のドライバはハンドルの切り替えし頻度が減り、車線を超えそうになる頻度があがることを報告している。そのため、ハンドル操作情報は、眠気の指標のひとつとなる。また、アクセルの操作情報や、自動車の速度の情報から、ドライバの状態を推定する手法も提案されている[17–19]。ただし、自動車の運転挙動は路面状況に大きく影響されるため、これらの情報を用いた推定手法は精度がよくないことが指摘されている[20]。

### 2.1.3 ドライバの挙動

画像センサなどによって取得したドライバの挙動をもとにしたドライバの状態推定は、生体信号と異なりドライバへの侵襲性が低いため、運転に影響を与えることが少なく実用上広く受け入れられている手法である。ドライバの挙動をもとにした推定では、ドライバの眼に関する情報やドライバの手の位置、運転姿勢が使われる。

眼の情報を使ってドライバの状態を推定するときに用いられる尺度に、PERCLOS[21]と呼ばれるまぶたの開き具合がある。PERCLOSは目が80%閉じている時間の割合で、もっとも広く受け入れられている眠気検知の指標のひとつである。ただし、PERCLOSは測定に長い時間間隔を要し、さらにまばたきは個人間でばらつきが多いことが問題となっている[13]。

運転姿勢を用いた推定手法では、体調急変や「ながら運転」といった注意散漫な挙動も推定することができる。ながら運転の多くは携帯電話の使用であることから、手の位置を推定することで運転中の携帯電話の使用を検出する手法が提案されている[22,23]。また、機械学習コンペティションプラットフォームであるKaggleでも、StateFarm保険会社によってドライバの姿勢分類タスクが出題されたことがある[24]。このコンペティションで公開されたデータセットは、姿勢分類タスクにおける最初のオープンなデータセットであった。このデータセットはドライバの運転姿勢の画像と、安全運転・右手での携帯への文字入力・左手での携帯への文字入力・右手での携帯の通話・左手での携帯の通話・ラジオの操作・飲み物・後ろに手を伸ばす、化粧や髪いじり、乗客との会話という10のクラスラベルから構成されている。StateFarmデータセットの使用は、Kaggleにおけるコンペティションでの使用に限られていたが、Abouelnagaら[25]が類似の運転姿勢データセットを公開している。これらのデータセットの存在は、収集の難しい実車環境での実験を容易にし、より精度のよい検出手法の発見を助けるという面で重要である。

山田ら[26]はドライバの体調急変による異常を、異常検知の枠組みで推定する手法を

提案している。この手法では、深度センサによってドライバの上半身の3次元骨格推定を行い、それをもとに Joint Relation Tensor(JRT) と呼ばれる相対関係表現に変換する。JRTとは関節の3次元座標を、関節間の  $x, y, z$  軸それぞれの相対関係に変換したテンソルである。この相対関係表現をもとに、通常運転時の姿勢辞書を作り、その姿勢辞書と現在の運転姿勢の乖離を測ることで運転時の異常姿勢を検知している。この手法では、網羅の難しい異常姿勢を定義せずに、通常時の姿勢のみから高い精度で異常を検知できている。しかし、山田らの手法で用いた3次元骨格推定は、推論環境下と同様の状況でラベル付けしたデータを訓練データとしているため、他の深度センサを使った場合や異なる車種など環境の変化に対して汎化性能が低いことが問題であった。

#### 2.1.4 姿勢推定

近年の画像技術の進歩により、RGB画像から人間の3次元の姿勢を精度よく推定することが可能となっている。3次元の骨格推定を行う手法としては、2次元の骨格推定器 [27,28] によって予測された2次元骨格情報をもとに、3次元の骨格位置を回帰したり [29,30]、モデルフィッティングによって求める手法が提案されている [31,32]。ただし、これらの2段階で3次元骨格を推定する手法は、画像から2次元骨格情報への変換で、3次元に推定するための多くの情報が捨てられてしまっていることが指摘されている [2]。

教師あり学習では一般に、入力画像とペアになる正解データが必要となるが、正確な3次元の骨格位置の取得は、実験室環境でモーションキャプチャなどを使わないと難しい。その結果、それらの教師データで訓練されたモデルは室内環境では3次元の姿勢推定をできても、それ以外の環境になるとうまく推定のできない汎用性の低いモデルになってしまう問題があった。Kanazawaらの提案する手法 [2] では、モデルに敵対的モデルを導入することでこの問題を解消している。この手法では2つのモデルを用い、一つのモデルは入力画像から SMPL [33] と呼ばれるパラメタライズされた人間模型のパラメタを予測し、もう一方のモデルはそのパラメタが人間の形状として自然であるかどうかを判別する。このように2つのモデルを敵対させることで、入力画像から合理的な3次元の人間形状を推定するモデルを訓練することができる。図 2.1 に、Microsoft COCO データセット [1] の1枚に対して、訓練済みの Kanazawa らの手法で予測した3次元の人間形状を示す。この図が示すように Kanazawa らの手法は、体の一部分しか写っていない画像を入力としても、もっとも合理的な3Dな人間の姿勢を予測することができる。



図 2.1 Microsoft COCO [1] を Kanazawa ら [2] の手法で予測した例

### 2.1.5 Approximate Computing

Approximate Computing とは計算クオリティと引き換えに計算量を削減する手法である [34]. Approximate Computing は厳密な解を必要としないアプリケーションに利用でき、実装や計算精度を適応的に省くことで性能やエネルギー効率に置き換えることを目的としている。実装方法はソフトウェアによる方法やハードウェアによる方法など多岐にわたる。一例として、Chippa ら [35] はスケーラブルなハードウェアを用いることで、クラスタリングアルゴリズムである k-means 法を、分類精度に 5% の損失を許容することで 50 倍のエネルギー効率を実現している。



## 2.2 フォークリフト運転支援のための棚検知システム

### 2.2.1 パレット検知

フォークリフトの操作は、主に工場内の目的地への移動と荷物の積み下ろしの2つから構成される。特に荷物の積み下ろすときに必要なフォーク部分を挿入するためのパレットの開口部分の位置推定は、フォークリフト操作の根幹をなすことから多くの手法が提案されている。無人搬送車に限らず、人が運転するフォークリフトの支援においても、操作ミスによってパレットや棚を傷つける事故を防止するために、パレットの位置推定は運転支援において重要なタスクである。

パレットの位置推定は主に画像処理の枠組みで行われる。Syuら[36]はフォークリフトに単眼のRGBカメラを搭載し、撮影された画像からHaar-like特徴量によってパレットの候補領域を選出した後、最適化をかけることでパレットの開口部を検出している。また、2Dレーザーや深度カメラによって取得した3次元情報からパレットの開口部を検出する手法も提案されている[37,38]。ただし、このようなパレットの位置推定手法は、パレットに対してフォークリフトが正面に位置している場合に有効であるため、図??左のような画像からパレット位置を推定することは難しい。さらに、工場内ではパレットの上にパレットを重ね置きする場合も多く、パレットの検出だけでは棚の空き状況を完全に検知することはできないという問題もある。

### 2.2.2 商品棚の管理システム

工場内の棚推定の類似のタスクとして商品棚の管理システムがある[39,40]。商品棚の管理システムは、スーパーマーケットにおける在庫管理システムで用いられる。Marderら[39]は物体検出によって、商品棚に陳列してある商品を監視し、店員に棚の状態と商品の補充リストを提供するシステムを提案している。この手法は、商品棚の画像から物体検出によって商品の領域を抽出し、その候補とあらかじめデータベースに登録してある画像を照らし合わせることで、商品棚に陳列されている商品を認識する。この方法は、コンテナなど限られた積荷が工場内の棚に積まれている場合は有効であるが、一般には不特定種類のパレットや積荷が混在するため、データベースの構築は現実的でない。また、フォークリフトは積み下ろしを行うとき以外は棚に対して正面を向いていないため、バウンディングボックスを用いる物体検出の手法は適していない。

### 2.2.3 セマンティックセグメンテーション

セマンティックセグメンテーションとは、画像内のピクセルごとの分類問題であり、画像処理分野の基本的なトピックのひとつである。セマンティックセグメンテーションは、自動運転 [3,4]、衛星画像解析 [41]、医療画像解析 [42] など多岐にわたって応用されている。深層学習技術によるセマンティックセグメンテーションは Fully Convolutional networks(FCNs) [43] を源流に、近年の深層学習技術の発展とともに精度をあげてきた [4,44–46]。図 2.2 に Cityscapes [3] の 1 枚に対して訓練済みの DeepLabV3+ [4] で予測した結果を示す。この図が示すようにセマンティックセグメンテーションは、車が存在しているピクセルは車のクラスに、人のピクセルは人のクラスに、といったようにピクセルごとにクラス分類を行う。

セマンティックセグメンテーションは自動運転技術にも応用されることから、車載を意識した省電力で軽量な深層学習モデルの提案もされている。MobileNetV2 [47] は、深層学習モデルにおける畳み込み層を、空間方向とチャンネル方向に分離することで計算量を減らし、高い精度とリアルタイム性を実現したモデル構造である。

また、深層学習モデルの学習には一般的に大規模な教師データが必要になる。教師データの作成は、物体認識タスクであれば 1 枚の画像に対して「犬」や「猫」などといったクラスラベルを付与するだけだが、セマンティックセグメンテーションの場合、各ピクセルに対してラベル付けを行う必要があり、非常に労力を必要とする。公道を走る自動車の車載カメラのデータセットは、自動運転タスクの応用にとって非常に重要なため、多くのデータセットが存在している [3,48,49] が、他の応用においてはデータセットの用意が問題となる場合が多い。そのような教師データが少ない状況では、データオーグメンテーションや半教師あり学習、弱教師あり学習が有効となる。

データオーグメンテーションとは移動や回転、ノイズ付加といった加工によって学習画像の数を増やす手法である。学習データを人工的に水増しすることにより、深層学習モデルの過学習を防ぐことができるため、多くのネットワークの学習で用いられている。データオーグメンテーションは対象とするタスクによって、有効となる手法が異なり、たとえば DeepLabV3+ [4] では CityScapes データセットの学習に、Horizontal Flip と呼ばれる水平方向の反転と、Random Crop と呼ばれる画像の一部を切り抜く手法によって学習データを増やしている。

また、教師データが少なくてもラベルのついていないデータが大量に使える場合は、半教師あり学習が有効な手法である。Hung ら [50] は、GAN の発想により、ネットワークに敵対的モデルを導入した半教師あり学習を提案している。この手法では、セグメンテーションのネットワークの他に、ディスクリミネーターと呼ばれる、セグメンテーションの

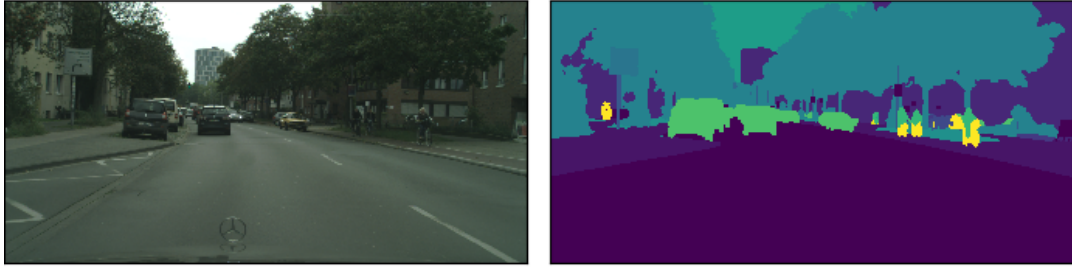


図 2.2 Cityscapes [3] を DeepLabV3+ [4] で予測した例

予測結果と正解データを識別する役割を持つネットワークを用いる。セグメンテーションのネットワークは正解に近い予測を行うだけでなく、ディスクリミネーターに正解データだと騙せるような予測を行うように学習が行われる。一方のディスクリミネーターはそれらを識別できるように学習を行う。

このようにしてネットワークを敵対させて学習させると、両者は拮抗し、セグメンテーションの予測に対してディスクリミネーターを騙すことができる領域が一部発生する。そのような領域はセグメンテーションモデルの確信度が高く、擬似的な正解と見なすことができる。そのため、ラベルのついていない画像を予測させ、さらにディスクリミネーターを騙せた領域を正解データとして用いることで、ラベルなしデータを学習に組み込むことができる。著者らは、この手法を用いることで、ネットワークの汎化性能があがり、評価用データに対する精度が向上することを報告している。

## 第 3 章

# ドライバモニタリングシステム

### 3.1 提案手法

#### 3.1.1 概要

本章では，車載システムを意識した運転姿勢によるドライバモニタリングシステムの詳細を述べる．このシステムの全体図を図 3.1 に示す．本手法では，関節の相対関係表現をもとに，深層学習による 3 次元姿勢推定と Approximate Computing によって高い汎化性能と省電力を実現するドライバモニタリングシステムを提案する．

本手法では，ドライバとその周辺環境はドライバ前方に設置された RGB カメラで計測される．RGB カメラで撮られた画像は Kanazawa らが提案する手法 [2] を通して，3 次元姿勢に変換され，そのうち上半身の関節 9 箇所 (鼻, 右/左肩, 右/左肘, 右/左手首, 右/左尻) の 3 次元座標を取得する．Kanazawa らの手法は，3 次元姿勢をあらゆる環境における人間の画像で学習してあるため，足の部分が写らない運転席であっても，再学習なしに予測することができる．

そのようにして取得した関節の 3 次元座標は山田ら [26] が提案する姿勢の相対関係表現である Joint Relation Tensor(JRT) に変換される．JRT を使うことで姿勢間に距離を導入でき，現在の運転姿勢と正常時の運転姿勢との距離を異常度と見なすことができる．このような異常検知の枠組みで運転姿勢の異常度を計測することは，網羅することの難しい異常運転状態を定義しなくていいというメリットがある．

この姿勢間距離は，瞬間的なノイズへの頑強性や，運転姿勢の時間的連続性のために，直近 3 秒間における平均をとる．この平均が閾値を超えていた場合，システムはドライバの異常運転状態を検知する．これにより画像の乱れなどの瞬間的なノイズへの頑強性や，一時的なドライバの例外的動作に対応できる．

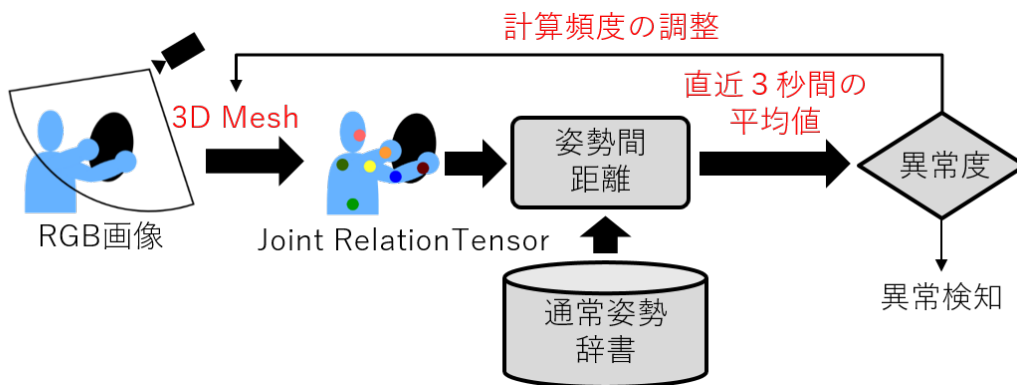


図 3.1 ドライバモニタリングシステムの全体図

### 3.1.2 姿勢推定

ドライバの運転姿勢の取得方法について説明する。本システムにおける運転姿勢推定は、Kanazawa らが提案する手法をもとにしている。この手法は、深度センサなどなしに RGB 画像のみから、3次元の姿勢を推定することができ、体の全体が写っていないでも体全体の自然な姿勢を推定することができる。これはドライバの体の一部分しか写らない、ドライバモニタリングシステム的环境に適した手法である。さらに、このモデルは画像を入力に、いくつかのパラメタを予測するだけの軽量なモデルであり、リアルタイム性の求められるアプリケーションに応用できるという利点がある。

また、Kanazawa らの手法は、モーションキャプチャを使った実験室環境に限らず、あらゆる環境下における人体の画像をモデルの訓練に使うことができるため、汎用性の高いモデルを構築でき、本システムのような運転席環境においても、モデルの再学習なしに推論が可能である。Kanazawa らの手法は、人体の3次元メッシュを予測するが、本手法では JRT による姿勢表現に変換する必要がある。そのため、図 3.2 に示すように、3次元メッシュから上半身の関節9箇所(鼻, 右/左肩, 右/左肘, 右/左手首, 右/左尻)の3次元座標を取得する。

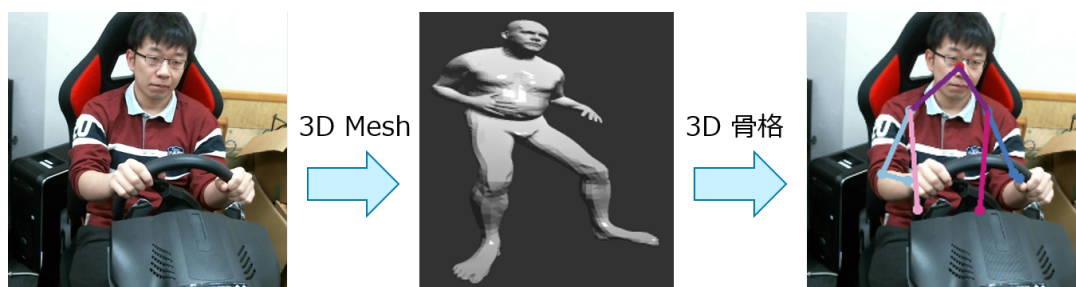


図 3.2 ドライバの3次元骨格推定

### 3.1.3 姿勢表現

姿勢推定によって得られた関節の3次元座標は山田らが提案する Joint Relation Tensor(JRT) と呼ばれる相対関係表現に変換される。JRTとは関節の3次元座標を図3.3に示すような関節間のx, y, z軸それぞれの相対関係<, =, >を表現したテンソルである。本手法では、上半身の関節9箇所を用いるためJRTは図3.3のようになる。

このJRTを用いることで2つの姿勢AとBの間に距離Dを導入できる。すなわち、姿勢AとBの間の距離DをJRTの不一致の数の重み付け和として定義する(3.1)。式中 $a_{ijk}, b_{ijk}$ がそれぞれ姿勢Aと姿勢BのJRTの要素に対応しており、 $i$ と $j$ はそれぞれ関節、 $k$ はx, y, z軸を表す。また、 $w_{ijk}$ は不一致した関節に加えられる重みである。

$$D = \sum_i \sum_{j(j>i)} \sum_k w_{ijk} \cdot \text{Match}(a_{ijk} - b_{ijk}) \tag{3.1}$$

$$\text{Match}(a_{ij} - b_{ij}) = \begin{cases} 0, & a_{ij} = b_{ij} \\ 1, & a_{ij} \neq b_{ij} \end{cases}$$

このように定義された姿勢間距離を用いることで運転姿勢の異常度を考えることができる。つまり、あらかじめ通常姿勢から作成されたJRTのリストを用意しておき、現在の運転姿勢のJRTとそれらすべてを比較する。そして、それらの距離のうち最小となるものを考える。もし、この距離が小さい値であるなら、現在の運転姿勢は通常と見なすことができるため、この最小距離はドライバーの運転可能能力の指標として振る舞う。このようにして定義された異常度はドライバーの体格に依存せず、さらに網羅の難しい異常姿勢を定義する必要がないという利点がある。

										z-axis													
										y-axis													
										x-axis													
	Nose	RS	LS	RE	LE	RW	LW	RH	LH														
																					<		
																						>	
																						=	
																							>
Nose																							>
RS																							>
LS																							>
RE																							>
LE																							>
RW																							>
LW																							>
RH																							>
LH																							>

図3.3 Joint Relation Tensor

### 3.1.4 異常検知

山田らは、(3.1) で定義される姿勢間距離の累積和をとり、その累積和が3秒間閾値を超えた場合を異常運転状態と定義していた。本システムでは、後述する *Approximate Computing* を導入するため、直近3秒間における姿勢間距離の平均値を異常度として定義する(3.2)。

ここで  $A_T$  は現在の時刻  $T$  における異常度で、 $\min D_t$  は最小の姿勢間距離、 $N$  は直近3秒間に行われた姿勢間距離の回数である。異常度が閾値  $threshold_{abn}$  に達したら、システムは異常を検知する。すなわちドライバは運転不能状態であると判断する。フレーム単位の異常度から判断するのではなく、移動平均によって判断することから、骨格検知の瞬間的なノイズに対してロバストなモデルとなっている。

なお、直近の3秒間で異常度を考えるのは、国土交通省によるドライバー異常時対応システム基本設計書 [52] において、ドライバの異常を検知してから3.2秒後に制御動作に移ることが提言されていることに依拠する。

$$A_T = \frac{\sum_{T-3 \leq t \leq T} \min D_t}{N} \quad (3.2)$$

### 3.1.5 Approximate Computing

深層学習モデルによる姿勢推定は、RGB画像を入力として3次元の姿勢推定を高い汎化性能と精度で予測することを可能にする。本システムで用いるモデルは比較的軽量であることを考慮しても、電力に制約のある車載環境において長時間連続して推論計算を行うことは負担となる。

そのため本システムでは *Approximate Computing* の考え方を導入し、高精度かつ省電力なドライバモニタリングシステムを実現する。運転中の時間のほとんどはドライバは通常姿勢を維持しており、そのような場合、ドライバが異常な姿勢をとっていないことさえ分かれば姿勢推定の精度や頻度は高い必要はない。

そのため本研究では、電力モードを通常モードと省電力モードの2つをもち、これらを動的に切り替えることを提案する。通常モードでは計算機が可能な限り毎フレーム姿勢間距離の計算を行う。一方の省電力モードでは、毎フレームは行わず、1秒間に2回程度まで計算頻度を低くする。省電力モードでは、通常モードに比べて姿勢間距離の計算を行わないため、その移動平均である異常度は通常モードで算出できる値の近似値となる。そのため、異常度の精度は低くなる一方、電力の節約ができる。

これらモードの切り替えは、図3.4のように行われる。通常モードでは計算機が可能な

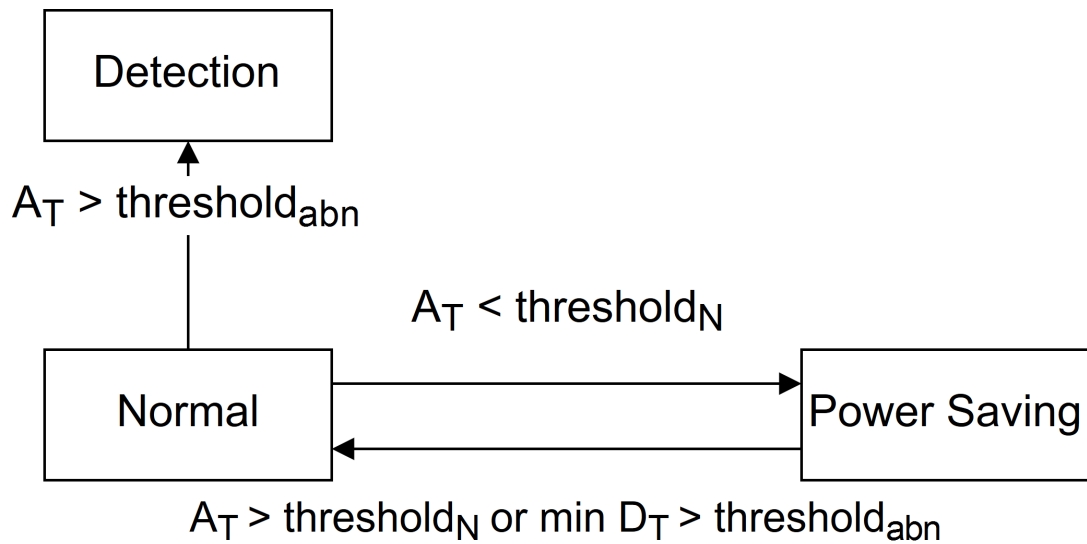


図 3.4 電力モードの状態遷移図

限り毎フレーム姿勢間距離の計算を行うが、通常姿勢が続き異常度  $A_T$  が  $\text{threshold}_N$  を下回りドライバが運転可能状態であると判断されたら、システムは省電力モードへ移行する。省電力モードは通常姿勢が続いている限り継続されるが、異常度  $A_T$  が  $\text{threshold}_N$  を上回るか、姿勢間距離  $\min D_T$  が  $\text{threshold}_N$  を超えるとシステムは即座に通常モードへ移行する。



## 3.2 実験

### 3.2.1 実装方法

本システムは車載できるように、組み込み GPGPU プラットフォームである NVIDIA Jetson TX2 上で実装した。Jetson TX2 の性能を表 3.1 に示す [53]。この組み込みプラットフォームは 15W 程度の電力で動作し本体サイズは  $50\text{mm} \times 87\text{mm}$  と小型で車載に適している。

表 3.1 JetsonTX2 の性能

CPU	HMP Dual Denver 2 + Quad ARM A57
GPU	Pascal, 256 CUDA cores
TDP	15W
メモリ	8GB
サイズ	$50\text{mm} \times 87\text{mm}$

### 3.2.2 実車実験

実車実験のデータセットは山田ら [26] が収集したデータセットのうち一部を用いた。山田らは Microsoft Kinect V2 を深度センサとして用いて、被験者を集め実車上でさまざまな運転姿勢を取得している。Kinect V2 は深度画像の他に RGB 画像も取得できているため、本研究では Kinect V2 で取得したデータのうち RGB 画像のみを使用した。

このデータセットは 30 歳から 50 歳の男性、女性 10 人を被験者とし、正常な運転姿勢をしているデータセットと、ドライバが異常姿勢をとっているデータセットの 2 つから構成される。

通常姿勢のデータセットは、ドライバが正常に運転している様子を撮影したデータで、道路状況や速度によって運転姿勢が変わることを考慮して、高速走行、低速走行、悪路走行の 3 パターンの環境下で取得している。これら走行条件の詳細は表 3.2 に示すとおりである。この正常運転データセットは通常姿勢の辞書の構築に用いられる。ドライバモニタリングシステム動作時は、この通常姿勢辞書をもとにドライバの異常を検知する。

異常運転データセットでは、図 3.5 に示すように、被験者に前傾姿勢・後方倒れ・右方倒れ・左方倒れ・脱力姿勢の 5 つの異常姿勢を演じてもらっている。それぞれの姿勢は 5 秒間ずつ維持してもらい、これを 2 回繰り返している。すなわち、各被験者に対して 10 回分の異常運転姿勢のデータを取得している。

本実験では、このデータセットのうち、入手可能であった 6 人のデータに対して評価実

表 3.2 通常姿勢時の運転環境

	走行距離	巡航速度の上限	走行時間
高速走行	2.3km	100 km/h	10 分
低速走行	0.8km	40 km/h	10 分
悪路走行	0.7km	30 km/h	1 周

験を行った。また、図 3.5 から見て取れるように後方倒れは通常運転時の姿勢と骨格位置の変化は少ないことがわかる。後方倒れは、山田らの実験においてもあまり検出できていない異常姿勢であり、骨格の相対関係だけでは検出が難しいと判断し、本研究では後方倒れを評価対象から除いた。

評価は、正常運転時に異常運転と誤検知されてしまう割合と、異常運転時に正しく異常運転を検知できるかで行った。ただし、誤検知の評価をする際には、姿勢辞書の構築に、評価対象の被験者の姿勢データが含まれないようにしている。



図 3.5 実車実験 (fig.4.3 [5])

### 3.2.3 電力実験

Approximate Computing を評価するために、本研究では実験室環境で電力評価を行った。この実験では図 3.6 のように、出力電力のログをとることができる電源装置に Jetson TX2 をつなぎ、ドライバがさまざまな運転姿勢をとることで、通常モードと省電力モードでどれだけ電力を消費するか計測した。

この実験では、JetsonTX2 上で Ubuntu16.04 が動作し、さらにその上でプログラムを動作させている。さらに USB カメラへの給電も JetsonTX2 を経由している。そのため、計測される電力はこれらをすべて含めた、システム全体の電力となる。



図 3.6 電力評価環境

## 3.3 評価

### 3.3.1 計算速度

本システムは電力モードが通常モードのとき、JetsonTX2 上で 7.8fps で動作することを確認した。これは直近 3 秒間で計測された異常度を、国土交通省のガイドラインで提言されている 3.2 秒以内に反映できることを示しており、リアルタイム性があるといえる。また、省電力モードでは、姿勢間距離の計算を 2fps で行うように調整した。

### 3.3.2 正常運転誤検知率

図 3.7 は通常運転時のテスト結果である。横軸が各通常姿勢で、縦軸は誤検出の割合を示している。誤検出の割合は、全フレームに対して、 $A_T$  が  $threshold_{abn}$  を超えているフレーム数である。アルファベットの A, B, C ... は被験者の ID を表している。また、それぞれのデータセットについて Approximate Computing を適応した場合と適応しなかった場合、それぞれで評価をとった。いくつかのケースで、誤検出が発生しており、Approximate Computing を適応しない場合の平均の誤検知率は 0.14%、適応した場合の誤検知率は 0.16% となった。

Approximate Computing の適応の有無で、誤検知率はほとんど変化がなかったものの、誤検知を起こすデータはその 2 つで少し違っている。これはデータセット中のノイズが影響しており、Approximate Computing 非適応時と適応時で拾ってしまうノイズのフレームが異なることが原因であった。

### 3.3.3 異常運転検知精度

図 3.8 は異常姿勢時のテスト結果である。横軸が各異常姿勢で、縦軸は異常を検知できた回数を示している。6 人の被験者に対して各異常運転動作を 2 回ずつ繰り返してもらったため、最大値は 12 となっている。誤検知率のときと同様に、それぞれの姿勢について Approximate Computing を適応しなかった場合とした場合について評価をとった。また、アルファベットの A, B, C ... は被験者の ID を表しており、被験者ごとに正しく異常運転として検知できたの回数がかかるようになっている。

ほとんどの異常姿勢を検出できているが、いくつかのケースでは検出に失敗している。検出できなかった異常姿勢を分析すると、そのときの関節の相対関係の多くが通常姿勢辞書に含まれていることが原因であった。特に、このデータセットは小型バスで撮影されていることから、個人によっては左方の巻き込み確認と異常姿勢である前傾姿勢や左方倒れ

との差異が少ない場合があった。

検出数は **Approximate Computing** を適応した場合においても変化がなかった。これはシステムが少しでも運転姿勢に異常を検知したら、省電力モードから通常モードへ即座に移行するからである。つまり **Approximate Computing** を適応した場合でも DMS の安全性は保たれていることがわかる。

### 3.3.4 消費電力

電力の動的な切り替えによる、消費電力の変化とその結果を図 3.9 に示す。図 3.9 の下図はシステム全体の電力の使用量を示している。通常モードでは、可能な限り異常度の計算を行うため、常に電力を多く使用している。一方、省電力モードでは、1 秒に 2 回行う異常度の計算時以外は低い電力で推移している。この実験の結果、省電力モードでは通常モードに比べ平均 35 % 電力を削減できることがわかった。これは、ドライバが異常度の低い通常姿勢を続け、システムが省電力モードで動作し続ける限りは、**Approximate Computing** により同じ精度で 35% 電力を削減したドライバモニタリングシステムを実現できるということを示している。

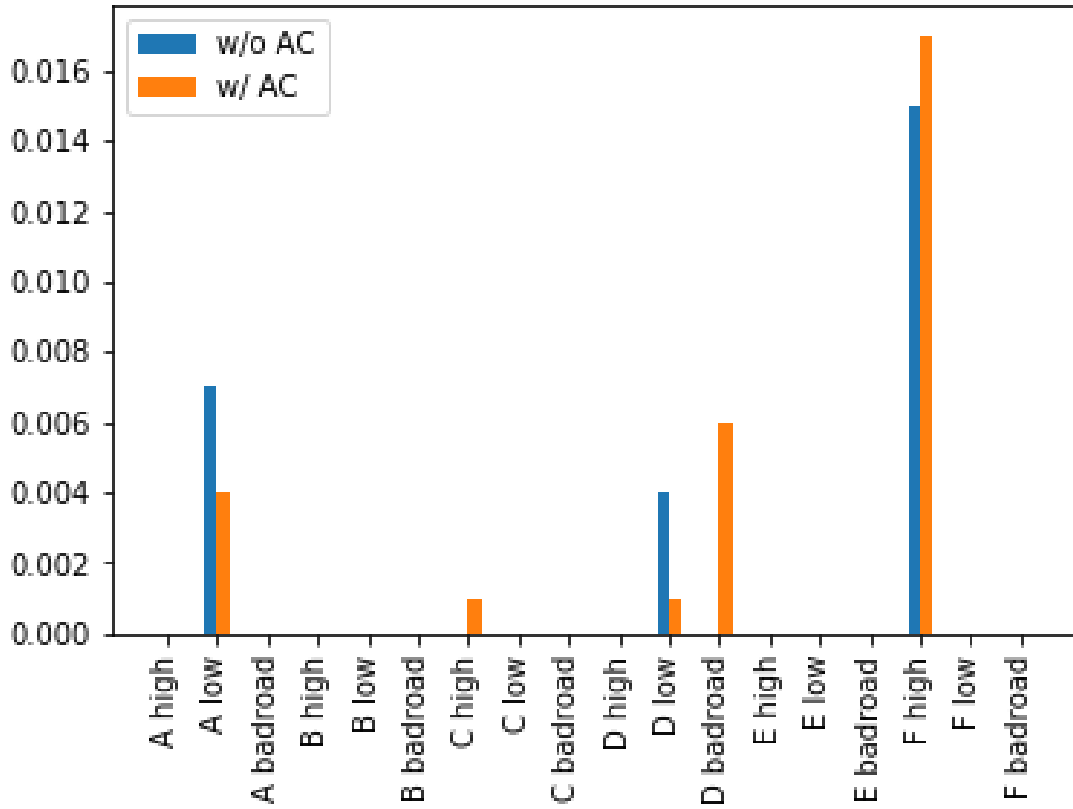


図 3.7 正常運転誤検知率

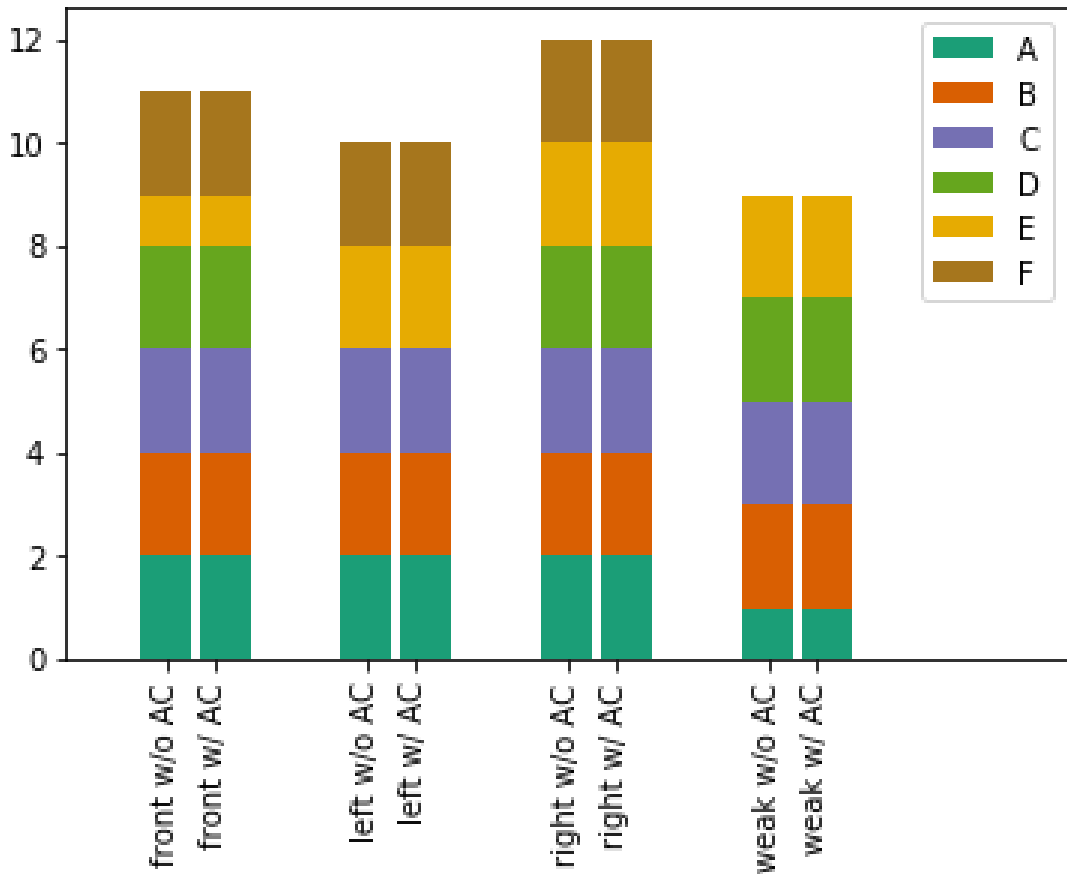


図 3.8 異常運転検出数





**Normal mode**  
**8.1 W**

**Power saving mode**  
**5.3 W**  
**(35% power saving)**

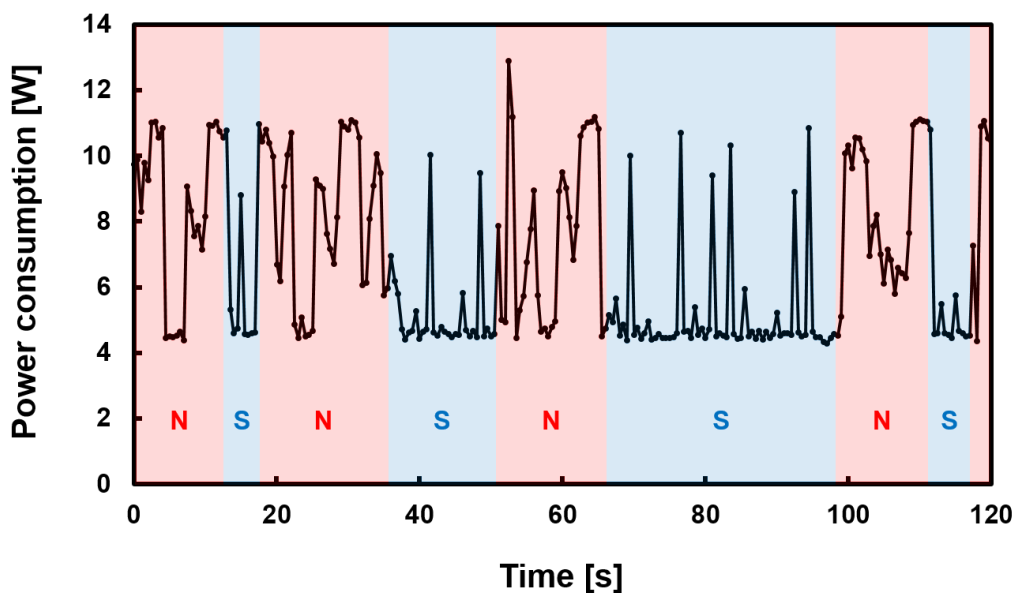


図 3.9 電力評価



## 第4章

# フォークリフト運転支援のための棚検知システム

本章では、フォークリフト運転支援のための棚検知システムについての詳細を述べる。本研究では、図 4.1 に示すような工場内の棚に対して、棚の検知と空き状況の推定を行い運転支援をすることを目的としている。

システムが棚の空き状況を推定し、その情報をドライバーに伝えることは、ドライバーとシステムの協調動作を可能にし、操作の効率化や運転事故の防止につながる。例えば、フォークリフトが棚の高所に積み下ろしを行う際はリフト部分が死角になることで、積荷が転落する事故が発生するが、死角をシステムが補うことでこのような事故は防ぐことができる。また、システムが棚の空き状況を推定することは運転支援に留まらず自動運転においても重要な役割を担う。



図 4.1 工場内の棚

## 4.1 棚検知における課題

フォークリフト視点で工場内の棚の位置や空き状況を推定することは、2章で紹介したパレットの位置推定や商品棚の管理システムとは異なる技術が要求される。図4.1のような棚に対して、棚の検知と空き状況の推定を行うために次のような課題を解決する必要がある。

1. 工場内の棚にはさまざまな種類があり、図4.1左のように棚が整列している場合もあれば、図4.1右のように収納場所が整列されていない場合もある。
2. 棚に積まれる荷物やパレットは、さまざまな工場から運ばれてくるため、その種類は多数存在する。
3. 工場内の棚は、棚の間に仕切りがない場合がある。そのため隣や奥にある棚が見えてしまう。

まず、1の課題から特徴量を使った単純なアプローチではうまくいかない。たとえば、ハフ変換などによる直線抽出では、十分に棚の輪郭領域を捉えることができない。図4.2は棚画像に対してハフ変換による直線抽出を行った結果であり、棚の輪郭領域を十分に検出できていないことがわかる。また、工場内には直線状のものが多く、ノイズが多くのもってしまう。

次に、2の課題から網羅的に荷物の形状をデータベース化することは難しい。図4.3は棚に積み立てられている積荷やパレットを一部拡大したものであるが、一つの棚においても多くの種類が存在している。また、図4.3が示すように、フォークリフトが棚に対して正面を向いていない限り、積荷やパレットの検出領域は長方形にならず、バウンディングボックスをベースとした物体検出手法は有効でない。

さらに、パレット推定に有効な手法である深度センサも、3の理由から得られる深度マップがスパースなものになってしまい、その情報から棚状況を推定するのが難しいという問題がある。

これらに対処するために工場内の棚や積荷にマーカを付与し、マーカ情報をもとに棚領域や積荷を検知する手法は考えられるが、それら全てにマーカをつける導入コストは高い。また、マーカは汚れなどで読み取れなくなる可能性があり、特に工場内ではそのような欠損が起りやすいという問題がある。



図 4.2 工場内の棚に対するハフ変換による直線抽出



図 4.3 工場内の棚の積荷

## 4.2 提案手法

これらの課題から、本研究では棚の検知と空き状況の推定をフォークリフトに載せた RGB カメラを入力にした、セマンティックセグメンテーションの問題に帰着させることを提案する。すなわち、本研究で提案するシステムでは工場内の様子を撮影した画像を、ピクセルごとに背景・埋まっている棚・空いている棚の3クラスに分類する。

セマンティックセグメンテーションはバウンディングボックスを用いる物体検出と異なり、対象領域が長方形でなくても過不足なく推定できる。この性質は図4.1のようにフォークリフトが棚に対して正面を向いていない状況でも有効であり、棚の空き状況の推定に適している。さらに、運転支援に留まらず、フォークリフトの自動運転に応用する際にも、セマンティックセグメンテーションによる予測は、自動運転研究の知見を取り入れやすいという利点がある。

この手法では、工場内の設備や積荷に対してマーカの付与などは一切行わず、フォークリフトに載せた RGB カメラと組み込み用計算機上のみで構築される。そのため、本システムは低い導入コストで実現できることも利点である。

また、フォークリフトの運転支援では組み込み用計算機で実行できるリアルタイム性の高いモデルが要求される。そのため、本システムでは高い精度とリアルタイム性を実現している MobileNetV2 [47] を用いる。[47] では、MobileNetV2 をベースに複数のセマンティックセグメンテーション用ネットワークが提案されているが、本システムは其中でも精度が高いことが報告されている、Output Stride が8の MobileNetV2 と簡易的なデコーダを組み合わせたネットワーク構成を採用した。

### 4.3 実装

本システムは3章で述べたドライバモニタリングシステム同様、組み込み GPGPU プラットフォームである NVIDIA Jetson TX2 上で実装した。JetsonTX2 の性能は表 3.1 に示すとおりである。

実験は図 4.4 のようにフォークリフトのマスト部分に GoPro カメラを載せることで撮影した工場内の棚画像に対して行った。撮影した工場は室内・室外を含み 5 箇所あり、それぞれで昼と夜に撮影することで、10 パターンの環境下におけるデータを取得した。この 10 パターンの環境それぞれについて 10 枚前後ずつ画像にアノテーションを行い、全体で 145 枚のラベル付きデータセットを構築した。さらに、それらを 121 枚の訓練データと 24 枚の評価用データに分割した。

一般に深層学習モデルは表現力が高いため、121 枚という訓練データは少なく過学習の恐れがある。しかし、セマンティックセグメンテーションのためのピクセル単位のアノテーションは時間的に多大な労力を要する。そのため本研究では水平方向の反転、回転、ノイズ付与、輝度調整によるデータオーグメンテーションと Hung らが提案する手法 [50] で約 4 万枚のラベルなしデータとともに半教師あり学習を行った。

GoPro によって取得される画像は  $1920 \times 1080$  であるが、軽量化のためモデルへの入力サイズは  $256 \times 256$  とした。予測時は  $1920 \times 1080$  の画像を  $1080 \times 1080$  にセンタークロップした後、 $256 \times 256$  にリサイズしたものがモデルへの入力となる。



図 4.4 撮影に使用したフォークリフトと GoPro カメラ



## 4.4 評価

### 4.4.1 計算速度

Jetson TX2 上で実行したときの fps は 5.66 であり，組み込み計算機上でリアルタイムで実行可能なことがわかった．ただし，フォークリフトの運転支援システムで用いる場合に 5.66fps が問題にならない遅延かは検討する必要がある．

### 4.4.2 予測精度

評価は IoU(Intersection over Union) によって行った．IoU は図 4.5 に示すように予測と正解の領域がどれだけ一致しているかを表している指標で，0 から 1 の値をとり，値が 1 に近いほど精度が良いことを表す．

本研究では，埋まっている柵に対する IoU，空いている柵に対する IoU，それらの平均，空き状況を考慮せずに柵の領域のみに対する IoU の 4 つを計測した．その結果を表 4.1 に示す．柵の領域のみであれば，0.829 と比較的高い値となったが，柵の空き状況を考慮した場合 IoU は 0.690 まで下がった．


$$IoU = \frac{\text{Area of Intersection}}{\text{Area of Union}} = \frac{\text{図 4.5 の図}}{\text{図 4.5 の図}}$$


図 4.5 Intersection over Union

表 4.1 評価

IoU(満)	IoU(空)	IoU(平均)	IoU(柵領域のみ)
0.733	0.646	0.690	0.829

図 4.6 に評価データに対する予測結果の一例を 4 つ示す。図の左の列が本システムによる予測結果であり、右の列が正解データである。

一番上段は、柵の領域やその空き状況を比較的良く予測できている結果である。このような予測結果になれば、柵の空き状況がドライバに正しく伝えることができ、直感的にシステムと協調動作することが可能となる。

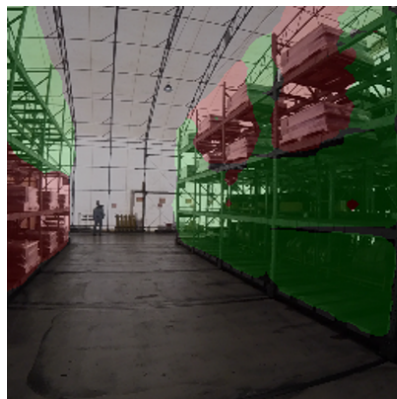
2 段目の例では、柵領域は十分に推定できているものの、空き状況に関しては境界が不明瞭であったり、奥にある状況の推定には失敗している。

3 段目は、一見問題なく推定できている例であるが、空き状況を考慮した IoU は 0.437 と低い値になっている。これは左下の領域を空いていると正しく推定できていないため、空き状況に関する IoU が 0 となってしまったことが原因となる。2 段目の例でも同様であるが、遠い場所の予測は、フォークリフトが近づいていくことで正しく推測を行う領域であると考えられ、実用上はあまり問題とならない。そのため、本システムを評価する指標として、IoU はあまり予測の良し悪しと結びつかない場合があり、検討する必要がある。

最下段の画像は、空き状況だけでなく柵の輪郭の推定も不明瞭なものとなっている。このような結果となった主な原因は、撮影されたデータの中に最下段の画像のような整列されていない柵の画像があまりなかったためであると考えられる。そのため、訓練データを増やすことで、このような結果は緩和できることが期待できる。また、このような予測は時間方向に連続していることから、時系列情報を活用することで、境界はより明瞭になり予測を安定させることができる。



(a) IoU(平均): 0.917 IoU(棚のみ): 0.928



(b) IoU(平均): 0.690 IoU(棚のみ): 0.842



(c) IoU(平均): 0.437 IoU(棚のみ): 0.891



(d) IoU(平均): 0.552 IoU(棚のみ): 0.580

図 4.6 予測結果



## 第 5 章

# 結論

本研究では運転事故を防止するためのリアルタイム性と消費電力を考慮した車載システムを目指し、その実事例としてドライバモニタリングシステムとフォークリフト運転支援のための柵検知システムを提案・実装した。

ドライバモニタリングシステムでは、山田らの提案するドライバ異常推定の手法を kanazawa らの手法によって汎化性能の高いモデルにした。さらにドライバモニタリングシステムが監視する時間のうちのほとんどを通常姿勢が多く占めることに着目し、Approximate Computing の考え方を導入することで、ドライバの状態に合わせて適切に計算精度を変化させることで消費電力を下げる手法を提案した。このシステムは組み込み計算機上でリアルタイムに動作し、正常姿勢に対する低い誤検出率と異常姿勢に対する高い検出率を実現した。また、Approximate Computing を適用することで、検出率を維持しつつ消費電力を最大 35% 削減できることを示した。

本システムでは、多少の誤検知と検出漏れが確認されてる。これは、本手法によって通常運転姿勢と異常運転姿勢が完全には分離できていないことを示唆している。これらを解決する方法としては 2 つあげられる。まず、ひとつは取得する関節数を増やすことである。異常運転を検出できなかった理由のひとつに、異常運転時の相対姿勢表現が、通常姿勢辞書に含まれることがあった。そのため、取得する 3 次元の関節座標の数を増やすことで、より通常運転姿勢と異常運転姿勢を分離することができると考えられる。もうひとつの改善方法は、他のドライバモニタリングシステム手法との組み合わせである。2 章で紹介したように、ドライバモニタリングシステムには生体信号や目の情報を利用したものがある。これらの手法は、姿勢検知とは直交した概念であるため、本研究の手法と組み合わせることができる。特に本手法では、対象としなかった後方倒れは、目の情報と組み合わせることで検知できる可能性がある。一方、目の情報を取得することが難しい前傾姿勢などは姿勢検知で補うことができる。

本研究により、計算頻度の間引きという単純な手法で大きく改善されたことからわか

るように、ドライバモニタリングシステムと **Approximate Computing** の相性はよい。そのため、今後はソフトウェア側の制御だけでなくハードウェア側と協調した **Approximate Computing** をすることで、より強力なドライバモニタリングシステムが期待される。ドライバモニタリング以外でも、この手法のような姿勢検知 **Approximate Computing** が電力事情の厳しいモバイル/IoT 上の HCI アプリケーションの電力消費削減に役立つと考えられる。

フォークリフト運転支援のための柵検知システムでは、柵の空き状況推定をセマンティックセグメンテーションに帰着させることを提案した。このシステムでは、評価用のデータに対して、空き状況の推定は IoU で 0.690、柵領域に関する IoU は 0.829 の精度で予測できた。さらに、このシステムは組み込み用 GPGPU プラットフォームである NVIDIA Jetson TX2 上で 5.66fps で動作し、車載上でもリアルタイムに動作できることが確認できた。

本システムの学習では、訓練に使ったラベル付きデータは 121 枚と、セマンティックセグメンテーションに使う学習データとしてかなり小規模であった。そのため、このモデルの精度にはまだ伸びしろがあることが期待でき、今後は学習データを増やすことでより正確なモデルを構築することが課題となる。

また、本システムでは画像を背景と埋まっている柵、空いている柵の 3 クラスに分類していたが、さらにパレットやパレットの開口部をクラスに加えることで、パレット位置の推定にも拡張可能である。柵の検知、さらにはパレット位置をリアルタイムで推定することは、フォークリフトの運転支援、さらには自動運転にとって必要不可欠な技術である。そのため本システムは、より幅広いフォークリフトの運転支援や自動運転に応用できることが期待される。

## 参考文献

- [1] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.
- [2] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [3] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [4] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. *arXiv preprint arXiv:1802.02611*, 2018.
- [5] 山田高寛. 運転姿勢を用いたドライバモニタリングシステム. Master’s thesis, 東京大学大学院情報理工学系研究科, 2018.
- [6] 国土交通省. トラック運送業の現状等について, 2018.
- [7] 厚生労働省. 「死亡災害報告」による死亡災害発生状況 (平成 29 年確定値), 2018.
- [8] Shigeki KAWACHI, Masahito HITOSUGI, Tetsuo MAKI, and Toshiaki SAKURAI. Vehicle driving posture at drivers’ unconsciousness and accident prevention. 日本職業・災害医学会会誌 = Japanese journal of occupational medicine and traumatology, Vol. 61, No. 2, pp. 144–147, mar 2013.
- [9] 交通事故総合分析センター. 四輪運転者の発作, 急病による交通事故の発生状況の研究, 2011.
- [10] 警察庁. 原付以上運転者 (第 1 当事者) の携帯電話使用等に係る交通事故の発生状況 (平成 23 年以降) .
- [11] S. Kaplan, M. A. Guvensan, A. G. Yavuz, and Y. Karalurt. Driver behavior analysis for safe driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*,

- Vol. 16, No. 6, pp. 3017–3032, Dec 2015.
- [12] Birgitta Thorslund. *Electrooculogram analysis and development of a system for defining stages of drowsiness*. Statens väg-och transportforskningsinstitut, 2004.
- [13] Y. Dong, Z. Hu, K. Uchimura, and N. Murayama. Driver inattention monitoring system for intelligent vehicles: A review. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 12, No. 2, pp. 596–614, June 2011.
- [14] Guosheng Yang, Yingzi Lin, and Prabir Bhattacharya. A driver fatigue recognition model based on information fusion and dynamic bayesian network. *Information Sciences*, Vol. 180, No. 10, pp. 1942 – 1954, 2010. Special Issue on Intelligent Distributed Information Systems.
- [15] Shuyan Hu and Gangtie Zheng. Driver drowsiness detection with eyelid related parameters by support vector machine. *Expert Systems with Applications*, Vol. 36, No. 4, pp. 7651 – 7658, 2009.
- [16] Stephen H Fairclough and Robert Graham. Impairment of driving performance caused by sleep deprivation or alcohol: a comparative study. *Human factors*, Vol. 41, No. 1, pp. 118–128, 1999.
- [17] M. Zhang, C. Chen, T. Wo, T. Xie, M. Z. A. Bhuiyan, and X. Lin. Safedrive: Online driving anomaly detection from large-scale vehicle data. *IEEE Transactions on Industrial Informatics*, Vol. 13, No. 4, pp. 2087–2096, Aug 2017.
- [18] Sarah Otmani, Thierry Pebayle, Joceline Roge, and Alain Muzet. Effect of driving duration and partial sleep deprivation on subsequent alertness and performance of car drivers. *Physiology & Behavior*, Vol. 84, No. 5, pp. 715 – 724, 2005.
- [19] Pierre Thiffault and Jacques Bergeron. Monotony of road environment and driver fatigue: a simulator study. *Accident Analysis & Prevention*, Vol. 35, No. 3, pp. 381 – 391, 2003.
- [20] Arun Sahayadhas, Kenneth Sundaraj, and Murugappan Murugappan. Detecting driver drowsiness based on sensors: A review. *Sensors*, Vol. 12, No. 12, pp. 16937–16953, 2012.
- [21] David F Dinges and Richard Grace. Perclos: A valid psychophysiological measure of alertness as assessed by psychomotor vigilance. *US Department of Transportation, Federal Highway Administration, Publication Number FHWA-MCRT-98-006*, 1998.
- [22] R. A. Berri, A. G. Silva, R. S. Parpinelli, E. Girardi, and R. Arthur. A pattern recognition system for detecting use of mobile phones while driving. In *2014 International Conference on Computer Vision Theory and Applications (VISAPP)*, Vol. 2, pp. 411–418, Jan

- 2014.
- [23] T. H. N. Le, Y. Zheng, C. Zhu, K. Luu, and M. Savvides. Multiple scale faster-rcnn approach to driver's cell-phone usage and hands on steering wheel detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 46–53, June 2016.
- [24] Kaggle. State farm distracted driver detection.
- [25] Yehya Abouelnaga, Hesham M Eraqi, and Mohamed N Moustafa. Real-time distracted driver posture classification. *arXiv preprint arXiv:1706.09498*, 2017.
- [26] T. Yamada, H. Irie, M. Kunitake, E. Nagano, and S. Sakai. Estimating driver's readiness by understanding driving posture. In *2018 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–4, Jan 2018.
- [27] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pp. 483–499. Springer, 2016.
- [28] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4724–4732, 2016.
- [29] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *International Conference on Computer Vision*, Vol. 1, p. 5, 2017.
- [30] Francesc Moreno-Noguer. 3d human pose estimation from a single image via distance matrix regression. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1561–1570. IEEE, 2017.
- [31] Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Reconstructing 3d human pose from 2d image landmarks. In *European conference on computer vision*, pp. 573–586. Springer, 2012.
- [32] Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, and Kostas Daniilidis. Sparse representation for 3d shape estimation: A convex relaxation approach. *IEEE transactions on pattern analysis and machine intelligence*, Vol. 39, No. 8, pp. 1648–1661, 2017.
- [33] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Smpl: A skinned multi-person linear model. *ACM Trans. Graph.*, Vol. 34, No. 6, pp. 248:1–248:16, October 2015.
- [34] Sparsh Mittal. A survey of techniques for approximate computing. *ACM Comput. Surv.*, Vol. 48, No. 4, pp. 62:1–62:33, March 2016.

- [35] V. K. Chippa, D. Mohapatra, K. Roy, S. T. Chakradhar, and A. Raghunathan. Scalable effort hardware design. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 22, No. 9, pp. 2004–2016, Sep. 2014.
- [36] Jia-Liang Syu, Hsin-Ting Li, Jen-Shiun Chiang, Chih-Hsien Hsia, Po-Han Wu, Chi-Fang Hsieh, and Shih-An Li. A computer vision assisted system for autonomous forklift vehicles in real factory environment. *Multimedia Tools and Applications*, Vol. 76, , 11 2016.
- [37] Z. He, X. Wang, J. Liu, J. Sun, and G. Cui. Feature-to-feature based laser scan matching for pallet recognition. In *2010 International Conference on Measuring Technology and Mechatronics Automation*, Vol. 2, pp. 260–263, March 2010.
- [38] B. Molter and J. Fottner. Real-time pallet localization with 3d camera technology for forklifts in logistic environments. In *2018 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, pp. 297–302, July 2018.
- [39] M. Marder, S. Harary, A. Ribak, Y. Tzur, S. Alpert, and A. Tzadok. Using image analytics to monitor retail store shelves. *IBM Journal of Research and Development*, Vol. 59, No. 2/3, pp. 3:1–3:11, March 2015.
- [40] Weidong Geng, Feilin Han, Jiangke Lin, Liuyi Zhu, Jieming Bai, Suzhen Wang, Lin He, Qiang Xiao, and Zhangjiong Lai. Fine-grained grocery product recognition by one-shot learning. In *Proceedings of the 26th ACM International Conference on Multimedia, MM '18*, pp. 1706–1714, New York, NY, USA, 2018. ACM.
- [41] Corentin Henry, Seyedmajid Azimi, and Nina Merkle. Road segmentation in sar satellite images with deep fully convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 02 2018.
- [42] Monjoy Saha and Chandan Chakraborty. Her2net: A deep framework for semantic segmentation and classification of cell membranes and nuclei in breast cancer evaluation. *IEEE Transactions on Image Processing*, Vol. 27, No. 5, pp. 2189–2200, 2018.
- [43] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.
- [44] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.
- [45] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE Conf. on Computer Vision and Pattern Recognition*

- (*CVPR*), pp. 2881–2890, 2017.
- [46] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, Vol. 40, No. 4, pp. 834–848, 2018.
- [47] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, June 2018.
- [48] Xinyu Huang, Xinjing Cheng, Qichuan Geng, Binbin Cao, Dingfu Zhou, Peng Wang, Yuanqing Lin, and Ruigang Yang. The apolloscape dataset for autonomous driving. *arXiv preprint arXiv:1803.06184*, 2018.
- [49] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2018.
- [50] Wei-Chih Hung, Yi-Hsuan Tsai, Yan-Ting Liou, Yen-Yu Lin, and Ming-Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. In *BMVC*, 2018.
- [51] George Papandreou, Liang-Chieh Chen, Kevin P. Murphy, and Alan L. Yuille. Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.
- [52] 国土交通省自動車局先進安全自動車推進検討会. 先進安全自動車 (asv) 推進計画報告書—第5期 asv 推進計画における活動成果について—, 2016.
- [53] NVIDIA. Embedded systems developer kits, modules, & sdks — nvidia jetson.

# 発表文献

## ■口頭発表（査読なし）

1. 谷台廣紀, 宮永 瑞紀, 入江 英嗣, 坂井 修一セキュアプロセッサにおける楕円曲線暗号の評価, 情報科学技術フォーラム講演論文集, No. 4, CL-004 (2017).



## 謝辞

指導教官である坂井修一教授には相談会やミーティングで、研究方針に様々な指導をいただいた上、個人的な課外活動に対しても配慮していただき、快適な研究生活を送ることができました。ありがとうございます。入江英嗣准教授には、研究の方向性や論文・発表の添削などで、いつも親身になって相談に乗っていただきました。大変お世話になりました、ありがとうございます。

また、秘書の八木原晴水さんと赤羽彩子さんには、各種事務手続きや研究環境のサポートで大変お世話になりました。ありがとうございました。また、研究室の方々には、常に優しく接していただき、とても充実した研究生活を送ることができました。坂井・入江研究室の皆様感謝いたします。

本研究の一部は株式会社小松製作所によります。心から感謝申し上げます。