

# エッジAIを用いたフォークリフト安全支援システム

## Edge AI-based Forklift Safety Support System

田窪 伸哉\*

Shinya Takubo

永田 篤樹

Atsuki Nagata

猪飼 隼吾

Shungo Ikai

李 漠飛

Mofei Li

西岡 進吾

Shingo Nishioka

吉村 明展

Akinobu Yoshimura

当社では工場で広く使用されているフォークリフトと作業者の接触事故を防ぐため、作業者がフォークリフトに接近したことを検知するフォークリフト安全支援システムを開発している。検知から発報まではリアルタイム性が要求されるため、現場での即時応答が可能なエッジコンピューティング方式での実現が望まれている。本稿では、小型のエッジコンピュータに搭載したエッジAIを用いたフォークリフト安全支援システムについて提案する。また、精度と速度の両立を図るためのエッジAI評価、効率よく精度を向上させる手法の検討結果についても報告する。

To mitigate accidents involving forklifts and people, we have developed a forklift safety support system capable of detecting the presence of people near a forklift. Real-time responsiveness is crucial, necessitating immediate detection and reporting. To address this, we propose utilizing edge computing to enable swift responses. This paper introduces a forklift safety support system leveraging edge AI installed in a compact edge computer. Additionally, we discuss the evaluation of edge AI's accuracy and speed at the edge, along with the examination of techniques to enhance the accuracy efficiently.

キーワード：フォークリフト安全支援、エッジAI、学習ツール、量子化、FPGA

## 1. 緒言

当社では生産現場の改善として、フォークリフトの「安全支援」を対象とした作業者検知システムを開発している。本システムでは、作業者がフォークリフトに接近した際にフォークリフトの運転手に発報することや、現場の状況を後から振り返ることで、生産現場の安全を支援することを目的としている。実現手段としてはカメラや超音波、ICタグなどの方式があるが、当社はカメラ方式を採用している(表1)。

作業者の検知から発報まではリアルタイムに行うことが望ましく、当社は小型のエッジコンピュータ(表2)を開発し、各フォークリフトに搭載し判定処理に用いている。また、ヘルメットや作業服、背丈や持ち物の有無など、工場内で想定される様々な作業者のパターンによらず高精度に検知するため、AI(人工知能)の中でも近年特に進化しているディープニューラルネットワーク(DNN)<sup>\*1</sup>をエッジコンピュータのFPGA<sup>\*2</sup>上に搭載することで、判定精度と速度を両立している。

本稿では2章で、フォークリフトのシステム構成、及びDNN開発サイクルを含むエッジ&クラウドシステムの全体構成、3章で判定精度と速度の両立のためのエッジAI評価、4章で効率的な作業者検知モデル開発手法について説明する。

表1 周辺作業者の検知方式比較

方式	超音波	ICタグ	カメラ
検知対象	反射物	専用タグ	人
マーカ	○不要	×必要	○不要
検知範囲	~4m	~10m	~10m
映像表示	×	×	○
録画	×	×	○

表2 エッジコンピュータ諸元

寸法[mm]	115mm × 63mm (W×H)
周辺I/F	USB, Display Port, GPIO等
プロセッサ	Arm + FPGA
OS	Yocto Linux

## 2. システム構成

本システム構成を図1及び図2に示す。

図1はフォークリフトのシステム構成である。前後方にカメラを1台ずつ設置し、運転手座席後方に設置したエッジコンピュータと接続する。カメラ映像内の周辺作業者を検知すると、運転手座席上部に設置したモニタに検知状況を表示するとともに、ブザーを発報する。また、エッジコンピュータは社内無線ネットワークと接続されているため、画像や検知結果をクラウドサーバに送信することや、クラ



図1 フォークリフトシステム構成

ウドから稼働状況を監視することが可能である。

図2は、クラウド側の機能を含めた全体構成である。エッジコンピュータに搭載したDNNで周辺作業者を検知すると、画像及び結果をエッジコンピュータ内のストレージに保存するとともに、社内無線ネットワーク経由でクラウドデータベースに送信する。クラウド側には年単位でデータを保存できるストレージがあるため、任意の期間の映像を後から振り返ることが可能である。また、画像と結果を基に危険度を分析した見える化ツールを各部門に公開し、安全教育などに活用している。

アノテーション<sup>※3</sup>に関しては、データ収集が図2の通り自動化されているため、大量のデータから検知対象を探索する作業が不要で効率よくアノテーションできる。

また、DNNの学習は通常、高度なプログラミングスキル

と複雑な環境構築を要するが、本ツールはWEBブラウザとマウス操作のみで学習サイクルを回すことが可能なツールを開発し、活用している (図3 (a))。

学習したDNNの結果は、履歴と精度表で定量化し管理している (図3 (b))。また、誤り傾向を画像で確認するために、精度表の任意のセルをクリックすることで該当画像にジャンプする機能を搭載しており、プログラムスキルレスで定量化と画像の両面からDNNの弱点分析が可能である。

学習したDNNは、社内無線ネットワークを介してエッジコンピュータにリモートアップデートし、国内外各拠点のDNNを運用管理している。

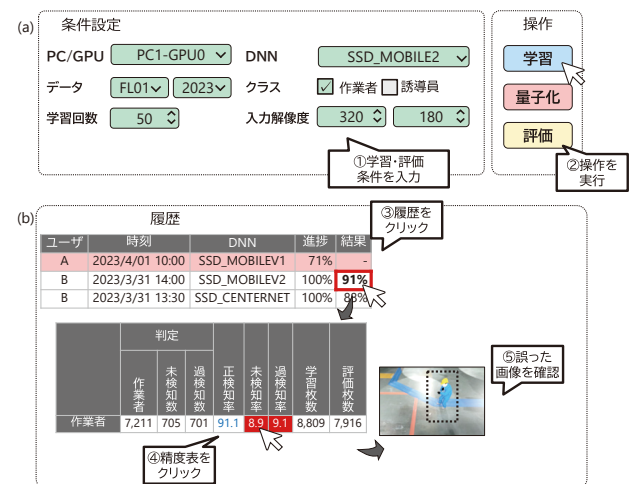


図3 独自開発したDNN学習ツール

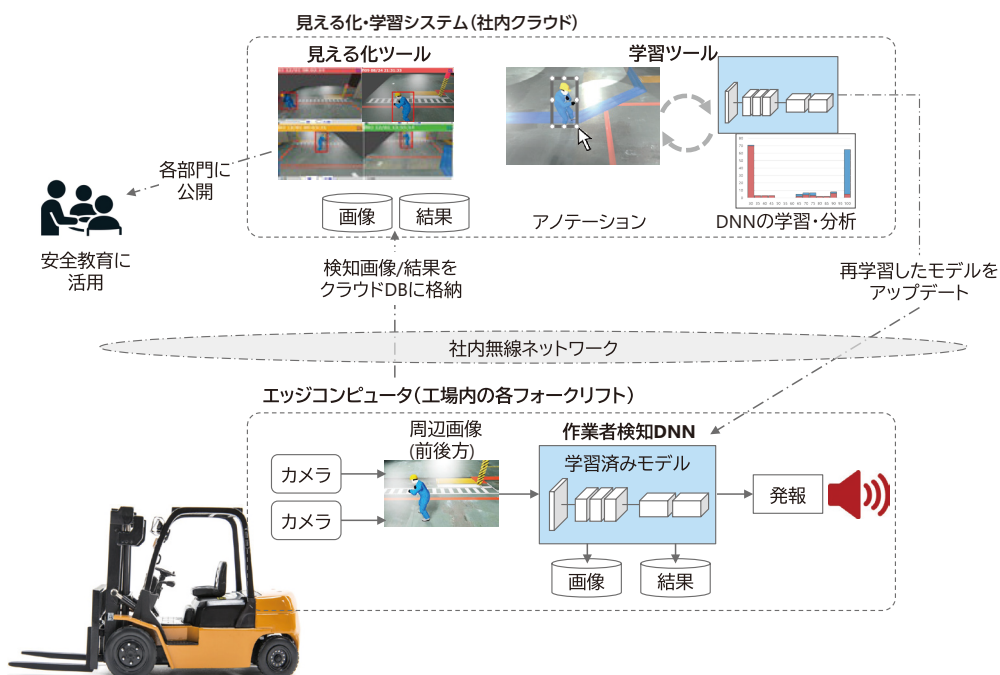


図2 エッジ&クラウドシステム全体構成

### 3. エッジAIモデル開発

DNNは、様々なデータを学習させることで画像処理などの従来方式よりも高い性能が出る一方、大量の演算が必要であることが知られている。そこで近年、リソースの限られた小型装置でDNNを高速・高精度に実行するエッジAIの研究が進んでいる。エッジAIを実現するため、「①エッジに適したDNNのベンチマーク」「②精度劣化を低減しつつ小型化する量子化ビットの評価」を実施した。本章における評価諸元を表3に示す。

表3 評価諸元

データ	当社工場内で取得した作業員データ 15,000枚 (学習：10,000枚、評価：5,000枚)
判定閾値	信頼度閾値: 0.3 IoU閾値 : 0.1

なお、信頼度閾値とは、検知枠に付与された信頼度を基に出力有無を決定する閾値である。また、IoU (Intersection over Union) 閾値とは、正解と検知枠の重なりを基に出力有無を決定する閾値である。

#### 3-1 エッジに適したDNNのベンチマーク

DNNはいくつかのタスクに分けられるが、今回は周辺作業員の接近を検知するアプリケーションであることから、画像中から対象物の位置を求めるObject DetectionタスクのDNNを選択した。Object Detection用のDNNにおいて、高速で精度の良いことが知られるSSD (Single Shot multi-box Detector)<sup>(1)</sup>は多くの派生手法が提案されているが、精度は学習データセットに依存する。今回、当社保有のフォークリフト周辺作業員の画像データを用いて、エッジに適したDNNをベンチマークした。なお、本稿で示す精度は、全て作業員の枠単位で算出した値である。

図4に比較結果を示す。横軸が演算回数、縦軸が精度、バブルサイズがパラメータサイズを示す。高精度・高速・小型の観点ではグラフの左上でバブルサイズが小さいことが望ましい。今回評価した8種類のDNNにおいては、精度の観点でSSD\_VGG16、速度の観点でSSD\_MOBILENETV3\_SMALLが優位であるものの、両立の観点で、SSD\_MOBILENETV2が適していると判断した。

#### 3-2 精度劣化を低減する量子化ビットの評価

DNNは図5のように、パラメータと入力値の積和演算を大量に繰り返す。この演算をfloat32ビット（浮動小数点精度）ではなく、int8ビット（整数）などで表すことで演算回路を小さくする「量子化」という技術が知られている。例えば32ビットと8ビットを比較すると、演算量は8ビットの方が32ビットに比べ1/16 ((8×8)÷(32×32))削減できる。このように量子化は高速化や省電力化の観点で、エッジAI実現に不可欠な技術である。今回はパラメータと入力値のビットを、エッジAIで一般的に使用される8

ビットを基準とし、1ビットまで順に量子化した際の精度及び速度について検証した。なお、本稿ではPTQ (Post Training Quantization) と呼ばれる、学習後にDNNを量子化する手法を用いた。

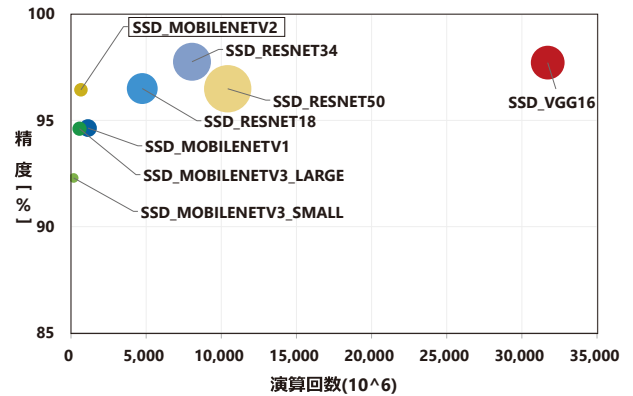


図4 DNNベンチマーク

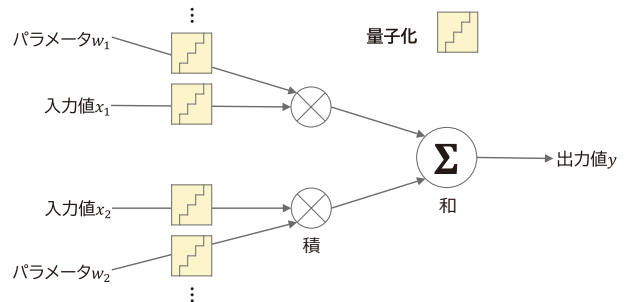


図5 量子化の原理

図6は、横軸を量子化ビット（パラメータ×入力値の降順）、縦軸を精度[%]とした図である。量子化ビットを8～1ビットまで変更させて評価したところ、6ビットまでの量子化では量子化前との精度差は見られなかった。また、パラメータと入力値が5ビット以下の量子化の場合、著しく精度が劣化する結果となった。

次に、パラメータと入力値それぞれが量子化に与える影響を調査した。図7は、横軸を量子化ビット、縦軸を精度[%]とし、パラメータ或いは入力値の一方を8ビット固定、他方を8～1ビットまで変化させた際の精度を比較した結果である。結果より、パラメータ・入力値のいずれも、5ビット以下の量子化の場合には著しく精度劣化し、影響の差は見られなかった。演算量の観点では、6ビット同士の演算よりも8ビット×5ビット（或いはその逆）の方が多いが、精度は6ビットの方が良い結果となった。よって単純に演算量だけでなく、パラメータ及び入力値の量子化ビットのバランスも精度に影響することがわかった。

結果より、本システムのエッジAIで精度と速度を両立させるには、量子化ビットはパラメータ6ビット、入力値6ビットが適しており、演算量は量子化前に比べ約1/28、一般的な8ビット量子化に比べ約2/3に削減できることがわかった。

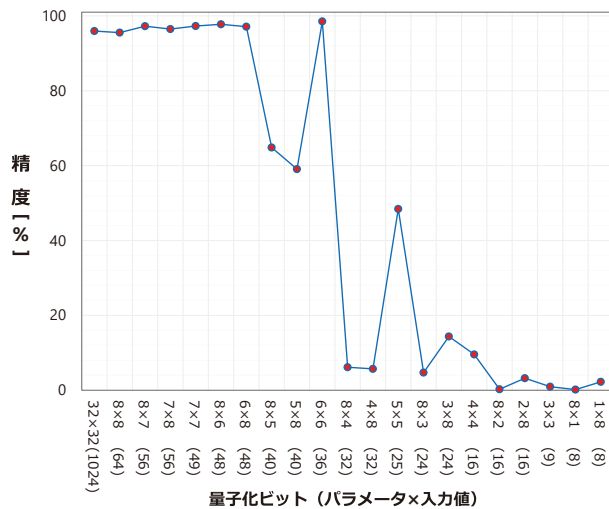


図6 量子化ビットごとの精度比較

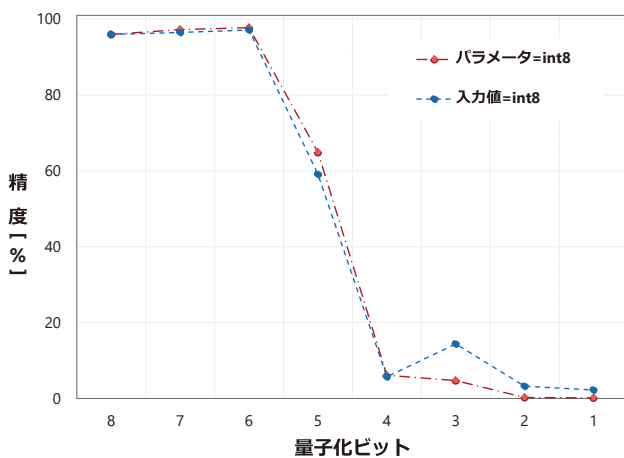


図7 異なる量子化ビットにおける精度比較

## 4. 作業員検知DNN開発

DNNの精度を上げるためには、学習データが重要となる。

本システムは2章で示した通り、検知データをクラウド側に上げることでデータ収集を自動化しているが、収集したデータを用いて効率よくDNN精度を高めるべく、我々は誤って検知したデータから学習データを自動で生成する手法の検証を行った。

### 4-1 作業員検知DNNの課題

DNNに工場のデータ10,000枚を学習させたところ、フォークリフト自身が放つ、進行方向を示すフォークリフ

ト接近灯（以降、「接近灯」）がカメラ内に映っており、作業員と誤って検知する事象が見られた（図8）。これは、学習データの約90%が当社規定の青色の作業服を着用した作業員であり、DNNが青色の物体を作業員と見誤る傾向にあるためであった。接近灯を新たなクラスと定義し、2クラスの物体検出DNNを学習させることで誤りを防ぐことも可能であるが、大量の接近灯のデータ収集や、アノテーションの工数がかかることから、我々は少量の接近灯データを背景画像と合成することで、学習データを効率よく水増しする手法を検証した。



図8 接近灯の誤検知

### 4-2 学習データの水増し手法

図9に、検証した学習データの水増し手法を示す。手順は下記1～3の通りである。

1. データセットを、接近灯が映った画像セット (A) と、映っていない背景画像セット (B) に分割する。
2. (A) から接近灯の座標領域を切り出し、切り出した画像から青色成分を抜き出す。それ以外の領域は透過させる。
3. 抜き出した青色成分を背景画像 (B) に合成し、合成画像セット (C) を得る。

なお、合成する接近灯の数は画像ごとに最大5つ、合成位置はランダムとした。また、合成すると同時に接近灯の位置座標を教師データに書き込むことで、人手でのアノテーションを省き、自動で教師データを作成した。

### 4-3 検証結果

図10に検証結果を示す。今回、撮影データ10,000枚に対し接近灯データ439枚を使用し、撮影データのみの場合と、合成データ10,000枚、20,000枚、30,000枚を撮影データ10,000枚に加えて学習した時の精度を比較した。また、学習のばらつきを考慮し、各条件で5回ずつ試行した。

検証の結果、合成データ10,000～20,000枚を加えて学習させることで、撮影データのみの場合に比べ過検知率が2-5%改善した。これは、作業員1クラスのDNNを、新た



に「接近灯」を加えた2クラスのDNNとすることで、DNNが作業者と接近灯の差を見分けるよう効果的な学習ができたためと考えられる。ただし、撮影データに対し合成データが多すぎると効果が見られず、撮影データとのバランスや、合成データの質が重要であることも判明した（今回の場合1：3）。合成データ作成時、教師データ作成も自動で行うため、学習データ作成効率が非常に高く、今回のように撮影データを集めることなく効率的に過検知率を低減可能であることがわかった。

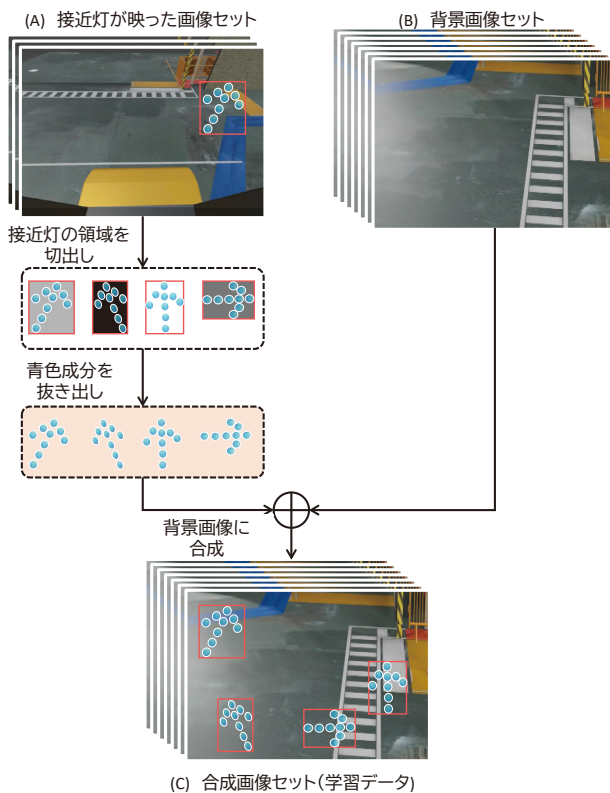


図9 学習データの水増し方法

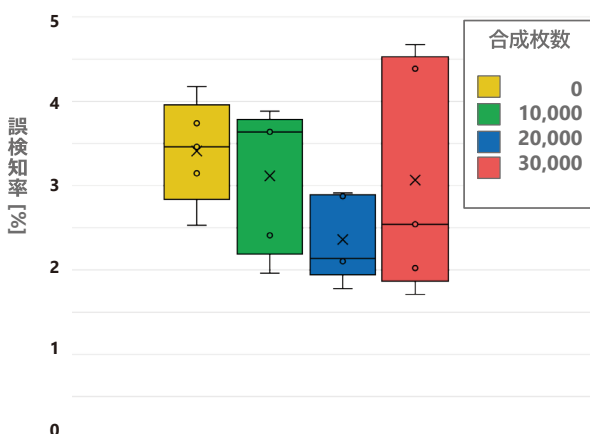


図10 合成枚数ごとの精度比較

## 5. 結 言

エッジコンピュータとエッジAIをフォークリフトに搭載し、検知から発報までリアルタイムに行うフォークリフト安全支援システムを開発した。また、精度と速度を両立するエッジ向けのAIのベンチマーク結果や、効率よく精度向上を図るための学習方法について示した。今後はDNNの精度/速度を更に向上させ、事故防止及び現場の安全確保を継続する。

### 用語集

#### ※1 ディープニューラルネットワーク

AI（人工知能）の1つ。入力データから、複数の積和演算の繰り返しによって出力を得る手法で、演算で用いるパラメータは事前の学習によって自動調整される。従来のニューラルネットワークの中間の演算数が1層なのに対し、ディープニューラルネットワークは2層以上の中間層で構成され、複雑なデータ処理に対応できる。

#### ※2 FPGA

Field Programmable Gate Arrayの略称。ユーザが自由にロジックや配線を書き換え可能な回路。AIのアクセラレータとして一般に利用されるGPUに比べ演算のカスタム性が高く、量子化技術と組み合わせることで電力効率の高いエッジAIを実現可能。

#### ※3 アノテーション

機械学習の一種である「教師あり学習」において、データにラベルなどのメタデータを付与して教師データを作成する作業。

・使用したDNNの実装は、BSDライセンスに基づいています。

### 参考文献

- (1) Wei Liu, "SSD: Single Shot MultiBox Detector," European Conference on Computer Vision 2016, pp.21-37

執 筆 者

田窪 伸哉\* : IoT 研究開発センター 主査



永田 篤樹 : IoT 研究開発センター



猪飼 隼吾 : IoT 研究開発センター



李 漠飛 : IoT 研究開発センター



西岡 進吾 : IoT 研究開発センター 主席



吉村 明展 : IoT 研究開発センター グループ長



\*主執筆者