

物流荷物検出 Convolutional Neural Network の学習における CG 活用の一検証*

A Study on Training a Convolutional Neural Network Using CG for Package Detection in Logistics

石神 裕丈
Hirotake ISHIGAMI

澤田 枝里香
Erika SAWADA

阿波 雅史
Masafumi AWA

A cargo monitoring system for delivery operations, that detects packages using images from cameras installed inside trucks, is under consideration. While the system aims to detect packages during cargo shifts, the low frequency of such incidents poses challenges in collecting sufficient training data. To address this, we propose a Convolutional Neural Network trained on both camera images and CG data simulating cargo shift states and verify its effectiveness.

Keywords :

Logistics, Object Detection, Convolutional Neural Network (CNN), Computer Graphics (CG)

1. まえがき

2024年4月1日からの“自動車運転業務における時間外労働時間の上限規制”の施行に代表されるように、物流業務における業務効率化の社会的ニーズが高まっている。これに対して、中継輸送や共同輸送といった輸送形態の工夫による効率化が検討されている。その一方で、関わる運送者が増え、荷崩れによる破損等の発生場所・原因の特定が困難となるといった、輸送品質の確保の点で懸念が挙がっている。

上記を踏まえて、我々はトラック庫内に設置したカメラの画像にて荷物を検出する、荷物配送業務見守りシステムの実現を検討している。当該システムにおいて、前述の様に輸送品質に関わる荷崩れ発生時の荷物

も検出したいが、その発生頻度が低く（協力会社へのヒアリングでは数か月に1度）、学習データの収集に時間を要する課題がある。

このような学習データ収集困難な事例に対して、Computer Graphics（以降CG）を用いる手法が検討されている¹⁾。先行研究では工場部品の在庫数量自動判定に向けて、ナットをCGで再現して学習した部品数量判定モデルの精度を検証している。

本稿では、データ収集困難な荷崩れ状態を再現したCGを生成して荷物検出用Convolutional Neural Network（以降CNN）を学習し、先行研究では述べられていない個々の物体（荷物）の検出性能を評価し、荷崩れ状態の荷物検出における有効性を検証した。

2. 検証環境

本稿における検証環境として、実際のトラックの1/10サイズを想定したミニチュア環境を構築した。Fig. 1にその外観を、Table 1にカメラ・エッジPCの型番・仕様を示す。トラック庫内を模した箱内の天井部分にカメラを設置している。Fig. 2の様に、当該環境で荷崩れを模擬した評価データを撮影して、画像中視認可能な荷物の検出性能を評価する。

Table 1 Details on equipment in use

| Items | Details |
|---------------|---|
| Edge PC | Nvidia Jetson Nano Developer Kit B01 |
| Camera Imager | Sony IMX477 (Capture at 3280x2464 pixels) |
| Camera Lens | Entaniya RPL-170 (FoV H185° x V130°) |

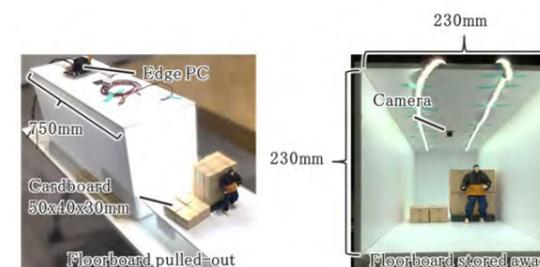


Fig. 1 Appearance of the verification miniature environment

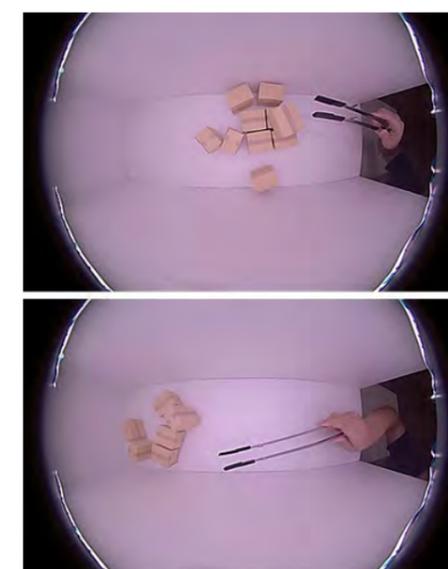


Fig. 2 Examples of captured data for evaluation

3. 荷物検出処理の構成

画像入力から荷物検出力までの処理フロー図をFig. 3に示す。以降の節にて、歪み補正・画像変換処理の詳細、CNN荷物検出処理におけるCNNの構成、学習データの詳細について述べる。

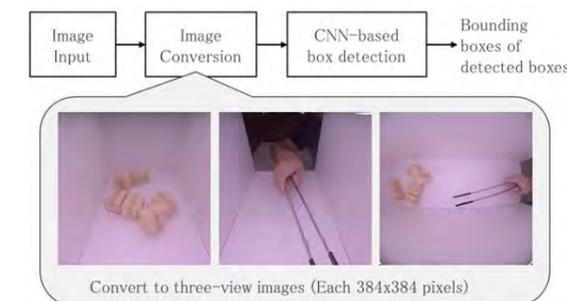


Fig. 3 Flowchart of box detection process

3.1 画像変換処理

本処理導入の動機として、瞬時に発生する荷崩れを捉えるために5fps以上での処理が必要と想定し、CNN荷物検出に入力する画像サイズを抑えたいと考えた。但し、撮影画像をそのまま縮小するとカメラから離れた範囲の荷物も相対的に小さくなり、CNNでの検出が困難となる。これに対して、Fig. 4に示すように、カメラから離れた範囲は水平垂直直角70°の中心射影カメラで撮影した相当の画像に変換し（Fig. 4中の視点#0、#1に対応）、各視点の画像を384x384pixで出力する手法を適用した。

変換に際して必要となる入射光角度に対する像高データは、Fig. 5に示すゴニオステージを用いた計測治具を作成し、ゴニオステージの回転に伴う撮影ターゲット板の初期位置格子点の変位量から計測した。

計測結果をFig. 6に示す。

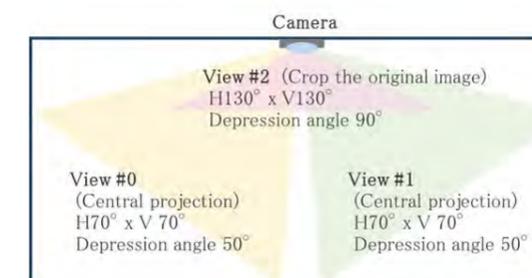


Fig. 4 Details of the Converted Views

* (公社) 精密工学会 画像応用技術専門委員会の子承を得て、「DIA2025 講演論文集」より一部加筆して転載

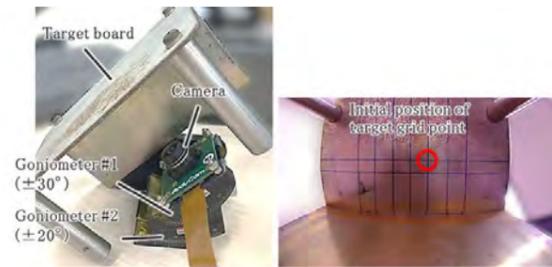


Fig. 5 Jig for measuring image height relative to incident light angle

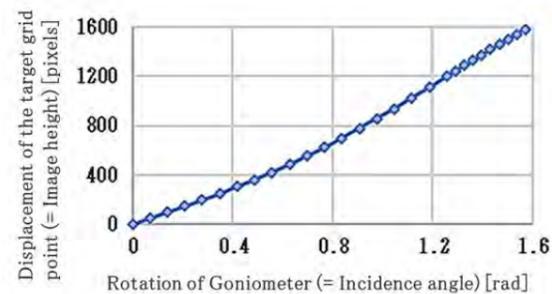


Fig. 6 Results of image height relative to incident light angle

3.2 CNN の構成

検出に用いる CNN は、Nvidia より公開の、ResNet18 を特徴抽出層として用いている DetectNet_v2²⁾ をベースモデルとして、次節で説明する学習データを用いてファインチューニングしている。当該モデルの畳み込み層を抽出した層構成概略図を Fig. 7 に示す。Fig. 7 中の赤矢印に相当する範囲の層をファインチューニングにより重み更新している。

学習には Nvidia 提供の TAO Toolkit (v5.3.0)³⁾ を使用している。当該 Toolkit には学習した CNN モデル中の最終出力に対して寄与度の低いパラメータを削除する “Pruning” 機能が含まれており、当該機能により最終モデルサイズをベースモデルの 5.8% に当たる 5.4MB (ベースモデル 93.3MB) まで削減して、省メモリ・処理の高速化を図っている。

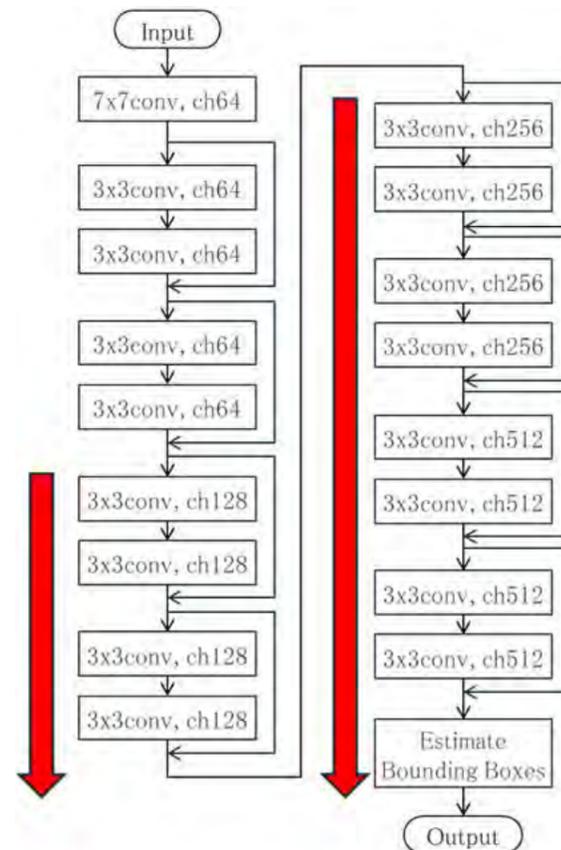


Fig. 7 Configuration diagram of ResNet18-DetectNet_v2 (The red arrow corresponds to the fine-tuning range)

3.3 学習データ

学習データは実際に撮影した通常積載のデータと、発生頻度が低い荷崩れを CG で再現したデータを併用する。

背景で説明したシステムの構成として、通常積載データは撮影して学習データとして収集可能と考える。このため本稿の検討においても、Fig. 8 に示すように、ミニチュア環境で撮影した通常積載相当のデータを学習データの一部として用いる。

さらに、CG で生成した荷崩れデータも学習に用いる。生成ツールとしては Nvidia Omniverse を用いて、Fig. 9 に示すような環境を構築した。当該ツールにおいて、重力加速度の下でランダムな高さ・位置から段ボールの 3D モデルを落下配置させるシミュレーションを実施して、荷崩れ状態を再現している。なお CG データを用いた CNN 学習に際して、CG 特有のシャープな輪郭を過学習してしまう懸念がある。このため、

Fig. 10 に示すように、CG データをベースにボケ・ノイズを付与して輪郭特徴をぼかした画像も生成し、こちらも学習データとして利用している。



Captured 13940 frames (Including 61668 boxes)

Fig. 8 Examples of captured data for training

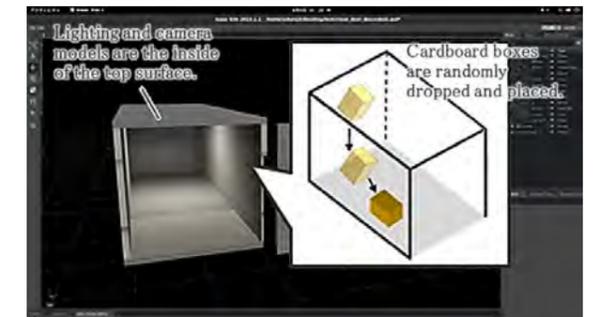
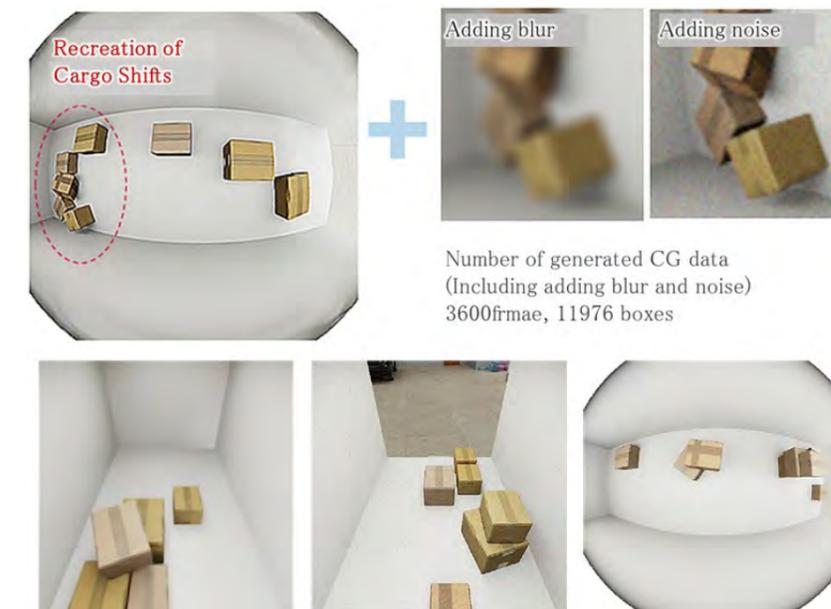


Fig. 9 Environment for generating CG that recreates cargo shifts

4. 結果

2 節で言及した評価データにて、箱の総数を母数とした検出率と、検出枠の誤検出件数を評価した。比較として、CG データを用いない通常積載状態の撮影画像のみを用いた場合も評価している。

評価結果を Tabel 2 に示す。表より、荷崩れを模擬した CG を併用して学習することで検出率が 1.4% 向上して 99.7% となることを確認した。通常積載状態の撮影画像のみ学習した CNN モデルと、CG も併用して学習した CNN モデルとで検出結果に差が生じたシーン例を Fig. 11 に示す。図より、CG データも併用した CNN モデルでは荷崩れシーンに特有の傾いた段



Number of generated CG data (Including adding blur and noise) 3600frames, 11976 boxes

Fig. 10 Examples of CG data for training

ボールで一部遮蔽されている段ボールも検出できている。これは崩れ方により様々な遮蔽パターンが生じる状況をCGで再現して学習できたためと想定する。

また Fig. 12 に、今回の評価において両手法共で誤検出が発生したシーンを示す。崩れ発生前の積み上げ状態において段ボールの境界に多重枠が発生している。これは検出対象がシンプルな模様のない段ボールであり、特徴とする輪郭のエッジに強く反応して多重枠として検出してしまったケースと考える。本ケースは、検出枠の重複量を考慮した適切なクラスタリングアルゴリズムの後処理により対処可能と考える。

最後に一連の処理のエッジ PC での処理周期を計測した。結果は最悪値 8fps (8 ~ 9fps 範囲) であり、荷崩れを捉えるために必要と想定する 5fps 以上での処理が可能であることを確認した。

Table 2 Result of Detection accuracy and False positive count

| | Trained on captured data during normal loading + CG data | Trained solely on captured data during normal loading |
|------------------------|--|---|
| Detection accuracy [%] | 99.7 | 98.3 |
| False positive count | 5 | 5 |

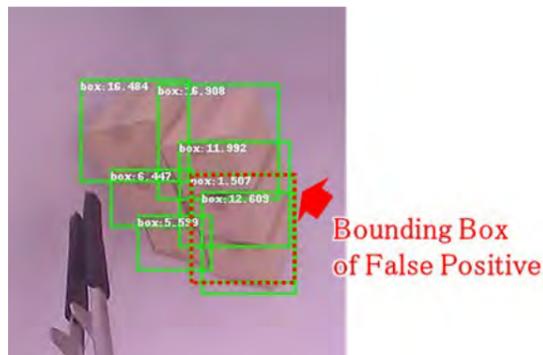


Fig. 12 An example of scene where false positive occurred

5. むすび

本稿では、トラック庫内に設置したカメラの画像における CNN による荷物検出において、発生頻度が低く学習データの収集に時間を要する荷崩れデータをCGで再現して、撮影した通常積載状態の画像と併せて学習した際の検出性能を評価し、その有効性を検証した。結果として、通常積載状態の撮影データのみを学習した CNN モデルよりも検出率が 1.4% 向上して 99.7% となり、荷崩れ特有の傾いた段ボールで一部遮蔽されている段ボールも検出できる点から有効性がある

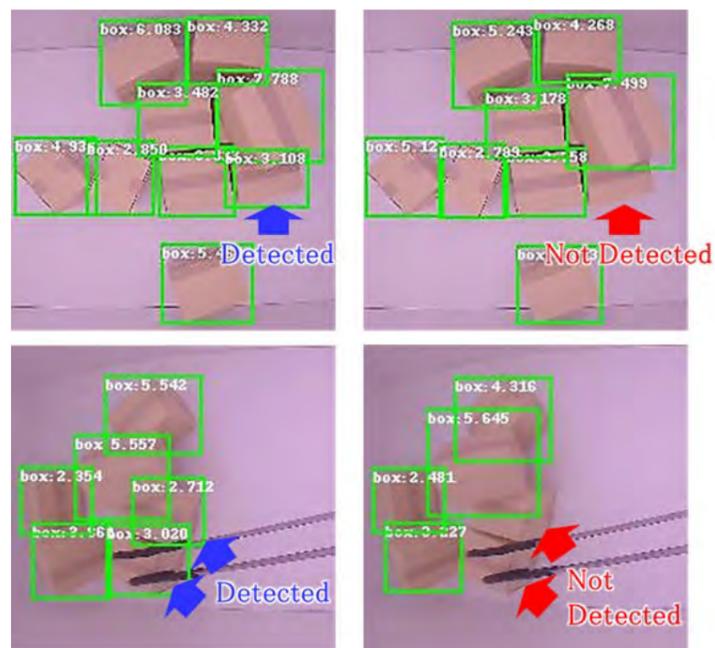


Fig. 11 Examples of scenes where detection results differed

ると考える。また、一連の処理をエッジ PC に実装しての処理周期計測も実施し、想定するシステムで必要となる 5fps を上回る 8fps で処理が可能であることを確認した。

本システムを実際のトラックに設置しての効果確認が今後の検証課題となる。

参考文献

- 1) 滝本廉, 工藤司. CG による学習データを活用した在庫把握方式. 電子情報通信学会 2019 年総合大会, D-17-3, 2019.
- 2) Nvidia, "TAO Pretrained DetectNetV2", https://catalog.ngc.nvidia.com/orgs/nvidia/teams/tao/models/pretrained_detectnet_v2, (参照2025-01-16)
- 3) Nvidia, "TAO Toolkit v5.3.0", <https://docs.nvidia.com/tao/archive/5.3.0/index.html>, (参照2025-01-16)

著者



石神 裕丈
いしがみ ひろたけ
株式会社 SOKEN 研究 2 部
画像認識・信号処理アルゴリズムの研究開発に従事



澤田 枝里香
さわだ えりか
社会イノベーション事業開発統括部
非モビリティ領域の新事業開発に従事



阿波 雅史
あわ まさふみ
社会イノベーション事業開発統括部
非モビリティ領域の新事業開発に従事