情報処理学会論文誌 Vol.67 No.3 1–11 (Mar. 2026)

変形ARマーカの3次元位置・姿勢推定と組み込みボードへの実装

浅野 右京[1,a)] 山内 悠嗣[1,b)]

受付日 2025年5月5日, 採録日 2025年12月9日

概要：近年，2 次元コードはキャッシュレス決済やロボットの自己位置推定など多様な用途で利用されている．しかし，2 次元コードを貼付する対象は平面であることが前提であり，円柱や非剛体などに貼り付けると変形し，認識や位置・姿勢推定が難化する．変形を除去し，データベースと照合を行うことで姿勢を推定する従来法が提案されたが課題があった．そこで本研究では，変形した AR マーカの認識と 3 次元位置・姿勢推定の高精度化・高速化を目的とし，従来法の課題である検出性能，姿勢推定精度，推定時間の問題に対し，複数の改良を提案する．提案手法では，物体検出モデルの変更や回帰の導入，変形を除去するモデルと姿勢を推定するモデルの交互最適化により，位置推定精度は従来法から 26.9%，姿勢推定精度は従来法から 62.7%向上した．また，実用化を見据え，提案手法を組み込みボード上に実装し，メモリ使用量が 2.0 [GB] と少量ながら，10.09 [fps] で動作することを確認した．

キーワード：AR マーカ，物体検出，姿勢推定，深層学習

6DoF Pose Estimation of Deformed AR Markers and Implementation on Embedded Boards

Ukyo Asamo[1,a)] Yuji Yamauchi[1,b)]

Received: May 5, 2025, Accepted: December 9, 2025

Abstract: In recent years, two-dimensional codes have been utilized for various purposes such as cashless payments and robot self-localization. However, these codes are designed under the assumption that they will be attached to flat surfaces, and they become deformed when placed on cylinders or non-rigid objects, making recognition and pose estimation more difficult. Previous methods that remove deformation and match with a database to estimate pose have been proposed; however, they have limitations. This study aims to improve the accuracy and speed of recognition and three-dimensional position/pose estimation of deformed AR markers. We propose several improvements to address the issues of detection performance, pose estimation accuracy, and processing time in conventional methods. Our proposed approach involves changing the object detection model, introducing regression, and alternately optimizing the deformation removal and pose estimation models. Compared to conventional methods, our proposed approach improves the position and pose estimation accuracies by 26.9% and 62.7%, respectively. Furthermore, to verify for practical implementation, we implemented the proposed method on an embedded board, confirming that it operates at 10.09 [fps] while using only 2.0 [GB] of memory.

Keywords: AR marker, object detection, pose estimation, deep learning

1. はじめに

近年，2 次元コードが普及し，キャッシュレス決済やロボットの自己位置推定など様々な用途で利用されている．しかし，2 次元コードを貼付する対象は平面であることが前提であり，円柱や非剛体などに貼り付けると変形し，認

[1] 中部大学大学院工学研究科
Graduate School of Engineering, Chubu University, Kasugai, Aichi 487–8501, Japan
[a)] tr24001-2371@sti.chubu.ac.jp
[b)] yuu@fsc.chubu.ac.jp

識や位置・姿勢推定が難化する．この問題を解決するために，変形した 2 次元コードを認識する手法 [1], [2], [3] がいくつか提案されている．そのうちの 1 つである従来法 [3] では，2 次元コードの一種である AR マーカを取り扱い，機械学習による変形 AR マーカの認識と 3 次元位置・姿勢推定法を提案した．従来法では，まず Single Shot Multibox Detector（SSD）[4] で変形 AR マーカの検出と ID の推定を行い，Augmented Autoencoder（AAE）[5] で検出した変形 AR マーカから背景と変形を除去した情報である潜在変数を取得する．最後に，潜在変数をあらかじめ作成したデータベースと照合することで姿勢を推定する．この従来法には 3 つの課題がある．1 つ目は検出性能であり，AR マーカの不規則かつ複雑な変形により，検出が困難な場合がある．2 つ目は姿勢推定精度である．Roll は回転角度に対して画像上の見えの変化が大きいため，ある程度の姿勢推定精度が担保されるが，Pitch と Yaw は回転角度に対しての見えの変化が小さいため姿勢推定精度が低下する．3 つ目は推定に要する時間である．先行研究では，数百次元の実数ベクトルとして表現される潜在変数とデータベースとの照合が行われるため姿勢推定に時間を要する．

そこで，本研究では 3 つの課題を解決するために次の提案と改良を行う．

- 物体検出モデルの変更
  物体検出モデルを SSD から NanoDet-Plus [6] に変更する．これにより変形 AR マーカの高精度な検出，位置推定を実現する．
- AAE の拡張
  変形を除去する AAE に連続性を考慮可能な Variational Autoencoder [7] を導入した Augmented Variational Autoencoder（AVAE）を提案する．AVAE により姿勢推定に適した潜在変数を取得する．
- 回帰による姿勢推定
  従来法では潜在変数とデータベースの照合処理に多大な計算時間を要していた．そこで，本研究では Multi-Layer Perceptron（MLP）を用いた回帰による姿勢推定に変更することで計算量を大幅に削減する．
- 2 つのモデルの交互最適化
  提案手法は，AVAE と MLP の 2 つのモデルで構成されている．AVAE と MLP を同時に最適化することが難しいため，交互に最適化するフレームワークを導入する．

また，モバイルデバイスや組み込みボードなどに実装し，実用化を想定している．そこで，提案手法を NVIDIA 製の組み込みボード Jetson Orin Nano [8] 上に実装し，処理速度とリソース消費量の検証を行う．

2. 関連研究

本研究は，物体検出や姿勢推定の技術を基盤とし，変形した 2 次元コードの 3 次元位置，姿勢を推定する．関連する 3 つの主要分野である 2 次元コードの認識，物体検出，姿勢推定の先行研究について述べる．

2.1 2 次元コードの認識

2 次元コードは，垂直方向と水平方向の 2 方向に配置された色のパターンで情報を表現したコードであり，カメラやスキャナで読み取ることでコードに含まれた情報を取得することが可能である．QR コード [9] は現代において最も普及しているコードの 1 つであり，白黒のドットパターンで情報を表現している．QR コードの位置を表すファインダパターンにより，あらゆる方向からでも認識が可能である．また，アライメントパターンにより歪みをある程度補正できる．AR マーカは拡張現実（AR）のアプリケーションやロボティクス分野で使用される 2 次元コードである．AR マーカは多種多様であり，白黒かつ正方形のマーカ以外にも円形や多色なものが存在している．その中でも ARToolKit [10] をベースにした多くのマーカが開発されており，ARTag [11] や AprilTag [12]，ArUco [13] などがある．

変形した 2 次元コードを認識する研究として，歪んだ 2 次元コードの復号手法 [1], [2] が提案されている．これらの研究は，QR コードに対して着色された補助線を引くことで歪みの補正を可能とした．また，DeepFormableTag [14] のような 2 次元コードの生成から，検出，情報の復号までの処理をすべて End-to-End で学習する手法も提案されている．

2.2 物体検出

物体検出は，画像や動画内において特定の物体の位置の検出とその物体の種類を分類するタスクである．現在では深層学習を用いた物体検出手法が主流となっており，これらの手法は大きく 2 つに分類される．1 つ目は，2 段階の処理を行う物体検出手法である．R-CNN [15]，Fast R-CNN [16]，Faster R-CNN [17] などが該当し，候補領域の検出とクラス分類の 2 つの処理で構成されており，高精度な検出が可能であるものの処理時間に課題がある．2 つ目は，単一のモデルで物体検出を行う手法である．SSD [4] や YOLO [18] などが該当し，候補領域の検出と分類を同時に行うことで，処理速度が大幅に向上し，リアルタイムでの物体検出が可能である．Faster R-CNN，SSD，YOLO は，事前に定義した矩形枠であるアンカーボックスを使用している．これは，特徴マップ上に複数のサイズやアスペクト比で配置され，物体の写る領域を示すバウンディングボックスを推定する際の基準となる．また，CornerNet [19] や CenterNet [20] のようなアンカーボックスを使用しない手法が提案されている．これらの手法はアンカーフリーの検出手法と呼ばれ，物体の中心点や境界を直接推定する手法であり，アンカーボックスに関するパラメータ調整が不

要かつ計算効率が高い．

近年では，Transformer [21] を導入した物体検出手法が注目されている．DETR [22] は，Transformer を初めて取り入れた End-to-End の物体検出手法であり，物体が密集した複雑なシーンにおいても高精度な検出が可能である．この手法は，大きな物体に対する検出精度は高いが，小さい物体の検出精度が低く，画像内のすべての画素間の関係性を計算するため処理コストが高い．この課題を解決した Deformable DETR [23] は，Deformable Attention により注目すべきリファレンスポイントの周辺に限定して情報を取得するため計算効率が高い．また，マルチスケールにも対応可能であり，小さな物体の検出精度を改善している．

2.3 姿勢推定

姿勢推定は，画像内の特定の物体の位置や姿勢を推定するタスクである．単一の RGB 画像から深層学習により，位置・姿勢を推定する研究がさかんに取り組まれている．PVNet [24] は，特徴点ベースの姿勢推定法であり，深層学習により画像中の物体のキーポイントを推定し，物体の 3D モデルのキーポイントとマッチングすることで姿勢を推定する．この手法は，物体のテクスチャが豊かな場合は高速かつ正確に姿勢を推定することができるが，テクスチャが乏しい場合には検出できるキーポイントが減少し精度が低下するという課題がある．

Augmented Autoencoder（AAE）[5] は，テンプレートベースの姿勢推定法である．この手法は，3D モデルの様々な姿勢の特徴量をテンプレートとして作成し，物体の特徴量をテンプレートと照合することで姿勢を推定する．この手法は，テクスチャが乏しい場合においてもテンプレートの数に比例して精度を確保できる．一方で，照合処理が必要となるため処理時間はテンプレートの数に反比例する．特徴点ベースやテンプレートベースは，画像中の物体の特徴を得るために間接的に深層学習を利用することが多い．

一方で，画像から深層学習により直接姿勢を推定する手法が研究されている．SSD-6D [25] は，分類ベースの姿勢推定法であり，姿勢を離散化することで分類問題として姿勢を推定する．分類ベースの姿勢推定法は，対称性を持つ物体を適切に処理できるという特徴がある．対称性を持つ物体とは，特定の軸周りの回転や球のような異なる角度でも同じ姿勢に見える物体を指す．分類ベースの姿勢推定法は，対称性を持つ物体の様々な姿勢を同じクラスとして扱うことで，視覚的に区別できない姿勢を学習することが可能である．しかし，姿勢を細かく離散化すると膨大なクラス姿勢数となるため学習が収束しない．一方，粗に離散する場合には正確な姿勢の推定ができない．PoseCNN [26] や DeepIM [27] は，回帰ベースの姿勢推定法であり，回帰により直接姿勢を推定する．PoseCNN は，物体のセマンティックラベリング，位置推定，姿勢推定の 3 つのタスクに分解して処理する手法である．CNN から得られた特徴マップとセマンティックラベリングから位置を推定し，特徴マップと位置から回帰により姿勢を推定する．DeepIM は，入力画像から回帰により位置と姿勢を推定し，その結果から物体の 3D モデルのレンダリングを行う．そして，レンダリング画像と入力画像を比較することで反復的に姿勢を推定する．回帰ベースは，姿勢の変化の連続性をとらえることが可能であり，学習データが不均衡である場合の影響を受けにくい．また，計算コストが低く高速に処理できる．しかし，対称性を持つ物体は正解姿勢が複数あるため，複数の正解姿勢の中間値に収束してしまう傾向がある．

3. 提案手法

本稿では，従来法の課題を解決した変形 AR マーカの高速，高精度な 3 次元位置・姿勢推定法を提案する．**図 1** に提案手法の流れを示す．まず，NanoDet-Plus [6] により画像中の変形 AR マーカのバウンディングボックスと AR マーカの種類を表す ID を推定する．次に，検出した変形 AR マーカを Augmented Variational Autoencoder（AVAE）のエンコーダに入力し，背景と変形を除去した潜在変数を取得する．そして，得られた潜在変数を Multi-Layer Perceptron（MLP）に入力することで姿勢を推定する．最後に，変形

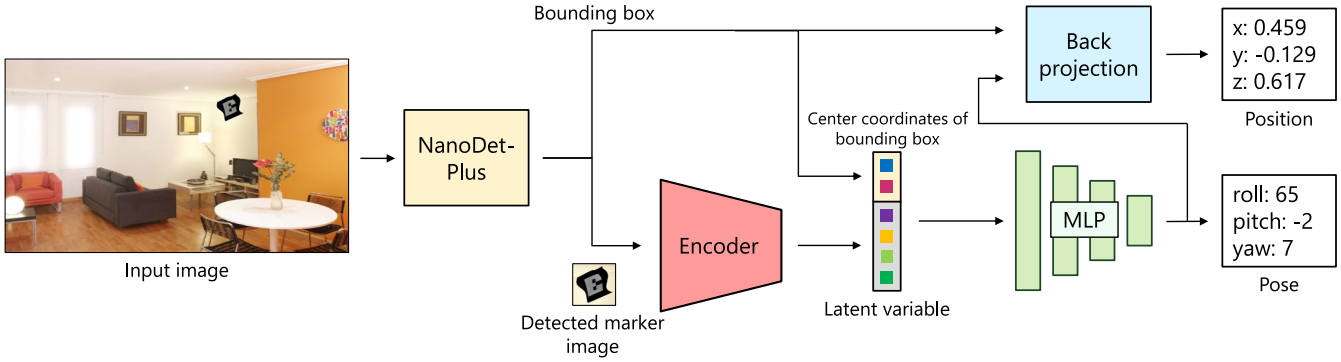

図 1 提案手法の流れ

Fig. 1 Flow of the proposed method.

AR マーカのバウンディングボックスと推定姿勢から逆射影変換により位置を推定する．

3.1 変形 AR マーカの検出

提案手法では，画像から変形 AR マーカを検出するために使用する物体検出モデルとして NanoDet-Plus [6] を採用する．NanoDet-Plus は，異なるスケールの物体を検出可能にする Feature Pyramid Network [28] を軽量化した Ghost-PAN や対象と非対象の物体クラスの不均衡問題を解決した損失関数 Generalized Focal Loss により軽量かつ高精度な物体検出を実現した．NanoDet-Plus は，クラス内における画像の見えの多彩さにも十分に対応しているため，複雑に変形した AR マーカの高精度な検出が期待できる．また，パラメータ数が少なく，低スペックのデバイスでも動作可能である．

NanoDet-Plus では，検出と同時に AR マーカの ID 分類について学習する．一方で，マーカ検出のみ機械学習で行い，切り取ったマーカ画像を解析して ID を分類するハイブリッドな手法も考えられる．提案手法は，マーカが大きく変形した場合，従来の解析手法による ID 分類は困難になるが，提案手法は変形した状態のまま特徴を学習するため，変形に対して頑健な ID 分類が期待できる．

3.2 変形 AR マーカの 3 次元姿勢推定

NanoDet-Plus を用いて検出された変形 AR マーカから AVAE のエンコーダと MLP により姿勢を推定する．

3.2.1 Augmented Variational Autoencoder

Augmented Autoencoder（AAE）[5] はオートエンコーダ [29] を拡張し，入力画像のノイズを除き本質的な情報のみを潜在変数に抽出することを目的としている．AAE のベースとなるオートエンコーダの学習では，入力データをエンコーダにより低次元のベクトルとして表現される潜在変数に圧縮後，デコーダにより入力画像と同じデータに復元する．これに対し AAE は，画像内の背景やオクルージョンなどのノイズを除去したうえで，対象となる物体に限定して特徴を抽出することを目的としたオートエンコーダである．入力画像をノイズを含む画像，ターゲット画像をノイズのない画像とし，入力画像からエンコーダとデコーダにより復元される画像がターゲット画像に近づくように学習する．本研究では，AR マーカの背景や変形，環境の変化をノイズとして除去することで，AR マーカのパターンや姿勢などの本質的な情報のみを抽出するように学習する．

本研究では，オートエンコーダ型の AAE を Variational Autoencoder（VAE）[7] 型に拡張した Augmented Variational Autoencoder（AVAE）を用いる．VAE は，入力データの背後にある潜在的な確率分布を学習し，入力データの法則性に基づく潜在変数の獲得や新たなデータの生成を可能としたオートエンコーダである．変形 AR マーカの見えの変化と姿勢は密接な関係にある．そこで，姿勢の連続性を潜在変数により表現することを目的として AVAE を採用する．なお，AVAE のエンコーダとデコーダは ResNet-18 [30] をベースとする．入力画像 $\boldsymbol{x}$ を AVAE に入力した際，復元画像 $\hat{\boldsymbol{x}}$ がターゲット画像 $\boldsymbol{y}$ に近づくように学習しながら，式 (2) に示すように潜在変数 $\boldsymbol{z}$ が事前分布に沿うように学習される．

$$\hat{\boldsymbol{x}} = (\Psi \circ \Phi)(\boldsymbol{x}) = \Psi(\boldsymbol{z}) \tag{1}$$

$$\boldsymbol{z} = \boldsymbol{\mu}(\boldsymbol{x}) + \boldsymbol{\sigma}(\boldsymbol{x}) \cdot \boldsymbol{\epsilon} \tag{2}$$

ここで，Φ はエンコーダ，Ψ はデコーダ，$\boldsymbol{z} \in \mathbb{R}^{256}$ である．潜在変数はエンコーダ Φ から得られた平均ベクトル $\boldsymbol{\mu}(\boldsymbol{x})$ と標準偏差ベクトル $\boldsymbol{\sigma}(\boldsymbol{x})$，および標準正規分布に従う乱数である $\boldsymbol{\epsilon}$ を用いて Reparameterization Trick により計算される．AVAE は，背景や変形の除去を主としながら姿勢の連続性を考慮した潜在変数の取得が期待できる．

3.2.2 Multi-Layer Perceptron

潜在変数から姿勢を推定するために Multi-Layer Perceptron（MLP）を導入する．MLP は，入力層，中間層 2 層，出力層の計 4 層で構成されている．入力層のノード数は 258，中間層 1 層目のノード数は 129，中間層 2 層目のノード数は 64，出力層のノード数は 4 であり，各層の間に活性化関数 Exponential Linear Unit（ELU）を配置した構造である．また，出力層の後に HardTanh$(0, 1)$ を適用し，最終的な出力値を $[0, 1]$ の範囲に制約した．カメラで物体を撮影した場合，物体の姿勢が同一であっても画像上の位置によって見え方が大きく変化する．そのため，バウンディングボックスの中心座標を MLP に入力することで，変形 AR マーカの画像上の位置による見えの変化を考慮する．MLP への入力は，AVAE より得られる潜在変数 256 次元に，NanoDet-Plus で検出した変形 AR マーカの中心座標 2 次元を連結した計 258 次元のベクトルとする．MLP の出力は，変形 AR マーカの姿勢情報であり，Roll を 2 次元，Pitch と Yaw を各 1 次元の計 4 次元のベクトルである．本研究では，変形 AR マーカの姿勢範囲を Roll は $[0\,\mathrm{deg}, 359\,\mathrm{deg}]$，Pitch と Yaw は $[-13\,\mathrm{deg}, 13\,\mathrm{deg}]$ とし，MLP から出力された $[0, 1]$ の値を式 (3)，(4)，(5) により変換する．

$$r_{angle} = \arctan2(2r_2 - 1, 2r_1 - 1) \tag{3}$$

$$p_{angle} = p \cdot (\theta_{p,max} - \theta_{p,min}) + \theta_{p,min} \tag{4}$$

$$y_{angle} = y \cdot (\theta_{y,max} - \theta_{min}) + \theta_{y,min} \tag{5}$$

ここで，MLP より出力された 2 次元の Roll を r_1 と r_2，変換した Roll を r_{angle} と表す．同様に，MLP より出力された各 1 次元の Pitch と Yaw を p，y，Pitch と Yaw の姿勢範囲を $[\theta_{p,min}, \theta_{p,max}]$，$[\theta_{y,min}, \theta_{y,max}]$，変換した Pitch と Yaw を p_{angle}，y_{angle} と表す．式 (3) による Roll の変換では，MLP から 2 次元で出力された $[0, 1]$ の値を $[-1, 1]$

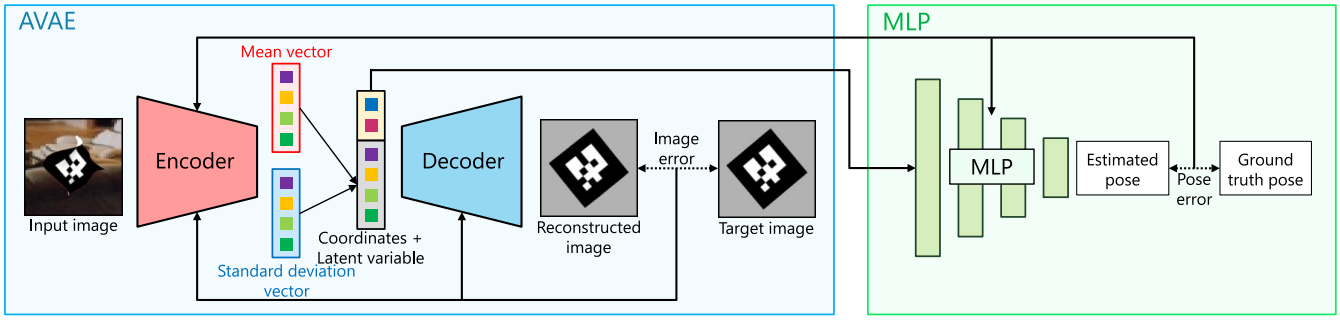

図 2 2 つのモデルの交互最適化の流れ

Fig. 2 Flow of the alternating optimization between the two models.

にスケーリングし，arctan2 により 1 次元の $[0, 359]$ に変換する．Roll のような周期性を持つデータを 1 次元で出力すると，本来連続である 0 [deg] と 359 [deg] が数値的に離れた値となり，損失計算において不連続性の問題が生じる．そこで，提案手法では，Roll の角度 θ を $(\cos\theta, \sin\theta)$ の 2 次元で出力し，1 次元に変換している．式 (4)，(5) による Pitch と Yaw の変換では，MLP から出力された $[0, 1]$ の値を $[-13, 13]$ に直接スケーリングする．

3.2.3 2 つのモデルの交互最適化

AVAE では変形除去，MLP では姿勢推定を行う目的でモデルが構成されており，2 つのモデルを同時に最適化することができない．そこで，提案手法では，2 つのモデルを交互に最適化する．

交互最適化の流れを図 2 に示す．AVAE を構成するエンコーダとデコーダの最適化では，まず変形 AR マーカ画像を AVAE のエンコーダに入力して潜在変数を得る．次に，潜在変数から AVAE のデコーダにより，背景や変形を除去した AR マーカ画像を復元する．そして，復元画像とターゲット画像の誤差 L_{AVAE} を式 (6) により計算し，AVAE のエンコーダとデコーダの重みを更新する．

$$L_{AVAE} = L_{rc} + \beta D_{KL} \tag{6}$$

$$L_{rc} = \frac{1}{2}\sum_{i=1}^{n}(\hat{x}_i - x_i)^2 \tag{7}$$

$$D_{KL} = -\frac{1}{2}\sum_{j=1}^{d}(1 + \log\sigma_j^2 - \mu_j^2 - \sigma_j^2) \tag{8}$$

ここで L_{rc} は，復元画像 $\hat{x}$ とターゲット画像 x の二乗和誤差を表す．D_{KL} は，潜在変数の分布が正規乱数に沿うように制御する正則化項であり，μ が潜在変数の平均，σ が潜在変数の標準偏差を示す．また，β は正則化の強さを調整するためのハイパーパラメータである．

エンコーダと MLP の最適化では，変形 AR マーカ画像を AVAE のエンコーダに入力して潜在変数を得る．次に潜在変数を MLP に入力して姿勢を推定する．そして，推定した姿勢と正解姿勢の姿勢誤差 L_{pose} を式 (9) により計算し，AVAE のエンコーダと MLP の重みを更新する．

$$L_{pose} = \lambda\left(\frac{|\hat{r}_1 - r_1| + |\hat{r}_2 - r_2|}{2} + |\hat{p} - p| + |\hat{y} - y|\right) \tag{9}$$

ここで，$(\hat{r}_1, \hat{r}_2)$ は Roll の推定姿勢，$\hat{p}$ は Pitch の推定姿勢，$\hat{y}$ は Yaw の推定姿勢，(r_1, r_2) は Roll の正解姿勢，p は Pitch の正解姿勢，y は Yaw の正解姿勢である．また，λ は AVAE のエンコーダに対する損失を調整するためのハイパーパラメータである．

2 つのモデルの最適化を交互に繰り返すことで，AVAE のエンコーダでノイズとなる背景，変形の情報を除去しながら，姿勢推定に必要な情報のみを表現する潜在変数の抽出が可能となる．

3.3 変形 AR マーカの 3 次元位置推定

変形 AR マーカの 3 次元位置は，NanoDet-Plus によって得られたバウンディングボックスと推定姿勢に基づいて逆射影変換により計算される．まず，事前にカメラキャリブレーションを行い，逆射影変換に必要なカメラパラメータを取得する．また，逆射影変換時の基準として，シミュレータ上のカメラパラメータと AR マーカの奥行方向の距離を 0.6 [m] に設定した場合のバウンディングボックスの大きさを記録する．次に，変形 AR マーカの奥行方向の距離 $\hat{z}_{tg}$ を式 (10) により計算する．

$$\hat{z}_{tg} = z_{bs} \times \frac{bb_{bs}}{bb_{tg}} \times \frac{f_{tg}}{f_{bs}} \tag{10}$$

ここで，z_{bs} と bb_{bs}，f_{bs} はそれぞれ基準となる奥行方向の距離とバウンディングボックスの対角線の長さ，焦点距離を表す．bb_{tg} と f_{tg} は変形 AR マーカにおけるバウンディングボックスの対角線の長さと焦点距離を表す．そして，変形 AR マーカの 3 次元位置 $\hat{\boldsymbol{t}}_{tg} = (\hat{x}_{tg}, \hat{y}_{tg}, \hat{z}_{tg})$ を式 (11)，(12) より求める．

$$\Delta\hat{\boldsymbol{t}} = \hat{z}_{tg}\boldsymbol{K}_{tg}^{-1}\boldsymbol{bb}_{tg,c} - z_{bs}\boldsymbol{K}_{bs}^{-1}\boldsymbol{bb}_{bs,c} \tag{11}$$

$$\hat{\boldsymbol{t}}_{tg} = \boldsymbol{t}_{bs} + \Delta\hat{\boldsymbol{t}} \tag{12}$$

ここで，$\boldsymbol{K}_{tg}$ と $\boldsymbol{bb}_{tg,c}$ は変形 AR マーカを撮影した際のカメラ行列とバウンディングボックスの中心座標，$\boldsymbol{K}_{bs}$ と $\boldsymbol{bb}_{bs,c}$ は基準となるカメラ行列とバウンディングボックスの中心座標を表す．

3.4 学習データセットの作成

変形 AR マーカの検出モデルおよび，位置・姿勢推定モデルの学習には大量のデータが必要となる．実環境で変形した AR マーカを撮影しアノテーションを付与するには多大な労力を要する．そこで，本研究ではシミュレータを用いることで学習データを自動的に作成する．

AR マーカには Robot Operating System（ROS）の ar_track_alvar パッケージ [31] を用いる．使用する AR マーカを**図 3** に示す．本研究には 10 種類の AR マーカを使用し，AR マーカの 1 辺の大きさは 50 [mm] とする．AR マーカの変形はベジェ曲面により与える．ベジェ曲面はベジェ曲線を 2 次元に拡張したものであり，ベジェ曲面の制御点の位置により複雑な変形を表現できる．本研究では制御点を**図 4** (a) のように x 軸，y 軸方向に 7 点ずつ，計 49 点を配置する．そして，制御点の x 軸，y 軸方向の位置を固定し，z 軸方向の位置を正規乱数に従い与える．これにより，正規乱数の標準偏差により変形度合いの調節が可能となる．変形の標準偏差は，$[0.2, 2.0]$ の範囲を 0.2 刻みで設定する．図 4 (b) に変形の標準偏差を 2.0 に設定した際の制御点と変形 AR マーカの関係を示す．制御点の位置に従い，AR マーカが滑らかに変形していることが確認できる．

本研究で生成する変形であるベジェ曲面は，布や紙などの非剛体物が緩やかにたわんだり，ねじれたりする状況を模擬している．これは，人の衣服にマーカを貼付することによるトラッキングや物が入った袋のような非剛体物をロボットがピッキングすることを想定している．実問題への変形の妥当性を確認するために，シミュレータでのベジェ曲面と実環境での軟包装における変形を定性的に比較した画像を**図 5** に示す．

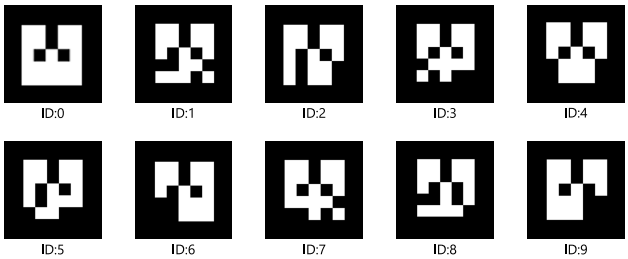

図 3 学習に使用する AR マーカ

Fig. 3 AR markers used for training.

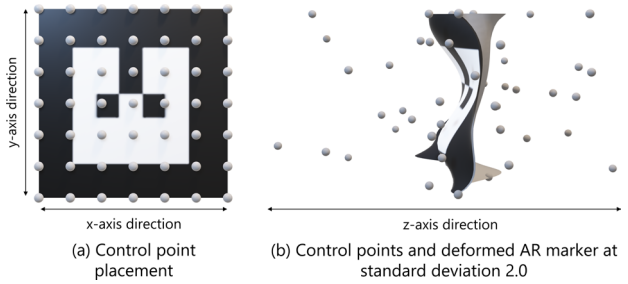

図 4 制御点の配置例と変形した AR マーカの例

Fig. 4 Examples of control point placement and deformed AR markers.

変形 AR マーカの 3D モデルには，3DCG 作製ソフトウェアである Blender を利用する．作製した変形 AR マーカの例を**図 6** に示す．作製した変形 AR マーカの 3D モデルを使用し，変形 AR マーカの画像をシミュレータ上で撮影する．変形 AR マーカがとりうる位置と姿勢は特定の範囲内でランダムに決定する．撮影環境の概要を**図 7** に示す．カメラの位置を原点とした際に，位置の範囲は x を $[-0.25\,\mathrm{m}, 0.25\,\mathrm{m}]$，y を $[-0.15\,\mathrm{m}, 0.15\,\mathrm{m}]$，z を $[0.4\,\mathrm{m}, 0.8\,\mathrm{m}]$ とする．姿勢範囲は，Roll を $[0\,\mathrm{deg}, 359\,\mathrm{deg}]$，Pitch を $[-13\,\mathrm{deg}, 13\,\mathrm{deg}]$，Yaw を $[-13\,\mathrm{deg}, 13\,\mathrm{deg}]$ とする．

撮影には Gazebo シミュレータを使用する．この方法により撮影した 1,920 × 1,080 [pixel] の画像とアノテーションデータを 1 セットとし，変形 AR マーカを検出する NanoDet-Plus の学習に使用するために 55,000 セットを用意する．2 つのモデルの交互最適化では，撮影した画像をア

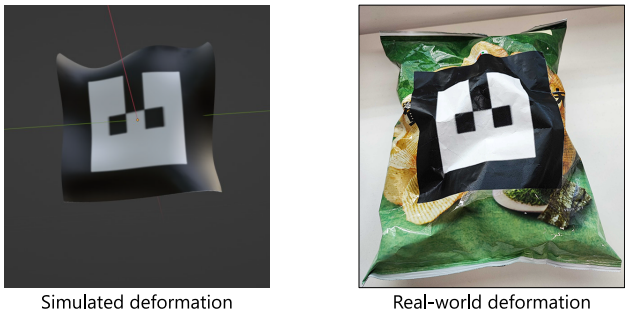

図 5 シミュレータの変形と実環境の変形の比較

Fig. 5 Comparison between simulated and real-world deformation.

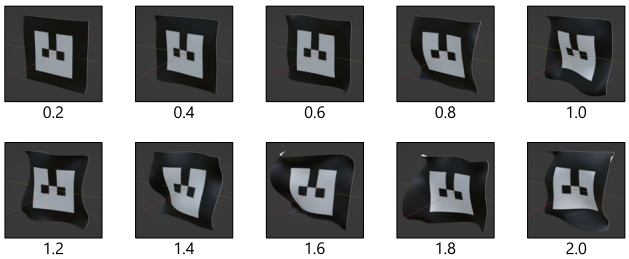

図 6 各標準偏差ごとの変形 AR マーカ

Fig. 6 Deformed AR markers for each standard deviation.

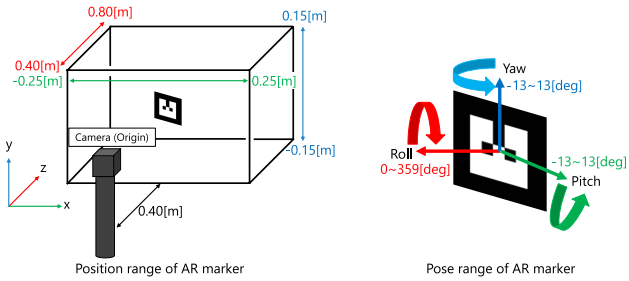

図 7 AR マーカの位置・姿勢範囲

Fig. 7 Range of position and pose of the AR marker.

ノテーションデータにより，変形 AR マーカを中心とした 128×128 [pixel] の画像に切り取ったものを使用する．入力画像，入力画像に対するアノテーションデータ，ターゲット画像の 3 つを 1 セットとして 22,000 セットを使用する．

4. 評価実験

提案手法の有効性を確認するために，従来法と提案手法で変形 AR マーカの検出性能，位置推定精度，姿勢推定精度の 3 つを比較する．本稿における変形 AR マーカの真の位置，姿勢とは，変形する前の平面状態の AR マーカの 3 次元的な位置，姿勢と定義する．

4.1 変形 AR マーカの検出性能

提案手法で使用する NanoDet-Plus と従来法で使用された SSD で変形 AR マーカの検出性能を比較する．評価用画像には，シミュレータ上で撮影された 11,000 枚の変形 AR マーカ画像を使用する．評価指標として，変形 AR マーカの検出精度には mean Average Precision（mAP），バウンディングボックスの正確性には Intersection over Union（IoU）を採用する．

変形 AR マーカの検出性能の結果を**表 1** に示す．mAP では SSD の 0.80 に対し，NanoDet-Plus は 0.96 であり，mAP が 0.16 向上した．また，IoU では SSD の 0.88 に対し，NanoDet-Plus では 0.96 であり，IoU が 0.08 向上した．

4.2 変形 AR マーカの位置推定精度

位置の推定精度を平均絶対誤差により比較する．評価には，シミュレータ上で撮影した画像から NanoDet-Plus により検出した 11,000 枚の変形 AR マーカ画像を使用する．

表 2 に x，y，z の位置推定誤差の結果を示す．提案手法の平均誤差は 4.90 [mm] であり，従来法 [3] の平均誤差 6.69 [mm] と比べて 26.9%減少した．これは，変形 AR マーカの検出性能が改善され，正確なバウンディングボックスの取得が可能となったためである．

表 1 各手法での検出性能
Table 1 Detection performance of each method.

Object detection model	mAP	IoU
SSD [4]	0.80	0.88
NanoDet-Plus [6]	0.96	0.96

表 2 各手法での位置推定誤差 [mm]
Table 2 Position estimation error [mm] for each method.

Method	x	y	z	Mean
Previous method [3]	3.84	2.40	13.84	6.69
Proposed method	2.51	1.51	10.68	4.90

4.3 変形 AR マーカの姿勢推定精度

姿勢の推定精度を平均絶対誤差により従来法と提案手法で比較する．提案手法においては，MLP への入力にバウンディングボックスの中心座標の有無，2 モデルの交互最適化の有無について有効性を検証する．なお，2 モデルの交互最適化をしない場合には，AVAE の最適化終了後に MLP を最適化する．評価には，シミュレータ上で撮影した画像から NanoDet-Plus により検出した 11,000 枚の変形 AR マーカ画像を使用する．

表 3 に Roll，Pitch，Yaw の姿勢推定誤差の結果を示す．提案手法は従来法の Roll の精度を保ちながら，Pitch と Yaw の精度を大きく改善できていることが分かる．提案手法の平均誤差は 1.97 [deg] であり，従来法の平均誤差である 5.28 [deg] と比べて 62.7%減少した．また，従来法と提案手法は，Roll に対して Pitch と Yaw の姿勢推定精度が低い．Roll は平面上の回転として表現されるため，見えの変化が大きく推定がたやすい．一方で，Pitch と Yaw は視点の奥行方向に回転し，見えの変化が小さく，マーカの位置によって見え方が大きく変化するため推定が難しい．

提案手法は，MLP にバウンディングボックスの中心座標を入力しない場合と比べ，平均誤差が 53.9%減少した．これは，位置の違いにより変形 AR マーカの見えの変化の影響が大きく，バウンディングボックスの中心座標を MLP に入力することで見えの変化を考慮した姿勢推定が可能となったためだと考えられる．また，提案手法は，個別にモデルを学習した場合と比べ，平均誤差が 48.6%減少した．モデルを交互に最適化する提案手法では，AVAE で変形を除去しながら姿勢推定に適した潜在変数を獲得するため姿勢推定精度が向上したと考えられる．また，**表 4** に提案手法における変形の標準偏差ごとの姿勢推定誤差を示す．変形が大きくなるにつれて，姿勢推定誤差も増加していることが確認できる．これは，変形が大きくなるほどマーカのパターンや輪郭といった特徴がとらえにくくなるためである．**図 8** に提案手法で学習した AVAE により復元した画

表 3 各手法での姿勢推定誤差 [deg]．Pos は MLP にバウンディングボックスの中心座標を入力することを表す．Indev は AVAE と MLP をそれぞれ個別に最適化することを表し，Alt は 2 つのモデルを交互に最適化することを表す
Table 3 Pose estimation error [deg] for each method. “Pos.” indicates that the center coordinates of the bounding box are input to the MLP. “Indiv.” indicates that AVAE and MLP are optimized individually, while “Alt.” indicates that the two models are optimized alternately.

Method	Pos	Indiv	Alt	Roll	Pitch	Yaw	Mean
Previous method [3]	—	—	—	0.69	7.84	7.32	5.28
Proposed method			✓	1.02	5.88	5.92	4.27
Proposed method	✓	✓		1.72	5.01	4.78	3.83
Proposed method	✓		✓	0.83	2.61	2.48	1.97

表 4 各変形の標準偏差での姿勢推定誤差 [deg]
Table 4 Pose estimation error [deg] for each standard deviation of deformation.

Axis	0.0	0.2	0.4	0.6	0.8	1.0	1.2	1.4	1.6	1.8	2.0
Roll	0.76	0.74	0.74	0.76	0.74	0.73	0.79	0.80	0.90	1.21	1.04
Pitch	2.32	2.29	2.18	2.28	2.38	2.41	2.66	2.64	2.93	3.14	3.43
Yaw	2.10	2.04	2.18	2.21	2.19	2.33	2.51	2.64	2.81	2.97	3.25
Mean	1.73	1.69	1.70	1.75	1.77	1.82	1.99	2.03	2.21	2.44	2.58

表 5 各姿勢範囲での検出性能と位置推定誤差 [mm]，姿勢推定誤差 [deg]
Table 5 Detection performance, position estimation error [mm], and pose estimation error [deg] for each pose range.

Method	Pose range	mAP	IoU	x	y	z	Roll	Pitch	Yaw
Proposed method	$[-13\,\mathrm{deg}, 13\,\mathrm{deg}]$	0.96	0.96	2.51	1.51	10.68	0.83	2.61	2.48
Proposed method	$[-30\,\mathrm{deg}, 30\,\mathrm{deg}]$	0.89	0.94	4.42	2.65	19.88	1.12	2.69	3.09

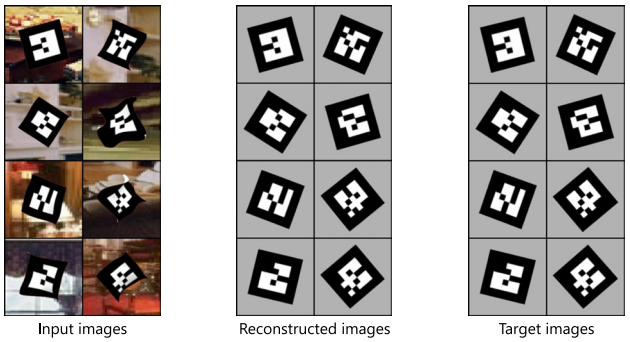

図 8 AVAE による復元画像の例
Fig. 8 Examples of images reconstructed by AVAE.

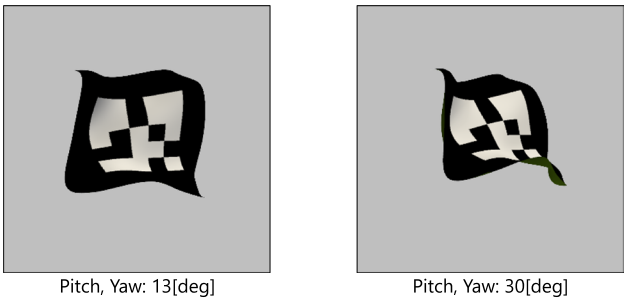

図 9 Pitch と Yaw が 13 [deg] と 30 [deg] での変形 AR マーカ
Fig. 9 Deformed AR marker at Pitch and Yaw angles of 13 [deg] and 30 [deg].

像の例を示す．復元画像はターゲット画像と同様の見た目をしており，入力画像から背景と変形が除去されていることが分かる．

4.4 姿勢範囲の拡大

従来法ではデータベースとの照合の処理が必要な都合上，姿勢範囲が拡大すると比例してデータベースが増大し，照合に要する計算時間とメモリ使用量が膨大となる．そのため，姿勢範囲が大きい場合，現実的な処理時間を考慮すると評価が不可能となる．この制約により，Pitch と Yaw の姿勢範囲を $[-13\,\mathrm{deg}, 13\,\mathrm{deg}]$ の狭い範囲に限定していた．本研究ではデータベースとの照合が必要なく，現実的に処理可能な計算時間であるため，姿勢範囲を $[-30\,\mathrm{deg}, 30\,\mathrm{deg}]$ に拡大して追加実験を行う．**図 9** に変形の影響を最大化した環境における Pitch と Yaw を 13 [deg] および 30 [deg] に設定した変形 AR マーカを示す．ここで変形の影響を最大化した環境とは，変形の標準偏差が 2.0 かつ，AR マーカの位置がカメラの中央から最も離れた状態を指す．13 [deg] より 30 [deg] は変形による歪みや隠れの影響が大きくなることが確認できる．

表 5 に，姿勢範囲別での検出性能と位置，姿勢推定誤差の結果を示す．検出性能では，mAP，IoU ともに若干低下した．これは，姿勢範囲が拡大したことで，マーカの変形やそれにともなう隠れが大きくなり，マーカの特徴をとらえにくくなったからだと考えられる．また，バウンディングボックスの正確性に依存する位置推定精度も低下している．姿勢推定精度においても若干低下しているものの，ほぼ同等であるといえる．$[-30\,\mathrm{deg}, 30\,\mathrm{deg}]$ の範囲では，$[-13\,\mathrm{deg}, 13\,\mathrm{deg}]$ と比較して誤差は増加するものの，大きな破綻をきたすことなく検出，位置，姿勢推定が可能であることを確認した．

ARToolKit [10] のような既存手法では $[-60\,\mathrm{deg}, 60\,\mathrm{deg}]$ で高精度な検出，位置，姿勢推定が可能である．既存手法は，マーカが平面であることを前提にパターンや輪郭，頂点などから検出，位置，姿勢を推定する．そのため，変形やそれにともなう隠れにより検出性能や位置，姿勢推定精度が低下もしくは検出自体が不可能となる．提案手法では，深層学習ベースの検出器の導入やエンコーダによる変形除去によって，変形があっても高精度な検出，位置，姿勢推定が可能である．提案手法は，高度なモーショントラッキングや非剛体物体のピッキングなどに独自の価値を発揮できると考える．高度なモーショントラッキングは，被験者が身につけたマーカのしわやねじれによる変形に対応しながら，位置，姿勢推定によるトラッキングが可能となる．ま

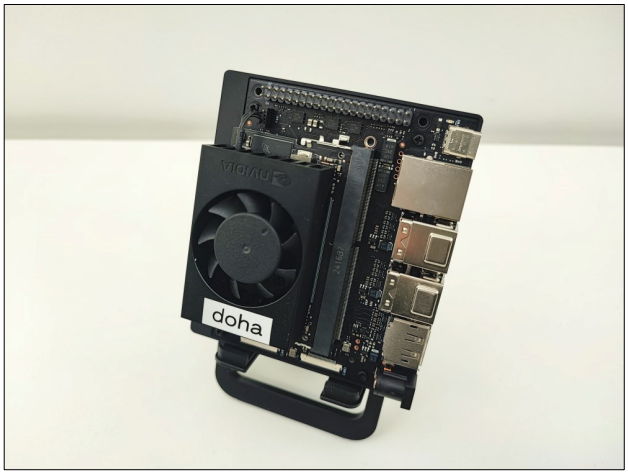

図 10 Jetson Orin Nano
Fig. 10 Jetson Orin Nano.

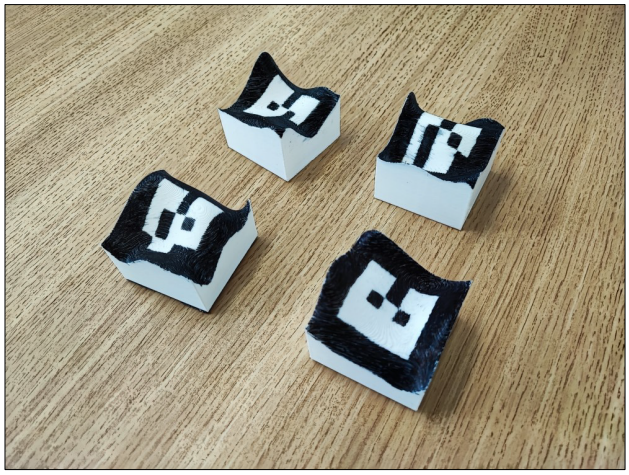

図 11 作成した変形 AR マーカ
Fig. 11 Deformed AR markers created in this study.

表 6 Jetson Orin Nano のスペック
Table 6 Specifications of Jetson Orin Nano.

CPU	6-core Arm Cortex-A78AE v8.2 64-bit CPU 1.5 MB L2 + 4 MB L3
GPU	1024-core NVIDIA Ampere architecture GPU 32 Tensor Cores
Memory	8 GB
Power consumption	7 W–15 W
Size	100 mm × 79 mm × 21 mm

た，非剛体物体のロボットによるピッキングでは物の入った袋のような非剛体の表面にマーカを貼付した場合でも検出，位置，姿勢推定を可能とし対象物体の形状にとらわれないロボットによるピッキングを実現できる．

5. 組み込みボードでの検証

組み込みボードを用いて，提案手法の処理速度とリソース消費量を検証し，実用性を評価する．本研究では，組み込みボードとして**図 10** に示す Jetson Orin Nano を採用する．Jetson Orin Nano は省電力で動作する小型の AI モジュールである．主なスペックを**表 6** に示す．カメラには Logicool C920n HD Pro ウェブカメラ（解像度は 1,920 × 1,080 [pixel]）を使用し，実環境で検証を行う．

検証の結果，提案手法はメモリ使用量が 2.0 [GB] と少量ながら 10.09 [fps] で動作することを確認した．提案手法と同様に変形マーカの検出や位置推定を行う手法である DeepFormableTag [14] では，高精度な変形マーカの検出，位置推定を実現しているが，Jetson Orin Nano において，メモリ使用量が 1.9 [GB]，3.10 [fps] での動作である．DeepFormableTag は，姿勢推定を行わず，独自のマーカを生成し，マーカから情報を復号する処理を含むなど，提案手法との単純な比較はできない．しかし，提案手法は既存の AR マーカを利用し，変形に頑健な検出，位置，姿勢推定を可能としながら，組み込みボードでも実用的な動作を実現した．

実環境で使用する変形 AR マーカを**図 11** に示す．変形 AR マーカの形状は 3D プリンターで作成し，AR マーカのパターンはマーカーペンで着色した．実環境の変形 AR マーカに対する位置・姿勢推定の例を**図 12** に示す．変形 AR マーカの ID をワイヤーキューブの色で表し，変形 AR マーカの位置と姿勢をワイヤーキューブの位置と姿勢で表現している．デモンストレーションの例から，変形 AR マーカを高精度に位置・姿勢推定できていることが視覚的に確認できる．

6. おわりに

本稿では，変形 AR マーカの高速，高精度な 3 次元位置・姿勢推定手法の提案と組み込みボードへの実装について述べた．

従来法 [3] の課題であった精度と処理時間を解決するために，下記の 4 つを提案した．

- 物体検出モデルの変更
- AAE を拡張した AVAE の提案
- 回帰による 3 次元姿勢推定
- 2 つのモデルの交互最適化

今後は，角や円柱，球体のような多種多様な変形での実験をするとともに，実環境の変形 AR マーカに対し，高速かつ高精度な位置・姿勢推定を行う予定である．

図 12 実環境におけるデモンストレーションの例．変形 AR マーカの ID をワイヤーキューブの色で表し，変形 AR マーカの位置と姿勢をワイヤーキューブの位置と姿勢で表現している

Fig. 12 Example of demonstration in a real environment. The ID of the deformed AR marker is represented by the color of the wire cube, and the position and pose of the deformed AR marker are represented by the position and pose of the wire cube.

参考文献

[1] 小野智司，川上雄大，伊藤拓也ほか：ゴミ袋に貼付された歪んだ 2 次元コードの復号，人工知能学会全国大会論文集，Vol.JSAI2012, 3F1OS196（オンライン），10.11517/pjsai.JSAI2012.0_3F1OS196 (2012).

[2] 小野智司，川上雄大，伊藤拓也ほか：ゆがんだ二次元コードの復号による廃棄物認識，人工知能学会誌，Vol.28, No.4, pp.575–582 (2013).

[3] 榎元洋平，山内悠嗣：機械学習による変形 AR マーカの 3 次元位置・姿勢推定，動的画像処理実利用化ワークショップ (2022).

[4] Liu, W., Anguelov, D., Erhan, D., et al.: SSD: Single Shot MultiBox Detector, *Proc. ECCV 2016*, Leibe, B., Matas, J., Sebe, N. and Welling, M. (Eds.), pp.21–37, Springer International Publishing (2016).

[5] Sundermeyer, M., Marton, Z., Durner, M., et al.: Implicit 3D Orientation Learning for 6D Object Detection from RGB Images, *Proc. ECCV 2018*, pp.699–715 (2018).

[6] RangiLyu: NanoDet-Plus: Super fast and high accuracy lightweight anchor-free object detection model, GitHub (online), available from ⟨https://github.com/RangiLyu/nanodet⟩(accessed 2025-4-18).

[7] Kingma, P.D. and Welling, M.: Auto-Encoding Variational Bayes, *Proc. ICLR 2014* (2014).

[8] NVIDIA: Jetson AGX Orin for Next-Gen Robotics, NVIDIA (online), available from ⟨https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin⟩ (accessed 2025-04-18).

[9] 長屋隆之，原　昌宏：高速読取り対応 2 次元コード［QR コード］の開発，全国大会講演論文集，Vol.52, pp.253–254（オンライン），入手先 ⟨https://ipsj.ixsq.nii.ac.jp/records/129358⟩ (1996).

[10] Kato, H. and Billinghurst, M.: Marker tracking and HMD calibration for a video-based augmented reality conferencing system, *Proc. IWAR 1999*, pp.85–94, IEEE Computer Society (1999).

[11] Fiala, M.: ARTag, a fiducial marker system using digital techniques, *Proc. CVPR 2005*, pp.590–596, IEEE Computer Society (2005).

[12] Olson, E.: AprilTag: A robust and flexible visual fiducial system, *Proc. ICRA 2011*, pp.3400–3407 (2011).

[13] Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F.J., et al.: Automatic generation and detection of highly reliable fiducial markers under occlusion, *Pattern Recognit.*, Vol.47, No.6, pp.2280–2292 (2014).

[14] Yaldiz, B.M., Meuleman, A., Jang, H., et al.: DeepFormableTag: End-to-end generation and recognition of deformable fiducial markers, *ACM Trans. Graph.*, Vol.40, No.4, pp.1–14 (2021).

[15] Girshick, R., Donahue, J., Darrell, T., et al.: Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation, *Proc. CVPR 2014*, pp.580–587, IEEE Computer Society (2014).

[16] Girshick, R.: Fast R-CNN, *Proc. ICCV 2015*, pp.1440–1448, IEEE Computer Society (2015).

[17] Ren, S., He, K., Girshick, R., et al.: Faster R-CNN:

Towards Real-Time Object Detection with Region Proposal Networks, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol.39, No.6, pp.1137–1149 (2017).
[18] Redmon, J., Divvala, S., Girshick, R., et al.: You Only Look Once: Unified, Real-Time Object Detection, Proc. *CVPR 2016*, pp.779–788, IEEE Computer Society (2016).
[19] Law, H. and Deng, J.: CornerNet: Detecting Objects as Paired Keypoints, *Int. J. Comput. Vision*, Vol.128, No.3, pp.642–656 (2020).
[20] Duan, K., Bai, S., Xie, L., et al.: CenterNet: Keypoint Triplets for Object Detection, *Proc. ICCV 2019*, pp.6568–6577, IEEE Computer Society (2019).
[21] Vaswani, A., Shazeer, N., Parmar, N., et al.: Attention is All you Need, *Proc. NeurIPS 2017*, Guyon, I., Von Luxburg, U., Bengio, S., et al. (Eds.), pp.6000–6010, Curran Associates Inc. (2017).
[22] Carion, N., Massa, F., Synnaeve, G., et al.: End-to-End Object Detection with Transformers, *Proc. ECCV 2020*, Vedaldi, A., Bischof, H., Brox, T., et al. (Eds.), pp.213–229, Springer International Publishing (2020).
[23] Zhu, X., Su, W., Lu, L., et al.: Deformable DETR: Deformable Transformers for End-to-End Object Detection, *Proc. ICLR 2021*, OpenReview.net (2021).
[24] Peng, S., Zhou, X., Liu, Y., et al.: PVNet: Pixel-Wise Voting Network for 6DoF Pose Estimation, *Proc. CVPR 2019*, pp.4556–4565 (2019).
[25] Kehl, W., Manhardt, F., Tombari, F., et al.: SSD-6D: Making RGB-Based 3D Detection and 6D Pose Estimation Great Again, *Proc. ICCV 2017*, pp.1530–1538, IEEE Computer Society (2017).
[26] Xiang, Y., Schmidt, T., Narayanan, V., et al.: PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes, *Proc. RSS 2018* (2018).
[27] Li, Y., Wang, G., Ji, X., et al.: DeepIM: Deep Iterative Matching for 6D Pose Estimation, *Proc. ECCV 2018*, Ferrari, V., Hebert, M., Sminchisescu, C., et al. (Eds.), pp.695–711, Springer International Publishing (2018).
[28] Lin, T., Dollar, P., Girshick, R., et al.: Feature Pyramid Networks for Object Detection, *Proc. CVPR 2017*, pp.936–944, IEEE Computer Society (2017).
[29] Hinton, E.G. and Salakhutdinov, R.R.: Reducing the Dimensionality of Data with Neural Networks, *Science*, Vol.313, No.5786, pp.504–507 (2006).
[30] He, K., Zhang, X., Ren, S., et al.: Deep Residual Learning for Image Recognition, *Proc. CVPR 2016*, pp.770–778 (2016).
[31] Niekum, S.: ar_track_alvar, ROS Wiki (online), available from ⟨https://wiki.ros.org/ar_track_alvar⟩ (accessed 2025-04-21).

浅野 右京 （学生会員）

2002 年生．2024 年中部大学工学部ロボット理工学科卒業．2025 年同大学大学院修士課程在学中．

山内 悠嗣 （正会員）

2012 年中部大学大学院博士後期課程修了．博士（工学）．同大学助手を経て，2018 年より同大学講師，2023 年同大学准教授となり，現在に至る．2010 年独立行政法人日本学術振興会特別研究員．画像認識，機械学習，知能ロボティクスの研究に従事．2013 年電子情報通信学会 ISS 論文賞，2014 年情報処理学会山下記念研究賞受賞．電子情報通信学会，日本ロボット学会，IEEE 各会員．