現場で使える低価格なAI実装モデルの構築

谷山清吾*，上薗　剛**

Development of a Low-Priced AI Implementation Model Used in the Field

Seigo TANIYAMA and Tsuyoshi UEZONO

AI（Artificial Intelligence：人工知能）技術の活用において，エッジのデバイス（一般的に，端末と端末側で収集したデータを回線に送り出すポイントのことで,本稿では,現場の製造装置等に設置されているコンピュータのこと）で処理を行う手法がある。インターネット環境が不要のため，通信速度による影響を受けないことや，セキュリティのリスクが少ないという利点があるが，処理能力の低いデバイスでAIの処理を行うことが問題になる。そこで，本研究ではAIモデルの圧縮による処理軽量化や使用デバイスの検証を行い，安価なデバイスを用いて，作業者による目視検査と同程度の時間で処理できることを確認した。

Keyword：エッジAI，AI実装，AI軽量化，RaspberryPi，画像分類

１．　緒　　言

AI技術の発展に伴って社会のAI活用も広まりつつある。本県においてもAI，IoT（Internet of Things：モノのインターネット）技術に関心の高い企業は多く，これらの技術の活用推進が急務である。しかし，AI技術の専門性や導入コストの高さもあり，県内中小企業におけるAI技術の活用例は非常に少ない。

その中で,近年注目されているエッジAIという分野では，エッジのデバイスでAI処理を行うため，インターネット環境が不要，セキュリティ面でのリスクが少ないという点から工場などの現場でのAI活用が容易になる。また，安価なデバイスを用いることでより低コストでのAI活用が期待できる。エッジ（edge）とは，一般には「ふち」や「端」,「刃物の刃」などの意味であり，IoTの分野では，端末と端末側で収集したデータを回線に送り出すポイントを「エッジ」という。例えば，エッジコンピューティングでは，センサや測定器が採取したデータをエッジ（データを送り出すポイント）に配置したコンピュータで解析，遠隔地には必要なデータだけを送信することで，ネットワークの負担を軽減することができる[1)]。

本研究では，県内企業のAI活用の促進を目的に，低価格なエッジAIデバイスの検討及び処理が軽量，かつ現場で扱える高精度なAIモデルの構築に取り組んだ。

２．　研究概要

２．１　AI圧縮手法の検討

処理能力の低いデバイスでAI処理を行うために，AIの処理を軽量化する圧縮手法の検討を行った。圧縮を行うことでAI処理の計算量削減や計算効率の向上によって処理時のデバイスへの負荷を小さくすることができる。ただし，圧縮をより強くかけるとデバイスへの負荷は小さくなるが，AIの推論精度が圧縮前に比べて落ちる可能性がある。本研究では，図１と図２に示す量子化と枝刈りの２つの圧縮を検討し，圧縮対象となるAIの精度が落ちないことを最優先として，推論精度を維持した範囲での圧縮パラメータの検証を行った。

・量子化：ニューラルネットワークの演算やニューロンの重みなどAIのパラメータを通常の32ビット浮動小数点から桁の小さな固定小数点に置き換えることでメモリの使用量削減や計算効率の向上を図る（図１）。

・枝刈り：ニューラルネットワーク内で重みの小さな接続や，影響の少ないノードを削除することで，計算量を削減しメモリの使用量を抑える（図２）。

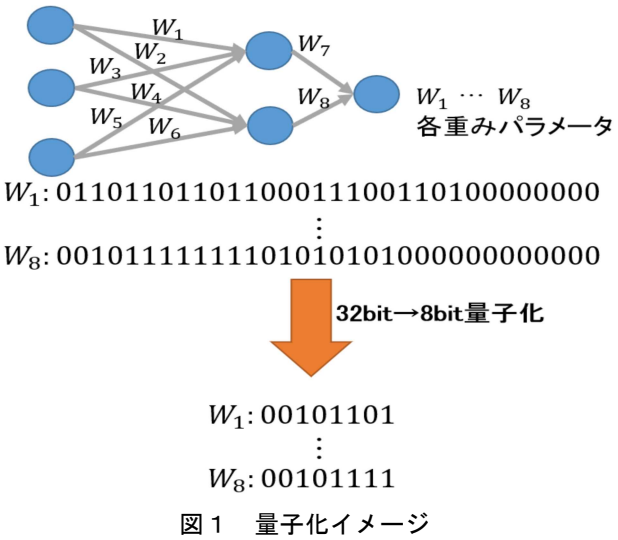

図１　量子化イメージ

*生産技術部
**地域資源部

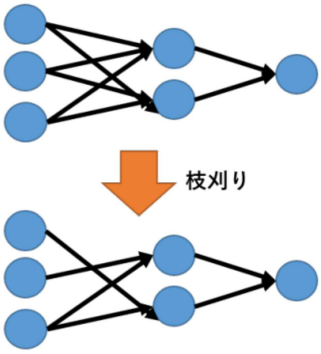

図２　枝刈りイメージ

２．２　AI実装デバイスの検討

AI実装を行うエッジデバイスの検討を行った。本研究では，低価格なAI実装モデルを構築するため，低価格で入手可能かつ一定の処理能力を有するエッジデバイスとしてRaspberry Pi3B+(以下，3B+)，Raspberry Pi4B(以下，4B)の２機種を検討した。

3B+，4Bは，IoTの導入でも広く使用されているデバイスであり，安価である。県内企業でIoT活用に使用されている例もあり，比較的馴染みのあるデバイスであるため，これらのデバイスを選定した。

各デバイスの性能などの概要を図３及び図４に示す。各デバイスを比較すると，CPU性能，メモリ容量どちらにおいても4Bの方が高性能のデバイスである。

CPU	ARM Cortex-A53 (ARM v7)64-bit 1.4GHz
メモリ	1GB LPDDR2 SDRAM

図３　Raspberry Pi3B+概要

CPU	Quad core Cortex-A72 (ARM v8)64-bit 1.5GHz
メモリ	4GB LPDDR4 SDRAM

図４　Raspberry Pi4B概要

２．３　AI実装モデルの性能評価

２．３．１　実装する対象AI

AI実装モデルを構築する上で実装を行う対象AIを２つ用意した。各AIの概要を表１に示す。また，今回のAI実装においては製品の不良検出などで県内製造業から需要の高い画像分類を対象とした。

表１　実装AI概要

	実装AI①	実装AI②
概要	CIFAR-10を用いた 10種類の画像分類	ねじ頭部の傷検出
入力画像	32×32（カラー）	140×140（カラー）
学習画像数	5000[枚]	756[枚]

実装AI①は広く，画像分類の機械学習チュートリアル等で使用されるCIFAR-10の公開されたデータセット[2)]を用いた。CIFAR-10データセットの10種類の画像例を図５に示す。

図５　CIFAR-10データセット画像例

実装AI②は，県内企業で製造しているねじ頭部に発生する傷の検出を目的とした。今回のサンプルの場合，専用の機械による画像判別または作業者による目視での検査を行っており，検査時間は，機械の場合１分で200個（１個あたり0.3秒），作業者の場合は１個あたりおよそ１～２秒かかっているとのことだった。また，データセットは，同様のねじに模擬的に傷をつけ画像を撮影することで作成した。作成したねじ画像について，不良画像（傷アリ）の例を図６に，良品画像（傷ナシ）を図７に示す。

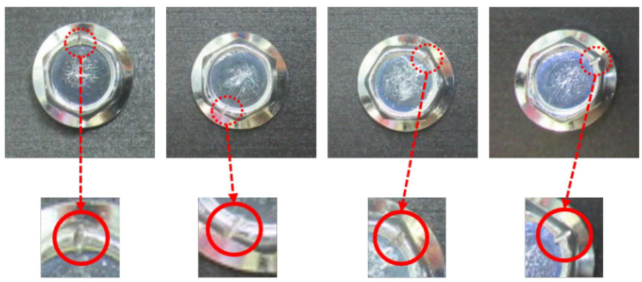

図６　ねじ画像（傷アリ）

図７　ねじ画像（傷ナシ）

２．３．２　評価方法

実装AI①と実装AI②において，[推論用のAI読み込み]，[テスト用画像１枚の入力]，[結果の出力]の一連の流れを完了するまでの処理時間及び推論の精度による評価を行った。ここでの精度の値は,実装AI①では,テスト用画像1,000枚における精度，実装AI②では，テスト用画像108枚における精度である（テスト用画像はAIの学習に用いていない未学習の画像)。処理時間は実装AI①では，テスト用画像1,000枚を処理した際の１枚処理にかかる平均時間,実装AI②では，テスト用画像108枚を処理した際の１枚処理にかかる平均時間としている。

また，実装AI①，②について，それぞれ圧縮によるAI軽量化の有無と，実装デバイス（3B+，4B）の違いによる評価結果の比較を行った。

２．３．３　評価結果

実装AI①の評価結果を図８に，実装AI②の評価結果を図９に示す。

実装AI①において，軽量化による処理時間の短縮は3B+の場合で約4.8秒，4Bの場合で約3.2秒であった。

また，3B+と4Bのデバイスの違いによる処理時間の差については,性能の高い4Bの方が処理時間が約4.3秒短くなった。

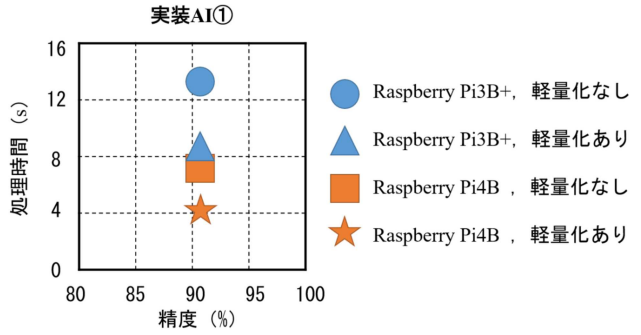

図８　実装AI①評価結果

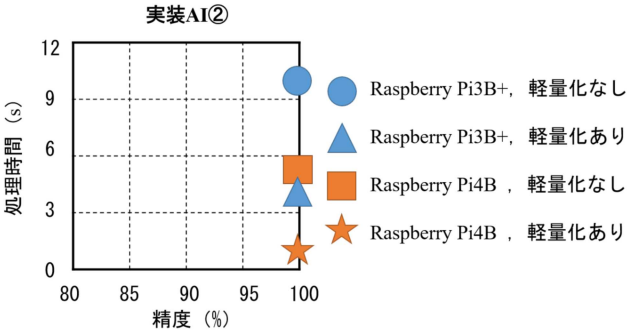

図９　実装AI②評価結果

実装AI②において，軽量化による処理時間の短縮は3B+の場合で約7.2秒，4Bの場合で約3.2秒であった。

また，3B+と4Bのデバイスの性能による処理時間の差については,性能の高い4Bの方が処理時間が約6.0秒短くなった。

以上の結果から，今回実施した評価実験の全体的な傾向として，圧縮を用いたAI軽量化により精度を維持しながら処理時間を短縮できていることがわかった。また，今回用いた２種類のデバイスにおいて，推論の精度については，両デバイスとも同様の圧縮を行ったため，デバイスによる精度の違いは発生していなかった。また，処理時間については，処理能力の高い4Bの方が全ての評価結果において，より高速に処理できていることがわかった。

なお，県内企業で製造しているねじ頭部に発生する傷の検出を目的として行った実装AI②の評価実験において，性能の高い4Bを用いて,圧縮を用いたAI軽量化を行った場合，処理時間が約1.4秒となり，作業者が目視で検査を行う場合の処理時間（１～２秒）とほぼ同程度の時間で処理できることを確認した。

２．４　AI実装モデルのシステム化

実装AI①の評価を行った際は，[推論用のAI読み込み]，[テスト用画像１枚の入力]，[結果の出力]の一連の流れをCUI（コマンド）で行っていたが，一般の作業者等が使用する際の使いやすさを考慮し，実装AI②の評価を行う際に，画像の撮影から推論の実行，結果の出力までをGUI（グラフィック）で行う簡易システムを構築した。構築したシステムの構成を図10に，システムの使用イメージを図11に示す。

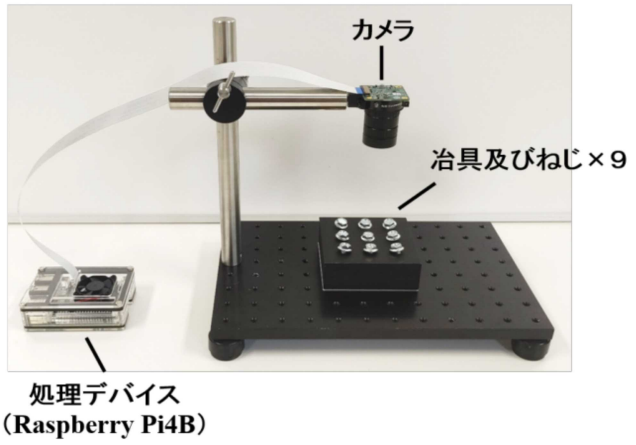

図10　簡易システムの構成

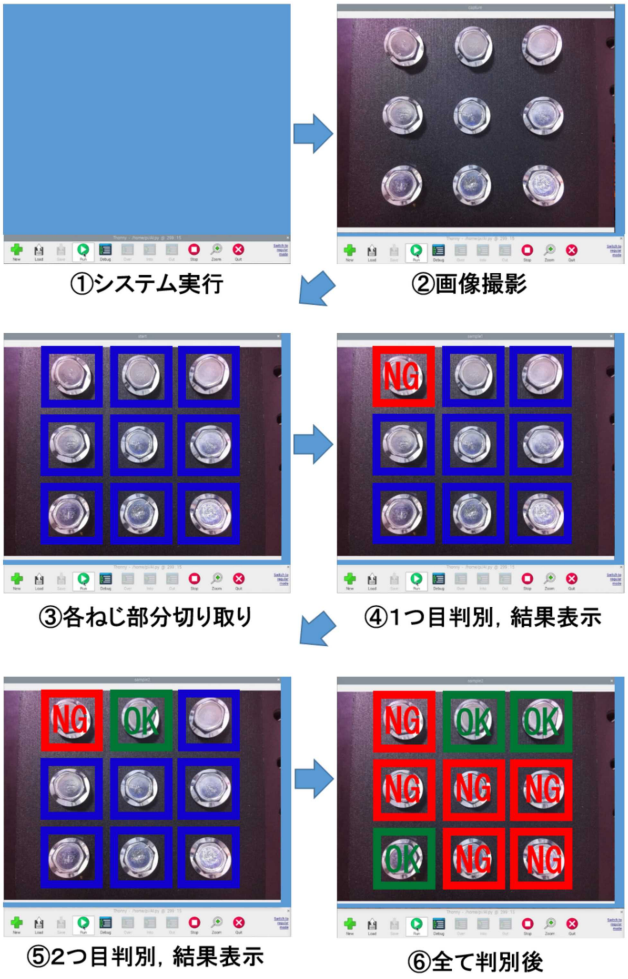

図11　簡易システム使用イメージ

今回構築したシステムでは，対象となるねじを撮影用冶具に設置することで１度に９つ撮影し，撮影した画像の中から１つずつねじ頭部を切り取り，それぞれのねじについて，傷の有無を判別して結果を画面上に示している。

３．　結　　言

AI圧縮による軽量化で，処理能力の低いエッジデバイスでも元となるAIの精度を維持しつつ，より短時間での処理が可能となった。これにより，数千円程度の低価格なデバイスでも，量子化や枝切りなどの圧縮手法を用いて処理時間を大幅に短縮することで，作業者による目視検査と同程度の時間で処理できることを確認した。

今後は，これらの研究成果の県内企業への技術移転を進めるとともに，今回用いた圧縮手法以外の圧縮手法を使用しての軽量化効果の検討やより多くのデバイスでの実装性能の比較を行うことでさらに短時間で処理可能となる実装モデルも検討したい。また，今回は画像処理のAIを対象として実装モデルを構築したが画像処理以外の数値解析などのAIに対する効果も検証することでより広い分野でのAI活用にも取り組みたい。

参　考　文　献

1）株式会社キーエンス「製造現場で役立つIoT用語辞典」.
https://www.keyence.co.jp/ss/general/iot-glossary/
（閲覧日 2022-7-19）

2）The CIFAR-10 dataset
https://www.cs.toronto.edu/~kriz/cifar.html
（閲覧日 2022-7-19）