
Development Stories

開発事例集 ～ 研究開発・Poc編 ～

当社R&D・顧客からのPoCなど様々な形態で日々研究開発を行っております。
主に、AI、機械学習、GPGPU、点群データ処理など、日進月歩の最新技術を使用し開発を行っております。

目次

	製造	建築	医療	農水産	インフラ	金融	その他	ページ
共同研究》外観検査における異常検知システム	●							p.4
自動車用部品の異常検知評価	●	●			●			p.5
半教師あり学習モデル	●	●			●			p.6
少ラベルデータでの教師あり学習の精度検証	●	●			●			p.7
LLMの歴史、学習方法、マルチモーダルLLMの調査							●	p.8
LLMにおける処理高速化、分散学習技術の調査							●	p.9
LLMの高速処理技術に関する調査							●	p.10
LLM向け 学習後量子化の開発・実証							●	p.11
LLMを使用したスポーツ試合映像分析							●	p.12
微小領域のセグメンテーション研究開発	●			●	●			p.13
小物体の検知を目的とする物体検出	●	●	●	●	●		●	p.14
スポーツ中継映像におけるDLホモグラフィー推定							●	p.15
AIを用いた物理方程式解析							●	p.16
深層学習バイナリモデルのエッジデバイス実装評価	●	●	●	●	●	●	●	p.17
充足性判定に向けたニューロンカバレッジの実装評価	●	●	●	●	●	●	●	p.18
説明可能AI	●	●	●	●	●	●	●	p.19
Egocentric Vision 一人称視点映像を用いた作業分析	●	●	●	●	●		●	p.20
一人称視点映像における物体検知技術の調査	●	●	●	●	●		●	p.21
FISH-QUANT CUDA開発	●		●		●			p.23
X線CT画像3D復元高速化	●	●	●		●			p.24

目次

	製造	建築	医療	農水産	インフラ	金融	その他	ページ
<u>点群データ関連最新技術の調査・実装</u>	●	●	●		●		●	p.26
<u>ドローン制御用自己位置推定モジュール</u>		●			●			p.27
<u>AIによるVisual SLAM技術の検証</u>		●			●			p.28
<u>大規模環境の三次元計測に向けたNeRF技術調査</u>		●			●			p.29
<u>電頭3次元再構築</u>		●	●					p.30
<u>宇宙空間における人工建造物の解析技術手法の調査</u>							●	p.31
<u>3D CADモデル生成アルゴリズムの調査</u>	●	●			●		●	p.32
<u>アームロボットを使用した強化学習</u>	●		●					p.34
<u>人工衛星の制御ソフトウェア開発支援</u>							●	p.35
<u>測点の変更を含む経路最適化高度化アルゴリズム実装評価</u>	●	●						p.37
<u>半導体製造装置 データ解析モジュール開発</u>	●							p.38
<u>半導体製造装置 計測箇所最適化</u>	●			●				p.39
<u>時系列データにおける外乱の検知手法の検討</u>	●		●					p.40
<u>マテリアルズインフォマティクス</u>	●		●				●	p.41
<u>組成提案スクリプトのモジュール化および新機能開発支援</u>	●		●				●	p.42

AI

～ artificial intelligence ～

2012年のILSVRCをきっかけに、第3次AIブームが始まっています。

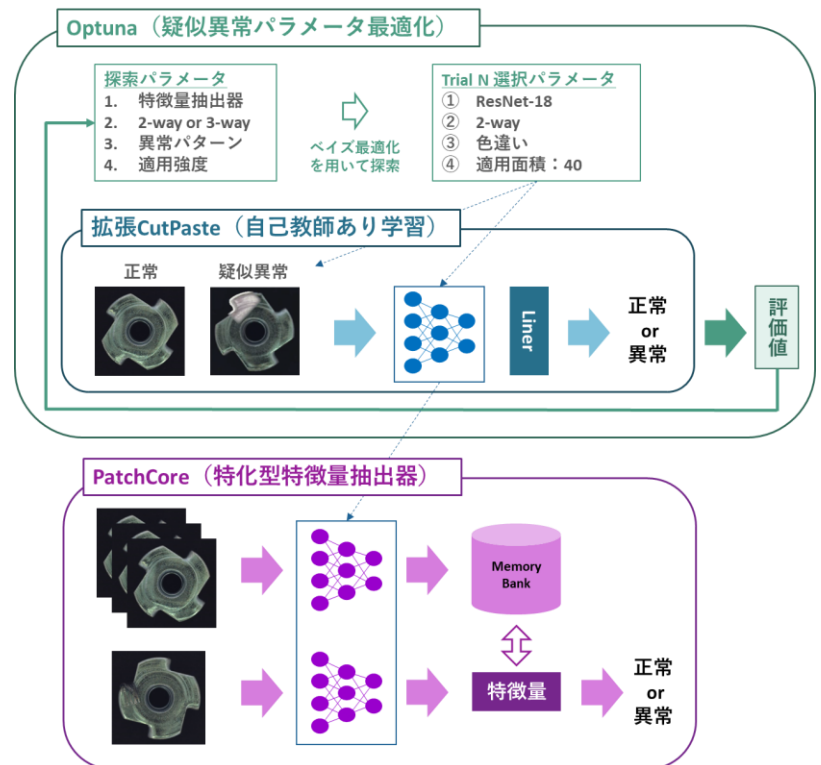
当社でもいち早くDeep Learningに着目し、これまでに様々な製品、システムを開発してきました。

外観検査における異常検知システムの研究開発 山梨大学との共同研究

深層学習を用いた外観検査の異常検知モデルを現場に落とし込めるようにする為に、検査対象に特化したモデルを自動で作成するシステムの研究開発を行っております。

近年では、ImageNetの学習済みモデルを使った特徴量抽出型の異常検知モデルなどが注目されておりますが、手軽で且つ汎用的に使える一方で現場に最適化された特徴量抽出器にはなっておりません。

そこで、自動で最適な特徴量抽出器を学習するシステムを開発する為に、自己教師あり学習手法の異常検知モデルであるCutPasteに注目しました。正常な画像の一部をカット＆ペーストすることで疑似異常として学習する手法となっておりますが、この疑似異常の付与を拡張し、実際の異常に少しだけ似た低レベルな異常を画像処理で付与出来るようにしました。これにより、従来手法と比べてより良い特徴量抽出器を学習することが可能となりました。しかし、様々な疑似異常パターンが付与出来るようになった反面、検査対象に最適な疑似異常の選択が難しくなりました。そこで、パラメータ最適化ツールのOptunaを用いることで適切な疑似異常パターンと適用強度を自動で探索可能にしました。



最適な特徴量抽出器の学習システムと、それにより生成した学習済みモデルを用いたPatchCoreによる異常検知

共同研究への取り組み

弊社が独自に新たな技術を獲得していくために、2021年より山梨大学と共同研究を行っております。

その中で、世の中での異常検知へのニーズの高まりから、上記のテーマを掲げてシステムの研究開発に取り組んできました。これらの成果は、プロダクト化して現場へ導入出来るようにしていきます。

今後も、共同研究を通して新しい技術のキャッチアップをしていきます。

開発例

項目	詳細情報
OS	Ubuntu 18.04 LTS
開発期間	42ヶ月 (2021年10月開始)
開発規模	9人 (山梨大学：3名、CM：6名)
開発言語	Python
使用モデル	CutPaste、PatchCore
GPU	Tesla V100
学習データ数	青果 11436枚、病理 3622枚
関連技術	・ 深層学習 ・ 異常検知 ・ 画像処理 ・ Optuna

自動車用部品の異常検知評価

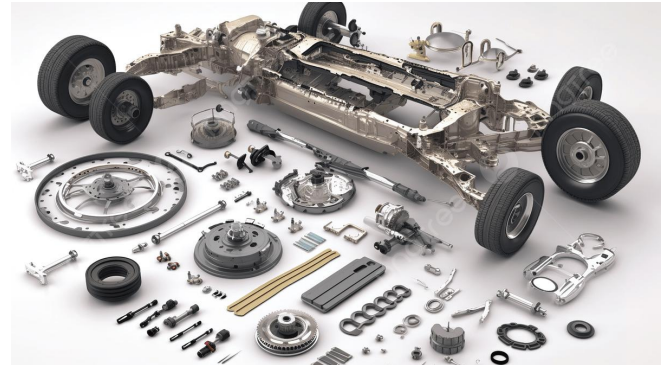
自動車に使用される部品の検査は、多くの場合は人間が目視で行っており、AIを活用した自動化によるコスト削減が期待されています。

AIを活用する場合は学習データが必要となりますが、一般的に工業製品などの多くは異常品のデータが少なく、所持している多くのデータが正常品である場合が多いです。

特に自動車業界など、製造する製品に対して高い信頼性が求められる業界については、日常的にも異常品が発生しにくい環境であり、異常品のデータを集めることが困難と想定されます。

そこで、正常品のみを学習に使用して正常品と異常品の判定を行う

「異常検知: Anomaly Detection」と呼ばれるタスクの中でも一般的な「PatchCore」という手法を利用し、自動車用部品の異常検知評価を行っております。



PatchCoreを用いた異常検知

PatchCore を用いた学習と評価は、以下の流れで実施します。

- ① 学習: 大量にある正常データから「正常」とであるという特徴をAIモデルが学習します。
- ② 評価: ①で学習したAIモデルを用いて評価データを推論します。この時、評価データが正常品である場合、学習済みの特徴と類似している可能性が高いことから正常と判定されます。対して、異常品である場合は、学習済みの特徴とかけ離れている可能性が高いことから異常と判定されます。

正常・異常判定の閾値は手動で調整も可能であるため、正常品の過検知を抑制したり、異常品の見逃しを低減したりと、用途に応じたチューニングが可能です。

弊社では自動車部品に限らず、様々な製品に対する異常検知を行っています。
PatchCoreに限らず、異常検知手法の論文調査から実装、評価の実施や、お客様自身で評価を実施できるよう、学習評価環境の提供なども行っています。

開発例

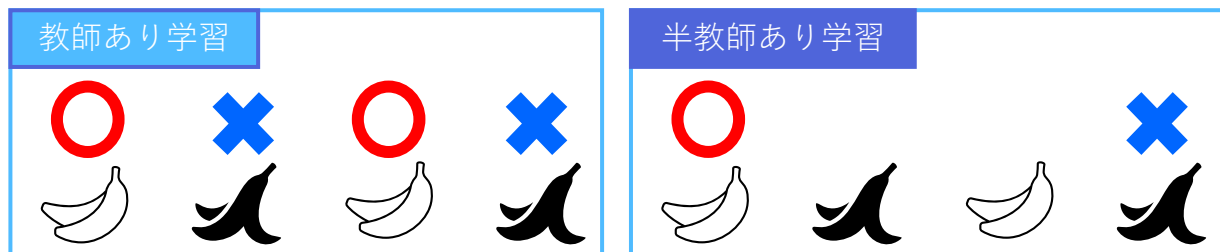
項目	詳細情報
OS	Windows(WSL2)
開発期間	3ヶ月
開発規模	1名
開発言語	Python
GPU	RTX 4080(laptop)
関連技術	Deep Learning 異常検知 PatchCore

半教師あり学習モデル

一般的に、深層学習では精度を高めるために大量のデータを必要とします。学習データを用意する上で、重要かつコストを要するのが正解ラベルを設定するアノテーション作業です。特にセグメンテーションでは、物体ごとに画像の領域を指定する必要があるため、アノテーション作業にかかるコストは画像分類と比較して高くなります。

アノテーション作業にかかるコストを削減する方法として、半教師あり学習が知られています。半教師あり学習は、大量のアノテーションなしの画像と、少量のアノテーション済みの画像を用いて学習する手法です。

産業界では、データは大量に取得できるもののアノテーションコストが高く、学習に利用できないことがあります。アノテーションなしの画像も学習に利用できる半教師あり学習は、非常に実用的な技術です。



論文から学習スクリプトを作成

深層学習は多くの手法が提案され、そのスクリプトもオープンソースソフトウェアとして公開されていますが、全ての手法のスクリプトが公開されているわけではありません。

本案件で使用した半教師あり学習手法は、画像分類タスクとして提案された半教師あり学習手法をセグメンテーションに拡張したものであり、スクリプトは非公開となっていました。そのため、公開されている画像分類用半教師あり学習スクリプトや論文の記述をもとに、半教師ありセグメンテーションスクリプトを作成しました。



論文



プログラム

上記のように弊社では実装の公開されていない深層学習手法であっても、論文をもとにスクリプトを作成し、提供することが可能です。

また、スクリプトの提供だけでなく、迅速かつ安価で検証作業だけを請け負うこともでき、学習においても理論に基づいた精度向上の提案までサポート致します。

開発例

項目	詳細情報
OS	Ubuntu CentOS
開発期間	2ヶ月
開発規模	1人
開発言語	Python、PyTorch
関連技術	・ 深層学習 ・ セグメンテーション ・ 半教師あり学習

少ラベルデータでの教師あり学習の精度検証

画像認識のタスクにおけるアノテーション作業（データに教師ラベルを付ける作業）は非常にコストが掛かる作業です。そこで、少量のラベルデータで学習を行える「教師なし学習、自己教師あり学習、半教師あり学習」などの手法から有効そうなアルゴリズムを調査・ピックアップし、その中から課題に適したアルゴリズムを1つ選択して実際にお客様のデータで精度検証を行ってアノテーションコストを抑えた有効な学習方法を検討しました。

教師なし学習

訓練データにラベルを与えずにモデルを生成する学習手法。

自己教師あり学習

教師なし学習と同じく訓練データにラベルを与えませんが、データから擬似的なラベルを自動で生成し、そのラベルを用いて教師あり学習を行う手法。

半教師あり学習

少量のラベルありデータと大量のラベルなしデータを用いて学習を行う手法。

調査したアルゴリズム一覧

学習手法	アルゴリズム名
教師なし学習	k-means
自己教師あり学習	DeepCluster
	UMDL
	SPICE
	SimCLR
半教師あり学習	FixMatch

半教師あり学習(FixMatch)を用いた精度検証

11クラスの分類問題に対して、1クラス当たり100枚の画像を用意し、その内25枚にラベル付けを行ってtrain/testに振り分けて学習・評価を行いました。その結果、比較的簡単な分類問題であったこともあり、均等にラベル付きデータがある場合ほぼ100%に近い精度が出ていました。

一方で、均等にラベルを付けないでクラスごとにラベル付きデータの枚数を変更して評価も行ったところ、2~5%程度の精度低下が見られました。特に一部のクラスのみ極端にラベル数が少ない不均衡データセットの場合に精度低下が顕著にみられました。この案件では、精度評価以外にも複数のデバイス(Jetson, Raspberry Pi)での速度評価なども実施しておりました。

開発例

項目	詳細情報
OS	Ubuntu
開発期間	6ヶ月
開発規模	2名
開発言語	Python
関連技術	・ Deep Learning ・ 教師なし学習 ・ 自己教師あり学習 ・ 半教師あり学習

LLMの歴史、学習方法、マルチモーダルLLMの調査

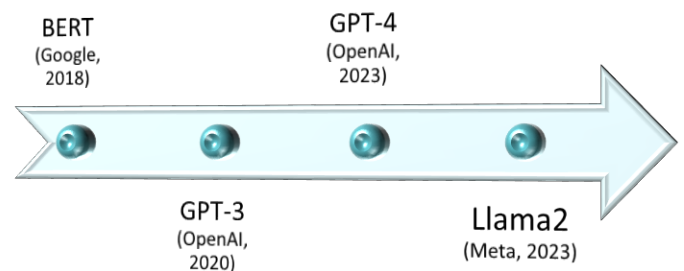
本調査は急速に発展し続ける大規模言語モデル「LLM」の技術について深く掘り下げて調査するというものです。ChatGPTの登場以来、LLMは人工知能分野で非常に高い注目を集めており、絶えず拡大する応用技術が見られます。

本案件では、LLM開発の歴史、学習・推論手法、マルチモーダルLLMについての調査を行いました。

LLM開発の歴史調査

これまでのLLM開発を時系列で調査し、各モデルにおいて工夫された点、モデルの構成要素の変化をまとめました。

また、日本語LLMについても調査を行い国内でのLLM開発にも目を向けております。



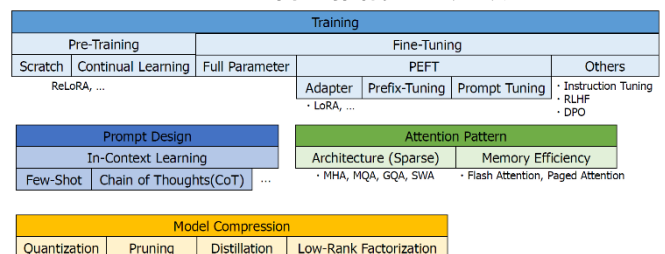
LLMの学習・推論手法調査

LLMの学習・推論手法を体系化した後、

- Parameter Efficient Fine Tuning (PEFT)
- Retrieval Augmented Generation (RAG)
- Mixture of Experts (MoE)

を重点調査対象として概要、最新動向の調査を行いました。

LLMの学習・推論手法分類



※RAG、MoEは上記分類外のモデル拡張技術とも言える

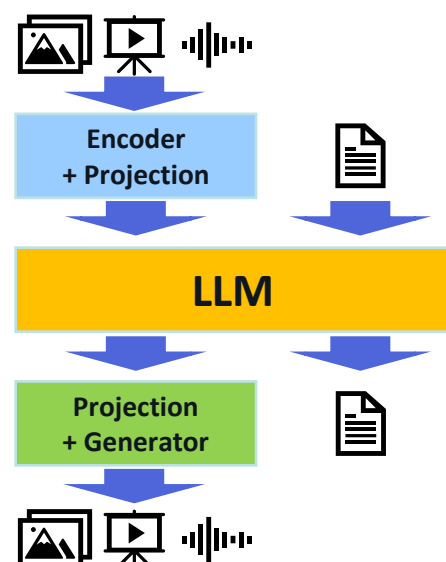
マルチモーダルLLM調査

マルチモーダルLLMのモデル構成、概要、現在公表されているモデルの詳細について調査を行いました。

マルチモーダルLLMはモデルへの入力 / 出力共にテキストだけでなく画像、動画、音声等の様々な形式に対応します。

推論の精度については現状課題もありますが、人工汎用知能 (Artificial General Intelligence、AGI) の実現に強く寄与していく分野と考えられます。

マルチモーダルLLMの構成要素



LLMにおける処理高速化、分散学習技術の調査

ChatGPTと同様の大規模言語モデル「LLM」の台頭は、AIと自然言語処理の領域に革命をもたらしました。人間のように自然な文章を生成するこれらのモデルは、カスタマーサービスや文書作成など、多岐にわたる用途で活躍しています。しかしながら、これらのLLMのトレーニングと維持には膨大な計算リソースが必要とされ、これは大きな課題となります。特に、トレーニングプロセスの高速化と、多くのGPUを駆使した分散システム上での学習実施は、効率性とコスト削減の観点から緊急を要する問題です。

本調査報告では、上記の課題に対処するための最先端技術に関して、数式やコーディングを含む包括的な理論調査を行いました。さらに、各社アクセラレータ、各種フレームワークの機能性、ベンチマークスコア、人気度について比較検討を行い、その成果をもとに実装へと移行しました。

実装とベンチマークの取得

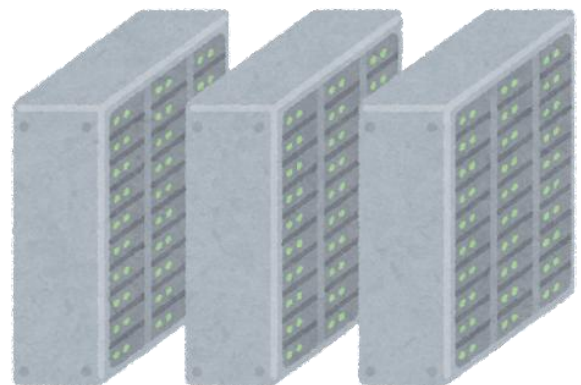
AI Bridging Cloud Infrastructure (ABCI) を使用し、Fully Sharded Data Parallelism (FSDP)、Flash Attention、vLLMなどの技術をシングルおよびマルチGPU環境において実装し、精密な性能検証を実施しました。

LLMモデルはLlama2を利用し得られた学習と推論のベンチマークをもとに、効率的なLLMトレーニングのための最適手法を探求しました。

また、実装過程で得られた洞察は、実際の運用に際しての重要なポイントとしてまとめ、報告書に記載しました。

この研究は顧客との対話を基に進められ、定期的に行われた報告会で顧客からの質問に答え、関心事項について議論しました。

このようなコミュニケーションの取り組みにより、研究の成果は顧客の期待に応えるものとなりました。



開発例

項目	詳細情報
OS	Linux Windows
開発期間	6ヶ月
開発人数	3人
開発言語	Python
関連技術	Llama2 PyTorch Transformers PEFT FSDP Flash Attention vLLM xFormers DeepSpeed

LLMの高速処理技術に関する調査 MoE

大規模言語モデル(LLM)の登場は社会に大きなインパクトを与え、LLMを使用したサービスは瞬く間に普及おります。それに伴い、LLMの性能向上や効率化に関する研究開発も活発に進められています。

しかし、LLMそのものの開発には膨大な計算リソースが必要という大きな課題があります。その解決策の一つとして注目されているのがMixture of Experts(MoE)です。実際に、MoEを採用したDeepSeek-R1は高いパフォーマンスを示し、大きな注目を集めました。

本調査ではMoEを中心として多くのモデルや研究について調査しました。MoEではモデルの内部に「エキスパート」と呼ばれる複数のサブネットワークを配置することで、効率の高い学習を可能にしています。その一方で、エキスパート間のデータ送受信がボトルネックとなり学習効率の低下を引き起こす課題があります。そこで、現在採用されている通信手段や、最新の研究における解決策について調査を行いました。

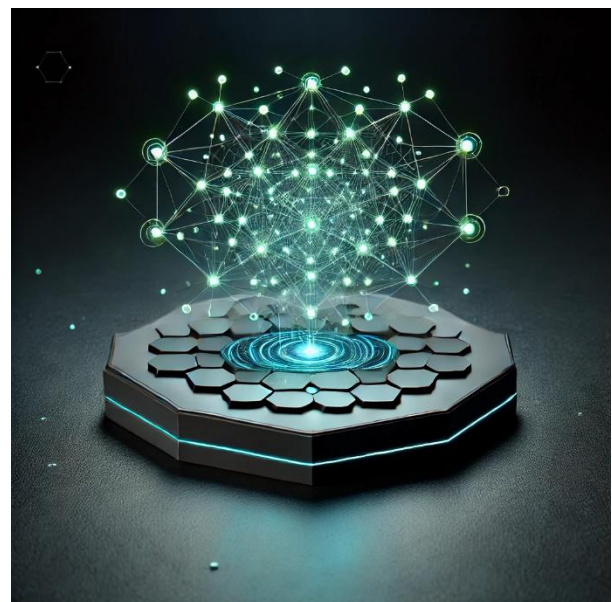
また、エキスパートの最適化に関する調査も実施しました。具体的には、エキスパートのサイズ設定、配置数、データのルーティング方法など、MoEの性能を最大限に引き出すための最適化手法について調査しました。

実運用を叶える様々な基礎要素技術の獲得・調査に取り組む

LLMの開発手法としては、事前学習によるモデル構築だけでなく、既存モデルのファインチューニングも重要です。そこで、ファインチューニング技術に関する調査を行い、一部の手法については実際にコードを実行し、動作検証を行いました。また、必要に応じてプロファイリングを取得し、コードの処理内容を詳細に分析しました。

LLMの計算不可の問題は学習時だけでなく、推論時にも大きな課題となります。その対策として、近年急速に注目を集めているSpeculative Decodingについても調査し、動作検証を行いました。

さらに、MoEと類似した特徴を持ちながら異なるアプローチをとる技術として、新たに登場したMixture of Depth (MoD)についても調査を実施しました。幅広い研究の進展をリアルタイムに追跡し、顧客との対話を通じて研究の方向性を議論しました。



開発例

項目	詳細情報
OS	Linux、Windows
開発期間	1年
開発人数	1人
開発言語	Python
関連技術	PyTorch Megatron-LM Hugging Face

LLM向け 学習後量子化の開発・実証

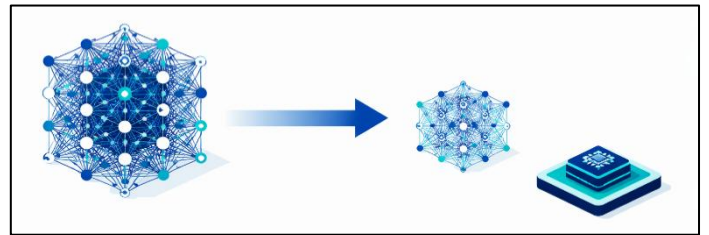
生成AIの普及に伴いオンプレミスやエッジ環境でのLLM活用が進んでいます。

一方で、モデル容量や計算・メモリ負荷が増大し、コストとレイテンシの面で課題が生じやすくなっています。また、データ保護など理由にローカルで軽量の推論を行いたいというニーズも高まっています。

追加学習を必要としない学習後量子化は導入障壁を下げる手段の一つですが、ビット幅や対象層で品質が変わるため、用途に応じた評価が欠かせません。

本取り組みでは、研究開発・実証（PoC）の観点から、学習済みモデルの学習後量子化で表現を圧縮し、推論パイプラインまで含めて検証できる枠組みを進めています。

単一アルゴリズムに閉じず複数の量子化アプローチを比較・組み合わせ可能とし、代表的小型LLMで動作と品質傾向を確認しています。

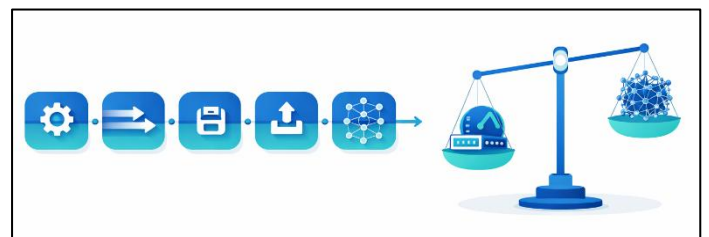


多手法対応の量子化パイプラインと、実運用を見据えた検証

具体的には、線形層などの層単位量子化、キャリブレーションに基づく手法、量子化誤差の扱いに関する知見を、変換から推論まで一連の手順で追いやすい形に整理し、既存のモデル変換および推論の流れに沿った検証を実施しています。

オープンソースのモデルと学習済み重みを前提に、設定変更から変換・保存・読み込み・推論までを同一の評価軸で実施したり、手法ごとに異なる前提条件を踏まえ、実験条件（対象層、ビット幅、キャリブレーション規模など）を明示し段階的に試したりすることで、再現性と比較の公平性を確保しています。

検証では、限定的な層やビット幅の設定から広げ、品質指標と実行コスト（時間・メモリ）のトレードオフを把握し、用途に応じた設計判断の材料を整理します。



開発例

項目	詳細情報
OS	Linux
開発期間	6か月
開発規模	5人
開発言語	Python、PyTorch
関連技術	- 深層学習 - 大規模言語モデル（LLM） - 学習後量子化

併せて、推論基盤・実行環境との組み合わせによる差や、変換後の互換性・運用時の切り分けといった導入論点も、実証の範囲で確認しています。

LLMを使用したスポーツ試合映像分析

大規模言語モデル（LLM）は、高性能な言語処理が可能であるとして近年注目を集めています。当社ではLLMを視覚処理分野に応用し、スポーツの試合動画内で起きている事象を分析、抽出するために活用しました。2023年2月にリリースされた軽量ながら高性能な推論が可能とされているオープンソースLLMであるLlamaにいち早く着目し、視覚処理分野に応用できないかのアプローチを行いました。

深層学習モデルを採用する場合において、学習データの準備の困難さは、ほぼ必ずついてまわる問題です。LLMを活用することで、スポーツの学習データを用意することなく高度な分析が実現可能かを検証しました。

本ケースでは、特にLlamaの使用によりスポーツ以外の動画像分析にも高い精度で活用できるという考察を得ることができました。さらに、ただの検証で終わるだけではなく、Llamaをユーザーフレンドリーに評価できるアプリの開発を行い、今後より深く広い検証に取り組める拡張性を持った研究開発を実施しました。



大規模言語モデルを動画分析モデルに機能拡張

本調査では、先述した通りスポーツの試合動画分析を高精度に行うことを目標としました。Llamaには「LLaMA -Adapter」という機能拡張ツールが備わっており、それを活用することで本件の目標達成を模索しました。このツールにより、画像データだけでなくテキストデータを含めたVQA形式と呼ばれる入力で推論することが可能となるためです。

これによって、一般的な画像キャプション技術を使うと出力結果をコントロールするのが難しいという問題を、理想とされる結果に近づけることができるのではないかと仮説のもと検証を行いました。

新しいオープンソースLLMが毎月のように発表されておりますが、その中でも今回は一定の評価を得ているLlamaにて具体的な検証結果を得ることができました。

コンピュータマインドでは今後も特徴を持ったLLMモデルをキャッチアップし、顧客の目的に応じたものを選択・提案できるよう致します。

開発例

項目	詳細情報
OS	Windows、Ubuntu
開発期間	12ヶ月
開発規模	1人
開発言語	Python
使用ネットワーク	Llama、LLaMA-Adapter
GPU	A6000
関連技術	深層学習

微小領域のセグメンテーション研究開発

セマンティックセグメンテーションは、画像内の一つ一つのピクセルに対して分類を行う領域分割の技術です。画像分類は、1枚の画像がどのクラスに属するかを判定するモデルですが、セマンティックセグメンテーションは、“画像のどの領域に何があるか”を認識することができるため、より詳細な情報を得ることができます。セマンティックセグメンテーションはその性質から、自動運転、医用画像の診断、外観検査等、様々なシーンで幅広く使用されています。一方で、検知対象の領域が小さく、背景領域が大部分を占めるような、クラスの存在比率が不均衡な画像については、学習及び推論がうまくいかないことがあります。

クラス不均衡な画像の例として、医用画像における病変部位や、外観検査における検査対象物体についての微小な傷・欠損等が挙げられます。また、セマンティックセグメンテーションのモデルは、エッジデバイス上で推論動作させるにはサイズが大きく、速度要求を満たさない場合があります。本事例では、クラス不均衡データへの対策と、エッジデバイスで動作させるためのモデルの軽量化に取り組みました。



クラス不均衡データセットの学習及びJetson上での動作

本事例で対象となるデータは、検知対象となる注目クラスの存在比率が極端に小さく、画像内の大部分が背景領域で構成されている画像でした。このようなクラスの存在比率が不均衡なデータセットで学習を行うと、損失関数を最小化しようとした結果、全てのピクセルを多数派クラスに分類するモデルになってしまう可能性があります。

本事例では、不均衡データセットに対する対策として、以下の取り組みを行いました。

- 画像分割による背景領域の削除（後処理で分割画像の統合が必要）
- 2段階方式モデルの適用（背景領域とそれ以外の領域を分類する前段モデルと、背景領域以外の領域を分類する後段モデルの構成）
- データ不均衡に対応した損失関数の適用

また、最終的に開発したモデルは、Jetson AGX Orin上で動作させる必要がありました。Jetson上でモデルを動作させるためには、メモリ容量の制約や実行速度の観点から、モデル軽量化が必要でした。そこで、TensorRT変換技術を用いてモデルの量子化を行うことにより、Jetson上で性能を大きく低下させることなく、モデルを高速に動作させました。

開発例

項目	詳細情報
環境	Ubuntu 22.04.3 LTS(WSL) Jetson AGX Orin
開発期間	3ヶ月
開発規模	1人
開発言語	Python、PyTorch
関連技術	・セグメンテーション ・クラス不均衡データセットの学習 ・TensorRTによる量子化 ・Jetsonでの動作

小物体の検知を目的とする物体検出

YOLOをはじめとするDeep Learningを利用する物体検出技術は、高精度かつリアルタイム処理が可能であるため、幅広いシーンで利用されています。

一般的な物体検出タスクのデータセットでは、対象物が比較的大きなサイズで画像内に存在します。例えば、車載カメラで撮影した前方を走る自動車や歩行者などの物体が挙げられます。一方、実社会では、画像に大きく映り込む物体だけでなく、小さな物体を検出したいというニーズも多く存在します。例えば、外観検査における微細な傷の検出などがその一例です。しかし、公開されている既存のモデルを使用するだけでは、小さい対象の検出が上手くできない場合があります。

そこで当社では、小物体の検出精度向上を目的とし、手法の調査や実装に加え、既存の手法の改良や組み合わせを行うことで、精度向上に取り組んでいます。

小物体の検知への取り組み

小物体を対象とする物体検出に関する当社の取り組みを紹介します。
まず、Deep Learningを利用した小物体の検出手法を調査し、下記に示す手法を採用しました。

■Attention機構を利用した手法

YOLO内部にAttention機構を組み込むことで、小物体に対する注目が強化され、より小さな物体に対する検出が促進される手法

■クロップによる対象物の拡大を利用した手法

学習時にデータをクロップして画像内の小物体の比率を高め、小物体の特徴を学習しやすくし、推論時には対象画像を一定数に分割し、拡大して推論を行う手法

これらの手法を実装してパラメータ調整等を行うことで検出精度の向上を確認しました。



開発例

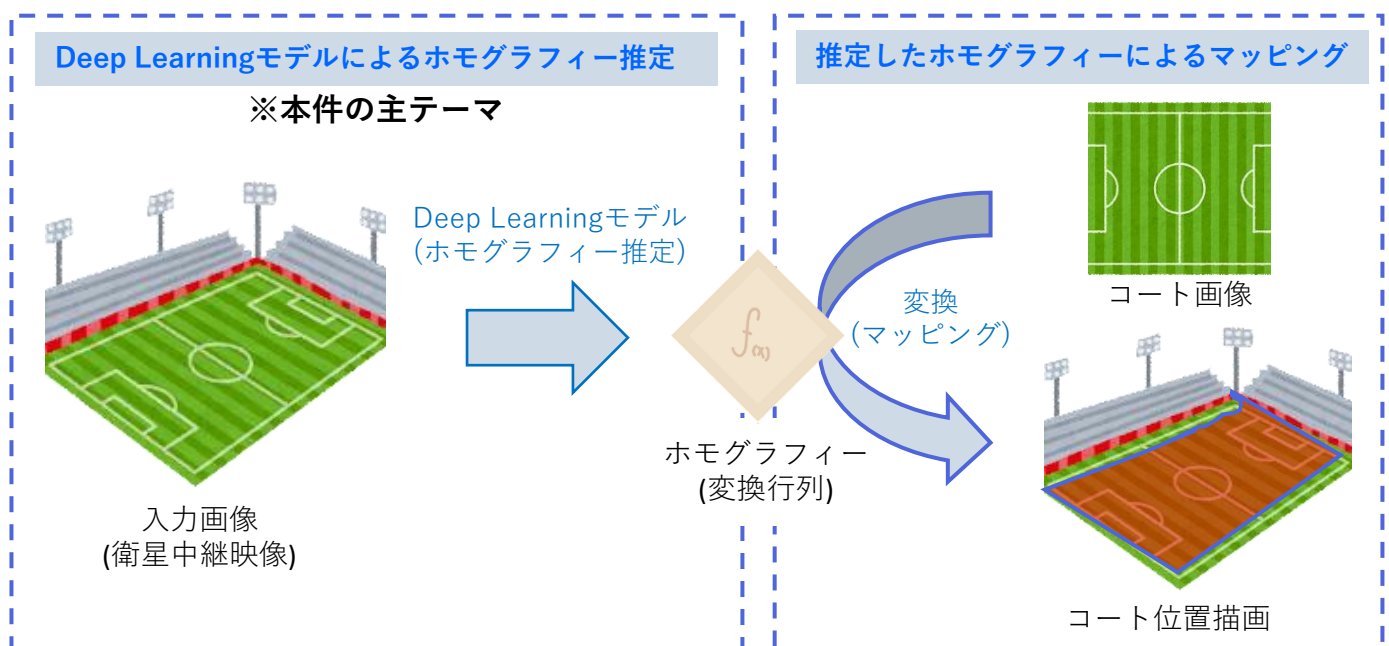
項目	詳細情報
開発状況	PoC
OS	Linux
開発期間	2か月
開発規模	1人
開発言語	Python
関連技術	・ 深層学習 ・ 物体検出

スポーツ中継映像におけるDLホモグラフィー推定

スポーツ（サッカー）の試合においてテレビ局による衛星中継が行われることが増えている中で、映像内のコート位置の特定は重要なタスクとなっています。

映像内のコート位置を特定する技術の一つにホモグラフィー推定があります。ホモグラフィーとはコートのみを描画した画像と映像内のコートとの対応を変換行列として表現したものです。

本件ではDeep Learning (DL) 技術を用いて、映像からホモグラフィーを自動で推定するモデルの調査、既存システムへの組み込みを目指した実装を行いました。



論文調査とシステム設計

本件は以下のフローで進めました。

1. DLによるホモグラフィー推定技術の論文調査
2. 1.で調査した論文に関するコード / 性能調査
3. 使用するDL技術の選定
4. 既存システムへの組み込み設計 / 実装

使用するDL技術は推定精度、実行時間、ライセンス等を基準として選定しました。

また、最終的にはエッジデバイス上での動作を想定しているため軽量なモデルであるかも選定基準の一つとなっております。

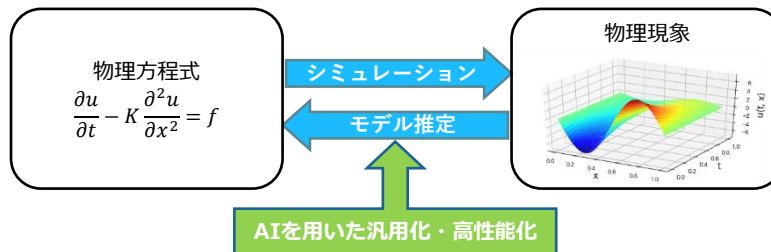
開発例

項目	詳細情報
OS	Windows
開発期間	6ヶ月
開発人数	1人
開発言語	Python
関連技術	ホモグラフィー エッジデバイス PyTorch キャリブレーション

AIを用いた物理方程式解析

物理現象を分析し、将来の状態を予測する手段として、物理理論に基づいて立式された物理方程式を解くことが現在の主流となっています。

しかし、一般に物理方程式を解くためには高度な知識と技術が必要であり、その専門性をAIの利用で解消する研究が注目を集めています。



本開発では、そのような最新の研究を対象として、論文を元に実装して性能評価を行っています。

そこで発覚した問題点については、AI・物理・数学といった関連分野での多角的な視点で原因を分析し、問題を解決する改良手法を考案してその効果を実証しました。

また、この手法の構造と特性を観察し、物理現象の解析に限らない広い分野に応用する実験も行っており、現在は統計的推定の問題をDeepLearningによって解く研究を進めています。

最新手法の調査と改良

論文で発表される最新の手法は、いずれも従来手法の問題を解決し得るものですが、現実の問題に適用するためには、その性質を明らかにして問題点を洗い出し、改良を加える必要があります。

そのため、論文で述べられた手法を実装するだけでなく、解析対象となる物理理論、手法の拠り所となるAIの理論、そして、多様な理論の共通言語である数学の理論に基づいて手法を分析し、いくつかの改良手法を考案しました。

それらの手法によって、対応する問題が解決することも実証しており、理論に基づく分析とデータに基づくAI処理の融合が効果的であることを示しています。

開発例

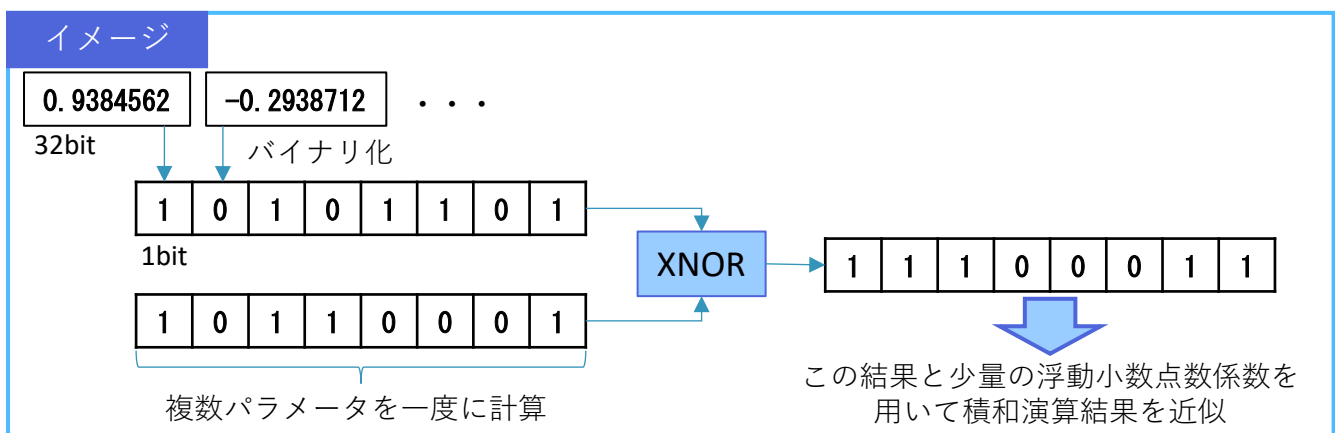
項目	詳細情報
OS	Windows
開発期間	24ヶ月
開発人数	1人
開発言語	Python、Matlab
使用ネットワーク	MLPをベースにした独自ネットワーク
関連技術	DeepLearning Tensorflow (low-level API) 偏微分方程式 統計的推定

深層学習バイナリモデルのエッジデバイス実装評価

AIというと高性能なマシンでの運用が必要なイメージがありますが、近年で開発の進む自動運転やIoTデバイスなどのように、より現場に近いデバイスに実装しAIによる判断を行うことへのニーズが高まっています。学習自体は高性能なマシンで行いAIによる判断は小型で軽量のエッジデバイスで行う、Deep Learningの現実的な実装評価に取り組んだ事例です。エッジデバイスの場合は小型・軽量ゆえに搭載メモリや処理速度の面で制限があり、推論処理を行う際には省メモリ化と高速化が課題となります。より少ないビット数で深層学習のパラメータを近似する量子化技術の中でも、計算量の多い畳み込み演算部分の入力・重みをバイナリ (1 bit)とし、わずかな浮動小数点型(32bit)の係数で積和演算結果を近似することで省メモリかつ高速に推論を行える XNOR-Net に着目し実装と検証を行いました。

バイナリ化のメリット

- パラメータのバイナリ化
 - …浮動小数点型(32bit)をバイナリ(1bit)化することで、メモリ容量と回路面積を削減
- 浮動小数点型(32bit)での積和演算を並列のビット演算にして高速に
 - …大量に発生する積和演算処理で時間がかかる浮動小数点型の乗算の一部を、ビット演算（加算とビット単位の否定排他的論理和 [XNOR]）に置き換えることで処理の高速化が期待できます



バイナリモデルの適用先

XNOR-Netをはじめとするバイナリモデルを適用させることで、エッジデバイス (FPGA)での推論の高速化と、計算時に確保するメモリ削減が期待できます。またバイナリ化の効果が表れるのはエッジデバイスだけではなく、PCのCPU環境であればSIMD命令を使うことで同様のモデルを実装することも可能です。NVIDIA社製のGPU環境であればTensorCoreを用いたバイナリ演算の高速化など、様々な用途への適用も考えられます。

開発例

項目	詳細情報
OS	Ubuntu
開発期間	6ヶ月
開発人数	1人
開発言語	Python、C
対応ネットワーク	AlexNet 独自作成のCNN
エッジデバイス (FPGA)	Ultra96v2
関連技術	DeepLearning、Vivado、 FINN、Brevitas、 TensorCore

充足性判定に向けたニューロンカバレッジの実装評価

Deep Learningシステムの開発フローにおいて、質の良いデータが十分に準備されていることは良いモデルを生成するために非常に重要です。

当社では、データが十分に準備されているかを評価する指標として、ニューロンカバレッジあるいはその派生形が有効であるかを判定するための実装評価を行いました。結果として、評価したどの指標もデータが充分であることを判定する可能性を示すことができました。

データ充足性というテーマは、世界的にニーズがあり今後も注目度が高いであろう興味深いテーマです。

ニューロンカバレッジとデータ充足性の関係

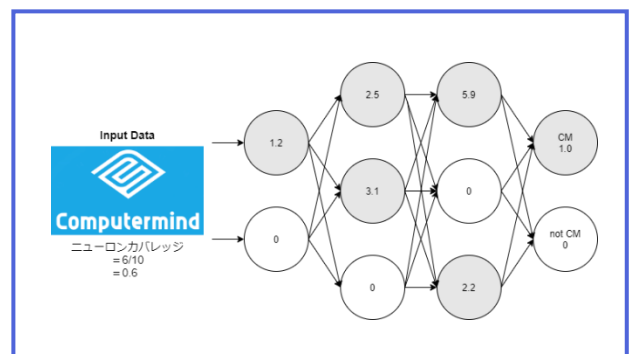
ニューロンカバレッジはモデルのニューロンの発火状況を数値化した指標です。

様々なデータが存在することにより多くのニューロンが発火する経験を得るという前提のもと、ニューロンカバレッジが大きいとデータが充分であると言えるだろうと推測されます。

この仮説を検証するために、質の良いオリジナルのデータセットと一律変換を加えた質の悪いデータセットを準備しました。

準備した質の悪いデータセットに対してニューロンカバレッジを評価すると、オリジナルの質の良いデータセットで評価した際の数値より減少傾向にあることが確認されました。

本件の調査では、傾向がみられることの確認に留まりましたが、ニューロンカバレッジのデータ充足性判定への適用可能性を示すことができました。



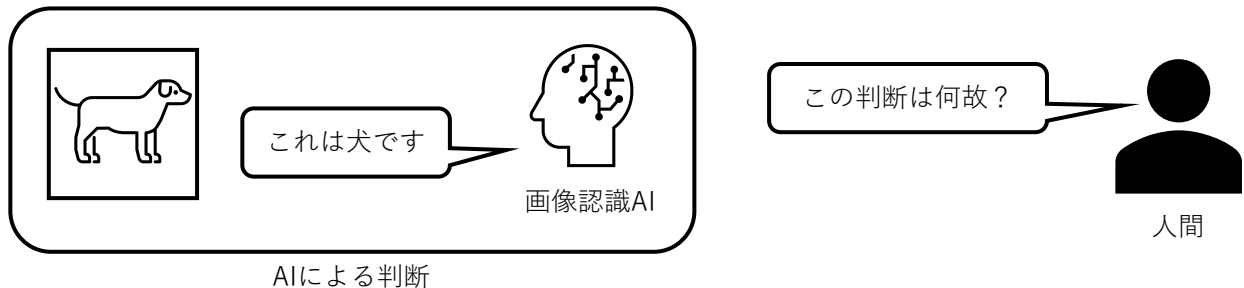
オリジナル データ破損 ノイズ付与 ぼかし

開発例

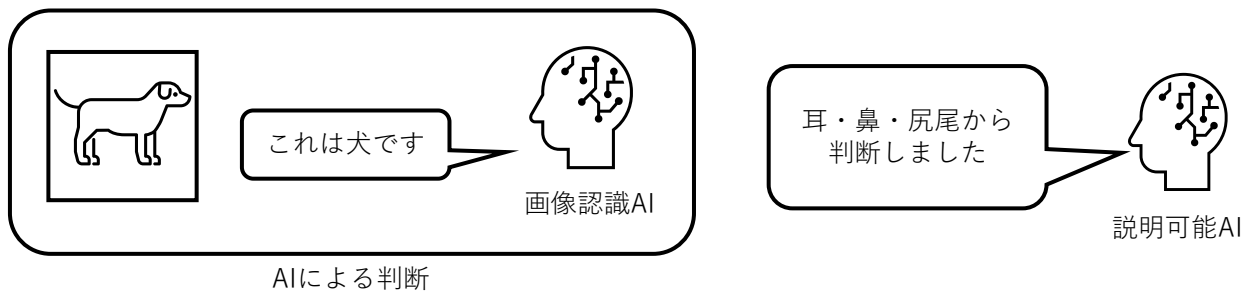
項目	詳細情報
OS	Windows
開発期間	12ヶ月
開発人数	1人
開発言語	Python
関連技術	DeepLearning Data Augmentation Adversarial attack

説明可能AI

DeepLearningを始めとするAI技術の発展により、大規模なデータに対して単純なルールベースの手法では実現できない複雑な判断を自動で行えるようになりました。しかし、それに伴って予測アルゴリズムも複雑化しており、単にその手続きを追うだけでは「何故AIがそう判断したのか」という判断結果の説明が困難になっています。

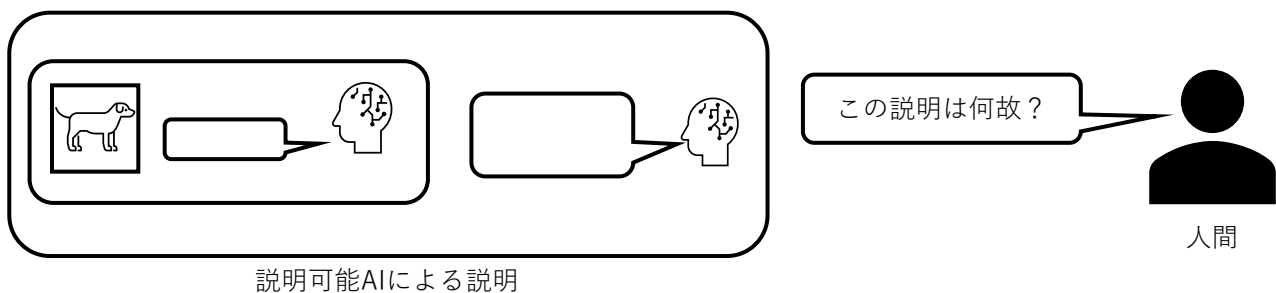


この問題に対して、AIの予測アルゴリズムから人間に解釈可能な情報を抽出する説明可能AIの技術が研究されており、AIの判断に説明性を持たせる試みが盛んになされています。



AIの説明結果についての説明の必要性

説明可能AI技術はAIモデルが持つ数理的な性質を抽出するもので、「説明可能AIが何を説明しているか」については人間が妥当な解釈を考える必要があります。



当社は説明可能AIのアルゴリズム解析も行い、当該技術がどのような説明を行っているかの解釈、及びに必要な説明を得るためにアルゴリズムを改良する調査・研究開発を行っています。

開発例

項目	詳細情報
OS	Windows
開発期間	12ヶ月
開発人数	1人
開発言語	Python
関連技術	DeepLearning GradCAM

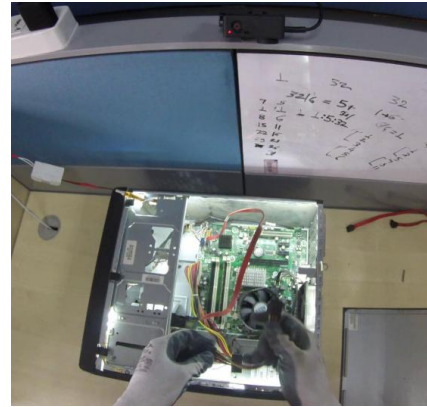
Egocentric Vision

一人称視点映像を用いた作業分析

工場などの製造現場では、作業者がどのような作業を間違えやすいか、どのような作業に時間を要しているかなどを把握することが、生産性向上や品質管理において重要な課題となっています。そこで近年注目されているのが、一人称視点映像（Egocentric Vision）を用いた作業分析です。作業者の頭や首に小型のカメラを装着することで、作業者自身の視点から手元の動作を詳細に記録・分析することが可能になります。



固定カメラ映像イメージ



一人称視点カメラ映像イメージ

作業ミス防止と工程進捗の自動把握を実現するAI技術の検証

本事例では、作業の誤りを検知する技術と作業の完了を検知する技術を調査しました。作業の誤りを検知する技術では、AIモデルが検知した作業の特徴量とその作業を代表する特徴量がどれだけ乖離しているかを算出することで、AIモデルが検知した作業が誤って実行されたものであるかを判定します。

作業の完了を検知する技術では、組み立て作業において、物体検知モデルによって組み立て品の状態遷移を捉え、ある状態から別の状態に遷移したときに遷移に必要な作業を完了した作業として判定します。



作業完了検知技術のイメージ図

引用元：IndustReal: A Dataset for Procedure Step Recognition Handling Execution Errors in Egocentric Videos in an Industrial-Like Setting, Tim J. Schoonbeek et al., 2023

開発例

項目	詳細情報
OS	Ubuntu 20.04
開発期間	6か月
開発規模	1人
開発言語	Python
関連技術	・アクションセグメンテーション ・対照学習 ・物体検知 ・グラフ畳み込みネットワーク

Egocentric Vision

一人称視点映像における物体検知技術の調査

前頁の作業検知技術の調査案件から継続作業として、作業者の手元の物体検知に関する調査を行いました。

作業者の手元の状況を分析するには、以下の工程を解析する技術が必要です。

- ・手の検出、物体の検出
- ・作業動作の認識
- ・作業工程の分析
- ・異常・ミスの検出

本事例では作業者がどんな部品を持っているのか一人称視点カメラ映像から解析するため、作業者の手と手が触れている物体をセグメンテーションマスクするAIモデルの調査を行いました。また、触れている物体の領域を検出するだけでなく物体名を検出するようなモデルの調査も行いました。

手と接触物体のセグメンテーション技術の調査

AIモデルの調査では実際に実装を行って手と接触物体のセグメンテーションマスクの精度を確認しました。このAIモデルは左手と右手、左手で触れている物体、右手で触れている物体、両手で触れている物体を検知できます。

一人称視点の映像では作業者の移動に合わせてカメラが移動するため、物体が画面外に見切れたり、作業者の手が物体とかぶっていたりして、物体の見え方が変わりやすいです。精度を確認する際はそのような状況でも手や物体を正しく検知できているかを確認しました。

精度を確認した後は、実運用への導入を見据え、精度向上のためのさまざまな手法を提案・検証しています。例えば、YOLOやSAMなど他のAIモデルと組み合わせて領域をより正確に検知することが可能かどうか調査したり、AIモデルのファインチューニングを行ったりして、それぞれの効果を定量的に比較・評価しました。

開発例



項目	詳細情報
OS	Ubuntu 20.04
開発期間	9か月
開発規模	1～2人
開発言語	Python
関連技術	・物体検知 ・セグメンテーション ・一人称視点映像 ・ファインチューニング

GPGPU

～ General-purpose computing on graphics processing units ～

AI（DeepLearning）で使用されているGPUですが、豊富なコア数を有効活用して、演算処理の高速化が可能です。

GPUを効率よく使用するためには専用言語《CUDA》で実装する必要があります。

当社では、いち早くGPUによる高速化の優位性を見出し、案件を成功させて来ました。

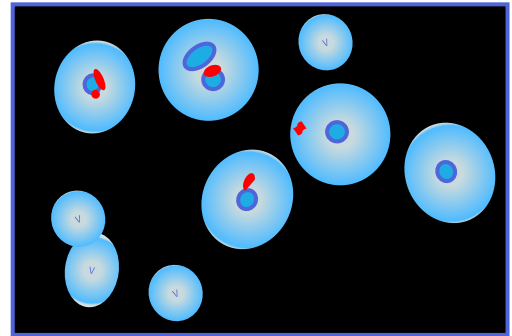
FISH-QUANT CUDA開発

FISH-QUANTと呼ばれるTranscripts(DNA塩基配列)をカウントするアルゴリズムがMATLABで実装されていたながら、

- ・処理時間が過分に掛かっている
- ・MATLABのライセンスを持っていないと利用することが出来ない

という点で有用でありながら使いづらい問題がありました。

そこで、C++/CUDAで解析処理部を実装し、C#で画面を実装することで、MATLABを使わずにFISH-QUANTが利用でき、従来のアプリケーションよりも高速に解析ができるようにしました。

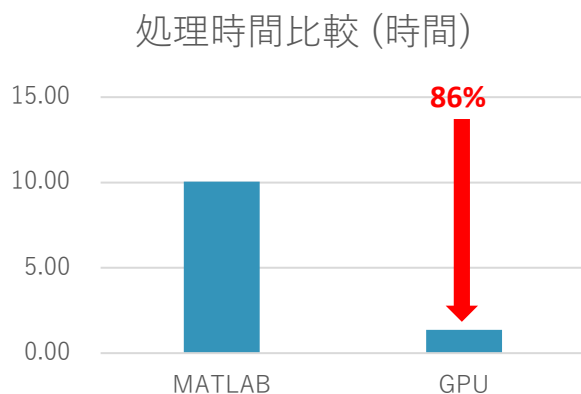


C++/CUDAに置き換えることで7倍の高速化

三次元フィッティングを行っている箇所をC++化して、処理が重くなっているフィッティング処理をCUDAで並列化しました。

MATLAB上では、処理時間が10時間掛かっていましたが、C++/CUDAで処理を置き換えたことで7倍の高速化を達成し、1.4時間で処理を完了させることが出来るようになりました。

また、処理の置き換えによる計算誤差についても検証を行い、90%近くの輝点がオリジナルと同等かそれ以上の精度であることを確認しました。



開発例

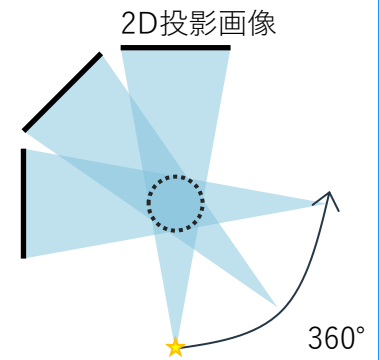
項目	詳細情報
OS	Windows
開発期間	3ヶ月
開発人数	1人
開発言語	C++、CUDA、C#
GPU	開発：Quadro M6000 動作：Tesla V100
関連技術	MATLAB 3D Gauss Fitting

X線CT画像3D復元高速化

レントゲン撮影は1方向からX線を照射し、被写体内部の2次元写真を撮影します。
X線CTスキャナでは360° 全方向からX線を被写体に照射し、数百～数千枚の2次元投影画像を取得します。

取得した投影画像を再構成(コンピュータ処理)し、3次元画像(2次元輪切り画像の積み重ね)を生成します。

X線CT画像の再構成には非常に時間がかかる上に、画質を良くするために補正処理を繰り返し行う必要があります。



実際のシステムに載る前のプロトタイプ開発として、GPGPUという技術を用いて時間のかかる再構成～補正処理の高速化を実現しました。

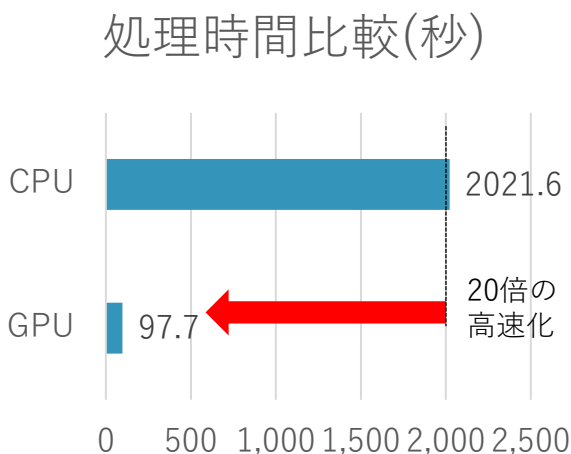
CUDA/C++ による高速化

X線CT画像を3次元復元するための再構成～補正処理の内、処理の重くなっている箇所をCUDAで並列処理することで処理を高速化しました。

再構成処理単体で約43倍の高速化を実現し、処理全体で約20倍の高速化(例：30分→1.5分)を実現しました。

実運用では大量にメモリを使用する処理のため、データフローについての調査と対処方法の検討まで含めて報告しました。

開発例



項目	詳細情報
OS	Win 10 Pro
開発期間	2ヶ月
開発人数	2人
開発言語	CUDA/C++ (VS2015 pro)
CUDA	9.2
GPU	開発：GTX 1080Ti 動作：GTX 1080Ti
関連技術	GPGPU

3Dデータ解析

～ 3D data analysis～

近年、医療・製造・建築・ロボット・地理情報など様々な領域で活用されている、3Dデータを使用したソリューションの提供を行っております。

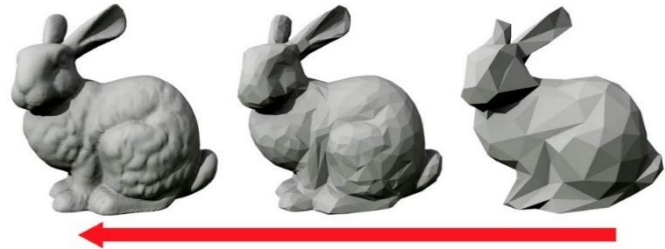
また、Visual-SLAMやNeRFなど、AI技術を取り入れて、画像等の2Dデータから3Dデータを作り出す技術も注目を集めております。

弊社では、3Dデータを活用した、様々な課題解決ソリューションのご提案いたします。

点群データ関連最新技術の調査・実装

現実世界の対象物を3次元カメラ等で撮影し取得した点群データを元に、地形や構造物、生き物の3Dポリゴンメッシュの高解像度化や修復等を行う様々な3次元処理の手法を研究してきました。

点群データは主に建築や自動運転の分野で広く取り扱われていますが、他にも点群ベースの姿勢解析アルゴリズムや、セキュリティ、外観検査への適用など、その用途は多岐に渡ります。

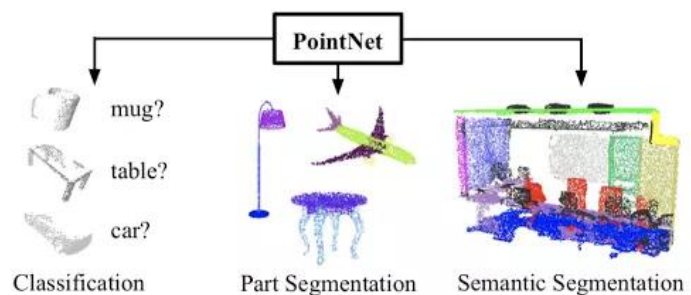


また、近年では点群データを対象とした深層学習モデルにも多数のアプローチが存在しているため、それらによって点群データの分類やセグメンテーションを行うためのより良い手法が常に議論され続けています。

論文解析から実装、改良までを広く対応 製品開発のための技術課題を解決

ソフトウェア、ハードウェア製品開発のための3次元点群処理のトレンドを追い、またその性能評価を行うことで、実用レベルでの手法の検討を行ってきました。

その他にも、複数視点の点群データを統合することによる高解像度な3Dオブジェクトの構築や3Dアニメーションに代表されるポリゴンの変形技法、3D点群のセグメンテーションや点群補完など、様々な技術の実装、評価を行ってきました。



開発例

項目	詳細情報
OS	Windows 10 Ubuntu 16.04 LTS
開発期間	36ヶ月
開発人数	2人
開発言語	C++, C#, python
関連技術	3D reconstruction Open3D 点群マッチング PointNet

また、GPGPUによる高速化や手法の改良、組み合わせによる高精度化にもトライしており、常に進化し続ける3次元処理の分野における技術課題解決をサポートしています。

ドローン制御用自己位置推定モジュール

ドローンの利用方法は現在、様々模索されています。その中でもここでは「巡回監視」を取り上げます。カメラ搭載ドローンを一定のコース内で巡回させ、周囲の様子を撮影・記録することが目的です。

この時、ドローンはもっぱらGPSを使用して自分の現在位置を特定し、適切に方向転換を行います。しかし、電波の届かない閉所などではGPS衛星からの信号を受信できず自己位置を特定できなくなります。

当社では、単眼カメラ(レンズモジュールが1つのカメラ)を用いて、周囲の情景からドローンの自己位置推定を行い、ドローンの自己位置に変換することを目指しました。



GPS無しで自機の緯度・経度を推定、電波の悪い環境での機体誘導が可能に

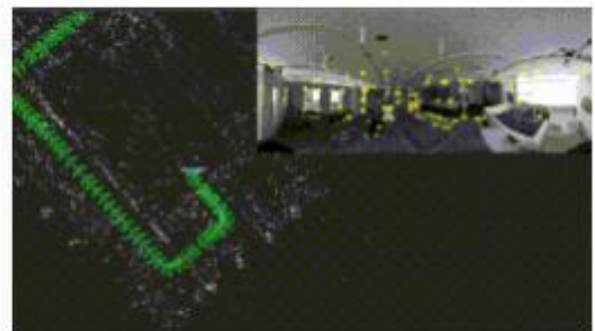
ドローンの自己位置をGPSによらず推定するため、OpenVSLAMを用いて単眼SLAMを行う処理を実装しました。

OpenVSLAMとは、カメラに映った画像から特徴点を抽出し、その移動の具合から周囲の地図を作成すると同時に自己位置の推定を行う、Visual SLAMと呼ばれる技術のC++実装です。

お客様の要望により、自己の緯度・経度を求めることになっていたため、ArUcoと呼ばれる技術を併用しました。これは、ARマーカと呼ばれる特殊な模様をカメラで撮影し、そのカメラに対する角度と紐付けられた番号を推定するものです。

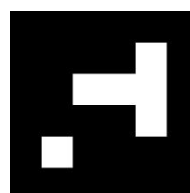
ARマーカの番号と緯度・経度を対応付け、SLAMで求めた自己位置と組み合わせることで、GPSに頼ることなく自己の緯度・経度を求めることが可能となりました。

本開発では、お客様のご要望により、全体をROS2のノードとして実装しています。ROS2ノード単体では、Raspberry Pi 4上で10FPS程度で動作します。



開発例

項目	詳細情報
OS	Ubuntu
開発期間	2ヶ月
開発人数	3人
開発言語	C++
関連技術	OpenVSLAM ArUco ROS 2



CC BY-SA

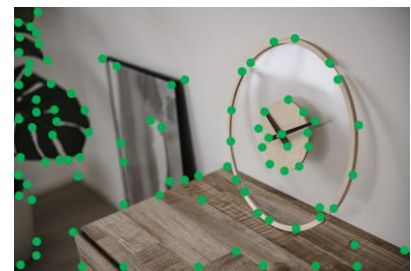
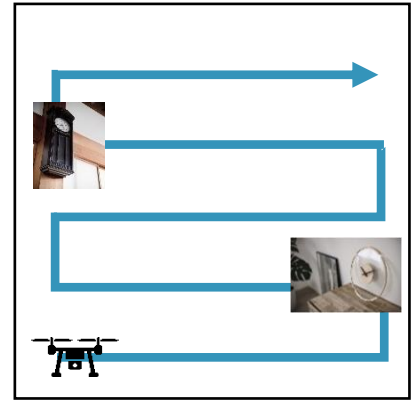
AIによるVisual SLAM技術の検証

自動運転やロボット掃除機などでは、自分の位置を把握するため、【SLAM】と呼ばれる自己位置推定技術が必要不可欠となっています。

現在SLAMを行う場合、LiDAR等の3次元計測デバイスを使用するケースもありますが、3次元計測デバイスは安くとも数十万円代と比較的高価なデバイスとなっています。

対して、比較的安価なデバイスである2次元カメラ画像からSLAMを行うVisual SLAMによる自己位置推定の実現も近年まで盛んに研究が行われています。

本案件では、近年のデファクトスタンダードとなっている【AI Visual SLAM】手法をベースに、その発展形や別のアプローチを取っている様々な手法から特徴を調査し、比較対象として検証を進めました。



アルゴリズム単体で解決できない課題に対する改善提案

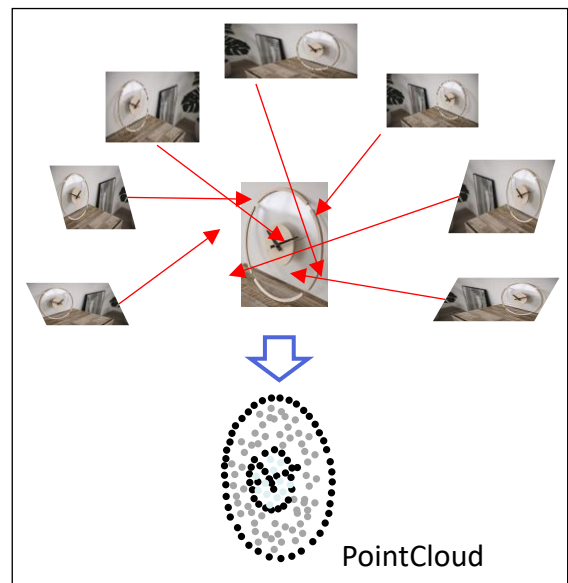
まず、AIアルゴリズムに基づく自己位置推定・マッピング手法についてトレンドを調査し、比較検証を行いました。

また、既存のAI Visual SLAM手法の精度検証だけでなく、各アルゴリズムをサービス化する際に課題となりうる改善点の提案と実装を行いました。

特に、Visual SLAMの課題であるマッピング点群のスパース性の解消という課題に着目し、別の3次元再構成手法と組み合わせた点群の高密度化についての改善を実施致しました。

高密度点群の生成にはカメラ画像と三角測量を用いた従来手法および、3Dレンダリングを主眼に置いた技術やAI・非AI手法など様々なアプローチがあり、手法の選定およびそれらの特徴を明らかにすることが重要となります。

高密度点群の手法の選定に際しては、課題の解決の観点だけでなく、汎用性・精度などの多角的な視点から、最適な手法を選択する必要があると考えております。



開発例

項目	詳細情報
OS	Ubuntu
開発期間	6ヶ月
開発人数	2人
開発言語	C++、python
関連技術	Visual SLAM 3D Reconstruction

大規模環境の三次元計測に向けたNeRF技術調査

大規模環境を三次元に捉えるために使用される、Structure from Motion (SfM)、Multi-View Stereo (MVS) という技術が存在します。これらはドローン等を用いて対象を多角的に連続撮影し、それらの画像から対象を3Dモデルとして復元する手法で、世界遺産の3Dモデル化などに用いられていることで知られています。

このような手法で大規模な環境を詳細に復元する場合、様々な角度、距離からの画像が大量に必要になります。

本件ではその画像の多様性をNeural Radiance Fields (NeRF) という機械学習手法を用いて補完できないかという観点でNeRFやその派生・後継手法について調査、実験を行いました。



NeRFとは

NeRFとは2020年に提案されたニューラルネットワークを用いた画像生成技術の一つです。撮影した画像とカメラパラメータからニューラルネットワークを学習することで、新たな任意視点からの対象の画像を生成できます。

あらゆる視点からの対象の画像を生成できるということは対象を三次元的に表現できるとも言えるため、NeRFモデルは点群データやメッシュデータ、ボクセルデータなどと並ぶ新たな3Dモデルと考えられています。



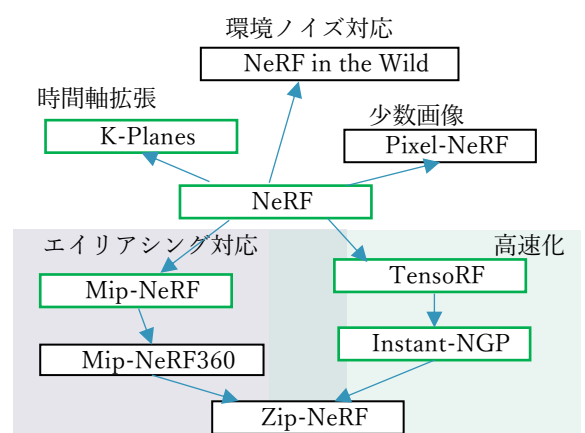
NeRFの派生・後継手法

NeRFを大規模環境に適用することを考えると、問題がいくつかあります。

- 規模に応じて学習にかかる時間が膨大になる
- データの粗密や質によりエイリアシングが発生しやすい
- 屋外であることが多いため、照明条件や天候など悪環境によるノイズが含まれる

これらは既存の3次元復元手法においても課題ですが、NeRFにおいても障害となります。

本件ではこれらを解決し得る派生・後継手法について調査し、実際の大規模データを用いてその効果を実験しました。



NerfStudio

本実験においては主にNerfStudio(<https://github.com/nerfstudio-project/nerfstudio/>)を用いて効果の検証を行いました。

NerfStudioはNeRFやその派生・後継手法に広く対応し、学習、評価、3Dモデルへの変換、高速化、ノイズ除去などのコンポーネントを使用できるAPIを提供しているOSSプラットフォームです。

電顕3次元再構築

バイオサイエンス分野において、生物組織の構造を三次元的に把握することは生命現象の理解に必要です。

生物組織の構造を三次元化するシステムは多く存在しますが、そのほとんどはデータ処理する過程で高性能なコンピュータを必要とすることと細胞構造に対する専門知識が必要であることが研究者にとって大きなボトルネックとなっていました。

これらの問題を解決することを目指し、一般的なWindows PCで機械学習を用いて簡単な操作で連続断層画像から効率よく生物組織の構造を三次元化が可能なシステムを開発しました。

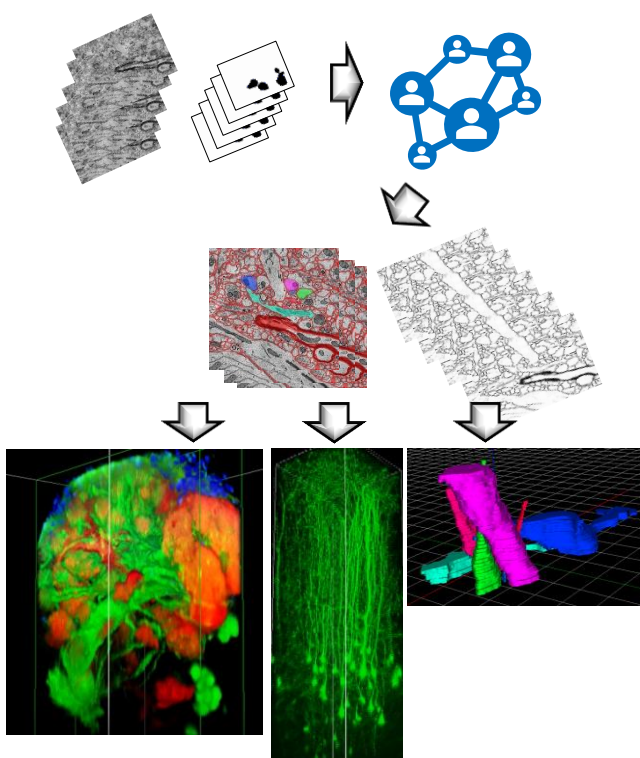
機械学習による効率化により少データ数からの3次元構築

操作が煩雑であること同様、3次元構築に必要な学習データの多さも研究者にとってのボトルネックとなっておりました。

本システムは機械学習を用いて効率よく生物組織の構造を抽出して三次元に再構築することが可能となり、10枚程度の学習データだけでセグメンテーションを行うことが可能です。

簡単な操作で、細胞の形状を視覚的に確認することができます。

本システムにより、研究者は操作方法の理解など、研究とは無関係な作業を省くことができ、多くの時間を研究に集中することが可能となりました。



開発例

項目	詳細情報
OS	Windows 10
開発期間	6ヶ月
開発人数	3人
開発言語	C#.net, C++, Python
使用ネットワーク	U-net
関連技術	動的輪郭 グラフカット (Graph Cut) LiveWire OpenCV OpenGL DeepLearning

宇宙空間の人工構造物の解析技術手法の調査・検証

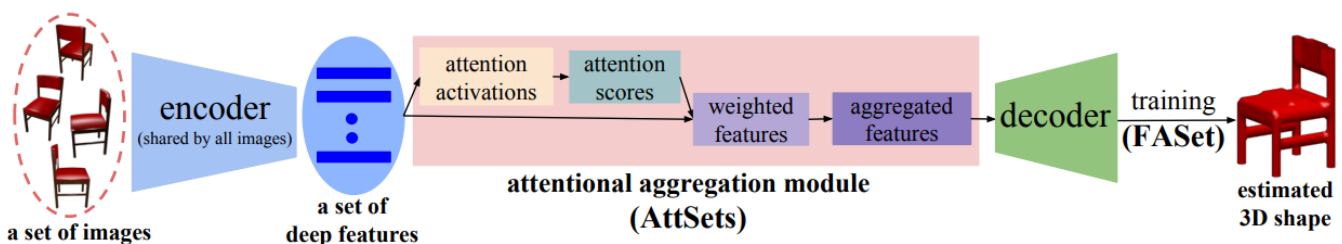
宇宙開発の進歩により宇宙空間には人工構造物が多数漂っています。ロケットや人工衛星などは他の人工構造物との衝突を避け安全に運用する必要があり、そのために人工構造物の観測と状態分析が不可欠となってきています。

宇宙空間における人工構造物の様々な観察方法の一つとして、地表から観測装置を用いて観察する方法が挙げられます。その方法には、宇宙空間という環境により観測画像が鮮明とは限らないという課題があります。さらに観測はできたものの、どのようにして姿勢推定や形状復元を行うのかという技術選定も必要です。

弊社では上記の背景を踏まえ、地上から観測された人工構造物の画像の鮮明化技術と分析技術について調査しました。具体的には、主に画像のボケ除去などを行う画像復元技術と被写体の軌道予測・姿勢推定・形状解析を行う画像解析技術について技術調査を行いました。さらに、それらの技術調査結果から選定した技術の実現性について調査しました。

特に「複数視点から撮影された2次元画像の入力による、3次元モデルの復元」技術に対する精度検証を経て、活用の可能性についての調査を実施しました。

画像と3Dモデル(Voxelデータ)を用いた形状解析の精度検証



人工構造物の実写シミュレーション画像とVoxel形式の3Dモデルを用いて形状解析を行い解析精度の検証を実施しました。論文や公開されたソースコードでの検証では、豊富な入力データで学習を行っていた一方で、今回の問題設定(宇宙空間の観測という取得難易度が高い画像を用いての解析)の制約上、弊社の行う検証では学習データの種類が限られるという壁がありました。

最初は公開されたソースコードとほぼ同条件で検証を実施しました。しかしどの画像で推論しても特定の学習データに近い形で曖昧に復元されてしまいました。そこで、敢えて過学習を行い、損失関数を工夫することで、学習データと同じ人工構造物に対しては形状復元を正確に行えることが確認できました。

しかし一方で、過学習により未知あるいは学習と異なる損傷状態の人工構造物の形状復元には課題があることも検証され、技術の活用のための課題を考察することができました。

開発例

項目	詳細情報
OS	Ubuntu 20.04.6 LTS
開発期間	6ヶ月
開発規模	1人
開発言語	Python
関連技術	- Deep Learning - Tensorflow - 画像復元 - 物体姿勢推定 - 軌道予測 3次元形状復元

3D CADモデル生成アルゴリズムの調査

CAD（Computer-Aided Design）とは、コンピュータを用いて設計や製図を支援するツール、あるいは設計すること自体を指します。本案件では、幾何学的深層学習を用いてCADデータを生成するアルゴリズムに関する調査を実施しました。

CADデータへの幾何学的深層学習(GDL)アプローチに関する現況

CADデータに対する従来の機械学習および深層学習によるアプローチは、主にクラスタリングタスクを扱うことが多く、その時には体積や密度といった物理的特徴に基づいた特徴表現学習に焦点を置くか、点群のような別のデータ形式で直接作業をしていました。その結果、モデルに埋め込まれた幾何学的特徴をとらえるには不十分な場合がしばしばありました。しかし近年、TransformerをはじめとするGDLの手法により、物理的特徴に関係なくCADデータの幾何学的特徴を学習する可能性を示すという研究がなされています。これらの特徴表現は、CADモデルの再生成や未完成のCADモデルの自動補完など、様々なCAD設計プロセスに応用が期待されています。

1枚の画像から3D CADモデルを生成

具体的な応用方法として、「画像から空間的な形状を推測し、その形状の3D CADモデルを構築するための設計手順を拡散モデルにより生成する」という手法があります。この手法を導入することで、高コストで手間のかかる3Dスキャン技術の代替となり、実世界の画像に対して高い汎用性を実現できます。



迅速な調査と確かな情報収集

3D CADモデル生成手法には、学習で用いるデータセットの不足や、モデルの複雑性に十分対応できていないといった課題が残されています。弊社では、これらの課題解決に向けて継続的に技術調査・分析・検証を実施することで、先端技術のキャッチアップ支援だけでなく、調査結果を開発判断につなげることができます。

開発例

項目	詳細情報
OS	Windows
開発期間	3ヶ月
開発規模	1人
開発言語	Python
関連技術	Geometric Deep Learning Transformer Diffusion Model CAD

ロボティクス

～ Robotics ～

様々な分野でオートメーション化の上で欠かせない存在となってきているのが、ロボット分野です。

AI、ROS等の技術を組み合わせて最適解をご提案差し上げています。

アームロボットを使用した強化学習

深層強化学習はカメラで撮影した画像などの複雑な情報を使って学習をし、動作を最適化する能力を持っています。

現在では、深層強化学習を利用したロボットの動作を最適化する研究が盛んに行われています。

ロボットに対して深層強化学習を適用する際に、以下のことが大きな障壁となります。

- ・学習に非常に多くの時間を要する
- ・学習中にロボットが破損する恐れがある

この解決策として、シミュレータ上にロボットを作成して学習する方法があります。ただし、現実の世界とシミュレータの世界では、取得できる情報が完全一致しているわけではないため、シミュレータ上で学習したモデルが、現実世界で上手く動作しないという問題が発生します。(ドメインシフト)

ある世界で学習したモデルを、別の世界に適応させることをドメイン適応と呼びます。当社では深層強化学習をシミュレータ上で実行し、そのモデルを現実世界のロボットに転送するドメイン適応に取り組んでいます。

Real World



Simulator World



シミュレータで高速に学習し、実機ロボットで動かす

深層強化学習で4軸ロボットアームに目的の動作を学習させる開発を行いました。

学習はシミュレータを用いてのみ行い、現実世界のロボットによる学習は一切行いませんでした。

そのため、ドメイン適応技術を使用せずに学習を行い、その学習モデルを実機のロボットに転送すると、ドメインシフトにより目的の動作をしません。

当社では、シミュレータ上での学習時にドメイン適応技術を使用することで、実機のロボットでも目的の動作を実行するモデルを作成しました。

開発例



項目	詳細情報
OS	Ubuntu 18.04 LTS
開発期間	6ヶ月
開発人数	1人
開発言語	Python
使用ネットワーク	Deep Q Network
GPU	開発: Nvidia Geforce RTX 2080Ti 動作: Nvidia Geforce RTX 2080Ti
関連技術	深層強化学習 ドメイン適応

人工衛星の制御ソフトウェア開発支援

近年、国内海外を問わず宇宙ビジネスが活発になっており、人工衛星による観測データの利用は農業、環境モニタリング、都市計画、災害対策などの分野で民間企業の参入も相次いでいます。

人工衛星には自身を制御するためのコンピュータが搭載されており、様々なソフトウェアが動作しています。

これはセンサーから取得したデータをもとに、人工衛星自らの制御を行うためのアクチュエータの出力を決定するもので、堅牢で信頼性のある設計・実装が要求されます。

本開発にあたっては、メンテナンスやデバッグがしやすく、複雑な機能群を管理できるシステムを提案し、効率的に開発できる環境を実現しています。



堅牢で柔軟性のある制御システムの実現

機能的かつ信頼性の高いソフトウェアへの要件をもとに、人工衛星制御の検証及びPoC開発を行いました。

フレームワークはROS2を提案し、その上で制御アルゴリズムの開発を行っております。

これにより、システムの一部を変更する際に他の部分に影響を与えずに済むため、デバッグや新規アルゴリズムの導入が容易となり、柔軟性のあるシステムの構築が可能となりました。

これによって人工衛星のようなミッションクリティカルなタスクへ対応した処理フローの実現が可能となりました。

ROS ROS2

また、今回開発したROS2の開発環境は、他のROS2ベースのシステムとの互換性が高く、既存の機能やアルゴリズムを再利用することができ、開発プロセスの効率化やコスト削減が見込めます。

開発例

項目	詳細情報
OS	Ubuntu 22.04 LTS
開発期間	24ヶ月
開発人数	1人
開発言語	C++, Python
使用フレームワーク	ROS2
関連技術	カルマンフィルタ PD制御 MATLAB/Simulink

数値データ分析

～ Numerical data analysis ～

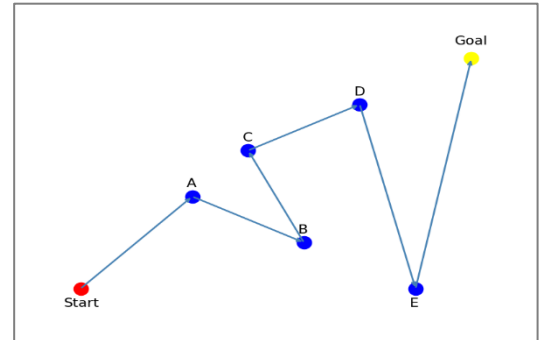
工場のIoT化やスマートフォンの登場などに伴い、様々な種類のビッグデータを取得することが可能となりました。しかし、取得したデータを有効活用するためには、データをクレンジング・解析し有用なデータする必要があります。当社では、機械学習・DeepLearningを用いてお客様のデータを有効化し、さらに高度な予測を行うための手助けを行っております。

経路最適化問題の産業用途への適用

経路最適化問題とは、点と点を繋ぐなどによって表される経路に対し、時間やコスト、距離などを最小化して、効率を最大化することを目的とした組合せ最適化の問題の一つです。与えられた問題で通過すべき点の数が多いと組み合わせ数が急激的に増加するため、最短経路の算出とその計算時間をトレードオフで考える必要があります。

実世界の経路最適化問題においては、距離・時間・コストなど、何を目的として最適化を行うのか、算出する経路に対する制限はあるか、適用する分野は何かなど、様々な制約条件の下で最適化を試みます。

本開発では、メーカー企業とともに産業用途への経路最適化問題の適用をテーマとし、ある装置が計測すべき規則性のない点群間を効率的に移動するための経路を得る問題について取り組みました。



既存の経路最適化アルゴリズムの応用

・ ヒューリスティックな手法の応用

本問題は、定められた始点から全ての点を一回ずつ通って定められた終点まで移動する、という制約から、**巡回セールスマン問題**に近い特徴がありました。そのため、我々は解決策としてヒューリスティックな手法である**反復局所探索法**を採用しました。局所探索法では最適解の算出は保証されませんが、局所解からの脱出と局所探索法の適用を繰り返すことで、十分に良質な解を得ることが可能です。

・ 算出時間の短縮のための工夫

与えられた点群の点数が少なければ前述の手法の適用のみで対応できますが、点数が膨大である場合、計算量は現実的ではなくなります。そのため、我々は大きさや形に応じて**点群を分割**し、分割したそれぞれの点群を最適化するというアプローチをとることで、最適化結果の品質を担保しつつ、現実的な計算量に留めることに成功しました。

また、経路が作成された点群に対して、**点群の一部が削除**されたり、**新しい点が追加**された時の経路の変更処理を実装するといった、顧客の要求に対応できる柔軟さも当社の強みの一つといえます。

開発例

項目	詳細情報
OS	Windows 10
開発期間	24ヶ月
開発規模	2人
開発言語	Python
関連技術	・ 局所最適化 ・ クラスタリング

半導体製造装置 データ解析モジュール開発

本開発の目的は、製造装置に紐づいた多くのデータに対し随時発生する課題を解決するためデータの解析、可視化を行い、最終的に実システム上で動作するブラックボックスではないモジュールを提供することです。
取り扱うデータは、主に個別の装置に紐づいた温度や製造物の形状、位置などの観測データ、制御パラメータなど多岐にわたります。

それらに対し、古典的手法から近年発表された論文の手法まで、課題にあった手法をご提案し、机上検討、実データ検証、試作運用と手順を踏んで実現していきます。

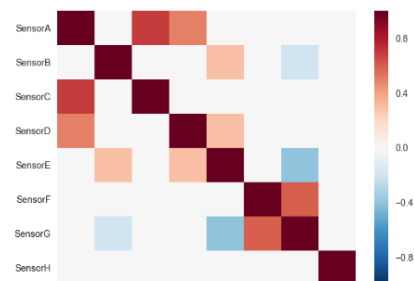
課題に対する理解を深め、現実的な解決を

データ解析における落とし穴として、解析技術者の暴走があります。
指定されたデータを解析し、目標性能を出すことに終始してしまい、別の方法で課題が解決できる可能性に目を向けなくなってしまう、というものです。

我々は「お客様が抱えている課題はどんなものか」「解決の方法は何パターン考えられるか」「より低コストで解決できる方法は何なのか」、「アルゴリズムの本質を理解し、計算過程を明瞭に提示する」お客様と同じ視点で解析に取り組んでいくよう努めています。

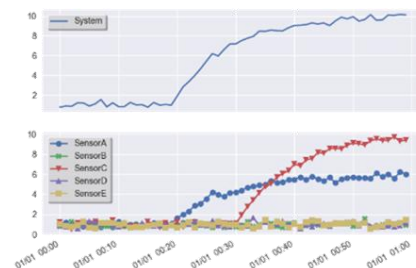
● 通常稼働時におけるセンサーデータの相関構造推定

通常稼働時のセンサーデータを学習させることで、各センサー間の相関関係を推定します。
これにより、通常センサーが取りうる値の範囲や、連動するセンサーを抽出することができます。
また、意図せず連動しているセンサーを特定することもでき、設計の改善などにも利用できます。



● システム・各センサーの異常検知

あらかじめ学習させた通常稼働時の相関構造を基に、センサーデータの異常度を計算します。
システム全体の異常度や、各センサーごとの異常度、特定のセンサーグループの異常度等、様々な粒度で解析・可視化を行うことができます。



● 異常発生時におけるセンサーデータの因果関係推定

異常発生時における各センサーデータの変動を解析することで、センサーデータ間の因果関係を推定することができます。異常が装置内で伝搬していく流れを可視化することができ、異常の原因となったセンサーの特定に活用できます。



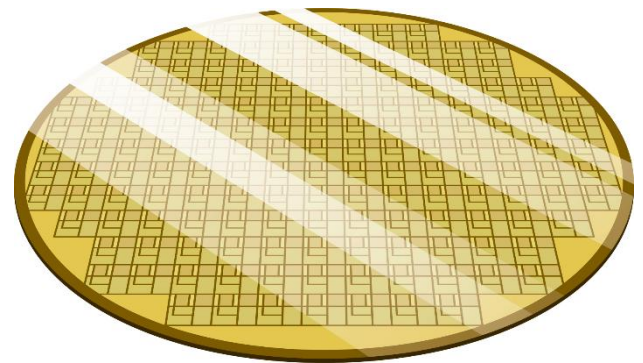
半導体製造装置 計測箇所最適化

半導体の製造工程には、ウェハと呼ばれるシリコンを円板状にカットしたものへ微細な電子回路パターンを露光する作業が含まれます。

これには非常に高い精度が要求され、ウェハのナノ単位の歪みが品質に影響を与えます。

そのためこの歪みを計測/補正することが必要となりますが、ウェハ全面を計測するのはコストが高く、時間当たりの生産性を示すスループットが低下します。

この課題を解決するためウェハに生じる歪みをモデリング/推定しながら、計測箇所が可能な限り少なくて済み、また歪みが正確に計測できるような計測箇所の組み合わせを最適化によって探索するという問題に取り組んでいます。



課題に寄り添ったソリューションのご提案

本開発で取り扱う課題は非常に専門性が高く、類似の事例もほとんどないため手法選定の段階から手探りのような状態でした。

そのような中でも、お客様へのヒアリングやディスカッションを通して要件を具体化し、課題の本質と解決手段を追求しました。

結果として、多目的最適化や組み合わせ最適化の代表手法から知名度の低い発展的手法まで様々な論文を調査し、それぞれについてアルゴリズム実装/評価を重ねることで適切な手法の選定に繋げることができました。

このような調査力と柔軟性も当社の強みの一つといえます。



開発例

項目	詳細情報
OS	Windows 10 Pro
開発期間	24ヶ月
開発規模	2人
開発言語	Python, C#
関連技術	・（制約付き）組み合わせ最適化 ・焼き鈍し法 ・モデル選択

時系列データにおける外乱の検知手法の検討

時系列データの異常検知は区間ごとでの正常と異常のパターンの分類問題となるのが一般的です。しかし、データを取得する環境によっては異常パターン以外の要素で外乱が発生することがあります。その場合、この外乱が発生しているかの検知を人力で行えば大きな手間となります。また、既に異常検知アルゴリズムがある場合には、外乱のせいでそのアルゴリズムが正常に動作しなくなることが起こります。

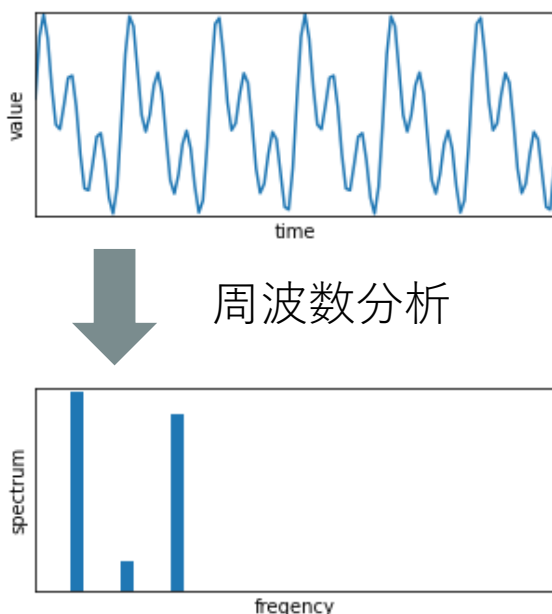
本開発は時系列データから外乱であるデータ区間を自動で検知し、今後の手間を軽減するAI手法の検討になります。前述の異常検知アルゴリズムを運用するような状況では、正常と異常データの区間を外乱と誤検知しないようにした上で外乱データの区間を検知することになります。

ここで難しくなるのは、異常のパターンの中に振幅の不規則性が認められるものが存在するときです。つまりデータの振幅の規則性がないことだけを理由に外乱であると検知することができないことにあります。

対象のドメイン知識の重要性

このようなときに助けになるのが、特定の専門分野や業界についての知識、つまりドメイン知識です。特にお客様から問題の対象のドメイン知識を頂くことで、問題設定が単純化及び整理できます。それにより開発の課題が解決に向かうことが多くあります。

本開発ではお客様から頂いたドメイン知識がきっかけで、外乱の方に正常パターンや異常パターンとは異なる法則性を発見できました。これにより単純な周波数分析と機械学習手法を組み合わせることで検知できるという方針を出すことができました。



開発例

項目	詳細情報
OS	Windows
開発期間	1カ月
開発規模	1人
開発言語	Python
関連技術	機械学習、異常検知、信号処理

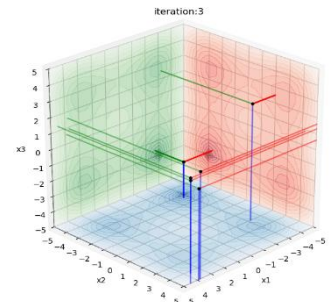
マテリアルズインフォマティクス

マテリアルズインフォマティクス（MI）とは、機械学習を材料分野に応用するという考え方です。

我々は、特にベイズ最適化をベースとした技術を用いて「狙った物性を持った新材料を少ない試作回数で合成する」ことを目標として本課題に取り組んでいます。

ベイズ最適化とはMIにおいて、「組成から物性を予測する」「狙った物性に近づける」という二つの指標から「次に試作すべき組成を提案する」といった動作を行い、一定の効果を示しますが、同時にいくつか課題もはらんでいます。

我々は、材料学を専門とするお客様のもと、ベイズ最適化の発展手法をいくつかの論文から実装し、検証、提供することでこれらの課題解決に寄与しています。



材料分野に限らず、製造実験全般における課題

- 狙いたい物性の間にトレードオフ関係がある

製品の示す特性といったものは多種多様で、特に材料の分野においては物性の種類は1億ともいわれています。

しかし、それらの物性は一方を狙った性質に近づけるように組成を変更すると、一方は離れていくといったトレードオフの関係が存在します。

こうなると、1つずつ物性を最適化していても、狙った物性には近づきません。

これを解決するために、ベイズ最適化を**多目的ベイズ最適化**に拡張しました。

本手法によって複数の物性を一つの指標に落とし込むことで、

「すべての物性が狙った物性にバランスよく近くなる組成の集合」を探索することができます。

- 新製品は作成に期間がかかる

製品を新規で作成する場合、受発注処理や原材料調達、スケジュール調整、運搬など時間のかかるプロセスを挟み、すぐに現物が得られるとは限りません。

材料の作成においても1回1回製造された結果を見てから次の組成を決めているようでは、待機時間が長くなってしまい、欲しい物性を得られるまで何年かかるかわかりません。

我々はこれを解決するために、**多点ベイズ最適化**を利用しました。

これは、すでに計測済みの結果から最適の可能性が高い複数の提案を得られるという手法で、一度の注文で複数の結果を得ることができます。

開発例

項目	詳細情報
OS	Windows
開発期間	12ヶ月
開発人数	1人
開発言語	python
関連技術	ベイズ最適化の拡張

これら二つを組み合わせ、

多点多目的ベイズ最適化も

期待された性能を示しています。

マテリアルズインフォマティクス

組成提案スクリプトのモジュール化および新機能開発支援

材料研究の現場では、研究のスピードや独自性が重視される中で、研究者が試行錯誤を重ねながらスクリプトを整備していくことが少なくありません。こうしたコードは、研究を前進させるうえで大きな役割を果たす一方で、長期的な運用や複数メンバーでの活用を見据えると、保守性や拡張性、知見の共有といった観点から整理が求められる場面もあります。

本事例では、顧客が研究の中で構築・改良してきた組成提案スクリプトを対象に、既存の知見や運用実態を踏まえながら、モジュール化、不具合修正を実施しました。その過程で明らかになった要件や新たな懸念事項についても継続的に対話を重ね、研究の進展に合わせて改善が続けられる、拡張性の高いツールへと発展させました。



研究者の知見を、組織で使い続けられる資産に

研究の中で育ててきたスクリプトを、継続的な活用や機能拡張に対応できる形へ整えるため、本案件では以下の3つの観点から支援を行いました。

・既存コードの理解と要件の可視化

数千行規模の既存 Jupyter Notebook を丁寧に読み解き、処理フローや構成を整理・文書化しました。その過程で、機能間の仕様差や暗黙の前提を洗い出し、顧客との対話を通じて要件を明確化しました。さらに、処理階層ごとのフロー図を含むスクリプト仕様書を作成し、改修方針の整理と今後の拡張に向けた土台づくりにつなげました。

・定量評価にもとづく性能改善

スクリプト全体の定量評価を進める中で、処理速度の改善余地に着目し、ボトルネックの調査を行いました。改善の具体例としては、逐次処理となっていた部分を処理効率の高いベクトル演算へ置き換え、対象処理の実行時間を元の1/10に短縮しました。これにより検証サイクルを回しやすくなり、より多様な条件での試行が可能になりました。

・対話を通じた仕様整理と新機能開発

新たな組成生成モードを追加するにあたり、既存機能との整合や利用時の分かりやすさを踏まえながら、顧客と継続的に仕様を整理しました。組成データ特有の制約や条件間の相互関係を考慮して設計・実装を進めることで、研究現場で扱いやすく、今後の拡張にも対応しやすいスクリプトへと改善しました。

開発例

項目	詳細情報
OS	Windows
開発期間	12ヶ月
開発規模	1人
開発言語	Python
関連技術	組成データ (Compositional Data, CoDa)

Memo

掲載内容の一部およびすべてを複製、転載または配布（電子媒体における転送含む）、印刷など、無断での使用を禁止します。

本カタログに記載の会社名及び商品名は各社の商標または登録商標となります。

本カタログの情報は2026年4月現在のものです。仕様と製品は製造・販売元が何ら責任を負うことなく予告なしに変更される場合があります。



Computermind

株式会社コンピュータマインド

東京本社

〒160-0023

東京都新宿区西新宿6丁目6-2 新宿国際ビルディング4F

☎ 03 - 6911 - 1855 (代)

🌐 <http://www.compmind.co.jp/>

本社

〒400-0064

山梨県甲府市下飯田1-10-8

☎ 055 - 230 - 1122 (代)

