2025/3/7

CNNによる画像分析タスクの実行時間予測のためのCPU/GPUベンチマーク手法の提案

大阪大学

源内 裕貴　川口 峻平　大下裕一　下西英之

Shimonishi Lab.
Osaka Univ.

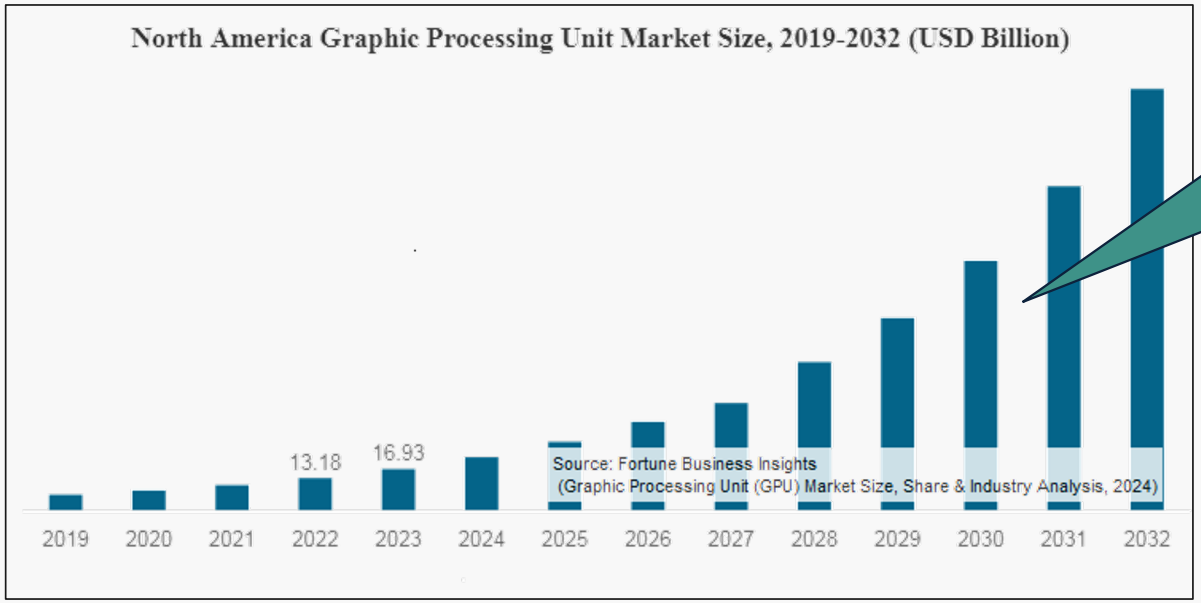
背景

GPU需要の増加 → クラウド利用の拡大

- オンプレミスでの確保が難しいため，クラウドGPUの利用が広がっている
- 例：AWS, Microsoft Azure, Google Cloud Platform

GPUクラウドサービスの課題

- 高額な初期費用が発生し、高い価格設定
- データセンター拡大による電力消費の増大



北アメリカにおけるGPU市場の規模
2025 → 2030で約3倍

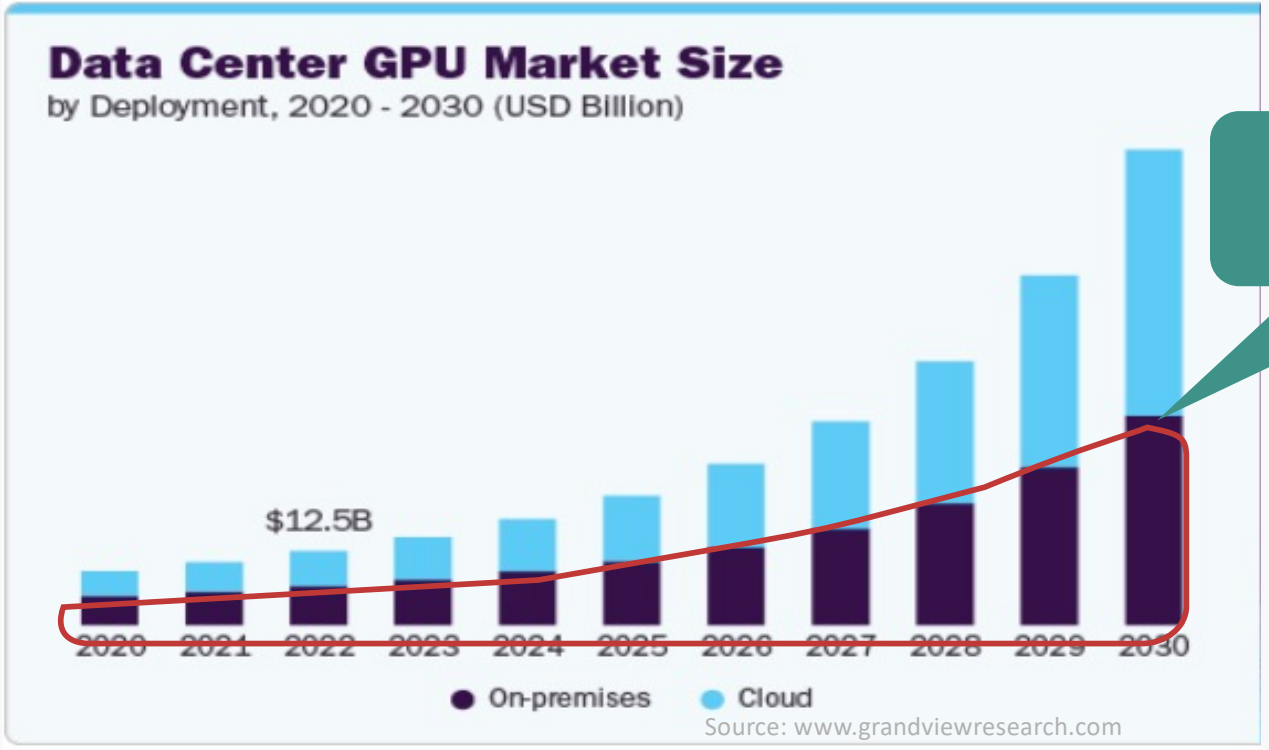
オンプレミスGPUの活用

未使用のオンプレミスGPUの共有（GPUシェアリング）

- クラウド利用されていないGPUサーバは2025年時点で**52%**ある

→ ・**既存インフラ**を使用することで，設備投資の費用を抑える

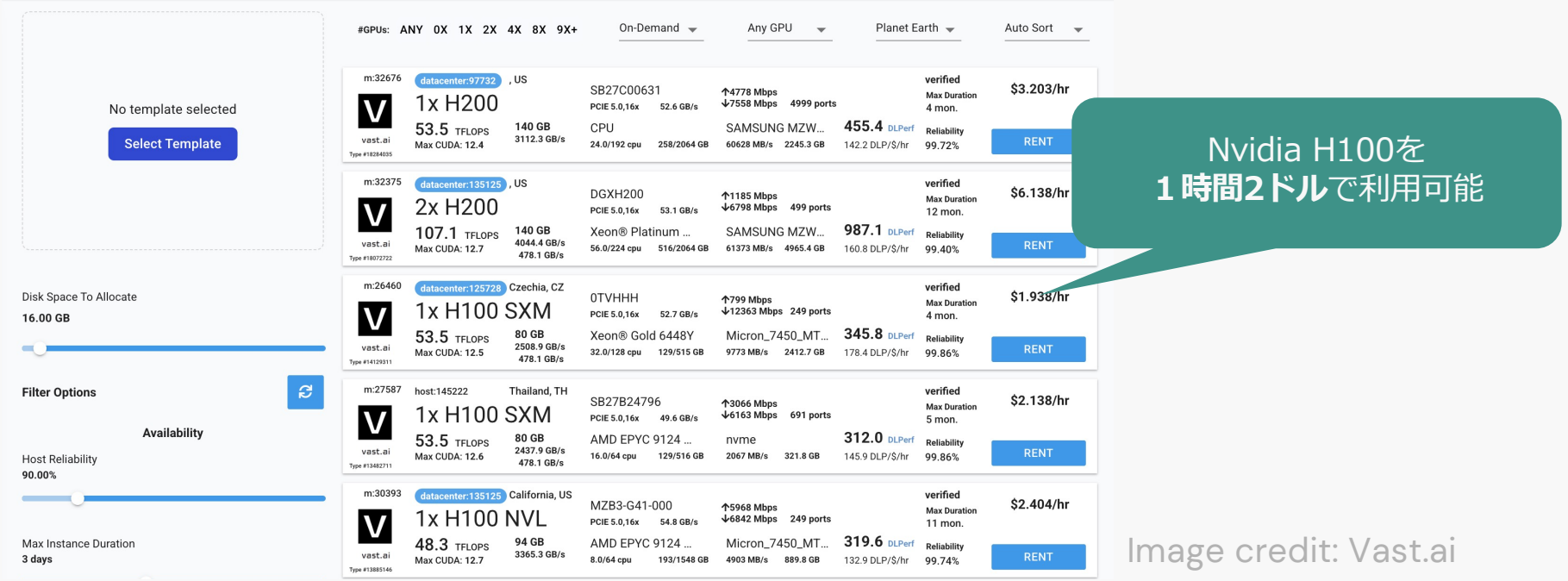
・未使用のGPUリソースのシェアでGPU市場全体として稼働率を向上



GPUシェアリングサービスの例

Vast.ai[1]

- 分散型GPUシェアリングプラットフォーム（**データセンタから個人レベル**）
- 余剰GPUリソースを活用し、低コストで計算資源を提供
- 課題：サービスにサーバを提供している間，オーナーは**個人利用できない**



Nvidia H100を
1時間2ドルで利用可能

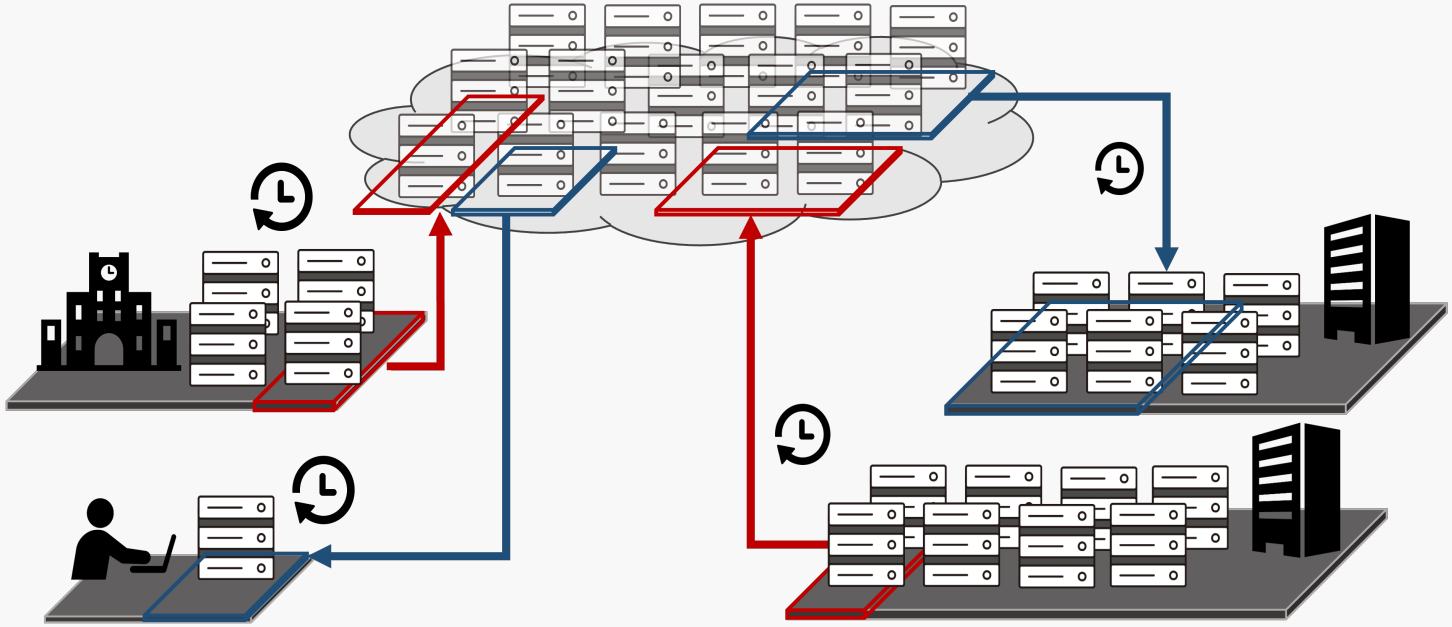
Image credit: Vast.ai

[1] : Vast.ai, Global GPU Market, https://vast.ai/, 2025/2/27

Air Computing Pool（ACP）：コンセプト

細粒度な時間貸し出し可能なGPUシェアリングサービス

- **オーナーはサービスに提供している間も個人利用可能**
- **未使用時間のみGPUサーバを貸し出し**（貸し出し時間はオーナーが決定）
- タスクの完了時間の予測をもとにGPUサーバを予約して利用
- ユーザは**予約時間が終了したら即時**にサーバの使用権を返却



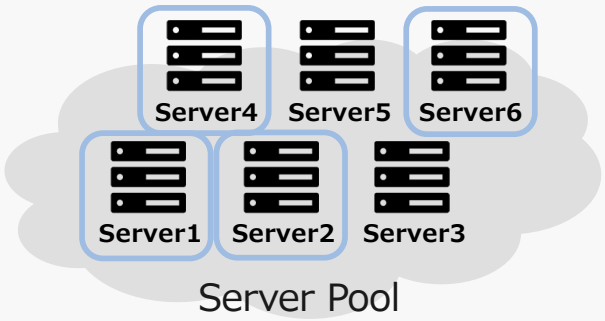
Air Computing Pool:利用手順

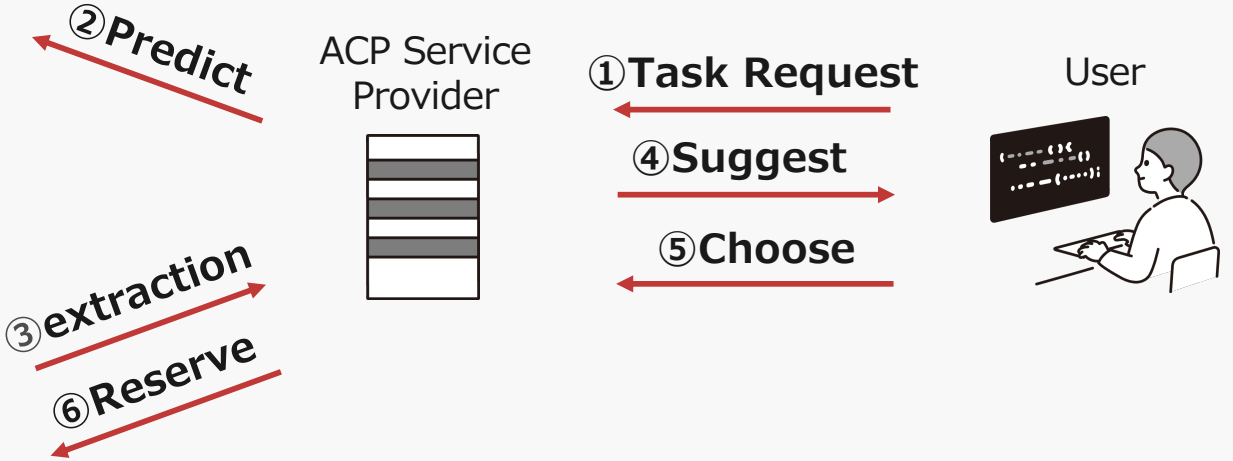
タスクの実行時間単位で貸し出し可能なGPUをシェアリング

- タスク情報を基にPool内のGPUにおける実行時間を予測
- **GPUの貸し出し可能時間**と**タスク完了予測時間**から候補をユーザに提示

Server Info	Available Time	Prediction Time
Server1	50 min	20 min
Server2	40 min	15 min
Server3	5 min	10 min
Server4	30 min	50 min
Server5	45 min	20 min
Server6	60 min	40 min

Available Time > Prediction Time





Air Computing Pool における課題

GPUサーバの種類/性能が不均一

- 各地の企業，研究機関からサーバを提供するため，CPUやGPU,メモリの構成が非常に多様
- サーバの内部構成によって処理性能が大きく変動
- GPUアーキテクチャが異なるとハードウェア特性や動作特性が変わり，それに伴いタスクによって計算性能が大きく変わる

予測モデルの更新はサービスの運用効率の低下を招く

- 新たなGPUサーバに対して予測精度向上を図り，都度モデルを更新していると，運用/保守コストが非常に重くなる

未知のGPUに対しても予測モデルを再構築することなく
正確な予測ができる手法が求められる

技術課題

機械学習モデルによる実行時間予測

- GPUハードウェアスペック（コア数，クロック数）を入力指標とする

→ 未学習のGPUアーキテクチャに対して予測精度が悪い

例：N体問題による
ベンチマーク[2]

GPUベンチマークを用いた実行時間予測

- 特定のタスクに対して単一の指標でGPUサーバの性能を抽象化

→ 計算処理の種類が異なるタスクに適用できない

GPUサーバの処理要素ごとに性能を測ることでタスクに依存しない
GPU性能指標の構築を目指す

- AIタスク全般に対応することを目標とし，本研究では**CNNによる画像分析**を扱う

[2] : D.P. Playne, M. Johnson, and K.A. Hawick.(2009). “Benchmarking GPU Devices with N-Body Simulations”.CDES.vol.9 p.132.

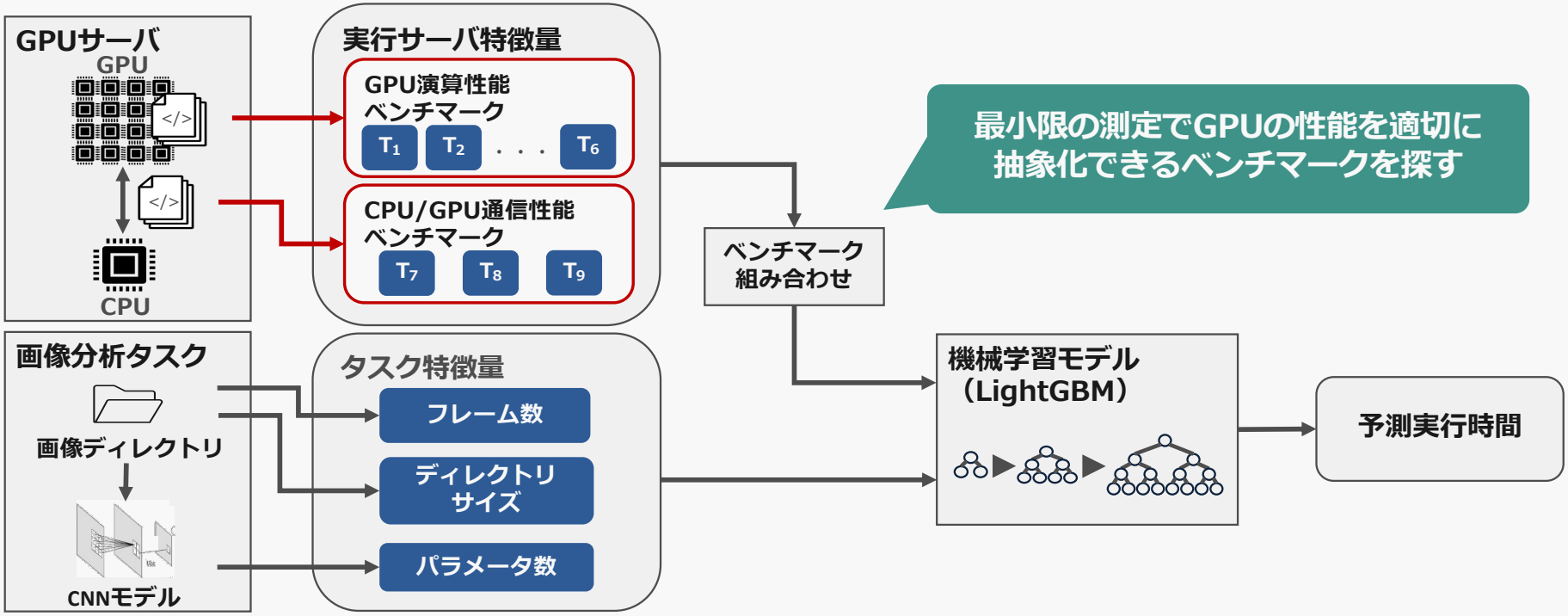
提案手法

GPU性能を2つの処理要素に分けてベンチマークを取得

- GPU演算とCPU/GPU通信の性能を複数の指標で計測し組み合わせる

ベンチマークを入力とする機械学習モデルを構築

- 分析アルゴリズムとして勾配ブースティング決定木を使用（LightGBM）



GPU演算性能を測るベンチマーク候補

演算の種類

- 行列の**和，積，畳み込み演算**の3種類
- 各演算において，1回の**単一演算**と100回の**繰り返し演算**の2パターン

設計の狙い

- SCO / SMO / SAO：大規模データでのGPUの**ピーク性能**評価
- MCO / MMO / MAO：反復処理における**スループット**評価

	ベンチマーク	処理内容
畳み込み	T_{SCO}	$(10^4 \times 10^4)$ 行列に対して（20×20）フィルタで畳み込みを1回実行
	T_{MCO}	$(10^3 \times 10^3)$ 行列に対して（20×20）フィルタで畳み込みを100回実行
行列積	T_{SMO}	$(10^4 \times 10^4)$ 行列の間で行列積を1回実行
	T_{MMO}	$(10^3 \times 10^3)$ 行列の間で行列積を100回実行
行列和	T_{SAO}	$(10^4 \times 10^4)$ 行列の間で行列和を1回実行
	T_{MAO}	$(10^3 \times 10^3)$ 行列の間で行列積を100回実行

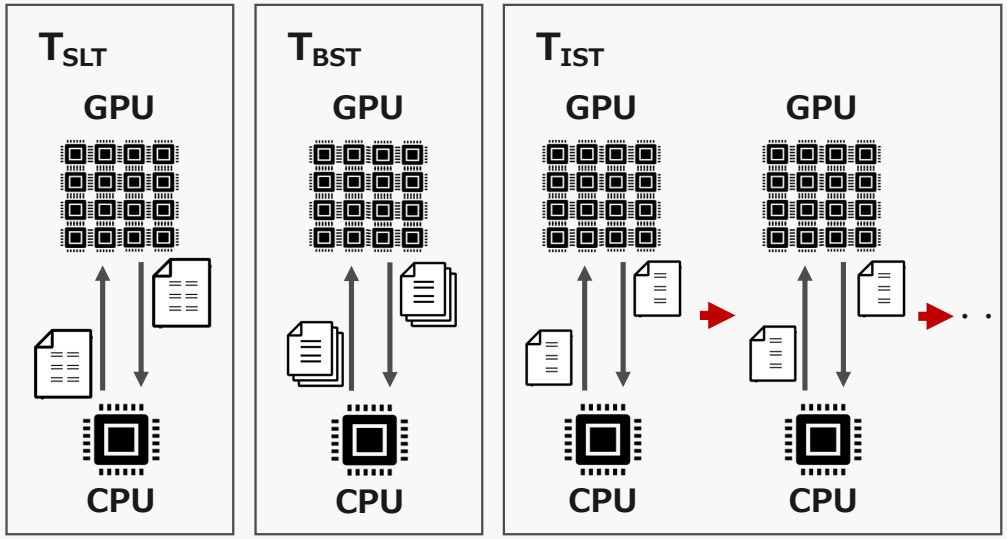
CPU/GPU通信性能を測るベンチマーク候補

転送の種類

- **単一**データと**複数**データの2種類

設計の狙い

- SLT：**メモリ帯域幅**の最大性能評価
- BST：**メモリ負荷**が通信性能に与える影響を評価
- IST：**転送オーバーヘッド**が通信性能におけるの影響を評価



ベンチマーク	処理内容
T_{SLT}	1個の（$10^4 \times 10^4$）行列を転送
T_{BST}	100個の（$10^3 \times 10^3$）行列を同時に転送
T_{IST}	100個の（$10^3 \times 10^3$）行列を1個ずつ逐次転送

評価環境

GPUサーバ

- **12種類**のCPU-GPU構成のサーバ
  （OS：Ubuntu 22.04）

CPU	GPU
Intel 13th Core i5	GTX 1080, GTX 1650 RTX 3050, RTX 3060 Ti RTX 4070
Intel 13th Core i7	GTX 1080, RTX 3050 RTX 3060 Ti, RTX 4070
Intel 9th Core i7	GTX 1080, RTX 4070
Intel 1th Xeon Gold	RTX 2080 Ti

CNN系画像処理モデル

- **YOLOv8, SSD, Faster R-CNN**の3種類

→ YOLOv8, SSD ：one-stage
　Faster R-CNN：two-stage

物体領域の抽出とクラス分類を
同時に処理 vs. 別々に処理

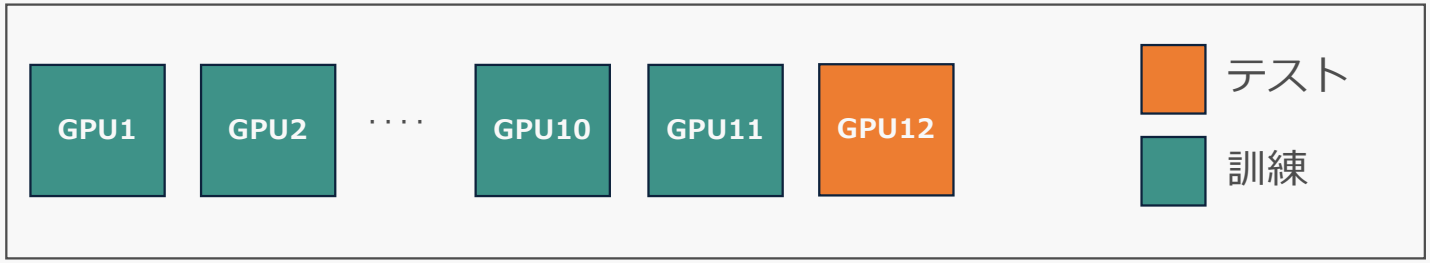
画像データセット

- Pexelsから取得した**100**種類の動画を推論用画像データとして使用

評価について

Leave-One-Out Cross-Validation法による評価

- 12種類のGPUサーバー → 1つをテスト用に除外、残りでMLモデルを学習
- 12回繰り返し、全てのサーバーに対する汎化性能を評価
- 評価指標：**MAPE（Mean Absolute Percentage Error）**
- **average MAPE**：12種類のテストに対するMAPEの平均

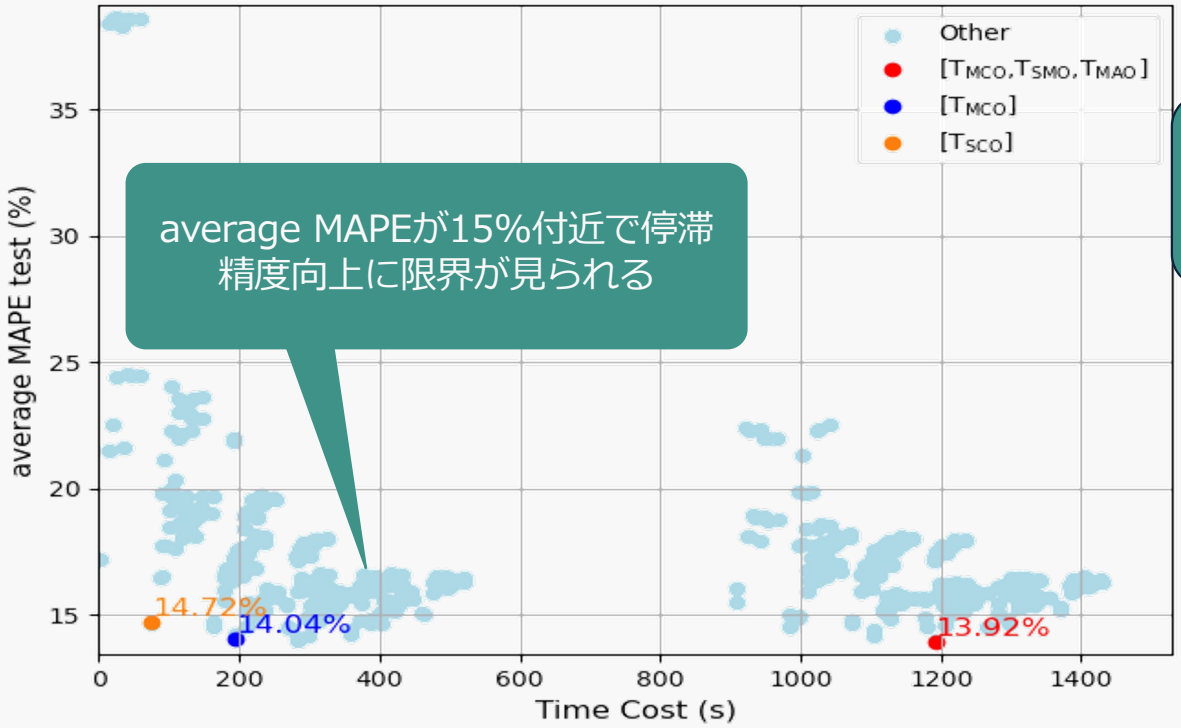


タスク実行時間予測の精度向上に寄与するベンチマークの評価

- 設計した9 種類のベンチマークのうち，少なくとも 1 種類を使用する
  計511 通りの組み合わせに対して機械学習モデルを作成
- **使用するベンチマークの組み合わせとモデルの精度の関係**を評価する

ベンチマーク測定時間と精度の関係

- **Time Cost：各ベンチマークの測定時間の総和**
- Time Costが大きいほど、GPUの性能差を捉えやすい
- 指標を増やしても**精度向上には限界**があり、過剰な指標追加は特定の学習サーバに対して**過剰適合**する



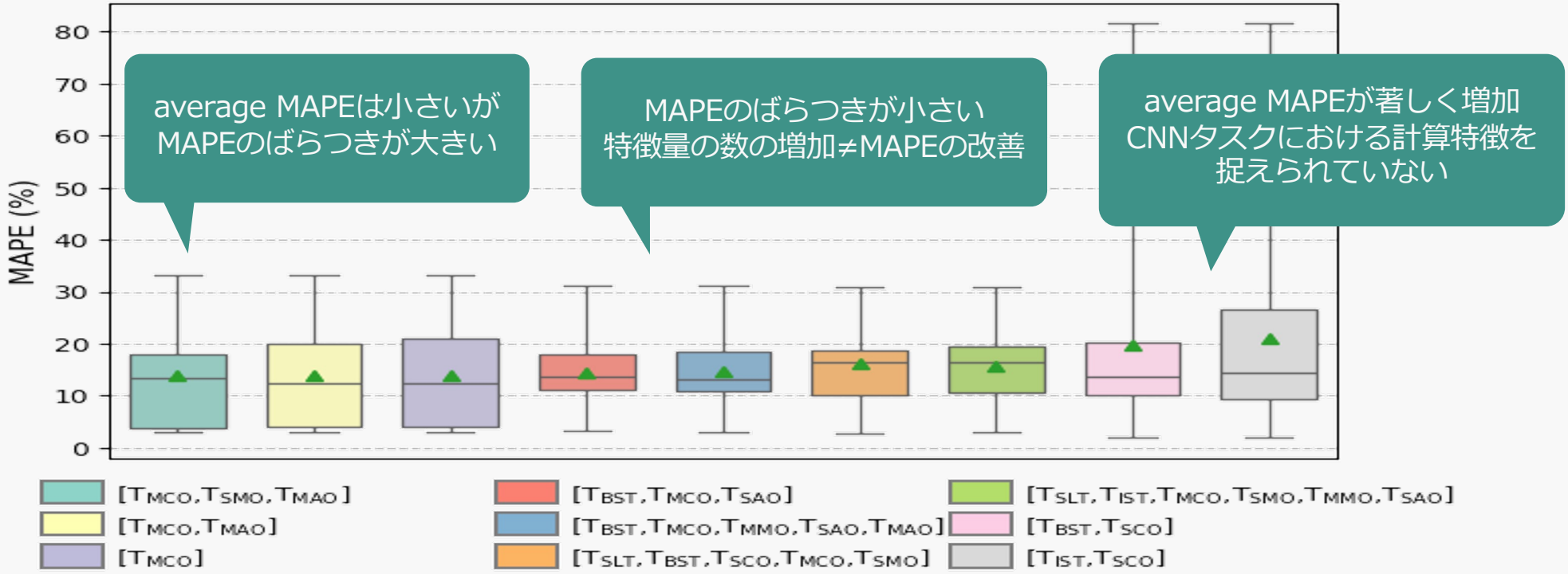
Time Costを大幅に増やしても精度は同程度
実運用では青や黄色のモデルが適切

- ●：average MAPE 13.92%
  Time Cost **1192**秒
- ●：average MAPE 14.04%
  Time Cost **195**秒
- ●：average MAPE 14.72%
  Time Cost 74.7秒

高精度モデルにおけるベンチマークの組み合わせ

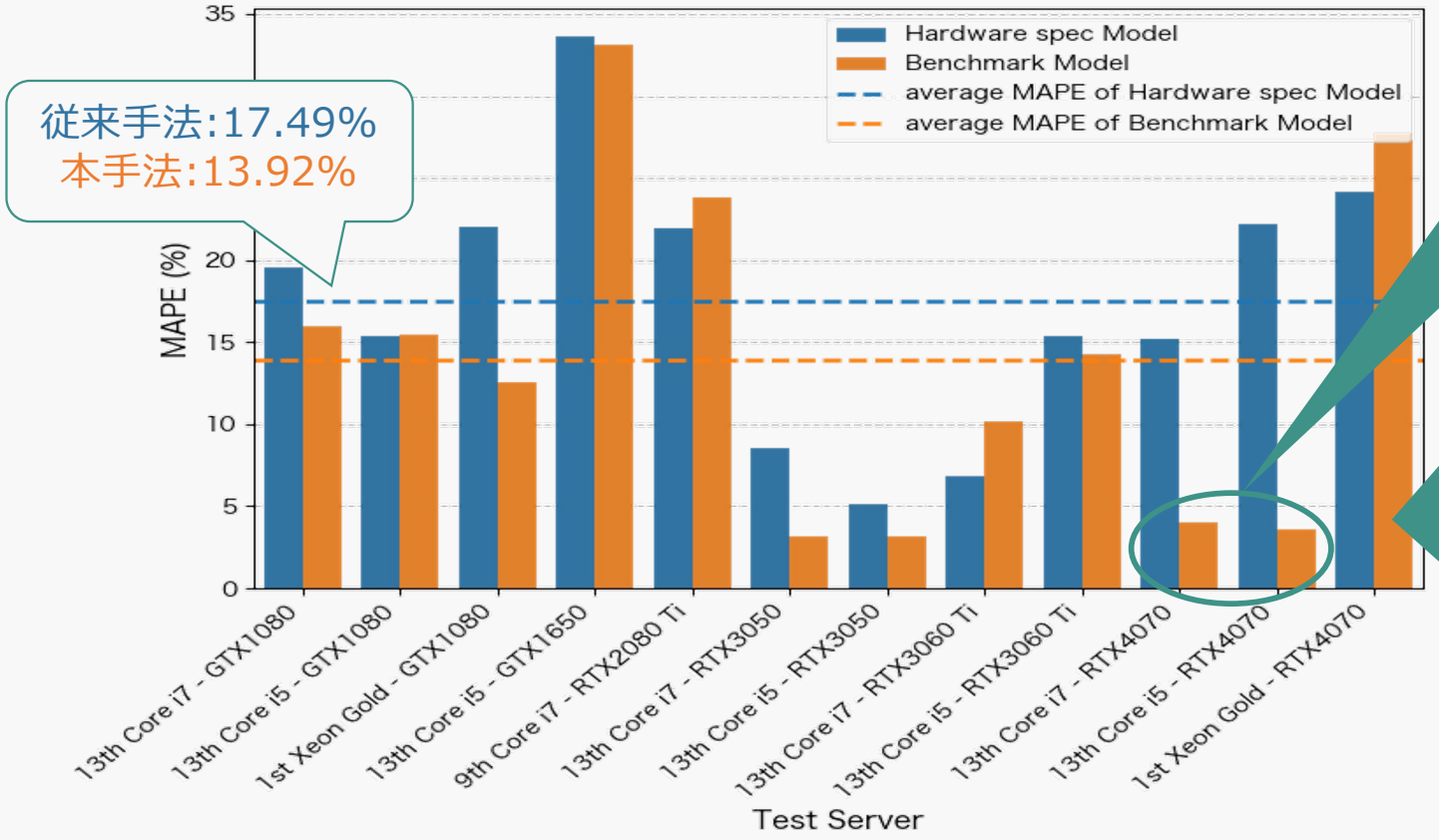
T_{MCO}（複数回畳み込み）を含む組み合わせが最良の予測精度を示す

- T_{MAO}（複数回行列和）との組み合わせて**average MAPE**が向上
- T_{BST}（複数個まとめて転送）との組み合わせて**MAPEのばらつき**が改善
  → 学習に用いるGPUサーバの種類に左右されにくい



従来手法との比較

ハードウェアスペック（コア数,クロック数 .etc）を入力とするモデルより，提案手法は未知のGPUに対する**予測精度が向上**



RTX4000番台キャッシュの進化でコア数がそれほど増加していないが大幅に並列計算速度向上

実動作に基づく指標のため予測精度向上

CPU性能がボトルネックでGPU本来の性能が制限

ベンチマークで測定した性能よりも実測の実行時間が大幅に長かった

テストベンチの計算負荷が軽すぎることが要因

まとめと今後の課題

まとめ

GPUサーバ処理要素ごとに計測するベンチマーク手法の提案

- 特定のタスクに依存しないベンチマークの手法の提案

機械学習モデルと組み合わせた実行時間予測モデルの構築

- 演算性能と通信性能を組み合わせることで予測のロバスト性が向上
- 従来手法よりも未知のGPUに対して実行時間予測精度向上

今後の課題

- ハイエンドGPUへの適用
- Transformerによる画像分析や，自然言語処理，AI学習の実行時間予測への適用