5E-05

メディア解析クラウドサービスにおける データ内容適応型の負荷制御手法の提案

Proposal of Content-Adaptive Load Control Method for Media Analysis Cloud Service

岩松洋介† 有熊威† 北野貴稔†

YOSUKE IWAMATSU† TAKESHI ARIKUMA† TAKATOSHI KITANO†

1. はじめに

顔認識や話者認識のように，画像や音声から実世界の状況を認識するメディア解析の技術が普及期を迎えており，これをクラウドサービスとして提供する動きが広まりを見せている．

このようなメディア解析においては，画像内の対象の数等のデータ内容に応じて，処理の負荷が大きく変動する．クラウドで複数のデータを同時に処理する場合，負荷の大きい一部の要求によってリソースが専有され，サービス品質が低下することが課題となる．

そこで，本論文では，データ内容に基づいて負荷を推定し，負荷の大きい要求の処理中でもその他の要求の処理が進むようにリソースの割当を制御することで，サービス品質を向上させる負荷制御手法を提案する．顔認識を用いた実験評価により，提案手法の有効性を示す．

2. メディア解析のクラウド提供における課題

メディア解析ではデータ内容に応じて処理の負荷が大きく変動し，そのクラウド提供においては，負荷の大きい一部の要求によるリソースの専有が課題となる．

メディア解析において負荷が変動する理由は，個々のデータに含まれる対象の数や量が異なり，これらの数や量に負荷が連動するためである．例えば，顔認識においては，画像から顔を検出し，検出した顔毎に特徴の抽出(特抽)を行うが，画像に含まれる顔数が多いほど特抽の実行回数が増え，負荷が大きくなる．一つの顔の特抽には一般に数百ミリ秒を要するため，顔数が少ない画像は数百ミリ秒で処理できる一方，顔数が多い画像には数秒を要する等，秒単位で負荷のばらつきが生じる．

クラウドサービスにおいて，このように負荷の異なる複数の要求を同時に処理する場合，一部の負荷の大きい要求によるリソースの専有が課題となる．例えば，CPU コアが 4 つのサーバで顔認識のサービスを実行する場合，顔数が多い画像が同時に 4 つ到着すると，数秒間，CPU コアを全て消費してしまい，その他の要求の処理が進まなくなる．結果，所定の時間内に処理を完了できる割合(以下，時間内処理率と呼ぶ)が減少し，リトライが多数発生する．即ち，サービス品質が低下するのである．

† 日本電気株式会社，NEC Corporation

サーバの負荷バランシング等，クラウド上のリソースの割当を制御する手法はこれまでも提案されてきたが[1][2]，これらの既存手法では，前述のような，メディア解析のサービスで発生する時間内処理率の減少を回避できない．なぜなら，既存手法においては，画像内の対象の数等のデータ内容を見ることなく要求をスケジューリングするため，負荷の大きい要求によるリソースの専有を防止できないからである．

したがって，メディア解析のクラウド提供においては，データ内容に基づいて負荷を推定し，負荷の大きい要求による専有が起こらないようにリソースの割当を制御する，データ内容適応型の負荷制御手法が求められるのである．

なお，このようなデータ内容適応型の負荷制御においては，負荷の推定自体にかかる処理時間の増加が問題となる．メディア解析では，画像内の被写体等，データ内容を解析して初めて負荷を推定できる．即ち，負荷を推定する処理自体に時間がかかる．そのため，全体の処理時間が増加しないよう抑制する仕組みが，併せて必要となる．

3. データ内容適応型の負荷制御手法

本論文では，データ内容に基づいて負荷を推定し，推定した負荷を元にした優先度付けによりリソースの専有を軽減することで，時間内処理率を向上させる負荷制御手法を提案する．負荷の推定においては，必要な条件下でのみ推定を実施する制御を導入し，処理時間の増加を抑制する．

図 1 に，提案手法を用いたサービス構成を示す．

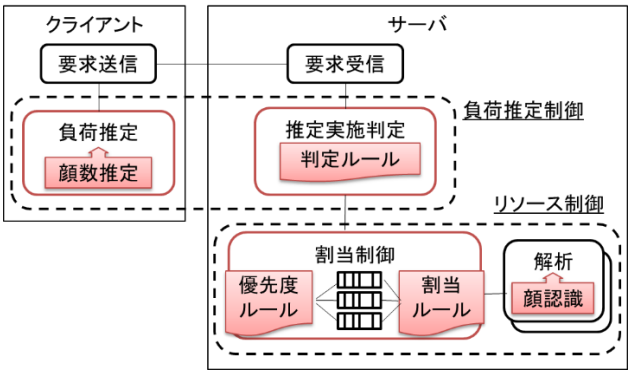

図 1 データ内容適応型の負荷制御のサービス構成

以下，負荷の推定の処理と，推定した負荷を用いたリソース割当・解析実行の処理について，詳細に説明する．

3.1 負荷の推定

初めに,クライアントからサーバに解析を要求する際の,負荷の推定の処理の流れを説明する.

まず,クライアントは,サーバに解析の要求を送信する.

サーバは，要求を受信し，判定ルールを元に，負荷推定を実施すべきかどうかを判定する．ここで，判定ルールとは，負荷の推定を実施すべき条件を指定するものである．例えば，判定ルールとして，サーバのリソース使用率が一定の閾値を超過した場合のみ負荷の推定を実施するという条件を指定する.負荷の推定の必要がない場合,サーバは,そのまま解析を実行する．負荷の推定が必要な場合，サーバは,クライアントに負荷の推定を実施するよう通知する.

クライアントは，通知を受け，負荷の推定を実施する.例えば顔認識の場合，画像内の顔数を推定し，負荷が顔数に対して線形に増加する等のモデルを用いて，負荷を推定する.クライアントは,推定した負荷をサーバに通知する.

以上のように，本手法では，サーバのリソース使用率が高い等，リソースの割当の制御が必要な場合に限定して負荷の推定を実施するため，負荷の推定自体による処理時間の増加を抑制することができる.

3.2 リソース割当と解析実行

続いて，推定した負荷を用いてリソースの割当を制御し,解析を実行する処理の流れを説明する.

サーバは，優先度ルールに従い要求の優先度を決定し,優先度毎のキューに要求を入れる．ここで，優先度ルールとは，推定した負荷から要求の優先度を決定するための条件を指定するものである．例えば，優先度を高と低の 2 段階とし，処理が 1 秒未満と推定される負荷のデータの要求は優先度高，1 秒以上と推定される負荷の大きい要求は優先度低と指定する.

次に，サーバでは，キューに入った要求について，割当ルールに従ってリソースを割り当て，解析を実行する．ここで，割当ルールとは，優先度に応じたリソースの利用の条件を指定するものである．例えば，優先度高の要求はリソースを 100%利用し，優先度低の要求はリソースのうち 50%のみ利用する，といった条件を指定する.

以上のように，本手法では，推定した負荷をもとにした優先度付けとリソース割当により，負荷の大きい要求によるリソースの専有を軽減し，時間内処理率の向上とする.

4. 評価

提案手法の有効性を確認するために，顔認識を対象とした実験評価を行った.

評価用データとして，2 種類の映像セットを用いた．一つは，リソースの専有が発生しやすい状況をシミュレートするもので，エスカレータ出口のように顔数が多いシーンと，人通りの少ない街路のように顔数が少ないシーンの両方を含む，計 12 カメラの映像セットである．もう一つは,比較用として，リソースに余裕がある状況をシミュレートするもので,上記 12 カメラから一部を除いた計 8 カメラの映像セットである．これらを 1 秒に 1 フレームずつ同時に入力した場合について，効果の計測や見積りを行った.

4.1 時間内処理率の向上の効果

図 2 に，各映像を入力し，所定の時間内に処理が完了した割合(時間内処理率)を計測した結果を示す．リソース制御の優先度ルール，割当ルールには，3.2 で例示した各条件を用いた．提案したリソース制御を用いた場合，リソースの専有が発生しやすい 12 カメラの入力のケースにおいて，時間内処理率が約 1.5 倍に増加した．即ち，提案手法にてサービス品質の向上が可能であることが確認できた.

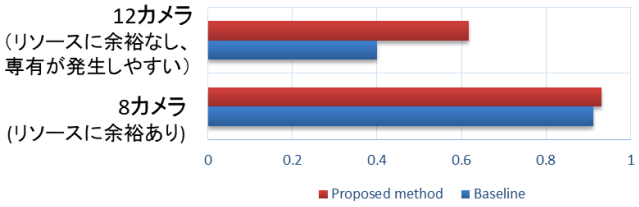

図 2 時間内処理率の比較

4.2 負荷の推定による処理時間増加の抑制の見積り

表 1 に，事前に解析した各映像内の顔数から，各映像が同時に入力された場合のリソース利用率を算出し，その時の処理時間の増加を見積もった結果を示す．負荷推定制御の判定ルールには，3.1 で例示した条件を用いた．提案した負荷推定制御を用いた場合，平均の処理時間の増加を 1.5%～4.6%と軽微に抑えられることを確認できた.

表 1 処理時間(平均)の見積りの比較

	負荷の推定なし	必要時に負荷を推定 (提案手法)
12 カメラ(リソースに余裕なし)	1053ms	1102ms (4.6%増)
8 カメラ(リソースに余裕あり)	785ms	798ms (1.5%増)

5. まとめ

メディア解析のクラウド提供に向けて，データ内容に基づいて負荷を推定し，負荷の大きい要求の処理中でも他の要求の処理が進むようにリソースの割当を制御することで,サービス品質を向上させる負荷制御手法を提案した．顔認識を用いた実験評価で，所定の時間内に処理が完了する割合が約 1.5 倍に増加する等,提案手法の有効性を確認した.

今後は，顔認識以外の多種のメディア解析に対して提案手法の適用可能性を検証し,実サービスへの活用を進める.

参考文献

[1] Nguyen Khac Chien et al, “Load balancing algorithm based on estimating finish time of services in cloud computing”, IEEE ICACT, 2016.

[2] Mayanka Katyal et al, “A Comparative Study of Load Balancing Algorithms in Cloud Computing Environment”, IJDCC,vol 1, issue 2, 2013.