

タプルストリームとメディアストリームの 統合的管理に関する研究

西村 和也[†] 川島 英之[‡] 北川 博之[‡]

筑波大学情報学群情報科学類[†] 筑波大学システム情報系情報工学域[‡]

1. はじめに

近年、高速ネットワークの発達、カメラの低価格化などの要因により、メディアストリームの配信、取得、蓄積を行うことが容易になってきた。そのため、ロンドンの監視カメラネットワークである Ring of Steel に代表されるような様々なシステムが実際に用いられている。

メディアストリームの普及に伴い、その処理要求も複雑化している。例えば、特定のカメラに人物が映っている時の映像データのみが欲しいといった要求がある。このように、利用者が求めている情報は、莫大な量のデータの一部分である場合が多い。そのためメディアストリームの配信、取得、蓄積といった処理はメディアデータの特徴量の解析などによって選択的に行われることが望ましい。さらに、各種センシングデバイスのデータストリームとの異種統合利用により、更に高度なデータ処理が可能になる。

これまで、ストリームデータを処理するための枠組みとしてストリーム処理システム (Stream Processing System) [1, 3] というものが研究、開発されてきた。しかし、既存のストリーム処理システムはタプル型のストリームデータの処理を前提としているために、画像、動画のようなメディアストリームの処理を行うことが難しい。

そこで、本研究ではメディアストリームとタプルストリームを連携動作させるシステムを提案する。なお、メディアストリームとしては Kinect を取り上げる。

2. 背景

2.1. タプルストリーム処理システム

ストリームデータ利用機会の増加に伴い、ストリーム処理システムというミドルウェアの研

究開発が盛んに行われてきた。タプルストリーム処理システム (タプル SPS) は関係演算処理のみを提供するため、ストリームストリームデータは一般的なデータベースと同じくタプル形式で表現される。多種多様なストリームデータをタプル形式で表現する事でそれらを統合的に扱う。

2.2. Kinect

Kinect は Microsoft 社が販売するゲーム用途のセンサデバイスである。公開ライブラリにより PC から利用可能である。本研究においては多様な OS で動作が可能な OpenNI を Kinect と連携して使用する。

3. 状況監視

状況監視の際、メディアストリームとタプルストリームは併用される。タプルはイベント検知、メディアは情報確認に使われる。例えば侵入者検知の場合、時刻 t で「人が侵入」というイベントが生じた時のみ、人間は時刻 t からのカメラデータで現象を確認する。

上記の仕組みを実現するには、タプルでのイベントを検知後にメディアを取得するスキームが必要になる。1つのスキームはタプル SPS を拡張してメディアを利用可能にすることだが、タプル SPS を拡張することは労力を要する。より現実的なスキームはメディアデータを管理するシステムを作成し、それをタプル SPS と連携させる方式である。本論文ではこのスキームを採用する。

4. 研究課題

上記のアプローチを実現するには 2 つの課題に挑む必要がある。第一の課題は問合せモデルの構築である。イベント検知時にメディアストリームを要求できる問合せ記述が必要になる。第二の課題はメディアデータを管理するサーバの実現である。過去のデータが閲覧される可能性もあるため、メディアサーバはデータを一定

A unified Management of Tuple streams and Media streams
Kazuya Nishimura[†], Hideyuki Kawashima[‡] and Hiroyuki Kitagawa[†]

[†]College of Information Science, University of Tsukuba

[‡]Faculty of Engineering, Information and Systems, University of Tsukuba

時間保持する必要がある。また、要求に応じてデータを配信する仕組みも必要である。

5. 提案

上記課題に対するアプローチを提案する。提案システムの構成を図 1 に示す。提案システムはダブル SPS, メディアデータサーバ, そしてデータ取得インタフェースから構成される。ダブル SPS については我々が開発してきた分散型ダブル SPS である SS*を利用した。メディアデータサーバはイベント検知に関連するデータ (例: 人数情報) をダブルとして SS*へ送信する。SS*はイベントを検知するとユーザプログラム中のデータ取得 I/F にダブルを送信する。データ取得 I/F はダブル受信後, 予め定められていたメディアデータをメディアデータサーバから取得する。本システムの開発言語は C++である。

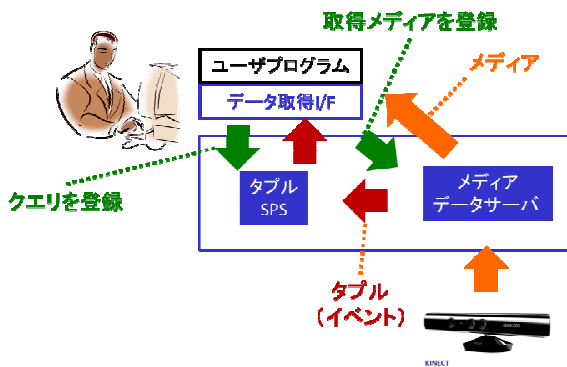


図 1 タプル・メディア連携機構

5.1. データ取得インタフェース

このモジュールは，データ取得時に動作させるメソッドを登録しておく．同メソッドにはデータ到着時にメディアデータを取得するロジックを記述する．このとき指定する項目は次の通りである：＜メディア名，開始時間，終了時間＞．イベント駆動のデータ取得 I/F は既に StreamSpinner[3] で提案されているが，本研究ではそれに時間属性を追加している．

5.2. メディアデータサーバ

メディアデータサーバは 2 つの機能をもつ。第一の機能はデータ管理である。Kinect を含むメディア系センサは膨大量のデータを生成する。本研究では OpenNI を対象とする。データ生成周期は約 30 フレーム/秒である。そして本研究では深度（解像度は 640×480 ）、画像データ（解像度は 640×480 、種類は RGB24, YUV422, 8bit グレースケール, 16bit グレースケール）、ユーザデータ（人数、骨格、重心、参入、消失）、ジェ

スチャ, 手の位置, を扱う.

メディアデータサーバは上述のデータを一定量メモリに保持しておく．メモリが不足した場合，一部のデータをストレージに移動する．本研究では移動アルゴリズムとして FIFO を採用した．この理由はイベント検知時に要求されるデータは新しいデータである事が多いという経験則に基づいている．そしてストレージのデータ量が一定量を超えると，FIFO に則りデータを削除する．

第二の機能はデータ配信である。メディアデータサーバはデータ取得 I/F からのメディア要求を受信すると、該当データを送信する。このとき、複数のクライアントから同一データへの要求がある場合には内部で要求時間情報を考慮しながら、データコピー回数を削減することで性能を向上させる。

6. まとめと今後の課題

本論文ではタプルストリーム処理システムとメディアストリーム処理システムを統合的に利用するスキームを提案し、そのシステム構成を示した。新規モジュールであるデータ取得インタフェースとメディアデータサーバについて述べ、ストリーム処理システムである SS*についても触れた。

本論文ではメディアデータとして Kinect データのみを記載して、今後、当研究グループで研究開発したメディアストリーム管理機構[2]と連携し、一般的なプロトコルによる映像データ配信を可能にするシステムを構築する。

謝辭

本研究の一部は科研費（#21240005, #22700090）による.

参考文献

- [1] D. J. Abadi, et, al. "Aurora: a new model and architecture for data stream management", VLDB Journal Vol.12, No.2, pp.120-139, 2003
- [2] 高橋翼, 川島英之, 北川博之. ストリーム処理システムにおける映像データ管理機構. Proc. SMDMS, 2011
- [3] Yousuke Watanabe and Hiroyuki Kitagawa, "Query Result Caching for Multiple Event-driven Continuous Queries", Information Systems, Vol. 35, No. 1, pp. 94-110, 2010.