

## 検出精度を維持した並列分散によるテロップ検出

## A Parallel Distributed Processing Method for Accurate Caption Detection

後藤 充裕† 佐藤 隆† 東野 豪†

Mitsuhiro GOTO Takashi SATOU Suguru HIGASHINO

## 1. はじめに

動画共有サービスやライブ配信サービスの流行により、現在は、ネットワーク上に流通する大量の映像を視聴する時代となってきた。

映像は、画像と音声データが時間的に変化するメディアであり、映像を視聴するには、映像長に応じた時間を消費することになる。映像内容の把握を短時間で支援するために、画像や音の特徴を利用した映像イベント検出技術が提案されている[1]。例えば、シーンの切り替わりを表すカット点や画面中にテロップ文字が表示されている映像区間を検出し、そのイベントに対応する映像区間のサムネイル画像を提示することにより、映像全体を俯瞰して見ることが可能になり、映像内容を把握する時間を短縮できる。

しかしながら、映像イベント検出は、映像が持つ大量のフレーム画像や音データから抽出した特徴量の時間変化をフレーム単位で評価しながら、映像中の特徴的な区間を探索する処理であるため、フレーム数に比例した処理時間を要する。

従来より、この高速化に向けて様々な試みが取られている。主なアプローチとして、マルチコア CPU 上での SIMD 命令を利用した並列化[2]や、GPU による高速化[3]がある。これらのアプローチは処理の高速化に有効である一方で、元の検出アルゴリズムに対して抜本的な変更を加える必要がある。そこで我々は、対象とするアルゴリズムへの変更を最小限に抑えながら、マルチサーバやマルチコア CPU を用いた並列分散方式による高速化の実現を目指している。

イベント検出の並列分散を実現する単純な方式は、処理対象の映像を固定長の処理区間に分割して、各処理区間をそれぞれを独立して動作する検出プロセス（以下、ワーカと呼ぶ）に割り当てる方式である。ワーカは、割り当てられた処理区間に対して、イベント検出を実行し、その結果を出力する。この方法は処理時間を短縮できる一方で、イベントの誤検出が発生することがある。誤検出の原因は、割り当てた処理区間の終了点がイベントの存在区間を横切り、イベントを全て検出する前に、ワーカが処理を終了してしまうためである。

我々は、この誤検出を防ぎながら並列分散を実現する方式として、イベント区間をカバーするように処理区間を動的に延長する方式を提案し、平均イベント長の短いカット点やカメラワーク検出において検出精度を損なうことなく、高速化できることを確認した[4]。

本稿では、平均イベント長が長く処理区間の延長の発生しやすいテロップ検出を対象にして、本方式の処理区間の動的な延長が検出精度の維持に有効であることを評価実験により確認したことを報告する。

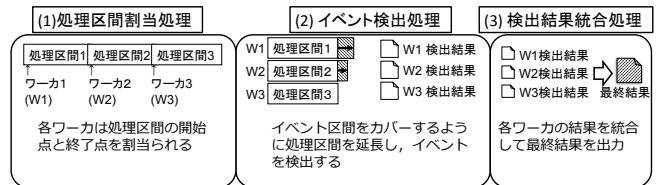


図 1 提案手法の概要

## 2. 提案手法

我々の提案する並列分散方式は、複数のワーカで 1 つの映像データを共有して、イベント検出を並列に実行して高速化を図る。図 1 は、本方式の概要であり、以下の順序で処理を進める。

- (1) **処理区間割当処理**: 映像長をワーカ数で等分割し、処理区間の開始点と終了点のタイムスタンプを求め、各ワーカに割当てる。
- (2) **イベント検出処理**: 各ワーカは割り当てられた処理区間に対して、イベント検出を進め、検出結果を出力する。
- (3) **検出結果統合処理**: 各ワーカが出力した検出結果を統合し、最終結果を出力する。

以降では、各処理のポイントについて詳細を述べる(図 2)。

## 2.1. 処理開始点の補正

映像を途中までシークして検出処理を実行すると、イベント検出器の内部状態が映像先頭から処理した場合と一致せず、処理開始点から正しくイベントを検出できないことがある。そこで、分割しない場合と検出器の内部状態を一致させるように、割り当てられた本来の処理開始点より前方にシークして少し前のフレームデータを読み込みイベント検出処理を実行する。対象とするテロップ検出では、内部状態の一致に 2,000ms 前のフレームデータを必要とする。

## 2.2. 処理区間の延長

ワーカは、割り当てられた処理区間の開始点まで映像をシークし、さらに上記補正点まで開始点を補正し、終了点までイベント検出処理を進める。処理区間の終了点に到達したときに、ワーカはイベント検出状態を基にして処理区間の延長を判断する。イベント検出中ではない状態で終了点に達した場合、そこでイベント検出を終了する。しかし、イベントを検出中の状態で終了点に達した場合、現在検出中のイベントが終了するまで処理区間を延長しながら、イベント検出を続行する。このように処理終了点を延長し、隣り合う処理区間にオーバーラップ区間を設けることで、正しくイベント区間を検出することができる。

†日本電信電話株式会社

サービスエボリューション研究所

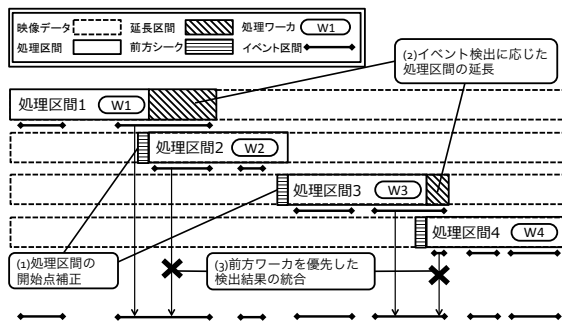


図 2 提案手法の特徴

### 2.3. 検出結果の統合

ワーカが処理区間を延長すると、後方のワーカ処理区間との重複してイベントを検出する映像区間が発生することになる。そこで、正しくイベントを検出するのに必要なフレームを全て参照している前方ワーカの検出結果を優先して統合をする。具体的には、検出結果が一部重複している場合や前方ワーカ検出結果が後方ワーカの検出結果を全て含んでいる場合に、後方ワーカの検出結果を破棄して、前方ワーカの検出結果のみを用いる。

## 3. 評価実験

### 3.1. 実験システム

2 節で述べた提案に基づきプロトタイプシステムを実装した。プロトタイプシステムは、ネットワーク接続された 1 台の管理サーバと、7 台のイベント検出サーバ（各サーバ最大 2 ワーカで実行）、映像ファイルを共有する 1 台の共有ストレージで構成される。管理サーバが解析区間割当処理と解析結果統合処理を、イベント検出サーバがイベント検出処理を、それぞれ実行する。

### 3.2. 実験条件

提案手法が平均イベント長の長いテロップ検出に有効であることを確かめるため、ドラマ、バラエティ、ニュースなど複数ジャンルの映像 11 本を用いて実験した。映像の長さは 3~10 分であり、映像フォーマットは Motion JPEG、画面解像度は 320×240 である。

これらの映像に対して、分割しない場合（1 ワーカ）での実行処理時間および検出結果を正解データとした。次に、ワーカ数を増やしながら、以下の 3 パターンの処理区間の設定方式で、処理時間と検出精度を評価した。

1. **提案手法**：イベント検出処理の状態に応じて、動的に処理区間を変更する。
2. **固定長分割**：隣り合う処理区間に 5,000ms のオーバーラップする区間を設ける。
3. **単純分割**：隣り合う処理区間にオーバーラップ区間を設けない。

### 3.3. 実験結果・考察

各設定方式におけるテロップ検出の結果を図 3 に示す。なお、各値は 11 本の映像の平均値である。

図 3 の(a)より提案手法の検出精度は、ワーカ数によらず正解データと一致した（全てのワーカ数で 100%）。

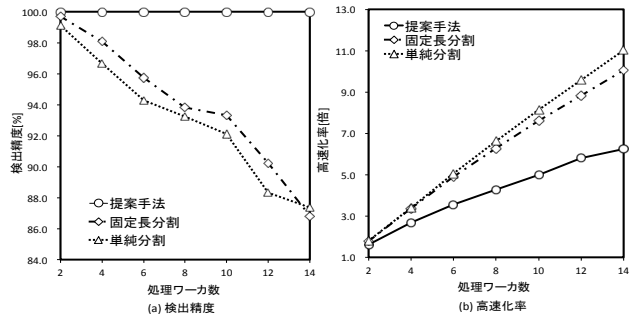


図 3 実験結果

一方で、固定長分割と単純分割の検出精度は、ワーカ数の増加に応じて検出精度の低下が発生した(14 ワーカ時に、固定長分割：86.8%，単純分割：87.4%)。

図 3 の(b)より提案手法の高速化率は、14 ワーカを用いて、1 ワーカでの処理の 6.2 倍を達成した。また、固定長分割と単純分割の結果(14 ワーカ時に、固定長分割：10.1 倍、単純分割：11.0 倍)と比較すると、ワーカ数の増加に応じて、高速化の度合いは鈍化した。

以上の実験結果は、提案する処理区間の延長が、並列分散時の検出精度の維持に有効である一方で、割当てたワーカ数に見合った高速化率の達成に影響があることを示している。

このことから、平均イベント長が長いテロップ検出の並列分散時には、検出精度と高速化率がトレードオフの関係にあり、このような特性は、要求される検出精度・高速化率に応じて、処理区間の設定方式を使い分ける上で有用であると言える。例えば、厳密な検出精度が必要な場合には提案手法を利用し、多少の誤検出を許容しても高速化が必要な場合には、延長しない方式を利用することが考えられる。

## 4. まとめ

本稿では、イベント区間をカバーするように処理区間に必要な長さだけ動的に延長する並列分散方式を平均イベント長が長いテロップ検出に適用し、検出精度と高速化率について評価した。実験結果より、平均イベント長が短いカット点やカメラワーク検出では、発生しなかった検出精度と高速化率のトレードオフの特性を持つことを確認した。

今後の課題は、効率的な高速化率を達成するワーカ数の設定について検証を進めることである。

### 参考文献

- [1] 谷口行信, 南憲一, 佐藤隆, 桑野秀豪, 児島治彦, 外村佳伸. Scenecabinet: 映像解析技術を統合した映像インデクシングシステム. 信学会論文誌. D-II84(6):pp. 1112-1121, 2001.
- [2] M. Diao, C. Nicopoulos, and J. Kim. Large-scale semantic concept detection on manycore platforms for multimedia mining. In Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International, pp. 384-394. IEEE, 2011.
- [3] K. E.A. van de Sande, T. Gevers, and C. G.M. Snoek. Empowering visual categorization with the gpu. Trans. Multi., 13(1), pp.60-70, 2011.
- [4] 後藤 充裕, 佐藤 隆, 東野 豪. 並列分散による映像イベント検出の高速化. 情報処理学会研究報告. AVM76, 2012(2):1-6, 2012