社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2003－FI－73 (9)
2003／11／13

無音を利用したイベントと映像の同期方法

○ 疋田聡、高橋望、鷹見淳一、國枝孝之

抄録：
マルチメディア技術の一つとして、別々のマシンで記録された映像、音声とイベントとの同期を、無音区間を利用して自動的にとることが考えられる。
しかし、①イベント発生時に必ずしも無音になっていない、②イベント発生時以外にも多くの無音部分が存在するため単に無音部分をイベントと対応付けても誤りが多数発生してしまう、という問題点があった。また、無音区間の検出数は、背景ノイズのレベルに依存するが、同期を行うためにはどの程度の無音区間の判定閾値が適当であるかが分からないという問題点もあった。
そこで、イベント発生時に必ずしも無音になっていない場合を含み、イベント発生時以外にも多くの無音部分が存在する場合にもロバストにずらし時間を検出し、同期を可能とする方法を示し、実験によりその有効性を確認した。

Robust Synchronous Function Using Silent Sections

○ S. Hikida, N. Takahashi, J. Takami, T. Kunieda

Abstract：
There is an idea of taking the synchronization with images, sounds and events automatically using the silent section. However, it does not work by the simple method. The first reason is that sounds are not always silent at the time of events generating. The second reason is that many silent sections exist besides the time of events generating. Then, we proposed the method of making a synchronization possible also under such conditions, and showed the effectiveness by experiments.

著者所属：株式会社リコー
Affiliation: RICOH COMPANY, LTD.

1. 背景

近年、マルチメディア技術の活用が盛んになってきている。例えば、電子情報通信学会第 14 回データ工学ワークショップ（DEWS2003）[1]では、研究発表の大部分をプレゼンテーション Web コンテンツ自動生成ツール「MPMeister」[2]を用いて撮影収録し、並列トラックのために聴講できなかった参加者のために会場でプレゼンテーション資料と同期した発表の様子を閲覧可能とすると同時に、そのコンテンツをアーカイブ[3]に格納して後で閲覧できるようにするなどの試みが行われるようになってきた。このようなマルチメディア技術の利用においては、映像や音声データの取得とプレゼンテーションのスライドをめくる等のイベントの記録とが別々のマシンで行われるようなパターンを考慮することも必要となってくる。この解決策を単純に考えると、マシン間の時計を合わせておくとか、ネットワーク通信で同期を取るなどが考えられるが、上記の DEWS2003 のようなワークショップのような状況では、さまざまな人が自分のマシンを持ち込んで次々とプレゼンテーションを行っており、事前にマシンの時計を合わせたり、ネットワーク設定を行っている時間が無い場合も多々発生する。そのため、マシン間の時間同期が取れていない場合にも、取得された映像、音声、イベントデータ自身から、データ間の同期を自動的に行える機構の必要性が認識されるようになった。

2. 解決すべき課題

映像や音声データの取得とプレゼンテーションのスライドをめくる等のイベントとが別々のマシンで行われ、その間で通信の利用ができない場合、映像や音声とイベントとの間の同期を行おうとすると、手動で同期をとらなければならない。例えば、スライドページがめくられたのを目で見て、その時の映像のカウンターの値をメモする、映像の一部にプレゼンの画像が入るようにするなどである。このような余分な手間を減らす方法として、無音区間を利用して映像や音声とページめくり等のイベントの同期を自動的にとることが考えられる。しかし、①イベント発生時に必ずしも無音になっていない、②イベント発生時以外にも多くの無音部分が存在する。したがって、単に無音部分をイベントと対応付けても誤りが多数発生してしまうという問題点があった（図1）。

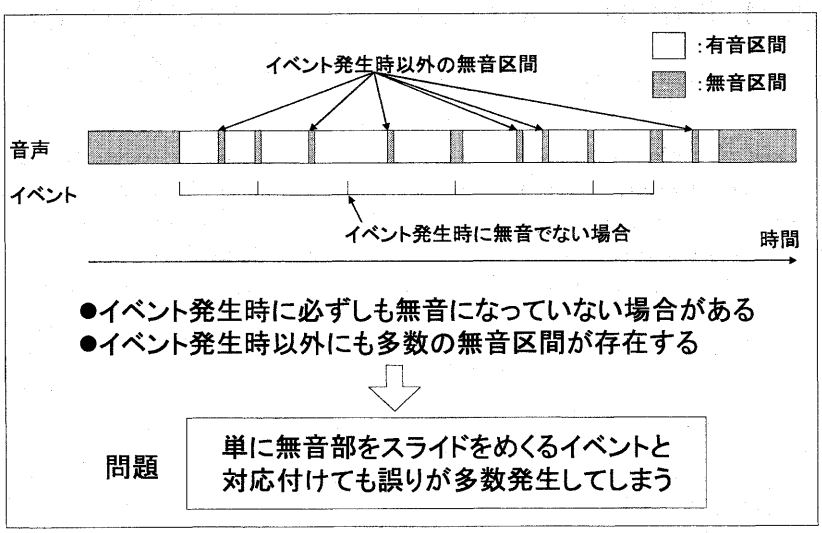

図1. 解決すべき課題

また、無音区間の検出数は、背景ノイズのレベルに依存するが、同期を行うためにはどの程度の無音区

間判定閾値が適当であるかが分からないという問題点もあった。

そこで、イベント発生時に必ずしも無音になっていない場合を含み、イベント発生時以外にも多くの無音部分が存在する場合にもロバストにずらし時間を検出し、同期を可能とすることが解決すべき課題となる。

3. 方法

イベント1個1個と無音部を対応付けるのでなく、イベント全体と無音部との適合度のスコア関数を定義し、スコアが最大となるずらし時間を探索する。スコア関数としては、例えばイベント時刻が音声の無音区間に入っていればスコアを1点足すなどが考えられる（図2）。

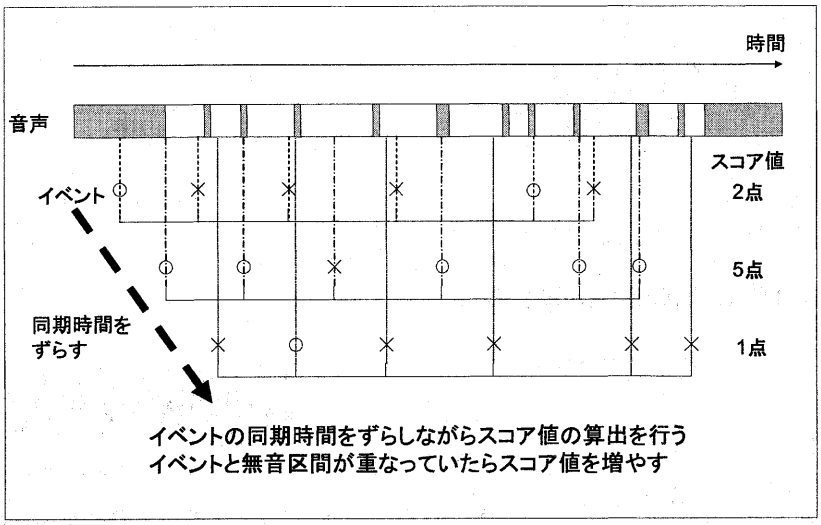

図2. 方式

このようなスコア値の最大値は事前には分からないが、スコア値のピーク検出によりスコアが最大となる同期時間を求めることができる。これにより、スライドをめくるときに無音でない場合が含まれていてもピーク検出により、ロバストに同期時間を得ることができる（図3）。

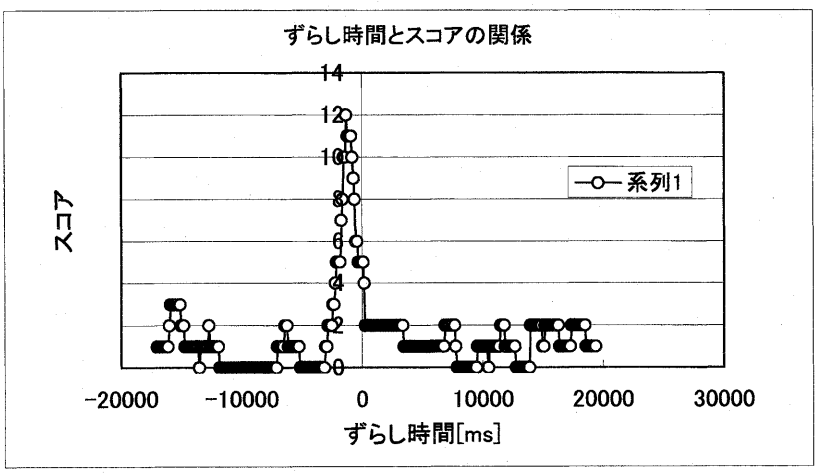

図3. スコアのピーク検出

また、無音区間の検出数は、背景ノイズのレベルに依存するので、無音区間を判定するための閾値を適当に調整しないとうまくいかない場合が出てくる。例えば、無音区間判定用閾値が高すぎると、無音区間が多くなりすぎ、正解以外のスコアも高くなってしまいうまくピーク検出ができない（図４）。無音区間判定用閾値が低すぎると、無音区間が少なくなりすぎ、正解のスコアも低くなってしまい、この場合もうまくピーク検出ができない（図５）。そこで、無音区間の数とイベントとの数との割合が適当になるように無音区間判定閾値を自動調整し、さらに、スコアの最大値と他のピークのスコア最大値との差が大きくなるように無音区間判定閾値を自動調整する。この方法により、録音時のノイズレベルに左右されずにロバストな同期が可能となる（図６）。

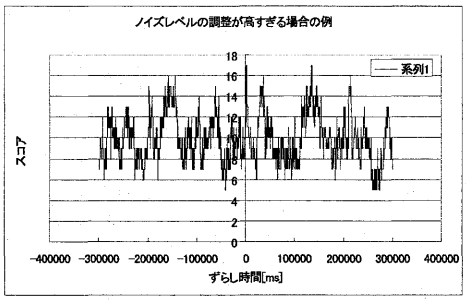

図4. 無音区間判定閾値が高すぎる例

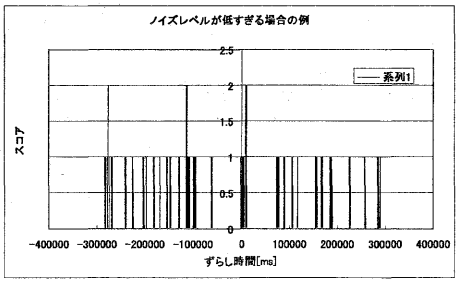

図5. 無音区間判定閾値が低すぎる例

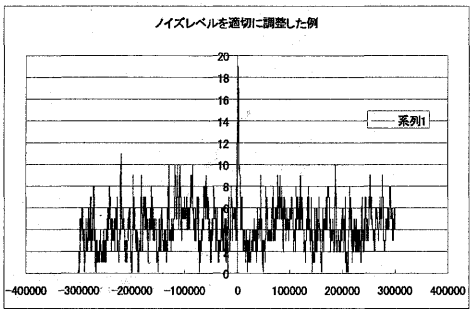

図6. 無音区間判定閾値が適当な例

図７に本方式のフローチャートを示す。

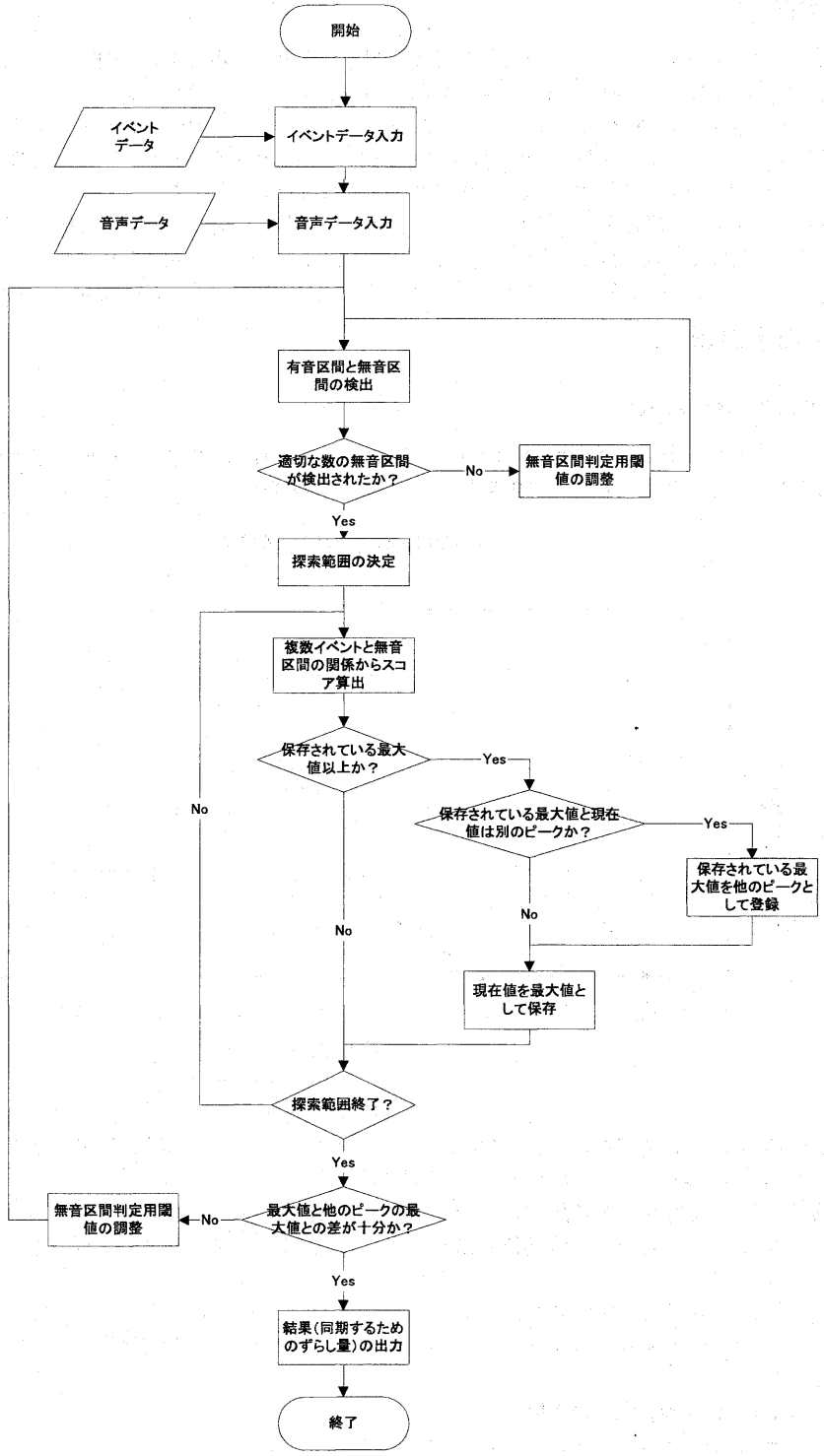
開始
イベント
データ
イベントデータ入力
音声データ
音声データ入力
有音区間と無音区
間の検出
適切な数の無音区間
が検出されたか？
No
無音区間判定用閾
値の調整
Yes
探索範囲の決定
複数イベントと無音
区間の関係からスコ
ア算出
保存されている最大
値以上か？
Yes
No
保存されている最大値と現在
値は別のピークか？
Yes
保存されている最
大値を他のピークと
して登録
No
No
現在値を最大値と
して保存
探索範囲終了？
Yes
無音区間判定用閾
値の調整
No
最大値と他のピークの最
大値との差が十分か？
Yes
結果（同期するため
のずらし量）の出力
終了

図7. フローチャート

4. 実験

過去にコンテンツとして作成したデータから、評価対象サンプルを１６個抽出し、本方式にて同期処理を行った。各コンテンツは、１５分から１時間程度の長さのプレゼンテーション発表であり、Web コンテンツ自動生成ツール「MPMeister」を用いて撮影収録してコンテンツ作成したものである。実験に用いたコンテンツは、ネットワークが使用可能な環境下で撮影収録したものであり、正解の同期ずらし時間は分かっているので、結果の評価は、その正解同期ずらし時間との差で行った。

また、手動による同期との比較として、ネットワークが使用できない環境下で撮影収録したもので、以前に手動にて同期処理を行ったものの内、目視で同期に誤りがあると認められるコンテンツ（正解と判断できるタイミングから３秒以上のずれがあるもの）４例についても本方式にて同期実験を行った。

5. 結果と考察

5.1. 同期のロバスト性

本方式により評価対象コンテンツ１６例全てについて±2 秒以内の同期時間の誤差範囲で、正しい同期を行うことができた。以下に、実験結果として同期誤差の度数分布グラフを図８に示す。

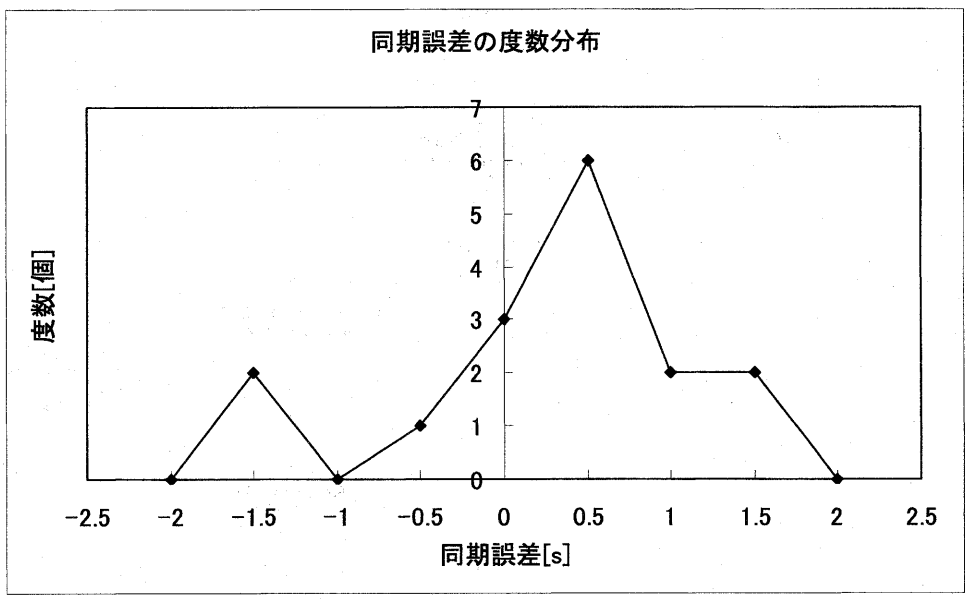

図 8. 実験結果

手動での同期処理で誤りのあった４例についても、本方式により目視では同期誤り認められない程度にまで同期時間を修正することができたことを確認した。

無音区間の判定閾値自動調整の効果については、自動調整機能を無効にしていた場合には、１６例中２例が同期できなかったが、自動調整機能を有効にした場合は、１６例全てが同期できたことにより確認できた。

また、①イベント発生時に必ずしも無音になっていない、②イベント発生時以外にも多くの無音部分が存在する、という点に対するロバスト性を調べるため、あるデータの無音区間とイベントの関係を測定したところ、

検出された無音区間の総数　　213

無音区間に入ったイベント数/全体イベント数　　16/60

となり、イベント数の３倍以上の無音部があり、しかも、ページめくりと無音区間が 1/3 以下しか対応していないような悪条件の下でも、正しい同期時間が得られることがわかった。

さらに、評価に用いた１６例中には、間違いや言いよどみがあり、無音部の多いデータも含まれており、このようなデータにおいても、正しい同期時間が得られることが確認できた。

5.2. 処理時間

図９に本評価に要した処理時間のグラフを示す。本評価で用いたコマンドは、同期のロバスト性の評価に重点を置いたプロトタイプであるため、高速化に関する考慮はあまり行っていないので、今後数倍の高速化が可能と考えられる。したがって、速度の傾向についてのみ参考のこと。

下のグラフで、ファイルにより処理速度に大きく差があるのは、図７フローチャートの下の方の条件判断文で何回かループした場合に、１回で終了した場合の何倍かの時間がかかるためである。このあたりは、ロバスト性と処理時間のトレードオフになっている。しかし、ファイル ID が 16 の処理時間は他のデータに比べて時間がかかりすぎていると思われるので、処理時間によってループを打ち切る処理を入れることを考えている。

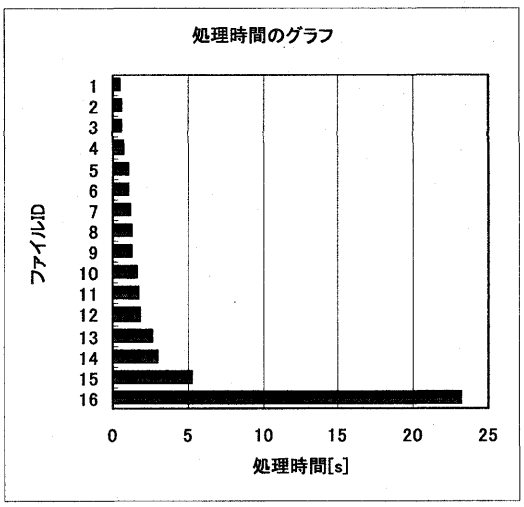

図 9. 処理時間

6. まとめ

映像、音声とイベントとを同期させる方法として、イベント発生時に必ずしも無音になっていない場合を含み、イベント発生時以外にも多くの無音部分が存在する場合にもロバストにずらし時間を検出し、同期を可能とする方法を示し、実験によりその有効性を確認した。

今後は、処理時間の短縮などを行うことが考えられる。

7. 参考

[1] http://www.ieice.org/iss/de/DEWS/DEWS2003/
[2] http://www.ricoh.co.jp/mpmeister/
[3] http://www.dbsj.org/Japanese/Archives/DEWS2003/DBSJarchives.html