目 次

1 はじめに **3**

2 映像アーカイブ利用のためのアプローチ **5**
2.1 映像アーカイブ構造化とは 5
2.1.1 デジタル映像の現状 5
2.1.2 テキストからの情報抽出 9
2.1.3 画像特徴による情報抽出 10
2.1.4 時空間特徴による情報抽出 13
2.1.5 目的に特化した情報抽出 15
2.2 映像照合 18
2.2.1 同一とされる映像とは 18
2.2.2 Near Duplicate 20
2.2.3 同一場面映像照合 23
2.3 手法概要 25

3 フラッシュパターンによる照合 **28**
3.1 概要 28
3.2 フラッシュ検出 29
3.3 フラッシュ判定 31
3.3.1 輝度差分画像の類似性による判定 31
3.3.2 輝度変化の発生パターンによる判定 34
3.4 パターン比較 35
3.4.1 パターン間類似度 35
3.4.2 高速化のためのアルゴリズム 37
3.5 実験による評価 39
3.5.1 フラッシュ検出 41
3.5.2 パターン比較 43

4 動きの変化パターンによる照合 **47**
4.1 概要 47
4.2 動きの検出とパターン生成 49
4.3 ショット間の比較 52
4.3.1 軌跡間の類似度 52
4.3.2 ショット間の類似度 54
4.4 実験による評価 56
4.4.1 映像照合性能の評価 56
4.4.2 アーカイブへの適用 59
4.5 アーカイブ構造化に向けて 65

4.5.1 ニュース番組における有益な映像区間の抽出 65
4.5.2 リンク数、リンク密度ベースの区間抽出 66
4.5.3 実験結果 67

5 結論 **70**
5.1 提案手法に関する結論 70
5.2 今後の課題 72

1　はじめに

本研究は、大量に蓄積された映像を効果的に利用するための、映像データベースの構造化を目標としたものである。大量の映像を扱う環境は近年著しく充実してきているが、蓄えた映像を内容的に扱うような技術はあまり発展しているとはいえない。例えばMPEG7ではメタデータの記述が可能であるが、このように人手で付加するものに関しては、記述する内容の選択基準が偏るのが問題である。そもそもあらゆるユーザが望む情報をメタデータとするのは不可能であるため、何らかの形で自動的に多くの映像を解釈し、それらの内容や関連性を抽出する手法が求められることとなる。

そこで、映像の内容に基づくようなアーカイブのインデキシングのために、同一の場面を撮影する部分映像の存在に注目した。本研究ではTV放送映像を解析対象とするが、TV放送映像の特徴として、同じ場面を撮影した映像が繰り返し利用される、というものがある。同一場面の映像が使われているということは、その映像が含まれる番組間に内容的な関連性が存在することを示し、インデキシングのための有意義な情報となりうる。また、繰り返し利用されるような映像は、それらが放映された時期に少なからず注目を集めたような内容を含むものであることが推測され、大きな内容的価値を有すると考えられる。

このような理由から、本研究は同一場面を撮影した映像のアーカイブからの発見、もしくは検索を目標とする。そして、そのために必要となる、場面という観点からの同一性を示すための映像照合手法を本稿において提案する。この手法において解決すべき最大の問題は、同一場面映像間の視覚的な差に関わらず照合を行うことである。撮影しているカメラ、そして編集により映像に追加されるテロップなどは、番組により異なるものであり、これらの違いがもたらす視覚的な差は非常に大きい。また、一つの場面の内、放送に利用される部分も番組ごとに異なるため、このような時間的な違いも吸収する必要がある。

これらの要件を満たす手法として、本稿では2種類の手法を提案している。この2つの手法に共通する点は、映像データから求められる特徴量が時間的に変化する、そのパターンに基づいて照合を行うことである。一つは、カメラのフラッシュのたかれる場面における、フラッシュ光の発生パターンを利用するものである。もう一つの手法では、映像内の物体の動きに着目し、その変化のパターンを利用している。これらはカメラの違いやテロップの存在などの影響をあまり受けないため、場面の同一性による照合の基準とすることができる。また、これらは特徴量の時間パターン、つまりフレームごとに検出される特徴量の配列であるため、放送される区間の時間的な差にも対応できる。

加えて本研究では、大規模な映像アーカイブを前提としているため、計算時間の抑制も最重要課題である。既存の画像もしくは映像に関する解析手法は、その精度が重要視される一方で、このような問題はあまり重視されない傾向にある。それに対し本稿の提案手法は、例えば検索などのアプリケーションに利用すること

を想定した上で、現実的な計算時間で処理を行うことを目標としている。このようなアプローチは、大規模な映像アーカイブといった環境が今後さらに充実するに従って、より強く求められる要素であると言える。

以上が本研究の要点であり、これらを詳細に述べるものとして、本論文は以下のように構成される。2 章では、研究内容の概要と、その背景についての議論を行う。3 章、4 章では提案手法の詳細を説明し、それらを適用した際の実験結果を示す。5 章では全体のまとめを行う。

2　映像アーカイブ利用のためのアプローチ

2.1　映像アーカイブ構造化とは

2.1.1　デジタル映像の現状

本研究の目的は、デジタル映像の蓄積された映像アーカイブを、映像内容という側面から構造化することである。近年、デジタルデータを保存するためのディスクの容量は拡大を続け、また、高速なネットワークが世界中に広まっている。このような著しいインフラの整備により、デジタル映像の利用も大幅に拡大している。元々映像は非常に大きな情報を持つものであり、例えばMPEGのような方式でデジタルデータとして圧縮した場合、そのサイズはテキストや画像に比較して遥かに大きなものとなる。よって、近年の技術の急速な進歩により、初めてデジタル映像が広く利用されるようになったといえる。

このような背景の下、映像を蓄積して閲覧するという方式が、段々に一般的なものとなりつつある。つまり大量の映像のアーカイブ化である。実際に国立情報学研究所 (NII)[1] では、7000時間分もの TV 放送映像を MPEG 形式で保存し、日々更新を続けるデータベースが稼動している。このデータベースでは、テラバイト単位のディスクを管理するシステムが利用されており、一昔前ではとても実現できなかった環境が、すでに実用化されているということである。また、大規模な映像アーカイブの例として、この他にも例えばNHKは放送した番組の一部分を川口アーカイブ [2] と呼ばれる大規模な施設に保存している。ここに保存される映像は、それらを検索・閲覧するためのメタデータとともに大規模なデータベースで管理されている。これらはともに放送映像をアーカイブ化している例であるが、放送映像というものは、それぞれ放送された時期の出来事や風潮を直接現すものであり、歴史的な遺産としての価値は非常に高い。また、デジタルデータであるために劣化の心配が無いという利点もあり、このような大規模な映像アーカイブは今後も様々な場所で利用されていくものと考えられる。

大規模なアーカイブシステムに対し、一般的にも DVD レコーダにより映像を保存する傾向が広まっている。以前、映像はビデオテープにより保存されていた。しかしテープはかさばるものであり、また、時が経てば自然とその画質は劣化する。デジタルデータによる映像の保存という方式は、これらの問題を解決するものである。また、番組情報などの外部のデジタル情報と連動させることが容易であり、録画や閲覧、さらにはそれらに対する編集など、以前のテープによる映像の保存に比べて遥かに便利な利用形態が可能となっている。例えば事前に登録することで、好きな番組を自動的に録画する、CM を自動的に飛ばす、放送中に見逃した場面を巻き戻して見れる、などのアプリケーションが実用化されている。実際に DVD レコーダは、すでにデジタル三種の神器と言われるほどに世の中に広まっており、以前は基本的に見たらおしまいという使い捨てのコンテンツであった映像が、保存することで好きな時に閲覧できるものへと進化を遂げている。また、ユー

ザの意識もこれに従って、より映像を積極的に、かつ便利な形で利用しようとする方向に変遷している。

図 1: NII の TV 放送映像アーカイブシステム

そこで、大規模なアーカイブシステムに関して、いかにして蓄積された映像をうまく閲覧するかということが問題となる。単に蓄積するだけではその後の利用がいかにも不便であり、何らかの映像内容に関する整理が必要となる。このような問題に関して、NII の TV 放送映像のアーカイブは (図 1)、放送日時とそれに対応する番組表を関連付けて保存しており、見たい番組に対応する映像ファイル (1 時間ごとの MPEG ファイルとして保存されている) にアクセス可能となっている。また、番組内の音声をテキストデータとしたクローズドキャプションデータが同時に保存され、それに対するテキスト検索を行える。これらの情報により、映像の選択的な閲覧が可能となっている。ただ、このような形態での映像へのアクセスだけで、ユーザにとって十分に便利な映像アーカイブになっているとまではいえない。例えば以下のような点において、十分であるとはいえない。

- 番組という映像単位は大まかなものであり、一つの番組内で扱われるトピックの量や種類は非常に多い。それらのトピックや、シーンごとのアクセスを行えることが望ましい。
- クローズドキャプションに対する単純なテキスト検索は、Web サイトの文書に対する単純なテキスト検索のようなもので、内容を直接表すものではない単語の存在などによるノイズが非常に大きい。また、映像内容と完全に同期が取れている保証もない。

前者の問題に関して、最近はユーザが注目する映像を部分的に抽出してそれを保存、共有するという傾向が顕著に見られる。多くのユーザにとって、興味をひかれる映像とは番組の一部分である場合が多く、そのような箇所がユーザの主観によって抽出されている。これは、映像の製作者により与えられる情報とは全く違った意味を持つものである。例えばYoutube[3]がその最たるところである。Youtubeでは、ユーザがアップロードした映像に対する検索を行うことが可能であり、このようなサービスとしては他にGoogle Video[4]などが存在する。Youtubeの一つの特徴として、それぞれの映像に対してユーザが独自にメタデータを付けることができる点が挙げられる。これは即ち映像製作者の提供する情報だけによらない、ユーザの側でのインデキシングが行われていることを示し、実際に放送映像内の一部分が抽出されたような映像が数多く存在する。この結果として、Youtubeは最大一日あたり一億とも言われるほどの膨大なアクセス数に表されるような、非常に大きな人気を得るサイトとなっている。この事実に示されるように、一般に映像そのものの持つ魅力は非常に大きい。そして、映像をより細かな単位で、ユーザの様々な嗜好に合わせたような形で分類する効果も非常に大きい。Youtubeのようなサービスでは、それらが人手により実現されているが、もし大量の映像に対して自動的にこのような処理を施すことが可能であれば、それはアーカイブの利便性を大きく向上させるものであると考えられる。

そこで、自動的な映像の分類といった目的を実現するためのアプローチとして、本稿は映像アーカイブの構造化という提案を行う。構造化、とはすなわち、なんらかの物や事象を階層的にとらえ、階層ごとに段階的に詳細化していくことを言う。映像アーカイブという例においては、映像の持つ様々な属性、例えばその映像はどの番組に属するものか、映像はどのような物体を映すものか、その映像はどれほど注目されるものか、と言ったようなあらゆる観点から、アーカイブ内の映像データに対する、より細かなアクセスを行えるだけの情報を抽出することにあたる。映像は本質的に非常に多くの情報を含むものであるため、そもそも単一の手法による映像把握を行うというのは現実的ではなく、映像データそのものにおける特徴から、映像の持つ意味合いなどのメタ情報に至るまで、多様な捉え方による構造化を行うことで、初めて利便性に富んだ映像アーカイブとなるものと考えられる。

構造化の一例として、映像を時間的に構造化することを考える。もともとデジタル映像とは、離散化されたフレームの集まりとして構成されるものであり、フレーム単位のアクセスが基本となる。また、番組表情報を利用することで番組単位のアクセスが可能となっている。これらの時間的な映像の分割と分類の手法に加え、一つの映像をショット単位に分割するショット分割点検出手法が古くから研究されている。ショットとは時間的に連続している部分映像であり、一つのショットであれば、カットやワイプなどの映像編集技法による、連続しない映像の接続点を含むことは無い。複数のショットが様々な編集方式により接続され、その後放映されるというのは、放送映像における一般的な形式であり、それゆえにショッ

ト分割点を検出することで、一つの映像を、内容的に連続することが保証される複数の部分映像へと分割することができる。現在でも、多くの研究者がこのテーマに携わっており (TRECVID2006[5] では、26 の研究グループがその成果を報告している)、精度と処理の高速性に関して、技術の向上が見られる。なお、ショット分割は映像データそのものだけを解析することで行われているが、最近ではクローズドキャプションに代表されるテキストデータや、番組構造などの予備知識を利用することで、関連するショットをまとめたようなシーン単位での分割を行う研究も盛んである [7, 9, 10]。シーンとは、低次元データに対する解析の結果であるショット分割を前提とし、関連するショットをまとめることでショットより大きな単位での時間的な階層化をもたらす概念である。これらの概念に基づいて映像アーカイブは時間的な構造化が成され、その結果としてユーザはより細かい単位での映像へのアクセスが可能となる (図 2)。

図 2: 時間的な構造化の例

ただしこれはあくまでも一例であり、細かく分割された映像も、それぞれが何を撮影したものであるかという情報なしでは、個別の部分映像へ有効にアクセスする手段に欠ける。例えばイマジカ [6] やいくつかの TV 局は、多くの映像を素材映像と言う形式で提供するサービスを行っている。映像内に何が映っているか、映像の撮影された場所はどこか、などといったメタ情報を利用することで、映像を利用するクリエイターの作業の手助けとなるような素材映像を検索できるようなシステムである。これは映像を単に分割するだけに留まらず、その内容の分類まで行うことで利便性を高めている好例と言えるだろう。しかし重要な問題として、そのようなメタ情報を付加するのも、またはメタ情報の分類を行うのも、全て映像提供側による人手作業であることが挙げられる。人手による作業は、当然コストがかかるものであるし、何より映像提供者の主観に基づく内容分類しか行えないことが大きな問題である。新しい概念により分類された映像を提供する場合は、

また人手による分類を行う必要があり、これがこのようなサービスの限界である。上で述べたように、現在ではユーザ側からの映像の選別を行えることが求められており、そのような現状を鑑みるに、映像の製作者や閲覧者の視点に拠らないような、映像内容の取得による構造化を行うことが重要であると考えられる。

このようなデジタル映像の利用形態から、現状必要とされていると考えられる要件を以下にまとめる。

- 大量の映像に対する、様々な視点からの分類。
- 映像内容を表すような、客観的な情報の自動抽出。
- 抽出された情報による映像のインデキシング。

これらを満たすような技術こそが、映像アーカイブに対する構造化技術にあたり、本研究ではその一端を担うような技術を開発することを目標とする。なお、このような目的に対して行われている研究は数多くあり、それらはテキストによるもの、映像内の画像的な特徴を扱うもの、といった具合に大まかに分類できる。それぞれに関しては、以下の節で例を取り上げて説明することとする。

2.1.2 テキストからの情報抽出

映像の内容を表現する情報として、最も一般に広く利用されているのはテキスト情報である。映像の内容を表現する上で、テキストが最も直観的に扱える形式であることは間違いない。MPEG7 では、テキストにより映像の内容をメタデータとして記述することが可能である。このようなメタデータは、映像の一部分に対して記述できるようになっており、テキストによるシーンの検索などに利用できる。しかし上で述べたように、テキストからなるメタ情報を人手で付加するという方式は、コストや客観性の面で問題がある。そこで、映像内の発話情報をテキスト形式に保存したような、クローズドキャプションの利用が注目されている。クローズドキャプションの作成は基本的に人手によるものであり (将来的には音声認識による自動化などが期待される)、その作成にかかるコストは大きい。しかし、既にクローズドキャプションを提供するサービスは現実に行われており、また、人手によるメタデータの記述と異なり、主観に左右されることのないデータであるため利用価値は高い。

映像内の音声にあたるテキストを利用することで、ニュース映像内のトピックの検出とその追跡を行う手法 (TDT:Topic Detection and Tracking) が [13] などで提案されている。テキストをベクトルモデルで表現する言語処理手法は古くから行われており、そのベクトル空間でのクラスタリングを行うことで、同じトピックを扱ったようなニュースが抽出される。また、そのクラスタを学習データとして利用することで、新たな映像が与えられたときに、そこから同じトピックを検

出する。このような手法では、テキストに対する自然言語処理や、TFIDF による単語の重み付けなどのテキスト検索手法が応用されている。

また、映像内の物体に直接注目するような手法として、NHK アーカイブスはクローズドキャプションの自然言語処理による被写体検出技術を提案している [11, 12]。クローズドキャプションに現れる多くの名詞から、その映像の中心として撮影されている被写体に対応するものを求めるというアプローチである。図 3 の例では、テキストの中に含まれる二種類の名詞の内、被写体となっているのは「ウサギ」であると判断し、「ウサギ」という単語とこの映像との関連付けを行っている。これは即ち、映像の主体となるオブジェクト、もしくはコンセプトにあたる名詞により、映像に対するラベル付けを自動的に行うものであり、その結果として、何らかの名詞をクエリーとして与えることで、それが中心となって映し出されているような映像を検索することが可能となる。クローズドキャプションに対する単なるテキスト検索と比べ、自然言語処理による単語の選別を行うことで、その精度を大幅に向上しているのが大きな特徴である。このシステムはマルチメディア百科事典とも呼ばれ、素材映像の提供といったアプリケーションとも関連する。アーカイブ構造化の観点からは、人手を介さない、自動的なテキストと映像との関連付けをしている点で重要な意味合いを持ち、映像アーカイブの効果的な利用のためにはこのようなアプローチは欠かせないものである。

図 3: 自然言語処理による被写体検出

このようにテキストによる情報抽出は、映像アーカイブの構造化においても大きな効果を持つ。特に放送のデジタル化が進んでいる現在、日本においてもクローズドキャプションの提供が活発に行われ始めており、映像の内容的なインデキシングのための有益なデータとして活用されるものと考えられる。

2.1.3 画像特徴による情報抽出

テキストを中心とした情報抽出手法に対し、映像の画像的な特徴、つまりデジタル映像を構成するフレームのピクセル値を利用した解析を行うことで、高次元の意味情報を取得することを目標としたアプローチがある。映像データに対して直接適用する手法であるため、基本的にはメタ情報やクローズドキャプションと

いった、外的な情報を必要としない。テキストからの情報抽出技術の紹介で示したように、クローズドキャプションは映像の内容把握をするにあたって非常に有効な情報ソースとなりうるものだが、特に放送映像においては、内容的に映像と完全に同期が取れているという保証は無い。また、テキスト検索は非常に発達しているとはいえ、テキストの内容とあまり関係のない単語により生ずるノイズの影響は避けられない。そこで映像に対する解析というアプローチにより、このようなテキストを利用する際の問題点を補い、さらには全く違った側面からのアーカイブ構造化を行う。映像データの解析は、映像の内容に直に触れられる点で有効であり、また、映像とは画像がさらに時間軸方向に広がったような時空間データであるため含む情報が豊富であるという利点がある。一方で、映像は直観的に構造化できるようなものではないため、取り扱うのが難しいという根本的な問題がある。

そこで、映像を扱う手法は時間軸方向のデータを大幅に圧縮した方法が採られることが多い。特に分割されたショットから、キーフレームとなるフレームを選出し、そのフレームに大して画像処理手法を適用するアプローチが一般的である。もともと画像処理の分野において、画像に映っているオブジェクトを抽出しようとする試みは古くから成されており、例えばエッジ検出 [30] や領域分割 [29] などがある。その中で、しばしば利用される手法の一つが特徴点検出に基づくものである [17, 22]。画像の中に映る物体の中の特徴的な点を自動的に検出し、それを何らかの特徴量で表現することでその物体のモデルとする。そこで得られたモデルを、別の画像に適用し、類似するものが見つかればその画像にも同じ物体が含まれているとする、とこのような流れで物体の特徴的な一部分という観点からの画像のラベリングを行うやり方が一般的である。これは汎用的に使える方法であるため、映像のフレーム画像に対しても適用可能であり、その映像内に含まれている物体という観点からの構造化へと応用される。

このような方式において、基本となる特徴点の抽出手法としては、Harris の特徴点検出手法 [17, 20] が広く利用される。ここで定義される特徴点とは、無地の領域からはほとんど検出されず、特徴的な模様のある領域から多く検出される。また、物体の角などにあたる点も、特徴点として選ばれやすい。このような手法は、同じ物体であれば、撮影環境に関わらず同じ点を特徴点として検出することが望ましい。つまり、ある物体を別の角度や距離から撮影した場合や、異なる証明条件の下で撮影した場合、いずれにおいても空間的に対応している点が特徴点として選ばれるべきということである。このような性質は、例えば映像の物体のサイズに関してはscale-invariant などと呼ばれ、特徴点の性能評価を行う一つの指標になっている。このような点で、Harris の手法は高い評価を得ており、そのため多くの特徴点ベースの手法の研究において利用されている。なお、点ではなく、特徴的な領域を抽出するような手法も存在する [18, 21]。特徴点検出をベースとし、その周囲の領域をスケールや向きの変化に対する不変性を持つような楕円として抽出する、もしくは窓のような周囲との境目がはっきりとしているフラットな領域

を直接抽出してくる、などの処理が特徴領域の検出のために行われている。

次に、検出された特徴点を利用して、物体をモデル化するための特徴量が必要となる。この特徴量としては、同一物体上の同一点に対応する特徴点であれば同じ特徴量となることが望ましく、やはり撮影したカメラの違いや照明条件の違いなどに左右されない不変性が求められることとなる。これは何をモデル化するかという目的に応じ、それぞれの手法において独自の特徴量が定義されることも多い。その中で、汎用的に画像の特徴を現す値として、SIFT と呼ばれる特徴ベクトル [19] が存在する。SIFT では、ある点の周囲の小領域内の全ての点において、近傍の点とのピクセル値の差分を求める。この差分を利用することで、それぞれの点におけるピクセル値の変化の大きさと向きを定義し、それらを 128 次元のベクトルとしてまとめたものである。この特徴量は、スケールやアフィン変換などに対する不変性を持つことで知られており、特徴点を表すための特徴ベクトルとして広く利用されている。検出されたそれぞれの特徴点から特徴量を求め、それをいかにして物体のモデルとするかは、手法の独自性が求められる部分である。この目的のため、正解データによる学習を行った上で SVM などにより同一物体が含まれるかどうかの判別を行わせるといったアプローチが広く行われている [28, 31, 32, 33, 34, 35, 36, 37, 38, 39]。その他、特徴点間の位置関係を利用することで、より厳密なモデルの作成を行うなどの工夫が見られる。

これらが特徴点ベースの画像処理でしばしば採られる、一般的な手法の流れである。映像に関しても、適切なキーフレームを選び出し、同様の処理を行うことで、映像内の物体のモデル化へとつなげることが可能である [26, 27]。ただし、一つのショット内で、ある特徴点が最初から最後まで存在し続ける保証は無く、多くの映像において、特徴点は消失したり途中のフレームから現れたりするものである。そこで、このような方式を、映像データに特化する形で応用した手法の典型が、Zisserman らによって提案された Video Google[26] である。Video Google では上で述べたような特徴領域の抽出と、SIFT による特徴ベクトルの算出を映像内の各フレームにおいて行った上で、K-Means クラスタリング手法による領域のクラスタリングを行い、同一と見なせる領域のクラスタによる映像のモデル化、ひいては構造化を行っている。この手法では、特徴領域をトラッキングし、すぐに消えてしまうような領域を捨てることで、映像をモデル化する上で重要な領域のみの選別を行っている。また、トラッキングの結果、同一と見なされる特徴領域をまとめることで、全てのフレームから特徴領域を抽出することによる冗長性も対処されている。その結果、画像領域をクエリーとして与えることで、それと同じような領域の現れる映像の一部分を検索することが可能となり、その名に示されるように映像検索を実現している手法の一つである。実際に彼らは映画の映像に対してこの手法を適用しており、例えばあるシーンで現れるネクタイと同じネクタイが現れる別のシーンの検索を、高速に行った結果などが報告されている (図 4)。なんらかの物体が現れるシーンを検索するようなシステムは、素材映像の提供サービスの存在からも示されるように、一般的な需要はかなり高いものであり、Video

Google の目指すような画像クエリーによる検索は、重要なテーマであると考えられる。なお Video Google では、現在広く行われているテキスト検索の手法に則り、TFIDF を利用することで、元々ノイズの影響に弱い特徴点ベースの手法の欠点を補完しており、検索システムとしての完成度を高めている。また、特徴領域の抽出やそのトラッキングといった処理は非常に高い計算コストを必要とするものであるが、映像に対して一度行うだけで、あとはモデル間の距離計算へと問題を単純化することができる。膨大なデータ量となる映像を扱う上で、検索のような繰り返し行われるであろう処理にかかる計算量を減らすことは非常に重要であり、映像処理を一度だけに抑える方式は、映像を扱う上で基本となるものである。

図 4: Video Google における画像クエリーによる検索例

2.1.4 時空間特徴による情報抽出

前節で取りあげた Video Google は、画像解析手法に時間軸に関わる情報を取り入れることで、高度な映像把握を行うものであった。それに対し、初めから映像を時空間データとして捉え、そこから直接情報を抽出する手法がある。これらは、映像の中に何が映っているかという、オブジェクト指向の構造化とは異なり、映像の中で何が起きているかという、イベント単位の映像の構造化を目的として、しばしば採られるアプローチである。映像内のイベントの把握を行うためには、映像

内でどのような動きが生じているかを観察することが必要不可欠であり、そのためには画像領域だけでは情報として不足である。そこで時空間における時間軸方向のピクセル単位の変化を調べることが求められ、そのような目的に沿って、多くの映像処理手法が考案されてきている。

古くから、映像内に生ずる動きを検出することを目的とした手法が様々に研究されている。あるフレームにおけるあるピクセルが、次のフレームでどこに動いたかを求める Optical Flow の推定は、その中でも最も一般的なものであろう。しかし単純な Optical Flow の計算では、あまり正確な動きの推定が行えないため、限定された点の時空間内での動きを追うような、トラッキング手法は様々な形で提案されてきている [14, 15, 16]。さらには得られたピクセルに関する動きの情報を利用し、似たような動きを示すピクセルに関するグルーピングを行うことで、映像内のオブジェクトの抽出を行うような手法 [23, 24] などが、時空間のピクセルデータを直接解析する手法の典型的なものであるといえる。例えば Video Google でも、特徴領域の選別などの目的でトラッキングが利用されていたように、これらの手法はより高度な次元の映像解析を行う際にもしばしば利用されるものであり、一つの要素技術としての価値は非常に高い。

また、映像は 3 次元空間を撮影した結果として 2 次元平面状に投影されるものであるが、撮影対象である 3 次元空間そのものに関わる情報の抽出を目指す試みも、映像処理手法の一つとして広く行われてきた。Factorization や Mosaic と呼ばれるような手法 [71, 72, 73, 74, 75, 76, 77, 78] がそれにあたる。Factorization とは、映像の動きを利用することで、その映像に映される場面の 3 次元構成の復元を行うことを目指している。それに対し Mosaic では、同じ場面を異なるカメラで撮影したような複数の映像間で、ピクセル単位の実空間における対応付けを推定することで、それらの映像を合わせてより広い視野の映像を合成している。特に [71] では、映像のピクセル間の対応を、スケールや角度、さらにはフレームレートの異なるような映像についても推定するような手法が提案されており、得られた情報を応用することで、サブピクセル、サブフレーム単位の映像を合成することが、理論的に可能である。

これらの手法から、映像内の動きの情報が取得されることとなるが、それを利用することで、映像内で起きているイベントの把握を行うようなアプローチが近年盛んに行われている [40, 41, 42, 43, 44, 45, 46, 47, 49, 50]。ここでいうイベントとは、手法により独自に定義されるものもあるが、基本的には動きを持つような全ての事象を含む。特に人の行動に関するものが多く、人の動作をモデル化し、そのモデルにより異なる映像から同様の動作を検出する、といったアプローチが成されている。例えば Lindeberg ら [42] は、時空間特徴点を定義している。上で述べたように、画像から特徴点を抽出する手法は非常に広く利用されているが、彼らは時空間における特徴点を、画像からの特徴点抽出手法である Harris の手法を応用することで抽出している。この時空間特徴点は、画像における特徴点と同様に特徴ベクトルとして表現され、時空間内でのそれらの位置関係もしくは分布など

により、一つのイベントをモデル化することとなる。Lindeberg らは、人の歩行動作時の足の動きのような繰り返し動作のみに限定し、時空間特徴点によるモデル化を行っている。図 5 に示されるように、歩行時の人の足の部分から特徴点が検出され、クラスタリングにより同じ動きに対応する点は同じクラスに属する。その結果、いくつかのクラス分けされた特徴点の組が一つの動作に対応することとなり、これが歩行動作のモデルとなる。このモデルを利用することで、異なる映像内における歩行動作を検出することができる。また、このモデルは人物の外見やサイズなどに対して影響されない点で、歩行動作というイベントそのものをモデル化したものであるといえる。

図 5: 歩行動作から検出される時空間特徴点

また、Yao ら [43] は動作のような局所的な動きではなく、画面全体での大局的な動きに注目することによるイベントのモデル化を行っている。彼らの手法では、映像はカメラの動きとオブジェクトの動きにより構成されるものとされ、時空間内の全てのピクセルにおけるそれらの動きを表すパラメータを算出している (図 6)。カメラの動きはフレームに対してアフィン変換の効果を与えるものという前提により算出され、それ以外の動きをオブジェクトごとの動きとして捉えて、それぞれをパラメータで表現する。この計算をショットを通じて行った後に、算出されたパラメータの集合をヒストグラムとして表現することで映像のモデルとしている。特に放送映像などでは、同じジャンルの映像は同じような画面構成となることが多く、そのような特性を利用することで、映像分類が行われる。[43] ではテニスの映像の分類を行う実験が行われており、テニスのプレイが行われているシーンとそうでないシーンのヒストグラムを学習データとして与えることで、未分類の映像に関しても同様の分類が行われたとの結果が得られている。この手法の特徴は、映像にあたる時空間全体の特徴をモデルとして扱っている点にあるが、それ以外にも、同じイベントを捉えた映像は同じような画面構成であることが多いという、放送映像の特性を時空間において利用している点が注目される。

2.1.5 目的に特化した情報抽出

ここまでに取りあげてきた映像処理手法は、本質的にはあらゆる映像に対して汎用的に適用できるように設計されたものである。画像領域による物体のモデル

図 6: 映像内のカメラモーションとオブジェクトの動作の分類

化は、あらゆる物体に対して適用可能であり、また、時空間特徴量を利用することであらゆるイベントを表すことができる。しかし上で述べたように、映像からの情報抽出を実用に耐えうるような精度で行うのは非常に難しく、紹介したような手法をあらゆる対象に対してそのまま適用しただけでは、人間の感性からは程遠い結果となることが予想される。実際にこれまで取りあげた手法も、本質的にはあらゆる映像に対して汎用的に利用できる形態をとってはいるものの、Video Google では物体のバリエーションの限られる映画の映像の構造化を行うに留まっており、また、時空間におけるイベントのモデル化を行う手法も、繰り返し動作やスポーツ映像といった限られたイベントのモデル化を主として行っている。このように、あらゆる映像に含まれる物体やイベントを、全て同等に扱いモデル化を行うというのは極めて困難であり、なんらかの制約を加えることが必要となるのが実情である。また、処理を行うための計算量の抑制も重要な課題であり、モデルの汎用性が高くなるほどそれを表現するパラメータ空間の次元も高くなり、結果として計算時間の増大につながるという問題がある。

そこで、映像の扱い方を限定することで、より高精度な情報抽出を実現するようなアプローチが現実的に有効であり、そのような試みが多く成されている。[52, 53, 54, 55, 56] などは、人物、ロゴ、テロップ、といった領域を映像から抽出もしくは探索している。例えば放送局や番組を直接表現するようなロゴが付加される放送映像は数多くあり、また、ロゴのような領域は極めて特徴的なものであるために検出し易い。さらに、ロゴやテロップに含まれるテキストなどは、直接映像内容に関わるような情報を含むものであり、放送映像アーカイブのような大量の映像からの検出は、アーカイブの構造化のために大きな効果をもたらすことが期待される。また、中でも人物の顔を検出する手法は発達しており [51]、最近ではデジタルカメラにその機能が搭載されるなど、すでに一般的に利用されるアプリ

ケーションとして確立している。これはオブジェクト検出の一種であると言えるが、人間の顔という極めて特徴的な領域の探索に狙いを絞ることで、高精度な顔領域抽出を可能としている。

これらの手法は、検出する対象であるオブジェクトを限定しているために、検出されたオブジェクトの持つ意味合いが明確なものとなるという利点がある。例えば番組や企業に対応するロゴを発見できれば、その映像の内容把握につながることは明らかである。また、映像内における人物の存在とは、それ自体が強い意味合いを持つものである。そこで金出らは、映像から検出される顔と、その映像に対応するクローズドキャプションの情報を同時に利用することで、顔と名前の自動的な対応付けを行う Name-It システム [57] を作成している。このシステムでは、上でも触れたようなクローズドキャプションに対する自然言語処理と、映像に対する顔検出処理といったマルチモーダルなアプローチを行うことで、人物とその名前といった、映像内容に極めて近い観点からの構造化を行っていると言える。

このように顔検出という手法は、一般的なオブジェクト検出と比較して、検出する対象を絞ることで実用化に至る程の精度の向上を実現した。それに対し、映像の中に何が映っているかというオブジェクト指向の把握までは行わず、単に映像間の関連性を取得することを目指した手法として、映像照合というアプローチが存在する。映像照合とはその名の通り、何らかの基準により同一と判断される映像同士を照合することである。何をもって同一とするかの定義が重要であるが、これは照合を行う目的により様々なものがあり、これらに関しては後に詳細を述べることとする。ただし、基本的に同じものを撮影していることが基準となる。このような映像照合技術をアーカイブに対して適用することで、アーカイブに含まれる映像間の同一性が取得されることとなるが、これは特に放送映像アーカイブにおいては、構造化という点で非常に大きな効果を持つものである。例えば多くの TV 局において、一日に複数のニュース番組が放送されるが、それらが扱う出来事は大体似通ったものである。特に重要な出来事はほぼ全てのニュース番組で扱われることとなり、その結果、同じ場面を撮影した映像が複数の番組で流されることとなる (図 7)。つまり放送映像アーカイブには、同一の場面を撮影したような映像の組が大量に存在し、それらを発見することで、番組や TV 局の壁を越えて内容の関連性を取得することができる。このようなアプローチはトピック追跡などという言葉で呼ばれ、現在注目を集める研究テーマの一つであり、アーカイブの内容的な構造化につながるものである。また、何度も放送されているような映像は、それだけ重要度の高い内容であったり、視聴者が興味を引かれやすい映像であったりするもので、映像の頻出性を調べることで、映像の注目度・人気度といった観点からのランキングなどが可能となる。このように、映像照合を行うことによる、放送映像アーカイブの構造化に関する効果は非常に大きい。そこで本研究では、映像照合というテーマに絞った手法の提案を行うが、上述のように、映像照合手法に分類される手法においても同一性の基準は曖昧である。本研究で提案する手法は、同一性の定義を明確に示すものであり、他の手法と比較しての

位置づけに関して、以降で詳述するものとする。

図 7: 放送映像アーカイブにおける映像照合

2.2 映像照合

2.2.1 同一とされる映像とは

本研究がテーマとする同一場面映像照合は、映像照合という分野における技術の一つと位置づけることができる。上で述べたように、映像を照合するという試み自体は古くから行われているが、それらは手法としての特徴や、対象とする映像に関して多くのバリエーションを持つ。これらの違いは、主に映像照合を利用する目的と、その目的に沿うような、照合するための同一性の定義の違いから生ずるものである。本節ではそれらのバリエーションを取りあげ、本研究の目的とする同一場面映像照合の特徴と位置づけをまとめる。なお、本研究では、既存のほぼ全ての映像処理手法と同様に、映像の単位としてショットを利用する。ショット分割点を発見する技術が広く研究されているのは既に紹介した通りであり、また、ショットの分割点とは完全に客観的な情報であるため、映像製作側からその情報が付加されることも期待できる。よって、ショットを照合単位とするのは、本研究の対象とする放送映像に関しても適切であると考えられる。

映像を照合するための同一性の基準としては、以下のようなものがある。

1. 全く同じ映像

2. 基本的に同じ映像であるが、編集の違いなどによる差がある

3. 同じ場面を撮影している映像

4. 同種のイベントを撮影した映像

5. 同じオブジェクトを含む、もしくは同じ場所で撮影されている映像

図 8: 映像照合における同一性の種類

それぞれにおける照合対象となる映像の関係を図 8 に示す。(1) は、例えば CM のような、TV 局の違いなどにほとんど影響されない同一の映像のことをいう。CM に関しては、規定の回数放送されているかなどの調査といった目的で、映像照合手法が利用されることがある。宮武らは映像を単純なシーケンスで表現することにより、同一の映像の非常に高速な探索が行える技術を提案している [58]。フレーム全体でのピクセルの RGB 各成分の平均、フレームを半分にした片面でのピクセルの RB 成分の平均、そしてもう一方をさらに半分にしたときの RB 成分の平均、をそれぞれ 1 バイトの値で表現した合計 7 バイトのデータによりフレームを表す。この結果、連続するフレーム画像により構成される映像が、連続する 7 バイトの値のシーケンスとしてモデル化されることとなる。さらにこれらの値は閾値処理によりグルーピングされ、映像は最終的にはグループを示す識別子のシーケンスという形に高速に変換される (図 9)。このようなシーケンス間でのパターンマッチングを行うことで、映像照合が行われる。この手法においては、処理を高速に行うことが技術的に重視されており、映像のモデルであるシーケンスへの変換、シーケンス間の照合、が共に高速に行えるような形で定義されているため、映像の入力に対して即座に照合処理を開始できることが最大の特徴である。この他にも [59, 60, 62, 61] などで、映像を時系列データに変換することで高速な検索

や照合を行う方法が提案されており、処理の高速性が映像照合の重要な目標の一つであることは明らかである。

図 9: シーケンスによる映像の表現

これに対し、(5) のような同一性の定義に基づくと、上で紹介したような Video Google のような画像領域による映像のモデル化などが有効となる。また、(4) のような定義には、時空間特徴点によるアプローチなどが応用できるであろう。宮武らの手法は完全に同一な映像を探索することを主眼においているため、その内部に何があるか、内部で何が起きているか、といった内容は全く関知しない。そのため、時空間の構造が異なる映像の関連性を取得することは不可能であり、同一の物体が存在する場合も、外見が違う場合は同一でないとして扱われる。このように、何をもって同一と見なすかの定義により、求められる技術は大幅に異なる。また、そのような定義は、映像アーカイブの構造化をどのような側面から構造化するかという問題と深く関わる。そのため映像照合を行う上で、同一性の基準を明確に定義することが何より重要であり、また、そうすることにより性能評価を適切な形で行うことができる。

なお、本研究の提案する同一基準とは (3) に相当する。これは一般に Near Duplicate と呼ばれる (2) の定義にかなり近く、また、そのような映像の照合を行うことによる構造化の効果も似ている。しかし、実際はこれらの間には明確な定義の違いが存在し、それが本研究の提案する映像照合手法の新規性を意味することとなる。そこで、Near Duplicate と本研究の提案する同一場面映像照合に関する解説を行うことで、本研究の位置づけを示していくこととする。

2.2.2 Near Duplicate

Near Duplicate という表現は、画像もしくは映像に関して利用されるが、その定義は非常に曖昧である。単にコピー映像 (画像) という言葉で表現される場合もあり、基本的に同一の映像ソースからなるものとすることが多い。同一の映像に異なる編集が成されることで、映像データとしては差が生じている、そのような

映像の組は放送映像ではしばしば見られる。例えば海外から配信される資料映像などは、基本的に全てのTV局が同じ映像を利用することとなるが、編集により付加されるテロップなどは放送される番組により違ったものとなる。また、微妙に明るさや画質が異なることも多く、結果として同じ映像ソースからも異なるデータをもつ映像が作成され放送されている。このような映像の組がNear Duplicateと呼ばれるが、例えば[64]では、図10のような例をNear Duplicateの正解例としており、撮影するカメラの向きが異なるような映像もNear Duplicateとして扱っている。このように厳密な定義は手法ごとに多少のバリエーションを持ち、そのため、検出された映像の組が正解であるかどうかの評価基準も、それぞれの手法に依存することとなる。

図 10: Near Duplicateとされる画像例

Near Duplicateの照合は、現実の放送映像に対して強い適正を持つアプローチであると言える。それは編集などによる、多少の視覚的な差に関わらない照合を行える点にある。例えばニュース映像であれば、発言内容の字幕、トピックに関わる情報を表示するテロップ、時刻や天気の情報、などが画面のどこかしらに常に表示されているもので、それらの要素がもたらす番組間の視覚的な差は非常に大きいものとなる。そこで、完全に同一な映像データを照合するだけでなく、このような差を許容した照合を行うことで、現実の映像アーカイブにおける映像照合の可能性が格段に大きなものとなる。ではどのようにしてNear Duplicateを照合するかというと、大きく分けて二つのアプローチが存在する。それは上で述べた映像処理手法の分類と同様、キーフレームにおける照合を利用するものと[64, 67, 68, 69, 70]、時空間情報を利用した照合を行うもの[66]である。なお、前節で示したような、シーケンスのパターンマッチングによる映像照合手法は後者に分類される。

キーフレームによる Near Duplicate の照合は、Near Duplicate 画像の照合を応用することで可能である。Chang らは [64] において、画像から抽出される特徴点を利用し、多少の画像的な差異によらない照合法を提案している。画像が与えられた時、まず特徴点を検出し、全ての特徴点を接続するようなグラフを作成する。各特徴点は、色情報や Gabor フィルタによる特徴量などから構成される 11 次元の特徴ベクトルとして表現され、特徴点の間の関連性は、特徴点の位置関係にあたる座標の差により表現される。このように画像をグラフによりモデル化し、グラフ間の照合を行うことで、テロップの違いや、物体の位置関係の多少の違いなどに影響されない照合を実現している (図 11)。また、佐藤ら [63] は、カラーヒストグラムの相関による照合という比較的単純な手法を利用することで、数百〜数千時間単位の映像データからの高速なフレームの照合を行っている。例えば有名な人物や建築物の映像の中には、それらに関するトピックが扱われた際に、それらを象徴するものとしてたびたび流されるものがある。また、特に海外の出来事に関しては、その資料となる映像の種類が限られている場合が多く、この場合も同じ映像が繰り返し流される。このような映像の組は、疎にしか存在しないものであるが、非常に広い範囲からの探索を行うことで、映像内容の時期的な関連性の推定にまでつながるものである。彼らのアプローチは、大規模な映像アーカイブを前提としている点で特徴的であり、映像照合がアーカイブの構造化にもたらす効果を示す好例である。

図 11: グラフによる Near Duplicate 照合

これに対し、時空間情報を活用することで画像的な差を吸収するような映像照合を行っている手法というものは多くない。これは Near Duplicate の定義において画像的な類似性が重視される傾向にあるため、時間変化があまり意識されないことによる。上で取りあげたような時空間特徴点などを利用する手法は、例えば歩行動作のようなイベントの同一性に基づく照合ならば行えるが、それは Near Duplicate における同一性より広い意味合いのものである。よって、全く異なる映像内の同一イベントを照合するようにデザインされているこのような手法は Near Duplicate の照合には利用できない。その点 Boujemaa ら [66] は、時空間の特徴といえる特徴点の軌跡を利用することで、特徴点ベースの Near Duplicate 照合の精度を高める試みを行っている。彼らの手法は、基本的には特徴点の類似性に基づく典型的

な形式に則っているが、特徴点を表す特徴量に軌跡から得られる情報を取り入れている。抽出された特徴点はショットを通じてトラッキングされ、各フレームにおいてそれぞれ算出される特徴量の平均が採用される。また、その特徴点がどの範囲に存在したかを空間・時間の両方の側面から求める。これらの軌跡から得られる情報により、特徴量はより強い弁別能力を持つこととなる。

このように、Near Duplicate の照合は、主に画像もしくは時空間の構造に関する類似性を求めることで、実現されている。データセットとして TRECVID のニュース映像が利用されることも多く、照合結果を利用することでのトピック追跡を行うなど、アーカイブ構造化を意識していることがうかがえる。

2.2.3 同一場面映像照合

本研究は映像アーカイブの構造化に向けてのアプローチとして、映像照合を行う。映像照合のもたらす構造化の効果に関しては、上で述べたとおりである。映像照合という手法の中にも様々なものがあり、それらの例を紹介してきた。これらを踏まえ、本研究ではこれまでのアプローチとは異なる新たな映像照合の方式を提案し、それを同一場面映像照合と呼ぶ。同一場面映像とは、その名の通り同一の場面を撮影した映像のことをいう。つまり、ある被写体を中心とする一つの場面に関して、それを撮影している全てのカメラから得られる映像が同一場面映像として扱われる。また、そのような映像にテロップの付加などの編集を施しても、それはやはり同一場面映像となる。このような定義は Near Duplicate の定義と比較して、同一基準の厳密さに関して改善されており、同一か否かの判定を明確に行える。

Near Duplicate との違いは、照合されるべき映像の性質にも現れる。Near Duplicate は手法により定義が曖昧であるとはいえ、コピー映像検出などという言葉で表現されることもあるように、基本的に視覚的な類似性がその照合基準となる。それに対し同一場面映像照合では、視覚的に類似しているかどうかは全く関係ない。映像に映される場面が同じであれば、視覚的に大きく異なっていても同一と見なす。そのため照合すべき映像の組が、画像的に大きく異なるものであることも多く (図 12)、これらに対して画像的な類似性を求めるような手法を適用することはできない。よって、キーフレームの解析というアプローチは、同一場面映像照合には適していないといえる。また、図 12 に示される例は、カメラの角度に大きな違いがある。さらには異なるテロップが追加され、画面の全体的な明るさも大きく異なる。このような映像の組を画像的な類似性から照合するのはそもそも困難であると考えられ、何らかの時間変化に関する情報の利用が必要不可欠である。

また、場面に関する同一性は、Near Duplicate のそれと比較してかなり広義のものとなるが、完全な包含関係にあるわけではない。Near Duplicate 照合の手法においては、例えば同じ景色を異なる時間に撮ったような映像も照合対象となるような定義が成されることがある。また、画像的な類似性に従った照合を行って

(a) (b)

図 12: 同一場面映像の例

いるため、似たような光景の映像ならば、異なる場面であっても照合されてしまう。このような照合が正解であるか不正解であるのかは、明確な区分けが成されていない。その点同一場面映像照合では、場面の同一性という定義が成されているため、視覚的に類似する異なる場面を撮影した映像を照合するのは明らかに誤りである。このような基準による評価は、Near Duplicate の照合などでは行われてこなかったものであるが、同一場面映像照合を行う上で重要な問題となる。例えば図 13 に示される映像の組は、同じ場所で同じ人物の会見を撮影した映像であるが、撮影された日時は異なり、もちろんその内容も違う。つまり、「似ずして同一の場面」を照合すると同時に、「似て非なる場面」を区別することが同一場面映像照合には求められ、このような点に関して新規性を持つ。

ここで示した映像例は、既存の手法では処理しきれない可能性が高い。しかし、実際に放送映像にはこの例のような映像の組が多く、放送映像アーカイブの構造化へとつなげるためには、これらの照合を適切に行う必要があると考えられる。そのため、本研究の行う同一場面映像照合では、このような映像間の照合を適切に行うことを目標としている。

また、映像アーカイブでの照合を行うために欠かせない条件として、計算量の問題がある。映像照合というアプローチは、上で述べたように放送映像アーカイブに適用されることで大きな意味を持つ性質の手法であるため、大量の映像が存在する場合を想定し、それらの間での照合処理を十分に短い時間で行えるように

図 13: 類似した非同一場面映像の例

設計されなければならない。既存の多くの映像処理手法は、その精度に関する評価と比べ計算量に関してはあまり記述が成されていない場合が多く、実際に短い時間で映像からの情報抽出を行うというのは難しい。それに対して映像照合を高速に行うようなアプローチを紹介したが、これらは映像が基本的に同一であることを前提としたもので、例えば異なるカメラで撮影したような映像の照合はそもそも想定していない。そのため、本研究は上で述べたような照合に関わる問題に加え、計算量に関する問題を同時に解くことを目標とする。

2.3 手法概要

同一場面映像照合に関する議論で触れたように、本研究は二つの目的を同時に満たす必要がある。照合に関しては、キーフレームを主体とする処理では困難であると考えられるため、ショットを時空間として捉え、そこから得られる時間的な情報を利用する必要がある。しかし、時空間データそのものを利用するのは、計算量的に好ましくない。そこで本手法では、ショットを比較的単純な時系列のデータとして表現することで、これらの問題に対処することとした。具体的には、一つのショットを、各フレームごとに一つの値を持つような数値のシーケンス (もしくはそのようなシーケンスの集合) という形式のモデルに変換し、そのシーケンス間での照合を行う方式で、宮武らの開発したシステム [58] に非常に近いアプローチである。このような方式を適用することで、映像アーカイブから図 14 に図示されるような形で、同一場面映像は照合される。ここで示されるショットの特徴量であるシーケンスの抽出はオフラインで行われ、その結果得られる特徴量はアーカイブ内に保存される。その後、抽出された特徴量空間での比較を行うことで、同一場面映像を取得するという手順である。

数値シーケンスを元とした映像照合を行う上で解決する必要のある問題は、主に以下の 3 つである。

1. シーケンスをいかに抽出するか

図 14: 映像アーカイブの拡張

2. シーケンスをいかに比較するか

3. ショット間のオフセットにいかに対応するか

（1）は照合手法の基本となる問題である。ここで抽出するシーケンスは、同一場面を撮影したショットのみを確実に照合できるようにデザインされたものでなければならない。本研究では、このために二通りの手法を提案するが、ともに画像的な特徴量の時間変化を表現するような値を利用している。時間変化に基づくモデルとすることで、視覚的な差に影響されない形での照合を行える可能性がある。また、時間変化を直接照合基準とすることで、「似て非なる場面」が照合されてしまうという問題は解決される。これは既存の画像的な類似度による映像照合手法と大きく異なる点であるといえる。（2）は精度や計算量といった、手法の性能に関わる問題である。単に数値シーケンスを比較する手法としては、編集距離や相関といったものが存在するが、ここではシーケンスの持つ特性に合うような調整が必要となる。また、そのような特性を踏まえつつ、計算量を抑えることが重要である。本研究で扱う二つの照合手法では、ともに数値のシーケンスの比較を行っているが、それらは各々のシーケンスの特性に合わせるために、異なる形式で行われている。最後に（3）は、同一場面を撮影した映像でも、実際に放送される時間的な区間は番組によって異なるという、放送映像の特性に対応するためのものである。ショットを表す特徴量としてのシーケンスという観点から言えば、同一場面映像であってもその始点と終点が一致する保証は無く、適切な映像間の時間的な差＝オフセットを求める必要があるということである (図 15)。

これらの問題を解決するために、本研究は二通りの手法を提案する。3 章ではフラッシュのたかれるショットを照合するための、フラッシュ発生パターンによる照合手法を取り上げ、続いて 4 章では、映像内のオブジェクトの動きによる照合手法を述べる。また、それぞれの手法を実装した上で行った実験の結果により、各手法の評価を行う。

図 15: 同一場面映像間の時間的なオフセット

3　フラッシュパターンによる照合

3.1　概要

本章で提案する手法は、カメラのフラッシュのたかれるような映像を対象とし、そのフラッシュ光の発生するパターンを数値のシーケンスとすることで、そのような映像間の照合を行うものである。フラッシュのたかれるような映像は、特にニュース番組などにおいて頻繁に現れるものであり、内容的にも放送時期を代表するようなものである場合が多い。そのためこのような映像照合は、トピック追跡や人気映像の抽出といった同一場面映像照合の目的に沿ったものであると考えられる。

フラッシュという現象を照合基準としたのには、大きく分けて二つの理由が挙げられる。その一つ目は、映像からフラッシュのたかれるフレームを検出することが、比較的容易であると考えられるためである。フラッシュ光のたかれるタイミングでは、被写体の周囲が急激に明るくなるため、その影響はフレーム全体の輝度の急激な変化として現れる。これは多少のカメラの向きの差異や、比較的小さいテロップの存在などにはほとんど影響されることはない。そのため、基本的にはフレームの輝度に注目するだけで、フラッシュ発生フレームを検出することが可能となる。また、ショット間の照合を行うような特徴量の特性が、二つ目の理由となる。フラッシュ光の発生パターンは、フレームごとに発生したか否かを示す二値のパターン、すなわち 0 か 1 で構成される数値シーケンスとして表現することが可能であり、これをショットを表現する特徴量とすることができる。図 16 に示されるように、フラッシュのたかれるフレームは明らかに周囲のフレームより明るく、かつ、異なるカメラで撮られていてもこのシーケンスは共通する。二値のシーケンスは、ショット特徴量としてアーカイブに保存する上で、そのデータ量の少なさは大きな利点であり、なおかつパターンマッチングを高速で行えることが期待できる。このため、前章で取りあげたような、同一場面映像照合手法に求められる要件を、十分に満たしていると言える。

図 16: フラッシュの発生を示す二値のシーケンスとその共通性

図 14 に本研究の目標とするシステムの概要を図示した。これをフラッシュパターンによる照合手法に当てはめるならば、ショットを示す特徴量はフラッシュの発生パターンということとなる (図 17)。デジタル映像において、ショットは連続するフレーム画像により構成されるもので、そのフレームごとにフラッシュが発

生したか否かを示す二値のシーケンスが、特徴量として扱われる。つまり、フラッシュがたかれたと判断されるフレームを 1 で表し、それ以外のフレームを 0 で表すことで、一つのショットの特徴量としている。これらのシーケンスが類似しているかどうかを調べることにより、ショット間の照合を行うのがこの手法の基本的な概念である。

図 17: フラッシュパターンによる照合の流れ

このようなシステムを実現するためには、フラッシュパターンを抽出する手法、つまりショットの中でフラッシュの発生していると考えられるフレームを検出する、いわばフラッシュ検出手法と、抽出された二値パターン同士を比較する、パターン比較の手法とが求められる。それらの詳細に関し、以下の節で解説を行う。

3.2 フラッシュ検出

まずは与えられた映像の全フレームにおいて、全てのピクセルの輝度の平均値を求める。そしてこの値が急激に変化しているフレームを、フラッシュがたかれたフレームの候補とする。フラッシュは一瞬のうちに光が照射されてすぐ消えるという現象であるため、フラッシュがたかれたフレームの前後での輝度の変化は、急激に値が跳ね上がった後すぐに元に戻る、という特徴的な変化を示す (図 18)。

そこで、ショット間の平均輝度値の変化を追跡し、前後 2 フレーム (1 フレームで輝度が完全に上がりきらない、もしくは下がりきらない可能性を考慮) と比較して急激に輝度が高くなっているフレームを探す。あるショットにおいて、i 番目のフレームの全ピクセルの平均輝度値を h_i とする。このとき平均輝度値に関する、フラッシュ発生を判断する閾値を θ_1 とすると、式 (1) に示される両方の条件を満たす場合に i 番目のフレームをフラッシュ発生フレームの候補とする。

$$\begin{aligned} &\max(h_i - h_{i-1}, h_i - h_{i-2}) > \theta_1. \\ &\max(h_i - h_{i+1}, h_i - h_{i+2}) > \theta_1. \end{aligned} \tag{1}$$

図 18: フラッシュのたかれるショットにおける平均輝度値の変化の例

この場合の 2 フレームという限定は、今回扱った 30 フレーム/秒のフレームレートを持つ MPEG 圧縮された映像についての観測結果による数値である。本稿では、以後も 30 フレーム/秒の MPEG 映像を利用していることを前提として話を進める。θ_1 で示される閾値を適当に調整することで、映像内のフラッシュをほぼ取りこぼし無く、かつ高い精度で検出できることが確認されている。ただ、そこには誤検出も少なからず存在し、その例を図 19 に示す。

図 19 のシーン (a) では蛍光灯がちらついているために輝度の変化が生じ、フラッシュとして検出されてしまっている。(b) の例は雪がちらついているために、フレームの平均輝度値が上下してフラッシュと誤認されたもの、(c) は炎のゆらめきが輝度値を大きく変化させてしまったものである。

誤検出が真のフラッシュと同じ発生パターンをとることはないという前提があるため、このような誤検出もパターンの照合結果に影響しないはずだが、図 19 の (a) のような例では輝度変化のパターンが非常に規則的であるために同種の誤検出間での照合が行われてしまうことがある。また、フラッシュの発生パターン同士の比較はショット数の二乗のオーダーで計算量に影響するため、比較するパターンの数は少ないほどいい。このように、精度・計算速度の両方の側面から輝度の変化のみによる検出で生じる誤検出を取り除く、つまり真にフラッシュであるかどうかの判定を行う必要がある。そのような処理をフラッシュ判定と呼び、3.3 で二つのフラッシュ判定の手法を述べる。

(a) (b) (c)

図 19: フラッシュとして誤検出されたフレームとその前のフレーム

3.3 フラッシュ判定

3.3.1 輝度差分画像の類似性による判定

図 19 の (b) や (c) のような誤検出を取り除くため、検出されたフラッシュ発生フレームの候補を画像的な観点から検査する。この処理は以下の前提に基づいて行う。

1. フラッシュ発生フレームの候補とそれに隣接する前後のフレーム間のオブジェクトの動きは、無視できるほど小さく、かつ局所的である。

2. カメラのパンやズームも非常に小さな動きとしてのみ観測されるが、フレーム全体に影響を及ぼす点でその影響は大きいといえる (図 20)。

3. 画面全体にちらつく雪や炎、特殊な光源 (パトカーのライトなど) は隣接フレーム間のような短い間にも大きくフレーム画像を変化させる (図 20。)。

（1）と（3）から、フラッシュ発生の候補となるフレームの前後間でのフレーム画像の変化を調べ、フレーム画像の変化が大きい場合はフラッシュではないと考えることで、図 19 の (b) や (c) に示されるような誤検出を取り除くことが可能である。ただし、そのような判定を行うためには（2）が問題となりうるため、まずはこれを取り除くために、候補となるフレームの前後のフレーム間での Optical Flow の推定を式 (2)、式 (3) により行う。

$$I_t\left(x - V_x(x,y), y - V_y(x,y)\right) = I_{t+1}(x,y). \tag{2}$$

図 20: フラッシュの近辺でズームされている例

$$\begin{pmatrix} \sum F_x^2 & \sum F_x F_y \\ \sum F_x F_y & \sum F_y^2 \end{pmatrix} \begin{pmatrix} V_x \\ V_y \end{pmatrix} = \begin{pmatrix} \sum -F_x F_t \\ \sum -F_y F_t \end{pmatrix}. \tag{3}$$

F_x、F_y は画像のそれぞれ xy 方向で隣り合うピクセル間の輝度値の差分、F_t は隣り合うフレームの対応するピクセル間の輝度値の差分を示す。V_x、V_y が各ピクセルの xy 方向それぞれの移動した距離となる。また、画像のガウシアンピラミッドを作成し、その上位のレベルの推定結果を下位に反映することで、比較的大きな動きに対しても対応できる。ただし、この手法は輝度の大きな変化が無いフレーム間で行うことを前提としているため、輝度がピークとなるフラッシュ発生候補のフレームに関しては適用できず、そのためその前後のフレーム間での推定を行っている。ピークとなるフレームの前後のカメラモーションが一定であるとすると、得られた Optical Flow の半分の数値で前後のフレームを補正できる。この推定は、フラッシュ前→後とフラッシュ後→前の二通りを行い、それぞれについて補正値の半分を、ピークのフレームに対しての Optical Flow$(V^-(x,y), V^+(x,y))$ として取得する。

次に推定した Optical Flow$(V_x(x,y), V_y(x,y))$ に従って、対応するピクセル間でピークの前後の各フレームに関して輝度差分画像を作成する。ショットの i 番目のフレームの前後のフレームに関して、i 番目のフレームに合わせて補正したフレーム画像を $I'_{i-1}(x,y), I'_{i+1}(x,y)$ とする。

$$\begin{aligned} I'_{i-1}(x,y) &= (I_i(x - V_x^-(x,y)), I_i(y - V_y^-(x,y))). \\ I'_{i+1}(x,y) &= (I_i(x + V_x^+(x,y)), I_i(y + V_y^+(x,y))). \end{aligned} \tag{4}$$

この場合、輝度差分画像 J_{i1}, J_{i2} は、$J_{i1}(x,y) = I_i(x,y) - I'_{i-1}(x,y)$、$J_{i2}(x,y) = I_i(x,y) - I'_{i+1}(x,y)$ として求められる。フラッシュの他の光源の変化が無いものと仮定すれば、この輝度差分画像のピクセルの輝度値は全てフラッシュ光の影響によるものと考えることができる。そこで J_{i1}, J_{i2} それぞれの全てのピクセルに関し

て、その輝度値が適当な閾値 θ_2 より大きい $(=1)$ か小さい $(=0)$ かにより二値化した画像 J^b_{i1}, J^b_{i2} を算出する。この J^b_{i1}, J^b_{i2} に関する類似度を定義し、それを真にフラッシュが発生したか否かを示す基準とする。これらの画像の例を図 21 に示す。

図 21: 隣接する輝度差分画像間の類似性による判定

二値化した輝度差分画像 J^b_{i1}, J^b_{i2} の類似性から判定を行う。仮に、輝度差分画像の J_{i1}, J_{i2} に関するフレームの全ピクセルの平均輝度値 h_{i1}, h_{i2} が $h_{i1} < h_{i2}$ であるとする (逆の場合は、J_{i1} と J_{i2} を入れ替えればよい)。この場合、$J^b_{i1}(x,y) = 1$ であれば、それよりフレーム間の輝度の変化の大きい J^b_{i2} において、$J^b_{i2}(x,y) = 1$ も満たされるはずである。この条件がどの程度満たされるかにより、フラッシュ判定を行う。手順は以下の通りである。

1. 二値化輝度差分画像の全ての対応するピクセルに関して、$J^b_{i1}(x,y) = 1$ かつ $J^b_{i2}(x,y) = 0$ となるピクセルの総数 e_i を求める。これはすなわち両画像間のエラーに相当する。

2. 同様に、$J^b_{i1}(x,y) = 1$、もしくは $J^b_{i2}(x,y) = 1$ となるピクセルの総数 N_i を求める。これは、少なくとも片方の輝度差分画像でフラッシュ光の影響が確認されたピクセル数となる。

3. フラッシュ光の影響が確認されたピクセルの内のエラーの割合 $d_{img} = e_i/N_i$ を求める。これを二値化輝度差分画像間の距離として考え、この判定における評価値とする。

この評価値 d_{img} が実際にフラッシュがたかれたかの判定基準となる。値が小さいほどフラッシュ現象に適合すると考えられる。なお、実際はピクセル単位ではなく、適当な大きさの矩形領域単位の対応を調べることで optical flow の推定の際に生じるノイズの影響を抑えている。

この結果として、フラッシュ発生の候補となる全てのフレームが評価値 d_{img} を持つこととなる。次にショット内の全ての d_{img} の平均値を算出し、ショット単位でのフラッシュ判定を行うための評価値とする。フラッシュの発生したフレームの中には、例えばカメラが非常に速く動いた、フラッシュ光が非常に密にたかれたために 1 フレームでは輝度が落ちきらない、などの理由により Optical Flow の推定などが適切に行われないようなものも稀に存在する。ショットに関する平均値による判定を行うことでこのようなノイズが吸収され、精度向上につながる。なおこのような処理では、実際のフラッシュとフラッシュと誤検出されるような現象が混在するショットを扱えないが、実際の TV 放送映像において、例えば記者会見映像などは安定した照明条件下で撮影される場合がほとんどであり、ショット単位での判定は有効といえる。

3.3.2 輝度変化の発生パターンによる判定

図 19 の (a) のような例は全体に当たる光がちらつくものであり、画像的な変化という点からはフラッシュと酷似した現象である。逆に発生パターンという点では非常に規則的なパターンを示す傾向にあり、このような誤検出同士でパターンが照合されがちである。

フラッシュとは人手による現象であり、その発生パターンは蛍光灯のちらつきのような規則的なパターンとは異なるのが普通である。そこでフラッシュの発生はポアソン到着に従うと仮定し、観測されたパターンのフラッシュ発生間隔の分布とポアソン到着における到着間隔の分布との類似度を求めて検査する。なお、フラッシュはあるタイミングに集中する傾向があり、それ以外の区間での発生は非常に疎である。そのため密にフラッシュがたかれる区間のみに注目してモデルとなる確率分布を算出する。ポアソン到着において、時間間隔 τ に関する到着間隔の分布 $g(\tau)$ は式 (5) で表される。

$$g(\tau) = \lambda e^{-\lambda\tau}. \tag{5}$$

λ は単位時間当たりの平均到着数、つまり平均フラッシュ発生数を示す。λ は各ショットで実際に観測されたフラッシュ候補フレーム数を、ショット内のフラッシュの存在区間のフレーム長で割ることで求め、利用する。

算出した確率分布との距離 d_{occ} を以下のように定義する。f_i は長さが i フレームであるフラッシュ発生間隔の回数であり、すなわちフラッシュ発生間隔のヒストグラムに相当する。n はフラッシュ発生間隔の総数 (=フラッシュ発生数-1、ショット

の両端は間隔としては扱わないため)、p_i はポアソン到着モデルから算出される、長さが i フレームであるフラッシュ発生間隔の生ずる理論確率を示す。フレームレートを r とすると、$p_i = g(i/r)$ となる。つまり、ここで定義される距離は観測値と理論値の相対誤差の累計である。

$$d_{occ} = \sum \frac{(f_i - np_i)^2}{np_i}. \tag{6}$$

式 (6) の d_{occ} が輝度変化の発生パターンによる判定の基準値であり、これが小さいほどポアソン分布に近い、つまりフラッシュとして考えても良いということとなる。なお、この距離は χ^2 検定に使われるものであるが、この場合はサンプル数が非常に少ないために通常 χ^2 検定において利用される理論的適合基準に基づく閾値は適用できない。そのため単に分布間の距離として扱うこととする。

この処理の結果として除かれるパターンは上記の規則的なものの他、フラッシュの発生頻度が高すぎて放送映像のフレームレートでは正確なパターンとならないシーンがある。このようなシーンは稀に存在し、結果的に常に (1 フレームおきに) フラッシュが発生しているパターンに近い形で検出される。よって、照合という点からも今回対象とする映像ではこのようなシーンは適切には扱えず、その他の誤検出と共に捨てることとなる。

3.4 パターン比較

3.4.1 パターン間類似度

検出されたフラッシュパターン間の類似度を定義する。ここで定義される類似度が、ある閾値を越えるショット同士を同一場面映像として検出する。扱うショット数が増えることでパターン間比較を行う回数も増大するために、低計算量のアルゴリズムでなければならない。また、フラッシュの発生パターンは、撮影したカメラの違いにより 1 フレーム程度の誤差が生じることがあり、それは許容する。さらに、番組によって同じシーンでも放送される範囲は異なるため、そのような時間的なオフセットの差に対応する必要がある。

フラッシュパターンの比較は、基本的にはオフセットごとのパターンの類似度計算の繰り返しにより行われる。あるオフセットにおける類似度は、二つのパターンの重複部分から計算される。そして重複部分のフレーム長が、ある閾値 θ_{len} 以上となるような範囲 $(-(n_2 - \theta_{len}) < \tau < n_1 - \theta_{len})$ でオフセットを 1 フレームずつずらしていき、その都度類似度の計算を行う。最後に、全てのオフセットから得られた類似度の内、最大のものを選び出す、といった流れである。

各オフセット τ における類似度の定義を以下に示す。また、図 22 にオフセットを変化させつつ類似度を求める処理の流れを示す。

1. 片方のパターン t_2 を τ だけずらした状態で t_1 と並べる。

2. そのオフセットにおける、両パターンの重複部分に注目する。このとき、片方のパターンの各々のフラッシュ発生フレームに関して、もう一方のパターンの対応する、もしくは隣接するフレームが、同様にフラッシュ発生フレームであるかを調べる。もしフラッシュ発生フレームであれば、その隣接関係に応じたスコアを加算する (s_τ)。

3. 比較範囲内に存在する両方のパターンのフラッシュの総数 (M_τ) に対する合計スコアの割合をもって、そのオフセットにおけるパターン間の類似度とする ($sim_\tau = s_\tau / M_\tau$)。

4. オフセット τ を 1 ずつ増やしつつ、その都度類似度 sim_τ を求める。

5. 全てのオフセットに関する最大の類似度を、パターン間の類似度として定義する ($sim_{12} = \max(sim_\tau)$)。

図 22: パターンの類似度の算出

フラッシュ発生フレームがマッチしたとされた際に加算するスコアとしては、一致するフレームに存在する場合は 1、隣接するフレームに存在する場合は 0.75 とした。これらは両パターンのフラッシュ発生フレームそれぞれについて与えられ、結果としてはこの 2 倍の値がスコアとして加算されることとなる。このように定義することで、ショット間の類似度として計算される s_τ は 0〜1.0 の範囲に収まることとなる。

なお、このような処理を全てのオフセットに関して行うと、計算量が膨大になるという問題がある。また、パターンの比較は重複部分のみにおいて行われるが、もしその範囲に存在するフラッシュ発生フレームの数が少ない場合は適切な検出が行えない。そこで、これらの問題に対処するために、比較範囲内に最低限存在する必要のあるフラッシュ発生フレームの個数の閾値 (θ_{num}) を設定した。パターン内のある区間に、いくつのフラッシュ発生フレームが存在するかは高速に求める

ことができる。一つのパターンに関して、先頭フレームからその後の各フレームまでに存在するフラッシュ発生フレームの累積数を表すシーケンスを別に作成し、比較範囲の終点と始点でその差分を求めればよい。そして、いずれかのパターンにおける、比較範囲内におけるフラッシュ数が θ_{num} 未満ならば、そのオフセットでの適切な類似計算は行えないとして、類似度を 0、つまりマッチしないものとする。また、両パターンにおけるフラッシュ数から、取り得る最大の類似度を求めることも可能である。比較範囲内における二つのパターンのフラッシュ発生数をそれぞれ n_1 と n_2 とする。また、$n_1 > n_2$ であるとする。この時、2 つ目のショットの全てのフラッシュ発生フレームに関して、1 つ目のショットの対応するフレームがフラッシュ発生フレームだった時、類似度は最大となる。つまり、$(2n_2)/(n_1 + n_2)$ の値が、そのオフセットにおいて取り得る最大の類似度ということになり、これがショットを照合するためのパターンの類似度の閾値より低い場合、そのオフセットで照合されることはありえない。そのため、そのオフセットにおいて、フレーム間の対応を調べる類似度計算を行う必要がなくなり、計算量を抑えることができる。これらの方法により、パターンがマッチしうるオフセットのみで、フレームごとの対応を調べるような類似度計算を行えばいいということになり、結果的に処理にかかる総時間を減らすことができる。

これらの比較手法において利用されるパラメータとして、フレーム数の閾値 θ_{len} として 150 フレーム、フラッシュ数の閾値 θ_{num} として 5 回、という値を採用した。フラッシュが連続するフレームから検出されることはなく、150 フレームから隣り合わない 5 フレームを選び出す組み合わせは膨大に存在するため、ポアソン分布との比較で適合するもののみに限定しても、パターンの一意性を十分に示すことができると考えられ、これは後述する実験における、結果の精度の高さとして証明される。また、フラッシュ発生フレームがマッチしたとされた際に加算するスコアとしては、一致するフレームに存在する場合は 1、隣接するフレームに存在する場合は 0.75 とした。これらは両パターンのフラッシュ発生フレームそれぞれについて与えられ、結果としてはこの 2 倍の値がスコアとして加算されることとなる。このように定義することで、ショット間の類似度として計算される sim_τ は 0 〜1.0 の範囲に収まることとなる。

3.4.2 高速化のためのアルゴリズム

前節で示したパターン比較手法は、比較的単純なアルゴリズムを利用しているために高速な処理が期待されるが、実際はオフセットごとの反復処理を含むためにかなりの計算時間を必要とする。フラッシュ検出手法と違い、パターンの比較は繰り返し行われる必要があるため、その計算速度の向上は特に重要な課題である。そこで、パターン比較処理の高速化のためのアルゴリズムを導入する。

処理の高速化のため、無駄な類似度計算を省くことを考える。ある 2 つのパターンが与えられた時、それらが全く類似しないものであるならば、そのパターン間

の類似度計算を行わずに、照合し得ないパターン対として処理するのが効率的である。ここではそのための手法に関してまとめる。実際にそのようなパターン対の照合処理のスキップを行うためには、各パターンをより高速に比較できるような別の特徴量に変換する必要がある。そのような目的のため、ここではフラッシュの発生間隔に注目した。ある二つのパターンが照合される場合、それらは共通するフラッシュ発生間隔を持つはずである。比較範囲のフラッシュ発生数が $\theta_{num} = 5$ の以上の場合についてのみ照合を行うため、発生間隔にすると 5-1=4 つの共通する発生間隔を持つこととなる (図 23(a))。次に、単一の発生間隔ではなく、連続する発生間隔の組の共通性を考える。2 つの連続するフラッシュ発生間隔の組に関して、5 つ以上のフラッシュが発生しているパターン同士が照合されるためには、少なくとも 5-2=3 組について共通しなければならない (図 23(b))。

図 23: フラッシュ発生間隔

このような考えに基づいて実験した結果から、発生間隔の組は 3 次元が適当であると推定された。そこで連続する 3 つのフラッシュ発生間隔を 3 次元ベクトルとして表現する。

図 24: フレッシュ発生間隔ベクトル

なお、フラッシュ発生間隔ベクトルは、図 24 に図示されるように、一つのフラッシュから 4 通り算出される。これはフラッシュ発生フレームの未検出に対応するためである。フラッシュの検出手法は recall という点で非常に高い精度を示したが、万が一取りこぼしがあった場合に備えてこのような処理を行っている。ショット内の全てのフラッシュ発生フレームからこれらのベクトルを抽出し、ベクトル空間における、それぞれのベクトルが対応する点の出現数を合計することで、そのショットを表す隣接フラッシュ間隔ヒストグラムが作成される。放送映像では 1 つのショットは数秒〜10 数秒で構成されることが多く、その中に発生するフラッシュ数は 20〜30 程度に過ぎない。そのため、このヒストグラムは非常に疎なものであり、異なるフラッシュ発生パターンを持つショット間では、ヒストグラム間での重複はほとんど存在しないと考えられる。そこで、2 つのショットを比較する際に、まずはこれらのヒストグラム間の比較を行う。ヒストグラム間の重複分を調べ、上述の理由により 2(=5-3) 以上の重複が存在すれば、照合される可能性があるため、前節で述べたように類似度を計算する。逆に 2 未満であれば、照合される可能性はないものとして、類似度の計算を行わずに 0 とする。

ヒストグラム間の重複度を調べる際に、フラッシュが 1 フレーム程度ずれて検出された場合に備え、ϵ 以下の距離にある点は重複するものとして扱う。しかしヒストグラム空間において、ϵ だけの距離の差を許した重複度計算は、比較的大きな計算量を必要とする。高速化のための処理に計算時間をかけるのは本末転倒である。そこで、ヒストグラム空間での点の拡張を行うことによる、近似的な重複度計算を実装することで効率化を図っている。全ての隣接フラッシュ間隔ヒストグラムは、生成されるベクトルに対応する点を追加する際に、その周囲 ϵ の距離にある点に関しても、同様に頻度をインクリメントすることで生成される。そしてヒストグラム間の比較をする際には、片方のパターンから生成されるベクトルに関して、もう一方のパターンの ϵ だけの拡張が成されたヒストグラムとの重複度を求める。このような近似を行うと、一つのベクトルを元に拡張されたヒストグラム上の値が、複数のベクトルと重複するとされることがあり、結果として重複度が余分に追加され、false alarm が生じる可能性がある。しかし、前述のように、このようなヒストグラムは非常に疎なものである場合が多く、また、ここでパターン同士の比較をこの段階で完全に行う必要はないため、高速化の効果を大きく損ねるものではない。実際に、この処理により多くのパターン対に関する類似度計算の省略が行われることが、後に示す実験結果により明らかになる。これらの実験では、ヒストグラム空間点の間の距離は L1 距離により求められ、$\epsilon = 2$ を用いて判定を行っている。

3.5 実験による評価

利用する TV 放送映像は、MPEG1 形式で 1 時間ごとに分割されたファイルで保存されている。その中からニュース番組に相当する 80 時間の映像を対象とした

実験を行い、各閾値などのパラメータの設定を行った。3.5.1 では、表 1 で示したパラメータ設定用の 80 時間分のニュース番組映像 (NHK、日本テレビ、TBS、フジテレビ、テレビ朝日の主に夕方、夜のニュース番組) に対してフラッシュの検出と判定を行い、3.3.1 や 3.3.2 で述べた評価値 d_{img}, d_{occ} それぞれに関して、フラッシュと判定できる閾値を決める。そしてその閾値を利用して、別の性能評価用の 80 時間分のニュース番組映像データセットに同様の処理を行い、その精度から利用した閾値の汎用性を確認する。なお、精度を求めるために両方のデータセットに関して、ショットがフラッシュのたかれるシーンを映したものであるか否かを示す Ground Truth を事前に確認し、利用している。

用途	**総時間**	**放送時期**
パラメータ設定用	3 時間	2002/12/10
	15 時間	2003/4/15-2003/4/18
	62 時間	2003/12/11-2003/12/13
性能評価用	80 時間	2003/12/13-2003/12/16

表 1: 実験に利用したデータセット

また、全ての実験には 1 台の PC(Pentium4 2.8GHz, Windows XP) を使用した。全てのコードは C++で書かれており、GCC によりコンパイルしたものを利用している。

まずは、これらの映像を各フレーム画像の特徴量によりショットごとに分割しつつ、フラッシュの候補となるフレームをフレームの平均輝度値から選び出す。ショット分割は、隣接フレームのカラーヒストグラムの差を閾値で区切ることにより行う。ただし、フラッシュはショット分割点と誤認されやすく、ショット分割とフラッシュ検出を平行して行う必要がある。フラッシュの「輝度が跳ね上がった後瞬時に元に戻る」という性質は、ショット分割点ではほとんど見られない特徴であるため区別できる。なお、このような方法では、ワイプやディゾルブなどの画面効果により接続されるショットの分割を適切に行えない。このため、本来同一シーンを撮影したものであるショット対が検出されなくなる可能性がある。しかし、ショット分割手法については本研究の対象外であり、正解・不正解の判定もこの手法で分割されたショットを正しく分割されたショットであるとして行っているために、性能評価を行う上ではほとんど問題は無い。

次に 3.2 の方法で、パラメータ設定用データセットからフラッシュの検出を行ったところ、フラッシュが発生したとされるフレームは約 30000 であった。5 回以上のフラッシュが 150 フレーム以上の区間にわたって存在するショットのみに限定するとその数は 928、それらのショットの内実際にフラッシュがたかれていると確認されたショットは 709 であった。なお、ここでは閾値を低めに抑えているためにフラッシュを取りこぼすことはほぼ無い。よってここで検出されたショットをフラッ

シュがたかれるショットの全てであるとして評価基準とする。また、この 709 のショットの合計フレーム数は約 30 万フレーム、つまり 2 時間 46 分であり、これは映像全体の約 3.5%である。

3.5.1 フラッシュ検出

検出されたフレームに関して、輝度差分画像の解析によるフラッシュの判定を行った。ここではフラッシュの候補となるフレームに関して、それぞれ 3.3.1 で定義される評価値 (d_{img}) が求まり、ショットごとにその平均値を算出して判定を行う。図 25 の横軸がその平均値、縦軸がショット数を示す。

図 25: 画像解析によるフラッシュ判定の評価値 (ショット単位) の分布

図 25 のヒストグラムにおいて、閾値 (θ_{img}) を 0.25 とすると 120 以上の誤検出されたショットが捨てられ、全体でショット数は 758、recall は 6.1%低下して 93.9%、precision は 11.5%向上して 87.9%となる。

次にフラッシュの発生パターンに関する判定の結果を図 26 に示す。横軸は 3.3.2 の式 (6) により定義されるポアソン到着との距離 (d_{occ})、縦軸はその距離を持つショットの数を示すヒストグラムである。実際に大きな d_{occ} を持つショットの大部分は、例えば蛍光灯が点滅しているような誤検出であることが確認された。逆に、実際にフラッシュがたかれているショットのパターンの d_{occ} は比較的小さい。図 26 では両者間のギャップが明確に現れているといえる。

ここで閾値 (θ_{occ}) を 80.0 とすると、ショット数は 730、recall は 0.8%低下して

図 26: フラッシュ発生パターンに関する評価値の分布

93.1%、精度は2.5%向上して90.4%となる。これらの判定を行うことによるprecision-recallの変化を表2に示す。

検出法	recall	precision
輝度差分のみ (3.2)	100%	76.4%
輝度差分画像の類似性 (3.3.1)	93.9%	87.9%
ポアソン到着との類似性 (3.3.2)	93.1%	90.4%

表 2: 各フラッシュ判定後の precision-recall

上で示した実験結果から、$\theta_{img} = 0.25$、$\theta_{occ} = 80$ という値を採用し、これらを別の異なる80時間分のニュース映像により構成される評価用データセットに適用した場合のprecision-recallの変化を、表2と同様に表3に示す。評価用データセットには、全部で478のフラッシュが発生するショットが含まれており、輝度差分のみによるフラッシュ検出では689のショットが検出された。

このデータセットでは、動きの激しいフラッシュシーンなどのエラーを引き起こし易いショットが多く含まれていたためかprecision、recall共に数%低下しているが、フラッシュ判定処理による効果は依然大きく、採用した各パラメータの値の汎用性を示すものと考えられる。また、これらのフラッシュ判定処理は、そのものの精度に加え3.5.2に示されるようにパターンの比較の精度の向上に大きく貢献するものである。さらには、誤検出同士の比較という無意味な処理を省けるこ

検出法	recall	precision
輝度差分のみ (3.2)	100%	61.2%
輝度差分画像の類似性 (3.3.1)	88.5%	84.6%
ポアソン到着との類似性 (3.3.2)	88.3%	88.3%

表 3: 評価用データセットにおける各フラッシュ判定後の precision-recall

とによる計算時間の大幅な短縮も効果が大きい。

3.5.2 パターン比較

評価用データセットに対するフラッシュ検出、そしてフラッシュ判定の結果として残されたショットに関して、全ての組み合わせにおけるフラッシュパターン同士の類似度のヒストグラムを図 27 に示す。横軸が類似度、縦軸がその範囲に含まれるパターンの対の数である。なお、0.65 以上の類似度を持つものに関しては人手で真偽を確認したが、それ以下のショット対は非常に数が多いため、ランダムに 100 のショット対を選び出し、それらについて同一シーンのショット対が含まれていないことを確認した上で全て FALSE とした。

図 27: パターンの類似度の分布

図 27 に見られるように、類似度 0.7〜0.9 あたりがギャップとなっており、類似度 1、つまりほぼ完全に一致するパターンに至ってパターン数は増加している。よっ

て、偶然似通ったフラッシュの発生パターンが生ずることはほとんど無く、逆に一致するショットのパターンは撮影環境の違いによらず一定であることが読み取れる。類似度の閾値を 0.8 に設定することで、約 90%の精度で同一シーンのショットを取得できる。この時検出された同一シーンを撮影したショットの組は全部で 154 である。

また、パターン比較の結果から、フラッシュ判定処理の効果を評価する。特に発生パターンによる判定で除くべき誤検出はそれら同士で照合されやすく、最終的な出力の精度を著しく損なう可能性がある。フラッシュ判定を行った場合と行わない場合の照合精度の変化を図 28 に示す。横軸が同一シーンと見なす類似度の閾値、縦軸が検出結果の内、実際に同一シーンのショット対であった精度を示す。このグラフにはフラッシュ判定による精度の向上効果が明らかに示されている。

図 28: フラッシュ判定の有無による精度の変化

このような精度の向上に加え、比較するパターンの数の減少により、同一パターンを検出するために必要となる計算時間も大幅に軽減される。80 時間分の性能評価用データセットには、総時間にして 2 時間程度、総数にして 422 のフラッシュ発生ショットが含まれていたが、フラッシュ判定処理により選別されたショットは、422 のフラッシュ発生ショットに誤検出のショットである 56 を加えた合計 478 であった。それらの間からの同一場面映像の検出という作業は、ショット数の二乗のオーダーの計算量を必要とするため、パターン比較を行うショットの組み合わせの総数は 30%以上減ることとなり、これに応じて計算時間も短縮される。また、TV

放送映像の特性として、フラッシュがたかれるショットとは短く編集されていることが多いため、誤検出されたショットはそれらに比べて長いパターンとなることが多い。よって、実際には、50%近い計算時間の短縮がこのデータセットにおいて確認されており、計算時間の面におけるフラッシュ判定の効果も大きいことが証明される結果となった。

次に、3.4.1 に示される、取り得る最大類似度を求めることにより照合し得ないオフセットでの類似度計算をスキップする処理と、3.4.2 に示される、マッチし得ないパターン対における類似度計算のスキップを行う高速化アルゴリズムの効果を表 4 にまとめる。この結果から、これらのアルゴリズムが大幅な計算時間の短縮をもたらしていることは明らかである。結果として、二つのフラッシュ発生ショット (平均 15 秒) の比較を行うために必要な時間は約 1.5[ms] であり、また、パターン対、つまりショット間の照合処理をスキップする効果も含めると、この時間は約 0.26[ms] にまで減少する。この結果は、映像アーカイブへの適用を考えても十分な速度での処理が行えたことを十分に示していると考えられる。

	パターン対	オフセット	計算時間 (分:秒)
高速化無し	114,003	29,790,376	06:00
パターン対スキップのみ	19,543	7,706,744	01:36
パターン対＋オフセットスキップ	19,543	2,162,427	00:30

表 4: 高速化アルゴリズムの効果

最後に検出されたショットの組の例を図 29 に示す。

図 29: 検出されたショットの例

これらはそれぞれショットの構成やテロップ、映像の質などは異なっているが、この時期の主要なニュースでありそれぞれ複数の TV 局で何度も放映されている

ショットである。視聴者にとってより興味深い映像が抽出されたと言えるだろう。なお照合結果には、図 29 にあるような明らかにカメラの異なる映像に加え、単にテロップが異なるだけといったほぼ同一の映像の組が多く含まれている。今回扱ったデータセットに含まれる映像は純粋なニュース映像であるため、例えば芸能ニュースなどの局ごとの差が強く出る内容の映像が少なく、また、世界的に重要なニュースが世間を賑わせていたことからほぼ同じ映像が繰り返し使われていた、などがその理由である。

4　動きの変化パターンによる照合

4.1　概要

前章で提案したフラッシュ発生パターンによる同一場面映像照合は、計算速度と精度に関しては非常に良い結果を示した。しかしこの手法は、フラッシュがたかれるようなショットの照合にしか利用できないという、映像アーカイブに適用する上での汎用性の低さが問題点として残されている。そこで、本章では、より幅広い映像に対して適用できるような手法として、映像内の物体の動きの共通性による同一場面映像照合のアプローチを提案する。

物体の動きに注目するとはいえ、映像を時空間として捉えた時、そこでの物体の動きは当然カメラの角度などの影響を受ける。つまり、物体の動きを単なる時空間における幾何学的な形状として捉え、それに対する比較を行うだけでは、撮影するカメラが違うような同一場面映像の照合を行うことは到底不可能である。そこで、動きを示すような時空間内の絶対的な位置座標を利用するのではなく、動きの向きや速度の変化の発生に注目し、そのような変化の発生パターンによる照合を行うというのが基本的な概念となる。向きや速度の絶対値はカメラの向きに依存するが、それらがどのように変化しているかという傾向に関しては、カメラの違いによらず、ある程度共通したものがあると考えられる。物体の動きが観察される映像の数は、フラッシュのたかれる映像に比較すればかなり多いと考えられ、その点で汎用性は大きく向上している。また、動きとはすなわち時空間情報であるため、そこに含まれる情報は静止画と比較して豊富であり、強い照合能力を持つものと考えられる。なお、動きによる照合を行う上で、一つのショット内でのカメラの動きは存在しない、さらにはカメラの向きの極端な差は存在しない、という二つの前提を設けた。このような撮影条件に関する大きな差は、動きの傾向の大きな変化をもたらすため、その共通性を見出すことも困難となる。しかし、実際に放送映像において、ひっきりなしにカメラが動いているようなショットはほとんど存在せず、また、被写体は主に正面から撮影されるものであり、例えば真下や真後ろから撮影されることなどはほぼありえない。このような放送映像の特性を考えると、これらの前提は、放送映像に関する一般性を大きく損なうものではないと考えられる。

ただし、時空間的な情報を扱うためには、時空間を効率的にモデル化し、比較できるような特徴量を抽出することが、映像アーカイブにおける照合という目的を果たすためには必要不可欠である。そのため、本手法では特徴点の軌跡をもって、ショットに対応する時空間のモデルとすることとした。特徴点の軌跡を利用するというアプローチ自体は、[66, 48] などが利用しているが、これらは軌跡自体を照合対象とするわけではなく、軌跡を利用することによる、別の特徴量の改善を目的としていた。それに対して、ショットを表すような特徴量として、軌跡を直接利用しているのが本手法の特徴である。軌跡を利用するために必要な、特徴点抽出と

それらをトラッキングすることによる軌跡の抽出といった手法は、古くから映像処理手法として研究されてきているものであり、ある程度の成果が現在までにもたらされている。よって、そのような既存の手法を利用することで、ショットのモデル化を行うこととする。また、特徴点の軌跡は、ショットの各フレームに対応するような、時空間点のシーケンスとして表されるものである。そのためこれらの時空間点を、その点の周囲における動きの特徴を表すような数値で表現することで、2.3 で述べたような数値シーケンスとして表現することができる。このようなアプローチを採ることで、最終的にショット同士を数値のシーケンス間のパターンマッチングにより高速に照合するというのが、本手法の概要である。

これらの処理を実現するために、以下の二つの処理が必要となる。

- ショットからの特徴点の軌跡の検出と、特徴量シーケンスへの変換
- 複数の軌跡間での比較による、ショット間の類似性の推定

フラッシュ発生パターンによる照合と同様に図 14 に当てはめるなら、前者が特徴量の抽出処理に当たり、後者は特徴量間の比較を指す。なお、この手法におけるショットの特徴量は、特徴点の軌跡を表現するシーケンスが集められたものであり、一つのショットが一つのシーケンスに対応した前章の手法とは、この点で大きく異なっている (図 30)。つまり、ここではショット間の類似性を算出するために、2 つのシーケンスの比較ではなく、多対多のシーケンス比較を行う必要がある。そのため、フラッシュ発生パターンによる照合に比べると、どのシーケンス同士が対応するかといった対応付けの問題など、かなり複雑化している。しかし、軌跡を表すシーケンス自体は、ショットの持つフレーム数分の長さの数値配列であることには変わりはなく、その保存性や計算速度などに関する同一場面映像照合に求められる要件を、依然として満たすものであるといえる。

図 30: 特徴点の動きパターンによる照合の流れ

以下の節では、特徴点の軌跡をどのように検出し、そして比較に利用するかに関しての詳述を行う。

4.2 動きの検出とパターン生成

全てのショットは、トラッキングされた特徴点の時空間における軌跡の組として表される。ここでは、そのようなショットを表す特徴量を生成する方法に関して述べる。

最初に特徴点の軌跡を得るために、与えられたショットからの特徴点の検出と、そのトラッキングを行う。ここでは KLT トラッカーと呼ばれる、特徴点の検出とトラッキングを同時に行う手法 [15] を利用する。この手法は古くから様々な映像処理手法に利用されているもので、映像を入力することで、追跡すべき特徴点の抽出と、それらに対するトラッキングを自動的に行うことができる。なお、特徴点を抽出する際に、実空間において対応する点を抽出するのが理想的だが、必ずしも完全な対応が必要では無い点が本手法の一つの特徴である。後に示すように、ショット間の比較は統計的処理により行われるもので、大体同じような動きを示す点が抽出されれば照合を行うことが可能である。そのため、対応する特徴点が抽出されるかどうかという点に関しては、それほど神経質になる必要は無く、正確なトラッキングが行われることが重要である。KLT トラッカーは、パラメータとして特徴点の個数を必要とするが、実験においては 200 個の特徴点を追跡させている。また、物体の移動やオクルージョンなどの理由により追跡不能となった特徴点の数が増えた場合に備え、特徴点数が 100 個以下になった場合は新たな点を抽出させるようにしている。結果として、一つのショットから複数の特徴点の軌跡が抽出されることとなる。抽出された軌跡のうち、存在区間の短すぎるものはこの段階で捨てられることとなる。

次にそれぞれの軌跡は、inconsistency として [40] に定義される、時空間画像内の点の周囲のピクセル値の動きの均一性を示す値のフレームごとのシーケンスとして表現される (図 31)。この inconsistency は、例えば一定方向への等速度運動のような、均一な動きしか含まない時空間領域においては低い値となり、逆に、物体の動きの向きや速度が大きく変化するような点を含む時空間領域において高い値を示す、という性質がある。つまり、各軌跡において、特徴的な変化を示している点が高い inconsistency の値として表現されることになり、この値は 4.1 で述べたような照合の基準として有用なものであると考えられる。

inconsistency は、時空間における全てのピクセルに関して定義される値であり、時空間の全ての方向での輝度の差分を元として算出される値である。ある時空間点での x, y, t それぞれの方向に関する輝度差分を $\nabla P = (P_x, P_y, P_t)$ とし、これをその周囲の小さな時空間領域内の全ての点に関して合計することで、行列 $\boldsymbol{M}$ を求める。

$$\boldsymbol{M} = \begin{pmatrix} \sum P_x^2 & \sum P_x P_y & \sum P_x P_t \\ \sum P_y P_x & \sum P_y^2 & \sum P_y P_t \\ \sum P_t P_x & \sum P_t P_y & \sum P_t^2 \end{pmatrix}. \tag{7}$$

図 31: 軌跡の inconsistency によるショットのモデル化

この時、もし領域内の全てのピクセルが、時間軸方向に一定の動きを示しているならば、$\boldsymbol{M}$ のランクは 2 以下になる。そこで、この行列の左上の 2×2 部分のみを抽出した、つまり時間成分を取り除いた行列 $\boldsymbol{M}^{\diamond}$ とのランクの差を求め (式 (8))、そのパッチにおける動きの不均一性として定義している。このランクの差は、$\boldsymbol{M}$ と $\boldsymbol{M}^{\diamond}$ の固有値の計算により、0〜1 の間の連続的な実数として近似される。$\boldsymbol{M}$ の固有値を $\lambda_1 \geq \lambda_2 \geq \lambda_3$ とし、$\boldsymbol{M}^{\diamond}$ の固有値を $\lambda_1^{\diamond} \geq \lambda_1^{\diamond}$ とすると、式 (9) により Δr は実数として得られる。なお、これらの詳細については [40] に記述されている。結果として、このような行列計算により、時空間点の周囲の時空間領域における動きの不均一性がスカラーとして表現されることとなり、本手法はこの値を利用することとする。

$$\Delta r = rank(\boldsymbol{M}) - rank(\boldsymbol{M}^{\diamond}). \tag{8}$$

$$\Delta r = \frac{\lambda_2 \cdot \lambda_3}{\lambda_1^{\diamond} \cdot \lambda_2^{\diamond}}. \tag{9}$$

なお、この inconsistency は、元々は時空間内の全てのピクセルから算出され、時空間における分布具合から、映像間の比較を行うように考案されたものである。それに対し本手法では、この値を利用してはいるものの、時空間全体からではなく、検出された特徴点の軌跡上の点のみを利用している。この理由を以下に示す。

- 時空間内の全てのピクセルにおける inconsistency を保持するのはデータ量的に膨大になるため好ましくない。また、全時空間からの inconsistency の算出は非常に大きな計算量を要する。

- 高い inconsistency で示されるような特徴的な動きの発生は、全時空間中で見ると非常に稀なものである。そして、そのような特徴的な動きは、物体の動作に沿って現れるものであるため、特徴点の軌跡にそのような動きが生ずる可能性は極めて高い。

- 二つのショットが同一場面であるかの比較を行う際、軌跡単位での比較は時空間画像全体での比較に比べて、高速に行うことが可能である。この比較処理は、ショット数に応じて繰り返し行われるものであるため、計算速度は非常に重要な問題である。

- 軌跡とは、場面内の物体のある一点に関する、時空間における動きを捉えたものである。つまり、軌跡単位の比較はオブジェクト単位の動作の比較に近く、同一場面の照合性能を強める効果がある。

以上の理由により、特徴点の軌跡に注目した解析を行うことで、inconsistency を効率的にも精度的にも元の手法とは異なる形で有効に利用していると考えられる。ただし、この inconsistency はカメラの向きやズームの度合いなどの違いに影響されて変動する値であり、同一場面内の対応する点であってもその値が同一である保証は無い。もともと、時空間内における点の周囲の不均一性の強さの傾向を示す値として定義されたものであり、物理的な定量性をもった値とはいえないのがその理由である。この問題は、ショット間の照合を行う際に解決する必要があり、4.3 で扱う。inconsistency は、軌跡上の全ての点の周囲の小さな時空間領域 (7x7x3) から算出され、その結果として各軌跡は軌跡上の各点に対応する inconsistency のシーケンスとして表現されることとなる。実際のショットにおける、ある特徴点に関する inconsistency の変化を図 32 に示す。これは右に示される二つの同一場面映像の中から、実際に対応している点 (円により示される、右耳部分から検出された特徴点) の inconsistency のフレームごとの変化を示したものである。対応する点において、inconsistency の変化の傾向はよい類似を示すことが読み取れる。

図 32: 同一場面を撮影したショット間の inconsistency

次に、このシーケンスから、inconsistency がある程度以上大きく、かつ周囲の値と比較した際に極大値をとるフレームを求める。本手法では、特徴点の動きの向きや速度の大きな変化の発生パターンの共通性が照合の鍵である。inconsistency の

シーケンスにおける激しい値の上昇は、そのような動きの大きな変化に対応していると考えられ、特に inconsistency のピークとなるフレームを中心として、その軌跡の周囲で何か特徴的な事象が起きているものと考えられる。本稿では、このような特徴的な inconsistency の変化を「イベント」という言葉で呼ぶこととする。これは前章で取りあげたフラッシュ発生パターンの照合におけるフラッシュに対応するもので、本手法の inconsistency は、前章の手法ではフレームの平均輝度値に対応する要素である。このようなイベントの定義に基づき、イベント発生フレームを inconsistency の変化から求める。フラッシュの場合は、平均輝度値に及ぼす変化がほぼ 1 フレームに収まる、非常に急激なものであったため、フレーム間の平均輝度値の差を調べることで適切な検出が行えたが、inconsistency のイベント周囲における変化は、それに比較してゆるやかである。また、inconsistency の値はノイズの影響によりかなりばらつくことが確認されている。そこで、ここではガウスフィルタで平滑化した inconsistency のシーケンス ($cons(i)$) に対し、適当な幅の窓による極大値を探索することで、イベント発生フレームを検出する。つまり、あるフレーム i の inconsistency($cons(i)$) が、前後 l_w フレームの全ての inconsistency より高い値である場合 ($\forall j \leq l_w, cons(i) > cons(i \pm j)$)、このフレームをイベント発生フレームと考える。ショットを通してイベント発生フレームを検出した結果を、各フレームが 0 か 1 で表される二値のシーケンスであるイベントフレーム発生パターン ($ev(i)$) として表現する。もし i 番目のフレームがイベント発生フレームならば $ev(i) = 1$、そうでなければ $ev(i) = 0$ となる。

これらの処理の結果、各軌跡は平滑化された inconsistency のシーケンス ($cons(i)$: inconsistency パターン) と、そのピークであるイベント発生フレームの存在を示す二値の配列 ($ev(i)$:イベント発生パターン) により表現されることとなる。各々のショットは、このような軌跡の組としてモデル化され、それぞれの軌跡の持つ二つの配列は適当な形式で保存されることとなる。これらの配列は、共に長さは軌跡の存在したフレーム数であり、データサイズは非常に小さく抑えられるために、ショットのモデルとして保存することは十分に可能である。

4.3 ショット間の比較

それぞれのショットは、軌跡の抽出と inconsistency による軌跡の表現といった処理により、数値のシーケンスである inconsistency パターンと、二値のシーケンスであるイベント発生パターンの二種類のシーケンスの集合によりモデル化された。ここでは、それらのモデル間の照合を行う手法についての解説を行う。

4.3.1 軌跡間の類似度

はじめに、二種類のシーケンスにより表現される軌跡同士がどれくらい似たものであるかを示す類似度の計算法について述べる。軌跡を表現するシーケンスと

しては、二値のシーケンスであるイベント発生パターンが存在し、この比較はフラッシュ発生パターンの比較手法に順ずる形で行えるように見える。しかし、フラッシュという現象が平均輝度値に対して極めて明確な変化を与えるのに対し、inconsistency パターンにおけるイベントの発生は比較的ゆるやかなものであり、フラッシュほどフレーム単位での厳密性があるとは言えない。また、イベント発生フレームを求めるために、inconsistency の極大値を探索しているが、そもそも極大値を求めるという処理自体がノイズに弱い。そのためイベント発生パターンのみからの、軌跡間の類似度計算は不適切である。よって、類似度計算は inconsistency パターン同士に対して行う。イベント発生パターンは、類似度計算を行う上で補助的に利用する。

また、軌跡間の比較を行う上で、時間のオフセットは既に与えられているものとする。後述するように、ショット間の類似度を計算する際、オフセットを変化させつつ、その都度類似度計算を繰り返し行っており、軌跡間の類似度計算は、そのループの内部の一つの計算処理として扱われる。そのため、軌跡間を比較する時点で、双方の軌跡間のオフセットは既に決められている。よってここで述べる軌跡間の比較とは、あるオフセットが与えられた状態で、二つの軌跡を表す二種類のシーケンスの重複部分のみを抽出し、その部分における類似度を算出することをいう。なお、特徴点の軌跡はショットの途中で消失することもあり、また途中から生ずることもあるため、ショット単位では十分に長く重複するようなオフセットでも、軌跡単位での重複部分は短くなることがありうる。そこで、二つの軌跡の重複部分が一定の閾値 (θ_{len}) より短い場合や、重複部分に存在するイベント発生フレームの数がやはり閾値 (θ_{num}) より少ない場合は、これらの軌跡の比較は適切には行えないものとして、類似度を 0 とする。これらの閾値の扱いは、フラッシュ発生パターンを照合する際にも利用されたもので、本手法においても同様の値を設定して ($\theta_{len} = 150$,$\theta_{num} = 5$) 処理を行っている。

軌跡間の重複部分の長さ、そこに存在するイベント発生フレーム数に関する両方の条件が満たされた場合の、類似度を算出する手順を以下に示す。ここでは、パターンの重複部分を抽出して、その区間のみについて計算を行う。また、比較するショットをそれぞれショット 1(s_1)、ショット 2(s_2) とし、それぞれに含まれる軌跡を p_{1i}, p_{2j} と表記する。ショット間のオフセットは τ で表す。

1. それぞれの軌跡のイベント発生パターン間の類似度を、3.4.1 で述べた手法で求める。なお、オフセットは与えられているため、変化させる必要は無い。

2. イベント発生パターンの類似度がある閾値 (θ_{ev}) より低ければ、この軌跡は照合されえないとして類似度を 0 として処理を終了する。これは類似度計算時間を減らすための手順であり、厳密なふるい分けを行うものではないため、閾値は低めの値に設定する。現在の実装においては、$\theta_{ev} = 0.5$ としている。

3. 両軌跡に存在する、全てのイベント発生フレーム (f_k) に関して、その前後

N フレームでの二つの inconsistency パターンの相関係数 ($C(f_k)$) を算出する (式 (10)〜式 (12))。

4. 求められた全ての相関係数の平均値を求め、それを与えられたオフセットにおける軌跡間の類似度として定義する (式 (13))。

$$\phi(t) = cons_1(t) - \sum_{\theta=-N}^{N} (cons_1(f_k+\theta))/(2N+1). \tag{10}$$

$$\psi(t) = cons_2(t) - \sum_{\theta=-N}^{N} (cons_2(f_k+\theta))/(2N+1). \tag{11}$$

$$C(f_k) = \frac{\sum_{\theta=-N}^{N} \phi(f_k+\theta)\psi(f_k+\theta)}{\sqrt{\sum_{\theta=-N}^{N} \phi(f_k+\theta)^2 \sum_{\theta=-N}^{N} \psi(f_k+\theta)^2}}. \tag{12}$$

$$sim(p_{1i}, p_{2j}, \tau) = avg_k(C(f_k)). \tag{13}$$

この類似度計算処理において、(3) では inconsistency パターン全体の相関ではなく、イベント発生フレーム周囲のみの相関を算出し、その平均を求めている。これは inconsistency の値そのものが、物理現象を定量的に表現するような値ではないという特性を持っているため、例えば背景の煩雑さが変わることで値の絶対値が変化するという特性に対処するためである。例えば図 32 で示した例でも、変化の具合は非常によく似ているが、値そのものが完全に一致してるわけではない。また、どちらかのショットの軌跡の inconsistency が常にもう一方の軌跡より高くなっている、といったような統一性も無い。そのため、異なる時点の inconsistency を同等に扱うことは適切ではないと考えられ、(3) のように、イベント毎に分割して相関性を調べている。なお、N の値としては 14 を実験結果から採用した。30 フレーム/秒の映像を扱っているため、片側 14 フレームとはすなわち周囲約 1 秒に相当する値である。

4.3.2 ショット間の類似度

次に、軌跡間類似度を利用して、ショット間の類似度を求める手法を述べる。ショットは複数の軌跡により表されるため、その類似度は多対多の比較を行うことで得られる。このため、どの軌跡同士が対応するのかという対応付け問題、多くの軌跡の組から算出される多くの軌跡間類似度をどのようにまとめるかという統計処理、といった問題を解く必要がある。このような問題に対して、本手法は以下の二つの前提に従って解法を考案した。

- 実空間において対応するような動きを捉えている場合、軌跡間の類似度は高くなる。

- 同一場面であっても、カメラや編集の違いなどにより、対応する軌跡を持たない軌跡も存在しうる。

前者は本手法の前提ともいえるもので、対応付けの問題に関係する。つまり、軌跡間類似度の高い軌跡の組が、実空間的にも対応するものとして考える。後者はカメラの違いなどを吸収するために、軌跡間類似度をショット単位でまとめるときに考慮しなければならない点である。対応する軌跡を持たないような同一場面映像の例を図 33 に示す。この例では、人物の左耳にあたる特徴点は、カメラの角度の違いにより左側のショットには存在しない。また、ネクタイなどに対応する点は、左側のショットでは字幕に隠されてしまっている。

図 33: 片方のショットのみに現れる特徴点の例

これらを踏まえ、軌跡の組としてモデル化されたショット間の類似度は、以下のような手順で算出される。

1. 与えられた 2 ショット (s_1,s_2) の間の時間的なオフセット (τ) を決める。なお、オフセットの取り得る範囲は、そのオフセットにおける 2 つのショットの重複部分が、ショット長の閾値 (θ_{len}) より長くなる範囲に相当する。

2. ショット間での全ての軌跡の組み合わせ (p_{1i},p_{2j}) に関して、決められたオフセットを与えた上で、軌跡の持つ二種類のパターンから重複する範囲における類似度 $(sim(p_{1i}, p_{2j}, \tau))$ を上記の手法に従って算出する。

3. 片方のショット (s_1) の全ての軌跡に関して、それを含む全ての組み合わせから算出された類似度の内から最大のものを選び、その軌跡ともう一方のショット (s_2) に含まれる軌跡の組との類似度 $(sim(p_{1i}, s_2, \tau) = \max_j(sim(p_{1i}, p_{2j}, \tau)))$ とする。

4. （3）で算出された類似度の内、高い方からある一定の割合 ($\rho\%$) に含まれるもののみを選び出し、それらの平均値を求める。この値をショット 1 から見たショット 2 との類似度 ($sim(s_1, s_2, \tau) = avg_{i \in top\rho\%}(sim(p_{1i}, s_2, \tau))$) とする。

5. 逆向きの類似度 ($sim(s_2, s_1, \tau)$) を同様に算出し、これら 2 つの値の平均値を求める。この値を、（1）で決められたオフセットにおける 2 つのショット間の類似度 ($sim_{12}(\tau) = (sim(s_1, s_2, \tau) + sim(s_2, s_1, \tau))/2$) として定義する。

6. オフセットを 1 フレームずつ変化させて（2）から（5）までの処理を繰り返し、その中の最大値をショット間の類似度 ($sim_{12} = \max_\tau(sim_{12}(\tau))$) とする。

この中で、(3) が軌跡間の対応付け問題にあたる。ここでは最大の類似度を持つ軌跡を対応する軌跡と考え、それらの間の類似度を利用している。また、(4) では片方のショットの全ての軌跡に関する、もう一方のショットとの類似度の内、上位 $\rho\%$ のみの値の平均を利用している。これは対応する軌跡を持たない軌跡の存在に対応するために行う処理である。つまり、全ての軌跡に関する類似度を平均すると、対応する軌跡を持たない、すなわち低い類似度しか持ちえない軌跡の影響で、そのような軌跡を含むショットから見たもう一方のショットとの類似度が下がるという問題がある。そこで、ある一定の割合の軌跡がよい一致を示していれば照合を行えるように、上位 $\rho\%$ に限定した平均値を求めている。$\rho\%$ の値としては、やはり実験結果から 50%を採用している。また、(3) と（4）の計算は、どちらのショットの軌跡を基準として最大類似度を選ぶかという点に関して非対称である。(5) で両方向での平均値を求めているが、これによりその非対称性は取り除かれる。

以上の手順で得られた値をもって 2 つのショット間の類似度と定義し、この値がある閾値以上ならば、それらを同一場面映像であるとして検出する。

4.4 実験による評価

4.4.1 映像照合性能の評価

以上の手法を現実の映像アーカイブ内の映像に適用し、その性能評価を行った。3.5 で利用したものと同じアーカイブに保存される、5 つの放送局の TV 放送映像の一部をデータセットとして利用する。まずは、本手法の照合精度に関する評価のため、人手により選び出された同一場面映像を 26 ショット、似たような画面構成の非同一場面映像を 9 ショット、アーカイブ内からランダムに選ばれた映像を 15 ショット、の合計 50 ショットに関して、全てのショットの組み合わせでの類似度を計算する実験を行った。映像照合を行うにあたって、$\theta_{num}(= 5)$ 以上のイベント発生フレームを持つ軌跡の個数が 20 未満のショットは、照合には不適切であると考

え取り除いた。この閾値処理により、24 ショットの非同一場面映像から 8 ショットが取り除かれ、全部で 42 ショットをデータセットとして利用した。類似度計算による照合は、これらのショットの全ての組み合わせである、861 のショット対について行う。なお、同一場面映像に関わる 26 ショットの内、4 ショットは全て同じ場面に対応しており、17 のショット対が同一場面映像の組として、データセットに存在することとなる。

図 34: ショット間類似度の分布

図 34 は、861 のショット対から算出された類似度の値の分布を示す。この結果において、類似度の閾値を 0.7 とすると 7 の映像対が照合結果として得られ、これは全て同一場面映像である。また、類似度の閾値を 0.6 とすると 12 の映像対が得られ、これらの内の 9 の映像対が同一場面映像となる。図 35 に、このデータセットから、高い類似度をもって検出された同一場面映像対の例を示す。これらの検出結果から、映像の色情報が大きく変わっていたり、もしくはカメラのズームによりスケールが変化していたとしても、そのような同一場面映像間での照合を行えることが読み取れる。

なお、この実験の結果として、照合結果の precision は高い反面、照合できない同一場面映像が残る、つまり recall が低くなる問題が確認された。それらの例を図 36 に示す。

このような問題は、主に以下の二つの理由により生じている。

- カメラの角度が大きく異なる
- 十分に動きの観察される軌跡の量が少ない

図 35: 高い類似度を示した同一場面映像例

図 36: 検出されなかった同一場面映像の例

カメラの角度の大きな差に関して、例えば図 36 の上の例では人物を左下からとほぼ真下から撮影しており、その差の影響により、これらのショットを適切に照合することはできなかった。それに対し、後者はトラッキングによる軌跡の抽出に関わる問題である。これは特に、特徴的な動きを示す画像領域が小さく、かつ、特徴的な模様を持つ背景や、テロップが多く存在する場合に生ずる問題である。KLT トラッカーにより抽出される特徴点は、そのような背景やテロップに集中する特性があり、その結果として、人物などの動きを示すオブジェクト上からはあまり特徴点が抽出されない。図 36 の下の例がそれにあたり、このショットで特徴的な動きを示す、人物の顔から抽出される軌跡はあまり多くない。その結果、ノイズの影響の割合が大きくなり、正しい照合が行われなくなっている。この問題は、テロップの含まれるような領域からは特徴点の軌跡を抽出しないようにすることで、改善が成されるものと考えられる。

4.4.2 アーカイブへの適用

次に、実際の映像アーカイブにおいて、2006 年 12 月 13 日に放送された、5 つの放送局のニュース番組映像に対して手法を適用した。特に朝のニュース番組は取り扱う内容が豊富であり、また、同じトピックが繰り返し流されるといった冗長性を持つため、そこに同一場面映像の組が存在する可能性は高い。そこで、表 5 に示す映像に対して、同一場面映像照合手法を適用した。

放送局	NHK	日本テレビ	TBS	フジテレビ	テレビ朝日
総時間	3 時間				
放送時刻	6:00〜8:00 19:00〜20:00	6:00〜9:00			
ショット数	80	83	75	89	97
(軌跡数≧ 20)	74	78	62	73	82

表 5: 実験に利用した映像データセット

これらのデータセットを構成するために、3.5 で行ったのと同様に、それぞれの映像に対してショット分割を行っている。次に、手法の特性に適応するために、二つの手作業による処理を行っている。その一つはスタジオ風景を撮影したような、いわゆるアンカーショットの除去である。司会者やコメンテーターがそれぞれのトピックについて発言を行うシーンは数多く含まれるが、それらに含まれるショットが同一場面映像となることはほぼありえない。なお、アンカーショットを自動的に判別するという研究も行われており、また、あるショットがアンカーショットであるかどうかは客観的に決まるものである。このような理由により、アンカーショットを除き、それ以外のショットを照合対象とする。また、4.1 で述べたように、本研究はカメラモーションの含まれるショットには対応していない。実際に、カメラが動き続けるようなショットはほぼ存在しないことが確認されているが、ショットの始まりに例えばズームなどが利用されるという例は比較的多く見られる。そこでそのような部分を取り除き、カメラモーションの含まれないような区間に関するモデル化と照合を行った。カメラモーションの推定はショット分割点検出などと同様に、TRECVID の一つのテーマとなっており、このような処理を組み込むことで、自動的にカメラの固定されている区間の抽出などが行えるものと考えられる。

これらの処理の結果得られたショットに対し、同一場面映像照合を行った。まずは特徴点のトラッキングと、inconsistency 計算による特徴量の抽出を行う。この処理は、非常に時間のかかるものであり、計算結果はそれぞれファイルとして保存しておくこととなる。計算時間は入出力を含め、トラッキングは 1 ショット (平均 8.7[s]) あたり約 30 秒、軌跡上の inconsistency の計算は、同様に 2 分半程度かかっている。現在は実装の都合上処理を分割しており、映像のデコードやファイルの IO も全て個別に行っている。このような無駄な処理を減らすことで、かなりの特徴量計算の短縮が見込まれる。次にショット間の類似度を、全てのショットの組み合わせに関して行う。なお、4.4.1 と同様にイベント発生フレーム数が 5 以上の軌跡が 20 未満のショットは、比較に適さないため類似度計算を行わない。この条件を満たすショット数は、表 5 の最下行に示されるものである。類似度の閾値としては、4.4.1 の結果から 0.7 を採用している。その結果として、64 のショット対が同一場面映像として照合され、その内の実際の同一場面映像対は 50 だった。こ

れらの間の閾値の内訳を図 37 に示す。高い類似度を示す映像対は、確かに同一場面映像対であることが見られる。ここに示されるように、同じ映像に対して画面の一部に司会者の映像が映し出されるような Near Duplicate と言えるような映像は、0.9 以上といったような非常に高い類似度を示す傾向がある。それに対し、カメラの違いを含むような同一場面映像対は、誤検出の混ざる閾値付近の類似度となり易く、カメラの違いによる影響の大きさが読み取れる。

図 37: 類似度ごとの、正解不正解の分布

同一場面映像として正しく取得されたショットの例を図 38 に示す。上の二つはスケールや角度に関して比較的大きな違いを持つ映像の組が正しく照合された例である。下の二つはそれほど大きな撮影角度の違いは見られないが、ズームの状態などからわかるように異なるカメラで撮影されたものである。いずれの例も、画面内におけるテロップの占める割合はかなり大きく、また、下の二つは画面下部に司会者やコメンテーターの映像が映し出されている。このようなノイズ影響の下、ほぼ画面中央部分の特徴点の軌跡のみによって、これらの映像対は照合されたことが読み取れる。軌跡による映像モデルの弁別能の強さを示していると言える。

一方、照合し損ねた同一場面映像として確認されたショット対の例を図 39 に示す。図 39 の内、上の例は 4.4.1 で示されたような、大きな撮影角度の違いがあると適切に照合できないという手法の限界を示す例である。その点、下の例は角度

図 38: 正しく検出された同一場面映像の例

の大きな違いもなく、照合の妨げとなる要素は見られないように思うが、左側のショットでは、途中に大きなテロップの出現と消失が含まれており、その結果正しい照合を行える軌跡の数が減っていることが確認された。テロップの出現はその近傍の特徴点に関する inconsistency を大幅に変化させる影響があり、それ以外の特徴点の軌跡間に関しては良い照合結果が得られている。

図 39: 検出されなかった同一場面映像の例

また、本手法は視覚的な類似性に依存しないために、「似て非なる場面」を区別することができる。そのような映像は、放送映像においては非常に一般的なものであり、確かに今回利用したデータセットにも存在した。当然これらは低い類似度を示し、区別されていることが確認された。その例を図 40 に示す。これらは同一の会見の映像ではあるが、その中の異なる部分をそれぞれ映しているものであり、全く別のトピックと関連付けられて放送に使われている。

図 41 は、同一場面映像照合を行った結果、このデータセットに含まれる番組間で生成されたリンクを図示したものである。異なる番組、もしくは異なる放送局間での同一場面映像が抽出されていることが読み取れる。

このように現在の実装において、動きの変化パターンによる映像照合手法により多くの同一場面映像対を発見できた。しかしその反面、全ての同一場面映像を発見するだけの照合能力は持ち合わせていない。それは例えば大きな撮影角度の違いには対処しきれない、また、特徴的な動きを捉えたような特徴点が十分に検出されない場合がある、といった理由による。また、非同一場面映像の組が非常

図 40: 視覚的に類似した非同一場面映像

図 41: 同一場面映像間のリンク

に高い類似度を示すということはなく、高い類似度を持つ映像対が同一場面映像であるという信頼度は高い。しかし、カメラの角度の差やテロップの影響の程度が大きくなると、非同一場面映像の誤検出との完全な弁別は難しく、類似度の閾値を 0.7 とした実験における精度は約 80%であった。

また、このデータセットの全てのショットの組み合わせに関する照合処理を行うのに必要とした時間は約 30 時間であった。実験環境やコーディングに関しては、3.5 と同様である。全てのショットに関する組み合わせの総数は約 7 万通りであり、1 つのショット対の類似度を求めるのに必要な時間は 1.6[s] 程度であるということが示される。フラッシュ発生パターンによる照合に比べ、多対多の比較を行うことにより照合にかかる計算時間は大幅に増加している。しかし、例えば関連手法であるところの Near Duplicate の照合に関して、[64] では 1 対のフレーム画像を照合するのに約 0.4[s] 必要という報告がされている。それに対して本手法では、平均して約 260 のフレーム (=8.7[s]) を含むショット単位の比較を行っており、計算

速度の面でも良い成果を示しているものと考えられる。

4.5 アーカイブ構造化に向けて

4.5.1 ニュース番組における有益な映像区間の抽出

前節の実験で扱ったデータセットは5チャンネル×3時間の合計15時間分であるが、これらの内でその日を代表するようなニュースが扱われている部分はそれほど多くない。例えば番組タイトルや司会者の挨拶などの内容には関わりの無い映像、天気予報のような当日にしか意味を持たない情報、さらには番組独自の特集や占いなどのコンテンツは、多くがその場限りの有用性しか持たないものである。このような部分に関わる映像は、特殊な目的を除いては、アーカイブ化して見るような性質のものではないと考えられる。その反面、世間を騒がせたようなニュースに関する映像の存在は、放送映像アーカイブという観点から見た価値は高いものであり、そのような映像の含まれる部分が明らかとなることで、効率的な閲覧につながるものと考えられる。

このような考えから、本章では同一場面映像を利用したニュース番組の構造化のためのアプローチを取り上げる。ここで目指すのは、重要と思われる映像のありかを明示するような情報を抽出することである。まず、視聴者にとって有益なニュース映像を、以下のようなものと考える。

- 多くの番組で扱われるような、注目されるニュース映像。
- 番組内で重点的に扱われている内容のニュース映像。

前者は映像照合と直接関連するものであり、同一の放送局の内部、もしくは異なる放送局間で張られるリンクの数が多いほど注目度が高いと見なせる。それに対して後者は、別の視点から捉えたニュース映像の区分である。例えば複数のニュース番組がいくつかの同じニュースを扱っているとき、それらの内どのニュースを重点的に伝えるかには偏りがあり、それは番組ごとに異なるのが常であり、たとえ放送局が同じあっても番組が違うことで大きな温度差が生ずる。これらを視聴することを考えると、ニュース内容に関する深い知識を得るためには、そのニュースに重きを置いて放送している番組を見ることが望ましい。そこで単に注目ショットを抽出するに留まらず、番組における重要性を推定することが、より良い視聴のために有効であると考える。

このような偏りを推定するために、同一場面映像照合の結果を利用する。図41を見るとわかるように、同一場面映像とは均等に存在するのではなく、時間的に短い部分にリンクが集中する傾向にあることがわかる。この一つの理由としては、重点的に扱われるようなニュースに関する複数の同一場面映像の存在がある。ニュース番組において、あまり重視されないニュースに関しては、一つの象徴的な映像

を流すだけで次のトピックに移行するが、重視されるニュースは複数のショットにより内容を伝えるのが一般的である。また、例えば人物の会見の映像を流した後、再びその映像を見せつつ音声は司会者やコメンテーターの発言に切り替える、といった編集はしばしば見られる。このような理由により、重点的に扱われるニュース映像の付近には、同一場面映像のリンクが集中しやすくなる。

放送映像の特性を踏まえたこれらの理由から、同一場面映像間のリンクを利用することで、有益なニュース映像を発見することを目指す。なお、この手法で取得することを目指すのは、視聴者にとって有益となるような映像がどのあたりにあるか、という映像内の時間的な位置である。ニュース番組において、その日の中心的なニュースを熱心に伝えているような部分がどこかを探索することが目的であり、例えばショットやシーン、もしくはトピックといった概念による厳密な区切りは一切意識しない。場合によっては複数のトピックが含まれるような部分が、一つの注目映像区間として抽出されることとなる。このような映像抽出は、内容を知らない映像を漠然と閲覧する上での大きな手助けとなるものと考えられる。

4.5.2 リンク数、リンク密度ベースの区間抽出

本節では、具体的な映像区間の抽出法について述べる。同一場面映像のリンクを利用するが、これは基本的には時間軸における K-Means クラスタリングを利用することで行う。それぞれの同一場面映像に対し、それに関わるリンクの数と種別により重み付けを行った後、その分布に関するクラスタリングを行っている。なお、リンクの種別とは放送局に関して内部的なリンクか外部リンクかということを示す。外部リンクの存在は、そのニュースが一般的に興味をひきやすいものであることを示すため、内部リンクと比較して、より大きな重み付けを行うのが適当であると考えられる。これらのリンクに関する重み付けには多くのバリエーションが考えられるが、現状の実装では内外問わずリンクの総数を w_l、外部リンクが張られている放送局数を w_{out} とし、ショットに対する重み付けを表すスコア $S = \sqrt{w_l}(w_{out} + 1)$ として定義する。

まず、ある局の映像内に存在する全ての同一場面映像 IVS_i を、3 つの特徴により表現する。ここで B, E はショットの開始、終了フレーム、S はスコアを表す。

$$IVS_i = (B_i, E_i, S_i)\,. \tag{14}$$

これらの同一場面映像の分布を、時間軸に関してクラスタリングする。K-Means 法では、最初に K 個の初期クラスタ中心を与えるが、ここでいうクラスタ中心 $(C_j\ (j = 1, ..., K))$ とはフレーム番号にあたる。C_j と IVS_i の距離 $D(C_j, IVS_i)$ は、単一のフレームからある程度のフレーム幅を持つショットへの距離により定義される。また、ショットの内部にフレームが含まれる場合、距離 $D(C_j, IVS_i)$ は 0 とする。その代わり、ショットの中心フレームとの距離 $ID(C_j, IVS_i)$ を定義して、同一

場面映像の分類はこれを利用する。つまり、ある同一場面映像に関して距離 0 となるクラスタが 2 つ以上存在した場合は、それぞれのクラスタ中心と、ショットの中心フレームの距離により、それらの内どのクラスタに属するか決定する。

$$D(C_j, IVS_i) = \min\left(|C_j - B_i|, |C_j - E_i|\right). \tag{15}$$

$$ID(C_j, IVS_i) = \left|C_j - \frac{(B_i + E_i)}{2}\right|. \tag{16}$$

全ての同一場面映像がクラスタに振り分けられたら、各クラスタ中心を更新する。これは同一場面映像であるショットの中心フレームをスコアによる重み付けを行った上で、その重心を求める。クラスタ中心 C_j をとるクラスタに含まれる同一場面映像を IVS_k とすると、以下の式によりクラスタ中心は C'_j へと更新される。

$$C'_j = \frac{\sum_k (B_k + E_k) S_k}{2 \sum_k S_k}. \tag{17}$$

この処理をクラスタの更新がなくなるまで行う。その結果得られた各クラスタが注目映像区間に対応すると定義し、それぞれが含む全ての同一場面映像の存在範囲全体をカバーする区間がそれにあたるとする。また、それぞれが含む全ての同一場面映像のスコアの合計 $(\sum_k S_k)$ をもって、そのクラスタのスコアと考える。これらの処理を各放送局ごとに行い、高いスコアを持つクラスタに含まれる区間を注目映像として抽出する。

4.5.3 実験結果

以上の方式に従い、4.4.2 と同じデータセットから注目映像区間の抽出を行った。なお、この方式で利用するリンク情報として、4.4.2 で抽出された同一場面映像間のリンクに加えて 3 章で述べたフラッシュ発生パターンによる映像照合を改めて適用し、そこから得られたリンクも活用した。フラッシュのたかれる映像は映像全体に占める割合は低いものの、存在するものに関しては大体繰り返し利用されるため、この結果としてさらに 70 ものリンクが得られた。なお今回のデータセットでは、フラッシュの発生パターンの共通するような放送局の異なる同一場面映像は存在せず、結果としてフラッシュ発生パターンからは内部リンクしか生成されていない。

K-Means アルゴリズムに与える初期値としては、1 時間当たり 6 個のクラスタを等間隔に与えた。この結果、求められたニュース番組内の注目度の分布を図 42 に示す。これは、放送局別の横軸に時間をとった時のクラスタのとる区間を塗りつぶしたもので、濃く塗られている部分ほど、同一場面映像間のリンクにより求められた注目度が高いことを示す。

図 42: 注目映像区間クラスタの様子

また、得られたクラスタは大体 10 秒〜10 分程度の区間を持っている。長いものは一つのトピックだけでなく、複数のトピックを一つのクラスタ内に含む。中でも特に高い注目度を示したクラスタの例を、良いクラスタの得られなかった NHK を除く 4 つの放送局それぞれに関して示す。

図 43 を見ると、比較的芸能ニュースなどがまとめられて抽出されていることがわかる。これらのニュースは、ほぼ確実に会見を映したような現場の映像含み、そこには特徴的な動きが見られたり、フラッシュがたかれていたりする可能性が高い。そのためショット間に張られるリンクの数が増大し、さらには芸能ニュースなどは一つの番組のコーナーとしてまとめられる傾向があるため、結果として大きなクラスタが作られている。その次に目立っているのは、その日の重要なトピックを個別に扱っているような区間である。このデータセットに含まれる映像の流された日は、ひき逃げ事件の時効成立 (図 43(a)) と松坂選手の入団交渉 (図 43(b))、そしてタウンミーティングのやらせ質問問題 (図 43(c))、といった辺りが主要なトピックであり、これらを含むようなクラスタは比較的高い値を示す。

しかしクラスタによっては、同一場面映像照合の false alarm のためにここでいうところの注目映像ではない、番組独自の特集のような映像が混ざるものがある (図 43(d))。そのために、それらの間のやはりニュースとは関係の無い区間までもがクラスタに含まれることとなり、クラスタの有益性を損ねている。これは同一場面映像をまとめる手法に関する工夫により、解くべき問題である。また、4.4 では動きの変化による同一場面映像照合に関して、recall 向上のために解決すべき問

図 43: 各クラスタに含まれる映像例

題があることが明らかになったが、この影響により、全体的に得られるクラスタの量が少なくなっている。この問題に関しては、映像照合の精度を上げることが最も重要である。

現在の実装ではこのような問題が明らかになっており、映像照合手法を具体的に構造化されたアーカイブとして利用するために、今後取り組むべき課題である。またこの実験では、クラスタリングを行う際に時間的な距離だけを利用しており、得られたクラスタは全て独立したものとして扱っている。つまり、同一場面映像間のリンクをショット単位の関連性としてのみ利用している。例えばクラスタ単位でのリンクの接続関係を利用するなど、より高度な利用法を考案することが、照合手法の改善とともに求められる課題であると考えられる。

5　結論

5.1　提案手法に関する結論

本研究で取り組んだ問題

本稿では、映像アーカイブの構造化を目標に掲げ、そのための一つのアプローチである映像照合手法の提案を行った。映像照合とは同一と見なされる映像を照合することであり、特に放送映像のような、同じ場面を撮影した映像がしばしば現れるアーカイブにおいてこのような手法は有効に働く。例えば映像間の照合を行うことで、関連するトピックの追跡や人気映像の自動抽出などのアプリケーションの実現の可能性がある。ただし、既存の映像照合手法の多くはフレーム画像の類似性に依存して照合を行うものであり、同じ場面を撮影した映像であっても、撮影するカメラの違いなどによる視覚的な差があることで照合が行えない、もしくは異なる場面であっても視覚的に類似していれば照合してしまう、という問題があった。また、そのような視覚的な差に対応しようとすると計算量が大きくなり、大容量の映像アーカイブへの適用が難しい。これらの問題に対して本研究では、場面が同一であれば、視覚的な差に影響されずに照合できるような映像照合手法である同一場面映像照合手法を提案した。これは照合基準に関して、既存の手法に比べて明確な定義が成されているのが特徴であり、そのために性能評価も行い易い。

同一場面映像照合を行うためのアプローチ

上記の問題に対応するため、提案手法は以下のような形をとる。なお、本研究では映像をショット単位で扱い、ショットへの分割は正しくできていることを前提としている。

- 視覚的な差に対応するため、フレーム画像だけではなく、ショットを表す時空間を扱う。
- 計算速度、データ量の問題に対応するため、ショットを数値のシーケンス、もしくはその集合でなる特徴量で表す。シーケンス内の各値は、ショット内の各フレームに対応する。
- シーケンス間の比較による照合を行う。

カメラやテロップなどの、視覚的な差を生ずるような要素の影響を受けずに場面の一意性を表現できるようなシーケンスを定義することで、照合に関する要件は満たされる。また、数値のシーケンスはデータ保存のコストは明らかに少ない。数値のシーケンス間の比較は、複雑な構造を持つ映像同士の比較に比べ、はるかに高速に行える。よって、シーケンスをいかにして求め、いかにして比較するか、

を定義することが同一場面映像照合の提案となる。本稿では二つの手法を提案し、その性能評価を行った。

フラッシュ発生パターンによる照合の結論

一つ目に提案した手法では、カメラのフラッシュの時間的な発生パターンの同一性によってショットの照合を行う。ショットを表す特徴として、フラッシュが観察されたフレームを 1、それ以外を 0 とするような二値のシーケンスが利用される。フレームの平均輝度値の変化により、フラッシュの発生するフレームは検出されるが、この処理は取りこぼしはほぼ無いものの、誤検出が混ざりやすいという問題がある。そこで、フラッシュという現象の特性を利用した判定を二種類行って、フラッシュ検出の精度を向上する。このようにして求めたシーケンス間の類似度を調べることで、ショット間の照合を行われる。

これらの手法を、大量の放送映像に適用することで、性能評価を行った。フラッシュ検出に関しては、二つの判定処理により多くの誤検出を取り除くことができた。これにより、ショット間の照合精度、照合にかかる計算時間も大きく向上することが確認された。また、ショット間の照合の結果、同一場面映像が高い正解率で、かつ非常に高速に検出された。フラッシュのたかれるようなショットに関する映像照合は、上で挙げたような要件を十分に満たす結果が得られたと言える。

動きの変化パターンによる照合の結論

次に、ショット内でのオブジェクトの動きの変化を照合基準とするような手法を提案した。フラッシュの発生パターンによる照合の欠点として、フラッシュがたかれるような映像が映像アーカイブ内に占める割合は小さいことによる汎用性の低さがある。それに対しこの手法は、ショット内にある程度の動きが存在すれば照合対象となるため、より多くのショットについて照合を行うことができる。この手法でショット間の照合基準となる要素は、特徴点のトラッキングによりショットから抽出される特徴点の動きの軌跡である。軌跡は既存のトラッキング手法により検出され、それぞれの軌跡が inconsistency と呼ばれる [40] にて定義される値のシーケンスとして表現される。このシーケンス間の類似度は、主に相関計算により算出される。ショット間の類似度は、全ての軌跡の組み合わせから算出される類似度から、軌跡同士の対応関係を求めつつ計算される。

この手法についての実験結果から、高い類似度を示すような映像は確かに同一場面映像であるという結果が確認された。しかしその反面、照合できなかった同一場面映像が多く存在した。例えば撮影するカメラの角度が大きく異なる、特徴的な動きを捉えたような軌跡の数が少ない、などがその主な原因であり、現状の手法の限界となっている。なお、色合いの違いやズームといった画像的な変化に対しては、高いロバスト性が示された。処理速度に関しては、フラッシュの発生

パターンによる照合手法と比較すると、約 1000 倍の計算時間が必要となる。しかし、既存のキーフレーム間の比較による照合手法と比較すると十分な計算時間に抑えられている。場面の同一性を示すには時間方向の情報を利用することが必須であり、扱うデータサイズは遥かに増大している。そのため、既存の手法では成し得なかった映像照合を高速に行うことができたと言えるだろう。

5.2 今後の課題

本研究で提案した映像照合手法は、映像アーカイブ構造化という大きな目的のための一つの要素技術であれう。これを有効利用するための一例として、4.5 では一つのアプリケーションを作成したが、いまだ解決すべき問題は多い。

- 動きの変化パターンによる照合手法の改善：4 章で提案した手法の実験による評価から、照合しきれない映像の特徴が明らかとなり、またその原因も判明している。これらの問題に対応することで、手法の精度を向上させる必要がある。
- テキスト情報との連携：同一場面映像であることが保証されることで、それらの間で共起するようなテキストが、内容に関わっている可能性は強まる。テキスト空間におけるノイズの影響が、場面的同一性により大きく緩和される。
- GUI の利用：同一場面映像間でリンクを張る、さらには頻出する映像を人気映像として抽出する、などの結果を目に見え利用できる形で表現することで、アーカイブの利用の大きな助けとなる。

このようなアプローチが、同一場面映像照合による映像アーカイブの構造化を有効に行うための今後の課題といえる。

発表文献

- 瀧本政雄，佐藤真一，坂内正夫，“大容量放送映像アーカイブからの同一フラッシュシーン映像の発見”, 電子情報通信学会論文誌 D Vol.J89-D No.12 pp.2699-2709, 2006.
- Masao Takimoto, Shin'ichi Satoh, Masao Sakauchi, “Identification and detection of the same scene based on flashlight patterns”, ICME2006, 2006.
- Masao Takimoto, Shin'ichi Satoh, Jun Adachi, “Identification Method for Video Shots of the Same Scene Based on Trajectories of Feature Points”, CVPR2007, 2007. （投稿中）
- 瀧本政雄，佐藤真一，坂内正夫，“フラッシュパターンに基づく映像照合手法”, 情報技術レターズ, 第二回情報科学技術フォーラム (FIT2003), 2003.
- 瀧本政雄，佐藤真一，坂内正夫，“フラッシュパターンに基づく類似映像照合”, 電子情報通信学会技術研究報告, IE2003-27, PRMU2003-57, MVE2003-39, 2003.

参考文献

[1] 国立情報学研究所, http://www.nii.ac.jp/

[2] NHK アーカイブス, http://www.nhk.or.jp/archives/

[3] Youtube, http://www.youtube.com/

[4] Google Video, http://video.google.com/

[5] TRECVID, http://www-nlpir.nist.gov/projects/trecvid/

[6] イマジカ映像バンク, http://audrey.imagica.co.jp/i-view/

[7] Wallapak Tavanapong and Junyu Zhou, "Shot Clustering Techniques for Story Browsing", IEEE Transactions on Multimedia, 2004.

[8] Y. Zhai, A. Yilmaz and M.Shah, "Story Segmentation in News Videos Using Visual and Text Cues", International Conference on Image and Video Retrieval, 2005.

[9] Y. Zhuang, Y. Rui, T. S. Huang and S. Mehrotra, "Adaptive Key Frame Extraction Using Unsupervised Clustering", In Proc. Intelligent Conference on Image Processing, vol. 1, pp. 866-870, 1998.

[10] K. Hoashi, M. Sugano, M. Naito, K. Matsumoto, F. Sugaya and Y. Nakajima, "Shot Boundary Determination on MPEG Compressed Domain and Story Segmentation Experiments for TRECVID 2004", TREC Video Retrieval Evaluation Forum, 2004.

[11] 三浦菊佳，山田一郎，住吉英樹，八木伸行, "クローズドキャプションを利用した映像主被写体の推定手法", 情報処理学会研究報告, NL171-1, Vol.2006, No.1, pp1-6, 2006.

[12] 山田一郎，三浦菊佳，住吉英樹，八木伸行，奥村学，徳永健伸，"AdaBoost を利用した字幕テキストからの定型表現文章区間抽出の検討", 言語処理学会第 12 回年次大会発表論文集，D1-4, 2006.

[13] Yiming Yang, Jaime Carbonell, Ralf Brown, Tom Pierce, Brian T. Archibald, Xin Liu, "Learning approaches for Detecting and Tracking News Events", IEEE Intelligent Systems, vol. 14, no. 4, 1999.

[14] Michal Irani, "Multi-Frame Optical Flow Estimation Using Subspace Constraints", Proc. ICCV, 1999.

[15] Carlo Tomasi and Takeo Kanade, “Detection and tracking of point features”, Technical report CMU-CS-91-132, 1991.

[16] Mei Han, Wei Xu, Hai Tao and Yihong Gong, “An Algorithm for Multiple Object Trajectory Tracking”, Proc. CVPR, 2004.

[17] K. Mikolajczyk and C. Schmid, “An affine invariant interest point detector”, In Proc. ECCV, Springer-Verlag, 2002.

[18] F. Schaffalitzky and A.Zisserman, “Multi-view matching for unordered image sets, or “How do I organize my holiday snaps?” ”, In Proc. ECCV, Vol.1, pp.414-431, 2002.

[19] David G.Lowe, “Object Recognition from Local Scale-Invariant Features”, In Proc. ICCV, pp.1150-1157, 1999.

[20] Cordelia Schmid, Roger Mohr and Christian Bauckhage, “Evaluation of interest point detectors”, International Journal of Computer Vision, vol.37, no.2, pp.151-172, 2000.

[21] J. Matas, O. Chum, M. Urban and T. Pajdla, “Robust Wide Baseline Stereo from Maximally Stable Extremal Regions”, In Proc. BMVC, pp.384-393, 2002.

[22] S. Smith, “A new class of corner finder”, In Proc. 3rd British Machine Vision Conference, pp. 139-148, 1992.

[23] John Wang and Edward Adelson, “Representing Moving Images with Layers”, In The IEEE Transactions on Image Processing, vol. 3, no. 5, pp. 625–638, Sept, 1994.

[24] Philip H.S. Torr, Richard Szeliski, P. Anandan, “An Integrated Bayesian Approach to Layer Extraction from Image Sequences”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.23, no.3, pp.297-303, Mar, 2001.

[25] N. Vasconcelos and A. Lippman, “Empirical Bayesian Motion Segmentation”, IEEE Trans. Pattern Anal., 2001.

[26] Josef Sivic and Andrew Zisserman, “Video Google: A Text Retrieval Approach to Object Matching in Videos”, Proc. ICCV, pp.1470-1477, 2003.

[27] Fred Rothganger, Svetlana Lazebnik, Cordelia Schmid and Jean Ponce, “Segmenting, Modeling, and Matching Video Clips Containing Multiple Moving Objects”, IEEE Conference on Computer Vision, 2004.

[28] R.Fergus, P.Perona and A.Zisserman, "Object Class Recognition by Unsupervised Scale-Invariant Learning" Proc. CVPR, Vol.2, June, 2003.

[29] J. Shi and J. Malik, "Normalized cuts and image segmentation" IEEE Transactions on Pattern Analysis. and Machine Intelligence 22(8), pp. 888–905, 2000.

[30] T. Lindeberg, "Edge detection and ridge detection with automatic scale selection", International Journal of Computer Vision, 30, 2, pp 117–154, 1998.

[31] D. Hoiem, R. Sukthankar, H. Schneiderman and L. Huston, "Object-Based Image Retrieval Using the Statistical Structure of Images", IEEE Conference on Computer Vision and Pattern Recognition, June, 2004.

[32] H. Schneiderman, "Feature-Centric Evaluation for Efficient Cascaded Object Detection" IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, June, 2004.

[33] R. Fergus, P. Perona and A. Zisserman, "A Visual Category Filter for Google Images", In Proc. ECCV. Springer-Verlag, May 2004.

[34] P. Montesinos, V. Gouet and R. Deriche, "Differential Invariants for Color Images", International Conference on Pattern Recognition (ICPR'98), pages 838-840, 1998.

[35] A. Baumberg, "Reliable Feature Matching Across Widely Separated Views", In. Proc. CVPR, vol. 1, pp. 774-781, 2000.

[36] M. Weber, M. Welling, P. Perona, "Unsupervised Learning of Models for Recognition", In Proc. 6th ECCV, 2000.

[37] J. Thureson and S. Carlsson, "Appearance Based Qualitative Image Description for Object Class Recognition", In Proc. ECCV, pages 518–529, 2004.

[38] V. Ferrari, T. Tuytelaars and V. Gool, "Simultaneous Object Recognition and Segmentation by Image Exploration", In Proc. ECCV, Vol. I, pp. 40-54, 2004.

[39] Y. Shan, H.S. Sawhney, B. Matei and R. Kumar, "Partial Object Matching with Shapeme Histograms", In Proc. ECCV, 2004.

[40] Eli Shechtman and Michal Irani, "Space-Time Behavior Based Correlation", Proc. CVPR, pp.405-412, 2005.

[41] Josef Sivic, Frederik Schaffalitzky and Andrew Zisserman, "Object Level Grouping for Video Shots", ECCV, 2003.

[42] Ivan Laptev and Tony Lindeberg, "Space-time Interest Points", ICCV, 2003.

[43] G.Piriou, P.Bouthemy and J.F.Yao. "Extraction of Semantic Dynamic Content from Videos with Probabilistic Motion Models", Proc ECCV, vol.3, pp.145-157, 2004.

[44] Y. Wexler, E. Shechtman, and M. Irani, "Space-Time Video Completion", IEEE Conference on CVPR, June 2004.

[45] Oren Boiman and Michal Irani, "Detecting Irregularities in Images and in Video", IEEE International Conference on Computer Vision (ICCV), Beijing, October 2005.

[46] Hua Zhong, Jianbo Shi and Mirko Visontai, "Detecting unusual activity in video", In Proc. CVPR, June, 2004.

[47] Chong-Wah Ngo, Ting-Chuen Pong and Hong-Jiang Zhang, "On Clustering and Retrieval of Video Shots Through Temporal Slices Analysis", IEEE Transactions on Multimedia, Vol.4, No.4, 2002.

[48] M. Grabner and H. Bischof, "Extracting object representations from local feature trajectories", In 1st Cognitive Vision Workshop, 2005.

[49] L. Xie, L. Kennedy, S. -F. Chang, A. Divakaran, H. Sun and C. -Y. Lin, "Discover meaningful multimedia patterns with audio-visual concepts and associated text", in ICIP, Singapore, 2004.

[50] M. Blank, L. Gorelick, E. Shechtman, M. Irani and R. Basri, "Actions as Space-Time Shapes", IEEE International Conference on Computer Vision (ICCV), Beijing, October 2005.

[51] P. Viola and M. Jones, "Robust Real-Time Object Detection", International Journal on Computer Vision, 2001.

[52] Nozha Boujemaa, Francois Fleuret, Valerie Gouet and Hichem Sahbi. "Automatic Textual Annotation Of Video News Based on Semantic Visual Object Extraction", Proc. SPIE, Vol.5307, pp.329-339, 2003.

[53] H. Schneiderman and T. Kanade, "A Statistical Method for 3D Object Detection Applied to Faces and Cars", IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2000), 2000.

[54] Paul Viola, Michael J.Jones and Daniel Snow, “Detecting Pedestrians Using Patterns of Motion and Apperance”, Proc. Ninth IEEE International Conference on Computer Vision (ICCV), 2003.

[55] C. Kong, A. Calway and M. Mirmehdi, “Recognizing Animals Using Motion Parts”, In Proc. 16th British Machine Vision Conference, pp. 796-808. 2005.

[56] M. A. Smith and Takeo Kanade, “Video Skimming and Characterization through the Combination of Image and Language Understanding Techniques”, CMU Technical Report CMU-CS-97-111, Feb 1997.

[57] Shin’ichi Satoh, Yuichi Nakamura, and Takeo Kanade, “Name-It: Association of Face and Name in Video”, IEEE MultiMedia, Vol. 6, No. 1, January-March, pp. 22-35, 1999.

[58] 長坂晃朗，宮武孝文 “時系列フレーム特徴の圧縮符号化に基づく. 映像シーンの高速分類手法”, 電子情報通信学会論文誌，Vol.J81-. D-II, No.8, pp.1831-1837 1998.

[59] M. Y. M. Naphade and B.-L. Yeo, “A novel scheme for fast and efficient video sequence matching using compact signatures”, In Proc. SPIE, Storage and Retrieval for Media Databases 2000, Vol. 3972, pp. 564-572, Jan. 2000.

[60] A. Hampapur and R. Bolle, “Comparison of Sequence Matching Technique for Video Copy Detection”, In Conf. on Storage and Retrieval for Media Databases, pp. 194-201, 2002.

[61] 柏野邦夫，木村昭悟，黒住隆行, “局所的・大域的クラスタリングに基づく音・映像の高速時系列探索法”, 電子情報通信学会技術報告， PRMU2005-25 (DE2005-3)，2005.

[62] 柏野邦夫, ガビン・スミス, 村瀬洋. “ヒストグラム特徴を用いた音響信号の高速探索法 –時系列アクティブ探索法–”, 信学論, J82-DII, No.9, pp.1365-1373, 1999.

[63] 山岸史典，佐藤真一 “同一映像断片探索に基づく映像ブラウジングの提案”, 第 2 回情報科学技術. フォーラム（FIT2003）講演論文集, vol.3, pp.103–104, 2003.

[64] Dong-Qing Zhang and Shih-Fu Chang, “Detecting Image Near-Duplicate by Stochastic Attributed Relational Graph Matching with Learning”, In ACM Multimedia, New York, 2004.

[65] A. Jaimes, S. -F. Chang and A. C. Loui, "Duplicate detection in consumer photography and news video", In Proc. tenth ACM international conference on Multimedia, 2002.

[66] Julien Law-To, Valerie Gouet-Brunet, Olivier Buisson and Nozha Boujemaa, "Local Behaviours Labelling for Content Based Video Copy Detection", Proc. ICPR, Vol.3, pp.232-235, 2006.

[67] Winston H. Hsu and Shih-Fu Chang, "Topic Tracking across Broadcast News Videos with Visual Duplicates and Semantic Concepts", Proc. ICIP, Atlanta, GA, October 2006.

[68] Yun Zhai and Mubarak Shah, "Tracking News Stories Across Different Sources", in ACM Multimedia, Singapore, 2005.

[69] Alexis Joly, Olivier Buisson and Carl Frelicot, "Statistical Similarity Search Applied to Content-Based Video Copy Detection", Proc. ICDEW, April, 2005.

[70] A. Joly, C. Frelicot and O.Buisson, "Feature statistical retrieval applied to content-based copy identification", In ICIP, 2004

[71] Yaron Caspi and Michal Irani, "Spatio-Temporal Alignment of Sequences", IEEE transaction on pattern analysis and machine intelegence, Vol.24, No.11, November, 2002.

[72] Carlo Tomasi and Takeo Kanade, "Shape and Motion from Image Streams: a Factorization Method", Techinical Report CMU-CS-92-104, Carnegie Mellon University, March, 1992.

[73] Richard Szeliski. "Video Mosaics for Virtual Environment", IEEE Computer Graphics and Applications, Vol.16, No.2, pp.22-30, March, 1996.

[74] G.Stein, R.Romano and L.Lee, "Monitoring Activities from Multiple Video Streams: Establishing a Common Coordinate Frame" IEEE Trans. Pattern Analysis and Machine Intelligence, vol.19, no.3, pp.268-273, March, 1997.

[75] T.Syeda-Mahmood, A.Vasilescu and S.Sethi, "Recognizing action events from multiple viewpoints" Proc. IEEE Workshop Detection and Recognition of Events in Video, 2001.

[76] P.H.S.Torr and A.Zisserman, "Feature Based Methods for Structure and Motion Estimation" Proc. Vision Algorithms Workshop, pp.279-295, 1999.

[77] M. Irani and P. Anandan, “About Direct Methods”, ICCV workshop on Vision Algorithms, pp. 267-277, Corfu, September 1999.

[78] L. Zelnik-Manor and M. Irani, “Multi-View Constraints on Homographies”, IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), Vol. 24, No. 2, pp. 214-223, February 2002.

[79] 中村裕一, 金出武雄. “ニュース映像からの重要セグメント抽出”, 第 3 回知能情報メディアシンポジウム, pp.61-68, 1997

[80] J. R. Kender and M. R. Naphade, “Visual Concepts for News Story Tracking: Analyzing and Exploiting the NIST TRECVID Video Annotation Experiment”, in CVPR, San Diego, 2005.