静止画内物体への変形指示による動画検索

川手 裕太　　岡部 誠　　尾内 理紀夫　　平野 廣美 *

概要. 我々は動画から視聴したいシーンを素早く，容易に検索するためのユーザインタフェースを提案する．既存の動画共有サイトで使用されている検索エンジンはテキストを使用したものであるため，視聴したいシーンの様子までは検索することができないといった問題点が存在する．我々のシステムでは，ユーザが動画内の一時停止フレーム等の静止画を入力し，静止画内の物体の変形を矢印 2 本のスケッチによって指示をすることができる．その後，システムはユーザが矢印で指定した変形に類似したシーンを検索し，ユーザは目的のシーンを視聴することができる．我々の手法では，ユーザが視聴したい目的のシーンを検索するために 2 つのステップを採用している．第 1 ステップでは標準的な画像検索技術を使用し，入力された静止内の物体と同じ外観のフレームを検索する．第 2 ステップでは前ステップで選択したフレームから自動的に前後フレームへと早送りと巻戻しを行うことで，目的のシーンを検索する．早送りと巻戻しを行い，ユーザが 2 本の矢印で指定した変形が発見された場合，早送りと巻戻しを停止させ，目的のシーンを出力する．この 2 つのステップを行うことにより，静止画内物体が横向きにもかかわらず，外観の異なる正面向きの物体が映るシーンを検索するといったことを可能にした．我々はこの手法が F1 カーと馬，飛行機の 3 種類の物体において有効であることを確認した．

1　はじめに

動画共有サイトの普及により，インターネット上で視聴することができる動画の数は日に日に増加している．ユーザはそれらの動画から視聴したいシーンを迅速かつ容易に検索したいと考えている．ただし，YouTube[1] やニコニコ動画 [2] などの動画共有サイトの動画検索エンジンは，テキストでしか検索することができない．その検索エンジンでは，各動画のタイトルや割り当てられているテキストタグの情報に基づいて検索が行われている．従って，これらの検索エンジンで，検索項目として視聴したい物体の様子に関したテキストを入力しても検索できない．例えば,「左上に飛んで行く飛行機」というテキストに対して，YouTube やニコニコ動画ではいずれも，図 1 のような左上方向へ飛行する飛行機のシーンを取得することはできない．さらに，そのようなテキストを理解できる動画検索エンジンがあったとしても，ユーザが物体の様子を言葉で正確に記述することは困難であり，面倒である．

このような問題を解決するために，我々はユーザが迅速かつ容易に視聴したい物体の様子を指定することができる新しいユーザインターフェースを提案する．ユーザインターフェースでユーザが検索を行う際の流れは以下の通りである．

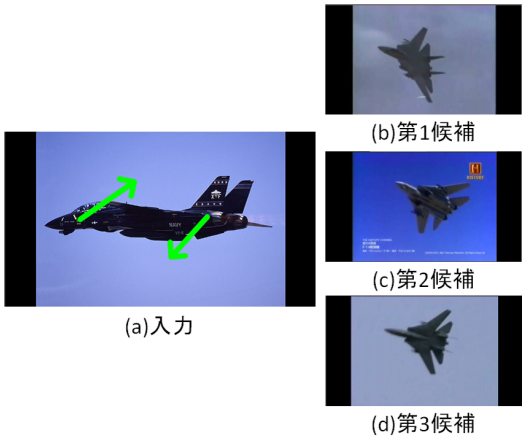

図 1. 提案システム

1. 視聴したい物体の映る動画の一時停止のフレームまたは静止画を入力静止画とする．
2. スケッチインターフェース (図 1(a)) にて，2 本の緑矢印の描画で入力静止画内の物体の変形を指定する．
3. システムが動画データベースから検索結果の候補のシーンを表示する (図 1(b,c,d))．

ユーザは 2 本の緑矢印で入力静止画内の物体の変形を指示することで視聴したいシーンをインタラクティブに検索することができる．また，この 2 本の緑矢印は 2 次元の入力であるが，横向きの入力静止画から正面向きのシーンを検索するといった，3 次元で変形させたようなシーンも検索することが可能

* Yuta Kawate, 電気通信大学, Makoto Okabe, 電気通信大学/JST CREST, Rikio Onai, 電気通信大学, Hiromi Hirano, 楽天株式会社/電気通信大学

[1] http://www.youtube.com/

[2] http://www.nicovideo.jp/

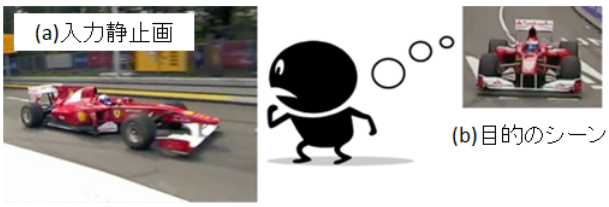

図 2. モチベーション

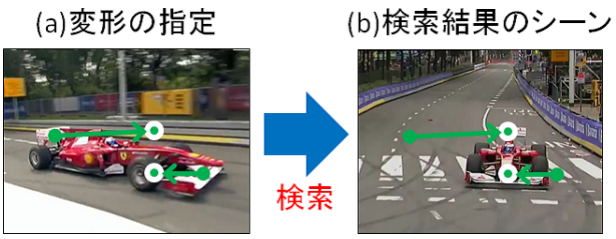

図 3. 提案するユーザインタフェース

である．実験において，この方法が F1 カー，馬，飛行機といった複数の種類の物体に有効であることを確認した．

2 関連研究

多くの動画からインタラクティブな検索を可能にする外観ベースのインタフェースの提案はなされている．bag-of-features を使用し，比較的低次元のベクトルとして各動画のフレームを表すことにより，入力した静止画から静止画や動画の検索が提案された [3, 15]．多くの動画や静止画から構成された 3 次元のシーンをユーザがインタラクティブに動きまわる事ができるシステムが提案された [1, 5, 16, 17]．これらは structure-from-motion を介してシーンやカメラの位置を 3 次元再構成を行なっている．しかし，これらの方法は，建物などの静止物体にしか適用されておらず，移動したり，変形したりする物体に適用することは困難であり，非常に計算時間がかかる．

動画内の物体を操作することで，連続した単一の動画内のシーンのインタラクティブなナビゲーションを可能にするシステムが提案された [4, 7, 8, 9, 10]．ユーザは，単一動画内の物体をドラッグによってインタラクティブな姿勢の操作ができ，ナビゲートできる．これらは，3 次元再構成ではなく，2 次元の画像処理技術が行われているので，比較的計算時間が少ない．しかし，我々のシステムは単一の動画だけではなく，複数の別々の動画から物体を検索することができる．

3 ユーザインタフェース

我々の提案するユーザインタフェースは，ある物体の動画の視聴時に別のシーンが見たいといった，物体の様子を検索するシステムである．例えば，図 2(a) のような F1 カーが左から右に走行するシーンの視聴している時，図 2(b) のように正面向きに走行する F1 カーのシーンを視聴したいという状況を考える．ユーザはそのようなシーンを視聴するために，早送りや巻き戻しを行うか，シークバーを操作する．もし，今視聴している動画にそのようなシーンが含まれない場合，ユーザが他に所持している F1 カーの動画を一つ一つ見る必要がある．それでも発見できなかった場合，YouTube 等の動画共有サイトで見つけ出さなければならない．その作業は非常に面倒である．

我々の提案するユーザインタフェースは目的のシーンを素早く，容易に検索が可能である．ユーザが入力するものは図 3(a) のように，視聴したい F1 カーが写っている静止画と 2 本の緑矢印だけである．このユーザインタフェースにおいてユーザは，「視聴したい F1 カーが現在見ている F1 カーの回転したバージョンである」という考えをもとに 2 本の緑矢印を入力する．この考えをもとにして，3 次元右向きの F1 カーを正面向きへ回転させることを考えた場合，正面向きの F1 カーを 2 次元の静止画上で見ると，図 2(b) のように F1 カーの前方は下部，後方は上部に配置される．このように F1 カーを回転させたと想定して 2 本の緑矢印で変形を指定する．ここでは，ユーザは図 3(a) のように F1 カーの前方を下方向へ，後方を上方向へと矢印を入力する．我々のシステムはその 2 本の緑矢印で指定した変形に類似した目的のシーン (図 3(b)) を検索する．

4 アルゴリズムの概要

動画や静止画検索の技術において，今視聴しているシーン (図 2(a)) を入力として，目的のシーン (図 2(b)) を直接探し出すことは困難である．例えば，SIFT 特徴量 [11] を動画から抽出し，今視聴しているシーンと目的のシーンのマッチングの計算を行う．マッチングに F1 カーを使用した場合の結果は図 4 で示す通りである．

右向きと左向きの F1 カーといった外観が類似している左のケースでは正確なマッチングが多く抽出されている．一方で，右向きと正面向きの F1 カーといった右のケースでは不正確なマッチングが抽出されているか，マッチングの数が少ない．ここで興味深いことに，図 4(a) と (b) のフレームは図 5 で示す通り同じシーンに存在している．

図 6(b) のような入力静止画 (図 6(a)) と外観が類似したフレームを探し出し，そのフレームから動画を巻き戻すことで図 6(c) のような目的のシーン (図 6(d)) を探し出すことができる．巻き戻しで図 5(d)

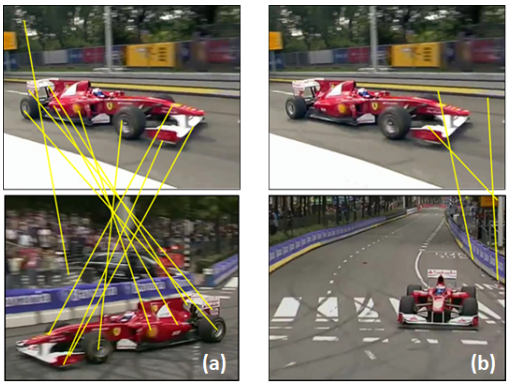

図 4. マッチングの例

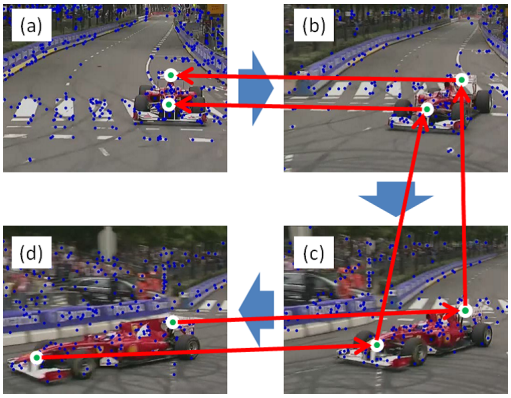

図 5. F1 カーが正面から左向きになる動画

と (a) を関連付ける方法として，図 5 の緑色の点のような物体上に配置された点を追跡する．

検索アルゴリズムは 2 ステップある．

1. フレーム検索ステップ

   SIFT 特徴量を使用したマッチングにより，入力静止画内の物体と類似したフレームを検索する．

2. フレーム追跡ステップ

   類似フレームから早送り・巻き戻しを行い，ユーザの入力矢印で指定された変形に類似したフレームを出力する．

2 ステップについて 4.1 と 4.2 で記述する．

4.1 フレーム検索

入力静止画と SIFT 特徴量を使用したマッチングを行った結果，マッチング数が最多のフレームを類似したフレームとして選択する．ただし，SIFT 特徴量をそのままマッチングに使用する方法では計算時間がかかる．計算時間を減らすために，Visual Words[15] を使用したマッチングを行う．この計算時間の比較は 5.1 で記述する．

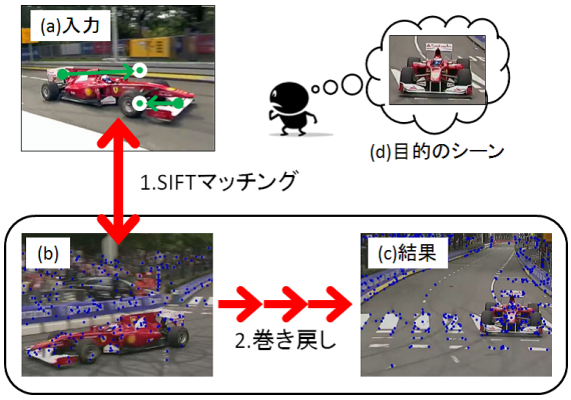

図 6. アルゴリズムの概要

Visual Words を使用したマッチングを行うために以下の前準備を行う．

1. 検索対象の動画の全フレームから SIFT 特徴量を抽出する．
2. 抽出した特徴量をランダムに選択し，クラスタリングを行う．クラスタリングには K-means 法を使用する．
3. それぞれのクラスタの中心のデータ (Visual Words) を辞書と定義する．
4. 辞書を使用し，残りの SIFT 特徴量についてどのクラスタに分類されるかを特定する．

この前準備で求めた動画の各フレームのクラスタの特定が行われた SIFT 特徴量は，ユーザインタフェースのシステム起動時に読み込まれる．ユーザインタフェースで静止画が入力された時，入力静止画と動画の各フレームとのマッチングは以下の流れで行う．

1. 入力静止画から SIFT 特徴量を抽出する．
2. 抽出した特徴量を辞書を使用し，クラスタの特定を行う．
3. 動画内フレームから抽出された各特徴量とクラスタの番号が一致する場合にマッチングとする．

マッチングは多く抽出されるが，その中には外観と比較すると不正確に見えるものが存在する．そのようなノイズを減らすために，RANSAC[6] を使用する．

Visual Words を使用し，入力静止画と外観が類似したフレームをマッチングを行った結果の例が図 7 である．図 7 下のフレームを含む動画において，マッチング数が最多であるので，このフレームをその動画での類似したフレームとして選択する．ただし，Visual Words を使用したことにより SIFT 特徴量をそのまま使用した場合よりも精度が低下している．よって，人間の見た目には類似していないにもかかわらず，マッチング数が動画内で最多のフレームが存在することがある．この問題に対して，マッ

図 7. Visual Words を使用したマッチングの例

チングの数が多い順から複数の候補のフレームを類似したフレームとして使用する．

4.2 フレーム追跡

4.1 で選択した類似フレームから前後フレームを探索し，ユーザが矢印で指定した変形にマッチングしたフレームを出力する．前後フレームの探索には図 5 のようにフレーム上に配置された点を追跡する．追跡点を抽出する処理は予め行い，システム実行時に追跡点データを読み込むようにする．物体の追跡には particle video[13] を使用したアルゴリズムと，SIFT 特徴量のマッチングを使用したアルゴリズムの 2 種類の方法を行った．前者は精度は高いが計算時間が非常にかかるため，実験では SIFT 特徴量のマッチングを使用する．SIFT 特徴量を使用した物体の追跡は多く行われている [2, 14] が，簡略化した以下のような流れで行う．

1. 動画内の各フレームで SIFT 特徴量を抽出する．
2. $n-1$ と n フレーム，n と $n+1$ フレームの SIFT 特徴量のマッチングをそれぞれ計算する．
3. $n-1$ と $n+1$ フレームとのマッチングが存在する n フレームの特徴点を追跡点とする．

そのアルゴリズムを使用し，追跡は以下のように行う．

1. 4.1 のマッチングの結果を使用し，入力矢印の始点と対応した類似フレーム上の点 (対応点) を求める．
2. 対応点と近傍の追跡点を求める．
3. フレームを移動させ，入力矢印の終点と追跡点が最近似のフレームを求める．

システムは，各動画の最近似フレーム中，上位 3 フレームを検索結果として出力する．

表 1. 各マッチングの計算時間 (秒)

	SIFT 特徴量 [12]	Visual Words
特徴量抽出	1.23	1.23
クラスタ特定	-	1.84
マッチング	91.11	1.78
合計	92.34	4.85

5 実験

すべての実験は CPU Intel i7-3930k 3.20GHz，メモリ 16GB のパソコンで行った．用意した全ての動画の大きさは 640×360 である．

5.1 計算時間比較

4.1 で記述したフレーム検索ステップにて，SIFT 特徴量をそのまま使用してマッチングを行った場合 [12] と，Visual Words を使用したマッチングを行った場合の計算時間は表 1 である．ただし，動画の SIFT 特徴量は予め抽出しており，それぞれのフレーム数は 761 である．また，ノイズを減らすために RANSAC[6] を使用している．

表 1 の結果から，Visual Words を使用したマッチングは，SIFT 特徴量をそのまま使用した場合より，マッチングを高速に行うことが分かった．

5.2 動画検索

我々は YouTube 上にある F1 カー，馬，飛行機の 3 種類の物体の動画でシステムの実験を行った．3 種類の物体の動画をダウンロードした後，1 シーンごとに動画を分割した．各物体ごとに検索対象は 20 本の動画である．それらの動画の全フレームから SIFT 特徴量を抽出し，Visual Words マッチングで使用する辞書とそれぞれの Visual Words データ，SIFT 追跡データを作成した．各物体の動画には 20 種類のシーンが存在する．

図 8 の 1 列目は静止画と矢印を入力したユーザインタフェースの画面，右の 3 列は各入力に対するシステムの検索結果である．実験に使用した動画のフレーム数と検索時間は表 2 に示す通りである．ここで，特徴量の抽出は 1 枚の入力静止画に対して 1 回だけ行えば良いので，2 回目以降の検索は抽出したデータを読み込むようにしているため，1 回目より高速になる．(クラスタ数　F1 カー 2500，馬・飛行機 500)

図 8 の 1 行目の (a) は F1 カーの動画を検索した結果である．この行では，ユーザが正面向きに移動する F1 カーのシーンを検索しようと，右向きの F1 カーと 2 本の緑矢印を入力した．それを想定して，1 本目の矢印は F1 カーの前方を指定して下に向けて入力する．続けて，2 本目の矢印は F1 カーの後方

図 8. 検索結果 (a) 正面向きに移動する F1 カーのシーン (b) 左向きに移動する F1 カーのシーン (c) 右向きに移動する馬のシーン (d) 正面向きに移動する馬のシーン (e) 右向きに飛行する飛行機のシーン

表 2. 検索時間

	F1 カー	馬	飛行機
フレーム数	3941	7018	3741
特徴量抽出 (秒)	7.08	7.04	1.41
特定 (秒)	3.29	2.86	1.64
追跡 (秒)	0.97	2.29	0.36
合計 (秒)	11.34	12.19	3.41

を指定し上に向けて入力する．検索結果は入力静止画とユーザが指定した矢印に近似した上位 3 フレームである．上位 3 フレームの中に，ユーザが検索しようとした正面向きに移動する F1 カーのシーンが含まれている．以降の行でもユーザが視聴したいと思ったシーンが上位 3 フレームの中に含まれている．

また，図 1 も飛行機について検索した結果である．これが我々の技術の限界を示している．ユーザは図 9 のような左上向きの飛行機のシーンを発見しようと 2 本の緑矢印を描画した．検索された結果は全てで左上を向いており，ユーザの指定が満たされている．

図 9. 左上へ飛行する飛行機

しかし，飛行機はその前方と後方を軸として回転しているシーンばかりである．ユーザが図 9 のように飛行機が回転していないシーンを望む場合であっても，我々のシステムを使用して，飛行機が回転しているシーンを排除することは困難である．

6 まとめ

本論文では，静止画内の物体に 2 本の矢印を描画することでシーンの検索ができる新しいユーザインタフェースを提案した．実験において，F1 カー，馬，飛行機といった 3 種類の物体において検索できることをを確認した．

今後として，解決すべき課題がある．まず，実験においてユーザが検索しようとしたシーンとは違う意図しないシーンが存在したという問題がある．SIFT 特徴量をそのまま使用し，Particle Video を使用して検索を行った場合 [12] では意図しないシーンはあまり出てこなかった．このような精度低下の原因は，使用するアルゴリズムを変更したからである．しかしながら，フレーム検索では 17 倍，フレーム追跡では 1.5 倍以上高速になったため，より多くの動画を以前よりも時間をかけること無く検索対象にすることができる．より多くの動画を使用することで，シーンの種類が増え，ユーザの意図に近い動画が増えることにより，意図しないシーンが上位に来なくなるのではないかと考える．

次に，色情報を使用していないという点がある．SIFT 特徴量のマッチングや追跡において，色情報を使用していない．そのため，例えば，F1 カーの動画を大量に用意した場合，形は同じだが色が異なる F1 カーが検索結果として表示される可能性がある．この場合のシステムの対応について，色の違う車を許容して検索結果を出すのか，色の違いを許さず色情報を使用する方法を検討するのか等を考える必要がある．

3 点目に，検索する対象が剛体で特徴的な部分が存在しないとうまく働かない点がある．F1 カーや飛行機といった乗り物では色や模様が異なることがあるが，形状についてあまり変わらない．また，タイヤやロゴ，翼等といったように特徴的な部分が存在する．馬は足を動かし形状が変化するが，馬衝や鞍といった馬具を付けられてこれが特徴的な部分となっている．一方で，小動物や魚を検索対象にして実験を行ったが，うまく機能しなかった．その原因は，動画内で形状が変化し，特徴的な部分が存在しないからである．形状が変化すると，SIFT 特徴量を使用したマッチングがうまく機能しない．また，特徴的な部分がないと正確なマッチングも判定するのが困難である．そのような物体に対しても検索可能にするためには，使用する特徴量について検討する必要がある．

参考文献

[1] S. Agarwal, N. Snavely, I. Simon, S. Seitz, R. Szeliski: Building rome in a day. In: ICCV 2009. (2009) 72-79

[2] S. Battiato, G. Gallo, G. Puglisi, and S. Scellato: Sift features tracking for video stabilization. In Proc. of International Conference on Image Analysis and Processing (2007) 825-830

[3] R. Datta, D. Joshi, J. Li, J.Z. Wang: Image retrieval: Ideas, influences, and trends of the new age. ACM Comput. Surv. 40(2) (2008) 5:1-5:60

[4] P. Dragicevic, G. Ramos, J. Bibliowitcz, D. Nowrouzezahrai, R. Balakrishnan, K. Singh: Video browsing by direct manipulation. In: Proc. of CHI ’08. (2008) 237-246

[5] J. M. Frahm, M. Pollefeys, S. Lazebnik, C. Zach, D. Gallup, B. Clipp, R. Raguram, , C. Wu, T. Johnson: Fast robust large-scale mapping from video and internet photo collections. ISPRS Journal of Photogrammetry and Remote Sensing 65(6) (2010) 538-549

[6] M. A. Fischler, R. C. Bolles: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. In: Comm. of the ACM 24(6) (1981) 381-395.

[7] A. Girgensohn, D. Kimber, J. Vaughan, T. Yang, F. Shipman, T. Turner, E. Rieffel, L. Wilcox, F. Chen, T. Dunnigan: Dots: support for effective video surveillance. In: Proc. of ACM Multimedia. (2007) 423-432

[8] D. B. Goldman, C. Gonterman, B. Curless, D. Salesin, S. M. Seitz: Video object annotation, navigation, and composition. In: Proc. UIST 2008. (2008) 3-12

[9] T. Karrer, M. Weiss, E. Lee, J. Borchers: Dragon: a direct manipulation interface for frame-accurate in-scene video navigation. In: Proc. of CHI ’08. (2008) 247-250

[10] D. Kimber, T. Dunnigan, A. Girgensohn, F. Shipman, T. Turner, T. Yang: Trailblazing: Video playback control by direct object manipulation. In: IEEE International Conference on Multimedia and Expo. (2007) 1015-1018

[11] D. G. Lowe: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision 60 (2004) 91-110

[12] M. Okabe, Y. Kawate, K. Anjyo, R. Onai: Video Retrieval based on User-Specified Appearance and Application to Animation Synthesis. In: Proc. of MMM (2013)

[13] P. Sand, S. Teller: Particle video: Long-range motion estimation using point trajectories. In: Proc. of CVPR ’06. (2006) 2195-2202

[14] S. N. Sinha, J. M. Frahm, M. Pollefeys, Y. Genc: Gpu-based video feature tracking and matching. In: Workshop on Edge Computing Using New Commodity Architectures (2006)

[15] J. Sivic, A. Zisserman: Video google: a text retrieval approach to object matching in videos. In: ICCV. (2003) 1470-1477

[16] N. Snavely, S. M. Seitz, R. Szeliski: Photo tourism: exploring photo collections in 3d. In: ACM SIGGRAPH 2006 Papers. (2006) 835-846

[17] J. Tompkin, K. Kim, J. Kautz, C. Theobalt: Videoscapes: Exploring sparse, unstructured video collections. In: ACM Transactions on Graphics (Proc. of SIGGRAPH). (2012)