

部分的に局所性を保持するハッシュ関数を用いた 画像列からの高頻度パターン検出

小川原光一 (九州大学) 田邊康史 (九州大学) 倉爪亮 (九州大学) 長谷川勉 (九州大学)

Detecting Frequent Patterns in Video using Partly Locality Sensitive Hashing

*Koichi OGAWARA (Kyushu University), Yasufumi TANABE (Kyushu University),
Ryo KURAZUME (Kyushu University), Tsutomu HASEGAWA (Kyushu University)

Abstract— Frequent patterns in video are useful clues to learn previously unknown events in an unsupervised way. This paper presents a novel method that detects relatively long variable-length frequent patterns in video efficiently. The major contribution of the paper is that Partly Locality Sensitive Hashing (PLSH) is proposed and used as a sparse sampling method so as to detect frequent patterns faster than the conventional method using LSH. The proposed method was evaluated by detecting frequent everyday whole body motions in video.

Key Words: Frequent Pattern Mining, Locality Sensitive Hashing, Unsupervised Learning, Video Analysis

1. はじめに

事前に学習された人間行動を動画画像から検出する研究 [1] や異常 (未学習) 行動を動画画像から検出する研究 [2] は、動画画像検索や広域監視の自動化など様々な応用が考えられることから盛んに取り組まれてきた。

このとき、人間行動のデータベースを事前に作成する必要があるが、これまでは大量の動画画像から人手で作成する方法が一般的であった。そのため、この作業を自動化するために、動画画像中に高頻度で発生する人間行動を自動検出してデータベースの作成に利用するための研究が行われてきた。

これらの研究は2つに大別される。1つは動画画像から時空間パッチを検出してこれをクラスタリングする方法 [3] であるが、比較的長いパターンを扱うことができないという問題点がある。もう1つは動的計画方法に基づく方法 [4] であるが、比較的長いパターンを扱うことができる反面、画像数 N に対して計算量が $O(N^2)$ になるという問題点があり、長時間の動画画像を取り扱う目的には不向きであった。

そこで、計算時間を短縮するために、Meng らは Locality Sensitive Hashing (LSH) [5] を用いて各時刻ごとに近傍の類似データを探索し、これらを接続することによって、モーションキャプチャデータから非線形伸縮する高頻度パターンを計算量 $O(N^{1+1/\alpha})$ で検出する手法を提案した [6]。この手法では、各時刻ごとに LSH を用いた近傍探索を行っているが、対象が連続した時系列データである場合には、時間方向に隣接するデータ点における探索結果には重複が大きい。

本稿では、主に [6] の問題点を解決する手法を提案する。提案手法では、毎時刻近傍探索を行わずに、近傍データをリンクリストで管理してこれを時間軸に沿って更新していく。しかし、リンクリストの更新は不完全であることから、各時刻ごとに近傍点を疎にサンプリングすることによって、リンクリストを修正する。

疎なサンプリング法として、本研究では近似最近傍

探索法の一つである Partly Locality Sensitive Hashing (PLSH) を提案し、LSH を使用した方法と比較して少ない計算時間で高頻度パターンが検出できることを示す。

2. 提案手法の概要

Fig.1 に、 d 次元時系列データを 2 次元に投影して表示した例を示す。もし時刻 t のデータ点 $o(t)$ がある高頻度パターン群に属している場合、このデータ点の近傍に他の類似パターンも存在することになる。そのため、近傍パターン数の多いデータ点が連続する区間は高頻度パターンのよい候補になると考えられる。

そこで、データ点 $o(t)$ を中心としたデータ密度を

$$D(t) = \sum_{i \in S(t)} \|o(i) - o(i+1)\|$$

$$\text{where } S(t) = \{i; \|o(i) - o(t)\| \leq R\}$$

と定義し、Table 1 に示す方法でこれを各時刻ごとに効率よく計算することによって高頻度パターンの有無を評価することを考える。

このとき、新規セグメントと分断セグメントは直ちには検出できないため、新規セグメントについては現超球内で近傍点を疎にサンプリングし、分断セグメントについては一定時間 (T_{delay}) 前の超球内で近傍点を疎にサンプリングすることによって、それぞれを発見することを考える。ここで、超球内のデータをランダムかつ効率よくサンプリングすることが重要になるが、これを実現する方法として Partly Locality Sensitive Hashing (PLSH) を 3. 章で提案する。

3. Partly Locality Sensitive Hashing

Partly Locality Sensitive Hashing (PLSH) は近似最近傍探索法の一つであり、局所性を保持するハッシュ関数を用いる Locality Sensitive Hashing (LSH) [5] に局所性を保持しないハッシュ関数を追加して拡張したものである。

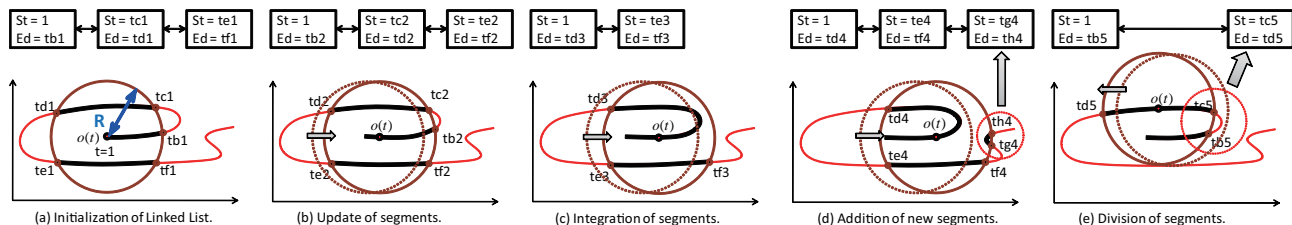


Fig.1 Calculation of data density

Table 1 Algorithm to find frequent patterns

1. 時刻 $t = 1$:
超球内のセグメントの情報を保持するリンクリストを初期化 (Fig.1(a))
2. 時刻 $t = 2$ から N :
リンクリストを更新 (Fig.1(b),(c))
PLSH を用いて新規セグメントを検出しリンクリストへ追加 (Fig.1(d))
PLSH を用いて分断セグメントを検出しリンクリストを更新 (Fig.1(e))
3. 時刻 $t = 1$ から N :
大域最適化に基づき全高頻度パターンを検出

3.1 部分的に局所性を保持するハッシュ関数

PLSH の枠組みでは、ハッシュ関数 $g_l(p)$ ($1 \leq l \leq L$) は以下のように定義される。

$$g_l(p) = \langle hs_{l,1}(p), \dots, hs_{l,K_s}(p), hi_{l,1}(p), \dots, hi_{l,K_i}(p) \rangle$$

ここで、 $hs(p)$ は入力値の局所性を保持する任意のハッシュ関数 $hs: R^d \rightarrow U$ を表し、 $hi(p)$ は入力値の局所性を保持しない任意のハッシュ関数 $hi: R^d \rightarrow U$ を表す。例えば線型写像に基づくハッシュ関数として、以下で定義するハッシュ関数を用いることができる。

$$hs(p) = \lfloor (a_s \cdot p + b_s) / w_s \rfloor$$

$$hi(p) = \lfloor (a_i \cdot p + b_i) \rfloor \bmod w_i$$

近傍探索の手順は、まず探索対象となる全てのデータ点に L 個のハッシュ関数 $g_l(p)$ を適用して L 個のハッシュ値を計算し、対応する L 個のバケットにそれぞれデータ点を格納する。入力データ点 p が与えられると、 L 個のハッシュ関数 $g_l(p)$ から L 個のハッシュ値を計算し、対応する L 個のバケット内のデータ点群を調べる。

3.2 PLSH を用いた疎なサンプリング法

2. 章で述べたとおり、各時刻ごとに $o(t)$ を中心とした超球内のデータ点をランダムかつ効率よくサンプリングしたい。もし Meng らの手法 [6] のようにデータ点の探索に LSH を使う場合は、バケットの大きさを表す w は超球内のデータ点を探索できるように半径 R に設定するため固定値となり、そのため Fig.2(a) に示

すようにデータ数が増えるとバケット内のデータ数も増加し実際の計算量は $O(N^2)$ に近くなる。

このようにバケットの大きさが固定されている場合に PLSH は有用である。PLSH を使う場合、本研究では入力データ値を $d+1$ 次元ベクトル $(p_1, \dots, p_d, t)^T$ で表現する。このうち最初の d 次元はデータを表し、最後の 1 次元はそのデータが観測された時刻を表す。

ハッシュ関数 $g_l(p)$ は、 K_s 個の局所性を保持するハッシュ関数と 1 個の局所性を保持しないハッシュ関数により構成する。 K_s 個の局所性を保持するハッシュ関数については、射影ベクトルである a_s の最後の要素は常に 0 とする。局所性を保持しないハッシュ関数については、射影ベクトルである a_i の最初の d 個の要素は常に 0 とし、最後の要素は 1 とする。これは、 w_i (LIH width) で決定される数のハッシュ空間にデータ点を分散することに相当する。

これにより、Fig.2(b) に示すように、時間軸上で近傍に存在するデータ点同士では、互いに重複するデータを持たない独立したハッシュ空間を探索することになる。そのため、新規データ点（もしくは分断データ点）が発見された場合には過去に遡ってデータ密度を修正することにより、新規セグメント（もしくは分断セグメント）上のデータ点が発見される確率を LSH と同じに保ったまま、1 つのバケット内に格納されるデータ数を大きく削減することが可能になり、探索時間を短縮することができる。バケットあたりのデータ数は $\frac{1}{w_i}$ となり、 w_i は検出したいパターンの時間長より大きくならない範囲で決めればよい。

LSH の場合も、Fig.2(c) に示すように、時系列データをあらかじめある間隔 w_r (Reduction width) で間引いておくことによって、同等のデータ削減効果を得ることは可能である。しかし、PLSH の場合は、Fig.2(d) に示すように、時系列データを間隔 w_r で間引いた上でさらに時間軸方向にデータを分散することによって、 $\frac{1}{w_i \cdot w_r}$ のデータ削減効果を得ることができる。

4. 高頻度パターンの検出

時系列データ $O = (o(1), \dots, o(N))$ が与えられたときに、未知の高頻度パターンを検出する問題を以下の 2 段階処理によって解く。

1. データ密度に基づく高頻度パターンの検出
2. 高頻度パターンのクラスタリング

4.1 データ密度に基づく高頻度パターンの検出

まず、パターンの種類とは無関係に、高頻度パターンである可能性の高い区間を O から検出する。この問題を、 $X = (x_1, \dots, x_N)$, $x_t \in \{1 = \text{高頻度}, 0 = \text{低頻}$

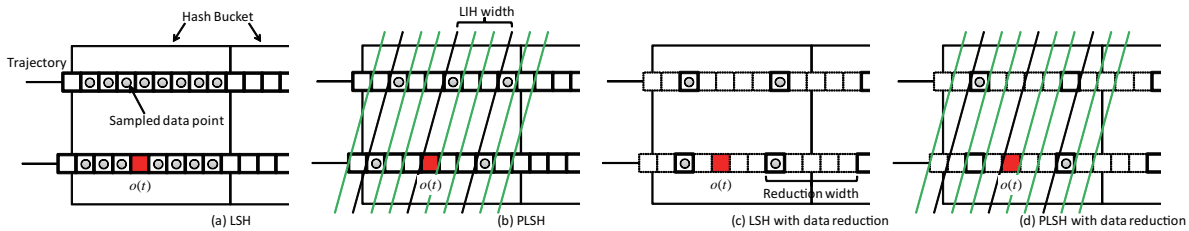


Fig.2 Difference between LSH and PLSH

度}のように各時刻に2値ラベルを割り当てる組み合わせ最適化問題として定式化する。

そして、以下のエネルギー関数 $E(O, X)$ を最小化することによって X を求める。

$$E(O, X) = E_v(O, X) + E_d(O, X) + E_s(X) \quad (1)$$

$$E_v(O, X) = \sum_t -\log(1 - \exp(-\frac{|\dot{o}_{x_t}(t)|}{\langle |\dot{o}_{x_t}(t)| \rangle}))$$

$$E_d(O, X) = \sum_t -\log(1 - \exp(-\frac{D_{x_t}(t)}{\langle D_{x_t}(t) \rangle}))$$

$$E_s(X) = \sum_t T(x_t \neq x_{t+1}) \cdot C_{\text{smooth}}$$

ただし $\langle |\dot{o}_{x_t}(t)| \rangle$ は $|\dot{o}_{x_t}(t)|$ の平均、 $\langle D_{x_t}(t) \rangle$ は $D_{x_t}(t)$ の平均、 C_{smooth} は定数、 $T(s)$ は $T(\text{true}) = 1, T(\text{false}) = 0$ である。

式(1)は一次のマルコフ性を満足しており、動的計画法によって解析的に最小化することができる。

4.2 高頻度パターンのクラスタリング

前節で検出されたパターンには異なる種類の動作が混在しているため、凝集的クラスタリングによって同じ種類の動作を統合する。

5. 実験

提案手法を評価するため、画像列からの高頻度な全身運動を検出する実験を行った。実験では、新規セグメントおよび分断セグメントのサンプリング法として、LSHに基づく方法と提案するPLSHに基づく方法との比較を行った。なお以降の実験では、LSHおよびPLSHのいずれも L は8に K_s は3に設定した。また、全ての計算はXeon 3.0GHzの計算機上で行った。

5.1 実験条件

本実験では、Fig.3に示す3種類の全身運動パターンのうちByeが5回、Stretchが6回、Stand-upが5回含まれた動画画像を用意し、これから高頻度パターンを検出した。また、画像中に局所的に現れる時系列パターンを位置に不変な特徴量で記述するため、高次局所自己相関特徴量(CHLAC)[7]を画像のデータ表現として用いた。

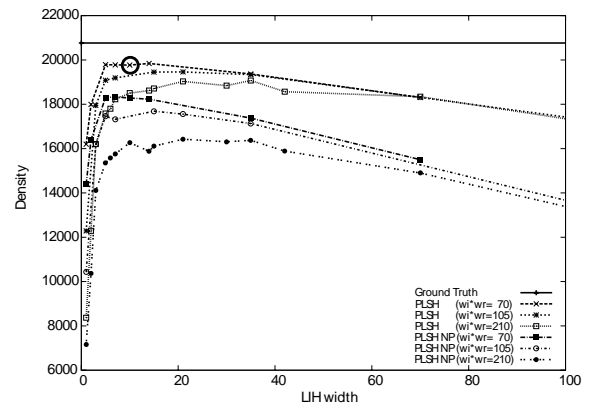


Fig.4 Estimation of data density

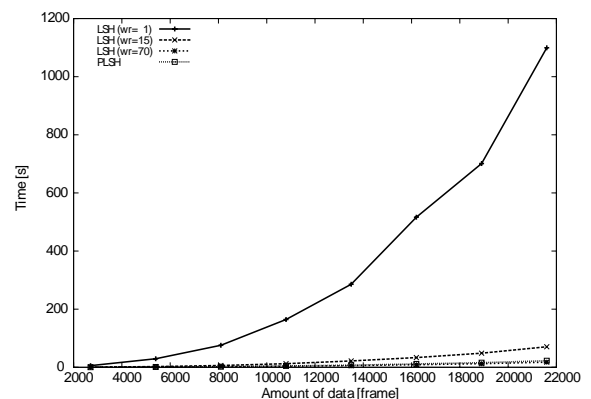


Fig.5 Computational time v.s. amount of data

5.2 データ密度推定とサンプリング方法の評価

3.2節で述べたように、PLSHでは w_r の間隔で時系列データを間引き、かつ w_i (LIH width) で決定される数のハッシュ空間にデータ点を分散することによって、ハッシュバケット内のデータ数を $\frac{1}{w_i \cdot w_r}$ に削減し、疎なサンプリングを実現する。計算時間は $\frac{1}{w_i \cdot w_r}$ に比例するため、データ密度の推定精度の低下を抑えた最適な w_i, w_r を決定する必要がある。

異なる w_i, w_r の組に対してデータ密度を計算した結果をFig.4に示す。グラフの縦軸はデータ密度の総和を表し、横軸は w_i を表す。グラフの折れ線は、 $w_i \cdot w_r$ の値が等しい点を結んだものである。NP (Non Propagation) は、新規セグメントもしくは分断セグメントがサンプリングによって発見された後に、過去に遡ってデータ密度を修正しない場合の結果を示す。

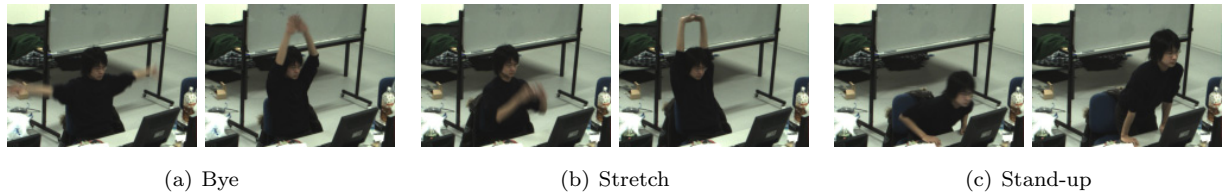


Fig.3 3 whole body motions to be detected

Table 2 Evaluation of Dataset [2700 frames]

Action	Bye	Stretch	Stand-up	False Positive	False Negative	Precision	Recall	Time [msec]
Presented #	5	6	5					
LSH ($w_r=1$)	5.00	5.00	5.00	0.00	1.00	1.00	0.94	5807
LSH ($w_r=15$)	5.00	5.00	5.00	0.00	1.00	1.00	0.94	512
LSH ($w_r=70$)	5.00	4.90	0.00	0.00	6.10	1.00	0.62	198
PLSH	5.00	5.00	5.00	0.00	1.00	1.00	0.94	224

折れ線の一番左側の点が, w_i (LIH width) が1の場合, つまり LSH を使用した場合に対応する. グラフより, $w_i \cdot w_r$ を一定にした場合には, w_i を1から大きくするにつれてデータ密度が厳密解に近くなり, さらに大きくすると厳密解から離れていくことが分かる. これは, w_i と w_r のいずれか片方が大きな値を取ると, サンプリングの間隔が広くなり十分なサンプリングがなされないことを意味する.

この結果をふまえ, 以降の実験ではグラフに丸で示したように PLSH の w_i は10に w_r は7に設定した.

5.3 高頻度パターンの検出結果

(1) LSH (間引き無し, $w_r = 1$), (2) LSH (PLSH と同等のデータ密度, $w_r = 15$), (3) LSH (PLSH と同等の削減率, $w_r = 70$), (4) PLSH の4つの手法に基づき動画から高頻度パターンを検出した結果を Table 2 に示す. ハッシュ関数のパラメータは乱数で決まるため, それぞれ10回試行を行った平均を計算した.

(1) と (2) では PLSH とほぼ同等の検出結果が得られたが, データの削減率で劣るため計算時間は長くなる. 一方, (3) は PLSH と比べてやや計算時間が短い, 検出結果は特に再現率の点で大きく劣っている. この結果から, PLSH は LSH と比較して少ないサンプリング数で高頻度パターンを検出できることが分かる.

5.4 データ量の増加に対する計算時間の評価

異なる量のデータに対して前節の4手法を比較した結果を Fig.5 に示す. 本実験では, 動画にノイズを加えて単純に連結する方法によって異なるデータ量の動画を10個作成した. グラフより, ハッシュバケット内のデータ数を PLSH に基づき削減することによって, 計算時間が大きく削減されたことが分かる.

6. まとめ

本稿では, 時系列データから非線形伸縮する比較的長い高頻度パターンを効率よく検出する方法を提案した.

局所性を保持するハッシュ関数と局所性を保持しないハッシュ関数を組み合わせた Partly Locality Sensitive

Hashing (PLSH) を提案し, (1) PLSH に基づくハッシュ空間の分割と (2) データ列の間引きの2つの効果により, 従来の LSH を用いた方法と比較して高頻度パターンの高速な検出を可能にした.

謝辞

本研究は, 科学研究費補助金若手 (B)(21700224) および科学技術総合推進費補助金「若手研究者の自立的な研究環境整備促進」の補助を受けている.

参考文献

- [1] Juan Carlos Niebles and null Li Fei-Fei. A hierarchical model of shape and appearance for human action classification. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition: CVPR*, pp. 1–8, 2007.
- [2] Oren Boiman and Michal Irani. Detecting irregularities in images and in video. *International Journal of Computer Vision: IJCV*, Vol. 74, No. 1, pp. 17–31, 2007.
- [3] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. In *Proc. of BMVC*, 2006.
- [4] Seiichi Uchida, Akihiro Mori, Ryo Kurazume, Rinichiro Taniguchi, and Tsutomu Hasegawa. Logical dp matching for detecting similar subsequence. In *Proc. of Asian Conference of Computer Vision*, 2007.
- [5] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Proc. of the twentieth annual Symposium on Computational Geometry*, pp. 253–262, 2004.
- [6] Jingjing Meng, Junsong Yuan, Mat Hans, and Ying Wu. Mining motifs from human motion. In *Proc. of EUROGRAPHICS'08*, 2008.
- [7] T. Kobayashi and N. Otsu. Action and simultaneous multiple-person identification using cubic higher-order local auto-correlation. In *Proc. Int. Conference on Pattern Recognition: ICPR*, pp. 741–744, 2004.