Multiple Feature Hashing for Real-time Large Scale Near-duplicate Video Retrieval

Jingkuan Song*, Yi Yang**, Zi Huang*, Heng Tao Shen*, Richang Hong***

*University of Queensland, Australia
**Carnegie Mellon University, USA
***Hefei University of Technology, China

1 December 2011

DEFINITIONS OF NEAR-DUPLICATE

Identical or approximately identical videos close to the exact duplicate of each other, but different in file formats, encoding parameters, photometric variations (colour, lighting changes), editing operations (caption, logo and border insertion), different lengths, and certain modifications (frames add/remove).	Wu et al. TMM 2009
Clips that are similar or nearly duplicate of each other, but appear differently due to various changes introduced during capturing time (camera view point and setting, lighting condition, background, foreground, etc.), transformations (video format, frame rate, resize, shift, crop, gamma, contrast, brightness, saturation, blur, age, sharpen, etc.), and editing operations (frame insertion, deletion, swap and content modification).	Shen et al. VLDB 2009
Videos of the same scene (e.g., a person riding a bike) varying viewpoints, sizes, appearances, bicycle type, and camera motions. The same semantic concept can occur under different illumination, appearance, and scene settings, just to name a few.	Basharat et al. CVIU 2008
NDVs are approximately identical videos that might differ in encoding parameters, photometric variations (colour, lighting changes), editing operations (captions, or logo insertion), or audio overlays. Identical videos with relevant complementary information in any of them (changing clip length or scenes) are not considered as NDVs. Two different videos with distinct people, and scenarios were considered to be NDVs if they shared the same semantics and none of the pairs has additional information.	Cherubini et al.ACM MM 2009

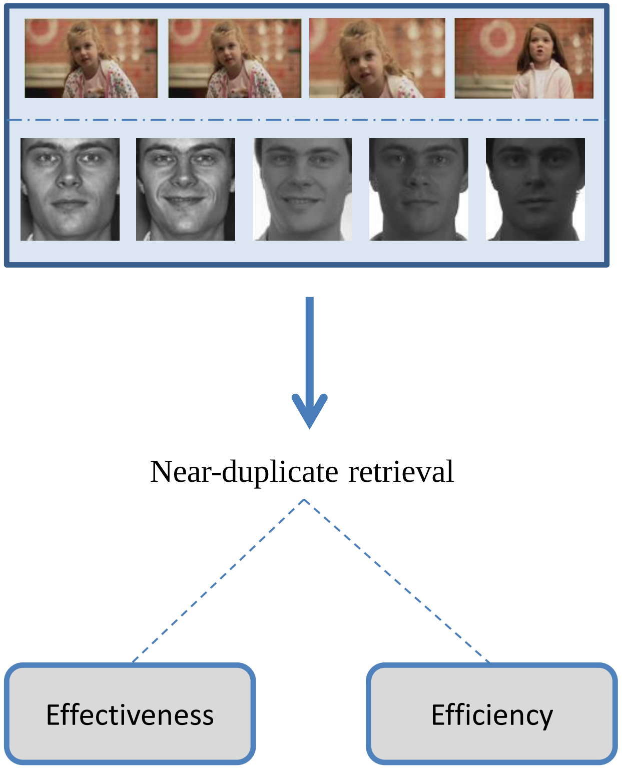
- Variants: duplicate, copy, (partial) near-duplicate

APPLICATIONS

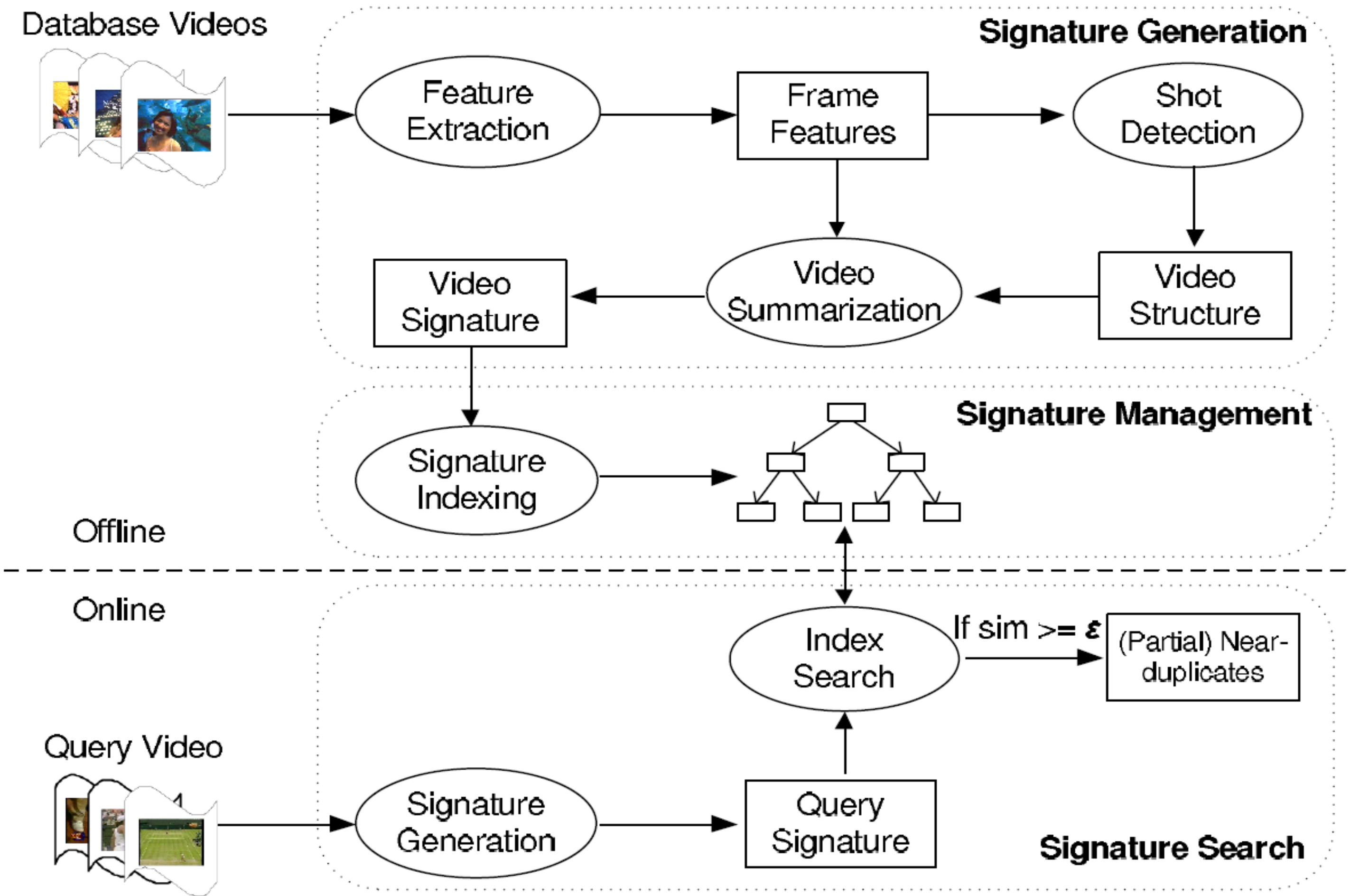
- Copyright protection
- Database cleansing
- Recommendation
- Video Monitoring
- Video Thread Tracking
- Multimedia reranking
- Multimedia tagging
- Border Security, …

OBJECTIVE

- Given a large-scale video dataset and a query video, find its near-duplicate videos in real-time

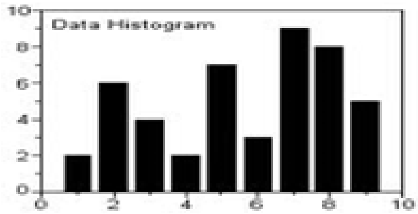


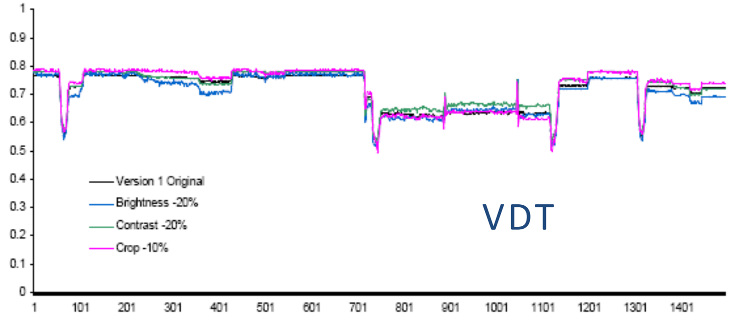
A GENERIC FRAMEWORK



SIGNATURES

- Frame-level Local Signatures
  - SIFT Feature, Local Binary Pattern (LBP), etc
- Frame-level Global Signatures
  - Color Histogram, Bag of Words (BoW), etc
- Video-level Global Signatures
  - Accumulative Histogram
  - Bounded Coordinate System (BCS), etc
- Spatio-temporal Signature
  - Video Distance Trajectory (VDT),
  - Spatio-temporal LBP (SP_LBP), etc



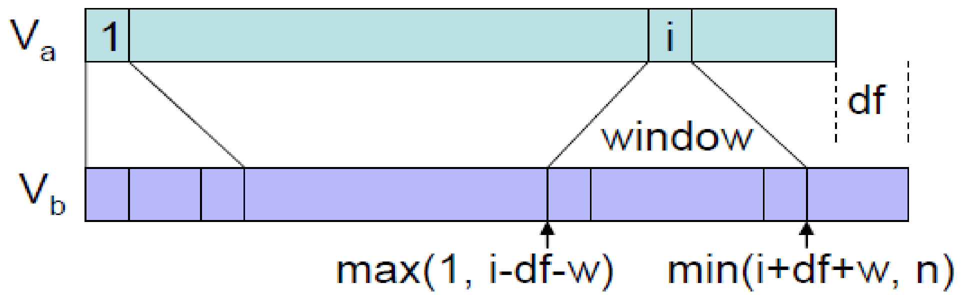


INDEXING METHODS

- Tree Structures
  - R-tree, M-tree, etc
- One-dimensional Transformation
  - iDistance, Z-order, etc
- Hashing
  - LSB-forest
  - Locality Sensitive Hashing (LSH)
  - Spectral Hashing
  - Self-taught Hashing (STH), etc

RELATED WORK – A HIERARCHICAL APPROACH

- Wu et al. outlines ways to filter out the near-duplicate video using a hierarchical approach
- Initial triage is fast performed using compact global signatures derived from colour histograms
- Only when a video cannot be clearly classified as novel or near-duplicate using global signatures, a more expensive local feature based near-duplicate detection is then applied to provide accurate near-duplicate analysis through more costly computation



Matching window for keyframes between two videos

- X. Wu, A. G. Hauptmann, and C.-W. Ngo. Practical elimination of near-duplicates from web video search. In ACM Multimedia, pages 218–227, 2007.

RELATED WORK – SPATIOTEMPORAL FEATURE APPROACH

- L. Shang introduce a compact spatiotemporal feature to represent videos and construct an efficient data structure to index the feature to achieve real-time retrieving performance.

(a)

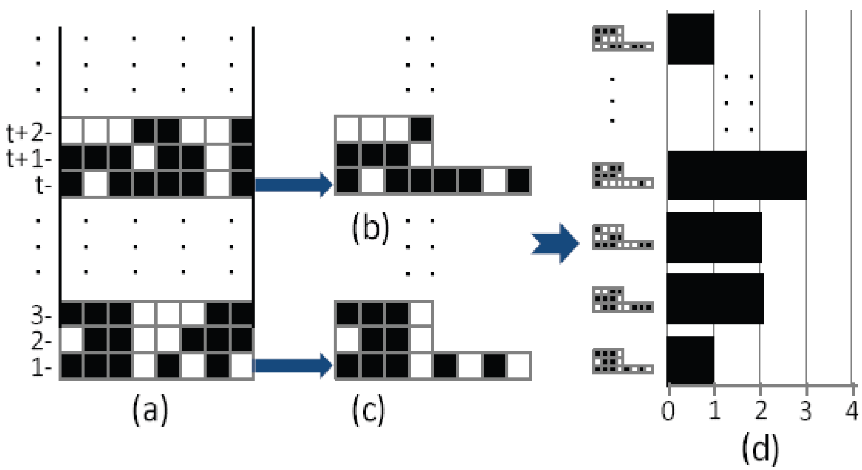
199	176	181
178	138	136
86	107	56

(b)

$$\begin{bmatrix} 8 & 5 & 7 \\ 6 & 4 & 3 \\ 1 & 2 & 0 \end{bmatrix}$$

(c)

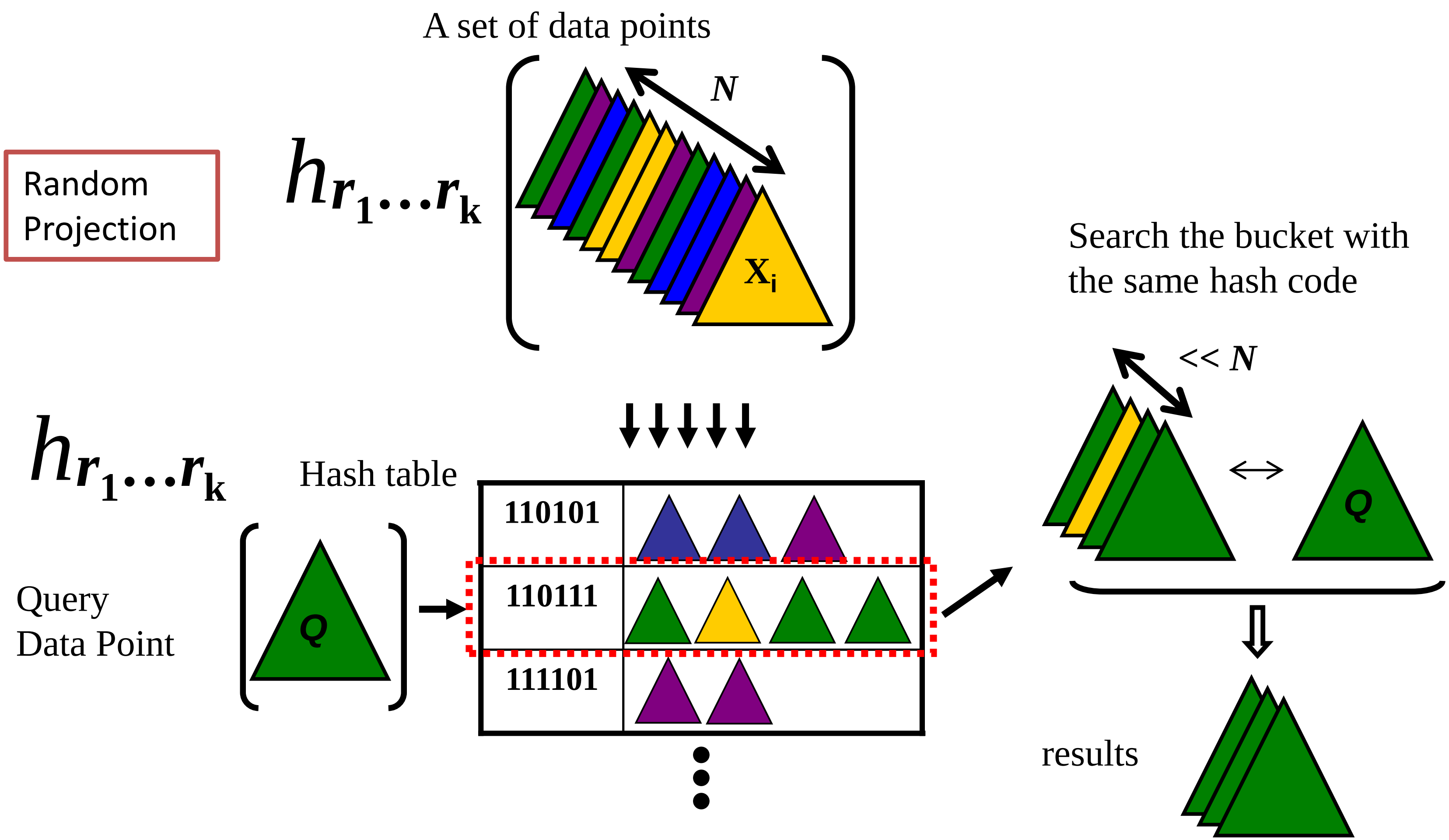
- Ordinal measure can be rewritten in the form a 36-dimensional (C_9^2) binary feature vector



- CE-based Spatiotemporal Feature
- LBP-based Spatiotemporal Feature

- L. Shang, L. Yang, F. Wang, K.-P. Chan, and X.-S. Hua. Real-time large scale near-duplicate web video retrieval. In ACM Multimedia, pages 531–540, 2010.

RELATED WORK - LOCALITY SENSITIVE HASHING

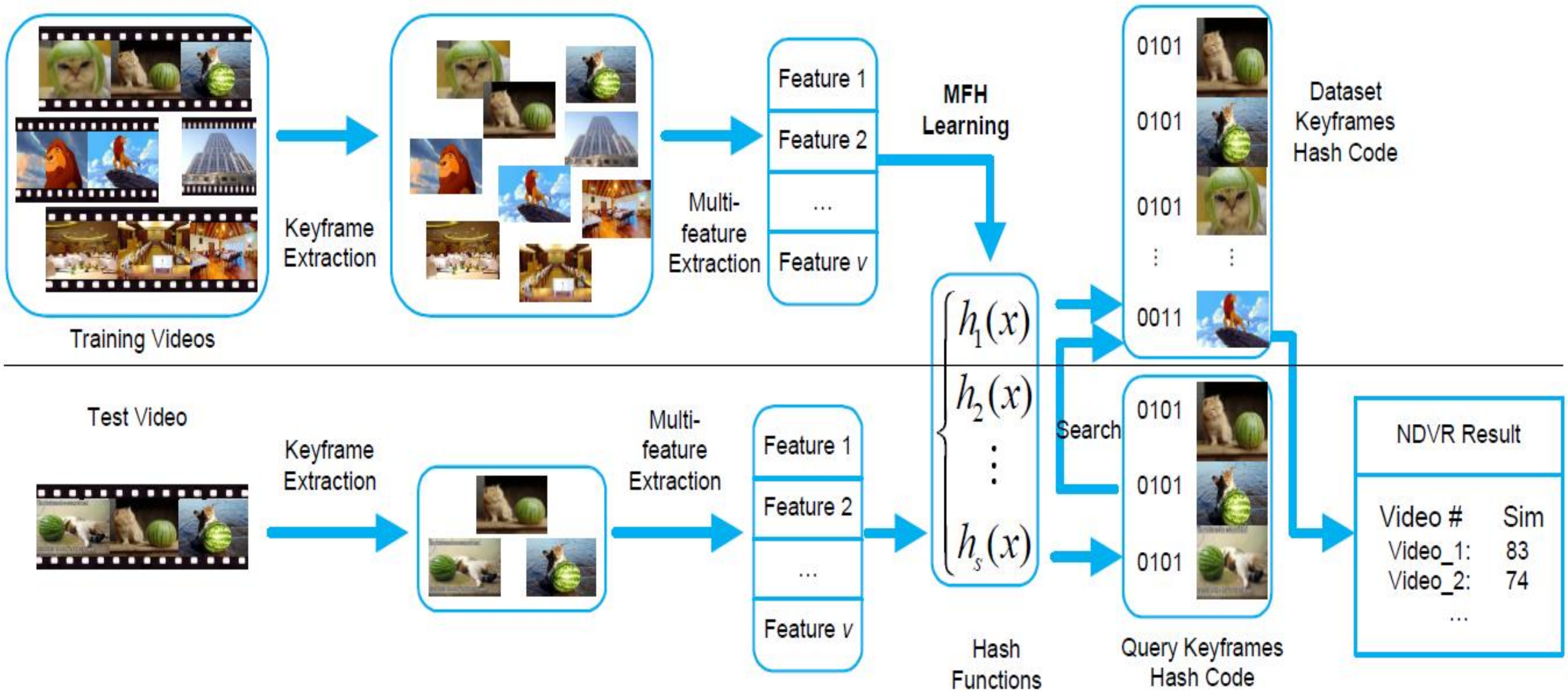


- M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In Symposium on Computational Geometry, pages 253–262, 2004.

OUR PROPOSAL - MULTIPLE FEATURE HASHING (MFH)

- Problems & Motivations
  - Single feature may not fully characterize the multimedia content
  - Exiting NDVR concerns more about accuracy rather than efficiency
- Methodology
  - We present a novel approach - Multiple Feature Hashing (MFH) to tackle both the accuracy and the scalability issues
  - MFH preserves the local structure information of each individual feature and also globally consider the local structures for all the features to learn a group of hash functions

MFH – THE FRAMEWORK



MFH – LEARNING HASHING FUNCTIONS

Target: Similar items have similar hash codes

$$(A^g)_{pq} = \begin{cases} 1, & \text{if } (x^g)_p \in \mathcal{N}_k((x^g)_q) \text{ or } (x^g)_q \in \mathcal{N}_k((x^g)_p) \\ 0, & \text{else} \end{cases}$$

KNN Graph

where $\mathcal{N}_k(\cdot)$ is the k-nearest-neighbor set and $1 \le (p,q) \le n$.

1: Sum the distance of nearby hash codes

$$\sum_{p,q=1}^{n} (A^g)_{pq} \left\| (y^g)_p - (y^g)_q \right\|_F^2$$

2: Sum the distance in each feature type, and globally consider the overall hash code

$$\sum_{g=1}^{v} \sum_{p,q=1}^{n} (A^g)_{pq} \left\| (y^g)_p - (y^g)_q \right\|_F^2 + \gamma \sum_{g=1}^{v} \sum_{t=1}^{n} \| y_t - (y^g)_t \|_F^2$$

3: Add the regression model to learn the hash functions

$$\begin{aligned} \min_{Y, Y^g, W, b} \quad & \sum_{g=1}^{v} \sum_{p,q=1}^{n} (A^g)_{pq} \left\| (y^g)_p - (y^g)_q \right\|_F^2 \\ & + \gamma \sum_{g=1}^{v} \sum_{t=1}^{n} \| y_t - (y^g)_t \|_F^2 \\ & + \alpha \sum_{l=1}^{s} \left(\sum_{t=1}^{n} \| h_l(x_t) - y_{tl} \|_F^2 + \beta \Omega(h_l) \right) \\ \text{s.t.} \quad & \begin{cases} y_t \in \{-1,1\}^s, (y^g)_t \in \{-1,1\}^s \\ YY^T = I \end{cases} \end{aligned} \tag{4}$$

Reformulating

The final objective function

$$\min_{YY^T = I} tr(Y^T D Y).$$

$$\begin{aligned} D &= \sum_{g=1}^{v} \left(C^g L^g C^g + \gamma (I - C^g)^2 \right) + \alpha B \\ &= \gamma \sum_{g=1}^{v} (I - C^g) + \alpha B \end{aligned}$$

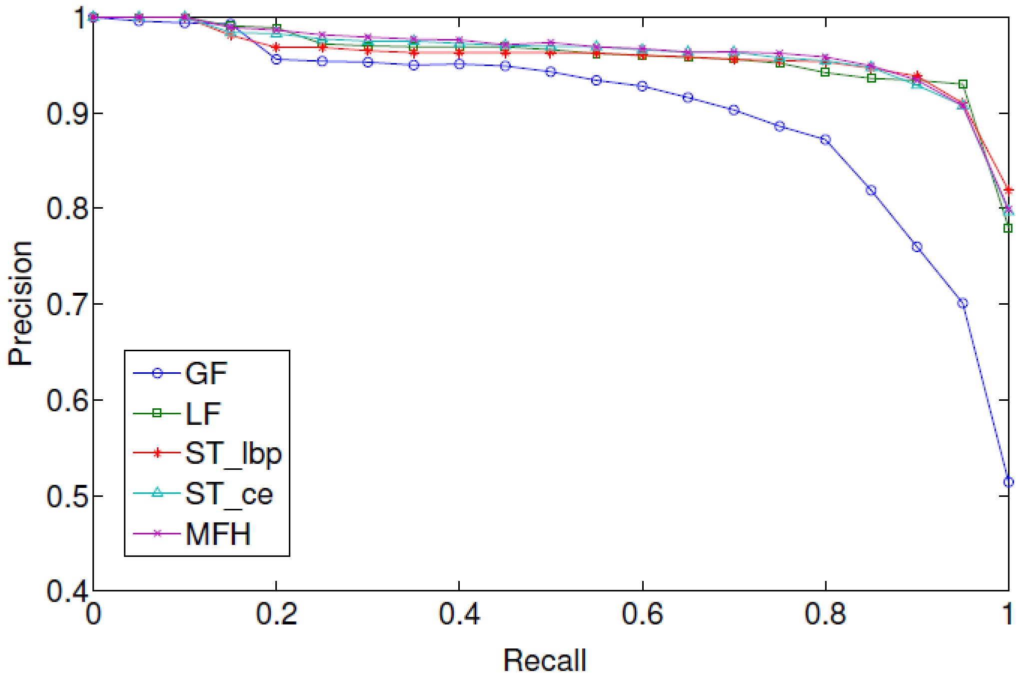
MFH - EXPERIMENTS

- We evaluate our approach on 2 video datasets
  - CC_WEB_VIDEO
    - Consisting of 13,129 video clips
  - UQ_VIDEO
    - Consisting of 132,647 videos, which was collected from YouTube by ourselves

- We present an extensive comparison of the proposed method with a set of existing algorithms, such as Self-taught Hashing, Spectral Hashing and so on

- Both efficiency and accuracy are reported to compare the performance

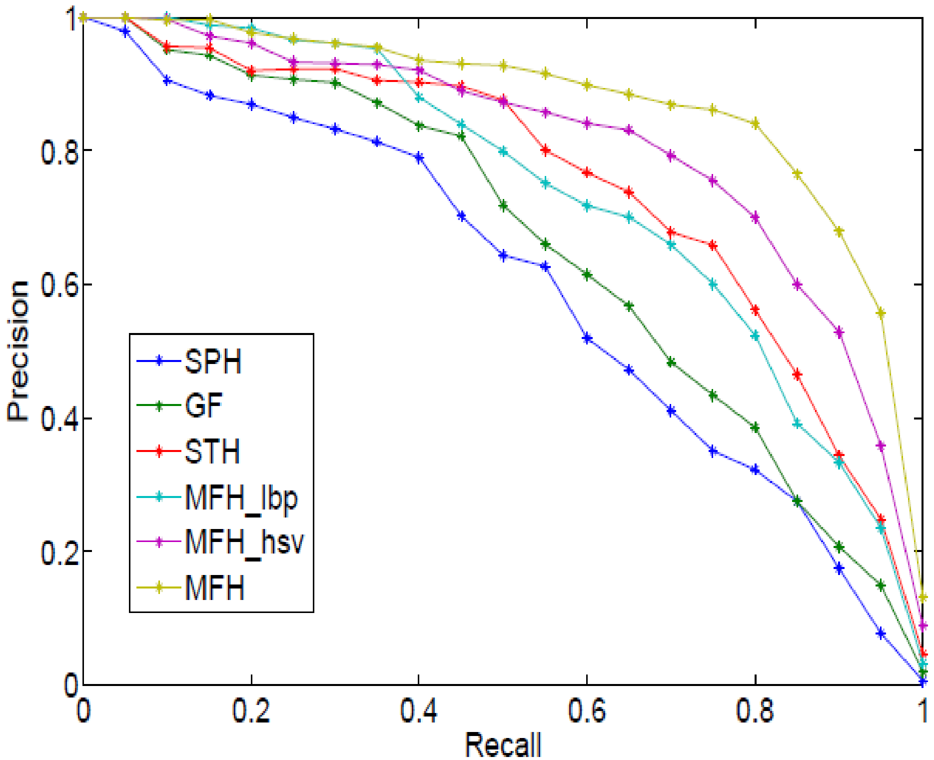
MFH - RESULTS



Methods	GF	LF	ST_lbp	ST_ce	MFH
MAP	0.892	0.952	0.953	0.950	0.954

Results On CC_WEB_VIDEO

MFH - RESULTS



Methods	MAP	Time(s)	Storage(MB)
SPH	0.5941	0.4907	4.8079
GF	0.6466	1.3917	211.5497
STH	0.7536	0.6439	6.3262
MFH_lbp	0.7526	0.6445	6.3262
MFH_hsv	0.8042	0.4508	4.4284
MFH	0.8656	0.5533	5.0610

Results On UQ_VIDEO

MFH - CONTRIBUTIONS

- As far as we know, it is the first hashing algorithm on indexing multiple features

- We propose a new framework to exploit multiple local and global features of video data for NDVR

- We have constructed a large scale video dataset UQ_VIDEO consisting of 132,647 videos which have 2,570,554 keyframes

THANK YOU

?