High Accuracy Perceptual Video Hashing via Low-Rank Decomposition and DWT

Keyword: Video hashing, Low-rank and sparse decomposition, Discrete wavelet transform, Copy detection

Lv Chen(B), Dengpan Ye, and Shunzhi Jiang

Ngày 26 tháng 9 năm 2022

Purpose/Output

- Đề xuất 1 thuật toán Video hashing mới với độ chính xác cao. Thuật toán được đề xuất tạo ra một fix-up hash thông qua phân rã cấp thấp (low-rank), Sparse decomposition và biến đổi wavelet rời rạc (DWT - Discrete Wavelet Transform).
- Sau đó, content-based feature matrices sẽ trích xuất từ một video được chuẩn hóa ngẫu nhiên với low-rank và sparse decomposition. Cuối cùng, nén dữ liệu với 2D-DWT của LL sub-band được áp dụng cho các ma trận đặc trưng và các thuộc tính thống kê của hệ số DWT được định lượng để tạo ra một video hash nhỏ gọn.

Data source/ input

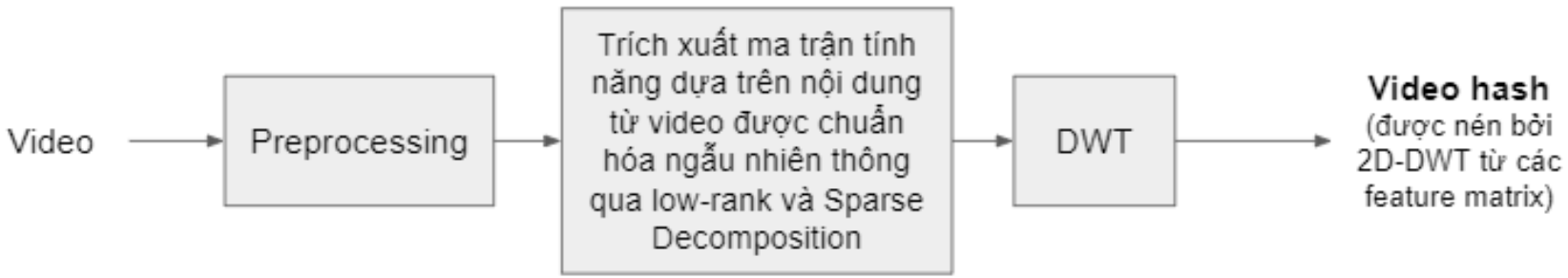
- Video dataset từ "The open video project" [1] với 76 video từ mục có tên là “University of Maryland HCIL Open House Video Reports”.
- Một số video thử nghiệm được chọn [1,2] được biểu diễn như Hình phía dưới.
- Video cho tập valid được lấy từ [2] với 200 video khác nhau

Một số video thử nghiệm được chọn Open databases.

1 The open video project. http://www.open-video.org/.
2 ReefVid: Free Reef Video Clip Database. http://www.reefvid.org/.

Method: Video Hashing

Có 3 bước:



Hình 1. Sơ đồ khối của phép băm video được đề xuất.

1. Preprocessing

- Video màu đầu tiên phải convert sang grayscale video.
- Input video được map thành video chuẩn hóa với cùng độ phân giải và khung hình bằng hai thao tác resampling.
  - **Thao tác 1:** Resample tạm thời, map các video có các frame khác nhau với cùng một số lượng frame. Cụ thể, các pixel của tất cả các frame ở cùng một vị trí có thể được xem như các ống có trật tự. *Phép nội suy tuyến tính (linear interpolation)* được áp dụng để thay đổi kích thước pixel của ống thành độ dài M thích hợp và được làm mịn bằng bộ lọc *Gaussian low-pass* với kernel size $= 1 \times$ f.
  - **Thao tác 2:** Resample lại không gian, thay đổi kích thước các video có kích thước khác nhau thành cùng độ phân giải. Mỗi frame được chuyển đổi thành square size M $\times$ M bằng *phép nội suy tuyến tính hai khối (bi-cubic linear interpolation)*.
- Sau thao tác 2, thu được video chuẩn hóa V_{norm}^{MxMxM} của M frame có size MxM

Để đảm bảo tính bảo mật của hàm băm, logistic map được áp dụng để xây dựng video chuẩn hóa ngẫu nhiên và được định nghĩa bằng phương trình:

$$x_{i+1} = rx_i(1 - x_i)$$

1. Preprocessing

Trong đó: r và x_0 là các control parameters.

- Khi r $\in$ (3.57, 4) và $x_0 \in$ (0, 1), logistic map có thể đạt đến trạng thái hỗn loạn.
- Trong paper này, chọn r = 3.85 và lấy giá trị ban đầu x_0 là key.
- Chuỗi ban đầu x = $x_1, x_2, ..., x_M$ được tạo ngẫu nhiên bằng cách sử dụng pseudo-random được điều khiển bằng khóa k_1 và $y = y_1, y_2, ...y_M$ được tạo ngẫu nhiên bằng cách sử dụng pseudo-random được điều khiển bằng khóa k_2. Sau đó chuỗi u = $u_1, u_2, ...u_M$ với M numbers được tạo ra, u_i được tính bằng x_i lần lặp y_i lần.
- Tiếp theo, dãy u được map thêm vào một dãy mới g, theo vị trí tương đối của dãy u theo thứ tự giá trị phần tử của chúng. Ví dụ, phần tử nhỏ nhất trong u tương ứng với vị trí tương đối 1 và phần tử lớn nhất trong u tương ứng với vị trí M (g = $g_1, g_2, ...g_M$).
- Cuối cùng, một video chuẩn hóa ngẫu nhiên V_{rank}^{MxMxM} được sắp xếp lại từ chuẩn V_{norm}^{MxMxM}. Lưu ý rằng, g là một dãy hỗn loạn, rất khó để đoán đúng số g nếu không biết các giá trị khóa k_1 và k_2.

2. Low-Rank and Sparse Decomposition

Phương pháp này được sử dụng trong lĩnh vực ảnh Y sinh và ảnh vệ tinh viễn thám (image classification, image denoising,...).

- $I \in R^{MxM}$ biểu diễn một ảnh có kích thước M $\times$ M.
- Kết quả của Low-rank và sparse decomposition của **I** là 2 ma trận $L \in R^{MxM}$ và $S \in R^{MxM}$. Low-rank component: phản ánh cấu trúc nguyên tắc của I và sparse component: đại diện cho các thành phần nổi bật của I, được biểu diễn:

$$\min_{\mathbf{L},\mathbf{S}} \mathbf{Rank}(\mathbf{L}) + \lambda \|\mathbf{S}\|_0$$
$$s.t.\ \mathbf{I} = \mathbf{L} + \mathbf{S},$$

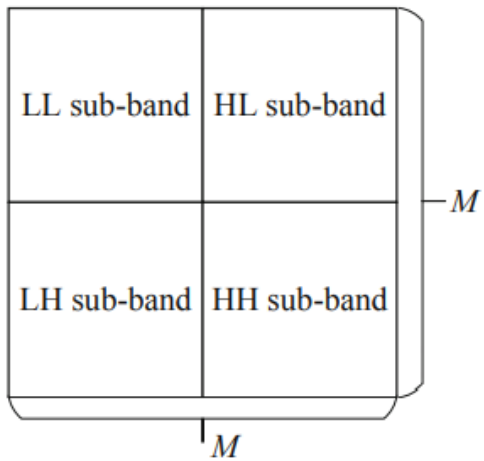
Trong đó: $||\cdot||_0$ biểu thị l_0 norm của ma trận và λ. Bài toán tối thiểu hóa lỗi bị ràng buộc có thể được giải quyết bằng một phương pháp là hệ số nhân Lagrange tăng cường không chính xác (IALM - inexact augmented Lagrange multipliers).

(a)low-rank component

(b)sparse component

3. DWT

- DWT là một kỹ thuật xử lý ảnh và một single-level 2-D DWT có thể phân tách hình ảnh đầu vào thành bốn sub-band như hình dưới LL sub-band, LH sub-band, HL sub-band và HH sub-band.
- **Trong đó:** LL sub-band có tần số thấp và 3 sub-band khác có tần số cao.



- Chọn LL sub-band để biểu diễn để lấy mã ngắn dựa trên hai chú ý:
  - Một số hoạt động bảo toàn nội dung, như lọc low-pass và ô nhiễm tiếng ồn, có ảnh hưởng nhẹ đến hệ số dải tần phụ tần số thấp.
  - Hệ số DWT trong LL sub-band là hệ số xấp xỉ của ma trận đặc trưng và chỉ là hệ số phần tư dành riêng.

$\Rightarrow$ *giúp hàm băm dẫn xuất thành một đoạn mã ngắn với các tính năng dựa trên nội dung.*

4. Minh họa thuật toán

- **Step 1:** Một video đầu vào được chuyển đổi thành normalized video V_{norm}^{MxMxM} với 2 lần resampling, sau đó được tạo thêm V_{rank}^{MxMxM} bằng cách sử dụng chuỗi vị trí tương đối tham chiếu đến một chuỗi hỗn loạn **g**.
- **Step 2:** Áp dụng low-rank và sparse decomposition cho từng frame của V_{rank}^{MxMxM} và trích xuất ma trận cấp thấp **L** dưới dạng content-based feature matrices cho từng frame.
- **Step 3:** Áp dụng 2D-DWT cho từng frame của video, và thu thập giá trị trung bình của tất cả hệ số LL sub-band để tạo thành 1 short string **s** (có tổng số M phần tử). Tiếp theo, các phần tử của s được lượng tử hóa theo quy tắc dưới đây:

$$h(i) = \begin{cases} 1 & \text{If s(i)} < \text{s(i+1)} \\ 0 & \text{Otherwise,} \end{cases} \qquad i = 1, 2, ..., M,$$

Trong đó: s(i) là phần tử thứ i của **s** $\rightarrow$ hàm băm được tạo như sau:

$$\text{h} = [\text{h(1)},\text{h(2)},...,\text{h(M)}].$$

5. Hash Similarity

- Vì hàm băm của chúng ta là một chuỗi bit, nên Hamming distance được lấy để đánh giá mức độ giống nhau giữa hai hàm băm.
- Giả sử V_1 và V_2 đại diện cho hai video khác nhau và h_1 và h_2 lần lượt đại diện cho hai hàm băm tương ứng của chúng. Khoảng cách giữa hai hàm băm có thể được tính theo phương trình sau:

$$d_H(h_1, h_2) = \sum_{i=1}^{M} |h_1(i) - h_2(i)|$$

Trong đó: phần tử thứ i của h_1 và h_2 lần lượt là h_1(i) và h_2(i), và M đại diện cho độ dài của hàm băm. Hamming distance càng nhỏ thì 2 video càng giống nhau.

  - Ngưỡng T xác định trước có thể được sử dụng để đánh giá mức độ giống nhau của hai video. Nói cách khác, hai video có thể được đánh giá là video giống nhau về mặt hình ảnh khi $d_H \leq T$. Nếu không, hai video có thể được coi là video khác nhau về hình ảnh.

Results

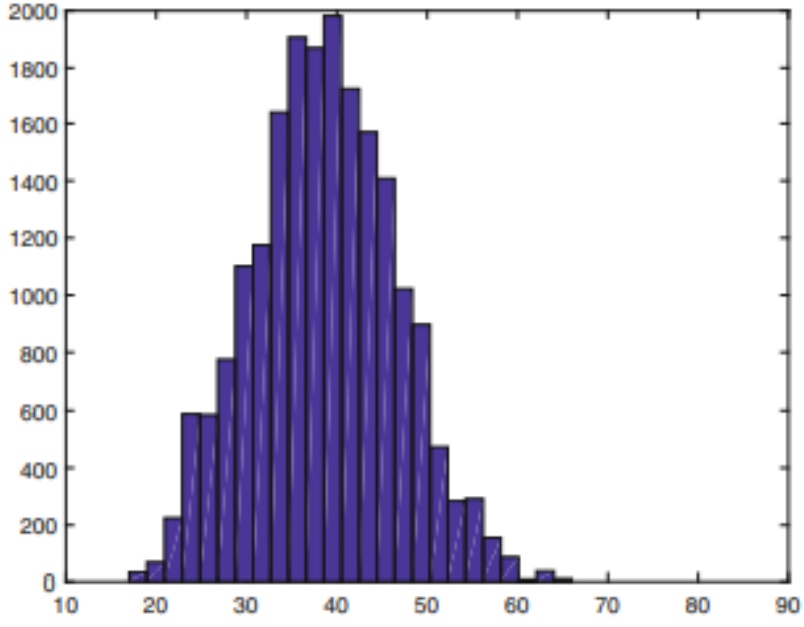
1. Robustness

- Video chuẩn hóa ngẫu nhiên có kích thước $128 \times 128 \times 128$
- Một bộ lọc Gaussian low-pass với kernel size là 1×20. Do đó, độ dài hàm băm là 128 bits.
- Chi tiết cách cài đặt hoạt động như sau:
  - Điều chỉnh độ sáng với thang đo của Photoshop là 20, 15, ..., 20
  - Bộ lọc Gaussian low-pass 3×3 với Độ lệch chuẩn là 0.1, 0.2, ..., 1
  - Salt và pepper noise của mật độ là 0.001, 0.002, ..., 0.01
  - AWGN của tỷ số nhiễu tín hiệu là 1, 2, ..., 6
  - Tốc độ bit cho MPEG-2 là 100, 200, ..., 1000
  - Số lượng khung hình drop là 5, 10, ..., 20
  - Tỷ lệ mở rộng khung hình là 0.8, 0.85, ..., 1.2 và góc quay là 2, 1, ..., 2
- Sau khi thử nghiệm, rút ra được hầu hết các giá trị trung bình đều nhỏ hơn 6 ngoại trừ hoạt động của Rotation và giá trị trung bình của Rotation không lớn hơn 15. Điều này ngụ ý rằng ngưỡng được chọn là $T = 15$

Results

2. Discrimination

- Trích xuất các hàm băm cho mỗi video và tính toán khoảng cách Hamming giữa hàm băm ban đầu của nó và các hàm băm của 199 video khác. Do đó, cặp khoảng cách $200 \times 199/2 = 19900$ Hamming thu được.
- Hình dưới cho thấy khoảng cách Hamming tối thiểu giữa hai video khác nhau là 16 và tối đa là 89.
- Giá trị trung bình và độ lệch chuẩn lần lượt là 38.52 và 8.06
- Đặt ngưỡng là 35, thì có 0,003% các cặp video khác nhau bị coi là các video giống nhau về mặt hình ảnh.

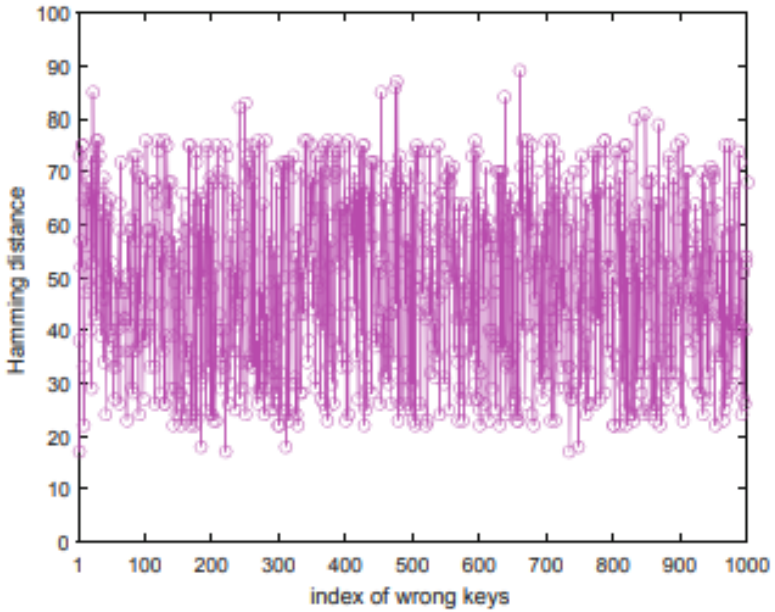


Results

3. Key Dependence

Để xác thực bảo mật của hàm băm, lấy video của phần Robustness và discrimination test video làm video kiểm tra.

- Đặt các nhóm key bí mật khác nhau k_1 và k_2 của logistic map để tạo các hàm băm của mọi video thử nghiệm và tính toán khoảng cách Hamming giữa các hàm băm của video thử nghiệm.
- Sau khi thử nghiệm hơn 1000 nhóm key khác nhau (k_1 và k_2) thì thấy khoảng cách Hamming của các khóa sai phân bố từ 17 đến 89, lớn hơn ngưỡng T = 15.



$\Rightarrow$ **phép băm phụ thuộc vào key-dependence.**