Discussion of a Pruning Scheme for Top-K Retrievals Among Vector Quantizer Encoded Signatures

Anindya Sarkar[1], Vishwakarma Singh[2], Pratim Ghosh[1], B. S. Manjunath[1], Ambuj Singh[2]

[1] Department of Electrical and Computer Engineering, University of California, Santa Barbara
[2] Department of Computer Science, University of California, Santa Barbara

April 25, 2008

1 Problem Statement

The problem we are considering here is duplicate video detection. We have a database of N videos and we store compact signatures, called fingerprints, for each of them. When a query video is presented, the system first returns the top-K most closely matched videos. Then, a more detailed search is performed among the top-K retrieved model videos to obtain the best match. Finally, a separate module is used to confirm whether the best matched video is indeed a duplicate. A complete overview of our duplicate detection framework is shown in Fig. 1. In this write-up, we focus on the VQ based pruned search where the effort is to return the top-K neighbors in the fastest possible manner without having to do a linear scan of all the N database signatures. The database videos are referred to as "model" videos in this write-up.

The N model video signatures in the database are denoted by $\{X^i\}_{i=1}^{N}$. On presenting a query video signature Q, the aim is to find the K model video signatures that are nearest to Q. The notion of similarity is with reference to a distance measure $d(X^i, Q)$ (1). To simplify matters and improve runtime, a vector quantizer (VQ) based approach is used, where the video signatures are VQ encoded and lookup table based methods are used to make the search faster.

$$d(X^i, Q) = \sum_{k=1}^{M} \left\{ \min_{1 \le j \le F_i} \left\| X^i_j - Q_k \right\|_1 \right\} \tag{1}$$

where $\left\| X^i_j - Q_k \right\|_1$ refers to the L_1 distance between X^i_j, the j^{th} feature vector of X^i and Q_k, the k^{th} feature vector of Q. For every vector in Q, the best match is obtained out of all the vectors in X^i and $d(X^i, Q)$ is the summation of the best matched distances.

Glossary of Notations

N : number of database videos
V_i : i^{th} model video in the dataset
V_{i^*} : best matched model video for a given query
p : dimension of the feature vector computed per video frame
$Z^i \in \mathbb{R}^{T_i \times p}$: feature vector matrix of V_i, where V_i has T_i frames after temporal sub-sampling
$X^i \in \mathbb{R}^{F_i \times p}$: fingerprint of V_i, which has F_i keyframes
X^i_j: j^{th} vector of video fingerprint X^i
U : size of the vector quantizer (VQ) codebook used to encode the model video and query video signatures
$Q_{orig} \in \mathbb{R}^{T_Q \times p}$: query signature created after sub-sampling, where T_Q refers to the number of sub-sampled query frames
$Q \in \mathbb{R}^{M \times p}$: keyframe based signature of the query video, where M is the number of query keyframes
C_i : the i^{th} VQ codevector
$\overrightarrow{x_i}$: VQ based signature of V_i
$\vec{q}$: VQ based query signature
$\mathcal{S}_{X^i_j}$: VQ symbol index to which X^i_j is mapped
$\mathbb{D} \in \mathbb{R}^{U \times U}$: Inter VQ-codevector distance matrix
$\mathbb{D}^* \in \mathbb{R}^{N \times U}$: Lookup distance matrix of shortest distance values from each model to each VQ codevector
$|E|$: the cardinality of the set E

2 Use of VQ-encoded signatures

We develop an algorithm that uses VQ-based encoding on the signature feature vectors. Thus, the distance between any two feature vectors reduces to an inter-symbol distance, after VQ-based encoding. By using a lookup table of inter-VQ codevector distances, the L_1 distance computation cost (e.g. $\|X^i_j - Q_k\|_1$) can be avoided.

Using the features extracted from the database video frames, a vector quantizer of codebook size U is constructed. Since each vector in a video signature can be mapped to one of U codevectors, the effective video signature can be thought of as a U-dimensional vector, where the i^{th} dimension denotes the fraction of vectors in the original signature which get mapped to the i^{th} codevector C_i.

Let $[q_1, q_2, \cdots, q_U]$ denote the normalized query video signature $\overrightarrow{q}$ and $[x_{i,1}, x_{i,2}, \cdots, x_{i,U}]$ denote the normalized model video signature $\overrightarrow{x_i}$ for the i^{th} video V_i.

$$q_k = |\{j : \mathcal{S}_{Q_j} = k,\ 1 \le j \le M\}|/M \quad (2)$$
$$x_{i,k} = |\{j : \mathcal{S}_{X^i_j} = k,\ 1 \le j \le F_i\}|/F_i \quad (3)$$

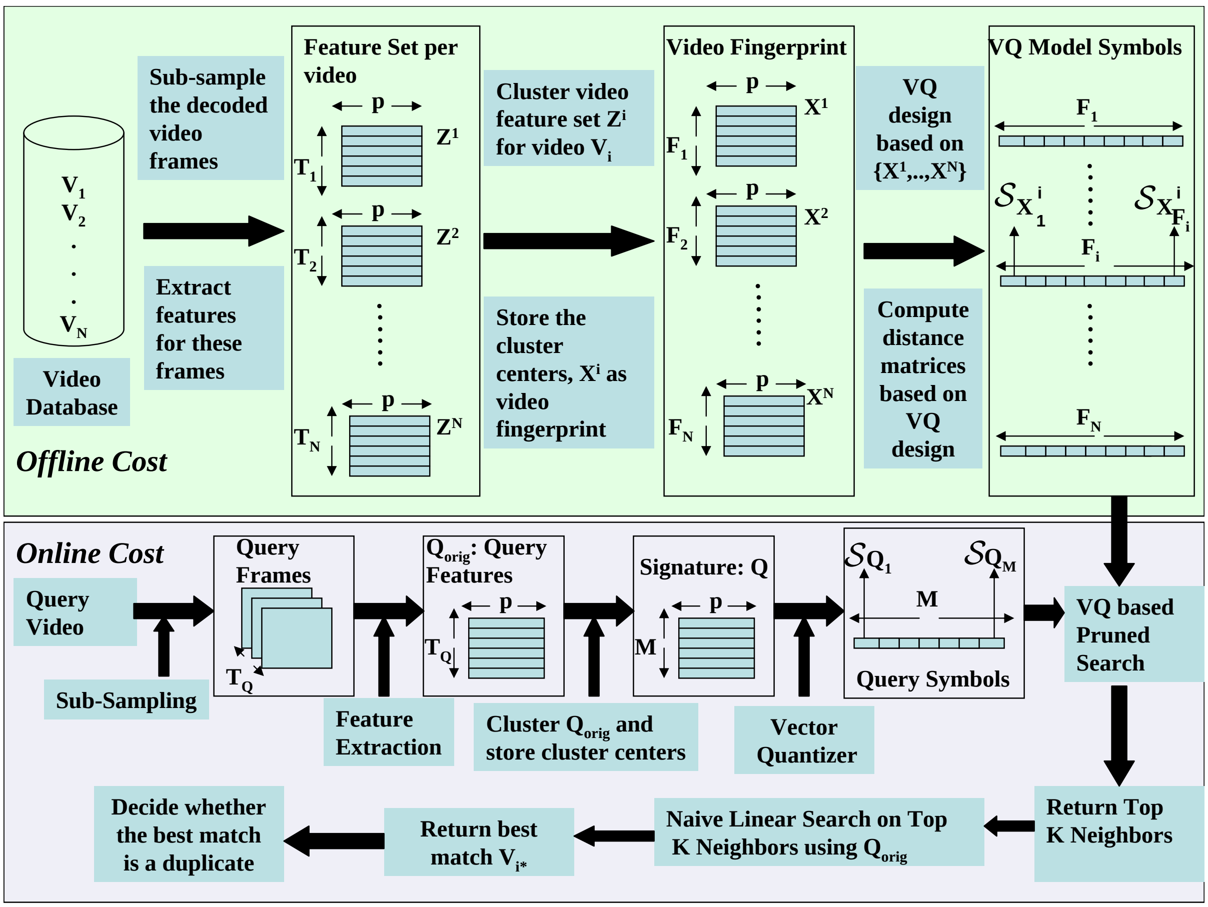

Figure 1: Block diagram of the proposed duplicate detection framework.

Generally, there is a high degree of redundancy among video frames; hence, many of them will get mapped to the same VQ codevector and there will be many VQ codevectors which will have no representative (assuming a large enough U). Let $\{t_1, t_2, \cdots, t_{N_q}\}$ and $\{n_{i,1}, n_{i,2}, \cdots, n_{i,N_{x_i}}\}$ denote the non-zero dimensions in $\overrightarrow{q}$ and $\overrightarrow{x_i}$, respectively.

The distance between them can be expressed as:

$$
\begin{aligned}
d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}) &= \sum_{k=1}^{N_q} q_{t_k} \times \left\{ \min_{1 \le j \le N_{x_i}} \mathbb{D}(t_k, n_{i,j}) \right\} && (4) \\
\text{where } \mathbb{D}(i,j) &= \|C_i - C_j\|_1,\ 1 \le i, j \le U && (5)
\end{aligned}
$$

where $\mathbb{D} \in \mathbb{R}^{U \times U}$ is the inter-VQ codevector distance matrix.

It can be easily shown that the distances in (1) and (4) are identical, apart from a constant scaling factor, when each vector in (1) is represented by its corresponding VQ codevector.

$$
d(X^i, Y) = M \times d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}) \tag{6}
$$

Further speedup is possible if we are able to directly lookup the distance of a query signature symbol to its nearest symbol in a model video signature (e.g. $\{\min_{1 \le j \le N_{x_i}} \mathbb{D}(t_k, n_{i,j})\}$ in (4)). We pre-compute a matrix $\mathbb{D}^* \in \mathbb{R}^{N \times U}$ where $\mathbb{D}^*(i,k)$ denotes the minimum distance of a query vector, represented by symbol i after the VQ encoding, to the k^{th} model.

$$
\begin{aligned}
d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}) &= \sum_{k=1}^{N_q} q_{t_k} \times \mathbb{D}^*(t_k, i) && (7) \\
\text{where } \mathbb{D}^*(i,k) &= \min_{1 \le n \le F_k} \mathbb{D}(i, \mathcal{S}_{X_n^k}) && (8)
\end{aligned}
$$

3 Theoretical Solution for Pruning Along the Model Video Search Space

For a big enough dataset (large N), a practical approach to pruning can be if we can avoid considering all the model videos, while ensuring that we still return the top-K model videos. The philosophy for this pruning is explained below.

Given a dataset of $\{\overrightarrow{x_i}\}$ signatures, where $i \in S$, we present a lower bound of the minimum model-to-query distance, $\{\min_{i \in S} d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q})\}$, found for all signatures in the dataset (9). Here, $\beta(i, t_k)$ denotes the best matching dimension in $\overrightarrow{x_i}$ for dimension t_k.

$$\begin{aligned}
\min_i d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}) &= \min_i \left[\sum_{k=1}^{N_q} q_{t_k} \times \mathbb{D}(t_k, \beta(i, t_k)) \right] \\
&\geq \min_i \left[\sum_{k=1}^{N_q} q_{t_k} \times \{ \min_j \mathbb{D}(t_j, \beta(i, t_j)) \} \right] \\
(\text{using } \sum_{k=1}^{N_q} q_{t_k} = 1) &= \min_i \{ \min_j \mathbb{D}(t_j, \beta(i, t_j)) \} \qquad (9)
\end{aligned}$$

Thus, the lower bound equals the smallest distance between a non-zero query dimension and any of the non-zero model dimensions.

We store two $(P \times P)$ matrices, a proximity matrix $\mathbb{P}$ and a distance matrix $\mathbb{D}'$, which store the nearest neighbors (NN), and their corresponding distances, respectively, for a certain VQ codevector. E.g. $\mathbb{P}(i, j)$ denotes the j^{th} nearest neighbor for the i^{th} VQ codevector. Similarly, $\mathbb{D}'(i, j)$ denotes the distance of the $\{\mathbb{P}(i, j)\}^{th}$ codevector from the i^{th} VQ codevector, i.e. $\mathbb{D}'(i, j) = \mathbb{D}(i, \mathbb{P}(i, j)) = \|C_i - C_{\mathbb{P}(i,j)}\|_1$.

We also store P clusters $\{\mathbb{C}(i)\}_{i=1}^{P}$, where $\mathbb{C}(i)$ denotes the cluster which contains those model video indices whose signatures which have the i^{th} dimension as non-zero.

$$\mathbb{C}(i) = \{ j : x_{j,i} > 0,\ 1 \leq j \leq N \} \qquad (10)$$

This method uses a multi-pass approach, where as soon as a certain distance based condition is satisfied, the search can be stopped at that pass and it can be guaranteed that the top-K candidates have been found, out of all N model videos. We provide a list of symbols with their definitions used in the algorithm:

1. $\mathbb{S}_j$: denotes the set of distinct model videos considered in the j^{th} pass
2. G: denotes the set of non-zero query dimensions;
   $G = \{t_1, t_2, \cdots, t_{N_q}\}$
3. d_j^*: denotes the minimum of the distances of all codevectors contained in the query to their j^{th} nearest neighbors
4. $d_{min,j}$: denotes the minimum possible distance value, between a certain non-zero query dimension and all the non-zero dimensions in the model videos found in $\mathbb{S}_j$
5. A_j: denotes the set of distinct VQ indices which are encountered on considering the first j nearest neighbors for each of the elements in G. Therefore, $(A_j \setminus A_{j-1})$ denotes the set of distinct (not seen in earlier passes) VQ indices encountered in the j^{th} pass, when we consider the j^{th} NN of the elements in G.

For a given query, the model videos which are nearest to it are likely to have some or all of the non-zero dimensions, as the query signature itself, as non-zero. In the first pass, we find all the model videos which

Algorithm 1 Pruning Along Model Video Search Space - here, unique(E) returns the unique (without repeats) elements in E

Input: N model video signatures, $\overrightarrow{x_i} \in \mathbb{R}^U,\ 1 \le i \le N$
Input: the query signature $\vec{q}$, and lookup matrices $\mathbb{P}$ and $\mathbb{D}'$
Output: Best sequence to search N videos

Initialization: (1^{st} **pass**)
$G = \{n_1, n_2, \cdots, n_{N_q}\}$
$A_1 = G$
$\mathbb{S}_1 = \bigcup_{1 \le i \le N_q} \mathbb{C}(n_i)$
$d_1^* = \min_{1 \le i \le |G|} [\mathbb{D}'(G_i, 1)] = 0$
We maintain the K-minimum distance values $\{L_i\}_{i=1}^K$ and the corresponding indices $\{I_i\}_{i=1}^K$, based on the elements in $\mathbb{S}_1$.
End of 1^{st} **pass**
for j=2 to U **do**
  $d_j^* = \min_{1 \le i \le |G|} \{\mathbb{D}'(G_i, j)\}$
  if $L_K \le d_j^*$ **then**
    break;
  end if
  $B_i = \mathbb{P}(n_i, j),\ 1 \le i \le N_q$
  $E = B \setminus A_{j-1},\ E =$ unique(E)
  $\mathbb{S}_j = \bigcup_{1 \le i \le |E|} \mathbb{C}(E_i)$
  $\mathbb{S}_j = \mathbb{S}_j \setminus \bigcup_{1 \le i < j} \mathbb{S}_i$, (get videos not seen in earlier iterations)
  $A_j = A_{j-1} \cup E$
  Update the lists I and L based on the elements in $\mathbb{S}_j$
end for
return The sequences observed so far $\{\mathbb{S}_1, \mathbb{S}_2, \cdots, \mathbb{S}_{j-1}\}$

have at least one of the non-zero query dimensions as non-zero - $\mathbb{S}_1$ is the set of these video indices. We store the top-K neighbors ($\{I_i\}_{i=1}^K$) and the K corresponding distance values ($\{L_i\}_{i=1}^K$, sorted in ascending order) from this set.

We now show why $d_j^* \leq d_{min,j}$ holds, $\forall j$. To compute $d_{min,j}$, we consider elements in $\mathbb{D}'$ where the column index is j and the rows correspond to U, a subset of G (only those elements in G, the j^{th} NN of which belongs to $(A_j \setminus A_{j-1})$, the set of new VQ indices encountered in the j^{th} pass, constitute U). Thus, $d_j^* \leq d_{min,j}$ as d_j^* is the minimum computed over a larger set than $d_{min,j}$.

$$U = \{G_i,\ i : \mathbb{P}(G_i, j) \in (A_j \setminus A_{j-1})\} \tag{11}$$

$$d_{min,j} = \min_i [\mathbb{D}'(U_i, j)] \tag{12}$$

$$U \subseteq G \Rightarrow d_j^* \leq d_{min,j}$$

We now show that $\{\min_{i \in \mathbb{S}_j} d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q})\} \geq d_{min,j}$. Out of all the distinct VQ indices contained in the model videos in $\mathbb{S}_j$, there cannot be any VQ index that is a $\hat{j}^{th}(\ \hat{j} < j)$ NN of any non-zero query dimension. This is because all $\hat{j}^{th}$ $(\hat{j} < j)$ NN indices are used up in the set $\cup_{\ell, 1 \leq \ell < j} \mathbb{S}_\ell$. Therefore, the smallest "query dimension-to-model dimension" distance is due to a model dimension which is the j^{th} NN of a certain query dimension. $J = \{\mathbb{P}(t_k, j)\}_{k=1}^{N_q}$ is the set of indices that serve as the j^{th} NN of non-zero query dimensions. Of these indices, some may have already been present in the model indices found in $\cup_{\ell, 1 \leq \ell < j} \mathbb{S}_\ell$. The set of VQ indices that are j-NN of the query dimensions and are newly encountered in the j^{th} pass is given by $(A_j \setminus A_{j-1})$.

$$\begin{aligned}
\min_{i,\ i \in \mathbb{S}_j} d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}) \quad &\geq \quad \min_{1 \leq k \leq N_q} [\min_{\ell : \mathbb{P}(G_\ell, j) \in (A_j \setminus A_{j-1})} \mathbb{D}'(t_k, J_\ell)] \\
&= \quad \min_{\ell : \mathbb{P}(G_\ell, j) \in (A_j \setminus A_{j-1})} [\mathbb{D}'(G_\ell, \mathbb{P}(G_\ell, j))] \\
&= \quad \min_{1 \leq k \leq |U|} [\mathbb{D}'(U_k, j)],\ \text{using (11)} \\
&= \quad d_{min,j}
\end{aligned} \tag{13}$$

When we consider videos in $\mathbb{S}_j$, during the j^{th} pass, $d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}) \geq d_{min,j}$, where model index $i \in \mathbb{S}_j$. Since $d_{min,j} \geq d_j^*$, and if $d_j^* \geq L_K$, then it is assured that $d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}) \geq L_K$, using (13). Now, if for the j^{th} pass, it is ensured that $d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}) \geq L_K,\ i \in \mathbb{S}_j$, then is it guaranteed that for videos in the ${j'}^{th}$ pass, (for $j' > j$), $d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}) \geq L_K,\ i \in \mathbb{S}_{j'}$?

Explanation : $d_{j'}^* \geq d_j^*,\ \because \mathbb{D}'$ is a sorted matrix

$d_{min,j'} \geq d_{j'}^*,\ \because d_{j'}^*$ is the minimum over a larger set than $d_{min,j'}$

$$\therefore L_K \leq d_j^*,\ d_j^* \leq d_{j'}^*,\ d_{j'}^* \leq d_{min,j'},\ d_{min,j'} \leq d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}),\ i \in \mathbb{S}_{j'}$$

$$\Rightarrow\ L_K \leq d_{VQ}(\overrightarrow{x_i}, \overrightarrow{q}),\ i \in \mathbb{S}_{j'}$$

Hence, it is confirmed that if $d_j^* \geq L_K$, we **will not** find a model video in any sequence $\mathbb{S}_{j'}$, where $j' > j$, with model-to-query distance less than L_K, where L_K is the K^{th} minimum distance computed over the set of videos constituted using sequences $\{\mathbb{S}_k\}_{k=1}^{j-1}$.

If this condition ($L_K \leq d_j^*$) is not satisfied, then we compute S_j and proceed to the $(j+1)^{th}$ pass. After the j^{th} pass, we need to maintain $\mathbb{S}_j$, A_j, and the updated lists $\{I_i\}_{i=1}^{K}$ and $\{L_i\}_{i=1}^{K}$.