

distance from the query fingerprint. Thus, fingerprints with a small number collisions per fingerprint codeword will result in shorter search duration. Databases containing fingerprints with an average low collision number are more scalable to large database sizes than databases with a high average collision number per fingerprint.

In our previous work [4], we studied the collision property of one of the robust hash functions proposed in [3]. The robust hash proposed in [3] first projects a feature matrix onto a set of projection matrices. Then, the projected values are compared to a threshold to generate hash bits. In [4], we showed that the collision performance of these methods can be optimized by a careful selection of the projection matrices. Specifically, the projection matrices were obtained using a Singular Value Decomposition (SVD) on a set of features from a training set that minimized the cross-correlation between projected values. However, the projection matrices did not consider the effect of content modifications on the projected values.

In this paper, we study two methods that create projection matrices that take into account the noise introduced by content modifications on these projected values. The first method derives the projection matrices based on the maximization of a generalized Rayleigh quotient and the second method derives the projection matrices based on an eigenvector analysis of difference between the signal and the noise correlation matrices. Both of these methods were originally proposed for deriving noise-robust features from audio spectrograms in [1]. Based on experimental results, we show that the first method improves the robustness of the projected values while losing heavily in terms of collision performance. The second method improves the robustness of the projected values while showing a poor collision performance. Also, we interpret the projection matrices obtained through these two methods and compare with those derived using the SVD procedure in [4].

2. PROPOSED APPROACHES

The first step in a media fingerprint extraction process is the derivation of robust features from the video content that are invariant to various processing operations of the content. Let us represent the extracted robust features by a matrix Q . For all the experiments in this paper, Q is a matrix with $G \times N$ elements that attempts to capture the appearance and motion information from the input video. The input video is first temporally downsampled to 12 fps. A letter-box detector detects eventual black bars on the sides of the picture and removes them. Then, every frame is spatially downsampled to 120×160 (*Height* \times *Width*) resolution. The N columns in the feature matrix Q correspond to N video frames in a buffer including the current frame. We use a buffer of 3s ($N = 12 \times 3$) to capture motion information. The G rows correspond to a set of moment invariants extracted from each frame in the buffer. We extract 7 moment invariants from 5 concentric circular regions in each frame ($G = 5 \times 7$). Moment invariants are global measures of an image surface that are invariant to translation, rotation and scaling and were originally proposed for text pattern recognition in [2]. Now, each column of the matrix Q attempts to capture the appearance of the corresponding frame of the video in the buffer by measuring how the 7 moment invariants change over the 5 regions. In the second step, hash

bits are extracted from the feature matrix Q using a robust hash function. The robust hash function projects the feature matrix (Q) onto L projection matrices P_i , $i = 1, 2, \dots, L$. Then, each projected value is compared with a threshold Th to obtain L bits b_i . If $B.P_i < Th$, then $b_i = 0$ else $b_i = 1$.

In [4], we proposed a method to obtain these projection matrices that are optimal in terms of collision performance. Here, we briefly recall the procedure in the following three steps.

Step 1: Obtain a training dataset of media content and extract features Q^1, Q^2, \dots, Q^M . Here M is the number of training instances. The dimension of each feature matrix Q^i is $R = G \times N$.

Step 2: Then, we compute the covariance matrix C^{feat} (dimension $R \times R$) of the features from the training set Q^1, Q^2, \dots, Q^M as given by

$$C^{feat}(k, l) = \frac{1}{M} \sum_{i=1}^M (Q_k^i - E(Q_k))(Q_l^i - E(Q_l))$$

Here $E(Q_k) = \frac{1}{M} \sum_{i=1}^M Q_k^i$ and $k, l = 1, 2, \dots, R$.

Step 3: Once we have computed the covariance matrix of the features from the training set, we can now compute the eigenvectors of the matrix C^{feat} that satisfy the relation, $V^{-1}C^{feat}V = D$, using PCA (Principal Components Analysis). Here the columns of V (dimension $R \times R$) are the eigenvectors of the covariance matrix C^{feat} and are represented as $\phi_1, \phi_2, \dots, \phi_R$. D is a diagonal matrix with eigenvalues (E_1, E_2, \dots, E_R) as its main diagonal elements and zero elsewhere.

Now, we transform the input feature vector Q to a L dimensional space by projecting it onto the first L eigenvectors $\phi_1, \phi_2, \dots, \phi_L$ and obtain $H_1^{svd}, H_2^{svd}, \dots, H_L^{svd}$ respectively. Here H_k^{svd} is the projection of Q onto ϕ_k . In [4], we showed that the fingerprints derived using these projection matrices have lower average number of collisions than fingerprints derived using random projection matrices.

However, these projection matrices do not consider the noise on the projected values due to content modifications (H_k^{svd}). In the following two subsections we describe two methods that derive the projection matrices which account for this noise.

2.1 Noise Robust Projection Matrices based on Rayleigh Quotient

Consider the case where the original video is modified by compression, cropping and/or rotation. These operations add noise to the extracted feature vector. The noisy feature matrix is denoted as \tilde{Q} . Let the projections obtained by projecting \tilde{Q} onto L projection matrices be represented as $\tilde{H}_1, \tilde{H}_2, \dots, \tilde{H}_L$. Then, the noise on i^{th} projected value is $Z_i = H_i - \tilde{H}_i$. Here H_i is the projection of original feature matrix Q onto the i^{th} projection matrix.

Step 1: Obtain a training dataset of media content and extract features Q^1, Q^2, \dots, Q^M . Here M is the number of training instances. The dimension of each feature matrix Q^i be $R = G \times N$. We also create modified versions of content and compute the corresponding modified features $\tilde{Q}^1, \tilde{Q}^2, \dots, \tilde{Q}^M$.

Step 2: We compute the noise in the feature vectors for the training set W^1, W^2, \dots, W^M . Here $W = Q - \tilde{Q}$ and W is of the same dimension as Q .

Step 3: We compute the covariance matrix C^{feat} (dimension $R \times R$) of the features from the training set Q^1, Q^2, \dots, Q^M as given by:

$$C^{feat}(k, l) = \frac{1}{M} \sum_{i=1}^M (Q_k^i - E(Q_k))(Q_l^i - E(Q_l))$$

Here $E(Q_k) = \frac{1}{M} \sum_{i=1}^M Q_k^i$ and $k, l = 1, 2, \dots, R$. We refer to C^{feat} as the signal covariance matrix.

Step 4: We compute the noise covariance matrix C^{noise} (dimension $R \times R$) of the noise in the features from the training set W^1, W^2, \dots, W^M as given by

$$C^{noise}(k, l) = \frac{1}{M} \sum_{i=1}^M (W_k^i - E(W_k))(W_l^i - E(W_l))$$

Here $E(W_k) = \frac{1}{M} \sum_{i=1}^M W_k^i$ and $k, l = 1, 2, \dots, R$. We refer to C^{noise} as the noise covariance matrix.

Step 5: Given the signal covariance matrix (C^{feat}) and the noise covariance matrix (C^{noise}), the generalized Rayleigh quotient is given by $\frac{pC^{feat}p}{pC^{noise}p}$ and can be maximized by solving the following generalized eigenvalue problem:

$$C^{feat}p = r_q C^{noise}p$$

Here, p is a generalized eigenvector. r_q is the corresponding generalized eigenvalue also denoted as the Rayleigh quotient. The most significant generalized eigenvector p obtained as a solution to this eigenvalue problem maximizes the generalized Rayleigh quotient. Maximizing Rayleigh quotient ensures that the projected values have a higher variance along the signal direction while having a small variance along the noise direction thereby improving the signal to noise ratio of the projected values. We select the L most significant generalized eigenvectors of this solution, $\phi_1^{rq}, \phi_2^{rq}, \dots, \phi_L^{rq}$, as our projection matrices. We can now transform any input feature vector Q to a L dimensional space by projecting it onto the projection matrices $\phi_1^{rq}, \phi_2^{rq}, \dots, \phi_L^{rq}$ to obtain $H_1^{rq}, H_2^{rq}, \dots, H_L^{rq}$, respectively. Here H_k^{rq} is the projection of Q onto ϕ_k^{rq} .

2.2 Noise Robust Projection Matrices based on Difference between Signal and Noise Covariance Matrices

In this method, we compute the signal covariance matrix (C^{feat}) and the noise covariance matrix (C^{noise}) from a training dataset as explained in steps 1-4 in the previous subsection. Instead of maximizing the Rayleigh quotient to obtain the projection matrices, here we compute projection matrices that minimize the following mean squared reconstruction error.

$$\sum_{i=1}^M (Q_i - \widehat{Q}_i)^2$$

Here $\widehat{Q}_i = (\widetilde{Q}_i \cdot p)p$. Recall that \widetilde{Q}_i is the noisy feature extracted from modified content. Then, \widehat{Q}_i is a projection of the noisy feature \widetilde{Q}_i along the direction p . By minimizing the reconstruction error in the equation above, we attempt to find projection matrices (p) such that the transformed noisy feature vector is as close as possible to the

original feature vector. The projection matrices that minimize this reconstruction error can be found as eigenvectors of $C^{feat} - C^{noise}$. We select the first L significant eigenvectors of this difference between signal and noise covariance matrices, $\phi_1^{s-n}, \phi_2^{s-n}, \dots, \phi_L^{s-n}$, as our projection matrices. We can now transform any input feature vector Q to a L dimensional space by projecting it onto the projection matrices $\phi_1^{s-n}, \phi_2^{s-n}, \dots, \phi_L^{s-n}$ to obtain $H_1^{s-n}, H_2^{s-n}, \dots, H_L^{s-n}$ respectively. Here H_k^{s-n} is the projection of Q onto ϕ_k^{s-n} .

3. EXPERIMENTAL RESULTS

In this section, we present experimental results on collision performance and robustness of three methods for selecting the projection matrices. The first method is proposed in [4]. It is based on a SVD of the signal covariance matrix (C^{feat}) computed from a training set. The L projection matrices obtained in this manner are denoted to as $\phi_1, \phi_2, \dots, \phi_L$. The other two methods considered in this paper are methods that take into account the noise introduced by content modifications to derive the projection matrices. The second method obtains projection matrices based on maximization of Rayleigh quotient. The projection matrices obtained using this method are denoted as $\phi_1^{rq}, \phi_2^{rq}, \dots, \phi_L^{rq}$. The third method obtains projection matrices based on the SVD of difference between the signal and the noise covariance matrix. The projection matrices obtained using this method are denoted as $\phi_1^{s-n}, \phi_2^{s-n}, \dots, \phi_L^{s-n}$. The feature matrix Q for all experiments in this paper is a matrix of dimensions 35×36 ($G \times N$) where 36 corresponds to a 3s buffer of frames at 12fps and 35 corresponds to the 7 moment invariants computed from 5 concentric circular regions of each frame in the buffer.

3.1 Collision analysis

First, we extract 22 bit fingerprint codewords ($L = 22$) from a 25 min video clip using three sets of projection matrices: (i) ϕ_i (svd), (ii) ϕ_i^{rq} (rq), (iii) ϕ_i^{s-n} (s-n). In all three cases, for the selection of the projection matrices, we used the same offline training set of 94 hrs of content to compute both the signal covariance matrix (C^{feat}) and the noise covariance matrix (C^{noise}). In the case of (i), the noise covariance matrix doesn't play a role in the selection of projection matrices. This method can be considered as a baseline method. Let us denote the collection of fingerprint codewords extracted from the 25 min clip using projection matrices from (i) as S^{svd} . We also let those extracted using projection matrices from (ii) be denoted as S^{rq} and those extracted using projection matrices from (iii) be denoted as S^{s-n} . Then, we compute the average number of collisions for the fingerprint codewords in each of the three sets namely S^{svd}, S^{rq} and S^{s-n} .

In order to study the collision performance of these three sets, we first add fingerprint codewords extracted from unique content (content that is unrelated to the 25 min clip) to the database. Then, we compute the average number of collisions as we increase the number of fingerprint codewords of unrelated content from 1 hr to upto 3hrs. Ideally, there should be no change in the average of number of collisions as we add unrelated content to the database.

Table 1 illustrates the collision performance for the three sets of projection matrices. The average of number of collisions for S^{svd} increases from 5.437 to 5.541 (see column C) as we add fingerprint codewords 3.37hrs of unrelated con-

Table 1: Collision Performance A: Number of fingerprint codewords; B: Number of hours in DB; C: Avg. num of collisions for ϕ_i ; D: Avg. num of collisions for ϕ_i^{rq} ; E: Avg. num of collisions for ϕ_i^{s-n} ;

| A | B | C(svd) | D(rq) | E(s-n) |
|--------|-------|--------|--------|--------|
| 20705 | 0.479 | 5.437 | 7.483 | 4.372 |
| 145813 | 3.375 | 5.541 | 43.114 | 4.49 |

Table 2: Robustness Performance ;

| | $\phi_i(svd)$ | ϕ_i^{rq} | ϕ_i^{s-n} |
|-------------|---------------|---------------|----------------|
| compression | 0.0222 | 0.0306 | 0.0207 |
| rotation | 0.0591 | 0.032 | 0.0534 |
| cropping | 0.1854 | 0.0986 | 0.1656 |
| bright up | 0.0972 | 0.0395 | 0.0865 |
| bright down | 0.1013 | 0.0483 | 0.0895 |
| asp ratio | 0.1000 | 0.0522 | 0.0926 |

tent. On the otherhand, the average number of collisions for S^{rq} increases from 7.483 to 43.114 (see column D) for the same increase in number of fingerprint codewords from unrelated content. The average number of collisions for S^{s-n} increases from 4.372 to 4.49 (see column D). We computed the increase in the average number of collisions for every additional hour of unrelated content included (referred to as collision slope) for all three methods from this table. The collision slopes for S^{svd}, S^{rq} and S^{s-n} are 0.035, 12.30 and 0.04 respectively. This implies that the projection matrices (ϕ_i^{s-n}) derived using the third method have similar collision performance as that of the base-line method (ϕ_i obtained using SVD of C^{feat}) whereas the collision performance of the projection matrices (ϕ_i^{rq}) obtained by the maximization of the Rayleigh coefficient is inferior to the baseline method and the third method.

3.2 Robustness analysis

In order to study the robustness of the fingerprints created by the three sets of projection matrices (ϕ_i, ϕ_i^{rq} , and ϕ_i^{s-n}), we created modified versions of the reference content. The attacks on the content include both geometric (rotation, cropping, aspect ratio change) and non-geometric attacks (compression, brightness change). Then, we compare the fingerprints of the original content to those of the attacked content and record the average Bit Error Rate (BER) for each of the three sets of projection matrices. Table 2 summarizes the robustness performance of these three sets of projection matrices. Since both ϕ_i^{rq} and ϕ_i^{s-n} are derived taking into the noise introduced by content modifications, they have a lower average BER than the fingerprints derived using ϕ_i . Note that the fingerprints derived using ϕ_i^{rq} have the lowest BER among the three. But the collision performance of this method was the worst among the three. This implies that ϕ_i^{rq} captures aspects of the feature that are not sensitive to content and are also most robust to noise induced by modifications of the content. The projection matrix ϕ_i^{s-n} has a similar collision performance to that of ϕ_i and also improves the robustness against attacks. For instance, the average BER for cropping attacks has improved from 0.1854 to 0.1656 (almost 2% improvement). For certain attacks such as compression, the improvement is not substantial even with ϕ_i^{s-n} .

3.3 Discussion

In the previous subsections, we illustrated the collision and robustness performance of the projections matrices derived using three methods. We showed that the third method ($s-n$) has a performance similar to that of the first method (svd) in terms of collision and has the best robustness among the three sets. In other words, the third method ($s-n$) improves robustness of the fingerprints while maintaining the collision statistics of the baseline method (svd). In this section, we further discuss and compare the three different types of projection matrices in all three cases. Figure 1 shows the first 8 projection matrices obtained through the SVD procedure. Each of the projection matrices are of the same dimension as the feature matrix Q (35×36). We recall that the number of rows correspond to 5×7 moment invariants extracted from each frame in the buffer and the number of columns correspond to the number of frames in the 3s buffer (3×12). Projection matrices on the top & bottom left of Figure 1 can be interpreted as those capturing appearance information alone from the feature matrix Q . This is represented by the fact that the values along the columns of the projection matrices 1 & 3 are similar for every row. Also, note that the patterns along rows are repeated 5 times. This is due to the fact that the 7 moment invariants computed from the 5 concentric circular regions are correlated and hence the projection matrix values are also correlated. The other projection matrices shown in the Figure 1 capture both appearance and motion information as can be seen from the changing patterns across the columns (time) for these projection matrices. Similarly, figure 2 shows the first eight projection matrices obtained using the difference between the signal and the noise covariance matrices (ϕ_i^{s-n}). We note that the projection matrices ϕ_i^{s-n} look very similar to ϕ_i and hence their collision performance is similar. In contrast to these projection matrices, the projection matrices obtained using maximization of Rayleigh Quotient look very different and are not effective in capturing the appearance or motion of the underlying video content (see Figure 3). Hence, the fingerprints derived using this set of projection matrices don't exhibit good collision properties. In contrast, they provide high robustness against content modifications.

We can also compare the robustness of the projected values against content modifications for the projection matrices (ϕ_i^{s-n} and ϕ_i). We consider the case where the original video is modified by compression, cropping and/or rotation. These operations add noise to the extracted feature vector. Let us represent the noisy feature matrix as \tilde{Q} . Also, let the projections obtained by projecting \tilde{Q} onto L projection matrices (ϕ_i 's) be represented as $\tilde{H}_1, \tilde{H}_2, \dots, \tilde{H}_L$. Then, the noise of the i^{th} projected value is $Z_i = H_i - \tilde{H}_i$. Here H_i is the projection of original feature matrix Q onto the i^{th} projection matrix (ϕ_i). Then, we can compute the ratio of the signal variance and the noise variance ($SNR_{svd} = \frac{\text{var}(H_i)}{\text{var}(Z_i)}$) as a measure of the robustness of the projections obtained using ϕ_i . The higher this ratio is, the lower is the fingerprint BER. Table 3 shows the SNR_{svd} for various modifications of the original content. We note that the cropping attack has the lowest SNR while the rotation attack has the highest SNR. Correspondingly, the BER is highest for cropping attack and lowest for rotation attacks. (see Table 2).

Similarly, let the projections obtained by projecting \tilde{Q} onto L projection matrices (ϕ_i^{s-n} 's) be represented as

Table 3: SNR table;

| | SNR_{svd} | SNR_{s-n} | $SNR_{s-n} - SNR_{svd}$ |
|-------------|-------------|-------------|-------------------------|
| rotation | 44.4927 | 45.6481 | 1.1553 |
| bright down | 23.1246 | 23.4984 | 0.3738 |
| bright up | 30.9174 | 31.5645 | 0.6471 |
| cropping | 8.9563 | 9.2605 | 0.3041 |
| asp ratio | 23.3428 | 23.8030 | 0.4602 |

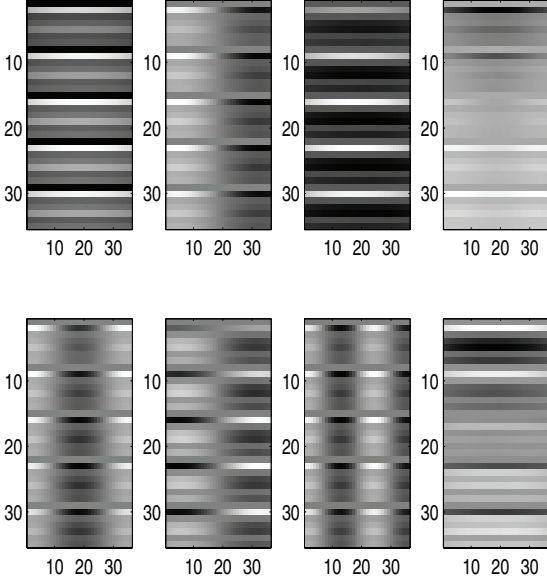


Figure 1: First eight projection matrices of ϕ_i .

$\widetilde{H}_1^{s-n}, \widetilde{H}_2^{s-n}, \dots, \widetilde{H}_L^{s-n}$. Then, the noise of the i^{th} projected value is $Z_i^{s-n} = H_i^{s-n} - \widetilde{H}_i^{s-n}$. Here H_i^{s-n} is the projection of the original feature matrix, Q onto the i^{th} projection matrix (ϕ_i^{s-n}). Again, we can compute the ratio of signal variance and noise variance ($SNR_{s-n} = \frac{var(H_i^{s-n})}{var(Z_i^{s-n})}$) as a measure of the robustness of the projections obtained using ϕ_i^{s-n} . Table 3 (second column) shows the SNR_{s-n} for various attacks on original content. Note that SNR_{s-n} is better than SNR_{svd} for all attacks as shown in the third column of this table.

4. CONCLUSIONS

In this paper, we studied two methods to improve the robustness property of projection based hashing methods. The two methods derive projection matrices taking into account the noise on the extracted features due to content modifications. Both methods use an off-line training set to estimate a signal covariance matrix and a noise covariance matrix. The first method considered in this paper, derives the projection matrix by maximizing the Rayleigh Quotient. The second method derives the projection matrices based on eigenvector analysis of the difference between the signal and noise covariance matrices. Both of these methods consider the effect of noise on extracted feature while deriving the fingerprints

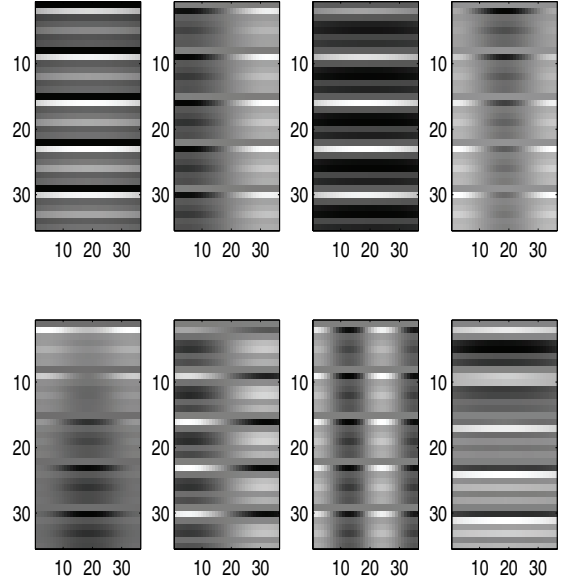


Figure 2: First eight projection matrices of ϕ_i^{s-n} .

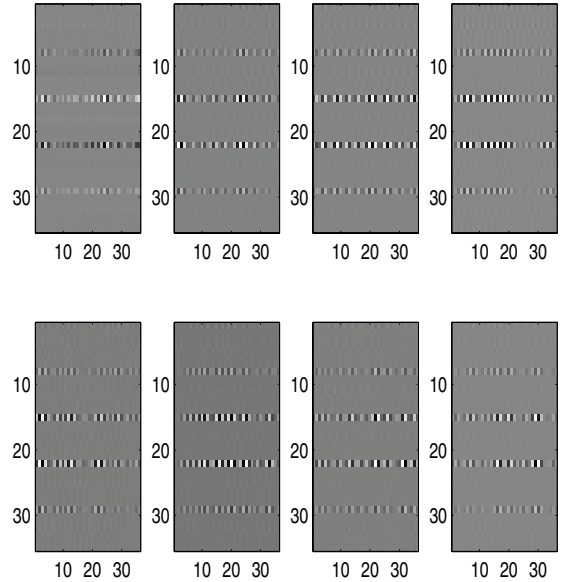


Figure 3: First eight projection matrices of $\phi_i^{r,q}$.

which is unlike our previous method proposed in [4]. Based on experimental results, we showed that the first method improves the robustness of the projected values but results in larger number of collisions than in the case of method in [4]. The second method improves the robustness of the projected values while maintaining the collision performance as in [4]. Although the robustness of the fingerprints obtained by using the second method improves, the amount of improvement is not substantial. Our future work will focus on further improving robustness while maintaining good collision characteristics.

5. REFERENCES

- [1] Burges, C.J.C. Platt, J.C. and Jana, S. Distortion discriminant analysis for audio fingerprinting. *IEEE Transactions on Speech and Audio Processing*, May 2003.
- [2] Hu, M.K. Visual pattern recognition by moment invariants. *IRE Trans. Info. Theory*, IT-8:179–187, 1962.
- [3] J.Fridrich and M.Goljan. Robust hash functions for digital watermarking. *ITCC*, 2000.
- [4] R.Radhakrishnan and C.Bauer. On improving the collision property of robust hashing based on projections. *Proc. of ICME*, 2009.