Robust Content Fingerprinting Algorithm Based on Sparse Coding

Yue nan Li, *Member, IEEE*

***Abstract*—Content fingerprinting is a powerful solution for media indexing, searching and digital right management, in which the perceptual content of digital media is summarized to a robust and discriminative digest. In this letter, we develop a general paradigm for image fingerprinting by exploiting the capability of sparse coding in capturing the visual characteristics of digital image. Furthermore, the impact of the dictionary for sparse coding on the performance of fingerprinting algorithm is analyzed. Accordingly, the problem of dictionary learning is studied in the context of content fingerprinting by incorporating the robustness and discriminability requirements. Comparative experiments indicate that the proposed work exhibits much higher content identification accuracy than the state-of-the-art ones, and the dictionary learned by the proposed work can substantially improve the performance of fingerprinting algorithm. In addition, our algorithm is highly efficient, and its average fingerprint computation time is less than 0.024s.**

***Index Terms*—Content identification, image fingerprinting, robust hashing, sparse coding.**

I. Introduction

RECENT explosive growth of digital media has imposed great challenges to media indexing, searching and copyright protection. Content fingerprinting is a promising technique for tackling these challenges. It aims at computing a unique identifier (i.e., fingerprint) for digital media that captures its perceptual characteristics, so that the contents of digital media can be identified via fingerprint comparison. Fingerprint should be robust and discriminative. The robustness property requires fingerprint remains stable when content-preserving manipulations are imposed on media file. At the same time, fingerprinting algorithm needs to map perceptually distinct media data to completely different fingerprints, and this capability is termed as discriminability. To alleviate storage and computational costs, fingerprint is expected to be compact and easy to compute. Content fingerprinting is sometimes referred to as robust hashing in the literature. It is worth mentioning that robust hashing also finds application in authentication, while the focus of fingerprinting is placed on content identification.

Most fingerprinting algorithms follow the two-component structure, as feature extraction and quantization. Below we present a brief review of image fingerprinting algorithms. The feature extraction schemes in image fingerprinting mainly stem from the research in image processing, according to which existing algorithms can be classified into four categories, namely, those based on spatial-domain features [1]–[3], transform-domain features [4]–[10], key points [11], [12] and coarse image representation [13]–[15]. The goal of feature quantization is to condense features while preserving their robustness and discriminability. The relationship-based quantizer is most commonly used due to its high robustness and simplicity. For instance, the binary fingerprints in [1], [3], [7] are generated by comparing features with a threshold. Likewise, the ordinal ranking based scheme collects the ranks of features as fingerprints [16]. Some feature quantization schemes were developed by adapting the quantizers in data compression. For example, the adaptive quantizer [17] computes fingerprints by encoding features with a randomized non-uniform scalar quantizer. To strike a better balance between robustness and discriminability, Li *et al.* proposed a dithered lattice vector quantizer based feature quantization scheme in [9]. Recently, the locality-sensitive-hashing (LSH), which was originally designed for approximate nearest neighbor searching, has also been exploited for feature quantization [18].

As mentioned above, the goal of image fingerprinting is to capture the perceptual essence of image. Recent research in signal processing indicates that sparse coding can provide compact and discriminative representation of signal due to its succinct nature [19], and it has shown promising results in several computer vision problems, such as face recognition [20], [21], object recognition [23] and content-based image retrieval [22] which bears some similarity to fingerprinting. The compact and discriminative properties of sparse coding are attractive to image fingerprinting. Inspired by this fact, we propose a sparse coding based fingerprinting algorithm in this letter, the novelty of which is twofold. First, it provides a general paradigm for content fingerprinting by exploiting the succinct nature of sparse coding. Second, dictionary learning has been investigated in the context of image fingerprinting, and we propose to enhance the robustness of fingerprinting algorithm by imposing the mutual coherence constraint.

The reminder of this letter is structured as follow. In Section II we describe the proposed fingerprinting algorithm

Manuscript received September 15, 2014; revised November 14, 2014; accepted January 19, 2015. Date of publication January 22, 2015; date of current version January 29, 2015. This work was supported by the National Natural Science Foundation of China under Grant 61202164, the Fund for Doctor Stations of Ministry of Education under Grant 20120032120026, the Tianjin Research Program of Application Foundation and Advanced Technology under Grant 14JCQNJC01500, the Shenzhen Strategic Emerging Industries Program under Grant ZDSY20120613125016389, and by the Seed Foundation of Tianjin University under Grant 2014XRG-0104. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Anthony Ho.

The author is with the School of Electronic and Information Engineering, Tianjin University, Tianjin, China (e-mail: ynli@tju.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2015.2395726

in detail. Experimental results are presented in Section III, and conclusions are drawn in the final section.

II. Sparse Coding Based Image Fingerprinting

A. Pre-processing and Feature Extraction

Prior to fingerprint computation, the input image is first smoothed by a 5×5 Gaussian low-pass filter with standard deviation 3 and then normalized to 128×128 pixels via bilinear interpolation based resizing. The low-pass filter can attenuate the influence of distortions (e.g., additive noise), and resizing makes the length of fingerprint independent of the size of input image. In this work, totally M square blocks $\{\boldsymbol{B}_i\}_{i=1}^{M}$ are randomly sampled from the normalized image using a secret key as in [14], and block widths are drawn from the uniform distribution $U(8, 24)$. Each block is characterized by its mean intensity, and all the mean intensities are sequentially grouped into 5-dimensional vectors: $\{\boldsymbol{y}_i \in \mathbb{R}^5\}_{i=1}^{N=M/5}$. We normalize each feature vector to zero-mean, which can result in better invariance to illumination changes and reduce the correlation between feature vectors. It deserves a mention that the proposed fingerprint computation paradigm is also applicable to other features. Mean-intensity is adopted due to its outstanding robustness, discriminability and efficiency. The comparative studies in [24] show that block-based spatial-domain features are most suitable for visual fingerprinting, and they are the winner in overall performance.

B. Feature Coding and Dictionary Learning

Fingerprints are generated using the sparse representation of feature vectors. Given an over-complete dictionary $\boldsymbol{D} = [\boldsymbol{d}_1, \cdots, \boldsymbol{d}_K] \in \mathbb{R}^{5\times K}$ with K atoms, sparse coding seeks to representing a feature vector $\boldsymbol{y}_i$ using at most T_0 atoms:

$$\min_{\boldsymbol{x}_i \in \mathbb{R}^K} \|\boldsymbol{y}_i - \boldsymbol{D}\boldsymbol{x}_i\|_2, \quad \text{s.t.} \quad \|\boldsymbol{x}_i\|_0 \leq T_0. \tag{1}$$

where $\boldsymbol{x}_i$ is the sparse representation of $\boldsymbol{y}_i$, and $\|\boldsymbol{x}_i\|_0$ is the ℓ_0-norm of $\boldsymbol{x}_i$. The atoms with non-zero weights in $\boldsymbol{x}_i$ are termed as active atoms. As (1) shows, sparse coding maps a given feature vector to a small number of most representative atoms. Hence, sparse representation can provide an informative description of feature vector and tolerate a certain degree of distortion [19]. We solve (1) by the orthogonal matching pursuit (OMP) algorithm [25] using its Matlab implementation in [26]. OMP sequentially identifies the active atoms that can best represent $\boldsymbol{y}_i$, and at most T_0 iterations are required. Since T_0 is usually sufficiently small, the complexity of solving (1) is quite low. As has been verified by experiments, this model demonstrates distinct advantage over the one based on ℓ_1-norm minimization in terms of complexity.

Fig. 1 shows the flowchart of the proposed algorithm, from which we see that all the fingerprints are computed using a fixed dictionary trained off-line. The performance of fingerprinting algorithm largely depends on this dictionary. Thus, before proceeding with fingerprint computation, we shall first discuss the problem of dictionary learning.

From a robustness point of view, sparse vectors are desired to be stable against content-preserving manipulations. As mentioned above, the OMP algorithm iteratively seeks T_0 most

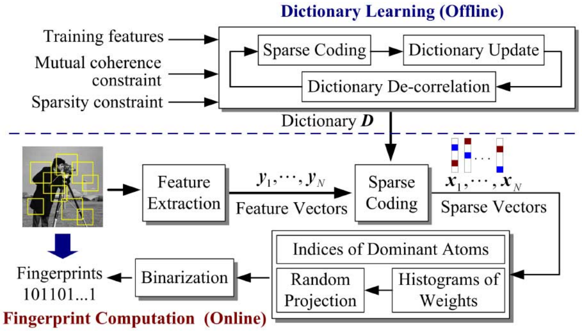

Fig. 1. Flowchart of the proposed fingerprinting algorithm.

representative atoms for $\boldsymbol{y}_i$. At each iteration, OMP computes the absolute inner products between the current approximation residue and all the available atoms, and the one yielding the largest absolute inner product is chosen as an active atom. Therefore, the atoms should be uncorrelated with each other; otherwise, a small disturbance could easily transfer the feature vector to another set of active atoms (i.e., alter the positions of non-zero entries in $\boldsymbol{x}_i$). The correlation between atoms in a dictionary $\boldsymbol{D}$ can be measured by mutual coherence [19]:

$$\mu(\boldsymbol{D}) = \max_{1\leq i,j\leq K, i\neq j} \frac{|\langle \boldsymbol{d}_i, \boldsymbol{d}_j\rangle|}{\|\boldsymbol{d}_i\|_2 \cdot \|\boldsymbol{d}_j\|_2} \tag{2}$$

Since atoms are normalized to unit ℓ_2-norm, $\mu(\boldsymbol{D})$ is in fact the largest absolute inner product between different atoms. Obviously, a dictionary with low $\mu(\boldsymbol{D})$ can guarantee the stability of sparse vector, and we thus propose to learn dictionary under the mutual coherence constraint. Some recent studies on pattern classification also support this observation [20], [28]. For instance, the research in sparse coding based face recognition implies that the training set (i.e., dictionary) with wide variation can result in higher robustness [20]. We now look at the discriminability requirement. We desire that the sparse representations of the feature vectors extracted from perceptually distinct images show remarkable differences. In other words, sparse coding should preserve the discriminability of feature vectors. In this regard, the discriminability of fingerprinting algorithm partially depends on the representational capability of dictionary, so the learned dictionary should minimize the approximation errors of training samples. By addressing the robustness and discriminability requirements, we cast dictionary learning into a constrained optimization problem. Denote the training set by $\boldsymbol{V} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_S]$, then the objective of dictionary learning can be expressed as:

$$\begin{aligned} \min_{\boldsymbol{D},\boldsymbol{X}} \quad & \|\boldsymbol{V} - \boldsymbol{D}\boldsymbol{X}\|_2, \\ \text{s.t.} \quad & \mu(\boldsymbol{D}) \leq \mu_0; \quad \|\boldsymbol{x}_i\|_0 \leq T_0, 1 \leq i \leq S. \end{aligned} \tag{3}$$

where $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_S]$ is the sparse matrix, whose i-th column $\boldsymbol{x}_i$ is the sparse vector of $\boldsymbol{v}_i$. K-SVD [27] is one of the most effective algorithms for dictionary learning, in which approximation error is minimized by alternatively updating $\boldsymbol{X}$ and $\boldsymbol{D}$. However, it does not take the mutual coherence constraint into account. To solve (3), we run K-SVD in conjunction with dictionary de-correlation. The de-correlation strategy proposed in [28] can fix the mutual coherence of a dictionary below a constraint while minimizing its approximation error, which is achieved by iterative projection and rotation (IPR). Algorithm 1 outlines the procedures of dictionary learning.

Algorithm 1 Dictionary learning under sparsity constraint T_0 and mutual coherence constraint μ_0

Input: Training set $\boldsymbol{V} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_S]$, Maximum number of iterations: MaxIter $= 250$ and error tolerance: $\epsilon = 10^{-4}S$, where S is the size of training set.

Initialize: Initial dictionary $\boldsymbol{D}^{(0)}$, $J = 0$.

repeat

— $J \leftarrow J + 1$

— Find the sparse representation of each $\boldsymbol{v}_i (1 \leq i \leq S)$ with respect to $\boldsymbol{D}^{(J-1)}$ via OMP:
$\boldsymbol{x}_i^{(J)} = \arg\min_{\boldsymbol{x}_i} \|\boldsymbol{v}_i - \boldsymbol{D}^{(J-1)}\boldsymbol{x}_i\|_2, \text{s.t.} \|\boldsymbol{x}_i\|_0 \leq T_0, \forall i.$

— Fix $\boldsymbol{X}^{(J)}$ and update dictionary by K-SVD:
$\boldsymbol{D}^{(J)} = \arg\min_{\boldsymbol{D}} \|\boldsymbol{V} - \boldsymbol{D}\boldsymbol{X}^{(J)}\|_2.$

— De-correlate $\boldsymbol{D}^{(J)}$ using the IPR algorithm:
$\boldsymbol{D}^{(J)} \leftarrow \text{IPR}(\boldsymbol{V}, \boldsymbol{D}^{(J)}, \boldsymbol{X}^{(J)}, \mu_0)$

Until $J >$ MaxIter or $\|\boldsymbol{V} - \boldsymbol{D}^{(J)}\boldsymbol{X}^{(J)}\|_2 < \epsilon$

C. Fingerprint Generation

In this subsection, we introduce the mapping from sparse vectors to fingerprint. For each feature vector, the corresponding sparse vector indicates the indices of its active atoms in the dictionary and their weights. Among all the active atoms of a feature vector, the one with the largest absolute weight (hereafter referred to as *dominant* atom) is most stable against content-preserving manipulations. Let us concatenate the indices of the dominant atoms corresponding to N feature vectors as $\boldsymbol{A} = \{\arg\max_k |\boldsymbol{x}_i(k)|\}_{i=1}^N$, then the first N fingerprint bits are generated by comparing each entry of $\boldsymbol{A}$ with their median. If $\boldsymbol{A}(i) > \text{Median}(\boldsymbol{A})$, the i-th bit is set to '1'; otherwise, a '0' is output. In addition, the weights of atoms are also involved in fingerprint computation. For better discriminability, their signs are taken into account, and we aggregate negative and positive weights into different histograms. In both histograms, each bin corresponds to an atom. Consider a dictionary with K atoms and take the histogram of negative weights ($\boldsymbol{H}^-$) for instance, the k-th ($k = 1, \cdots, K$) bin accumulates the absolute values of the negative weights associating with the k-th atom in all sparse vectors, as (4) shows. Similarly, we can construct the histogram of positive weights by (5).

$$\boldsymbol{H}^-(k) = \sum\nolimits_{i=1}^N |\boldsymbol{x}_i(k)|, \quad \forall \boldsymbol{x}_i(k) < 0 \tag{4}$$

$$\boldsymbol{H}^+(k) = \sum\nolimits_{i=1}^N \boldsymbol{x}_i(k), \quad \forall \boldsymbol{x}_i(k) > 0. \tag{5}$$

These two histograms are concatenated as $\boldsymbol{H} = [\boldsymbol{H}^+, \boldsymbol{H}^-] \in \mathbb{R}^{2K}$. Unlike bag-of-words [29] that aggregates the counts of visual words' occurrence into histogram, the proposed work takes account of the weights of atom in representing feature vectors, since they are more informative. We project $\boldsymbol{H}$ onto Q random vectors:

$$\left[\frac{\langle \boldsymbol{p}_1, \boldsymbol{H}\rangle}{\|\boldsymbol{p}_1\|_2}, \frac{\langle \boldsymbol{p}_2, \boldsymbol{H}\rangle}{\|\boldsymbol{p}_2\|_2}, \ldots, \frac{\langle \boldsymbol{p}_Q, \boldsymbol{H}\rangle}{\|\boldsymbol{p}_Q\|_2}\right], \tag{6}$$

where the entries of $\boldsymbol{p}_i \in \mathbb{R}^{2K} (i = 1, \cdots, Q)$ are randomly drawn from the normal distribution $N(0,1)$ via a key. The projections in (6) are mapped to Q binary bits by comparing each with their median, as in binarizing the indices of dominant atoms. Finally, the binary bits derived from the indices of dominant atoms and the histograms of weights are concatenated to form the final fingerprint.

III. Experimental Results

The performance of the proposed algorithm was evaluated by content identification experiments under the following parameter settings. The number of feature vectors $N = 80$, the sparsity constraint $T_0 = 3$, the number of projections $Q = 10$, and the dictionary contains $K = 15$ atoms learned from $S = 8000$ training features under the mutual coherence constraint $\mu_0 = 0.45$. The criterion for parameter setting is to trade off robustness, discriminability, compactness and efficiency. The testing database contains 130,000 gray-scale images[1], including 2,000 original images (collected from the public database ImageNet [30] and downloaded from Internet, with widths and heights vary from 158 to 3488) and their distorted copies produced by the manipulations in Table I[2]. The perceptually identical copies of each original image were identified by fingerprint comparison. The perceptual equality of two images was determined by comparing the normalized Hamming distance (NHD) between their fingerprints with a threshold τ. By sweeping τ from 0 to 1, the false acceptance rate (FAR) and false rejection rate (FRR) were computed, according to which the receiver-operating-characteristics (ROC) curve was plotted. The proposed work was compared with two state-of-the-art algorithms: the one based on ring partition and nonnegative matrix factorization (Ring-NMF) [15] as well as the one based on Gabor filtering and dithered lattice vector quantization (GF-DLVQ) [9]. The parameter settings of Ring-NMF and GF-DLVQ are exactly the same as described in [15] and [9], respectively, and experiments were conducted using the source codes provided by authors. From Fig. 2(a), we see that our proposed algorithm demonstrates the best performance, and its ROC curve is almost identical to the ideal one corresponding to error-free content identification. Moreover, to quantify the performance of each algorithm with respect to different manipulations, their equal error rates (EER) in content identification are tabulated in Table II. As expected, the proposed work exhibits the lowest EER in most cases. Moreover, our algorithm also outperforms the other two in terms of compactness. The output fingerprint is 90 bits long, which is more compact than those produced by Ring-NMF (64 real values) and GF-DLVQ (120 bits).

As discussed earlier, learning an incoherent dictionary is vital for the sparse coding based fingerprinting algorithm. This

[1]Training images for dictionary learning were not included in testing set.

[2]The manipulations in Table I were implemented using the functions in Matlab image processing toolbox, and the notation $a : s : b$ means that the strength of manipulation varies from a to b with step size s.

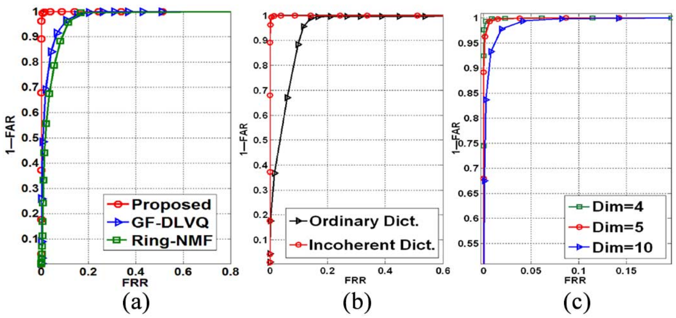

Fig. 2. Comparisons on ROC curve. (a) ROC curves of the proposed algorithm, GF-DLVQ and Ring-NMF, (b) ROC curves corresponding to ordinary and incoherent dictionaries, (c) ROC curves corresponding to different dimensions of feature vector.

TABLE I
CONTENT-PRESERVING MANIPULATIONS

Manipulation	Strength
JPEG Compression	Quality factor $\in \{10, 20:20:80, 90\}$
Gaussian Filtering	Filter size: 3, Standard deviation $\in \{0.3:0.1:1\}$
Circular Filtering	Radius $\in \{0.1:0.2:0.9, 1:1:3\}$
Gaussian Noise	Zero mean, variance $\in \{0.01, 0.05, 0.1:0.1:0.3, 0.4:0.2:1\}$
Speckle Noise	Zero mean, variance $\in \{0.01, 0.05, 0.1:0.1:0.3\}$
Rotation+Cropping	Degree $\in \{1:1:4, 6:2:10\}$
Histogram Equalization	Number of gray levels $\in \{2, 8, 16, 32:32:256\}$
Gamma Correction	$\gamma \in \{0.55:0.1:0.95, 1.05:0.1:1.45\}$

TABLE II
COMPARISON ON EER

Manipulation	Proposed	GF-DLVQ	Ring-NMF
JPEG Compression	$\mathbf{1.1 \times 10^{-4}}$	5.0×10^{-4}	1.6×10^{-3}
Gaussian Filtering	1.7×10^{-5}	$\mathbf{8.0 \times 10^{-6}}$	1.8×10^{-3}
Circular Filtering	$\mathbf{4.8 \times 10^{-5}}$	7.1×10^{-5}	3.1×10^{-3}
Gaussian Noise	$\mathbf{1.6 \times 10^{-3}}$	0.15	0.23
Speckle Noise	$\mathbf{9.4 \times 10^{-4}}$	2.2×10^{-2}	2.2×10^{-2}
Rotation+Cropping	8.1×10^{-3}	$\mathbf{1.9 \times 10^{-4}}$	2.8×10^{-3}
Histogram Equalization	$\mathbf{1.6 \times 10^{-2}}$	0.13	0.14
Gamma Correction	$\mathbf{7.4 \times 10^{-4}}$	5.6×10^{-2}	2.2×10^{-2}

set of experiments demonstrates the impact of atom de-correlation on the performance of fingerprinting algorithm. The above content identification experiment was repeated using an ordinary dictionary learned without the mutual coherence constraint. Fig. 2(b) displays the ROC curves corresponding to the ordinary and incoherent dictionaries, which clearly shows the superiority of the incoherent one. Our analysis in Section II-B implies that an incoherent dictionary can provide fingerprinting algorithm with higher robustness. Accordingly, we compared the robustness of the fingerprints computed using the two dictionaries, and Gamma correction ($\gamma = 0.75$) was taken as the example of distortion. For the ordinary dictionary, the average NHD between the fingerprints of original and distorted images is 0.18, while the value corresponding to the incoherent dictionary is 0.06.

We now demonstrate the influence of feature vector dimension (Dim) on the performance of fingerprinting algorithm. A large Dim results in less feature vectors and more compact fingerprint. The lengths of fingerprint corresponding to $Dim = 4$, 5 and 10 are 110, 90 and 50 bits, respectively. However, increasing Dim degrades the performance of fingerprinting algorithm. As depicted by Fig. 2(c), $Dim = 4$ and 5 show equally good performance, while the case in $Dim = 10$ is much worse. Therefore, to balance the compactness of fingerprint and the performance of fingerprinting algorithm, we set $Dim = 5$.

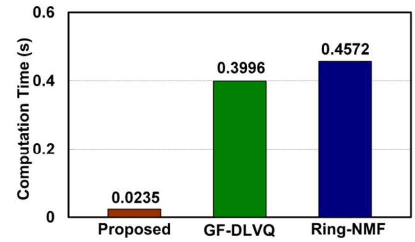

Fig. 3. Comparison on computation time.

To guarantee the security of image fingerprinting in copyright detection, both the feature extraction and binarization approaches described in Section II-A) and C) are key-dependent. For a secure fingerprinting algorithm, its output fingerprint needs to be sensitive to secret key, making it computationally impossible to estimate the fingerprint of a given image without knowing the secret key. To evaluate the security of the proposed work, we repeatedly computed the fingerprints of a given image using 2,000 different keys. Experimental results show that the average NHD between the fingerprints computed under different keys is 0.489, indicating that output fingerprints are uncorrelated with each other and highly sensitive to secret key.

The last set of experiments assessed the efficiency of the proposed algorithm by running 500 trials of fingerprint computation on a PC with Intel Core-i7 3.4 GHz CPU and 3.2 GB RAM. The average time spent by the proposed work in computing the fingerprint of a 512×512 image is 0.0235s, which is sufficient for real-time applications. Fig. 3 visualizes the comparison on computation time among the proposed work, GF-DLVQ and Ring-NMF, and we see that the proposed one is 16 and 18 times faster than GF-DLVQ and Ring-NMF, respectively. For comparison, we have also measured the complexity of the fingerprinting algorithm that uses ℓ_1-norm minimization based sparse coding, where the ℓ_1-min problem: $\min \|\boldsymbol{x}_i\|_1, \text{s.t.} \boldsymbol{y}_i = \boldsymbol{D}\boldsymbol{x}_i$, was solved by the gradient projection based algorithm in [31]. Its average fingerprint computation time is 1.0585s, which is 44 times slower than the proposed work.

IV. CONCLUSIONS

This letter has presented a sparse coding based fingerprinting algorithm. Our studies have revealed that sparse coding can provide a promising avenue for content fingerprinting with the following merits. Firstly, sparse coding can capture the intrinsic characteristics of perceptual contents. Secondly, the incoherent dictionary based sparse coding possesses a high degree of distortion tolerance, which is beneficial to the robustness of fingerprinting algorithm. Thirdly, the succinct property of sparse coding can well satisfy the compactness requirement of fingerprint.

References

[1] J. Fridrich and M. Goljan, "Robust hash functions for digital watermarking," in *Proc. IEEE Conf. Information Technology: Coding and Computing*, Mar. 2000, pp. 178–183.

[2] F. Khelifi and J. M. Jiang, "Perceptual image hashing based on virtual watermark detection," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 981–994, Apr. 2010.

[3] Y. S. Choi and J. H. Park, "Image hash generation method using hierarchical histogram," *Multimedia Tools Applicat.*, vol. 61, no. 1, pp. 181–194, Nov. 2012.

[4] R. Venkatesan, S. M. Koon, M. H. Jakubowski, and P. Moulin, "Robust image hashing," in *Proc. IEEE Conf. Image Processing*, Sep. 2000, vol. 3, pp. 664–666.

[5] S. S. Jin, J. Haitsma, T. Kalker, and C. D. Yoo, "A robust image fingerprinting system using the radon transform," *Signal Process. Image Commun.*, vol. 19, no. 4, pp. 325–339, Apr. 2004.

[6] A. Swaminathan, Y. Mao, and M. Wu, "Robust and secure image hashing," *IEEE Trans. Inf. Forensics Secur.*, vol. 1, no. 2, pp. 215–230, Jun. 2006.

[7] D. Wu, X. B. Zhou, and X. M. Niu, "A novel image hash algorithm resistant to print-scan," *Signal Process.*, vol. 89, no. 12, pp. 2415–2424, Dec. 2009.

[8] Y. Zhao, S. Wang, X. Zhang, and H. Yao, "Robust hashing for image authentication using zernike moments and local features," *IEEE Trans. Inf. Forensics Secur.*, vol. 8, no. 1, pp. 55–63, Jan. 2013.

[9] Y. Li, Z. Lu, C. Zhu, and X. Niu, "Robust image hashing based on random gabor filtering and dithered lattice vector quantization," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1963–1980, Apr. 2012.

[10] M. M. Esmaeili, M. Fatourechi, and R. K. Ward, "A robust and fast video copy detection system using content-based fingerprinting," *IEEE Trans. Inf. Forensics Secur.*, vol. 6, no. 1, pp. 213–226, Mar. 2011.

[11] V. Monga and B. L. Evans, "Perceptual image hashing via feature points: Performance evaluation and tradeoffs," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3453–3466, Nov. 2006.

[12] X. Lv and Z. J. Wang, "Perceptual image hashing based on shape contexts and local feature points," *IEEE Trans. Inf. Forensics Secur.*, vol. 7, no. 3, pp. 1081–1093, Jun. 2012.

[13] S. S. Kozat, R. Venkatesan, and M. K. Mihçak, "Robust perceptual image hashing via matrix invariants," in *Proc. IEEE Conf. Image Processing*, Oct. 2004, vol. 5, pp. 3443–3446.

[14] V. Monga and M. K. Mhçak, "Robust and secure image hashing via non-negative matrix factorizations," *IEEE Trans. Inf. Forensics Secur.*, vol. 2, no. 3, pp. 376–390, Sep. 2007.

[15] Z. J. Tang, X. Q. Zhang, and S. C. Zhang, "Robust perceptual image hashing based on ring partition and NMF," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 3, pp. 711–724, Mar. 2014.

[16] X. S. Hua, X. Chen, and H. J. Zhang, "Robust video signature based on ordinal measure," in *Proc. IEEE Conf. Image Processing*, Oct. 2004, vol. 1, pp. 685–688.

[17] M. K. Mhçak and R. Venkatesan, "A perceptual audio hashing algorithm: A tool for robust audio identification and information hiding," in *Proc. Workshop Information Hiding*, Pittsburgh, PA, USA, Apr. 2001, vol. 2137, pp. 51–65.

[18] M. Li and V. Monga, "Compact video fingerprinting via structural graphical models," *IEEE Trans. Inf. Forensics Secur.*, vol. 8, no. 11, pp. 1709–1721, Nov. 2013.

[19] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, Mar. 2009.

[20] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[21] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Jun. 2010, pp. 2691–2698.

[22] T. Ge, Q. Ke, and J. Sun, "Sparse-coded features for image retrieval," in *Proc. Brit. Machine Vision Conf.*, Sep. 2013, pp. 1–11.

[23] Z. Ji, J. Theiler, R. Chartrand, G. Kenyon, and S. P. Brumby, "Decoupling sparse coding of SIFT descriptors for large-scale visual recognition," in *Proc. SPIE*, May 2013, vol. 8750, p. 87500K1-11.

[24] J. Lu, "Video fingerprinting for copy identification: From research to industry applications," in *Proc. SPIE Media Forensics and Security*, Feb. 2009, vol. 7254, pp. 1–15.

[25] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. Conf. Signals, Systems and Computers*, Pacific Grove, CA, USA, Nov. 1993, vol. 1, pp. 40–44.

[26] Matlab codes of OMP [Online]. Available: http://www.cs.technion.ac.il/~elad/Various/KSVD_Matlab_ToolBox.zip(OMP.m)

[27] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[28] D. Barchiesi and M. D. Plumbley, "Learning incoherent dictionaries for sparse approximation using iterative projections and rotations," *IEEE Trans. Signal Process.*, vol. 61, no. 8, pp. 2055–2065, Apr. 2013.

[29] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. Int. Conf. Computer Vision*, Oct. 2003, pp. 1470–1477.

[30] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and F. F. Li, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Jun. 2009, pp. 248–255.

[31] L-1 Benchmark Package [Online]. Available: http://www.eecs.berkeley.edu/~yang/software/l1benchmark/l1benchmark.zip