LETTER

Approximate Nearest Neighbor Based Feature Quantization Algorithm for Robust Hashing

Yue nan LI[†a)], *Nonmember and* **Hao LUO**[††b)], *Member*

SUMMARY In this letter, the problem of feature quantization in robust hashing is studied from the perspective of approximate nearest neighbor (ANN). We model the features of perceptually identical media as ANNs in the feature set and show that ANN indexing can well meet the robustness and discrimination requirements of feature quantization. A feature quantization algorithm is then developed by exploiting the random-projection based ANN indexing. For performance study, the distortion tolerance and randomness of the quantizer are analytically derived. Experimental results demonstrate that the proposed work is superior to state-of-the-art quantizers, and its random nature can provide robust hashing with security against hash forgery.

key words: *robust hash function, feature quantization, approximate nearest neighbor, performance analysis*

1. Introduction

Robust hash function is a mapping from the perceptual content of media data to a short digest, and it was initially proposed to overcome the limitation of cryptographic hash functions in media authentication. Cryptographic hash functions such as SHA-1 are designed to be highly sensitive to the digital representation of the message. However, media authentication aims at verifying the authenticity of the perceptual content, instead of the digital representation, of media data. Accordingly, media authentication cannot be achieved by verifying the media data in a bit-by-bit manner. As a consequence, robust hashing was proposed to enable media authentication. Robust hashing is developed by capturing the perceptual essence of the media data, and the hash value can tolerate content-preserving manipulations. Meanwhile, robust hashing should possess adequate sensitivity to malicious tampering and perceptually distinct media data, which is referred to as the discrimination requirement.

Feature extraction and quantization are two primary concerns in developing robust-hashing algorithms. Compact features are extracted as the descriptor of perceptual content and then quantized to produce fix-length hash. In this letter, we address the feature quantization stage of robust hashing. Little research has been conducted on this topic. The hash values in a vast majority of robust-hashing algorithms are generated by quantizing features using scalar quantizers (SQ). A key-dependent quantization algorithm was proposed in [1], where each feature is randomly mapped to one of its neighboring indexes according to the secret key. In [2], random hash values are generated by dithering features and then quantizing the dithered features via distributed coding. Similarly, Tagliasacchi *et al.* opted for the Wyner-Ziv codec in their hashing system [3]. To meet the security requirement of content authentication and make a good balance between robustness and discrimination, the feature vectors in [4] are quantized by dithered lattice vector quantizer (DLVQ) that is a kind of random quantizer in multidimensional space.

By analyzing the problem of feature quantization, we find that the features of perceptually identical media data can be modeled as approximate nearest neighbors (ANN) in the feature set and the robustness and discrimination requirements are in good agreement with those of ANN indexing. Consequently, we propose to quantize feature vectors by means of ANN indexing. It has been observed that the ANN based quantizer can exhibit superior performance than both SQ and DLVQ. Moreover, its random nature can benefit the security of robust hashing.

The rest of this letter is organized as follows. Section 2 describes the formulation of feature quantization and then introduces the ANN based quantizer. Analytical and experimental results are provided in Sect. 3 and Sect. 4, respectively. Finally, conclusions are drawn in Sect. 5.

2. ANN Based Feature Quantization Algorithm

2.1 Problem Statement and Formulation

Given the media data $\boldsymbol{M}$ and its feature $\boldsymbol{f}$, let us denote the feature extracted from the perceptually identical counterpart of $\boldsymbol{M}$ as $\boldsymbol{f}_{\text{iden}}$, denote the one extracted from the media that is perceptually distinct from $\boldsymbol{M}$ as $\boldsymbol{f}_{\text{dist}}$. Consequently, the robustness and discrimination requirements of the quantizer $Q(\cdot)$ can be expressed as:

- Robustness: $\Pr\left(Q(\boldsymbol{f}) = Q(\boldsymbol{f}_{\text{iden}})\right) \geq 1 - \varepsilon_1$
- Discrimination: $\Pr\left(Q(\boldsymbol{f}) = Q(\boldsymbol{f}_{\text{dist}})\right) \leq \varepsilon_2$

where $\Pr(\cdot)$ denotes the probability of an event, $0 < \varepsilon_1, \varepsilon_2 \ll 1$. It has been observed that $\boldsymbol{f}$ and $\boldsymbol{f}_{\text{dist}}$ are usually far apart from each other, while most of the distorted features lie near to $\boldsymbol{f}$. However, it should be noted that although $\boldsymbol{f}_{\text{iden}}$ and $\boldsymbol{f}$ may be quite similar, $\boldsymbol{f}_{\text{iden}}$ is not necessarily the nearest neighbor of $\boldsymbol{f}$, but actually one of its *approximate* nearest

Manuscript received April 18, 2012.
Manuscript revised July 20, 2012.

[†]The author is with the School of Electronic and Information Engineering, Tianjin University, P. R. China.
[††]The author is with the School of Aeronautics and Astronautics, Zhejiang University, P. R. China.
a) E-mail: ynli@tju.edu.cn
b) E-mail: luohao@zju.edu.cn

DOI: 10.1587/transinf.E95.D.3109

neighbors. According to [5], given a point $\boldsymbol{q}$ and the point set P, $\boldsymbol{p} \in P$ is an ϵ-approximate nearest neighbor (ANN) of $\boldsymbol{q}$ if for all $\boldsymbol{p}' \in P$, $d(\boldsymbol{p}, \boldsymbol{q}) \leq (1+\epsilon)d(\boldsymbol{p}', \boldsymbol{q})$. Apparently, ANN relaxes the concept of nearest neighbor by defining a neighborhood around $\boldsymbol{q}$. Likewise, in the context of robust hashing, most of the distorted features concentrate in the neighborhood of $\boldsymbol{f}$. In this regard, ANN can well model the relationships between the features of perceptually identical media data. From this point of view, the object of feature quantization is consistent with that of ANN indexing.

2.2 Feature Quantization via ANN Indexing

The random-projection based locality-sensitive hashing[†] (LSH) [6], which is an effective solution for ANN indexing, is exploited in this work to devise the quantizer for robust hashing. The random-projection based LSH $Q(\cdot)$ is (r_1, r_2, p_1, p_2)-sensitive as shown below ($r_1 < r_2, p_1 > p_2$) [6], which agrees with the robustness and discrimination requirements of feature quantization.

- if $d(\boldsymbol{p}, \boldsymbol{q}) \leq r_1$, $\Pr(Q(\boldsymbol{p}) = Q(\boldsymbol{q})) \geq p_1$
- if $d(\boldsymbol{p}, \boldsymbol{q}) \geq r_2$, $\Pr(Q(\boldsymbol{p}) = Q(\boldsymbol{q})) \leq p_2$.

In addition, the random nature of the LSH can benefit the security of robust hashing by endowing the quantizer with the property of key-dependent. Based on these facts, a feature vector $\boldsymbol{f}$ can be quantized to hash value via the mapping defined by the random-projection based LSH as

$$Q(\boldsymbol{f}) = \left\lfloor \frac{\boldsymbol{a} \cdot \boldsymbol{f} + b}{r} \right\rfloor, \tag{1}$$

where $\boldsymbol{a} \cdot \boldsymbol{f}$ is the dot product, $\boldsymbol{a}$ is a vector whose elements are independently drawn from normal distribution, b is a random variable following the uniform distribution $U(0, r)$, and $\lfloor \cdot \rfloor$ denotes the rounding operation. In robust hashing, the parameters $\boldsymbol{a}$ and b are randomly generated under secret key. As shown in Fig. 1, the mapping can be decomposed into the following stages: the feature vector is first projected onto a random line with direction $\boldsymbol{a}$, the projection is then shifted with a random amount b, and the line is finally chopped into segments of size r. In this manner, neighboring vectors can fall into the same segment (i.e., mapped to identical hash) with high probability. To ensure features can be quantized to fix-length hash, they are first normalized into $[-16, 16)$ and then grouped into 4-D vectors. Four hash values are computed for each vector under distinct $\boldsymbol{a}$ and b. The final hash string is obtained by concatenating the hash values of each feature vector, and we use the normalized Hamming distance (NHD) to measure the distance between two hash strings.

3. Performance Analysis of the ANN Based Feature Quantization Algorithm

3.1 Analysis on Distortion Tolerance

The ANN based quantizer can map the features of perceptually identical media data to the same hash with high probability. Therefore, it can tolerate a certain amount of content-preserving distortion. In what follows, the robustness of the ANN based quantizer will be quantified by estimating the average distortion it can tolerate. Given the original feature vector $\boldsymbol{f}$ and the distorted one $\boldsymbol{f}_d$, the distortion on feature vector can be expressed as $\boldsymbol{d} = \boldsymbol{f}_d - \boldsymbol{f}$. Denote by $\boldsymbol{I}$ and $\boldsymbol{I}_d$ the hash values of the original and distorted features, and denote by $D(\boldsymbol{I}_d, \boldsymbol{I})$ the NHD between $\boldsymbol{I}_d$ and $\boldsymbol{I}$. Let $\Delta = \|\boldsymbol{d}\|_2$, then the average distortion that the quantizer can tolerate is

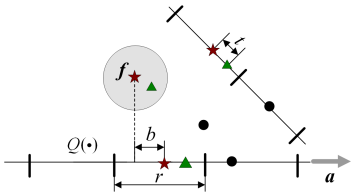

Fig. 1 Graphic illustration of the mapping in (1).

$$\overline{\Delta}_{ANN} = \int_0^{\infty} \Delta \cdot \Pr(D(\boldsymbol{I}_d, \boldsymbol{I}) = 0 | \Delta) d\Delta. \tag{2}$$

Since the four hash values of each feature vector are independently generated via the mapping $Q(\cdot)$, we have

$$\Pr(D(\boldsymbol{I}_d, \boldsymbol{I}) = 0 | \Delta) = \Pr(Q(\boldsymbol{f}_d) = Q(\boldsymbol{f}) | \Delta)^4. \tag{3}$$

As can be seen from Fig. 1, the probability of $Q(\boldsymbol{f}_d) = Q(\boldsymbol{f})$ depends on the distance between the projections of the two vectors, namely, $|\boldsymbol{a} \cdot \boldsymbol{f}_d - \boldsymbol{a} \cdot \boldsymbol{f}| = |\boldsymbol{a} \cdot \boldsymbol{d}|$. We focus on the case where the elements of $\boldsymbol{a}$ are drawn from the normal distribution $N(0, 1)$. It is easy to verify that $(\boldsymbol{a} \cdot \boldsymbol{d}) \sim N(0, \Delta^2)$ and the probability density function (pdf) of $t = |\boldsymbol{a} \cdot \boldsymbol{d}|$ is

$$p(t|\Delta) = \frac{2}{\Delta\sqrt{2\pi}} e^{-\frac{t^2}{2\Delta^2}}, \quad (t \geq 0). \tag{4}$$

As can be inferred from Fig. 1 that when $t \in [0, r]$, $Q(\boldsymbol{f}_d) = Q(\boldsymbol{f})$ holds with probability $(1 - t/r)$, since the shift b can vary from 0 to r [6]. Hence, we have

$$\Pr(Q(\boldsymbol{f}_d) = Q(\boldsymbol{f}) | \Delta) = \int_0^r p(t|\Delta) \left(1 - \frac{t}{r}\right) dt. \tag{5}$$

Up to now, given the segment size r, the distortion tolerance of the ANN based quantizer can be obtained by computing (5), (3) and (2) consecutively. To make the quantizer map each 4-D feature vector to four 4-bit hash values, we set $r = 4$ and numerically computing (2) reveals $\overline{\Delta}_{ANN} = 1.12$.

We now consider the case when each dimension of the 4-D feature vector is independently quantized by a uniform SQ and estimate the corresponding distortion tolerance. Since $\Delta = \|\boldsymbol{d}\|_2$, the average distortion on each dimension of the feature vector is approximately $\Delta/2$. Consider the uniform SQ with step size W, two features with

[†]It should be clarified that the term *hashing* in LSH has a different meaning from the one in robust hashing. In LSH, it refers to the data structure for ANN indexing. In robust hashing, it stands for the algorithm that computes the robust signature of media data.

distance $\Delta/2 \in [0, W]$ can be quantized to the same index with probability $\left(1 - \frac{\Delta}{2W}\right)$. As before, let us denote the hash values, which are obtained by concatenating the four quantization indexes produced by the uniform SQ, of the original and distorted feature vectors as $\boldsymbol{I}$ and $\boldsymbol{I}_d$, respectively. Then the probability of $D(\boldsymbol{I}_d, \boldsymbol{I}) = 0$ is $\left(1 - \frac{\Delta}{2W}\right)^4$. Accordingly, the average distortion that the uniform SQ can tolerate is

$$\overline{\Delta}_{SQ} = \int_0^{2W} \Delta \left(1 - \frac{\Delta}{2W}\right)^4 d\Delta. \tag{6}$$

Since the features are normalized into $[-16, 16)$, if the uniform SQ generates four 4-bit hash values for each feature vector as the case in the ANN based quantizer, then $W = 2$ and $\overline{\Delta}_{SQ} = 0.53 < \overline{\Delta}_{ANN}$. It is evident from the comparison on distortion tolerance that the ANN based quantizer can provide a higher degree of robustness against distortions.

3.2 Analysis on Randomness

As stated in [7], the degree of success that an adversary correctly forge or estimate hash values without knowing the key for hash computation depends on the randomness of the output hash. In this section, the security of the proposed quantizer against hash forgery is investigated by assessing the randomness of the output hash using the entropy-based metric proposed in [7]. Given the feature $\boldsymbol{f}$, we estimate the entropy of the output hash $Q(\boldsymbol{f})$ that is computed by rounding the random variable $S = (\boldsymbol{a} \cdot \boldsymbol{f} + b)/r$. According to [8], it is straightforward that $H(Q(\boldsymbol{f})) \approx \aleph(S)$, where $H(\cdot)$ and $\aleph(\cdot)$ are the discrete and differential entropies, respectively. Hence, the randomness of the quantizer can be measured by computing $\aleph(S)$. However, the pdf of S has a complicated form, which makes it impossible to derive the closed-form expression of $\aleph(S)$. Alternatively, we derive its lower and upper bounds. Let us consider the case where the elements of $\boldsymbol{a}$ are drawn from $N(0, \sigma^2)$, we have $(\boldsymbol{a} \cdot \boldsymbol{f})/r \sim N\left(0, \sigma^2 \|\boldsymbol{f}\|_2^2 / r^2\right)$. Recall that conditioning reduces entropy, the lower bound of $\aleph(S)$ can be expressed as

$$\aleph(S) > \aleph\left(\frac{\boldsymbol{a} \cdot \boldsymbol{f} + b}{r} \,\middle|\, \frac{b}{r}\right) = \aleph\left(\frac{\boldsymbol{a} \cdot \boldsymbol{f}}{r}\right) = \frac{1}{2}\log_2\left(\frac{2\pi e \sigma^2 \|\boldsymbol{f}\|_2^2}{r^2}\right). \tag{7}$$

We proceed with estimating the upper bound. Among all the distributions with the same variance, the normal distribution gives the maximum entropy. Since $b/r \sim U(0, 1)$, we have $\mathrm{Var}(S) = \sigma^2 \|\boldsymbol{f}\|_2^2 / r^2 + 1/12$, and the following upper bound can be obtained.

$$\aleph(S) < \frac{1}{2}\log_2\left[2\pi e\left(\frac{\sigma^2 \|\boldsymbol{f}\|_2^2}{r^2} + \frac{1}{12}\right)\right]. \tag{8}$$

Here, we average the lower and upper bounds to estimate the entropy of the ANN based quantizer.

$$H_{ANN} \approx \frac{1}{4}\log_2\left(\frac{2\pi e \sigma^2 \|\boldsymbol{f}\|_2^2}{r^2}\right) + \frac{1}{4}\log_2\left[2\pi e\left(\frac{\sigma^2 \|\boldsymbol{f}\|_2^2}{r^2} + \frac{1}{12}\right)\right] \tag{9}$$

4. Experimental Results

We first demonstrate the overall performance of the proposed quantizer and compare it with that of the uniform SQ and DLVQ [4] using the receiver operating characteristic (ROC) curves obtained from content identification experiments. The testing database contains 2×10^3 images that are with the size of 512×512 and have 256 gray levels. Forty features were computed for each image using the Radon-transform-based [9] and the random-Gabor-filtering-based [4] feature extraction schemes. To compute hash values using the ANN based quantizer, features were first normalized into $[-16, 16)$ and then grouped into 10 4-D vectors. For each vector, 4 hash values were generated and each has 4 bits, so the number of buckets is 16 and the length of the final hash is 160 bits. For the ANN based quantizer, the elements of $\boldsymbol{a}$ were independently drawn from $N(0, 1)$ and $r = 4$. The parameter setting of DLVQ is the same as described in [4] and the SQ has 16 quantization levels. To assess the overall performance of the proposed work (i.e., its capability in balancing robustness and discrimination), content identification experiments were carried out and several kinds of content-preserving manipulations were implemented to produce distorted images, as listed in Table 1. For comparison purpose, the ROC curves corresponding to the ANN based quantizer, uniform SQ, as well as DLVQ are displayed in Fig. 2. The ROC curves imply that the ANN based quantizer outperforms both SQ and DLVQ in terms of overall performance, which indicates that it can best strike the balance between robustness and discrimination.

The second set of experiments were devoted to investigating the randomness of the ANN based quantizer. We started by assessing the accuracy of the estimated entropy. For a given feature vector, 10^5 hash values were generated under different keys, and then the entropy of the output hash

Table 1 List of content-preserving manipulations.

Manipulations	Range of Strength
JPEG compression	Quality factor: from 95 to 5
Median filtering	Width of square window: from 1 to 10
Blurring	Radius of circular window: from 1 to 40
Gaussian noise addition	Variance of noise: from 0.01 to 0.4
Contrast enhancement	Number of gray levels: from 224 to 8

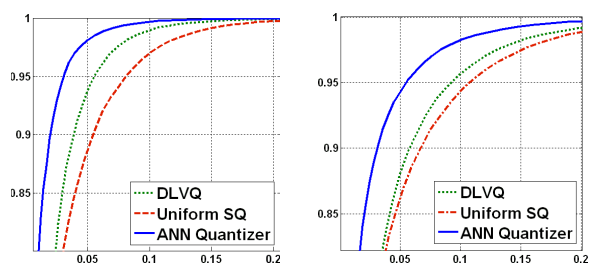

Fig. 2 ROC curves of quantizers. (a) Quantizing Radon-transform-based features; (b) Quantizing random-Gabor-filtering-based features.

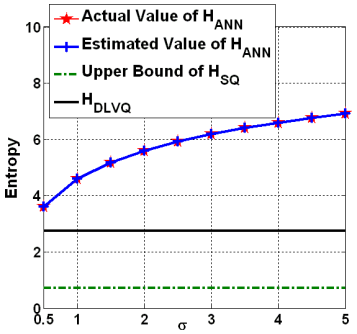

Fig. 3 Curves of entropy rates.

was computed. In Fig. 3, we plot the curves of the estimated and actual values of the entropy versus σ. It can be concluded that (9) can make an accurate estimation, since the two curves almost coincide. For comparison purpose, the entropy rates of the random SQ and DLVQ were also estimated and plotted in Fig. 3. As derived in [7], the entropy of the random SQ can be expressed as $H_{SQ} = r \log_2 e$, where $r \leq \frac{1}{2}$. Hence, the upper bound of H_{SQ} is $\frac{\log_2 e}{2} = 0.72$. In DLVQ, each feature vector is randomly mapped to a D_4 lattice point (i.e., 4-D integer vector with even component sum) by dithering and quantization. Since the components of D_4 lattice points are restricted to $[0, 8)$ in [4], it is easy to verify that there are $8^4/2 = 2048$ such lattice points. Each 4-D feature vector can be randomly mapped to any of them, so the entropy of DLVQ can be computed as $H_{DLVQ} = \frac{\log_2 2048}{4} = 2.75$. As shown in Fig. 3, the ANN based quantizer has the highest entropy among these random quantizers.

In addition, the sensitivity of output hash to the secret key for hash computation was also examined. Two hundred rounds of hash computation were conducted by quantizing the Radon-transform-based features extracted from a given testing image under distinct keys. The average distance between output hash values is 0.47, which indicates that the hash values produced by the proposed quantizer are quite sensitive to key variations. The studies in [10] show that this property can benefit the fragility of hashing algorithm against malicious tampering when image content is incorporated in key generation. For content-dependent key, even a slight content tampering can result in the change of the key and consequently lead to drastic change of the output hash. To verify this fact, tampering detection experiment was carried out following the key generation scheme presented in [10]. The Harris corner points, which are very sensitive to malicious tampering, were exploited to produce content-dependent key. We first defined 10 lines by linking the selected pairs of Harris corner points. For each line, a binary bit was generated according to the relationship between the numbers of Harris corner points lying on its two sides. Finally, a 10-bit content-dependent key can be obtained by concatenating these binary bits. The hash values of the original and tampered images shown in Fig. 4 were computed by quantizing the Radon-transform-based features using the

(a) (b)

Fig. 4 Original and tampered images. (b) is the tampered version of (a), where less than 1% of the pixels in (a) are modified. The sizes of the two images are 512×512.

ANN based quantizer under content-dependent key. The result of hash comparison shows that the NHD between the hash values of the original and tampered images is 0.49. Hence, the tampered image can be judged as inauthentic.

5. Conclusion

In this letter, we show that the features of perceptually identical media data can be modeled as ANNs. In light of this, we propose to quantize feature vectors using the random-projection based ANN indexing scheme. Analytical and experimental results reveal that the proposed work is better suited for robust hashing than other quantizers due to its higher tolerance against distortion, superior overall performance, and security against hash forgery.

References

[1] M.K. Mihcak and R. Venkatesan, "A perceptual audio hashing algorithm: A tool for robust audio identification and information hiding," Proc. Workshop Inf. Hiding, vol.2137, pp.51–65, April 2001.
[2] M. Johnson and K. Ramchandran, "Dither-based secure image hashing using distributed coding," Proc. IEEE Conf. Image Process., vol.3, pp.II-751–754, Sept. 2003.
[3] M. Tagliasacchi, G. Valenzise, and S. Tubaro, "Hash-based identification of sparse image tampering," IEEE Trans. Image Process., vol.18, no.11, pp.2491–2504, Nov. 2009.
[4] 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, April 2012.
[5] P. Indyk and R. Motwani, "Approximate nearest neighbors: towards removing the curse of dimensionality," Proc. ACM Symp. Theory of Computing, pp.604–613, May 1998.
[6] M. Datar, N. Immorlica, P. Indyk, and V.S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," Proc. ACM Symp. Computational Geometry, pp.253–262, June 2004.
[7] A. Swaminathan, Y. Mao, and M. Wu, "Robust and secure image hashing," IEEE Trans. Inf. Forensics Security, vol.1, no.2, pp.215–230, June 2006.
[8] T.M. Cover and J.A. Thomas, Elements of Information Theory, 2nd ed, Wiley, Hoboken, NJ, 2006.
[9] 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.
[10] W. Li and B. Preneel, "Attacking some perceptual image hash algorithms," Proc. IEEE Conf. Multimedia and Expo, pp.879–882, July 2007.