QUASI RATE DISTORTION OPTIMIZATION FOR BINARY HASHING

Yiding Liu, Wengang Zhou, and Houqiang Li

University of Science and Technology of China
Department of Electronic Engineering and Information Science
230026, Hefei, Anhui, China

ABSTRACT

Rate-distortion optimization has been a successful and significant method in video coding. By introducing Lagrange multiplier optimization into compress procedure, we can choose coding parameters simply and effectively. In nearest neighbor search problem, hashing has been a popular method to reduce computation and storage cost, which is consistent with video coding method. Conventionally, we evaluate a hashing method with mAP (mean average precision) w.r.t. different bit number, but leave bit cost as an independent measure index. In this paper, we make an attempt to combine retrieval accuracy and bit cost to make evaluation more comprehensive, using the concept of rate distortion optimization. Consequently, we obtain an evaluation criterion to judge which work point of a specific hashing method is better, taking both the accuracy and the bit cost into account. The exertion of an algorithm can be then determined.

Index Terms— Performance evaluation, Rate distortion theory, Information retrieval, nearest neighbor search

1. INTRODUCTION

With the rapid growth of multimedia communication and electronic equipment industry, we are crowded by mounts of digital images and videos, which are transmitted online. These huge amounts of data present significant challenges to signal processing, analysis and retrieval, including video coding and nearest neighbor search.

Video compression scheme is a complicated optimization procedure, which involves various motion representation possibility and coding of differences. Therefore, it is a crucial problem in video coding to control the encoder operationally. Blending various blocks and motions in video sequences and mode selection of different prediction modes, rate distortion efficiency varies.

The classical video coding standard MPEG-4 Part-10 AVC/H.264 [1] allows up to 7 block types and 16 reference frames in coding. Witnessing the fact that the various coding options show various coding efficiency at different bit rates with different video contents, we are encountered by a problem of selecting the best mode from various candidates. The mode with least distortion may be intuitively accepted, but it may not be globally optimal since the high rate cost may lead to a loss in the overall performance. In order to select the best operation globally, we determine the coding mode by rate-distortion optimization (RDO)[2, 3, 4].

Generally, the target of RDO is to minimize the distortion D subject to a given constraint rate R_c, which is formulated as follows:

$$\min\{D\} \text{ , subject to } R \leq R_c, \tag{1}$$

where R and D represent the rate and the distortion of each coding unit, respectively. The optimization task in Eq. (1) is popularly solved by Lagrangian optimization. The Lagrangian formulation of the minimization problem is given by

$$\min\{J\}, \text{ where } J = D + \lambda R, \tag{2}$$

where J is the Lagrangian rate-distortion function and λ is the so-called Lagrange multiplier. For each given value of Lagrangian multiplier λ, there is a solution to Eq. (2) corresponding to an optimal solution to Eq. (1) for a specific value of R_c.

Approximate nearest neighbors (ANN) search problem has been a popular and significant problem for research and are widely applied in retrieval systems [5, 6, 7, 8, 9], which are also involved with compression. With a query sample, ANN search tries to find the neighbors of the query in the whole dataset with an acceptable time cost. To solve this problem, tree-based techniques [10] and many hashing approaches [11, 12, 13, 14, 15, 16] have been proposed. Due to the effectiveness and high compressibility of encoding high-dimensional data points into binary codes, binary hashing is now widely used in various research tasks. By partitioning the continuous data space (usually Euclidean space) into discrete data regions, and assigning each data region a binary code, vectors in a region share a same binary representation and similar vectors are mapped to similar codes. In this way, we can find the distance of two vectors by checking their

This work was supported in part to Prof. Houqiang Li by 973 Program under contract No. 2015CB351803, NSFC under contract No. 61325009 and No. 61390514, and in part to Dr. Wengang Zhou by the Fundamental Research Funds for the Central Universities under contract No. WK3490000001 and WK3490000002.

978-1-5090-2175-8/17/$31.00 ©2017 IEEE

binary codes based on Hamming distance, which is efficient and simple.

2. PROBLEM FORMULATION

In traditional hashing method evaluation, we measure a hashing method in different bit conditions. Bit number in hashing method is always considered as an independent variable, which is typically set as 8, 16, 32, 64, etc. Performing a certain method, we input training data as well as the required bit number and then get hashing functions. For many hashing methods, there are random factors that affect the performance, such as random initializations and random projections. Due to such randomness, we may need to perform this procedure several times and obtain the final result of the method. After that we traverse all potential settings of bit number to evaluate the hashing method comprehensively.

We always compare the experiment results with a same bit number, which represents for the potency of the method in certain bit count. However, for the results with different bit numbers, it is still meaningful to determine which one is better, which indicates the exertion of the certain hashing method. It is common for the application scenarios of hashing methods to be storage-limited and computation-limited. In this circumstance, evaluation of each hashing model provide an important reference to model selection.

The exertion of a certain hashing method is a very abstract concept. For each attempt of a hashing method, the algorithm exerts differently, influenced by the randomness and the properties of the dataset and leading to different results. Under the condition of same bit number, we can tell the exertion by the search results. The better the results are, the better the exertion is. But in different bit number conditions, exertion still exists, but results are not valid to present it. We are trying to solve this problem.

In terms of probability, a random projection matrix can achieve the same performance to any hashing method, which takes projection matrix as hashing functions, but it is not stable and common. There is a performance limitation of a hashing method only related to the form of the hashing functions. For a certain dataset and projection matrix as hashing functions, there are optimized matrixes, which lead to the best searching result. Any hashing method in which the matrix is taken as the hashing function is possible to achieve this best result. However it is not proper to set this performance as the upper bound of the method, and the probability that this happens is too low to be considered. Assuming we always choose the best result as our final result in certain bit numbers, if we try sufficient amount of times, we can get an experimental upper bound of the effectiveness for a hashing method. Commonly, an attempt of this hashing method with certain bit number will get a result no better than the bound of the same bit number. This upper bound is "an upper bound in general", which can represent the best exertion of the hashing method in some way.

Hashing problem and video coding procedure have some concepts in common. We get an evaluation criteria like mean average precision (mAP) in hashing, corresponding to the distortion in video coding. Bit numbers in hashing and rate in video coding both represent the compression degree. In information theory, the rate and distortion has a theoretic lower bound only related to the information source, which rate-distortion optimization reflects in some way. And now we get an upper bound, if we set $D = 1-\text{mAP}$, we can also get a lower bound, which can perform a similar rate-distortion optimization so as to make it possible to evaluate the exertion.

The problem can be formulated as: Given a certain hashing method $\mathcal{M}$ with a set of bit numbers $\mathcal{B} = \{b_1, b_2, \ldots, b_k\}$ and dataset $\mathcal{D}$, we are trying to evaluate the exertion of $\mathcal{M}$, in other words, given working points $p_i(d_i, r_i)$, we can tell which working point is better, where $d_i \in [0, 1]$ represents the distortion of p_i and $r_i \in \mathcal{B}$ represents for the bit number of p_i.

Similar to video coding method, we are trying to consider the distortion and the rate. In rate-distortion optimization, we formulate a Lagrangian function $J = D + \lambda R$ to find a proper work point. Correspondingly, we formulate our objective function as:

$$\min\{J\}, \text{ where } J = D + \lambda R, \tag{3}$$

where $D = 1 - \text{mAP}$ represents for the distortion, $R \in \mathcal{B}$ represents for bit numbers.

First we get the lower bound of the performance of $\mathcal{M}$. We perform n times of $\mathcal{M}$ on $\mathcal{D}$, and get a series of working points $\mathcal{W}$ with different bit numbers. For each bit number, we choose the best work point which has the minimum d, a lower bound of $\mathcal{M}$: $\mathcal{L} = \{p(r, d)|p, p_j \in \mathcal{W}, \forall p_j, d < d_j \text{ if } r = r_j\}$, then we have an assumption:

Assumption 1 *Working points $p \in \mathcal{L}$ are all optimal.*

That is to say, after enough times of test, the best working points we get in each bit number can represent the best exertion of the method. Given this assumption, we have two ways of evaluation.

Direct evaluation (DE): Due to the assumption, all the working points in $\mathcal{L}$ are optimal, then we set:

$$\text{for } p \in \mathcal{L}, J(p) = d + \lambda r = const, \tag{4}$$

so we get:

$$\lambda = \frac{const - d}{r}. \tag{5}$$

Due to $r \in \mathcal{B}$, we acutally get a set of $\lambda_i = \lambda(b_i)$, for $p_j \in \mathcal{W}$ if $r_j = b_i$,then $J(p_j) = d_j + \lambda_i r_j$, then we can evaluate all the working points.

We can easily prove that, for any $p_j \in \mathcal{W}$,we can find a $p \in \mathcal{L}$ which $r = r_j, d < d_j$, so $J(s_j) = d_j + \lambda r_j > d + \lambda r$, then the loss for all the working points are higher than the points on the lower bound, which satisfies the assumption.

R-d optimization based evaluation (RDOE): in rate-distortion optimization, Eq. (2) is taken as Lagrangian cost function, which needs to be solved. If the rate-distortion curve is convex and both R and D are derivable, we can slove this Lagrangian optimization prolem. To minimize J, we take the derivative of R and set it to zero:

$$\frac{\mathrm{d}J}{\mathrm{d}R} = \frac{\mathrm{d}D}{\mathrm{d}R} + \lambda = 0 \tag{6}$$

$$\lambda = -\frac{\mathrm{d}D}{\mathrm{d}R} \tag{7}$$

Due to the lack of analytic function of rate and distortion, Lagrange multiplier method cannot directly figure out what λ will be. There are many kinds of ways to determine how to choose the value of λ. Here we select a simple one: according to Eq. (7), if we know the relationship between rate and distortion, we can easily find out what λ should be. The more accurate the rate-distortion model is, the better λ can be. Thus, many different algorithms were proposed to establish a robust and effective R-D model [17, 18, 19].

Back to hashing method, compared to video coding, we already have a lower bound so that we can directly find out a decent R-D model by fitting the bound curve. We choose to fit the curve by polynomials based on our observation. For different datasets, the model needs to be changed according to the effectiveness of the hashing method. For example, if we select quadratic model as our R-D model:

$$D = aR^2 + bR + c \tag{8}$$

$$\lambda = -2aR - b \tag{9}$$

In Rate-Distortion optimization, we have a limitation of R in the cost function, which is pre-determined before the coding procedure. In our rate-distortion optimization based evaluation, we need to designate an ideal working rate, after that, according to Eq. (9), we can get λ for evaluation.

3. EVALUATION

In this section, we evaluate several hashing methods in nearest neighbor search problem and try to compare which one of the working points is the best, finding out an evaluation on the exertion of the hashing methods.

3.1. Dataset

We evaluate the method on two datasets: (1) **ANN_SIFT10K** [20]: this dataset consists of 10,000 dataset vectors, 100 query vectors and 25,000 training vectors for learning phase. (2) **ANN_SIFT1M** [20]: this dataset consists of 1,000,000 dataset vectors, 10,000 query vectors and 100,000 training vectors. These two datasets all consist of 128-D SIFT [21] feature vectors, but the scale is different, which correspond to a small dataset and a larger one to simulate different cases.

3.2. Hashing methods

We choose three traditional hashing methods for evaluation. **LSH**: Local Sensitive Hashing [11]. In this method, we project the data onto several different random hyperplanes for hashing. **ITQ**: Iterative Quantization [12]: It learns a good projection matrix from training data instead of a random one. **SpH**: Spherical Hashing [13], in which hyperplane is replaced by hypersphere.

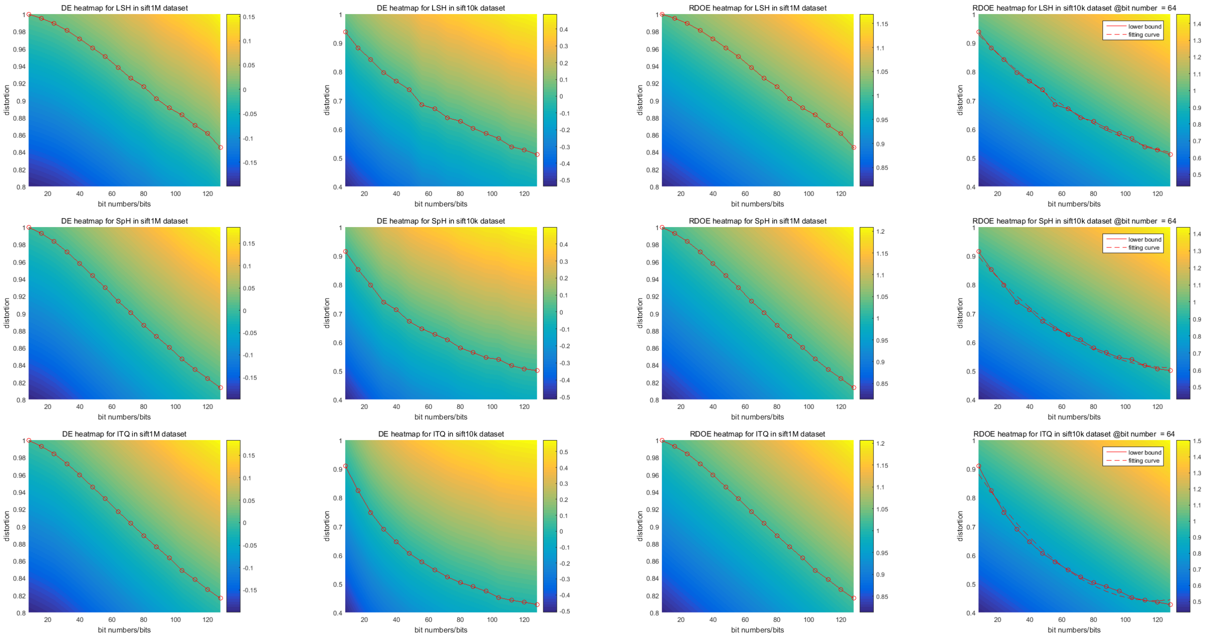
3.3. Exertion evaluation

For hashing method, mean average precision is most widely accepted as an evaluation criteria. We match 1-mAP to the distortion D in video coding and bit number to rate R.

Fig. 1 shows the experiment results. Values of heat map represents for the value of cost function. The higher the cost is, the worse the exertion is. We repeat each experiment 50 times to make the lower bound more convincing. Bit numbers are from 8 to 128, with step of 8 bits. We picked up the boundary in the experiment results and perform our evaluation method. For DE, as referred in Eq. (5), each bit number is corresponded to a λ. We set the $const$ as 0, which means the λ of each bit should be the minus slope of the boundary point with the zero point. As we can see in the figure, the contour line of the cost function is similar to the lower bound, which means nonuniformity in the working space. Low cost working points are more likely to concentrate near the bound. For RDOE, we need to first find a proper model to fit the curve. By observation, we choose linear model for experiments on SIFT1M dataset and quadratic model on SIFT10K dataset. For linear model, the derivative of D to R is a fixed number, which leads to a fixed λ, so no matter what the bit number is, we can use a same λ, we can then skip over the pre-installation of the working rate, every working rate shares a same λ. But as for quadratic model, it is not so simple. Setting a working bit number makes great influence on the results. Fig. 2 shows how the cost of the working space will change when varying the working rate. Testing on RDOE in the experiment applies a same λ in an attempt, where the contour lines are all parallel. It means the low cost working points are more likely to concentrate on the tangent line of the lower bound through the ideal working point corresponding to the given working rate, rather than the lower bound like DE.

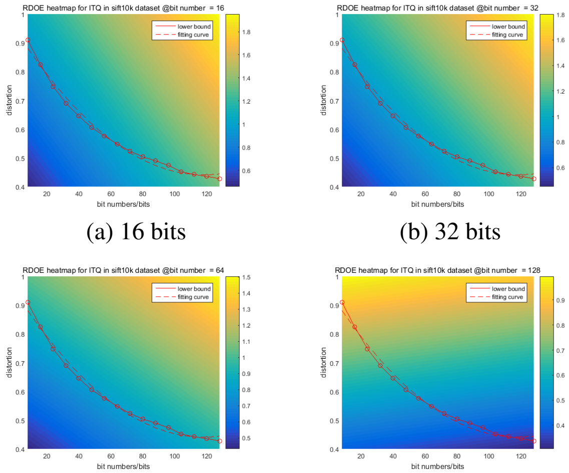
4. CONCLUSION

In this paper, we investigate the quasi rate-distortion optimization problem and design a new evaluation method on discovering how a hashing method works and if a hashing method works well on a certain experiment, which is quite similar to the rate-distortion optimization problem in video coding. Referring to the loss function of the problem, we introduce two different methods on this problem, direct evaluation (DE) and rate-distortion optimization based evaluation



(a) (b) (c) (d)

Fig. 1. Experiment results of the evaluation criteria. Values of heatmap represents for the value of cost function. The higher the cost is, the worse the exertion is. Only the regions above the red curve can be reached. Each row represents for a hashing method: LSH, SpH, ITQ, from top to down. Each column represents for different datasets and evaluation method: (a):DE, SIFT1M, (b):DE, SIFT10K, (c):RDOE, SIFT1M (d):RDOE, SIFT10K @bit number = 64



(a) 16 bits

(b) 32 bits

(c) 64 bits

(d) 128 bits

Fig. 2. Experiment results on different preinstall interested bit numbers. λ is only valid in the neighborhood.

(RDOE). DE treats the Lagrangian formulation as a cost function, fixing the working points on the lower bound on constant cost. We find reasonable values of λ and formulate a cost function for the task. RDOE performs the rate-distortion optimization on the function just like it is in video coding method. By constructing a proper model for bit number the rate and 1-mAP the distortion, we can solve the optimization problem by setting a work rate. Discussing the exertion of a certain algorithm is a novel problem, which worth a further studying.

5. REFERENCES

[1] Thomas Wiegand, Gary J Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the h. 264/avc video coding standard," *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 7, pp. 560–576, 2003.

[2] Bin Li, Houqiang Li, Li Li, and Jinlei Zhang, "λ domain rate control algorithm for high efficiency video coding," *IEEE transactions on Image Processing*, vol. 23, no. 9, pp. 3841–3854, 2014.

[3] Li Li, Bin Li, Houqiang Li, and Chang Wen Chen, "Domain optimal bit allocation algorithm for high efficiency video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, 2016.

[4] Li Li, Bin Li, Dong Liu, and Houqiang Li, "λ-domain rate control algorithm for hevc scalable extension," *IEEE Transactions on Multimedia*, vol. 18, no. 10, pp. 2023–2039, 2016.

[5] Wengang Zhou, Yijuan Lu, Houqiang Li, Yibing Song, and Qi Tian, "Spatial coding for large scale partial-duplicate web image search," in *Proceedings of the 18th ACM international conference on Multimedia*. ACM, 2010, pp. 511–520.

[6] Shaoyan Sun, Wengang Zhou, Qi Tian, and Houqiang Li, "Scalable object retrieval with compact image representation from generic object regions," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 12, no. 2, pp. 29, 2016.

[7] Zechao Li, Jing Liu, Jinhui Tang, and Hanqing Lu, "Robust structured subspace learning for data representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 10, pp. 2085–2098, 2015.

[8] Zhen Liu, Houqiang Li, Wengang Zhou, Richang Hong, and Qi Tian, "Uniting keypoints: Local visual information fusion for large-scale image search," *IEEE Transactions on Multimedia*, vol. 17, no. 4, pp. 538–548, 2015.

[9] Wengang Zhou, Houqiang Li, Jian Sun, and Qi Tian, "Collaborative index embedding for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[10] Jerome H Friedman, Jon Louis Bentley, and Raphael Ari Finkel, "An algorithm for finding best matches in logarithmic expected time," *ACM Transactions on Mathematical Software*, vol. 3, no. 3, pp. 209–226, 1977.

[11] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *Proceedings of the twentieth annual symposium on Computational geometry*. ACM, 2004, pp. 253–262.

[12] Yunchao Gong, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin, "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2916–2929, 2013.

[13] Jae-Pil Heo, Youngwoon Lee, Junfeng He, Shih-Fu Chang, and Sung-Eui Yoon, "Spherical hashing," in *Computer Vision and Pattern Recognition, 2012 IEEE Conference on*. IEEE, 2012, pp. 2957–2964.

[14] Jinhui Tang, Zechao Li, Meng Wang, and Ruizhen Zhao, "Neighborhood discriminant hashing for large-scale image retrieval," *IEEE Transactions on Image Processing*, vol. 24, no. 9, pp. 2827–2840, 2015.

[15] Min Wang, Wengang Zhou, Qi Tian, Zhengjun Zha, and Houqiang Li, "Linear distance preserving pseudo-supervised and unsupervised hashing," in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 1257–1266.

[16] Min Wang, Wengang Zhou, Qi Tian, and Houqiang Li, "Sparse matrix based hashing for approximate nearest neighbor search," in *Pacific Rim Conference on Multimedia*. Springer, 2016, pp. 559–568.

[17] Gary J Sullivan and Thomas Wiegand, "Rate-distortion optimization for video compression," *IEEE signal processing magazine*, vol. 15, no. 6, pp. 74–90, 1998.

[18] Thomas Wiegand and Bernd Girod, "Lagrange multiplier selection in hybrid video coder control," in *Image Processing, 2001. Proceedings. 2001 International Conference on*. IEEE, 2001, vol. 3, pp. 542–545.

[19] Lulin Chen and Ilie Garbacea, "Adaptive λ estimation in lagrangian rate-distortion optimization for video coding," in *Electronic Imaging 2006*. International Society for Optics and Photonics, 2006, pp. 60772B–60772B.

[20] Herve Jegou, Matthijs Douze, and Cordelia Schmid, "Product quantization for nearest neighbor search," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 1, pp. 117–128, 2011.

[21] David G Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.