An Image Compression Algorithm Based on the Karhunen Loève Transform

Jae Woong Soh*, Hyun-Seung Lee*†, and Nam Ik Cho*
* Department of Electrical and Computer Engineering, INMC, Seoul National University, Seoul, Korea
† Samsung Electronics Co., Ltd., Suwon, Korea
E-mail: soh90815@ispl.snu.ac.kr

Abstract—**The Karhunen-Loève Transform (KLT) is the optimal transform for a block of signal in terms of decorrelation and energy compaction performances. However, it is not used in practice because we need to transmit the transform kernel for every block, which requires huge amount of side information. Hence fixed-kernel discrete cosine transform (DCT) is used instead in many of current image compression algorithms. In this paper, considering a recent trend of requiring high resolution, high quality, multiple view images, we propose an image adaptive transform (IAT) approach where the transform kernels are derived from the KLT for a group of image blocks. The contribution of this paper is to develop a simple method to group the similar image blocks according to DCT coefficient statistics, and also to reduce the number of transform kernels by using the relationship between the DCT coefficients and the corresponding block properties such as edge directions. Since this method needs some amount of side information for transmitting IAT kernels, it does not perform well for the low resolution images at low bit rates, but it outperforms JPEG for the high resolution, high quality and multiple view images that can share the same transforms.**

I. Introduction

The KLT is statistically optimal for compressing a given block of signal, but it cannot be used in practice because the transform kernel for each block should also be transmitted to the receiver [1], [2]. Hence we normally apply the same transform kernel to every block of signal regardless of their statistical properties in practical compression methods. Specifically, the DCT is widely used for block-wise image and video compression algorithms, because it is known to perform similarly as KLT for the highly correlated signals and also it has fast algorithms like FFT [3], [4], [5], [6]. Also, the DCT can be adapted to the block properties by warping the frequency of input signal [8]. Nevertheless, the KLT was tried for the compression of hyperspectral or medical images, which can be considered the pile of similar images [2], [7], [9], [10]. However, they did not compare the performance with the JPEG or they showed that compression performance is at best similar to JPEG due to huge amount of side information.

In this paper, we again attempt to apply KLT to block-wise image compression, targeting the high resolution, high quality and multiple view images. The problem when we wish to apply the optimal transform for each block is the overwhelming amount of side information. We try to evade this problem by classifying the blocks into several groups, and then derive the KLT for each group of blocks. Hence we need to transmit only a few number of KLTs, which greatly reduces side information. In the case of low resolution and low bit rate compression, this method does not outperform the conventional JPEG. However, in the case of high resolution images at high bit rates, the amount of side information becomes relatively small compared to the main information. Moreover, capturing multiview images is also becoming very important these days, especially for increasing the value of virtual reality (VR) and augmented reality (AR) systems. In the case of multiview images for the same scene, we may find KLT only for the center view image and use the same transforms for the rest of views, which also greatly reduces the relative amount of side information.

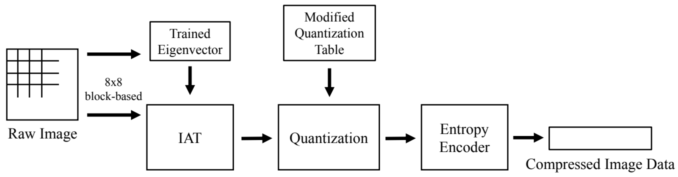


Fig. 1: Block diagram of proposed encoder.

In summary, we propose a new transform coding method for the still image compression, where the contributions are as follows:

- A simple method for the classification of image blocks is proposed, based on the analysis of DCT coefficients.
- The KLT for each group of blocks is derived, which compacts the signal energy better than the DCT on average.
- We demonstrate that the proposed method outperforms JPEG especially for high resolution, high quality and multiple view images.
- While the existing KLT based compression methods do not show better performance than the JPEG, the proposed method yields large gain over the JPEG.

II. Proposed image compression algorithm

The overview of proposed image compression procedure is shown in Fig. 1, which is basically the same as JPEG except that the IAT is used instead of DCT and the quantization table is modified. The process of deriving IAT is first to categorize the image blocks into 7 classes and then find the KLT for each class.

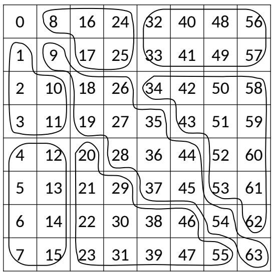


Fig. 2: Grouping the coefficient positions according to edge and frequency similarity.

JPEG

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Ours

10	16	24	40	57	69	92	103
11	16	24	49	58	72	92	103
12	16	24	51	60	77	95	104
12	17	26	51	61	78	98	109
13	18	29	55	62	80	99	112
14	19	35	55	64	81	100	113
14	22	37	56	64	87	101	120
14	22	40	56	68	87	103	121

Fig. 3: Quantization tables for JPEG and the proposed IAT.

A. Classification of image blocks

The basic idea of proposed method is to find similar blocks in the image and derive the IAT, which is actually the KLT of the average of similar blocks. The blocks can be classified according to the similarity of pixel values in the spatial domain. However, for the robustness to noise and brightness variation, we classify the blocks according to the similarity of DCT coefficients. Specifically, we transform the 8×8 image block by DCT and find the maximum absolute coefficient. Formally, for the m-th 8×8 image block $\mathbf{X}_m$, let us denote its DCTed block as $\mathbf{F}_m$ which is also an 8×8 matrix, the elements of which are denoted as $F_m(i,j)$, $i, j = 0, 1, \cdots, 7$. If we vectorize $\mathbf{F}_m$ into a 64×1 vector $\mathbf{f}_m$, its p-th element is given by

$$f_m(p) = F_m(i + 8j). \tag{1}$$

Then the blocks that have the maximum coefficients at the same index p (except for the DC at $p = 0$) are considered the ones that have similar properties. If we categorize the blocks into 63 groups according to their maximum DCT coefficient index, we need to prepare 63 KLTs in this case, which needs unduly large amount of side information. Hence we try to reduce the number of groups by merging the coefficient indices as shown in Fig. 2. The merging strategy is based on the relationship between the DCT coefficients and edge direction in a block, which has been studied for the vector quantization or block artifact reduction [11], [12], [13]. According to these studies, the large coefficients in the upper area in the 8×8 block are related with vertical edges, left area with horizontal edges, and the coefficients in the diagonal area are related with oblique edges. Hence grouping like Fig. 2 means that we classify the blocks according to their edge orientation and frequency contents (high or low frequencies) into 7 groups, with the group index l from 1 to 7. For example, for the DCT of the m-th image block ($\mathbf{F}_m = \text{DCT}(\mathbf{X}_m)$), if its vectorized version $\mathbf{f}_m$ has the peak at $p = 2$, and also for the DCT of k-th block ($\mathbf{F}_k = \text{DCT}(\mathbf{X}_k)$), if its vectorized version $\mathbf{f}_k$ has the peak at $p = 3$, then $\mathbf{X}_m$ and $\mathbf{X}_k$ are categorized into the same group, i.e., the blocks with indices m and k are classified into the same group.

B. Derivation of IAT and the compression algorithm

Let $\mathbf{x}_n$ be a 64×1 vector which is the vectorized version of $\mathbf{X}_n$. Then for all the vectors with indices n that are categorized into a group indexed l ($l = 1, 2, \cdots 7$), we compute their covariance matrix as

$$\mathbf{C}_l = \frac{1}{L} \sum_{n \in \text{group } l} \mathbf{x}_n \mathbf{x}_n^T \tag{2}$$

where L is the number of blocks in the l-th group. Then we find the eigenvectors of $\mathbf{C}_l$, which form the 64×64 KLT denoted as $\mathbf{K}_l$ that is considered optimal for the blocks in the group l on average. However, an image block that is classified into a group may be better compacted by the KLT that is optimal for the adjacent group, especially when the p is on the border of category shown in Fig. 2. Hence for each of image blocks, we try all the KLTs from $\mathbf{K}_1$ to $\mathbf{K}_7$, and also the DCT, and find the best one for the block. Then the block is given the index l when it is best compressed by $\mathbf{K}_l$ and the index 0 is given when the DCT performs the best for the block. This index (from 0 to 7) is transmitted to the receiver along with the seven 64×64 KLT coefficients. To be precise with the amount of side information, we need 3 bits for each block for the group index. In addition, we round the KLT coefficients to two byte integers, so that the amount of side information for the transform coefficients are $7 \times 64 \times 64 \times 2$ bytes.

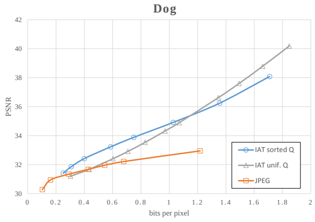
C. Quantization Table

Quantization table for the JPEG is designed considering human visual system (HVS), which is very sensitive to low frequency signals and less sensitive to high frequency [14]. We use two versions of the quantization table for IAT: sorted quantization table and uniform quantization table. In designing the quantization table for our IAT, we expect that the KLT will compact more energy into lower frequencies than the DCT, so that the order of coefficient magnitudes (from low to high frequency) would be closer to "monotonically decreasing pattern" than the DCT coefficients. Hence, while some of the elements in the JPEG quantization table is not in the monotonically increasing order, we set them to be monotonic as shown in Fig. 3. Also, to preserve high frequency components for very high bit rate applications, we use another quantization table which is uniform over the all frequencies.

III. Experimental Results

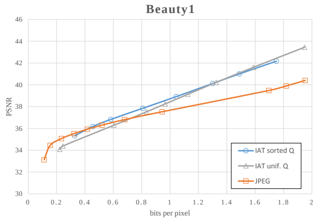
As stated previously, our goal is to develop an IAT based compression algorithm that works better than the DCT based JPEG for the high resolution images at high bit rates. For this,

(a)

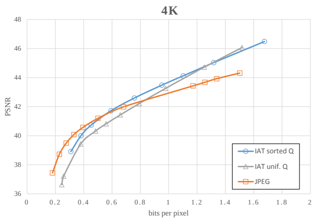


(b)

Fig. 4: (a) “Dog” image (b) RD-curves for the Dog.



(a)



(b)

Fig. 5: (a) RD-curves for ’Beauty1.bmp’ (b) averaged RD-curves for 4K images.

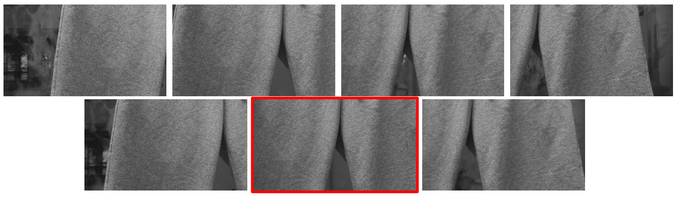
we prepare a very high resolution image and also some 4K images that are extracted from frequently used videos for the test of HEVC [15]. In addition, we also demonstrate that our compression method is especially effective when compressing a set of similar images such as multiview images.

A. Results for high resolution images

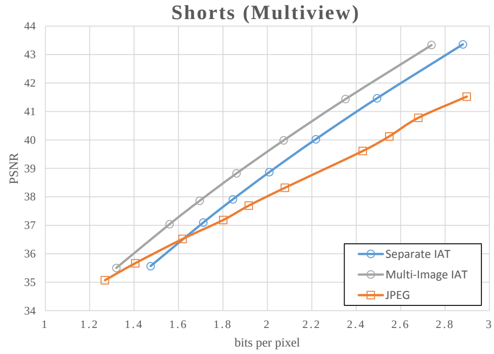
First, we apply the algorithm for a very high resolution image (6000×4000) shown in Fig. 4a. The RD curves compared with the JPEG are shown in Fig. 4b which shows that our IAT performs much better than the DCT. IAT with sorted quantization table tends to perform better in low bit rate regions, whereas IAT with uniform quantization table performs better in high bit rate regions. Second, we apply our algorithm for 21 images with 4K (3840×2160) resolution. We extract 3 frames from each of 7 videos in [15] and plot the RD curves in Fig. 5, where Fig. 5a shows the RD curves for a specific image, and Fig. 5b shows the averaged results of 21 images. As expected, the IAT also outperforms DCT at high bit rates.

B. Results for multiview images

For capturing a scene including the 3D information, we usually take multiple images of the same scene from different view points, or use light field and/or depth capturing devices. In the case of multiple view images, they share similar blocks and thus will be more benefited from the IAT than the single image cases. For the experiment, we take 7 images for a scene at 7 different view points with resolution of Full HD (1920×1080). Then the KLTs are derived from the center view image, and we let the images to share the same KLTs. Hence the required side information becomes relatively small (proportional to the number of views) as compared to the



(a)



(b)

Fig. 6: (a) Multiview images (b) RD-curves for the proposed method when we share the KLT (denoted as Multi-image IAT in the figure) and when we apply the algorithm to each of the view point images separately (Separate IAT).

single image case. Fig. 6a shows multiple images that we used for the test. The image with red box is the center view image. Fig 6b shows the RD curve for the proposed method, which is compared with the JPEG and also the proposed algorithm that is applied to each of the view point images separately. As expected, the RD curve is improved when we share the KLT such that the side information is relatively reduced.

IV. Conclusions

We have proposed a still image compression algorithm, based on the adaptive transform derived from the KLT. We first classify the image blocks into several groups according to the similarity of DCT coefficients, and find KLT for each group which is considered the optimal transform for the group of blocks on average. The group index for each block and the KLT coefficients are sent to the receiver as side information, which is quite large for low resolution and low bit rate cases. However, for the high resolution images, the side information becomes relatively small. From the experiments, the proposed method is shown to be effective for the compression of high resolution images at high bit rates and also for the compression of multiple view images. To the best of our knowledge, existing KLT based methods do not perform better than JPEG due to heavy side information, whereas our method yields large gain over the JPEG.

We have shown the potential of proposed block transform coding algorithm by comparing it with the JPEG. There are many things to do as future works for enhancing its performance and for extending it to video compression. First, we are going to develop an arithmetic coder that is suitable for the compression of KLT results, while we are currently using a simple run length coder like the JPEG. Second, we need to develop a more sophisticated block classification method. We group the blocks just by the maximum DCT coefficients in this paper, but we are going to develop k-means or mean-shift clustering based methods for grouping the similar blocks. Third, we are going to compare the (multiview) image compression performance with the HEVC (MVC) which uses inter-view correlations and arithmetic coding [16], when the KLT coefficients are further compressed by the arithmetic coding. Finally, we are going to apply the proposed method for the compression of residual images from the intra and/or inter prediction of frames, which may extend the proposed algorithm to video compression.

REFERENCES

[1] Karhunen, Kari. "ber lineare Methoden in der Wahrscheinlichkeitsrechnung." Vol. 37. *Universitat Helsinki,* 1947.

[2] RD Dony, "Karhunen-Lo'eve transform," *The transform and data compression handbook,* pp. 20-55, 2001.

[3] N Ahmed, T Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Transactions on Communications,* pp. 90-93, 1974.

[4] Nam Ik Cho and Sang Uk Lee, "Fast algorithm and implementation of 2-d discrete cosine transform," *IEEE Transactions on Circuits and Systems,* vol. 38, no. 3, pp. 297-305, 1991.

[5] Nam Ik Cho and Sang Uk Lee, "A fast algorithm for 2-d dct," *IEEE International Conf. Acoust., Speech, Signal Processing (ICASSP),* pp. 2197-2200, 1991.

[6] Nam Ik Cho, Il Dong Yoon, and Sang Uk Lee, "On the regular structure for the 2-d dct algorithm," *IEEE Transactions on Circuits and Systems II,* vol. 40, no. 4, pp. 259-266, 1993.

[7] Jingning Han, Ankur Saxena, and Kenneth Rose, "Towards jointly optimal spatial prediction and adaptive transform in video/image coding," *IEEE International Conf. Acoust., Speech, Signal Processing (ICASSP),* 2010.

[8] Nam Ik Cho, Sanjit K. Mitra, "Warped discrete cosine transform and its application in image compression," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 10, no. 8, pp. 1364-1373, 2000.

[9] Barbara Penna, Tammam Tillo, Enrico Magil, and Olmo Gabriella, "Transform coding techniques for lossy hyperspectral data compression," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 45, no. 5, pp. 1408-1421, 2007.

[10] Michelle Effros and Philip A. Chou, "Weighted universal transform coding: Universal image compression with the karhunen-loeve transform," *IEEE International Conf. Image Processing (ICIP),* 1995.

[11] Dong Sik Kim and Sang Uk Lee, "Image vector quantizer based on a classification in the dct domain," *IEEE Transactions on Communications,* vol. 39, no. 4, pp. 549-556, 1991.

[12] Y Itoh, "An edge-oriented progressive image coding," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 6, no. 2, pp. 135-142, 1996.

[13] Bong Gyun Roh, Nam Ik Cho, and Sang Uk Lee, "Blocking artifacts reduction using constrained least squares method with directional information," *Picture Coding Symposium (PCS),* 1999.

[14] Gregory K Wallace, "The JPEG still picture compression standard," *IEEE Transactions on Consumer Electronics,* vol. 38, no. 1, pp. xviii-xxxiv, 1992.

[15] "Ultra Video Group- Test Sequences," http://ultravideo.cs.tut.fi/#testsequences, [Online].

[16] Gerhard Tech, Ying Chen, Müller Karsten, Jens-Rainer Rom, Anthony Vetro, and Ye-Kui Wang, "Overview of the multiview and 3d extensions of high efficiency video coding," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 26, no. 1, pp. 35-49, 2015.