MANIPULATING IMAGE PATCHES FOR COMPRESSION

Dong Liu [1*], *Xiaoyan Sun* [2], *and Feng Wu* [2]

[1] MOE-Microsoft Key Laboratory of Multimedia Computing and Communication,
University of Science and Technology of China, Hefei 230027, China
[2] Microsoft Research Asia, Beijing 100190, China

E-mail: liud@mail.ustc.edu.cn, xysun@microsoft.com, fengwu@microsoft.com

ABSTRACT

We consider how to exploit the correlation in image for compression by virtue of studying image patches in a non-parametric manner. Instead of extracting and recording parameters, our approach directly operates on image patches. The basic assumption is that a subset of image patches can be well inferred from the others; therefore, they can be removed at encoder only to be restored at decoder. Meanwhile, assistant information is transmitted for the restoration, which actually encodes the similarity between removed and preserved patches. The entire scheme is built upon an optimization framework, which is decoupled and solved accordingly.

Index Terms— Image compression, inpainting, non-parametric, texture synthesis.

1. INTRODUCTION

Ubiquitous applications of image and video communications have been challenging the compression systems. To eliminate the bottleneck of signal-processing-based compression, many works have been done to exploit the visual redundancy in images, known as the second-generation coding [1]. Such coding methods target at developing concise descriptions of images without loss of important visual features. However, it remains a difficult issue which features should be involved in a specific description so as to facilitate a specific application. For example, we can use many mature tools to derive a set of parameters from an image, including histograms, edges, wavelet coefficients, and so on. But it is hard to assert that they have been enough to capture image features.

Instead of extracting parameters from an image, another way arises that samples image patches and tries to directly utilize these patches. The underlying story is that a number of image patches contain sufficient statistics, which are often of high-order and not easily inferred. To take advantage of image patches in a *non-parametric* manner has achieved great success in super-resolution [2] as well as in texture synthesis [3].

From the compression point of view, one tough problem is how to represent a huge set of image patches, which may be of different shapes, sizes, and orientations. Even if we restrict the patches to be regular ones, e.g. squares with the fixed size, the number of patches is still great since that patches can be overlapped with each other. At the same time, the numerous patches sampled from the same image exhibit strong correlations, which can be exploited accordingly.

Epitome is a good example that compiles the patches sampled from an image and generates a miniature. An epitome, with remarkably smaller size than the original image, still contains the important visual information of the image it represents [4, 5]. Recent work, known as inverse texture synthesis [6], also combines plenty of image patches into a so-called compaction, and has improved on epitome when the image contains rich textures.

In this paper we consider another approach to manipulating image patches for compression. Different from epitome or compaction which generates a smaller image, our approach will provide an *incomplete* image with the same size to the original, only partial regions removed. Intuitively, both approaches utilizes the correlations between image patches and provides condensed description. Moreover, two differences are noticeable. Firstly, the incomplete image still keeps the spatial locations of the preserved patches, while the epitome or compaction has lost such information. Secondly, the reconstruction in the epitome or compaction case requires a full-resolution map, which indicates where the patch can be found within the smaller image. In our approach, the reconstruction is only required for the removed regions by virtue of the preserved ones, which is virtually the inpainting problem.

This paper is related to our former work that integrates inpainting into compression, and assistant information is proposed to enhance the inpainting capability [7]. In this paper we also adopt the inpainting with assistant information as the reconstruction method. And we have considered the entire scheme based on the framework of image patches. Instead of extraction and utilization of edges in [7], we directly operate on image patches in this paper.

We will provide the mathematical formulation of our ap-

[*] This work has been done during D. Liu's internship at Microsoft Research Asia.

978-1-4244-2571-6/08/$25.00 ©2008 IEEE

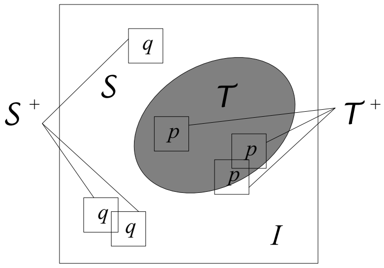

Fig. 1. Illustration of symbols.

proach in Section 2. Section 3 is devoted to the solution to the region removal problem. Section 4 will discuss the inpainting method for the reconstruction. Section 5 presents some experimental results and concludes this paper.

2. MATHEMATICAL FORMULATION

As shown in Figure 1, assume an observed image f is defined on I, where $I \subset \mathbb{Z}^2$ is a set of 2-D coordinates. Our approach essentially condenses image patches by dividing I into the removed regions $\mathcal{T}$, $\mathcal{T} \subset I$, and the preserved regions $\mathcal{S} = I - \mathcal{T}$. $\mathcal{S}$ is considerably smaller than I for the purpose of compression. Note that we perform our approach by studying image patches that sampled from I. To make this clear, we use the symbol f_p to denote an image patch sampled from f at the location p, where p is always 8×8 square in this paper. We further define two sets of patch locations, $\mathcal{T}^+ = \{p | p \subset I \text{ and } p \cap \mathcal{T} \neq \varnothing\}$ and $\mathcal{S}^+ = \{q | q \subset \mathcal{S}\}$. Figure 1 shows some patches belonging to $\mathcal{T}^+$ and $\mathcal{S}^+$, note that patches are overlapped with each other.

In our approach, decoder reconstructs the image in two steps. First, the preserved image regions are decoded and presented as an incomplete image $\bar{f}$. Second, the removed image regions are restored by inpainting as g,

$$\{g_p | p \in \mathcal{T}^+\} = \mathcal{F}_\theta(\{\bar{f}_q | q \in \mathcal{S}^+\}). \tag{1}$$

where $\mathcal{F}$ is an implicit inpainting functional, provided the assistant information θ. We emphasize that such θ does not exist in traditional inpainting problems. However in our approach, we adopt inpainting as the restoration module in the compression scenario. Therefore, kinds of distinctive information can be extracted from the removed regions at encoder and be utilized by inpainting at decoder. The introduction of θ into (1) thus makes inpainting a quite different problem.

In terms of compression, we consider how to achieve the best coding performance by minimizing the joint rate-distortion cost,

$$J = D_{\mathcal{S}}(f, \bar{f}) + D_{\mathcal{T}}(f, g) + \lambda(R_{\mathcal{S}} + R_{\bar{f}} + R_\theta). \tag{2}$$

The degrees of freedom in our approach include: region removal, coding of preserved regions, inpainting as well as assistant information. If all these factors are jointly considered, the optimization problem will be quite complicated. In this paper, we decouple the problem as follows.

Firstly, consider the region removal, i.e. to split I into $\mathcal{T}$ and $\mathcal{S}$. Intuitively, we translate the rate-distortion optimization as the following two questions: Which part of an image can be well restored from the other parts, while which part cannot? Now put aside the inpainting or compression for simplicity, we can solve the two questions by studying patches sampled from the original image. We formulate an energy function and minimize it, which will be detailed in Section 3.

Secondly, assume we have gotten the split of I, i.e. $\mathcal{T}$ and $\mathcal{S}$, consider the inpainting problem. At this step, we have been inspired by two works, texture optimization [3] and intra motion compensation [8], both are combined in our approach. To be specific, we first constructs an energy function of image patches, similar to that in [3], but obviously different due to the assistant information θ. Then, we restrict θ to directly encode the similarity between removed and preserved patches. Such θ can be acquired by encoder with an intra motion estimation module [8]. The details are given in Section 4.

At last, we remark that the decoupling of the joint cost (2) is not optimal but rather for practice. One can find from the following two sections that there are only direct manipulations of image patches in the solution, which is performed in the non-parametric manner.

3. REGION REMOVAL TO COMPRESSION

As mentioned before, in region removal we put aside the inpainting or compression, only consider original image patches. Here we define an hidden image h to indicate the inferred regions. Our objective is to find the optimal removal $\mathcal{T}^*$, in order to minimize the inference error as well as to introduce possible compression, i.e.,

$$\mathcal{T}^* = \arg\{\min E_{rem}, \text{ subject to } |\mathcal{T}| \geqslant T_0\}, \tag{3}$$

$$E_{rem} = \sum_{p \in \mathcal{T}^+} ||f_p - h_p||^2, \tag{4}$$

i.e. the removal energy E_{rem} is sum of patch distances where the distance is in the Euclidean sense. $|\mathcal{T}|$ is the cardinality of $\mathcal{T}$ and T_0 is a specified area of the removed regions.

Intuitively, h is related to inpainting (and compression as well), but since we want to decouple the problem, we estimate h by minimizing a heuristic energy function,

$$\begin{aligned} E_{inf} = & \frac{1}{|\mathcal{T}^+|} \sum_{p \in \mathcal{T}^+, p^* \in \mathcal{S}^+} ||h_p - f_{p^*}||^2 \\ & + \frac{\alpha}{|\mathcal{S}^+|} \sum_{q \in \mathcal{S}^+, q^* \in \mathcal{T}^+} ||f_q - h_{q^*}||^2, \end{aligned} \tag{5}$$

where h_p is an inferred patch while f_{p^*} is its match, where p^* belongs to $\mathcal{S}^+$ and the distance from f_{p^*} to h_p is minimal. f_q

is a preserved patch while h_{q^*} is its match with the minimal distance, $q^* \in \mathcal{T}^+$. α is a predefined positive weight.

The energy E_{inf} (5) consists in two parts, termed *forward* energy and *backward* energy, respectively. To minimize the forward energy means that each removed patch can find a good match from the preserved ones, thus it can be well restored. To minimize the backward energy means that the preserved patches can also find good matches from the restored ones. In other words, the preserved patches can *only* constitute the removed regions but not able to provide other information. Due to the minimization of the *backward* energy, the region removal process tries to remove as much redundancy as possible. Therefore, the backward energy is related to the rate in terms of compression.

The solution to (3) is not trivial in practice since the relationship between E_{rem} and $\mathcal{T}$ is not explicit. Here we consider a block-wise region removal, i.e. original image is divided into non-overlapped blocks and some are removed. For the purpose of sampling patches, the block size should be larger enough than the patch size. Each block B is related to an estimated removal energy calculated by (4), where $\mathcal{T}$ is replaced by B. The blocks with less energies will be removed, until the specified area T_0 is achieved.

To calculate the removal energy for each block also requires to estimate h by minimizing the energy E_{inf} (5). Here $\mathcal{T}$ is replaced by B while $\mathcal{S}$ is replaced by $I - B$. In practice, $I - B$ is further substituted by $\mathcal{N}_B$, which means the neighborhood of B, in order to reduce the computations. Note that p^* and q^* in (5) are implicit variables, so we adopt an EM (Expectation Maximization) like algorithm to minimize (5). In the E (Expectation) step, p^* and q^* are fixed and h is adjusted to minimize the energy. According to the derivatives,

$$\begin{aligned} h(x,y) =& \Big(\frac{1}{|B^+|}\sum_{p\in B^+}\delta_p(x,y) + \frac{\alpha}{|\mathcal{N}_B^+|}\sum_{q^*\in\mathcal{N}_B^+}\delta_{q^*}(x,y)\Big)^{-1} \\ & \Big(\frac{1}{|B^+|}\sum_{(x,y)\in p,(x_1,y_1)\in p^*} f(x_1,y_1) \\ & + \frac{\alpha}{|\mathcal{N}_B^+|}\sum_{(x,y)\in q^*,(x_2,y_2)\in q} f(x_2,y_2)\Big), \end{aligned} \tag{6}$$

where $\delta_p(x,y)$ is an indicator function that evaluates 1 if $(x,y) \in p$ and 0 otherwise, $\delta_{q^*}(x,y)$ has similar definition. (x_1,y_1) is the counterpart of (x,y) according to the relationship between p^* and p, while (x_2,y_2) is also corresponding to (x,y), but according to the relationship between q and q^*. In the M (Maximization) step, h is fixed and p^* and q^* are adjusted to minimize the energy, which is virtually to find the most similar patch for each h_p and f_q. E and M steps are iterated in turn until convergence.

4. INPAINTING TO RECONSTRUCTION

Now the removed regions $\mathcal{T}$ are determined. The incomplete image is fed into an image codec and the preserved regions are reconstructed as $\bar{f}$. We consider the inpainting problem formulated by (1). The optimal restoration in the rate-distortion sense is accomplished by minimizing the joint rate-distortion cost,

$$E_{res} = \sum_{p\in\mathcal{T}^+} ||f_p - g_p||^2 + \beta R_\theta, \tag{7}$$

where β is a positive weight. Furthermore, inspired by the texture optimization method [3], we explicitly express the inpainting (1) as to minimize the following energy function,

$$E_{inp} = \sum_{p\in\mathcal{T}^+, p^*\in\mathcal{S}^+} ||g_p - \bar{f}_{p^*}(\theta)||^2, \tag{8}$$

where the patch $\bar{f}_{p^*}$ is the match to g_p, i.e. p^* belongs to $\mathcal{S}^+$ and the "distance" from $\bar{f}_{p^*}$ to g_p is minimal. The "distance" here is not only in the Euclidean sense but also constrained by the assistant information θ, which is the essential difference between our formulation (8) and the one in [3].

Note that in [6] the energy function also contains a control map that acts similarly as θ, but the control map is often derived from specific applications in graphics (i.e. known *a priori*). In our formulation, we define θ within the image patches framework. Let θ directly link the patches p^* and p, i.e. θ encodes geometrical transformations including translation, rotation and scaling, or even deformations. In case the patches are regular ones, e.g. squares with the fixed size, θ actually reduces to displacement vectors for only translations.

To make our idea clear, rewrite the inpainting energy function as follows,

$$E_{inp} = \sum_{p\in\mathcal{T}_1^+} ||g_p - \bar{f}_{p^*}||^2 + \sum_{p\in\mathcal{T}_2^+} ||g_p - \bar{f}_{p^\theta}||^2, \tag{9}$$

where $\mathcal{T}^+$ is divided into $\mathcal{T}_1^+$ and $\mathcal{T}_2^+$. $\bar{f}_{p^*}$ does not explicitly depend on θ, i.e. p^* are searched out according to the criterion of minimal Euclidean distance. $\bar{f}_{p^\theta}$ is determined only by θ, or specifically, for each $p \in \mathcal{T}_2^+$ the p^θ is decided by θ.

The joint consideration of (7) and (9) leads to a practical algorithm for inpainting with assistant information. At the encoder side, by an intra motion estimation module, we search for each $p \in \mathcal{T}^+$ the best match $p^\#$ according to f_p and $\bar{f}_{p^\#}$. All such displacement vectors are stored in θ_0. Then encoder needs to select a subset of θ_0 to transmit. The selection can be a greedy one, which drives the joint cost (7) to decrease in the steepest direction. Start from the initial case $\theta = \varnothing$, calculate the energy (7). At each step, choose one displacement vector in θ_0 and move it into θ, check the energy (7) again. If the energy decreases, this displacement vector is accepted and continue to add; otherwise, the process halts. At the decoder side, according to the selected θ, (9) is minimized to restore the removed regions.

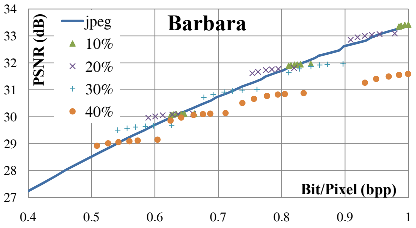

Fig. 2. Coding performance in PSNR versus bit-rate.

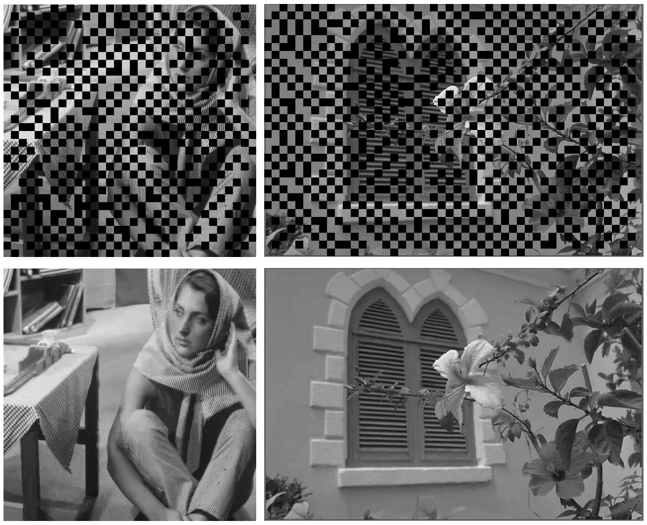

Fig. 3. Incomplete images, black for removed regions (top) and reconstructed images (bottom).

When calculating the energy (7), the restored image g is acquired by minimizing (9), which contains implicit variable p^*. Our solution to (9) is again the EM like iterative method. In the E step, p^* are fixed and g is adjusted according to,

$$g(x,y) = (\sum_{p\in\mathcal{T}^+} \delta_p(x,y))^{-1}(\sum_{(x,y)\in p} \bar{f}(x_1,y_1)), \tag{10}$$

where (x_1, y_1) is the counterpart of (x, y) according to the relationship between p^* and p if $p \in \mathcal{T}_1^+$, or between p^θ and p if $p \in \mathcal{T}_2^+$. In the M step, g is fixed and p^* is adjusted, which means to find the most similar patch for each g_p. E and M steps are iterated in turn until convergence.

5. RESULTS AND CONCLUSION

We construct an image compression system to verify the proposed approach. Input image is divided into 16 × 16 blocks, some of which are removed according to an input ratio. The block removal algorithm has been discussed in Section 3 and the weight α in (5) is set to 0.1 empirically. A binary mask encodes whether each block is removed or not. The incomplete image is coded by JPEG while the removed regions are skipped during JPEG coding. Moreover, the removed regions are restored by inpainting with assistance of a set of displacement vectors, as detailed in Section 4 and the weight β in (7) is an input parameter. Since JPEG is adopted to code the incomplete image, the results reported herein are not intended to be state-of-the-art. But our proposed approach is much more general, any existing compression methods can be readily integrated into our system.

Figure 2 shows the PSNR versus bit-rate performance of our system when testing on the Barbara image and the removal ratios are from 10% to 40%. The data points correspond to different JPEG quality parameters and different β's. Compared to baseline JPEG, our system can achieve PSNR improvement as high as 0.5dB. We remark that PSNR is not quite suitable to evaluate our approach since that the inpainting can present visually pleasing results but with large pixel-wise errors. How to assess the quality remains an open issue. Figure 3 shows the incomplete images for Barbara and Kodim07[1] at 40% removal, as well as the final reconstructed images at 0.62bpp and 0.48bpp, respectively. The restoration quality is satisfactory for most textural regions.

The compression approach proposed in this paper directly utilizes the correlations within image patches. We have discussed the region removal and the inpainting with assistant information within the framework of image patches. By solving optimization formulations, we show how the entire approach translates the rate-distortion theory into practice. Experimental results demonstrate the potentials of our approach.

6. REFERENCES

[1] M. M. Reid, R. J. Millar, and N. D. Black, "Second-generation image coding: An overview," *ACM Comput. Surveys*, vol. 29, no. 1, pp. 3–29, Mar. 1997.

[2] W. T. Freeman and E. C. Pasztor, "Learning low-level vision," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV'99)*, pp. 1182–1189.

[3] V. Kwatra, I. Essa, A. Bobick, and N. Kwatra, "Texture optimization for example-based synthesis," *ACM Trans. Graph. (SIGGRAPH 2005)*, vol. 24, no. 3, pp. 795–802, Jul. 2005.

[4] N. Jojic, B. J. Frey, and A. Kannan, "Epitomic analysis of appearance and shape," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV'03)*, pp. 34–41.

[5] V. Cheung, B. J. Frey, and N. Jojic, "Video epitomes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR'05)*, vol. 1, pp. 42–49.

[6] L.-Y. Wei, J. Han, K. Zhou, B. Guo, and H.-Y. Shum, "Inverse texture synthesis," Microsoft Research, Tech. Rep. MSR-TR-2007-35, 2007.

[7] D. Liu, X. Sun, F. Wu, S. Li, and Y.-Q. Zhang, "Image compression with edge-based inpainting," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 10, pp. 1273–1287, Oct. 2007.

[8] S.-L. Yu and C. Chrysafis, "New intra prediction using intra-macroblock motion compensation," JVT-C151, 3rd meeting of Joint Video Team (JVT), May 2002.

[1] In the Kodak image library: http://r0k.us/graphics/kodak/.