CLOUD-BASED DISTRIBUTED IMAGE CODING

Xiaodan Song[*1], *Xiulian Peng*[2], *Jizheng Xu*[2] *and Feng Wu*[2]

[1]Xidian University, Xi'an, China
[2]Microsoft Research Asia, Beijing, China
Email: {v-xias, xipe, jzxu, fengwu}@microsoft.com

ABSTRACT

This paper proposes a cloud-based distributed image coding scheme (Cloud-DIC) to exploit the strong correlations with external partial-duplicate images in the cloud. It features both high coding efficiency and low encoder complexity, which makes it suitable for photo sharing on mobile devices. To get the side information in the cloud, a thumbnail of the current image is transmitted to retrieve highly correlated images and reconstruct through geometrical registration and adaptive patched-based stitching. The current image is then compressed by a transform-domain syndrome coding, bitplane by bitplane. Once a bitplane is received, the decoded high-quality image is further used to refine the side information in the cloud, which will benefit the coding of following bitplanes and the reconstruction. Experimental results on a landmark image database show that it can largely enhance the coding efficiency both subjectively and objectively with up to 5 dB gains and 58% bits saving over JPEG.

Index Terms— cloud-based coding, distributed image coding, local feature descriptors

1. INTRODUCTION

With the explosion of multimedia on the web, when you would like to share one image with your friends, you can easily find some similar images, even of the same scene for landmark images, through sophisticated image retrieval techniques. This brings great opportunities to largely enhance the image coding efficiency by exploiting the strong correlations with external images. However, conventional image coding schemes, e.g. JPEG [1] and JPEG 2000 [2], based on transforms and predictions cannot exploit such correlations. To upload a high-resolution image, it will consume a lot of network bandwidths to get a high-quality image. It is desired to develop an efficient coding scheme that can make full use of external images.

Recent research on image processing using a large-scale image database has emerged in various applications, such as composition, reconstruction, super resolution and compression [3]-[7]. They tend to use local feature descriptors to retrieve and align correlated images and show some promising results. In compression, Yue et al. proposed to transmit local SIFT (Scale Invariant Feature Transform) descriptors instead of pixel values to better exploit correlations with images in the cloud [6]. With SIFT matching between the current image and correlated ones in the cloud, a geometrical registration is performed at the patch level to correct the geometrical distortions between them, which come from different camera viewpoints, focal lengths and scales. A down-sampled image is also transmitted to verify external patches and guide the patch stitching.

However, most of these schemes using local descriptors are inverse problems. One cannot guarantee that patches found from external images match the one from the current image without transmitting it. For example, when new objects come out or when occlusion occurs, false or no patches may be stitched to reconstruct the image, leading to a locally false or smoothed reconstruction. What's more, if there are locally varying illumination differences between the current image and the correlated ones retrieved from the cloud, even if a patch corresponding to exactly the same object at the same viewpoints and scale after geometrical registration can be found, the reconstructed image might be visually pleasing, but different from the original image. That is, the fidelity of the reconstructed image cannot be guaranteed.

In this paper, we propose a cloud-based distributed image coding scheme (Cloud-DIC) to solve this problem. Since distortions come only from quantization in distributed source coding (DSC) if a correct decoding is achieved, a high-fidelity reconstruction can be ensured [8]. By shifting computations from the encoder to the decoder, the proposed scheme is suitable for photo sharing on mobile devices. In Cloud-DIC, a thumbnail is transmitted to get the side information (SI) in the cloud using correlated images. The encoder then transmits syndrome bits bitplane by bitplane to correct the difference between the current image and the side information in the cloud, where the compression ratio is decided by the correlation between them. With a better reconstruction after decoding one bitplane, a successive refinement is performed in the cloud to get a better side information and a correspondingly better reconstruction.

The rest of this paper is organized as follows. Section 2 gives an overview of the proposed scheme. Section 3 explains it in detail. Experiments results are presented in Section 4. Section 5 concludes this paper.

*This work was done when X. Song was with Microsoft Research Asia as an intern.

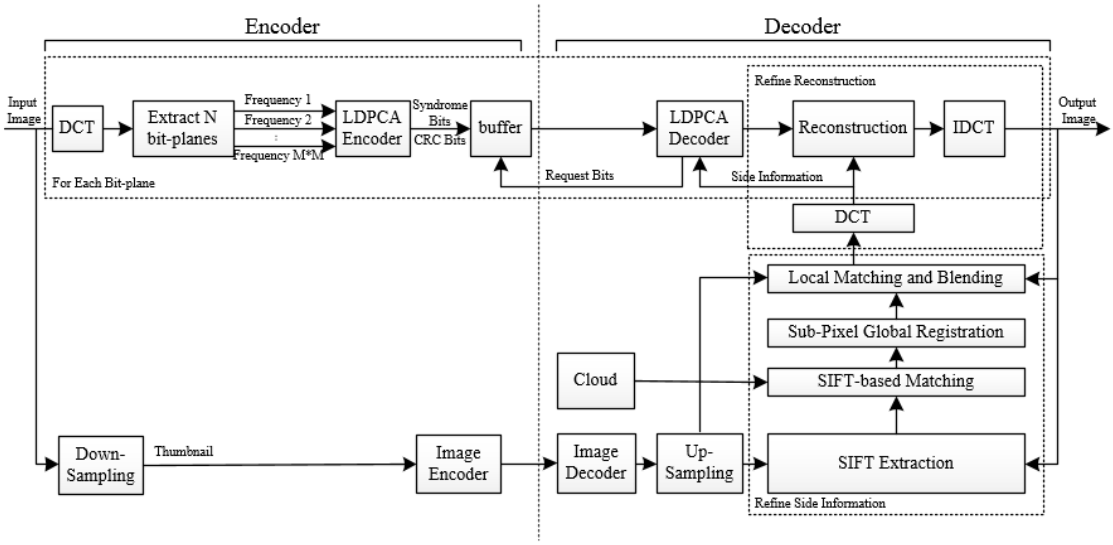


Fig. 1 Framework of the proposed scheme.

2. OVERVIEW OF THE PROPOSED SCHEME

The flowchart of the proposed scheme is depicted in Fig. 1. It mainly consists of three parts: side information generation using the compressed thumbnail, distributed encoding and decoding, and successive refinement between the two parts.

First, to get the side information from external images in the cloud, a thumbnail of the input image is compressed by conventional image coding. After decoding, it is used to reconstruct the current image similar to the super resolution approach in [7]. The processes include local SIFT descriptors extraction from the up-sampled thumbnail, SIFT-based matching, geometrical registration to compensate for geometric differences such as camera viewpoints and focal length, and the local matching and blending to locally align and stitch patches. After getting the reconstruction, a discrete cosine transform (DCT) is performed on the reconstruction to get the side information for distributed decoding.

In distributed coding, the syndrome-based coding using Low-Density Parity-Check Accumulate (LDPCA) codes is adopted in Cloud-DIC [9][10]. The input image is decorrelated by DCT and split into bitplanes. Each bitplane is sent to the LDPCA coder from the most significant to the least significant bitplane for syndrome encoding. For each bitplane, the encoder transmits a subset of the syndrome bits to the decoder on request. Except the syndrome bits, Cyclic redundancy check (CRC) bits of each bitplane are also sent to the decoder for validation of a correct decoding. After syndrome decoding of all received bitplanes, a reconstruction in frequency domain is obtained by an expectation-based reconstruction. After inverse DCT, the image is finally reconstructed.

At the syndrome decoding stage, once a bitplane is received and decoded, a reconstruction can be obtained which is better than the thumbnail. For better alignment between the current image and the correlated ones, the side information is further refined using the new reconstruction for registration and patch matching instead of the thumbnail. The refined side information can in turn lead to better reconstruction and is used to decode the next bitplane. The process continues until all received bitplanes are decoded.

3. THE PROPOSED SCHEME

3.1. Thumbnail Compression

The thumbnail is generated by downsampling from the original image and compressed by the conventional JPEG image coding. Larger thumbnails lead to better side information quality but more bits to transmit. Here we use bicubic downsampling with a ratio of 1:64. To ensure a good quality, the thumbnail is compressed by JPEG at quality 90.

3.2. Side Information Generation

At the decoder, the thumbnail is decoded to get the SI in the cloud for the DSC module. The SI generation process with the decoded thumbnail is similar to the super resolution (SR) approach in [7]. First, local SIFT descriptors are extracted from the up-sampled thumbnail, which are used to retrieve correlated images in the cloud. Since different images may be captured at different camera viewpoints and focal lengths, a 3D geometrical registration is performed between the retrieved images and the current one using matched SIFT pairs. After that, an adaptive patch-based matching is performed on thumbnails to locally adjust the 2D position for further alignment. Unlike [7] where the alignment is performed at the integer pixel precision, a ¼ pixel precision is used in the Cloud-DIC for better alignment. Finally, a blending process is performed

to stitch the high-frequency details of patches from the aligned images onto the up-sampled thumbnail and get the reconstruction for the current image. After DCT, it is used as the side information for syndrome decoding.

3.3. Transform-Domain Distributed Coding

To exploit the correlations between the current image and the SI, the transform-domain syndrome-based coding similar to that in [10] is adopted in the Cloud-DIC. One difference is that successive syndrome-based coding is employed in our scheme. At the encoder, the input image is first decorrelated by a 2D DCT and split into bitplanes. The DCT size is set to 8x8 in our implementation. Each bitplane is then sent to the LDPCA encoder from the most significant to the least significant bits to generate syndrome bits to be stored in the buffer. In addition, CRC bits of the current bitplane are stored in the buffer to ensure the correctness of decoding. The buffer transmits a subset of these syndrome bits with CRC to the decoder upon request.

At the decoder, syndrome decoding is first performed to decode each bitplane. We assume a Laplacian model for the correlation noise of each frequency between the original image and the SI. To estimate the model parameter, multiple images are offline reconstructed using a large database and statistics are estimated to get a fixed model for all test images. After decoding one bitplane, a reconstruction can be obtained by calculating the expectation given the quantization interval $\boldsymbol{q_i}$ and the SI, i.e.

$$\hat{\boldsymbol{x}}_i = E(\boldsymbol{x}_i|\boldsymbol{y}_i, \boldsymbol{q}_i), \quad (1)$$

where $\boldsymbol{x_i}$ and $\boldsymbol{y_i}$ are the i-th frequency of the input image and SI, respectively. After inverse DCT, the image is finally reconstructed.

3.4. Refinement between SI and DSC Reconstruction

One of the biggest challenges in reconstructing an image using correlated images is the alignment between them. Although the SIFT-based approach can be invariant to scale and rotations, its registration accuracy is questionable. The thumbnail somewhat improves the registration accuracy by removing some false patches and providing a local alignment, but the lack of high-frequency cannot make an accurate alignment. In Cloud-DIC, a better alignment can be achieved by using the received bitplanes. Once a bitplane is decoded, a new reconstruction superior to the thumbnail is obtained, which can be used to refine the SI.

Let $\hat{\boldsymbol{x}}$ denote the reconstruction in spatial domain after decoding one bitplane, whose quality is better than the up-sampled thumbnail $\boldsymbol{x_l}$. $\hat{\boldsymbol{x}}$ is used to refine the SI by another reconstruction process in the cloud using external images similar to that in Sec. 3.2. With more details than the thumbnail, more SIFT descriptors can be extracted from $\hat{\boldsymbol{x}}$, leading to a more stable geometrical registration. Unlike Sec. 3.2 where the matching is performed on thumbnails only, the local patch-based matching is performed in a residue domain with $\hat{\boldsymbol{x}}$ to compensate for local illuminance differences. Let $\boldsymbol{r_l}$ denote the up-sampled thumbnail of one registered correlated image $\boldsymbol{r}$ with the same down-sampling ratio as $\boldsymbol{x_l}$. The cost of two patches P and Q on the current image $\boldsymbol{x}$ and $\boldsymbol{r}$ is given by

$$D(P,Q) = \left\| \left(P_{\hat{x}} - P_{x_l}\right) - \left(Q_r - Q_{r_l}\right) \right\|_2, \quad (2)$$

where $P_{\hat{x}}$ and P_{x_l} are corresponding patches on $\hat{\boldsymbol{x}}$ and $\boldsymbol{x_l}$ respectively. Q_r and Q_{r_l} are corresponding patches on $\boldsymbol{r}$ and $\boldsymbol{r_l}$ respectively. Given the matched patch Q^* with a minimum cost by Eq. (2), the reconstructed high-resolution patch for P is

$$P_x^* = P_{x_l} + Q_r^* - Q_{r_l}^*, \quad (3)$$

where Q_r^* and $Q_{r_l}^*$ are corresponding matched patches on $\boldsymbol{r}$ and $\boldsymbol{r_l}$ respectively.

On the other hand, with a better SI $\boldsymbol{y'}$, the reconstruction can in turn be refined by

$$\hat{\boldsymbol{x}}_i{}' = E(\boldsymbol{x}_i|\boldsymbol{y}_i{}', \boldsymbol{q}_i), \quad (4)$$

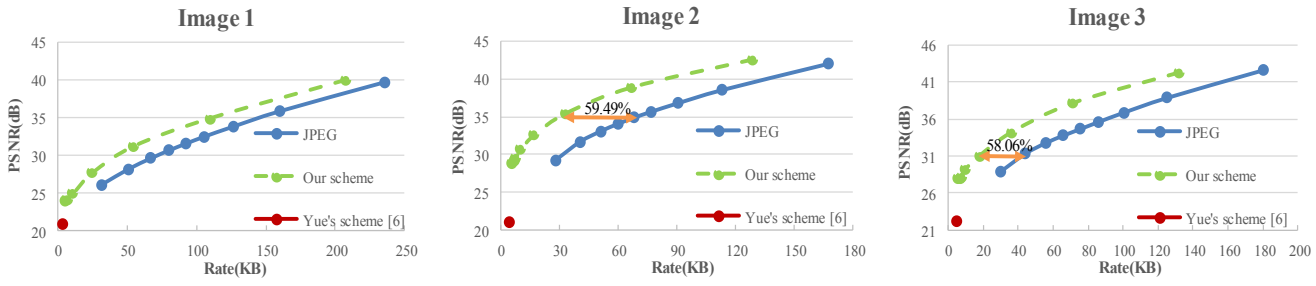
where $\hat{\boldsymbol{x}}_i{}'$ and $\boldsymbol{y}_i{}'$ are the i-th frequency of the reconstruction and $\boldsymbol{y'}$, respectively. We can see that there is an iterative refinement process between the reconstruction in syndrome decoding and the side information, when a bitplane is decoded. Our experiment shows that one back-and-forth iteration is enough. The refined side information $\boldsymbol{y'}$ will also help the coding of following bitplanes.

4. EXPERIMENTAL RESULTS

Image 1 Image 2 Image 3

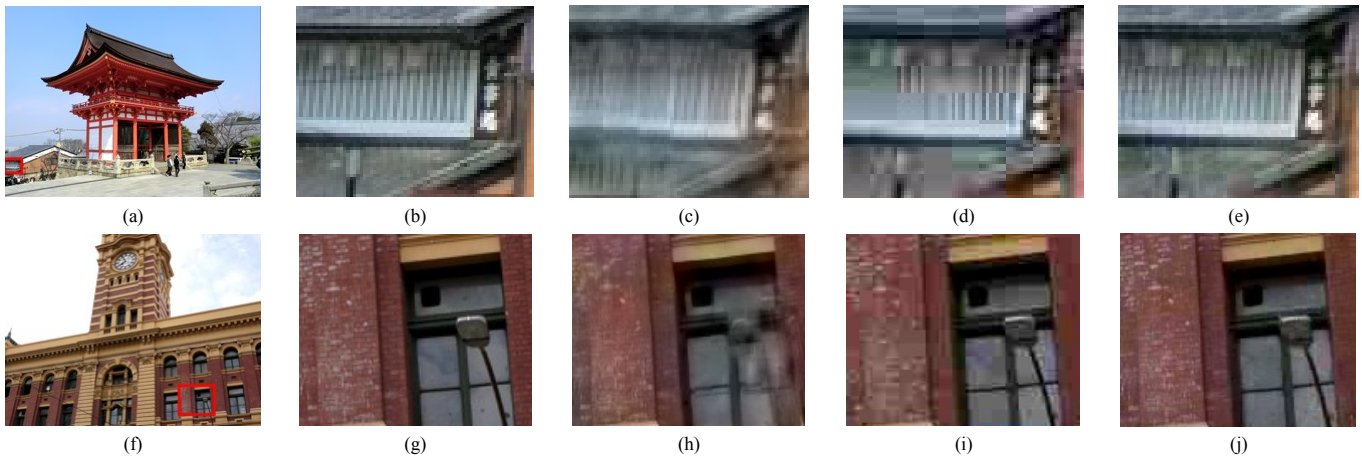
Fig.2 Test images.

To evaluate the performance of the proposed scheme, we build a database by crawling landmark images from Flickr, whose width or height is larger than 1024. There are in total 535,520 images in the database. Three images listed in Fig. 2 are used for evaluation in this experiment, which are retrieved using the keywords, summer palace in China, Kiyomizu Temple in Japan and Flinder street station in Melbourne, respectively. The test images are excluded from the dataset. At most top four correlated images are used for each test image in the reconstruction, as shown in [11]. We can see that there are differences in camera viewpoints, focal length and illuminations between the test images and the correlated ones.



(a) (b) (c)

Fig.3 RD performance comparison.



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

(f) (g) (h) (i) (j)

Fig.4 Visual quality comparison. (a) Image 2. (b) Cropped regions of image 2. (c) SR from compressed thumbnails [7]. (Rate: 4.99KB, PSNR: 28.32dB). (d) JPEG. (Rate: 33.12KB, PSNR: 30.45dB). (e) Proposed scheme. (Rate: 33.16KB, PSNR: 35.53dB). (f) Image 3. (g) Cropped regions of image 3. (h) SR from compressed thumbnails [7]. (Rate: 5.35KB, PSNR: 27.79dB). (i) JPEG. (Rate: 36.04KB, PSNR: 30.11dB). (j) Proposed scheme. (Rate: 36.05KB, PSNR: 34.19dB).

The RD performance of the Cloud-DIC is shown in Fig. 3. Two schemes are used for comparison, which are JPEG and one pioneering cloud-based image coding, Yue's scheme in [6] by vision-based approaches. In Yue's scheme, the thumbnail downsampling ratio is 1:64 and it is compressed by HEVC intra coding [12]. Only PSNR of the luminance component is shown here. We can see that by exploiting correlations with external images, the Cloud-DIC largely enhances the coding efficiency with 2-5dB gains over JPEG and up to 58% bits saving. Compared with Yue's scheme with only one RD point, Cloud-DIC provides different qualities at different rates. Besides, even at the lowest rate of Cloud-DIC by transmitting only a thumbnail, it shows a much better reconstruction fidelity over Yue's scheme with up to 7dB gains in PSNR for better registration and patch stitching.

The visual comparisons are shown in Fig. 4. The third column shows the result by cloud-based SR from the transmitted thumbnail [7]. One can observe that some parts are incorrectly reconstructed because SR is an inverse problem that cannot ensure the reconstruction fidelity. After transmitting some syndrome bits, those parts are corrected as shown in the fifth column. Compared with JPEG that shows severe blocking artifacts and loses details by quantization, the reconstruction by the proposed scheme looks much better at the same rate.

As for the complexity, since computations are shifted from the encoder to the decoder by DSC, the encoder complexity of the proposed Cloud-DIC is low, which makes it suitable for mobile devices with limited resources. With the powerful computing resources in the cloud, the high decoder complexity will not be a problem.

5. CONCLUSION

This paper proposes a cloud-based distributed image coding scheme to exploit correlations with external images. It can largely reduce the image upload bandwidth with a low encoder complexity if highly correlated images can be found in the cloud, which makes it suitable for mobile devices with limited resources. Considering that JPEG is the most widely used image coding scheme today, our scheme is based on JPEG currently. With the emergence of the latest HEVC standard [12], the thumbnail compression in Cloud-DIC can also benefit from its high coding efficiency. Future work will include better source correlation exploitation using DWT and spatial adaptivity exploitation in DSC.

6. REFERENCES

[1] G.K. Wallace, "The JPEG still picture compression standard," *Communications of the ACM*, vol.34, pp. 30-44, 1991.

[2] A. Skodras, C. Christopoulos and T. Ebrahimi, "The JPEG 2000 still image compression standard," *IEEE Signal Processing Magazine*, vol. 18, pp. 36-58, 2001.

[3] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "Photosketch: A sketch based image query and compositing system," *Proc. SIGGRAPH*, p. 60, 2009.

[4] H.J. Weinzaepfel and P. Pérez, "Reconstructing an image from its local descriptors", *Proc. IEEE Conf. CVPR*, pp. 337-344, 2011.

[5] M. Brown and D. Lowe, "Automatic panoramic image stitching using invariant features". *Intl. J. of Computer Vision*, pp. 59-73, 2007.

[6] H. Yue, X. Sun, and F. Wu, "Cloud based Image Coding for Mobile Devices –Toward Thousands to One Compression", *IEEE Trans. on Multimedia*, vol. 15, no. 4, pp. 845-857, 2012.

[7] H. Yue, X. Sun, and F. Wu, "Landmark Image Super-Resolution by Retrieving Web Images", *IEEE Trans. on Image Processing*, vol. 22, no. 12, pp. 4865-4878, 2013.

[8] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction", *IEEE Trans. Inf. Theory*, vol. 49, no. 3, pp. 626-643, 2003.

[9] D. Varodayan, A. Aaron and B. Girod "Rate-adaptive codes for distributed source coding", *Signal Process.*, vol. 86, pp. 3123-3130, 2006.

[10] A. Aaron, S. Rane, E. Setton, and B. Girod, "Transform-domain Wyner-Ziv codec for video", *Proc. SPIE Visual Communications and Image Processing Conf.*, vol. 5380, pp. 520-528, 2004.

[11] Correlated images. [Online]. Available: https://skydrive.live.com/?cid=75614bebe47f0556&id=75614BEBE47F0556%21113#cid=75614BEBE47F0556&id=75614BEBE47F0556%21113.

[12] G.J. Sullivan, J.-R. Ohm, W.-J. Han and T. Wiegand "Overview of the High Efficiency Video Coding (HEVC) standard", *IEEE Trans. Circuits Syst. VideoTechnol*., vol. 22, pp. 1648-1667, 2012.