STRUCTURE AND TEXTURE SYNTHESIS
Pragnesh Patel, Shailesh Gupta, Haider Zafar, Nilesh Deotale
Computer Department, Mumbai University
Lokmanya Tilak college of Engineering, Koparkhairne, Navi Mumbai.
patelpragnesh85@gmail.com, shaileshgpt47@gmail.com, haiderzafar72@gmail.com

Abstract
An approach for filling-in blocks of missing data in wireless image transmission is presented in this paper. When compression algorithms such as JPEG are used as part of the wireless transmission process, images are first tiled into blocks of 8x8 pixels. When such images are transmitted over fading channels, the effects of noise can destroy entire blocks of the image. Instead of using common retransmission query protocols, we aim to reconstruct the lost data using correlation between the lost block and its neighbours. If the lost block contained structure, it is reconstructed using an image inpainting algorithm, while texture synthesis is used for the textured blocks. The switch between the two schemes is done in a fully automatic fashion based on the surrounding available blocks. The performance of this method is tested for various images and combinations of lost blocks.

Keywords - Restoration, interpolation, inpainting, filling-in, texture synthesis, JPEG, wireless transmission, compression.

I. INTRODUCTION
General purpose images are most commonly compressed by lossy JPEG. JPEG divides the image into blocks of 8x8 pixels and calculates a two-dimensional (2-D) discrete cosine transform (DCT), followed by quantization and Huffman encoding; see [1]. In common wireless scenarios, the image is transmitted over the wireless channel block by block. Due to severe fading, we may lose an entire block, even several consecutive blocks of an image. In [2] the report that average packet loss rate in a wireless environment is 3.6% and occurs in a bursty fashion. In the worst case, a whole line of image blocks might be lost. Note that JPEG uses differential encoding for storing the average (dc) value of successive pixels. Hence, even if a single block is lost, the remaining blocks in that line (or reset interval) might be received without their correct average (dc) value. Two common techniques to make the transmission robust are forward error correction (FEC) and automatic retransmission query protocols (ARQ). Of these, FEC needs extra error correction packets to be transmitted. As noted in [3], ARQ lowers data transmission rates and can further increase the network congestion which initially induced the packet loss. Instead, we show that it is possible to satisfactorily reconstruct the lost blocks by using the available information surrounding them this will result in an increase in bandwidth efficiency of the transmission. The basic idea is to first automatically classify the block as textured or structured (containing edges), and then fill-in the missing block with information propagated from the surrounding pixels. In the case of structured
blocks, the inpainting algorithm in [4] is used; while for textured regions we follow [5]. We test the proposed scheme with a variety of images and simulated block losses. We also combine this approach with JPEG compression itself, where the encoder voluntarily skips blocks, and these are reconstructed at the decoder in the same fashion as in the wireless scenario. This process improves the compression ratio, at little or no quality degradation.

II. PROPOSED ALGORITHM

A. Image Transform Coding For JPEG Compression Algorithm.

The Joint Photographic Expert Group (JPEG), Formed as a joint ISO and CCITT working committee, is focused exclusively on still image compression. JPEG is compression standard for still color image and gray-scale image, otherwise known as continuous-tone images. The JPEG compression scheme is lossy and utilizes forward discrete cosine transform, a uniform quantizer and entropy encoding. The DCT function removes data redundancy by transforming data from a spatial domain to a frequency domain: the quantizer quantizes DCT coefficients with weighting functions to generate quantized DCT coefficients optimized for human eye; and the entropy encoder minimizes the entropy of the quantized DCT coefficients. By this methodology, the reduction of a large volume of data to a smaller version is achieved, discarding information that has little visual effect, and further compression of the data by taking advantage of its spatial characteristics.

B. The Discrete Cosine Transform (DCT)

DCT is a mathematical operation closely related to Fourier Transform. In the Spatial domain the image requires lots of data points. Once image is converted to frequency domain using Fourier transform family, only a few points are required to present the same image, because image contains only a few frequency components.

This technique can be applied to a color image. A color image is composed of pixels. These pixels have RGB color values, each with its x and y coordinates using 8X8 or 16X16 matrix for each primary color. When considered over an 8X8 matrix of 64 values, each with x and y coordinates, we have a three dimensional representation of pixel called a spatial representation or spatial domain. Compression algorithm, the input image is subdivided into 8-by-8 or 16-by-16 non-overlapping blocks, and the two-dimensional DCT is calculated for each block. The DCT coefficients are then quantized, coded, and transmitted. The JPEG receiver decodes the quantized DCT coefficients, calculates the inverse two-dimensional DCT of each block, and then puts the blocks back unruffled into a single image. For typical images, many of the DCT coefficients have values near to zero; these coefficients can be cast-off without seriously disturbing the quality of the reconstructed image. A two dimensional DCT of an M by N matrix A is defined as given in eqn(2):
\[
B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos \frac{\pi(2m + 1)p}{2M} \cos \frac{\pi(2n + 1)q}{2N},
\]

\[
0 \leq p \leq M - 1
\]
\[
0 \leq q \leq N - 1 \quad \text{... (2)}
\]

Where,

\[
\alpha_p = \begin{cases} 
\frac{1}{\sqrt{M}}, & p = 0 \\
\sqrt{\frac{2}{M}}, & 1 \leq p \leq M - 1 
\end{cases}
\]

\[
\alpha_q = \begin{cases} 
\frac{1}{\sqrt{N}}, & q = 0 \\
\sqrt{\frac{2}{N}}, & 1 \leq q \leq N - 1 
\end{cases}
\]

The DCT is invertible transformation and its inverse is given as:

\[
A_{mn} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} B_{pq} \cos \frac{\pi(2m + 1)p}{2M} \cos \frac{\pi(2n + 1)q}{2N},
\]

\[
0 \leq p \leq M - 1
\]
\[
0 \leq q \leq N - 1
\]

Where,

\[
\alpha_p = \begin{cases} 
\frac{1}{\sqrt{M}}, & p = 0 \\
\sqrt{\frac{2}{M}}, & 1 \leq p \leq M - 1 
\end{cases}
\]

\[
\alpha_q = \begin{cases} 
\frac{1}{\sqrt{N}}, & q = 0 \\
\sqrt{\frac{2}{N}}, & 1 \leq q \leq N - 1 
\end{cases}
\]

The DCT based encoder can be thought of as basically compression of a stream of 8 X 8 blocks of image sections. Each 8 X 8 block makes its way through each processing step, and produces Output in compressed form into the data stream. Because nearby image pixels are highly associated, the ‘forward’ DCT (FDCT) processing step lays the foundation for accomplishing data compression by concentrating most of the signal in the lower spatial frequencies. For atypical 8 X 8 sample block from a classic source image, most of the spatial frequencies have zero or near-zero amplitude and
need not be encoded. In principle, the DCT presents no loss to the source image samples; it purely transforms them to a realm in which they can be more competently encoded.

After output from the FDCT, each of the 64 DCT coefficients are uniformly quantized in aggregation with a carefully designed 64 – element Quantization Table. At the decoder, the quantized values are multiplied by the respective QT elements to recover the original values.

After quantization, all of the quantized coefficients are well-arranged into the “zig-zag” sequence. This arrangement helps to facilitate entropy encoding by assigning low-frequency non-zero coefficients before high-frequency coefficients. The DC coefficient, which contains a significant fraction of the complete image energy, is differentially encoded.

The reconstructions of lost blocks are in three steps:-

1. Classify lost blocks into two types i.e. texture and structure
2. Synthesize blocks which were classified as texture ;
3. To fill in blocks which were classified as structure

C. Block Classification:-

The first step in image reconstruction is to classify the errors i.e. whether they are texture or structure. This process is done after receiving the image on receiver side. After receiving the image receiver checks the pixels surrounding the lost block. For doing this we will use the method given by [11] . To determine whether the image contains texture or not we just define some threshold value to the local surrounding pixels. For assigning threshold values to the pixels we use method given as follows

\[
 n = s^2 \frac{(UB + LB)}{2} 
\]

Where UB and LB are upper and lower threshold value s. The threshold value vary between 0 and 1. We have used UB=0.16 and LB=0.04. as suggested in .

The above method is applied for each 8x8 block in the immediate neighborhood of the lost block. If the block contains even a single structure then we have to consider the structure first, also there is some limitation in algorithm stated in [11]

To overcome this limitation we will divide the image in 8 neighbor of 8x8 block and calculate the difference of them if 4 continuous difference is less than threshold value we will assign that pixel as structure.

D. Texture synthesis procedure

Most schemes reported in the literature deal with image transmission in error-prone environments using a combination of source and channel coding. we describe a packetization scheme in which the DCT coefficients array generated by JPEG is grouped such that bursty (consecutive) packet loss during transmission is scattered into a pseudo-random loss in the image domain (i.e., consecutive blocks are rarely lost in the image domain). The ensuing reconstruction scheme benefits because,
most frequency components can be recovered from adjacent blocks. However, large bursts may cause the errors to cluster in the image, and reconstruction suffers. It should be noted that the packetization scheme proposed in, when used with the reconstruction scheme described in our paper, is expected to further improve on the results reported here, and provide satisfactory reconstruction results even for very large bursts.

We also note that interleaving the image data before packetization avoids loss of contiguous areas in an image, facilitating reconstruction. This paper demonstrates reconstruction in the transform domain by expressing the lost data as a linear combination of blocks in the 4-neighborhood of the lost block. Four optimal weights (coefficients) need to be calculated per block based on combinations of available adjacent blocks. These weights, which result in a 10% space overhead, are used later in reconstruction. Strong diagonal edges are not well reconstructed by this method. Additional work on the reconstruction of missing data in block-based compression schemes is reported in, where the DCT coefficients of a missing block are interpolated from those with the same position in the neighboring blocks. The novelty of our proposed scheme is in the separation of the lost blocks into different classes, followed by the use of state-of-the-art image filling-in algorithms for textured and structured regions. This is done in a complete automatic fashion and without any side information.

A. Image Inpainting

Structure in an image can be an edge between two regions or the deterministic change in color or gray level. When the structure is classified in block classification then that pixel is replaced by digital image in painting method given by [4]

In this method let $\Omega$ be the region to be filled and $\partial \Omega$ be its boundary. The basic idea is to smoothly distribute the information around $\Omega$. Both the gray value and the isophote direction are distributed within the boundary. Let the image is denoted by $I$ then we can distribute the values by using the partial differential equation given as

$$\frac{\partial I}{\partial t} = \nabla(\Delta I).\nabla^\perp I$$

Where $\nabla$, $\Delta$ and $\nabla^\perp$ are gradient, laplacian and orthogonal gradient(isophote direction) respectively.
III. CONCLUSION AND FUTURE DIRECTIONS

We have proposed a new technique for the filling-in of missing blocks in wireless transmission of JPEG (or block based) compressed images. We have shown that as long as the features in the image are not completely lost, they can be satisfactorily reconstructed using a combination of computationally efficient image inpainting and texture synthesis algorithms. This eliminates the need for retransmission of lost blocks. When image resolution is increased, the quality of reconstruction improves & retransmission request is rarely required. We have tried to use image dependent information i.e. texture and structure to enhance the performance of JPEG. The compression ratio can be further increased by finding better masks by providing more image information. Missing blocks in the different channels need not be in the same image used in block classification & reconstruction. Adding this to current neighbouring information used is expected to improve even further the quality of results.
REFERENCES


