UDC 681.3.07

Serhii Zybin,
Candidate of Technical Sciences, State University of Telecommunications, Kyiv, Ukraine
Hennadii Kis,
Postgraduate, State University of Telecommunications, Kyiv, Ukraine

VIDEO ENCODING AND DECODING METHOD

Nowadays a multimedia content requires the improvement of communication technology. Compression is a process of data redundancy elimination or reducing.

Effective storing multimedia and transmission of the data is an actual task.

Inpainting can recover homogenous regions in a natural-looking manner, even when certain kinds of limited structure are present. However, conventional image inpainting is not effective at regenerating significant visual structure, especially if they are unique or have special, exact placement in the image.

Method for vision-based image compression (HVS), e.g., compression based on structure-aware in-painting discarding regions where features are not detected. Such systems and methods significantly boost image (and video) compression and coding efficiency.

The overall result of using the content information is a higher compression ratio than conventional techniques, especially for images that have some redundant visual structure, and a high perceptual fidelity to the original image.

Keywords: *multimedia coding, human visual system, compression rate, fractal image, self-similarity, computer vision.*

В настоящее время мультимедийный контент требует улучшения коммуникационных технологий. С целью снижения загрузки сети высоко ценятся методы сжатия видео.

Эффективное хранение мультимедиа и передача данных является актуальной задачей.

Рассмотрены кодирования второго поколения для передачи мультимедийных данных. Этот метод использует свойства человеческой зрительной сенсорной системы (HVS) для стратегии кодирования с целью достижения высоких показателей сжатия при сохранении приемлемого качества изображения.

Предложенный функциональный метод предусматривает существенное повышение степени сжатия, а также позволяет гибко адаптировать передачу данных с возможностями целевого устройства воспроизведения.

Ключевые слова: *мультимедиа кодирования, человеческая зрительная система, степень сжатия, фрактальное сжатие, самоподобие, компьютерное зрение.*

Introduction

Nowadays there are two major trends in mass-consuming multimedia.

The first tendency is a quality elevation of modern multimedia content. Resolution of photo matrices and frame rates are improved yearly by main HW vendors. So, recent camera standard of 4K (3840x2160) and 8K (7680x4320) UHDTV requires >100MB per a video frame. With frame rate growth (60 fps and more), it leads to increasing requirement of communication channels capacity.

© Зибін С., Кіс Г., 2018

Another trend is a widening of Internet access availability caused by mobile devices and global telecommunication market growth. Sharing any multimedia including TV and personal data is highly appreciated by customers. Rapid developing of cloud services can be observed; nobody stores data locally now.

Internet-based media devices have been added to the stack of home devices. Internet Protocol Television (IPTV) services offer cable TV-like services over home Internet connections. Every day customers are listening to and viewing the Internet, looking TV, video conferencing, sharing video in social networks and all these tasks require multimedia streaming.

Upcoming 3D TV and 360 TV standards require significantly larger volume of data to transmit. For example, if higher-resolution video or stereoscopic video were introduced, stereoscopy is achieved through 120 fps video synchronized with shuttered glasses, with 60 fps delivered to each eye the communication service should provide this capability without essential investment.

Additionally, modern electronic devices typically consume a great deal of power, which means they are expensive over time and wasteful of energy resource.

One of the traditional approaches of big data storage and transmission is a data compression. Compression is a process of data redundancy elimination or reducing started with pioneering research of Shannon information theory. State-of-the-art JPEG2000 and MPEG-4 AVC/H.264 are two such examples of coding efficiency. Video compression standard is capable to reduce memory requirements in tens times; a typical MPEG-4 lossy compression video has a compression factor between 20 and 200.

Until approximately 1980, the majority of image coding methods relied on techniques based on classical information theory (Huffman compression – Huffman 1952, LZV compression – Ziv and Lempel 1977; Welch 1984) to exploit the redundancy in the images in order to achieve compression. The techniques used were pixel-based and did not make any use of the information contained within the image itself. The compression ratios obtained with these techniques were moderate at around 2 to 1. Even with a lossy technique, such as discrete cosine transform (DCT) (Wallace 1991), a higher ratio (greater than 30 to 1) could be achieved only at the expense of image quality.

Attempts have recently been made to develop new image-compression techniques that outperform these first-generation image coding techniques considerably improving compression ratio. These methods attempt to identify features within the image and use the features to achieve compression. An awareness of how the HVS perceives various image features has been incorporated into coding methods for removing some of the visual redundancy inherent in images and for improving the visual quality of resulting images. These recent developments have been termed second generation image coding Kunt et al [1].

The recent success in multimedia data retrieval and image restoration (in-painting techniques) gives a cue for developing of novel way for multimedia compression. This approach is known as vision-based compression [2, 3].

Vision-based techniques are considered to augment a conventional signal-processing-based technique. For some regions of a source image, an exemplary system efficiently extracts and organizes structural information instead of compressing the regions. A structure-aware decoder then restores the regions via the feature information,

which occupies very little data space or minimal bandwidth in a bitstream that is transmitted from encoder to decoder. Key visual components of the image can still be conventionally compressed. Extracting feature information for some regions instead of compressing them can considerably increase overall image compression.

Background

Computer-Vision Background for Compression. We consider computer-vision approaches that can be used for reducing image data. At first it is feasible to separate texture and structure information at image. Taking into consideration the fact that textures consume the most of data volume and at the same time almost don't have human valuable information. So, for perception it is doesn't matter how exact look a grass or sand patch, a such patches can be completely generated by very small sample block from original image. Texture restoration looks to be a good point for data redundancy elimination.

Main problem for vision-based compression is a capture of scene structure.

Fractal image compression is one of the most promising techniques for image compression due to the advantages such as resolution independence and fast decompression. It exploits the fact that natural scenes present self-similarity to remove redundancy and obtain high compression rates with smaller quality degradation compared to traditional compression methods. The main drawback of fractal compression is its computationally intensive encoding process, due to the need for searching regions with high similarity in the image.

Several approaches have been developed to reduce the computational cost to locate similar regions. The use of robust features provides more discriminative and representative information for regions of the image. When the regions are better represented, the search for similar parts of the image can be reduced to focus only on the most likely matching candidates.

Texture synthesis and inpainting. Texture synthesis has a variety of applications in computer vision, graphics, and image processing. An important motivation for texture synthesis comes from texture mapping. Texture images usually come from scanned photographs, and the available photographs may be too small to cover the entire object surface. Learning of texture can be effectively used for image compression applying synthesis of unimportant for human perception parts of image instead of their encoding.

A.A. Efros and W.T. Freeman [4] have introduced image quilting, a method of synthesizing a new image by stitching together small patches of existing images (Fig. 1). Despite its simplicity, this method works remarkably well when applied to texture synthesis.

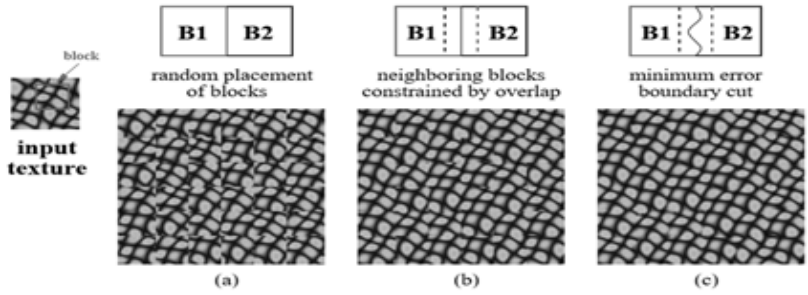

Fig. 1. Restoration of whole texture by initial block (a) simple stack (b) overlap blending (c) boundary cut on minimum error (Efros&Freeman algorithm)

Simple texture synthesis may give poor result for real scenes that have irregular structure. For restoration purpose texture transfer using correspondence map can produce result that is more realistic.

It is particularly well suited for texture transfer, because the image-quilting algorithm selects output patches based on local image information. We augment the synthesis algorithm by requiring that each patch satisfy a desired correspondence map, as well as satisfy the texture synthesis requirements. The correspondence map is a spatial map of some corresponding quantity over both the texture source image and a controlling target image. That quantity could include image intensity, blurred image intensity, local image orientation angles, or other derived quantities (see figure 2).

Constrained Texture Synthesis [5] is used for to better match the features across patch boundaries between the known texture around the hole and newly pasted texture patches, we fill the hole in spiral order. The authors developed real-time synthesis algorithm based on k-d trees and MRF (Markov random field).

Fig. 2. Texture transfer example from [4]. For Picasso the correspondence maps are the blurred luminance values

Recently great interest is to object removal and restoration of missed parts in photographs [6–8] called inpainting. The same approach can be applied for data redundancy elimination removing some part that can be realistically restored by surrounding.

The main problem is that boundaries between image regions are a complex product of mutual influences between different textures.

Image inpainting techniques fill holes in images by propagating linear structures (called isophotes in the inpainting literature) into the target region via diffusion. They are inspired by the partial differential equations of physical heat flow, and work convincingly as restoration algorithms. Their drawback is that the diffusion process introduces some blur, which becomes noticeable when filling larger regions.

Fig. 3. Restoration the background area of natural photo (a, b) and text by constrained texture synthesis from [6]. Restored text doesn't make sense but looks realistically

In [9] one is proposed to classify all blocks to texture and structure ones. Then, Synthesize blocks which were classified as texture; fill-in structure blocks with image completion. Blocks are classified based on their surroundings, see figure 4.

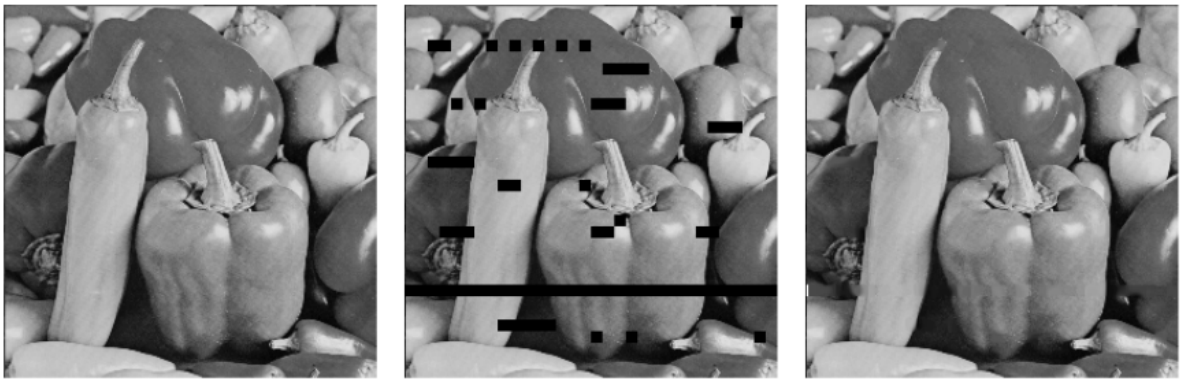

Fig. 4. Restoration result of missed block by [9]

Feature-based image retrieval and registration. Content-based image retrieval (CBIR) is the application of computer vision techniques to the problem of searching for digital images in large databases. "Content-based" means that the search analyzes the contents of the image rather than the meta-data such as keywords, tags, or descriptions associated with the image. The term "content" in this context might refer to colors, shapes, textures, or any other information that can be derived from the image itself.

One can see that approaches in image retrieval can be connected directly with second generation video coding because of visual content analysis are required by both applications.

To represent an image using BoW (bag of words) model [11, 12], an image can be treated as a document. Similarly, "words" in images need to be defined too. To achieve this, it usually includes following three steps: feature detection [14, 15], feature description, and codebook generation. A definition of the BoW model can be the "histogram representation based on independent features". Content based image indexing and retrieval (CBIR) appears to be the early adopter of this image representation technique.

After feature detection, each image is abstracted by several local patches. Feature representation methods deal with how to represent the patches as numerical vectors. These vectors are called feature descriptors. A good descriptor should have the ability to handle intensity, rotation, scale and affine variations to some extent. One of the most famous descriptors is Scale-invariant feature transform (SIFT) [13]. SIFT converts each patch to 128-dimensional vector. After this step, each image is a collection of vectors of the same dimension (128 for SIFT), where the order of different vectors is of no importance.

The final step for the BoW model is to convert vector-represented patches to "codewords" (analogous to words in text documents), which also produces a "codebook" (analogy to a word dictionary). A codeword can be considered as a representative of several similar patches. One simple method is performing k-means clustering over all the vectors. Codewords are then defined as the centers of the learned clusters. The number of the clusters is the codebook size (analogous to the size of the word dictionary).

Thus, each patch in an image is mapped to a certain codeword through the clustering process and the image can be represented by the histogram of the codewords. Codebooks can be considered as compressed representation of the image.

Fractal coding. Self-similarity. Representation an image as visual keyword document can be used for frame compression (see figure 5). Fractal compression exploits the fact that natural scenes present self-similarity [15], and then a significant amount of redundancy can be removed, providing high compression rates. Since it is difficult to detect self-similarity in a global scale, the image is usually partitioned into square blocks and affine transforms are used to describe the similarity between blocks.

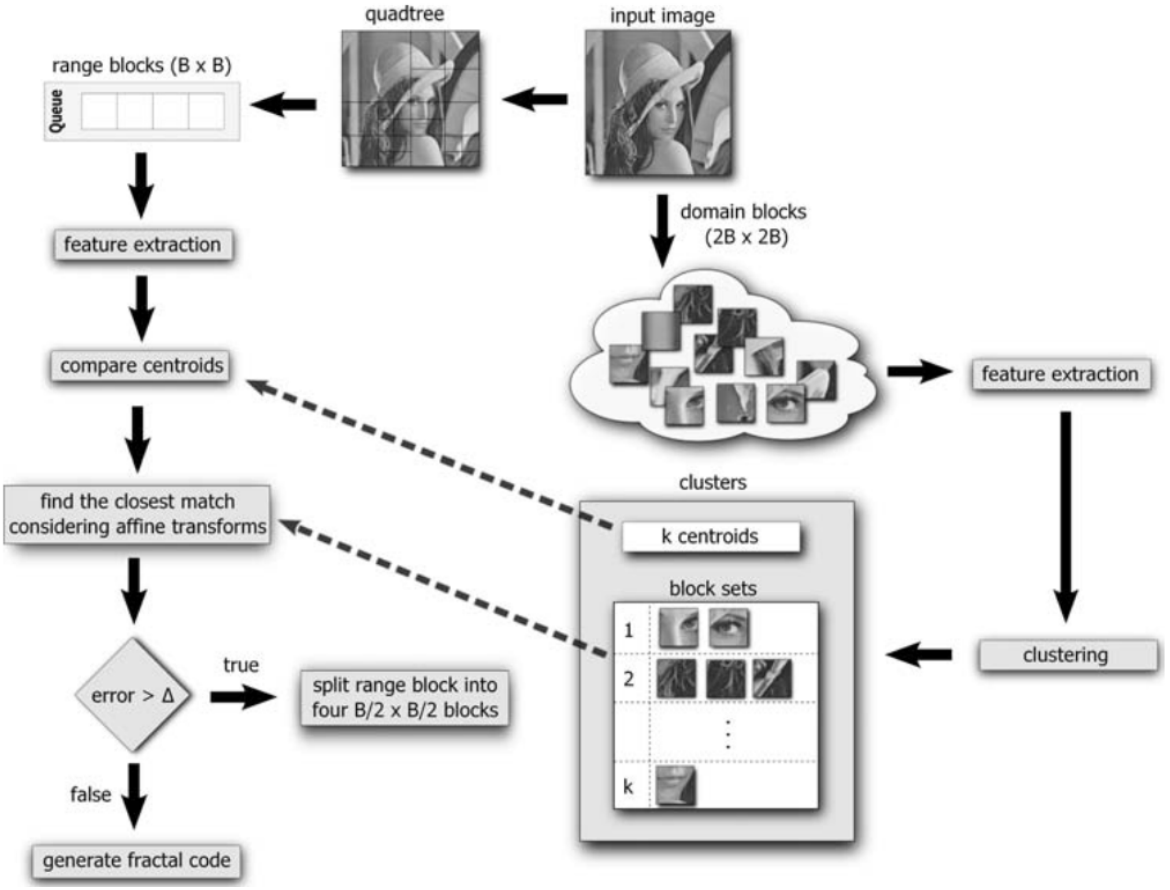

Fig. 5 Fractal image compression approach.

The resulting transforms, called fractal codes, are stored and used afterwards to reconstruct the original image by means of iterative function evaluations starting from an arbitrary image. By storing the fractal codes rather than bitmaps, images can be decompressed to resolutions that are higher or lower than the original resolution without distortions.

For a given level of the quadtree decomposition where range blocks have BЧB pixels are considered, domain blocks of 2Bx2B pixels are sampled from the input image. Using feature vectors as descriptor for each block, a clustering algorithm is applied to partition the domain blocks in k clusters. After extracting the feature vector for a range block, the closest cluster, according to the distance between the cluster centroid and range block feature vector, is estimated, then only domain blocks

belonging to the referred cluster are considered to find the best match. Once the best matching domain block is obtained, the reconstruction error is estimated. If it is smaller than the threshold, a fractal code is generated, otherwise, the range block is split into four sub-blocks of size B/2ЧB/2 pixels to be considered in the next level of the quadtree decomposition.

A method can be easily used for self-similarity detection and theirs transforms in consequence with [15] codebook.

Overview. Vision-based compression.

Mainly, the proposed method follows to idea introduced in [3]. Authors developed image compression techniques by identifying and utilizing visual features within images to achieve higher coding efficiency.

Inpainting can recover homogenous regions in a natural-looking manner, even when certain kinds of limited structure are present. However, conventional image inpainting is not effective at regenerating significant visual structure (e.g., structural edges), especially if they are unique or have special, exact placement in the image. These structural edges are conventionally relegated to conventional compression – so that they will reliably reappear in the regenerated image.

Nonetheless, structural data, especially edges, have a perceptual significance that is greater than the numerical value of their energy contribution to the entire image. Thus, the coding efficiency and the video quality of image coding techniques could be improved if the structural information might be properly exploited. What is needed is way to efficiently capture and organize structural information extracted from a source image so that an inpainter in a decoder can restore relatively large structural regions of the image with guidance that occupies very little data space/ minimal bitstream bandwidth to transmit from encoder to decoder.

As shown in fig. 6, in a typical implementation, an image is partitioned into blocks or “regions”. Regions that contain key visual components, are compressed by conventional signal-processing-based compression techniques. Remaining regions, which may still contain significant structure, are dropped from compression at the encoder being used. Instead of compression, the exemplary system extracts visual edge information from these remaining regions constructing (see correspondence map).

This method can be improved by exploiting feature-based approach borrowed from CBIR system.

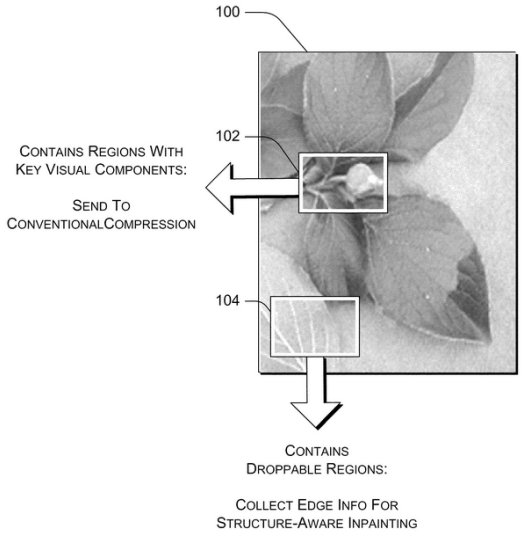

Fig. 6. Image segmentation to different blocks types: key regions and droppable regions [3].

The proposed method splits video frame stream as follows:

1. Low resolution band (optional).

2. Visual keywords (dynamic dictionary) comprising the most frequent visual patches. Visual patches contain descriptors and regions to compress in conventional manner.

3. Structure for key regions – transform coefficient for keywords to fit in key regions.

4. Structure for discardable regions in-painting.

5. Details, comprises high-frequency component bits (optional).

Every stream can be compressed with any conventional algorithm achieving highest compression rate.

Let I is original gray-scale image (gray scale is selected for simplicity of consideration);

Base layer B is low-resolution image that can be expressed as:

$$B = S(k) * \langle G(y) * I \rangle \quad (1)$$

here < * > means convolution;

G is 2D-Gaussian distribution;

S(k) is sub-sampling operation in k-times.

Then, дI = I – B details image

Encoding method

One of detect features with any suitable feature detection approach (SIFT, FAST, SURF or any other) at details layer: $F = \{f_i\}$ is set of feature points, $|F| = n$.

Let's calculate descriptors SIFT or SURF and add them into BoW, for example using k-D tree and k-means requires O(n log n). See above (Feature-based image retrieval and registration).

This BoW dictionary can take considerable size (f.e. SIFT descriptor occupies 128 floating point values) however taking into consideration high probability of sequential frames similarity in video stream encoding only data update is going to be transmitted.

Dynamic Huffman algorithm with dynamic dictionary can be good pattern for BoW update scheme. Keyword statistics should be updated every next video frame passed into encoder. Rarely used keywords should go away from keyword dictionary and replaced with new candidates. Only keyword patches should be transmitted. BoW is needed only for encoding.

Use to square patches to calculate intensity and contrast transform to minimize MSE between feature locality region and corresponding keyword like fractal compression does.

$$J = B + \Sigma L(i_j, t_j) G(s_j) \quad (2)$$

here *L(i)* is patch corresponding *i-th* visual keyword;

vector *t* is affine transformation that should be applied to patch to fit *j-th* feature;

(x, y, s, α; γ) comprises spatial parameters: shift, rotation, and scale and color space transform: gamma-correction parameters to fit intensity. Patches should be applied in pyramidal manner to keep needed scale details level.

To avoid edge effects patches are weighted with Gaussian of corresponding scale *G* gives confidence map for further inpainting to find areas of image completion. Confidence map is depending on actual matching score of the best keyword to the given region.

Areas where no features detected are considered as discardable ones:

$$\delta J = \text{argmin} |I - J + \delta J| \quad (3)$$

Finally, details layer δJ is calculated to keep pixel-wise structure that used for in-painting during restoration feature-less areas.

Only low-resolution edge information is extracted for them with Canny or any other edge detection approach to form correspondence map.

In one implementation, the system applies curve fitting to represent the edges. This edge information has a small data size compared with compressed versions of the same regions. The edge information, in turn, may be compressed to represent the dropped blocks.

Decoding method.

Firstly, base layer is decoded and patch dictionary is updated. Then key regions descriptors applied using (2). Finally, the dropped regions are inpainted based on the correspondence map information. The inpainter propagates structure in each unknown region by first finding nearby known structure elements (using confidence map) propagating texture form known regions to uncertain ones.

For example, in [3] from ends of known structural edges that abruptly terminate at the border of an unknown region, the inpainter applies the received edge information to propagate the edge as a mathematical construct or a 1-pixel-wide curve across the unknown region – so that the edge is made continuous with the neighboring regions and with a shape or path derived from the received edge information. The inpainter then applies a pixel classifier to label each pixel on or near the propagated edge as either a structure pixel, or as a pixel belonging to one of the objects on either side of the propagated edge (the edge inherently delineates two objects).

Then the inpainter fills in each pixel via an exemplary distance-based pair matching technique. For structure pixels, a given pixel is filled in with a value based on one of its neighboring pixels, depending on distance from the neighboring pixel and situational similarity between pixel pairs. For object pixels that are near the propagated structural edge within a distance threshold, a similar pair matching technique is used to fill in these object pixels except that the distance of each pixel in the pair from the structural edge is also taken into account in the calculations.

Once the propagated structure and nearby pixels have been filled in, a texture synthesizer completes each unknown region using texture appropriate for each object. Each object includes those pixels assigned to it by the pixel classifier.

Conclusions

The proposed scheme is feasible for further investigation. It is based on combination current MPEG-4 standard and mature computer-vision restoration techniques.

Described methods for vision-based image compression, e.g., compression based on structure-aware in-painting discarding regions where features are not detected. Such systems and methods significantly boost image (and video) compression and coding efficiency.

Key idea of fractal compression – self-similarity is exploited for visual dictionary creation improving overall compression ratio.

The overall result of using the content information is a higher compression ratio than conventional techniques, especially for images that have some redundant visual structure, and a high perceptual fidelity to the original image.

The proposed compression can produce much better compression rate than existed technology. Also there is an additional benefit for immediate usage for visual similarity and content search because of CBIR basis usage.

Additionally, due to data scale separation only patches of needed resolution can be rendered depending on target multimedia device capabilities reducing CPU resource.

There is possible reducing of needed bandwidth to transfer multimedia data to apply the proposed method in practice.

REFERENCES

1. *M. Kunt, A.* Ikonomopoulos and M. Koche, "Second Generation Image Coding Techniques", Proc. IEEE, vol. 73, 549–574. Apr. 1985.

2. *Feng Wu,* Advances in Visual Data Compression and Communication: Meeting the Requirements of New Applications. 2015.

3. *Xiaoyan Sun, Feng Wu, Shipeng Li,* Dong Liu patent US 8396312 B2. Vision-based compression. 2013.

4. *A.A. Efros and W.T.* Freeman, "Image Quilting for Texture Synthesis and Transfer" SIGGRAPH 01.

5. *Lin Liang, Ce Liu, Ying-Qing Xu, Baining Guo,* and Heung-Yeung Shum "Real-Time Texture Synthesis by Patch-Based Sampling", ACM Transactions on Graphics, Vol. 20, No 3, July 2001.

6. *A. Criminisi, P. Perez and K.* Toyama Region Filling and Object Removal by Exemplar-Based Image Inpainting IEEE Transactions on Image Processing, Vol. 13, No 9, SEP 2004.

7. Ouafek Naouel, Kholladi Mohamed-Khireddin "An Image Inpainting Algorithm based on K-means Algorithm". 2013. ACEEE.

8. *Feng Tang, Yiting Ying, Jin Wang,* and Qunsheng Peng "A Novel Texture Synthesis Based Algorithm for Object Removal in Photographs", Advances in Computer Science – ASIAN 2004. Higher-Level Decision Making. Vol. 3321. Lecture Notes in Computer Science. 248–258.

9. S. *Rane, G. Sapiro, and M. Bertalmio.* "Structure and Texture Filling-in of Missing Image Blocks in Wireless Transmission and Compression Applications".

10. *Mikolajczyk K., Leibe B., Schiele B.* "Local Features for the Object Class Recognition". In Computer Vision, ICCV 2005. Tenth IEEE International Conference. Vol. 2, 1792–1799. IEEE. 2005.

11. *Philbin J., Chum O., Isard M., Sivic J., Zisserman A.* (2007). Object Retrieval with Large Vocabularies and Fast Spatial Matching. In: Proc. CVPR.

12. *Li-Wei Kang, Chao-Yung Hsu, Hung-Wei Chen at al.* "Feature-Based Sparse Representation for Image Similarity Assessment", IEEE Transactions on Multimedia, Vol. 13. No 5. October 2011.

13. *D. Lowe* (2004). "Distinctive Image Features from Scale-Invariant Keypoints". International *Journal of Computer Vision.*

14. *E. Rosten; T. Drummond* (2006). "Machine Learning for High-Speed Corner Detection". European Conference on Computer Vision. Springer. 430–443. (The FAST corner detector).

15. *William Robson Schwartz, Helio Pedrini* (2011) "Improved Fractal Image Compression Based on Robust Feature Descriptors", International Journal of Image and Graphics. Vol. 11. No 4. 571–587.

УДК 681.3.07

С. Зибін,
кандидат технічних наук, Державний університет телекомунікацій,
м. Київ, Україна,
Г. Кіс,
аспірант, Державний університет телекомунікацій, м. Київ, Україна

МЕТОД КОДУВАННЯ ТА ДЕКОДУВАННЯ ВІДЕО

Сьогодні мультимедійний контент потребує вдосконалення комунікаційних технологій. Методи стиснення відео високо оцінюються для зниження навантаження на мережу.

Одним із традиційних підходів зберігання та передачі даних великого обсягу є стиснення даних. Компресія − це процес усунення або скорочення надмірності даних, розпочатий з піонерських досліджень теорії інформації Шеннона.

Ефективне зберігання мультимедіа і передача даних є актуальною задачею.

Розфарбовування може відновити однорідні регіони природним чином, навіть якщо присутні певні види обмеженої структури. Однак звичайне розфарбовування зображення не є ефективним при регенерації значних візуальних структур, особливо якщо вони є унікальними або мають спеціальне, точне розміщення на зображенні. Ці конструктивні грані традиційно віднесені до звичайного стиснення, тому вони напевно з'являться знову в регенерованому зображенні.

У типовій реалізації зображення розбивається на блоки або "регіони". Регіони, які містять ключові візуальні компоненти, стискаються за допомогою звичайних методів стиснення на основі обробки сигналів. Залишкові області, які все ще можуть містити значну структуру, усуваються від стиснення на кодері, який використовується. Замість стиснення, зразкова система витягує візуальну інформацію граней з цих залишкових регіонів, які будуються.

Спосіб стиснення зображень на основі зору, наприклад, стиснення, засноване на відкиданні структурного регіону, що розфарбовується, де функції не виявлені. Такі системи і способи значно підвищують стиснення зображень (і відео) і ефективність кодування. Загальним результатом використання інформації про вміст є більш високий коефіцієнт стиснення, ніж для звичайних методів, особливо для зображень, які мають певну надлишкову візуальну структуру, і високу перцепційну вірність оригінальному зображенню.

Таке стиснення може забезпечити набагато кращу швидкість стиснення, ніж існуюча технологія. Також існують додаткові переваги для негайного використання для візуальної схожості та пошуку контенту через використання бази CBIR.

Ключові слова: кодування мультимедіа, людська зорова система, ступінь стисненя, фрактальне стиснення, самоподібність, комп'ютерний зір.

Отримано 02.10.2018

Рецензент Рибальський О.В., д.т.н., проф.